

Content

1	Update Information	19
1.1	General information.....	19
1	Update Information No. 37 NC software 60642x-03	21
1.1	Overview.....	21
1.1.1	Released service packs	21
1.2	Important notes	22
1.3	New software options.....	23
1.4	New machine parameters.....	24
1.5	New functions	29
1.6	Software option 144—Motion Adaptive Control (MAC)	41
1.7	DHCP server.....	43
1.8	VNC settings.....	49
1.9	PLC functions.....	53
2	Introduction.....	71
2.1	Meaning of the symbols used in this manual	71
2.2	Proper and intended operation.....	71
2.3	Trained personnel	71
2.4	General information.....	72
2.4.1	HSCI interface	75
2.5	Overview of components	76
2.5.1	MC 6xxx main computer, HDR or SSDR hard disk and SIK 76	
2.5.2	CC 61xx controller unit.....	84
2.5.3	UEC 11x, UMC 11x controller unit with integrated inverter and PLC.....	86
2.5.4	PL 6xxx PLC input/output systems with HSCI interface	89
2.5.5	SPI expansion modules.....	93
2.5.6	PSL 13x low-voltage power supply unit	94
2.5.7	TE 6xx keyboard units	96
2.5.8	BF 2xx visual display units.....	101
2.5.9	MB 6xx machine operating panel.....	103
2.5.10	HSCI adapter for PLB 6001 OEM-specific machine operating panel.....	105
2.5.11	HR 4xx(FS), HR 5xx(FS) handwheels	106
2.5.12	Industrial PC with Windows 7	113
2.5.13	Key symbols	114
2.5.14	Touch probes.....	119
2.5.15	MS 110 / MS 111 installation kit	122
2.5.16	Other accessories	123
2.5.17	Documentation.....	126

2.6	Brief description	127
2.7	Software	139
2.7.1	Designation of the software.....	139
2.7.2	PLC software	140
2.7.3	Additional control loops or software options.....	140
2.7.4	Upgrade functions (feature content level).....	150
2.7.5	NC software exchange on the iTNC 530.....	152
2.7.6	Availability of software versions, software options and FCL	164
2.7.7	Installing a service pack	167
2.7.8	Special features of the software	169
2.7.9	HEROS 5 operating system	172
2.7.10	Functional safety (FS).....	174
2.7.11	Monitoring hardware changes	174
2.7.12	SELinux	175
2.7.13	Data backup	179
2.8	Released NC software	180
3	Mounting and electrical installation	181
3.1	General information.....	181
3.1.1	Safety precautions	181
3.1.2	Degrees of protection	182
3.1.3	Electromagnetic compatibility	182
3.1.4	ESD protection	183
3.2	Environmental conditions	185
3.2.1	Storage and operating temperatures	185
3.2.2	Heat generation and cooling	187
3.2.3	Limit values for ambient conditions	188
3.2.4	Installation elevation.....	188
3.2.5	Mounting attitude MC 6xxx, CC 61xx, UV xxx, UM xxx, UE 2xx B	189
3.2.6	Mounting attitude of UMC 111 (FS).....	190
3.2.7	Mounting attitude of UEC 11x (FS)	191
3.2.8	Mounting attitude of MC 6222, MC 7222	192
3.2.9	Mounting attitude of CC 6106, UV xxx, UM xxx, UE 2xx B(D)	193
3.2.10	Mounting attitude of PLB 6xxx	194
3.3	Handling the HDR hard disk.....	195
3.4	Mounting the Solid State Disk (SSD).....	196
3.5	Overview of components	197
3.6	HSCI.....	204
3.6.1	Introduction	204
3.6.2	Topology.....	207
3.6.3	HSCI interface	208

3.7	iTNC 530 HSCI connection overview	209
3.7.1	MC 6241	209
3.7.2	MC 6242	210
3.7.3	MC 6222	211
3.7.4	MC 7222	212
3.7.5	MC 6341	213
3.7.6	IPC 6341	214
3.7.7	CC 6106	215
3.7.8	CC 6108	216
3.7.9	CC 6110	217
3.7.10	UMC 111FS	218
3.7.11	UMC 111	219
3.7.12	UEC 11x	220
3.7.13	PLB 62xx	221
3.7.14	PLB 62xx FS	221
3.7.15	PLB 61xx	222
3.7.16	I/O modules PLD-H and PLA-H	223
3.8	Supply voltages in the HSCI system	226
3.8.1	Overview: Power supply for the iTNC	228
3.8.2	Current consumption of the HSCI components	229
3.8.3	X90: +24 V NC output of the UxC 11x (FS)	230
3.8.4	X101: NC power supply	231
3.8.5	Power supply of the CC 61xx	232
3.8.6	PSL 13x low-voltage power supply unit	237
3.9	MS 110 / MS 111 installation kit for double-row configuration	247
3.9.1	General information	247
3.9.2	Double-row configuration	248
3.9.3	Connection overview	250
3.10	UxC 11x (FS): Power supply and motor connection	254
3.10.1	UMC 111 (FS)	254
3.10.2	UEC 11x (FS)	257
3.11	UxC 11x (FS): Meaning of the LEDs	263
3.12	Power supply for PLC outputs	265
3.13	Power supply for PLB 6xxx (FS)	267
3.14	Meaning of the LED on PLB 6xxx	268
3.15	Power supply for control-is-ready signal	268
3.16	Digital PLC inputs/outputs	269
3.16.1	UxC 11x (FS): Digital PLC inputs/outputs	278
3.17	Analog PLC inputs/outputs	285
3.18	PROFIBUS connection	288
3.19	PROFINET connection	289
3.20	Configuring the PLC inputs/outputs with IOconfig	291
3.21	Buffer battery	294
3.22	Encoder connections	296
3.22.1	General information	296
3.22.2	Position encoder input	297
3.22.3	Speed encoder input	300
3.23	Adapters for encoder signals	304
3.24	Connecting the motor power modules (only CC 61xx)	307

3.25	Analog nominal value output	308
3.25.1	CMA-H 04-04-00—Pin layout	310
3.26	Touch probes.....	313
3.27	MB 620 machine operating panel.....	317
3.28	Data interfaces	325
3.28.1	USB interface (USB 2.0).....	328
3.28.2	HDL interface	329
3.29	MB 720 (FS) machine operating panel.....	330
3.30	HSCI adapter for PLB 6001 (FS) OEM-specific machine operating panel	341
3.31	TE 620 keyboard unit.....	355
3.32	TE 630 keyboard unit.....	356
3.33	TE 635Q (FS) keyboard unit	359
3.34	TE 645Q (FS) keyboard unit	360
3.35	TE 730, TE 720 keyboard unit.....	364
3.36	TE 735 (FS) keyboard unit	367
3.37	TE 740 keyboard unit.....	370
3.38	TE 745 (FS) keyboard unit	373
3.39	BF 250 15-inch TFT screen.....	377
3.40	BF 260 19-inch TFT display unit.....	379
3.41	Handwheel input.....	383
3.41.1	HR 4xx (FS) or HR 5xx (FS) portable handwheel.....	383
3.41.2	HR 520FS, HR 410FS safety-oriented electronic handwheels.....	387
3.41.3	HR 550 handwheel with wireless transmission.....	393
3.41.4	HR 410 portable handwheel.....	411
3.41.5	HR 130 panel-mounted handwheel.....	413
3.41.6	HRA 110 handwheel adapter	414
3.42	CML 110 capacitor module.....	416
3.43	Connecting cables: Specifications.....	418
3.44	Dimensions.....	419
3.44.1	MC 6241	420
3.44.2	MC 6341, MC 6x51, IPC 6341	421
3.44.3	MC 6242	422
3.44.4	MC 6222	423
3.44.5	MC 7222	424
3.44.6	MB 720 (FS)	425
3.44.7	TE 635Q (FS)	426
3.44.8	TE 645Q (FS)	427
3.44.9	TE 720	428
3.44.10	TE 730	429
3.44.11	TE 735 (FS).....	430
3.44.12	TE 630.....	431
3.44.13	TE 620.....	432
3.44.14	TE 740.....	433
3.44.15	TE 745.....	434
3.44.16	BF 250.....	435
3.44.17	BF 750.....	436
3.44.18	BF 260.....	437

3.44.19	BF 760.....	439
3.44.20	CC 6106.....	440
3.44.21	CC 6108 / CC 6110.....	441
3.44.22	UMC 111 (FS).....	442
3.44.23	UEC 11x (FS).....	443
3.44.24	PL 6xxx (FS).....	444
3.44.25	PLB 6001(FS).....	445
3.44.26	PSL 130.....	446
3.44.27	PSL 135.....	447
3.44.28	MS 11x.....	448
3.44.29	Adapter block for the data interface.....	449
3.44.30	USB hub.....	450
3.44.31	Line-drop compensator.....	451
3.44.32	Handwheels.....	451
3.44.33	Touch probes.....	457
3.44.34	CML 110.....	464
3.44.35	USB hub for operating panel.....	464
3.45	iTNC 530 HSCI grounding diagram with modular HEIDENHAIN inverter system— External PSL 13x.....	465
3.46	iTNC 530 HSCI grounding diagram with modular HEIDENHAIN inverter system—Power supply unit integrated in UV(R)	466
3.47	iTNC 530 HSCI basic circuit diagram.....	467
3.48	iTNC 530 HSCI cable overview—Basic configuration, MC in electrical cabinet	469
3.49	iTNC 530 HSCI cable overview—Basic configuration, MC in operating panel	470
3.50	iTNC 530 HSCI cable overview—Basic configuration, MC in electrical cabinet, UEC 1xx.....	471
3.51	iTNC 530 HSCI cable overview—Basic configuration, MC in operating panel, UEC 1xx.....	472
3.52	Cable overview for iTNC 530 HSCI with HEIDENHAIN inverter systems.....	473
3.53	iTNC 530 HSCI cable overview—Accessories.....	474

4	Machine parameters.....	475
4.1	What is a Machine Parameter?	475
4.2	The Machine Parameter Programming mode of operation	477
4.3	Input and output of machine parameters.....	478
4.3.1	Input format	478
4.3.2	Activating the machine parameter list.....	480
4.3.3	Changing the input values.....	480
4.4	List of machine parameters	494
4.4.1	Encoders and machines	494
4.4.2	Positioning.....	502
4.4.3	Operation with velocity feedforward control.....	510
4.4.4	Servo control with following error	511
4.4.5	Integrated speed and current control.....	512
4.4.6	Spindle	526
4.4.7	Integrated PLC	530
4.4.8	Configuration of the data interface.....	533
4.4.9	3-D touch probe	535
4.4.10	Tool measurement with TT	537
4.4.11	Tapping.....	542
4.4.12	Display and operation	543
4.4.13	Colors	553
4.4.14	Machining and program run	557
4.4.15	Hardware.....	566
4.4.16	Second spindle.....	582
5	Modules, markers and words.....	583
5.1	Overview of modules.....	583
5.2	Overview of markers and words	591
6	Configuring the axes and spindle.....	603
6.1	Control loops	603
6.1.1	Selecting the axes	603
6.1.2	Axis designation	604
6.1.3	Encoders	606
6.1.4	Assignment for axes	614
6.1.5	Assignment for spindles	616
6.1.6	Specifications of the controller unit:.....	618
6.1.7	Reading axis information	621
6.1.8	Traverse ranges.....	623
6.1.9	Lubrication pulse	629
6.2	PLC axes.....	630
6.2.1	Coupling function for PLC axes or auxiliary axes	635
6.3	PLC positioning	640

6.4	Axis-error compensation	647
6.4.1	Backlash compensation.....	647
6.4.2	Linear axis error compensation	651
6.4.3	Nonlinear axis error compensation.....	652
6.4.4	Compensation of thermal expansion.....	659
6.4.5	Compensation of reversal spikes during circular traverse	661
6.4.6	Compensation of stick-slip friction	661
6.4.7	Compensation of sliding friction (only for digital axes)	662
6.4.8	Torsion compensation	663
6.4.9	KinematicsComp	664
6.4.10	Software option 141—Cross Talk Compensation (CTC).....	688
6.4.11	Software option 142—Position Adaptive Control (PAC).....	693
6.4.12	Software option 143—Load Adaptive Control (LAC).....	694
6.4.13	Software option 144—Motion Adaptive Control (MAC).....	697
6.4.14	Software option 145—Active Chatter Control (ACC).....	699
6.5	Tilting and swivel axes	701
6.5.1	Determining the mechanical offset	701
6.5.2	Description of the kinematics configuration.....	714
6.5.3	Temperature compensation with tilting axes	741
6.5.4	Changing the milling heads	745
6.5.5	KinematicsOpt.....	746
6.5.6	“Tilt working plane” feature.....	761
6.5.7	Automatic compensation of offset for tilting axes	770
6.5.8	Virtual tool axis	776
6.5.9	Tilting functions with open-loop rotary axes.....	776
6.5.10	Cylindrical surface	777
6.6	Gantry/synchronized axes	778
6.6.1	Gantry axes	778
6.6.2	Master-slave torque control	782
6.6.3	Brake test for synchronized axes	792
6.7	Reference marks.....	796
6.7.1	Definition.....	796
6.7.2	Traversing the reference marks	797
6.8	The control loop	810
6.8.1	Relation between jerk, acceleration, velocity and distance	812
6.8.2	Interpolator	814
6.8.3	Position controller.....	832
6.8.4	Speed controller	848
6.8.5	Switching drives on and off.....	858
6.8.6	Current controller	866
6.8.7	Braking the drives for an emergency stop and a power failure	870
6.8.8	Power and torque limiting	873
6.8.9	Weakened field operation	880
6.9	Offset adjustment	890
6.9.1	Offset adjustment with integral factor	890
6.9.2	Offset adjustment by code number	890

6.10	Contouring behavior	891
6.10.1	Radial acceleration	891
6.10.2	Contour velocity at corners	892
6.11	Monitoring functions	894
6.11.1	Position monitoring	896
6.11.2	Nominal speed value monitoring.....	898
6.11.3	Movement monitoring	899
6.11.4	Standstill monitoring	900
6.11.5	Positioning window.....	900
6.11.6	Monitoring of the power supply unit.....	902
6.11.7	Temperature monitoring	904
6.11.8	Internal power supply and housing fan	907
6.11.9	I ² t monitoring	908
6.11.10	Momentary utilization of drive motors	918
6.11.11	Determining the current torque of a drive.....	920
6.11.12	Status of HEIDENHAIN power modules	922
6.11.13	Controlling the motor brakes.....	925
6.11.14	Emergency stop monitoring.....	930
6.11.15	EN 13849-1 on the iTNC 530 HSCI without functional safety (FS).....	943
6.12	Spindle	944
6.12.1	Position encoder of the spindle.....	945
6.12.2	Speed encoder of the spindle	950
6.12.3	Analog and digital closed-loop spindle control	952
6.12.4	Coded output of spindle speed	965
6.12.5	Volts-per-hertz control mode	967
6.12.6	Oriented spindle stop	968
6.12.7	Tapping with floating tap holder and nominal speed output.....	974
6.12.8	Tapping with floating tap holder and coded spindle-speed output.....	976
6.12.9	Rigid tapping	977
6.12.10	Switching the modes of operation	980
6.12.11	Operating a second spindle.....	985
6.12.12	C-axis operation.....	987
6.12.13	Software option 96—Advanced spindle interpolation	991
6.13	Integrated oscilloscope	992
6.14	Commissioning	1004
6.14.1	Tables for power modules, supply modules and motors	1004
6.14.2	Field orientation.....	1019
6.14.3	Preparation	1020
6.14.4	Commissioning digital control loops with TNCopt	1023
6.14.5	Commissioning of digital axes	1026
6.14.6	Commissioning the digital spindle	1055
6.14.7	Commissioning an analog axis	1060
6.14.8	Commissioning the analog spindle	1064
6.15	Block diagram for iTNC 530 with CC 61xx	1065

7	CC 61xx / UEC 1xx controller units ...	1067
7.1	Specifications	1067
7.2	Connecting the encoders	1069
7.2.1	General information	1069
7.2.2	Position encoders	1069
7.2.3	Speed encoders	1070
7.3	Relationship between speed input, position input and PWM output	1071
7.4	Single-speed, double-speed, PWM frequency.....	1072
7.4.1	General information	1072
7.4.2	Single-speed and double-speed (software option 49)	1074
7.4.3	Combination of several CC 61xx controller units	1078
7.4.4	PWM frequency	1082
7.5	PLC cycle time	1087
7.6	Monitoring functions	1087
7.7	Special functions	1088
7.7.1	Multifunction filter	1088
7.7.2	Filter order for separate low-pass filter in the speed controller	1093
7.7.3	Dynamic determination of load	1093
7.7.4	LIFTOFF function.....	1098
7.7.5	TRC – Torque Ripple Compensation	1101
7.7.6	Peculiarities in weakened-field operation	1103
7.7.7	Crossover Position Filter (CPF).....	1104
7.8	Stick-slip friction compensation at quadrant transitions	1106
7.9	Field orientation	1108
7.9.1	Possibilities for determining the field angle	1110
7.9.2	Determination of the field angle without motor motion (MP2254.x = 0)	1113
7.9.3	Determination of the field angle with motor motion (MP2254.x = 2/3).....	1115
7.9.4	Reading or setting the field angle via the PLC	1118
7.9.5	Saving the determined field angle.....	1120
7.9.6	Definition of the field angle	1121
7.10	Adjustment of linear and torque motors	1122
7.10.1	General information	1122
7.10.2	Safety precautions for linear and torque motors	1126
7.11	Commissioning linear and torque motors.....	1128
7.11.1	Machine parameters for linear motors	1128
7.11.2	Machine parameters for torque motors	1129
7.11.3	Adjustment of the current controller	1130
7.11.4	Adjustment of the speed controller.....	1133
7.12	Determining entries for motor tables	1140
7.12.1	Determining data for linear motors	1140
7.12.2	Determining data for torque motors.....	1143

8	Machine interfacing	1147
8.1	Display and operation.....	1147
8.1.1	Position and status display.....	1147
8.1.2	Operating modes	1165
8.1.3	Operating times	1169
8.1.4	Error messages	1176
8.1.5	Service files	1185
8.1.6	Help	1189
8.1.7	PLC pop-up window.....	1192
8.1.8	TNCguide – context-sensitive help system for the iTNC 530 (user documentation)	1195
8.1.9	Machine datum	1205
8.1.10	NC program.....	1210
8.1.11	Adaptive feed control (AFC)	1218
8.1.12	Global program settings (PGM).....	1236
8.1.13	Cycles.....	1241
8.1.14	End of program run	1244
8.1.15	Returning to the contour	1244
8.1.16	M functions.....	1251
8.1.17	Powering up and shutting down the control.....	1260
8.1.18	Arc end-point tolerance	1263
8.1.19	Limit-switch tolerance for M140 / M150.....	1263
8.1.20	Radius compensation	1264
8.1.21	User parameters	1264
8.1.22	Code numbers.....	1265
8.1.23	Programming station mode.....	1267
8.1.24	Color settings	1267
8.1.25	Graphic display	1275
8.1.26	Special characters	1277
8.1.27	iTNC character set.....	1278
8.1.28	Conversational language	1283
8.1.29	Logs	1286
8.1.30	Diagnostic functions.....	1298
8.1.31	BUS diagnosis	1320
8.1.32	Expanded PLC module interface for 18 axes	1329
8.1.33	Window manager.....	1332
8.1.34	Software option 133—Remote Desktop Manager.....	1338
8.2	PLC window	1354
8.2.1	Small PLC window	1354
8.2.2	Large PLC window	1358
8.3	PLC soft keys	1372
8.3.1	Soft-key project file for screen	1372
8.3.2	Soft-key project file for HR 420, HR 5xx	1385
8.3.3	Soft-key rows	1389
8.4	Keystroke simulation.....	1392
8.4.1	iTNC control panel.....	1392
8.4.2	Machine operating panel.....	1404
8.4.3	Touchpad on USB port.....	1405

8.5	Files	1406
8.5.1	Datum tables (*.D)	1409
8.5.2	Freely definable tables	1409
8.5.3	PLC files	1419
8.6	DCM – Dynamic Collision Monitoring	1424
8.6.1	DCM – monitoring the working space for collisions.....	1424
8.6.2	Fixture monitoring with DCM.....	1453
8.6.3	Tool carrier kinematics and DCM	1477
8.6.4	KinematicsDesign.....	1488
8.7	Pallet management.....	1490
8.8	Electronic handwheel	1502
8.8.1	HR 130 panel-mounted handwheel.....	1510
8.8.2	HR 410 portable handwheel.....	1510
8.8.3	HR 420 portable handwheel.....	1513
8.8.4	HR 520 portable handwheel.....	1516
8.8.5	HR 550FS wireless handwheel	1518
8.8.6	HR 150 panel-mounted handwheels with HRA 110 handwheel adapter	1524
8.9	PLC inputs/outputs.....	1526
8.9.1	24 V– switching input/outputs	1526
8.9.2	Analog inputs.....	1529
8.9.3	Analog outputs	1529
8.10	Incremental jog positioning	1530
8.11	Hirth coupling.....	1532
8.12	Datum shift.....	1534
8.13	Touch probe.....	1535
8.13.1	Using the touch probes	1536
8.13.2	Touch probe cycles	1539
8.13.3	Measurement log in manual touch probe cycles	1546
8.13.4	Measurement log in the touch probe cycles for probing from the NC program	1549
8.13.5	Tool measurement	1552
8.14	Special functions for laser cutting machines	1561
8.14.1	Analog voltage output	1561
8.14.2	Graphic simulation (without TOOL CALL)	1563
8.14.3	Program stop for M functions and TOOL CALL S.....	1564
8.15	Tool changer	1565
8.15.1	Tool and pocket table	1565
8.15.2	Tool-usage test.....	1597
8.15.3	Automatic calculation of cutting data	1601
8.15.4	Automatic tool recognition	1603
8.15.5	Controlling the tool changer	1603
8.15.6	PLC programming example.....	1629
8.15.7	Extended tool management.....	1641

9	PLC Programming.....	1661
9.1	PLC Functions.....	1661
9.1.1	Selecting the PLC mode	1667
9.1.2	PLC main menu.....	1667
9.1.3	File management	1670
9.1.4	The WATCH LIST function.....	1671
9.1.5	The I/O-FORCE LIST	1674
9.1.6	The TABLE function	1676
9.1.7	The TRACE function.....	1678
9.1.8	The logic diagram.....	1680
9.1.9	The COMPILE function	1682
9.1.10	ADVANCED SETUP.....	1683
9.2	Conditional Compilation	1684
9.3	Hard-Disk Organization	1689
9.3.1	Encrypted PLC partition (PLCE:).....	1690
9.4	System Files	1701
9.4.1	OEM.SYS	1701
9.4.2	NCMACRO.SYS	1709
9.4.3	MGROUPS.SYS	1712
9.4.4	MSPLIT.SYS.....	1712
9.4.5	PLCSOFTK.SYS.....	1712
9.4.6	CYCLE.SYS	1712
9.4.7	TNC.SYS.....	1712
9.5	Data Transfer NC > PLC, PLC > NC	1713
9.5.1	Data transfer NC program > PLC (FN19: PLC =)	1713
9.5.2	Data transfer PLC > NC program (Q-parameters)	1714
9.5.3	Data transfer NC program > NC (FN17: SYSWRITE)	1716
9.5.4	Data transfer NC > NC program (FN18: SYSREAD).....	1725
9.5.5	Data transfer of machine parameters > PLC.....	1737
9.5.6	Interrogate PLC operands in the NC program (FN20: WAIT FOR)	1740
9.5.7	Data transfer PLC operand <=> NC program	1741
9.6	Operands.....	1744
9.6.1	Overview of operands.....	1744
9.6.2	Operand addressing (byte, word and double word)	1747
9.6.3	Timers	1748
9.6.4	Counters.....	1751
9.6.5	Fast PLC inputs	1753
9.7	Program Creation.....	1754
9.7.1	ASCII editor	1755
9.7.2	Program format	1755
9.7.3	Program structure	1756
9.7.4	Logical names for files	1757

9.8	Command Set.....	1758
9.8.1	Overview.....	1758
9.8.2	LOAD (L).....	1761
9.8.3	LOAD NOT (LN).....	1763
9.8.4	LOAD TWO'S COMPLEMENT (L-).....	1765
9.8.5	LOAD BYTE (LB).....	1766
9.8.6	LOAD WORD (LW).....	1766
9.8.7	LOAD DOUBLE WORD (LD).....	1767
9.8.8	ASSIGN (=).....	1767
9.8.9	ASSIGN BYTE (B=).....	1768
9.8.10	ASSIGN WORD (W=).....	1769
9.8.11	ASSIGN DOUBLE WORD (D=).....	1769
9.8.12	ASSIGN NOT (=N).....	1770
9.8.13	ASSIGN TWO'S COMPLEMENT (=-).....	1770
9.8.14	SET (S).....	1771
9.8.15	RESET (R).....	1772
9.8.16	SET NOT (SN).....	1773
9.8.17	RESET NOT (RN).....	1774
9.8.18	AND (A).....	1775
9.8.19	AND NOT (AN).....	1777
9.8.20	OR (O).....	1779
9.8.21	OR NOT (ON).....	1781
9.8.22	EXCLUSIVE OR (XO).....	1783
9.8.23	EXCLUSIVE OR NOT (XON).....	1785
9.8.24	ADDITION (+).....	1787
9.8.25	SUBTRACTION (-).....	1788
9.8.26	MULTIPLICATION (X).....	1789
9.8.27	DIVISION (/).....	1790
9.8.28	REMAINDER (MOD).....	1791
9.8.29	INCREMENT (INC).....	1792
9.8.30	DECREMENT (DEC).....	1792
9.8.31	EQUAL TO (==).....	1793
9.8.32	LESS THAN (<).....	1794
9.8.33	GREATER THAN (>).....	1795
9.8.34	LESS THAN OR EQUAL TO (<=).....	1796
9.8.35	GREATER THAN OR EQUAL TO (>=).....	1797
9.8.36	NOT EQUAL (<>).....	1798
9.8.37	AND [] (A[]).....	1799
9.8.38	AND NOT [] (AN[]).....	1801
9.8.39	OR [] (O[]).....	1801
9.8.40	OR NOT [] (ON[]).....	1801
9.8.41	EXCLUSIVE OR [] (XO[]).....	1801
9.8.42	EXCLUSIVE OR NOT [] (XON[]).....	1801
9.8.43	ADDITION [] (+[]).....	1802
9.8.44	SUBTRACT [] (-[]).....	1803
9.8.45	MULTIPLY [] (x[]).....	1803
9.8.46	DIVIDE [] (/ []).....	1803
9.8.47	REMAINDER [] (MOD[]).....	1803
9.8.48	EQUAL TO [] (=[]).....	1804
9.8.49	LESS THAN [] (<[]).....	1805
9.8.50	GREATER THAN [] (>[]).....	1805
9.8.51	LESS THAN OR EQUAL TO [] (<=[]).....	1805
9.8.52	GREATER THAN OR EQUAL TO [] (>=[]).....	1805
9.8.53	NOT EQUAL [] (<>[]).....	1805

9.8.54	SHIFT LEFT (<<)	1806
9.8.55	SHIFT RIGHT (>>)	1807
9.8.56	BIT SET (BS)	1808
9.8.57	BIT CLEAR (BC)	1809
9.8.58	BIT TEST (BT)	1810
9.8.59	PUSH DATA ONTO THE DATA STACK (PS)	1811
9.8.60	PULL DATA FROM THE DATA STACK (PL)	1812
9.8.61	PUSH LOGIC ACCUMULATOR ONTO THE DATA STACK (PSL)	1812
9.8.62	PUSH WORD ACCUMULATOR ONTO THE DATA STACK (PSW)	1813
9.8.63	PULL LOGIC ACCUMULATOR FROM THE DATA STACK (PLL)	1813
9.8.64	PULL WORD ACCUMULATOR FROM THE DATA STACK (PLW)	1814
9.8.65	UNCONDITIONAL JUMP (JP)	1814
9.8.66	JUMP IF LOGIC ACCUMULATOR = 1 (JPT)	1815
9.8.67	JUMP IF LOGIC ACCUMULATOR = 0 (JPF)	1815
9.8.68	CALL MODULE (CM)	1816
9.8.69	CALL MODULE IF LOGIC ACCUMULATOR = 1 (CMT)	1816
9.8.70	CALL MODULE IF LOGIC ACCUMULATOR = 0 (CMF)	1817
9.8.71	END OF MODULE, END OF PROGRAM (EM)	1818
9.8.72	END OF MODULE IF LOGIC ACCUMULATOR = 1 (EMT)	1818
9.8.73	END OF MODULE IF LOGIC ACCUMULATOR = 0 (EMF)	1818
9.8.74	LABEL (LBL)	1818
9.9	INDEX register (X register)	1819
9.10	Commands for String Processing	1821
9.10.1	LOAD String (L)	1823
9.10.2	ADD String (+)	1823
9.10.3	STORE String (=)	1823
9.10.4	OVERWRITE String (OVWR)	1824
9.10.5	EQUAL TO Command for String Processing (==)	1825
9.10.6	LESS THAN Command for String Processing (<)	1825
9.10.7	GREATER THAN Command for String Processing (>)	1825
9.10.8	LESS THAN OR EQUAL TO Command for String Processing (<=)	1826
9.10.9	GREATER THAN OR EQUAL TO Command for String Processing (>=)	1826
9.10.10	NOT EQUAL Command for String Processing (<>)	1827
9.10.11	Modules for string processing	1828
9.11	Submit Programs	1831
9.11.1	Calling the submit program (SUBM)	1832
9.11.2	Interrogating the status of a submit program (RPLY)	1832
9.11.3	Canceling a submit program (CAN)	1833
9.12	Cooperative multitasking	1835
9.12.1	Starting a parallel process (SPAWN)	1835
9.12.2	Control of events	1836
9.13	Constants field (KF)	1841

9.14	Program Structures	1842
9.14.1	IF ... ELSE ... ENDI structure	1843
9.14.2	REPEAT ... UNTIL structure	1843
9.14.3	WHILE ... ENDW structure.....	1844
9.14.4	CASE branch	1844
9.15	Linking Files.....	1846
9.15.1	USES Statement (USES)	1847
9.15.2	GLOBAL statement (GLOBAL).....	1848
9.15.3	EXTERN Statement (EXTERN)	1848
9.16	PLC Modules.....	1849
9.16.1	Markers, bytes, words, and double words.....	1849
9.16.2	Number conversion	1852
9.17	Python.....	1856
10	Data Interfaces	1857
10.1	Introduction	1857
10.1.1	Principles of data transfer.....	1858
10.1.2	Data transfer check: Handshaking.....	1862
10.2	The Ethernet Interface of the iTNC.....	1863
10.3	Connecting the iTNC to the Network	1864
10.3.1	Settings at the iTNC	1864
10.4	Protection Against Data Tampering.....	1893
10.5	The USB Interface of the iTNC.....	1894
10.6	iTNC Serial Data Interfaces	1899
10.6.1	General information.....	1899
10.6.2	RS-232-C/V.24 interface	1899
10.7	Configuration of Interfaces	1902
10.7.1	Control characters	1902
10.7.2	Selection of interfaces and operating modes.....	1903
10.7.3	Configuration of interfaces.....	1903
10.8	Data Transmission Protocols	1907
10.8.1	Selection of transmission protocols	1907
10.8.2	Standard communications protocol.....	1908
10.8.3	Communications protocol with block check character....	1910
10.8.4	LSV2 transmission protocol.....	1913
10.9	Saving and Loading Files	1914
10.10	Data Transfer by PLC	1916
10.10.1	Settings	1916
10.10.2	PLC modules	1917
10.11	External Programming	1929

11	iTNC 530 and Windows	1931
11.1	General information and important notes	1931
11.2	IPC 6341—Industrial PC with Windows 7	1931
12	Error messages	1943
12.1	DSP/NC error messages.....	1943
12.2	iTNC Error Messages during Data Transfer.....	1944
12.3	Error messages of the file system	1945
13	Subject Index	1947

1 Update Information

1.1 General information

Update Information for the iTNC 530 HSCI appears at irregular intervals, often as part of a new software version. This is preliminary information in PDF format, containing brief descriptions of new software functions as well as new hardware components. After the Update Information has been published, the new items are included in the iTNC 530 HSCI Technical Manual.

The Technical Manual and each Update Information are saved in the HEIDENHAIN HESIS-Web including Filebase on the Internet, where registered users can access them at <http://portal.heidenhain.de>.

Registered users of the HEIDENHAIN HESIS-Web including Filebase on the Internet receive an e-mail notification when a new Update Information appears.

This version of the Technical Manual includes all Update Information documents of the iTNC 530 up to and including number 37, meaning that the contents of this Technical Manual for the iTNC 530 HSCI correspond to the scope of functions of software version 60642x-03.



1 Update Information No. 37 NC software 60642x-03

1.1 Overview

1.1.1 Released service packs



Note

Please keep in mind that this document is a prerelease! This preliminary version is not yet complete and subject to changes.

The following service packs were released for **60642x-01**:

- Service pack 04: March 2011
- Service pack 05: August 2011
- Service pack 06: August 2011
- Service pack 07: January 2012
- Service pack 08: June 2012
- Service pack 09: January 2013

The following service packs were released for **60642x-02**:

- Service pack 01: March 2012
- Service pack 02: March 2012
- Service pack 03: May 2012
- Service pack 04: July 2012
- Service pack 05: October 2012
- Service pack 06: November 2012
- Service pack 07: November 2012
- Service pack 08: January 2013
- Service pack 09: April 2013

The following software versions were released for 60642x:

- 60642x-03: May 2013

1.2 Important notes

With the release of the new software version, new variants of the memory media will be introduced. This makes it possible to order a memory medium with a certain NC software version by indicating the ID number and variant:

Memory media with NC software 60642x:

HDR hard disk for iTNC 530 with HSCI	ID
NC software 60642x-02	
No export license required, NC software 606641-02	682272-52
Export license required, NC software 606420-02	682272-02
NC software 60642x-03	
No export license required, NC software 606421-03	682272-53
Export license required, NC software 606420-03	682272-03

SSDR solid state disk for iTNC 530 with HSCI	ID
NC software 60642x-03	
No export license required, NC software 606421-02	736591-52
Export license required, NC software 606420-02	736591-02
NC software 60642x-03	
No export license required, NC software 606421-03	736591-53
Export license required, NC software 606420-03	736591-03

1.3 New software options

You can enable the following new software options by entering a code number. HEIDENHAIN can give you the code number after having been informed of the SIK number:

Option	Description	ID
#144	Motion Adaptive Control (MAC) The MAC option provides a way to change machine parameters depending on motion-dependent input quantities (e.g speed or following error). This makes it possible, for example, to realize a velocity-dependent adaptation of the kV factor on motors whose stability changes through the various traversing velocities.	800546-01
#145	Active Chatter Control (ACC) With the ACC option you can significantly reduce a machine's inclination to chattering and increase the attainable metal removal rate during heavy cutting by 25 % and more, depending on the type of machine.	800547-01

1.4 New machine parameters

■ **New: MP118, MP119 bit 4, bit 5 – Select encoder for safety functions of functional safety (FS)**

These machine parameters have been introduced as a preparation for future expansions.

■ **Removed: MP734 – Interpolating non-linear axis error compensation**

The nominal axis value is now taken into account by default in direction-dependent axis-error compensation. MP709 (time constant for backlash compensation) is used as a filter for the compensation value. MP734 has therefore been removed.

■ **New: MP1061.0, MP1061.1**

MP1061.0 – Path acceleration for machining feed rate

MP1061.1 – Path acceleration for rapid traverse

MP1061 (Limitation of path acceleration) was divided into the two indices MP1061.0 (Path acceleration for machining feed rate) and MP 1061.1 (Path acceleration for rapid traverse). This division enables you to enter a lower value for the acceleration at machining feed rate, which makes it possible to optimize the behavior of drives with regard to backlash compensation.

Input: 0.001 to 500 [m/s²] or [1000°/s²]

■ **Enhanced: MP1350.x – Functional sequence for reference mark traverse**

The new input value 8 makes it possible to reference incremental encoders with EnDat 2.2 protocol. With these encoders the absolute value is not available after switch-on, so that reference marks must be traversed.

If this type of encoder is connected to the position input, bit 3 of MP118.x must be set. This type of encoder cannot be used at the speed encoder input yet.

Input: 8: Incremental encoder with EnDat 2.2 interface, reference run required

■ **New: MP1358.x – Multiturn information**

MP1358 is used to activate axis-specific evaluation of the multiturn information on the respective encoders.

The reference position of the axis can be read out as usual with PLC Module 9041. If MP1358.x = 1, the return value is the position of the encoder plus the multiturn information (e.g. 730 degrees) plus the value in MP960.x. As this return value of the module now also includes the multiturn information of the encoder, this needs to be considered when defining MP960.x and MP1356.x.

Input: %0000000000000000000000

0 = Do not evaluate multiturn information

1 = Evaluate multiturn information

■ **New: MP2206.x – Type of speed encoder**

Input: 12: EnDat2.2 incremental, position encoder (e.g. LIP 200)

19: EnDat 2.2 incremental, speed encoder
(e.g. ERM + EIB)

■ **Enhanced: MP2223 – Special controller functions**

Bit 2 of <Mode> of PLC Module 9173 (speed-dependent wye/delta switchover) is used to set whether readiness of the current controller is to be waited for during restart. Now bit 7 of MP2223 can be set as an alternative.

Input: Bit 7

0 = Wait for readiness of current controller

1 = Do not wait for readiness of current controller

■ **New: MP2547.x – Damping/phase increase for filter 6**

Input: 0 to 40.0 [dB]

■ **New: MP2557.x – Center/cutoff frequency for filter 6**

Input: 0 to 30000.0 [Hz]

■ **New: MP2567.x – Filter type for filter 6**

Input: 0 = No filter
1 = PT2 low-pass filter (speed controller)
2 = Band-rejection (speed controller)
3 = Phase increase (speed controller)
11 = PT2 low-pass filter (position controller)
12 = Band-rejection (position controller)
13 = Phase increase (position controller)

■ **New: MP2577.x – Bandwidth for filter 6**

Input: 0 to 30000.0 [Hz]

■ **New: MP2609.x – Only CC 6xxx: Filter frequency for CPF**

Filter frequency for Crossover Position Filter, see page 29.

Input: 0 to 300 Hz
0 = Not active

■ **New: MP2690 – Bandwidth limiting of the compensation values**

The compensation values for backlash, static friction and reversal error can be calculated dynamically depending on the acceleration, or by using fixed time constants. The new parameter MP2690 can be used to prevent the compensation values from changing too rapidly at very high acceleration. A minimum time constant is entered to limit the bandwidth of the change of the compensation values.

The reversal error is compensated dynamically depending on the acceleration. The reversal error is compensated more rapidly at high acceleration, and more slowly at low acceleration. The new MP708 is used to define the distance within which the complete reversal error is to be compensated. This dynamic compensation is active only if no time constant is defined in MP752. The bandwidth of the compensation value can be limited in MP2690.

Input: 0.0001 to 1.0000 [seconds]
0 = Not active

■ **Enhanced: MP2700.x – Only CC 6xxx: Activation of the expanded compensation**

Bit 4 is added to MP2700.x. This enables you to activate and deactivate the ACC compensation for specific axes.

Bit 4: ACC compensation
Input: %0000000000000000
0 = Not active
1 = Active

■ **New: MP 2710.x – Only CC 6xxx:**

ACC: Selection of principle of function of automatic mode

Only valid if bit 4 is set in MP2700

Input: 0 = Robust standard mode with an average application of a damping signal (recommended setting)
1 = Chatter damping is increased for the selected axis. In some cases this is useful for one or two axes (e.g. vertical axes), but it slightly reduces the control precision (slightly higher following error).
2 = Chatter damping is significantly increased for the selected axis. This is usually advisable for no more than one axis, but it slightly reduces the control precision (increased following error).

■ **New: MP4020, bit 17 – PLC functions**

MP4020, bit 17, is used to suppress the message "Debug functions of the PLC are active" if MP4753 = 1 has triggered this message.

■ **Enhanced: MP7261.x = Number of pockets in magazine**

The indices MP7261.8 to MP7261.15 were added to MP7261.

■ **Enhanced: MP7330.x – Definition of the user parameters 1 to 16**

The input range was increased to 99999.00 maximum.

Input: 0 to 99999.00 (no. of the user parameter)

■ **New: MP7412 – Settings for cycles**

Bit 0 of MP7412 is used to influence the behavior of marker M4031 in Cycle 209. If bit 0 is set, the marker M4031 is set before the spindle orientation (M05) in Cycle 209.

Bit 0: Behavior of M4031 in Cycle 209

Input: %0000000000000000
0 = Behavior as before
1 = Set M4031 before the spindle orientation

■ **Enhanced: MP7481.x – Sequence for new and returned tool when changing tools**

The indices MP7481.8 to MP7481.15 were added to MP7481.

■ **Enhanced: MP7484.x – Search sequence in tool magazines**

The indices MP7484.8 to MP7484.15 were added to MP7484.

■ **New: MP7266.44 – Elements of the tool table**

MP7266.44 is used to show or hide the column for the new function Active Chatter Control (ACC).

Input: 0 or 1

■ **New: MP7268.x – Sorting/hiding columns of the preset table**

MP7268.x is used to hide the columns of the preset table, or to sort them as desired. MP7268.x is used to define the sequence of the columns. If MP7268.x = 0, the respective column is not displayed and is not saved during external data backup.

Input: 1 to 99
0 = Do not display the column

■ **Changed: MP7620 – Feed-rate override and spindle-speed override**

Machine parameter MP7620 bit 3 and bit 4 is used to configure the characteristic curve and the behavior of the spindle-speed override and feed-rate override. If bit 3 is not set (= 0), the feed-rate and spindle potentiometers are now available as a linear characteristic curve. On TE keyboard units without rapid-traverse override, bit 4 serves to additionally select the resolution used for the evaluation of the overrides.

Until now, bit 3 was not evaluated for TEs with rapid-traverse override and all three potentiometers were operated with a non-linear characteristic curve. The rapid-traverse override cannot be configured and is always evaluated with a non-linear characteristic curve in steps from 0 to 15000.

New behavior of MP7620 bit 3 on TEs without rapid-traverse override:

0: Linear characteristic curve for spindle-speed and feed-rate override, resolution depends on bit 4 (0 to 150 or 0 to 15000)

1: Non-linear characteristic curve for spindle-speed and feed-rate override, resolution: 0 to 15000

New behavior of MP7620 bit 4 on TEs without rapid-traverse override:

0: Spindle-speed and feed-rate override with a resolution of 0 to 150

1: Spindle-speed and feed-rate override with a resolution of 0 to 15000

New behavior of MP7620 bit 3 on TEs with rapid-traverse override:

0: Linear characteristic curve for spindle-speed and feed-rate override, resolution: 0 to 15000

1: Non-linear characteristic curve for spindle-speed and feed-rate override, resolution: 0 to 15000

New behavior of MP7620 bit 4 on TEs with rapid-traverse override:

No effect



Note

- Please note that until now MP7620 bit 3 was evaluated only if the keyboard unit did not have a rapid-traverse override. Now bit 3 is always evaluated.
- MP7620 bit 4 used to be reserved and has now been assigned a function. If bit 4 was set in your MP file, this can now have an effect on the evaluation of the overrides.

■ **Enhanced: MP7640.x – Type of handwheel**

Indices were added to MP7640. MP7640.0 to MP7640.4 are used to configure the respective type of handwheel connected. MP7640.0 for X23 of the last machine operating panel in the HSCI chain, and MP7640.4 for X29 of the MC.

■ **New: MP7642.x – Handwheel name**

With MP7642.0 to MP7642.4, you can define a name for every connected handwheel. MP7642.0 for X23 of the last machine operating panel in the HSCI chain, and MP7642.4 for X29 of the MC. For handwheels with display, the name of the currently active handwheel is shown on the display of the disabled handwheel. In addition, the handwheel name is also displayed in the pop-up window of the control when a handwheel is active. Keep in mind that the handwheel display can show a maximum number of 20 characters for the name.

Input: Handwheel name, max. 31 characters

- **Changed: 7645.x – Initialization parameter for handwheel at X23 of the last machine operating panel in the HSCI chain**
- **New: 7646.x – Reserved for future functions**
- **New: 7647.x – Reserved for future functions**
- **New: 7648.x – Reserved for future functions**
- **New: 7649.x – Initialization parameter for handwheel at X29 of the MC main computer**
- **Enhanced: MP7650.x – Counting direction for handwheel**
 Indices were added to MP7650. MP7650.0 to MP7650.4 are used to configure the counting direction of the respective handwheel connected. MP7650.0 for X23 of the last machine operating panel in the HSCI chain, and MP7650.4 for X29 of the MC.
- **Enhanced: MP7680 – Machine parameter with multiple function**
 Setting bit 18 in MP7680 suppresses the error message "Axis angle not equal to tilt angle" when a manual probing cycle is used to measure a basic rotation or to align the workpiece by means of the rotary axis. This makes it possible to measure the basic rotation using reference marks that could not be reached without a tilted head. Determining a datum is not possible, however.
 Input:
 Bit 18 – Manual measurement of basic rotation with inclined rotary axes
 0 = Error message "Axis angle not equal to tilt angle" is active
 1 = Error message is not active
- **Enhanced: MP7682 bit 16 – Reserved**
 Input: 0
- **Enhanced: MP7682 bit 17 – Reserved**
 Input: 0

1.5 New functions

■ Keys are also effective for active third desktop

The NC start key is now also effective on the third desktop (OEM screen) and is transferred to the NC (start of an NC program, if required). Furthermore, the ERR or HELP key now opens the error list and/or help text window on the third desktop.

■ Speed monitoring for slave axes

Machine parameter MP855.x (synchronization monitoring) is now effective for all slave axes of a synchronized drive (if a value > 0 has been entered), regardless of whether it is a gantry or master-slave combination. Until now this monitoring function was only effective for position controlled slave axes of a gantry combination.

■ Preset table in inches

The values in preset tables will now be shown in inches if the control was set to inches. When the files are output, the data will be saved without unit as before, i.e. converted files must always be interpreted in mm format.

■ Compensation values with higher resolution in the compensation tables

Compensation values with six decimal places (1 nm) can now be entered in the compensation tables (*.COM).

■ New function for HE Logging

The new tab **Memcheck** is now available in the **HE Logging** menu. Just like the existing tabs for the settings regarding **Syslog**, **FTrace**, and **TCPDump**, these possibilities for diagnostics and the associated settings are to be used only together with HEIDENHAIN service personnel.

■ Access to data from the NC program

The new NC blocks DATA ACCESS READ and DATA ACCESS WRITE have been introduced. These blocks make it possible for an NC program to access specific data of the NC or PLC. The same interface is used as for data access with Python. The subscription to and unsubscription from data take place automatically. The access rights are determined by the directory where the executing NC program is stored (TNC:\, PLC:\). Example:

```
0 BEGIN PGM DATA_RW MM
1 ;Read tool name of T2 from TOOL.T
2 DATA READ QS8 = "\TABLE\TOOL\T\2\NAME"
3 ;Tool radius of T2 from TOOL.T
4 DATA READ Q8 = "\TABLE\TOOL\T\2\R"
5 ;Write tool name of T5 in TOOL.T
6 DATA WRITE "\TABLE\TOOL\T\5\NAME" = QS8
7 ;Write tool radius of T5 in TOOL.T
8 DATA WRITE "\TABLE\TOOL\T\5\R" = Q8
9 ;Coordinate from line 5 of the KINEMATO.TAB
10 DATA READ QL1 = "\TABLE\TNC:\KINEMATO.TAB\NR\5\COORD"
11 ;File name from line 1 of the KINEMATO.TAB
12 DATA READ QS15 = "\TABLE\TNC:\KINEMATO.TAB\NR\1\FILE"
13 ;Write coordinate to line 7 of the KINEMATO.TAB
14 DATA WRITE "\TABLE\TNC:\KINEMATO.TAB\NR\7\COORD" = QL1
15 ;Write file name to line 2 of the KINEMATO.TAB
16 DATA WRITE "\TABLE\TNC:\KINEMATO.TAB\NR\2\FILE" = QS15
17 ;Read PLC word 4000
18 DATA READ QR1 = "\PLC\memory\W\4000"
19 ;Write to PLC word 4002
20 DATA WRITE "\PLC\memory\W\4002" = QR1
21 END PGM DATA_RW MM
```

■ Write protection for lines of the preset table

Lines in the preset table of the TNC can now be write-protected by the user. A write protection for individual lines or entire areas (e.g. **PR.LINESLOCKED = 1,4-8,22**) in the preset table can now be set by an entry (**PR.LINESLOCKED =**) in the **TNC:\TNC.SYS** file. Write-protection can be activated only for the first 255 lines of the preset table. Protected lines are indicated in the editor by a different color.

■ Automatic connection of network drives

The waiting time for the automatic connection (mounting) of network drives was increased. Up to now, after start-up of the operating system the control tried to connect the network drives automatically for approx. 7 seconds. Now it keeps trying for up to a minute. In this way, the automount succeeds even when a device in the network (e.g. a switch in the machine) does not start up until the machine switches on.

■ Backing up and restoring operating system settings

During backup and restoring via TNCremo or TNCcmd, the following operating system settings are now also saved and restored:

- System time settings via Settings/Date/Time dialog
- Language settings via Settings/Language dialog
- Network settings via Settings/Network dialog
- Settings for the connected network drives via Settings/Shares dialog
- Settings of the VNC server via Settings/VNC server
- Screen saver settings via Settings/Screen Saver dialog
- Settings of the XFCE Window Manager via Settings/WindowManagerConfig dialog
- Settings of the diagnostic tools under HELogging
- Settings of the Remote Desktop Manager

■ Two handwheels possible in the HSCI system

As of software version 60642x-03, two handwheels can be connected and used alternately in the HSCI system.

With software version 03, one handwheel can be connected to X23 of the machine operating panel as before. It has the index .0 in the handwheel MPs. The second handwheel must be connected to X29 of the MC main computer unit. It has the index .4 in the handwheel MPs. Machine parameters MP7640 (type of handwheel) and MP7650 (counting direction) were therefore expanded to 5 indices. MP7642 (handwheel name) was added, as well as MP7646 to MP7649, which—just like MP7645—contain handwheel-specific parameters for the other handwheels. In future software versions the number of possible handwheels is planned to be increased further. The corresponding indices of the mentioned machine parameters have already been reserved for this purpose. MP7641 for the handwheel settings is globally effective for all handwheels in the system. Please note that at present only the handwheel connected to X23 of the last machine operating panel in the HSCI chain is supported if there is more than one machine operating panel in the HSCI system. A machine operating panel at X501 of the MC main computer unit is automatically always the last machine operating panel in the HSCI system.

As before, the permissive buttons of all handwheels must be evaluated by suitable external devices. No more than one permissive device may be active on the machine at any one time. If you are using a control system with integrated functional safety (FS), FS inputs can be used for the evaluation of the permissive buttons. As before, the emergency stop buttons of all handwheels must be evaluated by suitable external devices. All emergency-stop devices of the machine must function reliably at all times. Under no circumstances are these devices to be deactivated. If you are using a control system with integrated functional safety (FS), the emergency-stop inputs (–ES.A, –ES.B) should be used for the evaluation of the emergency-stop buttons. Several emergency-stop buttons can be connected in series for this purpose.

If only handwheels with display (HR 420, HR 520, HR 550) are involved, the active handwheel is defined solely through the selection key on the handwheel. If there is exactly one HR 110 panel-mounted handwheel in the system, it becomes active automatically if control is not assigned to the display handwheel.

In all other cases, the handwheel must be activated via the PLC program if more than one handwheel is involved. The active handwheel can be defined with PLC Module 9036 and the new selection parameters 20 (active handwheel) and 21 (standard handwheel). Number 70 (active handwheel) and number 71 (standard handwheel) of the PLC Module 9035 can be used to read back the currently active handwheel.

The PLC interface for the handwheels will not be expanded. The currently active handwheel always accesses the markers and double words of the PLC that have already been used before (e.g. M466x for the HR keys).

Further constraints:

- If there is only one handwheel in the HSCI system, this is the standard handwheel. In this case, the behavior of software version 03 is completely compatible with the previous versions without the need for any modifications.
- If one of the display handwheels is active, the other handwheels are disabled, and this is indicated by a corresponding dialog in the display.
- If there is more than one display handwheel in the system, control can be handed over only if override control is also withdrawn from this handwheel.
- If there are several handwheels in the system and one of them is an HR 110, the panel-mounted handwheel is automatically the standard handwheel.

- If there is more than one handwheel without display (HR 410 and earlier versions), this type of handwheel must be activated via the PLC program (e.g. through evaluation of the permissive buttons). PLC Module 9036 was expanded accordingly.
- If the PLC program wants to prevent an HR 420 from withdrawing control from an HR 410, the Marker 4680 (Disable HR 420 activation) must be set.
- Software version 60642x-03 supports only one HR 550 wireless handwheel in the system.



Note

The PLC program must ensure that only one of the operating units (handwheels, machine operating panels) is active at any one time so as to avoid danger to the operator.

Please note that it may be necessary for the operator to confirm which operating unit is active before it can actually be used. The risk assessment of your machine must show whether this is necessary. Such a requirement could be realized, for example, by pressing the permissive button of the respective operating unit. Then the operating unit that should be active according to the PLC program cannot be used for any machine functions until the permissive button of this operating unit has been pressed.

The emergency stop function of the individual operating stations must always be active, however!

■ Notes on the adaptive control functions

- Only axes that are controlled by the same DSP can be compensated with each other for software option 141 (CTC). This means that such axes cannot be distributed to multiple controller units.
- Up to five controller parameters can be changed by each of the two options PAC and MAC. A controller parameter must be changed either by PAC or by MAC, and not by both PAC and MAC.

■ Default tool axis for test run

When an NC program is started in the Test Run operating mode, a default tool axis is now set. It is taken from the **ToolAxis** entry (permitted tool axis) of the kinematics active in the test run.

■ The number of variables in the kinematic temperature compensation has been increased

PLC words can be entered as variables in the kinematic temperature compensation ("TEMPCOMP" column of the kinematics table) and in the kinematic axis-error compensation (columns C1 to C6 of compensation value tables).

The number of maximum permitted variables has been increased from 32 to 64.

■ Switching between rapid-traverse and feed-rate potentiometers

When an NC program was run, no new block was started if an OEM-specific rapid-traverse potentiometer was used for changing from machining feed rate to FMAX or vice versa, and the newly activated potentiometer was set to 0 %. Now a new block is started and the distance-to-go display shows the values of the new NC block.

■ HR 550FS wireless handwheel

The communication between the wireless handwheel and the control was improved so that it is now less sensitive to interference in the radio connection.

■ PLC output to be set after shut-down (MP4044) now also on machine operating panel

The 24 V outputs of HSCI machine operating panels (X7) can be used for the switch-off function of a machine. Until now, only the single-channel outputs of a PL 62xx could be used.

However, the following must be kept in mind:

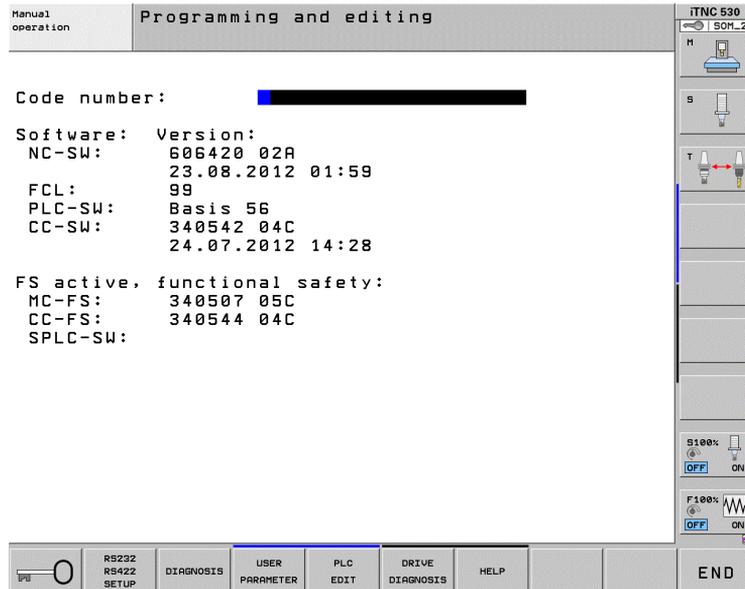
- The maximum current of an output on a machine operating panel is limited to 150 mA (500 mA with the PL 62xx).
- Galvanic isolation of the PLC supply must be ensured. The machine operating panels and their outputs are supplied with +24 V NC.

■ Modified handling of the spindle during LIFTOFF due to power failure

If the DC-link voltage falls below the threshold of 450 V, the spindle is decelerated to feed energy into the DC-link. Until now the spindle was accelerated again once the DC-link voltage exceeded the threshold of 450 V. Now the spindle is decelerated until the DC-link voltage has risen above 500 V. Then the spindle speed is maintained. This process can be repeated as often as necessary if the DC-link voltage falls below the threshold of 450 V.

■ Revised MOD dialog of the software versions

The MOD dialog for displaying the installed software version was revised:



- **Code number:** Input field for code numbers
- **NC-SW:** NC software version and date
- **FCL:** Currently set Feature Content Level (FCL)
- **PLC-SW:** Number of the PLC program
- **CC-SW:** Software version of the CC, UEC and UMC controller units and date

If the HEIDENHAIN integrated functional safety is active:

- **MC-FS:** SKERN software version of the MC main computer
- **CC-FS:** SKERN software version of the CC, UEC, UMC controller units
- **SPLC-SW:** Number of the SPLC program

■ Behavior of DCM

Depending on the settings for collision monitoring (DCM) via the COLLISION soft key and the actual operating situation, the collision monitoring behavior was sometimes ambiguous. If DCM was **active for Program Run** and **inactive for Manual Operation**, collision monitoring was inactive in the following cases:

1. If the program was interrupted in the Program Run operating modes by pressing NC stop and the MANUAL TRAVERSE soft key, for example. Collision monitoring was not active when the axes were moved with the axis-direction keys or the handwheel.
2. If the handwheel was moved in the Program Run operating modes while the program was stopped, collision monitoring was inactive during M118 or handwheel superimposition via the Global Settings.

In order to provide optimum protection, as of software version 60642x-03 the collision monitoring function is active in the above-mentioned cases if it is enabled in the MOD dialog for Program Run OR Manual Operation.

■ Nesting depth for NC programs

A maximum of 30 NC programs can be nested (subprograms, cycles, macros).

■ **Switchable characteristic curve for potentiometers**

See changed behavior of MP7620 bits 3 and 4.

■ **Input for speed encoder of the spindle can be switched**

In MP113.0 and MP 113.1 you can now set different inputs for the speed encoder of the spindle, even if the PWM output for the spindle drive stays the same. The PWM output and the two encoder inputs must be on the same drive-control motherboard (DSP) of a CC, however. This means that they must be configured on CC connectors with the same letter A or B. This may be necessary for applications in which you switch between two spindles or spindle motors that are operated with the same UM power module.

■ **Number of multifunction filters has been increased**

Another multifunction filter is available with the new machine parameters MP2547.x, MP2557.x, MP 2567.x and MP2577.x.

■ **Switch-on behavior of vertical and rotary axes**

If a vertical or rotary axis is switched on, the current that was active at the time of switch-off is now used. This is to prevent the axis from sagging when the brake is released, even if the feedforward control for the axis is not optimally set.

■ **Protection for the configuration of the Ethernet interfaces**

The settings of the non-primary Ethernet interface(s), the selection of the primary interface and the DHCP server settings are now password protected. This protects the settings from unauthorized changes. The code number 807667 or the PLCPASSWORD defined in the OEM.SYS are now required for these settings. The code number is entered using the **OEM authorization** button or soft key.

■ **Different effect of the two DIADUR keys**

The meaning of the two DIADUR keys in the lower row of the ASCII keyboard on the TE keyboard unit was changed.

The left key now opens the JH menu (operating system setting) of HEROS 5. Within a remote-desktop application, this key functions as a Windows key.

The right key switches to a remote-desktop application if the "Remote Desktop Manager" software option is enabled. If you happen to be in a remote desktop, this key switches to the next remote desktop or back to the NC application.

In a remote-desktop application, the screen switchover key now has locally the same effect as an F12 key, but you have to use the right Diadur key to switch back.

■ **Spindle override with AFC**

If the AFC soft key is set to ON, the spindle potentiometer is fixedly set to a factor of 100 % for the Program Run operating modes right from the start of cutting.

■ **Tool change within an AFC cut**

If a change to another tool (not replacement tool) takes place within an active cut, the error message "Faulty assignment of tool/cut no." is issued and an NC stop is initiated.

If AFC detects at the beginning of a cut that the tool and the associated cut number do not match, it stops instead of executing an automatic teach-in cut (NC stop) and the error message "Faulty assignment of tool/cut no." is issued. After NC start has been pressed again, a teach-in cut is carried out with the new tool.

In both cases the user can either cancel with internal stop and correct the cause of the error message, or continue with the actual data by pressing NC start.

■ SELinux

The directory itself and the data in the **PLC:\etc\selinux** directory should not be deleted. These files are directly related to the function of the SELinux safety software.

■ MP6508 switchover via PLC/FN (3 touch probes)

The machine parameter MP6508 (SE 642 transmitter/receiver unit at X112) can now be changed via the PLC or an FN function. Hence, the transmitter/receiver unit can be activated or deactivated using a macro or cycle, for example. This makes it possible to use different TT touch probes at X113 and on the SE.

■ MP subfiles

Placeholders **?** can be entered in bit-specific machine parameters in MP subfiles. This makes it possible to change individual bits specifically. For example, you can overwrite only bits 2 and 5 in an MP subfile by entering **MP10:????1?0??**.

■ New FN functions

- FN 17: SYSWRITE ID 59 NR10 =+0

When this function is called in the tool change macro, the spindle speed is recalculated using the current tool radius from TOOL.T if a cutting speed is programmed in the TOOL CALL block. The spindle speed is entered in the system datum that can be read with FN18: SYSREAD ID60 NR3.

Example of a tool-change macro showing how the changed tool data can be made effective after a new tool has been inserted:

; Change the tool

78 TOOL CALL Q1 .Q7 Z SQ3 DL+Q4 DR+Q5 DR2:+Q6

; Data in TOOL.T are updated

79 M...

; Recalculation of the spindle speed from the cutting speed

80 FN 17: SYSWRITE ID 59 NR10 =+0

; Read the spindle speed

81 FN 18: SYSREAD Q3 = ID60 NR3 IDX0

; Changed tool data become effective

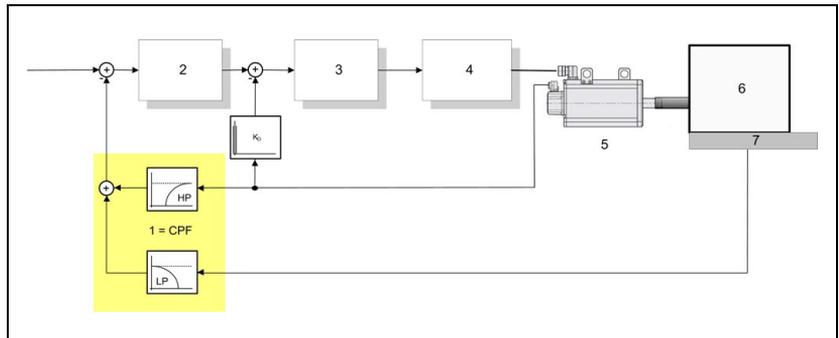
82 TOOL CALL Q1 .Q7 Z SQ3 DL+Q4 DR+Q5 DR2:+Q6

■ Only CC 6xxx: Crossover Position Filter (CPF)

The CPF can be used only in dual-encoder systems with rotary and position encoders. In such systems, the CPF is used to passively dampen certain machine vibrations caused by the drive components in the drive chain. In order to stabilize the position control loop in such systems with resonances, the position signal from the position encoder, which is filtered through a low-pass filter, and the position signal from the motor speed encoder, which is filtered through a high-pass filter, are added and made available to the position controller as actual position value. The filter frequency (separation frequency) is set in MP2609 for this. The typical filter frequency values in MP2609 are between 20 Hz and 80 Hz.

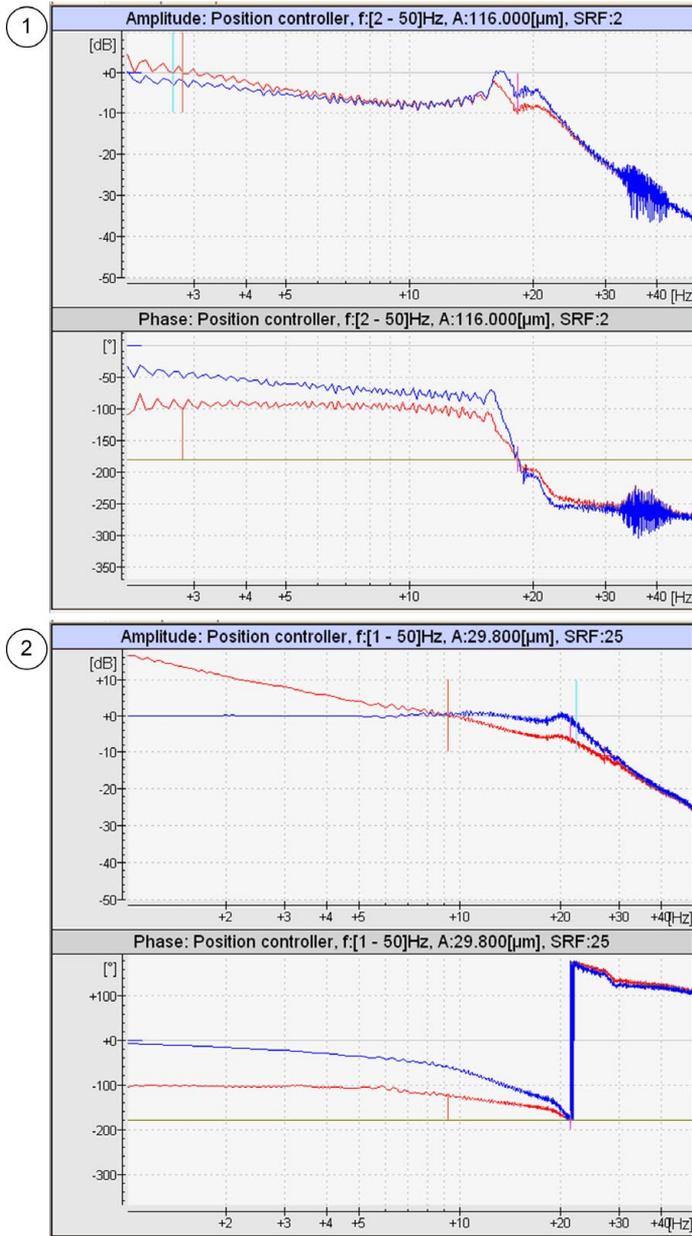
The separation frequency can be determined with the position-controller Bode diagram of the TNCOpt PC tool. The frequency at which an excessive resonance occurs is to be selected as the value for MP2609. After the separation frequency has been defined, the kV factor can be increased.

The block diagram shows how the crossover position filter works:



- 1: Crossover position filter (CPF) with high-pass (HP) and low-pass (LP)
- 2: Position controller
- 3: Speed controller
- 4: Current controller – power module
- 5: Motor
- 6: Machine
- 7: Linear encoder

The two Bode diagrams in TNCopt illustrate the procedure for setting the separation frequency for the CPF. Diagram 1 shows that the excessive resonance is reached at 15 Hz. Diagram 2 shows the frequency response after 15 Hz has been entered in MP2609.



■ Only CC 6xxx: Active Chatter Control (ACC) (software option 145)

Strong forces come into play during heavy cutting (power milling). These forces are transmitted between the milling spindle and the machining axes via the tool and the workpiece. Depending on the tool spindle speed, the resonances in the machine tool and the removal volume (metal-removal rate during milling), the tool can sometimes begin to "chatter." This chattering places heavy strain on the machine, and causes ugly marks on the workpiece surface. The tool, too, is subject to heavy and irregular wear from chattering. In extreme cases it can result in tool breakage.

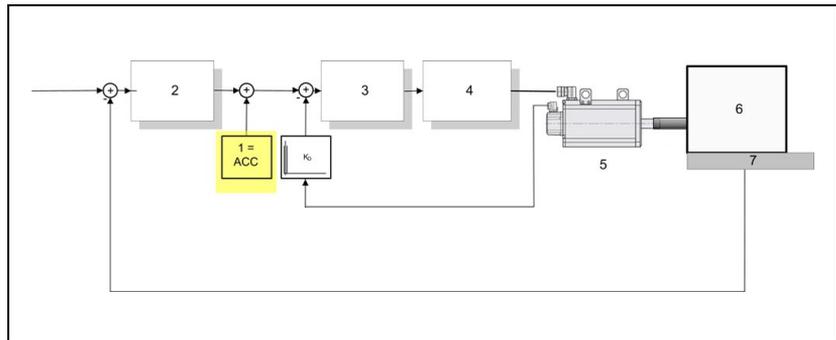
To reduce a machine's inclination to chattering, HEIDENHAIN offers software option 145 Active Chatter Control (ACC). The use of this option is particularly advantageous regarding the attainable metal removal rate during heavy cutting. Depending on the type of machine, the metal removal rate can be increased significantly within the same machining time. Software option 145 must be enabled in order to be able to use ACC. ACC must then be enabled for all machining axes via MP2700 bit 4. The machining axes are usually the axes X, Y, and Z. Also, for all tools that are used for heavy cutting, "1" must be entered in the **ACC** column of the tool table, and the number of tool teeth in the **CUT.** column.

If a tool with active ACC function is inserted, a damping signal reducing the chatter tendency is automatically added in the control loop.

You have the additional possibility of using machine parameter MP2710 to influence the weighting of the damping signal between the X, Y, and Z axes. In this machine parameter you can select the axis to which the higher-weighted damping signal is to be applied.

Besides this automatic function, you have the option of creating an ACC parameter file with TNCopt 7.0 and transfer it to the control. Since every tool influences the chatter characteristics of the cutting process, an individual file can be created for each tool. Alternatively you can also create a machine-specific file. The respective chatter frequency range and the intensity of the damping signal can be set using the parameter files. If a tool with active ACC function is inserted, the NC first automatically searches for a tool-specific parameter file. If no such file is available, it searches for a machine-specific file. If this is not available either, ACC operates in automatic mode.

The block diagram shows how ACC works:



- 1: Active Chatter Control (ACC)
- 2: Position controller
- 3: Speed controller
- 4: Current controller – power module
- 5: Motor
- 6: Machine
- 7: Linear encoder

1.6 Software option 144—Motion Adaptive Control (MAC)

The MAC option (only CC 6xxx) provides a way to change machine parameters depending on motion-dependent input quantities (e.g velocity or following error). This makes it possible, for example, to realize a velocity-dependent adaptation of the kV factor on motors whose stability changes through the various traversing velocities.

Considerations before using MAC

Considerations before using MAC:

Does the machine have a pronounced dependence on velocity or acceleration regarding the controller parameters?

This becomes apparent:

- The internal control oscilloscope if resonances occur at different velocities or accelerations, and therefore the controller variables (actual speed, nominal current) are subject to oscillations.
- Audible oscillations at certain velocities and/or accelerations, or at a certain velocity threshold.

Is there a need to change any controller settings or feedforward functions depending on the velocity or acceleration?

This may be required for:

- Master-slave torque drives whose tensioning is to be reduced for rapid traverse (high acceleration), for example.

Preparation and activation

Activating MAC:

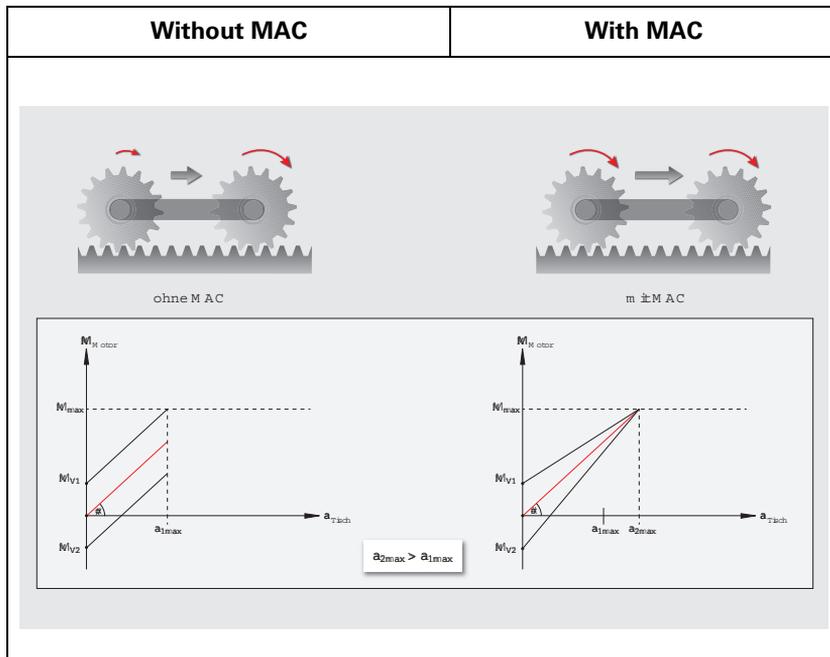
- ▶ Activate MAC via TNCopt (using the "Expanded compensation types" menu item).
- ▶ Set the controller parameters of the axis for different velocities or accelerations.
- ▶ Use TNCopt to enter the parameters you have determined for up to four support points. You can define up to five machine parameters and up to five position-dependent filters (with two support points). The parameters are interpolated linearly between the support points.
- ▶ Following error, jerk, acceleration and velocity are available as input quantities, to which the parameters are adjusted. Optionally, the input quantity can be considered unsigned. This is useful in symmetrical situations, for example to obtain the same adjustment for negative and positive accelerations.

This function is described in detail in the TNCopt 7.0 online help. Available as of the beginning of 2013.

Example

One possible application is the acceleration-dependent change of the tensioning torque between master axis and slave axis for master-slave torque control.

With the MAC option, this arrangement makes it possible to attain a significantly higher maximum acceleration at rapid traverse, for example through parameterized reduction of the tensioning torque with increasing acceleration.



1.7 DHCP server

As of software version 60642x-03, a DHCP/DNS server can be configured and started. This server can be used to supply the devices that are installed in the machine network with dynamic IP addresses and to provide a name resolution ("DNS service").

The server cannot be started on the primary network interface of the MC. For this function to be available, the MC must have at least two network interfaces so that the server can be activated.

The server provides the following services for this interface:

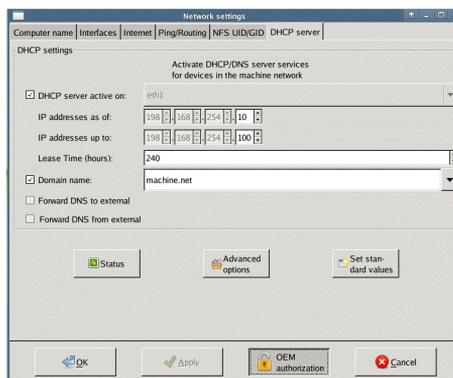
- DHCP service (dynamic IP address assignment)
- DNS service (name resolution)
- NTP service (time synchronization)

Configuring the DHCP/DNC server

■ DHCP/DNC server configuration

Open the network settings of your control. The configuration dialog is protected from unauthorized changes by the PLCPASSWORD.

- ▶ Select the **DHCP** tab
- ▶ Press the **OEM authorization** button or soft key
- ▶ Enter the defined PLCPASSWORD or the code number 807667



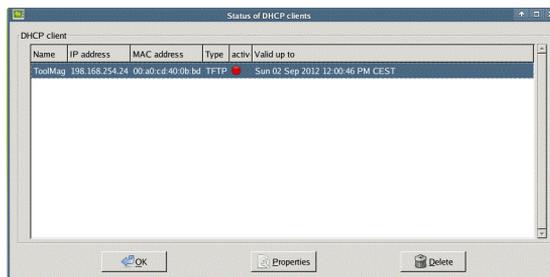
Tab	Option	Meaning
DHCP		Configuration of the DHCP/DNS server:
	DHCP server active on:	<p>Activation of the DHCP server and configuration of the Ethernet interface. Also activates other services, such as DNS or NTP. The option appears dimmed if the MC has only one Ethernet interface.</p> <ul style="list-style-type: none"> ■ Selection of the interface on which the DHCP server is to start. The primary interface (eth0) cannot be selected here. The primary interface can be selected on the first tab Computer name. If the MC has two interfaces, the non-primary interface appears dimmed here. The interface used here must have a static IP address and network mask. This is necessary because the address range of the dynamic IP addresses of the DHCP server is derived from this IP address and network mask. ■ IP addresses as of: Defines which IP addresses are output by the DHCP server. The values that appear dimmed are transferred from the static IP address of this network interface of the MC. The number of segments that appear dimmed in the IP-range setting is defined in the network mask of the static IP address of the network interface (Interfaces tab -> Configure). The start (lowest IP address) of the dynamically used IP address range is defined here. ■ IP addresses up to: Definition of the highest IP address of the dynamically used IP address range. The IP addresses are not retrieved in consecutive order. ■ Lease Time (seconds): Lease Time is the time during which an IP address is reserved for a client. If the client logs on again within this time, he will definitely be assigned the same dynamic address again. During operation, a client automatically extends the validity of the assigned dynamic IP address. This means that if a client is switched off longer than the Lease Time, it can happen that the dynamic IP address is used for another client.
	Domain name:	Here you can enter the name of the internal network. The name is required, for example, to distinguish devices in the internal network from devices with the same name in an external network, or if you want to access the internal network from devices in an external network. The domain name can be chosen freely or you can select one of the three suggestions (machine.net, tnc.net or <hostname>-net.com) using the arrow.
Forward DNS to external:	This option can only be selected if the IP-Forwarding option has also been selected on the Interfaces tab. If this option is activated, the name resolution for the devices in the internal network can also be used by devices in the external network. The option is required, for example, if you want to reach a device in the internal network with dynamic IP address from a PC in the external network by using the host name of the device.	

Tab	Option	Meaning
	Forward DNS from external:	This option can only be selected if the IP-Forwarding option has been selected on the Interfaces tab. If this option is set, DNS requests from devices in the internal network that cannot be answered by the MC's DNS server are forwarded to the name servers of the external network. In the configuration of the primary interface, you can define the name servers to be dynamic (via DHCP) or static.
	Status	Using the Status button or soft key, you can call an overview of the internal devices with dynamic IP address and define settings for these devices, see page 46.
	Advanced options	Here, experts can enter additional parameters for the DNS/DHCP server, see page 48.
	Set standard values	Using the Set standard values button or soft key, you activate the DHCP/DNS server (if there are several network interfaces and a static IP address on the non-primary network interface). The factory default settings are set.

Status of DHCP clients

■ Requesting the status

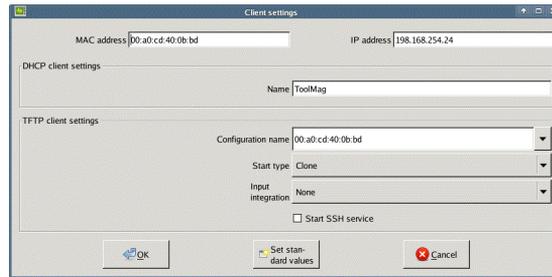
All valid IP addresses output by the DHCP server are displayed in the **Status of DHCP clients** window. Valid means that the **Lease Time** of the IP addresses has not expired yet. If you select a line in the list, you can define settings for the device using the **Properties** button, see page 47. The IP address and all settings for the device can be deleted by pressing the **Delete** button. The displayed list is updated approx. every 5 seconds.



Column	Meaning
Name	Host name of the device. If the device logs on to the DHCP server using a host name, this name will be used. If the device does not report a name, the standard name will be generated from the MAC address (all colons will be replaced by hyphens). For DHCP devices, the device name can be overwritten using the Properties button.
IP address	The dynamically assigned IP address of the device in the internal network
MAC address	The MAC address of the device in the internal network
Type	For external devices, the type DHCP must be entered if they request a dynamic IP address. Devices with static IP in the internal network are not shown because these devices do not log on to the DHCP server.
Active	If the device replies to an ICMP echo request (ping), a green dot is entered here. If the device is not available or does not react, a red dot is entered.
Valid up to	Time at which the validity of the IP address will expire without being renewed. If you subtract the Lease Time from this time you can ascertain at what time the device last booted or extended the validity of the IP address.

DHCP client settings

- Client settings
In this window you can enter the names of all devices (overwrite).

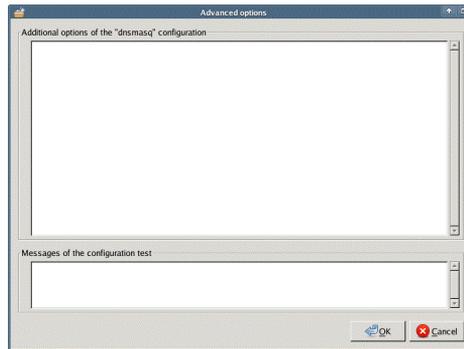


Dialog	Option	Meaning
Client settings	MAC address:	MAC address of the client
	IP address:	IP address of the client
	DHCP client settings	<ul style="list-style-type: none"> Name: Here you can overwrite the host name of the device. The DHCP server tries to set the host name of this device as indicated here. It depends on the device, however, whether it will really change its name as specified by the DHCP server. On most devices the name must be specified in the device itself, or at least the configuration in the device must be modified such that the device retrieves its host name from the DHCP. Regardless of whether the device itself changes its name as specified, this IP-address name is entered in the name server on the MC, and therefore this name must be used to access the device in the network.
	TFTP client settings	Currently not yet supported. Preparation for future expansions.
	Start SSH service	Currently not yet supported. Preparation for future expansions.

Advanced options for DHCP

■ Advanced options

The advanced options are intended for experts who want to exploit the full range of possibilities offered by the "dnsmasq" program that provides the DHCP and DNS services.



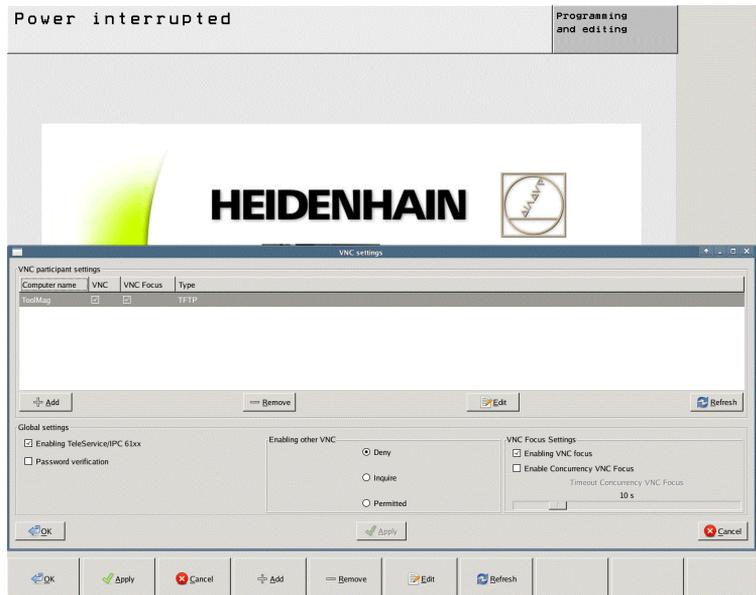
Column	Meaning
Additional options	Here you can enter the additional options.
Messages	If you confirm the window with OK , the options entered are checked. If an error is found, the error is described here and the window will not be closed. If no error is found, the window will be closed. The window can always be closed with Cancel . All changes are rejected in this case.

1.8 VNC settings

The behavior of the different VNC clients (e.g. IPC) can be configured using the VNC settings. This includes, for example, operation via soft keys, mouse and the ASCII keyboard.

Open the VNC settings as follows:

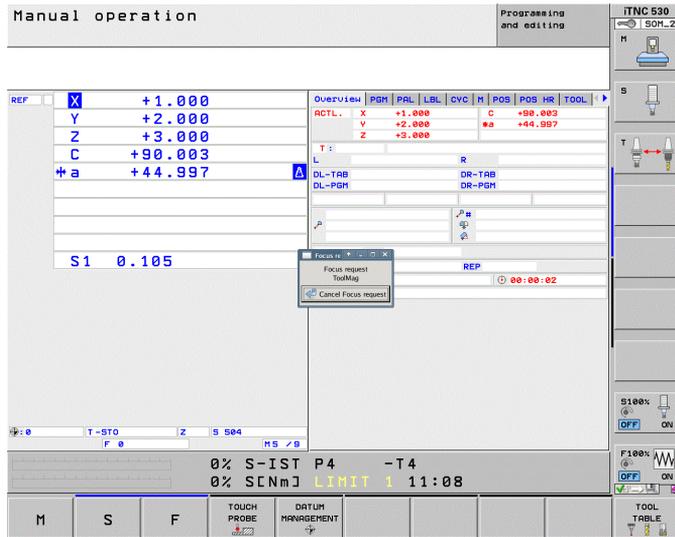
- ▶ Make the taskbar at the bottom of the screen visible by moving the mouse pointer over it
- ▶ Press the green HEIDENHAIN button to open the JH menu
- ▶ Select the **Settings** menu item
- ▶ Select the **VNC settings** menu item



Dialog	Option	Meaning
VNC settings		
Overview of the VNC settings		
	Computer name:	IP address or computer name
	VNC:	Connection of the client to the VNC viewer
	VNC Focus	The VNC client participates in the focus assignment
	Type	<ul style="list-style-type: none"> ■ Manual Manually entered VNC client ■ Denied This VNC client is not permitted to connect ■ TeleService/IPC 61xx VNC client via TeleService connection ■ TFTP Currently not yet supported. Preparation for future expansions. ■ DHCP Other computer that retrieves an IP address from this computer ■ Add Add new VNC viewer/client ■ Remove Deletes the selected client. Possible only for clients that were entered manually. ■ Edit Edit the configuration of the selected client ■ Refresh Updates the display. Necessary during connection attempts while the dialog is open.
Global settings		
	Enabling TeleService/IPC 61xx	Connection via TeleService/IPC 61xx is always permitted
	Password verification	The client must enter a password for verification. If this option is active, the password must be entered when the connection is established.
Enabling other VNC		
	Deny	Access generally denied to all other VNC clients.
	Inquire	During connection attempts a corresponding dialog is opened.
	Permitted	Access is generally granted to all other VNC clients.
VNC Focus Settings		
	Enabling VNC focus	Enable focus assignment for this system. Otherwise there is no central focus assignment. In the default setting, the focus is actively reassigned by the owner of the focus by clicking the focus symbol. This means that the owner of the focus must first release the focus by clicking the focus symbol before any other client can retrieve the focus.

Dialog	Option	Meaning
	Enable Concurrency VNC Focus	In the default setting, the focus is actively reassigned by the owner of the focus by clicking the focus symbol. This means that no other VNC client can retrieve the focus until the focus has been released by clicking the focus symbol on the VNC client owning the focus. If concurrency focus is selected, any VNC client can retrieve the focus at any time without having to wait for the current owner of the focus to release it.
	Timeout Concurrency VNC Focus	Time period within which the current owner of the focus can object to the focus being withdrawn or can prevent the reassignment of the focus. If a client requests the focus, a dialog in which the reassignment of focus can be refused appears on all clients' screens. If a VNC client requests this focus, a corresponding dialog window appears on all VNC clients' screens. The current owner of the focus can refuse the reassignment of focus in this dialog.
Focus icon		
		Current status of VNC focus on the respective client: Focus is owned by other client. Mouse and keyboard are disabled. Focus assignment is controlled either by clicking the focus icon with the mouse or by using the Connect key.
		Current status of VNC focus on respective client: Focus is owned by current client. Entries can be made.
		Current status of VNC focus on respective client: Owner of focus is requested to give the focus to another client. Mouse and keyboard are disabled until the focus is assigned unambiguously.

If **Enable Concurrency VNC Focus** is selected, the dialog window shown below appears. This dialog makes it possible to refuse that the focus be given to the requesting client. If this is not done, the focus will be given to the requesting client after the specified time period has elapsed.



The PLC Modules 9286, 9287 and 9288 enable you to use the PLC program to control the focus assignment.

For multiple clients or operating units, the focus assignment procedure depends on the design and the operating situation of the machine. The task of realization of and the responsibility for the focus assignment therefore lies with the machine tool builder.

All external VNC clients (non-HEIDENHAIN devices) have read-only rights to display the screen contents, and not write rights.

1.9 PLC functions

- Version entry for SPLC program in the OEM.SYS
For an SPLC program, a version identification can now be entered in the OEM.SYS file. The identification must be assigned to the new token SPLCSOFTVERS. The entry can be displayed by pressing the MOD key and can be requested via LSV2.
- The COPY SAMPLE FILES soft key can now also be used to create the PLC:\PROTO\plc\SPLcApiMarker.def file again.
- Detection of a spindle that is coasting to a stop after a restart
The current spindle speed is already available in the PLC word W322 or D368 before the control voltage is switched on (not with CC 422).
- Backing up and restoring the nonvolatile PLC memory
When creating service files or shutting down the control, the nonvolatile memory of the PLC program is now saved automatically to the file PLC:\JHPLCLOG\PLCREMADATA.LOG. Restoring the nonvolatile area from the file is possible only by using the existing RESTORE M/B/W/D soft key in the PLC mode of operation under the TABLE soft key.
- Default language for text file
If you use the TEXTFILE command to show a text file in PLC mask files (PLC Module 9210) and this file does not exist in the set language (MP7230.3), the NC software now also searches the English language directory (PLC:\LANGUAGE\ENGLISH) for the text file .
- LSV2 message window possible without software option 18
Displaying a message window via an LSV2 telegram is now also possible without software option 18 HEIDENHAIN DNC. The window was changed to "GTK" and can be closed on the control by using CTRL + END or the Close button of the window. You can now also define whether this window is to be displayed in the Machine mode of operation, the Editing mode of operation or in both modes of operation.
- New word W1004
While an NC error message is being displayed, the associated error number is saved to the new PLC word 1004. If no error message is active, the PLC word has the value 0.
- Modified: Behavior of M4623
The start of DNC operation can be disabled with the M4623. Until now this also applied to TNCopt. As of this software version, you can start TNCopt independently of marker M4623.

After the SHUTDOWN soft key has been pressed, the PLC program can delay or prevent the display of the pop-up window (shutdown: yes/no). The message "The TNC will be shut down and machine functions will be performed" in the operating modes window indicates this condition. An appropriate user guidance must have been programmed by using the available tools, such as dialog texts, pop-up windows, vertical soft key, etc., in order to inform the user about concluding operating steps that may be necessary.

The previous possibilities of configuring a shutdown (MP4040 and the subsequent functions) can continue to be used, because these settings do not become effective until the prompt "Do you really want to switch off?" has been acknowledged.

■ **New: M4195**

The new marker M4195 is set when the SHUTDOWN soft key is pressed (= 1). The PLC program can prevent the control from shutting down by resetting the marker in the next PLC cycle. In this phase (M4195 = 0), the text "The TNC will be shut down and machine functions will be performed" is displayed in the operating modes window and the PLC program can execute specific actions before shutting down the control. The TNC remains in this condition until the marker M4195 is set again.

■ **New: M4197**

If the PLC actions cannot be concluded, the PLC can cancel shutdown of the control by setting marker M4197 = 1. Then the TNC returns to its original mode of operation and the two markers also return to their initial state.

Diagnostics

In order to make debugging in PLC programs easier, the following markers can be used to record additional PLC module calls and their parameters. The data are written to the new log file PLC:\JHPLCLOG\PLCMODULES.LOG. If one of the three markers is set, this results in the one-time NC error message "Debug functions of the PLC are active." The existing log files PLC:\PLCDEBUG.LOG, PLC:\MYDEBUG.LOG, PLC:\VERSION.TXT and PLC:\MakeCycle.log are now contained in the directory PLC:\JHPLCLOG\. Also, the file PLC:\VERSION.TXT was renamed to PLC:\VERSION.LOG. When updating the software, the previous log files will be identified by the extension .OLD and will be copied to the new directory. Then they will be deleted from the previous directory PLC:\.

■ **New: M4751**

Recording of transfer and return parameters of PLC modules that are called in the cyclic PLC program.

■ **New: M4752**

Recording of transfer and return parameters of PLC modules that are called in submit/spawn processes.

■ **Modified: M4753**

The existing function no longer writes the recorded data to the previous file PLC:\DEBUG.LOG, but to the new file PLC:\JHPLCLOG\PLCMODULES.LOG.



Note

If the PLC program Basic_54 is used and the message "Debug functions of the PLC are active" occurs, you need to correct the PLC project with respect to M4753 or set MP4020 bit 17, because M4753 is set permanently in this version of the PLC basic program.

More information on the use of M4751, M4752 and M4753:

- The markers may be used for debugging only temporarily and must not be set in the factory default setting of a PLC.
- M4753 is/was set permanently by the PLC basic program Basic_54. MP 4020 bit 17 must be set in order to suppress the message without having to modify the PLC program.
- A maximum of 1 MB of data can be written to the file PLCMODULES.LOG; any data exceeding this limit will be copied to .OLD.
- A maximum of 2 MB is available for both the file PLCMODULES.LOG and the file PLCMODULES.LOG.OLD. With the PLC basic program for example, the 2 MB are exhausted some seconds after the power interrupted message has been acknowledged if all debug markers are set.
- During debugging in the PLC program, it is advisable to position the markers as close as possible to the code position to be examined, and not for the complete PLC scan.
- On OEM machines, PLC debug code that has been output for debugging must afterwards be replaced by the original code again.

Some PLC modules that return an error code to the PLC program via the PLC program stack (PL command) did not influence the error word W1022 if an error occurred (marker M4203 set). The value in W1022 remained unchanged at a previous value. The PLC modules affected will be modified with this NC software version and will also write to word W1022 in the future. As a result, all PLC modules will have the same behavior and will transfer an error code in W1022. You can then evaluate M4203 and W1022 in order to display an error message. The following PLC modules are affected by this change:

- PLC Module 9031 – Overwrite machine parameter
- PLC Module 9032 – Read machine parameter
- PLC Module 9084 – Display PLC error message with additional data
- PLC Module 9090 – Select a line in the pallet table
- PLC Module 9092 – Search for an entry in the tables selected for execution (.T/.D/.TCH)
- PLC Module 9095 – Select active line in configuration file
- PLC Module 9220 – Renewed traversing of the reference marks
- PLC Module 9221 – Start a PLC positioning movement
- PLC Module 9223 – Free rotation
- PLC Module 9227 – Position auxiliary axes and NC axes
- PLC Module 9281 – Select a line in the pallet table
- PLC Module 9310 – Change machine parameters in the current machine-parameter file
- PLC Module 9312 – Change machine parameters in the current machine-parameter file
- PLC Module 9313 – Read machine parameter

Information on the individual, new values for W1022 can be found in the respective module description in the Technical Manual.

In rare cases this modification can influence the behavior of the PLC program.

HEIDENHAIN recommends:



Note

- Test the behavior of the PLC program on the machine.
- If the modification of word W1022 causes problems, you need to adapt your PLC program to the new behavior.

Changes to PLC modules

- PLC Module 9036—Write status information, activate/deactivate functions

The virtual tool axis VT can be selected under number 6 using the value 999 or -1.

The active handwheel can be defined with the new selection parameters 20 (active handwheel) and 21 (standard handwheel) if more than one handwheel is connected to the control. Number 70 (active handwheel) and number 71 (standard handwheel) of the PLC Module 9035 can be used to read back the currently active handwheel.

- Note on PLC Module 9193 – Set the operating times

So as not to impair the behavior of the overall system, the module may only be used for re-initializing the machine times, and may not be called too often. The behavior of the module was changed such that the module (and therefore the calling submit or spawn job) waits for acknowledgment.

- Note on PLC Module 9132 – Selection of override potentiometers

If PLC Module 9132 was used to select another machine operating panel, no automatic reset to the operating panel with the number 0 took place when the PLC was restarted (e.g. compile, acknowledgment of PLC run-time error).

- Note on PLC Module 9228, 9229 – Coupling function for auxiliary axes

As long as the coupling function in the interpolator has not taken effect, status module 9229 reports the status "0" for the coupling axis (0: No coupling function active for programmed axes).

Until now, PLC Module 9229 reported an active coupling function as soon as the coupling function was requested via PLC Module 9228, even if the coupling function had not yet taken effect in the interpolator.

Module 9046 Copy controller data to the PLC memory

Module 9046 is used to copy controller data (data of the CC) to the PLC memory via configured channels.

Constraints:

- The module must not be called in the cyclic PLC program part.
- The module should only run once for each channel. Once a channel has been configured successfully, the programmed CC data is copied to the PLC memory at the PLC clock rate.
- If a channel that has already been configured must be reconfigured (other CC data or other axis), the channel must be reset first. Direct reconfiguration leads to a module error.
- Only with CC 6xxx.

Call:

PS	B/W/D/K	<Mode> 0: Configuration of cyclic data channels from the CC to the PLC
PS	B/W/D/K	<Axis number> 0 to n: Dependence on MP4020 bit 14
PS	B/W/D/K	<Code for CC data> Mode 0: -1: Reset channel (default setting) 0: Mass moment of inertia of the table [kg*m ²] 1: Mass of the table [kg] 2: Total mass moment of inertia (table and motor) [kg*m ²] 3: Total mass (table and motor) [kg] 4: Total power [W] 5: Axis power [W] 6: Torque-producing nominal current (with sign) [mA] 7: Total actual current (with sign) [mA]
PS	B/W/D/K	<Data channel> Mode 0: 0 to 15, up to 16 channels can be programmed
PS	B/W/D/K	<Target address in PLC memory> Mode 0: Double-word address to which the value of the programmed data channel is copied
CM	9046	
PL	B/W/D	<Error code> 0: No error. Data channel was (already) configured 1: Call was not in a submit/spawn job 2: NC/CC does not support the desired function 3: Invalid value in the <Mode> parameter 4: Invalid value in the <Axis number> parameter 5: Invalid value in the <Code for CC data> parameter 6: Invalid value in the <Channel number> parameter 7: Invalid double-word address in the <Target address in PLC memory> parameter 8: Programmed data channel is already being used 100: Programmed CC data is not available for programmed axis

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Axis does not exist or status interrogation faulty
W1022	1	Invalid value for Mode parameter, CC code
	2	Invalid axis number programmed
	3	Invalid value for Channel parameter programmed
	4	Memory range for the programmed double-word address is too small or does not exist
	5	Programmed address is not a double-word address (not divisible by 4)
	6	Programmed data channel is already being used
	20	Call was not in a submit/spawn job
	45	Internal error
	51	Programmed CC data is not available for this axis on the CC, or the PLC module is not supported by this NC software
	61	Internal error

Module 9047 Reset or initialize controller algorithms

Module 9047 can reset controller algorithms or reinitialize them.

Constraints:

- The module must not be called in the cyclic PLC program part.
- Only with CC 6xxx.

Call:

PS B/W/D/K <Mode>
0: Commands to the CC (reset/reinitialize)

PS B/W/D/K <Axis number>
-1: All axes
0 to n: Dependence on MP4020 bit 14

PS B/W/D/K <CC command>
0: Reset Load Adaptive Control (SW option 143)
1: Deactivate all adaptive controller parameters
2: Activate all adaptive controller parameters

CM 9047

PL B/W/D <Error code>
0: No error. Reinitialization successful
1: Call was not in a submit/spawn job
2: NC/CC does not support the desired function
3: Invalid value in the <Mode> parameter
4: Invalid value in the <Axis number> parameter
5: Invalid value in the <CC command> parameter
100: Programmed CC data is not available for programmed axis

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Axis does not exist or status interrogation faulty
W1022	1	Invalid value for Mode parameter, CC code
	2	Invalid axis number programmed
	20	Call was not in a submit/spawn job
	45	Internal error
	51	Programmed CC command is not available for this axis on the CC, or the PLC module is not supported by this NC software
	61	Internal error

Module 9170 Find the current torque

Module 9170 makes the averaged, maximum and minimum load values of the drive of an axis available between two PLC cycles as a reference value for the rated torque value or rated power. One mode also supplies the algebraic sign of the averaged load.

Call:

PS B/W/D/K <Axis>
Axis-specific: Dependence on MP4020 bit 14

PS B/W/D/K <Mode>
0: Torque value in per cent of the rated torque
1: Torque value in tenths of per cent of the rated torque
2: Torque value in tenths of per cent of the rated torque with algebraic sign; minimum and maximum torque will not be calculated (=0)

CM 9170

PL B/W/D <Averaged torque value>
Mode 0: In per cent
Mode 1: In tenths of per cent
Mode 2: In tenths of per cent with algebraic sign

PL B/W/D <Maximum torque value>
Mode 0: In per cent
Mode 1: In tenths of per cent
Mode 2: 0

PL B/W/D <Minimum torque value>
Mode 0: In per cent
Mode 1: In tenths of per cent
Mode 2: 0

Error code:

Marker	Value	Meaning
M4230	0	Torque ascertained
	1	Error code in W1022
W1022	1	Invalid mode
	2	Invalid axis number

Module 9279 Shut down control (configurable)

PLC Module 9279 carries out the reset function of the NC. Depending on the <Mode> parameter, configuration parameters concerning the shutdown process can be taken into account.

Constraints:

- The functionality of the shutdown delay with PLC markers M4195 and M4197 is not available when shutting down via this module.

Call:

PS B/W/D/K <Mode>

0: Shut down the control without evaluating the MP configuration for shutting down (MP4040 to MP4042)

1: Shut down and restart the control

2: Shut down the control depending on the MP configuration for shutting down (MP4040 to MP4042)

CM 9279

Error code:

Marker	Value	Meaning
M4203	0	Control reset was carried out
	1	Error code in W1022
W1022	2	Mode parameter is invalid
	20	Call was not in a submit/spawn job

Module 9283 Generate tool-usage file

This module is used to generate a tool-usage file for a specified part program, including its subprograms.

Constraints:

- In order to create the file, the Test Run mode must be the active editing mode on the control.
- The NC program is selected automatically and started in Test Run.
- Generation of the tool-usage file must be permitted with bit 2 of MP7246.
- The module must be programmed in a spawn job, since program run can take a long time and submit subprograms would be blocked for a long time.

Call:

```
PS    B/W/D/K <Mode>
        0: NC program including subprograms
PS    B/W/D/K <Additional parameter>
        Content is currently not evaluated, but must be
        programmed
PS    B/W/D/K/S<Path>
        0: Path of the NC program
CM    9283
PL    B/W/D <Error code>
        0: No error
        1: Call was not in a submit/spawn job
        2: Invalid value in the <Mode> or <Additional parameter>
        parameter or
        3: Invalid value in the <Path> parameter
        4: Internal error
        5: File in <Path> parameter does not exist, or invalid file
        type
        6: File generation not permitted or test run already started
        7: Generation of .DEP not enabled (MP7246, bit 2)
        8: Error during program run
```

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Faulty execution of module
W1022	1	Mode parameter is invalid
	2	NC program does not exist or wrong file type
	6	Generation of .T.DEP files not enabled in the configuration
	8	Control is in an operating mode from which the call is not possible, or test run is already started
	11	An invalid PLC string was programmed
	20	Call was not in a submit/spawn job
	45, 58, 61	Internal error

Module 9286 Select the VNC client and control the focus assignment

PLC Module 9286 is used to program the following status requests regarding the control and assignment of multiple VNC clients:

- Select the active VNC client and simultaneously the assignment of the focus
- Enable or disable focus assignment (mouse operation, touchpad) for the user

Constraints:

- Call only from a submit/spawn job
- The names of the VNC clients (host name or MAC address) can be ascertained with Module 9287

Call:

```
PS    B/W/D/K <Mode>
        0: Define the active VNC client and simultaneously assign
            the focus
        1: Enable or disable focus assignment
PS    B/W/D/K <Name>
        Mode 0: PLC string with host name or IP address
        Mode 1: - - -
PS    B/W/D/K <Timeout [ms]>
        Module rounds up to the next 100 ms limit
        0: Default timeout 3000 ms
PS    B/W/D/K <Additional parameter>
        Mode 0:
        0: Enable focus assignment
        1: Disable focus assignment
        2: Focus assignment unchanged
        Mode 1:
        0: Enable focus assignment
        1: Disable focus assignment
CM    9286
PL    B/W/D <Error code>
        0: No error
        1: PLC error, error code in PLC error word (see below)
        >= 100: Operating-system error
```

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Faulty execution of module
W1022	1	Mode parameter is invalid
	2	Timeout parameter or additional parameter parameter is invalid
	3	PLC string address parameter invalid
	20	Call was not in a submit/spawn job
	45	PLC module cancelled (operating-system error)
	51	NC software does not support module
	61	Function could not be executed
	100	Invalid parameter
	110	Focus assignment unclear, currently no master
	111	Client could not be set as master
	120, 121, 122	Disabling of focus assignment not supported
	130	Unknown status of focus assignment
	131, 132, 133, 134	Interference in communication for focus assignment
200, 201	Internal error	

Module 9287 General status requests regarding VNC clients

PLC Module 9287 is used to program the following status requests regarding control and assignment of multiple operator terminals (IPC, ThinClient):

- Number of VNC clients
- Status of focus assignment
- Name of the active VNC client (master)
- List of VNC clients
- Individual name of a VNC client from the client list

Constraints:

- Call only from a submit/spawn job
- If no name is configured for a client, its MAC address is returned in the PLC strings.
- Mode 3/4: The sequence and numbering of the client list can vary from one boot instance to the next, since this information depends on the point in time at which the clients register themselves in the system.

Call:

PS	B/W/D/K	<Mode>	0: Number of active clients suitable for focus assignment 1: Status of focus assignment 2: Name of the active VNC client 3: List of all clients 4: Name of client n from the client list
PS	B/W/D/K	<Timeout [ms]>	Module rounds up to the next 100 ms limit 0: Default timeout 3000 ms
PS	B/W/D/K	<Additional parameter 1>	Mode 0: - - - Mode 1: - - - Mode 2: Address of PLC string Mode 3: Address of PLC string Mode 4: Address of PLC string
PS	B/W/D/K	<Additional parameter 2>	Mode 0: - - - Mode 1: - - - Mode 2: - - - Mode 3: - - - Mode 4: Number of clients n
CM	9287		
PL	B/W/D	<Error code>	0: No error 1: PLC error, error code in PLC error word (see below) >= 100: Operating-system error
PL	B/W/D	<Status>	Mode 0: Number of clients Mode 1: Focus assignment 0/1: Enabled/disabled Mode 2: - - - Mode 3: - - - Mode 4: - - -

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Faulty execution of module
W1022	1	Mode parameter is invalid
	2	Timeout parameter or additional parameter parameter is invalid
	3	PLC string address parameter invalid
	11	List of clients is empty
	12	Error while evaluating the client list
	20	Call was not in a submit/spawn job
	45	PLC module cancelled (operating-system error)
	51	NC software does not support module
	61	Function could not be executed
	100	Invalid parameter
	110	Focus assignment unclear, currently no master
	111	Client could not be set as master
	120, 121, 122	Disabling of focus assignment not supported
	130	Unknown status of focus assignment
	131, 132, 133, 134	Interference in communication for focus assignment
200, 201	Internal error	

Module 9288 Status requests regarding a VNC client

PLC Module 9288 (call only from within a submit/spawn job) is used to program the following status requests regarding control and assignment of multiple operator terminals (IPC):

- Properties of a client (master / can be master / can be displayed)

Constraints:

- Call only from a submit/spawn job.
- The names of the clients (host name or MAC address) can be ascertained with Module 9287.

Call:

PS	B/W/D/K	<Mode> 0: Properties of the ThinClient
PS	B/W/D/K	<Name> PLC string with name of the client
PS	B/W/D/K	<Timeout [ms]> Module rounds up to the next 100 ms limit 0: Default timeout 3000 ms
PS	B/W/D/K	<Additional parameter 1> Mode 0: - - -
CM	9288	
PL	B/W/D	<Error code> 0: No error 1: PLC error, error code in PLC error word (see below) >= 100: Operating-system error
PL	B/W/D	<Status> Mode 0: Bit 0: 1 = Client is master Bit 1: 1 = Client can be master Bit 2: 1 = Client can be displayed

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Faulty execution of module
W1022	1	Mode parameter is invalid
	2	Timeout parameter or additional parameter parameter is invalid
	3	Parameter name is invalid
	20	Call was not in a submit/spawn job
	45	PLC module cancelled (operating-system error)
	51	NC software does not support module
	61	Function could not be executed
	100	Invalid parameter
	110	Focus assignment unclear, currently no master
	111	Client could not be set as master
	120, 121, 122	Disabling of focus assignment not supported
	130	Unknown status of focus assignment
	131, 132, 133, 134	Interference in communication for focus assignment
200, 201	Internal error	



2 Introduction

2.1 Meaning of the symbols used in this manual



Danger

Failure to comply with this information could result in most serious or fatal injuries, and/or in substantial material damage.



Attention

Failure to comply with this information could result in injuries and interruptions of operation, including material damage.



Note

Tips and tricks for operation as well as important information, for example about standards and regulations as well as for better understanding of the document.

2.2 Proper and intended operation

The described components may only be installed and operated as described in this manual. Commissioning, maintenance, inspection and operation are only to be performed by trained personnel.

HEIDENHAIN contouring controls and their accessories are designed for integration in milling, drilling and boring machines, and machining centers.

2.3 Trained personnel

Trained personnel in the sense of this manual means persons who are familiar with the installation, mounting, commissioning, and operation of the HEIDENHAIN components. Furthermore, electrical engineering work on the system may be carried out only by trained electrical engineering technicians or persons trained specifically for the respective application.

Basically, persons who perform work on HEIDENHAIN components must meet the following requirements:

- They must have been trained or instructed in the standards of safety engineering.
- They must have appropriate safety equipment (clothing, measuring systems).
- They should be skilled in first-aid practice.

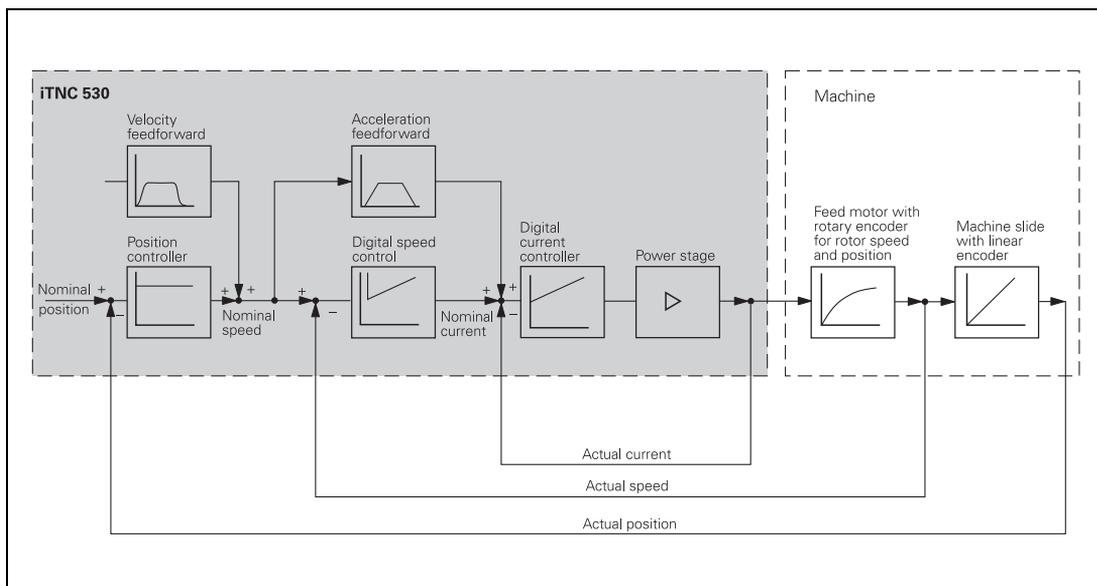
2.4 General information

HEIDENHAIN contouring controls are designed for use with milling, drilling and boring machines as well as machining centers.

The **iTNC 530** features integrated digital drive control and drives the power modules through PWM signals.

Integrating the drive controllers in the iTNC 530 provides the following benefits:

- All the software is contained centrally in the NC; this means that the individual components of the NC, such as feed axes, spindle, NC and PLC, are optimally matched.
- High control quality, because the position controller, speed controller and current controller are combined into one unit.
- The same functions are available for commissioning, optimizing and diagnosing feed drives as well as spindles.



The **iTNC 530** offers digital control for up to 18 axes and 1 spindle at spindle speeds up to $100\,000\text{ min}^{-1}$.

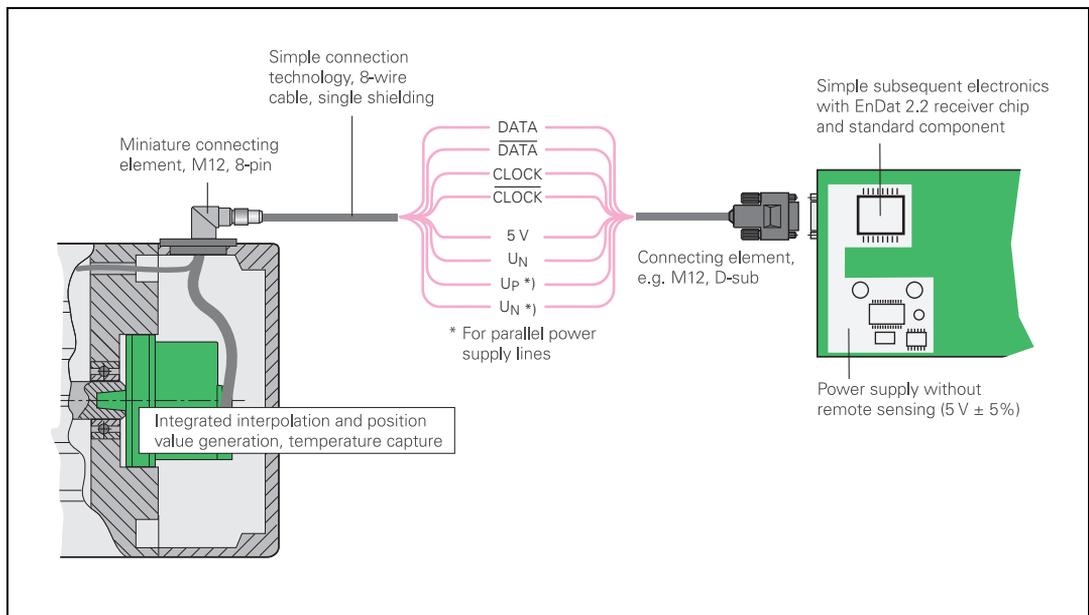
The **iTNC 530** is designed for connection of a compact or modular inverter system. A complete control package, including drives and HEIDENHAIN motors, can be delivered (see the "Inverter Systems and Motors" Technical Manual).

The **MC 6222** or **MC 7222**, the compact main computer of the iTNC 530, takes up very little space, as it is housed in the operating console, directly behind the integrated 15-inch TFT display.

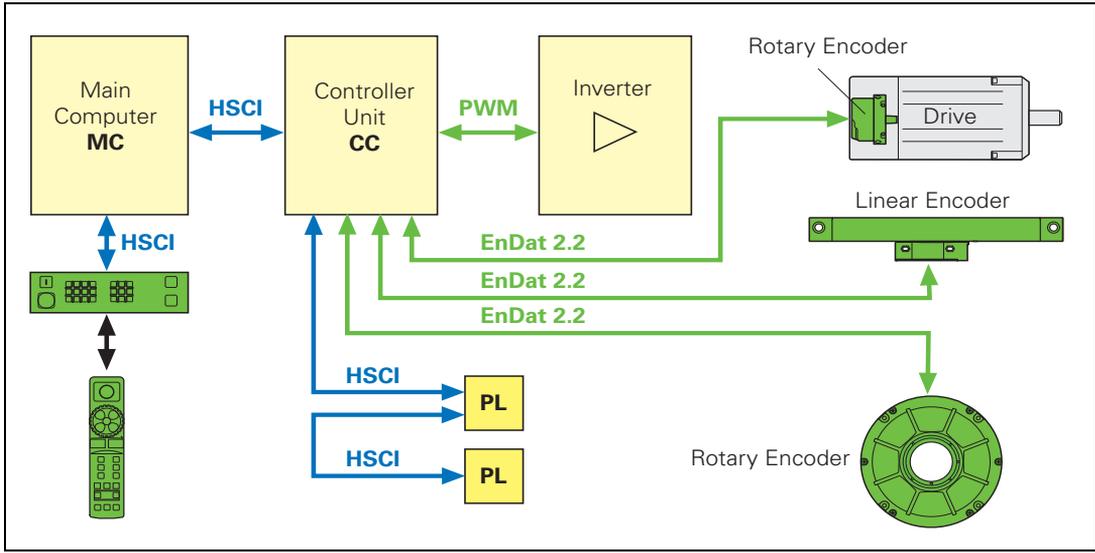
The MC is connected to the CC controller unit, the MB machine operating panel and the PL 6xxx PLC input/output systems via **HSCI** (HEIDENHAIN Serial Controller Interface). The connection of the various control components via HSCI offers numerous benefits, including:

- Simple and uncomplicated wiring
- High noise immunity
- Simple commissioning
- Comprehensive yet straightforward possibilities for diagnostics

The iTNC 530 is prepared for the connection of incremental and absolute position and shaft-speed encoders. EnDat 2.2, which is purely digital and compatible to version 2.1, makes it possible to very rapidly transmit highly resolved position values over long cable lengths. An overview of EnDat 2.2:



Overview of the purely digital control architecture with HSCI and EnDat 2.2:



2.4.1 HSCI interface

The individual control components communicate with each other via the HSCI connection (HEIDENHAIN Serial Controller Interface). A connection via HSCI is only permitted for HEIDENHAIN components that are part of the machine tool's control system. In addition, the HSCI connecting cable may only be installed in a protected manner (e.g. within the electrical cabinet, cable ducts).

The following features characterize the HSCI connection:

- Based on standard 100BaseT Ethernet hardware
- Telegrams of the HSCI connection are not compatible with the Ethernet
- Line structure
- Only one master in the system (MC), all other devices are HSCI slaves

Different addresses are assigned to the individual participants in the HSCI network. The addresses are assigned dynamically during booting of the MC. The HSCI addresses of the participants are formed from an HSCI address (8 bits) and a device type address (6 bits).

After the machine has undergone acceptance testing, the nominal configuration of the control is saved in the IOC file on the control's memory card. This nominal configuration contains the assignment of the device-type address and serial number of the device to the individual HSCI addresses. The momentary configuration is ascertained during startup of the system by requesting the serial numbers. The momentary configuration is compared with the nominal configuration. If there is a deviation, the machine operator is prompted to check the configuration.

The following applies to the assignment of the HSCI address:

- The HSCI address (bus address) is the result of the device's position in the bus
- The master (MC) always has the HSCI address 0.
- The HSCI addresses of the slaves result from their position in the bus:
 - First device after the master (MC): Bus address 1
 - Second device after the master (MC): Bus address 2
 - etc.

The device type address is for internally distinguishing between connected HSCI participants. Each device type (MC, CC, PL, MB, etc.) is assigned a type specification that is used to address all HSCI participants of this type.

More information on the connection of the HSCI components is provided in Chapter 3.

2.5 Overview of components

2.5.1 MC 6xxx main computer, HDR or SDR hard disk and SIK

The iTNC 530 HSCI always includes at least the following components:

- MC 6xxx main computer
(MC = Main Computer)

and either:

- CC 61xx controller unit
(CC = Controller Computer)
- PL 620x system PL
- Modular or compact HEIDENHAIN inverter system

and / or

- UEC 11x controller unit with integrated inverter and PLC

MC 6241 main computer

Main computer for installation in an electrical cabinet:

- Processor:
Pentium M with 1.8 GHz
- 1 GB RAM
- Slot for HDR
- HSCI interface
- 2 x Ethernet interface 100BaseT
- 2 x USB 2.0
- 1 x RS-232C
- Weight: 4 kg

MC 6241 **ID 573398-xx**

MC 6241
with Profibus **ID 653220-xx**



MC 6341 main computer

Main computer for installation in an electrical cabinet:

- Pentium Dual Core with 2.2 GHz
- 2 GB RAM
- Main computer unit
- Electrical cabinet version
- Slot for HDR
- HSCI interface
- 2 x Ethernet interface 100BaseT
- 2 x USB 2.0
- 1 x RS-232C
- Weight: 4 kg

MC 6341 **ID 671226-xx**

MC 6341
with Profibus **ID 735873-xx**



Additionally required for MC 6x41:

- **HDR** hard disk of the iTNC 530 HSCI
 - Contains the NC software
 - Memory capacity 160 GB, of which 138 GB are for the TNC:\ partition and 1 GB for the PLC:\ partition
Remaining memory for system data

HDR iTNC **ID 682272-xx**



MC 6222 main computer

Compact main computer for incorporation in the operating console, with integrated TFT flat-panel display and soft keys.

- Processor:
Pentium M with 1.8 GHz
- 1 GB RAM
- Slot for SSDR
- HSCI interface
- 2 x Ethernet interface 100BaseT
- 3 x USB 2.0
(1 in the operating panel, 2 on the rear)
- 1 x RS-232C
- Weight: 5.8 kg, including SSDR

MC 6222 **ID 634109-xx**

MC 6222
with Profibus **ID 634113-xx**



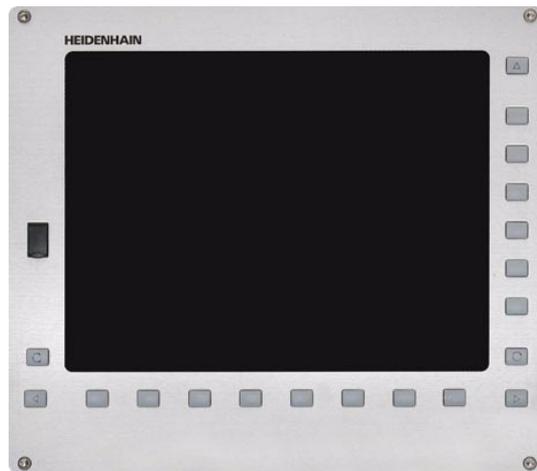
MC 7222 main computer

Compact main computer for incorporation in the operating console, with integrated TFT flat-panel display and soft keys.

- Processor:
Pentium M with 1.8 GHz
- 1 GB RAM
- Slot for SSDR
- HSCI interface
- 2 x Ethernet interface 100BaseT
- 3 x USB 2.0
(1 in the operating panel, 2 on the rear)
- 1 x RS-232C
- Weight: 7.5 kg, including SSDR

MC 7222 **ID 806868-xx**

MC 7222
with Profibus **ID 806869-xx**



MC 6242 main computer

Main computer for integration in the operating panel or electrical cabinet:

- Processor:
Pentium M with 1.8 GHz
- 1 GB RAM
- Slot for SSDR
- HSCI interface
- 2 x Ethernet interface 100BaseT
- 2 x USB 2.0
- 1 x RS232C
- Weight: approx. 4 kg

MC 6242
ID 804780-xx

MC 6242 with Profibus
ID 804786-xx



- **SSDR** hard disk of the iTNC 530 HSCI
 - Contains the NC software
 - Memory capacity 32 GB, of which 21.4 GB are for the TNC:\ partition and 1 GB for the PLC:\ partition; Remaining memory for system data

SSDR iTNC **ID 736591-xx**



Additionally required for MC 6xxx:

- **SIK** (System Identification Key)
 - Contains the NC software license for enabling control loops and software options.
 - The SIK number provides the control with a unique identification.



HDR

HDR hard disk for iTNC 530 with HSCI	ID
NC software 60642x-02	
No export license required, NC software 606641-02	682272-52
Export license required, NC software 606420-02	682272-02
NC software 60642x-03	
No export license required, NC software 606421-03	682272-53
Export license required, NC software 606420-03	682272-03

The following sizes are valid for the partitions on HDR hard disks for the iTNC 530 that were delivered with the NC software 60642x-01:

Partition	Contents	Size
SYS	System files	2 GB (for multiple software versions in packed format)
PLC	OEM files	1 GB
TNC	User files	> 138 GB

SSDR

SSDR solid state disk for iTNC 530 with HSCI	ID
NC software 60642x-03	
No export license required, NC software 606421-02	736591-52
Export license required, NC software 606420-02	736591-02
NC software 60642x-03	
No export license required, NC software 606421-03	736591-53
Export license required, NC software 606420-03	736591-03

The following sizes are valid for the partitions on the SSDR solid state disk for the iTNC 530 that were delivered with the NC software 60642x-01:

Partition	Contents	Size
SYS	System files	2 GB (for multiple software versions in packed format)
PLC	OEM files	1 GB
TNC	User files	> 21 GB



Note

A single NC program can be up to 2 GB large.

**SIK MC 62xx,
MC 63xx**

SIK with NC software license for MC 62xx/MC 63xx without software option		ID (standard)	ID (export)
iTNC 530 HSCI			
	For 4 control loops without software options	586084-20	586084-70
	For 5 control loops without software options	586084-24	586084-74
	For 6 control loops without software options	586084-25	586084-75
	For 7 control loops without software options	586084-26	586084-76
	For 8 control loops without software options	586084-27	586084-77

SIK with NC software license for MC 62xx/MC 63xx with software option 1		ID (standard)	ID (export)
iTNC 530 HSCI			
	For 4 control loops with software option 1	586084-09	586084-59
	For 5 control loops with software option 1	586084-17	586084-67
	For 6 control loops with software option 1	586084-18	586084-68
	For 7 control loops with software option 1	586084-19	586084-69
	For 8 control loops with software option 1	586084-23	586084-73

SIK with NC software license for MC 62xx/MC 63xx with software options 1 and 2		ID (standard)	ID (export)
iTNC 530 HSCI			
	For 4 control loops with software options 1 and 2	586084-01	586084-51
	For 5 control loops with software options 1 and 2	586084-02	586084-52
	For 6 control loops with software options 1 and 2	586084-03	586084-53
	For 7 control loops with software options 1 and 2	586084-04	586084-54
	For 8 control loops with software options 1 and 2	586084-05	586084-55
	For 9 control loops with software options 1 and 2	586084-06	586084-56
	For 10 control loops with software options 1 and 2	586084-07	586084-57
	For 11 control loops with software options 1 and 2	586084-10	586084-60
	For 12 control loops with software options 1 and 2	586084-11	586084-61
	For 13 control loops with software options 1 and 2	586084-12	586084-62
	For 14 control loops with software options 1 and 2	586084-13	586084-63
	For 15 control loops with software options 1 and 2	586084-14	586084-64
	For 16 control loops with software options 1 and 2	586084-15	586084-65

If desired, you can enable the software options 1 and 2 as well as all further software options afterward with code numbers.



SW option: Control loop groups

Further control loops can be enabled either as groups or individually. The combination of control-loop groups and individual control loops makes it possible to enable any number of control loops. Up to 20 control loops are possible.

In addition to the control loops of the basic version, further control loop groups can be enabled through software options. There are control-loop groups with four and with eight control loops:

SW option: Control loop groups:	ID
#77: 4 additional control loops	634613-01
#78: 8 additional control loops	634614-01

In addition to the control loops of the basic version and the control-loop groups, up to eight individual control loops can be enabled through software options. These can also be enabled in addition to the control-loop groups:

SW option: Control loop groups:	ID
#0: 1st additional control loop	354540-01
#1: 2nd additional control loop	353904-01
#2: 3rd additional control loop	353905-01
#3: 4th additional control loop	367867-01
#4: 5th additional control loop	367868-01
#5: 6th additional control loop	370291-01
#6: 7th additional control loop	370292-01
#7: 8th additional control loop	370293-01

The control loops of the CC 61xx can operate selectably as single-speed or double-speed axes. This requires software option 49:

SW option:	ID
#49: Double speed for control loops	632223-01

2.5.2 CC 61xx controller unit

CC 6xxx controller unit

CC 6106 with 6 control loops consists of:

1 drive-control motherboard and
2 drive-control expansion boards

It is equipped with:

- 6 PWM outputs
- 6 speed encoder inputs (1 V_{PP} or EnDat 2.2)
- 6 position encoder inputs (1 V_{PP} or EnDat 2.2)
- 2 SPI expansion slots
- Power supply through UV(R) power supply unit
- Weight: 4.1 kg

CC 6106 ID **662636-xx**



CC 6108 with 8 control loops consists of:
 2 drive-control motherboards and
 2 drive-control expansion boards
 It is equipped with:

- 8 PWM outputs
- 8 speed encoder inputs (1 V_{PP} or EnDat 2.2)
- 8 position encoder inputs (1 V_{PP} or EnDat 2.2)
- 4 SPI expansion slots
- Power supply through UV(R) power supply unit
- Weight: 4.7 kg

CC 6108 ID 662637-xx



CC 6110 with 10 control loops consists of:
 2 drive-control motherboards and
 3 drive-control expansion boards
 It is equipped with:

- 10 PWM outputs
- 10 speed encoder inputs (1 V_{PP} or EnDat 2.2)
- 10 position encoder inputs (1 V_{PP} or EnDat 2.2)
- 4 SPI expansion slots
- Power supply through UV(R) power supply unit
- Weight: 4.8 kg

CC 6110 ID 662638-xx



2.5.3 UEC 11x, UMC 11x controller unit with integrated inverter and PLC

UEC 11x

Controller unit with integrated inverter and PLC for up to 5 control loops for providing line voltage. Compact unit for machines with limited number of axes and low power demands.

It is equipped with:

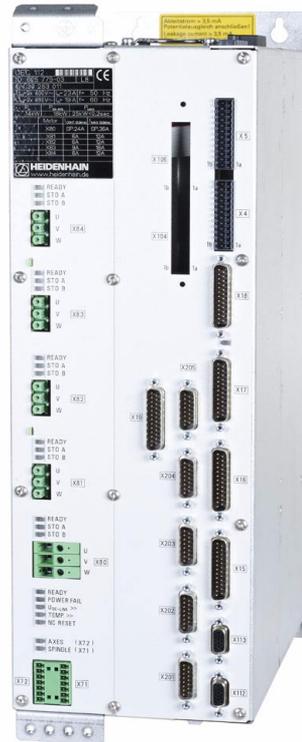
- HSCI interface
- 4 (UEC 111) or 5 (UEC 112) speed encoder inputs
- 4 (UEC 111) or 5 (UEC 112) position encoder inputs
- Connection for 3 axes plus spindle (UEC 111) or
Connection for 4 axes plus spindle (UEC 112)
- Braking resistor
- Connection for TS and TT touch probes if the first PLB is in the HSCI chain
- 38 PLC inputs, 23 PLC outputs (expandable via PL 61xx)
- Integrated power supply unit 24 V NC / 3.5 A for supplying the HSCI components
- Weight: 20 kg

UEC 111 with 4 control loops

ID 625777-xx

UEC 112 with 5 control loops

ID 625779-xx



UMC 11x

Controller unit with integrated inverter and PLC for up to 4 control loops for auxiliary axes and supply with external DC-link voltage. Automatically enables the control loops required for its four auxiliary axes without the need to buy any additional software options.

The maximum number of axes of the control remains unchanged.

It is equipped with:

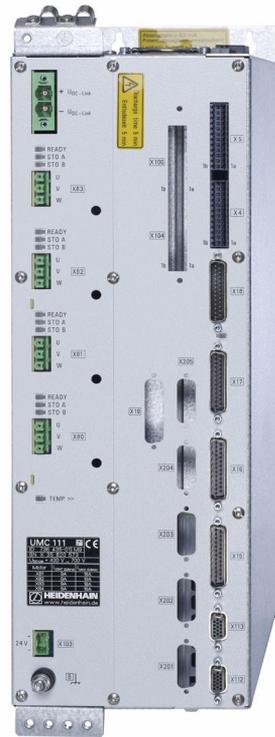
- HSCI interface
- 4 speed encoder inputs
- No position encoder inputs
- Connection for 4 auxiliary axes
- Connection for external DC link
- Connection for TS and TT touch probes if the first PLB is in the HSCI chain
- Braking resistor
- 38 PLC inputs, 23 PLC outputs (expandable via PL 61xx)
- UMC 111 FS: 8 additional FS inputs, 8 FS outputs, 5 PLC outputs
- Weight: 20 kg

UMC 111 with 4 control loops

ID 736435-xx

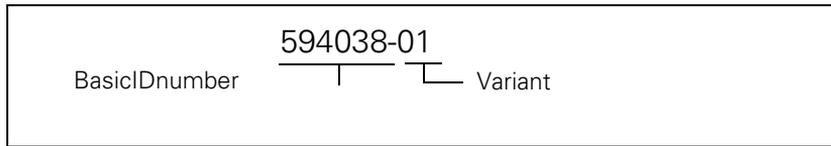
UEC 111FS with 4 control loops

ID 664231-xx



**Designation of the
MC xxxx, CC 61xx
and UMC 11x**

ID of MC 62xx:



The basic ID number indicates hardware differences.
This first digit of the variant number indicates hardware changes.

Variant	Changes to MC 6241
xxx xxx-y1	Prototype
xxx xxx-y2	Prototype
xxx xxx-y3	Initial version

Variant	Changes to MC 6222
xxx xxx-y1	Prototype
xxx xxx-y2	Initial version

Variant	Changes to CC 6106
xxx xxx-y3	Initial version

Variant	Changes to CC 6108
xxx xxx-y3	Initial version

Variant	Changes to CC 6110
xxx xxx-y3	Initial version

Variant	Changes to UEC 111
xxx xxx-y1	Prototype
xxx xxx-y2	Initial version

Variant	Changes to UEC 112
xxx xxx-y1	Prototype
xxx xxx-y2	Initial version



2.5.4 PL 6xxx PLC input/output systems with HSCI interface

The PLC inputs and outputs of the iTNC 530 HSCI are available via the external modular PL 6xxx PLC input/output systems.

The PL 6xxx consists of the PLB 6xxx basic module and one or more I/O modules. The basic modules are connected to the MC main computer via the HSCI interface.



Note

The MC xxxx main computer of the iTNC 530 HSCI does **not** have an integrated PLC, and has **no** connections for TS or TT touch probes. In order to operate the control, at least the PL 62xx system PL (when using a CC 6106) or the UEC 11x controller unit with integrated inverter and PLC is necessary (the system PL is integrated in the UEC).

The PLC inputs/outputs are configured with the PC software IOconfig.

System PL

PL 62xx

System PL, consisting of PLB 620x basic module and I/O modules.

- One module must be in the HSCI system if no UEC 11x is used
- Available with 4, 6 or 8 slots
- HSCI interface
- Connections for TS and TT touch probes
- Safety-relevant PLC inputs/outputs

For an overview of the available I/O modules, see "I/O modules" on page 91.

Mounted on standard NS 35 rails (DIN 46 227 or EN 50 022)

PLB 6204 ID 591832-xx

PLB 6206 ID 630054-xx

PLB 6208 ID 630055-xx

PLB 6204 FS ID 586789-xx

PLB 6206 FS ID 622721-xx

PLB 6208 FS ID 620927-xx



Expansion PL

PL 61xx

Expansion PL, consisting of PLB 620x basic module and I/O modules.

- Available with 4, 6 or 8 slots
- HSCI interface
- Up to 7 PL 61xx can be present in the HSCI system

For an overview of the available I/O modules, see "I/O modules" on page 91.

Mounted on standard NS 35 rails (DIN 46 227 or EN 50 022)

PLB 6104 ID 591828-xx

PLB 6106 ID 630058-xx

PLB 6108 ID 630059-xx

PLB 6104 FS ID 590479-xx



Variant	Changes to PL 62xx
xxx xxx-y1	Prototype
xxx xxx-y2	Prototype
xxx xxx-y3	Initial version

Variant	Changes to PL 61xx
xxx xxx-y1	Prototype
xxx xxx-y2	Prototype
xxx xxx-y3	Initial version

I/O modules

I/O modules are available with digital and analog inputs and outputs. For partially occupied PLB basic modules, the unused slots must be occupied by an empty housing.

PLD-H xx-xx-xx	
<p>Digital I/O module:</p> <ul style="list-style-type: none">■ PLD-H 16-08-00: I/O module with 16 digital inputs and 8 digital outputs■ PLD-H 08-16-00 I/O module with 8 digital inputs and 16 digital outputs■ PLD-H 04-08-00 FS: I/O module with 4 digital dual-channel inputs and 8 digital dual-channel outputs■ PLD-H 08-04-00 FS I/O module with 8 digital dual-channel inputs and 4 digital dual-channel outputs <p>PLD-H 16-08-00 ID 594243-xx PLD-H 08-16-00 ID 650891-xx PLD-H 04-08-00 FS ID 727219-xx PLD-H 08-04-00 FS ID 675572-xx</p>	
PLA-H xx-xx-xx	
<p>Analog I/O module:</p> <ul style="list-style-type: none">■ PLA-H 08-04-04: Analog module with 4 analog inputs for PT 100 thermistors and 8 analog inputs ± 10 V and 4 analog outputs ± 10 V. <p>PLA-H 08-04-04 ID 675572-01-xx</p>	
Empty housing	
<p>...for partial assembly. Empty housing for slots of a PL 6xxx</p> <p>ID 383022-11</p>	

Variant	Changes to PLD-H 16-08-00
xxx xxx-y1	Prototype
xxx xxx-y2	Initial version

Variant	Changes to PLD-H 08-16-00
xxx xxx-y1	Prototype
xxx xxx-y2	Initial version

Variant	Changes to the PLA-H 08-04-04
xxx xxx-y1	Prototype
xxx xxx-y2	Initial version



2.5.5 SPI expansion modules

Expansion module for analog axes/spindles in the HSCI system

Expansion module for analog axes/spindles in the HSCI system:

■ **CMA-H 04-04-00:**

Module for controlling analog axes or spindles. The CMA-H is inserted in one of the SPI slots on the bottom of the controller units. The analog control loop outputs are accessed only via the NC.

As of software version 60642x-02, interpolated movements of analog axes with other axes are usually possible. However, please discuss such applications with your contact person at HEIDENHAIN beforehand.

CMA-H 04-04-00 ID 688721-xx



2.5.6 PSL 13x low-voltage power supply unit

PSL 130

PSL 130 power supply unit for HSCI components with +24 V power supply when using a HEIDENHAIN inverter system.

The **PSL 130** power supply unit was conceived in order to be able to provide the HSCI components of the iTNC 530 with +24 V NC voltage and +24 V PLC voltage.

The output voltages of the **PSL 130** fulfill the requirements for Protective Extra Low Voltage (PELV) according to EN 50178. The power supply unit is powered with line voltage (L1, L2) and the DC-link voltage U_z . This is used to produce the +24 V NC and +24 V PLC output voltages.

Please observe the information and regulations for the power connection cited under "PSL 13x low-voltage power supply unit" on page 237 and in the "Inverter Systems and Motors" Technical Manual.

PSL 130 **ID 575047-xx**



PSL 135

PSL 135 power supply unit for supplying the HSCI components in a multi-row configuration or for using a non-HEIDENHAIN inverter system.

The **PSL 135** power supply unit was conceived in order to be able to provide the HSCI components of the iTNC 530 with +24 V NC voltage, +24 V PLC voltage and +5 V.

The output voltages of the **PSL 135** fulfill the requirements for Protective Extra Low Voltage (PELV) according to EN 50178. The power supply unit is powered with line voltage (L1, L2) and the DC-link voltage U_z . This is used to produce the +24 V NC, +24 V PLC, and +5 V output voltages.

Please observe the information and regulations for the power connection cited under "PSL 13x low-voltage power supply unit" on page 237 and in the "Inverter Systems and Motors" Technical Manual.



PSL 135 ID 627032-xx

Variant	Changes to PSL 130
xxx xxx-y1	Prototype
xxx xxx-y2	Initial version
xxx xxx-y3	Change

Variant	Changes to PSL 135
xxx xxx-y1	Initial version
xxx xxx-y2	Change

2.5.7 TE 6xx keyboard units

TE 620

NC control panel:

- The IV and V keys, the blank key to the left of V and the blank key above MOD in the operating panel are snap-ons.
- USB interface
- Weight: 2.2 kg
- Fulfills IP54 degree of protection when installed

TE 620 ID 625806-xx



TE 630

NC control panel:

Same features as TE 620

- The IV and V keys, the blank key to the left of V and the blank key above MOD in the operating panel are snap-ons.
- USB interface
- Touchpad
- Weight: 2.2 kg
- Fulfills IP54 degree of protection when installed

TE 630 ID 617976-xx



TE 635Q

NC control panel:

Same features as TE 630

Machine operating panel:

- 15-inch design
- 6 axis-direction keys
- 16 function keys
- Keys for NC start and NC stop (illuminated)
- Keys for spindle start and spindle stop
- All keys in the machine operating panel are snap-on keys
- EMERGENCY STOP key
- Key for Control Voltage ON (RAFI key, illuminated)
- Two bore holes (22 mm) for additional RAFI buttons (shipped blocked with a cover) or detachable key switches
- HSCI interface
- Weight: 3.3 kg
- Fulfills IP54 degree of protection when installed

TE 635Q ID 617975-xx



TE 645Q

NC control panel:

Same features as TE 630

Machine operating panel:

- 19-inch design
- 6 axis-direction keys
- 16 function keys
- 3 potentiometers
- USB connection
- Keys for NC start and NC stop (illuminated)
- Keys for spindle start, spindle stop and control voltage ON
- All keys in the machine operating panel are snap-on keys
- EMERGENCY STOP key
- Key for Control Voltage ON (RAFI key, illuminated)
- Four bore holes (22 mm) for additional RAFI buttons or detachable-key switches (shipped blocked with a cover)
- HSCI interface
- Weight: 4.2 kg
- Fulfills IP54 degree of protection when installed

TE 645Q ID 682104-xx



TE 730

NC control panel:

- Suitable for BF 750 (15-inch)
- Keys IV and V can be snap-ons
- Contouring keys
- Operating mode keys
- Spindle-speed and feed-rate override potentiometers
- USB interface
- Touchpad
- Weight: 2.4 kg
- Fulfills IP54 degree of protection when installed

TE 730 ID 805488-xx



TE 735 (FS)

NC control panel:

like TE 730

- Suitable for BF 750 (15-inch)
- Keys IV and V can be snap-ons
- Contouring keys
- Operating mode keys
- Spindle-speed and feed-rate override potentiometers
- USB interface
- HSCI interface
- Touchpad
- Machine operating panel (same as MB720)
- Weight: 3.4 kg
- Fulfills IP54 degree of protection when installed

TE 735 ID 771898-xx

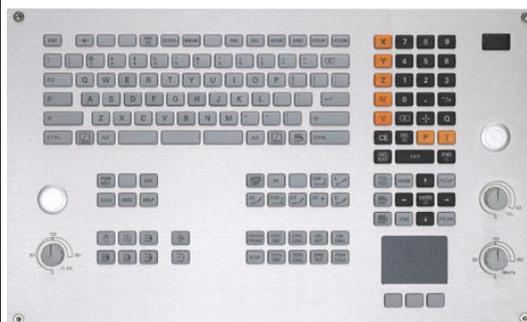
TE 735FS ID 805493-xx



TE 740

NC control panel:

- Suitable for BF 760 (19")
- Keys IV and V can be snap-ons
- Contouring keys
- Operating mode keys
- Spindle-speed, feed-rate and rapid-traverse override potentiometers
- USB interface
- Touchpad
- Weight: 3.2 kg
- Fulfills IP54 degree of protection when installed



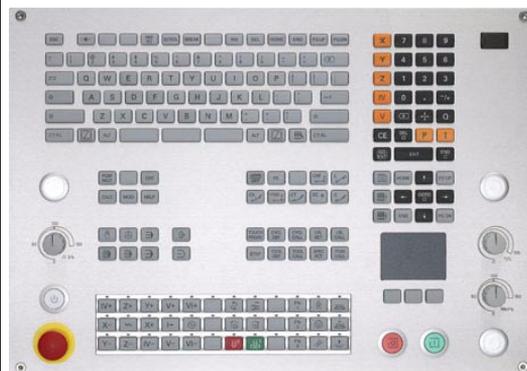
TE 740 ID 805488-xx

TE 745 (FS)

NC control panel:

like TE 740

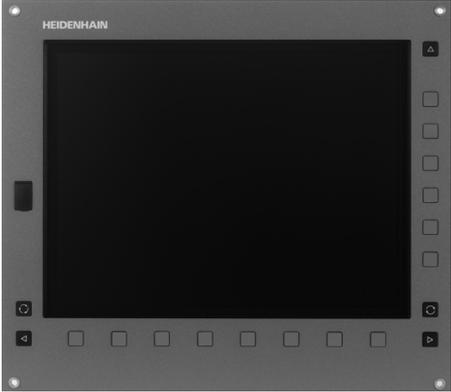
- Suitable for BF 760 (19")
- Keys IV and V can be snap-ons
- Contouring keys
- Operating mode keys
- Spindle-speed, feed-rate and rapid-traverse override potentiometers
- USB interface
- HSCI interface
- Touchpad
- Machine operating panel (same as MB720)
- Power supply 24 V-NC/ approx. 4 W
- HSCI interface
- Handwheel connection X23
- 36 snap-on (exchangeable) keys, see "Key symbols" on page 114. The key functions are freely definable via the PLC
- TE 745: 7 free PLC inputs and 5 free PLC outputs
- TE 745FS: 4 free FS inputs and 5 free PLC outputs
- Weight: 4.2 kg
- Fulfills IP54 degree of protection when installed



TE 745 ID 771898-xx

TE 745FS ID 805493-xx

2.5.8 BF 2xx visual display units

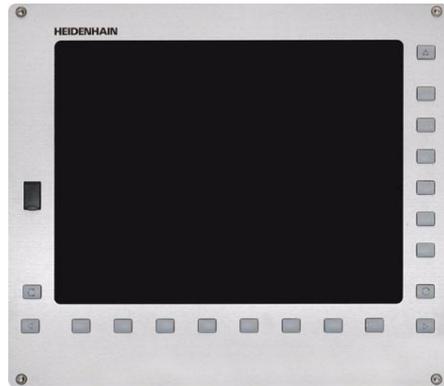
<p>BF 250</p> <p>Technical characteristics:</p> <ul style="list-style-type: none">■ 15-inch design■ Resolution: 1024 x 768 pixels■ 8 horizontal soft keys, 6 vertical soft keys for PLC■ 3 soft keys for switching soft-key rows■ Key for screen layout and operating mode switching■ Additional USB interface (USB 2.0) on the front of the visual display unit■ Integrated USB hub with (USB 2.0) with 4 USB interfaces on the back of the display unit■ HDL connection■ Weight: 5.2 kg■ Fulfills IP54 degree of protection when installed <p>BF 250 ID 599916-xx</p>	 <p>The image shows the Heidenhain BF 250 visual display unit. It features a 15-inch screen with a dark background. The screen is surrounded by a grey bezel. On the left side, there is a vertical column of six square soft keys. On the right side, there is a vertical column of six square soft keys. At the bottom, there is a horizontal row of eight square soft keys. The Heidenhain logo is visible at the top left of the bezel. There are also some navigation buttons (back, forward, home) at the bottom corners.</p>
<p>BF 260</p> <p>Technical characteristics:</p> <ul style="list-style-type: none">■ 19-inch design■ Resolution: 1280 x 1024 pixels■ 1 x 10 horizontal NC soft keys■ 1 x 8 soft keys for PLC to the right and 1 x 10 soft keys to the left of the screen (vertical)■ 3 soft keys for switching soft-key rows■ Key for screen layout and operating mode switching■ Integrated USB hub with 6 USB interfaces on the back of the display unit■ HDL connection■ Weight: 7.8 kg■ Fulfills IP54 degree of protection when installed <p>BF 260 ID 617978-xx</p>	 <p>The image shows the Heidenhain BF 260 visual display unit. It features a 19-inch screen with a dark background. The screen is surrounded by a grey bezel. On the left side, there is a vertical column of ten square soft keys. On the right side, there is a vertical column of eight square soft keys. At the bottom, there is a horizontal row of ten square soft keys. The Heidenhain logo is visible at the top left of the bezel. There are also some navigation buttons (back, forward, home) at the bottom corners.</p>

BF 750

Technical characteristics:

- 15-inch design
- Power supply 24 V-NC/ approx. 50 W
- Resolution: 1024 x 768 pixels
- 1 x 8 soft keys, 1 x 6 soft keys for PLC
- 3 soft keys for switching soft-key rows
- Key for screen layout and operating mode switching
- Additional USB interface (USB 2.0) on the front of the visual display unit
- Integrated USB hub with (USB 2.0) with 4 USB interfaces on the back of the display unit
- HDL connection
- Weight: 4.0 kg
- Fulfills IP54 degree of protection when installed

BF 750 ID 785080-xx



BF 760

Technical characteristics:

- 19-inch design
- Power supply 24 V-NC/ approx. 65 W
- Resolution: 1280 x 1024 pixels
- 1 x 10 horizontal NC soft keys
- 1 x 8 soft keys for PLC to the right and 1 x 10 soft keys to the left of the screen (vertical)
- 3 soft keys for switching soft-key rows
- Key for screen layout and operating mode switching
- Integrated USB hub with 6 USB interfaces on the back of the display unit
- HDL connection
- Weight: 7.8 kg
- Fulfills IP54 degree of protection when installed

BF 760 ID 732589-xx



2.5.9 MB 6xx machine operating panel

MB 620

The MB 620 is equipped with:

- HSCI interface
- Handwheel connection X23
- 21 snap-on (exchangeable) keys, see "Key symbols" on page 114. The key functions are freely definable via the PLC
- 8 PLC inputs and 8 PLC outputs
- Two bore holes for additional keys or keylock switches
- 15-inch design
- Weight: 1.2 kg
- Fulfills IP54 degree of protection when installed

Controls and displays:

- 12 axis keys
- 16 function keys
- NC start¹
- NC stop¹
- EMERGENCY STOP key
- Control Voltage ON¹

¹) Keys illuminated



MB 620 ID 617973-xx

MB 720

- 15-inch design, suitable for BF 750
- Power supply 24 V-NC/ approx. 4 W
- HSCI interface
- Handwheel connection X23
- 36 snap-on (exchangeable) keys, see "Key symbols" on page 114. The key functions are freely definable via the PLC
- MB 720: 7 free PLC inputs and 5 free PLC outputs
- MB 720FS: 4 free FS inputs and 5 free PLC outputs
- Two bore holes for additional keys or keylock switches
- Weight: 1.2 kg
- Fulfills IP54 degree of protection when installed

Controls and displays:

- 12 axis keys
- 16 function keys
- 6 function keys that are not assigned
- NC start¹
- NC stop¹
- EMERGENCY STOP key
- Control Voltage ON¹

¹⁾ Keys illuminated

MB 720 **ID 784803-xx**

MB 720FS **ID 805474-xx**



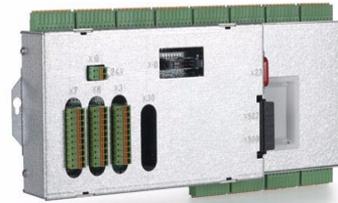
2.5.10 HSCI adapter for PLB 6001 OEM-specific machine operating panel

PLB 6001

The PLB 6001 is equipped with:

- HSCI interface
- Handwheel connection X23
- 64 PLC inputs, 32 PLC outputs for keys / key illumination
- Connection for spindle-speed and feed-rate override potentiometers
- Screw fastening or top-hat-rail mounting
- Weight: 1.2 kg

PLB 6001 **ID 668792-xx**



2.5.11 HR 4xx(FS), HR 5xx(FS) handwheels

All handwheels are available with and without detent. For handwheels with detent, the cogging torque prevents movements of the handwheel due to motions or vibrations of the machine. On handwheels without detent, this is prevented by a defined holding torque.

Handwheels with detent feature 100 detent positions per revolution, i.e. every 3.6°. The increment is set as follows:

- HR 130: By the user via conversational programming on the control
- HR 410(FS): By the user on the HR 410 via 3 different step increments predefined by the machine manufacturer
- HR 520(FS): By the user on the HR 520

The HR 520FS and HR 410FS handwheels that are available in addition to the HR 550 (wireless handwheel system) are devices that, due to their cross-circuit-safety, can also be integrated in safety-related applications according to EN 13849. The HR 410 and HR 520 standard handwheels do not feature this cross-circuit safety of the permissive buttons.

The handwheels with FS are plug-compatible with the respective handwheels without FS, thanks to the M23 connecting element, and they have the same scope of functions. As a result of the new logic behind the permissive buttons, only one of the two buttons needs to be pressed during permissive mode.



Note

- Due to the changed logic of the permissive buttons, do not replace the HR 520FS and HR 410FS with the respective variant without FS.
- Please note that even if servicing is required, an HR 520FS may not be replaced by an HR 520. This also applies to replacing an HR 410FS with an HR 410, and vice versa.

Because of the change in the principle behind the permissive buttons, we generally recommend using the safety-oriented handwheels with FS, due to the cross-circuit-safety of their permissive function.

Only for the HR 550FS wireless handwheel system with HRA 551FS can the behavior of the permissive buttons be defined by MP7645.3.

Also, a new dummy plug is being introduced for FS handwheels in case servicing is necessary. As before, this dummy plug bridges the emergency stop, and is adapted to the changed logic for the permissive buttons. The new dummy plug for FS handwheels is identified by the designation "Adapterstecker FS" and the ID 271958-05 on the ID label.





Note

- Please note that the dummy plug for FS handwheels may not be exchanged for the respective variant without FS.
- For reasons of technical safety, only dummy plugs for FS handwheels (ID 271958-05) may be used on systems that have an FS handwheel.
- For reasons of technical safety, only dummy plugs for standard handwheels (ID 271958-03) may be used on systems that have a standard handwheel (without FS).



HR 410 handwheel

Portable electronic handwheel with snap-on (exchangeable) keys, (see "Key symbols" on page 114).

- Five axis selection keys
- Keys for traverse direction
- Keys for preset feeds
- Actual-position-capture key
- Three keys for machine functions (definable via PLC)
 - Spindle right/left/stop
 - NC start/stop, spindle start; for HEIDENHAIN basic PLC program
- Two permissive buttons (24 V)
- Emergency stop button (24 V)
- Magnetic holding pads

ID 296469-54 HR 410 handwheel (spindle right/left/stop)

ID 296469-55 HR 410 handwheel (NC start/stop, spindle start)

ID 535220-05 HR 410 handwheel (NC start/stop, spindle start) with detent



HR 410FS handwheel

- Five axis selection keys
- Keys for traverse direction
- Keys for preset feeds
- Actual-position-capture key
- Three keys for machine functions (definable with PLC)
- Two cross-circuit-proof permissive buttons
- Integrated emergency stop button
- Magnetic holding pads
- Bottom row of keys: spindle start, NC start, NC stop
- Additional snap-on keys for spindle stop, spindle left, and spindle right are included with the handwheel
- Weight: Approx. 1 kg

ID 337159-21 HR 410FS handwheel

ID 578114-11 HR 410FS handwheel with detent



HR 520 handwheel

- Display for operating mode, actual position value, programmed feed rate and spindle speed, error messages
- Graphic display, resolution: 128 x 64 pixels, 6-line display
- Spindle speed and feed-rate override
- Selection of axes via keys or soft keys
- Actual position capture
- NC start/stop, spindle start/stop
- 6 freely programmable PLC keys with LED
- Keys for traverse direction
- Exchangeable snap-on keys for PLC functions and maintenance
- Integrated permissive buttons and emergency stop button (24 V)
- Magnetic holding pads
- Mount for attaching the handwheel to the machine (**ID 591065-xx**)
- Weight: Approx. 1 kg

ID 670302-xx HR 520 handwheel

ID 670303-xx HR 520 handwheel with detent



HR 520FS handwheel

- Graphic display, resolution: 128 x 64 pixels, 6-line display
- Six NC keys
- Six PLC keys with LEDs that are controlled through the PLC
- Two override potentiometers (feed rate and spindle speed)
- Two cross-circuit-proof permissive buttons
- Exchangeable snap-on keys for PLC functions and maintenance
- Integrated emergency stop button
- Magnetic holding pads
- Additional snap-on keys for spindle stop, spindle left, and spindle right are included with the handwheel
- Weight: Approx. 1 kg

ID 670304-xx HR 520FS handwheel

ID 670305-xx HR 520FS handwheel with detent



HR 5xx mount

Optional handwheel mount for attaching to the operating panel or electrical cabinet

ID 591065-xx



HR 550FS handwheel

Portable electronic handwheel with wireless transmission:

- Graphic display, resolution: 128 x 64 pixels, 6-line display
- Six NC keys
- Six PLC keys with LEDs that are controlled through the PLC
- Two override potentiometers (feed rate and spindle speed)
- Two permissive buttons
- Exchangeable snap-on keys for PLC functions and maintenance
- Integrated emergency stop button
- Vibration alarm when leaving the radio range
- Battery warning
- Display of radio field strength
- Weight: Approx. 1 kg

Without mechanical detent:

ID 598515-xx

With mechanical detent:

ID 606622-xx

Accessories (1 x included with the HR 550FS):

Handwheel batteries
(battery pack, 4-tray)

ID 623166-xx



HR 551FS handwheel adapter
 Handwheel adapter (docking tray) with integrated charger for handwheel batteries
ID 731928-xx



ID 312879-01 Connecting cable to cable adapter (spiral cable 3 m)
 ID 296467-xx Connecting cable to cable adapter (normal cable)
 ID 296687-xx Connecting cable to cable adapter (with metal armor)
 ID 296466-xx Adapter cable to MB 6xx
 ID 281429-xx Extension to adapter cable

Dummy plug for handwheels:

- Bridge for emergency stop
- Dummy plug for FS handwheels: No permission for cross-circuit safety logic of permissive buttons
- Dummy plug for standard handwheels: No permission for cross-circuit safety logic of permissive buttons

Dummy plug for handwheels with FS
ID 271958-05

Dummy plug for standard handwheels
ID 271958-03



HR 130 handwheel

Panel-mounted handwheel

ID 254040-05 With ergonomic knob, radial cable outlet

ID 540940-01 Same as above, but with detent



HRA 110 handwheel adapter

For connecting up to three **HR 150** handwheels to the TNC.

The axes and the subdivision factor are selected via rotary switch.

The most important characteristics of the HR 150 are:

- 11 μA_{PP} signals, line count 5000
- 1 m cable with 9-pin connector (male)
- Cable outlet for axial and radial use
- Ergonomic knurled control knob
- Version with mechanical detent, 100 positions per 360°

ID 261097-03	HRA 110
ID 540940-06	HR 150 with mechanical detent
ID 540940-07	HR 150 without mechanical detent
ID 270908-01	HRA selector switch



2.5.12 Industrial PC with Windows 7

<p>Industrial PC IPC 6341</p> <p>Main computer for installation in an electrical cabinet:</p> <ul style="list-style-type: none">■ Pentium Dual Core with 2.2 GHz■ 2 GB RAM■ Main computer unit■ Electrical cabinet version■ 2 x Ethernet interface 100BaseT■ 2 x USB 2.0■ 1 x RS-232C■ Weight: 4 kg <p>IPC 6341 ID 749963-xx</p>	 <p>A vertical, rack-mountable industrial PC unit with a silver and black finish. The front panel features a variety of ports including two Ethernet ports, two USB ports, and an RS-232C port. The unit is designed for installation in an electrical cabinet.</p>
<p>Additionally required for IPC 6341:</p> <ul style="list-style-type: none">■ Hard disk HDR with Windows 7 embedded<ul style="list-style-type: none">• Memory capacity 160 GB with approx. 140 GB free capacity for the user's own applications <p>HDR IPC ID 599956-xx</p>	 <p>A hard disk drive (HDD) mounted in a black metal cage. The drive is a 3.5-inch format and is connected to a green printed circuit board (PCB) via a ribbon cable. The cage has a front panel with a handle and a lock mechanism.</p>

2.5.13 Key symbols

Key symbols for the spindle

Key	Description Print/Background ID	Key	Description Print/Background ID
	Spindle stop White/Red 330816-08		Spindle start White/Green 330816-09
	Spindle direction left Black/Gray 330816-40		Spindle direction right Black/Gray 330816-41
	Spindle stop White/Red 330816-47		Spindle start White/Green 330816-46
	Clamp the spindle Black/Gray 330816-48		Spindle on Black/Gray 330816-1N

Key symbols with axis designations

Key	Description Print/Background ID	Key	Description Print/Background ID
	X Black/Orange 330816-24		Y Black/Orange 330816-36
	Z Black/Orange 330816-25		A Black/Orange 330816-42
	B Black/Orange 330816-26		C Black/Orange 330816-23
	U Black/Orange 330816-43		V Black/Orange 330816-38
	W Black/Orange 330816-45		IV Black/Orange 330816-37
	VI Black/Orange 330816-3D		Without symbol Black/Orange 330816-3P
	VT Black/Orange 330816-4U		

**Key symbols for
axis direction keys
for the
principal axes**

Key	Description Print/Background ID	Key	Description Print/Background ID
	X- Black/Gray 330816-63		X+ Black/Gray 330816-64
	X- <- Black/Gray 330816-18		X+ -> Black/Gray 330816-17
	X'- -> Black/Gray 330816-0W		X'+ <- Black/Gray 330816-0V
	X- <- Black/Gray 330816-0N		X+ -> Black/Gray 330816-0M
	X'- Black/Gray 330816-3C		X'+ Black/Gray 330816-3B
	Y- Black/Gray 330816-67		Y+ Black/Gray 330816-68
	Y'- -> Black/Gray 330816-21		Y'+ <- Black/Gray 330816-20
	Y- <- Black/Gray 330816-0P		Y+ -> Black/Gray 330816-0R
	Y- -> Black/Gray 330816-0D		Y+ <- Black/Gray 330816-0E
	Z- Black/Gray 330816-65		Z+ Black/Gray 330816-66
	Z- <- Black/Gray 330816-19		Z+ -> Black/Gray 330816-16
	Z'- -> Black/Gray 330816-0L		Z'+ <- Black/Gray 330816-0K

Key symbols for axis direction keys for rotary and additional linear axes

Key	Description Print/Background ID	Key	Description Print/Background ID
	A- Black/Gray 330816-95		A+ Black/Gray 330816-96
	B- Black/Gray 330816-97		B+ Black/Gray 330816-98
	C- Black/Gray 330816-99		C+ Black/Gray 330816-0A
	U- Black/Gray 330816-0B		U+ Black/Gray 330816-0C
	V- Black/Gray 330816-70		V+ Black/Gray 330816-69
	W- Black/Gray 330816-0G		W+ Black/Gray 330816-0H
	W- arrow Black/Gray 330816-1T		W+ arrow Black/Gray 330816-1S
	IV- Black/Gray 330816-71		IV+ Black/Gray 330816-72
	VI- Black/Gray 330816-1Z		VI+ Black/Gray 330816-2A

**Key symbols
for machine
functions**

Key	Description Print/Background ID	Key	Description Print/Background ID
	Special function Black/Gray 330816-0X		Function A White/Black 330816-30
	Function B White/Black 330816-31		Function C White/Black 330816-32
	Function 1 Black/Gray 330816-73		Function 2 Black/Gray 330816-74
	Function 3 Black/Gray 330816-75		Function 4 Black/Gray 330816-76
	Function 5 Black/Gray 330816-77		Unlock door Black/Gray 330816-78
	Unlock door Black/Gray 330816-79		Coolant Black/Gray 330816-80
	Coolant (internal) Black/Gray 330816-0S		Coolant (external) Black/Gray 330816-0T
	Coolant on Green/White 330816-1L		Coolant off Light red/White 330816-1M
	Rinse water jet Black/Gray 330816-81		Spotlight Black/Gray 330816-82
	Chip removal Black/Gray 330816-83		Chip conveyor Black/Gray 330816-84
	Tool change Black/Gray 330816-89		Tool changer left Black/Gray 330816-85
	Tool changer right Black/Gray 330816-86		Unclamp tool Black/Gray 330816-87
	Unclamp tool Black/Gray 330816-88		Clamp tool Black/Gray 330816-94
	Clamp tool Black/Gray 330816-0U		Retract axis Black/Gray 330816-91
	XY clamping Black/Gray 330816-0Z		Open the magazine door Black/Gray 330816-3L

Key	Description Print/Background ID	Key	Description Print/Background ID
	Clamp the fixture Black/Gray 330816-3M		Unclamp the fixture Black/Gray 330816-3N
	Machine illumination Black/Gray 330816-4V		Spindle switch-over Black/Gray 330816-5F
	Jog spindle to right Black/Gray 330816-5G		Jog spindle to left Black/Gray 330816-5H
	Spindle to percentage Black/Gray 330816-5K		Spindle to percentage Black/Gray 330816-5L
	Spindle to 100 percent Black/Gray 330816-5M		Clamp/unclamp pallet Black/Gray 330816-5V

Other key symbols

Key	Description Print/Background ID	Key	Description Print/Background ID
	No symbol -/Black 330816-01		No symbol -/Gray 330816-61
	NC start White/Green 330816-11		NC stop White/Red 330816-12
	NC start White/Green 330816-49		NC stop White/Red 330816-50
	Feed rate 1 Black/Gray 330816-33		Feed rate 2 Black/Gray 330816-34
	Rapid traverse Black/Gray 330816-13		Permissive button White/Green 330816-22
	Permissive button Black/Gray 330816-90		Actual position capture White/Black 330816-27
	- White/Black 330816-28		+ White/Black 330816-29
	Menu selection → Black/Gray 330816-92		Menu selection ← Black/Gray 330816-93
	0 Black/Gray 330816-0Y		Connect White/Black 330816-3A



2.5.14 Touch probes

Touch probes are connected via the systemPL62xx or the UEC11x. The touch probes generate a trigger signal that captures the current position value. For more information about touch probes, please request the "Touch Probes" brochure or CD-ROM from HEIDENHAIN.

The TS touch trigger probe has a stylus with which it probes workpieces. The iTNC 530 provides standard routines for datum setting and workpiece measurement and alignment. The touch probes are available with various taper shanks. Assorted styli are available as accessories.

TS 220 touch probe

Triggering touch probe with signal transmission over cable connection for machines with manual tool change. For workpiece setup and measurement during machining.

ID 293488-xx

TS 220

ID 633613-xx

Adapter cable for connection to the system PL or the UEC



TS 740, TS 640, TS 642, TS 444, TS 440 touch probes

Triggering touch probe with infrared transmission, for workpiece setup and measurement during machining. For machines with automatic tool changer.

- TS 440 with compact dimensions
- TS 444 with alternative battery-free power supply via compressed air through the spindle head
- TS 640 with wide-range infrared transmission and long operating time
- TS 642 with mechanical activation switch in the clamping shank
- TS 740 with high probing accuracy and repeatability, and low probing forces

The infrared transmission is established between the TS touch probe and the SE transceiver unit. The following SE units can be combined with the TS touch probes:

- SE 640 for integration in the machine workspace
- SE 642 for mounting in the workspace, for operation of a TS 44x, TS 64x, TS 74x infrared workpiece touch probe and the TT 449 infrared tool touch probe
- SE 540 for integration in the spindle head

ID 573757-xx	TS 740
ID 620189-xx	TS 640
ID 653217-xx	TS 642
ID 620046-xx	TS 440
ID 588008-xx	TS 444

ID 631225-xx SE 640 transmitter-receiver unit

ID 652792-xx SE 642 transmitter-receiver unit

ID 626001-xx SE 540 transmitter-receiver unit



TS 640, TS 740



TS 440, TS 444



SE 640



SE 642

Tool measurement

TT 140 tool touch probe

Touch trigger probe with rated break point of the connection pin for the probe head and optical deflection display. An additional connection pin is delivered with the touch probe.

ID 527797-03 TT 140

ID 559758-01 Connection pin

ID 633616-xx Adapter cable for connection to the PL 62xx



TT 449 tool touch probe

Touch trigger probe with rated break point of the connection pin for the probe head and optical deflection display. An additional connection pin is delivered with the touch probe. The TT 449 is wireless and communicates over infrared beam with the SE 642.

ID 593007-xx TT 449

ID 559758-01 Connection pin



2.5.15 MS 110 / MS 111 installation kit

MS 110 without additional 24 V supply

Sometimes limited space prevents the control and inverter system from being mounted in the same row in a machine's electrical cabinet, meaning that they must be mounted in two separate rows. In other cases the design calls for a second electrical cabinet to house the inverter system. This means that the distribution and arrangement of the components can be very different from case to case.

In order to establish an electrical connection (immune to noise) between the components of the inverter system, the MS 1xx installation kits are needed.

In most cases the additional 24 V are not needed, since the UVR 1xxD provides enough current for the fans and the MS 110 is sufficient.

ID 658132-xx MS 110



MS 111 with 24 V supply

For inverter systems with many powerful UM 1xxD power modules, it might be the case that the current provided by the UV(R) 1xx power supply unit for the fans of the UMs does not suffice to guarantee safe and reliable operation of the fans.

If it is exceeded, then an MS 111 must be used in the inverter row where the current consumption is very high.

ID 673685-xx MS 111



Variant	Changes to MS 11x
xxx xxx-01	Initial version

2.5.16 Other accessories

Further components	ID
Adapters for encoder signals	
TTL (HEIDENHAIN layout)/1 V _{PP}	317505-01
TTL (SIEMENS layout)/1 V _{PP}	317505-02

CML 110 capacitor module for 24 V power supply

Specifications

- Supply voltage: 24 V
- Capacitance: 8.3 F
- Max. charging current: 2.4 A
- Internal resistance (discharge)
 - Maximum: 156 milliohm
 - Typically: 65 milliohm
- Discharge current: 30 A
- A CML 110 is usually not required when using a PSL 130 because the PSL 130 supplies +24 V buffered in the DC link.



ID 574087-02

CML 110

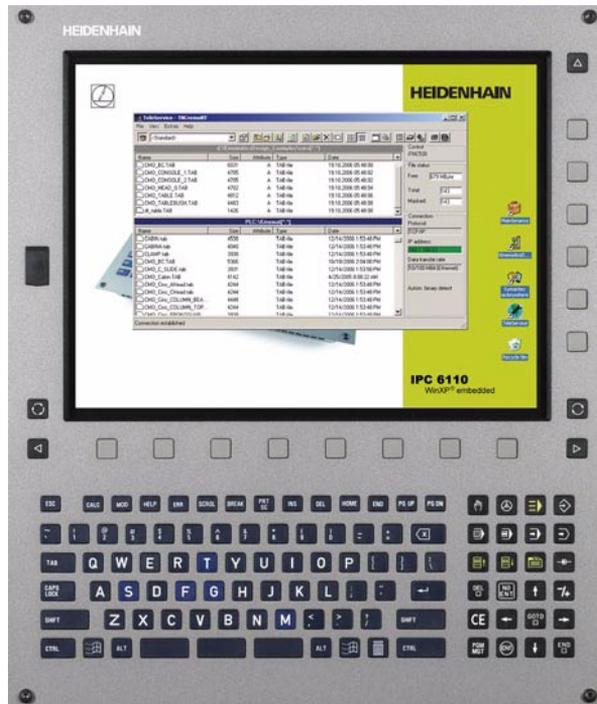
IPC 6110 main computer

ID 597968-02

- Main computer
- Integrated ASCII keyboard with additional keypad
- 15-inch screen with soft keys
- 3 USB connections
- Slot for CompactFlash memory card, types I and II
- Ethernet connection
- RS-232-C/V.24 data interface
- Power supply connection

Controls on the IPC 6110

- English ASCII keyboard with function keys
- 24-key keyboard block with snap-on keys
- 8 horizontal soft keys
- 6 vertical soft keys
- 2 keys for switching the horizontal soft-key row
- Key for switching the vertical soft-key row
- Key for view change
- Key for changing screen layout



IPC 6120 package

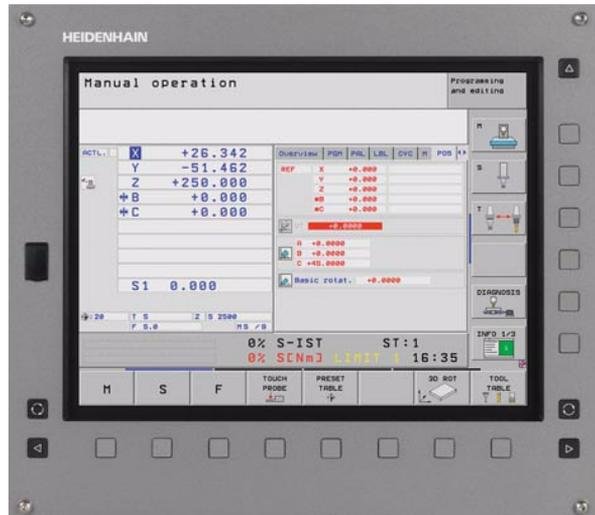
ID 664940-01 "IPC 6120 adv. package" (with TE 630 and CFR)

ID 664939-01 "IPC 6120 basic package" (with TE 620 and CFR)

- Main computer
- Separate TNC operating panel TE 630 or TE 620
- 15-inch screen
- 3 USB connections
- Slot for CompactFlash memory card, types I and II
- Ethernet connection
- RS-232-C/V.24 data interface
- Power supply connection
- CFR TNCterm

Controls on the IPC 6120 (can be used only as a package with TE 6xx)

- Complete TNC operating panel with programming keys
- Two override potentiometers (can be accessed for switching with analog inputs, not included in delivery)
- Mouse pad with keys (only "IPC 6120 adv. package")
- 8 horizontal soft keys
- 6 vertical soft keys
- 2 keys for switching the horizontal soft-key row
- Key for switching the vertical soft-key row
- Key for view change
- Key for changing screen layout



2.5.17 Documentation

HEIDENHAIN offers the following User's Manuals for the iTNC 530:

User's Manual	ID
HEIDENHAIN conversational programming	679351-xx
ISO Programming	679355-xx
Cycle Programming	679295-xx

The HEIDENHAIN inverters and motors for the iTNC 530 are described in the "Inverter Systems and Motors" Technical Manual.

The components required for operating the iTNC 530 with non-HEIDENHAIN inverter systems are described in "Technical Information for Operation of SIMODRIVE and POWER DRIVE Inverter Systems."



2.6 Brief description

Specifications	iTNC 530 HSCI
MC 6241	<p>Main computer unit for installation in an electrical cabinet</p> <ul style="list-style-type: none"> ■ Processor (Pentium M 1.8 GHz) ■ 1 GB SDRAM main memory ■ Bus frequency 400 MHz ■ HSCI interface ■ Unique identification of MC 6241 through System Identification Key (SIK)
MC 6222, MC 7222	<p>Main computer unit with integrated 15-inch TFT flat panel display for installation in the operating panel</p> <ul style="list-style-type: none"> ■ Processor (Pentium M 1.8 GHz) ■ 1 GB SDRAM main memory ■ Bus frequency 400 MHz ■ HSCI interface ■ Unique identification of MC 6222, MC 7222 through System Identification Key (SIK)
MC 6341	<p>Main computer unit for installation in an electrical cabinet</p> <ul style="list-style-type: none"> ■ 2.2 GHz dual core processor ■ 2 GB SDRAM main memory ■ 1066 MHz bus frequency ■ HSCI interface ■ Unique identification of MC 6341 through System Identification Key (SIK)
CC 6106	<p>All position and speed encoder inputs 1 V_{PP} or EnDat</p> <ul style="list-style-type: none"> ■ HSCI interface ■ Max. 6 digital control loops ■ 6 position and 6 speed encoder inputs with 1 V_{PP} or EnDat 2.2 for axes and spindle (EnDat 2.2 is backward-compatible to EnDat 2.1) ■ 6 PWM outputs ■ Power supply via UV(R), UE or UR power supply unit

Specifications	iTNC 530 HSCI
CC 6108	<p>All position and speed encoder inputs 1 V_{PP} or EnDat</p> <ul style="list-style-type: none"> ■ HSCI interface ■ Max. 8 digital control loops ■ 8 position and 8 speed encoder inputs with 1 V_{PP} or EnDat 2.2 for axes and spindle (EnDat 2.2 is backward-compatible to EnDat 2.1) ■ 8 PWM outputs ■ Power supply via UV(R), UE or UR power supply unit
CC 6110	<p>All position and speed encoder inputs 1 V_{PP} or EnDat</p> <ul style="list-style-type: none"> ■ HSCI interface ■ Max. 10 digital control loops ■ 10 position and 10 speed encoder inputs with 1 V_{PP} or EnDat 2.2 for axes and spindle (EnDat 2.2 is backward-compatible to EnDat 2.1) ■ 10 PWM outputs ■ Power supply via UV(R), UE or UR power supply unit
UEC 11x	<p>Controller unit with integrated inverter and PLC, for machines with low power demands</p> <ul style="list-style-type: none"> ■ HSCI interface ■ Controller unit with position, speed and current controller ■ UEC 111: Up to 4 digital control loops, connection for 3 axes and spindle ■ UEC 112: Up to 5 digital control loops, connection for 4 axes and spindle ■ Integrated braking resistor ■ 38 PLC inputs, 23 PLC outputs (expandable via PL 61xx) ■ Interfaces to the speed encoders ■ Interfaces to the position encoders ■ Interfaces for one TS and TT touch probe each ■ +24 V NC power supply with 2.5 A for MC and other control components



Specifications		iTNC 530 HSCI
Axis feedback control		
		Velocity feedforward control / Operation with following error
		Connection of the CC or UEC controller unit for digital axis control over HSCI
		Analog axis control over SPI expansion module
Cycle time for path interpolation		3 ms
Options		
		Additional control loops or software options can be enabled by entering a code number.
Display		
		15- or 19-inch TFT color flat-panel display
Program memory		
		> 138 GB with HDR hard disk
		> 21 GB solid state disk (SSD)
Input resolution and display step		
Linear axes		Linear axes down to 0.1 μm Angular axes down to 0.0001°
Block processing time		
		0.5 ms
Real-time operating system		
		HeROS 5
Interpolation		
Linear		5 of 18 axes 4 of 18 axes ^a
Circle		2 of 18 axes 3 of 18 axes with tilted working plane
Helix		Superimpositioning of circular and straight paths
Spline		Cubic splines can be executed

Specifications	iTNC 530 HSCI														
Regulation with CC 6xxx / UEC 1xx															
Position control resolution	<table border="0"> <tr> <td style="text-align: center; border-bottom: 1px solid black;">Signal period</td> <td rowspan="2" style="padding-left: 20px;">or encoder resolution (EnDat 2.2 interpol.)</td> </tr> <tr> <td style="text-align: center;">4096</td> </tr> </table>	Signal period	or encoder resolution (EnDat 2.2 interpol.)	4096											
Signal period	or encoder resolution (EnDat 2.2 interpol.)														
4096															
Path interpolation	3 ms														
Fine interpolation	Cycle time for fine interpolation = Cycle time of position controller														
Cycle time of current controller	<table border="0"> <tr> <td style="padding-right: 20px;">PWM frequency</td> <td>Cycle time if MP2182.x = 0:</td> </tr> <tr> <td>3333 Hz</td> <td>150 μs</td> </tr> <tr> <td>4000 Hz</td> <td>120 μs</td> </tr> <tr> <td>5000 Hz</td> <td>100 μs</td> </tr> <tr> <td>6666 Hz</td> <td>75 μs (option 49)</td> </tr> <tr> <td>8000 Hz</td> <td>60 μs (option 49)</td> </tr> <tr> <td>10000 Hz</td> <td>50 μs (option 49)</td> </tr> </table>	PWM frequency	Cycle time if MP2182.x = 0:	3333 Hz	150 μs	4000 Hz	120 μs	5000 Hz	100 μs	6666 Hz	75 μs (option 49)	8000 Hz	60 μs (option 49)	10000 Hz	50 μs (option 49)
PWM frequency	Cycle time if MP2182.x = 0:														
3333 Hz	150 μs														
4000 Hz	120 μs														
5000 Hz	100 μs														
6666 Hz	75 μs (option 49)														
8000 Hz	60 μs (option 49)														
10000 Hz	50 μs (option 49)														
Cycle time of speed controller	<ul style="list-style-type: none"> ■ Speed controller cycle time = 2 · current controller cycle time ■ Unless PWM frequency ≤ 5 kHz and double-speed performance, then: Speed controller cycle time = Current controller cycle time 														
Cycle time of position controller	Position controller cycle time = Speed controller cycle time														

Specifications	iTNC 530 HSCI
<p>Maximum motor speed</p> <p>or by way of:</p>	$n_{\max} = \frac{f_{\text{PWM}} \cdot 60000 \text{ min}^{-1}}{p \cdot 5000 \text{ Hz}}$ <p>n_{\max}: Maximum motor speed [min^{-1}] f_{PWM}: PWM frequency [Hz] p: Number of pole pairs</p> <p>The following PWM frequencies are available: 3333 Hz, 4000 Hz, 5000 Hz With option 49: 6666 Hz, 8000 Hz, 10 000 Hz</p> <hr/> <p>Maximum signal frequency of motor encoder = 400 kHz or 800 kHz (depending on MP2561.x bit 0)</p> <p>Calculation of maximum motor speed:</p> $n_{\max} = \frac{f_{\max} \cdot 60000 \text{ [s/min]}}{\text{STR}}$ <p>n_{\max}: Maximum motor speed [min^{-1}] f_{\max}: Maximum signal frequency of motor encoder [kHz] STR: Encoder line count</p> <p>Example 1:</p> $n_{\max} = \frac{400 \text{ kHz} \cdot 60000 \text{ [s/min]}}{2048}$ $n_{\max} = 11718.75 \text{ min}^{-1}$ <p>Example 2:</p> $n_{\max} = \frac{800 \text{ kHz} \cdot 60000 \text{ [s/min]}}{2048}$ $n_{\max} = 23437.5 \text{ min}^{-1}$
<p>Maximum feed rate by way of:</p>	<p>Maximum signal frequency of the position encoder:</p> <ul style="list-style-type: none"> ■ At 1 V_{PP}: 27 kHz or 400 kHz ■ At 11 μA_{PP}: 27 kHz or 140 kHz <p>Calculation of the maximum feed rate F_{max}:</p> $F_{\max} = \text{SP} \cdot f_{\max} \cdot 60^{-3} \left[\frac{\text{m} \cdot \text{s}}{\mu\text{m} \cdot \text{min}} \right]$ <p>F_{\max}: Maximum feed rate [m/min] f_{\max}: Maximum signal frequency of position encoder [kHz] SP: Signal period of the position encoder [kHz]</p> <p>Example:</p> $F_{\max} = 20 \mu\text{m} \cdot 400 \text{ kHz} \cdot 60^{-3} \left[\frac{\text{m} \cdot \text{s}}{\mu\text{m} \cdot \text{min}} \right]$ $F_{\max} = 480 \text{ m/min}$

Specifications		iTNC 530 HSCI
Error compensation		
		<ul style="list-style-type: none"> ■ Linear and nonlinear axis error ■ Backlash ■ Hysteresis ■ Thermal expansion ■ Friction, stick-slip ■ Sliding friction
Monitoring functions without FS		
		<ul style="list-style-type: none"> ■ Amplitude of encoder signals ■ Edge separation of encoder signals ■ Absolute position for encoders with distance-coded reference marks ■ Following error ■ Movement monitoring ■ Standstill monitoring ■ Nominal speed value ■ Checksum of safety-related functions ■ Supply voltage ■ Buffer battery ■ Operating temperature ■ Run time of PLC program ■ Motor current ■ Motor temperature ■ Temperature of power module ■ DC-link voltage
Integrated PLC		
PLC memory		1 GB on HDR or SSDR
Program format		Statement list (STL)
PLC main memory RAM		Dynamic, determined by the free main memory of the control
PLC cycle time		12 ms to 60 ms (adjustable at intervals of 3 ms)
PLC inputs, 24 V–		Via PL
PLC outputs, 24 V–		Via PL
Analog inputs, ±10 V		Via PL
Analog outputs, ±10 V		Via PL
Inputs for thermistors		Via PL

a. Export version



Machine interfacing	iTNC 530 HSCI
Commissioning and diagnostic aids	
	<ul style="list-style-type: none"> ■ Oscilloscope ■ Trace function ■ Table function ■ Watchlist function ■ Logic diagram (integrated in oscilloscope) ■ Log ■ TNCopt PC software ■ TNCscoptNT recording software ■ DriveDiag ■ Bus diagnosis ■ I/O force list ■ S status trace
Permissible temperature range	Operation: 5 °C to +40 °C Storage: -20 °C to +60 °C

User functions

You will find the user functions of the iTNC 530 HSCI in the User's Manuals and the brochure for end users (ID 363807-xx).

Accessories

Accessories	iTNC 530 HSCI
PL 6xxx PLC input/output systems with HSCI	<p>Up to eight PL 6xxx can be connected</p> <ul style="list-style-type: none"> ■ PL 620x (system PL) <ul style="list-style-type: none"> • Necessary once for each control system (except with UEC) • Has connections for TS and TT touch probes • Safety-relevant inputs/outputs • Available for 4, 6 or 8 I/O modules ■ PL 610x (expansion PL) <ul style="list-style-type: none"> • As addition to the system PL for increasing the number of PLC inputs/outputs • Available for 4, 6 or 8 I/O modules ■ I/O modules <ul style="list-style-type: none"> • PLD-H 16-08-00 I/O module with 16 digital inputs and 8 digital outputs • PLD-H 08-16-00 I/O module with 8 digital inputs and 16 digital outputs • PLA-H 08-04-04 I/O module with 8 x +/- 10 V inputs, 4 x +/- 10 V analog outputs, 4 x PT 100 inputs • Empty housing for slots of a PL
Power supply for HSCI components	<ul style="list-style-type: none"> ■ PSL 13x 24 V power supply unit for supplying the HSCI components <ul style="list-style-type: none"> • Outputs: NC: 24 V– (double insulation) PLC: 24 V– (basic insulation) Per output: max. 21 A/ 500 W Total: Max. 32 A / 750 W • Outputs can be connected in parallel
Electronic handwheels	<ul style="list-style-type: none"> ■ One portable HR 410, HR 520 handwheel, or ■ One HR 550 handwheel with wireless transmission ■ One HR 130 panel-mounted handwheel ■ Up to three HR 150 via HRA 110 ■ Rotary encoder at position input can be configured as a handwheel
Touch probes	<ul style="list-style-type: none"> ■ TS 220 triggering 3-D touch probe with cable connection, or ■ TS 440, TS 640, TS 642 three-dimensional touch trigger probe with infrared transmission ■ TT 449 for tool measurement



Accessories	iTNC 530 HSCI
Software	<ul style="list-style-type: none"> ■ PLCdesignNT^a PLC software developing environment ■ IOconfig^a Software for configuring PLC I/O and PROFIBUS-DP components ■ TNCremoNT Data transfer software ■ TNCremoPlus Data transfer software with "live" screen ■ CycleDesign^a Software for creating cycle structures ■ TNCscopeNT^a Software for recording data online or evaluating oscilloscope measurement series ■ DriveDiag^a Software for diagnosis of digital control loops ■ TNCopt^a Software for putting digital control loops into service ■ KinematicsDesign^a Software for configuring the machine kinematics ■ TeleService Software for remote diagnostics, monitoring, and operation ■ RemoTools SDK 3.0 Function library for developing customized applications for communication with HEIDENHAIN controls (option #18 required)

a. Registered customers can download these software products from the Internet.



Software options

Software options	iTNC 530
#8: Software option 1 ID 367591-01	<ul style="list-style-type: none"> ■ Cylinder surface interpolation ■ Feed rate in distance per minute ■ Tilting the working plane ■ Circular interpolation in 3 axes with tilted working plane
#9: Software option 2 ID 367590-01	<ul style="list-style-type: none"> ■ HSC path control: <ul style="list-style-type: none"> • Cycle 32: Additional options in Cycle 32 for roughing and finishing mode, and tolerances for rotary axes ■ 3-D compensation <ul style="list-style-type: none"> • of the tool length and diameter via surface normal vectors (DL/DR in the tool table) • via radius compensation perpendicular to the tool direction (RL/RR in the NC program) • Programming with LN blocks (machine-neutral 3-D programming with various options) ■ 3-D machining <ul style="list-style-type: none"> • TCPM: Tool Center Point Management—Maintain position of the tool tip when positioning with tilting axes (via M128 in the NC program) and with additional options (via Function TCPM in the NC program) • M114: Automatic correction of machine geometry when positioning tilting axes • M144: Compensating the machine's kinematic configuration for ACTUAL/NOMINAL positions at end of block ■ Keeping the tool normal to the contour ■ Straight-line interpolation in 5 axes (export license required) ■ Spline interpolation: Execution of splines (third-degree polynomials) <p>Note: With software version 60642x-02 and higher, the fast block processing time (0.5 ms) and all nominal position value filters (HSC filters) are available as a standard feature without the need to enable a software option.</p>
#18: HEIDENHAIN DNC ID 526451-01	<ul style="list-style-type: none"> ■ All functions of HEIDENHAIN DNC are available (see "Remo Tools SDK" documentation) ■ The following functions of the LSV2 ActiveX control become available (all other options do not require option 18): <ul style="list-style-type: none"> • OpenChatWindow • RunProgram • SetOverride • SetPreset • TransmitChatText • TransmitPLCCommand • TransmitPLCString • HostFunction • EventReceived • NCMsgReceived • PLCMsgReceived



Software options	iTNC 530
#40: DCM—Dynamic Collision Monitoring ID 526452-01	■ Dynamic collision monitoring of defined machine components
#41: Additional Language: ID 530184-01	■ Additional conversational languages: <ul style="list-style-type: none"> • Slovenian • Norwegian • Slovak • Latvian • Korean • Estonian • Turkish • Romanian • Lithuanian
#42: DXF Converter ID 526450-01	■ Conversion of DXF files (also just parts of files) for editing in HEIDENHAIN conversational programming or with smart.NC (see iTNC 530 User's Manual)
#44: Global PGM Settings ID 576057-01	■ Possibility of superimposing various coordinate transformations and settings in the program-run modes of operation (see iTNC 530 User's Manual)
#45: AFC - Adaptive Feed Control ID 579648-01	■ In adaptive feed control, after a teach-in cut the TNC automatically controls the feed rate as a function of the respective spindle power consumption.
#46: Python OEM Process ID 579650-01	■ Possibility of running a Python process on the control
#48: KinematicsOpt ID 630916-01	■ Performing an initial measurement and optimizing the active kinematics of rotary axes
#52: KinematicsComp ID 661879-01	■ Three-dimensional compensation of geometry errors and thermally induced errors
#53: Upgrade functions (Feature Content Level) ID 529969-01	■ Enabling of expanded, useful functions in updates of NC software. However, these are available without option #53 for the initial installation (see iTNC 530 User's Manual)

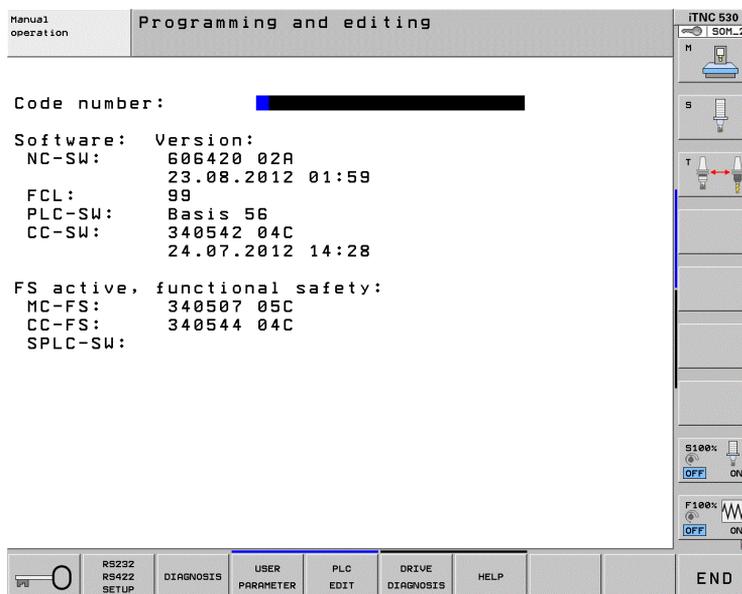
Software options	iTNC 530
#77: 4 Additional Axes ID 634613-01	■ Enabling four additional control loops
#77: 8 Additional Axes ID 634614-01	■ Enabling eight additional control loops
#92: 3D-ToolComp ID 679678-01	■ 3-D tool radius compensation, which uses a compensation-value table to define angle-dependent delta values that describe the tool deviation from the ideal circular shape.
#93: Extended Tool Management ID 676938-01	■ The enhanced tool management function additionally includes the tooling list table and the T usage order table.
#96: Advanced Spindle Interpolation ID 751653-01	■ Advanced functions for an interpolated spindle
#98: CAD Viewer ID 800553-01	■ Opening CAD files directly on the TNC
#101 to #130: OEM option 1 to OEM option 30 ID 579651-01 to 579651-30	■ Possibility for the OEM to enable his applications for the end user via the SIK from HEIDENHAIN
#133: Remote Desktop Manager ID 894423-01	■ Display and remote operation of an external computer unit (e.g. IPC 6341, Windows computer)
#141: Cross Talk Compensation ID 800542-01	■ CTC: Compensation of position error due to axis couplings (only with CC 61xx)
#142: Position Adaptive Control ID 800544-01	■ PAC: Position-dependent adaptation of control parameters (only with CC 6xxx)
#143: Load Adaptive Control ID 800545-01	■ LAC: Load-dependent adaptation of control parameters (only with CC 6xxx)
#144: Motion Adaptive Control ID 800546-01	■ MAC: Motion-dependent adaptation of control parameters (only with CC 6xxx)
#144: Motion Adaptive Control ID 800546-01	■ MAC: Motion-dependent adaptation of control parameters (only with CC 6xxx)
#145: Active Chatter Control ID 800547-01	■ ACC: Active suppression of chatter (only as of CC 6xxx)

2.7 Software

2.7.1 Designation of the software

The iTNC 530 HSCI features a separate software for the NC and the PLC. The NC software is identified with an eight-digit number.

If you press the MOD key in any operating mode, you can display the ID numbers of the NC software, the DSP software (CC) and the SKERN software with active functional safety (FS). An installed service pack is shown by **SPx** after the ID number of the NC software.



- **Code number:** Input field for code numbers
- **NC-SW:** NC software version and date
- **FCL:** Currently set Feature Content Level (FCL)
- **PLC-SW:** Number of the PLC program
- **CC-SW:** Software version of the CC, UEC and UMC controller units and date

If the HEIDENHAIN integrated functional safety is active:

- **MC-FS:** SKERN software version of the MC main computer
- **CC-FS:** SKERN software version of the CC, UEC, UMC controller units
- **SPLC-SW:** Number of the SPLC program

Type

The iTNC 530 HSCI is shipped with the following NC software:

Standard	Export	Comment
606420-xx	606421-xx	iTNC 530 HSCI
606422-xx	606423-xx	iTNC 530 HSCI

Due to restrictions on the export of the iTNC 530, HEIDENHAIN can also supply a special export version. This export version differs from the standard control through the installed NC software type. HEIDENHAIN releases a new NC software type whenever it introduces extensive new functions.

2.7.2 PLC software

The PLC software is stored on the hard disk or the solid state disk of the iTNC. You can order a PLC basic program directly from HEIDENHAIN. With the PLC development software **PLCdesignNT**, the PLC program can very easily be adapted to the requirements of the machine.

2.7.3 Additional control loops or software options

The system identification key (SIK) determines the available software options and the number of available control loops on the iTNC 530 HSCI. Each MC 6xxx can clearly be identified by the system identification key (SIK). You will find the SIK number on the outside of the MC 6xxx housing (below the ID label) and on the SIK board. To install the SIK, see "Handling the HDR hard disk" on page 195.



Note

If you replace the MC 6xxx, you must also replace the SIK in order to ensure that the enabled control loops and software options will also be enabled on the new hardware.

There are different versions of SIKs for the iTNC 530 HSCI, depending on the MC main computer used and the number of required control loops. Additional control loops (up to the maximum permissible number) can be enabled at any later time. See "SIK MC 62xx, MC 63xx" on page 82.

Software options can be used to adapt the scope of functions of the control to the respective needs and applications. See "Software options" on page 136.

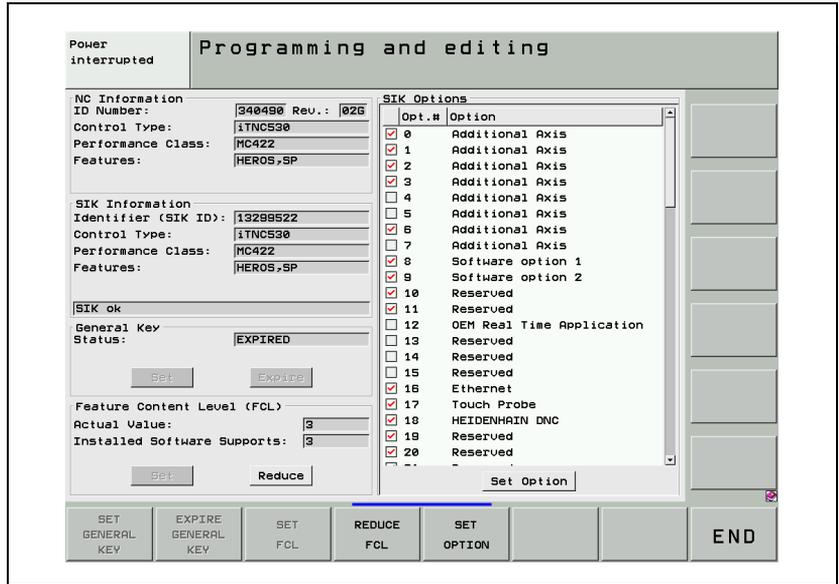
If you want the code number for enabling additional control loops or software options, please contact HEIDENHAIN for the code number. HEIDENHAIN can give you the code number after you state your SIK number.

In order to enable additional control loops or software options, proceed as follows:

- ▶ While in the **Programming and Editing** operating mode, press the MOD key.
- ▶ Enter the code number SIK and confirm your entry with the ENT key.



The following display will appear:



The display gives you the following information and possibilities for settings:

NC Information:

Display	Meaning
ID Number	Software
Rev.	Software version
Control Type	Control model
Performance Class	Type of main computer
Features	Characteristics of the control

SIK information:

Display	Meaning
Identifier (SIK ID)	SIK number
Control Type	Type of control
Performance Class	Type of main computer
Features	Characteristics of the SIK
SIK ok, wrong SIK (Control Type mismatch), wrong SIK (Features mismatch), wrong SIK (Performance Class mismatch), no SIK (Programming Station) or no SIK	Status of the SIK



General Key:

The general key permits you to enable and test all new feature content level functions and software options. The general key is valid for 90 days after the first enabling. After these 90 days have expired, the general key can only be used again after the software version on your control has been updated.

Display	Meaning	
Status	NONE	General key was not used yet for this software version.
	dd.mm.yyyy	Date up to which all options will be available. It is not possible to enable them again after this date.
	EXPIRED	General key has expired for this software version. Enabling is not possible.
Set	<p>Pressing the Set button or the SET GENERAL KEY soft key opens a window in which you press the Apply button or the SET GENERAL KEY button again to enable all options for 90 days.</p> <p>If this is done successfully, the General key has been set message appears and the expiration date of the general key is shown in the State field.</p>	
Expire	<p>Pressing the Expire button or the EXPIRE GENERAL KEY soft key opens a window in which you press the Apply button or the EXPIRE GENERAL KEY button again to expire the general key immediately.</p> <p>Be aware that it is then no longer possible to enable the general key for this software version!</p>	

Feature Content Level (FCL):



Danger

If you reduce the feature content level, then all new FCL functions and enhanced functions will be set to the desired lower version. Only error fixes remain active. This action can only be reversed by entering a valid code number again.

Display	Meaning
Actual Value	Current version of the feature content level
Installed Software Supports	Highest possible FCL with the current software
Set	Pressing the Set button or the SET FCL soft key opens a window in which you can enter the code number for the desired feature content level under Enter Key Code . HEIDENHAIN can give you the code number after you state your SIK number. Then press the Set FCL button or the SET FCL soft key to confirm the entry. If this is done successfully, the message Feature Content Level has been set appears.
Reduce	Pressing the Reduce button or the REDUCE FCL soft key opens a window in which you can enter the desired lower feature content level under New (lower) FCL . Please note that this action can only be reversed by entering a valid code number again. If you really want to reduce the feature content level, then press the Apply button or the REDUCE FCL soft key to confirm the entry. If this is done successfully, the message Feature Content Level has been reduced appears.



SIK Options:

All available software options and their corresponding numbers are listed in a table. The check marks in the first column indicate which software options have been enabled on your control.

To enable additional software options, proceed as follows:

- ▶ Press the MOD key and enter the code number **SIK** to display the input mask for SIK functions.
- ▶ In the table on the right side of the screen, use the arrow keys or the mouse to select the software option you want to enable.
- ▶ Pressing the **Set Option** button or the **SET OPTION** soft key opens a window in which you can enter the code number for the desired software option under **Enter Key Code**. HEIDENHAIN can give you the code number after you state your SIK number.
- ▶ Enter the code number and confirm the entry by pressing the **Apply** button or the **SET OPTION** soft key.
- ▶ If this is done successfully, the message **Option <number> has been set** appears, and the option is checked in the table.

When you leave this table of SIK functions by pressing the **END** soft key or the **END** key, you are requested to reboot the control if you have made any changes. Press the emergency stop button and press the **Reboot Now!** button or the **REBOOT NOW** soft key to reboot the control and activate the changes.

To display the corresponding number of machine-parameter indexes:

- ▶ In OEM.SYS, enter the code word **PWMPARAMETER** = followed by the required number of machine-parameter indexes MP2xxx.y for the current and speed controller.
- ▶ In OEM.SYS, enter the code word **AXISNUMBER** = followed by the required number of remaining machine-parameter indexes.

Status of options that have been set

Module 9067 Status of software settings

PLC Module 9067 enables you to request status information about software settings. Module 9067 can currently be used to interrogate the software options set in the SIK.

Call:

PS B/W/D/K <Mode>
0: Interrogate whether SW option is set in the SIK

PS B/W/D/K <Number>
If mode is 0: Number of SIK option

CM 9067

PL B/W/D <Status>
Status of SIK option (if mode is 0):
0: Not set
1: Set

Error code:

Marker	Value	Meaning
M4203	0	Function was performed correctly
	1	Error code in W1022
W1022	1	Invalid value for number
	2	Invalid value for mode

Temporary enabling of an option

You have the possibility of enabling software options with a temporary code number for a limited time. You can define a time between 10 to 90 days for enabling the software options. However, each option can only be enabled once with a temporary code number.

If you want to enable a software option temporarily on the control via the temporary key, proceed in the same way as for the standard enabling of software options. Press the **Set Option** button or the SET OPTION soft key. This opens a window in which you can enter the code number for the desired software option in **Enter Key Code**.

If the software option was enabled successfully, the expiration date of the temporary enabling is shown in the **Expires** column under **SIK Options**. After the defined period has expired, the entry in the **Expires** column will change to **EXP**, meaning "expired." The software option is then no longer available.

A software option can be enabled for an unlimited period at any time by means of the code number, which you will receive from HEIDENHAIN after stating the SIK number.

HEIDENHAIN would like to point out that it is not possible to use the OEM-specific options with the SIKs of the first generation. If you encounter any problems in this respect, please contact your HEIDENHAIN service agency.

You can generate the temporary code number with the TNCOEMOption tool for PCs. The tool is available for cost-free download from our HEIDENHAIN HESIS-Web including Filebase:

- ▶ FileBase
- ▶ PC Software
- ▶ NC Service

When you generate the code number, you must specify the number of the respective software option, the number of days you want to enable the option, and an optional OEM Key. The OEM Key provides protection against unauthorized persons generating code numbers for your machines with the help of the PC tool in order to enable software options without your approval. However, it is not essential to specify an OEM Key.

The OEM Key for generating the code number must be identical to the OEM Key on your control.

The OEM key was added to the display and management of the SIK options, feature content level and general key on the control. Press the MOD key and enter the code number **SIK** to display the input form for SIK functions:

■ **OEM Key**

The OEM Key on the control can only be used for enabling software options with a temporary code number if the same OEM Key was specified for the generation of the temporary code number.

If the OEM Key on the control is not identical to the one used for generating the code number, the software option will not be enabled. Once the OEM Key has been set, it cannot be reset.

Display	Meaning	
OEM Key for temp. options	NONE	OEM Key was not set
	SET	OEM Key was set
Set OEM Key	<p>Press the Set OEM Key button or the SET OEM KEY TEMP. OPT. soft key to open a window in which you can enter the OEM Key.</p> <p>Keep in mind that this process cannot be undone. The OEM Key must consist only of numbers. Confirm your entry by pressing the Apply button, or by pressing the SET OEM KEY TEMP. OPT. soft key again.</p> <p>If the OEM Key was set successfully, the message OEM Key has been set appears, and the status in OEM Key for temp. options changes to SET.</p>	

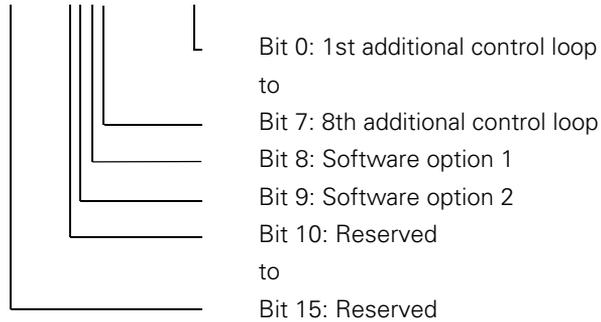


Displaying the status of an option

The status of an option (enabled or not) can be displayed either as a menu by entering the code number SIK (see previous pages), or bit-encoded after pressing the MOD key in the **OPT** line.

Each bit has the following meaning:

OPT: %xxxxxxxxxxxxxxxx



Options for the OEM

The SIK options 101 to 130 are available as OEM-specific options for you, the machine tool builder. You can assign your own OEM-specific software options to these options.

This makes it possible for you to enable your own applications (e.g. cycles) for the end user via the SIK from HEIDENHAIN.

The options can be enabled by entering a code number. HEIDENHAIN can give you the code number after having been informed of the SIK number. You also have the possibility of creating a temporary key for these OEM-specific options in order to enable the options for a period of max. 90 days.

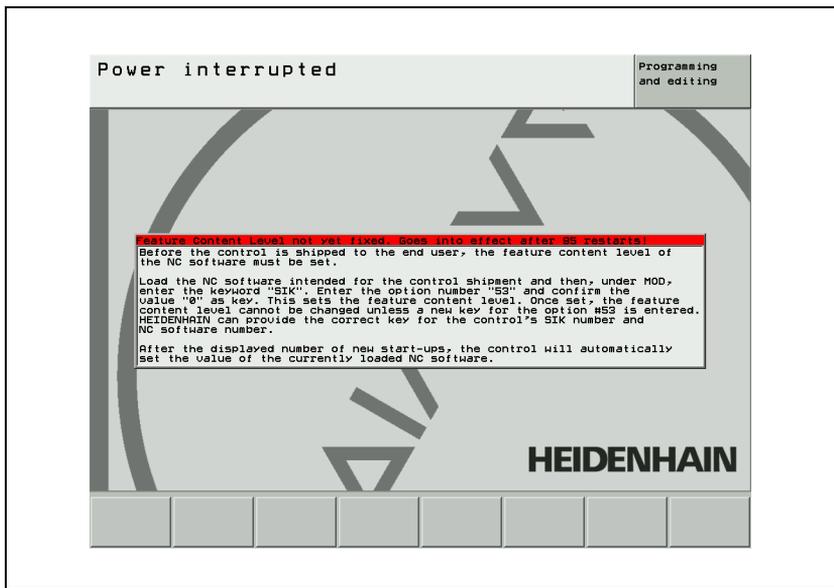
The PLC Module 9067 enables you to request the status of the SIK options. Use this module to request information from the SIK about whether certain software options are enabled via the SIK. Depending on the response, you can activate the software option via the PLC program.

2.7.4 Upgrade functions (feature content level)

In the past, each new NC software version contained error fixes as well as expanded functions. Users who wanted only the NC software update to eliminate the errors often felt bothered by the enhanced functions. For this reason, error fixes and expanded functions within the software are now handled in part separately.

The upgrade functions can then be enabled by entering a code number. HEIDENHAIN can give you the code number after it is informed of the SIK number and NC software version.

- The **upgrade functions** are defined as "feature content level" (FCL) in the SIK under option 53.
- The first time an NC software with upgrade functionality is installed on a control (i.e. no FCL has been set in the SIK), then the entire scope of functions can be used (including the **upgrade functions**).
The FCL is then automatically set after 255 restarts, or by entry of the code number 0 under **option #53**, and all upgrade functions belonging to this software version are enabled as well.
A note appears asking to confirm the installed NC software as the initial version, or if another initial software version is to be installed.
- If the FCL has already been set in the SIK of a control, then after a software update the new **upgrade functions** of the newer software version can be used only after entry of a code number from HEIDENHAIN under **option #53** (see "Software options" on page 136).
- After pressing the MOD key, the current status of the FCL is displayed in addition to the software versions.
- The FCL is incremented with each new version of the NC software.
- If the **upgrade functions** are enabled via the FCL for a software version, then all **upgrade functions** of this software version and all its predecessors are available. For example, if the FCL for version 606420-03 is set, then all **upgrade functions** from version 606420-02 are also available immediately.
- If a newer software version, e.g. 606620-06, is simply loaded onto a control, then the already existing **upgrade functions** remain available, but the **upgrade functions** of the newer version cannot be used. They must be enabled by entering a new code number.



Availability of feature content levels:

Software version	Available FCL	Notes on the FCL
60642x-01	FCL 04	
60642x-02	FCL 04	
60642x-03	FCL 04	

2.7.5 NC software exchange on the iTNC 530



Note

- The NC software must be exchanged only by trained personnel.
- For exchanging the NC software, HEIDENHAIN provides compressed files (*.zip) with the NC software. The packed files are transferred to the hard disk of the control and unpacked. The packed files remain on the hard disk of the control.

General Notes

- The following files are necessary for an update, and must be located in the same directory:
 - **setup.zip**
 - **setup.elf** (setup.exe, as an alternative to **setup.omf**)
 - **setup.ini** (this is to be created by the OEM, and is only necessary if the update is to occur according to a specific sequence)Please note that this currently means that the version of the software to be installed is not displayed in advance.



Note

You can download the two files required for the update (**setup.zip**, **setup.elf**) from the HEIDENHAIN HESIS-Web including Filebase and unzip them in your update directory.

- Software updates and service packs are loaded in the same manner.
- Automated updating is possible. If your directories are structured appropriately and you use a **setup.ini** control file, this method makes it possible to update the control automatically when it is booted (see "Automated update (setup.ini)" on page 153).
- Copying of the update files (setup.elf and setup.zip) to the system partition occurs automatically as part of the update program in **directories** with the following naming convention:
 - iTNC 530 HSCI
 - Software update: **SYS:\zip**
 - Service pack: **SYS:\zip**



Attention

No manual changes may be made to these directories, since they might be required for restoring earlier software versions.



- Before the update, a query appears asking whether the necessary binary to ASCII conversion should be performed. HEIDENHAIN recommends always performing this conversion. Only files whose binary version has **changed** are converted. As part of the recommended procedure, immediately thereafter a query appears if there is not enough memory space:
 - Cancel if not enough memory
 - Delete largest or oldest files first
 After successful installation, these ASCII files are automatically reconverted to binary format.
 This procedure can be automated in the control file mentioned above.
- The NC software has been prepared in such a manner that when an update is performed or a service pack loaded, the PLC program and PLC partition can be updated as well, according to the requirements of the OEM. When the NC software is updated, the OEM uses the HEIDENHAIN PC software PLCdesignNT to add all necessary files to the **setup.zip** archive. These files are copied to the appropriate locations during an update.

Automated update (setup.ini)

Automated updating is possible with the iTNC 530. In order to automate an update as much as possible, a control file with the name **setup.ini** is necessary in addition to the **setup.zip** and **setup.elf** files. If your directories are structured appropriately and you use this **setup.ini** control file, this method makes it possible to update the control automatically when it is booted. If there is a certain "install" directory on your iTNC containing a **setup.ini** control file in addition to the update files, an update is performed automatically according to the instructions in this control file. The following directories are checked during booting for the presence of a control file:

- iTNC 530
 - **TNC:\install**
 - or, if a USB memory device is connected
 - **USB0:\install** (**USB0:** first partition of the first USB memory device)

The **setup.ini** file can be created with a simple text editor.

An example of a setup.ini file:

```
Interactive=1
Confirm=1
Language=ENGLISH
SavePlc=TNC:\backup\606420_001.zip
```

The following settings are possible:

Parameter	Description
Interactive=[0, 1]	Deletion of the NC software archive (old software versions) and binary to ASCII conversion must be confirmed by the user. 0: No 1: Yes [default]
Confirm=[0, 1]	Start of update and reboot process must be confirmed by the user 0: No 1: Yes [default]

Parameter	Description
ServiceRequest=[0, 1]	A service request is triggered after an update (only if remote diagnosis is active) 0: No [default] 1: Yes
ConvertToAscii=[0, 1] (only if Interactive=0)	Binary to ASCII conversion with automated update process (not with service pack) 0: No 1: Yes [default]
CopyToSys=[0, 1]	The setup archive to be installed is copied to the directory SYS:\zip (iTNC with Windows: C:\Program Files\install). This makes it possible to return to this software version later. 0: No 1: Yes [default]
DeleteFiles= [DATE,SIZE,CANCEL] (only if Interactive=0)	Procedure during binary to ASCII conversion during automated update if there is not enough memory available on the TNC or PLC partition DATE: Delete oldest files first SIZE: Delete largest files first CANCEL: Cancel the update [default]
DeleteArchives= [DATE,SIZE,CANCEL] (only if Interactive=0)	Procedure if there is not enough room on the SYS partition for the update DATE: Delete oldest archive first SIZE: Delete largest archive first CANCEL: Cancel the update [default]
Language=[ENGLISH, GERMAN] (only if Confirm=0)	Language for the dialog guidance during the update if no user activities are required for starting the update and rebooting (Confirm=0) GERMAN: German dialog text ENGLISH: English dialog text [default]
DelSource=[0, 1]	Delete the source files (setup.zip, setup.ini, setup.elf/exe) once the update has completed successfully 0: No [default] 1: Yes
DeleteIni=[0, 1]	Delete the setup.ini file after a successful update. 0: No [default] 1: Yes

Parameter	Description
SavePlc=<name>	<p>The entire content of the PLC partition can be saved in binary form as a .zip file. This makes it possible to restore this software state including the PLC files.</p> <p>Here you enter the path and file name for the ZIP file in which the entire PLC partition is saved in binary format. Please consider the software version when assigning the file name.</p> <p>After an update, the -SETUP code number is entered on a single-processor control, and the Setup Back button is pressed in the iTNC control panel on a dual-processor control in order to reverse an update. The previous software version is then reinstalled. The PLC partition is restored from the *.zip file that was entered under SavePlc= in the setup.ini file. Please note that a subdirectory in which you want to save the *.zip file must be created on the TNC partition before the update. Otherwise the update is canceled when backing up the old PLC partition.</p> <p>Only updates from one full software version to another can be reversed. You cannot reverse the installation of service packs.</p>
RestorePlc=<name>	<p>The Restore function is used to restore the PLC data to a certain software state.</p> <p>Here you enter the path and file name for the ZIP file containing the PLC data state (in binary format) appropriate to the NC software version to be installed. This should be the ZIP file which was saved with the SavePlc function, containing the corresponding PLC state and saved on the PLC partition.</p>

Automated updating of machine parameters

If a PLC:_mpupdate directory is created, then files that automatically update or expand the active and selected MP or OEM.SYS files when the control is started can be saved here.

Files containing the name **merge.*** (**merge.mp** and **merge.oem.sys**) expand the MP and OEM.SYS files by the entries contained in them. Files containing the name **overwrite.*** (**overwrite.mp** and **overwrite.oem.sys**) contain updated entries for the corresponding files, and overwrite entries with the same names in the MP and OEM.SYS files.

This means that MP subfiles cannot be considered.

Manual update

The following procedure is used to perform a manual update (without a **setup.ini** file in the setup directory) or an installation of a service pack:

- ▶ HEIDENHAIN recommends using the PC program TNCremoNT from HEIDENHAIN to make a backup for the control.
- ▶ Connect the appropriate network drive or USB memory medium containing the files necessary for the update (**setup.zip**, **setup.elf**). If the network drive is not displayed, please proceed as follows:
 - Enter the code number 95148 and confirm your entry with the ENT key.
 - While in the **Machine-parameter programming** mode, press the PGM MGT key.
 - Press the **NETWORK** soft key.
 - Select the PC to be connected with the arrow keys and press the **CONNECT DRIVE** soft key.
 - To exit the list of network drives, press the **END** soft key.
 - To exit the program manager, press the **END** soft key.
 - Press the END key to exit the **Machine-parameter programming** mode.
- ▶ In the **Programming and Editing** mode, press the **MOD** key.
- ▶ Enter the code number **SETUP** and confirm your entry with the **ENT** key.
- ▶ In the upper part of the window, select the folder containing the setup files for the new NC software, and confirm with the **ENT** key. The file to be called for the installation (**setup.elf**) is displayed in the lower part of the window.
- ▶ To switch to the lower part of the window, press the **FILES** soft key. (Press the **PATH** soft key to return to the top part of the window.) In the lower part of the window, select the *.elf file of the new NC software with the cursor keys and press **ENT**. Following that, the NC software exchange will start.
- ▶ Use the cursor keys to select the language desired for the update guidance, and confirm your entry with the **ENT** key.
- ▶ Choose the desired action for the event that there is not enough memory available on the TNC or PLC partition for the binary to ASCII conversion:
 - **Cancel if not enough space:** The update procedure is canceled if there is not enough space available, and a message to this effect appears. In this case you must save the files (*.h, *.i, and tables) externally, if they are needed, and then remove them manually in order to make space for the conversion.
 - **Delete largest files first:** The largest files (*.h, *.i, and tables) are deleted until there is enough space for the conversion.
 - **Delete oldest files first:** The oldest files (*.h, *.i, and tables on the PLC and TNC partitions) are deleted until there is enough space for the conversion.



- ▶ If there is not enough space on the SYS partition, select deletion of the oldest setup files. This procedure is repeated until there is enough memory space available.
- ▶ Confirm the update actions listed and that are to be performed.
- ▶ After the update has finished successfully, confirm the restart of the control.
- ▶ Read-in the files which you had saved to a PC using TNCremoNT.
- ▶ With the **COPY SAMPLE FILES** soft key, the HEIDENHAIN standard tables for cutting data, the tilting-axis geometry, and the M-function macros as well as a prototype for a freely definable table (contains only the column **Name**) can be copied into the corresponding directories.
- ▶ The NC software exchange is completed.



Note

If a **setup.ini** file exists in the setup directory when you update **manually**, then the update is performed according to the instructions in this file. See "Automated update (setup.ini)" on page 153.

Automatic software exchange

Automated update

An automated update is usually a part of a manual update. In addition, this means that you can use the control file to reduce the number of user actions necessary for the update to a minimum, and at the same time make a backup of the PLC partition.

If one of the following directories exists when an iTNC is booted,

- TNC:\install\

or, if a USB memory device is connected

- USB0:\install\ (**USB0**: first partition of the first USB memory device)

and if a **setup.ini** control file is saved in this directory, then an automated update is performed according to the instructions in this control file (see "Automated update (setup.ini)" on page 153).



Note

Please note that if automated updating is selected, the update program may start with a delay due to the file size.

Software exchange via remote operation

Update via remote operation

How to perform an update via remote operation:

- ▶ In the **Programming and Editing** mode of the iTNC, press the **MOD** key.
- ▶ On the iTNC, activate remote maintenance by pressing the **Service ON** soft key.
A service request is triggered and a connection is established.
- ▶ Transmit the **setup.elf** and **setup.zip** files to a suitable directory on the control (e.g. **TNC:\update**).
- ▶ Continue the update via remote operation as described under "Manual update".

Notes on updating via remote operation

- If remote maintenance is active when the update is started, the remote maintenance is deactivated (this does not affect the current remote operation).
- After the reboot or after the update was canceled due to an error, a service request is triggered. The service request information indicates whether the update was successful.
- If the control does not boot due to missing or incorrect machine parameters, a service request is triggered until the "power interrupt" stage of the boot process is reached.

Soft keys for update functions

Soft key	Function
	Convert the files on the hard disk from binary format to ASCII format and save nonvolatile markers in the PLCMEM.A file.
	Convert the files on the hard disk from ASCII format to binary format and save nonvolatile markers in the PLCMEM.A file.
	Copy cutting-data tables, tables for tilting-axis geometry, and the table of M-function macros from the SYS partition into the corresponding directories of the TNC or PLC partition, and create prototypes of the tables.
	Activate or delete existing NC software.
	Exchange NC software.

Equivalent file name extensions in ASCII and binary format					
.H	.H%	.I	.I%	.T	.T%
.TCH	.TC%	.D	.D%	.P	.P%
.PNT	.PN%	.COM	.CO%	.CMA	.CM%

- ▶ Press the **NCV** → **iTNC** soft key.
- ▶ In the upper part of the window, you select the folder containing the *.zip file of the new NC software. The folder contents are displayed in the lower part of the window.
- ▶ To switch to the lower part of the window, press the **FILES** soft key. To return to the upper part of the window, use the **PATH** soft key. In the lower part of the window, select the *.zip file of the new NC software with the cursor keys and press ENT. Following that, the NC software exchange will start.
- ▶ All NC software versions that exist in the control are shown in the following list box.
- ▶ Select the new NC software with the arrow keys and press the **SELECT** soft key. The selected NC software is marked with an asterisk (*) in the **Sel** column. Confirm your selection with the **YES** soft key. The control activates the selected NC software and performs a reset. Press the END key, the **NO** soft key or the **END** soft key to exit the list box without making a new selection.
- ▶ If required, complete or delete the machine parameters.
- ▶ While in the **Programming and Editing** operating mode, press the MOD key.
- ▶ Enter the code number 95148 and confirm your entry with the ENT key.
- ▶ While in the **Machine-parameter programming** mode, press the MOD key.
- ▶ Press the **UPDATE DATA** soft key.
- ▶ Press the **ASC** → **BIN** soft key to convert the files on the hard disk from ASCII format back to binary format.
- ▶ Read-in files which you had saved to a PC.
- ▶ The NC software exchange is completed.
- ▶ With the **COPY SAMPLE FILES** soft key, the HEIDENHAIN standard tables for cutting data, the tilting-axis geometry, and the M-function macros as well as a prototype for a freely definable table (contains only the column **Name**) can be copied into the corresponding directories.

**NC software
exchange from
standard version to
export version (and
vice versa)**

The control version (iTNC 530 or iTNC 530 E) is stored in the SIK. If the NC software is exchanged on a software version different from that stored on the SIK, after the control starts up a message appears that the control can be operated only as a programming station. This message must be acknowledged.

Procedure for exchanging the NC software:

- ▶ Exchange the NC software as described above.

After the control powers up, **Incorrect software version** appears.

- ▶ Switch off the control.
- ▶ Exchange the SIK for a new, appropriate SIK (for the ID, see page 82); for the location of the SIK in the MC 6xxx, see page 209.
- ▶ Switch on the control.

Since the new SIK has another SIK number, the options that are enabled on the old SIK must be re-enabled on the new SIK. After you state your SIK number, HEIDENHAIN can give you the code numbers for enabling the options.

To make it possible to identify the control from outside, after you indicate the ID and serial number of the control, you will receive a new ID label with the new data.

- ▶ Stick the new ID label with the new control designation on the MC 6xxx.



Attention

After you have changed the NC software from the standard version to the export version, you must delete the packed files of the standard version from the hard disk, since the packed files are also subject to export authorization.

After the export version has been started, a prompt appears, asking if the compressed files of the standard version are to be deleted from the hard disk. If you answer with YES, all compressed files with the names of the standard version are deleted from the hard disk.

The procedure for deleting compressed files is on page 162.

Activating existing NC software

Before activating existing NC software, ensure that the free space on the storage medium of the MC 6xxx is at least 50 % the size of the occupied space. If that is not the case, you must save the files to a PC, e.g., with the TNCremoNT data-transfer software for PCs.

- ▶ While in the **Programming and Editing** operating mode, press the MOD key.
- ▶ Enter the code number 95148 and confirm your entry with the ENT key.
- ▶ While in the **Machine-parameter programming** mode, press the MOD key.
- ▶ Press the **UPDATE DATA** soft key.
- ▶ The name and path of a log file can be entered after **Path =** in the header.
- ▶ Press the **BIN → ASC** soft key to convert the files on the hard disk from binary to ASCII format.

Equivalent file name extensions in ASCII and binary format					
.H	.H%	.I	.I%	.T	.T%
.TCH	.TC%	.D	.D%	.P	.P%
.PNT	.PN%	.COM	.CO%	.CMA	.CM%

- ▶ Press the **SELECT** soft key.
- ▶ All NC software versions that exist in the control are shown in the selection window that appears.
- ▶ Select the NC software to be activated with the arrow keys and press the **SELECT** soft key. The selected NC software is marked with an asterisk (*) in the **Se1** column. Confirm your selection with the **YES** soft key. The control activates the selected NC software and performs a reset. Press the END key, the **NO** soft key or the **END** soft key to exit the list box without making a new selection.
- ▶ If required, complete or delete the machine parameters.
- ▶ While in the **Programming and Editing** operating mode, press the MOD key.
- ▶ Enter the code number 95148 and confirm your entry with the ENT key.
- ▶ While in the **Machine-parameter programming** mode, press the MOD key.
- ▶ Press the **UPDATE DATA** soft key.
- ▶ Press the **ASC → BIN** soft key to convert the files on the hard disk from ASCII format back to binary format.
- ▶ The activation of the NC software is completed.
- ▶ With the **COPY SAMPLE FILES** soft key, the HEIDENHAIN standard tables for cutting data, the tilting-axis geometry, and the M-function macros as well as a prototype for a freely definable table (contains only the column **Name**) can be copied into the corresponding directories.

Deleting the packed files of existing NC software

- ▶ While in the **Programming and Editing** operating mode, press the MOD key.
- ▶ Enter the code number 95148 and confirm your entry with the ENT key.
- ▶ While in the **Machine-parameter programming** mode, press the MOD key.
- ▶ Press the **UPDATE DATA** soft key.
- ▶ Press the **SELECT** soft key.
- ▶ All NC software versions that exist in the control are shown in the following options display.
- ▶ Select the NC software to be deleted with the arrow keys and press the **DELETE** soft key to delete all packed NC software files. The currently active NC software is marked with an asterisk (*) in the **Sel** column. Confirm your selection with the **YES** soft key. Press the **NO** soft key or the **END** soft key to exit the list box without deleting an NC software.



Note

If the packed files of an NC software, including the currently active software, are deleted, the respective software cannot be activated via the selection window any longer. The software concerned must then again be transferred to the control.

The deletion of the packed files of the currently active NC software has no other effects.

Entries in the log file

If errors occur during conversion, the TNC will display error messages and record them in the log file. During the NC software switch, the name and path of a log file can be entered in the header after **Path =**; the extension .A must be used. If no entry is made in this line, the file TNC:\CVREPORT.A is created.

Each error message contains:

- Error message
- Error code
- Error cause
- File concerned

Example:

```
=====
ERROR           :REMANENT PLC DATA NOT RESTORED
ERRNO          :2
ERROR MESSAGE  :Program name not found
FILE           :PLCMEM.A
=====
```

Error message	Meaning
CANNOT OPEN DIRECTORY	File could not be opened
REMANENT PLC DATA NOT RESTORED	No access to the file PLCMEM.A
NOT ENOUGH SPACE	Too little free memory on the hard disk
CONVERSION BIN ASC FAILED	A binary file has an incorrect format (e.g., binary format from an old NC software)
CONVERSION ASC BIN FAILED	An ASCII file on the hard disk is incorrect

2.7.6 Availability of software versions, software options and FCL

Software options

The table shows which software option is available as of which software version.

Software option		Available as of software version:
# 0 to # 4	Additional Axis	60642x-01
# 5 to # 7	Additional Axis	60642x-01
# 8	Software option 1	60642x-01
# 9	Software option 2	60642x-01
# 18	HEIDENHAIN DNC	60642x-01
# 40	Dynamic Collision Monitoring (DCM)	60642x-01
# 41	Additional language	
	Slovenian	60642x-01
	Norwegian	60642x-01
	Slovak	60642x-01
	Latvian	60642x-01
	Korean	60642x-01
	Estonian	60642x-01
	Turkish	60642x-01
	Romanian	60642x-01
	Lithuanian	60642x-01
# 42	DXF Converter	60642x-01
# 44	Global PGM Settings	60642x-01
# 45	Adaptive feed control (AFC)	60642x-01
# 46	Python OEM process	60642x-01
# 48	KinematicsOpt	60642x-01
# 52	KinematicsComp	60642x-01
# 53	Feature Content Level	60642x-01
# 77	Enabling four additional control loops	60642x-01
# 78	Enabling eight additional control loops	60642x-01

Software option		Available as of software version:
# 92	3D-ToolComp	60642x-01
# 93	Extended tool management	60642x-01
# 96	Advanced Spindle Interpolation	60642x-02
# 98	CAD Viewer	60642x-02
# 101 to # 130	OEM option 1 to OEM option 30	60642x-01
# 133	Remote Desktop Manager	60642x-02
# 141	Cross Talk Compensation (only with CC 6xxx)	60642x-02
# 142	Position Adaptive Control (only with CC 6xxx)	60642x-02
# 143	Load Adaptive Control (only with CC 6xxx)	60642x-02
# 144	Motion Adaptive Control (only with CC 6xxx)	60642x-03
# 145	Active Chatter Control (only with CC 6xxx)	60642x-03

This table shows which FCL is available with which software version.

FCL	Functions of the FCL	Available as of software version:
FCL 2	<p>Conversational programming functions:</p> <ul style="list-style-type: none"> ■ Cycle 441: Global touch-probe parameters ■ CAD point filter (offline) ■ 3-D line graphics ■ Virtual tool axis ■ In the active tool-axis system <p>smarT.NC functions:</p> <ul style="list-style-type: none"> ■ Coordinate transformation: Units 7, 8, 10, 11 ■ PLANE function: Unit 140, Tilting the Working Plane ■ Contour-pocket with variable depth ■ Graphic block scan <p>General:</p> <ul style="list-style-type: none"> ■ Expanded USB functionality ■ Network connection over DHCP and DNS 	60642x-01
FCL 3	<p>The TNCguide FCL function and Asian languages require at least 256 MB RAM</p> <ul style="list-style-type: none"> ■ TNCguide, context-sensitive help system (user documentation) ■ KinematicsOpt requires FCL 3 <p>smarT.NC functions:</p> <ul style="list-style-type: none"> ■ Feed-rate reduction during contour pocket machining ■ Units 408, 409 for datum setting ■ Unit 130 contour pocket on point pattern ■ Parallel programming ■ Behavior between two machining points <p>Conversational programming functions:</p> <ul style="list-style-type: none"> ■ Feed-rate reduction during contour pocket machining <p>Touch probe functions:</p> <ul style="list-style-type: none"> ■ Probe Units 408, 409 for datum setting ■ 3-D Probe Cycle 4 	60642x-01
FCL 4	<ul style="list-style-type: none"> ■ DCM for handwheel superimposition ■ DCM: Graphic simulation of the protected space or of collision objects ■ 3-D basic rotation light 	60642x-01



2.7.7 Installing a service pack

When needed, HEIDENHAIN prepares service packs for certain NC software versions. A service pack is loaded in addition to an already completely installed NC software.

When the control is started, a note regarding the installed service pack is shown. This can be replaced by a special logo (see "Powering up and shutting down the control" on page 1260).

The latest service pack always includes all changes from earlier service packs.

HEIDENHAIN recommends installing all released service packs.

Soft key	Function
	Installing a service pack



Danger

- When needed, HEIDENHAIN prepares service packs for the various versions of the NC software. Installation of a service pack in addition to the already installed NC software implements important error fixes.
- Please check whether there are service packs for the respective NC software available in the download area for your control. Please also note the Update Information releases, which contain important notes on the service packs in addition to the information contained in the Technical Manual.
- Please ensure that the NC software always contains the latest service pack before you ship the machine. Perform all tests required of the machine or the NC software again after having installed a service pack.



Attention

If a service pack has already been installed, it will not be possible to install a service pack with a lower index. This will be checked during the installation of a service pack for the software and a message will be displayed if an error is found.



Note

A service pack may only be loaded onto completely installed NC software (files in binary format), and may only be done by trained personnel.

The service pack consists of packed files (*.zip). The packed files are transferred to the hard disk of the control and unpacked.

The installation of a complete software version (**setup.elf** or **setup.omf**) is also possible via the **LOAD SP** soft key.

It is not necessary to convert from binary format to ASCII format.

- ▶ If you want to use the Ethernet interface for transferring the service pack from a PC, proceed as follows:
 - While in the **Programming and Editing** operating mode, press the PGM MGT key.
 - Press the **NETWORK** soft key.
 - Select the PC to be connected with the arrow keys and press the **CONNECT DRIVE** soft key.
 - To exit the list of network drives, press the **END** soft key.
 - To exit the program manager, press the **END** soft key.
- ▶ While in the **Programming and Editing** operating mode, press the MOD key.
- ▶ Press the **LOAD SP** soft key.
- ▶ In the upper part of the window, select the folder containing the *.zip file of the service pack. The folder contents are displayed in the lower part of the window.
- ▶ To switch to the lower part of the window, press the **FILES** soft key. To return to the upper part of the window, use the **PATH** soft key. In the lower part of the window, select the *.zip file of the service pack with the cursor keys and press ENT. The installation of the service pack starts.
- ▶ Service pack installation finishes.



Note

When installing a service pack, it is not necessary to convert the binary format to ASCII format or vice versa, or to make a back-up of the non-volatile operands.

2.7.8 Special features of the software

Firmware

HSCI components have their own firmware, which must match the currently installed NC software version. Every time the control software is started, the NC software checks the firmware versions of the individual HSCI components. After a software update, during initial operation or after replacing a hardware component a firmware update may be required.



Note

The control must not be switched off during a firmware update!

If the NC software detects that a firmware update is necessary, you will be prompted by a dialog box to start the firmware update. Press the **Start update** button in the dialog box for the update to be run automatically. The firmware updates are included in the NC software update package. When the firmware versions of all HSCI participants have been updated, the control boots until the "Power interrupted" message appears.

After the firmware update you are prompted to switch off the control via the main switch and leave it off for 60 seconds. This time must be maintained to ensure that all components concerned are completely reset. For technical reasons, some of the individual HSCI components must be updated successively. This makes it necessary to switch off the control repeatedly via the main switch.

Configuration

During startup, a control in an HSCI system expects the complete configuration of the HSCI system in the form of an IOC file. The IOconfig software for PCs is used to configure the system, the PLC input/output systems (PL6xxx, UEC11x, MB machine operating panel, etc.) and the PROFIBUS components. The inputs and the keys of the handwheel cannot be configured freely. The key information of the handwheels continues to be mapped on to the known PLC markers.

With IOconfig, the components of an HSCI/PROFIBUS system are specified in a project and their arrangement is described. The actual sequence of the components must be identical with the arrangement of the components in the IOC file. An MB machine operating panel or a PLB 6001, which is connected to X501, must be arranged as the last participant of the HSCI system. On the basis of this data, IOconfig generates the IOC file (*.IOC) that contains all relevant configuration data for the HSCI system. Then you transfer the IOC file to the control. Enter the path and name of the project file (e.g. **IOCCFG = PLC:\IOC\Structur.ioc**) in **IOCCFG =** in the OEM.SYS file. When the control is next booted, the file is read in and evaluated by the control.



Note

All information required for using the IOconfig PC software and configuring HSCI and PROFIBUS components is contained in the IOconfig Technical Information, which is available for registered customers from the HEIDENHAIN HESIS-Web including Filebase on the Internet.

During start-up of the control, the nominal configuration of the IOC file is compared with the actual configuration. If there is a difference, you will be informed of it in a dialog box ("Hardware changed" dialog). Press the **Accept** button to confirm the new configuration and continue start-up of the control.

The "Hardware changed" dialog appears when, in comparison with the last boot-up of the control

- one or more devices have been added,
- one or more devices have been removed,
- the sequence of the HSCI participants has changed, or
- on one or more devices the "identification data" have changed. The identification data include the following:
 - All available information from the electronic keyboard (e.g. device type, ID)
 - All serial numbers
 - All firmware versions, if the firmware is permanently programmed on the device (e.g. firmware of the PL, MB)
 - All configuration information of an IOC file (e.g. HSCI address, slot number)

As a rule the dialog appears after an intentional change, e.g. after exchange of a component for repair or after a firmware update after software installation. If the dialog appears without a known change, this indicates an unintentional change, e.g. loss of an HSCI connection, power supply failure, or a defect in an HSCI assembly. For this reason, displayed changes must always be checked by qualified personnel!



Danger

If the "Hardware changed" dialog appears, the change has to be checked and, if necessary, accepted!

The control also boots without the IOC file or the entry "IOCCFG =" in the OEM.SYS. The control then automatically detects all components connected to the HSCI chain and the current configuration—without the PLC input/output systems, however. Press the **Accept** button in the dialog box to confirm the detected configuration and continue start-up of the control. However, further configuration of the control is not possible. In the **Programming and Editing** mode of operation, you can start BUS DIAGNOSIS after pressing the MOD key and the DIAGNOSIS soft key. The control will display the structure of the detected HSCI system. If required, you can use this information for the configuration with IOconfig.



Note

HEIDENHAIN recommends updating all IOC files and EAZ projects with the IOconfig version 2.2.02. This is the only way to make sure that you can use all of the features provided by the new HSCI components, see page 291.



2.7.9 HEROS 5 operating system

Boot manager

A selection menu for selecting the operating mode is provided temporarily when the control is booting. If you press one of the arrow keys of the ASCII keyboard, the booting process is interrupted at this point and the control expects you to select an operating mode ("runlevel"). If the booting process is not interrupted at this point, the control continues booting in **Runlevel 5** and starts the NC control software. You can select one of the following operating modes:

- **Runlevel 5: Start Control** (standard mode of operation)
After the HEROS 5 operating system has booted, the NC control software is started automatically. The iTNC 530 is started in the standard condition.
- **Runlevel 4: Service Mode (graphics)**
This operating mode is only for trained service personnel!
After the HEROS 5 operating system has booted, the booting process is stopped. The NC control software is not started. The iTNC 530 is now in the graphic desktop of the operating system.
- **Runlevel 3: Service Mode (text)**
This operating mode is only for trained service personnel!
After the HEROS 5 operating system has booted, the booting process is stopped. The NC control software is not started. The iTNC 530 is in a text console of the operating system.

Select the desired operating mode with the arrow keys and press the ENTER key to start it.

HE Logging

In controls with HSCI, the TRACE settings in MP7691.x are replaced by the **HE Logging** settings. These are settings for internal diagnostic files (FILO memory) for finding errors. These files can only be evaluated by HEIDENHAIN. You can access **HE Logging** as follows if HEIDENHAIN service personnel requests you to do so:

- ▶ Make the taskbar at the bottom of the screen visible by moving the mouse pointer over it.
- ▶ Press the green HEIDENHAIN button to open the JH menu.
- ▶ Select the **Diagnostics** menu item.
- ▶ Select the **HE Logging** menu item.

This takes you to the **HE Logging** menu containing tabs for possible settings for **Syslog**, **FTrace** and **TCPDump**. These possibilities for diagnostics and the associated settings are to be used only together with HEIDENHAIN service personnel.

Proceed as follows if HEIDENHAIN requests you to send one of the diagnosis files:

- ▶ Open the corresponding tab **Syslog**, **FTrace** or **TCPDump**.
- ▶ Under **View** at the bottom of the respective tab, select the file requested by HEIDENHAIN.
- ▶ Press the **Copy** button to the right of the file selection in the **View** settings. This opens the **Savepath...** file browser with which you can select the target folder for the diagnostic file. Keep in mind that the file is saved in the folder that is open in the file browser.



- ▶ Press the **Copy** button to save the diagnostic file in the open target folder. There you will later find the diagnostic file, can copy it from the control (e.g. with TNCremo) and send it by e-mail.

The folder **mnt** under **file system** is the folder that contains the HEIDENHAIN partitions TNC, PLC and SYS.

Tools

The HeROS 5 operating system provides additional software tools that make it possible to also open external file types and edit some of these file types. These tools are not software tools from HEIDENHAIN. They are tools from external suppliers. HEIDENHAIN will not provide any support or guarantee for the proper functioning of these tools. The following tools are available:

- File manager
File manager of the operating system
- Mousepad
Text editor for opening and editing TXT and INI files
- PDFViewer
Viewer software for opening PDF files
- Ristretto
Viewer software for opening GIF, JPG, PNG and BMP files
- Gnumeric
Editor for opening and editing XLS and CSV files
- Firefox
Web browser for opening HTML files

If file types that are not known to the NC control software are selected in the file manager of the control, then the operating system automatically uses the corresponding tools to open the file types it knows. The tools themselves can only be opened in the PLC mode of operation. Proceed as follows:

- ▶ While in the Programming and Editing operating mode, press the MOD key.
- ▶ Enter the code number 807 667 to switch to the PLC operating mode.
- ▶ Make the taskbar at the bottom of the screen visible by moving the mouse pointer over it.
- ▶ Press the green HEIDENHAIN button to open the JH menu.
- ▶ Select the **Tools** menu item.
- ▶ Start the desired tool.

Operating-system password

Certain accesses to or changes in the operating system are password-protected. The code number 807667 or the PLCPASSWORD defined in the OEM.SYS are required for these areas.

2.7.10 Functional safety (FS)

The HSCI hardware also makes it possible to set up HEIDENHAIN control systems with integrated functional safety (FS). However, the full range of features will not yet be available in the first software version for the iTNC 530 HSCI. This Technical Manual describes the HSCI hardware for systems with and without functional safety. Components for systems with functional safety have the identifier FS, e.g. MB 620 FS, PLB 6204 FS.



Note

Before planning a machine with functional safety, please inform yourself of whether the current scope of functional safety features suffices for your machine design.

You will find the special features and limitations of the hardware and software for functional safety in the Technical Manual for the iTNC 530 with functional safety (FS). Your contact person at HEIDENHAIN will be glad to answer any questions concerning the iTNC 530 HSCI with functional safety.



Note

NC software 606 42x has not yet been approved for applications that use the integrated functional safety (FS) of the control. Separate approval is required for the use of integrated functional safety (FS) according to EN ISO 13849-1!

2.7.11 Monitoring hardware changes

The iTNC 530 HSCI automatically monitors and checks whether hardware changes to control components took place. If the control detects a hardware change, it displays a dialog window after start-up. The control operator must confirm the hardware change(s).

To accept a control hardware change:

- ▶ Press the right arrow key to navigate to the **Accept** button. Then press the ENT key:

To reject a control hardware change:

- ▶ Press the **Reject** button. The control will continue booting and start in the Programming Station mode. When the control is next booted, the dialog box will be displayed again.



2.7.12 SELinux

SELinux is an expansion for Linux-based operating systems. SELinux is an additional safety software in the sense of Mandatory Access Control (MAC) and protects the system against the execution of unauthorized processes or functions, i.e. viruses and other malicious software. MAC means that every action must be explicitly permitted, otherwise it will not be executed. The software is intended as protection in addition to the normal access restriction in Linux. Certain processes and actions can only be executed if the standard functions and access control of SELinux permit it. SELinux is only available with NC software version 606 42x.

The SELinux installation of the TNC is prepared to permit running of only those programs installed with the HEIDENHAIN NC software. Other programs cannot be run with the standard installation.

The directory itself and the data in the **PLC:\etc\selinux** directory should not be deleted. These files are directly related to the function of the SELinux safety software.

The access control of SELinux under HEROS 5 is regulated as follows:

- Only files that are installed with the HEIDENHAIN NC software can be executed.
- Files in connection with the safety of the software (SELinux system files, HEROS 5 boot files etc.) may only be changed by programs that are selected explicitly.
- New files generated by other programs must never be executed.
- There are only two processes that are permitted to execute new files:
 - Starting of a software update
A software update from HEIDENHAIN can replace or change system files. Software updates may only be downloaded from HESIS-Web Including FileBase. Install a software update only if you can trust the source of the files (e.g. USB stick).
You can prohibit software updates via the configuration of SELinux.
 - Starting of the SELinux configuration
The configuration of SELinux should be protected by a password.

HEIDENHAIN generally recommends activating SELinux because it provides additional protection against attacks from outside. When commissioning the machine you should first install all of the required components (PLC program, OEM help files, Python scripts etc.) before you activate SELinux. When the control boots, it will only display a short note then.

In rare cases it may be necessary to check the application of SELinux:

- Python modules or libraries that are not included in the HEIDENHAIN standard installation or were modified by the user. In such cases the user's changes must be checked and corrected, if necessary.
- OEM specific applications with executable files. In such cases the application of SELinux must be tested more thoroughly and possibly be adapted.

In these two cases, SELinux must be deactivated temporarily if Python scripts or executable programs are to be transferred to the control via the file manager, TNCremo or TNCbackup. Afterwards the file system must be relabeled (Relabel Files) and SELinux must be activated again.

SELinux configuration:

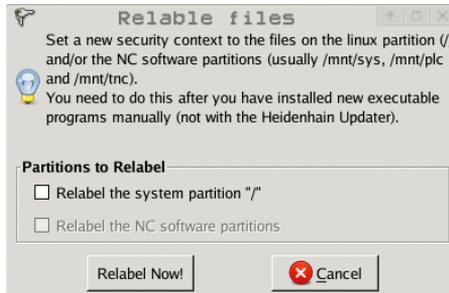
- ▶ Make the taskbar at the bottom of the screen visible by moving the mouse pointer over it.
- ▶ Press the green HEIDENHAIN button to open the JH menu.
- ▶ Select the Settings menu item.
- ▶ Select the SELinux menu item.



Menu item		Function
Password Protection		
	Set Password	Enter a password for the SELinux configuration. HEIDENHAIN recommends protecting the settings for SELinux with a password. Please note that the password needs to be entered again before it can be changed or deleted. If the password is lost, the storage medium must be replaced before access to the settings is permitted again.
	Delete password	Delete password for the SELinux configuration.
Security Policy NOT active (System is not protected)		Deactivate SELinux
	Don't warn if SELinux is not active	Do not display the message anymore.



Menu item		Function
Security Policy ACTIVE		
	Allow software updates	NC software update via setup.elf file permitted.
	Allow openVPN tunnel for remote service	Data exchange via a VPN tunnel permitted.
	Allow program execution from VFAT	Permit execution of files or programs from a storage medium with VFAT formatting. Software version 60642x-01 was shipped with VFAT formatting on the hard disk and SSDR. During a software update from 60642x-01 to 60642x-02 the formatting of the data carrier remains VFAT and this menu item is activated automatically. Please note that USB sticks, for example, are usually also VFAT formatted, meaning that any programs can be executed from such data carriers! If the item is not set automatically, it must be activated manually if the option of the encrypted PLCE partition is being used and Python programs located on PLCE: are to be started.
	Allow NC to change HeROS config files	Permit changes to the operating system files by HEIDENHAIN PC software. This menu item must be active if the Backup and Restore functions are to be executed with the TNCremo PC tool.
Relabel Files		If Python scripts or executable programs were transferred to the control, the file system must be relabeled and SELinux must be activated again.



Menu item		Function
Partitions to Relabel		
	Relabel the system partition "/"	Sets a new label only for the SYS partition
	Relabel the NC software partitions	Sets a new label for the NC partitions SYS, PLC and TNC

HEIDENHAIN recommends activating SELinux in the default configuration and protecting the settings for SELinux by an individual password:

- Activate the **Security Police**
- Activate the options **Allow software updates, Allow openVPN tunnel for remote service**
- Deactivate the options **Allow program execution from VFAT, Allow NC to change HeROS config files**

This way SELinux prevents unknown programs from being executed. From the point of view of HEIDENHAIN, this ensures sufficient protection of the control system and its software from e.g. viruses and malicious software.

Installing additional virus scanners on the control is not necessary or advisable from the point of view of HEIDENHAIN because it may cause problems with the execution of the actual NC software or NC programs.

NC updates from HEIDENHAIN or updates that are created with PLCdesign have a signature for identification. Only updates with such a signature can be executed by activating the **Allow software updates** option. After an update the newly installed programs are relabeled automatically—these may then be executed. If the option is not enabled this has the effect that the end user can no longer run NC software updates without entering the SELinux password (the option would have to be enabled again temporarily). The same applies to automatic updates via USB sticks.

2.7.13 Data backup

For data backup, HEIDENHAIN offers the PC software TNCbackup free of charge. TNCbackup provides convenient functions for backing up and restoring data. TNCbackup is part of the TNCremoNT software package and can be downloaded from the HEIDENHAIN HESIS-Web including Filebase on the Internet. (<http://portal.heidenhain.de>)

HEIDENHAIN recommends that the machine manufacturer use the TNCbackup software to save all his machine-specific data and supply them with the machine (e.g. on CD-ROM).

The customer, too, can save his data before exchanging the control. It is also advisable that the customer save all of the files and programs created on the control at regular intervals on a PC. The CD-ROM supplied with the machine should therefore also contain the current version of TNCremoNT.

2.8 Released NC software

The following NC software was released for the iTNC 530 with HSCI and MC 6xxx/CC 6xxx:

- NC software 606420-01 and 606421-01 July 2010
- NC software 606422-01 and 606423-01 July 2010
- NC software 606420-02 and 606421-02 December 2011
- NC software 606422-02 and 606423-02 December 2011
- NC software 606420-03 and 606421-03 April 2013
- NC software 606422-03 and 606423-03 April 2013



3 Mounting and electrical installation

3.1 General information



Attention

Remember the following during mounting and electrical installation:

- National regulations for power installations
- Interference and noise immunity
- Operating conditions
- Mounting attitude

3.1.1 Safety precautions



Danger

Ensure that the main switch of the control or machine is switched off when you engage or disengage connecting elements or connection clamps.



Danger

Ensure that the grounding conductor is connected. Interruptions in the equipment grounding conductor may cause damage to persons or property.



Danger

Incorrect or non-optimized input values can lead to faulty machine performance and therefore to serious injury to persons and damage to equipment. Modifications of the machine configuration should be done with caution and uncontrolled axis motions should be taken into account.



Attention

In order to be able to judge the behavior of an NC controlled machine, you need to have fundamental knowledge about drives, inverters, controls and encoders. Inappropriate use may cause considerable damage to persons or property.

HEIDENHAIN does not accept any responsibility for direct or indirect damage caused to persons or property through improper use or incorrect operation of the machine.



Danger

The interfaces for the PLC inputs/outputs, machine operating panel and PL connection comply with the requirements for basic insulation in accordance with **IEC 742 EN 50 178**.

Only units that comply with the requirements of **IEC 742 EN 50 178** for basic insulation may be connected, otherwise damage to persons or property may be caused. The maximum dc voltage mean value of the PLC inputs is 31 V.

3.1.2 Degrees of protection

The following components fulfill the requirements for IP54 (dust and splash-proof protection).

- Visual display unit (front, when properly installed)
- Keyboard unit (front, when properly installed)
- Machine operating panel (front, when properly installed)
- Handwheel (built-in or connected, depending on the type)

All electric and electronic control components must be installed in an environment (e.g. electrical cabinet, housing) that fulfills the requirements of protection class IP54 (dust and splash-proof protection) in order to fulfill the requirements of contamination level 2.



Note

All components of the OEM operating panel must also comply with protection class IP54, just like the HEIDENHAIN operating panel components.

3.1.3 Electromagnetic compatibility

This unit fulfills the requirements for EN 50370-01 and is intended for operation in industrially zoned areas.

Protect your equipment from interference by observing the following rules and recommendations.

Likely sources of interference

Interference is mainly produced by capacitive and inductive coupling from electrical conductors or from device inputs/outputs, such as:

- Strong magnetic fields from transformers or electric motors
- Relays, contactors and solenoid valves
- High-frequency equipment, pulse equipment and stray magnetic fields from switch-mode power supplies
- Power lines and leads to the above equipment

Protective measures

- Keep a minimum distance of 20 cm from the control and its leads to interfering equipment.
- Keep a minimum distance of 10 cm from the control and its leads to cables that carry interference signals. For cables in metallic ducting, adequate decoupling can be achieved by using a grounded separation shield.
- Shielding according to EN 50 178.
- Use equipotential bonding conductors with a cross section of 6 mm²
- Use only genuine HEIDENHAIN cables, connectors and couplings



3.1.4 ESD protection

Always assume that all electronic components and assemblies are endangered by electrostatic discharge (ESD).

To ensure protection from ESD, follow the precautionary measures described in IEC 61340-5-1, IEC 61340-5-2 and IEC 61340-4-1.



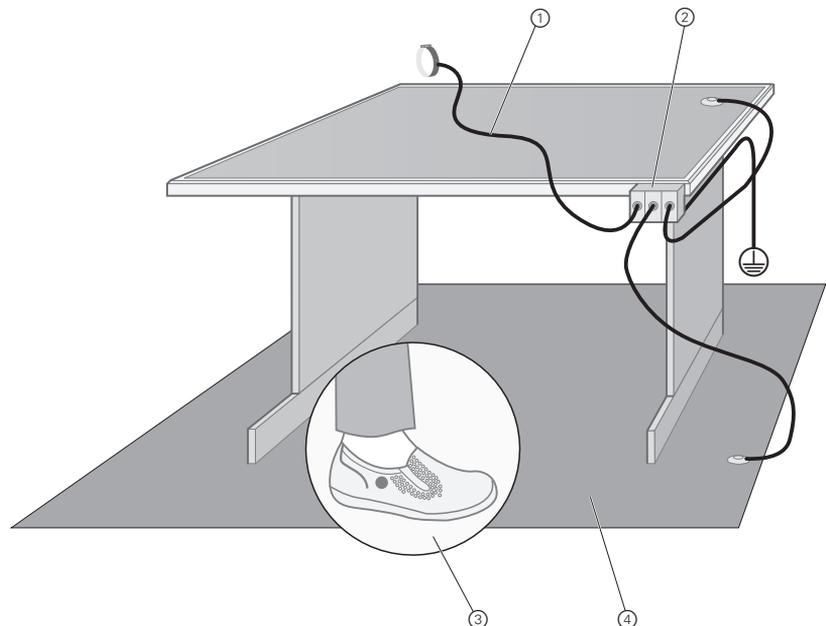
Note

Improper handling can result in damage to the components or assemblies due to ESD!

The following are some points covered in the above mentioned standards:

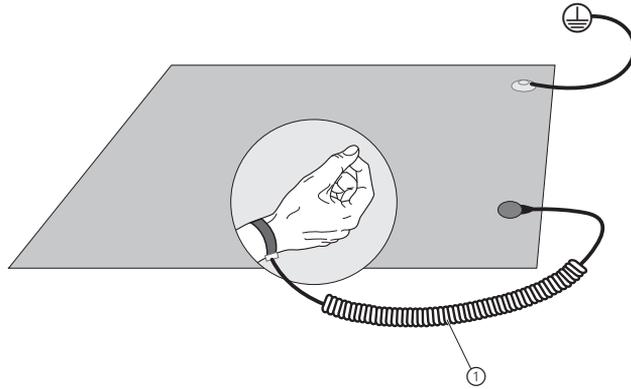
- When handling electrostatically endangered components or assemblies (e.g. exchange, installation, shipping), always comply with the precautionary measures described in IEC 61340-5-1, IEC 61340-5-2 and IEC 61340-4-1.
- Store and transport ESD-sensitive components in ESD protective containers.
- Ensure during handling the proper grounding of the working area (e.g. tool, workbench, packaging) and the person.
- Inspect the ESD protection system regularly.

The following figure shows how a suitable working area could look in accordance with IEC 61340.



- 1: Wristband with 1 MOhm grounding cable for grounding the person
- 2: Grounded connection for wristbands, floor mats, table mats etc. for equipotential bonding
- 3: Dissipative shoes
- 4: Dissipative flooring or floor mat

An important part of the working area is a suitable working surface with a wristband with 1 MOhm grounding resistance for personal grounding:



3.2 Environmental conditions

3.2.1 Storage and operating temperatures

Limit values

Device	Air approaching the device in the panel / electrical cabinet	Temperature range outside the panel / electrical cabinet
MC 6xxx, MC 7xxx in panel with SSDR	0 °C to +50 °C	0 °C to +45 °C (no direct exposure to sunlight)
MC 63xx in panel	0 °C to +50 °C	0 °C to +45 °C (no direct exposure to sunlight)
MB 6xx, TE 6xx	0 °C to +50 °C	0 °C to +45 °C
MB 7xx, TE 7xx	0 °C to +50 °C	0 °C to +45 °C
HR 4xx, HR 5xx		0 °C to +45 °C
MC 6241, MC 6341, IPC 6341	+5 °C to +40 °C	
PLB 6xxx, PLD-H, PLA-H	+5 °C to +40 °C	
BF 2xx	0 °C to +50 °C	0 °C to +45 °C (no direct exposure to sunlight)
BF 7xx	0 °C to +50 °C	0 °C to +45 °C (no direct exposure to sunlight)
HDR hard disk	+5°C to +55°C	0 °C to +45 °C (no direct exposure to sunlight)
SSDR solid state disk	0 °C to +70 °C	0 °C to +45 °C (no direct exposure to sunlight)
CompactFlas h card	0 °C to +70 °C	0 °C to +45 °C (no direct exposure to sunlight)

Conditions for integrating the MC in the operating panel

Because the MC is installed in the operating panel, additional heat is generated that must be emitted to the surroundings through the panel wall. The required temperature difference depends on the design and the size of the panel.

The maximum permissible temperature of the air surrounding the panel is 45 °C. However, the temperature of the air inside the panel, which flows into the HEIDENHAIN devices (e.g. the MC), is decisive for the HEIDENHAIN devices integrated in the panel.



Attention

The temperature of the air inside the panel, which flows into the HEIDENHAIN devices, must not exceed a maximum temperature of +50 °C.

Ensure that the operating panel is designed in such a way that the air approaching the MC does not exceed a max. temperature of +50 °C. Please verify this through an appropriate temperature measurement.

With the recommended temperature limits, active cooling is not required for a typical panel. If the maximum permissible temperature for the integrated components is exceeded, however, HEIDENHAIN recommends using heat exchangers. They ensure appropriate heat dissipation for the components in the operating panel.



Note

HEIDENHAIN recommends against using fans with filter units for dissipating the heat in the operating panel. Due to filter contamination problems they are often not capable of ensuring the required heat dissipation.

Humidity during operation



Attention

Condensation on the electronics is not permitted!

Condensation can form, for example, if warm, moist air flows along cool surfaces of the electronics. Therefore, cooling units with discontinuous (e.g. two-position) temperature control must not be used. The resulting cyclic changes in temperature and humidity can cause condensation on the cool surfaces of the electronics. Furthermore, you must prevent condensate from reaching the electronics and ensure that it can drain away without causing damage.

In tropical areas it is recommended that the control not be switched off, so that condensation is avoided on the circuit boards.

Storage temperatures

For all control components: -20 °C to +60 °C.

NiMH rechargeable batteries in the HR 550FS: -20 °C to +21 °C at a charge status of 40 %.

Humidity during storage of NiMH rechargeable batteries

NiMH rechargeable batteries in the HR 550FS: Relative humidity $5 < 60$ % rF

3.2.2 Heat generation and cooling

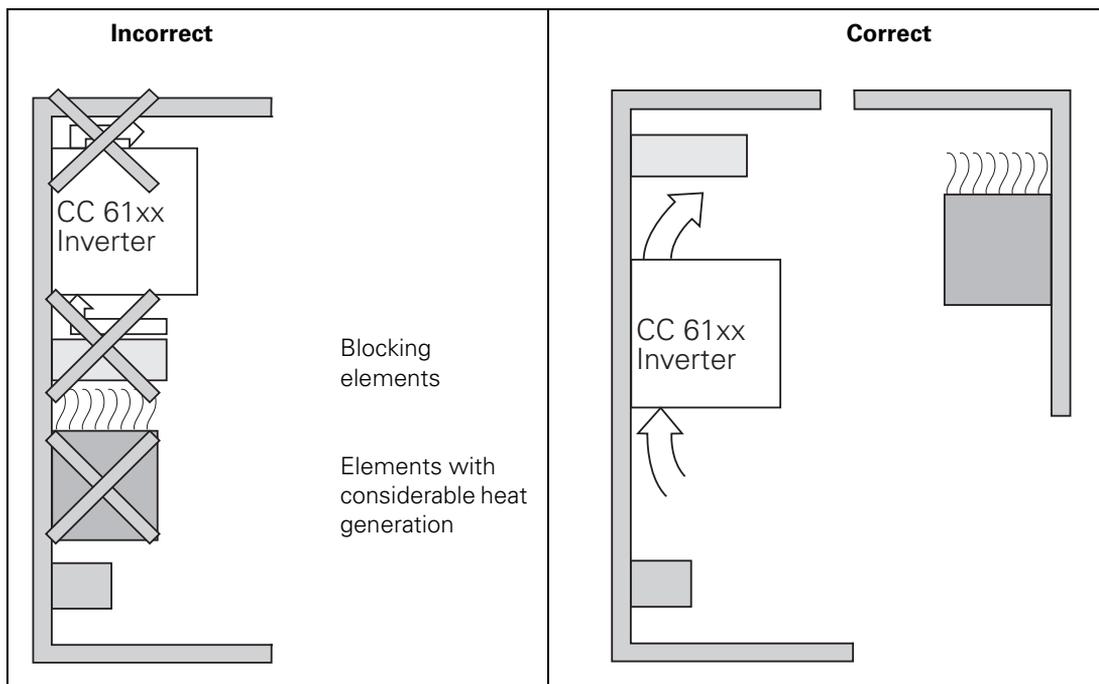
A heat exchanger or a cooling unit is preferable for controlling the internal temperature of the electrical cabinet.

If filtered air is blown into the electrical cabinet for cooling purposes, the standard IEC 61800-5-1 applies, which permits contamination level 2.



Danger

Be sure to take the measures required for preventing dust or water from entering the electrical cabinet or the housing. Dust depositing inside electrical devices may cause them to fail and impair the safety of the system. Max. contamination level 2 is permitted for the components.



3.2.3 Limit values for ambient conditions

HEIDENHAIN specifies the range of application 2 for the use of its control products.

Furthermore, the following limit values apply:

Characteristic values during operation:	Limit values to be maintained	Standard to be complied with
Vibration	± 0.075 mm, 10 Hz to 41 Hz; 5 m/s ² , 41 Hz to 500 Hz;	DIN EN 60068-2-6
Shock	50 m/s ² , 11 ms	DIN EN 60068-2-27
Relative air humidity	75 % in continuous operation; 95% for not more than 30 days a year (randomly distributed)	
IP protection	IP 54 for HW, TFT, MB, TE	DIN EN 60529
ESD 61000-4-2	Severity level 3	DIN EN 6100-4-2
RF field 61000-4-3	Severity level 3	DIN EN 6100-4-3
Burst 61000-4-4	Severity level 3	DIN EN 6100-4-4
Surge 61000-4-5	Severity level 3	DIN EN 6100-4-5
Conducted disturbances	Severity level 3	DIN EN 6100-4-6

3.2.4 Installation elevation

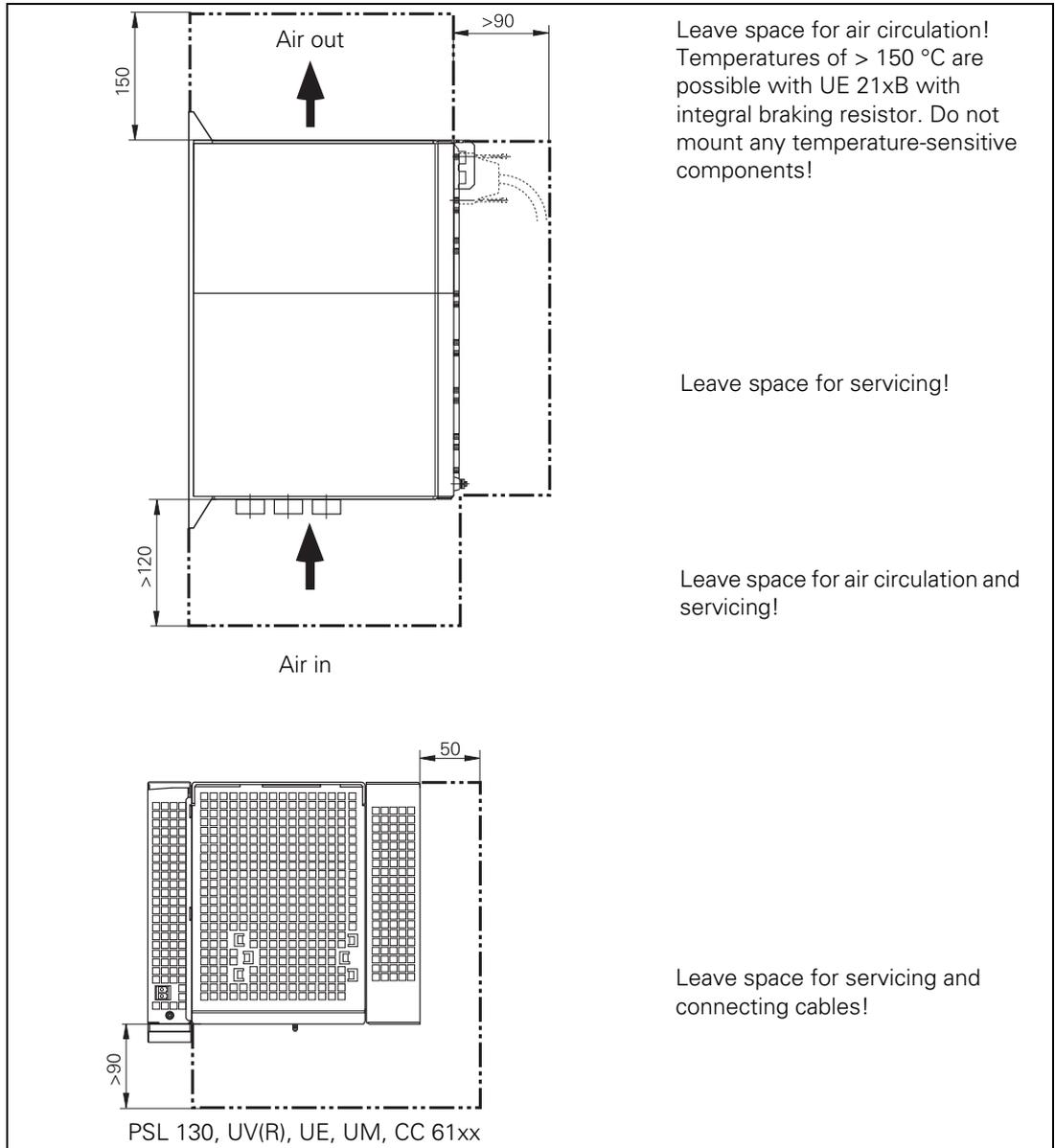
The maximum elevation for installation is 3000 m above sea level.

3.2.5 Mounting attitude MC 6xxx, CC 61xx, UV xxx, UM xxx, UE 2xx B



Attention

- When mounting, please observe proper minimum clearance, space requirements, length and position of the connecting cables.
- Please ensure during installation that the components are properly screwed to the wall of the electrical cabinet by using all of the provided cutouts (holes). For example, the MC 6x4x, CC 61xx each provide four cutouts for fastening.

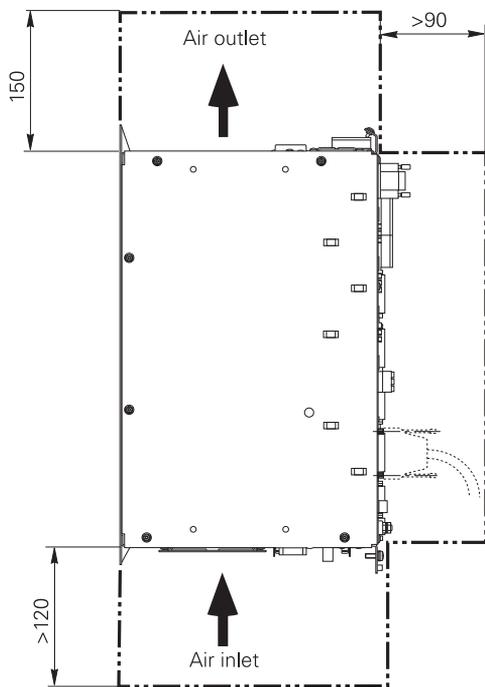


3.2.6 Mounting attitude of UMC 111 (FS)



Attention

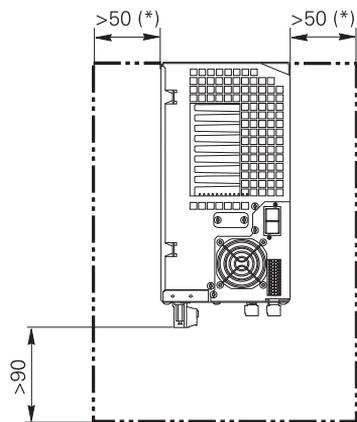
When mounting, please observe proper minimum clearance, space requirements, length and position of the connecting cables.



Leave space for air circulation!

Leave space for servicing!

Leave space for air circulation and servicing!



UMC 111 (FS)

(*) Recommended free space for air circulation >50 mm next to the last HEIDENHAIN component in the combination.

Leave space for servicing and connecting cables >90 mm!

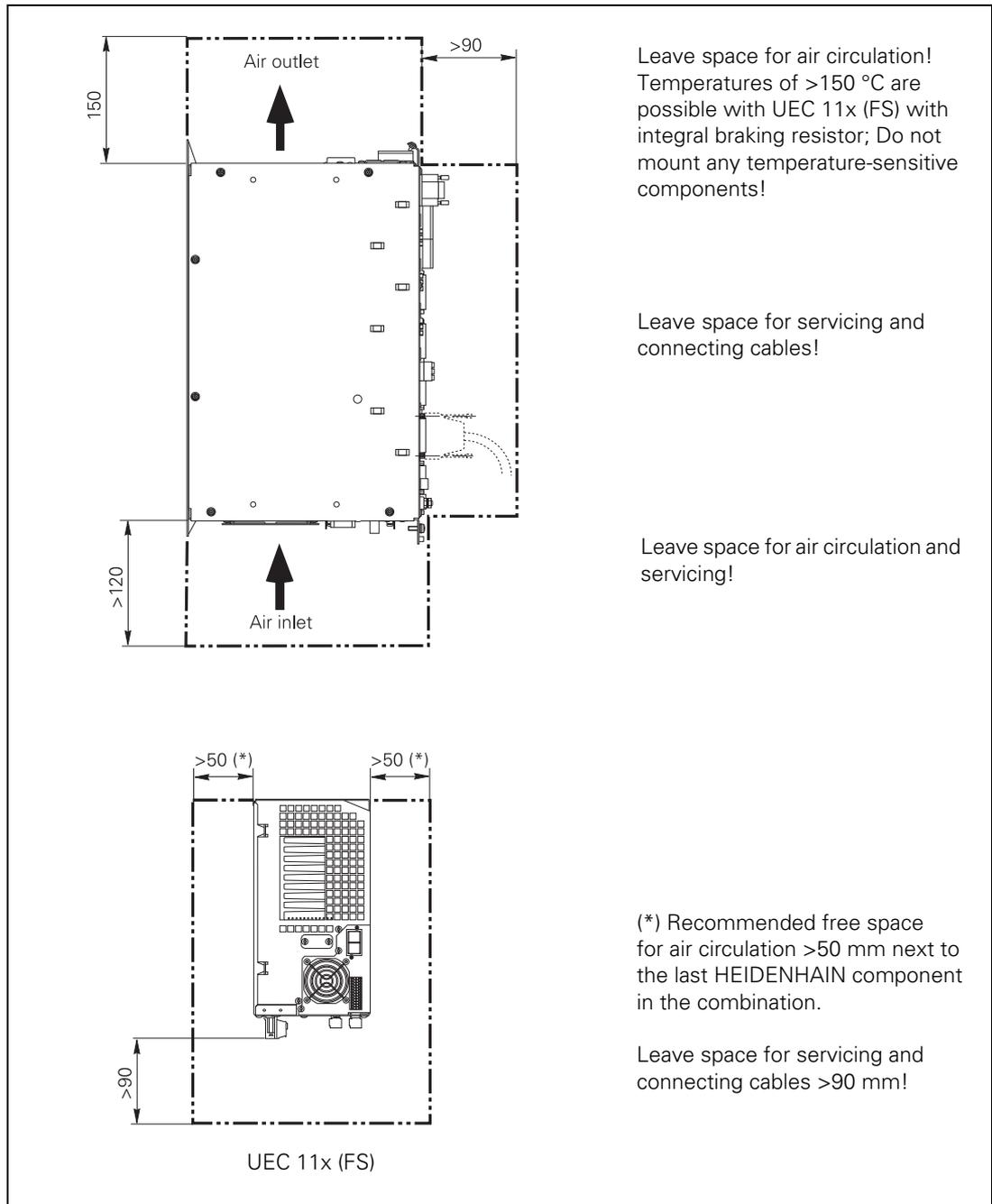


3.2.7 Mounting attitude of UEC 11x (FS)



Attention

When mounting, please observe proper minimum clearance, space requirements, length and position of the connecting cables.

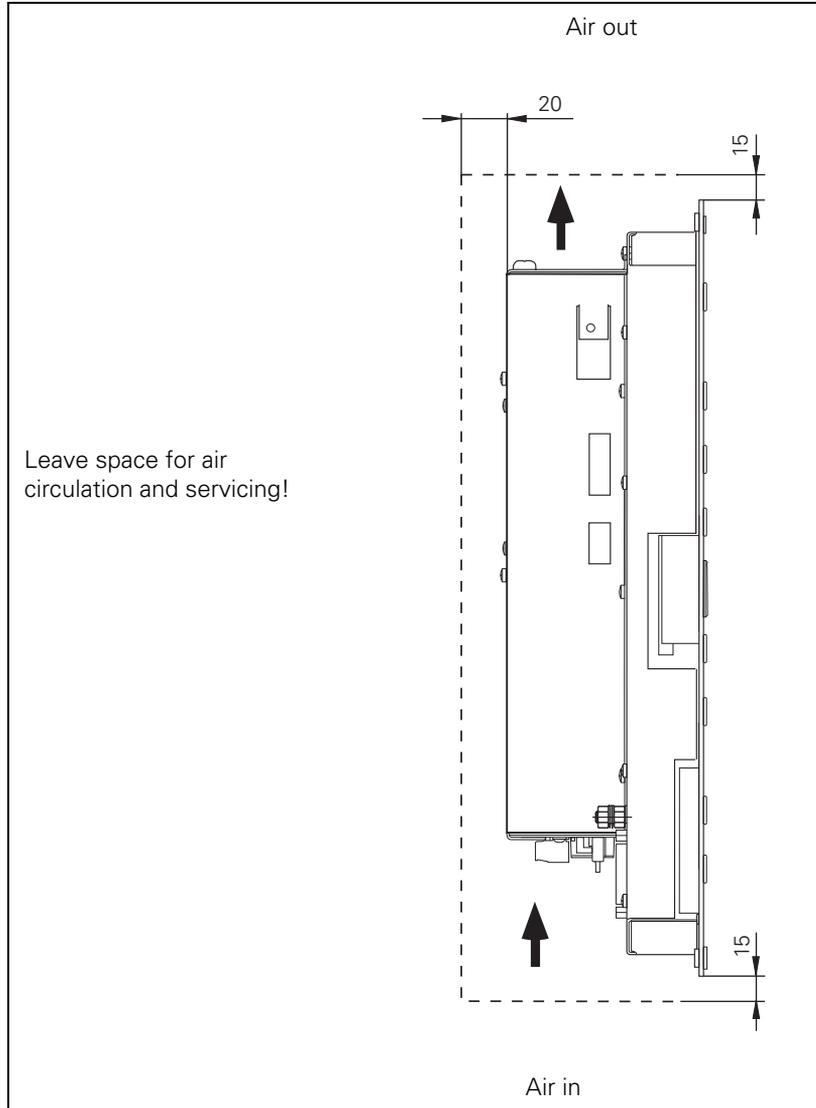


3.2.8 Mounting attitude of MC 6222, MC 7222



Attention

When mounting, please observe proper minimum clearance, space requirements, length and position of the connecting cables.

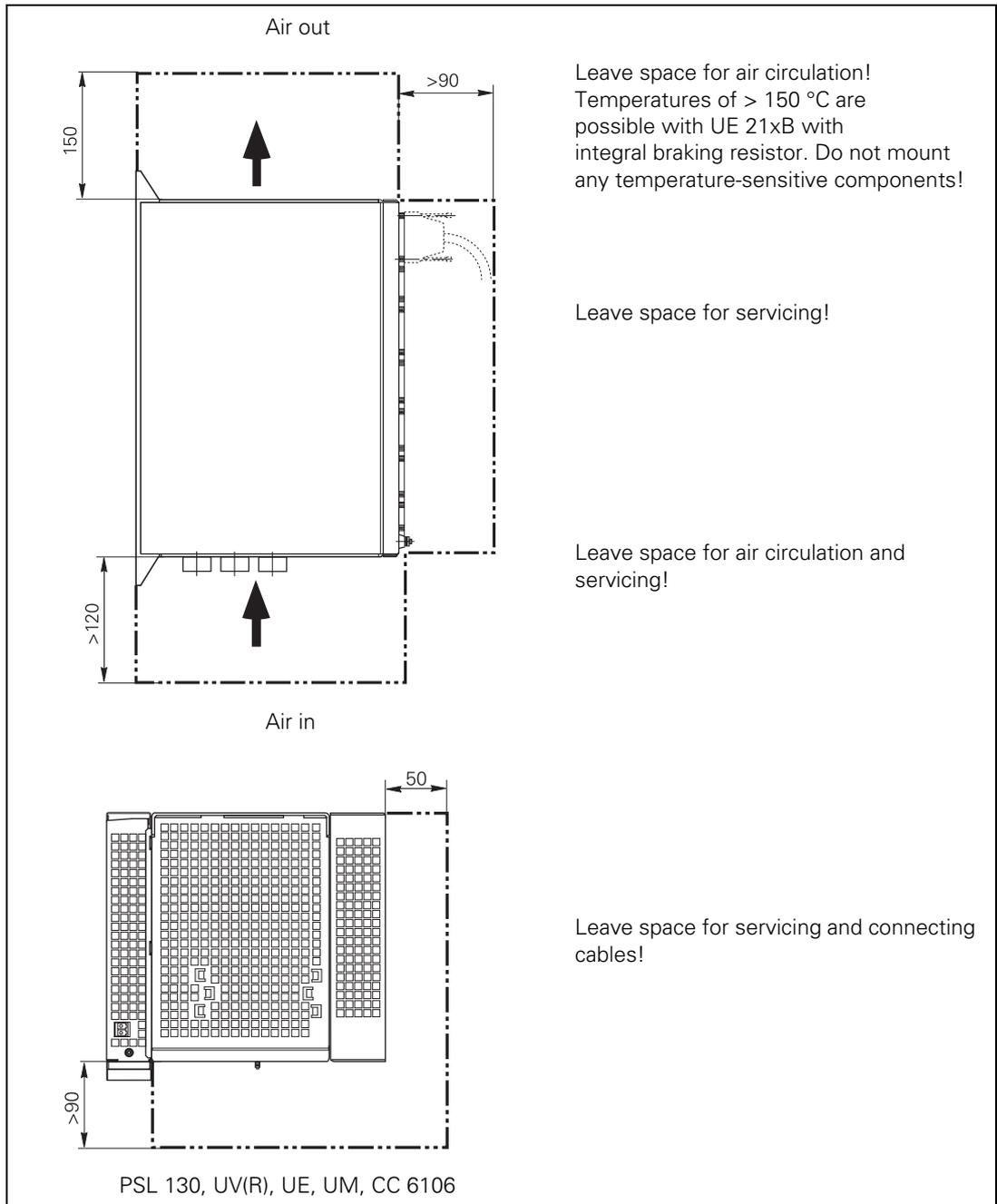


3.2.9 Mounting attitude of CC 6106, UV xxx, UM xxx, UE 2xx B(D)



Attention

When mounting, please observe proper minimum clearance, space requirements, length and position of the connecting cables.

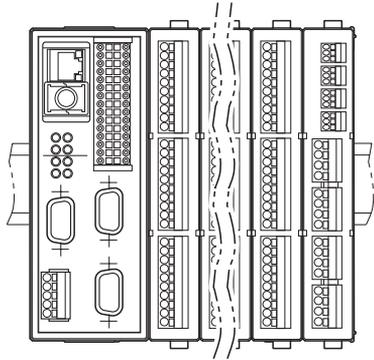


3.2.10 Mounting attitude of PLB 6xxx

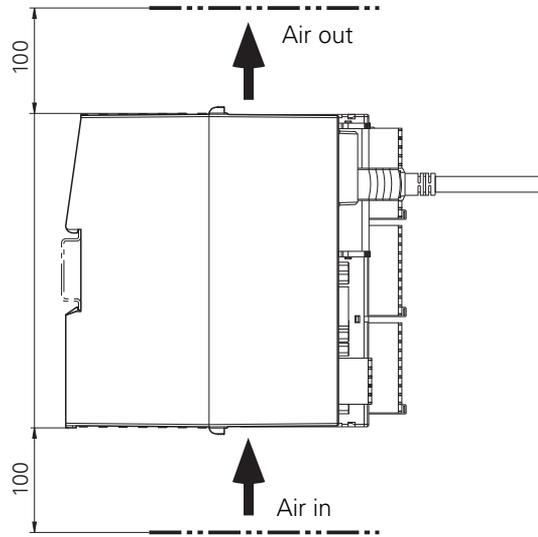


Attention

When mounting, please observe proper minimum clearance, space requirements, length and position of the connecting cables.



PLB 6xxx



Leave space for air circulation and servicing!



3.3 Handling the HDR hard disk

Shipping brace of the hard disk

The HDR hard disks of the MC 6x4x are equipped with a shipping brace. Before putting the iTNC 530 HSCI into service, the shipping brace of the hard disk must be removed.



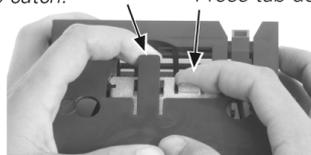
Attention

Do not transport the HDR with the MC 6x4x after you have installed the HDR. If the entire machine is being transported, or the MC is being transported inside the electrical cabinet, the shipping brace for the hard disk is usually not required. However, if the possibility exists that the hard disk could be subject to increased shock or vibration loads, then you must remove the hard disk from the MC for transport, reinstall the shipping brace in the HDR, and send the HDR separately in the original packaging.

Should servicing become necessary (i.e. the HDR is being shipped on its own), the hard disk must be secured with the shipping brace.

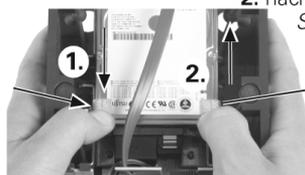
Festplatte entriegeln · *Unlocking the hard disk*

1. Sicherung anheben.
Lift the catch.
2. Lasche nach hinten drücken.
Press tab down.



Festplatte verriegeln · *Locking the hard disk*

1. hineindrücken, nach hinten schieben.
Press hard disk down, slide it backwards.
2. nach vorne schieben. (Click)
Slide it forwards. (click)



Griffflaschen benützen.
Use holding tabs.

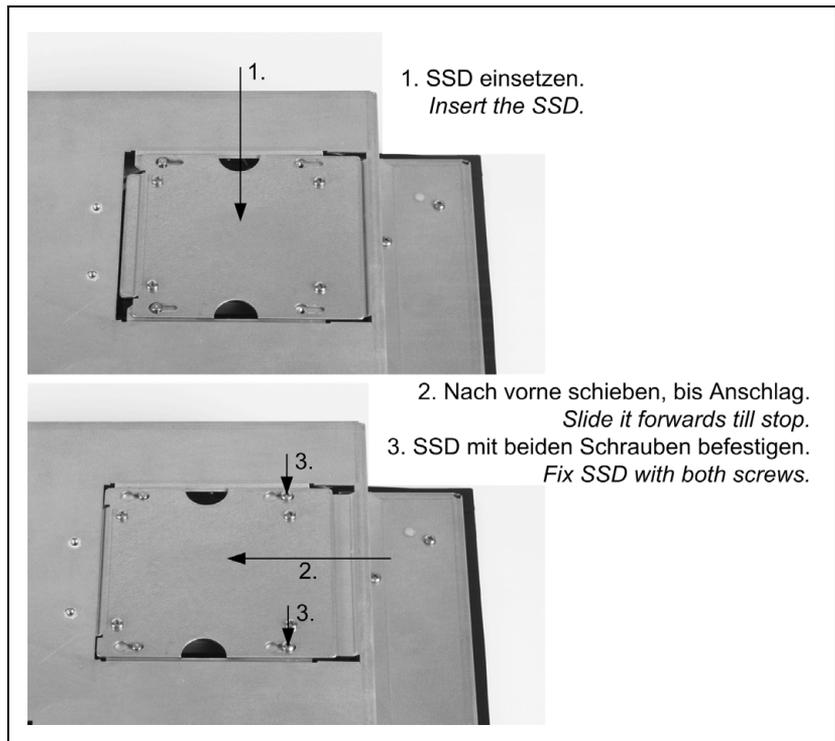


Note

When handling the HDR disk and the SIK, ensure electrostatic discharge protection. Improper handling can result in damage to the components or assemblies due to ESD!

3.4 Mounting the Solid State Disk (SSD)

A special shipping brace is not required for the Solid State Disk (SSD). The SSD is to be mounted as follows:



Note

When handling the Solid State Disk (SSD), ensure electrostatic discharge protection. Improper handling can result in damage to the components or assemblies due to ESD!

3.5 Overview of components

Hardware component		ID
MC 6241	Main computer 1.8 GHz with HDR, electrical cabinet version, without Profibus	573398-xx
MC 6241	Main computer 1.8 GHz with HDR, electrical cabinet version, with Profibus	653220-xx
MC 6222	Main computer with 15-inch TFT display, 1.8 GHz with SSDR, operating-panel version, without Profibus	634109-xx
MC 6222	Main computer with 15-inch TFT display, 1.8 GHz with SSDR, operating-panel version, with Profibus	634113-xx
MC 7222	Main computer with 15-inch TFT display, 1.8 GHz with SSDR, operating-panel version, without Profibus	806868-xx
MC 7222	Main computer with 15-inch TFT display, 1.8 GHz with SSDR, operating-panel version, with Profibus	806869-xx
MC 6341	Main computer with 15-inch TFT display, 2.2 GHz dual core with HDR, electrical-cabinet version	671226-xx
MC 6341	Main computer with 15-inch TFT display, 2.2 GHz dual core with HDR, electrical-cabinet version, with Profibus	735873-xx
HDR iTNC	Hard disk for MC 6x41, 160 GB, NC software 60642x-xx	682272-0x
HDR iTNC	Hard disk for MC 6x41 (export version), 160 GB, NC software 60642x-xx	682272-5x
SSDR iTNC	Solid State Disk for MC 6222 or MC 7222, 32 GB, NC software 60642x-xx	736591-0x
SSDR iTNC	Solid State Disk for MC 6222 or MC 7222 (export version), 32 GB, NC software 60642x-xx	736591-5x
SIK iTNC	SIK for MC 61x1, single-processor version	617776-01
SIK iTNC	SIK for MC 61x1, single-processor version (export version)	617776-51
SIK iTNC	SIK for MC 62xx, MC 63xx single-processor version	586084-01
SIK iTNC	SIK for MC 62xx, MC 63xx single-processor version (export version)	586084-51
BF 760	19-inch TFT display with HDL connection, stainless-steel design	732589-xx
BF 250	15-inch TFT display with HDL connection	599916-xx
BF 260	19-inch TFT display with HDL connection	617978-xx
IPC 6341	Industrial PC for installation in an electrical cabinet	649963-xx
HDR IPC	HDR hard disk for IPC 6341 with Windows 7 Embedded	599956-xx
CC 6106	Controller unit for HSCI for max. 6 control loops	662636-xx
CC 6108	Controller unit for HSCI for max. 8 control loops	662637-xx
CC 6110	Controller unit for HSCI for max. 10 control loops	662638-xx
Connecting cables	Ribbon cable for connecting four X69 connectors of the CC (with each other: two CC 6108, or two CC 6110, or one CC 6108 to one CC 6110)	325816-22

Hardware component		ID
Connecting cables	Ribbon cable for connecting 2 x X69 connectors of the CC 6106 (2 x CC 6106)	325816-24
UMC 111 FS	Controller unit with inverter and PLC for power supply via external DC link, 4 control loops, functional safety	664231-xx
UMC 111	Controller unit with inverter and PLC for power supply via external DC link, 4 control loops	736435-xx
UEC 111	Controller unit with inverter and PLC, 4 control loops	625777-xx
UEC 112	Controller unit with inverter and PLC, 5 control loops	625779-xx
MS 110	Mounting case for multi-row configuration	658132-xx
MS 111	Mounting case for multi-row assembly, additional connection for 24 V supply to the fan	673685-xx
TE 620	Keyboard unit without touchpad (15-inch)	625806-xx
TE 630	Keyboard unit with touchpad (15-inch)	617976-xx
TE 635Q	TE with touchpad and integrated MB for HSCI connection (15-inch)	617975-xx
TE 635Q FS	TE with touchpad and integrated MB for HSCI connection, functional safety (15-inch)	662255-xx
TE 645Q	TE with touchpad and integrated MB for HSCI connection (19-inch)	682104-xx
TE 645Q FS	TE with touchpad and integrated MB for HSCI connection, functional safety (19-inch)	685394-xx
TE 720	Keyboard unit without touchpad (15-inch), stainless steel	805488-xx
TE 730	Keyboard unit with touchpad (15-inch), stainless steel	805489-xx
TE 740	Keyboard unit with touchpad (19-inch), stainless steel	886546-xx
TE 735	Keyboard unit with touchpad and integrated MB for HSCI connection, 15-inch, stainless steel	771898-xx
TE 745	Keyboard unit with touchpad and integrated MB for HSCI connection, 19-inch, stainless steel	679817-xx
TE 735 FS	Keyboard unit with touchpad and integrated MB for HSCI connection, FS functional safety, 15-inch, stainless steel	805493-xx
TE 745 FS	Keyboard unit with touchpad and integrated MB for HSCI connection, functional safety (FS), 19-inch, stainless steel	805482-xx
PSL 130	Low-voltage power supply unit, 750 W, for +24 V NC and +24 V PLC	575047-xx
PSL 135	Low-voltage power supply unit, 750 W, for +24 V NC and +24 V PLC, also suitable for non-HEIDENHAIN systems	627032-xx
MB 620	Machine operating panel for HSCI connection	617973-xx
MB 620 FS	Machine operating panel for HSCI connection, functional safety	660090-xx
MB 720	Machine operating panel for HSCI connection (15-inch), stainless steel	784803-xx
MB 720 FS	Machine operating panel for HSCI connection, functional safety, 15-inch, stainless steel	805474-xx
PLB 6001	HSCI adapter for OEM-specific machine operating panel, 72 digital inputs, 40 digital outputs	668792-xx



Hardware component		ID
PLB 6001 FS	HSCI adapter for OEM-specific machine operating panel, functional safety (FS), 36 digital FS inputs, 40 digital outputs	722083-xx
PLB 6104	PLB for HSCI, 4 slots	591828-xx
PLB 6106	PLB for HSCI, 6 slots	630058-xx
PLB 6108	PLB for HSCI, 8 slots	630059-xx
PLB 6204	PLB for HSCI, 4 slots, with system module	591832-xx
PLB 6206	PLB for HSCI, 6 slots, with system module	630054-xx
PLB 6208	PLB for HSCI, 8 slots, with system module	630055-xx
PLB 6104 FS	PLB for HSCI, 4 slots, functional safety	590479-xx
PLB 6204 FS	PLB for HSCI, 4 slots, with system module, functional safety	586789-xx
PLB 6206 FS	PLB for HSCI, 6 slots, with system module, functional safety	622721-xx
PLB 6208 FS	PLB for HSCI, 8 slots, with system module, functional safety	620927-xx
PLD-H 16-08-00	PL for PLB 6xxx: 16 digital inputs, 8 digital outputs	594243-xx
PLD-H 08-16-00	PL for PLB 6xxx: 8 digital inputs, 16 digital outputs	650891-xx
PLD-H 08-04-00 FS	PL for PLB 6xxx FS: 8 digital inputs, 4 digital outputs, functional safety	598905-xx
PLD-H 04-08-00 FS	PL for PLB 6xxx FS: 4 digital inputs, 8 digital outputs, functional safety	727219-xx
PLA-H 04-00-04	PL for PLB 6xxx, 4 x \pm 10 V inputs, 0 analog outputs, 4 x PT 100 inputs	599070-xx
PLA-H 08-04-04	PL for PLB 6xxx, 8 x \pm 10 V inputs, 4 x \pm 10 V analog outputs, 4 x PT 100 inputs	675572-xx
PL empty housing	Empty housing for slots of a PL 6xxx	383022-11
CMA-H 04-04-00	SPI expansion module for CC/UEC for analog axes/spindles for HSCI, 4 analog outputs \pm 10 V	688721-01
HSCI cable	HSCI connecting cable	618893-xx
HDL cable	Connecting cable for connection to BF	625901-xx



Attention

In HSCI systems with integrated functional safety (FS) you may use only devices that have been certified for use in such systems. For components and devices permitted for use in systems with functional safety, please refer to the Technical Manual for Functional Safety (FS). Upon request, your contact partner at HEIDENHAIN can provide you with the manual.

If you want to design the control system in accordance with the new standard ISO 13849-1, you need the corresponding validation values of the individual control components for calculating the required performance level. This also applies to the inverters and power supply modules used in a system with HSCI but without integrated functional safety (FS). Only devices whose index or version number is greater than or equal to the device version numbers listed below in the right column are permitted for use in accordance with ISO 13849-1. The devices listed in the middle column can also be used.

Below you will find an overview of the devices that—according to ISO 13849-1—are permitted for use:

Device designation	Device ID for systems with integrated FS	Device ID for systems without integrated FS
Inverter modules		
UM 117DW	689572-01	
UM 116D	667954-01	542998-01
UM 116DW	667946-01	369629-01 index B
UM 115D	671566-01	387852-01 index E
UM 114D	671288-01	510509-01 index E
UM 113D	730435-01	518703-01 index B
UM 112D	731984-01	519971-01 index C
UM 122D	667633-01	519972-01 index C
UM 121BD	667942-01	513037-01 index C
UM 111BD	671968-01	513035-01 index E
UM 121D	667838-01	392319-01 index F
UM 111D	667945-01	392318-01 index F
Power supply modules		
UVR 120D	728252-01	390188-01 index K
UV 130D	728250-01	389311-01 index E
UVR 130D	728248-01	377639-01 index K
UVR 140D	728253-01	390281-01 index N
UVR 150D	728255-01	390421-01 index P
UVR 160D	728257-01	530341-01 index G
UVR 160DW	728258-01	560106-01 index G
UVR 170DW	546911-02	
UVR 170D	807429-01	



Device designation	Device ID for systems with integrated FS	Device ID for systems without integrated FS
Non-regenerative compact inverters		
UE 210D	733421-01	558302-01 index C
UE 211D	733423-01	558303-01 index C
UE 212D	733424-01	558304-01 index C
UE 230D	733425-01	558305-01
UE 240D	733426-01	558306-01
UE 241D	733427-01	558307-01
UE 242D	733428-01	558308-01
UEC 111(FS)	Not yet available	625777-xx
UEC 112(FS)	Not yet available	625779-xx
UE 110	Not yet available	375713-02 index B
UE 111	Not yet available	375714-02 index B
UE 112	Not yet available	375715-02 index B
UMC 111FS	664231-02	
Regenerative compact inverters		
UR 242D	741359-01	536565-01 index A
UR 230D	741356-01	536561-01 index A
UR 240D	741357-01	536564-01 index B



With the following HSCI control components, you must also make a distinction between those that are required in a system with functional safety and those that can be used in a system without functional safety. Devices with FS are listed below in the middle column:

Device designation	Device ID for systems with integrated FS	Device ID for systems without integrated FS
Machine operating panels and keyboard units		
In systems with FS you must use a machine operating panel for functional-safety applications. In these operating panels, all keys have twin channels. A movement can therefore be executed without additional permissive button/key.		
MB 620 (FS)	660090-xx	617973-xx
TE 635Q (FS)	662255-xx	617975-xx
TE 645Q(FS)	685394-xx	682104-xx
MB 720(FS)	805474-xx	784803-xx
TE 735 (FS)	805493-xx	771898-xx
TE 745 (FS)	805482-xx	679817-xx
PLB basic modules		
In FS systems, mixed use of PLB basic modules with and without FS is possible. However, at least one PLB 62xx FS must be used in systems with FS.		
PLB 6104 (FS)	590479-xx	591828-xx
PLB 6106 (FS)	804755-xx	630058-xx
PLB 6108 (FS)	804756-xx	630059-xx
PLB 6204 (FS)	586789-xx	591832-xx
PLB 6206 (FS)	622721-xx	630054-xx
PLB 6208 (FS)	620927-xx	630055-xx
PLB 6001 (FS)	Not yet available	668792-xx
PLD-H I/O modules		
In systems with FS, the mixed use of PLD-H modules with and without FS is possible in PLB basic modules with FS. However, do not insert PLD-H modules with FS in PLB basic modules without FS. Furthermore, the modules with FS must always be inserted into the PLB with FS starting from the left.		
PLD-H 16-08-00, PLD-H 08-04-00 FS	598905-xx	594 243
PLD-H 08-16-00, PLD-H 04-08-00 FS	727219-xx-xx	650891-xx



Device designation	Device ID for systems with integrated FS	Device ID for systems without integrated FS
Handwheels In FS systems, handwheels with cross-circuit proof permissive buttons must be used. Handwheels for which this has been implemented are identified with FS.		
HR 410 (FS)	337159-xx, 578114-xx (with detent)	296469-xx, 535220-xx (with detent)
HR 520 (FS)	670304-xx, 670305-xx (with detent)	670302-xx, 670303-xx (with detent)
HR 550 FS	598515-02, 606622-02 (with detent)	---

The other control components (MC 6xxx or MC 7xxx, CC 6xxx, PSL 13x, MS 1xx, BF 2xx) can be used regardless of whether the system is a system with or without functional safety (FS).



Attention

In HSCI systems with integrated functional safety (FS) you may use only devices that have been certified for use in such systems. For components and devices permitted for use in systems with functional safety, please refer to the Technical Manual for Functional Safety (FS). Upon request, your contact partner at HEIDENHAIN can provide you with the manual.

3.6 HSCI

3.6.1 Introduction

The main computer is connected to the controller units (CC or UxC) and the PLB 6xxx PLC basic modules in the electrical cabinet via **HSCI** (HEIDENHAIN Serial Controller Interface). The MB machine operating panel is also connected via HSCI. The connection of the various control components via HSCI offers numerous benefits, including:

- Simple and uncomplicated wiring
- High noise immunity
- Comprehensive yet straightforward possibilities for diagnostics

The logic area of the control system is supplied with power via two separate power loops:

- The MC, MB, PLs and BF are supplied with power via the PSL 130 power supply unit. The voltages of the PSL 130 correspond to the PELV system as per EN 50178.
- The logic voltages of the CC are voltages derived from the DC-link voltage of the inverters, and are transmitted via connector X69 "NC supply voltages and control signals." These voltages also correspond to PELV as per EN 50178.



The following table shows the maximum permissible number of individual HSCI participants:

HSCI component	Maximum number	
MC (HSCI master)	1 in the system	
CC, UEC, UMC (HSCI slave)	4 drive-control motherboards, (distributed to CC, UEC, UMC as desired)	
MB (FS), PLB 6001 (FS) (HSCI slave)	2 in the system	Total number of 9 components must not be exceeded in this case
PLB 62xx (FS) (HSCI slave)	1 in the system (not with UEC 11x)	
PLB 61xx (FS), PLB 62xx (FS) (HSCI slave)	7 in the system	
HR (FS) handwheel (on MB or PLB 6001)	1 in the system	
PLD-H xx-xx FS (in PLB 6xxx FS)	8 in the system	Total number of 1000 I/Os must not be exceeded in this case
PLD-H xx-xx (in PLB 6xxx FS)	64 in the system	



Danger

If you use more than one operating station or machine operating panel, the PLC program must ensure that only one of the operating devices is active at any one time so as to avoid danger to the operator.

The order of the HSCI participants at X500 of the main computer can be freely chosen. Each HSCI participant is assigned its HSCI address based on its position in the HSCI chain:

- First device after the master (main computer): Bus address 1
- Second device after the master (main computer): Bus address 2, etc.

The main computer selects a master controller unit, based on the position of the controller unit in the HSCI network. The first controller unit in the system (nearest the main computer) becomes the master controller unit.

Module 9132 Selection of override potentiometer

In a control system with more than one HSCI machine operating panel or keyboard unit, you can use Module 9132 to define which potentiometers are to be active. The potentiometer values of the selected operating panel or keyboard unit are used to calculate the override.



Danger

If you use more than one operating station or machine operating panel, the PLC program must ensure that only one of the two operating devices is active at any one time so as to avoid danger to the operator.

Constraints:

- The override potentiometers are selected by selecting the machine operating panel to which the potentiometers are connected.
- The numbering of the machine operating panels starts with 0 and is based on the arrangement or sequence in the IOC configuration or the HSCI chain. A machine operating panel or PLB 6001 at connector X501 of the MC always receives the last, i.e. the highest number of the machine operating panels in the HSCI system.
- After the control has been restarted and the PLC has been restarted/compiled, the potentiometers on the machine operating panel with the number 0 or the assignment with the number 0 are active (first operating panel in the chain).
If you want the previously active operating panel to be active again, the PLC program must write the corresponding information to the nonvolatile memory and evaluate it during a restart of the PLC. For safety reasons, HEIDENHAIN recommends setting the override values to 0 (PLC words W752, W764 and W766) for the time it takes the PLC program to complete this task.
- There is no basic switchover of operation between the machine operating panels or the keyboard units. Only the evaluation of the potentiometers is switched.

Call:

PS K/B/W/D <Number of machine operating panel or assignment>
0...n

PS K/B/W/D <Mode>
0: Selection of the active machine operating panel or the active keyboard unit

PS K/B/W/D <Reserved>

CM 9132

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Invalid value for <Mode> parameter
	2	Programmed MB machine operating panel or programmed assignment not found
	47	Selection of a machine operating panel was not entered in the control's log



3.6.2 Topology

The HSCI slaves are connected—in series—to connector X500 of the main computer. Connector X502 is always the HSCI input to the HSCI slaves and X500 the HSCI output to the next HSCI slave.

The nonsynchronized second HSCI output X501 of the MC can be used to connect a machine operating panel or a PLB 6001. However, do not connect any devices other than a machine operating panel or a PLB 6001 to X501 (no other or further HSCI components).

The HSCI components are connected via special shielded HSCI cables adapted for the increased demands of machine tool environments.

Terminating resistors are not required in the HSCI system.

The HSCI network is to be structured as an "open" ring (line structure). Connector X500 (or X501) on the main computer is the beginning of the HSCI network. The HSCI connection is then led to X502 of the next HSCI participant (CC, UxC, PL or MB), where X500 again serves as the output for continuing the HSCI network to the next participant (X502). The last HSCI participant in the ring detects its position automatically (X500 remains open) and internally and independently closes the ring of the HSCI network.



Note

A star configuration of the HSCI network (e.g. by using a hub) is not possible.

An HSCI segment always starts from the connector X500 or X501. The first HSCI segment starts at X500 of the MC and ends at the last HSCI component connected to X500 of the MC. The second HSCI segment starts at X501 of the MC and ends at the machine operating panel or a PLB 6001. The HSCI cabling used within an HSCI segment must not exceed a total length of 70 m. The individual cables can have any length, as long as the sum of their lengths does not exceed 70 m.

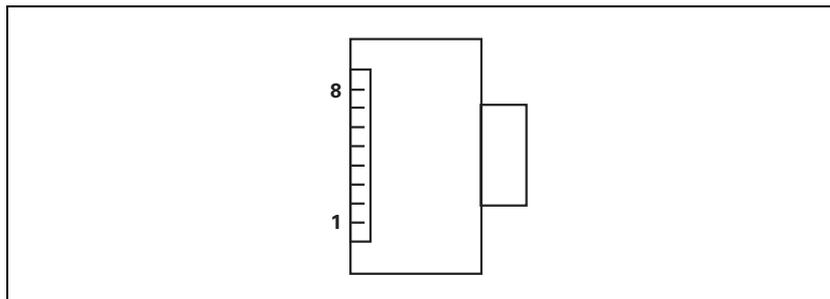


Note

The maximum permissible total length of the HSCI cables used within an HSCI segment is 70 m.

3.6.3 HSCI interface

Face of the connector:



X500
X501
X502

Pin layout:

The order of the HSCI participants at X500 of the MC is freely selectable. Each HSCI participant is assigned its HSCI address based on its position in the HSCI chain, see page 207.

Pin layout of the HSCI cable:

ID 618893-xx			
Female	Color	Assignment	Female
1	White/Green	Data	1
2	Green	Data	2
3	White/Orange	Data	3
4	Vacant	Vacant	4
5	Vacant	Vacant	5
6	Orange	Data	6
7	Vacant	Vacant	7
8	Vacant	Vacant	8

3.7 iTNC 530 HSCI connection overview

3.7.1 MC 6241

MC 6141, MC 6241: Main computer for installation in electrical cabinet			
Pin layout	Connector	Function	Page
	X29	Reserved	–
	X26, X116	Ethernet data interface	325
	X27	RS-232-C/V.24 data interface	327
	X141, X142, X143, X144	USB 2.0 interface (Type A)	380
	X101	+24 V NC power supply	253
	X500	HSCI output 1 (synchronized) to CC, PL, MB (not with MC 6x5x)	208
	X501	HSCI output 2, only to MB or PLB 6001 (not with MC 6x5x)	208
	X600	Reserved	–
	X249	HDL interface for visual display unit	329
	X121	MC 6x4x: Profibus (option)	288
	X125	SIK (System Identification Key)	–
	X601	Reserved	–
	⊕	Protective ground	–



Attention

Do not engage or disengage any connecting elements while the unit is under power!

3.7.2 MC 6242

MC 6242: Main computer for integration in the operating panel or electrical cabinet

Pin layout	Connector	Function	Page
	X29,	Reserved	–
	X26, X116	Ethernet data interface	325
	X27	RS-232-C/V.24 data interface	327
	X141, X142	USB 2.0 interface (Type A)	380
	X101	+24 V NC power supply	253
	X500	HSCI output 1 (synchronized) to CC, PL, MB (not with MC 6x5x)	208
	X501	HSCI output 2, only to MB or PLB 6001 (not with MC 6x5x)	208
	X600	Reserved	–
	X249	HDL interface for visual display unit	329
	X121	MC 6x4x: Profibus (option)	288
	X125	SIK (System Identification Key)	–
	X601, X28, X702	Reserved	–
	⊕	Protective ground	–



Attention

Do not engage or disengage any connecting elements while the unit is under power!



Note

With integration of the MC 6242 into the operating panel, special integration conditions must be observed, see page 186.

3.7.3 MC 6222

MC 6222: Main computer for installation in the operating panel,

Pin layout	Connector	Function	Page
	X29	Reserved	
	X26, X116	Ethernet data interface	325
	X27	RS-232-C/V.24 data interface	
	X141, X142, X143, X144	USB 2.0 interface (Type A)	380
	X101	+24 V NC power supply	253
	X500	HSCI output 1 (synchronized) to CC, PL, MB (not with MC 6x5x)	208
	X501	HSCI output 2, only to MB or PLB 6001 (not with MC 6x5x)	208
	X600	Reserved	
	X601	Reserved	
	X116	Reserved	
	X121	MC 6222: Profibus (option)	
	X125	SIK (System Identification Key)	
	X3	Connection for screen soft keys	378
	⊕	Protective ground	



Attention

Do not engage or disengage any connecting elements while the unit is under power!

3.7.4 MC 7222

MC 7222: Main computer for installation in the operating panel

Pin layout	Connector	Function	Page
	X29	Reserved	
	X26, X116	Ethernet data interface	325
	X27	RS-232-C/V.24 data interface	
	X141, X142, X143, X144	USB 2.0 interface (Type A)	380
	X101	+24 V NC power supply	253
	X500	HSCI output 1 (synchronized) to CC, PL, MB (not with MC 6x5x)	208
	X501	HSCI output 2, only to MB or PLB 6001 (not with MC 6x5x)	208
	X600	Reserved	
	X601	Reserved	
	X116	Reserved	
	X121	MC 7222: Profibus (option)	
	X125	SIK (System Identification Key)	
	X3	Connection for screen soft keys	378
	⊕	Protective ground	



Attention

Do not engage or disengage any connecting elements while the unit is under power!

3.7.5 MC 6341

MC 6341: Main computer for installation in electrical cabinet			
Pin layout	Connector	Function	Page
	X29	Reserved	–
	X26, X116	Ethernet data interface	325
	X27	RS-232-C/V.24 data interface	327
	X141, X142, X143, X144	USB 2.0 interface (Type A)	380
	X101	+24 V NC power supply	253
	X500	HSCI output 1 (synchronized) to CC, PL, MB (not with MC 6x5x)	208
	X501	HSCI output 2, only to MB or PLB 6001 (not with MC 6x5x)	208
	X600	Reserved	–
	X249	HDL interface for visual display unit	329
	X121	MC 6x4x: Profibus (option)	288
	X125	SIK (System Identification Key)	–
	X601	Reserved	–
	⊕	Protective ground	–



Attention

Do not engage or disengage any connecting elements while the unit is under power!

3.7.6 IPC 6341

IPC 6341: Industrial PC for installation in electrical cabinet			
Pin layout	Connector	Function	
	X29	Reserved	–
	X26, X116	Ethernet data interface	325
	X27	RS-232-C/V.24 data interface	327
	X141, X142, X143, X144	USB 2.0 interface (Type A)	215
	X101	+24 V NC power supply	253
	X500	Reserved	–
	X501	Reserved	–
	X600	Reserved	–
	X249	VGA interface for monitor	215
	X121	Reserved	–
	X125	Reserved	–
	X601	Reserved	–
		Protective ground	–



Attention

Do not engage or disengage any connecting elements while the unit is under power!



Note

Apart from that, the same operating conditions apply as for the MC 6xxx main computer units.

X249: For commissioning and maintenance purposes, a VGA monitor with a resolution of 1280 x 1024 pixels can be connected to X249.

X141 to X144: For commissioning and maintenance purposes, USB devices (for example mouse and keyboards) can be attached to connectors X141 to X144.

Pin layout for USB 2.0 connection (Type A):

USB connection (female) 4-pin	Assignment
1	+5 V
2	USBP-
3	USBP+
4	GND

3.7.7 CC 6106

CC 6106, controller unit with 6 control loops and HSCI interface			
Pin layout	Connector	Function	Page
	X15 to X20	Speed encoder	300
	X51 to X56	PWM output	307
	X69	Supply bus	233
	X201 to X206	Position encoder	297
	X500	HSCI output	208
	X502	HSCI input	208
	–	SPI slot 1 (on bottom, reserved for expansion modules)	–
	–	SPI slot 2 (on bottom, reserved for expansion modules)	–
	X74	+ 5 V supply	235
	X7	Bridge for signal ground (= functional ground) (on bottom)	236
	⊕	Protective ground	–



Attention

Do not engage or disengage any connecting elements while the unit is under power!

CC 6108 controller unit with 8 control loops and HSCI interface

Pin layout	Connector	Function	Page
	X15A - X18A	Speed encoder Drive-control motherboard A	300
	X15B - X18B	Speed encoder Drive-control motherboard B	300
	X51A - X54A	PWM output Drive-control motherboard A	307
	X51B - X54B	PWM output Drive-control motherboard B	307
	X69A	Supply bus Drive-control motherboard A	233
	X69B	Supply bus Drive-control motherboard B	233
	X201A - X204A	Position encoder Drive-control motherboard A	297
	X201B - X204B	Position encoder Drive-control motherboard B	297
	X500A	HSCI output Drive-control motherboard A	208
	X502A	HSCI input Drive-control motherboard A	208
	X500B	HSCI output Drive-control motherboard B	208
	X502B	HSCI input Drive-control motherboard B	208
	X74	+ 5 V supply	235
	-	SPI slot 1 (on bottom, reserved for expansion modules)	-
	-	SPI slot 2 (on bottom, reserved for expansion modules)	-
	X7	Bridge for signal ground (= functional ground) (on bottom)	236
	⊕	Protective ground	-



Attention

Do not engage or disengage any connecting elements while the unit is under power!



3.7.9 CC 6110

CC 6110 controller unit with 10 control loops and HSCI interface			
Pin layout	Connector	Function	Page
	X15A - X18A	Speed encoder Drive-control motherboard A	300
	X15B - X20B	Speed encoder Drive-control motherboard B	300
	X51A - X54A	PWM output Drive-control motherboard A	307
	X51B - X56B	PWM output Drive-control motherboard B	307
	X69A	Supply bus Drive-control motherboard A	233
	X69B	Supply bus Drive-control motherboard B	233
	X201A - X204A	Position encoder Drive-control motherboard A	297
	X201B - X206B	Position encoder Drive-control motherboard B	297
	X500A	HSCI output Drive-control motherboard A	208
	X502A	HSCI input Drive-control motherboard A	208
	X500B	HSCI output Drive-control motherboard B	208
	X502B	HSCI input Drive-control motherboard B	208
	X74	+ 5 V supply	235
	–	SPI slot 1 (on bottom, reserved for expansion modules)	–
	–	SPI slot 2 (on bottom, reserved for expansion modules)	–
	X7	Bridge for signal ground (= functional ground) (on bottom)	236
	⊕	Protective ground	–

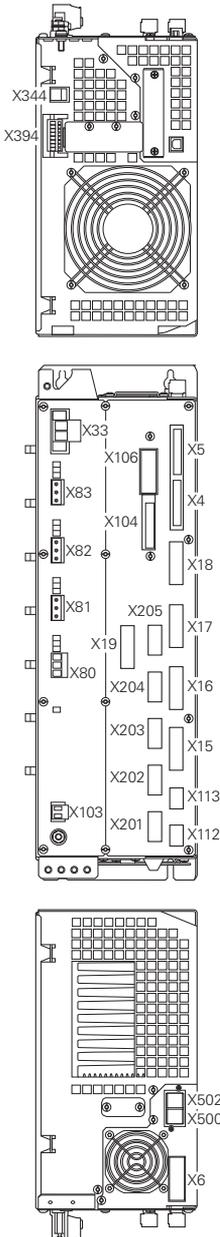


Attention

Do not engage or disengage any connecting elements while the unit is under power!

3.7.10 UMC 111FS

UMC 111 FS: Compact controller unit with inverter, external DC-link power supply, PLC I/Os and functional safety (FS)

Pin layout	Connector	Function	Page
	X4, X5	Single-channel PLC inputs	279
	X6	Single-channel PLC outputs	279
	X15 to X18	Speed encoder	300
	X33	DC-link power supply	258
	X81	Motor connection for axis 1 (9 A rated current at 3.3 kHz)	260
	X82	Motor connection for axis 2 (9 A rated current at 3.3 kHz)	260
	X83	Motor connection for axis 3 (9 A rated current at 3.3 kHz)	260
	X84	Motor connection for axis 4 (9 A rated current at 3.3 kHz)	260
	X103	24 V power supply	261
	X104	Single/dual-channel PLC inputs	261
	X106	Single/dual-channel PLC outputs	284
	X112	Reserved (do not use)	–
	X113	Reserved (do not use)	–
	X344	24 V supply for motor holding brake	
	X394	Motor holding brake 1 to 4	
	X500	HSCI output	208
	X502	HSCI input	208
	B	Signal ground (= functional ground)	–
	⊕	Protective ground M5	–



Attention

Do not engage or disengage any connecting elements while the unit is under power!



3.7.11 UMC 111

UMC 111: Compact controller unit with inverter, external DC-link power supply, PLC I/Os			
Pin layout	Connector	Function	Page
	X4, X5	Single-channel PLC inputs	279
	X6	Single-channel PLC outputs	279
	X15 to X18	Speed encoder	300
	X33	DC-link power supply	258
	X81	Motor connection for axis 1 (9 A rated current at 3.3 kHz)	260
	X82	Motor connection for axis 2 (9 A rated current at 3.3 kHz)	260
	X83	Motor connection for axis 3 (9 A rated current at 3.3 kHz)	260
	X84	Motor connection for axis 4 (9 A rated current at 3.3 kHz)	260
	X103	24 V power supply	261
	X112	Reserved (do not use)	–
	X113	Reserved (do not use)	–
	X344	24 V supply for motor holding brake	
	X394	Motor holding brake 1 to 4	
	X500	HSCI output	208
	X502	HSCI input	208
	B	Signal ground (= functional ground)	–
	⊕	Protective ground M5	–

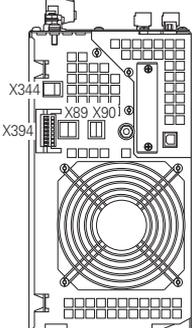
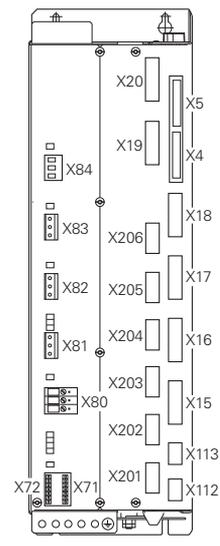
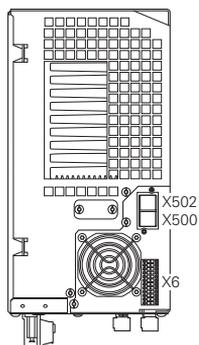


Attention

Do not engage or disengage any connecting elements while the unit is under power!

3.7.12 UEC 11x

UEC 11x: Compact controller unit with integrated inverter and PLC I/Os (**without** functional safety (FS))

Pin layout	Connector	Function	Page
	X4, X5	PLC inputs	279
	X6	PLC outputs	282
	X15 to X19	Speed encoder	300
	X31	Supply voltage for UEC 11x (3 x 400 V ± 10 %)	258
	X71	Safety relay for spindle (pulse inhibitor for spindle)	260
	X72	Safety relay for axes (pulse inhibitor for axes)	260
	X80	Motor connection for spindle (24 A rated current at 3.3 kHz)	260
	X81	Motor connection for axis 1 (6 A rated current at 3.3 kHz)	260
	X82	Motor connection for axis 2 (6 A rated current at 3.3 kHz)	260
	X83	Motor connection for axis 3 (9 A rated current at 3.3 kHz)	260
	X84	Motor connection for axis 4 (6 A rated current at 3.3 kHz)	260
	X89	Braking resistor	261
	X90	24 V NC output / 3.5 A	261
	X112	TS touch trigger probe	313
	X113	TT touch trigger probe	313
	X201 to X205	Position encoder	297
	X344	24 V supply for motor holding brake	261
	X394	Motor holding brake 1 to 4	261
	X500	HSCI output	208
	X502	HSCI input	208
	⊕	Protective ground M5	-

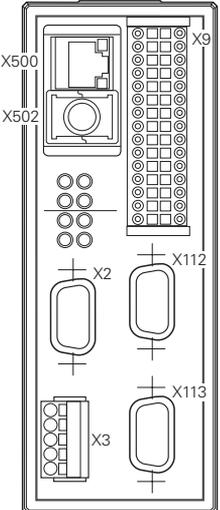


Attention

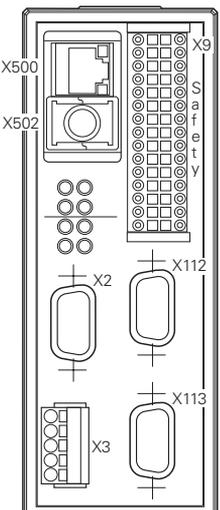
Do not engage or disengage any connecting elements while the unit is under power!



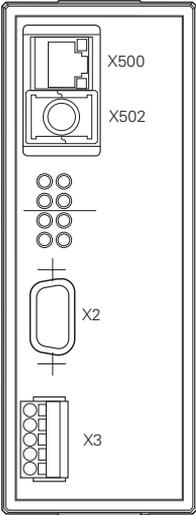
3.7.13 PLB 62xx

System module			
Pin layout	Connector	Function	Page
	X500	HSCI output	208
	X502	HSCI input	208
	X9	Safety-related PLC inputs/outputs	273
	X2	Reserved	–
	X3	+ 24 V NC, + 24 V PLC power supply	267
	X112	TS or TT touch trigger probe	313
	X113	TS or TT touch trigger probe	313
	Diagnosis: For meanings of the LEDs, see page 268		

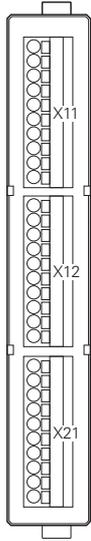
3.7.14 PLB 62xx FS

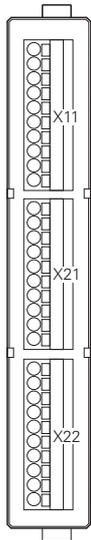
System module			
Pin layout	Connector	Function	Page
	X500	HSCI output	208
	X502	HSCI input	208
	X9	Safety-related PLC inputs/outputs	274
	X2	Reserved	–
	X3	+ 24 V NC, + 24 V PLC power supply	267
	X112	TS or TT touch trigger probe	313
	X113	TS or TT touch trigger probe	313
	<p>On the PL 62xx FS, the I/O terminals of connector X9 are dual-channel terminals and can therefore be used for safety-related applications.</p> <p>PLD-Hs with functional safety (FS) can be used only in the PL 62xx FS. A safety-related PLD must not be inserted after a PLD without FS, or after an empty slot. Safety-related PLDs must be inserted—one after the other—into successive slots without interruption (empty slot). The normal PLDs or PLAs must not be inserted until all PLDs with FS have been inserted.</p>		
Diagnosis: For meanings of the LEDs, see page 268			

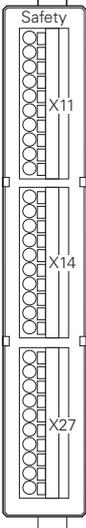
3.7.15 PLB 61xx

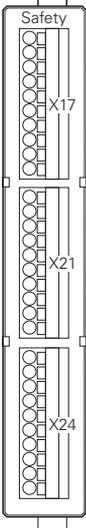
Expansion module			
Pin layout	Connector	Function	Page
 <p>The diagram shows the front panel of the expansion module. At the top, there are two circular connectors labeled X500 and X502. Below them is a row of six small circular LEDs. Further down is a rectangular connector labeled X2 with polarity markings. At the bottom is a multi-pin connector labeled X3.</p>	X500	HSCI output	208
	X502	HSCI input	208
	X2	Reserved	–
	X3	+ 24 V NC, + 24 V PLC power supply	267
	Diagnosis: For meanings of the LEDs, see page 268		

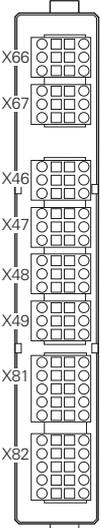
3.7.16 I/O modules PLD-H and PLA-H

PLD-H 16-08-00			
Pin layout	Connector	Function	Page
	X11	PLC inputs, channel A	275
	X12	PLC inputs, channel A	275
	X21	PLC outputs, channel A	277
	<p>Diagnosis (meanings of the LEDs):</p> <ul style="list-style-type: none"> ■ Red (X11/pin 1) status LED <ul style="list-style-type: none"> • Flashes: status of I/O module OK • Continuously on or off: error on I/O module ■ Yellow (per output): Status of the output <p>Error recognition:</p> <ul style="list-style-type: none"> ■ Short circuit: A short circuit is reported when a current ≥ 20 A flows for approximately 3 ms. Both the output-specific message and the group message are modal. After the short circuit has been removed, the PLC must reset the output before it can be activated again. 		

PLD-H 08-16-00			
Pin layout	Connector	Function	Page
	X11	PLC inputs, channel A	275
	X21	PLC outputs, channel A	277
	X22	PLC outputs, channel A	277
	<p>Diagnosis (meanings of the LEDs):</p> <ul style="list-style-type: none"> ■ Red (X11/pin 1) status LED <ul style="list-style-type: none"> • Flashes: status of I/O module OK • Continuously on or off: error on I/O module ■ Yellow (per output): Status of the output <p>Error recognition:</p> <ul style="list-style-type: none"> ■ Short circuit: A short circuit is reported when a current ≥ 20 A flows for approximately 3 ms. Both the output-specific message and the group message are modal. After the short circuit has been removed, the PLC must reset the output before it can be activated again. 		

PLD-H 08-04-00 FS			
Pin layout	Connector	Function	Page
	X11	PLC inputs, channel A	275
	X14	PLC inputs, channel B	275
	X27	PLC outputs, channel A/B	277
<p>PLD-Hs with functional safety (FS) can be used only in the PL 62xx FS. A safety-related PLD must not be inserted after a PLD without FS, or after an empty slot. Safety-related PLDs must be inserted—one after the other—into successive slots without interruption (empty slot). The normal PLDs or PLAs must not be inserted until all PLDs with FS have been inserted.</p>			

PLD-H 04-08-00 FS			
Pin layout	Connector	Function	Page
	X17	PLC inputs, channel A/B	275
	X21	PLC outputs, channel A	277
	X24	PLC outputs, channel B	277
<p>PLD-Hs with functional safety (FS) can be used only in the PL 62xx FS. A safety-related PLD must not be inserted after a PLD without FS, or after an empty slot. Safety-related PLDs must be inserted—one after the other—into successive slots without interruption (empty slot). The normal PLDs or PLAs must not be inserted until all PLDs with FS have been inserted.</p>			

Pin layout	Connector	Function	Page
 <p>The diagram shows a vertical module with eight connector locations labeled on the left side: X66, X67, X46, X47, X48, X49, X81, and X82. Each connector location is represented by a grid of pins. X66 and X67 are at the top, X46 through X49 are in the middle, and X81 and X82 are at the bottom.</p>	X66 ... X67	±10 V analog outputs	287
	X46 ... X49	±10 V analog inputs	286
	X81 ... X82	PT 100 analog inputs	287
	<p>A maximum of two PLA-H-08-04-04 I/O modules can be used per PLB 6xxx.</p>		

3.8 Supply voltages in the HSCI system

Two separate 24 V power supplies must be used to supply the **+24 V power** to the individual control components in the HSCI system: +24 V NC and +24 V PLC.

HEIDENHAIN recommends using the DC-link buffered PSL 13x for supplying power to the HSCI components, see "PSL 13x low-voltage power supply unit" on page 237.

If you are using the UEC 11x, the main computer and other NC components are usually supplied via the 24 V power supply unit of the UEC (connection X90). A PSL 130 is not necessary if the total current consumption of the NC supply of all HSCI components does not exceed 3.5 A.

External PL assemblies with HSCI (e.g. the PL 6xxx or the PL in the UxC 11x) consist of an HSCI part (bus module and logic) and a PLC part (PLC input/output assemblies). Due to the topology of the HSCI system, the 24 V NC voltage supplied to the HSCI part must comply with the requirements for double basic insulation according to EN 50 178 (PELV). All other NC components with HSCI interface (e.g. main computer and machine operating panel) must also be supplied with 24 V NC voltage with double basic insulation. The reason for the double basic insulation is electrical safety, e.g. accessibility of connecting elements supplied with +24 V NC voltage.

PLC components, such as motor holding brakes and solenoid valves, usually have simple basic insulation. The PLC part must therefore be powered by another +24 V supply voltage. The two supply voltages must not be connected to each other. The double basic insulation of the NC power supply is removed through "mixed operation," i.e. +24 V NC voltage with double basic insulation is connected to PLC components with simple basic insulation. This is not permitted in an HSCI system.

The following components are powered by +24 V NC supply voltage:

- MC 6xxx, MC 7xxx main computer unit or industrial PC
- BF 2xx TFT visual display unit
- MB 6xx or MB 7xx machine operating panel
- TE 6xx or TE 7xx keyboard unit

Protective Extra Low Voltage (PELV) according to EN 61800-5-1 must be complied with by the power supply unit for the +24 V NC supply voltage.

The following components are powered by +24 V PLC supply voltage:

- PLB 6xxx input/output module
- UxC 1xx input/output module
- PLD-H digital plug-in module for PLB
- PLA-H analog plug-in module for PLB
- Motor holding brakes, further components in the PLC circuit



Note

HEIDENHAIN recommends also using a power supply unit complying with Protective Extra Low Voltage (PELV) according to EN 61800-5-1 for the +24 V PLC power supply, although the circuit has only ELV status. In addition, HEIDENHAIN recommends connecting the 0 V PLC supply voltage to protective earth (PE). This is not strictly required according to the VDE standards. However, it provides additional safety in the event of insulation failure in the PLC circuit.

The CC 6xxx controller unit is supplied by the X69 supply bus of the HEIDENHAIN supply module and X74 (+5 V).



Danger

- The +24 V NC supply voltage (PELV system according to EN 50178) is required to be safely separated voltage for the entire HSCI system and must not be connected to the +24 V PLC supply voltage (ELV) of the system.
- Protective Extra Low Voltage (PELV) according to EN 61800-5-1 must be complied with for the +24 V NC power supply of the machine.
- VDE 0160/EN 50178 is to be observed for the +24 V NC voltage lines and cable routing. Lines or cables for safely separated electric circuits thus must have double or reinforced insulation between the wire and the surface if they are routed without spatial separation from other cables and lines.
- Due to the structure of the PLC area in the HSCI system, the +24 V PLC supply voltage is a voltage with basic insulation (ELV as per EN 61800-5-1).
- The 0 V signal of the NC power supply must be connected by a 6 mm² conductor to the machine's central functional ground (B).
- The 0 V signal of the PLC power supply must be connected by a 6 mm² conductor to the machine's central protective ground (PE).

The signal ground is used for functional-equipotential bonding. The signal-ground connections (B) of the HEIDENHAIN control components must be connected to the central functional ground of the machine (minimum cross section 6 mm²). The 0 V PLC and all of the protective-ground connections of the HEIDENHAIN control components must be connected separately from the signal-ground connections to the central protective ground (PE) of the machine (minimum cross section 6 mm²). The central signal ground and the central protective ground must be connected with each other for the machine! The cross section of this conductor must be at least as large as the largest cross section of the conductors for connecting the components used to protective ground or functional ground.



Note

The line cross section of the +24 V NC power supply must be designed for the power consumption of the connected devices. EN 60204-1 lists the protection provided by line cross sections.

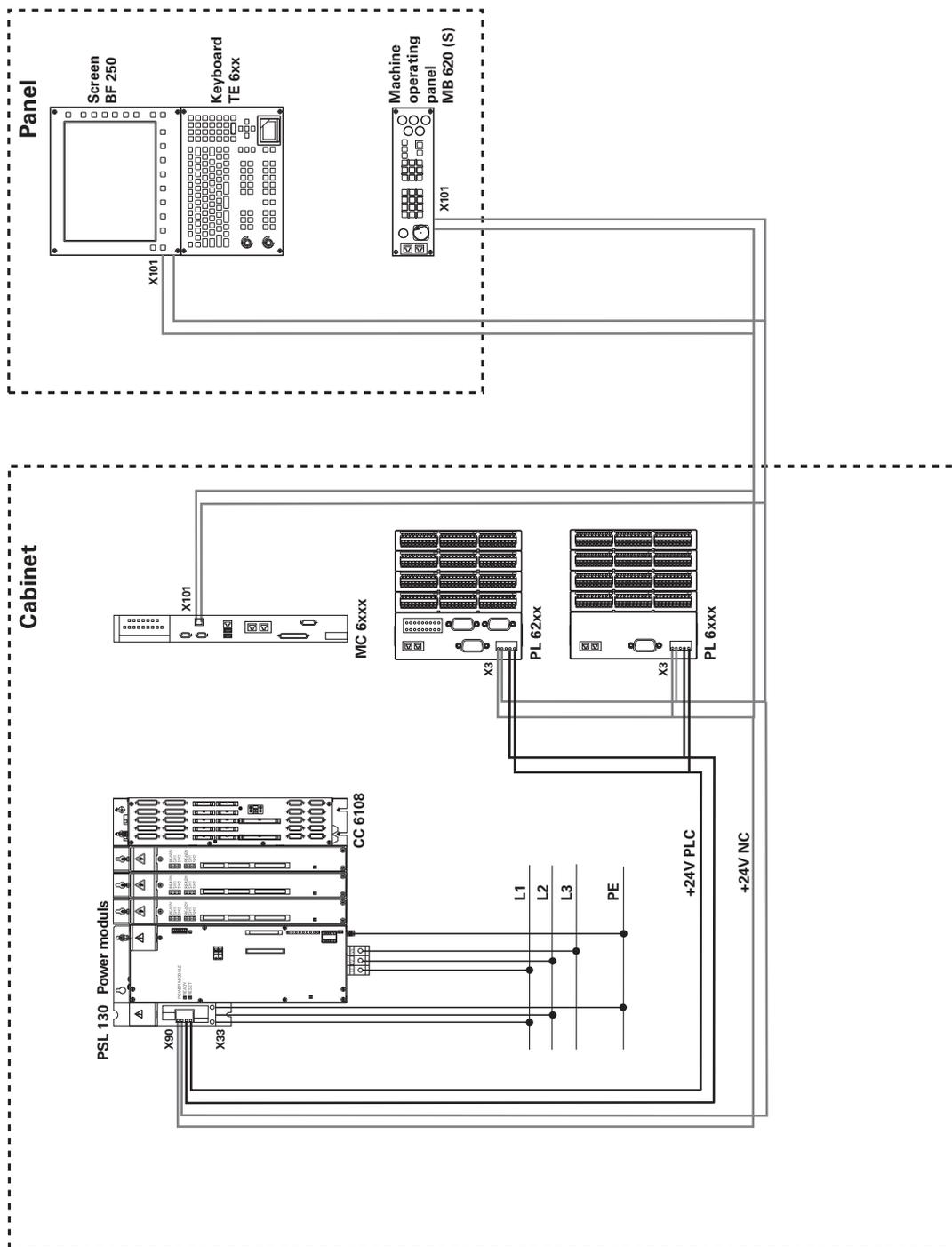
Minimum cross section of the +24 V NC power supply: 0.75 mm²



Note

The motor brakes are controlled by 24 V PLC voltage. The trigger circuit and the brake itself are usually separated from the line power only by basic insulation according to EN 618100-5-1 (also EN 50178). Also, other add-on devices that are controlled by PLC circuits usually have only basic insulation from the line power.

3.8.1 Overview: Power supply for the iTNC



Note

Please also refer to the Technical Manual for your control and the "Inverter Systems and Motors" Technical Manual.



3.8.2 Current consumption of the HSCI components

The following table shows the current consumption of the HSCI components at 24-V NC:

HSCI component		Current consumption at 24 V DC
	MC 6240 / 6241 / MC 6242	1.7 A
	MC 6341	1.8 A
	MC 6222, MC 7222	2.5 A
Machine operating panel	PLB 6001	0.2 A (without handwheel)
	MB 6xx / MB 7xx	0.2 A (without handwheel)
Keyboard unit	TE 6xx / TE 7xx	0.2 A (without handwheel)
PLC inputs/outputs	PLB 62xx	0.3 A (without touch probe)
	PLB 61xx	0.2 A
	PLD	0.05 A
	PLA	0.1 A
Visual display unit	BF 760	2.5 A
	BF 260	2.5 A
	BF 750	2.1 A
	BF 250	2.1 A
Handwheels	HR 410	0.05 A
	HR 520	0.05 A
	HRA 551FS + HR 550FS	0.5 A (during charging)
	HR 130	0.05 A
	HR 110 + 3 x HR 150	0.2 A
Touch probes	See specifications of the touch probes	

3.8.3 X90: +24 V NC output of the UxC 11x (FS)

Assignment

Connecting terminal X90	Assignment
+ or terminal 1	+24 V NC (max. 3.5 A)
– or terminal 2	0 V



Attention

The interface is electrically separated (protective extra-low voltage (PELV) as per EN 61800-5-1).

Load capacity

Device	Load capacity of 24 V NC supply (X90)
UEC 11x (FS)	3.5 A

Current consumption of the HSCI components

Device	Current consumption of the 24 V NC supply
MC 6110 or MC 7110	1.8 A
MB 620 or MB 720 (including all PLC outputs)	1.0 A
PL 62xx (incl. TS and TT)	0.5 A
PL 61xx	0.2 A

Example: Typical TNC 620 configuration with UEC 11x

Device	Current consumption 24 V NC
MC 6110 or MC 7110	1.8 A
MB 620 or MB 720 (including all PLC outputs)	1.0 A
PL 61xx	0.2 A
Total	3.0 A < 3.5 A

A PSL 130 unit is not needed for this application. The +24 V NC supply of the UEC 11x (X90) suffices for the connected components.



3.8.4 X101: NC power supply

The MC main computer is supplied with +24 V NC (control voltage) of the machine, for example by the PSL 130, see "PSL 13x low-voltage power supply unit" on page 237 or by the integrated 24 V power supply unit of the UEC 11x (FS).

Protective Extra Low Voltage (PELV) according to EN 61800-5-1 must be complied with for the +24 V NC power supply.

Power supply: Minimum absolute value: +20.4 V–
Maximum absolute value: +28.8 V–

Pin layout:

Connecting terminals at X101	Assignment	Fuse
+ / terminal 1	+24 V NC	7 A safety fuse integrated in the MC
- / terminal 2	0 V NC	



Attention

Ensure that either the DC-link power supply unit is switched off or the line power is disconnected before connecting the power cables!

Power consumption:

Device	Power consumption
MC 6241	40 W
MC 6341, IPC 6341	43 W
MC 6222, MC 7222	60 W



Note

If USB components that are connected to X141/X142 require more than 0.5 A, a separate power supply becomes necessary for these components. One possibility is the USB hub from HEIDENHAIN (582884-02).

3.8.5 Power supply of the CC 61xx

The CC 61xx controller unit is supplied with **+5 V power** by the power supply units via supply bus X69 and connector X74, see page 233.

The control monitors the 5 V supply voltage. If it drops below 4.75 V, the error message **5 V power supply too low** appears. If it rises above 5.4 V, **5 V power supply too high** is indicated.

For information on the power supply units, refer to the "Inverter Systems and Motors" Technical Manual.

Device	Load capacity
UVR 1xxD, UE 2xxD	20,00 A via X74 10.00 A via X69

Device	Current consumption of the 5 V supply
CC 6106/6 control loops	3.80 A Consisting of: 1 drive-control motherboard: 2.00 A 2 controller expansion PCBs: 0.90 A each
CC 6108/8 control loops	5.80 A Consisting of: 2 drive-control motherboards: 2.00 A 2 controller expansion PCBs: 0.90 A each
CC 6110/10 control loops	6.70 A Consisting of: 2 drive-control motherboards: 2.00 A 3 controller expansion PCBs: 0.90 A each
UxC drive control board	2.50 A
LS, LB	0.15 A
ERN, ROD, RON	0,20 A
Absolute rotary encoders	0.25 A (+0.085 A with line-drop compensator) ^a
Absolute angle encoders	0.35 A (+0.085 A with line-drop compensator) ^a
LC	0.30 A (+0.085 A with line-drop compensator) ^a

- a. For cable lengths > 10 m between the logic unit and the encoders with EnDat interfaces, a line drop compensator is required (efficiency = 75 %).

Power consumption of CC 61xx:

- CC 6106: 25 W
- CC 6108: 35 W
- CC 6110: 40 W

X69: CC-supply voltage and control signals

Pin layout:

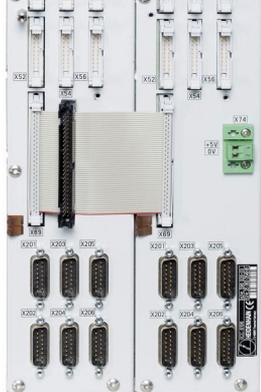
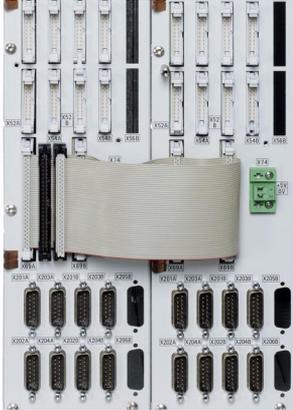
50-pin ribbon connector	Assignment	50-pin ribbon connector	Assignment
1a to 5b	+5 V	16b	GND
6a to 7b	+12 V	17a	RDY.PS
8a	+5 V (low-voltage separation)	17b	GND
8b	0 V (low-voltage separation)	18a	ERR.ILEAK
9a	+15 V	18b	GND
9b	-15 V	19a	PF.PS.AC (only UV 120, UV 140, UV 150, UR 2xx)
10a	UZAN	19b	GND
10b	0 V	20a	Do not assign
11a	IZAN	20b	GND
11b	0 V	21a	Do not assign
12a	RES.PS	21b	GND
12b	0 V	22a	Do not assign
13a	PF.PS.ZK	22b	GND
13b	GND	23a	Reserved (SDA)
14a	ERR.UZ.GR	23b	GND
14b	GND	24a	Reserved (SLC)
15a	ERR.IZ.GR	24b	GND
15b	GND	25a	RES.LE
16a	ERR.TMP	25b	GND

Combination of several CC 61xx controller units

Two drive-control motherboards are housed in one CC 6108 or CC 6110. Both boards must be connected to the supply bus via the connectors X69A and X69B. A short ribbon cable (ID 325 816-15) with three connectors for X69 is therefore included with either of the two CCs. Two of the connectors are connected to X69A and X69B of the CC, respectively. The third connector of the cable serves to establish the connection to X69 of the power supply module.

The ribbon cable with ID 325816-24 (100 mm) must be used for connecting two CC 6106 controller units.

However, if you want to arrange two CC 61xx controller units other than the CC 6106 controller unit next to each other in one row, you need to use the ribbon cable with ID 325 816-22. This cable is required for the following configurations: 2 x CC 6108, 2 x CC 6110 or CC 6110 with CC 6108. The four connectors X69 of the CCs are then connected via the additional accessory cable. The fifth connector of the cable is used to establish the connection to X69 of the power supply module.

Ribbon cable	
ID 325 816-24 on two CC 6106 controller units	
ID 325 816-22 on two CC 6108 controller units	

If several CC 61xx controller units are connected via a cable with ID 325 816-xx, the +5 V power must usually only be supplied by connector X74 of the rightmost CC controller unit. This +5 V supply must be tested.



X74: +5 V power supply for CC

Maximum wire cross section: 2.5 mm²

Connecting terminal at X74	Assignment
1	+5 V NC from the UV supply module (X74)
2	0 V



Attention

The +5 V NC supply via X74 from the supply module is mandatory for the CC 61xx!



Attention

The interface is electrically separated (protective extra-low voltage (PELV) as per EN 61800-5-1).

If the system includes several CCs 61xx that are connected to the same supply bus (X69) via a UV supply module, it is usually sufficient to connect only the last CC 61xx (usually the unit at the extreme right) to the UV via X74. The other CC 61xx units are then supplied via supply bus X69.

If several CC 61xx units are supplied by more than one UV supply module, which means that they are connected to different supply bus systems (X69), then the last CC 61xx (usually the unit at the extreme right) of the respective supply bus must also be supplied with the additional + 5 V of the UV via X74.

If the +5 V power supply to the CC 61xx is too low, the error message **Self-test cannot be started** appears after the control has booted. In this case, please open the bus diagnosis on your control to display more detailed information on the cause of the error:

- ▶ Press the MOD key
- ▶ Press the DIAGNOSIS soft key.
- ▶ Press the BUS DIAGNOSIS soft key.

If the error was caused by too low a supply voltage, the error **(PF.BOARD) Error in the supply voltage** is displayed for the affected CC controller unit(s) in the bus diagnosis. In these cases, check the wiring of the power supply at X74 and contact your HEIDENHAIN service agency.

**X7: Bridge for
signal ground
(= functional
ground)**

Connecting terminal X7	Assignment
1	Connection for signal ground (= functional ground)
2	Connection on housing

In shipping condition of the CC 61xx, the signal ground (pin 1) is connected to the housing (pin 2) over an external bridge. If only one CC 61xx is in the system, it ensures the correct signal-ground connection of the CC.

If there are two or more CC 61xx units in the system that are connected over a UV power module to the same supply bus (X69), this external signal-ground bridge can stay connected with only one CC 61xx. Disengage these bridges on all other CCs in order to prevent ground loops.

If there are two or more CC 61xx units powered over more than one UV power module and are therefore connected with different supply bus systems (X69), then the external signal-ground bridge is to be connected with only one CC 61xx of the respective supply bus. In order to prevent ground loops, disengage this bridge for the signal ground on all other CC 61xx units that are on a common supply bus (X69).



3.8.6 PSL 13x low-voltage power supply unit

PSL 130 pin layout	Connector	Function
 <p>The diagram shows two views of the PSL 130 power supply unit. The top view shows the top panel with a terminal block labeled X90. The bottom view shows the bottom panel with a terminal block labeled X33.</p>	Conductor bar	Connection of DC-link voltage U_z
	B – Signal ground (= functional ground)	Signal ground (0 V signal of the +24 V NC signal connected internally to protective ground)
	X90	Output for supply voltages: <ul style="list-style-type: none"> ■ Terminal 1: +24 V NC ■ Terminal 2: 0 V NC (ground +24 V NC) ■ Terminal 3: + 24 V PLC ■ Terminal 4: 0 V PLC (ground +24 V PLC)
	X33	Input voltages L1, L2
		Protective ground



Attention

Do not engage or disengage any connecting elements while the unit is under power!

PSL 135 pin layout	Connector	Function
<p>The diagram shows the PSL 135 pin layout with the following components labeled:</p> <ul style="list-style-type: none"> X31: A 3-pin connector with terminals labeled L1/L2, +U_{DC}, and -U_{DC}. X74: A 2-pin connector with terminals labeled +5V and 0V. X69: A 2-pin connector. X90: A 4-pin connector with terminals labeled 24V, 0V, 24V, and 0V. Protective ground: A ground symbol at the bottom. 	X31	Input voltages L1, L2 and connection of the DC-link voltage U _z
	B – Signal ground (= functional ground)	Signal ground (0 V signal of the +24 V NC signal connected internally to protective ground)
	X74	Output for supply voltages: <ul style="list-style-type: none"> ■ Terminal 1: +5 V ■ Terminal 2: 0 V
	X69	Power supply and control signals for CC 61xx (for X69 on CC)
	X90	Output for supply voltages: <ul style="list-style-type: none"> ■ Terminal 1: +24 V NC ■ Terminal 2: 0 V NC (ground +24 V NC) ■ Terminal 3: + 24 V PLC ■ Terminal 4: 0 V PLC (ground +24 V PLC)
	⊕	Protective ground



Attention

Do not engage or disengage any connecting elements while the unit is under power!

General information

The two 24 V output voltages of the PSL 13x are generated by two separate power supplies. The + 24 V NC and + 24 V PLC voltages are separated from each other by basic insulation and fulfill the requirements of EN 61800-5-1 for low voltage electrical separation.

The 0 V line of the NC supply voltage must be connected separately to the central grounding point of the machine (= central functional ground). The 0 V NC voltage in the PSL 130 is therefore connected internally with the outward conductor to signal ground (= central functional ground). An outward connector to protective ground will not be available for the 0 V PLC voltage until variant 03. With the variants up to and including variant 02, the 0 V PLC must be tapped at terminal X90.4 and led to the central protective ground. In variant 03 and later variants, an outward connector at the PSL will be available for connecting the 0 V PLC voltage, too.

With the PSL 135, the connections to the central functional ground and protective ground must be realized over the 0 V connections. The respective outward connections will not be available until the next hardware variants of the PSL 135.

The +24 V PLC voltage is also electrically separated by the power supply unit. However, because of the connection to the PLC part of the HSCI system this is a supply voltage with basic insulation. This voltage must not be linked with other voltages.

The +5 V NC supply voltage for X74 of the PSL 135 is taken from the +24 V NC supply voltage and is therefore galvanically connected with it.



Danger

- For the entire HSCI system, the +24 V NC power supply voltage is required to be safely separated voltage and must not be connected with the +24 V PLC voltage of the system.
- Protective Extra Low Voltage (PELV) according to EN 61800-5-1 must be complied with for the +24 V NC power supply of the machine.
- The 0 V signal of the PLC power supply must be connected by a 6 mm² conductor to the machine's central protective ground (PE).
- Refer to your control's grounding diagram!

Specifications

Specifications	PSL 130	PSL 135
Power supplies: at X33 (L1, L2)	400 V~ ± 10 % 50 Hz	
Power supplies: DC-link power bar or X31	400 V- to 750 V-	
Protection	Page 242, 243	
Output voltages: Accuracy of the +24 V NC Accuracy of the +24 V PLC	+24 V NC: ± 5 % +24 V PLC: Variations depending on the load, between 20 V and 28 V (adjustment to 25.0 V)	+24 V NC: ± 5 % +24 V PLC: Variations depending on the load, between 20 V and 28 V (adjustment to 25.0 V)
Accuracy of the +5 V NC	---	+5 V NC: ± 5 % (power supply unit is adjusted to 5.2 V)
Output power	24 V NC output: max. 500 W 24 V PLC output: max. 500 W Total power output: max. 750 W	24 V NC output: max. 350 W 24 V PLC output: max. 500 W 5 V NC output: max. 100 W ±15 V at X69: max. 50 W Total power output: max. 750 W
24 V NC output current	Max. 20 A	Max. 14.5 A
24 V PLC output current	Max. 20 A	Max. 20 A
5 V NC output current	---	Max. 20 A
24 V output current total	Continuous load: max. 31 A Max. load time 1 s: max. 41 A Max. load time 0.1 s: max. 46 A The outputs are short-circuit proof and switch off automatically when overloaded.	
5 V output current	---	Max. 20 A The outputs are short-circuit proof and switch off automatically when overloaded.
Power consumption	Max. 1000 W	
Power loss	Max. 100 W	
Degree of protection	IP 20	
Module width	50 mm	
Weight	2.1 kg	3.2 kg
ID	575047-xx	627032-xx



It is possible to connect both 24 V output voltages of the PSL 13x in parallel. In this way the PSL supplies only a maximum output power of 750 W, which fulfills the requirements for PELV according to EN 61800-5-1. However, because of the parallel circuit, a PSL 130 supplies only an output voltage of +24 V, which can then be used as desired for the 24 V NC voltage or the 24 V PLC voltage. An additional PSL 13x can be used for the respective other voltage.



Danger

When using it to supply the PLC, the common 0 V signal must be connected by a 6 mm² conductor to the machine's central ground.

U_Z DC-link current

Since the power to the PSL 130 is supplied through the DC-link, the voltage fed into the DC-link by the motors that are still running can be used during line voltage failures. The PSL 130 uses the supply voltage buffered via the DC-link to maintain the power supply for the control until the non-HEIDENHAIN inverter system has been shut down properly by the control.

If the PSL is connected over litz wires instead of the DC-link conductor bars, a wire cross section of at least 1.5 mm² must be used. Fuses or a motor protection switch of 6.3 A or greater for conductor protection must be selected depending on the wire cross section used. The PSL 130 is protected internally by a fuse (4 A).

Connecting terminals	Assignment
-U _Z	DC-link voltage -
+U _Z	DC-link voltage +

HEIDENHAIN offers insulated conductor bars if you want to position the PSL 130 next to the left of the UVR inverter and connect it to the DC-link via conductor bars. Two conductor bars are required for each connection. The conductor bars are shipped in packaging units. Also, this position makes it possible to connect the grounding conductor of the PSL via conductor bars. Therefore, angulated non-insulated conductor bars are also included in this packaging unit. This makes it possible to continue using the straight conductor bar included with the UVR for the grounding conductor connection from the UVR to the inverters via the right side.

Length	For connection to	ID (conductor bar set)
150 mm	UVR 120D/130D	687669-01
200 mm	UVR 140D/150D	687670-01
250 mm	UVR 160D/160DW	687671-01



Note

Tightening torque for the screws of the DC-link conductor bars = 3.5 Nm.

X33: Input voltage of the PSL 130

Supply voltage: 400 V \pm 10 %

Connection:

Connecting terminal	Assignment
L1	Phase 1 / 400 V~ \pm 10 % / 50 Hz to 60 Hz
L2	Phase 2 / 400 V~ \pm 10% / 50 Hz to 60 Hz
	Protective ground (YL/GN), $\geq 10 \text{ mm}^2$
	Connecting lead: Wire cross section: at least 1.5 mm^2 (AWG 16) Conductor protection (back-up fuse): Fuses or a motor protection switch of 6.3 A or greater depending on the wire cross section used. Line fuse: Internal protection of the PSL (4 A).
Tightening torque: for the connecting terminals 0.5 to 0.6 Nm Grounding terminal: $\geq 10 \text{ mm}^2$ (AWG 6) Strain relief: Ensure that the connecting cables are not subject to excessive strain.	

Through the connection to the 400 V AC line voltage (L1, L2) via X33, the output voltages of the PSL 130 are available as soon as the machine's main switch has been turned on. The connection of L1 and L2 is absolutely essential for booting the control.

Since the power to the PSL 130 is supplied through the DC-link, the voltage fed into the DC-link by the motors that are still running can be used during line voltage failures. The PSL 130 uses the supply voltage buffered via the DC-link to maintain the power supply for the control until the inverter system has been shut down properly by the control.



Note

HEIDENHAIN recommends connecting the PSL 130 power supply unit to the U_z DC-link voltage and the 400 V supply voltage (X33).

Operation of the PSL 130 without a connection to the U_z dc-link voltage is possible, e.g. as PLC supply in combination with the UEC 11x.

However, be aware that operation with the U_z DC-link voltage but without the power connection (L1 and L2) at X33 is not possible. The line voltage is necessary for the PSL 130 to start correctly.



X31: Input voltage of the PSL 135

Supply voltage: 400 V \pm 10 % or 400 V– to 750 V–

Connection:

Connecting terminal	Assignment
L1	Phase 1 / 400 V~ \pm 10 % / 50 Hz to 60 Hz
L2	Phase 2 / 400 V~ \pm 10% / 50 Hz to 60 Hz
+UDC	400 V– to 750 V–
–UDC	0 V–
	Protective ground (YL/GN), $\geq 10 \text{ mm}^2$
	Connecting lead: Wire cross section: at least 1.5 mm^2 (AWG 16) Conductor protection (back-up fuse): Fuses or a motor protection switch of 6.3 A or greater depending on the wire cross section used. Line fuse: Internal protection of the PSL (4 A).
Tightening torque: for the connecting terminals 0.5 Nm to 0.6 Nm Grounding terminal: $\geq 10 \text{ mm}^2$ (AWG 6) Strain relief: Ensure that the connecting cables are not subject to excessive strain.	

Through the connection to 400 V (L1, L2) via X31, the output voltages of the PSL 135 are available as soon as the machine's main switch has been turned on. These voltages are indispensable to be able to boot the control.

Since the power to the PSL 135 is supplied through the DC-link, the voltage fed into the DC-link by the motors that are still running can be used during line voltage failures. The PSL 135 uses the supply voltage buffered via the DC-link to maintain the power supply for the control until the inverter system has been shut down properly by the control.



Note

If the power supply is other than 400 V, an autotransformer is required. Otherwise, an isolating transformer is not necessary when using a PSL 135.



Note

HEIDENHAIN recommends connecting the PSL 135 power supply unit to the Uz DC-link voltage and the 400 V supply voltage (X31).

X90: Output voltage of the PSL 13x

Output voltages: + 24 V (2 x)

Connection:



Attention

The interface is electrically separated (protective extra-low voltage (PELV) as per EN 61800-5-1).

Connecting terminal	Assignment
Terminal 1 (top)	+ 24 V NC
Terminal 2	0 V NC (ground + 24 V NC)
Terminal 3	+ 24 V PLC
Terminal 4 (bottom)	0 V PLC (ground + 24 V PLC)

Tightening torque:
for the connecting terminals
0.5 Nm to 0.6 Nm

Strain relief:
Ensure that the connecting cables are not subject to excessive strain.

X74: Output voltage of the PSL 135

Output voltages: + 5 V

Connection:



Attention

The interface is electrically separated (protective extra-low voltage (PELV) as per EN 61800-5-1).

Connecting terminal	Assignment
Terminal 1 (top)	+ 5 V NC
Terminal 2	0 V NC (also ground +24 V NC)

Tightening torque:
for the connecting terminals
0.5 Nm to 0.6 Nm

Strain relief:
Ensure that the connecting cables are not subject to excessive strain.

Maximum wire cross section: 4 mm²

Signal ground (= functional ground)

Connections for signal ground, which are connected internally in the PSL 13x to 0 V NC and 0 V PLC signals.

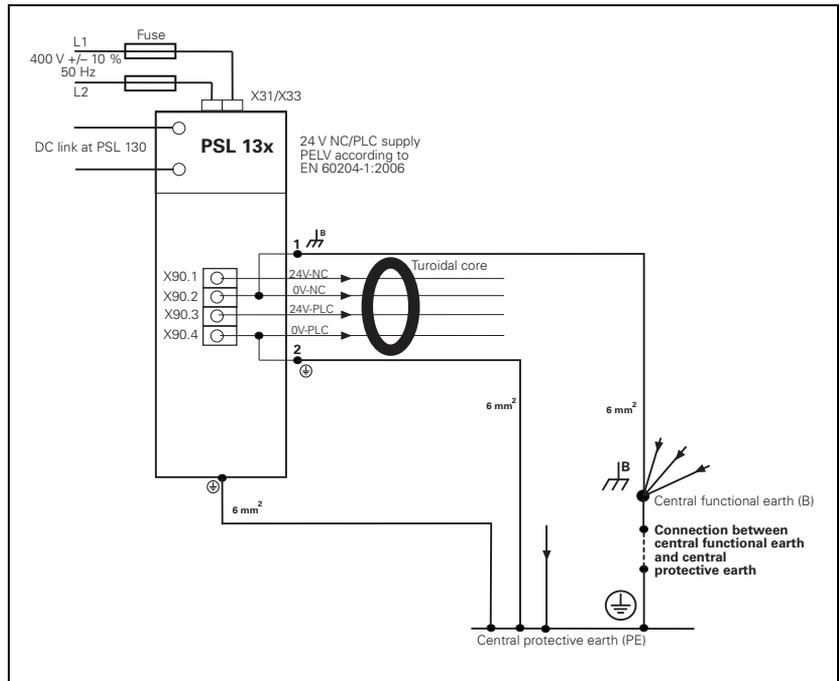
Connection:



Note

Conductor cross section of at least 6 mm² for connecting to signal ground (= central functional ground).

Power connection



The 0 V line of the NC supply voltage must be connected separately to the central grounding point of the machine (= central functional ground). The 0 V NC voltage of the PSL 130 is therefore connected internally with the outward conductor to signal ground (= central functional ground).

An outward connector to protective ground will not be available for the 0 V PLC voltage until variant 03. With the variants up to and including variant 02, the 0 V PLC must be tapped at terminal X90.4 and led to the central protective ground. In variant 03 and later variants, an outward connector at the PSL will be available for connecting the 0 V PLC voltage, too.

With the PSL 135, the connections to the central functional ground and protective ground must be realized over the 0 V connections. The respective outward connections will not be available until later hardware variants of the PSL 135.

If a suitable type of network (see Technical Manuals for Inverters and Motors) is used, the inverter system from HEIDENHAIN and the PSL 13x are connected to the main power line without an additional isolating transformer.

If an isolating transformer is required for the inverter system due to the type of network, then the PSL 13x must also be powered via the isolating transformer of the inverter system.

If the line voltage is 3 x 480 V~ and inverters suitable for this voltage (UE compact inverters) are used, then the PSL 13x must be powered via an additional autotransformer. The required output voltage of the autotransformer is 400 V~ +0 % / -15 %.

Conducted interference

To suppress conducted interference, the conductors for the 24 V NC and 24 V PLC output voltages (X90 of the PSL) must be passed through a toroidal core (ID 309694-07, inside diameter 14 mm). There are different ways to do this depending on the PSL variant:

■ Variants 02 and 03 of the PSL 130, variant 01 of the PSL 135

The toroidal core is included with the PSL. The output lines of connector X90 (24 V NC, 24 V PLC) must be passed through the toroidal core in order to suppress conducted interference. The lines must not be wound around the toroidal core, however.

■ Variant 02 of the PSL 135, variant 04 of the PSL 130

The toroidal core is integrated in the unit. No further external measures for noise suppression must be taken.

3.9 MS 110 / MS 111 installation kit for double-row configuration

3.9.1 General information

Sometimes limited space prevents the control and inverter system from being mounted in the same row in a machine's electrical cabinet, meaning that they must be mounted in two separate rows. In other cases the design calls for a second electrical cabinet to house the inverter system. This means that the distribution and arrangement of the components can be very different from case to case.

In order to establish an electrical connection (immune to noise) between the components of the inverter system, the MS 1xx installation kits are needed. The ribbon cables (unit bus, PWM lines, supply bus) from the other components are connected to the MS mounting cases, and shielded round cables of the appropriate lengths connect the MS mounting cases with each other.

In some cases, in order to ensure that the power supply for the fans of the inverters is maintained under all circumstances, it is also necessary to feed 24 V from an external power supply unit to the unit bus (since this is handled by the unit bus).

In most cases the additional 24 V are not needed, since the UVR 1xxD provides enough current for the fans.

If this is the case, and a double-row configuration is used, then two MS 110 mounting cases are necessary (see basic circuit diagram). For the current consumption of the fans, refer to the "Inverter Systems and Motors" Technical Manual. Based on these values you can calculate whether feeding in the additional 24 V is necessary.

For inverter systems with many powerful UM 1xxD power modules, it might be the case that the current provided by the UV(R) 1xx power supply unit for the fans of the UMs does not suffice to guarantee safe and reliable operation of the fans. The sum of the currents must not exceed the maximum current provided by the UV(R).

If it is exceeded, then an MS 111 must be used in the inverter row where the current consumption is very high.

With the MS 111, the 24 V from an external 24 V power supply unit are fed to the unit bus X79 in order to ensure reliable operation of the fans, and therefore the reliable cooling of the inverters. The 24 V supply that is routed via X79C is interrupted internally at the MS 111.



Note

With the MS 111, an additional power supply unit must be used for the 24 V power supply of the fans.

3.9.2 Double-row configuration

Components and cables for double-row configuration:

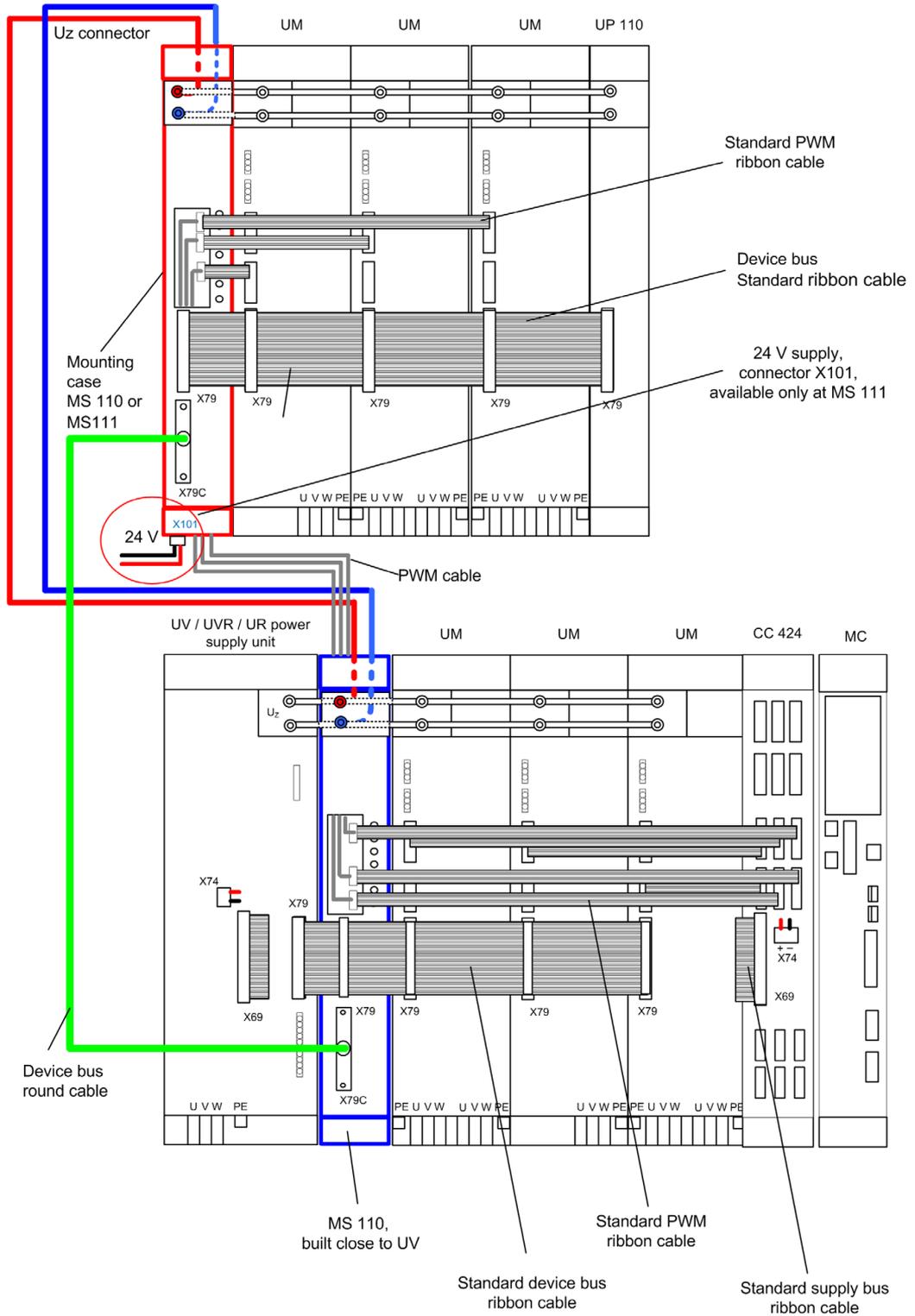
Component/Cable	ID
Unit bus cable (shielded, round) with 37-pin D-sub at both ends; max. length: 3 m	ID 664023-xx
PWM cable (round) with ribbon connector at both ends; max. length: 5 m	ID 664332-xx
Supply bus cable (round) with ribbon connector at both ends; max. length: 5 m (only necessary if the UV(R) 1xxD is not in the same row as the CC/MC)	ID 361508-xx
Wire for DC-link (16 mm ² , shielded, color: red); max. length: 3 m	ID 655440-xx
Wire for DC-link (16 mm ² , shielded, color: blue); max. length: 3 m	ID 655438-xx

When using a double-row configuration, please keep the following in mind:

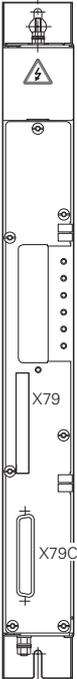
- The litz wires used for the DC-link connection of the power modules in the "second row" must not be longer than 3 m.
- Litz wires with 16 mm² cross section make a DC-link current of approx. 67 A possible. In a regenerative system, this results in approx. 35 kW of continuous power for the system connected by these wires.
- In a nonregenerative system the resulting maximum power is approx. 25 kW.
- Use fast-acting semiconductor fuses for protection of the UV(R) 1xxD on the primary side.
- The length of the unit bus ribbon cable must not exceed 1 m!
- If necessary, place the MS 110 or MS 111 in the "second row" in the center of the UMs.
- When calculating the length of the ribbon cables, make sure to include the module width of the MS 110 or MS 111.



Basic circuit diagram for double-row configuration:



3.9.3 Connection overview

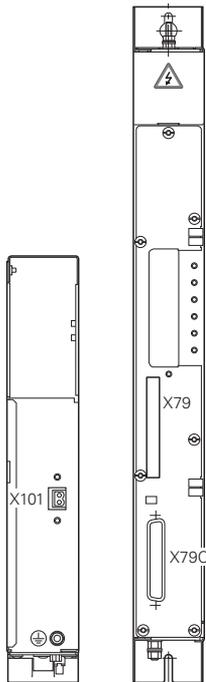
MS 110 pin layout	Connector	Function	Page
	DC-link conductor bar	400 V– to 750 V–	–
	Screw strip	For fastening the PWM round cable shields	–
	X79	Unit bus (connection for ribbon cable)	252
	X79C	Unit bus (connection for round cable)	253
		Protective ground	



Attention

Do not engage or disengage any connecting elements while the unit is under power!



MS 111 pin layout	Connector	Function	Page
	DC-link conductor bar	400 V– to 750 V–	–
	Screw strip	For fastening the PWM round cable shields	–
	X79	Unit bus (connection for ribbon cable)	252
	X79C	Unit bus (connection for round cable)	253
	LED	+24 V supply for fans is available	
	X101	Connection for +24 V supply for fans	253
		Protective ground	



Attention

Do not engage or disengage any connecting elements while the unit is under power!

40-pin ribbon connector	Assignment
1a to 3b	0 V *1
4a	+24 V *1
4b	+24 V *1
5a	+15 V *1
5b	+24 V *1
6a	+15 V *1
6b	+15 V *1
7a to 8b	Do not assign
9a	Reserved (SDA)
9b	Do not assign
10a	Reserved (SCL)
10b	$\overline{\text{ERR.TEMP}}$
11a	$\overline{\text{PF.PS}}$
11b	0 V
12a	$\overline{\text{RES.PS}}$
12b	0 V
13a	$\overline{\text{PWR.OFF}}$
13b	0 V
14a	5 V FS (spindle enable)
14b	0 V
15a	5 V FA (axis enable)
15b to 16b	0 V
17a and 17b	-15 V
18a and 18b	+15 V
19a to 20b	+5 V

These voltages must not be linked with other voltages (only basic insulation)!



Danger

The interface complies with the requirements of EN 61800-5-1 for low voltage electrical separation (except for 1a to 6b).

X79C: Unit bus

Connection:

Round cable connector 37-pin	Assignment
17 -19, 35 -37	0 V *1
16	+24 V *1
34	+24 V *1
15	+15 V *1
33	+24 V *1
14	+15 V *1
32	+15 V *1
12, 13	Do not assign
11	Reserved (SDA)
30, 31	Do not assign
29	Reserved (SCL)
10	$\overline{\text{ERR.TEMP}}$
28	$\overline{\text{PF.PS}}$
9	0 V
27	$\overline{\text{RES.PS}}$
0	0 V
26	$\overline{\text{PWR.OFF}}$
25	0 V
7	5 V FS (spindle enable)
6	5 V FA (axis enable)
5, 23, 24	0 V
4, 22	-15 V
3, 21	+15 V
1, 2, 20	+5 V

These voltages must not be linked with other voltages (only basic insulation)!

**Danger**

The interface complies with the requirements of EN 61800-5-1 for low voltage electrical separation (except for 1a to 6b).

X101: Power supply

Pin layout:

Connecting terminals	Assignment
+ / terminal 1	+24 V NC
- / terminal 2	0 V NC

3.10 UxC 11x (FS): Power supply and motor connection

3.10.1 UMC 111 (FS)

General information

Number of available control loops

- UMC 111 (FS): 4 control loops

Specifications	UMC 111FS 4 axes
External DC-link voltage	565 V
Power loss	Max. 400 W
Rated current at a PWM frequency of 3333 Hz	9.0 A
4000 Hz	8.3 A
5000 Hz	7.5 A
6666 Hz	6.3 A
8000 Hz	5.5 A
10000 Hz	4.6 A
Maximum current ^a at a PWM frequency of 3333 Hz	18.0 A
4000 Hz	16.5 A
5000 Hz	15.0 A
6666 Hz	12.6 A
8000 Hz	11.0 A
10000 Hz	9.2 A
Current consumption +24 V (incl. 4 speed encoders)	Max. 2.6 A
Current consumption of PLC outputs	Max. 48 x 0.15 A = 7.2 A
Weight	Approx. 20 kg

- a. 0.2 s cyclic duration factor for duty cycle time of 10 s with 70 % rated current preload

X81: Axis motor 1
X82: Axis motor 2
X83: Axis motor 3
X84: Axis motor 4

Connection:

Connecting terminals	Assignment
U	Motor connection U
V	Motor connection V
W	Motor connection W

For information on synchronous motors, asynchronous motors and power cables, refer to the Technical Manual on Inverter Systems and Motors, "Motors for Axis and Spindle Drives" chapter.

X103: 24 V input

Connecting terminals	Assignment
1	+24 V NC or +24 V PLC (max. 2.6 A)
2	0 V NC



Note

The integrated logic part, the inverter electronics and the integrated controller unit of the UMC are supplied via X103. The UMC generates the individual internal supply voltages with an integrated power supply unit, which complies with Protective Extra Low Voltage (PELV) according to EN 61800-5-1. The internally generated voltages thus have double basic insulation. This means that a +24 V PLC voltage with simple basic insulation (ELV as per EN 61800-5-1) is sufficient as supply via X103.

X33: DC-link power supply

Connection at UMC 11x(FS):

Connecting terminals	Assignment
1	+U _Z
2	-U _Z



Note

The line cross section should be 10 mm².

For connection to a regenerative inverter system, use shielded cables for reasons of EMC. Route the cables closely to each other and ground on both ends.

Cable lengths exceeding 3 m require fuse protection (e.g. SITOR 20 A fast).

X344: 24 V supply for motor holding brakes

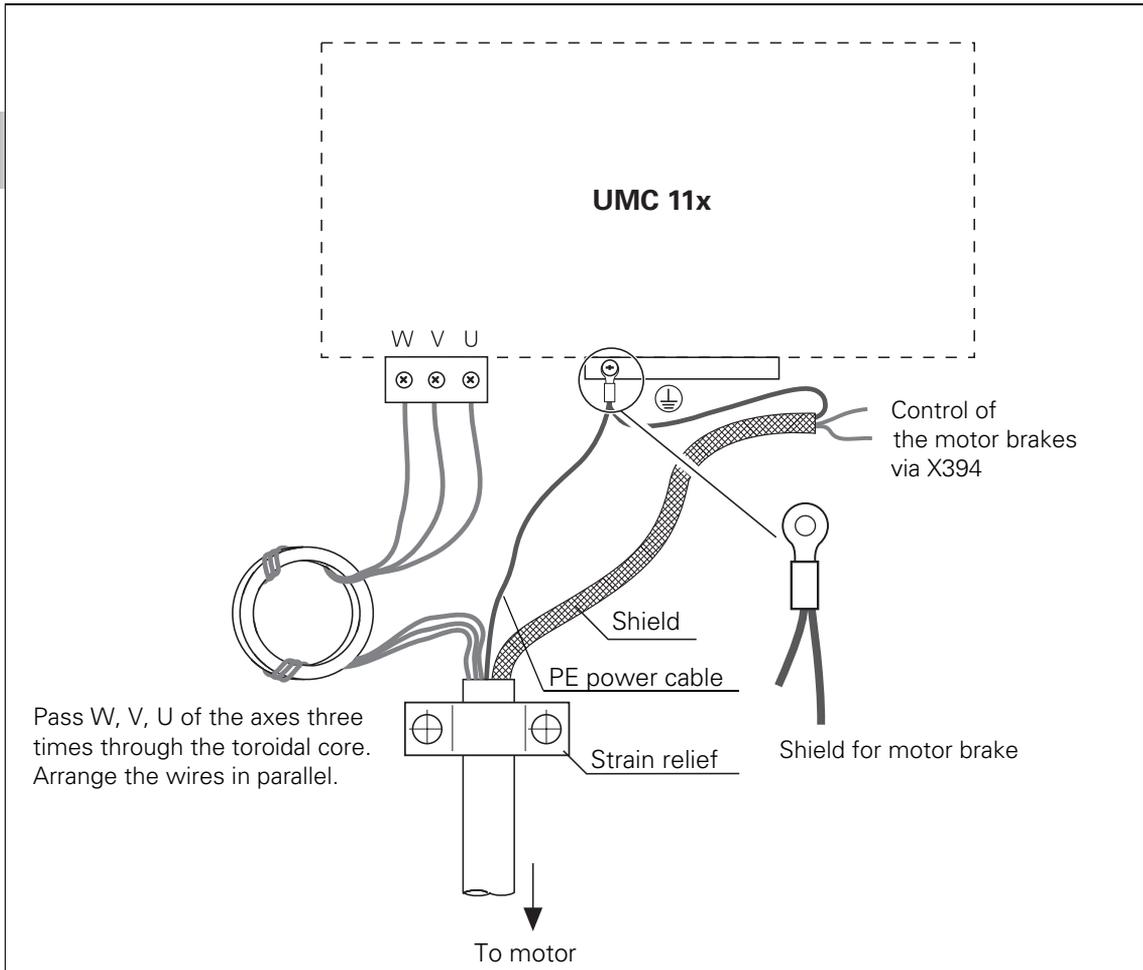
Connecting terminals	Assignment
1	+24 V PLC
2	0 V PLC

X394: Motor holding brakes

Connecting terminals	Assignment
1	Holding brake X80
2	0 V PLC
3	Holding brake X81
4	0 V PLC
5	Holding brake X82
6	0 V PLC
7	Holding brake X83
8	0 V PLC

Mounting the toroidal cores

To suppress conducted interference, toroidal cores must be mounted in the motor leads (X80 to X83).



3.10.2 UEC 11x (FS)

General information

Number of available control loops

- UEC 111 (FS): 4 control loops
- UEC 112 (FS): 5 control loops

Specifications	UEC 112 (FS)			UEC 111 (FS)		
	3 axes	1 axis	Spindle	2 axes	1 axis	Spindle
Supply voltage	3 x 400 V to 480 V ~ ±10 % (50 Hz to 60 Hz)					
DC-link voltage	565 V– (with supply voltage of 400 V)					
Power loss	Approx. 450 W			Approx. 450 W		
Rated current at a PWM frequency of 3333 Hz	6.0 A	9.0 A	24.0 A	6.0 A	9.0 A	24.0 A
4000 Hz	5.5 A	8.3 A	22.0 A	5.5 A	8.3 A	22.0 A
5000 Hz	5.0 A	7.5 A	20.0 A	5.0 A	7.5 A	20.0 A
6666 Hz	4.2 A	6.3 A	16.8 A	4.2 A	6.3 A	16.8 A
8000 Hz	3.6 A	5.5 A	14.6 A	3.6 A	5.5 A	14.6 A
10000 Hz	3.0 A	3.0 A	12.2 A	3.0 A	3.0 A	12.2 A
Peak power 6-40 % ^a	18 kW			18 kW		
DC-link power	14 kW			14 kW		
Maximum current ^b at a PWM frequency of 3333 Hz	12.0 A	18.0 A	36.0 A	12.0 A	18.0 A	36.0 A
4000 Hz	11.0 A	16.5 A	33.0 A	11.0 A	16.5 A	33.0 A
5000 Hz	10.0 A	15.0 A	30.0 A	10.0 A	15.0 A	30.0 A
6666 Hz	8.4 A	12.6 A	25.2 A	8.4 A	12.6 A	25.2 A
8000 Hz	7.3 A	11.0 A	21.9 A	7.3 A	11.0 A	21.9 A
10000 Hz	6.0 A	6.0 A	18.3 A	6.0 A	6.0 A	18.3 A
Integral braking resistor						
Continuous power	2.1 kW			2.1 kW		
Peak power ^c	27 kW			27 kW		
Resistance	18 ohms			18 ohms		
Load capacity +24 V NC	3.5 A			3.5 A		
Weight	Approx. 20 kg			Approx. 20 kg		

- a. Spindle: 40 % cyclic duration factor for duty cycle time of 10 minutes (S6-40 %)
- b. Axes: 0.2 s cyclic duration factor for duty cycle time of 10 s with 70% rated current preload
Spindle: 10 s cyclic duration factor for duty cycle time of 60 s with 70% rated current preload
- c. 1.5 % cyclic duration factor for duration of 120 s



Note

Under some circumstances, certain high-speed spindles cannot be satisfactorily controlled with 3.3 kHz PWM frequency. In this case, increase the PWM frequency of the spindle. The spindle (X80) and the 1st axis (X81) share a controller group. You therefore have to set the same PWM frequency for X80 and X81. (Note the D rating!)

For more information about the controller groups of the UEC 11x (FS), see page 1067.

X31: UEC power supply



Danger

Danger of electrical shock!

The UEC 11x controller unit must be opened only by HEIDENHAIN service engineers.

Do not engage or disengage any terminals while they are under power.



Note

EN 61800-5-1 requires a non-detachable connection to the line power supply.



Note

If the power supply is other than 400 V/ 480 V, an autotransformer is required. It must comply at least with the connection specifications of the UEC 11x.

With a power supply of 400 V, the inverter voltage U_z is 565 V~, and with a power supply of 480 V it is 678 V~.

For information on the power connection, refer to the Technical Manual for "Inverter Systems and Motors."

Connecting terminals	UEC 111, UEC 112
Operation on 400 V~	
L1	400 V~ ± 10 % 50 Hz to 60 Hz
L2	
L3	
	Cable / single conductor (HT wire): 6 mm ² (AWG 10) Single conductor H07 V2-K: 4 mm ² (AWG 10) Line fuse: 25 A (gR) Siemens Sitor type Grounding terminal: ≥ 10 mm ² (AWG 6)
	Tightening torque for connecting terminals: 0.7 Nm (6.5 to 7 pound-inch)
Operation on 480 V~	
L1	480 V~ ± 10 % 50 Hz to 60 Hz
L2	
L3	
	Cable / single conductor (HT wire): 6 mm ² (AWG 10) Single conductor H07 V2-K: 4 mm ² (AWG 10) Line fuse: 25 A (gR) Siemens Sitor type Grounding terminal: ≥ 10 mm ² (AWG 6)
	Tightening torque for connecting terminals: 0.7 Nm (6.5 to 7 pound-inch)



X80: Spindle motor
X81: Axis motor 1
X82: Axis motor 2
X83: Axis motor 3
X84: Axis motor 4

Connection:

Connecting terminals	Assignment
U	Motor connection U
V	Motor connection V
W	Motor connection W

For information on synchronous motors, asynchronous motors and power cables, refer to the Technical Manual on Inverter Systems and Motors, "Motors for Axis and Spindle Drives" chapter.

X71: Safety relay for spindle
X72: Safety relay for axes

For information on the wiring and function, see the Basic Circuit Diagram for your control. Registered customers can download the Basic Circuit Diagram from the HEIDENHAIN FileBase on the Internet.

Connecting terminals X71 to X72	Assignment
1	+24 V pulse release output (max. 250 mA) for control of the relays at X71.3 and X72.3 for drive enabling (Axis ON, Spindle ON).
2	0 V for pulse release output
3	+24 V pulse release input for Axis ON, Spindle ON
4	Do not assign
5	Do not assign
6 ^a	Normally closed contact (OE1, OE1A or OE1S)
7 ^a	Normally closed contact (OE2, OE2A or OE2S)

a. Max. 125 V



Note

The +24 V pulse release voltage at terminals X71.1 and X72.1 is generated internally by a separate power supply unit of the UxC 11x (FS). Use this voltage exclusively for drive enabling—for supplying the relay coils that are internally connected to X71.3 and X72.3.



Attention

The +24 V pulse release voltage must not be linked with other voltages (e.g. +24 V NC or +24 V PLC) of the HEIDENHAIN control system.



Attention

A recovery diode is required in the proximity of inductive loads, e.g. relay or contactor coils.



X89: Braking resistor

Connection at the UEC 11x:

Connecting terminal X89 UE 11x	Assignment	PW 21x	PW 1x0(B); connecting terminal X1
1	+U _Z	RB1	1
2	Switch to -U _Z	RB2	2

On the UEC 11x compact inverters, there must be no bridge at connector X89. A bridge at X89 would cause the internal braking resistor to be bridged and lead to a short circuit of the DC-link voltage. If necessary, you can connect an external braking resistor (PW 210) via X89 in parallel to the internal braking resistor.

X90: 24 V NC output

Connecting terminal X90	Assignment
+ or terminal 1	+24 V NC (max. 3.5 A)
- or terminal 2	0 V



Attention

The interface is electrically separated (protective extra-low voltage (PELV) as per EN 61800-5-1).

X344: 24 V supply for motor holding brakes

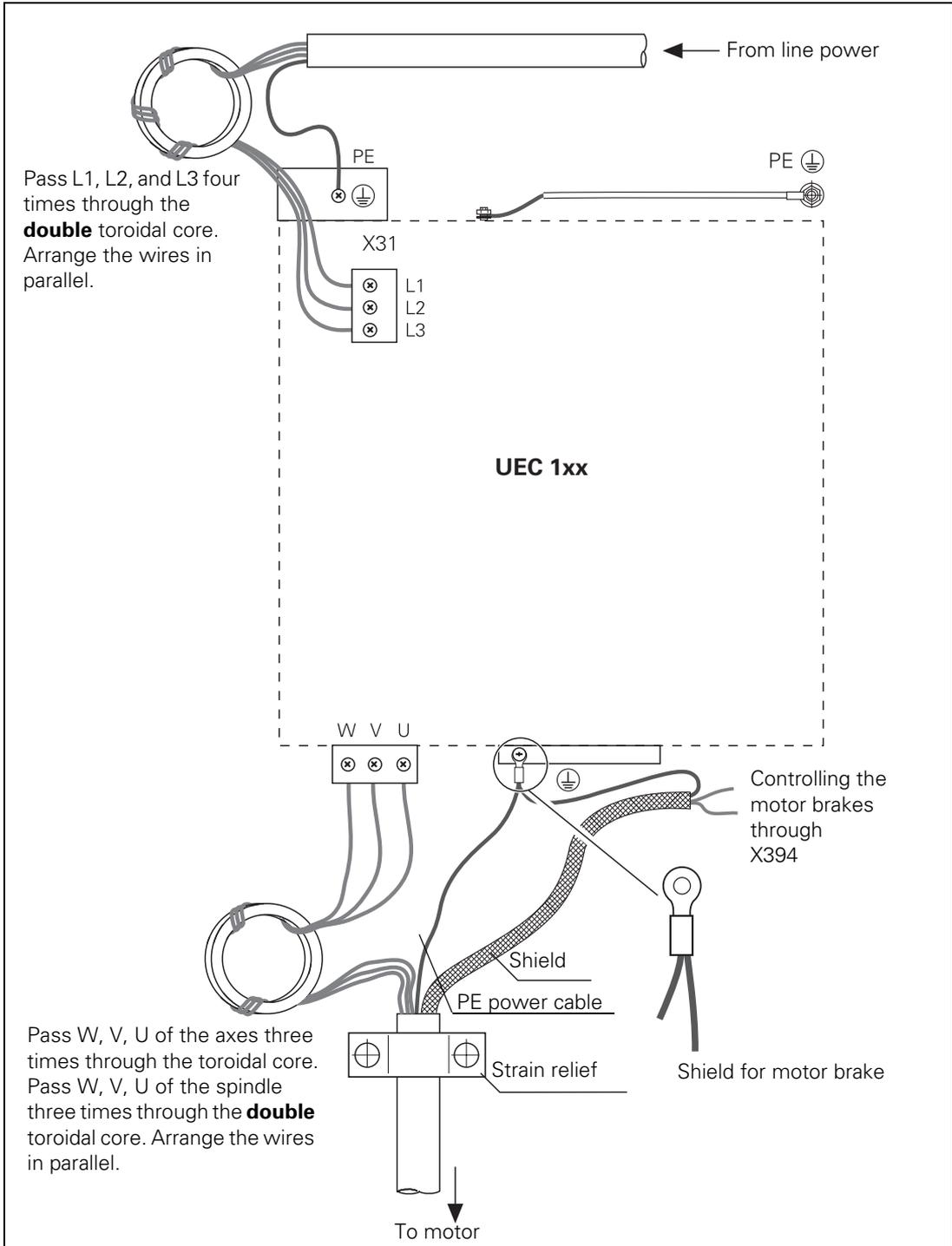
Connecting terminals	Assignment
1	+24 V PLC
2	0 V PLC

X394: Motor holding brakes

Connecting terminals	Assignment
1	Holding brake X80
2	0 V PLC
3	Holding brake X81
4	0 V PLC
5	Holding brake X82
6	0 V PLC
7	Holding brake X83
8	0 V PLC
9	Holding brake X84
10	0 V PLC

Mounting the toroidal cores

To suppress conducted interference, toroidal cores must be mounted in the motor leads (X80 to X84) and in the voltage supply lead (X31).



3.11 UxC 11x (FS): Meaning of the LEDs

On the front of the UxC 11x (FS) are several LEDs for functional control, with the following meaning:

UxC 11x	LED	Meaning	Signal direction	Signal
	NC RESET	Reset signal from the MC computer unit to the UxC	MC → UxC	RES.LE
	PWR FAIL	U_z too low, $U_z < 410$ V (e.g. failure of a phase under load, power < 290 V)	UxC → MC	PF.PS
	PWR RES	Reset signal from the UxC to the MC computer unit	UxC → MC	RES.PS
	READY	Inverter ready	UxC → MC	RDY
	TEMP >>	Temperature of heat sink too high (> 100 °C)	UxC → MC	ERR.TEMP
	U_{DC} LINK >>	U_z too high (> approx. 850 V); power modules are switched off	UxC → MC	ERR.UZ.GR
	STO A (RED)	Safe Torque Off; no enable from control (main contactor not active, DSP error, PLC error with emergency stop, hardware or software error of MC, CC) Axis/Spindle enabled	MC → UEC	STO.A.x
	READY (GREEN)		UxC → MC	RDY
	STO B	Safe Torque Off; no drive enable from control (e.g. by the PLC, active via external signal or STO A)	MC → UxC	STO.B.x

UxC 11x	LED	Meaning	Signal direction	Signal
LED at X9 on top surface of the UxC	PL green	<ul style="list-style-type: none"> ■ PL not ready ■ PL ready ■ PL initialization ■ Error with SS2 reaction is present ■ Error with SS1F reaction is present 	<ul style="list-style-type: none"> ■ Off ■ On ■ Slow blinking ■ Blinking twice ■ Fast blinking 	
	HSCI green	<ul style="list-style-type: none"> ■ Not ready for HSCI communication ■ Ready for HSCI communication ■ Error in HSCI communication ■ No HSCI communication 	<ul style="list-style-type: none"> ■ Off ■ On ■ Blinking twice ■ Fast blinking 	



3.12 Power supply for PLC outputs

The PLC outputs of the PLB 62xx are powered by the 24 V control voltage of the machine (PELV in accordance with EN 61800-5-1). The power to the PLC outputs is supplied via the corresponding terminals on the respective I/O module connectors for PLC outputs.

The control voltage must be smoothed with a capacitance of 150 μF per amp of rated current, and in any case with at least 1000 μF . At a current load of 15 A, for example, this corresponds to a capacitance of 2250 μF . If the PSL 130 is used as 24 V– supply unit, this additional smoothing is not necessary.



Note

HEIDENHAIN recommends the PSL 130 (575047-01) as 24 V power supply unit, see "PSL 13x low-voltage power supply unit" on page 237.

EN 61 131-2:1994 permits:

- Minimum absolute value: 20.4 V–
- Maximum absolute value: 25.4 V– at 200 W power output
- Maximum absolute value: 28.8 V– at 100 W power output



Attention

Use only original replacement fuses.

Power consumption

If half of the outputs are switched at the same time, the following are the values for power consumption:

PL 6xxx: approx. 485 W
UxC 11x: 48 W

Power output

The maximum permissible power output of a PLD-H xx-xx-xx is 200 W.

**Rated operating
current per output**

UxC 11x: 0.150 A

PLD-H xx-xx-xx: 2 A

Simultaneity with a supply voltage of 25.4 V:

2 outputs with 4 A each

4 outputs with 2 A each

8 outputs with 1 A each

Total current:

Out0 to Out7: ≤ 8 A

Out0 to Out3: ≤ 4 A

Out4 to Out7: ≤ 4 A

For all PLD-H xx-xx-xx units, it must be remembered that a total current of max. 8 A per slot (PLD-H) must not be exceeded! This applies regardless of the number of outputs of the PLD-H. With the UxC 11x, a total current of 1.2 A for the eight outputs of a connector must not be exceeded.

In the event of an overload or short circuit and as a consequence of excessive temperature, the individual PLC outputs of a PLD or UxC switch off automatically. Then the outputs will try to switch on again cyclically.

Furthermore, the modules have a fuse that prevents an excessive total current per output module with eight outputs (> 8 A not self-healing on PLD, > 1.2 A self-healing on UxC) and thus destruction of the modules.



3.13 Power supply for PLB 6xxx (FS)



Note

The control cyclically monitors the supply voltage of the PL 6xxx.

X3: +24 V NC, +24 V PLC power supply

Power consumption of the PL 6xxx via X3 if every slot is used, including the TS, TT:

- At +24 V NC: max. 48 W
- At +24 V PLC: max. 21 W

For more details regarding the power supply, see page 229.

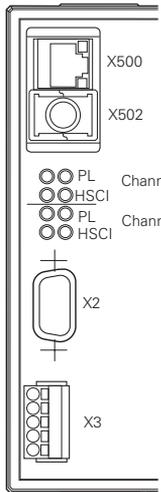
The power to the PLC outputs is also supplied via the corresponding terminals on the respective I/O module connectors for PLC outputs. The power consumption of the +24 V PLC via X3 and the power consumption of the PLC outputs add to each other.

Pin layout of X3:

Supply voltage for logic and PLC outputs

Connecting terminal	Assignment
1 (top terminal)	+ 24 V NC
2	0 V NC (ground + 24 V NC)
3	 Protective ground Minimum wire cross section of the power cables for 24 V PLC
4	+ 24 V PLC
5 (bottom terminal)	0 V PLC (ground + 24 V PLC)

3.14 Meaning of the LED on PLB 6xxx

PLB 6xxx	LED	LED status	Meaning
	PL green (right)	<ul style="list-style-type: none"> ■ Off ■ On ■ Slow blinking 	<ul style="list-style-type: none"> ■ PL not ready ■ PL ready ■ PL initialization
	PL yellow (left)	<ul style="list-style-type: none"> ■ Off ■ Fast blinking ■ Blinking twice 	<ul style="list-style-type: none"> ■ No error ■ Error with SS1F reaction is present ■ Error with SS2 reaction is present
	HSCI green (right)	<ul style="list-style-type: none"> ■ Off ■ On 	<ul style="list-style-type: none"> ■ PL not ready for HSCI communication ■ PL ready for HSCI communication
	HSCI yellow (left)	<ul style="list-style-type: none"> ■ Off ■ Fast blinking ■ Blinking twice 	<ul style="list-style-type: none"> ■ No HSCI communication error ■ No HSCI communication ■ Error in HSCI communication
	Channel A/Channel B refers to the safety channels A and B of the PLB 6xxxFS		

3.15 Power supply for control-is-ready signal

X9: Power supply for control-is-ready signal

The control-is-ready signal output is powered by 24 V– provided by the UE 2xxB inverter or the UV 1xx power supply unit. The voltage is connected with terminal X9 of the PLB 620x.

Pin layout:

Connecting terminal X9	Assignment	Connection when using a HEIDENHAIN inverter
1a	+24 V	X72/1
2b	0 V	X72/2

Power consumption via X9, +24 V PLC: max. 120 W (depending on the connection of PLC outputs to X9)



3.16 Digital PLC inputs/outputs

Input signals and addresses

Input signals of the switching inputs:

Voltage range	PLD-H (with LED)	UEC 11x, X9 of PL 62xx, and machine operating panel (without LED)
"1" signal: U_i	11 V to 30.0 V	11 V to 30.0 V
"0" signal: U_i	-3.0 V to 2.2 V	-3 V to 2.2 V

Current range	PLD-H (with LED)	UEC 11x, X9 of PL 62xx, and machine operating panel (without LED)
"1" signal: I_i	2.0 mA to 6.1 mA	2.1 mA to 6.0 mA
"0" signal: I_i at $U_i = 2.2$ V	0.3 mA	0.43 mA

If 24 V DC is present, and therefore a "1" signal is transmitted, the PLC inputs of the HSCI system consume a current of 5 mA.

Output signals and addresses

The switching outputs are transistor outputs with current limitation.

Please note:

- Permissible load: Resistive load (ohmic load). Inductive loads (e.g. relay, contactor) with an energy content of up to 100 mJ do not require a quenching diode. If the energy content exceeds 100 mJ: only with quenching diode parallel to inductance. Pay attention to the manufacturer's specification of the energy content when selecting the switching devices.
- For the rated operating currents of the PLC outputs, see "Rated operating current per output" on page 266.
- If an output is operated with an inductive load without a quenching diode and is read back to an input, the input must be protected by varistors or RC circuits.
- PLD-H: The outputs are short-circuit proof.
- For component-related reasons, the switching outputs should be loaded with at least 5 mA in "1" state. They conform to EN 61131-2. If a resistive load consumes less than 5 mA, it is necessary to either insert a relay or perform a usability test in accordance with the calculation described in the following, see "Calculation of the voltage drop in "0" state" on page 270.
- For component-related reasons, a current of $I_{Off} = 500 \mu\text{A}$ flows through the switching outputs also in "0" state. If high-impedance loads with a low-level lower switching threshold are connected directly to the output, the voltage drop can lead to a "1" state. In such a case, a shunt resistor must be connected to the output, see "Calculation of the voltage drop in "0" state" on page 270.
- If the holding brakes of motors are not driven exclusively over the inverter outputs for brakes, a protective circuit in the form of a varistor must be used. Due to the inductance of the holding brakes and any relays used, a voltage peak that may exceed 1000 V occurs when the exciting current is switched off. This may destroy other electronics, such as connected PLC inputs/ outputs.
- A protective circuit is not necessary only if the holding brakes are driven exclusively over the inverter outputs for brakes, because the voltage is limited internally by electronic switches in the inverters.

Output signals:

	PLD-H
Min. output voltage for "1" signal	3 V below supply voltage



Attention

PLC outputs must neither be connected to a 24 V supply, nor to other PLC outputs with a difference in potential. Otherwise, the voltage present at the PLC outputs is transmitted to the power supply. As a result, the PLC outputs that can be switched off may nevertheless be supplied with this voltage.

Calculation of the voltage drop in "0" state

Current ($I_{\text{Off}} = 500 \mu\text{A}$) also flows in "0" state of the PLC output for component-related reasons.

If resistive loads with high input impedance ($> 15 \text{ k}\Omega$) and a low-level lower switching threshold are operated directly (without an interconnected relay) on the PLC output, a shunt resistor may be necessary under certain circumstances. Typical values of the shunt resistor are e.g. $5 \text{ k}\Omega$ or $10 \text{ k}\Omega$. The voltage at the output in "0" state should be clearly (approx. 50 %) below the lower switching threshold of the resistive load.

Calculation example—Resistive load does not require shunt resistor:

Reversing load relay directly on PLC output, data sheet specifications:

Rated control voltage U_c :	24 V DC
Rated operating current I_c :	3 mA
Switching threshold "0" state U_{Low} :	9.6 V
Switching threshold "1" state U_{High} :	19.2 V

In "0" state of the output, the voltage drop has to be less than 9.6 V. Calculation of the input resistance R_c of the reversing load relay:

$$R_c = \frac{U_c}{I_c} = \frac{24 \text{ V}}{3 \text{ mA}} = 8 \text{ k}\Omega$$

Calculation of the voltage drop U_{Low} in "0" state at the output:

$$U_{\text{Low}} = R_c \cdot I_{\text{Off}} = 8 \text{ k}\Omega \cdot 500 \mu\text{A} = 4 \text{ V}$$

With 4 V, the voltage drop is clearly below the switching threshold for the "0" state (9.6 V). The reversing load relay can be operated directly at the PLC output.



Calculation example—Resistive load requires shunt resistor:

Interlocking switch directly at PLC output, data sheet specifications:

Rated control voltage U_c :	28 V DC
Rated operating current I_c :	2.2 mA
Switching threshold "0" state U_{Low} :	3.6 V
Switching threshold "1" state U_{High} :	17.4 V

In "0" state of the output, the voltage drop has to be less than 3.6 V .

Calculation of the input resistance R_c of the interlocking switch:

$$R_c = \frac{U_c}{I_c} = \frac{28 \text{ V}}{2.2 \text{ mA}} = 12.73 \text{ k}\Omega$$

Calculation of the voltage drop U_{Low} in "0" state at the output:

$$U_{Low} = R_c \cdot I_{Off} = 12.73 \text{ k}\Omega \cdot 500 \mu\text{A} = 6.36 \text{ V}$$

The shunt resistor at the output should be such that the voltage drop at the output in "0" state goes clearly below the lower switching threshold, e.g. $R_p = 5 \text{ k}\Omega$:

$$\frac{1}{R_{Total}} = \frac{1}{R_c} + \frac{1}{R_p}$$

$$\frac{1}{R_{Total}} = \frac{1}{12.73 \text{ k}\Omega} + \frac{1}{5 \text{ k}\Omega}$$

$$R_{Total} = 3.59 \text{ k}\Omega \quad U_{Low} = 3.59 \text{ k}\Omega \cdot 500 \mu\text{A} = \mathbf{1.79 \text{ V}}$$

With a voltage drop of 1.79 V in "0" state, the value falls below the lower switching threshold of the interlocking switch. If the shunt resistor of 5 k Ω is used, operation directly at the PLC output is possible in this case.

**Terminal X9 on PLB
62xx (FS)**

Properties	Socket connectors on the PLA-H 08-04-04
Connection:	Socket connector with tension clamp connection, type: Weidmüller B2L 3.5/30 SN SW 2-row, 30-pin
Connectable conductors:	Usable conductor cross sections without wire-end sleeve: 0.08 mm ² to 1.0 mm ² Usable conductor cross sections with wire-end sleeve: 0.14 mm ² 0.34 mm ² 0.5 mm ² (only with Weidmüller PZ 6/5 crimping pliers)



Note

HEIDENHAIN recommends:

Preferably use a conductor cross section of 0.34 mm^2 if you use stranded wires with wire-end sleeves. This cross section can be clamped appropriately and ensures a reliable terminal connection.

If you use stranded wires with wire-end sleeves and a conductor cross section of 0.5 mm^2 , the Weidmüller PZ 6/5 crimping pliers (setting $0.25\text{--}0.5 \text{ mm}^2$) must be used for crimping. In this case, orient the crimped wire-end sleeve before inserting it into the socket connector. If crimping pliers from other manufacturers are used for crimping conductors with a cross section of 0.5 mm^2 , the crimped wire-end sleeves cannot be inserted into the socket connector and clamped appropriately, and therefore do not result in a reliable terminal connection.



X9: Safety-related PLC inputs/outputs

Pin layout of PLB 620x:

The triggering outputs at X9 each supply up to 150 mA of output current. The only exceptions are the two outputs -STOS.A.G and -STO.A.G with max. 2 A of output current. In the event of an overload or short circuit and as a consequence of excessive temperature, the individual PLC outputs switch off automatically. Then the outputs will try to switch on again cyclically. Seven outputs and twelve inputs are available at X9 of a PLB 620x for free use. Further PLC inputs/outputs must be realized by means of I/O modules.

Terminal	NEW signal design.	OLD connector/signal design. (MC 42xC)	Assignm. / Function
1a	24 V.A	X34	24 V supply of the outputs MC.RDY, O.0 to O.2
2a	Do not assign	–	–
3a	MC.RDY ^a	–SH1A (safe stop) X41.34 / O33	24 V output: (safe torque off) Control-is-ready signal
4a	O0		24 V outputs (high-side driver)
5a	O1		
6a	O2		
7a	–ES.A	–NE1 / X42.4 / I3 Acknowledgment: "Control is ready"	24 V input Emergency Stop input 1
8a	I0		24 V inputs (PLC)
9a	I1		
10a	I2		
11a	I3		
12a	I4		
13a	I5		
14a	–PF.PS.AC	–PF.PS.AC (signal at X69)	24 V outputs for powerfail
15a	–PF.PS.DC	–PF.PS.ZK (signal at X69)	
1b	24 V.B	X44	24 V supply of the outputs O.3 to O.6
2b	0 V		0 V PLC for all I/Os
3b	O3 ^a		24 V outputs (high-side driver)
4b	O4		
5b	O5		
6b	O6		
7b	–ES.B		
8b	I6		24 V inputs (PLC)
9b	I7		
10b	I8		
11b	I9		
12b	I10		
13b	I11		
14b	–SP.REF	X30	Rapid 24 V input for spindle REF
15b	Do not assign	–	–

a. 2 A outputs

X9 – Safety: Safety-related PLC inputs/ outputs

Pin layout of PLB 620x FS:

The triggering outputs at X9 each supply up to 150 mA of output current. The only exceptions are the two outputs -STOS.A.G and -STO.A.G with max. 2 A of output current. In the event of an overload or short circuit and as a consequence of excessive temperature, the individual PLC outputs switch off automatically. Then the outputs will try to switch on again cyclically.

Two outputs and six inputs are available at X9 of a PLB 620x FS for free use. Further PLC inputs/outputs must be realized by means of I/O modules.

Terminal	NEW signal design.	OLD connector/signal design. (MC 42xC)	Assignm. / Function
1a	24 V.A	X44	24 V supply of the outputs MC.RDY, O.0 to O.1
2a	-STOS.A.G ^a	-SHS1A (safe stop of spindle) X41.32	24 V output: (spindle safe torque off)
3a	-STO.A.G ^a	-SH1A (safe stop) X41.34 / O33	24 V output: (safe torque off) Control-is-ready signal
4a	TEST.A	T.2 / X165.2	24 V output for emergency stop chain
5a	O0.A		24 V outputs (high-side driver)
6a	O1.A		
7a	-ES.A		
8a	I0.A		24 V inputs (PLC)
9a	I1.A		
10a	I2.A		
11a	I3.A		
12a	I4.A		
13a	I5.A		
14a	-PF.PS.AC	-PF.PS.AC (signal at X69)	24 V outputs for powerfail
15a	-PF.PS.DC	-PF.PS.ZK (signal at X69)	
1b	24 V.B	X44	24 V supply of the outputs O.0 to O.1
2b	0 V		0 V PLC for all I/Os
3b	Do not assign		24 V outputs (high-side driver)
4b	TEST.B	T.1 / X165.1	24 V output for emergency stop chain
5b	O0.B		24 V outputs (high-side driver)
6b	O1.B		
7b	-ES.B	-NE2 / X42.33 / I32 "Drive enabling"	24 V input Emergency Stop input 2



Terminal	NEW signal design.	OLD connector/signal design. (MC 42xC)	Assignm. / Function
8b	I0.B		24 V inputs (PLC)
9b	I1.B		
10b	I2.B		
11b	I3.B		
12b	I4.B		
13b	I5.B		
14b	-SP.REF	X30	Rapid 24 V input for spindle REF
15b	Do not assign	-	-

a. 2 A outputs

PLC inputs on the PLD-H

Pin layout on the PLD-H xx-xx-xx input/output module:



Note

The 0 V terminals of X11, X12 and X14 of the PLD-H are connected internally. These connections are used for connecting the potential of the electronics and for operating the LEDs. Since only a low current is required (max. 50 mA), it is sufficient to establish only one 0 V connection (preferably at X11).

X11: PLC inputs, channel A

Assignment	Terminal									
	1	2	3	4	5	6	7	8	9	10
PL 6xxx Slot 1	0 V PLC	0 V PLC	I0	I1	I2	I3	I4	I5	I6	I7

X12: PLC inputs, channel A

Assignment	Terminal									
	1	2	3	4	5	6	7	8	9	10
PL 6xxx Slot 1	0 V PLC	0 V PLC	I8	I9	I10	I11	I12	I13	I14	I15

X14: PLC inputs, channel B

Assignment	Terminal									
	1	2	3	4	5	6	7	8	9	10
PL 6xxx Slot 1	0 V PLC	0 V PLC	I0.B	I1.B	I2.B	I3.B	I4.B	I5.B	I6.B	I7.B

X17: PLC inputs, channel A/B

Assignment	Terminal									
	1	2	3	4	5	6	7	8	9	10
PL 6xxx Slot 1	0 V PLC	0 V PLC	I0.A	I1.A	I2.A	I3.A	I0.B	I1.B	I2.B	I3.B

Please note that a system with functional safety (FS) permits up to 256 safe, dual-channel inputs (channels A and B). An MB 6xx FS from HEIDENHAIN requires 56 of the safe inputs, and connector X9 of a system PL requires eight additional safe inputs. This means that max. 192 safe inputs can be distributed to the PLD-H xx-xx-xx FS.

The designation "Channel A" or ".A" for the PLC inputs is relevant only for I/O modules with functional safety (FS).

Fast PLC inputs

Only the first four slots of a PL 6xxx can be used for fast PLC inputs. The fifth slot and the successive slots (on PL 6x06, PL 6x08) must not be defined as fast PLC inputs.

The configuration of fast PLC inputs in the HSCI system corresponds to the previous configuration using machine parameters.



PLC outputs on the PLD-H Pin layout on the PLD-H xx-xx-xx input/output module:

X21: PLC outputs, channel A										
Assignment	Terminal									
	1	2	3	4	5	6	7	8	9	10
PL 6xxx Slot 1	O0.A	O1.A	O2.A	O3.A	O4.A	O5.A	O6.A	O7.A	24 V PLC for O0.A to O3.A	24 V PLC for O4.A to O7.A

X22: PLC outputs, channel A										
Assignment	Terminal									
	1	2	3	4	5	6	7	8	9	10
PL 6xxx Slot 1	O8.A	O9.A	O10.A	O11.A	O12.A	O13.A	O14.A	O15.A	24 V PLC for O8.A to O11.A	24 V PLC for O12.A to O15.A

X24: PLC outputs, channel B										
Assignment	Terminal									
	1	2	3	4	5	6	7	8	9	10
PL 6xxx Slot 1	O0.B	O1.B	O2.B	O3.B	O4.B	O5.B	O6.B	O7.B	24 V PLC for O0.B to O3.B	24 V PLC for O4.B to O7.B

X27: PLC outputs, channel A / channel B										
Assignment	Terminal									
	1	2	3	4	5	6	7	8	9	10
PL 6xxx Slot 1	O0.A	O1.A	O2.A	O3.A	O0.B	O1.B	O2.B	O3.B	24 V PLC for O0.A to O3.A	24 V PLC for O0.B to O3.B

Please note that a system with functional safety (FS) permits up to 126 safe, dual-channel outputs (channels A and B). Connector X9 of a system PL requires three of the outputs. This means that max. 123 safe outputs can be distributed to the PLD-H xx-xx-xx FS.

"Channel A" or ".A" of the PLC outputs needs to be entered only for I/O modules with functional safety (FS).

3.16.1 UxC 11x (FS): Digital PLC inputs/outputs

Type of terminals on the UxC 11x (FS)

Socket connectors X4, X5, X6, X104, X106 on the UxC 11x (FS)	
Connection:	Socket connector with tension clamp connection, type: Weidmüller B2L 3.5/24 SN SW 2-row, 24-pin
Connectable conductors:	Usable conductor cross sections without wire-end sleeve: 0.08 mm ² to 1.0 mm ² Usable conductor cross sections with wire-end sleeve: 0.14 mm ² 0.34 mm ² 0.5 mm ² (only with Weidmüller PZ 6/5 crimping pliers)



Note

HEIDENHAIN recommends:

Preferably use a conductor cross section of 0.34 mm² if you use stranded wires with wire-end sleeves. This cross section can be clamped appropriately and ensures a reliable terminal connection.

If you use stranded wires with wire-end sleeves and a conductor cross section of 0.5 mm², the Weidmüller PZ 6/5 crimping pliers (setting 0.25–0.5 mm²) must be used for crimping. In this case, orient the crimped wire-end sleeve before inserting it into the socket connector. If crimping pliers from other manufacturers are used for crimping conductors with a cross section of 0.5 mm², the crimped wire-end sleeves cannot be inserted into the socket connector and clamped appropriately, and therefore do not result in a reliable terminal connection.

The digital inputs/outputs described below are available on the UxC 11x (FS).

X4: Single-channel PLC inputs (on the front)

Connections on the front of the UxC 11x (FS):

- 18 single-channel PLC inputs are freely available:
I0 to I17

Terminal	Signal designation	Assignm. / Function
1a	+24 V PLC.01	24 V supply of the outputs MC.RDY, O16 to O22
2a	+24 V PLC.02	24 V supply of the outputs O8 to O15
3a	+24 V PLC.03	24 V supply of the outputs O0 to O7
4a	0 V PLC	0 V for all I/Os
5a	-SP.REF	Rapid 24 V input for spindle REF
6a	0 V PLC	0 V for all I/Os
7a	I12	24 V inputs
8a	I13	
9a	I14	
10a	I15	
11a	I16	
12a	I17	
1b	I0	24 V inputs
2b	I1	
3b	I2	
4b	I3	
5b	I4	
6b	I5	
7b	I6	
8b	I7	
9b	I8	
10b	I9	
11b	I10	
12b	I11	

X5: Single-channel PLC inputs (on the front)

Connections on the front of the UxC 11x (FS):

- 20 single-channel PLC inputs are freely available:
I18 to I37

Terminal	Signal designation	Assignm. / Function
1a	I30	24 V inputs
2a	I31	
3a	I32	
4a	I33	
5a	I34	
6a	I35	
7a	I36	
8a	I37	
9a	-ES.A	+24 V input Acknowledgment: "Control is ready"
10a	-ES.B	24 V input "Drive enabling"
11a	Do not assign	
12a	Do not assign	
1b	I18	24 V inputs
2b	I19	
3b	I20	
4b	I21	
5b	I22	
6b	I23	
7b	I24	
8b	I25	
9b	I26	
10b	I27	
11b	I28	
12b	I29	



Note

If the integrated PLC inputs do not suffice for your application, you can connect up to 7 additional external PL 61xx expansion PLs to the UxC 111FS via the HSCI interface.



X104— Safety:
Dual-channel PLC
inputs (on the front)

Connections on the front of the UxC 11x FS:

- 8 dual-channel PLC inputs:
 I0.A to I7.A
 I0.B to I7.B

Terminal	Signal designation	Assignm. / Function
1a	+24 V.A	24 V supply of the outputs O0.A to O7.A
2a	+24 V.B	24 V supply of the outputs O0.B to O7.B
3a	+24 V.C	24 V supply of the outputs O8.A to O15.A
4a	Do not assign	
5a	I4.B	24 V inputs
6a	I5.B	
7a	I6.B	
8a	I7.B	
9a	I4.A	
10a	I5.A	
11a	I6.A	
12a	I7.A	
1b	Do not assign	
2b	Do not assign	
3b	Do not assign	
4b	Do not assign	
5b	I0.B	24 V inputs
6b	I1.B	
7b	I2.B	
8b	I3.B	
9b	I0.A	
10b	I1.A	
11b	I2.A	
12b	I3.A	



Note

If the integrated PLC inputs do not suffice for your application, you can connect up to 7 additional external PL 61xx expansion PLs to the UxC 111FS via the HSCI interface.

X6: Single-channel PLC outputs (at the top)

Connections at the top of the UxC 11x:

- 23 single-channel PLC outputs
O0 to O22

Terminal	Signal designation	Assignm. / Function
1a	O4	24 V outputs, can be switched off via terminal X4.3a (+24 V PLC.03)
2a	O5	
3a	O6	
4a	O7	
5a	O12	24 V outputs, can be switched off via terminal X4.2a (+24 V PLC.02)
6a	O13	
7a	O14	
8a	O15	
9a	O20	24 V outputs, cannot be switched off
10a	O21	
11a	O22	
12a	MC.RDY	24 V output Control-is-ready signal
1b	O0	24 V outputs, can be switched off via terminal X4.3a (+24 V PLC.03)
2b	O1	
3b	O2	
4b	O3	
5b	O8	24 V outputs, can be switched off via terminal X4.2a (+24 V PLC.02)
6b	O9	
7b	O10	
8b	O11	
9b	O16	24 V outputs, cannot be switched off
10b	O17	
11b	O18	
12b	O19	



Note

Each output of the UxC 11x can be loaded with a maximum current of 150 mA.



Note

If the integrated PLC outputs do not suffice for your application, you can connect up to 7 additional external PL 61xx expansion PLs to the UxC 11x via the HSCI interface.



**X6—Safety:
Single-channel PLC
outputs (at the top)**

Connections at the top of the UxC 11x FS:

- 20 single-channel PLC outputs

Terminal	Signal designation	Assignm. / Function
1a	O4	24 V outputs, can be switched off via terminal X4.3a (+24 V PLC.03)
2a	O5	
3a	O6	
4a	O7	
5a	O12	24 V outputs, can be switched off via terminal X4.2a (+24 V PLC.02)
6a	O13	
7a	O14	
8a	O15	
9a	–TEST.A	24 V output for emergency stop chain
10a	–TEST.B	24 V output for emergency stop chain
11a	–STOS.A.G	24 V output: (spindle safe torque off)
12a	–STO.A.G	24 V output: (safe torque off) Control-is-ready signal
1b	O0	24 V outputs, can be switched off via terminal X4.3a (+24 V PLC.03)
2b	O1	
3b	O2	
4b	O3	
5b	O8	24 V outputs, can be switched off via terminal X4.2a (+24 V PLC.02)
6b	O9	
7b	O10	
8b	O11	
9b	O16	24 V outputs, cannot be switched off
10b	O17	
11b	O18	
12b	O19	



Note

Each output of the UxC 11x can be loaded with a maximum current of 150 mA.



Note

If the integrated PLC outputs do not suffice for your application, you can connect up to 7 additional external PL 61xx expansion PLs to the UxC 11x via the HSCI interface.

**X106—Safety:
Single-/dual-
channel PLC
outputs (on the
front)**

Connections on the front of the UxC 11x FS:

- 8 dual-channel PLC outputs:
O0.A to O7.A
O0.B to O7.B
- 8 single-channel PLC outputs:
O8.A to O15.A

Terminal	Signal designation	Assignm. / Function
1a	O4.B	24 V outputs, can be switched off via terminal X104.2a (+24 V.B)
2a	O5.B	
3a	O6.B	
4a	O7.B	
5a	O4.A	24 V outputs, can be switched off via terminal X104.1a (+24 V.A)
6a	O5.A	
7a	O6.A	
8a	O7.A	
9a	O12.A	24 V outputs, can be switched off via terminal X104.3a (+24 V.C)
10a	O13.A	
11a	O14.A	
12a	O15.A	
1b	O0.B	24 V outputs, can be switched off via terminal X104.2a (+24 V.B)
2b	O1.B	
3b	O2.B	
4b	O3.B	
5b	O0.A	24 V outputs, can be switched off via terminal X104.1a (+24 V.A)
6b	O1.A	
7b	O2.A	
8b	O3.A	
9b	O8.A	24 V outputs, can be switched off via terminal X104.3a (+24 V.C)
10b	O9.A	
11b	O10.A	
12b	O11.A	



Note

Each output of the UxC 11x (FS) can be loaded with a maximum current of 150 mA.



Note

If the integrated PLC outputs do not suffice for your application, you can connect up to 7 additional external PL 61xx expansion PLs to the UxC 111FS via the HSCI interface.



3.17 Analog PLC inputs/outputs



Note

The interfaces of the PLA-H 08-04-04 module are electrically separated from the 230 V line power in accordance with EN 50178.

Sensors and external devices that are connected to the PLA-H 08-04-04 module must be supplied exclusively either with 24 V NC or with 24 V PLC. The inputs and outputs of the module are not galvanically isolated internally. In the HSCI system, the two supply voltages must not be connected with each other, however.

Sensors and external devices that are connected to the PLA-H 08-04-04 module and are supplied with 24 V NC must have double basic insulation.

Specifications:

Analog inputs

Voltage range:	-10 V to +10 V
Input resistance:	> 40 k Ω
Resolution:	10 mV

Analog outputs

Voltage range:	-10 V to +10 V
Load impedance:	> 5 k Ω
Output current:	< 2 μ A
Resolution:	10 mV

Inputs for Pt 100 thermistors

Measurement current:	2.25 mA
Temperature range:	-4 °C to 120 °C (0xF230 to 0x0000)
Resolution:	0.002 °C



Note

If the module has not been correctly configured with the IOconfig software for PCs or if errors occur during measurement (measuring range exceeded, broken cable, loose connection, etc.), the module transmits the value 0.000 V for voltages or 120 °C for temperatures to the control, along with a diagnostic message. To avoid excessive temperature measurement values, these 120 °C values must **not** be taken into account in averaging the measured values. For values below the measuring range, the module transmits the value -4 °C to the control.

Terminals on the PLA-H 08-04-04

Properties	Socket connectors on the PLA-H 08-04-04
Connection:	Socket connector with tension clamp connection, type: X81, X82: Weidmüller B2L 3.5/10 SN SW 2-row, 10-pin X66, X67, X46, X47, X48, X49: Weidmüller B2L 3.5/6 SN SW 2-row, 6-pin
Connectable conductors:	Usable conductor cross sections without wire-end sleeve: 0.08 mm ² to 1.0 mm ² Usable conductor cross sections with wire-end sleeve: 0.14 mm ² 0.34 mm ² 0.5 mm ² (only with Weidmüller PZ 6/5 crimping pliers)



Note

HEIDENHAIN recommends:

Preferably use a conductor cross section of 0.34 mm² if you use stranded wires with wire-end sleeves. This cross section can be clamped appropriately and ensures a reliable terminal connection.

If you use stranded wires with wire-end sleeves and a conductor cross section of 0.5 mm², the Weidmüller PZ 6/5 crimping pliers (setting 0.25–0.5 mm²) must be used for crimping. In this case, orient the crimped wire-end sleeve before inserting it into the socket connector. If crimping pliers from other manufacturers are used for crimping conductors with a cross section of 0.5 mm², the crimped wire-end sleeves cannot be inserted into the socket connector and clamped appropriately, and therefore do not result in a reliable terminal connection.

Power consumption of a PLA-H 08-04-04 I/O module:

- Without load: 2.6 W
- Under full load: 4.5 W

X46 to X49: Analog inputs

Pin layout

Connecting terminals	Assignment
1a/1b	–10 V to +10 V (input)
2a/2b	0 V (reference potential)
3a/3b	Shield



**X66 to X67:
Analog output**

Pin layout

Connecting terminals	Assignment
1a/1b	-10 V to +10 V (output)
2a/2b	0 V (reference potential)
3a/3b	Shield

**X81 to X82:
Connection for
Pt 100**

Pin layout:

Connecting terminals	Assignment
1a/1b	I+ Constant current for PT 100
2a/2b	U+ Measuring input for Pt 100
3a/3b	U- Measuring input for PT 100
4a/4b	I- Constant current for PT 100
5a/5b	Shield

3.18 PROFIBUS connection

Pin layout on X121 of the MC main computer or IPC and on X1 of the PLB 550 PROFIBUS slave

Main computer X121		Connecting cable ID 515845-01			PLB 550 X1	
D-sub connection (female) 9-pin	Assignment	D-sub connectr. (male) 9-pin		D-sub connectr. (male) 9-pin	X1 D-sub connection (female) 9-pin	Assignment
1	Do not assign	1	–	1	1	Do not assign
2	Do not assign	2	–	2	2	Do not assign
3	B line	3	B line	3	3	B line
4	RTS (signal type: TTL)	4	–	4	4	RTS (signal type: TTL)
5	GND	5	–	5	5	GND
6	+5 V	6	–	6	6	+5 V
7	Do not assign	7	–	7	7	Do not assign
8	A line	8	A line	8	8	A line
9	Do not assign	9	–	9	9	Do not assign
Housing	External shield	Housing	External shield	Housing	Housing	External shield

All signals on the PL 550 are electrically isolated.

All signals are electrically isolated at X121 of the MC 6xxx main computer or the IPC 6xxx.

The +5 V and GND pins supply the terminating resistor in the connector.



Note

For more information about PROFIBUS on HEIDENHAIN controls, refer to the documentation about the IOconfig software for PCs.



3.19 PROFINET connection

PROFINET devices are connected to the MC main computer at the connector X621 or X622 on the plug-in interface board (PC-MIP or PCI Express slot).

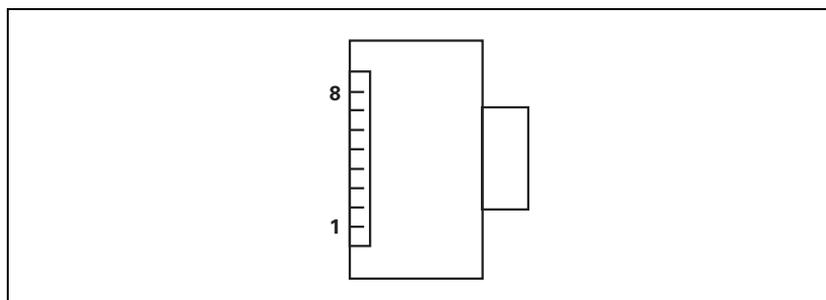
Properties of the PROFINET connection:

- Network topology: Star or linear configuration
- Data transfer rate: 100 mbps

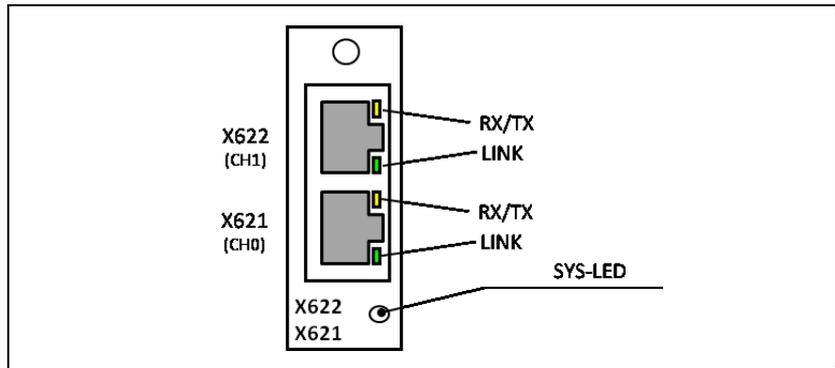
Pin layout of connectors X621 and X622 on the PROFINET interface board:

RJ45 connection (female) 8-pin	Assignment
1	TX+
2	TX-
3	RX+
4	Do not assign
5	Do not assign
6	RX-
7	Do not assign
8	Do not assign
Housing	External shield

Face of the connector:



Meanings of the LEDs on the PROFINET data interface:



LED	Status	Meaning
LINK green	On	Connection established
	Off	No connection
RX/TX yellow	Blinking	Communication active
	Off	No communication
SYS green	On	Controller running
SYS yellow	On	Controller waiting for software
SYS yellow	Off	Controller defective or not correctly connected



Note

For more information about PROFINET on HEIDENHAIN controls, refer to the documentation about the IOconfig software for PCs.

HEIDENHAIN provides device files (HDD files) for all HSCI components with the technical characteristics of these components. These HDD files are of critical importance for the configuration of the HSCI system. You should therefore make sure that you always use the latest version of IOconfig and the HDD files for the configuration.

When configuring a project, the IOconfig program takes the description of the HSCI components from the HDD file and transfers it to the configuration file (IOC file).



Note

HEIDENHAIN recommends updating all IOC files and EAZ projects with the IOconfig version 2.2.02. This is the only way to ensure that you can use all of the features provided by the new HSCI components.

For more information about using IOconfig, please refer to the program's on-line help.

Updating the HSCI configuration

To update the HSCI configurations, proceed as follows:

Install the IOconfig version 2.2.02 or a later IOconfig version. In the HEIDENHAIN FileBase, you will find the current version of the IOconfig software for PCs for downloading. With IOconfig version 2.2.02, however, changes to the HDD files are not transferred automatically.

If new HDD files exist on the PC, IOconfig 2.x does not update these files automatically until these files:

- are copied into the directory in which IOconfig searches for HDD files (to be defined in Extras/Options and Project/Settings), and
- are read in with the HDD import function or imported via an IOC import of GSD/HDD files.

HDD files cannot be imported in any other way, e.g. when EAZ files are loaded. Any existing new HDD files will therefore be ignored.

Do the following steps manually in order to be able to use all available features of new HSCI components:

1. Transfer the HDD files

Check where the standard directory and the project directory for HDD files are located. The two directories should usually be identical:

- Extras -> Options: Standard directory for HDD library: xxxxxxxx
- Project -> Project Settings: Directory of HDD library: xxxxxxxx

Move or delete all HDD files being used.

Then copy all current HDD files from IOconfig (e.g. from C:\Programs\HEIDENHAIN\IOconfig\HDD) to the standard directory or the project directory of the HDD library.



2. Update the EAZ project

The only way to ensure that all HDD files specified in the IOC file are evaluated and used for generating the HSCI components in an IOC project is exporting and importing an IOC file. These steps are required to be able to detect new components, diagnostic information and parameters in new HDD files. The essential steps are creating an IOC file and re-importing this IOC file. These steps ensure that all diagnostic information is updated and all new parameterization possibilities become visible.

If the message **Missing HDD files** is displayed during import, press **CONTINUE** to continue the process. Then the slave is inserted, but it will not have any diagnostic or parametric properties.

This can happen if the names of HDD files were changed and the IOC file still contains the previous file names. A new IOconfig version with current HDD files will usually not find these old files. **If it finds them nevertheless, then move the old HDD files and make sure that you do not use them anymore.**

Something similar can occur if the name of an HDD file is correct, but the name of the slave contained in it was changed and the IOC file still contains the old name of the slave. The error message **Missing HDD files** will then not be displayed, but the diagnostic and parametric properties of the slave will be missing.

In such cases, the slave in question must be deleted and reinserted from the correct HDD file. The new functions provided by IOconfig 2.2.02 for this purpose can be used to transfer the terminal names during the Delete/Paste actions.

3. Check the machine operating panels and keyboard units

If the **Override** module is shown for an MB/TE slave, then the respective slave must be deleted and reinserted from the correct HDD file. The new functions provided by IOconfig 2.2.02 for this purpose can be used to transfer the terminal names during the Delete/Paste actions.

3.21 Buffer battery



Note

Make a data backup before changing the buffer battery.



Danger

When exchanging the buffer battery, remember:

- Switch off the machine and the TNC.
- The buffer battery may be exchanged only by trained personnel.
- When exchanging the buffer battery, ensure electrostatic discharge protection. Improper handling can result in damage to the components or assemblies due to ESD!

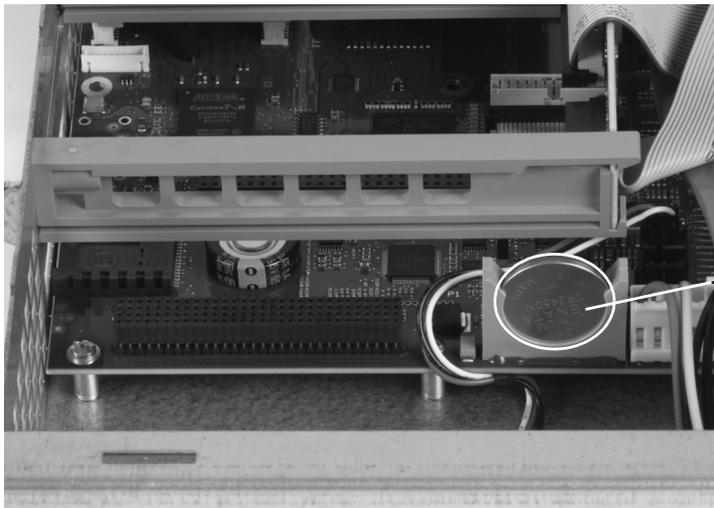
Battery type: 1 lithium battery, type CR 2450N (Renata), ID 315878-01

If the voltage of the buffer battery falls below 2.6 V, the error message **Exchange buffer battery** appears. If the voltage does not rise above 2.6 V again, the error message is reactivated after 30 minutes.

To exchange the battery:

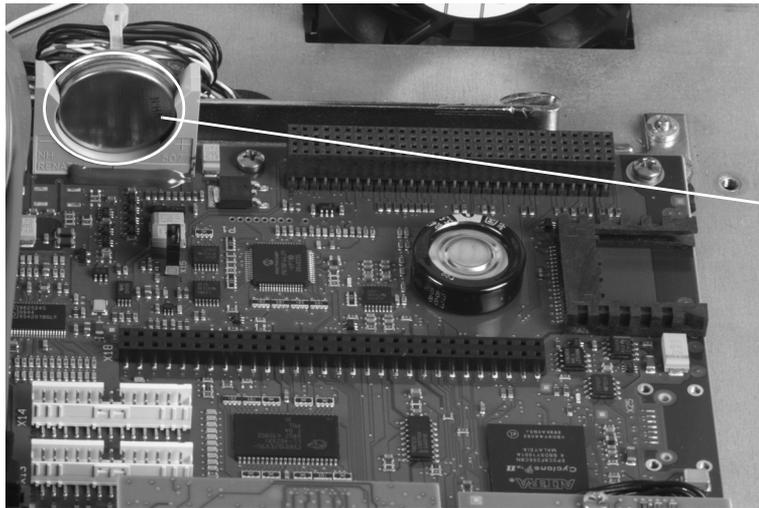
- ▶ The buffer battery is on the motherboard of the MC 6xxx.
- ▶ Loosen the screws securing the housing cover of the MC 6xxx.
- ▶ Remove the cover.
- ▶ The buffer battery is at the border of the PCB:

In the MC 6x41, it is next to the slot for the HDR:



Buffer battery in
MC 6x41

In the MC 6222, MC 7222 it is at the rear close to the fans:



Buffer battery
in the MC 6222

- ▶ Exchange the buffer battery. The new battery can be inserted in only one position.

3.22 Encoder connections

3.22.1 General information

HEIDENHAIN contouring controls are designed for use with incremental or absolute linear and angular encoders as measuring systems. The encoder signals are subdivided 1024-fold.

Encoders with one reference mark or distance-coded reference marks and with EnDat interface are permissible.

HEIDENHAIN recommends the use of absolute encoders with EnDat interface or the use of encoders with distance-coded reference marks because they greatly reduce the traverse distance required to establish the absolute position.



Note

On controller units with HSCI interface (e.g. CC 61xx, UEC 11x) you can no longer directly connect encoders with 11 μ A signals. You need an adapter ID 313119-xx to adjust the signals; see "Adapters for encoder signals" on page 304.

Please use only HEIDENHAIN encoder cables, connectors and couplings. For maximum cable lengths, see "Cable Overview" at the end of this chapter.

	Position encoder	Speed encoder
Signal amplitude, input frequency 1 V _{PP}	EnDat, 1 V _{PP} (adjustable in MP118, MP119)	EnDat, 1 V _{PP}



Note

Keep in mind the line count of the speed encoders when choosing the motors:

$$x = \frac{f \cdot 60 \cdot 1000}{n}$$

x: line count of the speed encoder

f: maximum input frequency

n: maximum speed

Example:

f = 350 kHz; n = 10 000 min⁻¹

$$x = \frac{350 \cdot 60 \cdot 1000}{10000} \approx 2048$$



3.22.2 Position encoder input

X201 to X206:
Position encoder
1 V_{PP}

Pin layout:

CC 6106		Adapter cable 309783-xx Adapter cable 310199-xx			Encoder	
Male	Assignment	Female	Color	Female	Male	Color
1	+5 V (U _P)	1	Brown/Green	12	12	Brown/Green
2	0 V (U _N)	2	White/Green	10	10	White/Green
3	A+	3	Brown	5	5	Brown
4	A-	4	Green	6	6	Green
5	Do not assign	5				
6	B+	6	Gray	8	8	Gray
7	B-	7	Pink	1	1	Pink
8	Do not assign	8				
9	+5 V (sensor)	9	Blue	2	2	Blue
10	R+	10	Red	3	3	Red
11	0 V (sensor)	11	White	11	11	White
12	R-	12	Black	4	4	Black
13	0 V	13				
14	Do not assign	14	Violet	7	7	Violet
15	Do not assign	15				
Hsg.	External shield	Hsg.	External shield	Hsg.	Hsg.	External shield



Note

The interface complies with the requirements of EN 50178 for "low voltage electrical separation."

X201 to X206:
Position encoder
with EnDat interface

Pin layout:

CC 6106		Adapter cable 332115-xx			Connecting cable 323897-xx			Adapter cable 313791-xx		
Male	Assign.	Female	Color	Female	Male	Color	Fem.	Male	Color	Fem.
1	+5 V (U _P)	1	Brown/ Green	7	7	Brown/ Green	7	7	Brown/ Green	5b
2	0 V (U _N)	2	White/ Green	10	10	White/ Green	10	10	White/ Green	6a
3	A+	3	Green/ Black	15	15	Green/ Black	15	15	Green/ Black	2a
4	A-	4	Yellow/ Black	16	16	Yellow/ Black	16	16	Yellow/ Black	2b
5	Data	5	Gray	14	14	Gray	14	14	Gray	3b
6	B+	6	Blue/ Black	12	12	Blue/ Black	12	12	Blue/ Black	1a
7	B-	7	Red/ Black	13	13	Red/ Black	13	13	Red/ Black	1b
8	$\overline{\text{Data}}$	8	Pink	17	17	Pink	17	17	Pink	3a
9	+5 V (sensor)	9	Blue	1	1	Blue	1	1	Blue	5a
10	Vacant	10		3	3	Red	3	3		
11	0 V (sensor)	11	White	4	4	White	4	4	White	6b
12	Vacant	12		2	2	Black	2	2		
13	Internal shield	13	Internal shield	11	11	Internal shield	11	11	Internal shield	
14	Clock	14	Violet	8	8	Violet	8	8	Violet	4a
15	$\overline{\text{Clock}}$	15	Yellow	9	9	Yellow	9	9	Yellow	4b
Hsg.	Housing	Hsg.	External shield	Hsg.		External shield		Hsg.	External shield	

Line drop compensator ID 336 697-02, if required



Note

The interface complies with the requirements of EN 50178 for "low voltage electrical separation."



Note

For cable lengths > 10 m between the CC 61xx or UxC 11x and the encoders with EnDat interfaces (EnDat 2.1), a line-drop compensator (336697-xx) is required (efficiency = 75 %).



**X201 to X206:
Position encoder
with EnDat 2.2
interface**

Pin layout:

CC 6106		Adapter cable 673459-xx		
Male	Assignment	Female	Color	Female
1	+5 V (U _P)	1	Brown/ Green	11
2	0 V (U _N)	2	White/ Green	12
3	A+	3	Vacant	
4	A-	4	Vacant	
5	Data	5	Gray	2
6	B+	6	Vacant	
7	B-	7	Vacant	
8	$\overline{\text{Data}}$	8	Pink	1
9	+5 V (sensor)	9	Blue	14
10	Vacant	10	Vacant	
11	0 V (sensor)	11	White	13
12	Vacant	12	Vacant	
13	Internal shield	13	Vacant	
14	Clock	14	Violet	10
15	$\overline{\text{Clock}}$	15	Yellow	9
Hsg.	Housing	Hsg.	External shield	Hsg.



Note

The interface complies with the requirements of EN 50178 for "low voltage electrical separation."

3.22.3 Speed encoder input



Attention

If you connect angle or linear encoders from HEIDENHAIN to the speed encoders (such as for torque motors), you must pay attention to the different connector layouts!

HEIDENHAIN offers special cables and line-drop compensators for such applications. More information is in the Cable overviews.

X15 to X20: Shaft speed encoder 1 V_{pp}

Pin layout:

CC 61xx		Adapter cable 289440-xx				Connecting cable 336847-xx		
Male	Assignment	Female	Color	Female		Male	Color	Female
1	+5 V (U _P)	1	Brown/Green	10	Line drop compensator ID 370 226-01, if required	10	Brown/Green	10
2	0 V (U _N)	2	White/Green	7		7	White/Green	7
3	A+	3	Green/Black	1		1	Green/Black	1
4	A-	4	Yellow/Black	2		2	Yellow/Black	2
5	0 V							
6	B+	6	Blue/Black	11		11	Blue/Black	11
7	B-	7	Red/Black	12		12	Red/Black	12
8	0 V	8	Internal shield	17		17	Internal shield	17
9	Do not assign							
10	Do not assign							
11	Do not assign							
12	Do not assign							
13	Temperature+	13	Yellow	8		8	Yellow	8
14	+5 V (sensor)	14	Blue	16		16	Blue	16
15	Do not assign							
16	0 V (sensor)	16	White	15		15	White	15
17	R+	17	Red	3		3	Red	3
18	R-	18	Black	13		13	Black	13
19	C+	19	Green	5		5	Green	5
20	C-	20	Brown	6		6	Brown	6
21	D+	21	Gray	14		14	Gray	14
22	D-	22	Pink	4		4	Pink	4
23	Do not assign							
24	0 V							
25	Temperature -	25	Violet	9		9	Violet	9
Hsg.	Housing	Hsg.	External shield	Hsg.	Hsg.	External shield	Hsg.	



Note

The interface complies with the requirements of EN 61800-5-1 for "protective extra-low voltage" (PELV).



X15 to X20: Speed encoder with EnDat interface

Pin layout:

CC 61xx		Adapter cable 336376-xx				Connecting cable 340302-xx		
Male	Assignment	Female	Color	Female		Male	Color	Female
1	+5 V (U _P)	1	Brown/Green	10	Line drop compensator ID 370 224-01, if required	10	Brown/Green	10
2	0 V (U _N)	2	White/Green	7		7	White/Green	7
3	A+	3	Green/Black	1		1	Green/Black	1
4	A-	4	Yellow/Black	2		2	Yellow/Black	2
5	0 V							
6	B+	6	Blue/Black	11		11	Blue/Black	11
7	B-	7	Red/Black	12		12	Red/Black	12
8	0 V	8	Internal shield	17		17	Internal shield	17
9	Do not assign							
10	Clock	10	Green	5		5	Green	5
11	Do not assign							
12	$\overline{\text{Clock}}$	12	Brown	14		14	Brown	14
13	Temperature +	13	Yellow	8		8	Yellow	8
14	+5 V (sensor)	14	Blue	16		16	Blue	16
15	Data	15	Red	3		3	Red	3
16	0 V (sensor)	16	White	15		15	White	15
17	Do not assign							
18	Do not assign							
19	Do not assign							
20	Do not assign							
21	Do not assign							
22	Do not assign							
23	$\overline{\text{Data}}$	23	Black	13		13	Black	13
24	0 V							
25	Temperature -	25	Violet	9		9	Violet	9
Hsg.	Housing	Hsg.	External shield	Hsg.	Hsg.	External shield	Hsg.	



Note

The interface complies with the requirements of EN 61800-5-1 for "protective extra-low voltage" (PELV).



Danger

Only units that comply with the requirements of EN 61800-5-1 for "protective extra-low voltage (PELV)" may be connected.

Pin layout (for the LC or RCN):

CC 61xx		Adapter cable 336376-xx				Adapter cable 369124-xx Adapter cable 369129-xx	
Male	Assignment	Female	Color	Female		Male	Color
1	+5 V (U _P)	1	Brown/Green	10	Line drop compensator ID 368 210-02, if required	7	Brown/Green
2	0 V (U _N)	2	White/Green	7		10	White/Green
3	A+	3	Green/Black	1		15	Green/Black
4	A-	4	Yellow/Black	2		16	Yellow/Black
5	0 V						
6	B+	6	Blue/Black	11		12	Blue/Black
7	B-	7	Red/Black	12		13	Red/Black
8	0 V	8	Internal shield	17		11	Internal shield
9	Do not assign						
10	Clock	10	Green	5		8	Violet
11	Do not assign						
12	Clock	12	Brown	14		9	Yellow
13	Temperature +	13	Yellow	8			
14	+5 V (sensor)	14	Blue	16		1	Blue
15	Data	15	Red	3		14	Gray
16	0 V (sensor)	16	White	15		4	White
17	Do not assign						
18	Do not assign						
19	Do not assign						
20	Do not assign						
21	Do not assign						
22	Do not assign						
23	Data	23	Black	13		17	Pink
24	0 V						
25	Temperature -	25	Violet	9			
Hsg.	Housing	Hsg.	External shield	Hsg.	Hsg.	External shield	
					1		
					2	temperature +	
					3	temperature -	
					4		



Note

The interface complies with the requirements of EN 61800-5-1 for "protective extra-low voltage" (PELV).



Pin layout (for the LC or RCN):

CC 61xx		Adapter cable 509667-xx			Adapter cable 369124-xx Adapter cable 369129-xx or RCN	
Male	Assignment	Female	Color	Female	Male	Color
1	+5 V (U _P)	1	Brown/ Green	7	7	Brown/ Green
2	0 V (U _N)	2	White/ Green	10	10	White/ Green
3	A+	3	Green/ Black	15	15	Green/Black
4	A-	4	Yellow/ Black	16	16	Yellow/ Black
5	0 V					
6	B+	6	Blue/ Black	12	12	Blue/Black
7	B-	7	Red/Black	13	13	Red/Black
8	0 V	8	Internal shield	11	11	Internal shield
9	Do not assign					
10	Clock	10	Green	8	8	Violet
11	Do not assign					
12	Clock	12	Brown	9	9	Yellow
13	Temperature +	13	Yellow	5		
14	+5 V (sensor)	14	Blue	1	1	Blue
15	Data	15	Red	14	14	Gray
16	0 V (sensor)	16	White	4	4	White
17	Do not assign					
18	Do not assign					
19	Do not assign					
20	Do not assign					
21	Do not assign					
22	Do not assign					
23	Data	23	Black	17	17	Pink
24	0 V					
25	Temperature -	25	Violet	6		
Hsg.	Housing	Hsg.	External shield	Hsg.	Hsg.	External shield

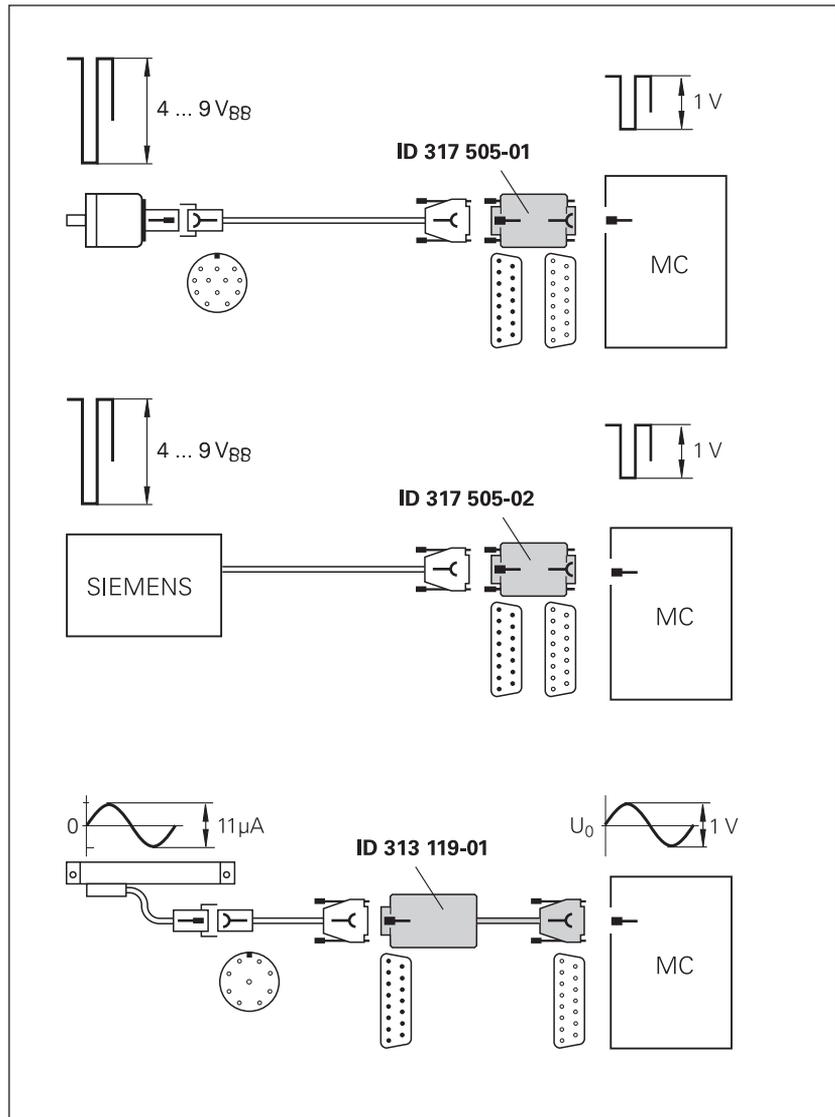


Note

The interface complies with the requirements of EN 61800-5-1 for "protective extra-low voltage" (PELV).

3.23 Adapters for encoder signals

Encoder signals with $11 \mu\text{A}_{\text{PP}}$ or TTL levels can be adapted to the 1 V_{PP} interface with HEIDENHAIN adapter connectors.



Note

Please note:

- The adapters adjust only the levels, not the signal shape.
- The contamination signal of the square-wave encoder cannot be evaluated.
- A square-wave signal can be subdivided no more than 4-fold.

**Adapter connector
TTL (HEIDENHAIN)/
1 V_{PP}**

Pin layout of D-sub connector (female) and D-sub connector (male):

D-sub connection (female) 15-pin	Assignment	D-sub connection (male) 15-pin	Assignment
1	+5 V (U _P)	1	+5 V (U _P)
2	0 V (U _N)	2	0 V (U _N)
3	A+	3	U _{a1}
4	A-	4	-U _{a1}
5	Not assigned	5	Not assigned
6	B+	6	U _{a2}
7	B-	7	-U _{a2}
8	Not assigned	8	Not assigned
9	+5 V	9	+5 V
10	R+	10	U _{a0}
11	0 V	11	0 V
12	R-	12	-U _{a0}
13	Not assigned	13	Not assigned
14	Do not assign^a	14	Do not assign^a
15	Not assigned	15	Not assigned

- a. The control assigns the EnDat clock to pin 14. Therefore, you must not assign any signals to this pin when using the TTL adapter connector.

**Adapter connector
TTL (SIEMENS)/1 V_{PP}**

Pin layout of D-sub connector (female) and D-sub connector (male):

D-sub connection (female) 15-pin	Assignment	D-sub connection (male) 15-pin	Assignment
1	Not assigned	1	Not assigned
2	0 V	2	0 V
3	A+	3	U _{a1}
4	A-	4	-U _{a1}
5	Not assigned	5	Not assigned
6	B+	6	U _{a2}
7	B-	7	-U _{a2}
8	Not assigned	8	Not assigned
9	Not assigned	9	Not assigned
10	R+	10	Not assigned
11	Not assigned	11	Not assigned
12	R-	12	U _{a0}
13	Not assigned	13	-U _{a0}
14	Not assigned	14	Not assigned
15	Not assigned	15	Not assigned

Adapter connector
11 μA_{PP} / 1 V_{PP}

Pin layout of D-sub connector (female) and D-sub connector (male):

D-sub connection (female) 15-pin	Assignment	D-sub connection (male) 15-pin	Assignment
1	+5 V (U_P)	1	+5 V (U_P)
2	0 V (U_N)	2	0 V (U_N)
3	A+	3	0°+
4	A-	4	0°-
5	0 V	5	0 V
6	B+	6	90°+
7	B-	7	90°-
8	0 V	8	0 V
9	+5 V	9	+5 V
10	R+	10	R+
11	0 V	11	0 V
12	R-	12	R-
13	0 V	13	0 V
14	Not assigned	14	Not assigned
15	Not assigned	15	Not assigned



3.24 Connecting the motor power modules (only CC 61xx)

The iTNC 530 HSCI is connected with HEIDENHAIN or non-HEIDENHAIN inverters through a PWM interface.

For a description of the HEIDENHAIN inverter systems, refer to the Technical Manual "Inverter Systems and Motors." The components required for operation of the iTNC 530 HSCI with non-HEIDENHAIN inverter systems are described in the manual "Technical Information for the Operation of SIMODRIVE and POWER DRIVE Inverter Systems." The individual PWM outputs of the CC 61xx are assigned to different controller groups.

For more information on the controller groups of the CC 61xx, see page 1067.

The following applies to the output signals to the power module:

Logic level: 5 V
 Analog signals I_{ACTL}: ±7.5 V
 PWM frequency: Can be set at
 3333 Hz, 4166 Hz, 5000 Hz, 6666 Hz,
 8333 Hz and 10000 Hz

X51 to X56: PWM output

Pin layout:

Ribbon connector, 20-pin	Assignment
1a	PWM U1
1b	0 V U1
2a	PWM U2
2b	0 V U2
3a	PWM U3
3b	0 V U3
4a	SH2
4b	0 V (SH2)
5a	SH1
5b	0 V (SH1)
6a	+IACTL 1
6b	-IACTL 1
7a	0 V (analog)
7b	+IACTL 2
8a	-IACTL 2
8b	0 V (analog)
9a	Do not assign
9b	BRK
10a	ERR
10b	RDY



Note

The interface complies with the requirements of EN 61800-5-1 for "protective extra-low voltage" (PELV).

3.25 Analog nominal value output

In the HSCI system, analog nominal-value outputs are available via the CMA-H 04-04-00, e.g. for controlling spindles and auxiliary axes.

The CMA-H 04-04-00 is an optional SPI expansion module (ID 688721-xx). It adds four analog nominal-value outputs to the CC 61xx controller unit or the UEC 1xx.

Controller unit	Number of CMA-H 04-04-00 modules per unit	Max. number of nominal-value outputs
CC 61xx	2	8
UEC 11x	1	4

The CMA-H 04-04-00 is supported by the following CC 61xx and UEC 11x:

- CC 6106 as of ID 662636-01 (as of index A, 29.06.2010)
- CC 6108 as of ID 662637-01 (as of Index A, 29.06.2010)
- CC 6110 as of ID 662638-01 (as of Index A, 29.06.2010)
- UEC 111 as of ID 625777-02 (as of Index A, 03.02.2010)
- UEC 112 as of ID 625779-02 (as of Index A, 03.02.2010)



Note

- The analog nominal value outputs can only be accessed via the NC, and not via the PLC. The PL 6xxx provides PLC analog outputs, see "Analog PLC inputs/outputs" on page 285.
- With software version 60642x-02 and higher, the CMA-H can generally also be used to control axes that are interpolated together with other digital axes. However, please discuss such applications with your contact person at HEIDENHAIN beforehand.

Properties	Socket connectors
Output:	±10 V
Smallest voltage step:	0.328 mV
Maximum load capacity of the outputs:	10 mA
Maximum capacity:	3 nF
Connection:	4 socket connectors with tension spring connection, type: Weidmüller B2L 3.5/6 SN SW 2-row, 6-pin
Connectable conductors:	Usable conductor cross sections without wire-end sleeve: 0.08 mm ² to 1.0 mm ² Usable conductor cross sections with wire-end sleeve: 0.14 mm ² 0.34 mm ² 0.5 mm ² (only with Weidmüller PZ 6/5 crimping pliers)



Note

HEIDENHAIN recommends:

Preferably use a conductor cross section of 0.34 mm² if you use stranded wires with wire-end sleeves. This cross section can be clamped appropriately and ensures a reliable terminal connection.

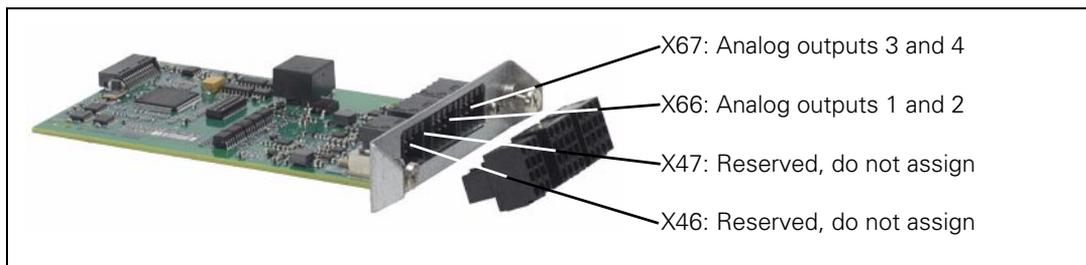
If you use stranded wires with wire-end sleeves and a conductor cross section of 0.5 mm², the Weidmüller PZ 6/5 crimping pliers (setting 0.25–0.5 mm²) must be used for crimping. In this case, orient the crimped wire-end sleeve before inserting it into the socket connector. If crimping pliers from other manufacturers are used for crimping conductors with a cross section of 0.5 mm², the crimped wire-end sleeves cannot be inserted into the socket connector and clamped appropriately, and therefore do not result in a reliable terminal connection.

Each CMA-H 04-04-00 has four analog outputs, see "CMA-H 04-04-00—Pin layout" on page 310.

Behavior of CMA-H 04-04-00 in case of error:

An error is deemed to be an interruption of the HSCI transmission, insufficient supply voltage, overtemperature, overcurrent or short circuit of a nominal-value output. If the module identifies an error, all outputs for 1 ms are actively switched to 0 V. Then all outputs are switched to 0 V (high impedance) and remain in this state until a reset.

3.2.5.1 CMA-H 04-04-00—Pin layout



CMA-H Analog outputs

X66: Analog outputs 1 and 2			
Function	Analog output 1		
Connecting terminal	1a	2a	3a
Assignment	±10 V	0 V	Shield
Function	Analog output 2		
Connecting terminal	1b	2b	3b
Assignment	±10 V	0 V	Shield

X67: Analog outputs 3 and 4			
Function	Analog output 3		
Connecting terminal	1a	2a	3a
Assignment	±10 V	0 V	Shield
Function	Analog output 4		
Connecting terminal	1b	2b	3b
Assignment	±10 V	0 V	Shield



Note

The terminals X46 and X47 on the CMA-H 04-04-00 are reserved for future functions and are not yet supported by the NC software. Please do not assign them.



Note

Please note:

- Connect the shield of the connecting cable leading to the nominal-value output both to pin 3 a/b on the CMA-H and to the ground potential of the housing of the CC 61xx or UEC 11x.
HEIDENHAIN recommends using EMC shielding terminals. The max. distance between the CMA-H and the ground point is 500 mm.
- On the analog servo amplifier, you also connect the shield of the connecting cable to the ground potential of the housing via an EMC shielding terminal.
- Use only shielded twisted-pair connecting cables.
- The connecting cables to the nominal-value outputs must not have more than one intermediate terminal.

Configuration with MP120 and MP121

Use MP120.x to assign analog auxiliary axes to the nominal-value outputs on the CMA-H 04-04-00, and MP121.x to assign analog spindles to the nominal-value outputs on the CMA-H 04-04-00.

One CC 61xx has slots for up to two CMA-H 04-04-00 modules. MP12x.x = 1...4 is used to address the first CMA-H, and MP12x.x = 5...8 is used to address the second CMA-H (see table below):

Setting in MP120.x or MP121.x	Active analog output on the CMA-H
1	X66: Analog output 1 (SPI slot 1 on CC 61xx or UEC 11x)
2	X66: Analog output 2 (SPI slot 1 on CC 61xx or UEC 11x)
3	X67: Analog output 3 (SPI slot 1 on CC 61xx or UEC 11x)
4	X67: Analog output 4 (SPI slot 1 on CC 61xx or UEC 11x)
5	X66: Analog output 1 (SPI slot 2 on CC 61xx)
6	X66: Analog output 2 (SPI slot 2 on CC 61xx)
7	X67: Analog output 3 (SPI slot 2 on CC 61xx)
8	X67: Analog output 4 (SPI slot 2 on CC 61xx)

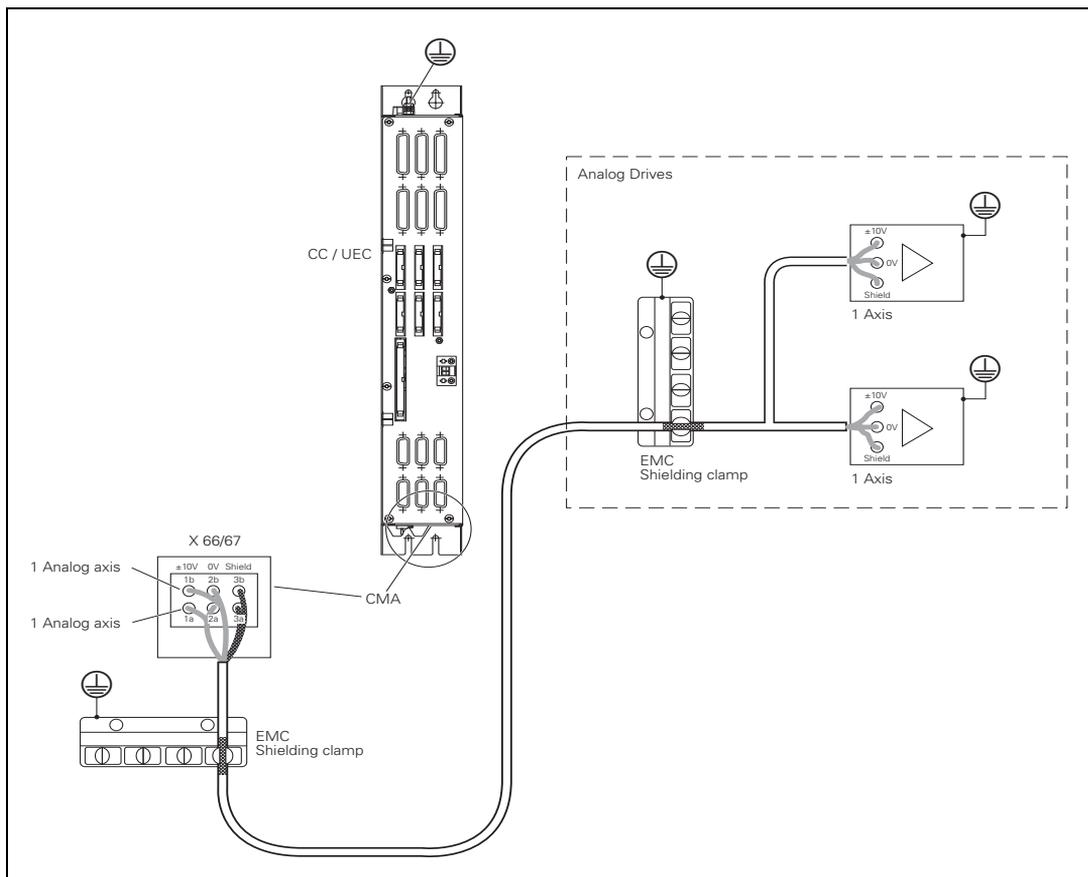
During rigid tapping with digital axes and analog spindle using the CMA-H, the following must be considered:

- MP7160 bit#4 = 0 (tool axis follows the spindle) is supported
- MP7160 Bit#4 = 1 (tool axis and spindle interpolated) is not supported



Overview of wiring

CMA-H 04-04-00—Schematic wiring diagram



3.26 Touch probes

The following touch probes can be connected to the iTNC 530 HSCI:

- TS 220, touch-trigger probe with cable connection for workpiece setup and measurement during machining
- TS 440, TS 444, TS 640, TS 642, TS 740, touch-trigger probes with infrared transmission for workpiece setup and measurement during machining
- TT 130, TT 140, TT 449 touch probes for tool measurement



Note

The touch probes are connected to the PLB 620x PLC system module or UEC 11x at X112 (TS) and X113 (TT).



Note

For the PLB 62xx up to and including variant -02, please note:
Variant -02:

To connect a TT touch probe to the PLB 62xx, you have to connect the touch probe adapter, ID 667674-01 to X113, see "Adapter for connection of a TT touch probe to X113" on page 316. The adapter is supplied with the PL.

For suitable connecting cables, see "Cable Overview" at the end of the chapter.

X112/X113: Triggering touch probe



Note

The touch probes are connected to the PLB 620x PLC system module or UEC 11x at X112 (TS) and X113 (TT).



Note

For the PLB 62xx up to variant -02, please note:

To connect a TT touch probe to the iTNC 530 HSCI, you have to connect the touch probe adapter, ID 667674-01, to X113.

X112/X113 pin layout on PLB 62xx as of variant-03 and UEC 11x (FS):

(15-pin D-sub, triple-row)



Note

The interface complies with the requirements of EN 60204-1:2006 for protective extra-low voltage (PELV).

Female	Assignment of X112 (TS)	Assignment of X113 (TT)
1	Trigger signal	Trigger signal
2	Trigger signal ^a	Trigger signal ^a
3	TS ready	Do not assign
4	Battery warning	Do not assign
5	+ 5 V NC (+/- 5%)	+ 5 V NC (+/- 5%)
6	TS start	Do not assign
7	TT start	TT start
8	0 V NC	0 V NC
9	0 V NC	0 V NC
10	+ 24 V NC	+ 24 V NC
11	TT ready ^b	TT ready
12	Do not assign	Do not assign
13	Do not assign	Do not assign
14	Do not assign	Do not assign
15	Do not assign	Do not assign

a. Stylus at rest means logic level HIGH.

b. Only if SE 642 is configured at X112 via machine parameter

Load capacity of output voltages of X112, X113:

- Socket 10, +24 V NC: max. 0.8 A
- Socket 5, +5 V NC: max. 0.8 A

The currents are total currents for X112 and X113 together; they are not the output currents per connector.



Danger

Please note that the outputs of connectors X112 and X113 are supplied internally with 24 V NC and thus supply NC voltages derived from it at the outputs.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.

Readiness of the TS at socket 3 is detected at a voltage of +8 V NC to +24 V NC.



Wire colors of adapter cable ID 633608-xx for X112/X113 to TS or TT:

X112/X113 on PLB 62xx or UEC 11x (FS) Female (D-sub)	Adapter cable 633608-xx		
	Male (D-sub)	Color	Female (M12)
1	1	Not assigned	
2	2	Pink	4
3	3	Green	5
4	4	Gray	6
5	5	Not assigned	
6	6	Blue	3
7	7	White	7
8	8	Not assigned	
9	9	White/Green	1
10	10	Brown/Green	2
11	11	Brown	8
12 to 15	12 to 15	Not assigned	



Adapter for connection of a TT touch probe to X113

Only up to variant -02 of PLB 62xx:

The items supplied with the PLB 62xx include a cable adapter with the ID 667674-01.

If you want to connect a TT touch probe to X113 of the PLB 62xx, you need the cable adapter. It makes the Start and Ready signals of the TT touch probes available on the correct pins of X113.

Pin layout of adapter with ID 667674-01:

(15-pin D-sub, triple-row)

Female	Assignment of X113 (TT) up to variant -02	Adapter ID 667674-01	TT adapter cable ID 633616-xx
1	Trigger signal	_____	Trigger signal
2	Trigger signal	_____	Trigger signal
3	TS ready	_____	Do not assign
4	Battery warning	_____	Battery warning
5	+ 5 V NC (+/- 5%)	_____	+ 5 V NC (+/- 5%)
6	Start	_____	Do not assign
7	0 V NC	_____	TT start
8	0 V NC	_____	0 V NC
9	0 V NC	_____	0 V NC
10	+ 24 V NC	_____	+ 24 V NC
11	TT ready	_____	TT ready
12	Do not assign	_____	Do not assign
13	Do not assign	_____	Do not assign
14	Do not assign	_____	Do not assign
15	Do not assign	_____	Do not assign



3.27 MB 620 machine operating panel

For machines with up to six axes, HEIDENHAIN offers the MB 620 machine operating panel (15-inch) with HSCI interface.

On the underside of the machine operating panel are terminal strips bearing the PLC inputs as well as the PLC outputs. Also, connection X23 (347) for HR handwheels is on the underside of the MB 620.

It must be ensured that negative pressure cannot develop in the panel in which the machine operating panel is installed. Negative pressure in the panel can lead to increased contamination of the machine operating panel, which must be avoided by enabling pressure compensation, for example via the cable duct between the electrical cabinet and the panel.

Pin layout for MB 620 (FS)	Connector	Function	Page
	X17	Emergency stop (MB)	318
	X18	Emergency stop (MB)	318
	X500	HSCI output	208
	X502	HSCI input	208
	X6	PLC inputs	319
	X7	PLC outputs	318
	X10	Interface to keyboard and potentiometers	322
	X23	Handwheel connection	347
	X30	Connection for handwheel adapter	323
	X31	Permissive button, NC start, NC stop	321
	X101	24 V NC power supply	323
	⊕	Protective ground	

Besides the snap-on keys, there are also keys with switch contacts that can be wired by the machine tool builder depending on the application. Specifications of the switch contacts on keys such as EMERGENCY STOP, NC start, NC stop, Control voltage on:

- Rated voltage: +24 V DC
- Maximum current: 1.2 A

X17/X18: EMERGENCY STOP on MB

Connectors X17 and X18 are electrically parallel.

With the MB 620 without FS, the EMERGENCY STOP must be wired externally in the EMERGENCY STOP chain as before. In the MB 6xx without FS, the connectors X17 and X18 do not support dual-channel evaluation of the EMERGENCY STOP button. These inputs must therefore not be used for evaluating EMERGENCY STOP!

With the MB 620 FS, the EMERGENCY STOP is evaluated by the control via HSCI and has HEIDENHAIN standard wiring.

Pin layout X18 on MB 620 FS (X17 without function):

Connecting terminals	Assignment
1	Power supply for emergency stop channel A (–TEST.A)
2	Emergency stop channel A
3	Power supply for emergency stop channel B (–TEST.B)
4	Emergency stop channel B

Emergency stop buttons are to be used exclusively for emergency stop purposes. Under normal operating conditions, a machine must not be switched off via the emergency stop buttons. The proper functioning of all emergency stop buttons is to be tested annually by pressing these buttons.

X7: PLC outputs

Pin layout:

Connecting terminals	Assignment
1	O0 (illumination for the NC start key) ^a
2	O1 (illumination for the NC Stop key) ^a
3	O2 (illumination for the Control Voltage ON key) ^a
4	O3
5	O4
6	O5
7	O6
8	O7
9	+24 V NC (available here)
10	0 V NC (available here)

a. With standard wiring

In the event of an overload or short circuit and as a consequence of excessive temperature, the individual PLC outputs switch off automatically. Then the outputs will try to switch on again cyclically.

Current carrying capacity of the outputs: Maximum 150 mA per output



Danger

Please note that the outputs of connector X7 are powered internally by +24 V NC, and therefore supply +24 V NC at HIGH level.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.

Each of the switching outputs at X7 supplies up to 150 mA of output current and are provided for driving the lamps on the MP 620.



Note

Immediately after the supply voltage has been switched on, all switching outputs at X7 switch to HIGH level for approx. 100 ms, which is indicated by short flashing of the connected lamps. Keep this characteristic in mind if you connect inductive loads or other consumers to the MB 620.

HEIDENHAIN recommends:

Connect inductive loads and other consumers to the PLC outputs of the PL 6xxx or UxC 11x. The described effect during switch-on does not occur there. Use the outputs at X7 only for the control of lamps.

X6: PLC inputs

Pin layout of MB 620 without FS:

Connecting terminals	Assignment
1	I0
2	I1
3	I2 (Control Voltage ON, CVO) ^a
4	I3
5	I4
6	I5
7	I6
8	I7
9	Reserved (do not use)
10	Reserved (do not use)

a. With standard wiring

Pin layout of MB 620 FS:

Connecting terminals	Assignment
1	I0.A
2	I1.A
3	I2.A (Control Voltage ON, CVO.A) ^a
4	I3.A
5	I0.B
6	I1.B
7	I2.B (Control Voltage ON, CVO.B) ^a
8	I3.B
9	Power supply for channel A (-TEST.A)
10	Power supply for channel B (-TEST.B)

a. With standard wiring



Danger

Please note that the MB 620 is powered by +24 V NC.

For the entire HSCI system, the +24 V NC power supply voltage is required to be safely separated voltage. It must also be safely separated from the +24 V PLC!



**X31: MB 620,
permissive button /
NC start / NC stop**

Standard wiring of connection X31 for MB 620 without FS:

Connecting terminals	Assignment
1	Reserved (do not use)
2	Reserved (do not use)
3	Reserved (do not use)
4	Reserved (do not use)
5	NC Start ^a
6	Reserved (do not use)
7	NC Start power supply (+24 V NC) ^a
8	NC stop ^a
9	Reserved (do not use)
10	NC Stop power supply (+24 V NC) ^a

a. With standard wiring

Pin layout of MB 620 FS:

Connecting terminals	Assignment
1	Optional: Permissive button channel A / terminal 1 (+24 V)
2	Optional: Permissive button channel A / terminal 2
3	Optional: Permissive button channel B / terminal 1 (+24 V)
4	Optional: Permissive button channel B / terminal 2
5	NC Start channel A / terminal 1 ^a
6	NC Start channel B / terminal 2 ^a
7	NC Start shared terminal (+24 V NC) ^a
8	NC Stop channel A / terminal 1 ^a
9	NC Stop channel B / terminal 2 ^a
10	NC Stop shared terminal (+24 V NC) ^a

a. With standard wiring

NC start and NC stop are normally-open contacts on the MB 620 (FS).



Danger

Please note that the MB 620 is powered by +24 V NC.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.

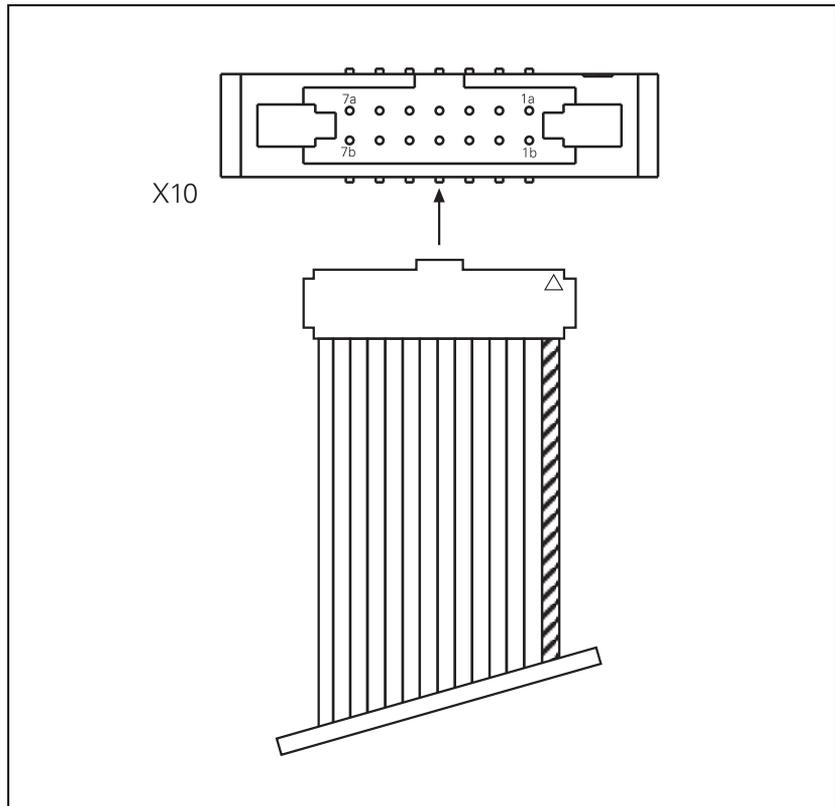


Attention

The interface is electrically separated (protective extra-low voltage (PELV) as per EN 61800-5-1).

X10: Interface to keyboard and potentiometers

Designation of the terminals:



Pin layout of the ribbon cable (included, length 54 cm):

Connecting terminals	Assignment
1a	Potentiometer 1
2a	Potentiometer 3
3a	Do not assign
4a	Do not assign
5a	Do not assign
6a	+ 5 V
7a	0 V
1b	Potentiometer 2
2b	---
3b	Do not assign
4b	Do not assign
5b	---
6b	+ 5 V
7b	0 V

X101: Power supply

Pin layout:

Connecting terminals	Assignment
+ / terminal 1	+24 V NC
- / terminal 2	0 V NC

Power consumption of the operating panel units without HR handwheel and controlled inputs/outputs:

Power consumption of the MB 620: 4.0 W

Power consumption of the PLB 6001: 5.0 W

X30: Handwheel connection, permissive button / emergency stop

With the MB 620 without functional safety (FS), the permissive buttons and the EMERGENCY STOP of the handwheel must be wired externally in corresponding safety circuits as before.

Pin layout of MB 620 FS:

Connecting terminals	Assignment
1	Permissive button channel A / terminal 1 (+24 V NC)
2	Permissive button channel A / terminal 2
3	Permissive button channel B / terminal 1 (+24 V NC)
4	Permissive button channel B / terminal 2
5	Emergency stop channel A / supply for channel A (-TEST.A)
6	Emergency stop channel A / terminal 2
7	Emergency stop channel B / supply for channel B (-TEST.B)
8	Emergency stop channel B / terminal 2

The emergency stop inputs (pin 6 and pin 8) are active even if the HR handwheel has been deactivated via machine parameter MP7640. It is therefore always necessary to wire the emergency stop inputs correctly.

Emergency stop buttons are to be used exclusively for emergency stop purposes. Under normal operating conditions, a machine must not be switched off via the emergency stop buttons. The proper functioning of all emergency stop buttons is to be tested annually by pressing these buttons.



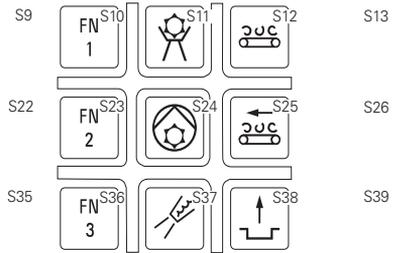
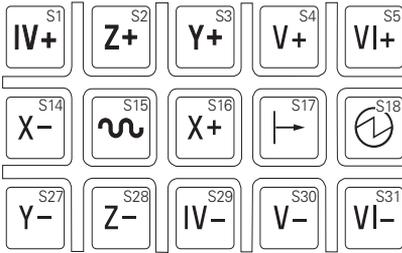
Danger

Please note that the MB 620 is powered by +24 V NC.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.

Connection X30 not used on MB 620 without FS.

Key assignment for machine operating panel



3.28 Data interfaces

X26, X116: Ethernet interface RJ45-port

- Maximum cable length:
 - Unshielded: 100 m
 - Shielded: 400 m
- Network topology: Star configuration

A hub serves as a central node that establishes the connection to the other participants.



Danger

The Ethernet interfaces of the MC 6xxx or MC 7xxx comply with the requirements of PELV ("low voltage electrical separation") according to EN 61800-5-1 and are powered internally by 24 V NC. All devices connected to these Ethernet interfaces must comply with the requirements of SELV or PELV according to EN 61800-5-1.

Maximum data transfer rate:

- For integration into the company network via NFS or SMB protocol:
Up to 1 Gbit/s
- For **LSV2 protocol** (in conjunction with TNCremoNT or RemoTools):
2 to 5 Mbps (depending on file type and network utilization)

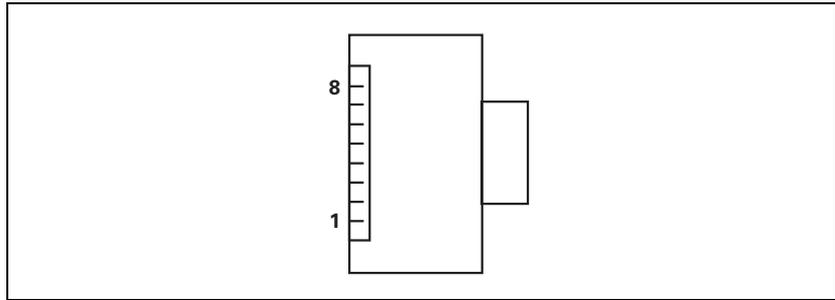
Assignment with 100 Mb Ethernet interface:

RJ45 connection (female) 8-pin	Assignment
1	TX+
2	TX-
3	RX+
4	Do not assign
5	Do not assign
6	RX-
7	Do not assign
8	Do not assign
Housing	External shield

Assignment with 1 Gbit Ethernet interface (available on all current MCs):

RJ45 connection (female) 8-pin	Assignment
1	MX0+
2	MX0-
3	MX1+
4	MX2+
5	MX2-
6	MX1-
7	MX3+
8	MX3-
Housing	External shield

Face of the connector:



Meanings of the LEDs on the Ethernet data interface:

LED	Status	Meaning
Green	On	Interface active
	Off	Interface inactive
Orange	On	1 Gbit interface active
Yellow	Blinking	100 Mb network or 1 Gbit network
	Off	10 Mb network

For more information on the Ethernet interface, see "X26, X116: Ethernet interface RJ45-port (100BaseT)" on page 1863.

X27:
RS-232-C/V.24 data
interface

■ Maximum cable length with RS-232-C/V.24 is 20 meters

Pin layout:



Note

The interface complies with the requirements of EN 50178 for "low voltage electrical separation."

For more information on the USB interface, see "iTNC Serial Data Interfaces" on page 1899.

25-pin adapter block:

MC 6xxx		Conn. cable 365725-xx			Adapter block 310085-01		Conn. cable 274545-xx		
Male	Assignment	Female	Color	Female	Male	Female	Male	Color	Female
1	Do not assign	1		1	1	1	1	White/ Brown	1
2	RXD	2	Yellow	3	3	3	3	Yellow	2
3	TXD	3	Green	2	2	2	2	Green	3
4	DTR	4	Brown	20	20	20	20	Brown	8
5	Signal GND	5	Red	7	7	7	7	Red	7
6	DSR	6	Blue	6	6	6	6		6
7	RTS	7	Gray	4	4	4	4	Gray	5
8	CTS	8	Pink	5	5	5	5	Pink	4
9	Do not assign	9					8	Violet	20
Hsg.	External shield	Hsg.	External shield	Hsg.	Hsg.	Hsg.	Hsg.	External shield	Hsg.

9-pin adapter block:

MC 6xxx		Conn. cable 355484-xx			Adapter block 363987-02		Conn. cable 366964-xx		
Male	Assignment	Female	Color	Male	Female	Male	Female	Color	Female
1	Do not assign	1	Red	1	1	1	1	Red	1
2	RXD	2	Yellow	2	2	2	2	Yellow	3
3	TXD	3	White	3	3	3	3	White	2
4	DTR	4	Brown	4	4	4	4	Brown	6
5	Signal GND	5	Black	5	5	5	5	Black	5
6	DSR	6	Violet	6	6	6	6	Violet	4
7	RTS	7	Gray	7	7	7	7	Gray	8
8	CTS	8	White/ Green	8	8	8	8	White/ Green	7
9	Do not assign	9	Green	9	9	9	9	Green	9
Hsg.	External shield	Hsg.	External shield	Hsg.	Hsg.	Hsg.	Hsg.	External shield	Hsg.

3.28.1 USB interface (USB 2.0)

**X141, X142, X143,
X144**

Pin layout for USB connection (type A):

USB connection (female) 4-pin	Assignment
1	+5 V
2	USBP-
3	USBP+
4	GND



Note

If USB components that are connected to one of the USB ports require more than 0.5 A, a separate power supply becomes necessary for these components. One possibility is the USB hub (USB 2.0) from HEIDENHAIN (582884-02).

If a USB hub is connected to one of the USB ports, the maximum permissible length of the USB cable (ID 624775-xx) is reduced to 20 m.

USB hub

The power supply for the USB hub must comply with EN 50 178, 5.88 requirements for "low voltage electrical separation."

For more information on the USB interface, see "The USB Interface of the iTNC" on page 1894.



3.28.2 HDL interface

X249

Pin layout:

MC 6xxx, X249		Connecting cable ID 625901-xx	BF 2xx, X2	
25-pin connection	Assignment		25-pin connection	Assignment
1	TD2+		1	TD2+
2	TD2-		2	TD2-
3	TD1+		3	TD1+
4	TD1-		4	TD1-
5	TD0+		5	TD0+
6	TD0-		6	TD0-
7	TC+		7	TC+
8	TC-		8	TC-
9	Do not assign		9	Do not assign
10	Do not assign		10	Do not assign
11	+ 5 V		11	Do not assign
12	TxD+		12	RxD+
13	TxD-		13	RxD-
14	TD2S		14	TD2S
15	Do not assign		15	EQSEL1
16	TD1S		16	TD1S
17	Do not assign		17	GND for EQSEL
18	TD0S		18	TD0S
19	Do not assign		19	EQSEL0
20	TCS		20	TCS
21	Do not assign		21	Do not assign
22	Do not assign		22	Do not assign
23	GND		23	GND
24	RxD+		24	TxD+
25	RxD-		25	TxD-

3.29 MB 720 (FS) machine operating panel

For machines with up to six axes, HEIDENHAIN offers the MB 720 machine operating panel (15-inch) with HSCI interface.

A blue status LED is located over each of the 36 snap-on machine keys. The LEDs are internally permanently connected with PLC outputs and can therefore be freely controlled via the PLC program. The output addresses of the LEDs come from the HDD file of the machine operating panel.



Note

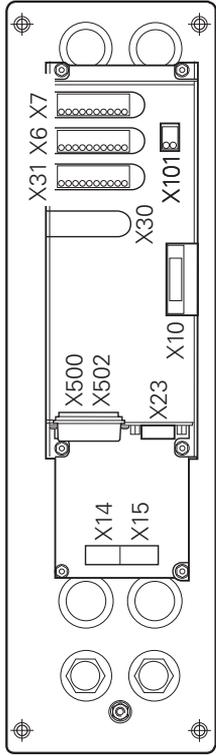
The current HDD files of the HEIDENHAIN operating panel components are a part of the installation package of IOconfig. You can download the current HDD files as a ZIP archive from the HESIS-Web Including Filebase on the Internet (<http://hesis.heidenhain.de>).

For this area you need access rights.

As an alternative, please get in touch with your contact partner at HEIDENHAIN to obtain the current HDD file.

On the underside of the machine operating panel are terminal strips bearing the PLC inputs as well as the PLC outputs. Also, connection X23 for HR handwheels is on the underside of the MB 720.

It must be ensured that negative pressure cannot develop in the panel in which the machine operating panel is installed. Negative pressure in the panel can lead to increased contamination of the machine operating panel, which must be avoided by enabling pressure compensation, for example via the cable duct between the electrical cabinet and the panel.

Pin layout for MB 720 (FS)	Connector	Function	Page
	X18	Emergency stop (MB) (pre-assigned by HEIDENHAIN)	332
	X500	HSCI output	208
	X502	HSCI input	208
	X6	PLC inputs	332
	X7	PLC outputs	333
	X10	Interface to keyboard and potentiometers	322
	X23	Handwheel connection	347
	X30	Connection for handwheel adapter	323
	X31	Permissive button	335
	X13	Key for Control Voltage On (pre-assigned by HEIDENHAIN)	
	X14	Connection for the NC start key (pre-assigned by HEIDENHAIN)	
	X15	Connection for the NC stop key (pre-assigned by HEIDENHAIN)	
	X101	24 V NC power supply	323
	⊖	Protective ground	

Besides the snap-on keys, there are also keys with switch contacts that can be wired by the machine tool builder depending on the application. Specifications of the switch contacts on keys such as EMERGENCY STOP, NC start, NC stop, Control voltage on:

- Rated voltage: +24 V DC
- Maximum current: 1.2 A

In standard wiring configuration, these keys are supplied with +24 V NC and the lamps in these keys are driven with +5 V-NC / 10 mA. The keys and switch contacts are not designed to power the two switch contacts or the LED with different voltages, e.g. on contact with 24 V NC and the other with 24 V PLC.



Note

If the key information is needed for external systems that are supplied with 24 V PLC, HEIDENHAIN recommends forwarding the information through the PLC program and a normal PLC output.

X18: EMERGENCY STOP on MB

MB 720 without FS:



Note

With the MB 720 without FS, the EMERGENCY STOP must be wired externally in the EMERGENCY STOP chain as before.

Pin layout of the emergency stop button, only relevant to MB 720 without FS:

Connecting terminals	Assignment
11	Emergency stop contact 1 / terminal 1
12	Emergency stop contact 1 / terminal 2
21	Emergency stop contact 2 / terminal 1
22	Emergency stop contact 2 / terminal 2

MB 720 FS:

With the MB 720 FS, the EMERGENCY STOP is evaluated through two channels by the control via HSCI and is normally wired via connector X18.

General information on the emergency stop button:

Emergency stop buttons are to be used exclusively for emergency stop purposes. Under normal operating conditions, a machine must not be switched off via the emergency stop buttons. The proper functioning of all emergency stop buttons is to be tested annually by pressing these buttons.

X7: PLC outputs

Pin layout:

Connecting terminals	Assignment
1	O0
2	O1
3	O2
4	O3
5	O4
6	O5
7	O6
8	O7
9	+24 V NC (available here)
10	0 V NC (available here)

In the event of an overload or short circuit and as a consequence of excessive temperature, the individual PLC outputs switch off automatically. Then the outputs will try to switch on again cyclically.



Current carrying capacity of the outputs: Maximum 150 mA per output



Danger

Please note that the outputs of connector X7 are powered internally by +24 V NC, and therefore supply +24 V NC at HIGH level.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.

Each of the switching outputs at X7 supplies up to 150 mA of output current and are provided for driving the lamps on the MP 720.



Note

Immediately after the supply voltage has been switched on, all switching outputs at X7 switch to HIGH level for approx. 100 ms, which is indicated by short flashing of the connected lamps. Keep this characteristic in mind if you connect inductive loads or other consumers to the MB 720.

HEIDENHAIN recommends:

Connect inductive loads and other consumers to the PLC outputs of the PL 6xxx or UxC 11x. The described effect during switch-on does not occur there. Use the outputs at X7 only for the control of lamps.

X6: PLC inputs

Pin layout of MB 720 without FS:

Connecting terminals	Assignment
1	I0
2	I1
3	I2
4	I3
5	I4
6	I5
7	I6
8	I7
9	Reserved (do not use)
10	Reserved (do not use)

Pin layout of MB 720 FS:

Connecting terminals	Assignment
1	I0.A
2	I1.A
3	I2.A
4	I3.A
5	I0.B
6	I1.B
7	I2.B
8	I3.B
9	Power supply for channel A (-TEST.A)
10	Power supply for channel B (-TEST.B)



Danger

Please note that the MB 720 is powered by +24 V NC.

For the entire HSCI system, the +24 V NC power supply voltage is required to be safely separated voltage. It must also be safely separated from the +24 V PLC!

**X31: MB 720,
permissive button /
NC-start / NC-stop**

The NC start and NC stop keys are connected directly on the keyboard PCB of the MB via the X14 and X15 connections and are evaluated as with the other machine keys, see "Key assignment for machine operating panel" on page 324. The keys are configured via the HDD file with IOconfig. Alternatively the keys can be connected to the X31 connection. In this case the assignment specified in the tables below applies and separate handling of the keys with IOconfig is necessary.

The NC start key is a normally open contact, and the NC stop key is normally closed.



Standard wiring of connection X31 for MB without FS:

Connecting terminals	Assignment
1	Reserved (do not use)
2	Reserved (do not use)
3	Reserved (do not use)
4	Reserved (do not use)
5	Reserved, only use if X14 terminal is vacant: NC start
6	Reserved (do not use)
7	Reserved, only use if X14 terminal is vacant: NC start power supply (+24 V NC)
8	Reserved, only use if X15 terminal is vacant: NC stop
9	Reserved (do not use)
10	Reserved, only use if X15 terminal is vacant: NC stop power supply (+24 V NC)

Pin layout of MB with FS:

Connecting terminals	Assignment
1	Optional: Permissive button channel A / terminal 1 (+24 V)
2	Optional: Permissive button channel A / terminal 2
3	Optional: Permissive button channel B / terminal 1 (+24 V)
4	Optional: Permissive button channel B / terminal 2
5	Reserved, only use if X14 terminal is vacant: NC start channel A / terminal 1
6	Reserved, only use if X14 terminal is vacant: NC start channel B / terminal 2
7	Reserved, only use if X14 terminal is vacant: NC start shared terminal (+24 V NC)
8	Reserved, only use if X15 terminal is vacant: NC stop channel A / terminal 1
9	Reserved, only use if X15 terminal is vacant: NC stop channel B / terminal 2
10	Reserved, only use if X15 terminal is vacant: NC stop shared terminal (+24 V NC)



NC Start and NC Stop are normally open contacts on the MB (FS).



Danger

Please note that the MB is powered by +24 V NC.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.

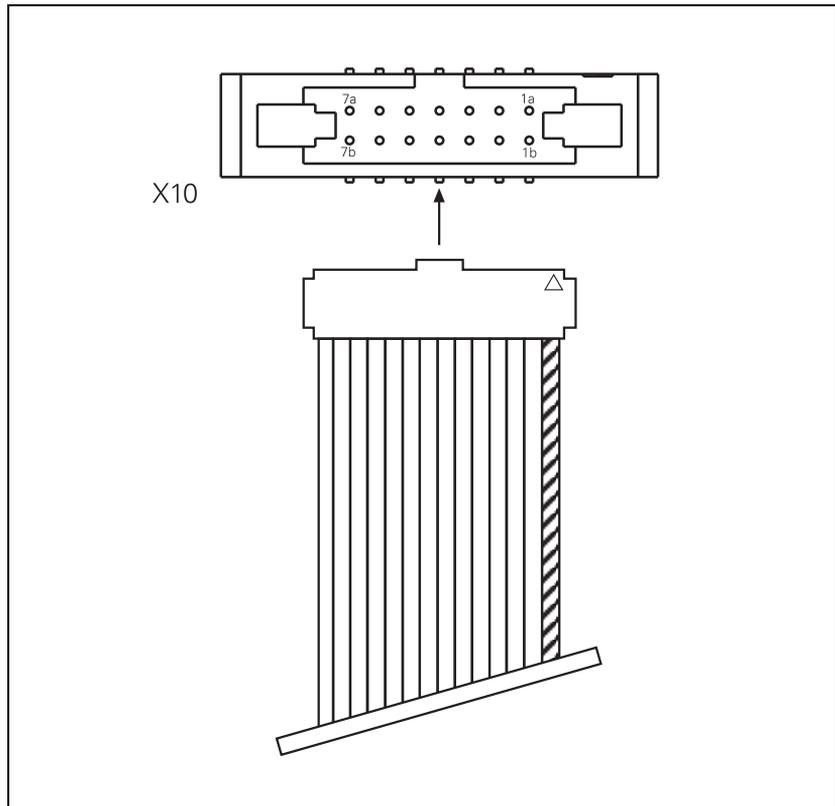


Attention

The interface is electrically separated (protective extra-low voltage (PELV) as per EN 61800-5-1).

X10: Interface to keyboard and potentiometers

Designation of the terminals:



Pin layout of the ribbon cable (included, length 54 cm):

Connecting terminals	Assignment
1a	Potentiometer 1
2a	Potentiometer 3
3a	Do not assign
4a	Do not assign
5a	Do not assign
6a	+ 5 V
7a	0 V
1b	Potentiometer 2
2b	---
3b	Do not assign
4b	Do not assign
5b	---
6b	+ 5 V
7b	0 V

X101: Power supply

Pin layout:

Connecting terminals	Assignment
+ / terminal 1	+24 V NC
- / terminal 2	0 V NC

Power consumption of the operating panel units without HR handwheel and controlled inputs/outputs:

Power consumption of MB 720, MB 620: 4.0 W

X30: Handwheel connection, permissive button / emergency stop



MB 720 without FS:

Connection X30 not used on MB 720 without FS.

Note

With the MB 720 without FS, the permissive buttons and the EMERGENCY STOP of the handwheel must remain externally wired in corresponding safety circuits!

MB 720 FS:

Pin layout of MB 720 FS:

Connecting terminals	Assignment
1	Permissive button channel A / terminal 1 (+24 V NC)
2	Permissive button channel A / terminal 2
3	Permissive button channel B / terminal 1 (+24 V NC)
4	Permissive button channel B / terminal 2
5	Emergency stop channel A / supply for channel A (-TEST.A)
6	Emergency stop channel A / terminal 2
7	Emergency stop channel B / supply for channel B (-TEST.B)
8	Emergency stop channel B / terminal 2

The emergency stop inputs (pin 6 and pin 8) are active even if the HR handwheel has been deactivated via machine parameter MP7640. It is therefore always necessary to wire the emergency stop inputs correctly.

Emergency stop buttons are to be used exclusively for emergency stop purposes. Under normal operating conditions, a machine must not be switched off via the emergency stop buttons. The proper functioning of all emergency stop buttons is to be tested annually by pressing these buttons.

General:



Danger

Please note that the MB 720 is powered by +24 V NC.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.

Key assignment for machine operating panel

□01 S1 IV+	□02 S2 Z+	□03 S3 Y+	□04 S4 V+	□05 S5 VI+	□06 S6	□07 S7	□08 S8	□09 S9	□010 S10 FN 1	□011 S11	□012 S12	O13 S13 NC-Stop
□014 S14 X-	□015 S15	□016 S16 X+	□017 S17	□018 S18	□019 S19	□020 S20	□021 S21	□022 S22	□023 S23 FN 2	□024 S24	□025 S25	O26 S26 NC-Start
□027 S27 Y-	□028 S28 Z-	□029 S29 IV-	□030 S30 V-	□031 S31 VI-	□032 S32	□033 S33	□034 S34	□035 S35	□036 S36 FN 3	□037 S37	□038 S38	O39 S39 CVO



3.30 HSCI adapter for PLB 6001 (FS) OEM-specific machine operating panel

HEIDENHAIN offers the PLB 6001 (FS) HSCI adapter with HSCI interface for connecting an OEM-specific machine operating panel. The HSCI system therefore interprets and treats the PLB 6001 (FS) as an MB machine operating panel.

On the underside of the machine operating panel are terminal strips bearing the PLC inputs as well as the PLC outputs. Also, connection X23 for HR handwheels is on the underside of the PLB 6001 (see "X23: Handwheel input" on page 347).



Danger

Please note that the PLB 6001 (FS) is supplied with +24 V NC.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.

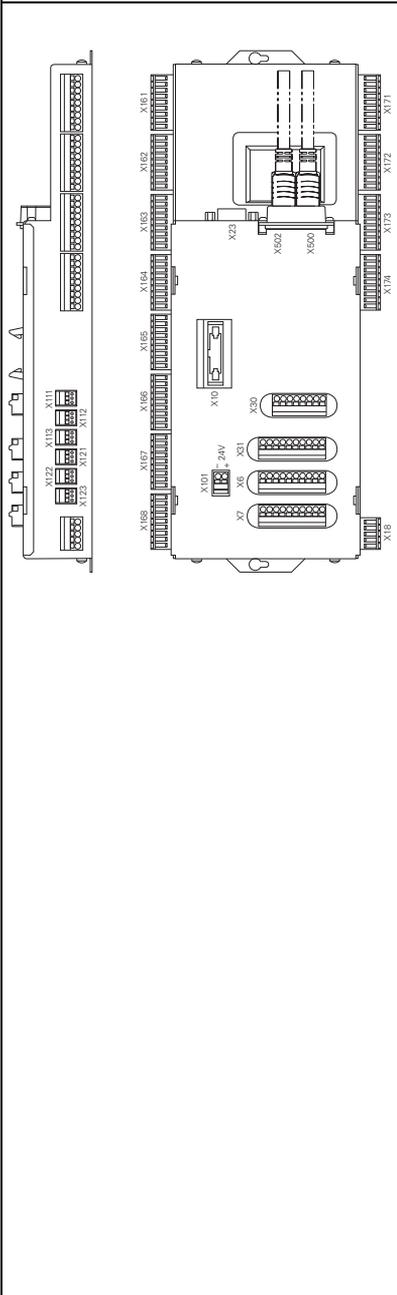
PLB 6001 pin layout	Connector	Function	Page
	X500	HSCI output	208
	X502	HSCI input	208
	X6	PLC inputs	345
	X7	PLC outputs	344
	X10	Interface to keyboard and potentiometers	346
	X18	Reserved	–
	X23	Handwheel connection	347
	X30	Reserved	–
	X31	Permissive button, NC start, NC stop	348
	X101	24 V NC power supply	350
	X111	Potentiometer connection 1	354
	X112	Potentiometer connection 2	354
	X113	Potentiometer connection 3	354
	X121	Reserved	–
	X122	Reserved	–
	X123	Reserved	–
	X161	PLC inputs I0 to I7	351
	X162	PLC inputs I8 to I15	351
	X163	PLC inputs I16 to I23	351
	X164	PLC inputs I24 to I31	351
	X165	PLC inputs I32 to I39	351
	X166	PLC inputs I40 to I47	351
	X167	PLC inputs I48 to I55	351
	X168	PLC inputs I56 to I63	351
	X171	PLC outputs O0 to O7	352
	X172	PLC outputs O8 to O15	352
	X173	PLC outputs O16 to O23	352
	X174	PLC outputs O24 to O31	352
	⊕	Protective ground	352



Attention

Do not engage or disengage any connecting elements while the unit is under power!



PLB 6001 FS pin layout ID 722083-xx	Connector	Function	Page
	X500	HSCI output	208
	X502	HSCI input	208
	X6	PLC inputs	345
	X7	PLC outputs	344
	X10	Interface to keyboard and potentiometers	346
	X18	Emergency stop input	347
	X23	Handwheel connection	347
	X30	Permissive button, EMERGENCY STOP on handwheel	348
	X31	Permissive button, NC start, NC stop	348
	X101	24 V NC power supply	350
	X111	Potentiometer connection 1	354
	X112	Potentiometer connection 2	354
	X113	Potentiometer connection 3	354
	X121	Reserved	–
	X122	Reserved	–
	X123	Reserved	–
	X161	FS inputs I0.A to I7.A	353
	X162	FS inputs I8.A to I15.A	353
	X163	FS inputs I16.A to I23.A	353
	X164	FS inputs I24.A to I31.A	353
	X165	FS inputs I0.B to I7.B	353
	X166	FS inputs I8.B to I15.B	353
	X167	FS inputs I16.B to I23.B	353
	X168	FS inputs I24.B to I31.B	353
	X171	PLC outputs O0 to O7	352
	X172	PLC outputs O8 to O15	352
	X173	PLC outputs O16 to O23	352
	X174	PLC outputs O24 to O31	352
	⊕	Protective ground	352



X7: PLC outputs

Pin layout:

Connecting terminals	Assignment
1	O0
2	O1
3	O2
4	O3
5	O4
6	O5
7	O6
8	O7
9	+24 V NC (available here)
10	0 V NC (available here)

Current carrying capacity of the outputs: Maximum 150 mA per output. In the event of an overload or short circuit and as a consequence of excessive temperature, the individual PLC outputs switch off automatically. Then the outputs will try to switch on again cyclically.



Danger

Please note that the outputs of connector X7 are powered internally by +24 V NC, and therefore supply +24 V NC at HIGH level.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.

Each of the switching outputs at X7 supplies up to 150 mA of output current. They are provided for driving the lamps on the operating panel.



X6: PLC inputs

Pin layout of PLB 6001 without FS:

Connecting terminals	Assignment
1	I0
2	I1
3	I2
4	I3
5	I4
6	I5
7	I6
8	I7
9	Reserved (do not use)
10	Reserved (do not use)

Pin layout of PLB 6001 FS:

Connecting terminals	Assignment
1	I0.A
2	I1.A
3	I2.A
4	I3.A
5	I0.B
6	I1.B
7	I2.B
8	I3.B
9	Power supply for channel A (-TEST.A)
10	Power supply for channel B (-TEST.B)



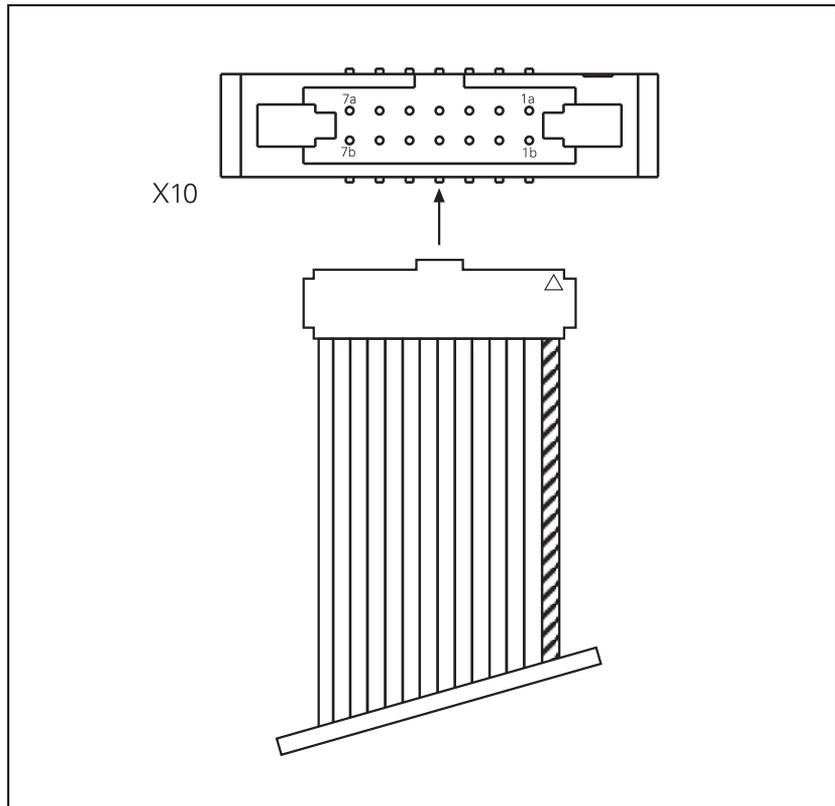
Danger

Please note that the PLB 6001 is supplied with +24 V NC.

For the entire HSCI system, the +24 V NC power supply voltage is required to be safely separated voltage. It must also be safely separated from the +24 V PLC!

X10: Interface to keyboard and potentiometers

Designation of the terminals:



Pin layout of the ribbon cable (included, length 54 cm):

Connecting terminals	Assignment
1a	Potentiometer 1
2a	Potentiometer 3
3a	Do not assign
4a	Do not assign
5a	Do not assign
6a	+ 5 V
7a	0 V
1b	Potentiometer 2
2b	---
3b	Do not assign
4b	Do not assign
5b	---
6b	+ 5 V
7b	0 V

X18: EMERGENCY STOP

With the PLB 6001 without FS, the EMERGENCY STOP must be wired externally in the EMERGENCY STOP chain as before. The connector X18 does not support a dual-channel evaluation of the EMERGENCY STOP button. These inputs must therefore not be used for evaluating EMERGENCY STOP!

With the PLB 6001 FS, the EMERGENCY STOP is evaluated through two channels by the control via HSCI.

X18 pin layout on PLB 6001 FS:

Connecting terminals	Assignment
1	Power supply for emergency stop channel A (-TEST.A)
2	Emergency stop channel A
3	Power supply for emergency stop channel B (-TEST.B)
4	Emergency stop channel B

Emergency stop buttons are to be used exclusively for emergency stop purposes. Under normal operating conditions, a machine must not be switched off via the emergency stop buttons. The proper functioning of all emergency stop buttons is to be tested annually by pressing these buttons.

X23: Handwheel input

Pin layout:

D-sub connector (female) 9-pin	Assignment
1	CTS
2	0 V
3	RTS
4	+12 V
5	Do not assign
6	DTR
7	TxD
8	RxD
9	DSR
Housing	External shield



Attention

The interface is electrically separated (protective extra-low voltage (PELV) as per EN 61800-5-1).



Danger

The connector for the handwheel on the machine operating panel, as well as the connector on the handwheel itself, may be removed only by trained and qualified personnel, even if it can be removed without using a tool. If the handwheel connector is removed, only basic insulation from line power (230 V) is provided!

X30: Permissive button, EMERGENCY STOP on handwheel

Pin layout of PLB 6001 FS:

Connecting terminals	Assignment
1	Permissive button channel A / terminal 1 (+24 V NC)
2	Permissive button channel A / terminal 2
3	Permissive button channel B / terminal 1 (+24 V NC)
4	Permissive button channel B / terminal 2
5	Emergency stop channel A / supply for channel A (-TEST.A)
6	Emergency stop channel A / terminal 2
7	Emergency stop channel B / supply for channel B (-TEST.B)
8	Emergency stop channel B / terminal 2

X31: Permissive button/NC-start, NC-stop

Connection X31 of PLB 6001:

Connecting terminals	Assignment
1	Reserved (do not use)
2	Reserved (do not use)
3	Reserved (do not use)
4	Reserved (do not use)
5	NC start
6	Reserved (do not use)
7	NC start power supply (+24 V NC)
8	NC stop
9	Reserved (do not use)
10	NC stop power supply (+24 V NC)

X31 pin layout on PLB 6001 FS:

Connecting terminals	Assignment
1	Optional: Permissive button channel A / terminal 1 (+24 V)
2	Optional: Permissive button channel A / terminal 2
3	Optional: Permissive button channel B / terminal 1 (+24 V)
4	Optional: Permissive button channel B / terminal 2
5	NC start channel A / terminal 1
6	NC start channel B / terminal 2
7	NC start shared terminal (+24 V NC)
8	NC stop channel A / terminal 1
9	NC stop channel B / terminal 2
10	NC stop shared terminal (+24 V NC)





Danger

Please note that the PLB 6001 (FS) is supplied with +24 V NC.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.



Attention

The interface is electrically separated (protective extra-low voltage (PELV) as per EN 61800-5-1).

X101: Power supply Pin layout:

Connecting terminals	Assignment
+ / terminal 1	+24 V NC
- / terminal 2	0 V NC

Power consumption of the operating panel units without HR handwheel and controlled inputs/outputs:

Power consumption of the PLB 6001 (FS):5.0 W



**X161 to X168:
PLC inputs**

PLB 6001 without FS:

X161: PLC inputs									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001	0 V PLC	I0	I1	I2	I3	I4	I5	I6	I7

X162: PLC inputs									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001	0 V PLC	I8	I9	I10	I11	I12	I13	I14	I15

X163: PLC inputs									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001	0 V PLC	I16	I17	I18	I19	I20	I21	I22	I23

X164: PLC inputs									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001	0 V PLC	I24	I25	I26	I27	I28	I29	I30	I31

X165: PLC inputs									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001	0 V PLC	I32	I33	I34	I35	I36	I37	I38	I39

X166: PLC inputs									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001	0 V PLC	I40	I41	I42	I43	I44	I45	I46	I47

X167: PLC inputs									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001	0 V PLC	I48	I49	I50	I51	I52	I53	I54	I55

X168: PLC inputs									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001	0 V PLC	I56	I57	I58	I59	I60	I61	I62	I63

0 V PLC must be connected to terminal 1.



**X171 to X174:
PLC outputs**

PLB 6001 with or without FS:

X171: PLC outputs									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001	O0	O1	O2	O3	O4	O5	O6	O7	24 V PLC for O0 to O7

X172: PLC outputs									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001	O8	O9	O10	O11	O12	O13	O14	O15	24 V PLC for O8 to O15

X173: PLC outputs									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001	O16	O17	O18	O19	O20	O21	O22	O23	24 V PLC for O16 to O23

X174: PLC outputs									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001	O24	O25	O26	O27	O28	O29	O30	O31	24 V PLC for O24 to O31

+24 V PLC must be connected to terminal 9.

Current carrying capacity of the outputs: Maximum 150 mA per output

Please note that the outputs of connectors X171 to X174 are galvanically isolated internally from the +24 V NC supply voltage supplied via X101. The outputs can therefore be supplied with +24 V PLC or +24 V NC.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.



**X161 to X168:
PLC inputs**

PLB 6001 FS:

X161: PLC inputs, channel A									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001 FS	0 V PLC	I0.A	I1.A	I2.A	I3.A	I4.A	I5.A	I6.A	I7.A

X162: PLC inputs, channel A									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001 FS	0 V PLC	I8.A	I9.A	I10.A	I11.A	I12.A	I13.A	I14.A	I15.A

X163: PLC inputs, channel A									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001 FS	0 V PLC	I16.A	I17.A	I18.A	I19.A	I20.A	I21.A	I22.A	I23.A

X164: PLC inputs, channel A									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001 FS	0 V PLC	I24.A	I25.A	I26.A	I27.A	I28.A	I29.A	I30.A	I31.A

X165: PLC inputs, channel B									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001 FS	0 V PLC	I0.B	I1.B	I2.B	I3.B	I4.B	I5.B	I6.B	I7.B

X166: PLC inputs, channel B									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001 FS	0 V PLC	I8.B	I9.B	I10.B	I11.B	I12.B	I13.B	I14.B	I15.B

X167: PLC inputs, channel B									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001 FS	0 V PLC	I16.B	I17.B	I18.B	I19.B	I20.B	I21.B	I22.B	I23.B



X168: PLC inputs, channel B									
Assignment	Terminal								
	1	2	3	4	5	6	7	8	9
PLB 6001 FS 0 V PLC	I24.B	I25.B	I26.B	I27.B	I28.B	I29.B	I30.B	I31.B	

0 V PLC must be connected to terminal 1.

Pin layout:

**X111 to X113:
Potentiometer
connection**

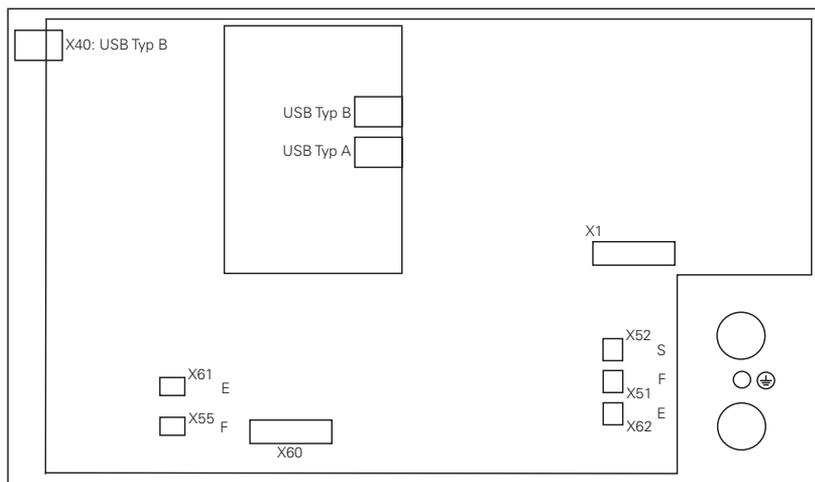
Connecting terminals	Assignment
1	0 V potentiometer
2	Potentiometer arm
3	+5 V potentiometer



3.31 TE 620 keyboard unit

NC operating panel for 15-inch without touchpad.

Connection overview



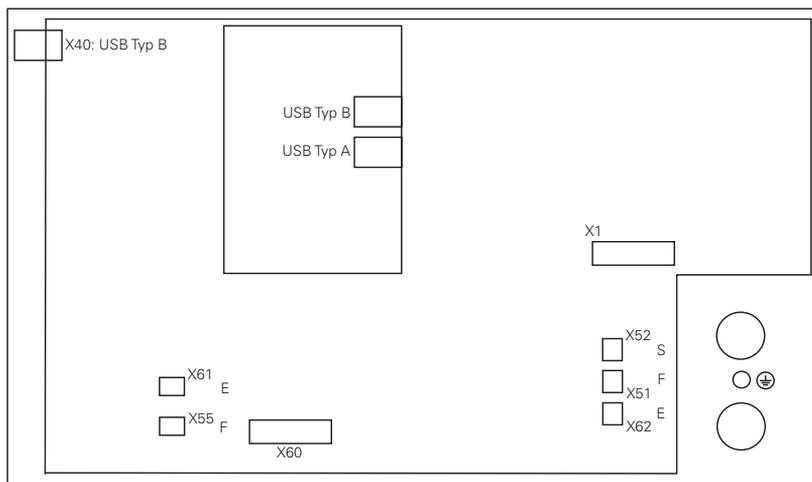
Connector	Function
USB Type B	USB connection to BF 2xx. The TE 6xx is connected to the USB hub of the BF 2xx. The maximum permissible cable length for this USB connection is 3 m. The USB hub of the TE 6xx only supports USB 1.1.
USB Type A	Freely available USB connection (USB 1.1) Maximum load capacity of USB output: 1 x 100 mA
X1	Screen soft keys of BF 2xx X3 via ribbon cable to keyboard unit X1
X60	Potentiometer of keyboard unit via ribbon cable to machine operating panel X10
X52 (S)	Connection for spindle-speed override potentiometer
X51 (F)	Connection for feed-rate override potentiometer
X62 (E)	Connection for rapid-traverse override potentiometer, alternative to X61 (not with TE 63x)
X55	Connection for feed-rate override potentiometer; alternative to X51 (not with TE 63x)
X61	Connection for rapid-traverse override potentiometer
⊕	Protective ground (M5)

The TE keyboard units are supplied with power via the USB port.

3.32 TE 630 keyboard unit

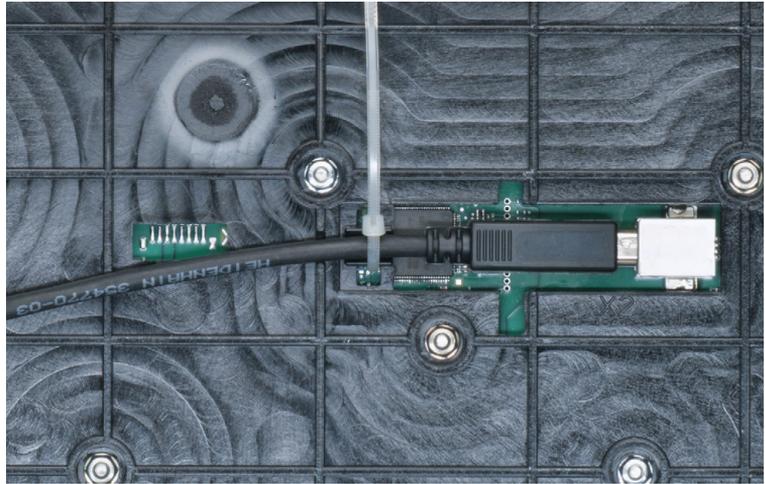
NC operating panel for 15-inch with touchpad.

Connection overview



Connector	Function
USB Type B	USB connection to BF 2xx. The TE 6xx is connected to the USB hub of the BF 2xx. The maximum permissible cable length for this USB connection is 3 m. The USB hub of the TE 6xx only supports USB 1.1.
USB Type A	Freely available USB connection (USB 1.1) Maximum load capacity of USB output: 1 x 100 mA
X1	Screen soft keys of BF 2xx X3 via ribbon cable to keyboard unit X1
X60	Potentiometer of keyboard unit via ribbon cable to machine operating panel X10
X52 (S)	Connection for spindle-speed override potentiometer
X51 (F)	Connection for feed-rate override potentiometer
X62 (E)	Connection for rapid-traverse override potentiometer, alternative to X61 (not with TE 63x)
X55	Connection for feed-rate override potentiometer; alternative to X51 (not with TE 63x)
X61	Connection for rapid-traverse override potentiometer
⊕	Protective ground (M5)

The TE keyboard units are supplied with power via the USB port (USB type B). USB cables that are connected to the USB ports of the TE keyboard units must be fixed with the corresponding cable tie (strain relief) to protect the USB connectors from mechanical load.



Possible configurations for the potentiometers of a TE 6xx

The override potentiometers of a TE 6xx can be configured in machine parameter MP7620 bit 5 and bit 9. If bit 5 is set in MP7620, the potentiometers are evaluated as feed-rate and rapid-traverse override potentiometers. If an HSCI TE keyboard unit with three override potentiometers (spindle, rapid traverse and feed rate) is used, bit 9 must be set in MP7620. If bit 5 and bit 9 are both set at the same time, an error message is triggered due to invalid configuration.

If you configure a rapid-traverse override (bit 5), the characteristic curve is nonlinear (range 0 to 10000/15000), regardless of bit 3. Then the nonlinear characteristic curve applies to all overrides.

For the evaluation of the potentiometers, the following distinctions are made between HSCI and non-HSCI hardware:

- TE for systems without HSCI
Configuration with feed-rate and rapid-traverse overrides (MP7620, bit 5). The spindle-speed override is used as rapid-traverse override. The NC keys are used to realize the spindle speed override. If hardware for systems without HSCI is used, bit 9 is ignored.
- TE for systems without HSCI
Configuration for three potentiometers with feed-rate, spindle-speed and rapid-traverse overrides (MP7620, bit 5). This configuration has no effect. The potentiometers are evaluated, as usual, as feed-rate and spindle-speed overrides. Bit 9 is ignored.
- TE for systems with HSCI
Configuration with feed-rate and rapid-traverse overrides (MP7620, bit 5). The spindle-speed override input **S** (X52) on the rear of the TE is not evaluated. The rapid-traverse override must be connected to the provided connector **E** (X61, X62). There are no NC keys for spindle-speed override. Machine operating panel keys must be used instead, and the function must be realized in the PLC program (the PLC program must transmit the information to W764).
The +, – and 100% keys for spindle-speed override can be snapped on to an HSCI machine operating panel. Contrary to keyboards for systems without HSCI, these keys are only available as PLC inputs, and not as NC keys. They are not evaluated automatically by the NC software. The PLC program must evaluate the keys and transmit the calculated value to the NC in W764 (spindle-speed override). The NC software limits this value depending on MP3310.0 (max. spindle-speed override value) and MP3310.1 (minimum spindle-speed override value).
- TE with three potentiometers for systems with HSCI
Configuration for three potentiometers with feed-rate, spindle-speed and rapid-traverse overrides (MP7620, bit 9): Feed-rate, spindle-speed and rapid-traverse overrides are evaluated.

MP_7620 **Bit#9 – HSCI keyboards with three potentiometers**

Input: 0: Keyboard with two potentiometers
 1: HSCI keyboard with three potentiometers



3.33 TE 635Q (FS) keyboard unit

The operating panel for 15 inch is the TE 630 plus the machine operating keys.

Machine operating panel

The description of the TE 635Q machine operating panel is the same as that of the MB 620, see page 317. The TE 635Q, too, is also available in a TE 635Q FS version, in which the machine operating panel is designed for use with functional safety (FS).



Danger

Please note that the TE 635Q is supplied with +24 V NC.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.

Keyboard unit

The description of the TE 635Q control panel corresponds to the description of the TE 630, see page 356.

3.34 TE 645Q (FS) keyboard unit

The operating panel for 19 inch is the TE 630 plus the machine operating keys.

Machine operating panel

The description of the TE 635Q machine operating panel is the same as that of the MB 620, see page 317. The TE 635Q, too, is also available in a TE 635Q FS version, in which the machine operating panel is designed for use with functional safety (FS).



Danger

Please note that the TE 635Q is supplied with +24 V NC.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.

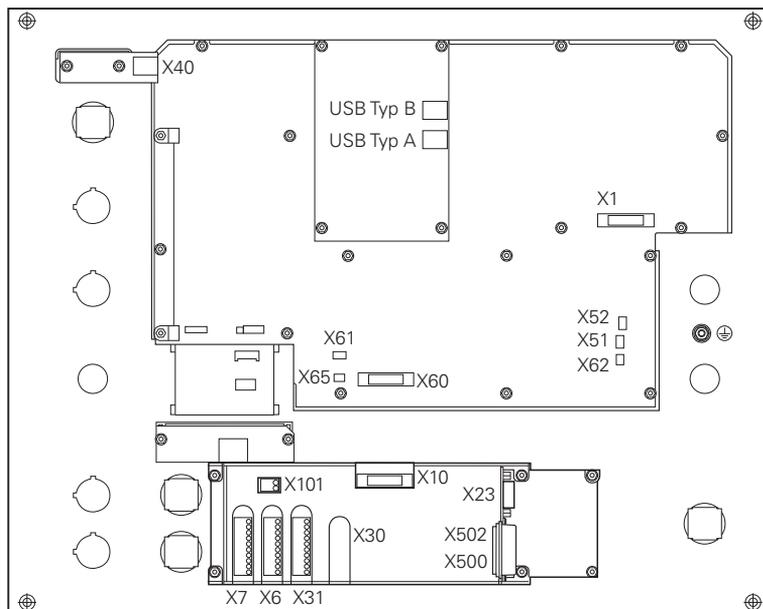
Keyboard unit

The description of the TE 635Q control panel corresponds to the description of the TE 630, see page 356.

Connection overview for the TE 645Q

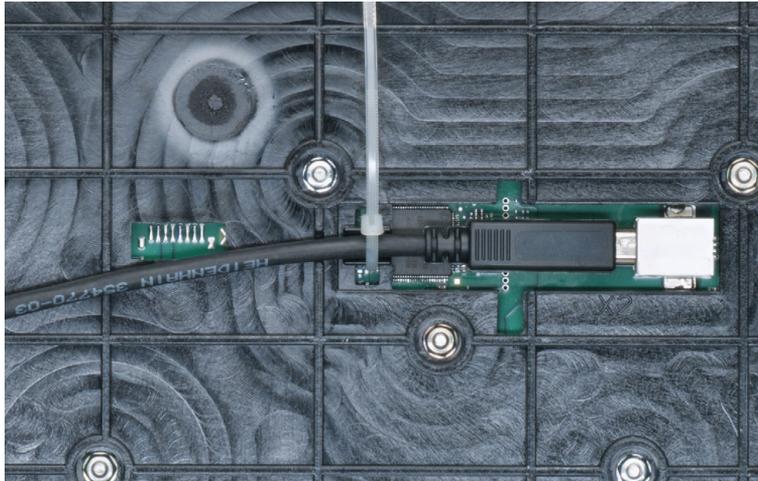
TE 645Q, 19-inch keyboard unit, integrated machine operating panel

TE 645 Q pin layout



Connector	Function
USB Type B	USB connection to BF 2xx. The TE 6xx is connected to the USB hub of the BF 2xx. The maximum permissible cable length for this USB connection is 3 m. The USB hub of the TE 6xx only supports USB 1.1.
X40: USB Type B	USB connection to USB connector on the front of the TE
USB Type A	Freely available USB connection (USB 1.1) Maximum load capacity of USB output: 1 x 100 mA
X1	Screen soft keys of BF 2xx X3 via ribbon cable to keyboard unit X1
X60	Potentiometer of keyboard unit via ribbon cable to machine operating panel X10
X52 (S)	Connection for spindle-speed override potentiometer
X51 (F)	Connection for feed-rate override potentiometer
X62 (E)	Connection for rapid-traverse override potentiometer, alternative to X61
X55	Connection for feed-rate potentiometer, alternative to X51
X61	Connection for rapid-traverse override potentiometer
⊕	Protective ground (M5)

The TE keyboard units are supplied with power via the USB port (USB type B). USB cables that are connected to the USB ports of the TE keyboard units must be fixed with the corresponding cable tie (strain relief) to protect the USB connectors from mechanical load.



Possible configurations for the potentiometers of a TE 6xx

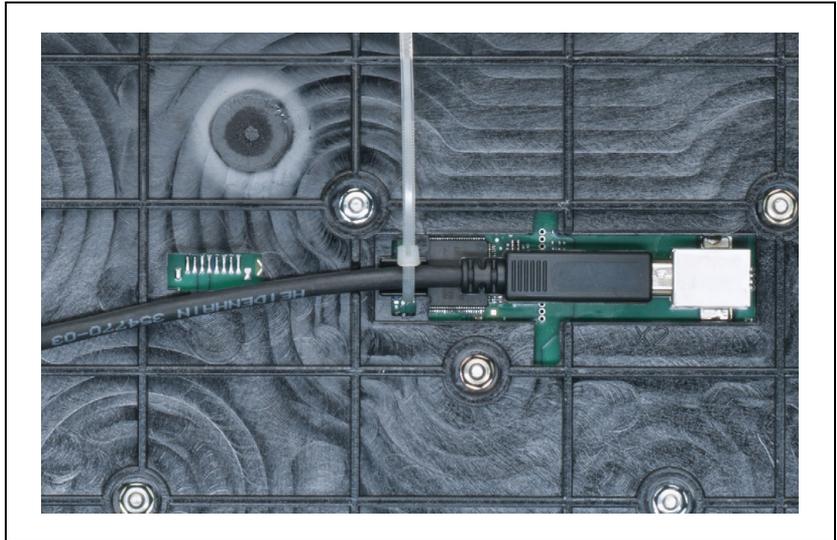
The override potentiometers of a TE 6xx can be configured in machine parameter MP7620 bit 5 and bit 9. If bit 5 is set in MP7620, the potentiometers are evaluated as feed-rate and rapid-traverse override potentiometers. If an HSCI TE keyboard unit with three override potentiometers (spindle, rapid traverse and feed rate) is used, bit 9 must be set in MP7620. If bit 5 and bit 9 are both set at the same time, an error message is triggered due to invalid configuration.

For the evaluation of the potentiometers, the following distinctions are made between HSCI and non-HSCI hardware:

- TE for systems without HSCI
Configuration with feed-rate and rapid-traverse overrides (MP7620, bit 5). The spindle-speed override is used as rapid-traverse override. The NC keys are used to realize the spindle speed override. If hardware for systems without HSCI is used, bit 9 is ignored.
- TE for systems without HSCI
Configuration for three potentiometers with feed-rate, spindle-speed and rapid-traverse overrides (MP7620, bit 5). This configuration has no effect. The potentiometers are evaluated, as usual, as feed-rate and spindle-speed overrides. Bit 9 is ignored.

- TE for systems with HSCI
Configuration with feed-rate and rapid-traverse overrides (MP7620, bit 5). The spindle-speed override input **S** (X52) on the rear of the TE is not evaluated. The rapid-traverse override must be connected to the provided connector **E** (X61, X62). There are no NC keys for spindle-speed override. Machine operating panel keys must be used instead, and the function must be realized in the PLC program (the PLC program must transmit the information to W764).
The +, – and 100% keys for spindle-speed override can be snapped on to an HSCI machine operating panel. Contrary to keyboards for systems without HSCI, these keys are only available as PLC inputs, and not as NC keys. They are not evaluated automatically by the NC software. The PLC program must evaluate the keys and transmit the calculated value to the NC in W764 (spindle-speed override). The NC software limits this value depending on MP3310.0 (max. spindle-speed override value) and MP3310.1 (minimum spindle-speed override value).
- TE with three potentiometers for systems with HSCI
Configuration for three potentiometers with feed-rate, spindle-speed and rapid-traverse overrides (MP7620, bit 9): Feed-rate, spindle-speed and rapid-traverse overrides are evaluated.

The TE keyboard units are supplied with power via the USB port (USB type B). USB cables that are connected to the USB ports of the TE keyboard units must be fixed with the corresponding cable tie to protect the USB connectors from mechanical load (strain relief).



Possible configurations for the potentiometers of a TE 7xx

The override potentiometers of a TE 6xx can be configured in machine parameter MP7620 bit 5 and bit 9. If bit 5 is set in MP7620, the potentiometers are evaluated as feed-rate and rapid-traverse override potentiometers. If an HSCI TE keyboard unit with three override potentiometers (spindle, rapid traverse and feed rate) is used, bit 9 must be set in MP7620. If bit 5 and bit 9 are both set at the same time, an error message is triggered due to invalid configuration.

If you configure a rapid-traverse override (bit 5), the characteristic curve is nonlinear (range 0 to 10000/15000), regardless of bit 3. Then the nonlinear characteristic curve applies to all overrides.

For the evaluation of the potentiometers, the following distinctions are made between HSCI and non-HSCI hardware:

- TE for systems without HSCI
Configuration with feed-rate and rapid-traverse overrides (MP7620, bit 5). The spindle-speed override is used as rapid-traverse override. The NC keys are used to realize the spindle speed override. If hardware for systems without HSCI is used, bit 9 is ignored.
- TE for systems without HSCI
Configuration for three potentiometers with feed-rate, spindle-speed and rapid-traverse overrides (MP7620, bit 5). This configuration has no effect. The potentiometers are evaluated, as usual, as feed-rate and spindle-speed overrides. Bit 9 is ignored.
- TE for systems with HSCI
Configuration with feed-rate and rapid-traverse overrides (MP7620, bit 5). The spindle-speed override input **S** (X52) on the rear of the TE is not evaluated. The rapid-traverse override must be connected to the provided connector **E** (X61, X62). There are no NC keys for spindle-speed override. Machine operating panel keys must be used instead, and the function must be realized in the PLC program (the PLC program must transmit the information to W764).
The +, – and 100% keys for spindle-speed override can be snapped on to an HSCI machine operating panel. Contrary to keyboards for systems without HSCI, these keys are only available as PLC inputs, and not as NC keys. They are not evaluated automatically by the NC software. The PLC program must evaluate the keys and transmit the calculated value to the NC in W764 (spindle-speed override). The NC software limits this value depending on MP3310.0 (max. spindle-speed override value) and MP3310.1 (minimum spindle-speed override value).
- TE with three potentiometers for systems with HSCI
Configuration for three potentiometers with feed-rate, spindle-speed and rapid-traverse overrides (MP7620, bit 9): Feed-rate, spindle-speed and rapid-traverse overrides are evaluated.

MP_7620 **Bit#9 – HSCI keyboards with three potentiometers**

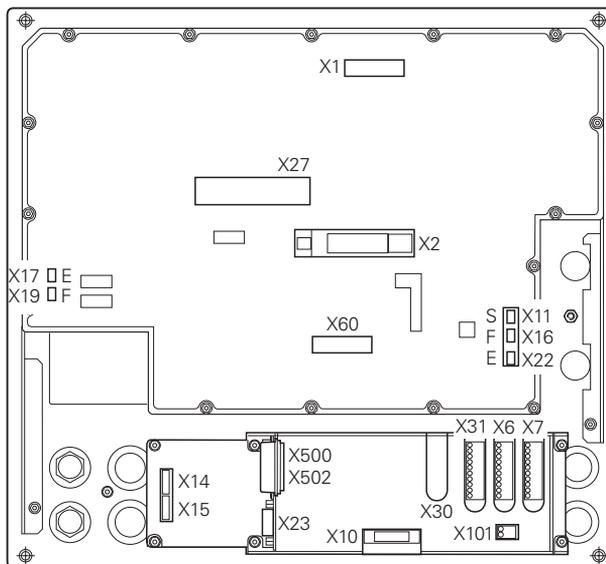
Input: 0: Keyboard with two potentiometers
 1: HSCI keyboard with three potentiometers



3.36 TE 735 (FS) keyboard unit

TE 735(FS), NC operating panel in stainless steel design, for 15-inch, with touchpad and integrated machine operating panel

TE 735(FS) pin layout

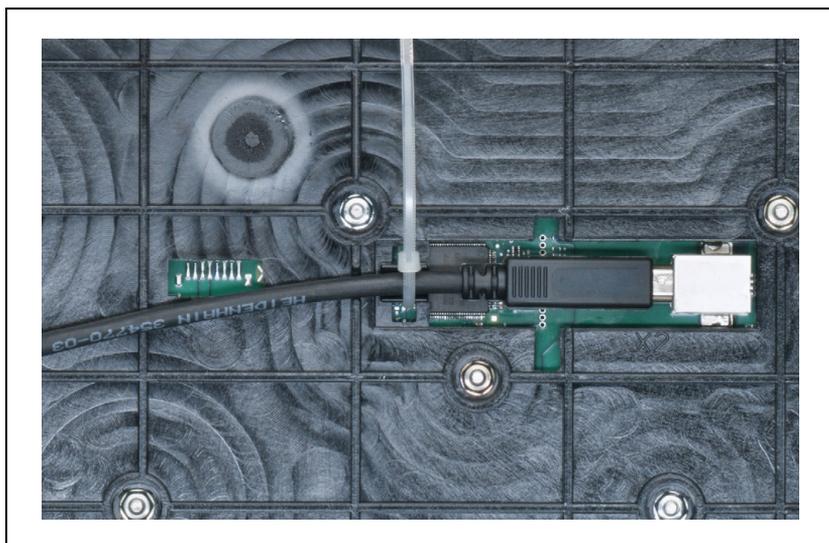


Connector	Function
X2: USB Type B	USB connection with BF 750. The TE is connected with the USB hub of the BF 750. The maximum permissible cable length for this USB connection is 3 m. The USB hub of the TE only supports USB 1.1.
X1	Screen soft keys of BF 750 X3 via ribbon cable to keyboard unit X1
X60	Potentiometer of keyboard unit via ribbon cable (enclosed) to machine operating panel X10
S, X11	Connection for spindle-speed override potentiometer (pre-assigned by HEIDENHAIN)
F, X19	Connection for feed-rate override potentiometer. There are two connections—preassigned by HEIDENHAIN depending on the arrangement of the feed-rate potentiometer.
E, X17	Connection for rapid-traverse override potentiometer. There are two connections—preassigned by HEIDENHAIN depending on the arrangement of the rapid-traverse potentiometer.
F, X16	Connection for feed-rate potentiometer, alternative to X19
E, X22	Connection for rapid-traverse override potentiometer, alternative to X17
X27	Not used
⊕	Protective ground (M5)

MB connector	Function, identical to MB 720	Page
X14	Connection for the NC start key (pre-assigned by HEIDENHAIN)	330
X15	Connection for the NC stop key (pre-assigned by HEIDENHAIN)	330
X18	EMERGENCY STOP	332
X500	HSCI output	208
X502	HSCI input	208
X23	Handwheel connection	347
X10	Potentiometer of X60 keyboard unit via ribbon cable (enclosed)	338
X30	Connection for handwheel: permissive buttons and emergency stop (only with functional safety (FS))	339
X31	NC start / NC stop, permissive buttons	335
X6	PLC inputs	333
X7	PLC outputs	332
X101	24 V DC NC power supply	339
⊕	Protective ground (M5)	

For the pin layouts of the MB, see page 330.

The TE keyboard units are supplied with power via the USB port (USB type B). USB cables that are connected to the USB ports of the TE keyboard units must be fixed with the corresponding cable tie to protect the USB connectors from mechanical load (strain relief).



Possible configurations for the potentiometers of a TE 7xx

The override potentiometers of a TE 6xx can be configured in machine parameter MP7620 bit 5 and bit 9. If bit 5 is set in MP7620, the potentiometers are evaluated as feed-rate and rapid-traverse override potentiometers. If an HSCI TE keyboard unit with three override potentiometers (spindle, rapid traverse and feed rate) is used, bit 9 must be set in MP7620. If bit 5 and bit 9 are both set at the same time, an error message is triggered due to invalid configuration.

If you configure a rapid-traverse override (bit 5), the characteristic curve is nonlinear (range 0 to 10000/15000), regardless of bit 3. Then the nonlinear characteristic curve applies to all overrides.

For the evaluation of the potentiometers, the following distinctions are made between HSCI and non-HSCI hardware:

- TE for systems without HSCI
Configuration with feed-rate and rapid-traverse overrides (MP7620, bit 5). The spindle-speed override is used as rapid-traverse override. The NC keys are used to realize the spindle speed override. If hardware for systems without HSCI is used, bit 9 is ignored.
- TE for systems without HSCI
Configuration for three potentiometers with feed-rate, spindle-speed and rapid-traverse overrides (MP7620, bit 5). This configuration has no effect. The potentiometers are evaluated, as usual, as feed-rate and spindle-speed overrides. Bit 9 is ignored.
- TE for systems with HSCI
Configuration with feed-rate and rapid-traverse overrides (MP7620, bit 5). The spindle-speed override input **S** (X52) on the rear of the TE is not evaluated. The rapid-traverse override must be connected to the provided connector **E** (X61, X62). There are no NC keys for spindle-speed override. Machine operating panel keys must be used instead, and the function must be realized in the PLC program (the PLC program must transmit the information to W764).
The +, – and 100% keys for spindle-speed override can be snapped on to an HSCI machine operating panel. Contrary to keyboards for systems without HSCI, these keys are only available as PLC inputs, and not as NC keys. They are not evaluated automatically by the NC software. The PLC program must evaluate the keys and transmit the calculated value to the NC in W764 (spindle-speed override). The NC software limits this value depending on MP3310.0 (max. spindle-speed override value) and MP3310.1 (minimum spindle-speed override value).
- TE with three potentiometers for systems with HSCI
Configuration for three potentiometers with feed-rate, spindle-speed and rapid-traverse overrides (MP7620, bit 9): Feed-rate, spindle-speed and rapid-traverse overrides are evaluated.

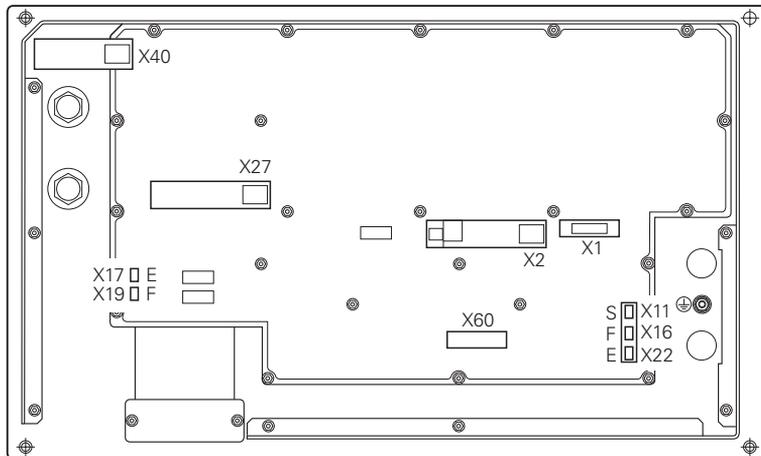
MP_7620 Bit#9 – HSCI keyboards with three potentiometers

Input: 0: Keyboard with two potentiometers
 1: HSCI keyboard with three potentiometers

3.37 TE 740 keyboard unit

TE 740, NC operating panel in stainless steel design, for 19-inch, with touchpad

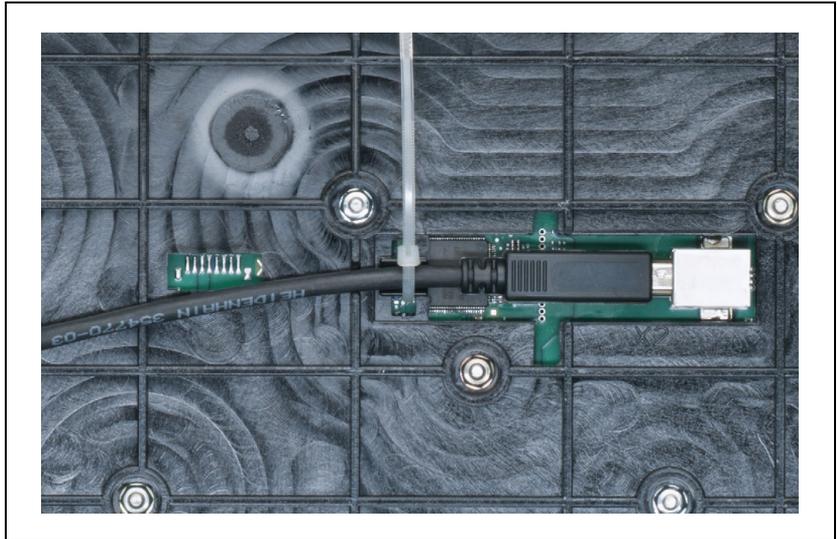
TE 740 pin layout



Connector	Function
X2: USB Type B	USB connection with BF 760. The TE 740 is connected with the USB hub of the BF 760. The maximum permissible cable length for this USB connection is 3 m. The USB hub of the TE 740 only supports USB 1.1.
X40: USB Type B	USB connection for execution of the USB connection from the BF 760 to the USB connector in the front of the TE
X1	Screen soft keys of BF 760 X3 via ribbon cable to keyboard unit X1
X60	Potentiometer of keyboard unit via ribbon cable to machine operating panel X10
S, X11	Connection for spindle-speed override potentiometer (pre-assigned by HEIDENHAIN)
F, X19	Connection for feed-rate override potentiometer. There are two connections—preassigned by HEIDENHAIN depending on the arrangement of the feed-rate potentiometer.
E, X17	Connection for rapid-traverse override potentiometer. There are two connections—preassigned by HEIDENHAIN depending on the arrangement of the rapid-traverse potentiometer.
F, X16	Connection for feed-rate potentiometer, alternative to X19
E, X22	Connection for rapid-traverse override potentiometer, alternative to X17
X27	Not used
⊕	Protective ground (M5)



The TE keyboard units are supplied with power via the USB port (USB type B). USB cables that are connected to the USB ports of the TE keyboard units must be fixed with the corresponding cable tie to protect the USB connectors from mechanical load (strain relief).



Possible configurations for the potentiometers of a TE 7xx

The override potentiometers of a TE 6xx can be configured in machine parameter MP7620 bit 5 and bit 9. If bit 5 is set in MP7620, the potentiometers are evaluated as feed-rate and rapid-traverse override potentiometers. If an HSCI TE keyboard unit with three override potentiometers (spindle, rapid traverse and feed rate) is used, bit 9 must be set in MP7620. If bit 5 and bit 9 are both set at the same time, an error message is triggered due to invalid configuration.

If you configure a rapid-traverse override (bit 5), the characteristic curve is nonlinear (range 0 to 10000/15000), regardless of bit 3. Then the nonlinear characteristic curve applies to all overrides.

For the evaluation of the potentiometers, the following distinctions are made between HSCI and non-HSCI hardware:

- TE for systems without HSCI
Configuration with feed-rate and rapid-traverse overrides (MP7620, bit 5). The spindle-speed override is used as rapid-traverse override. The NC keys are used to realize the spindle speed override. If hardware for systems without HSCI is used, bit 9 is ignored.
- TE for systems without HSCI
Configuration for three potentiometers with feed-rate, spindle-speed and rapid-traverse overrides (MP7620, bit 5). This configuration has no effect. The potentiometers are evaluated, as usual, as feed-rate and spindle-speed overrides. Bit 9 is ignored.
- TE for systems with HSCI
Configuration with feed-rate and rapid-traverse overrides (MP7620, bit 5). The spindle-speed override input **S** (X52) on the rear of the TE is not evaluated. The rapid-traverse override must be connected to the provided connector **E** (X61, X62). There are no NC keys for spindle-speed override. Machine operating panel keys must be used instead, and the function must be realized in the PLC program (the PLC program must transmit the information to W764).
The +, – and 100% keys for spindle-speed override can be snapped on to an HSCI machine operating panel. Contrary to keyboards for systems without HSCI, these keys are only available as PLC inputs, and not as NC keys. They are not evaluated automatically by the NC software. The PLC program must evaluate the keys and transmit the calculated value to the NC in W764 (spindle-speed override). The NC software limits this value depending on MP3310.0 (max. spindle-speed override value) and MP3310.1 (minimum spindle-speed override value).
- TE with three potentiometers for systems with HSCI
Configuration for three potentiometers with feed-rate, spindle-speed and rapid-traverse overrides (MP7620, bit 9): Feed-rate, spindle-speed and rapid-traverse overrides are evaluated.

MP_7620 **Bit#9 – HSCI keyboards with three potentiometers**

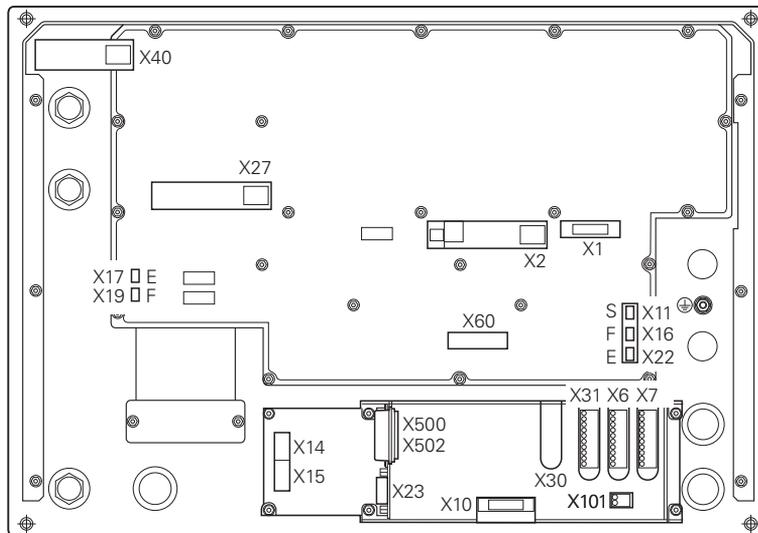
Input: 0: Keyboard with two potentiometers
 1: HSCI keyboard with three potentiometers



3.38 TE 745 (FS) keyboard unit

TE 745 (FS), NC operating panel in stainless-steel design (corresponds to TE 740), machine operating panel is integrated (same as MB 720), for 19 inches, with touchpad

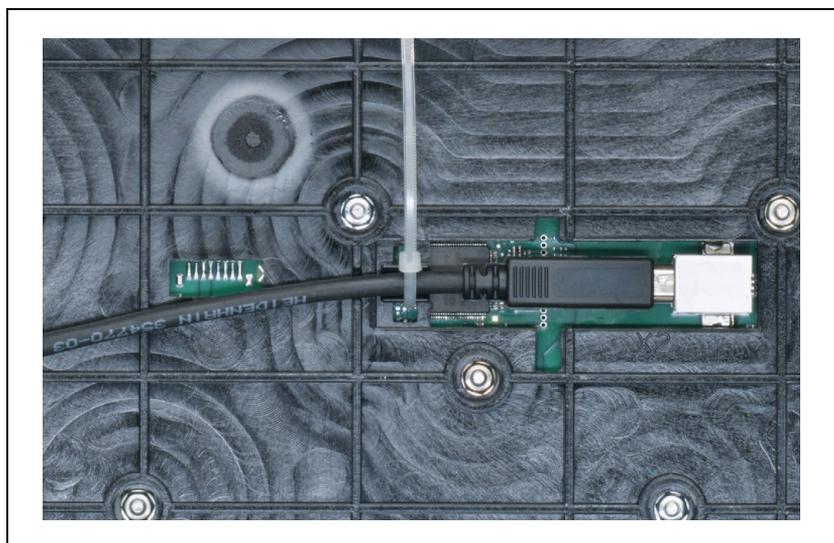
TE 745 (FS) pin layout



TE connector	Function
X2: USB Type B	USB connection with BF 760. The TE 745 is connected with the USB hub of the BF 760. The maximum permissible cable length for this USB connection is 3 m. The USB hub of the TE 745 only supports USB 1.1.
X40: USB Type B	USB connection for execution of the USB connection from the BF 760 to the USB connector in the front of the TE
X1	Screen soft keys of BF 760 X3 via ribbon cable to keyboard unit X1
X60	Potentiometer of keyboard unit via ribbon cable (enclosed) to machine operating panel X10
S, X11	Connection for spindle-speed override potentiometer (pre-assigned by HEIDENHAIN)
F, X19	Connection for feed-rate override potentiometer. There are two connections—preassigned by HEIDENHAIN depending on the arrangement of the feed-rate potentiometer.
E, X17	Connection for rapid-traverse override potentiometer. There are two connections—preassigned by HEIDENHAIN depending on the arrangement of the rapid-traverse potentiometer.
F, X16	Connection for feed-rate potentiometer, alternative to X19
E, X22	Connection for rapid-traverse override potentiometer, alternative to X17
X27	Not used

MB connector	Function, identical to MB 720	Page
X14	Connection for the NC start key (pre-assigned by HEIDENHAIN)	330
X15	Connection for the NC stop key (pre-assigned by HEIDENHAIN)	330
X18	EMERGENCY STOP	332
X500	HSCI output	208
X502	HSCI input	208
X23	Handwheel connection	347
X10	Potentiometer of X60 keyboard unit via ribbon cable (enclosed)	338
X30	Connection for handwheel: permissive buttons and emergency stop (only with functional safety (FS))	339
X31	NC start / NC stop, permissive buttons	335
X6	PLC inputs	333
X7	PLC outputs	332
X101	24 V DC NC power supply	339
⊕	Protective ground (M5)	

The TE keyboard units are supplied with power via the USB port (USB type B). USB cables that are connected to the USB ports of the TE keyboard units must be fixed with the corresponding cable tie to protect the USB connectors from mechanical load (strain relief).



Possible configurations for the potentiometers of a TE 7xx

The override potentiometers of a TE 7xx can be configured in machine parameter MP7620 bit 5 and bit 9. If bit 5 is set in MP7620, the potentiometers are evaluated as feed-rate and rapid-traverse override potentiometers. If an HSCI TE keyboard unit with three override potentiometers (spindle, rapid traverse and feed rate) is used, bit 9 must be set in MP7620. If bit 5 and bit 9 are both set at the same time, an error message is triggered due to invalid configuration.

For the evaluation of the potentiometers, the following distinctions are made between HSCI and non-HSCI hardware:

- TE for systems without HSCI
Configuration with feed-rate and rapid-traverse overrides (MP7620, bit 5). The spindle-speed override is used as rapid-traverse override. The NC keys are used to realize the spindle speed override. If hardware for systems without HSCI is used, bit 9 is ignored.
- TE for systems without HSCI
Configuration for three potentiometers with feed-rate, spindle-speed and rapid-traverse overrides (MP7620, bit 5). This configuration has no effect. The potentiometers are evaluated, as usual, as feed-rate and spindle-speed overrides. Bit 9 is ignored.
- TE for systems with HSCI
Configuration with feed-rate and rapid-traverse overrides (MP7620, bit 5). The spindle-speed override input **S** (X52) on the rear of the TE is not evaluated. The rapid-traverse override must be connected to the provided connector **E** (X61, X62). There are no NC keys for spindle-speed override. Machine operating panel keys must be used instead, and the function must be realized in the PLC program (the PLC program must transmit the information to W764).
The +, – and 100% keys for spindle-speed override can be snapped on to an HSCI machine operating panel. Contrary to keyboards for systems without HSCI, these keys are only available as PLC inputs, and not as NC keys. They are not evaluated automatically by the NC software. The PLC program must evaluate the keys and transmit the calculated value to the NC in W764 (spindle-speed override). The NC software limits this value depending on MP3310.0 (max. spindle-speed override value) and MP3310.1 (minimum spindle-speed override value).
- TE with three potentiometers for systems with HSCI
Configuration for three potentiometers with feed-rate, spindle-speed and rapid-traverse overrides (MP7620, bit 9): Feed-rate, spindle-speed and rapid-traverse overrides are evaluated.

Machine operating panel

The integrated machine operating panel of the TE 745 (FS) is suitable for machines with up to six axes. It is connected with the control system over the HSCI interface.

A blue status LED is located over each of the 36 snap-on machine keys. The LEDs are internally permanently connected with PLC outputs and can therefore be freely controlled via the PLC program. The output addresses of the LEDs come from the HDD file of the machine operating panel.



Note

The current HDD files of the HEIDENHAIN operating panel components are a part of the installation package of IOconfig. You can download the current HDD files as a ZIP archive from the HESIS-Web Including Filebase on the Internet (<http://hesis.heidenhain.de>).

For this area you need access rights.

As an alternative, please get in touch with your contact partner at HEIDENHAIN to obtain the current HDD file.

On the underside of the machine operating panel are terminal strips bearing where further PLC outputs and PLC inputs are located. Also, connection X23 for HR handwheels is on the underside of the MB.

It must be ensured that negative pressure cannot develop in the panel in which the machine operating panel is installed. Negative pressure in the panel can lead to increased contamination of the machine operating panel, which must be avoided by enabling pressure compensation, for example via the cable duct between the electrical cabinet and the panel.



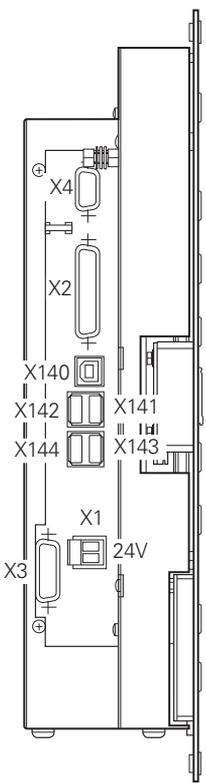
Danger

Please note that the integrated machine operating panel of the TE 745 (FS) is powered by +24 V NC.

For the entire HSCI system, the +24 V NC supply voltage is required to be safely separated voltage. The +24 V NC supply voltage must not, under any circumstances, be connected with the +24 V PLC supply voltage, because this removes the double basic insulation.

3.39 BF 250 15-inch TFT screen

Thanks to its HDL interface, the 15-inch TFT display unit is compatible with all MC 6xxx main computers with HDL output.

BF 250 15-inch TFT screen			
Pin layout	Connector	Function	Page
	X4	Reserved	
	X2	HDL connection	378
	X140	USB input (Type B)	378
	X141 to X144	USB 2.0 output (Type A)	378
	X1	+24 V power supply	377
	X3	Connection for screen soft keys	378

X1: +24 V-power supply for BF-2xx

Connection for the +24 V power supply of the BF 2xx. The screen can be powered by +24 V NC or +24 V PLC through the integrated power supply unit of the BF 2xx.

Connecting terminals	Assignment
+ / terminal 1	+24 V
- / terminal 2	0 V

Power supply: Minimum absolute value: +20.4 V–
Maximum absolute value: +28.8 V–

Power consumption of the BF 250: 50 W

- X2: HDL connection** Port for the HEIDENHAIN display link (HDL) connection for the MC 6xxx connector X249, see page 329.
- X3: Connection of screen soft keys** Connection of the screen soft keys to the TE 6xx keyboard unit connector X1 via ribbon cable.
- X140: USB input** Connection for the integrated USB hub to the MC 6xxx connector X141 or X142. The USB hub of the BF 2xx supports USB 2.0.
- X141 to X146: USB output** BF 250: Five freely available connections for additional USB devices. One of them on the front of the BF 250.
Maximum load of the five USB outputs of the BF 250:
4 x 500 mA, 1 x 100 mA, distributed as desired



3.40 BF 260 19-inch TFT display unit

Thanks to its HDL interface, the 19-inch TFT display unit is compatible with all MC 6xxx main computers with HDL output.

BF 260, 19-inch TFT display unit			
Pin layout	Connector	Function	Page
	X4	Reserved	
	X2	HDL connection	378
	X140	USB input (Type B)	378
	X141 to X146	USB output (Type A)	378
	X1	+24 V power supply	377
	X3	Connection for screen soft keys	378
	X5	Reserved	

X1: +24 V power supply for BF 2xx

Connection for the +24 V power supply of the BF 2xx. The screen can be powered by +24 V NC or +24 V PLC through the integrated power supply unit of the BF 2xx.

Connecting terminals	Assignment
+ / terminal 1	+24 V
- / terminal 2	0 V

Power consumption of the BF 260: 65 W

- X2: HDL connection** Port for the HEIDENHAIN display link (HDL) connection to the MC 6xxx connector X249.
- X3: Connection of screen soft keys** Connection of the screen soft keys to the TE 6xx keyboard unit connector X1 via ribbon cable.
- X140: USB input** Connection for the integrated USB hub to the MC 6xxx connector X141 or X142.
- X141 to X146: USB output** BF 260: Six freely available connections for additional USB devices.
Maximum load capacity of the six USB outputs of the BF 260:
4 x 500 mA, 2 x 100 mA, distributed as desired
- Settings** To be able to use the entire 19-inch screen, you first have to adjust the resolution setting.
You must enter **GRAPHICSRESOLUTION = 1280x1024** in the **OEM.SYS** for this. This setting goes into effect after you have restarted the control.
- Soft keys** The introduction of the 19-inch screen made it possible to offer a second vertical soft-key row. It is on the left edge of the screen. The keystrokes are evaluated and processed by the PLC program or a Python program. The table below shows the key codes of the soft keys in word W274:

Soft key	Key code, left row	Key code, right row
Soft key 1 (top)	0x150	0x160
Soft key 2	0x151	0x161
Soft key 3	0x152	0x162
Soft key 4	0x153	0x163
Soft key 5	0x154	0x164
Soft key 6	0x155	0x165
Soft key 7	0x156	0x166
Soft key 8	0x157	0x167
Soft key 9	0x158	0x168
Soft key 10 (bottom)	0x159	0x169

Also, new parameters and keywords were introduced for the left soft-key row. The left soft-key row is displayed as a pop-up menu.



The table below shows the key codes of the horizontal soft keys in word W274:

Soft key	Key code of horizontal row
Scroll to left	0x19E
Soft key 1 (left)	0x180
Soft key 2	0x181
Soft key 3	0x182
Soft key 4	0x183
Soft key 5	0x184
Soft key 6	0x185
Soft key 7	0x186
Soft key 8	0x187
Soft key 9	0x188
Soft key 10 (right)	0x189
Scroll to right	0x19D

Permissible parameters for SKMENU

Entry in *.SPJ	In connection with	Meaning
VRROOT	SKMENU	The menu for the right vertical soft-key row is defined in the header of the main menu.
HRROOT	SKMENU	The menu for the horizontal soft-key row is defined in the header of the main menu.
LROOT	SKMENU	The menu for the left vertical soft-key row is defined in the header of the main menu.
HRMENU	SKMENU	Optional: This soft-key menu is a handwheel menu
HRRROOT	SKMENU	Optional: This soft-key menu is the start menu of the handwheel
IHROOT	SKMENU	Optional: This soft-key menu is the horizontal start menu of an ITC terminal.
IVROOT	SKMENU	Optional: This soft-key menu is the vertical start menu of an ITC terminal.
EMODE	SKMENU	The menu for the programming operating modes is defined in the header of the main menu.
MMODE	SKMENU	The menu for the machine operating modes is defined in the header of the main menu.
OMODE	SKMENU	Optional: This (Python) soft-key menu is only visible in the OEM operating mode (third screen).
IDENT:<id>	SKMENU	Optional: Identifier for access from Python processes.

Keywords in the soft-key menus

Entry in *.SPJ	Meaning
Keywords for menu definition:	
ROWDEF	Specifies the position of a soft-key menu that does not span the entire soft-key row.
SKPREVDEF	Defines the key that triggers the page left/up function.
SKNEXTDEF	Defines the key that triggers the page right/down function.

3.41 Handwheel input

The following handwheels can be used with the iTNC 530 HSCI:

- HR 130 panel-mounted handwheel
- Up to three HR 150 panel-mounted handwheels via HRA 110 handwheel adapter
- HR 410 (FS) portable handwheel
- HR 520 (FS) portable handwheel with display
- HR 550 wireless handwheel with display and wireless transmission

The handwheels are connected to the iTNC 530 HSCI via X23 on the MB 620 machine operating panel.

3.41.1 HR 4xx (FS) or HR 5xx (FS) portable handwheel

The HR 4xx and HR 5xx are portable electronic handwheels.

For the assignment of the keys of the HR 410 to the PLC inputs and outputs, see "HR 410 portable handwheel" on page 1510.

Pin layout

Pin layout for the various extension cables, adapter cables, connecting cables, and the handwheel:

Extension cable, ID 281429-xx			Adapter cable, ID 296466-xx			For connecting cable ID, see "Introduction" chapter.			HR 4xx HR 5xx	
D-sub connector (male) 9-pin		D-sub connector (female) 9-pin	D-sub connector (male) 9-pin		Mounted coupling (female) (5+7)-pin	Cnnctr. (male) (5+7)-pin		Cnnctr. (female) (5+7)-pin	Cnnctr. (male) (5+7)-pin	
Hsg.	Shield	Housing	Hsg.	Shield	Housing	Hsg.	Shield	Housing	Hsg.	Shield
2	White	2	2	White	E	E	White	E	E	
4	Brown	4	4	Brown	D	D	Brown	D	D	
6	Yellow	6	6	Yellow	B	B	Yellow	B	B	
7	Gray	7	7	Gray	A	A	Gray	A	A	
8	Green	8	8	Green	C	C	Green	C	C	
				WH/BK	6	6	BK	6	6	
				YL/BK	7	7	RD/BL	7	7	
				WH/RD	5	5	Red	5	5	
				WH/BL	4	4	Blue	4	4	
				WH/GN	2	2	WH/GN	2	2	
				WH/YL	3	3	BN/GN	3	3	
				WH/GN	1	1	GY/PK	1	1	
				WH/BN	X1/3	Contacts 1 + 2 (supply)				
				WH/YL	X1/1	Contact 2 (left) permissive button				
				WH/GN	X1/2	Contact 1 (right) Permissive button				
				WH/BL	X2/1	Contact 1/Terminal 1 (EMERGENCY STOP)				
				WH/RD	X2/2	Contact 1/Terminal 2 (EMERGENCY STOP)				
				YL/BK	X2/4	Contact 2/Terminal 1 (EMERGENCY STOP)				
				WH/BK	X2/3	Contact 2				



Note

The interfaces comply with the requirements of EN 61800-5-1 for "protective extra-low voltage (PELV)."



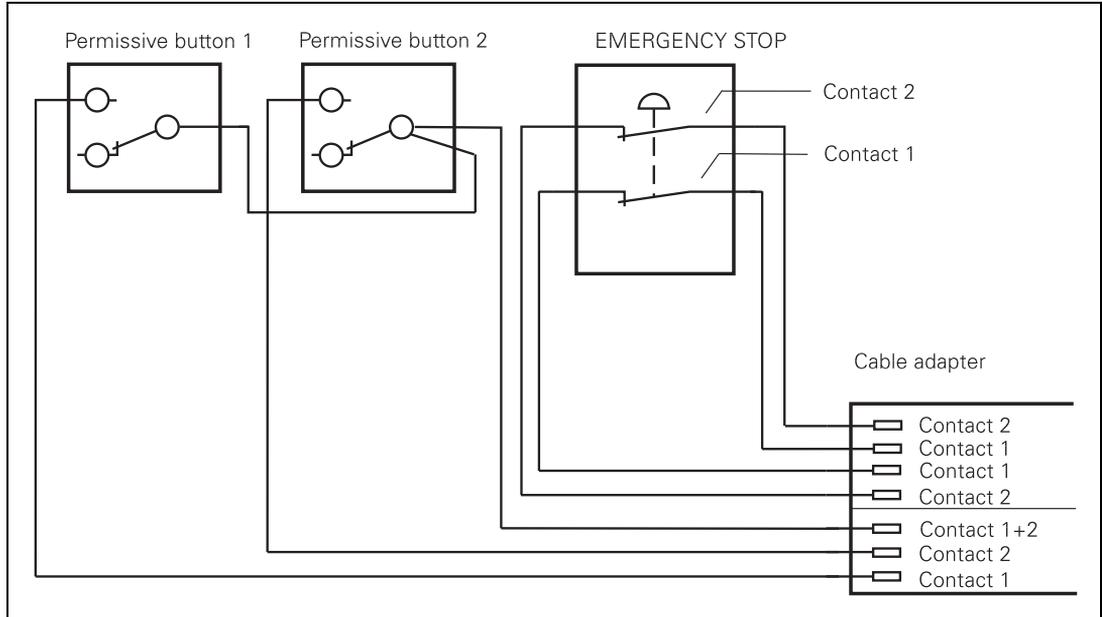
Danger

Only units that comply with the requirements of EN 61800-5-1 for "protective extra-low voltage (PELV)" may be connected.

The adapter includes plug-in terminal strips for the contacts of the emergency stop button and permissive button (max. load 24 V-, 1.2 A).

The plug-in terminal strips are supplied together with the adapter cable. If you have an immediate need for these terminal strips, they can be ordered in advance. See the "Additional components" table below.

Internal wiring of the contacts for the emergency stop button and permissive button:



Note

HEIDENHAIN would like to point out that the permissive buttons on the HR 520, HR 420 and HR 410 handwheels are not cross-circuit proof. If your machine requires a handwheel with cross-circuit proof permissive buttons, you must use a HR 410FS or HR 520FS handwheel.

You must calculate the probability of failure according to EN 13849-1 for the safety circuit with permissive-button information. This value is used to determine a maximum testing time for this safety circuit, which must not be exceeded. The control must be turned off and on again within this testing time.

Specifications of the switch contacts in permissive buttons and emergency stop button

- Rated voltage: +24 V DC
- Maximum current: 1.2 A

Additional components	ID
Dummy plug for emergency stop circuit	271958-03
Connecting cable	
Spiral cable	312879-01
Normal cable	296467-xx
Metal armor tubing	296687-xx
Plug-in terminal strips for advance ordering	
3-pin terminal block	266364-06
4-pin terminal block	266364-12

Pin layout of dummy plug for standard handwheels without FS:

Dummy plug without FS ID 271958-03	
Pin	Assignment
1	Not assigned
2	Not assigned
3	Not assigned
4	Bridge on pin 4 (emergency stop)
5	Bridge on pin 5 (emergency stop)
6	Bridge on pin 7 (emergency stop)
7	Bridge on pin 6 (emergency stop)
8	Not assigned
A	Not assigned
B	Not assigned
C	Not assigned
D	Not assigned



3.41.2 HR 520FS, HR 410FS safety-oriented electronic handwheels

The HR 520FS and HR 410FS handwheels that are now available in addition to the HR 550FS (wireless handwheel system) are devices that, due to their cross-circuit-safety, can also be integrated in safety-related applications according to EN 13849.

The handwheels with FS are plug-compatible with the respective handwheels without FS, thanks to the M23 connecting element, and they have the same scope of functions. As a result of the new logic behind the permissive buttons, only one of the two buttons need to be pressed during permissive mode.



Note

- Due to the changed logic of the permissive buttons, do not replace the HR 520FS and HR 410FS with the respective variant without FS.
- Please note that even if servicing is required, an HR 520FS may not be replaced by an HR 520. This also applies to replacing an HR 410FS with an HR 410, and vice versa.

Because of the change in the principle behind the permissive buttons, we generally recommend using the safety-oriented handwheels with FS, due to the cross-circuit-safety of their permissive function.

Only for the HR 550FS wireless handwheel system with HRA 551FS can the behavior of the permissive buttons be defined by MP7645.3.

Also, a new dummy plug is being introduced for FS handwheels in case servicing is necessary. As before, this dummy plug bridges the emergency stop, and is adapted to the changed logic for the permissive buttons. The new dummy plug for FS handwheels is identified by the designation "Adapterstecker FS" and the ID 271958-05 on the ID label.



Note

- Please note that the dummy plug for FS handwheels may not be exchanged for the respective variant without FS.
- For reasons of technical safety, only dummy plugs for FS handwheels (ID 271958-05) may be used on systems that have an FS handwheel.
- For reasons of technical safety, only dummy plugs for standard handwheels (ID 271958-03) may be used on systems that have a standard handwheel (without FS).

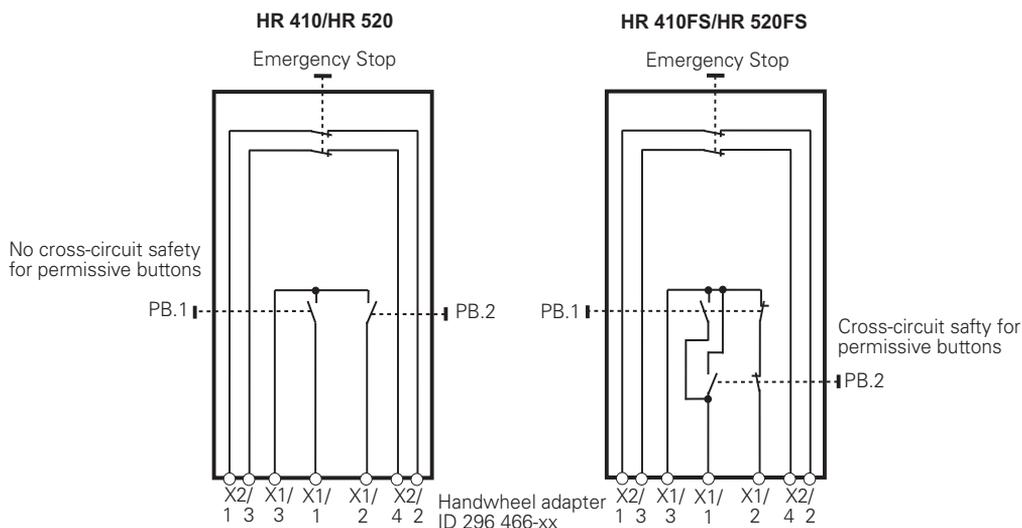
Differences between the HR 520 and HR 410 handwheels

Due to the modified internal wiring of the permissive buttons, a cross-circuit-safe signal for evaluation through the safety circuit is now available for the HR 520 FS and the HR 410 FS. Cross-circuit-safety is necessary in order to prevent a line short-circuit (e.g. defective handwheel cable) from leading to an unintentional permissive situation on a machine tool.

With the current principle behind the permissive buttons (handwheels without FS), in normal operation a faulty switching contact (permanently at "1") cannot be distinguished from a cross-circuit.

With functional safety, only a specific retrograde switching signal of the permissive buttons is used for enabling (permission), and a cross-circuit or a defective permissive button can already be detected during the idle state.

Another advantage of the new FS handwheels is that only one of the two permissive buttons (PB) must be pressed during permissive mode. This is realized through an additional switching contact per permissive button (see figure). Each of the two permissive buttons triggers a normally closed contact and a normally open contact. The two contacts are not positively driven, meaning that if one of the contacts gets jammed, it will not continuously and erroneously grant permission.



As a result of the cross-circuit-safety, the information from the permissive buttons of handwheels with FS must be evaluated differently in the safety circuit and in the PLC program:

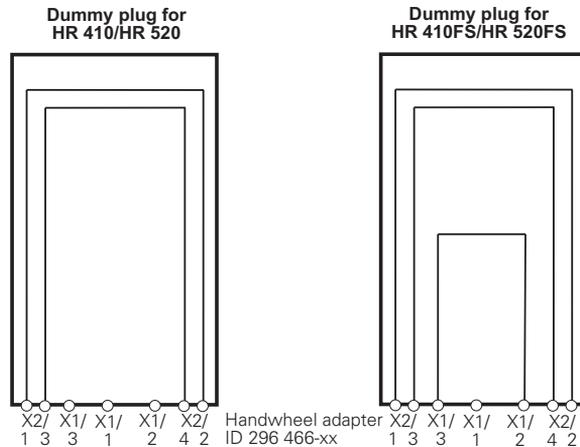
Handwheel without FS				Handwheel with FS		
X1/1	X1/2	Permission		X1/1	X1/2	Permission
0	0	No		0	0	No ^a
0	1	No		0	1	No
1	0	No		1	0	Yes
1	1	Yes		1	1	No ^a

- a. Invalid state: Error (broken wire or cross circuit). This state should be interrogated in the PLC program, and a corresponding error message should be displayed.

The safety circuit (e.g. safety contactor combination) and the PLC program must evaluate the new logic behind the permissive buttons. Only the state X1/1 = 1 and X1/2 = 0 at the handwheel adapter (ID 296466-xx) is valid for permission.

Dummy plug

The dummy plug is wired as follows:



Dummy plug without FS ID 271958-03				Dummy plug for FS ID 271958-05		
X1/1	X1/2	Permission		X1/1	X1/2	Permission
0	0	No		0	1	No

This results in a unique state that must be achieved after the machine has been switched on and no permissive button is pressed. At the handwheel adapter (ID 296466-xx) this state is $X1/1 = 0$, $X1/2 = 0$ for standard handwheels and $X1/1 = 0$, $X1/2 = 1$ for FS handwheels. The same state must also be achieved if the corresponding dummy plug is used. The PLC program should check this unique state when the machine is switched on, to make sure that the correct handwheel or dummy plug for the respective machine configuration is connected. If the expected state is not achieved, an error message is to be triggered and the machine operator is to be prompted to attach the correct handwheel or dummy plug. Using an incorrect handwheel or dummy plug leads to a PLC input (one channel) always carrying the information that a permissive button is pressed.

Pin layout of dummy plug for FS:

Dummy plug for FS ID 271958-05	
Pin	Assignment
1	Bridge on pin 2 (permissive button)
2	Bridge on pin 1 (permissive button)
3	Not assigned
4	Bridge on pin 4 (emergency stop)
5	Bridge on pin 5 (emergency stop)
6	Bridge on pin 7 (emergency stop)
7	Bridge on pin 6 (emergency stop)
8	Not assigned
A	Not assigned
B	Not assigned
C	Not assigned
D	Not assigned

General notes on handling and operation

Please note the following information when using handwheels and dummy plugs, and also give this information to your customers:



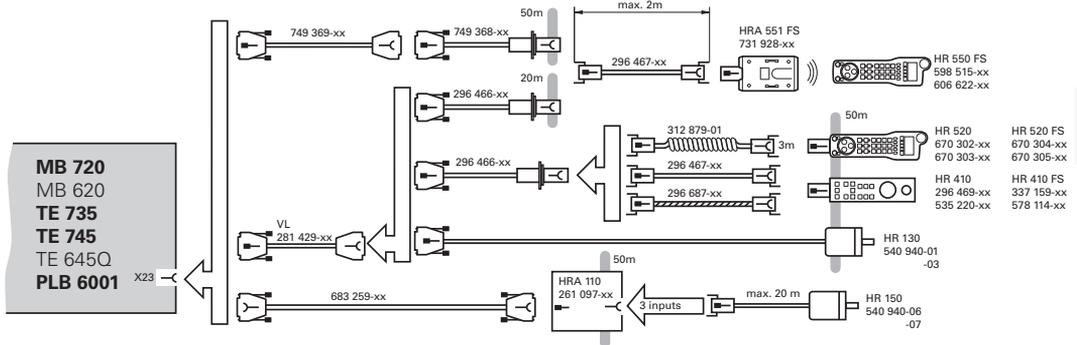
Note

- Only qualified personnel may connect and exchange handwheels and dummy plugs.
- It is essential that operative actions (such as pressing the emergency stop) on handwheels that are not connected are not possible. Handwheels and dummy plugs that are not connected to the machine, and are therefore without function, must be removed from the machine. Handwheels and dummy plugs that are not being used must clearly be stored away from the machine, such as in a locked cabinet.
- Only FS handwheels and dummy plugs for FS handwheels (ID 271958-05) may be used on systems that are prepared for FS handwheels.
- Only standard handwheels and dummy plugs for standard handwheels (ID 271958-03) may be used on systems that are prepared for standard handwheels (without FS).



Connection

Handwheels with FS are connected with the current cables of the HR 4xx and HR 5xx series of HEIDENHAIN handwheels:



Pin layout for the various extension cables, adapter cables, connecting cables, and the handwheel:

Extension cable, ID 281429-xx		Adapter cable, ID 296466-xx				For connecting cable ID, see "Introduction" chapter.		
D-sub connector (male) 9-pin		D-sub connector (female) 9-pin	D-sub connector (male) 9-pin		Mounted coupling (female) (5+7)-pin	Cnnctr. (male) (5+7)-pin		Cnnctr. (female) (5+7)-pin
Hsg.	Shield	Housing	Hsg.	Shield	Housing	Housing	Shield	Housing
2	White	2	2	White	E	E	White	E
4	Brown	4	4	Brown	D	D	Brown	D
6	Yellow	6	6	Yellow	B	B	Yellow	B
7	Gray	7	7	Gray	A	A	Gray	A
8	Green	8	8	Green	C	C	Green	C
					6	6	BK	6
					7	7	RD/BL	7
					5	5	Red	5
					4	4	Blue	4
					2	2	WH/GN	2
					3	3	BN/GN	3
					1	1	GY/PK	1
					WH/BN	X1/3	Contacts A + B (power supply)	
					WH/YL	X1/1	Contact B (permissive button)	
					WH/GN	X1/2	Contact A (permissive button)	
					WH/BL	X2/1	Contact A/Terminal 1 (EMERGENCY STOP)	
					WH/RD	X2/2	Contact A/Terminal 2 (EMERGENCY STOP)	
					YL/BK	X2/4	Contact B/Terminal 1 (EMERGENCY STOP)	
					WH/BK	X2/3	Contact B/Terminal 2 (EMERGENCY STOP)	



Note

The interfaces comply with the requirements of EN 61800-5-1 for "protective extra-low voltage (PELV)."



Danger

Only units that comply with the requirements of EN 61800-5-1 for "protective extra-low voltage (PELV)" may be connected.



3.41.3 HR 550 handwheel with wireless transmission

A wireless handwheel system consists of the HR 550FS mobile part and the HRA 551FS handwheel adapter. The HR 550FS has a rechargeable battery for mobile operation on the machine. During operation it continuously communicates via radio with the HRA 551FS handwheel adapter, which communicates directly with the control via a serial cable connection.

The HRA 551FS handwheel adapter, which functions both as receiving and charging station for the mobile part, features an M23 connector system for connection to the control via a standard handwheel adapter. If the wireless handwheel system is damaged, the dummy plug for the emergency stop circuit (ID 271958-03) can be used so that the operational readiness of the machine can be restored quickly.

The wireless handwheel system can be used in many different places thanks to the 2.4 GHz ISM radio band, which is freely accessible almost all over the world. The HR 550FS complies with the internationally accepted standards. For radio communication, the HEIDENHAIN HR 550FS uses 2.4 GHz ZigBee transmission technology with a special HEIDENHAIN communications protocol.

The HR 550FS wireless handwheel must always be used together with the HRA 551FS handwheel adapter. The handwheel and the handwheel adapter communicate with each other via radio transmission. The handwheel adapter is connected to X23 of the control system via the connecting cable.

When the handwheel is located in the HRA, contacts on the HRA and on the rear of the HR are used to communicate via serial data transmission instead of via radio transmission. The function of the HR 550FS remains the same, regardless of whether it communicates via radio transmission or via the handwheel adapter.

The HRA handwheel adapter features an integrated charger for the handwheel batteries. In addition, the safety-related signals (emergency stop, permissive buttons) are translated into relay contacts through the HRA. They must be used to connect the emergency stop button and the permissive buttons of the handwheel to the control and the associated safety circuits.



Danger

If more than one control with a portable HR 550FS handwheel is located in a factory hall, a room or a working area, the handwheels must be identified by unique color markings. The color marking must be applied to the handwheel **and** the associated machine tool at a clearly visible location. The marking on the handwheel and the machine tool must enable the machine operator to identify unambiguously which handwheel belongs to which machine tool. The color marking must ensure that no confusion occurs. A set of different markers is included with the wireless handwheel.

In addition, the machine operator must be informed that he must ensure/verify every time before using the handwheel that he uses the correct handwheel for the respective machine tool.

If the machine operator moves with the HR too far away from the HRA, a vibration alarm announces that transmission is weak before the machine operator has actually left the radio range. If the machine operator leaves the radio range anyway, or if radio communication is interrupted for other reasons, the HRA relay contacts for emergency stop and for the permissive buttons will open. You must ensure through appropriate wiring that this triggers an emergency stop on the control.

HEIDENHAIN also recommends that you always place the wireless handwheel into the HRA 551FS adapter when you are not using it. This prevents you from confusing the different handwheels. It also ensures that the handwheel battery is recharged and prevents an unexpected emergency stop reaction triggered by an empty battery.



Specifications	HR 550FS with HRA 551FS
Radio range (max. distance between the HR and the HRA)	Max. 20 m in direct line of sight (depending on the ambient conditions)
Ambient temperature during operation	0 °C to + 40 °C
Ambient temperature for charging process of battery	+ 5 °C to + 40 °C (recharging the battery at a temperature of < + 10 °C may reduce the battery life)
Storage temperature of NiMH rechargeable battery	-20 °C to +21 °C with a state of charge of 40 %
Power supply for HRA 551FS	12 V
Display resolution	128 x 64 pixels
Power consumption	Max. 10 W during charging phase
Batteries	Pack with 4 x AA batteries NiMh 4.8 V / 2400 mAh
Complete charging time	Approx. 150 minutes
Life of rechargeable battery	Approx. 16 hours with a utilization of 50 %
Radio frequency	2.4 GHz ISM band
Transmitter power	10 mW
Usable number of channels	16 (ZigBee)
Number of possible HRs with wireless transmission	Only one wireless HR per machine tool
Degree of protection	IP 54 for HR 551FS IP 54 for HR 550FS
Service life of the product	20 years
Max. voltage for relay contacts in HRA 551FS	250 V ~
Max. current for relay contacts in HRA 551FS	3.6 A
Reaction times of emergency-stop and permissive-button safety functions	Time until the reaction of the relay contacts: max. 200 ms; typically < 100 ms
FCC-ID	YJKHR550FS
Weight of HR 550	1 kg
ID for HR 550FS without mech. detent	598515-xx
ID for HR 550FS with mech. detent	606622-xx
ID for HRA 551FS	731928-xx

Radio transmission regulations

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

- This device must not cause harmful interference, and
- this device must accept any interference received, including interference that may cause undesired operation.

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy. If the equipment is not installed and used in accordance with the instructions, the equipment may cause harmful interference to radio communication. There is no guarantee, however, that such interference will not occur in a particular installation.



Note

This wireless handwheel from HEIDENHAIN must be installed and used in strict accordance with the manufacturer's instructions as described in this technical documentation and the User's Manual for your control. Any other installation or use will violate FCC Part 15 regulations. Modifications not expressly approved by Dr. JOHANNES HEIDENHAIN GmbH could void your authority to operate the equipment.

This device must not be co-located or operated in conjunction with any other antenna or transmitter.

Thanks to the wireless connection between the handwheel and the control, the HR 550FS wireless handwheel system and the HRA 551FS handwheel adapter provide maximum ease of use to the machine operator. The wireless communication uses the public, and therefore freely available, 2.4 GHz radio band via a special radio transmission protocol from HEIDENHAIN.

The wireless handwheel system attains a high and certified safety standard because of its dual-channel transmission protocol. The safety functions of the HR 550FS fulfill the requirements of performance level d, category 3, according to EN ISO 13849-1. Single errors do not lead to the loss of the safety functions. The service department must be notified if an error occurs. Only the functions of the permissive buttons and the emergency stop switch are classified as safety functions of the HR 550FS.

The safety functions must not be bypassed or deactivated in some other way.

Modifying or rebuilding the HRA 551FS and HR 550FS is not permitted and may lead to a loss of the safety functions.

The machine operator is responsible for identifying the area in which the wireless handwheel may be used. The wireless handwheel itself does not provide any configuration possibilities or limit values for defining a certain maximum range of use.

Because a handwheel is always connected with the movement of axes, the operator must be able to trigger an emergency stop of the machine by means of the handwheel at any time. This results in a special situation for the emergency stop switch of the HR 550FS wireless handwheel system. An unsafe situation in the radio communication must always trigger a safe reaction from the control. Therefore, an interference in the radio communication also means an emergency stop at the HRA 551FS as reaction.

An unexpected emergency stop of the machine is undesirable and may cause damage to the contour of the workpiece being machined. It is therefore essential to ensure a noise-free radio communication. Reliable operation of the wireless handwheel always requires planning the coexistence of the radio users active in the frequency and radio range concerned. If this cannot be guaranteed because there is no vacant radio channel available, we strongly advise against using the wireless handwheel system. HEIDENHAIN cannot guarantee noise-free radio communication of the HR 550FS wireless handwheel system and the HRA 551FS.

Even if an available radio channel has been found and the proper functioning of the system has been ascertained, it is possible that the availability situation of the 2.4 GHz ISM band will change over a longer period of time. If this situation cannot be corrected by changing the wireless handwheel system or the competing radio system to another channel, it may become necessary to revert to using a handwheel with cable.

The following situations lead to an emergency stop through the wireless handwheel system:

- The HR 550FS wireless handwheel is switched off outside of the HRA 551FS handwheel adapter.
- The radio communication between the HR 550FS wireless handwheel and the HRA 551FS handwheel adapter is interrupted.
- The charge of the handwheel battery becomes too low.



Note

- The power-up test of the HR 550FS must be repeated within no more than 168 hours. The HR 550FS handwheel and the HRA 551FS handwheel adapter must be turned off and on again to conduct the test. It must be ensured by the PLC program that this requirement is met.
- To guarantee the HR's emergency-stop and permissive-button safety functions, the respective contacts must be connected to the emergency stop chain and further safety circuits (if present) or safe inputs of the control.

The handwheel requires no maintenance work as long as the HR 550FS can be operated in conjunction with the HRA 551FS without disturbances or error messages. In the event of a fault, please inform your service agency. The only expendable part of the HR 550FS are the rechargeable batteries. If you notice a marked decrease in battery life, it's time to exchange them (handwheel batteries, ID 623 166-02).

Revised variants:

- HR 550FS mobile wireless handwheel: ID 598 515-03
- HR 550FS mobile wireless handwheel (with detent) ID 606 622-03
- HRA 551FS base and charging station for HR 550FS: ID 731 928-02

In mid-2012, the wireless handwheel system was revised with respect to the radio connection. During this modification, the transmitter-receiver electronics of the wireless handwheel system was improved significantly regarding the sensitivity to interference.

Please note that the previous versions of the HR550FS and HRA551 and the new, revised versions cannot be exchanged with each other anymore. This means that both units (HR and HRA) must be replaced if replacement of a HR 550FS (variant -02 by -03) or HRA 551FS (variant -01 by -02) is necessary.

Moreover, HEIDENHAIN recommends an update of the control hardware to the NC software 34049x-07 SP5, NC-Software 60642x-02 SP5 or higher due to further improvements in communication with the control.

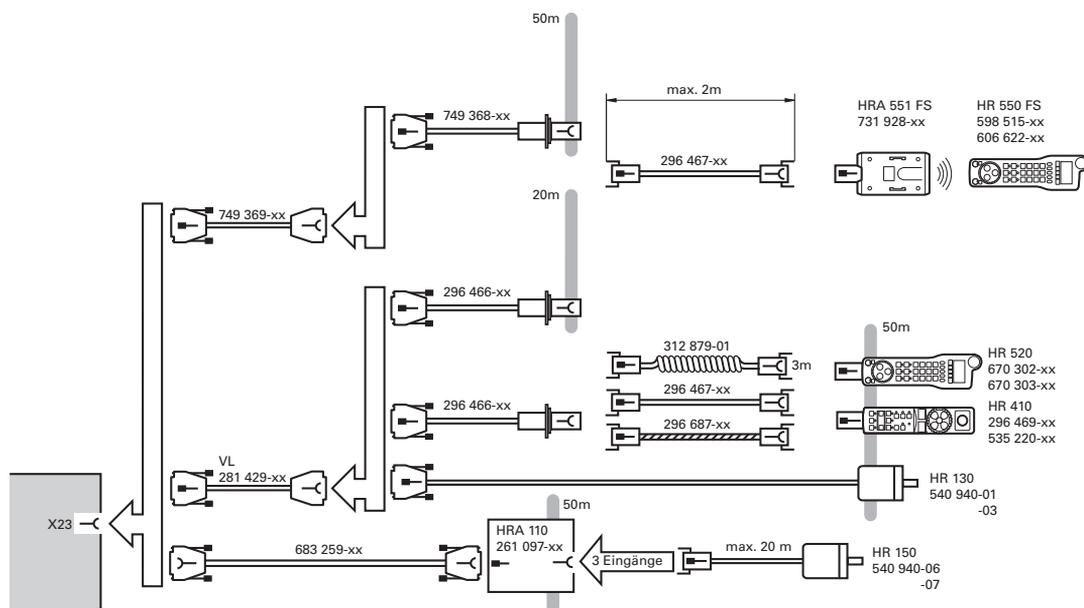


Connecting the wireless handwheel system

A wireless handwheel system is always connected to the control via the M23 plug connection of the HRA 551FS. The advantage of this is that the wireless handwheel system is also supplied with power via this connection so that no additional supply lines must be provided.

The serial communication, dual-channel emergency stop and dual-channel permissive buttons of the HR 550FS mobile part are routed from the HRA 551FS (M23 plug connection with handwheel cable, ID 296467-xx) via the corresponding handwheel adapter, ID 749368-xx, to the control hardware. For this purpose, special, short handwheel cables (ID 296467-xx) in lengths of 1.0 m, 1.5 m and 2.0 m were introduced. The reason for this is the charging current required for the HR 550FS mobile part. Due to the small cross section of the handwheel cable it can be transmitted only over short distances. This means that longer cables with ID 296467-xx cannot be used for the wireless handwheel system. If longer handwheel cables are required, HEIDENHAIN should be consulted. A maximum distance of 50 m between X23 (MC 4xx or MB 620/PLB 6001 for HSCI) and the HRA 551FS base station can be attained by using the adapter cable ID 749368-xx and the optional extension cable ID 749369-xx (with a handwheel cable, ID 296467-xx, with a length of max. 2.0 m).

Please note that the new adapter cables, ID 749368-xx, and extension cables, ID 749369-xx, will not be available until April 2011. Until then, the wireless handwheel system must be connected with the currently available cables, which limits the attainable cable length to the base station to 22 m!



Use of available cable connections:

For cable lengths up to 22 m, the existing cable connections for HEIDENHAIN handwheels of the HR 4xx and HR 5xx series can be used to retrofit a wireless handwheel system. The existing handwheel connecting cables must be checked regarding their length and the possibility of connecting a wireless handwheel system. Please refer to the cable overview shown above for the wireless handwheel system.

The relay contacts for the permissive buttons and emergency stop button on the HRA are four individual, normally open contacts. This information can then be picked up at the cable adapter.

■ Behavior of the emergency stop contacts

When the emergency stop button is not pressed, the relays are actuated, and therefore the relay contacts are closed. If an emergency stop is triggered, the control voltage of the relays will be switched off and the relay contacts will open. For the control and the associated safety circuits, the relay contacts have the same effect as normally closed contacts. If the batteries of the wireless handwheel are empty or if the wireless handwheel is out of range, the relay contacts are open.

■ Behavior of the contacts for the permissive buttons

Depending on the setting in MP7645.3, the permissive buttons are represented as two normally open contacts, or as one normally open contact and one normally closed contact. This makes it possible to adapt the handwheel (permissive buttons) to the requirements of the planned safety design of the respective machine.

HEIDENHAIN recommends setting the MP7645.3 to 1 to make the handwheel permissive buttons cross-circuit proof. To ensure compatibility with previous machine designs, MP7645.3 is set to 0 by default.

• MP7645.3 = 0:

When neither of the two permissive buttons is pressed, both relays are without control voltage, and therefore the relay contacts are open.

When at least one permissive button is pressed, both relays are actuated and both relay contacts close.

• MP7645.3 = 1:

When neither of the two permissive buttons is pressed, one of the relays is without control voltage, and relay contact A is open. When at least one of the permissive button is pressed, the relay is actuated and relay contact A closes. When neither of the two permissive buttons is pressed, control voltage is applied to the other relay and relay contact B is closed. When at least one of the permissive button is pressed, the relay is not actuated anymore and relay contact B opens. MP7645.3 must be set to 1 for machines with HEIDENHAIN functional safety.



Note

The interfaces comply with the requirements of EN 60204-1:2006 for "protective extra-low voltage (PELV)."

MP_7645.3 HRA 55xFS: Evaluation of the permissive buttons

Input: 0: Permissive buttons not pushed → Contacts A and B are open, but not cross-circuit proof
 1: Handwheel permissive buttons are cross-circuit proof

Pin layout for the various extension cables, adapter cables, connecting cables, and the handwheel:

Extension cable, ID 281429-xx			Adapter cable, ID 296466-xx			For connecting cable ID, see "Introduction" chapter.			HRA 551FS	
D-sub connector (male) 9-pin		D-sub connector (female) 9-pin	D-sub connector (male) 9-pin		Mounted coupling (female) (5+7)-pin	Cnnctr. (male) (5+7)-pin		Cnnctr. (female) (5+7)-pin		
Hsg.	Shield	Housing	Hsg.	Shield	Housing	Hsg.	Shield	Housing	Hsg.	Internal X230
2	White	2	2	White	E	E	White	E	E	
4	Brown	4	4	Brown	D	D	Brown	D	D	
6	Yellow	6	6	Yellow	B	B	Yellow	B	B	
7	Gray	7	7	Gray	A	A	Gray	A	A	
8	Green	8	8	Green	C	C	Green	C	C	
				WH/BK	6	6	BK	6	6	8
				YL/BK	7	7	RD/BL	7	7	7
				WH/RD	5	5	Red	5	5	5
				WH/BL	4	4	Blue	4	4	6
				WH/GN	2	2	WH/GN	2	2	3
				WH/YL	3	3	BN/GN	3	3	1
				WH/BN	1	1	GY/PK	1	1	4, 2
				WH/BN	X1/3	Contacts A + B				
				WH/YL	X1/1	Contact B (permissive button)				
				WH/GN	X1/2	Contact A (permissive button)				
				WH/BL	X2/1	Contact A/Terminal 1 (EMERGENCY STOP)				
				WH/RD	X2/2	Contact A/Terminal 2 (EMERGENCY STOP)				
				YL/BK	X2/4	Contact B/Terminal 1 (EMERGENCY STOP)				
				WH/BK	X2/3	Contact B/Terminal 2 (EMERGENCY STOP)				



Note

The interfaces comply with the requirements of EN 61800-5-1 for "protective extra-low voltage (PELV)."



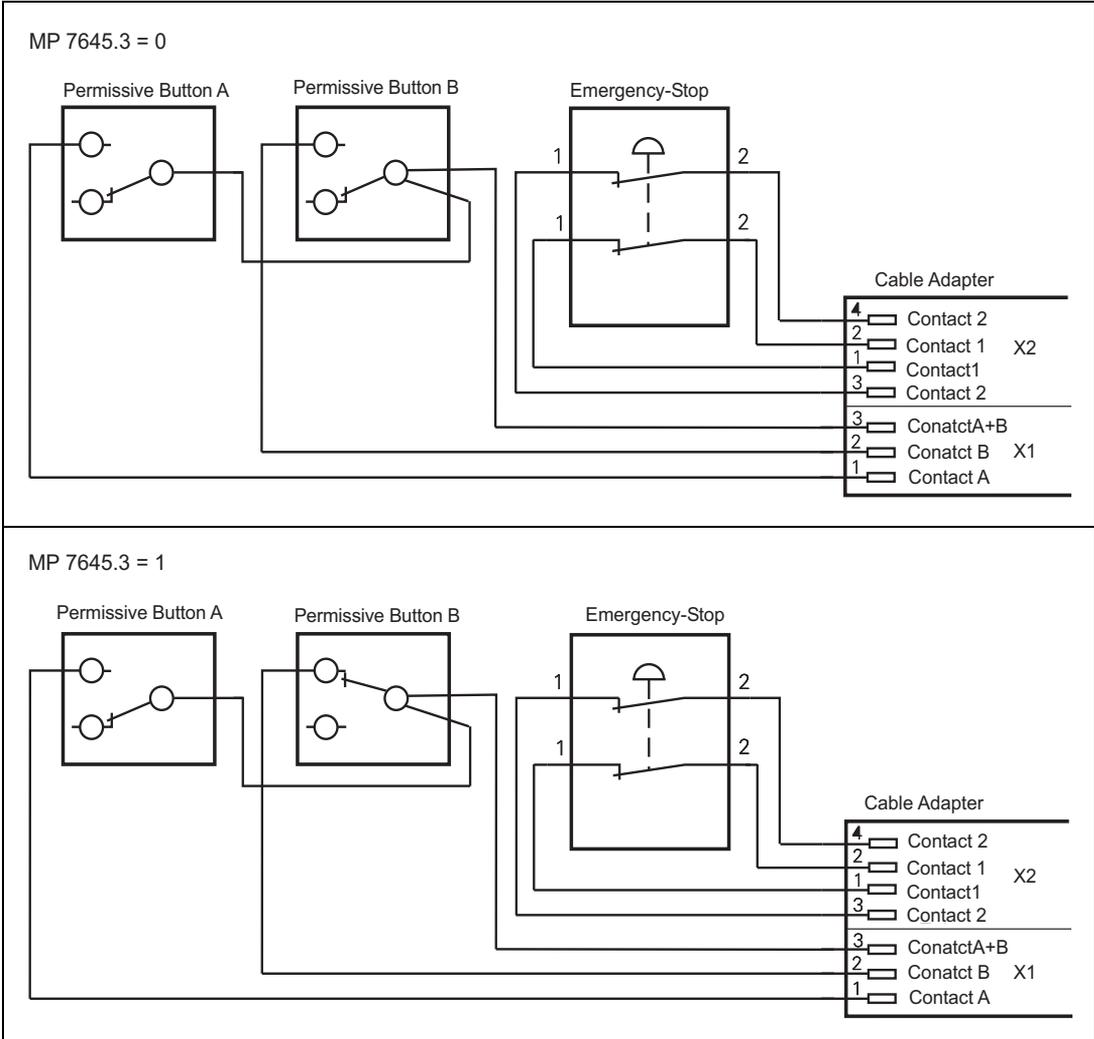
Danger

Only units that comply with the requirements of EN 61800-5-1 for "protective extra-low voltage (PELV)" may be connected.

The cable adapter includes plug-in terminal strips for the contacts of the emergency stop button and permissive button (max. load 24 V-, 1.2 A).

The plug-in terminal strips are supplied together with the adapter cable. If you have an immediate need for these terminal strips, they can be ordered in advance. See the "Additional components" table below.

Wiring of the contacts for the emergency stop button and permissive button:



Additional components	ID
Connecting cable	
Spiral cable	312879-01
Normal cable	296467-xx
Metal armor tubing	296687-xx
Plug-in terminal strips for advance ordering	
4-pin terminal block	266364-12



Prerequisites for use of the wireless handwheel system

The following conditions must be taken into account for the operation of a wireless handwheel system:

- The HR 550FS wireless handwheel system with the HRA 551FS is currently only available in combination with the iTNC 530 control.
- The iTNC 530 requires NC software version 34049x-06 or 60642x-01, or higher.
- The use of radio equipment in the 2.4 GHz range must be generally permitted at the intended place of use (end user's site).
- Country-specific and customer-specific requirements and regulations for radio equipment at the intended place of use of the wireless handwheel system must be taken into account.
- At the intended place of use, every wireless handwheel system from HEIDENHAIN needs a suitable and vacant radio channel in the 2.4 GHz ISM band. If this cannot be guaranteed, we advise against using the wireless handwheel system (see "Important notes on the radio network of the wireless handwheel system" on page 404 and see "Checking the radio environment at the intended place of use" on page 405).
- If several wireless handwheel systems from HEIDENHAIN are to be used within a certain range, a vacant channel must be selected for each individual wireless handwheel system.
- If there are not enough vacant radio channels available for the wireless handwheel systems and identical channel numbers have to be used for the wireless handwheel systems because of other ISM band participants, a minimum distance of 200 m must be maintained between the HRA 551FS base stations. In addition, the transmitting and receiving power of the wireless handwheel system can be reduced during commissioning. Three power stages are available for this. They are set using the wireless handwheel configuration dialog on the control.

Ideally, "coexistence planning" for all ISM band participants is available for the intended place of use of the wireless handwheel system. The coexistence planning includes all radio participants and the radio channel or radio range they are using. This planning can be used to determine whether a vacant radio channel is available for the wireless handwheel, and therefore whether a wireless handwheel system can be used.

Important notes on the radio network of the wireless handwheel system

The HEIDENHAIN wireless handwheel system uses the ISM band in the frequency range of 2.405 GHz to 2.480 GHz up to a maximum transmission power of 10 mW. As the ISM band generally is a freely accessible frequency range worldwide (country-specific restrictions must be checked individually for the country of destination), many radio systems have been developed for this range. As other radio systems can disturb the radio transmission of the wireless handwheel system, a vacant radio channel (ZigBee channels 11 to 26) is always required to ensure noise-free radio communication between the HR 550FS and HRA 551FS.

Since a handwheel is always connected with the movement of axes, the operator must be able to trigger an emergency stop of the machine by means of the handwheel at any time. This results in a special situation for the emergency stop switch of the HR 550FS wireless handwheel system. An unsafe situation in the radio communication must always trigger a safe reaction from the control. Therefore, an interference in the radio communication also means an emergency stop at the HRA 551FS as reaction. An unexpected emergency stop of the machine is undesirable and may cause damage to the contour of the workpiece being machined. It is therefore essential to ensure a noise-free radio communication. Reliable operation of the wireless handwheel always requires planning the coexistence of the radio users active in the frequency and radio range concerned. If this cannot be guaranteed because there is no free radio channel available, we advise against using the wireless handwheel system.

HEIDENHAIN cannot guarantee noise-free radio communication of the HR 550FS wireless handwheel system with the HRA 551FS. Even if an available radio channel has been found and the proper functioning of the system has been ascertained, it is possible that the availability situation of the 2.4 GHz ISM band will change later on. If this situation cannot be corrected by changing the wireless handwheel system or the competing radio system to another channel, it may become necessary to revert to using a handwheel with cable. HEIDENHAIN therefore offers the HR 520 handwheel with cable as an alternative.

If a wireless handwheel is to be operated in an unsafe radio environment anyway, the HR 550FS mobile part must be placed into the charging position of the HRA 551FS if a critical situation occurs, in which a workpiece could be damaged through an emergency stop. The contacts on the HR 550FS and the HRA 551FS provide a direct contact between the mobile part and the base station, and therefore communication is independent of the radio connection.

Checking the radio environment at the intended place of use

The following procedure is very important to determine whether a wireless handwheel system can be used in a certain radio environment:

Procedure

- Clarification of whether the HEIDENHAIN wireless handwheel system is allowed to use the 2.4 GHz ISM band in the customer's area (country-specific and company-internal regulations at the intended place of use and the end user's site must be taken into account).
- Inquiry about known radio participants at the end user's site. Ideally, the end user can provide information regarding the utilization of the 2.4 GHz ISM band at the intended place of use of the wireless handwheel.
- If the customer cannot provide exact information about which devices are using the 2.4 GHz ISM band at the intended place of use, we recommend performing measurements with a suitable spectrum analyzer and a WLAN detector. HEIDENHAIN recommends performing the measurements using a spectrum analyzer to obtain a utilization profile of the 2.4 GHz ISM band, see page 406. It is also possible to use a WLAN-capable laptop and the "Network Stumbler" PC software which is available free of charge on the Internet, see page 406.
- Selection and determination of a vacant radio channel for the wireless handwheel system.
- If no vacant radio channel can be found despite the above-mentioned measures, we advise against using the wireless handwheel system.

WLAN networks

In production environments, the ISM band is most frequently used by WLAN networks. Due to their transmitting/receiving characteristics and the transmitting power to be expected, these networks are not suitable for being operated in the same frequency range as a wireless handwheel system. A wireless handwheel system must always be operated outside the frequency range of a WLAN network.

Evaluation of the radio environment

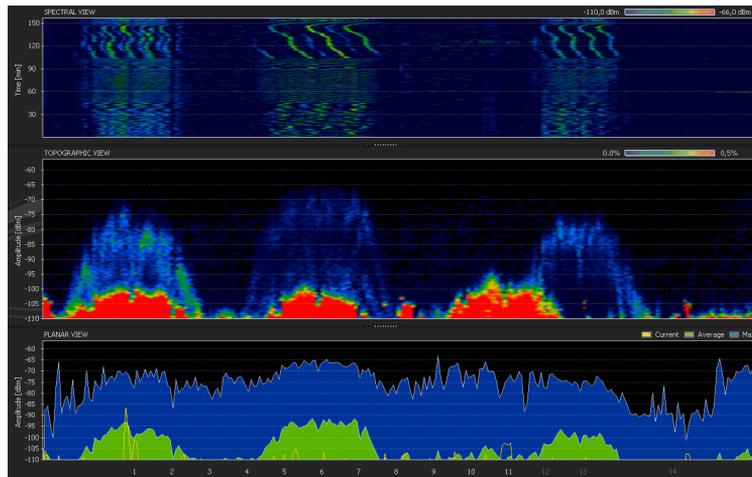
The radio environment must be evaluated at the intended place of use under the specific local conditions. All radio devices in the environment should be active for the evaluation.

HEIDENHAIN recommends performing the measurements using a "Wi-Spy WLAN USB 2.4x Spectrum Analyzer" to obtain a utilization profile of the 2.4-GHz ISM band. This USB device for PCs together with the associated Chanalyzer 3.4 software makes it possible to appropriately visualize the radio activity in the frequency spectrum and to identify other WLAN networks that could interfere with the connection.

The "Network Stumbler" software for PCs can be used as an alternative. However, it is only capable of detecting WLAN networks. The identified WLAN networks are displayed in an easy-to-read list. The channel used by the network is listed next to the name of the WLAN network. A radio channel outside the frequency range of a WLAN network must be selected for the wireless handwheel system.

Figure 1: Spectrum analyzer view of radio environment

Three WLAN networks have been identified in this example environment. In this example, the wireless handwheel must be set on a vacant radio channel outside the frequency range of the WLAN networks.



Note

The channel designations in the dialog of the control are numbered consecutively from 0 to 15 and do not correspond to the official ZigBee channel numbers. WiFi (WLAN) networks have channel frequencies and also channel designations, which, however, differ from those of the wireless handwheel system (ZigBee), see page 408.

Figure 2: Possible position of radio channel

Three WLAN networks have been identified in this example environment. In this example, the wireless handwheel could be set on the vacant ZigBee channel 15 outside the WLAN networks. The diagram shows the typical curve of a ZigBee radio device (white curve), such as the HR 550FS. Another possibility would be one of the vacant ZigBee channels 14, 20, 21 or 26.

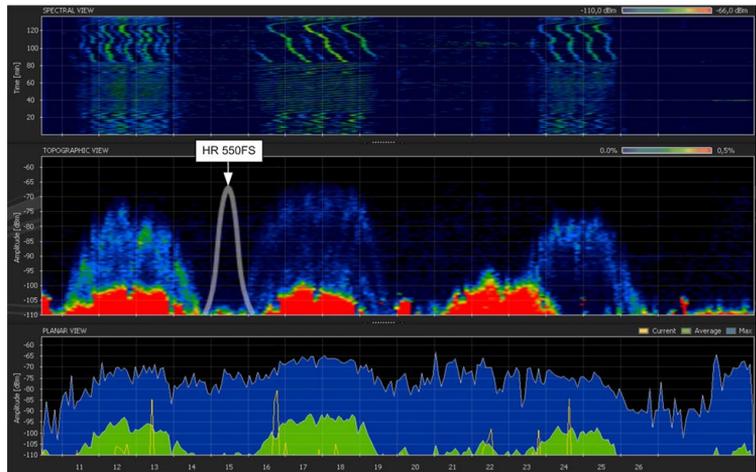
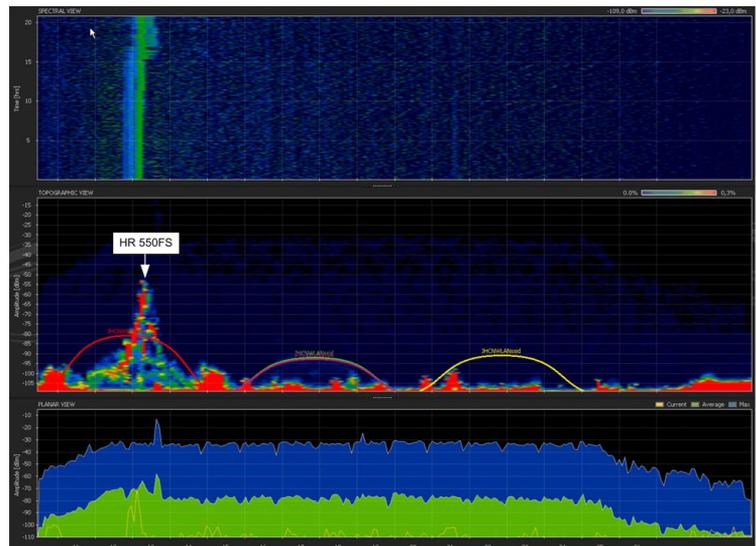


Figure 3: Radio channel of wireless handwheel collides with WLAN

In this example, the radio channel of the wireless handwheel is located in the range of the first WLAN network. This makes noise-free operation of the wireless handwheel impossible. Another channel must be selected for the wireless handwheel. In this case, ZigBee channel 20, 25 or 26 would be possible.

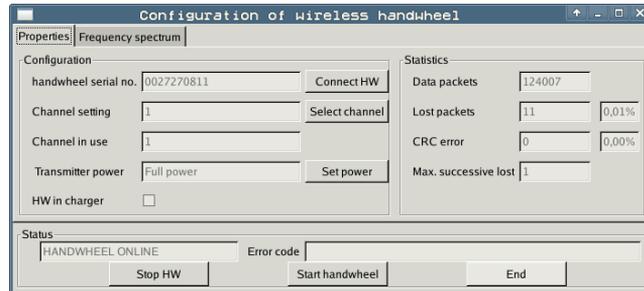


Information in the handwheel configuration dialog

The general procedure for the configuration of the HR 550FS handwheel is described in the iTNC 530 User's Manual.

To access the handwheel configuration dialog, proceed as follows:

- ▶ Press the MOD key
- ▶ Scroll through the soft-key row
- ▶ Press the SET UP WIRELESS HANDWHEEL soft key



HEIDENHAIN generally recommends that in **Channel setting** you manually define the radio channel to be used. To select a vacant radio channel, please follow the information provided in this document. We advise against using the **Best channel** setting because it does not allow planning for the radio environment, and therefore unexpected disturbances during the operation of the wireless handwheel system may occur.

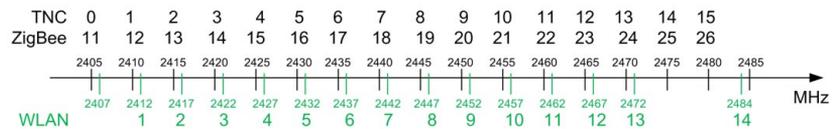
Channel setting

The channel designations in the dialog of the control are numbered consecutively from 0 to 15 and do not correspond to the ZigBee channel numbers. Channel 0 at 2.405 GHz in the dialog on the control corresponds to ZigBee channel 11; channel 1 at 2.410 GHz in the dialog on the control corresponds to ZigBee channel 12, and so on. With the next software versions, the ZigBee channel numbers will also be displayed in the dialog on the control. The diagram below shows the relationship between the channel designations in the dialog on the control and the ZigBee channel numbers.

WiFi (WLAN) networks have channel frequencies and also channel designations, which, however, differ from those of the wireless handwheel system (ZigBee). For example, the WLAN channel 13 is at 2.472 GHz, and the wireless handwheel channel 13 is at 2.415 GHz. The diagram below shows the relationship between the channel designations in the dialog on the control, the ZigBee channel numbers and the WIFI (WLAN) channels. In addition, a detected WLAN channel has a channel width of 20 MHz, which must be taken into account in the coexistence planning. The four ZigBee channels next to a WLAN channel must therefore not be used in order to ensure that 20 MHz of channel width remain available.

Example: WLAN on channel 6

As a result, the ZigBee channels 16, 17, 18 and 19 must not be used. A wireless handwheel should be set only to ZigBee channel 15 or 20, which corresponds to channel 4 or 9 in the dialog on the control.



Formula for the next vacant wireless handwheel channel (TNC channel) depending on a WLAN channel:

$$\text{TNC channel+} = \text{WLAN channel} + 3$$

$$\text{TNC channel-} = \text{WLAN channel} - 2$$

Example:

$$\text{TNC channel+} = 6 + 3 = 9$$

$$\text{TNC channel-} = 6 - 2 = 4$$

Radio connection during operation

After installation and during normal operation, the following options of the handwheel configuration dialog on the control provide information about the quality of the radio connection:

■ Data packets

Number of data packets transmitted by radio transmission since the last activation of the wireless handwheel.

■ Lost packets – Number

Number of data packets lost since the last activation of the wireless handwheel. For example, if this value shows that more than two packets are lost within 10 seconds during radio transmission (wireless handwheel is outside the handwheel adapter), the limit range has been reached. The radio connection should be checked. You should change to another radio channel.

■ Lost packets – Percentage

Percentage of data packets lost since the last activation of the wireless handwheel.

- Value < 0.05 %
The radio connection functions properly.
- Value between 0.05 % and 0.1 %
The radio connection functions properly, but it is in the limit range. The radio connection should be checked. You should change to another radio channel.
- Value > 0.1 %
The radio connection is not reliable. We advise against operating the wireless handwheel on this channel. The radio connection should be checked. You must change to another radio channel.

■ Max. successive lost

Number of data packets that were lost successively, one after the other. If four successive data packets are lost, an emergency stop is triggered.

- Value = 1
The radio connection functions properly.
- Value = 2
The radio connection functions properly, but it is in the limit range. The radio connection should be checked. You should change to another radio channel.
- Value = 3
The radio connection is not reliable. We advise against operating the wireless handwheel on this channel. The radio connection should be checked. You must change to another radio channel.



3.41.4 HR 410 portable handwheel

The HR 410 is a portable electronic handwheel.

For the assignment of the keys of the HR 410 to the PLC inputs and outputs, see "HR 410 portable handwheel" on page 1510.

Pin layout

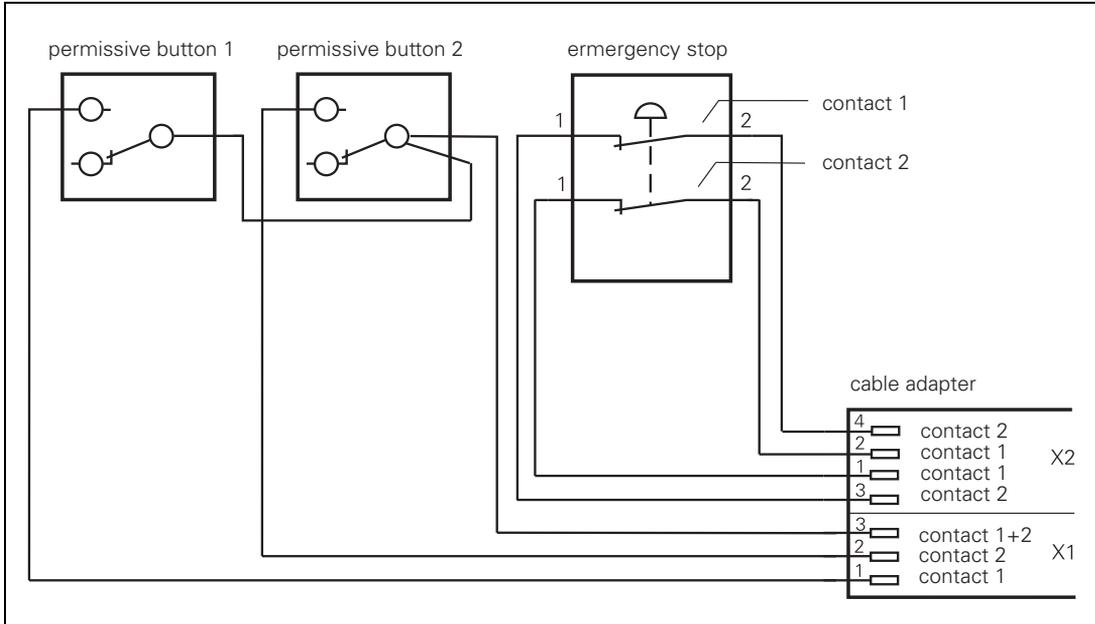
Pin layout for the various extension cables, adapter cables, connecting cables, and the handwheel:

Extension cable, ID 281429-xx		Adapter cable, ID 296466-xx				For connecting cable ID, see "Introduction" chapter.			HR 4xx	
D-sub connector (male) 9-pin		D-sub connect or (fem.) 9-pin	D-sub connector (male) 9-pin		Mounted coupling (female) (5+7)-pin	Cnnctr. (male) (5+7)-pin		Cnnctr. (female) (5+7)- pin	Connector (male) (5+7)-pin	
Housg.	Shield	Housg.	Housg.	Shield	Housing	Housing	Shield	Housing	Housg.	Shield
2	White	2	2	White	E	E	White	E	E	
4	Brown	4	4	Brown	D	D	Brown	D	D	
6	Yellow	6	6	Yellow	B	B	Yellow	B	B	
7	Gray	7	7	Gray	A	A	Gray	A	A	
8	Green	8	8	Green	C	C	Green	C	C	
					6	6	BK	6	6	
					7	7	RD/BL	7	7	
					5	5	Red	5	5	
					4	4	Blue	4	4	
					2	2	WH/GN	2	2	
					3	3	BN/GN	3	3	
					1	1	GY/PK	1	1	
					WH/BN	X1/3	Contacts A + B (power supply)			
					WH/YL	X1/1	Contact B (permissive button)			
					WH/GN	X1/2	Contact A (permissive button)			
					WH/BL	X2/1	Contact A/Terminal 1 (EMERGENCY STOP)			
					WH/RD	X2/2	Contact A/Terminal 2 (EMERGENCY STOP)			
					YL/BK	X2/4	Contact B/Terminal 1 (EMERGENCY STOP)			
					WH/BK	X2/3	Contact B/Terminal 2 (EMERGENCY STOP)			

The adapter includes plug-in terminal strips for the contacts of the emergency stop button and permissive button (max. load 24 V-, 1.2 A).

The plug-in terminal strips are supplied together with the adapter cable. If you have an immediate need for these terminal strips, they can be ordered in advance. See the "Additional components" table below.

Internal wiring of the contacts for the emergency stop button and permissive button:



Specifications of the switch contacts in permissive buttons and emergency stop button

- Rated voltage: +24 V DC
- Maximum current: 1.2 A

Additional components		ID
Dummy plug for emergency stop circuit		271958-03
Connecting cable		
Spiral cable		312879-01
Normal cable		296467-xx
Metal armor tubing		296687-xx
Plug-in terminal strips for advance ordering		
3-pin terminal block		266364-06
4-pin terminal block		266364-12

3.41.5 HR 130 panel-mounted handwheel

Standard cable length for the HR 130 is 1 meter.

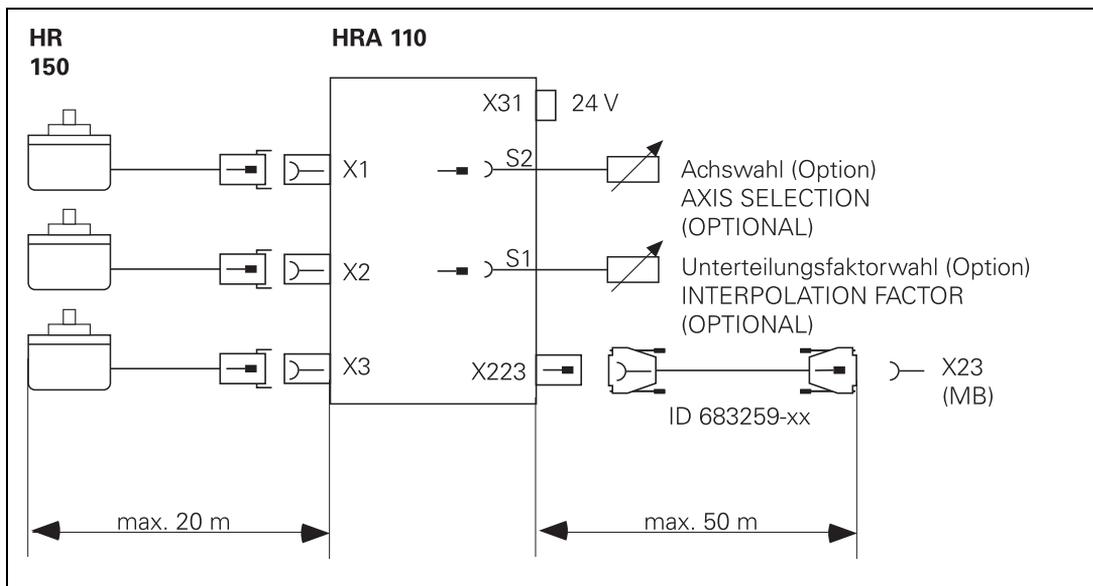
Pin layout for extension cable and handwheel:

Extension cable, ID 281429-xx			HR 130 ID 254040-xx	
D-sub connctr. (male) 9-pin		D-sub connector (female) 9-pin	D-sub connctr. (male) 9-pin	
Housing	Shield	Housing	Housing	Shield
2	White	2	2	White
4	Brown	4	4	Brown
6	Yellow	6	6	Yellow
8	Green	8	8	Green
7	Gray	7		

3.41.6 HRA 110 handwheel adapter

With the handwheel adapter you can connect two or three HR 150 panel-mounted handwheels to the iTNC 530 HSCI.

The first and second handwheels are assigned to the X and Y axes. The third handwheel can be assigned through a selection switch (option).



An additional switch enables you to select, for example, the interpolation factor for the handwheels. In the PLC you must evaluate the current position of the handwheel selection switch and activate the corresponding interpolation factor with Module 9036.

X1 to X3: Inputs on the HRA 110 for the HR 150

Pin layout:

HRA 110	
Connection (female) 9-pin	Assignment
1	$I_1 +$
2	$I_1 -$
5	$I_2 +$
6	$I_2 -$
7	$I_0 -$
8	$I_0 +$
3	+5 V
4	0 V
9	Internal shield
Housing	External shield



Attention

The interface is electrically separated (protective extra-low voltage (PELV) as per EN 61800-5-1).



X23: Connection to MB 620

Pin layout on the HRA 110:

HRA 110	
D-sub connector (female) 9-pin	Assignment
1	RTS
2	0 V
3	CTS
4	+12 V +0.6 V (U_V)
5	Do not assign
6	DSR
7	RxD
8	TxD
9	DTR
Housing	External shield



Attention

The interface is electrically separated (protective extra-low voltage (PELV) as per EN 61800-5-1).

X31: HRA 110 supply voltage

Pin layout on the HRA 110:



Attention

The interface is electrically separated (protective extra-low voltage (PELV) as per EN 61800-5-1).



Attention

The power supply of the PLC must not be used simultaneously for the HRA 110, otherwise the galvanic isolation of the PLC inputs/outputs would be bridged.

HRA 110	
Connecting terminal	Assignment
1	+24 V– (PELV as per EN 61800-5-1)
2	0 V

Maximum current consumption 200 mA.

3.42 CML 110 capacitor module



Danger

Before service or maintenance work, you must ensure that the CML 110 has been completely discharged.

The CML 110 (Capacitor Module Low Voltage) can be used to realize the LIFTOFF function if line power fails.

The LIFTOFF function can protect workpieces and tools from damage. When a power failure occurs and the LIFT OFF function is active, the iTNC 530 HSCI tries to lift the tool off of the contour, using the residual energy of the DC-link. In this case, the various releases for operating the control system must be maintained during the LIFTOFF. The CML 110 ensures the 24 V supply for this.

Connection

The CML 110 capacitor module is connected via X1 parallel to the 24 V power supply (+ +/ -).

The two ++ and -- terminals of the CML 110 are each connected to each other internally. This makes it possible to connect several CMLs 110 in parallel without needing to use additional external terminals.

Utilizability

Calculating the utilizability of the CML 110:

A successful LIFTOFF mainly depends on sufficient energy being available in the DC-link of the inverter system. Generally it suffices if the energy in the DC-link is available for the duration of one second. The 24 V supply must also be ensured for precisely this time. The following formula can be used to check this:

$$t = R_L \times C \times \ln (U_O/U_C)$$

where:

t = time until U_C is reached

R_L = ohmic load of the consumers

C = capacitance of the CML (for CML 110 = 8.3 F)

ln = natural logarithm

U_O = output load of the power supply unit with which the CML is operated

U_C = lowest voltage at which the consumers still fulfill their function



Example:

During operation at 24 V, a total current of 10 A is required for switching the control components on. This corresponds to an ohmic load of 2.4 ohms. In addition, the voltage for the 24 V components may not sink below 18 V (e.g. switching voltage of the contactors), for example.

This means:

$$t = 2.4 \text{ Ohm} \times 8.3 \text{ F} \times \ln(24 \text{ V}/18 \text{ V})$$

$$t = \mathbf{5.73 \text{ s}}$$

If the line voltage fails, then in the best case the voltage will not fall below 18 V until 5.73 seconds have passed. This is significantly longer than 1 second, and so the CML 110 is suitable for LIFTOFF here.

If the capacitance of the CML 110 should not suffice, then you can also switch more than one CML 110 in parallel. However, here you must note that a maximum charging current of 2.4 A per CML 110 is to be expected at switch-on. The full power of the 24 V power supply unit can only be used once all CMLs have finished charging.

3.43 Connecting cables: Specifications

Device	ID	Max. bend radius (rigid configuration)	Max. bend radius (frequent flexing)	Cable diameter
HSCI	618893-xx	≥ 40 mm	≥ 100 mm	ø 6.8 mm
HDL	625901-xx	≥ 60 mm	≥ 150 mm	ø 11.4 mm
Position 1 V _{PP}	298429-xx, 298430-xx	≥ 20 mm	≥ 75 mm	ø 6 mm
Position 1 V _{PP}	310199-xx, 309783-xx	≥ 40 mm	≥ 100 mm	ø 8 mm
Position EnDat	332115-xx, 323897-xx	≥ 40 mm	≥ 100 mm	ø 8 mm
Position EnDat	313791-xx, 332790-xx	≥ 20 mm	≥ 75 mm	ø 6 mm
Speed 1 V _{PP}	289440-xx, 336376-xx	≥ 40 mm	≥ 100 mm	ø 8 mm
Speed EnDat	336376-xx, 340302-xx, 369502-xx	≥ 40 mm	≥ 100 mm	ø 8 mm
Analog output	290110-xx, 290109-xx	≥ 40 mm	^a	ø 7.3 mm
TS 220	274543-xx	≥ 40 mm	≥ 100 mm	ø 8 mm
SE 640, SE 540	310197-xx, 517518-xx	≥ 40 mm, ≥ 10 mm	≥ 100 mm, ≥ 50 mm	ø 8 mm ø 4.5 mm
HR 130, HR 410 (extension cable)	281429-xx	≥ 20 mm	^a	ø 5.6 mm
HR 410 (ext. cable)	296466-xx	≥ 20 mm	^a	ø 5.6 mm
HR 410	296687-xx	≥ 40 mm	≥ 100 mm	ø 8 mm
RS232, 9-pin	355484-xx	≥ 20 mm	≥ 75 mm	ø 6 mm
RS232, 9-pin (ext. cable)	366964-xx	≥ 20 mm	≥ 75 mm	ø 6 mm
RS232, 25-pin	365725-xx	≥ 40 mm	≥ 100 mm	ø 7.1 mm
RS232, 25-pin (extension cable)	274545-xx	≥ 20 mm	≥ 75 mm	ø 6 mm
USB ^b	354770-xx	≥ 20 mm	≥ 75 mm	ø 4.5 mm
USB (with hub) ^c	624775-xx	Cable like 354770-xx, hub: ø approx. 20 mm, length approx. 115 mm		

a. Conditionally resistant to frequent flexing and torsion

b. These USB cables support USB 1.1 and USB 2.0

c. The hubs integrated in the USB cable (ID 624775-xx) only support USB 1.1

Cables— Instructions for wiring

Keep the following in mind for wiring inside the electrical cabinet:

The stray magnetic field of a KDR 1xx commutating reactor can disturb conductors routed in its proximity. This means that no conductors (such as dc-voltage lines and signal lines), except for the power connections of the motor and the connections between the UVR 1xx and KDR 1xx, should be routed in the proximity of the KDR 1xx. HEIDENHAIN recommends leaving a space of 20 cm around the commutating reactor. This distance is to be maintained regardless of the KDR 1xx used.



3.44 Dimensions



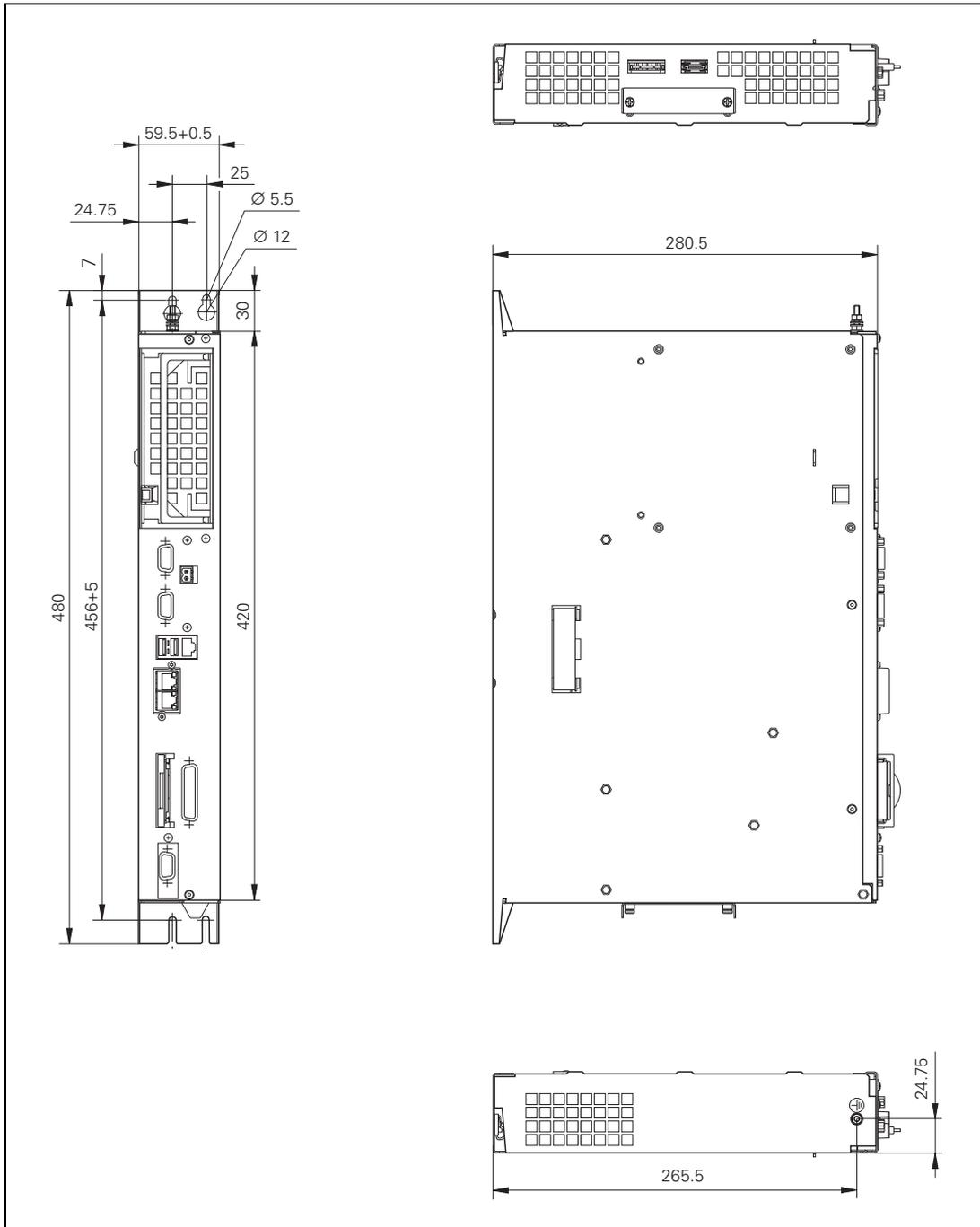
Note

All dimensions are in millimeters [mm].



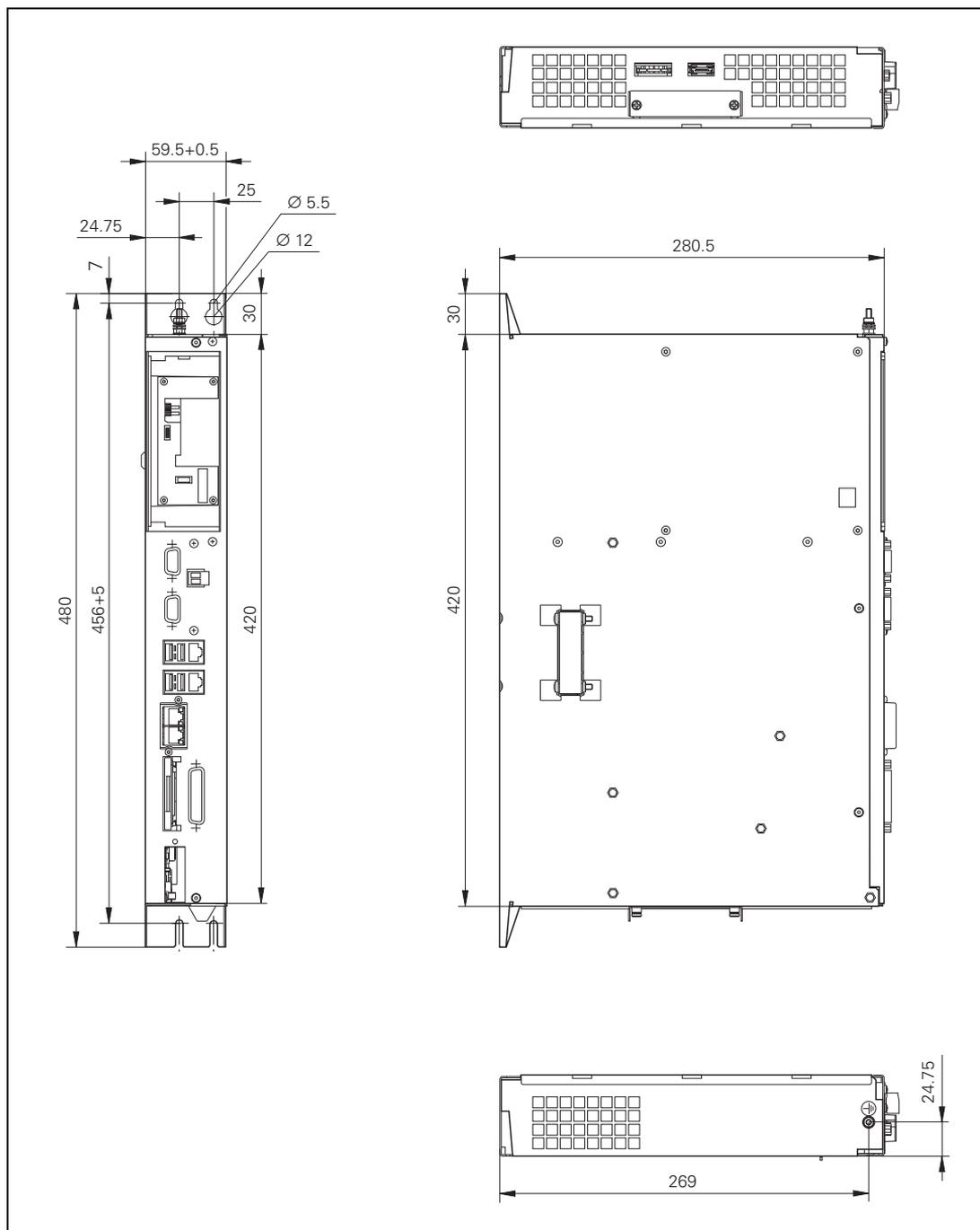
3.44.1 MC 6241

Weight: 4.0 kg



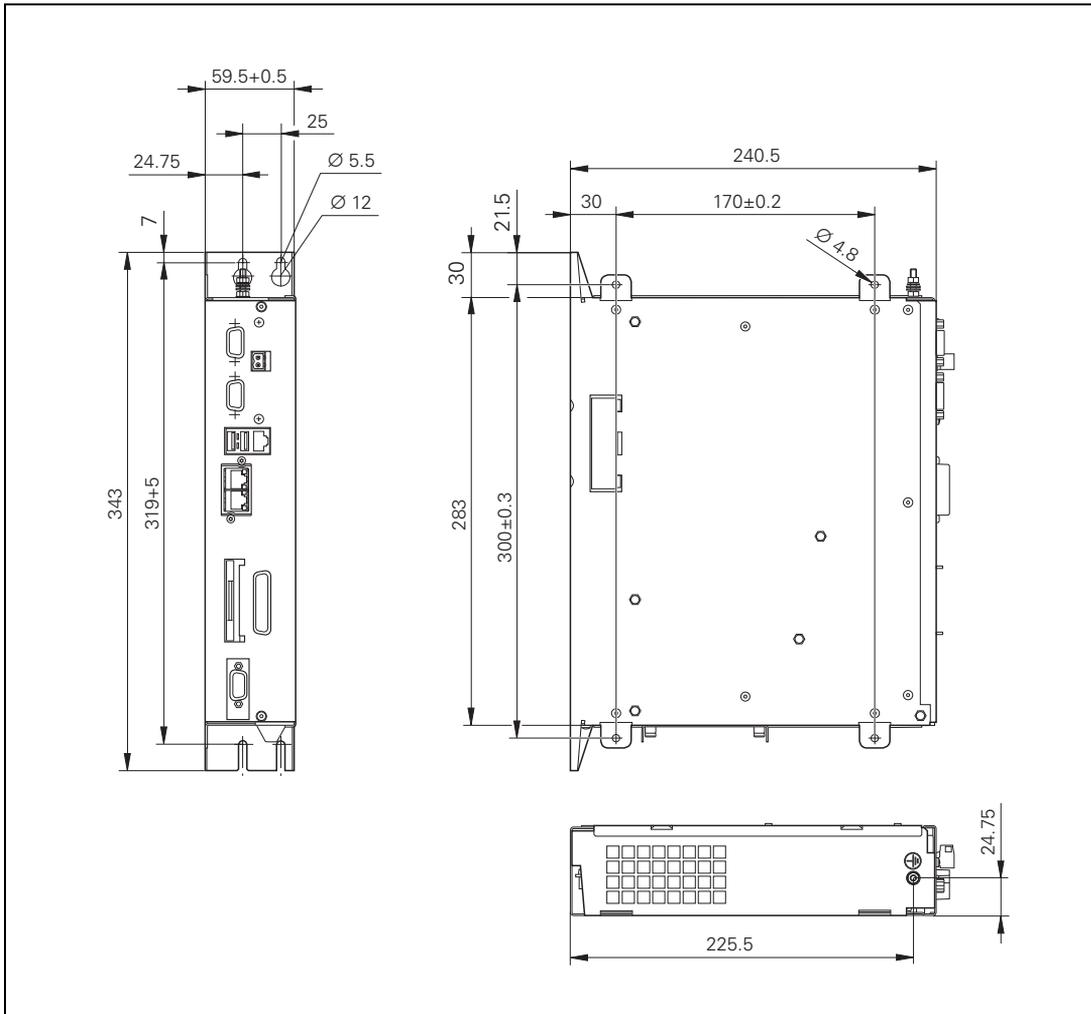
3.44.2 MC 6341, MC 6x51, IPC 6341

Weight: 4.0 kg



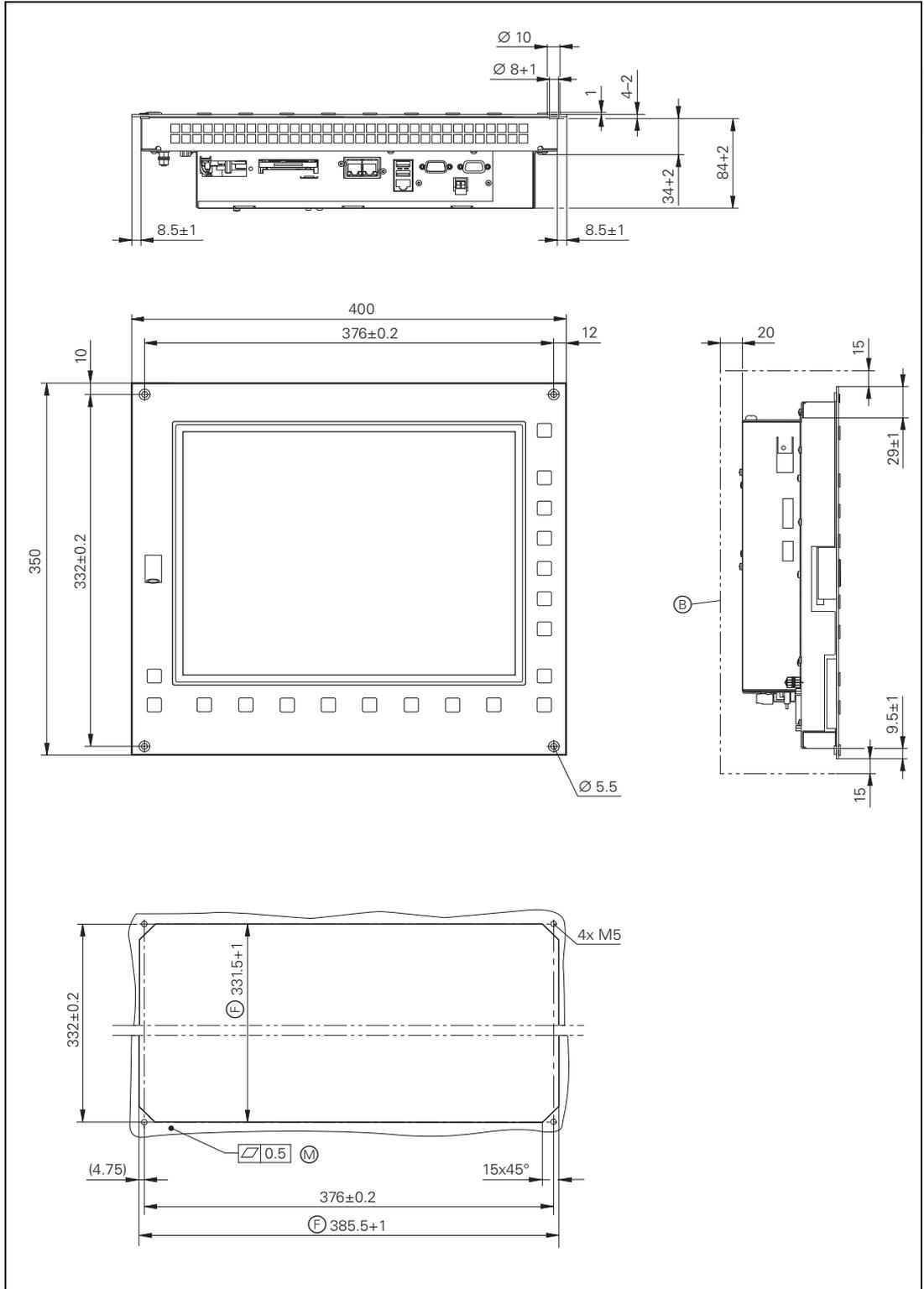
3.44.3 MC 6242

Weight: Approx. 4.0 kg



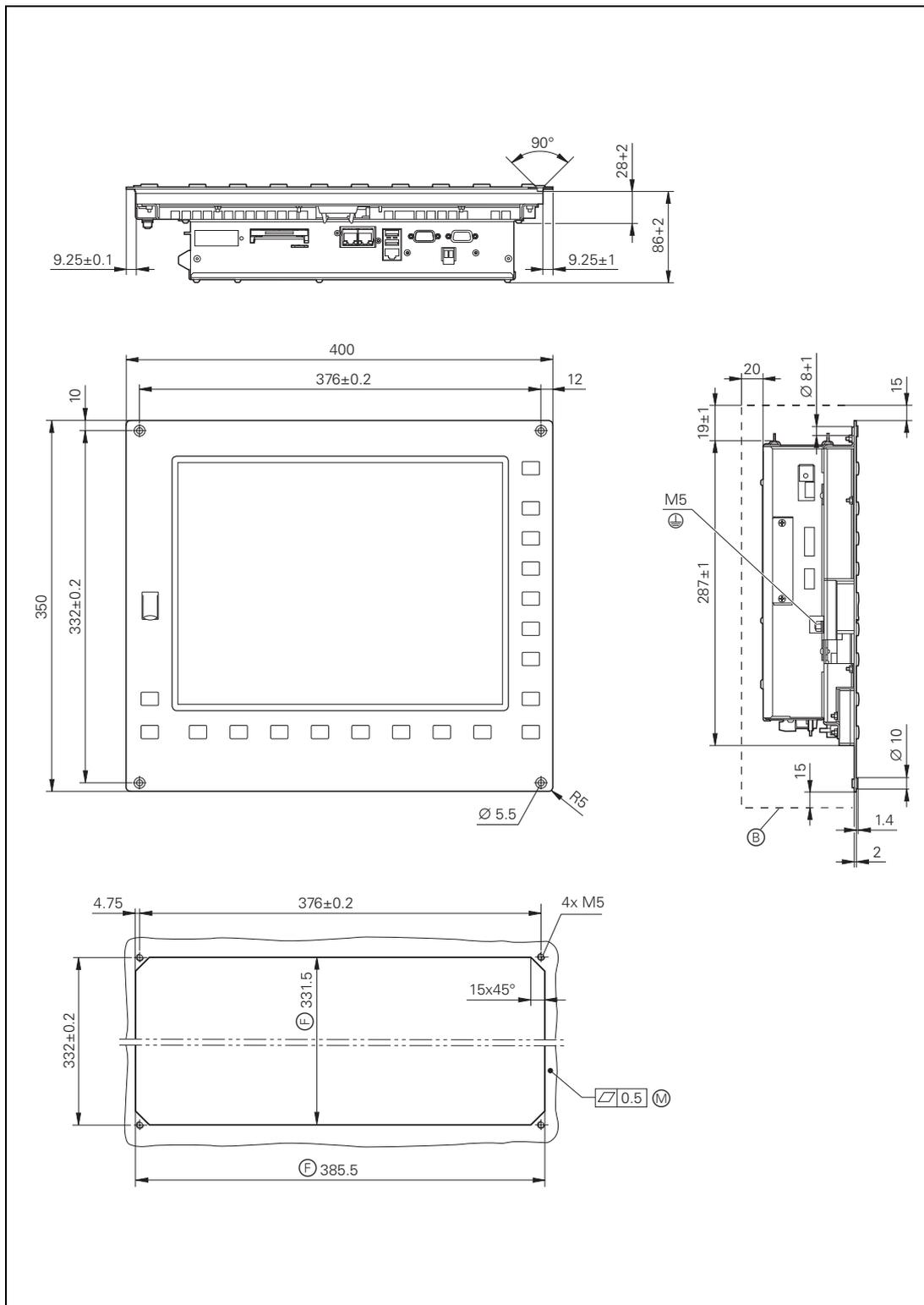
3.44.4 MC 6222

Weight: 5.8 kg

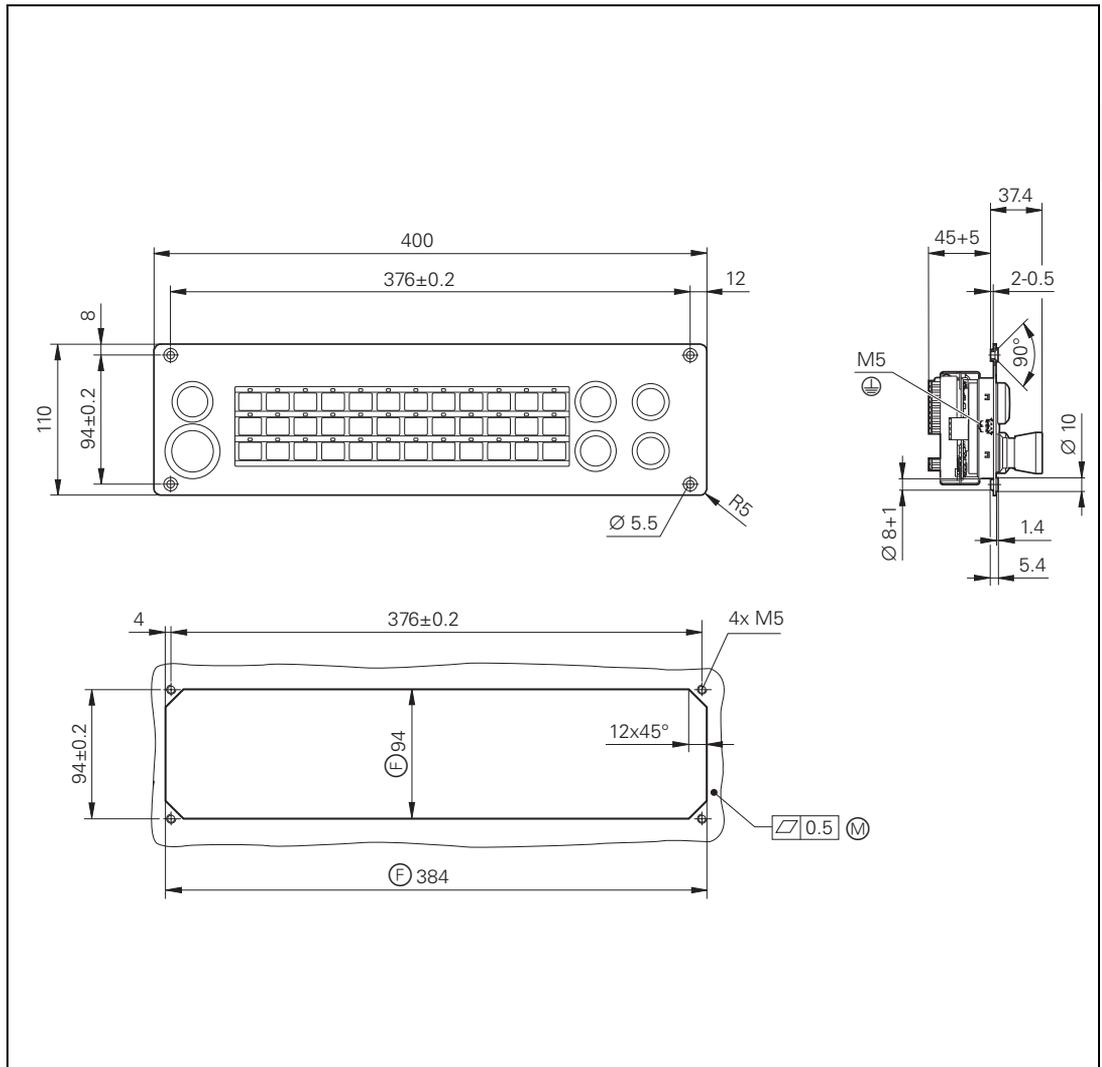


3.44.5 MC 7222

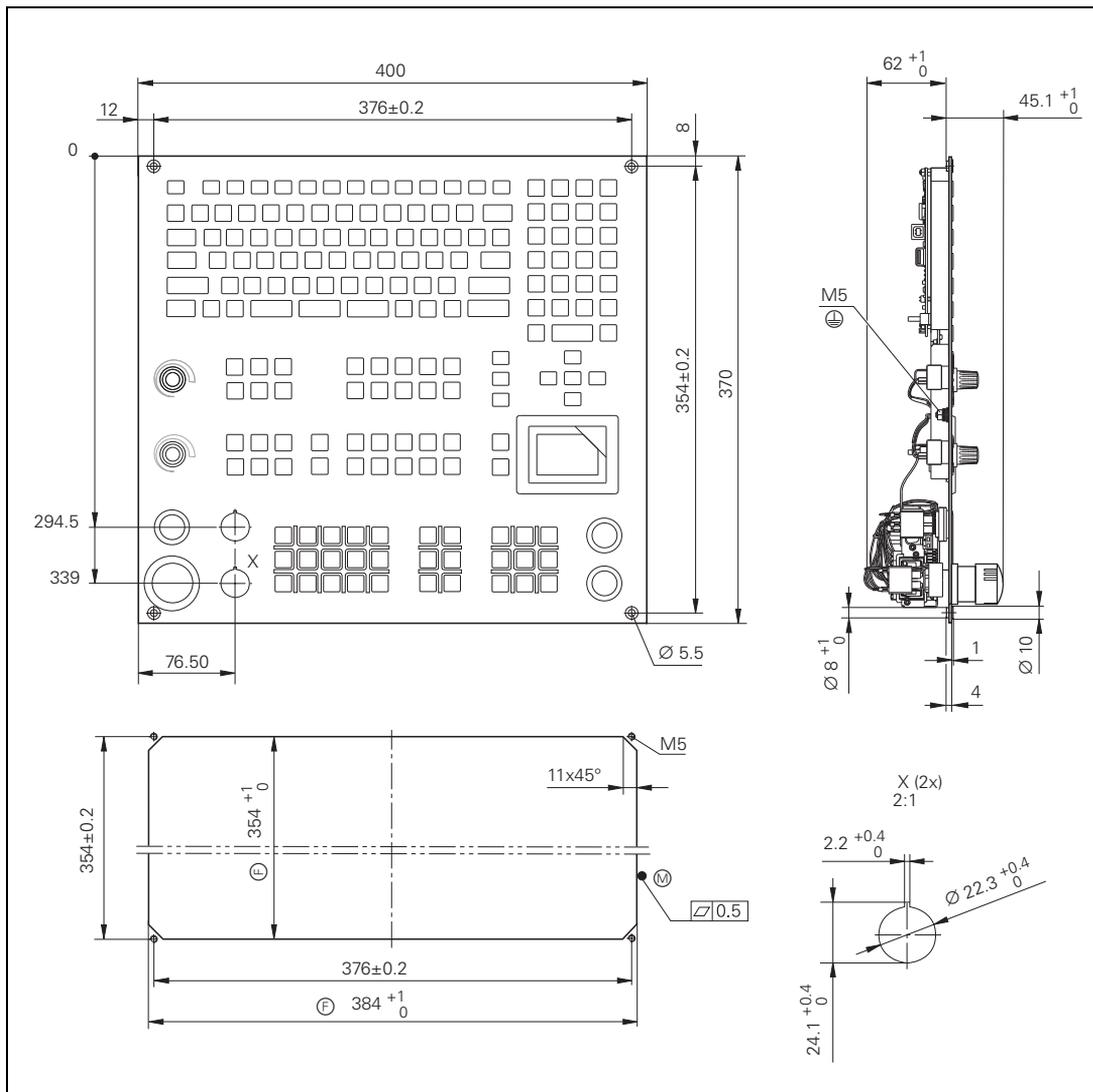
Weight: 7.5 kg



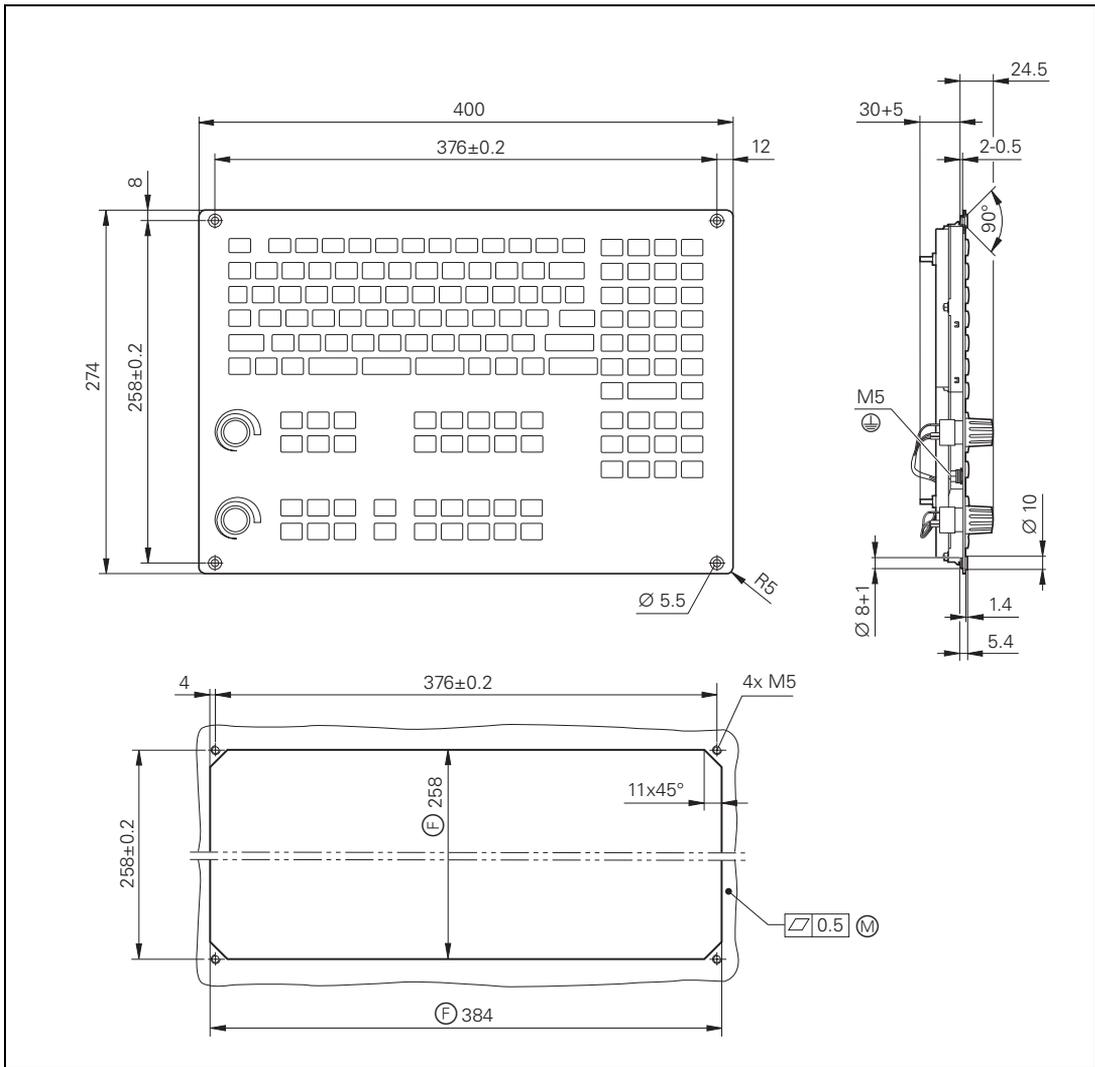
3.44.6 MB 720 (FS)



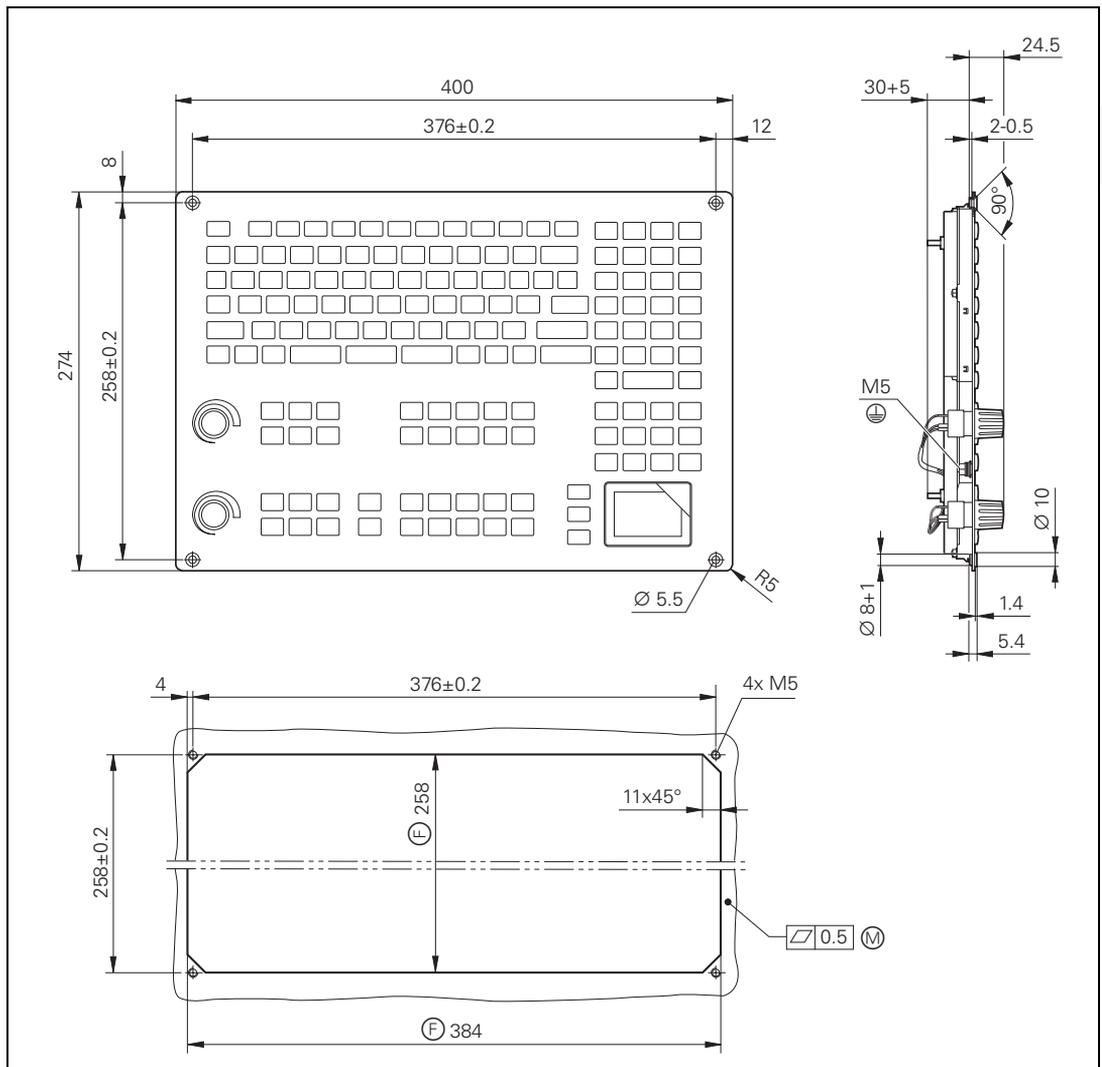
3.44.7 TE 635Q (FS)



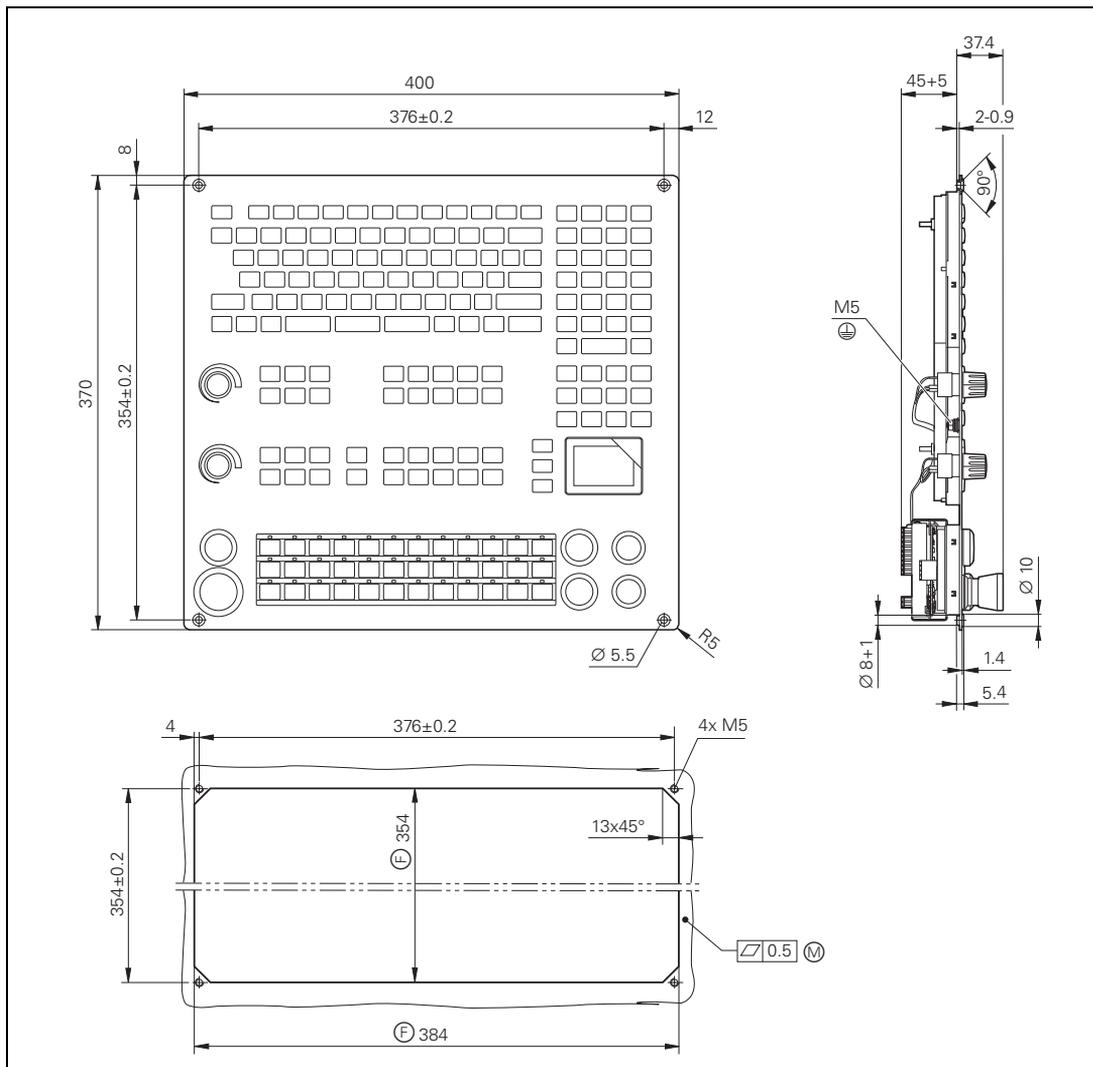
3.44.9 TE 720



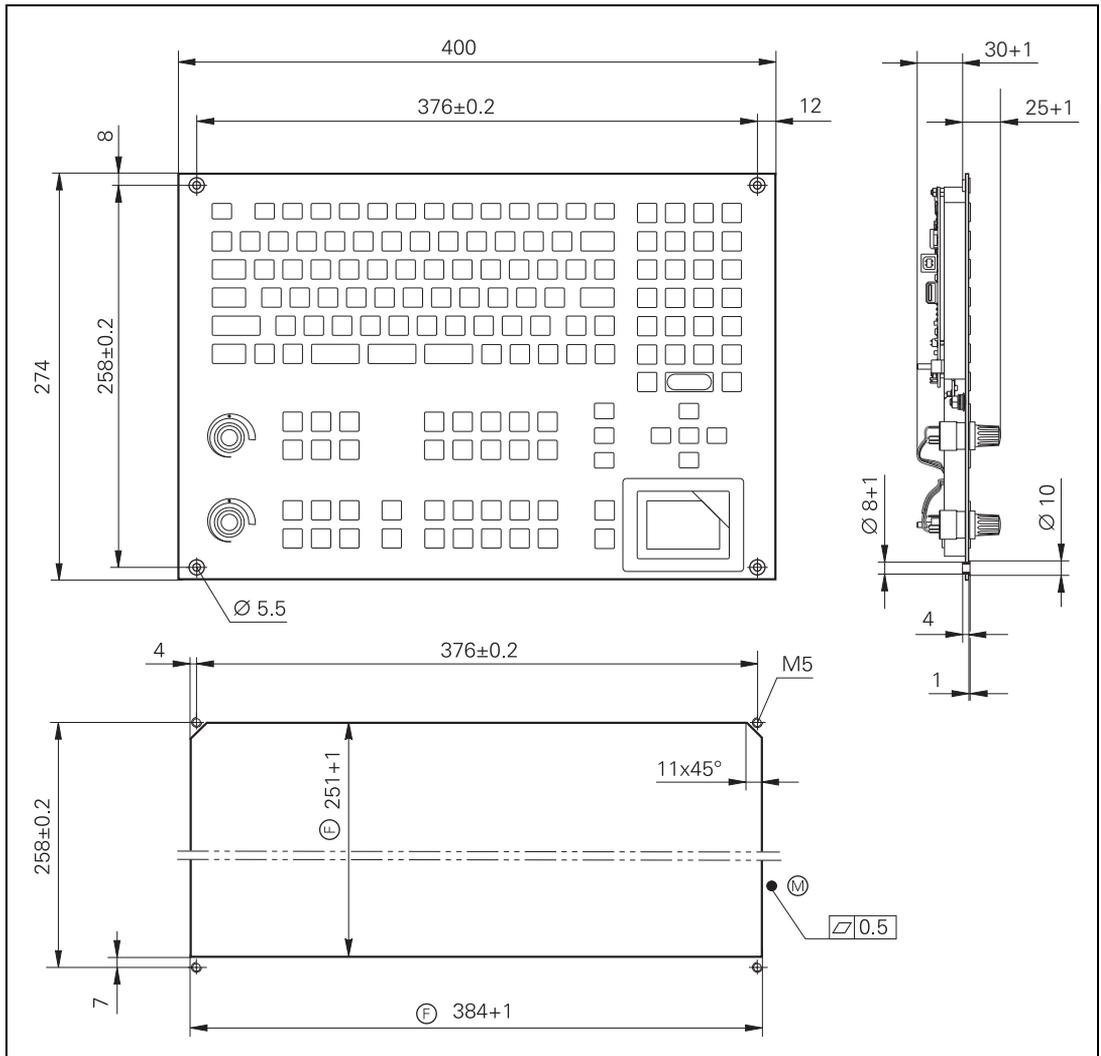
3.44.10 TE 730



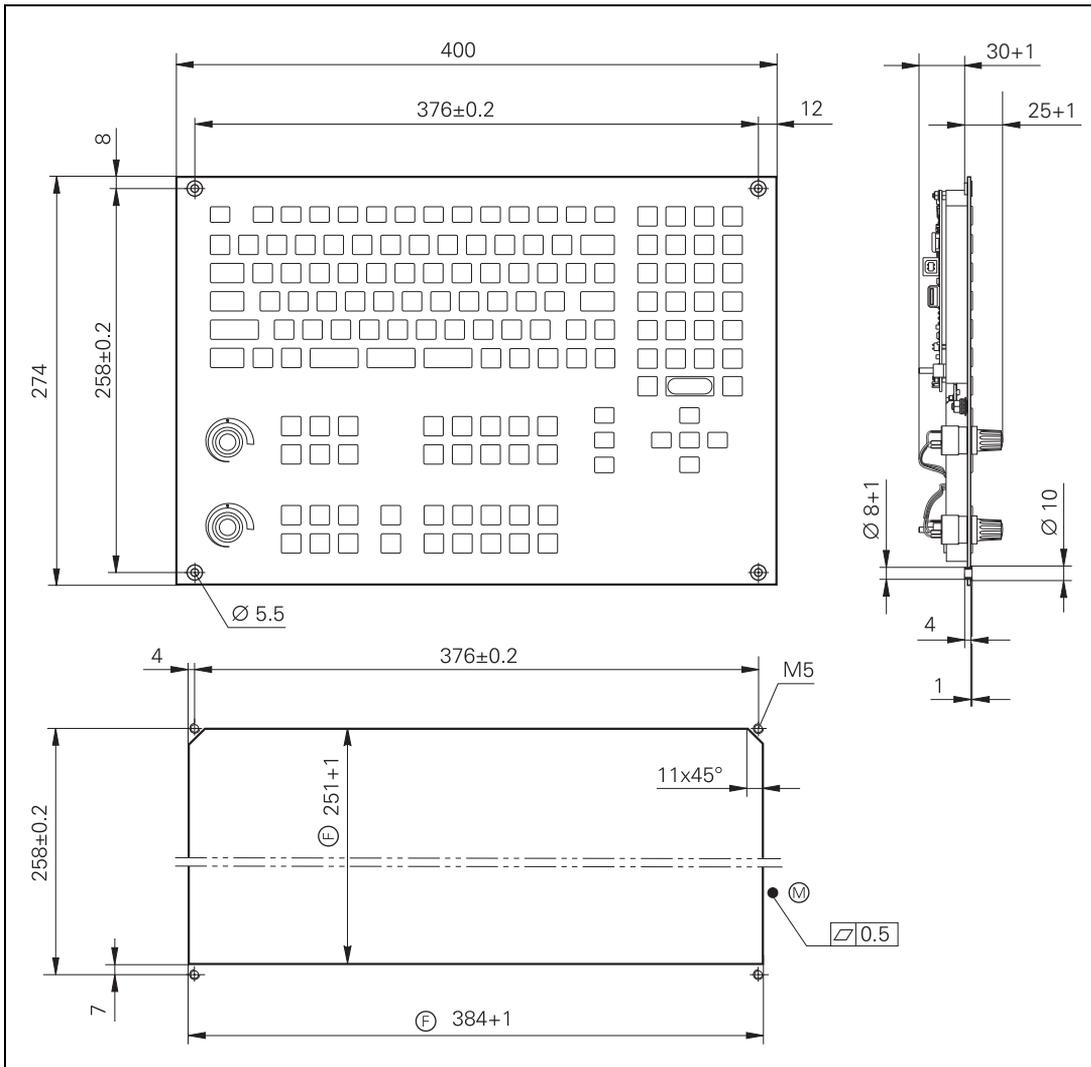
3.44.11 TE 735 (FS)



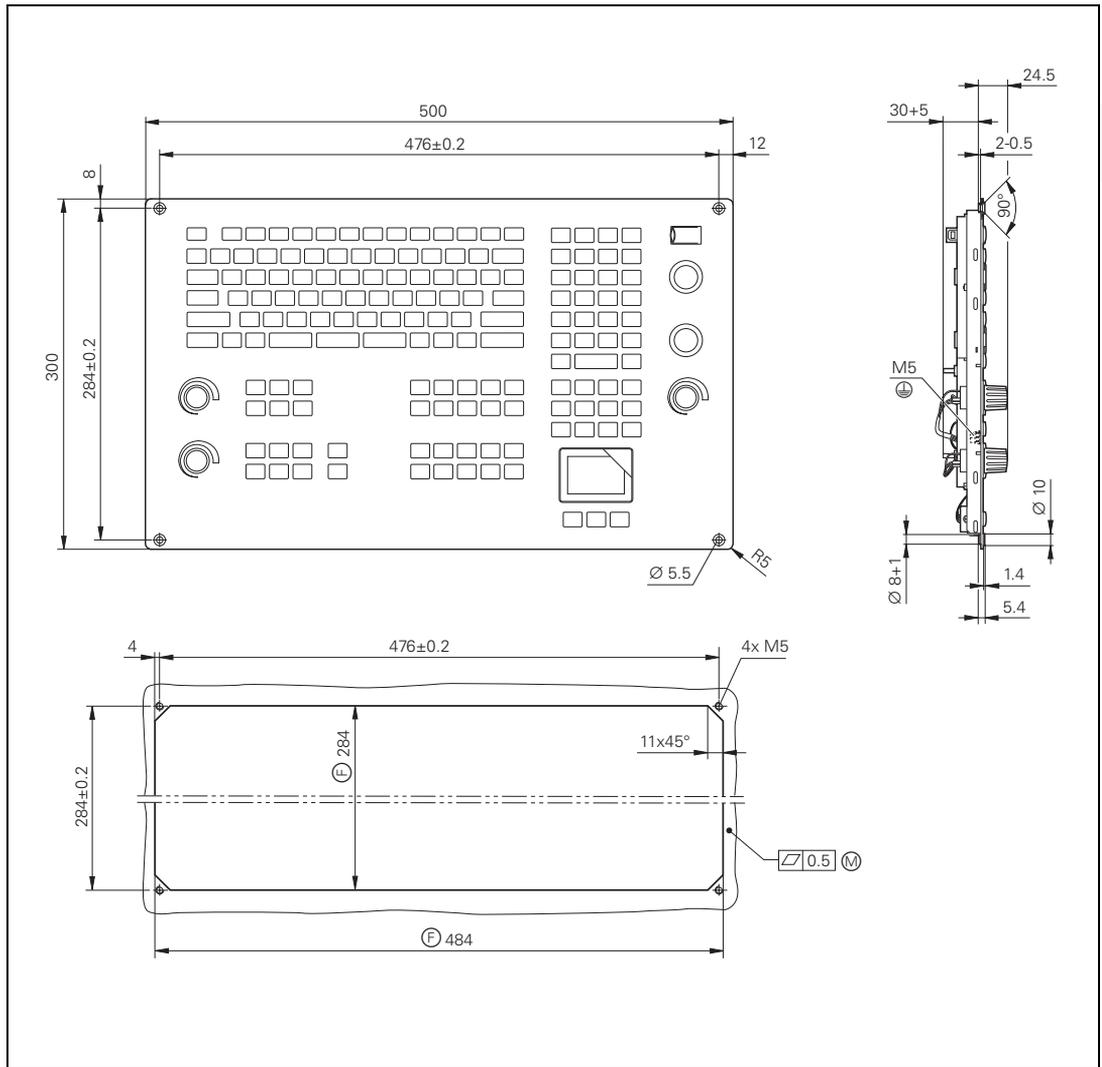
3.44.12 TE 630



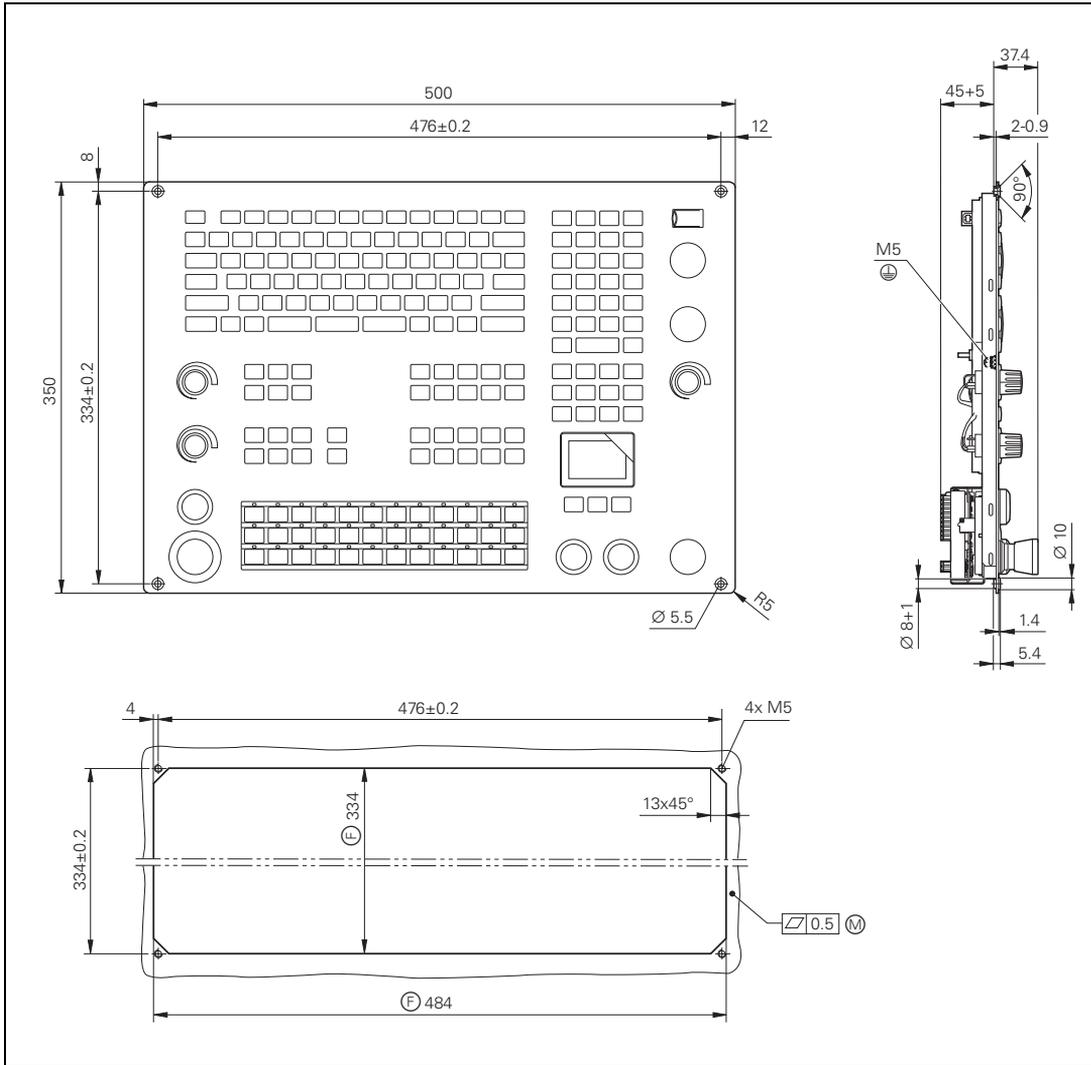
3.44.13 TE 620



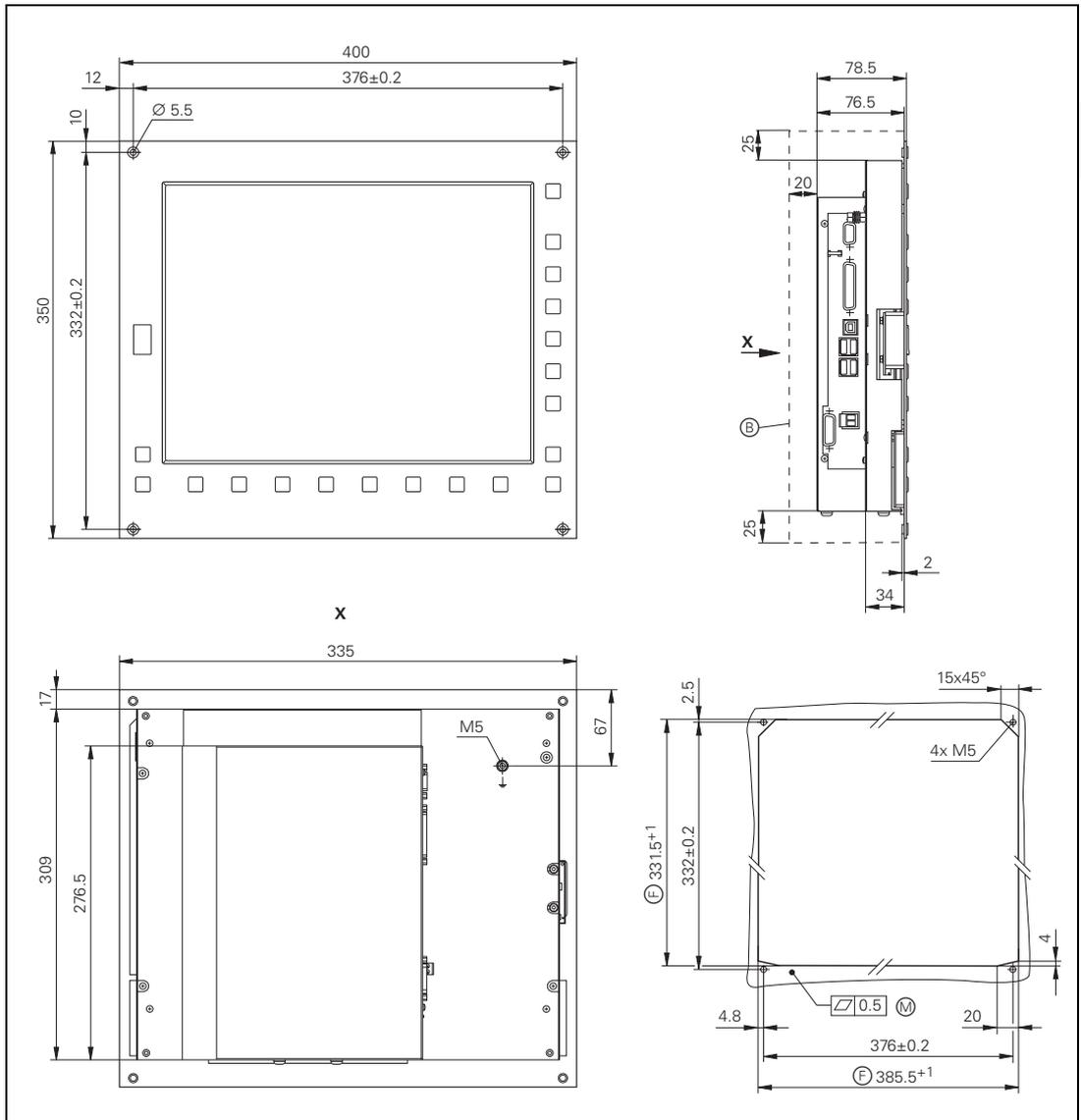
3.44.14 TE 740



3.44.15 TE 745



3.44.16 BF 250

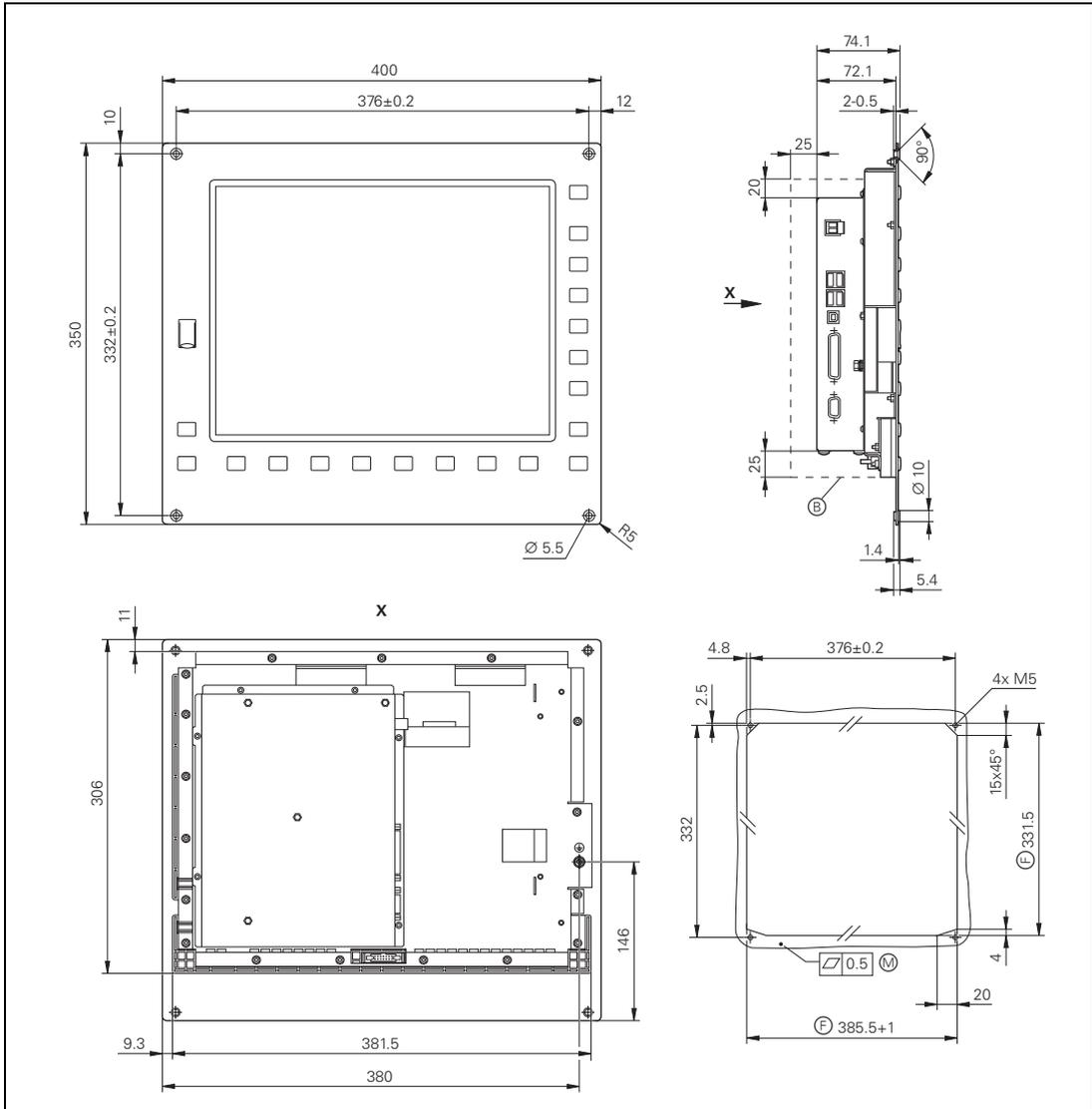


B = Space for air circulation

F = Front panel opening

M = Mounting surface

3.44.17 BF 750

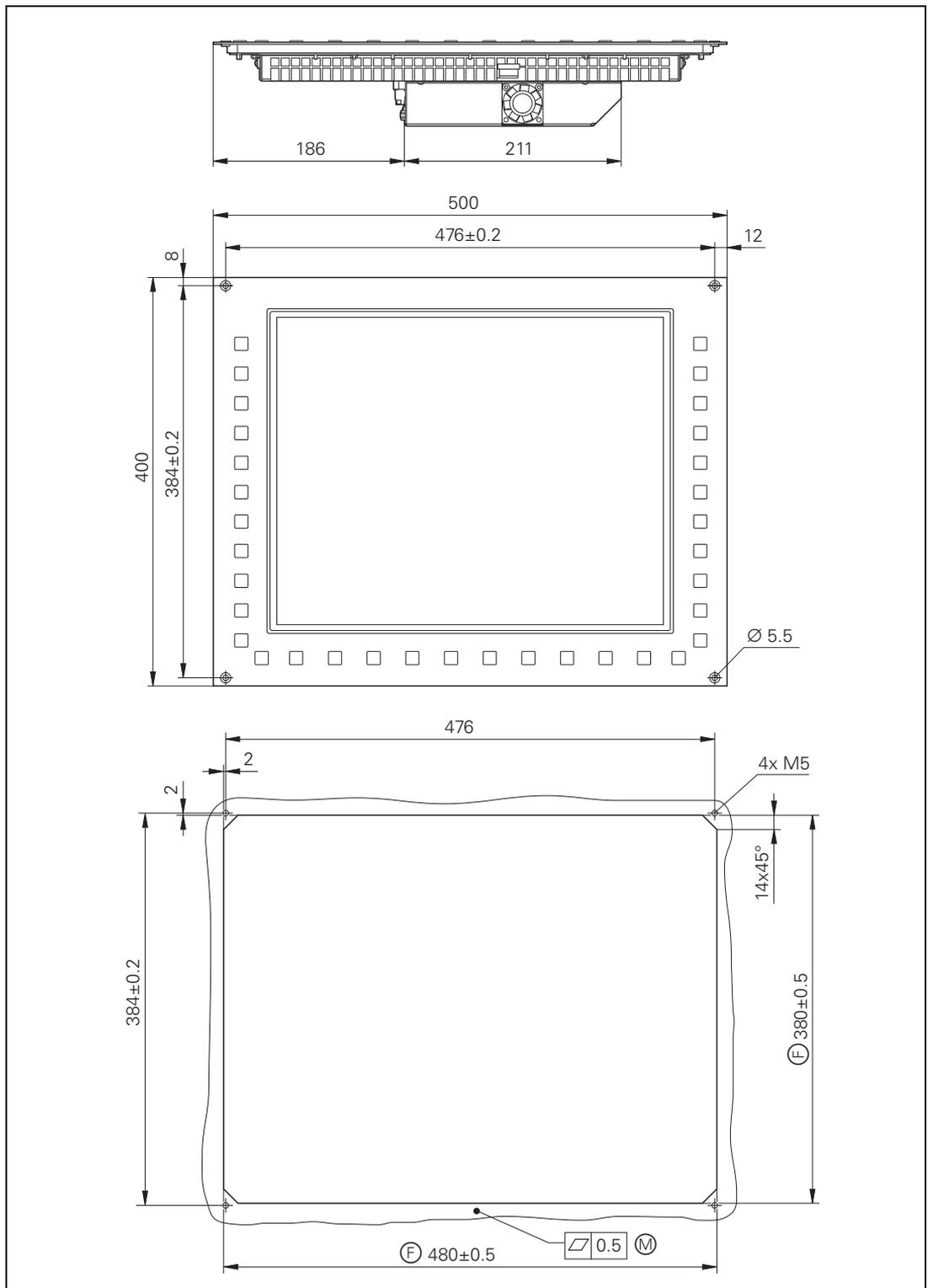


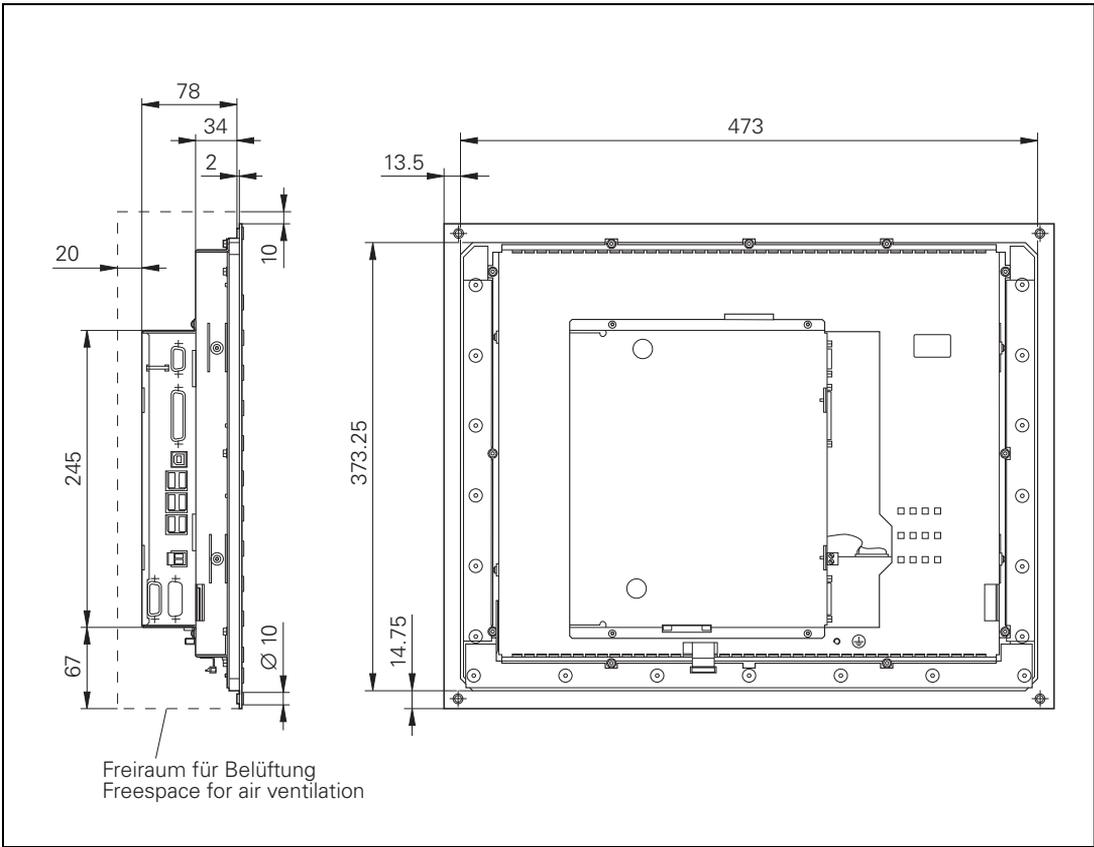
B = Space for air circulation

F = Front panel opening

M = Mounting surface

3.44.18 BF 260



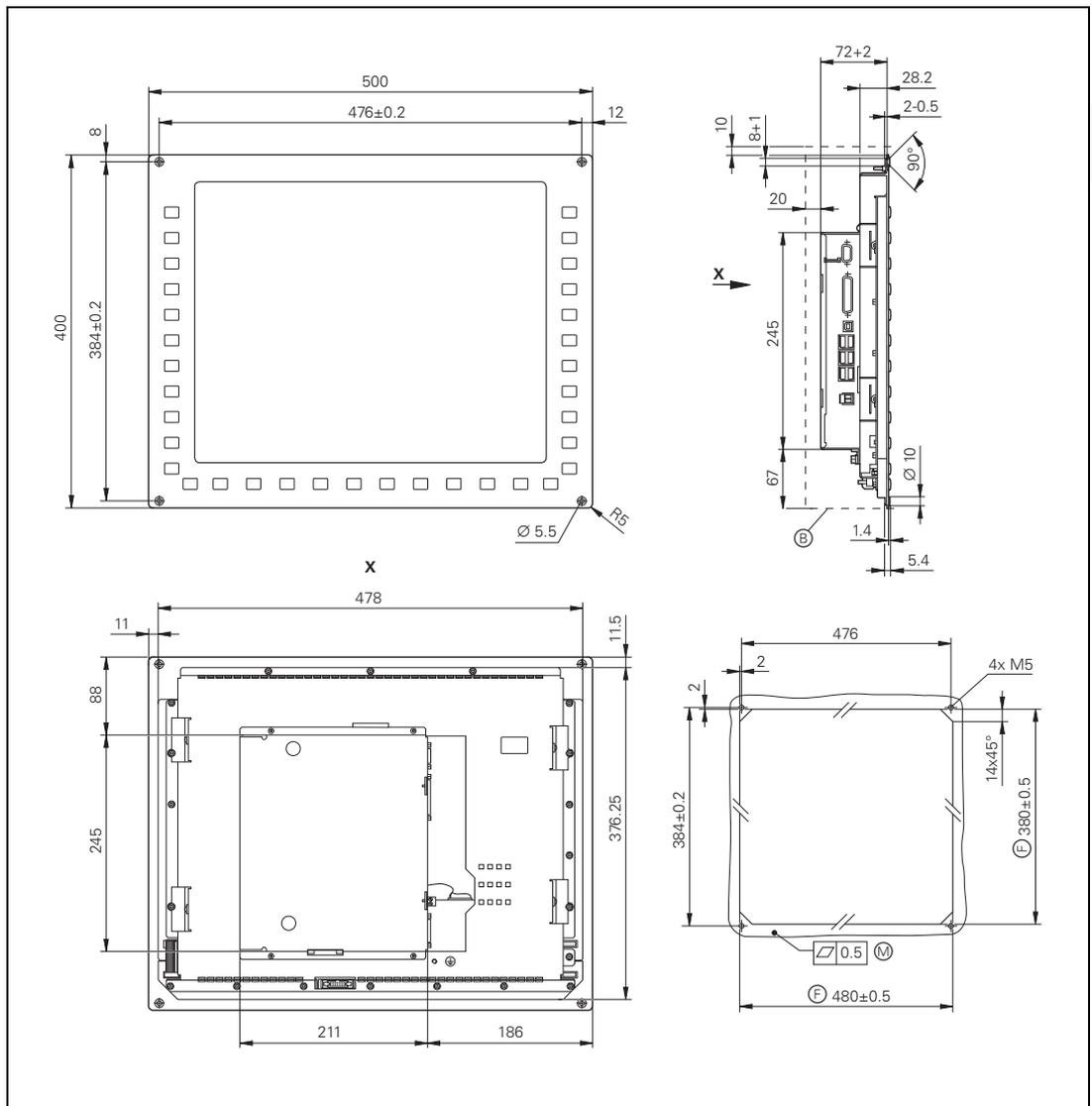


B = Space for air circulation

F = Front panel opening

M = Mounting surface

3.44.19 BF 760

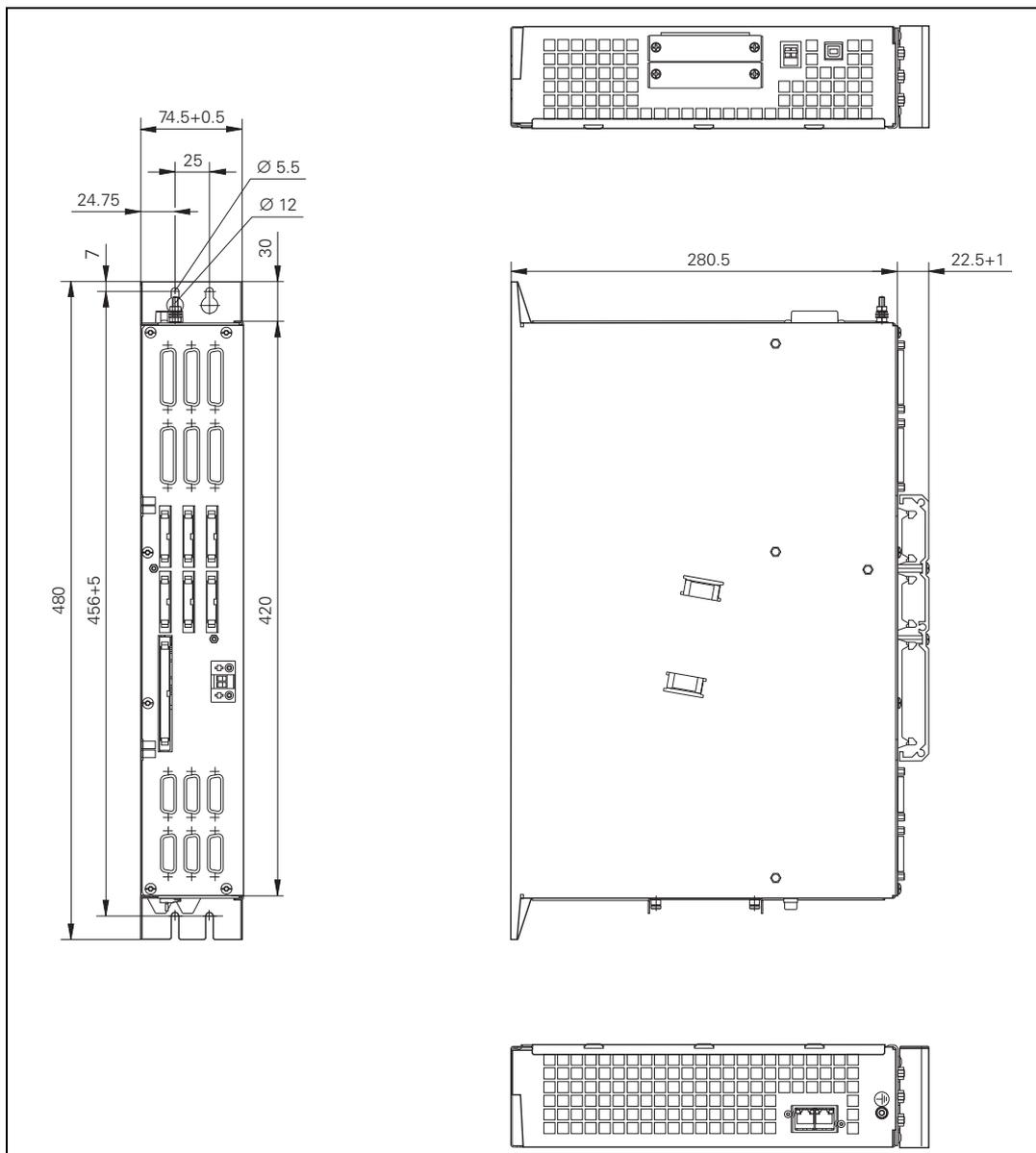


B = Space for air circulation

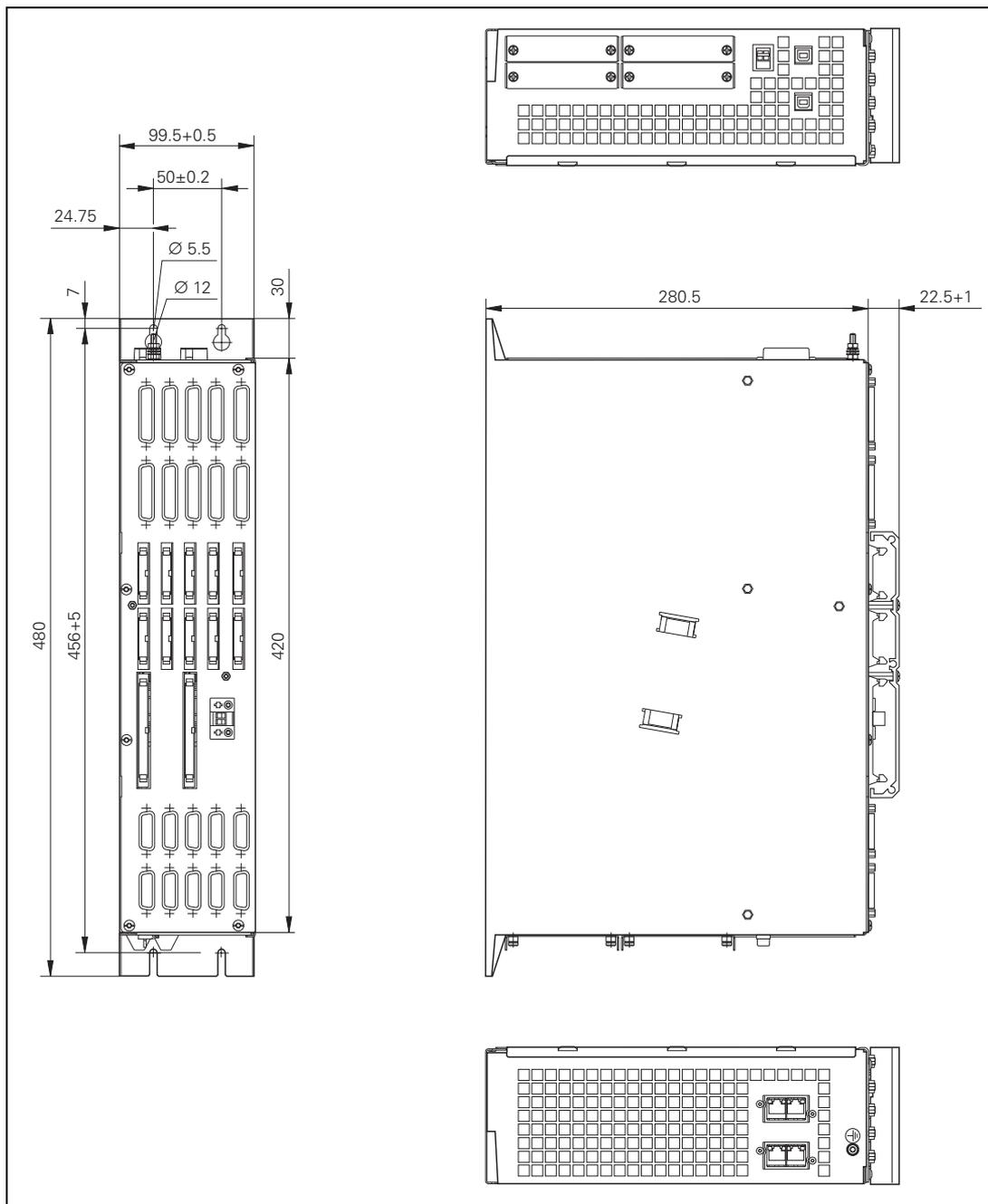
F = Front panel opening

M = Mounting surface

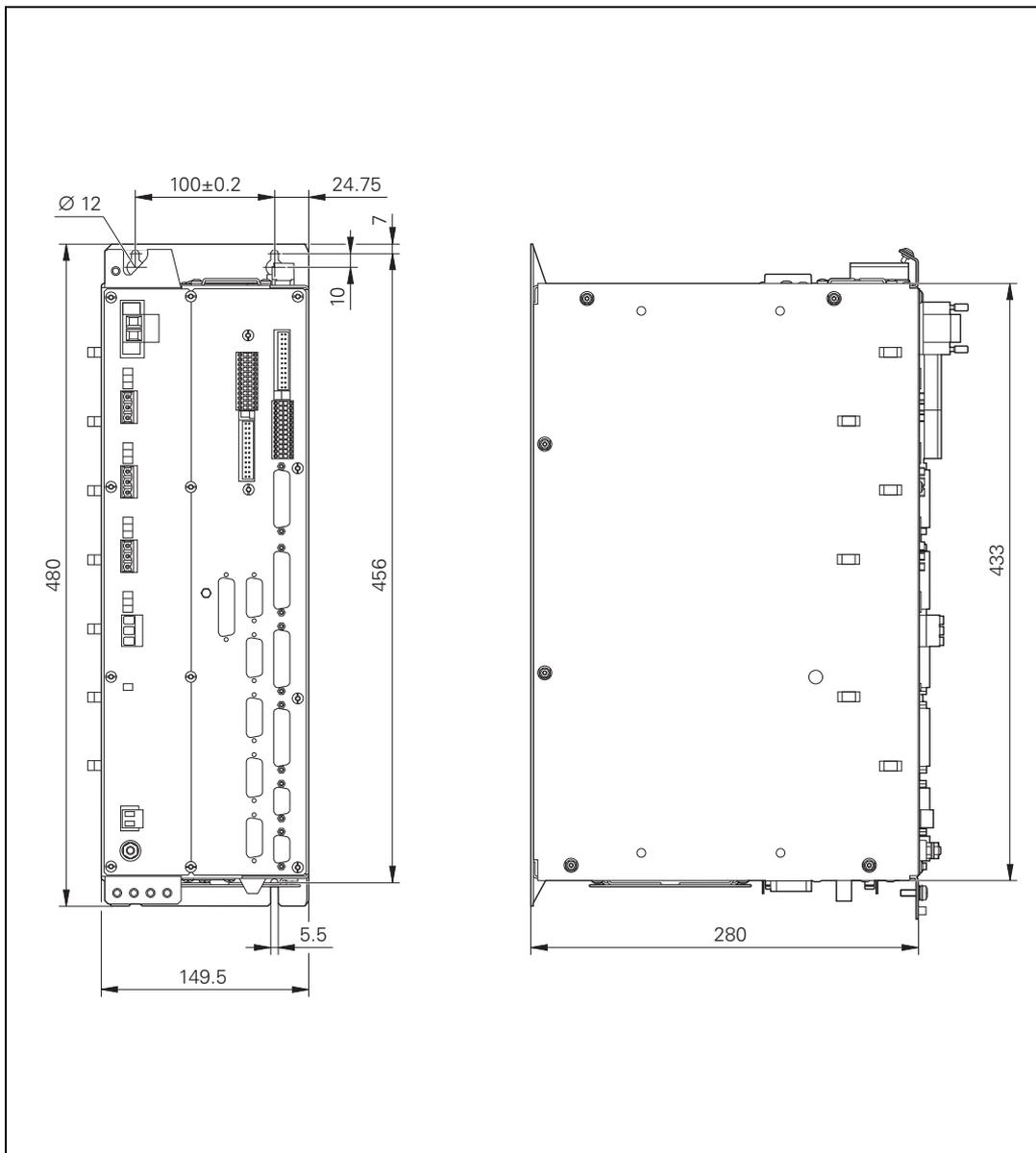
3.44.20 CC 6106



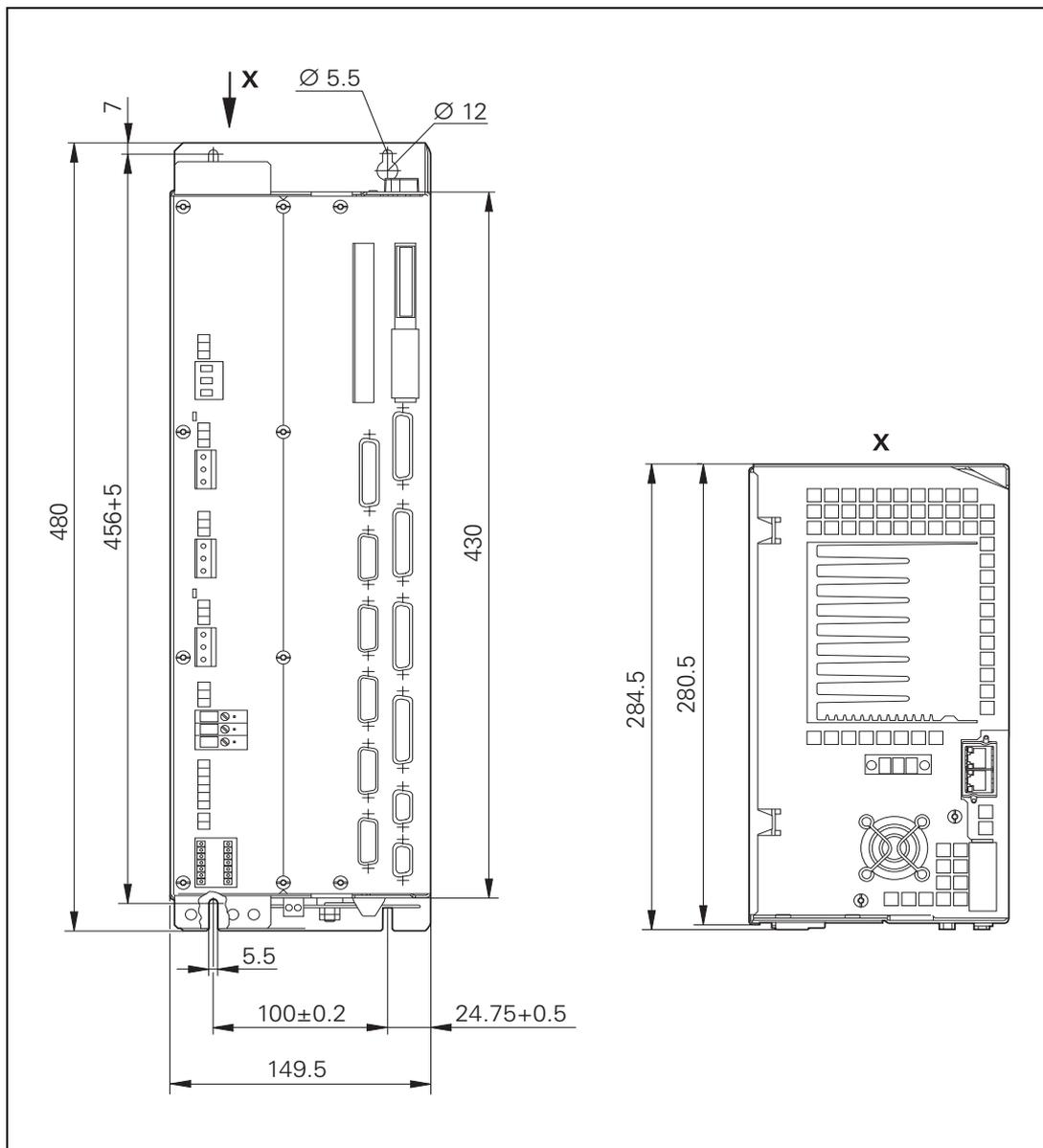
3.44.21 CC 6108 / CC 6110



3.44.22 UMC 111 (FS)

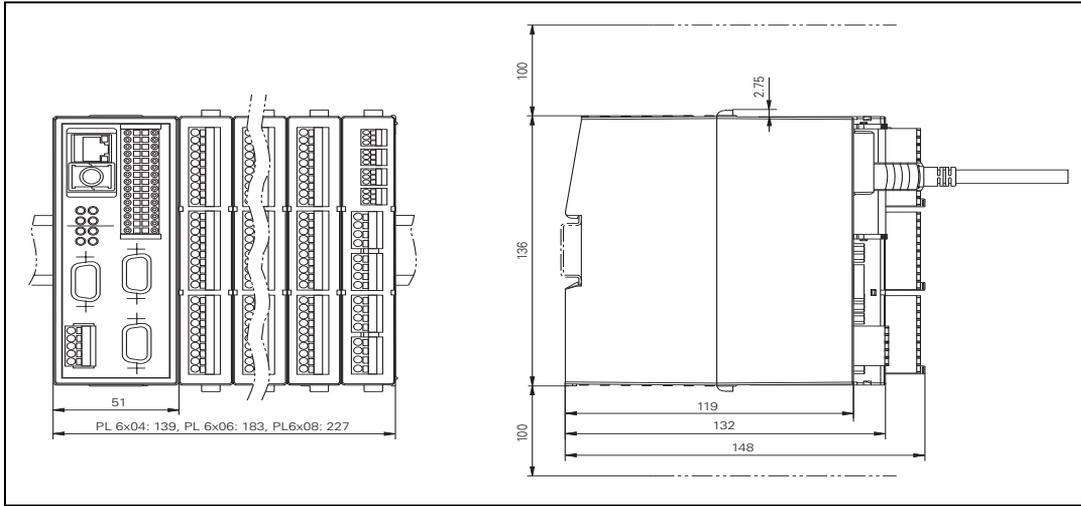


3.44.23 UEC 11x (FS)



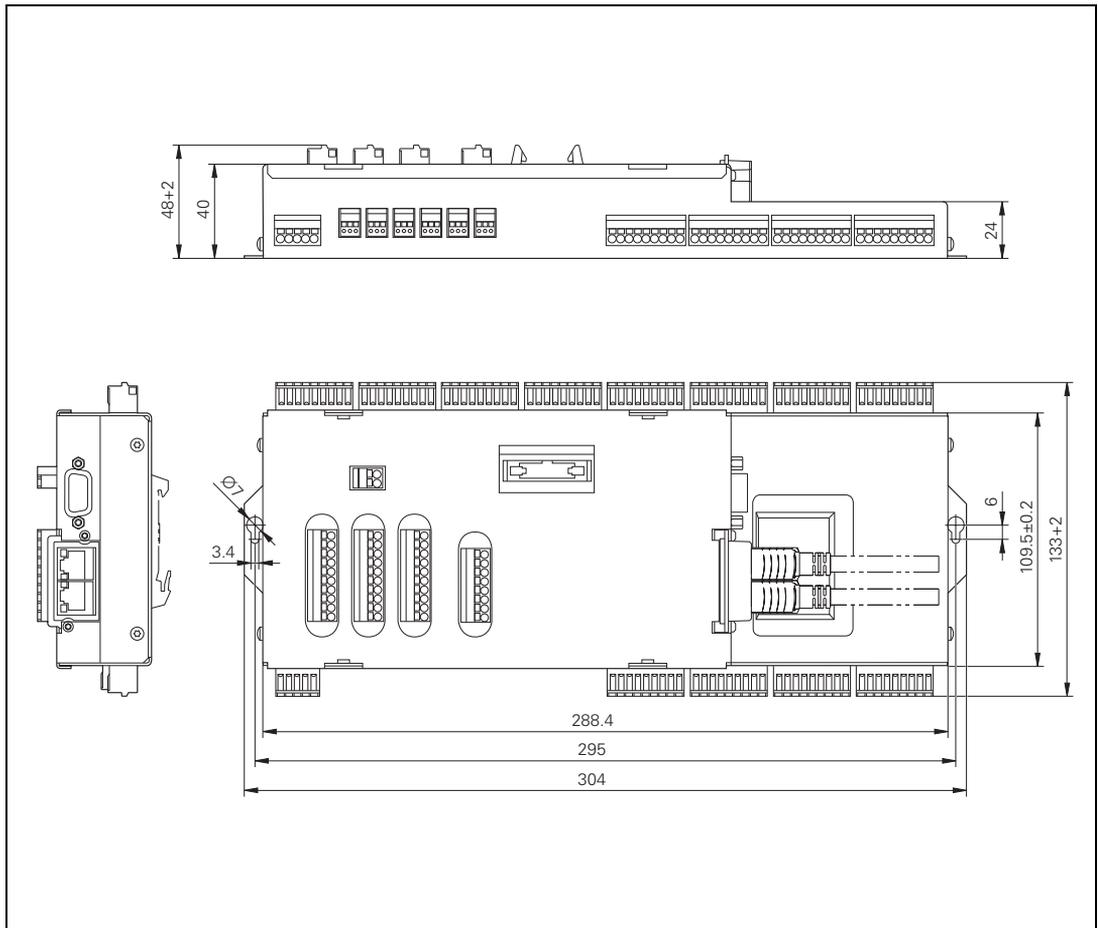
3.44.24 PL 6xxx (FS)

Weight: 1.0 kg

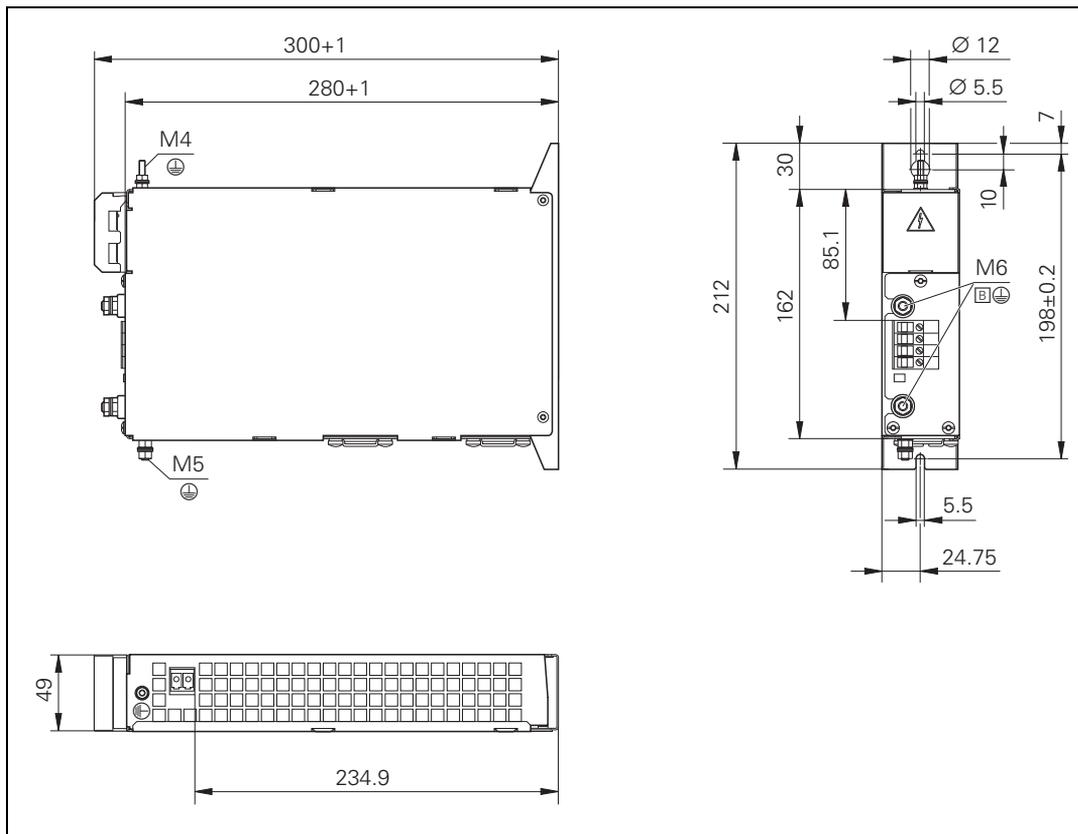


Dashed line: Space for air circulation

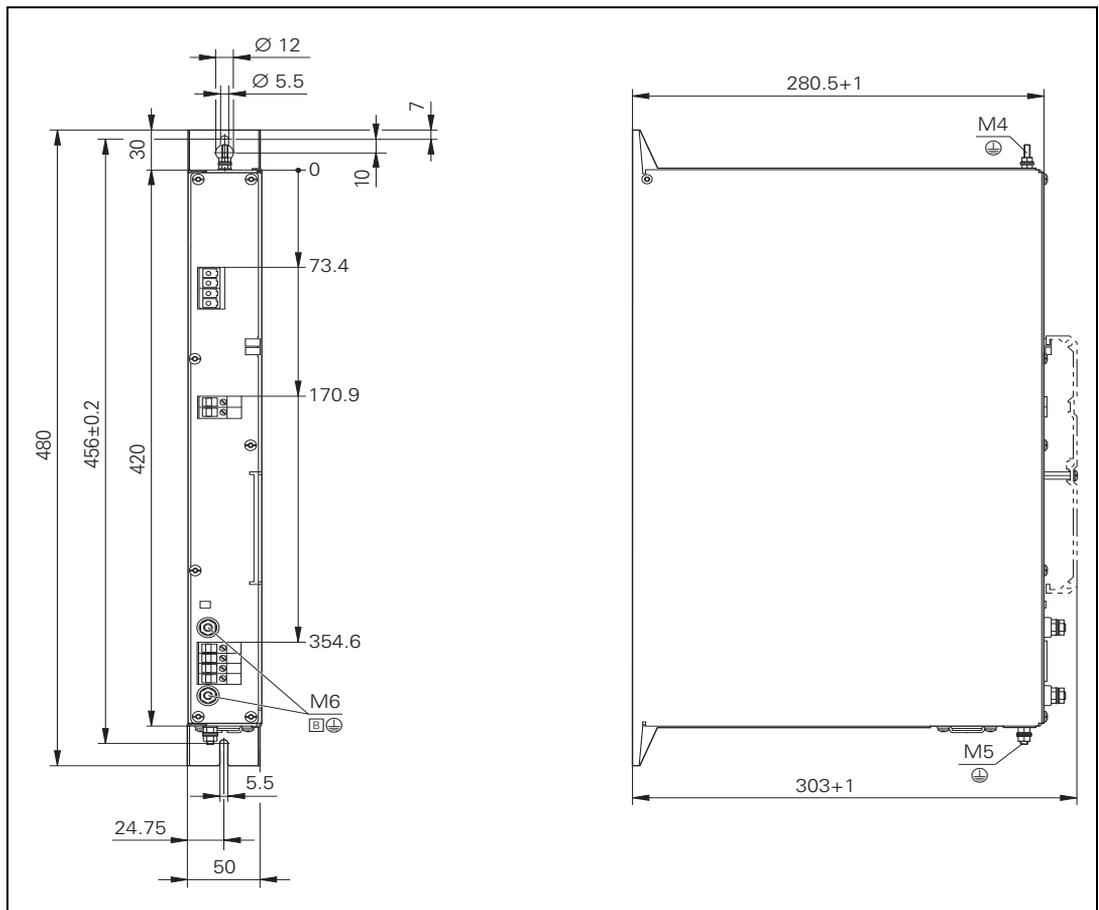
3.44.25 PLB 6001(FS)



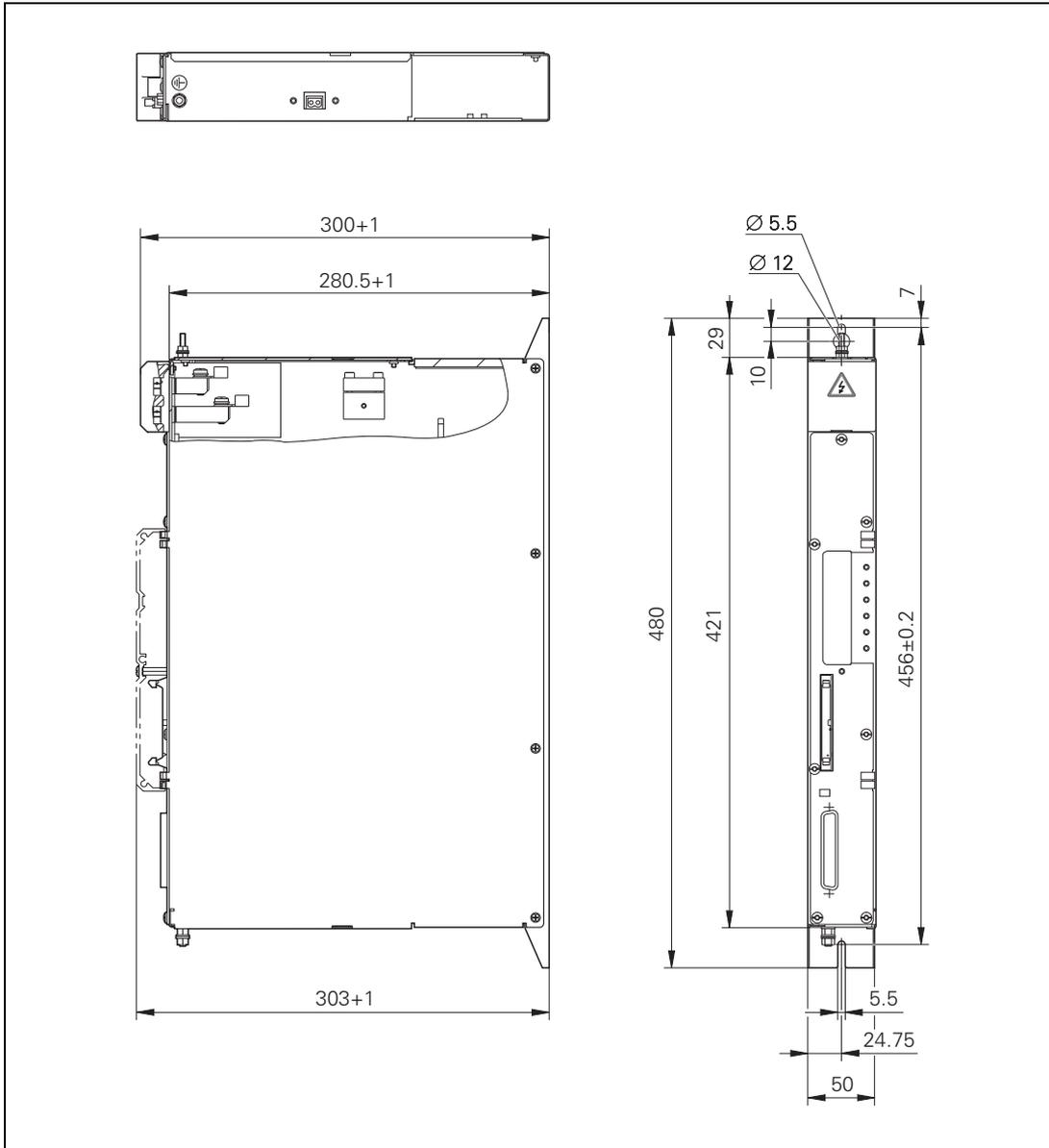
3.44.26 PSL 130



3.44.27 PSL 135

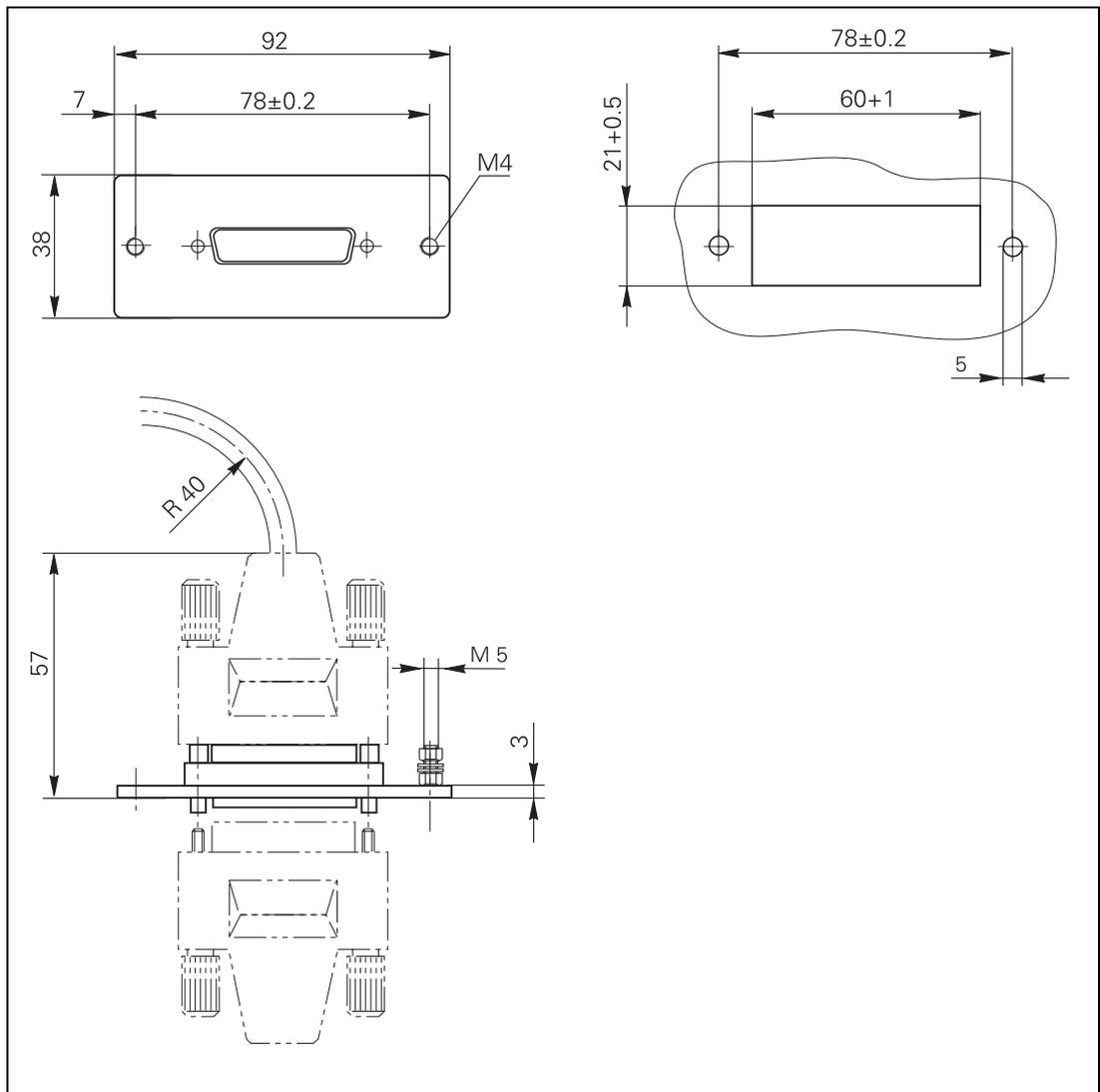


3.44.28 MS 11x



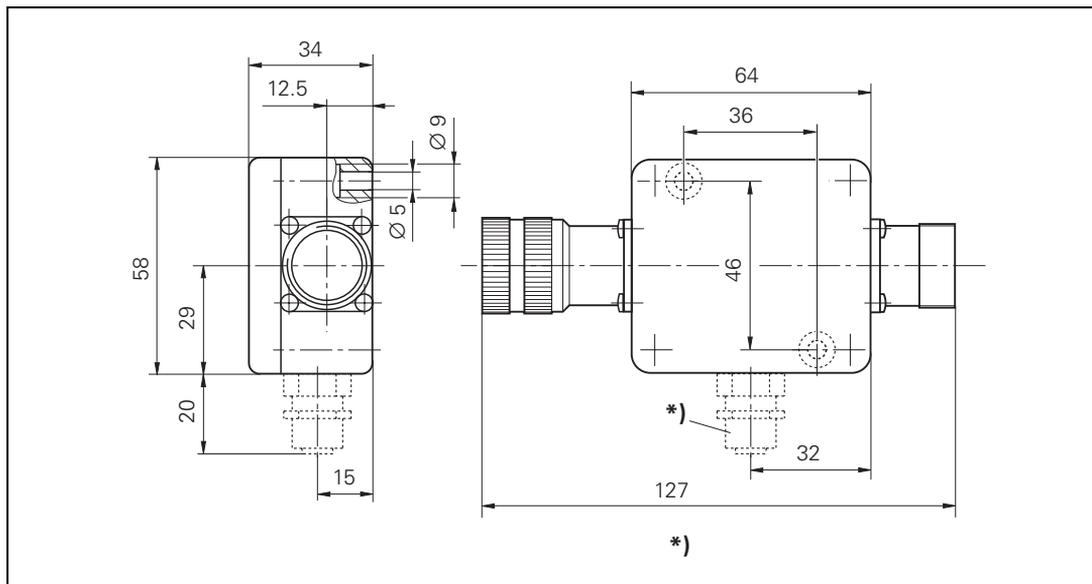
3.44.29 Adapter block for the data interface

RS-232-C/V.24 adapter block and
RS-422/V.11 adapter block



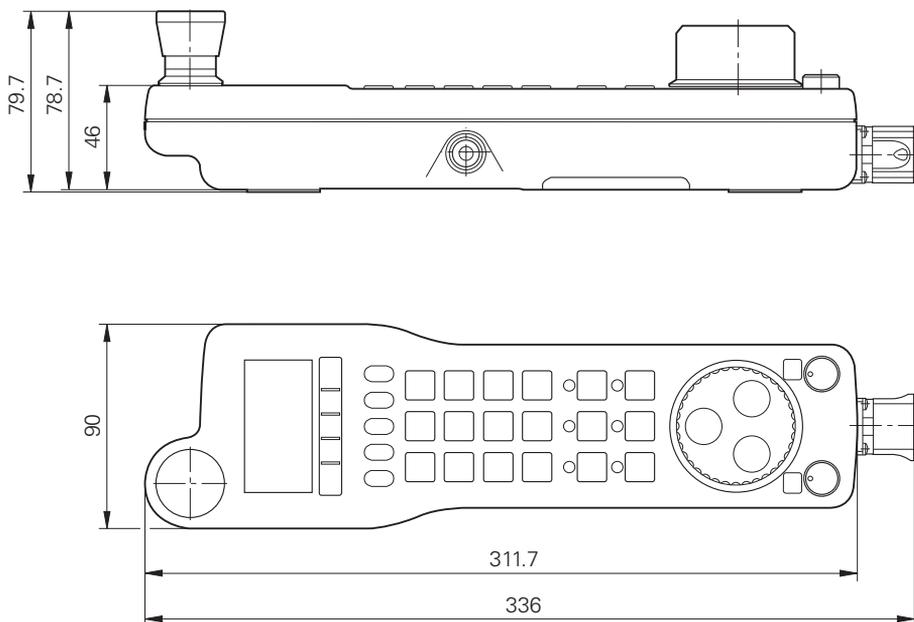
3.44.31 Line-drop compensator

Line-drop compensator for encoders with EnDat interface

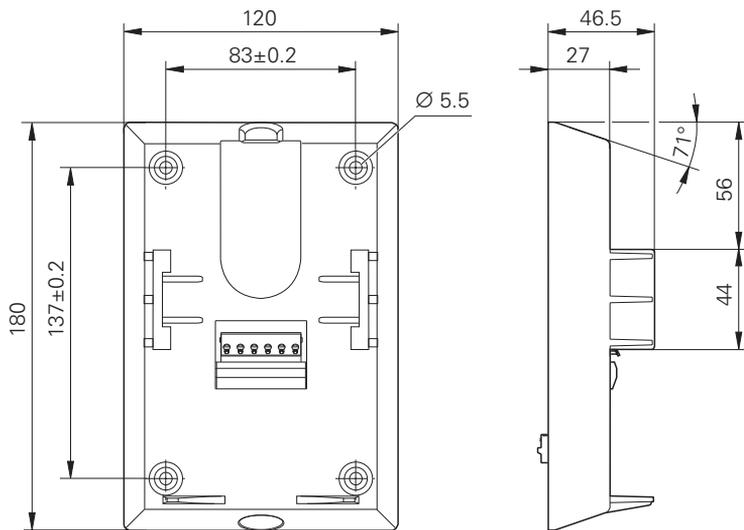


3.44.32 Handwheels

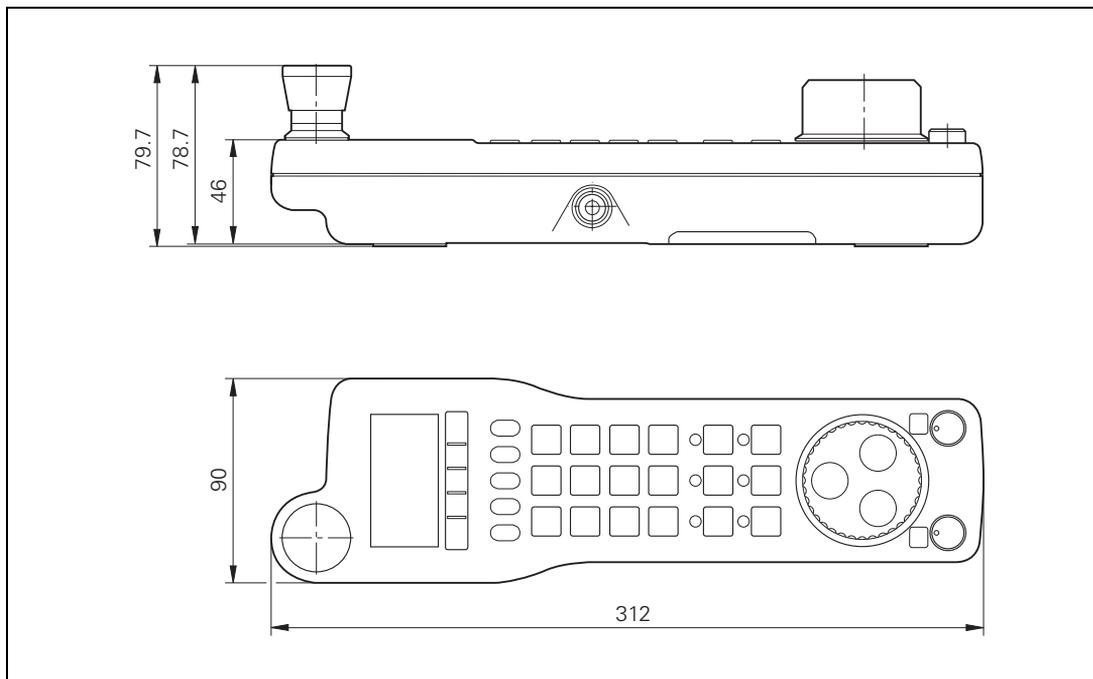
HR 520

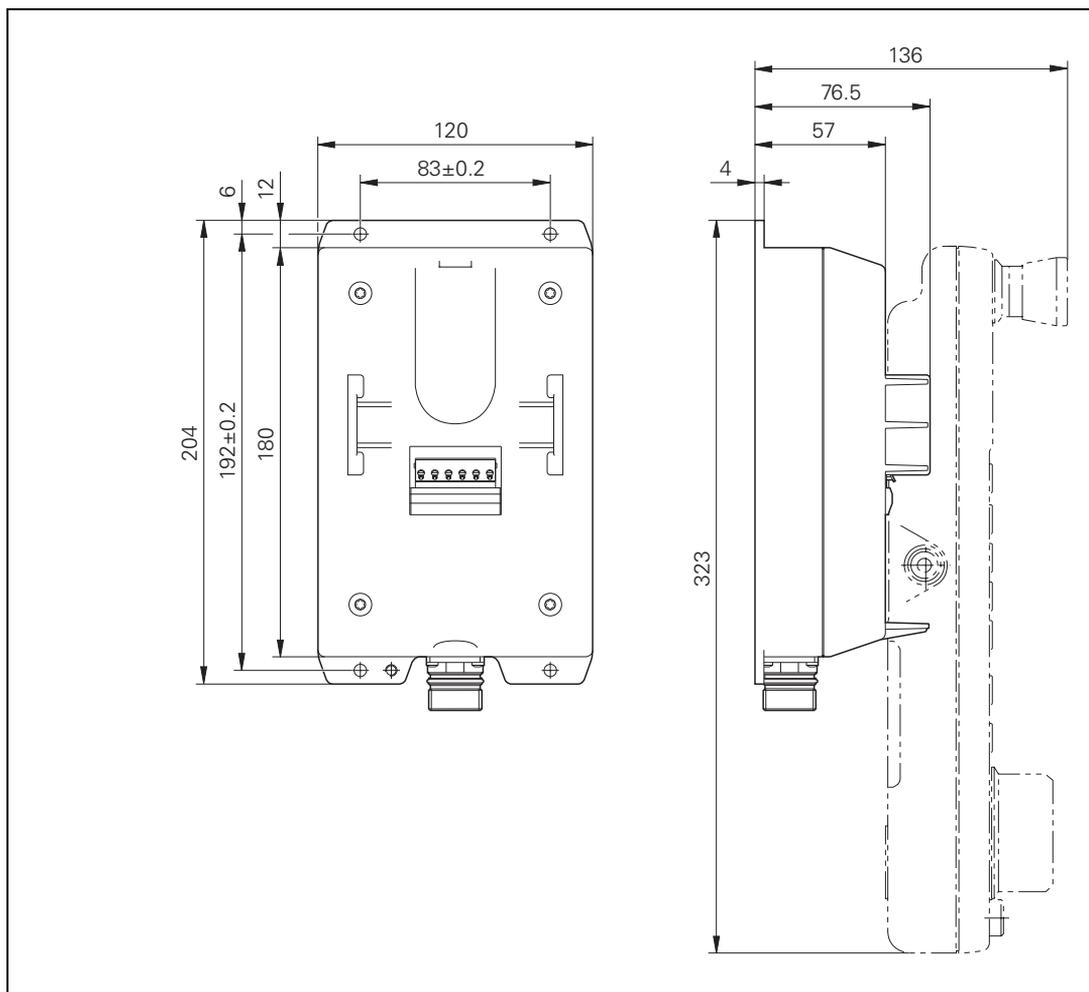


Mount for HR 520

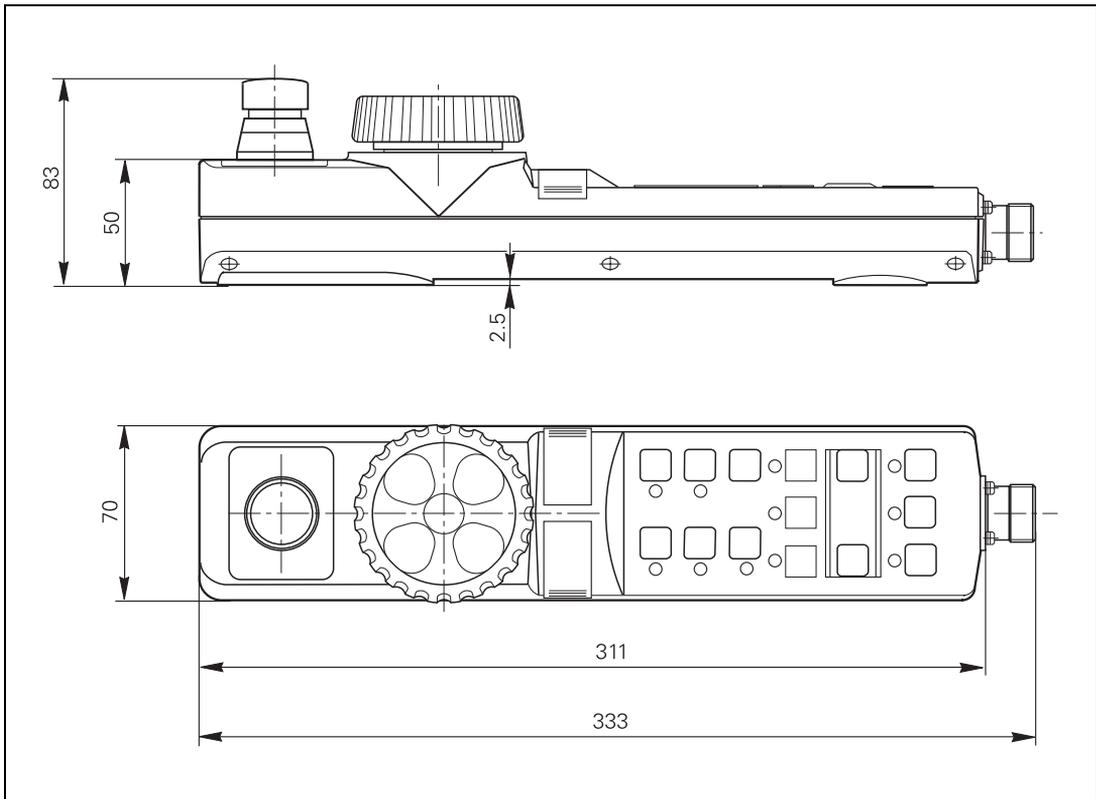


HR 550FS



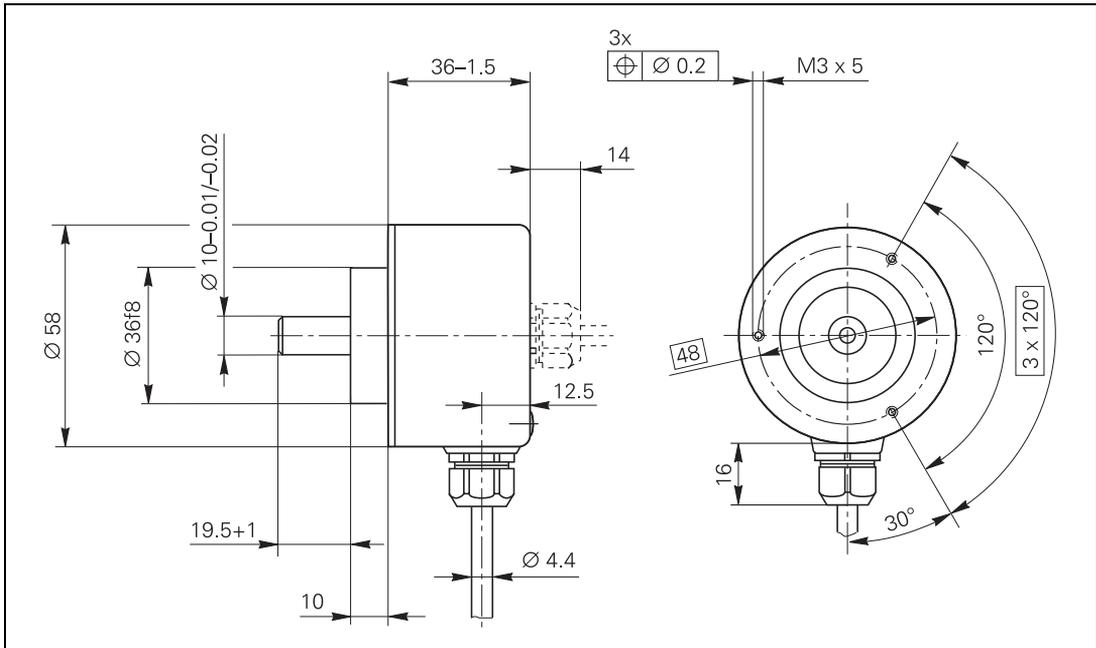


HR 410

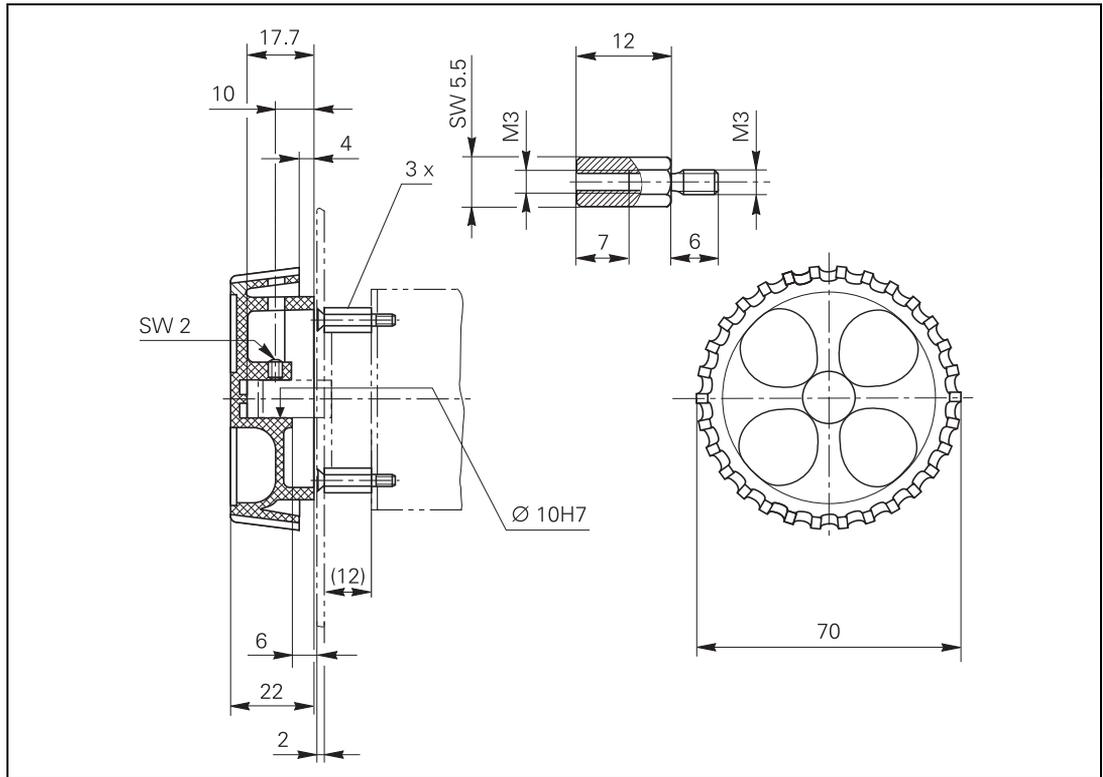


HR 130

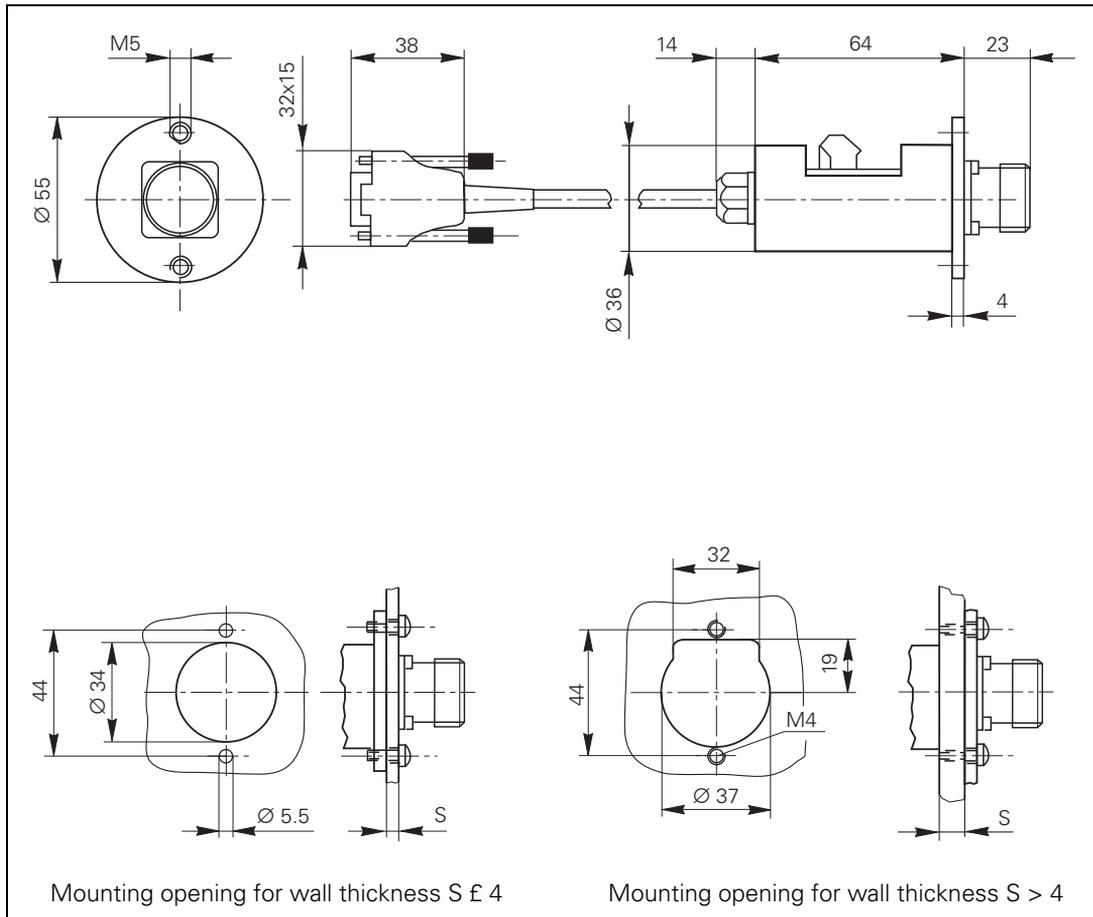
Weight: Approx. 0.7 kg



**Control knob for
HR 130**

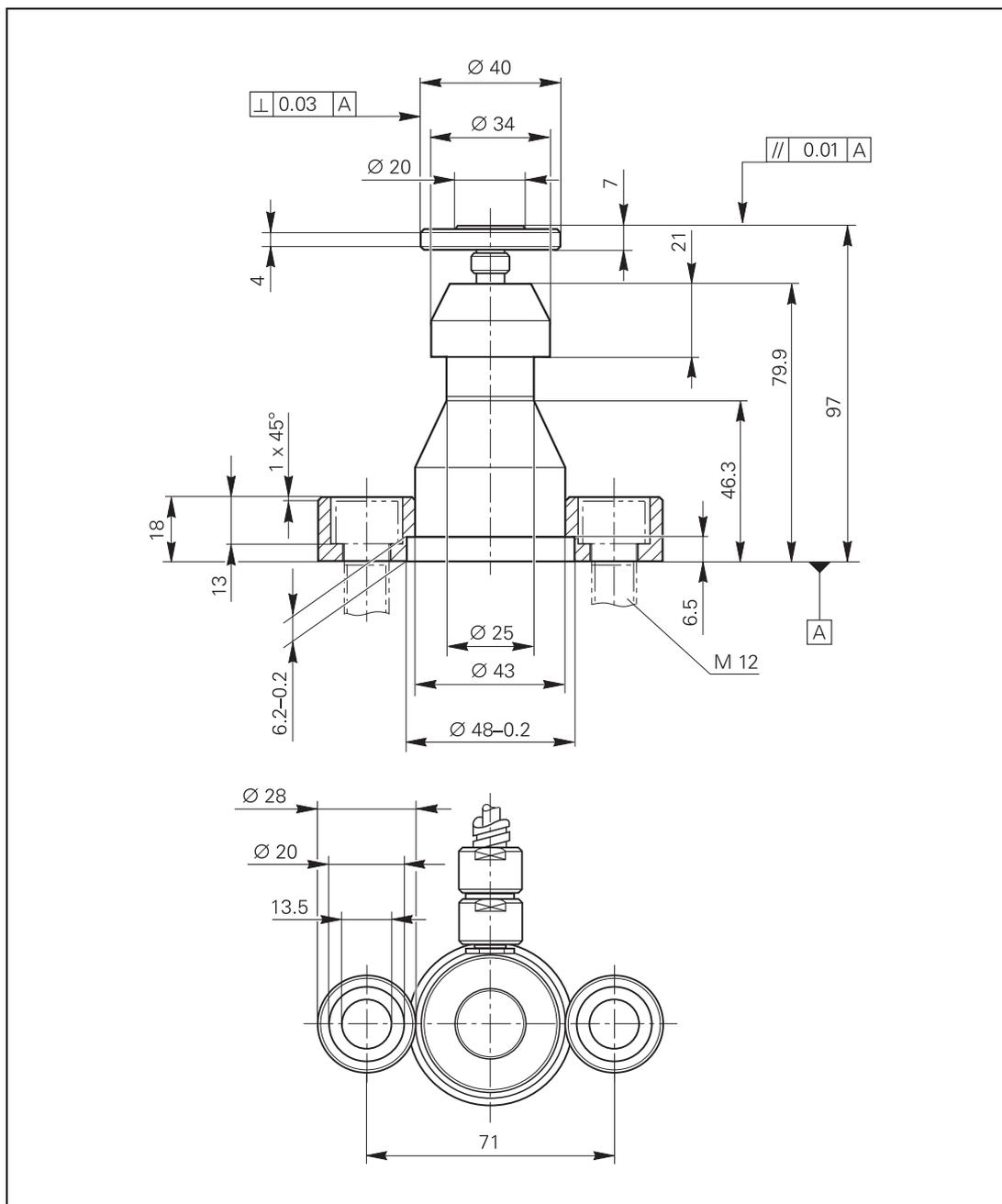


Adapter cable



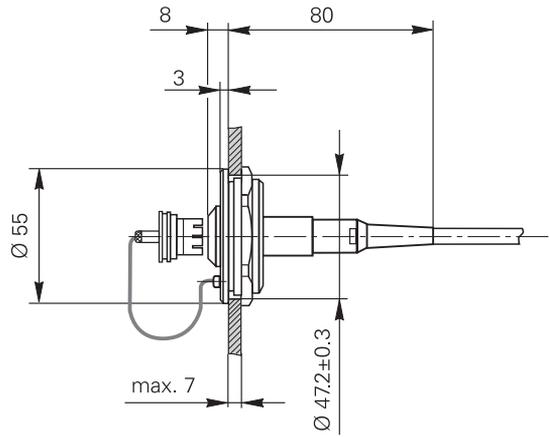
3.44.33 Touch probes

TT 130 / TT 140

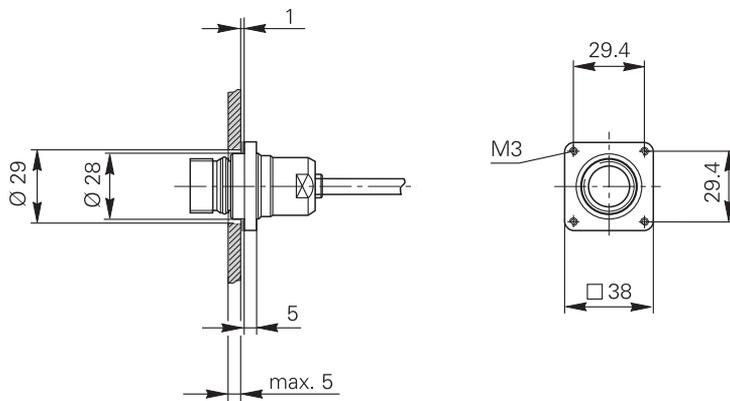


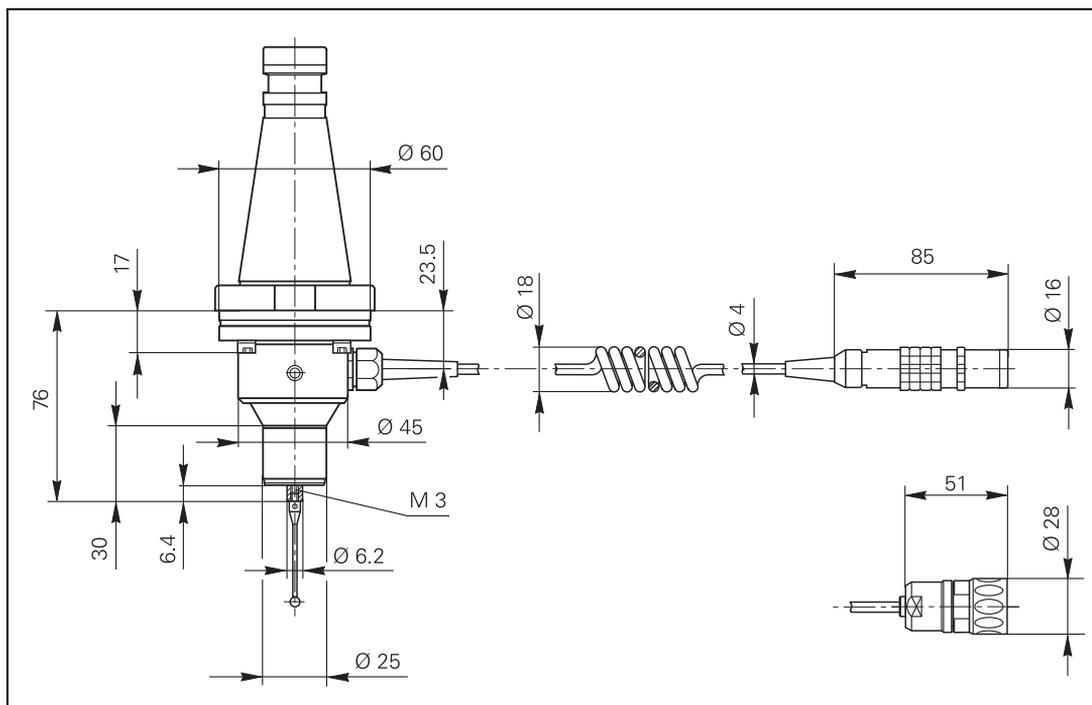
**Adapter cable for
TT and TS**

Mounted coupling for quick connection

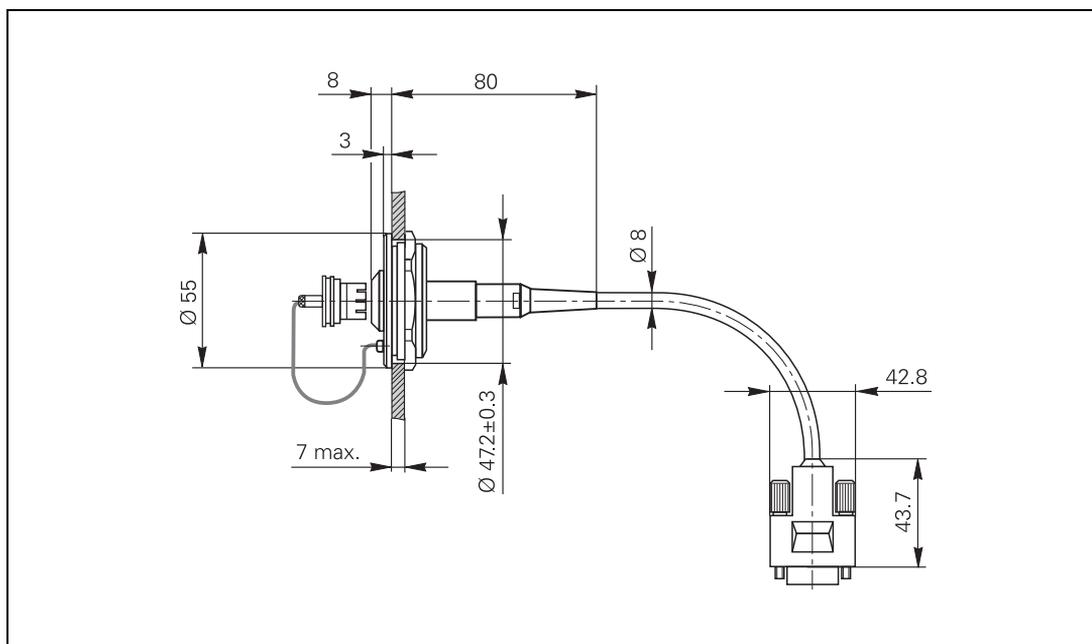


Mounted coupling for HEIDENHAIN standard connector:

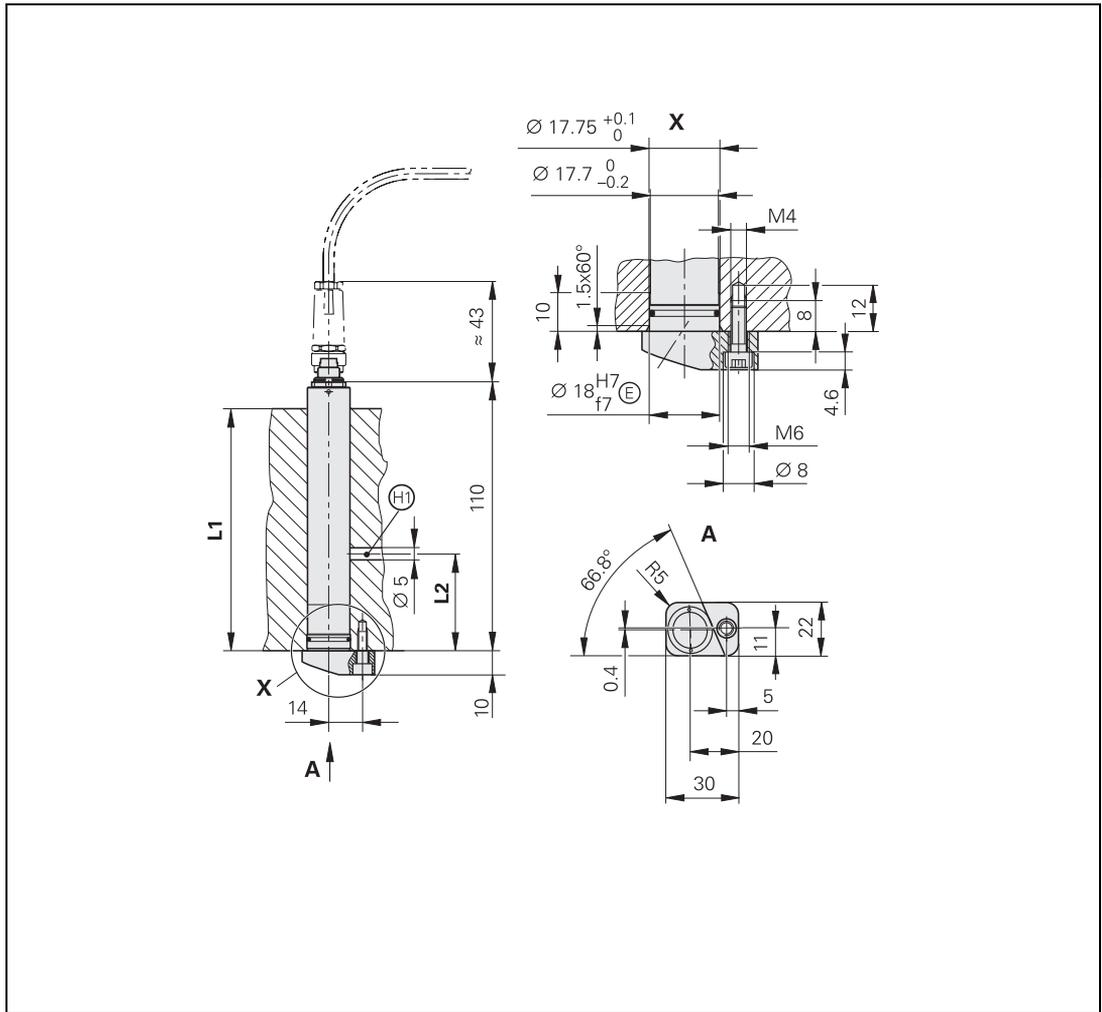


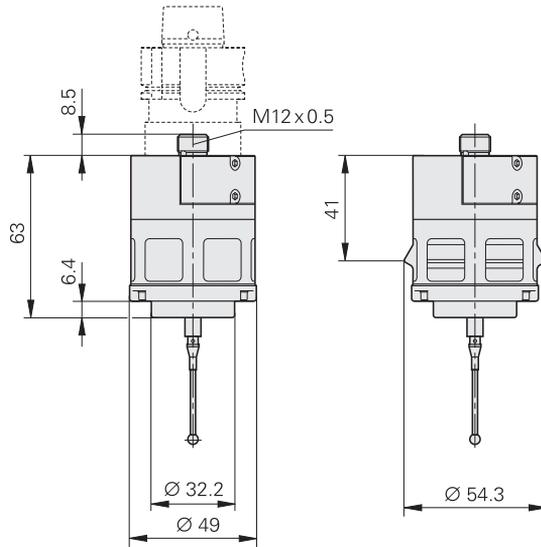


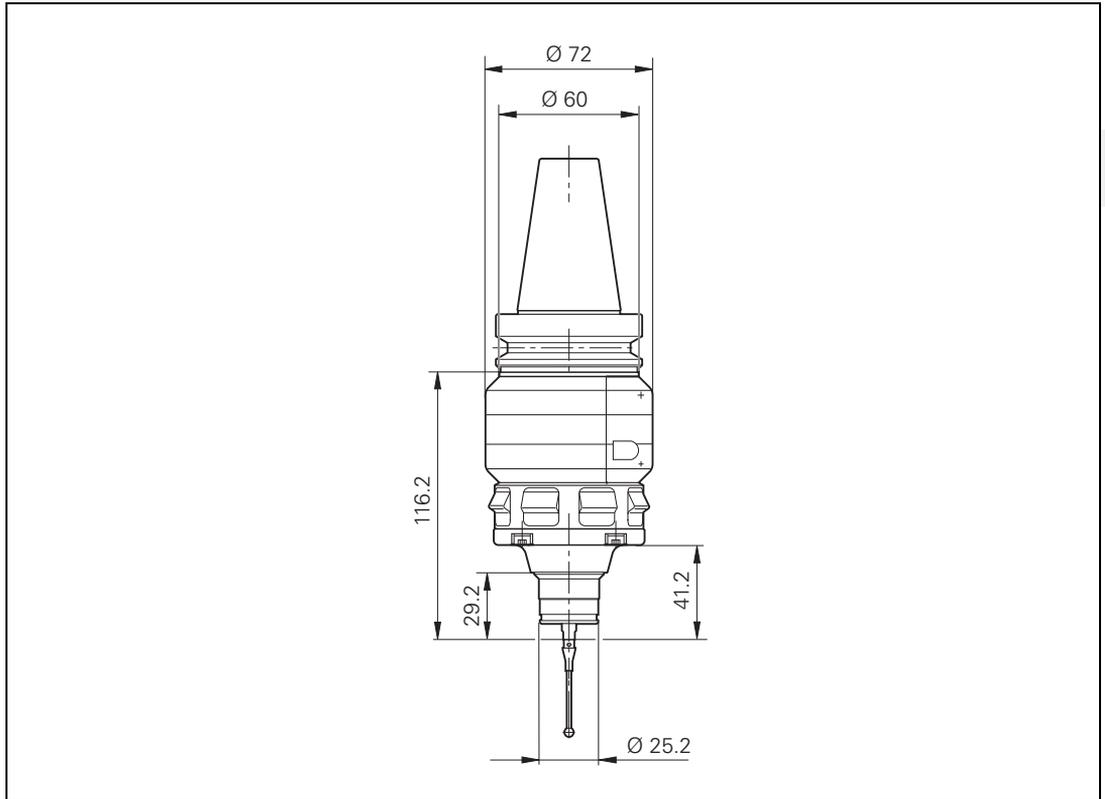
Adapter cable for
TS 120/TS 220



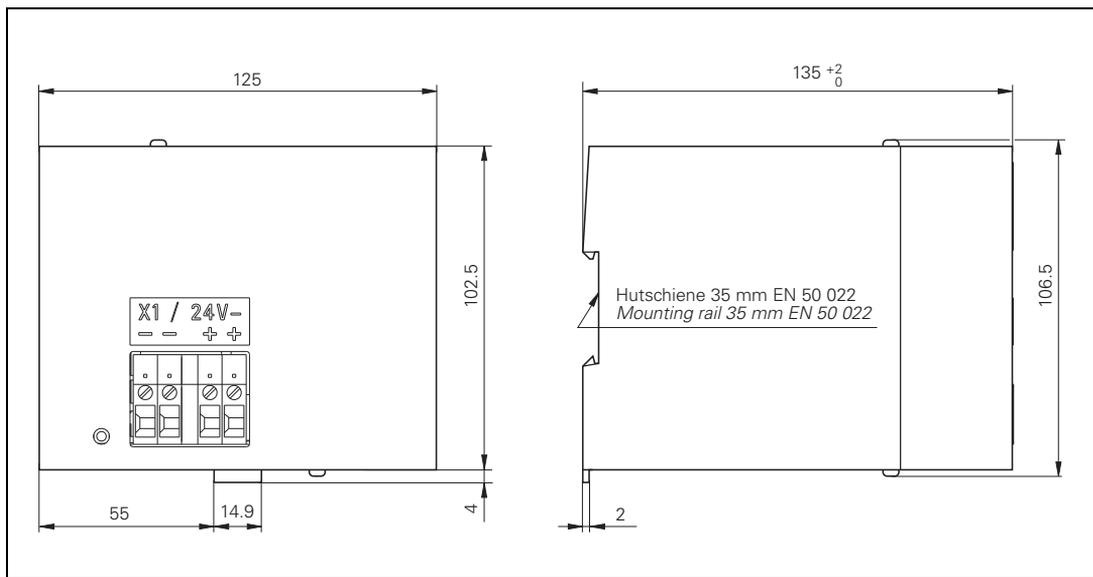
SE 540 transmitter-receiver unit



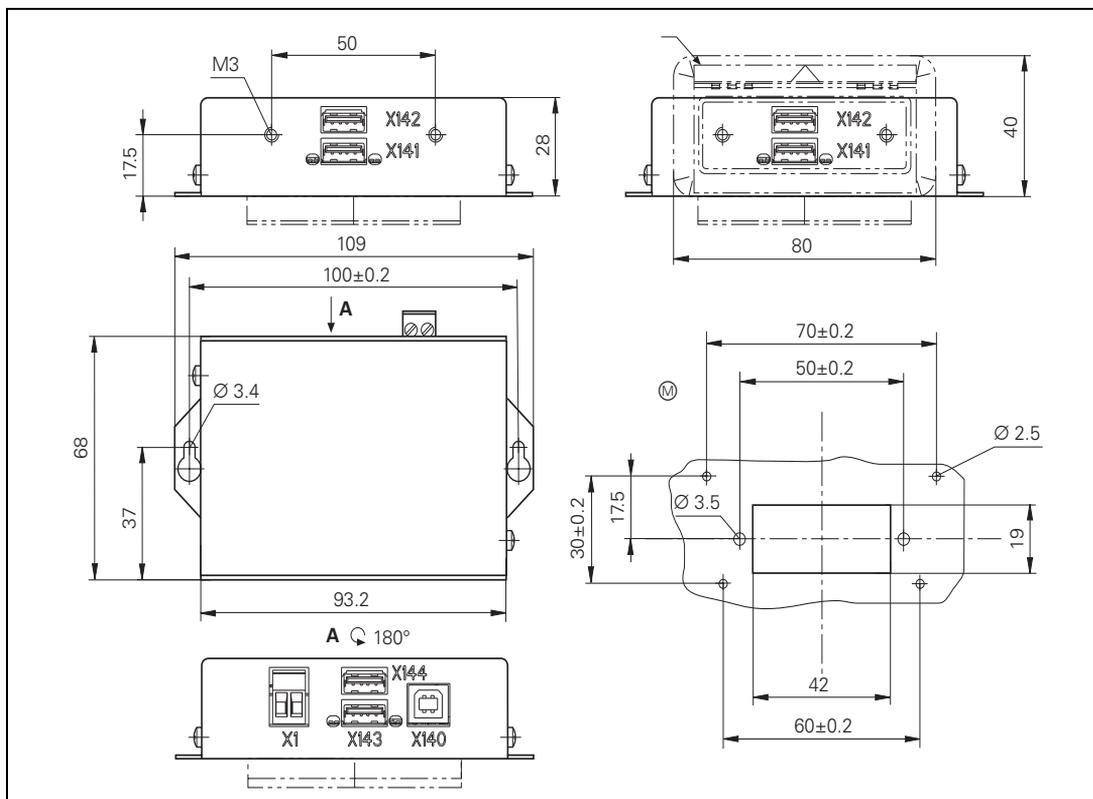




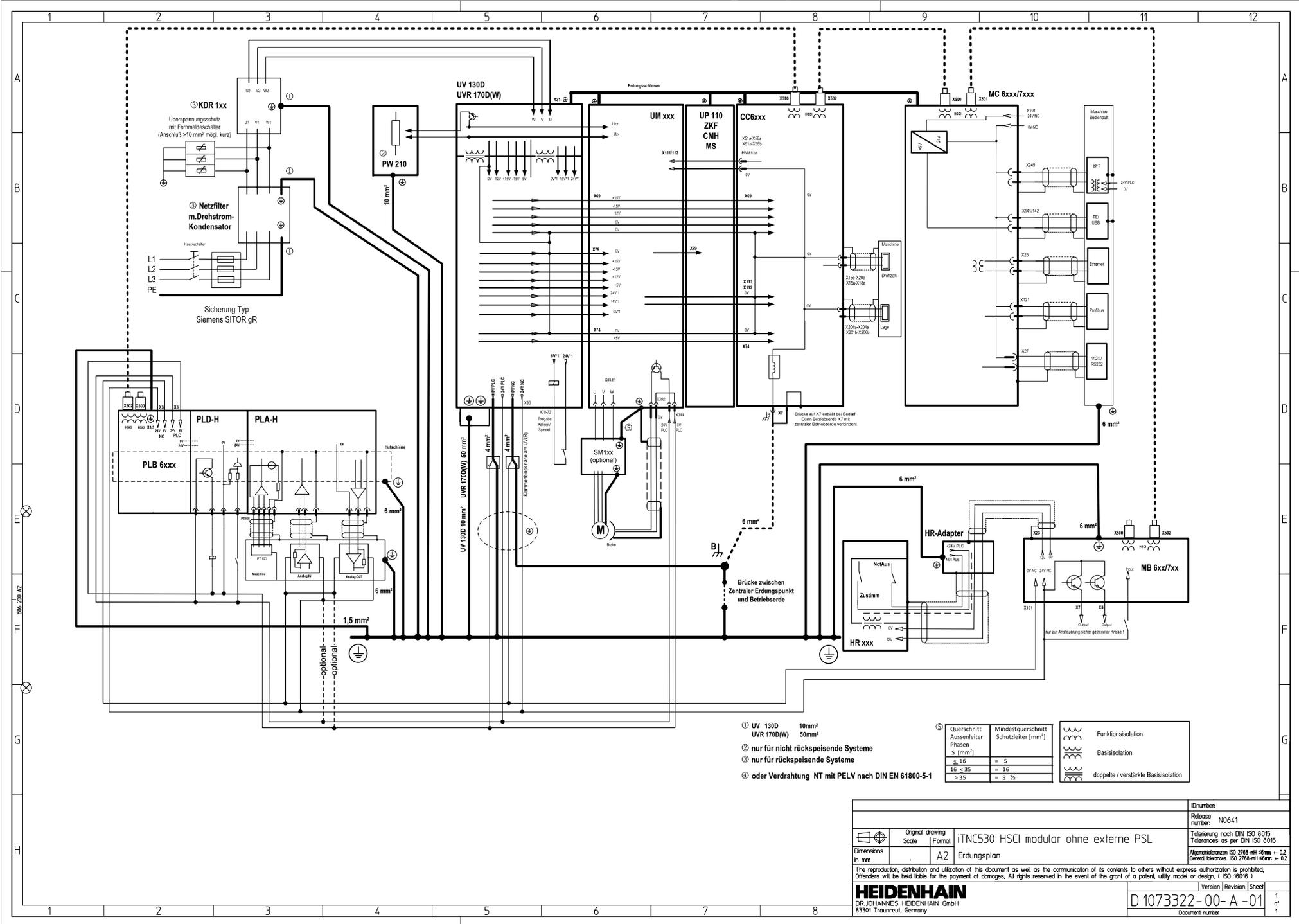
3.44.34 CML 110



3.44.35 USB hub for operating panel



3.46 iTNC 530 HSCI grounding diagram with modular HEIDENHAIN inverter system – Power supply unit integrated in UV(R)



- ① UV 130D 10mm²
UVR 170D(W) 50mm²
- ② nur für nicht rückpeisende Systeme
- ③ nur für rückpeisende Systeme
- ④ oder Verdrahtung NT mit PELV nach DIN EN 61800-5-1

⑤	Querschnitt Aussenleiter Phasen S (mm²)	Mindestquerschnitt Schutzleiter (mm²)	Funktionsisolation
	≤ 16	= S	Basisisolation
	16 < 35	= 16	doppelte / verstärkte Basisisolation
	> 35	= S ½	

Original drawing		iTNC530 HSCI modular ohne externe PSL		ID number:	
Scale	A2	Erdungsplan		Release number:	N0641
Dimensions in mm				Tolerierung nach DIN ISO 8015	
				Tolerances as per DIN ISO 8015	
				Abmessungen ISO 2768-mS	
				General tolerances ISO 2768-mS	
The reproduction, distribution and utilization of this document as well as the communication of its contents to others without express authorization is prohibited. Offenders will be held liable for the payment of damages. All rights reserved in the event of the grant of a patent, utility model or design. (ISO 10166)				Version	Revision
HEIDENHAIN DR. JOHANNES HEIDENHAIN GmbH 83301 Traunreut, Germany				D 1073322-00-A-01	1 of 1
				Document number	



3.47 iTNC 530 HSCI basic circuit diagram

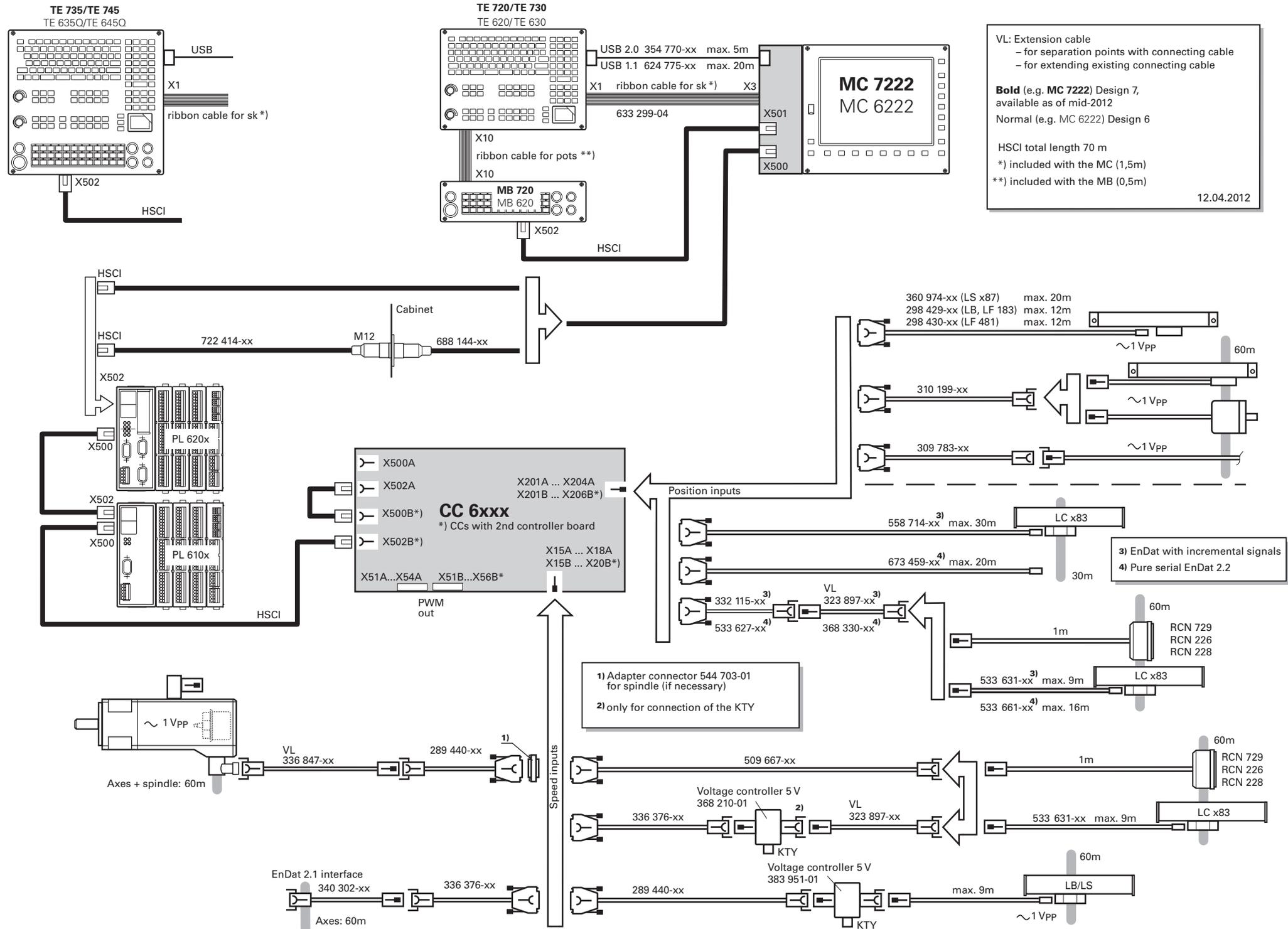
You can find the current basic circuit diagrams in the download area of the HEIDENHAIN HESIS-Web including Filebase on the Internet at <http://portal.heidenhain.de>. For this area you need access rights that you can request via e-mail.





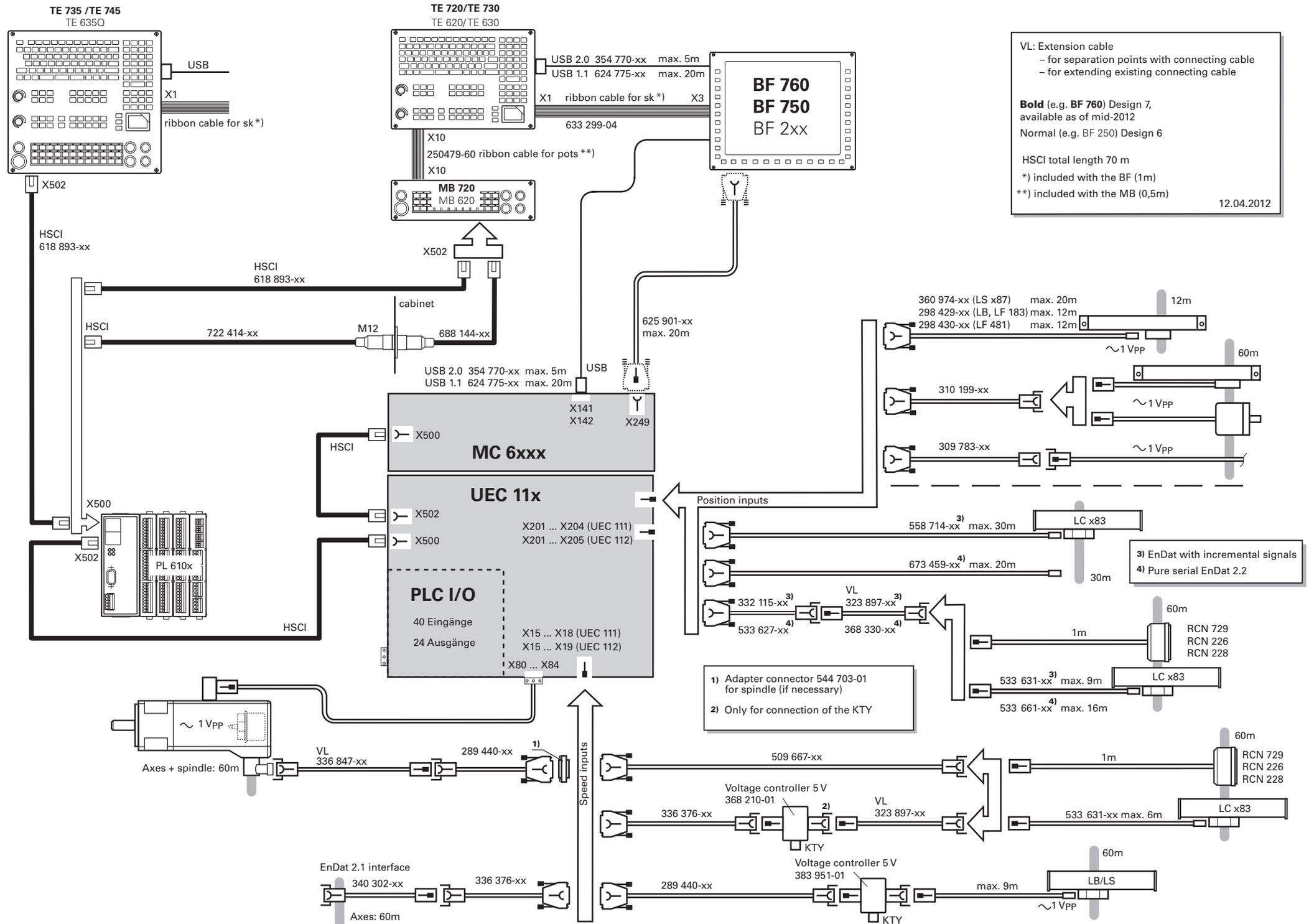
3.49 iTNC 530 HSCI cable overview – Basic configuration, MC in operating panel

Please note that the cable overviews already contain components in the new design (Design 7).

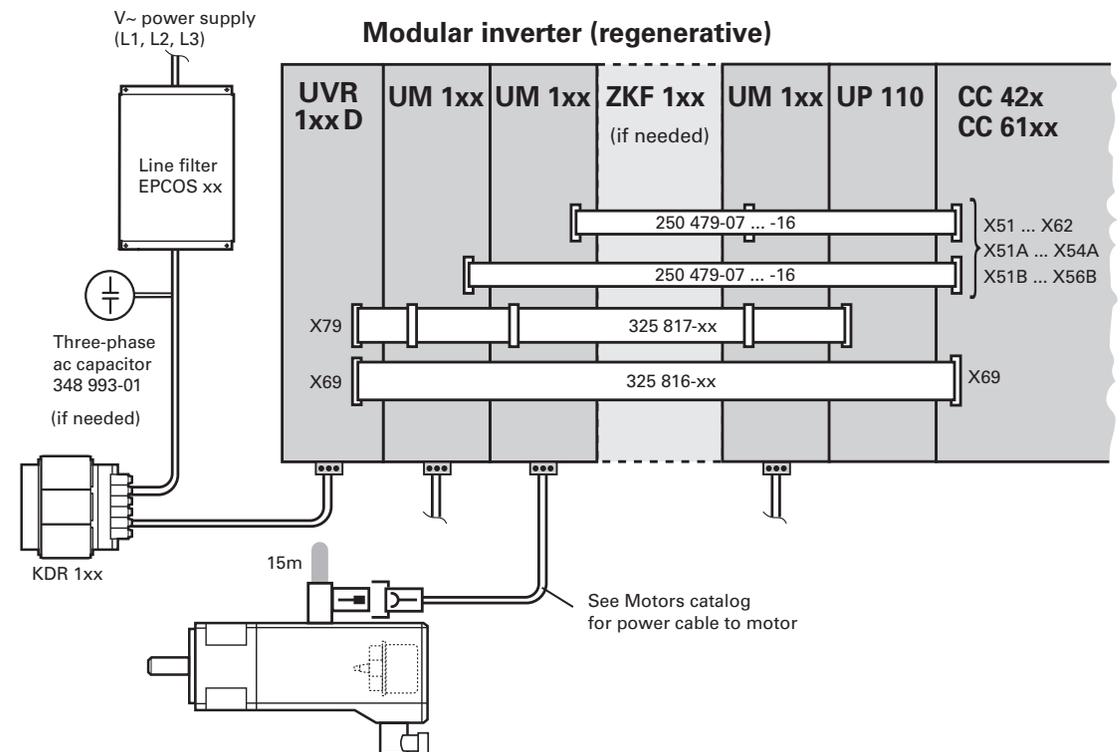
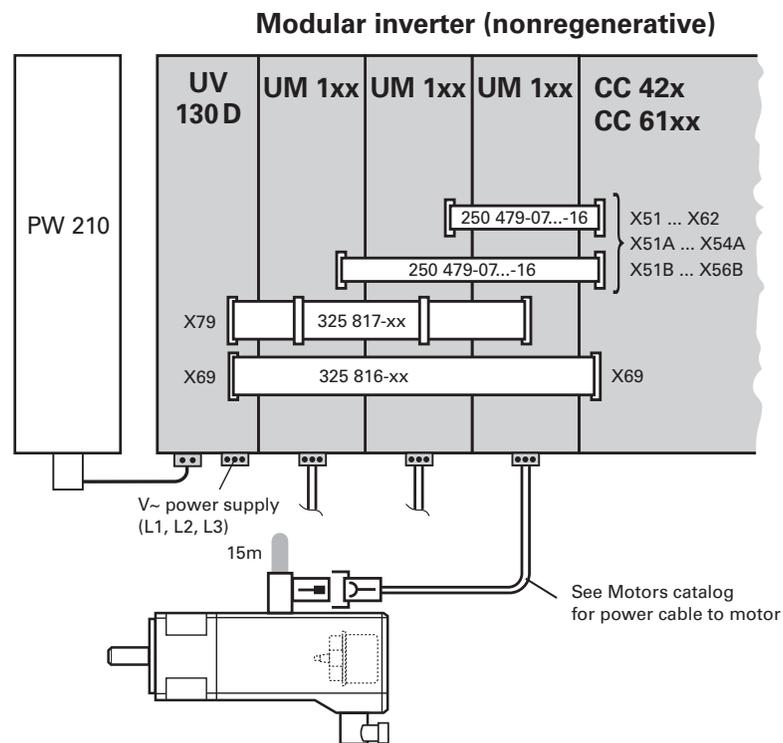
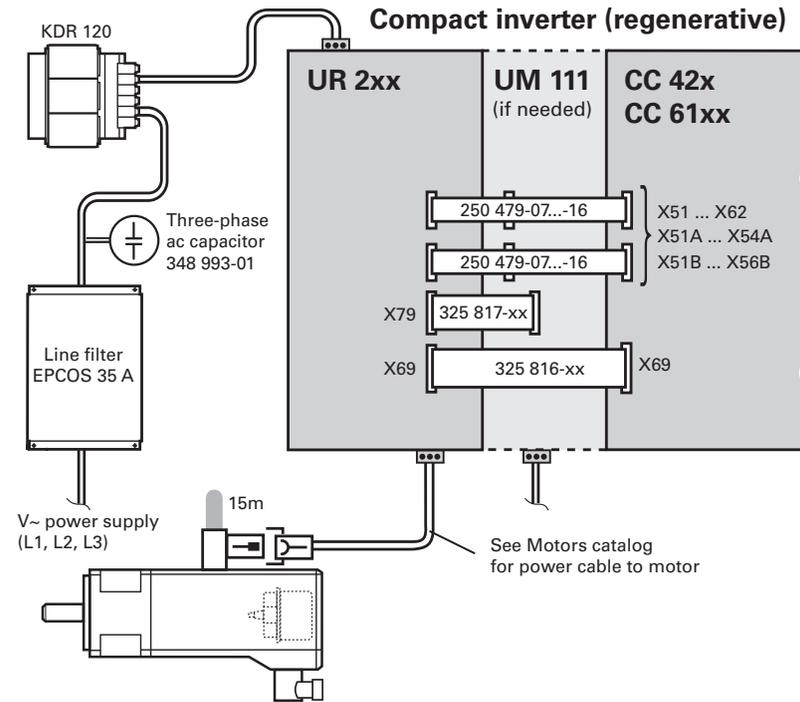
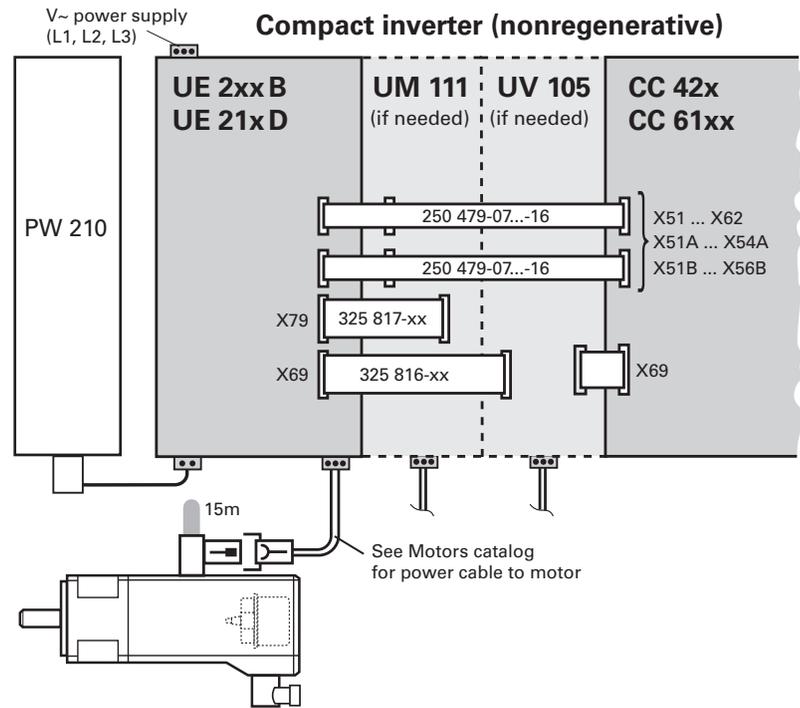


3.50 iTNC 530 HSCI cable overview – Basic configuration, MC in electrical cabinet, UEC 11x

Please note that the cable overviews already contain components in the new design (Design 7).

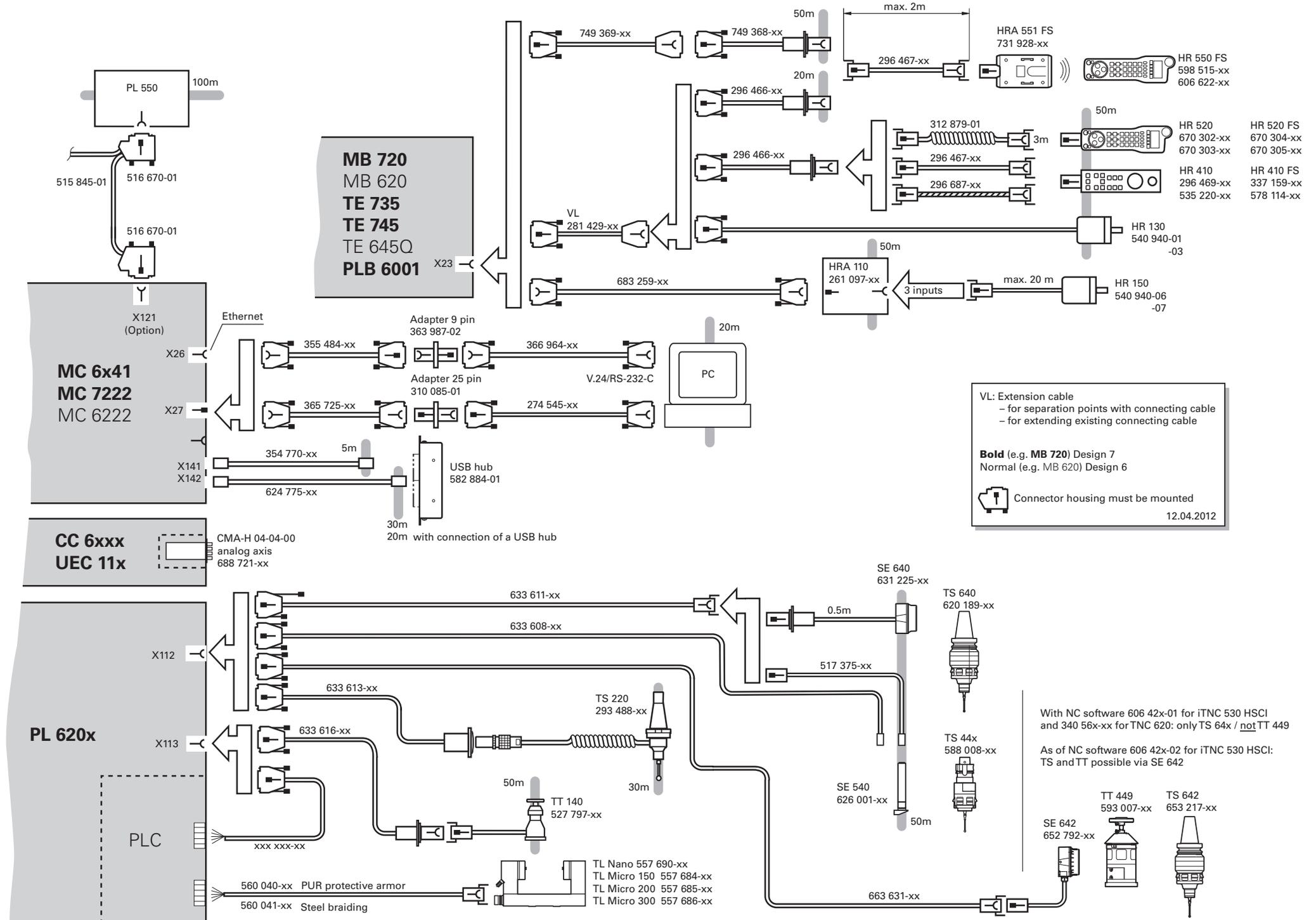


3.52 Cable overview for iTNC 530 HSCI with HEIDENHAIN inverter systems



3.53 iTNC 530 HSCI cable overview – Accessories

Please note that the cable overviews already contain components in the new design (Design 7; in boldface type).



4 Machine parameters

4.1 What is a Machine Parameter?

A contouring control must have access to specific data (e.g., traverse distances, acceleration) before it can execute its programmed instructions. You define these data in machine parameters.

This list of machine parameters is divided into groups according to topic.

Machine parameters	Topics
10 to 999	Encoders and machines
1000 to 1399	Positioning
1400 to 1699	Operation with velocity feedforward control
1700 to 1999	Servo control with following error
2000 to 2999	Integrated speed and current control
3000 to 3999	Spindle
4000 to 4999	Integrated PLC
5000 to 5999	Data interface
6000 to 6199	3-D touch probe
6500 to 6599	Tool measurement with triggering touch probe
7100 to 7199	Tapping
7200 to 7349	Display and programming
7350 to 7399	Colors
7400 to 7599	Machining and program run
7600 to 7699	Hardware

If there is more than one input value for a single function (e.g., a separate input for each axis), the parameter number is extended by indices. Index zero is always axis 1, index one is axis 2, etc.

Example:

MP1010.x	Rapid traverse
MP1010.0	Rapid traverse for axis 1
MP1010.1	Rapid traverse for axis 2
MP1010.2	Rapid traverse for axis 3
MP1010.3	Rapid traverse for axis 4
MP1010.4	Rapid traverse for axis 5
MP1010.5	Rapid traverse for axis 6
MP1010.6	Rapid traverse for axis 7
MP1010.7	Rapid traverse for axis 8
MP1010.8	Rapid traverse for axis 9
MP1010.9	Rapid traverse for axis 10
MP1010.10	Rapid traverse for axis 11
MP1010.11	Rapid traverse for axis 12
MP1010.12	Rapid traverse for axis 13
MP1010.13	Rapid traverse for axis 14
MP1010.14	Rapid traverse for axis 15
MP1010.15	Rapid traverse for axis 16
MP1010.16	Rapid traverse for axis 17
MP1010.17	Rapid traverse for axis 18

Enter into OEM.SYS, using the code word **AXISNUMBER =**, the number of axes being used, so that only the necessary index parameters are displayed.

With other machine parameters you can activate specific functions. In this case, the parameters serve as on/off switches for these functions. These parameters are bit-encoded. Each bit is assigned either to an axis or a function.



4.2 The Machine Parameter Programming mode of operation

- ▶ Enter the code number 95148 to access the **Machine Parameter Programming** mode of operation

Meaning of the soft keys in the **Machine Parameter Programming** mode of operation:

Meaning of the soft keys:	
	Switch between insertion and overwrite modes
	Jump to the beginning of the next word in the line
	Jump to the beginning of the previous word in the line
	Go back one page in the machine parameter file
	Go forward one page in the machine parameter file
	Jump to the beginning of the machine parameter file
	Jump to the end of the machine parameter file
	Search the machine parameter file for a text string
	Delete the character covered by the cursor
	Delete the word that the cursor is in
	Delete the line that the cursor is in
	Reinsert last deleted word or line
	Open the selection list for power modules
	Open the selection list for motors
	Open the selection list for power supply modules

4.3 Input and output of machine parameters

If the machine parameters have not yet been entered in a HEIDENHAIN contouring control (e.g., before commissioning), the iTNC presents the list of machine parameters after the memory test:

- ▶ Enter the values for the machine parameters either by hand on the keyboard or download them through the data interface.

4.3.1 Input format

You can enter the input values in decimal, binary (%) or hexadecimal (\$) format.

- ▶ Enter a number for each machine parameter.

The value represents, for example, the acceleration in mm/s^2 or the analog voltage in V. You can add a comment to your entry by preceding it with a semicolon (;). Binary input (%) is the best format for machine parameters that activate individual functions bit-encoded.

Example: Disabling soft keys for file types with MP7224.0

Bit 0	HEIDENHAIN programs	.H
Bit 1	DIN/ISO programs	.I
Bit 2	Tool tables	.T
Bit 3	Datum tables	.D
Bit 4	Pallet tables	.P
Bit 5	Text files	.A
Bit 6	HELP files	.HLP
Bit 7	Point tables	.PNT

The soft keys for datum tables and text files are to be disabled:

0: Do not disable

1: Disable

Input value for MP7224.0 =	Binary	%00101000
	Hexadecimal	\$28
	Decimal	40 (32+8)

**Special case:
Entering a formula**

Currently only for MP1054.x (linear distance of one motor revolution) and for MP7530 (within description tables for the kinematics).

You can enter a formula instead of a fixed value. When entering the formula, you must pay attention to the case of the letters (whether they are small or capital). Functions are written small, variables are written in capitals.

Functions:

+	Addition	sin	Sine
-	Subtraction	cos	Cosine
·	Multiplication	tan	Tangent
/	Division	asin	Arc sine
log	Logarithm	acos	Arc cosine
ln	Logarithm to the base of 10	atan	Arc tangent
exp	Exponent	sqrt	Square root
()	Expressions in parentheses are solved		
^	Exponential calculation		

Variable:

"REF" Current position of the axis relative to the machine datum
or "x" (resolution 0.0001 mm or °)

'x' was simply introduced as a short form

Example:

MP1054.0: $x*0.1^{6+15}$ (x in 0.0001 mm)

Entering a formula in MP1054.x is only useful if a position encoder is connected to the axis.

For machine parameters with formula input (e.g. MP1054), the formula can be indicated in an ASCII file. For identification, the character "@", followed by the absolute path name, is entered in the machine parameter.

Example:

@PLC:\MP\1054-X.A

The file must contain only the formula. Comments are not allowed. The formula in the file may consist of several lines and contain blank lines. The first 1024 characters of the file are evaluated.

4.3.2 Activating the machine parameter list

After you have entered all the values for the machine parameters:

- ▶ Exit the machine parameter list by pressing the END key.

Missing or incorrect entries result in error messages from the control that prompt you to correct your entry. The following errors are displayed:

Input error	Meaning
0	No MP number found
1	Invalid MP number
2	No separator (:) found
3	Input value incorrect
4	MP defined twice
6	MP cannot be saved

If the control does not recognize any errors, it automatically exits the machine parameter editor and is ready for operation.

If you do not make any entries in the machine parameter list during initial commissioning and exit the editor with the END key, the iTNC generates a standard machine parameter list (MP NAME). In this list the iTNC is defined as a programming station with the HEIDENHAIN standard colors. In all other machine parameters a default value is entered.

You can enter more than one machine parameter list in the iTNC:

- ▶ Select the lists with the PGM MGT key and the **SELECT** soft key. The last selected machine parameter list becomes active when you exit the machine parameter editor.

4.3.3 Changing the input values

A machine parameter list can be changed either with the machine parameter editor or directly through the PLC. The "List of Machine Parameters" includes the following symbols:

Indicator	Change by / Reaction
CN123	The MP is also accessible through the code number 123.
PLC	The MP can be changed via the PLC; it can also be changed in a running NC program during a strobe output.
RUN	The MP can also be changed while an NC program is running.
RESET	Changing the MP results in a reset.
REF	The axis must be moved over the reference mark again.



Manual input

- ▶ Call the machine parameter editor through the MOD function "code number":
 - By entering the code number **95148** you gain access to the complete list of machine parameters.
 - By entering the code number **123** you gain access to a subset of machine parameters. This subset can be changed by the user (see the User's Manual). Machine parameters that can be accessed through the code number 123 are indicated in the list with the symbol CN123.
- ▶ To exit the machine parameter editor, press the END key.

User parameters

You can access some machine parameters without first entering a code number.

- ▶ In MP7330.x, define up to 16 machine parameters and define the associated dialog in MP7340.x. The dialog is shown whenever the **USER PARAMETERS** soft key is pressed (up to 37 characters).
- ▶ Select the MOD function **USER PARAMETER**.

Protecting the machine parameter list

To protect the current machine parameter list from being edited through the code number 95148:

- ▶ In the OEM.SYS file, define a new code number in the entry **MPPASSWORD =** for editing the machine parameter list. Then it is **no longer possible** to edit through the code number 95148.

To protect individual machine parameters against editing:

- ▶ In the **MPLOCKFILE =** entry in the OEM.SYS file, enter the path of a machine-parameter subfile. Then it is only possible to edit those machine parameters that have no value assigned in this file. If there is a difference between the current MP value and the MP value in this subfile, the control displays an error message and a window offering the value from the subfile for your acceptance.

The following code numbers control access rights to MPs and protected MPs in the iTNC 530:

- **Entry of the code number 95148 or 984651 if no MPLOCKFILE is present**
Reading and editing of all machine parameters. Since there is no **MPLOCKFILE**, there are no SMPs. The code number 95148 or 984651 can be changed using the token **MPPASSWORD** in the OEM.SYS file. After that, the code number 95148 or 984651 only gives you read access to the MP file.
- **Entry of code number 95148 if MPLOCKFILE is present**
Reading and editing of all machine parameters that are not listed in **MPLOCKFILE**. The machine parameters of **MPLOCKFILE** can only be read, not edited. This code number cannot be changed if **MPLOCKFILE** is present, and is therefore always valid.
- **Entry of code number 984651 and MPPASSWORD if MPLOCKFILE is present**
Reading and editing of all machine parameters and safety-related machine parameters (MPs in **MPLOCKFILE**). After the control has been rebooted, any changes to the SMPs must be confirmed by entering the OEM password. You can change this code number by means of the keyword **MPPASSWORD** = in the **OEM.SYS** file to protect the machine parameters of **MPLOCKFILE** from unauthorized changes. This renders 984651 invalid for the changing of MPs. After that, the code number 984651 will only give you read access to the machine parameters of **MPLOCKFILE**.
- **PLC password**
Access to the OEM.SYS file is only possible upon entry of the PLC password. The OEM can change the PLC password via the **PLCPASSWORD** = entry in order to protect the OEM.SYS file from unauthorized changes.



Overwriting machine parameters

Machine parameters can be overwritten by the PLC or from an NC macro. In the **Program run, full sequence, Program run, single block** and **Positioning with manual data input** operating modes, machine parameters can be overwritten only when the drives are stationary, and not during a movement.



Attention

In the **Manual** and **Electronic Handwheel** operating modes machine parameters should not be overwritten during a movement, since this might result in critical conditions.

Changing the input values via PLC

You can also change the machine parameters through the PLC. The following modules are available for this purpose:

- Module 9031 Overwrite machine parameter
- Module 9032 Read machine parameter
- Module 9310 Read the machine parameter from the run-time memory
- Module 9033 Select machine parameter file
- Module 9034 Load machine parameter subfile
- Module 9312 Overwrite machine parameters in the active MP file or in the run-time memory
- Module 9313 Read machine parameters from the active MP file or from the run-time memory
- Module 9314 Activate/Deactivate machine parameters

The machine parameters that you can change with Module 9031, Module 9034 or Module 9312 are indicated with PLC in the overview.

Module 9031 Overwrite machine parameter

With this module you can overwrite the value of the given machine parameter with a new value. The input value must be a natural number with the decimal point shifted by the number of possible decimal places.

Example:

MP910.0 = 100.12 [mm] Transfer value: 1001200 (4 decimal places)

The value in the run-time memory is changed. The value from the editable machine parameter file does not change. The old value becomes valid again after the machine parameter file is edited and exited.

For non-indexed machine parameters, zero must be transferred as the index. Once the NC program has started, the module operates only during the output of an M/S/T/Q strobe.

Call only in a submit job.

Call:

PS B/W/D/K <MP number>

PS B/W/D/K <MP index>

PS B/W/D <MP value>

CM 9031

PL B/W/D <Error code>

0: No error

1: MP does not exist / is not changeable / is not changeable during a running program

2: MP value outside the valid range

3: Error while saving (fatal error)

4: Call was not in a submit or spawn job

5: Call during running program without strobe

Error code:

Marker	Value	Meaning
M4203	0	MP was overwritten
	1	MP could not be overwritten
W1022	1	MP value is outside the valid value range
	2	MP does not exist / is not changeable / is not changeable during a started program
	20	Call was not in a submit or spawn job
	21	Call during running program without strobe
	61	Error while saving (fatal error)



Module 9032 Read machine parameters

With this module you can read the value of the given machine parameter from the active machine parameter file. The input value is transferred as a natural number with the decimal point shifted by the number of possible decimal places.

Only the value from the editable machine parameter file is read, not any value modified in the run-time memory by PLC Module 9031.

For non-indexed machine parameters, zero must be transferred as the index.

Call only in a submit job.

Call:

PS B/W/D/K <MP number>

PS B/W/D/K <MP index>

CM 9032

PL B/W/D <MP value / Error code>

1: MP number does not exist

2: No separator (:)

3: MP value outside the valid range

4: MP not found in file

5: No MP file found

6: Call was not in a submit or spawn job

7: MP is of the "string" type

8: No system memory

Error code:

Marker	Value	Meaning
M4203	0	MP was read
	1	MP could not be read from the table
W1022	2	MP number does not exist; no separator (:); MP value outside the valid value range
	20	Call was not in a submit or spawn job
	41	MP is of the "string" type
	36	No MP file found
	61	No system memory
	62	MP not found in file

Module 9310 Read the machine parameter from the run-time memory

Use this module to read the value of the given machine parameter from the run-time memory. The input value is transferred as a natural number with the decimal point shifted by the number of possible decimal places. Machine parameters whose contents exceed the 32-bit limit cannot be read.

A value is read from the run-time memory.

For non-indexed machine parameters, zero must be transferred as the index.

Call:

PS B/W/D/K <MP number>

PS B/W/D/K <MP index>

CM 9310

PL B/W/D <MP value/error code>

1: MP number does not exist

3: MP outside value range

6: Call was not in a submit or spawn job

7: MP is of the "string" type

8: No system memory

Error code:

Marker	Value	Meaning
M4203	0	MP was read
	1	Error code in W1022
W1022	2	MP number does not exist or MP is outside value range
	20	Module was not called in a submit or spawn job
	41	MP is of the "string" type
	61	No system memory



Module 9312 Change machine parameters in the current machine parameter file

Module 9312 enables you to dynamically overwrite the values of the machine parameters in the active machine parameter file and the run-time memory of the system, or only in the run-time memory. For machine parameters defined as numerical values, the new value can also be programmed as a string.

Conditions:

- For numbers the values must be returned as an integer. The decimal point is shifted by the number of possible decimal places. For example, if MP910.0 is to be set to 100.12 mm, the transferred <MP value > must be equal to 1001200. (4 possible decimal places lead to multiplication by 10,000).
- For non-indexed machine parameters, zero must be programmed as the index.
- Once the NC program has started, the module operates only during the output of M/G/S/T/T2/Q (strobe) change signals.
- Depending on the changed machine parameter's type, the geometry is re-initialized.
- Not every MP can be changed by the PLC.
- If a new value for a machine parameter defined as a numerical value is programmed as a string, the value is converted to the decimal system (even if the new value is programmed in the "\$0004" or "%0000011111" format, for example).
- If the new values are immediately transferred after the modification of the active machine-parameter file and the run-time memory (bits 1 and 2 not set in the parameter "Mode"), the marker 4174 (first run after MP was changed) is set.
- If the new machine parameters are not transferred immediately, the new values will become effective when the module is called again without bit 2 being set.

Call:
 PS B/W/D/K <Mode>
 Bit 0:
 0 = Modify MP numerical value (MP is a number)
 1 = Modify MP string (MP is a string)
 Bit 1:
 0 = Modify parameter in the file and in the run-time memory
 (Bit 2 is then also relevant)
 1 = Modify parameter in run-time memory
 (behavior like Module 9031)
 Bit 2:
 0 = Change the file and the run-time memory and
 immediately transfer value(s) into the NC
 1 = Change the file and the run-time memory,
 do not immediately transfer new value(s) into NC
 Will be transferred at next call with bit 2 = 0
 PS B/W/D/K/S<File name>
 not supported at present
 (a value must be transferred (e.g. S""), but the value is not evaluated
 yet)
 PS B/W/D/K <MP number>
 PS B/W/D/K <MP index>
 PS B/W/D/K/S<MP value>
 Depending on the mode, the MP value is interpreted as a PLC
 string number, PLC constant string or number
 CM 9312
 PL B/W/D <Error code>
 0: No error
 1: Parameter does not exist or cannot be changed
 2: New value for parameter is invalid
 3: Error while saving
 4: Call was not in a submit or spawn job
 5: Call during running NC program without change signal
 6: Invalid PLC string for file name or MP value
 7: Parameter file does not exist

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	See error code
W1022	1	New value for parameter is invalid
	2	Parameter does not exist or cannot be changed
	11	Invalid PLC string for file name or MP value
	20	Call was not in a submit or spawn job
	21	Call during running NC program without change signal
	36	No MP file found
	61	Error while saving



Module 9313 Read machine parameters

Module 9313 reads the contents of machine parameters from the run-time memory or from the current machine-parameter file. For machine parameters defined as numerical values, the content can also be read as a string. Machine parameters defined as strings cannot be read as numbers.

Machine parameters that have no function in the currently active machine configuration (highlighted in the MP editor) previously reported the value 0 when they were read by PLC module 9313. As of software 340 49x-05, the value entered in inactive machine parameters will also be reported. For machine parameters, that are not in the MP file, the default value is read.

Call:

PS B/W/D/K <Mode>

Bit 0:

0 = Read MP as numerical value (MP must be a number)

1 = Read MP as string (MP can be a number or text)

Bit 1:

0 = Read MP from file

1 = Read MP from memory

PS B/W/D/K/S<File name>

not supported at present

(a value must be transferred, but the value entered is not evaluated yet)

(only the active MP file can be read from)

PS B/W/D/K <MP number>

PS B/W/D/K <MP index>

PS B/W/D/K/S<Target word or target string>

Mode bit 0 = 0: Interpretation as a double-word address

Mode bit 0 = 1: Interpretation as a PLC string number

CM 9313

PL B/W/D <Error>

0: No error

1: MP number, PLC string or PLC double-word address invalid

2: No ":" separator in MP file, e.g. (MP 10: %0111)

3: MP value outside permissible value range

4: MP not found in file

5: No MP file found

6: Call was not in a submit or spawn job

7: MP (defined as string) cannot be read as a number

8: No system memory

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	See error code

Marker	Value	Meaning
W1022	2	Invalid MP number programmed or MP value outside the permissible value range
	11	Invalid PLC string or double-word address programmed
	20	Call was not in a submit or spawn job
	36	No MP file found
	41	MP (defined as string) cannot be read as a number
	61	No system memory
	62	No MP file found



Module 9314 Activate/Deactivate machine parameters

PLC module 9314 is used to axis-specifically activate/deactivate other values from machine parameters during run time.

Condition:

- The module transmits the information at the end of a PLC cycle.
- After the PLC program has been restarted, the module will no longer be effective and the original configuration will become active again.
- HEIDENHAIN recommends using power limiting (mode 0) via MP2393.x only for spindles. Power limiting for axes might cause positioning errors.

Call:

PS K/B/W/D <Bit mask for axes/spindles>

Bit for axis/spindle = 1: Value from MPxxx.y is active

Bit for axis/spindle = 0: Value from MPxxx.y is not active

PS K/B/W/D <Mode>

0: Activation/Deactivation of power limiting (MP2393.x)

CM 9314

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Invalid value for this mode is active
	24	Call was not from a cyclic program

Module 9033 Select machine parameter file

With this module you can select a new machine parameter file. If machine parameters that set off a reset were changed, the control system will restart.



Attention

The module does not take any existing safety problems into account when setting off a control reset (e.g., axes and spindle coasting to a stop).

The file to be selected is checked; a faulty file is not selected. If file selection is successful, there is no return to the calling PLC program.

The file name is transferred in a string that must contain the complete path, name and file extension. Further characters, even space characters, are not permitted.

If the PLC program is created externally, ensure that lower-case letters are not used for the file name!

The path for the MP (sub)file can only be entered as absolute path (e.g. PLC:\BASIS\PROGRAM\MP\subfile1.A) or relative to the directory of the PLC main program (e.g. \MP\subfile1.A). If a relative path is entered, the MP (sub)file must be located in the directory of the PLC main program or an associated subdirectory. The possibility of entering a relative path eliminates the need for adapting the source code of the PLC program if the PLC program is moved to the encrypted PLCE partition.

Once the NC program has started, the module operates only during the output of an M/S/T/Q strobe.

Call only in a submit job.

Call:

PS B/W/D/K <String number>

CM 9033

Note: If a new file is selected, program execution ends here.

PL B/W/D <Error code>

0: No error. File was already selected

1: String does not contain a valid file name

2: File not found

3: File is faulty

4: Incorrect string number transferred

5: Call was not in a submit job

6: Call during running program without strobe



Machine parameter subfile

A machine parameter subfile can be activated via Module 9034, or from the NC program via **FN17: SYSWRITE** (also see page 1703).

Module 9034 Load a machine parameter subfile

With this module you load the contents of the given machine parameter into the main memory. All MPs not listed in this file remain unchanged.

Placeholders **?** can be entered in bit-specific machine parameters in MP subfiles. This makes it possible to change individual bits specifically. For example, you can overwrite only bits 2 and 5 in an MP subfile by entering **MP10:????1?0??**. The other bits remain unchanged.

The MP file to be selected is checked. A faulty file is not loaded. If the MP file contains parameters that require a system reset, the file is not loaded.

The file name is transferred in a string that must contain the complete path, name and file extension. Further characters, even space characters, are not permitted.

If the PLC program is created externally, ensure that lower-case letters are not used for the file name!

The path for the MP (sub)file can only be entered as absolute path (e.g. PLC:\BASIS\PROGRAM\MP\subfile1.A) or relative to the directory of the PLC main program (e.g. \MP\subfile1.A). If a relative path is entered, the MP (sub)file must be located in the directory of the PLC main program or an associated subdirectory. The possibility of entering a relative path eliminates the need for adapting the source code of the PLC program if the PLC program is moved to the encrypted PLCE partition.

Once the NC program has started, the module operates only during the output of an M/S/T/Q strobe.

Call only in a submit job.

Call:

PS B/W/D/K <String number>
 0 to 99

CM 9034

PL B/W/D <Error code>

0: No error

1: String does not contain a valid file name,
or the name (including the path) is too long

2: File not found

3: File is faulty / contains reset parameters

4: Incorrect string number was transferred (0 to 3)

5: Call was not in a submit job

6: Call during running program without strobe

4.4 List of machine parameters

4.4.1 Encoders and machines

MP	Function and input	Software version and behavior	Page
MP10	Active axes Format: %xxxxxxxxxxxxxxxxxxx Input: Bits 0 to 17 represent axes 1 to 18 0: Axis not active 1: Axis active	PLC RUN	603
MP12	Axis-specific demo operation for NC axes Format: %xxxxxxxxxxxxxxxxxxx Input: Bits 0 to 17 represent axes 1 to 18 0: Demo operation not active 1: Demo operation active	PLC RUN	604
MP20	Monitoring functions for the axes Format: %xxxxxxxxxxxxxxxxxxx Input: Bits 0 to 17 represent axes 1 to 18 0: Monitoring not active 1: Monitoring active	PLC RUN	610
MP20.0	Absolute position of the distance-coded reference marks		
MP20.1	Amplitude of encoder signals		
MP20.2	Edge separation of encoder signals		
MP21	Monitoring functions for the spindle Format: %xx Input: Bit 0 – Spindle 1 0: Monitoring not active 1: Monitoring active Bit 1 – Spindle 2 0: Monitoring not active 1: Monitoring active	PLC RUN	610
MP21.0	Absolute position of the distance-coded reference marks		
MP21.1	Amplitude of encoder signals		
MP21.2	Edge separation of encoder signals		
MP100	Designation of axes Format: -wvucbazyxWVUCBAZYX Any lowercase letters for auxiliary axes, any uppercase letters for slave axes Input: Characters 1 to 9 from the right represent axes 1 to 18	PLC RUN	603, 630
MP100.0	Traverse range 1		
MP100.1	Traverse range 2		
MP100.2	Traverse range 3		



MP	Function and input	Software version and behavior	Page
MP108.x	Assignment of axes to drive-control motherboards Input: 0 to 4: Number of drive-control motherboard in the HSCI chain	RESET	615
MP109.x	Assignment of spindle(s) to drive-control motherboards Input: 0 to 4: Number of drive-control motherboard in the HSCI chain	RESET	617
MP110.x	Assignment of position encoder inputs to the axes Input: 0: No position encoder input 1 to 6: Position encoder inputs X1 to X6 35 to 38: Position encoder inputs X35 to X38 201 to 214: Position encoder inputs X201 to X214	RESET	615
MP111.x MP111.0 MP111.1	Position encoder input for the spindle/spindles Input: 0: No position encoder input 1 to 6: Position encoder inputs X1 to X6 35 to 38: Position encoder inputs X35 to X38 201 to 214: Position encoder inputs X201 to X214 Position encoder input for the first spindle Position encoder input for the second spindle	REF	617, 948
MP112.x	Assignment of speed encoder inputs to the axes Input: 0: No speed encoder input 15 to 20: Speed encoder inputs X15 to X20 80 to 85: Speed encoder inputs X80 to X85	RESET	615
MP113.x MP113.0 MP113.1	Speed encoder for the spindle/spindles Input: 0: No speed encoder input 15 to 20: Speed encoder inputs X15 to X20 80 to 85: Speed encoder inputs X80 to X85 Speed encoder for the first spindle Speed encoder for the second spindle	REF	617

MP	Function and input	Software version and behavior	Page
MP118.x	Position encoder input for axes Format: %xxxx Input: Bit 0: Position encoder input 1 V _{PP} 0: 1 V _{PP} 1: Reserved Bit 1: Terminating resistor for encoder signal line 0: Active 1: Deactivated Bit 2: Input frequency of the position encoder input at 1 V _{PP} : 0: 27 kHz 1: 400 kHz Bit 3: Analog or digital position encoder input: 0: Analog encoder signal control (1 V _{PP}) 1: Digital encoder signal control (EnDat 2.2) Bit 4: Reserved Function is not yet supported. Bit 5: Reserved Function is not yet supported.	60642x-03	608
MP119.x	Position encoder input for the spindle/spindles Format: %xxxx Input: Bit 0: Position encoder input 1 V _{PP} 0: 1 V _{PP} 1: Reserved Bit 1: Terminating resistor for encoder signal line 0: Active 1: Deactivated Bit 2: Input frequency of the position encoder input at 1 V _{PP} : 0: 27 kHz 1: 400 kHz Bit 3: Analog or digital position encoder input: 0: Analog encoder signal control (1 V _{PP}) 1: Digital encoder signal control (EnDat 2.2) Bit 4: Reserved Function is not yet supported. Bit 5: Reserved Function is not yet supported.	60642x-03	608
MP120.x	Nominal speed command outputs of the axes Input: 0: No servo-controlled axis 1 to 8: Analog CMA output X66, X67 51 to 56: Digital CC outputs X51 to X56 80 to 85: Digital UEC outputs X80 to X85	RESET	615



MP	Function and input	Software version and behavior	Page
MP121.0	Nominal speed command output of the first spindle Input: 0: No servo-controlled axis 1 to 8: Analog CMA output X66, X67 51 to 56: Digital CC outputs X51 to X56 80 to 85: Digital UEC outputs X80 to X85	RESET	617
MP121.1	Nominal speed command output of the second spindle Input: 0: No servo-controlled axis 1 to 4: Analog CMA output X66, X67 51 to 56: Digital CCC outputs X51 to X56 80 to 85: Digital UEC outputs X80 to X85	RESET	
MP130.x	Y index of the machine parameters MP2xxx.y for the axes Input: 0 to 17	PLC RUN	615
MP131.x	Y index of the machine parameters MP2xxx.y for the spindle(s) in operating mode 0 Input: 0 to 17	PLC RUN	617
MP131.0	Index for the first spindle		
MP131.1	Index for the second spindle		
MP132.x	Y index of the machine parameters MP2xxx.y for the spindle(s) in operating mode 1 Input: 0 to 17	PLC RUN	617
MP132.0	Index for the first spindle		
MP132.1	Index for the second spindle		
MP210	Counting direction of position encoder output signals Format: %xxxxxxxxxxxxxxxxxxxx Input: Bit 0 to Bit 17 represent axis 1 to 18 0: Positive 1: Negative	REF	609
MP331.x	Distance for the number of signal periods in MP332 Input: 0.0001 to +1.797693135E+308 [mm] or [°]	PLC RUN REF	607
MP332.x	Number of signal periods for the distance in MP331 Input: 1 to +1.797693135E+308	PLC RUN REF	607
MP334.x	Nominal increment between two fixed reference marks on encoders with distance-coded reference marks Input: 1 to 65 535 0: 1 000	PLC RUN REF	607
MP340.x	Interpolation factor for external interpolation Input: 0 to 99 0 = 1: No external interpolation	RESET	607

MP	Function and input	Software version and behavior	Page
MP410	Assignment of axis keys IV, V and VI Input: Axis labels A/B/C/U/V/W/T	PLC RUN	604
MP410.3	IV axis key		
MP410.4	V axis key		
MP410.5	VI axis key (only HR 5xx)		
MP420.x	Hirth coupling Input: 0: No Hirth coupling 1: Hirth coupling	PLC RUN	1532
MP430.x	Prescribed increment for Hirth coupling Input: 0.0000 to 180.0000 [°]	PLC RUN	1532
MP708.x	Traverse distance for acceleration-dependent backlash compensation Input: 0.0001 to 1.000 [mm] 0: Function inactive	PLC RUN	649
MP709.x	Time constant for backlash compensation Input: 1 to 1000 [ms]	PLC RUN	649

MP	Function and input	Software version and behavior	Page
MP710.x	Backlash compensation Input: -9.9999 to +9.9999 [mm] or [°]	PLC RUN	649
MP715.x	Height of peaks during circular movement (analog only) with M105 Input: -1.0000 to +1.0000 [mm] (digital: 0)	PLC RUN	
MP716.x	Compensation value per control loop cycle time with M105 Input: 0.000000 to 99.999999 [mm] (digital: 0)	PLC RUN	
MP720.x	Linear axis error compensation Input: -1.000 to +1.000 [mm/m]	PLC RUN	651
MP730	Selection of linear/nonlinear axis error compensation Format: %xxxxxxxxxxxxxxxxxxxx Input: Bit 0 to Bit 17 represent axis 1 to 18 0: Linear axis error compensation 1: Nonlinear axis error compensation	PLC RUN	651, 657
MP732	Nonlinear axis-error compensation for rotary axes Format: %xxxxxxxxxxxxxxxxxxxx Input: Bit 0 to Bit 17 represent axis 1 to 18 0: Not active (usual compensation) 1: Active (mapped to traverse range)		658
MP750.x	Reversal error (backlash compensation) Input: -9.9999 to +9.9999 [mm] or [°]	PLC RUN	650
MP752.x	Compensation time for reversal error Input: 0 to 1000 [ms]	PLC RUN	650
MP810.x	Display mode for rotary axes and PLC auxiliary axes Input: 0.0000 to 99 999.9999 [°] 0: Display +/-99 999.9999 1: Modulo value for display	PLC RUN REF	1149
MP812	Activate software limit switches for rotary axes with modulo display, M94 and encoders with EnDat interface Format: %xxxxxxxxxxxxxxxxxxxx Input: Bit 0 to Bit 17 represent axis 1 to 18 0: Software limit switch not active 1: Software limit switch active	RESET	1149

MP	Function and input	Software version and behavior	Page
MP850.x	Synchronized axes Input: 0: Master axis 1: Slave axis to axis 1 2: Slave axis to axis 2 3: Slave axis to axis 3 4: Slave axis to axis 4 5: Slave axis to axis 5 6: Slave axis to axis 6 7: Slave axis to axis 7 8: Slave axis to axis 8 9: Slave axis to axis 9 10: Slave axis to axis 10 11: Slave axis to axis 11 12: Slave axis to axis 12 13: Slave axis to axis 13 14: Slave axis to axis 14 15: Slave axis to axis 15 16: Slave axis to axis 16 17: Slave axis to axis 17	PLC RUN	779
MP855.x	Synchronization monitoring Input: 0 to 100.0000 [mm] 0: Monitoring not active	PLC RUN	781
MP860.x	Datum for synchronous control Format: %xxx Input: Bit 0 – Datum position for synchronous control 0: Datum at position after switch-on 1: Datum at reference marks (machine datum) Bit 1 – Master-slave torque control 0: Axis is not slave axis in master-slave torque control 1: Axis is slave axis in master-slave torque control Bit 2 – Brake test of slave axis 0: Test the brakes of a synchronized axis simultaneously 1: Test the brakes of a synchronized axis successively	PLC RUN	781, 785
MP910.x	Positive software limit switches, traverse range 1 (default setting after power on) Input: –99 999.9999 to +99 999.9999 [mm] or [°]	PLC RUN	628
MP911.x	Positive software limit switches, traverse range 2 Input: –99 999.9999 to +99 999.9999 [mm] or [°]	PLC RUN	628
MP912.x	Positive software limit switches, traverse range 3 Input: –99 999.9999 to +99 999.9999 [mm] or [°]	PLC RUN	628



MP	Function and input	Software version and behavior	Page
MP920.x	Negative software limit switches, traverse range 1 (default setting after power on) Input: -99 999.9999 to +99 999.9999 [mm] or [°]	PLC RUN	628
MP921.x	Negative software limit switches, traverse range 2 Input: -99 999.9999 to +99 999.9999 [mm] or [°]	PLC RUN	628
MP922.x	Negative software limit switches, traverse range 3 Input: -99 999.9999 to +99 999.9999 [mm] or [°]	PLC RUN	628
MP950.x	Datum for positioning blocks with M92 for axes 1 to 9 Input: -99 999.9999 to +99 999.9999 [mm] or [°] Values with respect to the machine datum	PLC RUN	1207
MP951.x	Simulated tool-change position for TOOL CALL during mid-program startup (block scan) Input: -99 999.9999 to +99 999.9999 [mm] or [°]	PLC RUN	1246
MP960.x	Machine datum Input: -1.79769313486E+308 to +1.79769313486E+308 [mm] or [°] Values with respect to the scale reference point	PLC RUN	802, 1207

4.4.2 Positioning

MP	Function and input	Software version and behavior	Page
MP1010.x	Rapid traverse Input: 10 to 1 000 000 [mm/min or °/min]	PLC RUN	842
MP1011	Limit of rapid traverse on the path Input: 10 to 1 000 000 [mm/min or °/min]	PLC RUN	842
MP1012.x	Second axis-specific rapid traverse Input: 10 to 1 000 000 [mm/min or °/min]		842
MP1020.x	Manual feed Input: 10 to 300 000 [mm/min]	PLC RUN	842
MP1030.x	Positioning window Input: 0.0001 to 2.0000 [mm] or [1000°/s ²]	PLC RUN	900
MP1040	Analog axes: Polarity of nominal value voltage Digital axes: Algebraic sign of the nominal speed value Format: %xxxxxxxxxxxxxxxxxxxx Input: Bit 0 to Bit 17 represent axis 1 to 18 0: Positive 1: Negative		609
MP1050.x	Analog axes: Analog voltage at rapid traverse Input: 1.000 to 9.000 [V] Digital axes: without function Input: 1	PLC RUN	842
MP1054.x	Distance of one motor revolution [mm or °] Input: Analog axes: Without function Digital axes: Entry of a formula possible, *** "Special case: Entering a formula" on page 479 ***	PLC RUN	899
MP1060.x	Acceleration Input: 0.001 to 500 [m/s ²] or [1000°/s ²]	PLC RUN	818
MP1061	Limitation of the path acceleration Input: 0.001 to 500 [m/s ²] or [1000°/s ²]	PLC RUN	818
MP1061.0	For machining feed rate		
MP1061.1	For rapid traverse		
MP1070	Radial acceleration Input: 0.001 to 500 [m/s ²]	PLC RUN	891
MP1080.x	Analog axes: Integral factor for offset adjustment Input: Enter 0 to 65 535 Digital axes: No function Input: 0	PLC RUN	890



MP	Function and input	Software version and behavior	Page
MP1085.x	Maximum permissible axis-specific jerk for path movements in the operating modes Program Run Full Sequence, Program Run Single Block, and Positioning with Manual Data Input Input: 0.0 to 9999.9 [m/s ³ or 1000° /s ³]		818
MP1086.x	Maximum permissible axis-specific jerk for rapid traverse movements in the operating modes Program Run Full Sequence, Program Run Single Block, and Positioning with Manual Data Input Input: 0: Function inactive 0.0 to 9999.9 [m/s ³ or 1000° /s ³]	PLC RUN	818
MP1087.x	Maximum permissible axis-specific jerk for Manual operating mode Input: 0.1 to 1000.0 [m/s ³ or 1000° /s ³]	PLC RUN	818
MP1088.x	Axis-specific jerk limiting for unfiltered positioning movements Input: 0.1 to 9999.9 [m/s ³ or 1000° /s ³]		818
MP1089.x	Max. permissible axis-specific jerk for Pass Over Reference Point mode Input: 0.1 to 1000.0 [m/s ³ or 1000°/s ³]	PLC RUN	818
MP1090	Maximum permissible jerk on the tool path Input: 0: Not active 0.0 to 9999.9 [m/s ³ or 1000° /s ³]	PLC RUN	818
MP1090.0	For movements not at rapid traverse or feed rate < MP1092		
MP1090.1	For movements at rapid traverse or feed rate > MP1092		
MP1092	Feed rate threshold for MP1085.x and MP1086.x Input: 1 to 300 000 [mm/min] (previous behavior) 0: Not active	PLC RUN	818
MP1094	HSC filter (no longer recommended with HSCI, improved nominal position value filters as of MP12xx) Input: 0: HSC filter inactive 0.1 to 166.0: Cutoff frequency for HSC filter	Only through MPMODE = 340422 in OEM.SYS	
MP1095	Nominal position value filter (no longer recommended with HSCI, improved nominal position value filters as of MP12xx) Input: 0: Single filter 1: Double filter	PLC RUN	
MP1095.0	In the Program Run Full Sequence, Program Run Single Block, and Positioning With Manual Data Input operating modes	Only through MPMODE = 340422 in OEM.SYS	
MP1095.1	In the Manual, Handwheel, Jog Increment and Pass Over Reference Point operating modes		

MP	Function and input	Software version and behavior	Page
MP1096	Tolerance for contour transitions at corners (no longer recommended with HSCI, improved nominal position value filters as of MP12xx) Input: 0: No nominal position value filter 0.001 to 3.000 [mm]	PLC RUN Only through MPMODE = 340422 in OEM.SYS	,
MP1096.0	With machining feed rate		
MP1096.1	With rapid traverse		
MP1097.x	Max. permissible axis-specific jerk (single filter / HSC filter) (no longer recommended with HSCI, improved nominal position value filters as of MP12xx) Input: 0.1 to 1000.0 [m/s ³ or 1000°/s ³]	PLC RUN Only through MPMODE = 340422 in OEM.SYS	
MP1098.x	Max. permissible axis-specific jerk (double filter / HSC filter) (no longer recommended with HSCI, improved nominal position value filters as of MP12xx) Input: 0.1 to 1000.0 [m/s ³ or 1000°/s ³]	PLC RUN Only through MPMODE = 340422 in OEM.SYS	
MP1099	Minimum filter order (no longer recommended with HSCI, improved nominal position value filters as of MP12xx) Input: 0 to 20	PLC RUN Only through MPMODE = 340422 in OEM.SYS	
MP1099.0	Minimum filter order for single filter (MP1095 = 0)		
MP1099.1	Minimum filter order for double filter (MP1095 = 1)		
MP1110.x	Standstill monitoring Input: 0.0010 to 30.0000 [mm]	PLC RUN	900
MP1120.x	Standstill monitoring when determining the field angle Input: 0.0000 to 300.0000 [mm] or [°]	PLC RUN	1019
MP1140.x	Threshold above which the movement monitoring functions Input: Analog axes: 0.030 to 10.000 [V] Digital axes: 0.030 to 10.000 [1000 min] Recommended: 0.030 [1000 min]	PLC RUN	899
MP1144.x	Motion monitor for position and speed Input: Analog axes: Without function Digital axes: 0 to 99 999.999 [mm] 0: No monitoring	PLC RUN	899

MP	Function and input	Software version and behavior	Page
MP1146.x	Difference between the position at shutdown and the position read in via the EnDat interface Input: 0.0000 to 300.0000 [mm] or [°] 0: No difference permitted	PLC RUN	897
MP1150.0	Delay time for erasing the nominal velocity value with the erasable error message EXCESSIVE SERVO LAG IN <AXIS> Input: 0 to 65.535 [s] Recommended: 0	PLC RUN	896, 894
MP1150.1	Time period for which the monitoring function is to remain off after the fast PLC input defined in MP4130.0 is set. With HSCI, this must be realized via the PLC program. Input: 0 to 65.535 [s] 0: Monitoring functions on Recommended: 0.2 to 0.5		
MP1150.2	Minimum time period for which the monitoring functions are to remain effective after expiration of the time from MP1150.1. With HSCI, this must be realized via the PLC program. Input: 0 to 65.535 [s]		
MP1160	LIFTOFF at powerfail Input: 0 to 30.0000 [mm] Default: 0.1 [mm]	PLC RUN	1099
MP1200	Selection of the nominal position value filter used Input: 0: Single filter 1: Double filter 2: HSC filter 3: Advanced HSC filter	PLC RUN	827
MP1201	Nominal position value filter in manual operation Input: 0: Single filter 1: Double filter	PLC RUN	827
MP1202	Predefined tolerance for Cycle 32 Input: 0.0000 to 3.0000 [mm]	PLC RUN	827
MP1202.0	Tolerance at corners for movements at machining feed rate		
MP1202.1	Tolerance at corners for movements at rapid traverse		
MP1205	Reduction of the contouring feed rate at the beginning of a contour element Input: 0: Not active (fast, possibly less precise) 1: Active (slow but likely more precise)		827
MP1210	Limit frequency for single filter Input: 0: Filter is switched off 0.0 to 166.0 [Hz]	PLC	827
MP1211	Limit frequency for double filter Input: 0: Filter is switched off 0.0 to 166.0 [Hz]	PLC	827

MP	Function and input	Software version and behavior	Page
MP1212	Limit frequency for HSC filter Input: 0: Filter is switched off 0.0 to 166.0 [Hz]	PLC	827
MP1213	Limit frequency for advanced HSC filter Input: 0: Filter is switched off 0.0 to 166.0 [Hz]	PLC	827
MP1218.x	Cutoff frequency for nominal position value filters or filters of rotary axes (A, B, C) Input: 0: Triangle filter (standard) 0.0 to 166.0 [Hz]	606 42x-02 PLC	827
MP1222	Tolerance for curvature changes with HSC filter (only effective if MP7684 bit 4 = 0) Entry: 0: Do not include the tolerance 1: Include the tolerance	PLC RUN	827
MP1223	Tolerance for curvature changes with advanced HSC filter (only effective if MP7684 bit 4 = 0) Entry: 0: Do not include the tolerance 1: Include the tolerance	PLC RUN	827
MP1230.x	Max. permissible axis-specific jerk at corners for single filter Input: 0.1 to 1000.0 [m/s ³]	PLC RUN	828
MP1231.x	Max. permissible axis-specific jerk at corners for double filter Input: 0.1 to 1000.0 [m/s ³]	PLC RUN	828
MP1232.x	Max. permissible axis-specific jerk at corners for HSC filter Input: 0.1 to 1000.0 [m/s ³]	PLC RUN	828
MP1233.x	Max. permissible axis-specific jerk at corners for advanced HSC filter Input: 0.1 to 1000.0 [m/s ³]	PLC RUN	828
MP1240.x	Max. permissible axis-specific jerk at curvature changes for single filter Input: 0.1 to 1000.0 [m/s ³]	PLC RUN	828
MP1241.x	Max. permissible axis-specific jerk at curvature changes for double filter Input: 0.1 to 1000.0 [m/s ³]	PLC RUN	828
MP1242.x	Max. permissible axis-specific jerk at curvature changes for HSC filter Input: 0.1 to 1000.0 [m/s ³]	PLC RUN	828
MP1243.x	Max. permissible axis-specific jerk at curvature changes for advanced HSC filter Input: 0.1 to 1000.0 [m/s ³]	PLC RUN	828

MP	Function and input	Software version and behavior	Page
MP1250.x	Factor for axis-specific jerk at corners at rapid traverse (from value in MP123x.x) Input: 0.0000 to 30.0000 1: No change at rapid traverse	PLC RUN	828
MP1262	Filter order used for HSC filters Input: 0 to 31 [filter order] 31: Default	PLC RUN	1100
MP1263	Filter order used for advanced HSC filters Input: 0 to 31 [filter order] 31: Default	PLC RUN	1100
MP1290	Only with option #40: Maximum angle tolerance for DCM (Dynamic Collision Monitoring) Input: 0.0000° to 3.0000° 3: Default	PLC RUN	1430
MP1292	Only with option #40: Manual oversize for DCM (Dynamic Collision Monitoring) Input: 0 to 1000 [mm] 0: Default	PLC RUN	1430
MP1294	Only with option #40: Higher traversing speed for Dynamic Collision Monitoring (DCM) through movement of only a single axis Input: 0: Function inactive 1: Function active	PLC RUN	1427
MP1320	Direction for traversing the reference marks Format: %xxxxxxxxxxxxxxxxxxxx Input: Bit 0 to Bit 17 represent axis 1 to 18 0: Positive 1: Negative	PLC RUN	802
MP1330.x	Velocity for traversing the reference marks Input: 80 to 1 000 000 [mm/min]	PLC RUN	802
MP1331.x	Velocity for leaving the reference mark end position for axes 1 to 18 (only for rotary encoders MP1350 = 2) Input: 10 to 1 000 000 [mm/min]	PLC RUN	802
MP1340.x	Sequence for traversing the reference marks Input: 0: No evaluation of reference marks 1 to 18: Axis 1 to 18	PLC RUN REF	802

MP	Function and input	Software version and behavior	Page
MP1350.x	Sequence for finding the reference mark Input: 0: Linear encoder with distance-coded reference marks (old routine), reference run required 1: Position encoder with one reference mark 2: Special type (length measurement with ROD) 3: Linear encoder with distance-coded reference marks (new routine) 4: Same as 3 except that two reference marks are evaluated 5: Encoder with EnDat interface, no reference run required 6: Reference pulse via fast PLC input 7: Speed encoder with EnDat interface at the position and speed encoder inputs 8: Incremental encoder with EnDat 2.2 interface, reference run required (only at position encoder input)	PLC RUN REF	802
MP1352	Activate the software limit switches before traversing the reference marks Format: %xxxxxxxxxxxxxxxxxxxx Input: Bit 0 to Bit 17 represent axis 1 to 18 0: Software limit switch not active 1: Software limit switch active		802
MP1355	Double reference run Format: %xxxxxxxxxxxxxxxxxxxx Input: Bit 0 to Bit 17 represent axis 1 to 18 0: Reference run as defined in MP1350.x 1: Double reference run	PLC RUN REF	803
MP1356.x	Distance between speed and position encoder for double reference run. Input: -1.79769313486E+308 to +1.79769313486E+308 [mm] or [°]	PLC RUN REF	803
MP1357.x	W1032 for double reference run Input: 0: Reset W1032 if the reference run has been over the EnDat interface of the speed encoder 1: Reset W1032 if the reference mark was traversed with the position encoder	PLC RUN	803
MP1358.x	Evaluate multiturn information Format: %xxxxxxxxxxxxxxxxxxxx Input: Bit 0 to Bit 17 represent axis 1 to 18 0: Do not evaluate multiturn information 1: Evaluate multiturn information	606420-03 PLC RUN	614
MP1360.x	Fast PLC input for reference pulse Input: 0: No fast PLC input for reference pulse 1 to 5: Fast PLC inputs 1 to 5 (MP4130.x)	PLC RUN REF	803



MP	Function and input	Software version and behavior	Page
MP1391 MP1391.0 MP1391.1	Velocity and acceleration feedforward control in the MANUAL and HANDWHEEL operating modes Format: %xxxxxxxxxxxxxxxxxxxx Velocity feedforward control Input: Bit 0 to Bit 17 represent axis 1 to 18 0: Inactive 1: Active Acceleration feedforward control Input: Bit 0 to Bit 17 represent axis 1 to 18 0: Inactive 1: Active	PLC RUN	832, 832
MP1392	Velocity feedforward in the operating modes Program Run Single Block, Program Run Full Sequence and Positioning with Manual Data Input Format: %xxxxxxxxxxxxxxxxxxxx Input: Bit 0 to Bit 17 represent axis 1 to 18 0: Operation with following error (lag) 1: Operation with velocity feedforward control	PLC RUN	832
MP1396.x	Feedback control with velocity semifeedforward Input: 0.001 to 0.999 1: Velocity feedforward control	PLC RUN	838



4.4.3 Operation with velocity feedforward control

MP	Function and input	Software version and behavior	Page
MP1410.x	Position monitoring for operation with velocity feedforward control (erasable) Input: 0.0010 to 30.0000 [mm] Recommended: 0.5 mm	PLC RUN	896
MP1420.x	Position monitoring for operation with velocity feedforward control (emergency stop) Input: 0.0010 to 30.0000 [mm] Recommended: 2 mm	PLC RUN	896
MP1510.x	k_V factor for velocity feedforward control Input: 0.100 to 1 000.000 [(m/min)/mm]	PLC RUN	836
MP1511.x	Factor for stick-slip friction compensation Input: 0 to 16 777 215 [s]	PLC RUN	661
MP1512.x	Limitation of the amount of the stick-slip friction compensation Input: 0 to 16 777 215 [counting steps]	PLC RUN	661
MP1513.x	Feed-rate limitation for stick-slip friction compensation Input: 0 to 300 000 [mm/min]	PLC RUN	661
MP1515.x	k_V factor for velocity feedforward control effective after M105 Input: 0.100 to 1000.000 [m/(min*mm)]	PLC RUN	836
MP1516.x	k_V factor for velocity semifeedforward control Input: 0.100 to 20.000 [(m/min)/mm]	PLC RUN	838
MP1521	Transient response during acceleration and deceleration Input: 1 to 255 [ms] 0: Function inactive	PLC RUN	818
MP1522	Feed-rate smoothing Input: 0 to 60 [ms] 0: Function inactive	PLC RUN	820



4.4.4 Servo control with following error

MP	Function and input	Software version and behavior	Page
MP1710.x	Position monitoring for operation with following error (erasable) Input: 0.0000 to 300.0000 [mm] Recommended: 1.2 · following error	PLC RUN	896
MP1720.x	Position monitoring for operation with following error (emergency stop) Input: 0.0000 to 300.0000 [mm] Recommended: 1.4 · following error	PLC RUN	896
MP1810.x	k_V factor for control with following error Input: 0.100 to 20.000 [(m/min)/mm]	PLC RUN	834
MP1815.x	k_V factor for control with following error effective after M105 Input: 0.100 to 20.000 [(m/min)/mm]	PLC RUN	834
MP1820.x	Multiplier for the k_V factor Input: 0.001 to 1.00000	PLC RUN	843
MP1830.x	Characteristic curve kink point Input: 0.000 to 100.000 [%]	PLC RUN	843

4.4.5 Integrated speed and current control

MP	Function and input	Software version and behavior	Page
MP2000.x	Performance of control loop (software option 49) Input: 0: Single-speed axis 1: Double-speed axis		1077
MP2040 MP2040.0-7	Axis groups (for drive enabling through MP4132) Format: %xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx (Bit 31 = spindle) Input: 0: Axis/spindle not assigned (switching only through emergency-stop inputs) 1: Axis/spindle assigned Axis group 1 to 8	PLC RUN	863
MP2100.x	Type of axis power modules (change possible without automatic restart) Input: Name from file <Motor.amp>	PLC RUN	1007
MP2160.x	Field weakening with synchronous motors Input: 0: No voltage-protection module 1: Voltage-protection module present 2: Limited field weakening without voltage-protection module for EcoDyn motors 9: Weakened-field operation permitted, alternative to 1 10: Feed axis in EcoDyn mode, alternative to 2		889, 882
MP2170	Waiting time between the switch-on of the drive and the drive's standby signal Input: 0.001 to 4.999 [s] 0: 2 [s]		863
MP2172	Delay of the STO.A.x signal (inverter enable) at internal emergency stop (e.g. standstill monitoring, PLC via error table...) Input: 0 to 60 [s] as an integer 0: 3 [s] default		931
MP2173.x	Pulse switch-off of the power modules Input: 0.2 to 100.000 [s] 0: 3 [s] default		934



MP	Function and input	Software version and behavior	Page
MP2180.x	PWM frequency Input: 0: $f_{PWM} = 5000$ Hz 3200 to 3999: $f_{PWM} = 3333$ Hz 4000 to 4999: $f_{PWM} = 4000$ Hz 5000 to 5999: $f_{PWM} = 5000$ Hz 6000 to 7999: $f_{PWM} = 6666$ Hz 8000 to 9999: $f_{PWM} = 8000$ Hz 10000: $f_{PWM} = 10000$ Hz	PLC, RUN	1083
MP2182.x	Cycle time of current controller at double the fundamental PWM frequency Input: 0: Standard case: MP2180 = [3333 Hz to 5000 Hz] with single-speed axes MP2180 = [3333 Hz to 10000 Hz] with double-speed axes (current controller cycle time = $1 / (2 * MP2180)$) 1: Reserved 2: Special case 2, CC61xx: MP2180 = [3333 Hz to 5000 Hz] with speed-dependent doubling (MP2186, MP2188) of the fundamental PWM frequency from MP2180 with double speed axes (current controller cycle time = $1 / (4 * MP2180)$)	PLC RUN	1083
MP2184.x	Only CC424(B) (not CC61xx): Reserved Input: 0		
MP2186.x	Shaft speed at which the factor 1 PWM frequency is switched to a factor 2 (twice the PWM frequency) Input: 0 to 100 000 [rpm]		1086
MP2188.x	Shaft speed at which the factor 2 PWM frequency is switched to a factor 1 Input: 0 to 100 000 [rpm]		1086
MP2190.x	DC-link voltage U_z of the power supply module Input: 0 to 3000 [V] *: Entry from the power supply module table HEIDENHAIN inverters: Non-regenerative: 565 V Regenerative: 650 V		1020

MP	Function and input	Software version and behavior	Page
MP2195	<p>Handling of status signals from HEIDENHAIN power supply units. MP2195 can also be overwritten by the PLC and the LSV2 protocol.</p> <p>Input: Bit 0 – Status signals that are already active during control power-up</p> <p>0: Missing signals are ignored 1: Missing signals are evaluated</p> <p>Bit 1 – $\overline{\text{ERR.UZ.GR}}$ signal</p> <p>0: Error message is not suppressed 1: Error message is suppressed</p> <p>Bit 2 – $\overline{\text{ERR.TMP}}$ signal</p> <p>0: Error message is not suppressed 1: Error message is suppressed</p> <p>Bit 3 – Reserved</p> <p>Bit 4 – $\overline{\text{ERR.IZ.GR}}$ signal</p> <p>0: Error message is not suppressed 1: Error message is suppressed</p> <p>Bit 5 – RDY.PS signal</p> <p>0: Error message is not suppressed 1: Error message is suppressed</p> <p>Bit 6 – $\overline{\text{ERR.ILEAK}}$ signal</p> <p>0: Error message is not suppressed 1: Error message is suppressed</p> <p>Bit 7 - PF.PS.AC</p> <p>0: Error message is not suppressed 1: Error message is suppressed</p> <p>Bit 8 - PF.PS.DC</p> <p>0: Error message is not suppressed 1: Error message is suppressed</p> <p>Bit 9 – Monitoring the DC-link voltage Uz or LIFT-OFF function</p> <p>0: Uz monitoring and LIFTOFF function are active 1: Uz monitoring and LIFTOFF function are not active</p> <p>Bit 10 – Power monitoring of the UV(R)</p> <p>0: Monitoring active 1: Monitoring not active</p>	PLC	924, 902
MP2196.x	<p>Identifier for power supply module in MP2198.x</p> <p>Input: P, Q, R or T</p>	PLC RUN	1007
MP2198.x	<p>Type of power supply module</p> <p>Input: Name from file SUPPLY.SPY</p> <p>Default setting: Empty string</p>	PLC RUN	1007
MP2199.x	<p>Assignment of the drive to the power supply module</p> <p>Input: 0: The axis/spindle is assigned to the UV in MP2198.0 1: The axis/spindle is assigned to the UV in MP2198.1</p>	PLC RUN	1007
MP2200.x	<p>Motor model</p> <p>Input: Name of the selected motor (is entered by the iTNC)</p>	PLC RUN	1007

MP	Function and input	Software version and behavior	Page
MP2000.x	Performance of control loop (software option 49) Input: 0: Single-speed axis 1: Double-speed axis	RESET	1077
MP2202.x	Overwrite "Line count" from the motor table Input: *: Input from the motor table active 0: No speed encoder (volts-per-hertz control mode) 1 to 999 999	PLC RUN	1007
MP2204.x	Overwrite "Counting direction" from the motor table Input: *: Input from the motor table active +: Positive counting direction -: Negative counting direction	RESET	1007
MP2206.x	Overwrite "Type of encoder" from the motor table Input: *: Input from the motor table active 0: No speed encoder (volts-per-hertz control mode) 1: Incremental rotary encoder with Z1 track 2: Absolute rotary encoder with EnDat interface (aligned) 3: Absolute linear encoder with EnDat interface 4: Linear motor with one reference mark (CC424(B)) 5: Absolute rotary encoder with EnDat interface (not aligned) 6: Incremental rotary encoder without Z1 track 7: Incremental rotary encoder with distance-coded reference marks (nonaligned) 8: Incremental linear encoder with distance-coded reference marks (not aligned) 9: Aligned rotary encoder with EnDat 2.2 interface 10: Nonaligned rotary encoder with EnDat 2.2 interface 11: Linear encoder with EnDat 2.2 interface 12: EnDat2.2 incremental, position encoder (e.g. LIP 200) 13 bis 18: Reserved 19: EnDat 2.2 incremental, speed encoder (e.g. ERM + EIB)	RESET	1007
MP2208.x	Inductance of the series reactor Input: * = Entry from the motor table active Value of the series reactor in [μ H]	RESET	1015
MP2209.x	Mass moment of inertia of a drive motor Input: * = Entry from the motor table active Value of the mass moment of inertia in [kgm^2]	RESET	1145
MP2210.x	Reduction of the nominal voltage (and, as a result, the nominal magnetizing current) at the rpm for field weakening during idle running. Input: 0 to 60 [%] 0 = Function inactive		1103

MP	Function and input	Software version and behavior	Page
MP2220.x	<p>Monitoring functions</p> <p>Format: %xxxxxxxxxxxxxxxx</p> <p>Input: Bit 0 – Monitoring the reference mark</p> <p>0: Monitoring active 1: Monitoring inactive</p> <p>Bit 1 – Monitoring the direction of rotation</p> <p>0: Monitoring active 1: Monitoring inactive</p> <p>Bit 2 – Power limit of spindle with $\overline{\text{ERR.IZ.GR}}$ (only for HEIDENHAIN inverters, except UE 2xx)</p> <p>0: Power limit active 1: Power limit inactive</p> <p>(All HEIDENHAIN inverters except UE 2xx)</p> <p>Bit 3 – Switching off the controller when the motor brakes are activated</p> <p>0: Suppress oscillations 1: Oscillations are allowed</p> <p>Bit 4 – Monitoring for excessive temperature</p> <p>0: Active 1: Inactive</p> <p>Bit 5 – Monitoring for insufficient temperature</p> <p>0: Active 1: Inactive</p> <p>Bit 6 – Reserved</p> <p>Bit 7 – Monitoring of encoder input frequency</p> <p>0: Active 1: Inactive</p> <p>Bit 8 – Adjust mechanical offset by gradually increasing the k_V factor</p> <p>0: Active 1: Inactive</p> <p>Bits 9 to 15 – Reserved</p>	PLC RUN	875; 925, 951, 1087
MP2221.x	<p>Special motor functions</p> <p>Format: %xxxxxxxxxxxxxxxx</p> <p>Input:</p> <p>Bit 7 – Switch-on time of the drive</p> <p>0: Reduction of the switch-on time is active 1: Reduction of the switch-on time is not active</p> <p>Bit 10 – Handling of linear and synchronous motors to attain higher milling power with AFC</p> <p>0: Handling for attaining higher milling power is active 1: Handling for attaining higher milling power is not active</p>	PLC RUN	863 1233



MP	Function and input	Software version and behavior	Page
MP2222.x	Special controller functions Format: %xxxxxxxxxxxxxxxxxxx Input: Bit 2 – CC 61xx, EnDat 2.2: Temperature signal not from EnDat 2.2-log but from KTY input 0: Inactive 1: Temperature signal from KTY input Bit 10 – Special sequence for wye/delta switchover on asynchronous motors: Activate the power module and the current controller simultaneously 0: Not active 1: Active Bit 11 – Special sequence for wye/delta switchover on asynchronous motors: Demagnetize the motor during switch-off 0: Not active 1: Active	606 42x-02	613 984
MP2223.x	Special controller functions Format: %xxxxxxxxxxxxxxxxxxx Input: Bit 1 – Automatically take the holding torque from MP2630 in PLC Module 9128/9158 into account 0: Not active 1: Active Bit 7 – Wait for readiness of current controller 0: Wait for readiness of current controller 1: Do not wait for readiness of current controller	606 42x-02 60642x-03	875 982



MP	Function and input	Software version and behavior	Page
MP2230.x	Multiplier for motor standstill current during test of motor brake Input: 0.1 to 30.0 [- motor standstill current] 0: No test of motor brakes, or motor without brake	PLC	927
MP2232.x	Maximum permissible path during test of motor brakes Input: 0 to 10.0000 [mm] or [°]		927
MP2236.x	Reserved Input: 10	606 42x-02	
MP2250.x	Determining the field angle without motor motion Input: 0: Same as input value 2 1: Reserved 2: Method 2 (brakes applied) 3: Method 3 (same as Method 2, but motor brake is not applied) 4: Method 4 (if there is a lot of noise in the encoder signals)	PLC RUN	1114
MP2252.x	Reserved Input: Enter 0	PLC RUN	1114
MP2254.x	Determining the field angle Input: 0: Field angle is determined during operation; soft key has no function (without plausibility test) 1: Reserved 2: Field angle determination via soft key, motor motion is permitted (with plausibility test) 3: Same as 2, but the drive must no longer be switched on via the PLC. The drive is moved immediately!	PLC RUN	1019, 1110
MP2256.x	Determined field angle Input: 0: Field angle does not need to be determined, or has not been determined	PLC RUN	1019, 1120
MP2257.x	Control or encoder identification for the field angle from MP2256.x Input: 0: Field angle does not need to be determined, or has not been determined	PLC RUN	1019, 1120
MP2260.x	"TRC – Torque Ripple Compensation" File name for the torque-ripple-compensation file Input: xx_<MotorNamefromMotorTable>.TRC (generated in TNCopt) No input: No compensation	PLC RUN	1101
MP2261.x	Deactivate compensation Bit 0: Torque ripple compensation Bit 1: Gear error compensation Input: %0000000000000000 1: Compensation not active	PLC RUN	1102

MP	Function and input	Software version and behavior	Page
MP2302.x	Reference value for I^2t monitoring of motor Input: 0 to 1000.000 [\cdot rated current of motor] 0: I^2t monitoring of motor switched off 1: Rated current of motor as reference value	PLC	913
MP2304.x	Reference value for I^2t monitoring of power module Input: 0 to 1000.000 [\cdot rated current of power module] 0: I^2t monitoring of power module switched off 1: Rated current of power module as reference value	PLC	913
MP2308.x	Time between output of the braking signal \overline{BRK} and switching off of the controller (overlap time) Input: 0.001 to 5.000 [s] 0: 0.200 s		925
MP2309.x	Controller parameters adjusted to closed brake Input: 0: Not active 0.001 to 5.000 [s]		925
MP2312.x	Factor for utilization of motors Input: 0 to 1000.000 0: Factor = 1		919
MP2390.x	Maximum braking power Input: 0.1 to 3000.000 [kW] 0: Braking power is not limited		872
MP2392.x	Power limit Input: 0: No power limit 0.1 to 3000.000 [kW]		875
MP2393.x	Power limiting after PLC request Input: 0: No power limit 0.001 to 3000.000 [kW]		875
MP2394.x	Maximum brake power for power failure Input: 0.1 to 3000.000 [kW] 0: Braking power is not limited		872
MP2396.x	Maximum torque Input: 0.1 to 30 000.0 [Nm] 0: Torque is not limited	PLC	875
MP2420.x	Proportional factor of the current controller Input: 0.00 to 9999.99 [V/A] * = automatic calculation of the P factor	PLC	867
MP2430.x	Integral factor of the current controller Input: 0.00 to 9 999 999 [Vs/A] * = automatic calculation of the I factor	PLC	867
MP2440.x	Cutoff frequency of the feedforward current controller Input: 0.1 to 5000.0 [Hz] 0: Feedforward control not active	PLC	869

MP	Function and input	Software version and behavior	Page
MP2450.x	Dead-time compensation Input: 0: Dead-time compensation not active 1: Dead-time compensation active	PLC	869
MP2500.x	Proportional factor of the speed controller Input: 0 to 1 000 000.000 [As]	PLC RUN	848
MP2510.x	Integral factor of the speed controller Input: 0 to 100 000 000 [A]	PLC RUN	848
MP2512.x	Limit of integral factor of the speed controller Input: 0.000 to 30.000 [s] (realistic values: 0.1 to 2.0)	PLC RUN	661, 854
MP2520.x	Differential factor of the speed controller Input: 0 to 1.0000 [As]	PLC RUN	850
MP2530.x	PT ₂ element of the speed controller (second-order time delay element) Input: 0 to 1.0000 [s]	PLC RUN	851
MP2540.x	Reserved	PLC RUN	851
MP2542.x	Damping/phase increase for filter 1 Input: 0 to 40.0 [dB]	PLC RUN	1090
MP2543.x	Damping/phase increase for filter 2 Input: 0 to 40.0 [dB]	PLC RUN	1090
MP2544.x	Damping/phase increase for filter 3 Input: 0 to 40.0 [dB]	PLC RUN	1090
MP2545.x	Damping/phase increase for filter 4 Input: 0 to 40.0 [dB]	PLC RUN	1090
MP2546.x	Damping/phase increase for filter 5 Input: 0 to 40.0 [dB]	PLC RUN	1090
MP2547.x	Damping/phase increase for filter 6 Input: 0 to 40.0 [dB]	606420-03 PLC RUN	1090
MP2550.x	Reserved	PLC RUN	851
MP2552.x	Center/cutoff frequency for filter 1 Input: 0 to 30000.0 [Hz]	PLC RUN	1090
MP2553.x	Center/cutoff frequency for filter 2 Input: 0 to 30000.0 [Hz]	PLC RUN	1090
MP2554.x	Center/cutoff frequency for filter 3 Input: 0 to 30000.0 [Hz]	PLC RUN	1090
MP2555.x	Center/cutoff frequency for filter 4 Input: 0 to 30000.0 [Hz]	PLC RUN	1090



MP	Function and input	Software version and behavior	Page
MP2556.x	Center/cutoff frequency for filter 5 Input: 0 to 30000.0 [Hz]	PLC RUN	1090
MP2557.x	Center/cutoff frequency for filter 6 Input: 0 to 30000.0 [Hz]	606420-03 PLC RUN	1090
MP2560.x	Low-pass filter Input: 0: No low-pass filter 1: 1st-order low-pass filter 2: 2nd-order low-pass filter	PLC RUN	850
MP2560.x	Filter order of the low-pass filter Input: 0 to 20	PLC RUN	1093
MP2561.x	Maximum input frequency for motor encoders Format: %xxxxxxxxxxxxxxxxxxxx Input: Bit 0 – Maximum input frequency for motor encoders 0: 400 kHz 1: 800 kHz	PLC RUN	618
MP2562.x	Filter type for filter 1 Input: 0: No filter 1: PT2 low-pass filter (speed controller) 2: Band-rejection filter (speed controller) 3: Phase increase (speed controller) 11: PT2 low-pass filter (position controller) 12: Band-rejection filter (position controller) 13: Phase increase (position controller)	PLC RUN	1091
MP2563.x	Filter type for filter 2 Input: 0: No filter 1: PT2 low-pass filter (speed controller) 2: Band-rejection filter (speed controller) 3: Phase increase (speed controller) 11: PT2 low-pass filter (position controller) 12: Band-rejection filter (position controller) 13: Phase increase (position controller)	PLC RUN	1091
MP2564.x	Filter type for filter 3 Input: 0: No filter 1: PT2 low-pass filter (speed controller) 2: Band-rejection filter (speed controller) 3: Phase increase (speed controller) 11: PT2 low-pass filter (position controller) 12: Band-rejection filter (position controller) 13: Phase increase (position controller)	PLC RUN	1091

MP	Function and input	Software version and behavior	Page
MP2565.x	Filter type for filter 4 Input: 0: No filter 1: PT2 low-pass filter (speed controller) 2: Band-rejection filter (speed controller) 3: Phase increase (speed controller) 11: PT2 low-pass filter (position controller) 12: Band-rejection filter (position controller) 13: Phase increase (position controller)	PLC RUN	1091
MP2566.x	Filter type for filter 5 Input: 0: No filter 1: PT2 low-pass filter (speed controller) 2: Band-rejection filter (speed controller) 3: Phase increase (speed controller) 11: PT2 low-pass filter (position controller) 12: Band-rejection filter (position controller) 13: Phase increase (position controller)	PLC RUN	1091
MP2567.x	Filter type for filter 8 Input: 0: No filter 1: PT2 low-pass filter (speed controller) 2: Band-rejection filter (speed controller) 3: Phase increase (speed controller) 11: PT2 low-pass filter (position controller) 12: Band-rejection filter (position controller) 13: Phase increase (position controller)	606420-03 PLC RUN	1091
MP2572.x	Bandwidth for filter 1 Input: 0 to 30000.0 [Hz]	PLC RUN	1092
MP2573.x	Bandwidth for filter 2 Input: 0 to 30000.0 [Hz]	PLC RUN	1092
MP2574.x	Bandwidth for filter 3 Input: 0 to 30000.0 [Hz]	PLC RUN	1092
MP2575.x	Bandwidth for filter 4 Input: 0 to 30000.0 [Hz]	PLC RUN	1092
MP2576.x	Bandwidth for filter 5 Input: 0 to 30000.0 [Hz]	606420-03 PLC RUN	1092
MP2577.x	Bandwidth for filter 5 Input: 0 to 30000.0 [Hz]	PLC RUN	1092
MP2590.x	Braking ramp in an emergency stop Input: 0.001 to 999.999 [min ⁻¹ /ms] 0: Function inactive	PLC RUN	871
MP2600.x	Acceleration feedforward control Input: 0 to 100.0000 [A/(rev/s ²)]	PLC	854

MP	Function and input	Software version and behavior	Page
MP2602.x	IPC time constant T_1 Input: 0.0001 to 1.0000 [s] 0: IPC inactive	PLC RUN	855
MP2604.x	IPC time constant T_2 Input: 0.0001 to 1.0000 [s] 0: IPC inactive	PLC RUN	855
MP2606.x	Jerk feedforward control. Minimizing the following error (due to mechanical deformation) during the jerk phase Input: 0.000 to 0.800	PLC RUN	856
MP2607.x	Damping factor for active damping Input: 0 to 30.000 0: No damping 1.5: Typical damping factor	606 42x-02 PLC RUN	852
MP2608.x	Damping time constant for active damping Input: 0.000 to 0.9999 [s] 0: No damping 0.005 to 0.02: Typical damping time constant	606 42x-02 PLC RUN	852
MP2609.x	Filter frequency for CPF Input: 0 to 300.0 [Hz]	606420-03 PLC RUN	1105
MP2610.x	Friction compensation at low speeds (effective only with velocity feedforward control) Input: 0 to 100.0000 [A] 0: No friction compensation (or axis is analog)	PLC RUN	662
MP2610.x	Friction compensation at low speed Input: 0 to 30.0000 [A] (effective value) 0: No friction compensation	PLC RUN	1106
MP2612.x	Delay of friction compensation (effective only during operation with velocity feedforward control) Input: 0.0000 to 1.0000 [s] (typically: 0.015 s) 0: No friction compensation (or axis is analog)	PLC RUN	662
MP2612.x	Input: 0.000 to 1.000 [mm] or [°] 0: No friction compensation 0.1: Typical input value Distance before the reversal point from which a reduction of the current from MP2610.x is to go into effect	PLC RUN	1106
MP2614.x	Distance after the reversal point from which a reduction of the current from MP2610.x is to go into effect Input: 0.000 to 1.000 [mm] or [°] 0: Friction compensation same as CC 422 0.1: Typical input value	PLC RUN	1106

MP	Function and input	Software version and behavior	Page
MP2620.x	Friction compensation Input: 0 to 100.000 [A] 0: No friction compensation (or axis is analog)	PLC RUN	662
MP2630.x	Holding current Input: -100.000 to +100.000 [A]	PLC RUN	857
MP2640.x	Torsion compensation between position encoder and speed encoder Input: 0.001 to 100.000 [μ m/A] 0: Not active		663
MP2690.x	Band limitation of the compensation values Input: 0.0001 to 1.0000 [seconds] 0 = Not active	60642x-03	649
MP2700.x	Activation of expanded compensations (software options) Bit 0: CTC compensation Bit 1: PAC compensation Bit 2: LAC compensation Bit 3: MAC compensation Bit 4: ACC compensation Input: %000000000000000000 0: Not active 1: Active	606 42x-02	700
MP2710.x	ACC: Principle of function of automatic mode Input: 0 to 2 0: Robust standard mode with an average application of a damping signal (recommended setting) 1: Chatter damping is increased for the selected axis; this can sometimes be advisable for one axis or two axes (e.g. vertical axes) and results in slightly reduced control accuracy (following error increases slightly) 2: Chatter damping is significantly increased for the selected axis; this is usually advisable for no more than one axis and results in slightly reduced control accuracy (following error increases)	606 42x-03	700



MP	Function and input	Software version and behavior	Page
MP2900.x	Tensioning torque between master and slave for master-slave torque control (entry for the slave axis) Input: -100.00 to +100.00 [Nm]	PLC	791
MP2910.x	P factor of the torque controller for master-slave torque control (entry for the slave axis) Input: 0.00 to 999.99 [1/(Nm · min)]	PLC	791
MP2912.x	Setting for master-slave torque control Input: 0: The output of the torque-adjustment controller (= speed compensation value) is distributed evenly to master and slave. 1: The output of the torque-adjustment controller (= speed compensation value) is only distributed to the slave.	PLC	791



4.4.6 Spindle

MP	Function and input	Software version and behavior	Page
MP3010	Output of speed, gear range Input: 0: No output of spindle speed 1: Speed code if the speed changes 2: Speed code at every TOOL CALL 3: Nominal speed value always, G code if the gear range shifts 4: Nominal speed value always, G code at every TOOL CALL 5: Nominal speed value always, no G code 6: Same as 3, but with controlled spindle for orientation 7: Same as 4, but with controlled spindle for orientation 8: Same as 5, but with controlled spindle for orientation	PLC RUN	944
MP3011	Function of analog output S, if MP3010 < 3 Input: 0: No special function 1: Voltage is proportional to the current contouring feed rate, depending on MP3012 2: Voltage is defined as through Module 9130 3: Voltage is defined through M functions (M200 to M204)		1561
MP3012	Feed rate from output of an analog voltage of 10 V, MP3011 = 1 Input: 0 to 300 000 [mm/min]		1561
MP3013.x	Characteristic curve kink points (velocity) for output of the analog voltage with M202 Input: 10 to 300 000 [mm/min]	PLC RUN	1562
MP3014.x	Characteristic curve kink points (voltage) for output of the analog voltage with M202 Input: 0.000 to 9.999 [V]	PLC RUN	1562
MP3020	Speed range for S code output Format: xxyz xx: S code for minimum speed yy: S code for maximum speed z: Speed increment Input: 0 to 99 999	PLC RUN	965
MP3030	Behavior of the spindle Input: Bit 0 – 0: Axis stop for TOOL CALL S 1: No axis stop for TOOL CALL S Bit 1: Zero spindle speed when switching to another gear range 0: Reduce speed to 0 1: Do not reduce speed to 0	PLC RUN	960, 1564



MP	Function and input	Software version and behavior	Page
MP3120	Zero speed permitted Input: 0: S = 0 permitted 1: S = 0 not permitted	PLC RUN	958
MP3130	Polarity of the nominal spindle speed Input: 0: M03 positive, M04 negative 1: M03 negative, M04 positive 2: M03 and M04 positive 4: M03 and M04 negative	PLC RUN	957
MP3140	Counting direction of spindle position encoder output signals Input: 0: Positive counting direction with M03 1: Negative counting direction with M03	PLC RUN	957
MP3142	Line count of the rotary encoder on the spindle Input: 100 to 100 000 [lines]	PLC RUN	948
MP3143	Mounting configuration of the spindle position encoder Input: 0: Position encoder directly on the first spindle 1: Position encoder via transmission (ratio in MP3450.x and MP3451.x); X30 pin 1: reference pulse 2: Position encoder via transmission (ratio in MP3450 and MP3451); X30 pin 1: reference pulse release 3: Same as input value 1, except that the second reference pulse is evaluated. 4: Reference-mark evaluation of the spindle via EnDat. The encoder must be mounted directly (without transmission). No reference pulse is necessary. A new reference-mark evaluation via marker M4015 may only be performed at standstill. 5: The position encoder is mounted directly (same as input value 0) and the transmission ratio is also evaluated. (rigid tapping will also be possible without a spindle position encoder!)	PLC RUN	948
MP3210.0-7	Analog nominal spindle voltage at rated speed for the gear ranges 1 to 8 Input: 0 to 100.000 [V] Digital spindle motor revolutions at rated speed for the gear ranges 1 to 8 Input: 0 to 100.000 [1000 min ⁻¹]	PLC RUN	958
MP3240.1	Analog spindle: Minimum nominal value voltage Input: 0 to 9.999 [V] Digital spindle: Minimum motor speed Input: 0 to 9.999 [1000 min ⁻¹]	PLC RUN	958, 958

MP	Function and input	Software version and behavior	Page
MP3240.2	Analog spindle: Spindle jog voltage for gear shifting (M4009/M4010) Input: 0 to 9.999 [V] Digital spindle: Motor speed for gear shifting (M4009/M4010) Input: 0 to 9.999 [1000 min ⁻¹]		
MP3310 MP3310.0 MP3310.1	Limitation for spindle speed override Input: 0 to 150 [%] Upper limit Lower limit	PLC RUN	962
MP3350	Maximum permissible overshoot of the spindle speed in percent of the nominal spindle speed Input: 0 to 100 [%]	PLC RUN	955
MP3351	Entry of an absolute value for the permissible overshoot of the spindle speed Input: 0.001 to 100000.000 [rpm] 0 = Monitoring off	PLC RUN	955
MP3411.0-7	Ramp gradient of the spindle with M03 and M04 for gear ranges 1 to 8 Input: Analog axes: 0 to 1.999 [V/ms] Digital axes: 0 to 1.999 [1000 min ⁻¹ /ms]	PLC RUN	954
MP3412 MP3412.0 MP3412.1 MP3412.2 MP3412.3	Multiplication factor for MP3411.x Input: 0.000 to 1.999 With M05 With oriented spindle stop With tapping with floating tap holder With rigid tapping	PLC RUN	954, 970, 975, 979
MP3415 MP3415.0 MP3415.1 MP3415.2 MP3415.3	Overshoot behavior of the spindle with M03, M04 and M05 Input: 0 to 1000 [ms] With M03, M04 and M05 For oriented spindle stop With tapping With rigid tapping	PLC RUN	958, 958, 958, 958
MP3420	Spindle positioning window Input: 0 to 360.0000 [°]	PLC RUN	970
MP3430	Deviation of the reference mark from the desired position (spindle preset) Input: 0 to 360 [°]	PLC RUN	970
MP3440.0-7	k _V factor for spindle orientation for gear ranges 1 to 8 Input: 0.1 to 10 [(1000°/min) /°]	PLC RUN	970

MP	Function and input	Software version and behavior	Page
MP3450.0-7	Number of spindle position-encoder revolutions for gear ranges 1 to 8 Input: 0 to 65 535 0: No transmission	PLC RUN	948
MP3451.0-7	Number of spindle revolutions for gear ranges 1 to 8 Input: 0 to 65 535 0: No transmission	PLC RUN	948
MP3510.0-7	Rated speed for the gear ranges 1 to 8 Input: 0 to 99 999.999 [min ⁻¹]	PLC RUN	958
MP3515.0-7	Maximum spindle speed for gear ranges 1 to 8 Input: 0 to 99 999.999 [min ⁻¹]	PLC RUN	962
MP3520.0 MP3520.1	Speed activation through marker M4011 Input: 0 to 99 999.999 [min ⁻¹] Spindle speed for oriented stop Input: 0 to 99 999.999 [min ⁻¹]	PLC RUN	973, 970
MP3530	Increased spindle power for roughing Input: 0 = Not active 1 = Increased spindle power for roughing		956
MP3540	Permissible spindle speed shortfall Input: 0.001 to 0.999 0: Monitoring not active		956
MP3542	Minimum spindle speed as of which the monitoring in MP3540 becomes active Input: 0.001 to 0.999 0: Monitoring not active		956
MP3550	Delay of emergency-stop reaction of spindles Input: 0.001 to 0.100 [s] 0: Delay not active		872

4.4.7 Integrated PLC

MP	Function and input	Software version and behavior	Page
MP4000.0-63	Options for the conditional compilation of the PLC program		1688
MP4020	<p>PLC functions</p> <p>Format: %xxxxxxxxxxxxxx</p> <p>Input: Bit 0 to bit 4: Reserved</p> <p>Bit 5: Single or double spindle operation</p> <p>0: Single-spindle operation</p> <p>1: Double-spindle operation</p> <p>Bit 6 – Reserved</p> <p>Bit 7 – Reserved</p> <p>Bit 8 – Behavior after an ext. emergency stop</p> <p>0: "Approach position" is not automatically activated</p> <p>1: "Approach position" is automatically activated</p> <p>Bit 9 – Behavior of a simulated key</p> <p>0: Simulated key is transferred immediately to the NC</p> <p>1: Simulated key is processed first by an active PLC window before being transferred to the NC</p> <p>Bit 10 – Behavior of a locked key</p> <p>0: Locked key only works on the active PLC window</p> <p>1: Locked key works on neither the active PLC window nor on the NC</p> <p>Bit 11 – PLC counter in MP4120.x</p> <p>0: Input in PLC cycles</p> <p>1: Input in seconds</p> <p>Bit 12 – Font size in PLC window</p> <p>0: Automatic adaptation of font size to screen</p> <p>1: Font size for BF 120</p> <p>Bit 13 – Reserved, set to 0</p> <p>Bit 14 – PLC module interface for 18 axes</p> <p>0: PLC module interface for 14 axes plus spindle (axes = bit 0 to bit 14, spindle = bit 15).</p> <p>1: PLC module interface for 18 axes plus spindle (axes = bit 0 to bit 17, spindle = bit 31).</p> <p>Bit 15 – Saving the changes to tables to memory medium</p> <p>0: Cyclic saving of tables (behavior same as up to and including 606 42x-01 SP 04)</p> <p>1: Save tables each time they are closed</p> <p>Bit 16 – Time at which strobe markers are reset</p> <p>0: Do not reset strobe markers until the action triggered by acknowledgment of the strobe is concluded by the NC</p> <p>1: Reset strobe markers immediately upon acknowledgment (behavior same as up to and including 606 42x-01 SP 04)</p> <p>Bit 17 – PLC functions</p> <p>0: No suppression of "Debug functions of the PLC are active" error message</p> <p>1: Suppression of "Debug functions of the PLC are active" message if MP4753 = 1 has triggered this message.</p>	RESET	844, 985, 1392, 1331, 1567, 1611



MP	Function and input	Software version and behavior	Page
MP4030.x	Assignment of physical/logic PL (not required for HSCI systems)		
MP4031	Monitoring of number of PLs (not required for HSCI systems)		
MP4040	Set PLC output after shutdown	PLC RUN	1261
MP4041	Time after shutdown until setting of the PLC output from MP4044 Input: 0 to 1000 [s]	PLC RUN	1261
MP4043	Delay during shutdown for the PLC to execute final actions Input: 1 to 1000 [s] 0: No delay	PLC RUN	1527
MP4044	PLC output to be set after shutdown Input: Symbolic name of PLC output 0: Cannot be evaluated, results in an error message	PLC RUN	1261
MP4045	Reserved		
MP4050.0-8	Traverse distance for lubrication of axes 1 to 18 Input: 0 to 99 999.999 [m or 1000°]	PLC RUN	629
MP4070	Compensation amount per PLC cycle for lagged-tracking axis error compensation Input: 0.0001 to 0.5000 [mm]	PLC RUN	659
MP4110.0-47	Run time PLC timer T0 to T47 Input: 0 to 1 000 000.000 [s]	PLC RUN	1749
MP4111.96-x	Run time PLC timer T96 to x (defined in OEM.SYS) Input: 0 to 1 000 000.000 [s]	PLC RUN	1749
MP4120.0-47	PLC counter preset value Input: 0 to 1 000 000.000 [s or PLC cycles, depending on MP4020, bit 11]	PLC RUN	1752
MP4130.0 MP4130.1 MP4130.2-5	Fast PLC input for switching the monitoring functions off (with HSCI, this is also possible via MP4132.x) Reserved Fast PLC inputs Input: 0 to 20 000 [Number of the PLC input or symbolic PLC operand] -1: Function inactive		895, 863, 1753

MP	Function and input	Software version and behavior	Page
MP4131.0 MP4131.1 MP4131.2-5	Activation criterion for the fast PLC input for switching the monitoring functions off (with HSCI, this is also possible via MP4132.x) Reserved Activation criterion for fast PLC inputs Input: 0: Activation at low level 1: Activation at high level		895, 863, 1753
MP4132.0-7	Axis-specific drive enable, switch-off of monitoring functions Input: 0 to 20000 [Number of the PLC input or symbolic PLC operand] -1: Function not active		863, 930
MP4210.0-47	Setting a number in the PLC (D768 to D956) Input: -99 999.9999 to +99 999.9999		973, 1739
MP4220.0-4	Setting a number in the PLC (W960 to W968) Input: 10 to 30 000		1739
MP4230.0-31	Setting a number in the PLC (Module 9032) The number of indexes can be increased via an entry in OEM.SYS. Input: -99 999.9999 to +99 999.9999		1739
MP4231.0-31	Setting a number in the PLC (Module 9032) Input: -99 999.9999 to +99 999.9999		1739
MP4310.0-9	General parameters in the PLC (W976 to W994, M4300 to M4459) Format: Number, \$xxx [Hex], %xxxxxxxxxxxxxxxx [Bin] Input: 0 to 65535		1739
MP4400	Profibus configuration Input: %xx Bit 0 – Definition of option sequence 0: Sequence as in *.IOC file 1: Sequence as in MP4000.x Bit 1—reserved		



MP	Function and input	Software version and behavior	Page
MP5030 MP5030.0 MP5030.1 MP5030.2 MP5030.3	Communications protocol Input: 0 = Standard data transfer 1 = Blockwise transfer 2 = Without protocol (only for MP5030.2) Operating mode EXT1 Operating mode EXT2 Operating mode EXT3 (PLC) Operating mode EXT4 (PLC)	PLC RUN CN123	1906
MP5040 MP5040.0 MP5040.1	Data transfer rate in operating mode EXT3 or EXT4 (data transfer through PLC) Input: 0: 110 bps 1: 150 bps 2: 300 bps 3: 600 bps 4: 1200 bps 5: 2400 bps 6: 4800 bps 7: 9600 bps 8: 19200 bps 9: 38400 bps 10: 57600 bps 11: 115200 bps Operating mode EXT3 (PLC) Operating mode EXT4 (PLC)	PLC RUN	1916



4.4.9 3-D touch probe

MP	Function and input	Software version and behavior	Page
MP6010	Selection of the touch probe Input: 0: Touch probe with cable transmission (TS 120, TS 220) 1: Touch probe with infrared transmission (TS 632) 2: Touch probe with infrared transmission (TS 440, TS 640) 3: Battery-free TS 444 touch probe	PLC CN123	1535
MP6120	Probing feed rate (triggering touch probe) Input: 1 to 10 000 [mm/min]	PLC RUN CN123	1540
MP6130	Maximum measuring range Input: 0.001 to 99 999.9999 [mm]	PLC RUN CN123	1540
MP6140	Setup clearance above measuring point Input: 0.001 to 99 999.9999 [mm]	PLC RUN CN123	1540
MP6150	Rapid traverse in probing cycle Input: 10 to 20 000 [mm/min]	PLC RUN CN123	1540
MP6151	Pre-positioning in probing cycle with rapid traverse Input: 0: Pre-position with speed from MP6150 1: Pre-positioning at rapid traverse	PLC RUN CN123	1539
MP6160	M function for probing from opposite orientations Input: -1: Spindle orientation directly by NC 0: Function inactive 1 to 999: Number of the M function for spindle orientation by the PLC	PLC RUN CN123	1544
MP6161	M function for orienting the touch probe before every measuring process Input: -1: Spindle orientation directly by the NC 0: Function inactive 1 to 999: Number of the M function	PLC RUN CN123	1542
MP6162	Orientation angle Input: 0 to 359.9999 [°]	PLC RUN CN123	1542
MP6163	Minimum difference between the current spindle angle and MP6162 before executing an oriented spindle stop Input: 0 to 3.0000 [°]	PLC RUN CN123	1542

MP	Function and input	Software version and behavior	Page
MP6165	Orient the probe before approaching with Cycle 0 or 1, or with manual probing Input: 0: Probe is not oriented before each probing 1: Probe is oriented and always deflected in the same direction	PLC RUN CN123	1540
MP6166	Probing direction of the touch probe with consideration of an active basic rotation (only manual measuring cycles) Input: 0: Inactive 1: Active	PLC RUN CN123	1540
MP6170	Number of measurements in a programmed measurement (touch probe block) Input: 1 to 3	PLC RUN CN123	1545
MP6171	Confidence range for programmed measurement (MP6170 > 1) Input: 0.002 to 0.999 [mm]	PLC RUN CN123	1545
MP6180 MP6180.0 MP6180.1 MP6180.2	Coordinates of the ring gauge center for automatic calibration (Probing Cycle 2) with respect to the machine datum (traverse range 1) Input: 0 to +99 999.9999 [mm] X coordinate Y coordinate Z coordinate	PLC CN123	1544
MP6181 MP6181.0 MP6181.1 MP6181.2	Coordinates of the ring gauge center for automatic calibration (Probing Cycle 2) with respect to the machine datum (traverse range 2) Input: 0 to +99 999.9999 [mm] X coordinate Y coordinate Z coordinate	PLC CN123	1544
MP6182 MP6182.0 MP6182.1 MP6182.2	Coordinate of the ring gauge center for Probing Cycle 2 with respect to the machine datum (traverse range 3) Input: 0 to +99 999.9999 [mm] X coordinate Y coordinate Z coordinate	PLC CN123	1545
MP6185	Distance of probing point below ring top surface during calibration Input: +0.001 to +99 999.9999 [mm]	PLC CN123	1545

4.4.10 Tool measurement with TT

MP	Function and input	Software version and behavior	Page
MP6500	<p>Tool measurement with TT 130</p> <p>Format: %xxxxxxxxxxxxxx</p> <p>Input: Bit 0 – Cycles for tool measurement</p> <p>0: Disabled 1: Not disabled</p> <p>Bit 1 –</p> <p>0: Tool radius measurement allowed. Tool length measurement with rotating spindle 1: Tool radius measurement and individual tooth measurement disabled</p> <p>Bit 2 –</p> <p>0: Tool length measurement with stationary spindle (bit 1=1) 1: Tool length measurement with rotating spindle, only if a tool radius offset (TT: R-OFFS) has been entered in the tool table</p> <p>Bit 3 –</p> <p>0: Tool measurement with spindle orientation 1: Tool measurement without spindle orientation. Individual tooth measurement not possible. Tool radius measurement possibly faulty</p> <p>Bit 4 –</p> <p>0: Automatically determine speed 1: Always use minimum spindle speed</p> <p>Bit 5 – NC stop during tool checking</p> <p>0: The NC program is not stopped when the breakage tolerance is exceeded 1: If the breakage tolerance is exceeded, the NC program is stopped and the error message Tool broken is displayed</p> <p>Bit 6 – NC stop during tool measurement</p> <p>0: The NC program is not stopped when the breakage tolerance is exceeded 1: If the breakage tolerance is exceeded, the NC program is stopped and the error message Touch point inaccessible is displayed</p>	<p>PLC RUN</p>	<p>1552, 1553, 1555, 1557, 1559, 1559</p>

MP	Function and input	Software version and behavior	Page
MP6500	<p>Tool measurement with TT 130</p> <p>Format: %xxxxxxxxxxxxxxxx</p> <p>Input: Bit 7 – Reserved</p> <p>Bit 8 – Probing routine</p> <p>0: Probe contact is probed from several directions</p> <p>1: Probe contact is probed from one direction</p> <p>Bit 9 – Automatic measurement of the direction of the probe contact basic rotation (bit 8 = 1)</p> <p>0: Basic rotation is not measured</p> <p>1: Basic rotation of the probe element is automatically measured</p> <p>Bit 10 – Probing routine (bit 8 = 1)</p> <p>0: Pre-positioning to starting point in all three principal axes</p> <p>1: Pre-positioning to starting point in the tool axis and in the axis of the probing direction (MP6505) (bit 9 = 0)</p> <p>Bit 11 – Tool checking and changing in the tool table</p> <p>0: After Tool checking the tool table is changed</p> <p>1: After Tool checking the tool table is not changed</p> <p>Bit 12 – PLC datum shift</p> <p>0: Do not include</p> <p>1: Include</p> <p>Bit 13 –</p> <p>0: Tool is measured in the tilt position in which the tool touch probe was also calibrated</p> <p>1: Tool is measured in another tilt position</p> <p>Bit 14 – Tool measurement with number of teeth = 0</p> <p>0: Tool measurement with rotating spindle</p> <p>1: Tool measurement with stationary spindle</p>	<p>PLC</p> <p>RUN</p>	
<p>MP6505</p> <p>MP6505.0</p> <p>MP6505.1</p> <p>MP6505.2</p>	<p>Probing direction for tool radius measurement for 3 traverse ranges</p> <p>Input: 0: Positive probing direction of the angle reference axis (0° axis)</p> <p>1: Positive probing direction in the +90° axis</p> <p>2: Negative probing direction of the angle reference axis (0° axis)</p> <p>3: Negative probing direction in the +90° axis</p> <p>Traverse range 1</p> <p>Traverse range 2</p> <p>Traverse range 3</p>	<p>PLC</p> <p>RUN</p> <p>CN123</p>	1554
MP6507	<p>Calculation of the probing feed rate</p> <p>Input: 0: Calculation of the probing feed rate with constant tolerance</p> <p>1: Calculation of the probing feed rate with variable tolerance</p> <p>2: Constant probing feed rate</p>	<p>PLC</p> <p>RUN</p> <p>CN123</p>	1557



MP	Function and input	Software version and behavior	Page
MP6508	SE 642 transmitter/receiver unit at X112 Input: 0 = Not active (TS at X112, TT at X113) 1 = SE 642 with TT and TS at X112	606 42x-02	1535
MP6510 MP6510.0 MP6510.1	Maximum permissible measuring error for tool measurement with rotating tool Input: 0.002 to 0.999 [mm] First measurement error Second measurement error	PLC RUN CN123	1558
MP6520	Probing feed rate for tool measurement with non-rotating tool Input: 1 to 10 000 [mm/min]	PLC RUN CN123	1557
MP6530 MP6530.0 MP6530.1 MP6530.2	Distance from the tool end to the top of the probe contact during tool radius measurement for 3 traverse ranges Input: 0.001 to 99.9999 [mm] Traverse range 1 Traverse range 2 Traverse range 3	PLC RUN CN123	1554
MP6531 MP6531.0 MP6531.1 MP6531.2	Diameter or edge length of the TT 130 probe contact for 3 traverse ranges Input: 0.001 to 99.9999 [mm] Traverse range 1 Traverse range 2 Traverse range 3	PLC RUN	1555
MP6540 MP6540.0 MP6540.1	Safety zone around the probe contact of the TT 130 for pre-positioning Input: 0.001 to 99 999.9999 [mm] Safety clearance in tool axis direction Safety clearance in the plane perpendicular to the tool axis	PLC RUN CN123	1554
MP6550	Rapid traverse in probing cycle for TT 130 Input: 10 to 1 000 000 [mm/min]	PLC RUN CN123	1554
MP6560	M function for spindle orientation during individual tooth measurement Input: -1: Spindle orientation directly by NC 0: Function inactive 1 to 999: Number of the M function for spindle orientation by PLC	PLC RUN CN123	1553

MP	Function and input	Software version and behavior	Page
MP6562	M function before and after tool measurement cycle (TT cycle) Input: -1: Function inactive 0 to 999: Number of the M function	349 490-06 PLC RUN CN123	1560
MP6562.0	M function before the cycle start		
MP6562.1	M function after cycle end		
MP6570	Max. permissible surface cutting speed at the tooth edge Input: 1.0000 to 129.0000 [m/min]	PLC RUN CN123	1557
MP6572	Maximum permissible speed during tool measurement Input: 1 to 1000 [min ⁻¹] 0: 1000 [min ⁻¹]	PLC RUN CN123	1557
MP6580.0-2	Coordinates of the TT 130 probe contact center with respect to the machine datum (traverse range 1) Input: -99 999.9999 to +99 999.9999 [mm]	PLC RUN CN123	1555
MP6581.0-2	Coordinates of the TT 130 probe contact center with respect to the machine datum (traverse range 2) Input: -99 999.9999 to +99 999.9999 [mm]	PLC RUN CN123	1555
MP6582.0-2	Coordinates of the TT 130 probe contact center with respect to the machine datum (traverse range 3) Input: -99 999.9999 to +99 999.9999 [mm]	PLC RUN CN123	1555
MP6585	Monitoring the position of the rotary and additional linear axes during the tool measurement cycles Format: %xxxxxx Input: 0: Axis is not monitored 1: Axis is monitored Bit 0 – A axis Bit 1 – B axis Bit 2 – C axis Bit 3 – U axis Bit 4 – V axis Bit 5 – W axis	PLC RUN CN123	1558

MP	Function and input	Software version and behavior	Page
MP6586 MP6586.0 MP6586.1 MP6586.2 MP6586.3 MP6586.4 MP6586.5	Ref. coordinate for monitoring the position of the rotary and additional linear axes during the tool measurement cycles Input: -99 999.9999 to +99 999.9999 [mm or °] A axis B axis C axis U axis V axis W axis	PLC RUN CN123	1558
MP6600	KinematicsOpt: Maximum permitted change value Input: 0.010 to 1.000 [mm]	PLC RUN CN123	749
MP6601	KinematicsOpt: Radius deviation of the calibration sphere Input: 0.010 to 0.100 [mm]	PLC RUN CN123	749
MP6602	KinematicsOpt: M-function macro for positioning the rotary axes Input: 0 to 999 -1: Function inactive	PLC RUN CN123	750

4.4.11 Tapping

MP	Function and input	Software version and behavior	Page
MP7110.0	Minimum for feed-rate override during tapping Input: 0 to 150 [%]	PLC RUN	975
MP7110.1	Maximum for feed-rate override during tapping Input: 0 to 150 [%]		
MP7120.0	Dwell time for reversal of spindle rotational direction Input: 0 to 65.535 [s]	PLC RUN	975, 975
MP7120.1	Advanced switching time of the spindle during tapping with coded spindle-speed output Input: 0 to 65.535 [s]		
MP7120.2	Spindle slow-down time after reaching the hole depth Input: 0 to 65.535 [s]		
MP7130	Run-in behavior of the spindle during rigid tapping Input: 0.001 to 10 [°/min]	PLC RUN	979
MP7150	Positioning window of the tool axis during rigid tapping Input: 0.0001 to 2 [mm]	PLC RUN	979
MP7160	Spindle response during Cycles 17, 207 and 18 Format: %xxxxx Input: Bit 0 – Oriented spindle stop with Cycles 17 and 207 0: Oriented spindle stop before execution of the cycle 1: No oriented spindle stop before execution of the cycle Bit 1 – Spindle speed 0: Spindle speed is not limited 1: Spindle speed is limited so that it runs with constant speed approx. 1/3 of the time Bit 2 – Spindle in position feedback control 0: Spindle operated without position feedback control 1: Spindle operated with position feedback control Bit 3 – Acceleration feedforward control 0: Active 1: Not active Bit 4 – 0: Tool axis tracks the spindle 1: Tool axis and spindle interpolated	PLC RUN CN123	979



4.4.12 Display and operation

MP	Function and input	Software version and behavior	Page
MP7210	Programming station Input: 0: Controlling and programming 1: Programming station with PLC active 2: Programming station with PLC inactive 3: Programming station with PLC and emergency stop active	CN123	1267
MP7212	Power interrupted message Input: 0: Acknowledge Power interrupted message with CE key 1: Power Interrupted message does not appear	PLC RUN CN123	1263
MP7220	Block number increment for DIN/ISO programs Input: 0 to 250	PLC RUN CN123	1210
MP7224	Disable file types Input: 0: Do not disable 1: Disable Bit 0 – HEIDENHAIN programs *.H Bit 1 – ISO programs *.I Bit 2 – Tool tables *.T Bit 3 – Datum tables *.D Bit 4 – Pallet tables *.P Bit 5 – Text files *.A Bit 6 – HELP files *.HLP Bit 7 – Point tables *.PNT	PLC RUN CN123	1406
MP7224.0	Disable soft keys for file types		
MP7224.1	Protect file types		
MP7224.2	Disable the EDIT ON/OFF soft key		
MP7225	Reserved	PLC RUN	
MP7226.0	Reserved	PLC	
MP7226.1	Size of the datum table Input: 0 to 255 [lines]	RUN CN123	1409

MP	Function and input	Software version and behavior	Page
MP7237 MP7237.0 MP7237.1 MP7237.2	<p>Display and reset the operating times</p> <p>Display PLC operating times</p> <p>Input: Bits 0 to 12 represent PLC operating times 1 to 13 0: Do not display 1: Display</p> <p>Reset PLC operating times with code number 857282</p> <p>Input: Bits 0 to 12 represent PLC operating times 1 to 13 0: Do not reset 1: Reset</p> <p>Reset NC operating times with code number 857282</p> <p>Input: Bit 0 – No function Bit 1 – "Machine on" operating time Bit 2 – "Program run" operating time 0: Do not reset 1: Reset</p>	<p>PLC</p> <p>RUN</p>	1170
MP7238.0-12	<p>Dialog messages for PLC operating times 1 to 13</p> <p>Input: 0 to 4095 Dialog no. from the file (OEM.SYS)</p>	<p>PLC</p> <p>RUN</p>	1170
MP7245	<p>Disable auxiliary cycles and Cycle 32</p> <p>Input: Bit 0 – Auxiliary cycles (18, 33) 0: Auxiliary cycles disabled 1: Auxiliary cycles 18, 33 enabled Bit1 – Cycle 32 0 = Cycle 32 enabled 1 = Cycle 32 disabled</p>	<p>PLC</p> <p>RUN</p>	1258
MP7246	<p>Machine parameter with multiple function</p> <p>Input: %xxxx Bit 0 – Paraxial positioning blocks 0: Permitted 1: Disabled Bit 1 – Clear with DEL key 0: Does not need confirmation 1: Must confirm via soft key Bit 2 – Tool usage file 0: Do not generate 1: Generate Bit 3 – Settings file for AFC 0: Do not generate 1: Generate Bit 4 – ASCII file for machining time per NC block 0: Do not create ASCII file for machining time per NC block 1: Create ASCII file for machining time per NC block</p>	<p>PLC</p> <p>RUN</p>	1212, 1264, 1597
MP7251	<p>Number of global Q parameters starting from Q99 (up to Q60) that are transferred from the OEM cycle to the calling program.</p> <p>Input: 0 to 40</p>	<p>PLC</p> <p>RUN</p>	1715

MP	Function and input	Software version and behavior	Page
MP7260	Number of tools in the tool table MP7260 can also be overwritten by the PLC and the LSV2 protocol. Input: 0 to 30 000	CN123	1567
MP7261.0-15	Number of pockets in the tool magazine 1 to 16 MP7261 can also be overwritten by the PLC and the LSV2 protocol. Input: 0 to 9999	CN123	1567
MP7262	Maximum tool index number for indexed tools MP7262 can also be overwritten by the PLC and the LSV2 protocol. Input: 0 to 9	CN123	1596
MP7263	Pocket table MP7263 can also be overwritten by the PLC and the LSV2 protocol. Format: %xxxxxxx Input: Bit 0 – 0: Show POCKET TABLE soft key 1: Hide POCKET TABLE soft key Bit 1 – Output of the columns for file functions 0: Output only the displayed columns 1: Output all columns Bit 2 – Show the "Edit ON/OFF" soft key in the pocket table 0: Display soft key 1: Do not display soft key Bit 3 – Soft keys "Reset pocket table" and "Reset column T" 0: Display soft keys 1: Do not display soft keys Bit 4 - Deletion possible for a tool that is in the pocket table. Deletion must be confirmed. 0: Deletion impossible 1: Deletion possible (with confirmation) Bit 5 – Deletion of a tool possible even without confirmation (if bit #4 = 1) 0: Deletion not possible without confirmation 1: Deletion possible without confirmation Bit 6 – Deletion of index entries of a tool behaves like deletion of a tool. The settings of bits 4 and 5 also apply to the index entries if bit 6 is set. 0: Deletion always impossible 1: Deletion possible depending on settings in bits 4 and 5	CN123	1567



MP	Function and input	Software version and behavior	Page
MP7266	Elements of the tool table	CN123	1567
	MP7266 can also be overwritten by the PLC and the LSV2 protocol.		
	Input: 0: No display		
	1 to 99: Position in the tool table		
MP7266.0	16-character alphanumeric tool name (NAME)		
MP7266.1	Tool length (L)		
MP7266.2	Tool radius (R)		
MP7266.3	Tool radius 2 for toroidal cutter (R2)		
MP7266.4	Oversize in tool length (DL)		
MP7266.5	Oversize in tool radius (DR)		
MP7266.6	Oversize in tool radius 2 (DR2)		
MP7266.7	Locked tool? (TL)		
MP7266.8	Replacement tool (RT)		
MP7266.9	Maximum tool age, M4543 (TIME1)		
MP7266.10	Maximum tool age TOOL CALL (TIME2)		
MP7266.11	Current tool age (CUR.TIME)		
MP7266.12	Comment on the tool (DOC)		
MP7266.13	Number of tool teeth (CUT)		
MP7266.14	Wear tolerance for tool length (LTOL)		
MP7266.15	Wear tolerance for tool radius (RTOL)		
MP7266.16	Cutting direction of the tool (DIRECT)		
MP7266.17	Additional information for PLC, module 9093 (PLC)		
MP7266.18	Tool offset for tool length (TT:LOFFS)		
MP7266.19	Tool offset for tool radius (TT:ROFFS)		
MP7266.20	Breakage tolerance for tool length (LBREAK)		
MP7266.21	Breakage tolerance for tool radius (RBREAK)		
MP7266.22	Tooth length (LCUTS)		
MP7266.23	Plunge angle (ANGLE)		
MP7266.24	Tool type (TYP)		
MP7266.25	Tool material (TMA)		
MP7266.26	Cutting-data table (CDT)		
MP7266.27	PLC value (PLC-VAL)		
MP7266.28	Probe center offset in reference axis (CAL-OF1)		
MP7266.29	Probe center offset in minor axis (CAL-OF2)		
MP7266.30	Spindle angle during calibration (CAL-ANG)		
MP7266.31	Tool type for pocket table (PTYP)		
MP7266.32	Maximum shaft speed [rpm] (NMAX)		
MP7266.33	Retract tool (LIFTOFF)		

MP	Function and input	Software version and behavior	Page
MP7266.34	PLC value (P1)		
MP7266.35	PLC value (P2)		
MP7266.36	PLC value (P3)		
MP7266.37	Additional kinematics description (KINEMATIC)		
MP7266.38	Point angle for DRILL and CSINK (T-ANGLE)		
MP7266.39	Thread pitch for TAP (PITCH)		
MP7266.40	Control strategy name for AFC (Adaptive Feed Control)		
MP7266.41	Tolerance value or tool radius R2 (R2TOL)		
MP7266.42	Compensation value table for 3DToolComp (DR2TABLE)		
MP7266.43	Time stamp during tool changing (LAST_USE)		
MP7266.44	Active Chatter Control on/off (ACC)		
MP7267	Elements of the pocket table MP7267 can also be overwritten by the PLC and the LSV2 protocol. Input: 0: No display 1 to 99: Position in the pocket table	CN123	1571
MP7267.0	Tool number (T)		
MP7267.1	Special tool (ST)		
MP7267.2	Fixed pocket (F)		
MP7267.3	Locked pocket (L)		
MP7267.4	PLC status (PLC)		
MP7267.5	Tool name (TNAME)		
MP7267.6	Comment on the tool (DOC)		
MP7267.7	Tool type for pocket table (PTYP)		
MP7267.8	Value 1 (P1)		
MP7267.9	Value 2 (P2)		
MP7267.10	Value 3 (P3)		
MP7267.11	Value 4 (P4)		
MP7267.12	Value 5 (P5)		
MP7267.13	Reserve pocket (RSV)		
MP7267.14	Pocket above locked (LOCKED_ABOVE)		
MP7267.15	Pocket below locked (LOCKED_BELOW)		
MP7267.16	Pocket at left locked (LOCKED_LEFT)		
MP7267.17	Pocket at right locked (LOCKED_RIGHT)		
MP7267.18	S1 value (P6)		
MP7267.19	S2 value (P7)		
MP7268	Sorting/hiding columns of the preset table Input: 1 to 99 = Order of the columns 0: Do not display the column	PLC RUN CN123	1209

MP	Function and input	Software version and behavior	Page
MP7270	Feed rate display in the operating modes MANUAL OPERATION and ELECTRONIC HANDWHEEL Input: 0: Display of axis feed rate through pressing an axis direction key (axis-specific feed rate from MP1020) 1: Display of axis feed rate also before an axis direction key is pressed (smallest value from MP1020 for all axes)	PLC RUN CN123	1156
MP7280	Decimal character Input: 0: Decimal comma 1: Decimal point	PLC RUN CN123	1285
MP7281	Depiction of the NC program Input: 0: All blocks completely 1: Current block completely, others line by line 2: All blocks line by line; complete block when editing	PLC RUN CN123	1212
MP7285	Tool length offset in the tool-axis position display Input: 0: Tool length is not offset 1: Tool length is offset	PLC RUN CN123	1148
MP7289	Position display step for the spindle Input: 0: 0.1° 1: 0.05° 2: 0.01° 3: 0.005° 4: 0.001° 5: 0.0005° 6: 0.0001°	PLC RUN CN123	1148
MP7290.0-8	Position display step for axes 1 to 18 Input: 0: 0.1 mm or 0.1° 1: 0.05 mm or 0.05° 2: 0.01 mm or 0.01° 3: 0.005 mm or 0.005° 4: 0.001 mm or 0.001° 5: 0.0005 mm or 0.0005° 6: 0.0001 mm or 0.0001°	PLC RUN CN123	1148
MP7291 MP7291.0 MP7291.1 MP7291.2	Display of axes on the screen Format: SXYZABCUVWxyzabcuvw- Input: Characters 1 to 9 from the right represent lines 1 to 9 Character 10 is spindle S which is always output in line 10. MP7291.0 Display in traverse range 1 MP7291.1 Display in traverse range 2 MP7291.2 Display in traverse range 3	PLC RUN	603

MP	Function and input	Software version and behavior	Page
MP7294	Disable axis-specific "Datum setting" in the preset table Format: %xxxxxxxxxxxxxxxxxxxx Input: Bits 0 to 17 represent axes 1 to 18 0: Not disabled 1: Disabled	PLC RUN CN123	1209
MP7295	Disable "Datum setting" Format: %xxxxxxxxxxxxxxxxxxxx Input: Bits 0 to 17 represent axes 1 to 18 0: Not disabled 1: Disabled	PLC RUN CN123	1207
MP7296	"Datum setting" through axis keys Input: 0: Datum can be set by axis keys and soft key 1: Datum can be set only by soft key	PLC RUN CN123	1207
MP7300	Erasing the status information, tool data and Q parameters Input: 0: Erase the status information, Q parameters and tool data if a program is selected. 1: Erase the status information, Q parameters and tool data if a program is selected and in the event of M02, M30, and END PGM. 2: Erase the status information and tool data if a program is selected. 3: Erase the status information and tool data if a program is selected and in the event of M02, M30, END PGM. 4: Erase the status information and Q parameters if a program is selected. 5: Erase the status information and Q parameters if a program is selected and in the event of M02, M30, END PGM. 6: Erase the status information if a program is selected and in the event of M02, M30, END PGM. 7: Erase the status information if a program is selected and in the event of M02, M30, END PGM.	PLC RUN CN123	1161



MP	Function and input	Software version and behavior	Page
MP7310	<p>Graphic display mode</p> <p>Format: %xxxxxxx</p> <p>Input: Bit 0 – Projection in three planes: 0: German-preferred projection 1: US-preferred projection</p> <p>Bit 1 – Rotating the coordinate system in the working plane by 90°: 0: No rotation 1: Rotation by +90°</p> <p>Bit 2 – BLK form after datum shift: 0: Shifted 1: Not shifted</p> <p>Bit 3 – Display of the cursor position: 0: Not displayed 1: Displayed</p> <p>Bit 4 – Reserved</p> <p>Bit 5 – 3-D graphics during program test 0: 2.5-D and 3-D (only with MC 420 or MC 422B and higher) 1: 2.5-D</p> <p>Bit 6 – Stock removal with an inclined tool 0: Not active 1: Active</p> <p>Bit 6 – Exact evaluation of the column LCUTS (cutting length) from the TOOL.T table in order to display special tools (e.g. saw blade). 0: Free evaluation 1: Exact evaluation for special tools</p>	PLC RUN CN123	1276
MP7312	<p>Limitation of the tool tooth length LCUTS if no value was given for the tooth length in the tool table</p> <p>Input: = 0: No limitation, infinitely long tooth length > 0: Tooth length = 2 * tool radius * MP7312</p>	PLC RUN	1563
MP7315	<p>Tool radius for graphic simulation without TOOL CALL</p> <p>Input: 0.0000 to 99 999.9999 [mm]</p>	PLC RUN CN123	1563
MP7316	<p>Penetration depth of the tool</p> <p>Input: 0.0000 to 99 999.9999 [mm]</p>	PLC RUN CN123	1563

MP	Function and input	Software version and behavior	Page
MP7317	M function for graphic simulation	PLC	1563
MP7317.0	Beginning of graphic simulation Input: 0 to 88	RUN	
MP7317.1	Interruption of the graphic simulation Input: 0 to 88	CN123	
MP7330.0-15	Specification of user parameters 1 to 16 Input: 0 to 99999.00 (no. of the user parameter)	PLC RUN	1264
MP7340.0-15	Dialog messages for user parameters 1 to 16 Input: 0 to 4095 (line number of the PLC dialog message file)	PLC RUN	1264



4.4.13 Colors

MP	Function and input	Software version and behavior	Page
MP7350	Window frames	PLC RUN	1268
MP7351	Error messages	PLC	1268
MP7351.0	Priority 0 (error)	RUN	
MP7351.1	Priority 1 (warning)		
MP7351.2	Priority 2 (information)		
MP7352	"Machine" operating mode display	PLC	1268
MP7352.0	Background	RUN	
MP7352.1	Text for operating mode		
MP7352.2	Dialog		
MP7353	"Programming" operating mode display	PLC	1268
MP7353.0	Background	RUN	
MP7353.1	Text for operating mode		
MP7353.2	Dialog		
MP7354	"Machine" program text display	PLC	1268
MP7354.0	Background	RUN	
MP7354.1	General program text		
MP7354.2	Active block		
MP7354.3	Color of the comments and unused machine parameters in the machine parameter file		
MP7354.4	Background of inactive window		
MP7355	"Programming" program text display	PLC	1268
MP7355.0	Background	RUN	
MP7355.1	General program text		
MP7355.2	Active block		
MP7355.3	Color of the comments and unused machine parameters in the machine parameter file		
MP7355.4	Background of inactive window		
MP7356	Status window and PLC window	PLC	1269
MP7356.0	Background	RUN	
MP7356.1	Axis positions in the status display		
MP7356.2	Status display other than axis positions		
MP7357	"Machine" soft-key display	PLC	1269
MP7357.0	Background	RUN	
MP7357.1	Text color		
MP7357.2	Inactive soft-key row		
MP7357.3	Active soft-key row		

MP	Function and input	Software version and behavior	Page
MP7358	"Programming" soft-key display	PLC	1269
MP7358.0	Background	RUN	
MP7358.1	Text color		
MP7358.2	Inactive soft-key row		
MP7358.3	Active soft-key row		
MP7360	Graphics: 3-D view and plan view	PLC	1269
MP7360.0	Background	RUN	
MP7360.1	Surface		
MP7360.2	3-D: Front face		
MP7360.3	Text display in the graphics window		
MP7360.4	3-D: Lateral face		
MP7360.5	Lowest point of blank form		
MP7360.6	Highest point of blank form (below surface)		
MP7361	Graphics: Projection in three planes	PLC	1269
MP7361.0	Background	RUN	
MP7361.1	Top view		
MP7361.2	Front and side view		
MP7361.3	Axis cross and text in the graphic display		
MP7361.4	Cursor		
MP7362	Additional status display in the graphics window	PLC	1269
MP7362.0	Background of graphic window	RUN	
MP7362.1	Background of status display		
MP7362.2	Status symbols		
MP7362.3	Status values		
MP7362.4	Color of the unselected tabs in the graphics window		
MP7362.5	AFC tab – Background color		
MP7362.6	AFC tab – Color of actual override factor		
MP7362.7	AFC tab – Color of actual spindle factor		
MP7363	Programming graphics	PLC	1269
MP7363.0	Background	RUN	
MP7363.1	Resolved contour		
MP7363.2	Subprograms and frame for zooming		
MP7363.3	Alternative solutions		
MP7363.4	Unresolved contour		
MP7363.5	Rapid traverse movements		
MP7364	Color of the help illustrations for cycles	PLC	1270
MP7364.0-6	Colors 1 to 7 of the graphic program used	RUN	
MP7364.7	Line color (color 8 of the graphic program)		
MP7364.8	Color for highlighted graphic elements if defined in the help illustration		
MP7364.9	Background		



MP	Function and input	Software version and behavior	Page
MP7365	Oscilloscope	PLC	1270
MP7365.0	Background	RUN	
MP7365.1	Grid		
MP7365.2	Cursor and text		
MP7365.3	Selected channel		
MP7365.4-9	Channel 1 to 6		
MP7366	Pop-up window (HELP key, pop-up menus etc.)	PLC	1270
MP7366.0	Background	RUN	
MP7366.1	Text or foreground		
MP7366.2	Current line		
MP7366.3	Title bar		
MP7366.4	Scroll-bar field		
MP7366.5	Scroll bar		
MP7366.6-14	Reserved		
MP7367	Large PLC window	PLC	1270
MP7367.0	Background	RUN	
MP7367.1-7	Colors 1 to 7 (Color 8: MP7350)		
MP7367.8-14	Colors 9 to 15		
MP7368	Calculator	PLC	1270
MP7368.0	Background	RUN	
MP7368.1	Background of displays and keys		
MP7368.2	Key texts ("os" in "cos")		
MP7368.3	Key symbols		
MP7369	Directory tree in PGM MGT	PLC	1270
MP7369.0	Text background	RUN	
MP7369.1	Text		
MP7369.2	Text background of the active folder		
MP7369.3	Line color of the tree structure		
MP7369.4	Folders		
MP7369.5	Drives		
MP7369.6	Text background of the heading in the browser window		
MP7370	Small PLC window	PLC	1270
MP7370.0	Background	RUN	
MP7370.1-15	Colors 1 to 15		
MP7371.0	Status window and PLC window; background window	PLC	1270
MP7371.1	Background elements	RUN	
MP7371.2	Values except positions		
MP7371.3	Positions		

MP	Function and input	Software version and behavior	Page
MP7375.0	smarT.NC; background: Input forms	PLC	1270
MP7375.1	Background: Unselected tab	RUN	
MP7375.2	Background: Tree view and input fields		
MP7375.3	Background: Inactive input field		
MP7375.4	Background: Help graphic		
MP7375.5	Cursor: Tree view and background of current field		
MP7375.6	Cursor: Tree view if the input focus is in the form		
MP7375.7	Text color: Inactive input field		
MP7375.8	Text color: Active input field		
MP7375.9	Text color: Radio buttons		
MP7375.10	Text color: Inactive label		
MP7375.11	Background: Radio/check buttons during "mouse-over"		
MP7375.12	PREDEF global data: Background		
MP7375.13	PREDEF global data: Text		
MP7375.14	Changed global data: Background		
MP7375.15	Changed global data: Text		
MP7375.16	Tool tip: Background		
MP7375.17	Tool tip: Text		
MP7375.18	Dialog title: Background		
MP7375.19	Dialog title: Text		
MP7375.20	Pattern generator: Points of the same height		
MP7375.21	Pattern generator: Currently active points		
MP7375.22	Pattern generator: Deleted points		
MP7375.23	Pattern generator: Hidden points		
MP7375.24	Pattern generator: Rectangle for zoom		



4.4.14 Machining and program run

MP	Function and input	Software version and behavior	Page
MP7400	Look-ahead – Number of NC blocks for advance calculation of the path Input: 0: 256 [blocks] (default) 1: 512 [blocks] 2: 1024 [blocks]	PLC RUN	1210
MP7410	Scaling cycle in two or three axes Input: 0: Scaling cycle is effective in all three principal axes 1: Scaling cycle is effective only in the working plane	PLC RUN CN123	1243
MP7411	Tool data in the touch probe block Input: Bit 0 – 0: Use the calibrated data of the touch probe 1: Use the current tool data from the last TOOL CALL Bit 1 – 0: Only one set of touch probe calibration data 1: Use the tool table to manage more than one set of touch probe calibration data; display the tool name and tool number	PLC RUN CN123	1541
MP7412	Settings for cycles Input: Bit 0: Behavior of marker M4031 in Cycle 209 0 = Set M4031 after spindle orientation 1 = Set M4031 before the spindle orientation	PLC RUN CN123	

MP	Function and input	Software version and behavior	Page
MP7420	Cycles for milling pockets with combined contours Format: %xxxxx Input: Bit 0 – Milling direction for channel milling: 0: Counterclockwise for pockets, clockwise for islands 1: Clockwise for pockets, counterclockwise for islands Bit 1 – Sequence for rough-out and channel milling (only for SL 1): 0: First channel milling, then pocket rough-out 1: First pocket rough-out, then channel milling Bit 2 – Merging of listed contours: 0: Contours are merged only if the tool-center paths intersect 1: Contours are merged if the programmed contours intersect Bit 3 – Rough-out and channel milling to pocket depth or for every infeed 0: Each process uninterrupted to pocket depth 1: Both processes for each pecking depth before proceeding to the next depth Bit 4 – Position after completion of the cycle 0: Tool moves to the same position as before the cycle was called 1: iTNC retracts the axis to the "clearance height"	PLC RUN CN123	1243
MP7430	Overlap factor for pocket milling Input: 0.001 to 1.414	PLC RUN CN123	1241
MP7431	Arc end-point tolerance Input: 0.0001 to 0.016 [mm]	PLC RUN CN123	1263
MP7432	Limit-switch tolerance for M140 / M150 Input: 0.0001 to 1.0000 [mm] 0: Limit-switch tolerance not active		1263



MP	Function and input	Software version and behavior	Page
MP7440	<p>Output of M functions</p> <p>Format: %xxxxxxx</p> <p>Input: Bit 0 – Program stop with M06</p> <p>0: Program stop with M06</p> <p>1: No program stop with M06</p> <p>Bit 1 – Modal cycle call M89</p> <p>0: Normal code transfer of M89 at beginning of block</p> <p>1: Modal cycle call M89 at end of block</p> <p>Bit 2 – Program stop with M functions</p> <p>0: Program stop until acknowledgment of the M function</p> <p>1: No program stop. No waiting for acknowledgment.</p> <p>Bit 3 – Switching of k_v factors with M105/M106:</p> <p>0: Function is not in effect</p> <p>1: Function is effective</p> <p>Bit 4 – Reduced feed rate in the tool axis with M103:</p> <p>0: Function is not in effect</p> <p>1: Function is effective</p> <p>Bit 5 – Reserved</p> <p>Bit 6 – Automatic activation of M134</p> <p>0: M134 must be activated in the NC program</p> <p>1: M134 is automatically activated when an NC program is selected</p> <p>Bit 7 – Internal spindle status of the geometry</p> <p>0: With M00 the internal spindle status switches to M05</p> <p>1: With M00 the internal spindle status does not switch to M05</p>	<p>PLC</p> <p>RUN</p> <p>CN123</p>	<p>771, 834, 836, 1257, 1564, 954</p>
MP7441	<p>Error message during cycle call</p> <p>Format: %xxx</p> <p>Input: Bit 0 –</p> <p>0: Error message Spindle ? is not suppressed</p> <p>1: Error message Spindle ? is suppressed</p> <p>Bit 1: Reserved, enter 0</p> <p>Bit 2 –</p> <p>0: Error message Enter depth as negative is suppressed</p> <p>1: Error message Enter depth as negative is not suppressed</p>	<p>PLC</p> <p>RUN</p> <p>CN123</p>	<p>1257</p>
MP7442	<p>Number of the M function for spindle orientation in the fixed cycles</p> <p>Input: 1 to 999: Number of the M function</p> <p>0: No oriented spindle stop</p> <p>-1: Oriented spindle stop by the NC</p>	<p>PLC</p> <p>RUN</p> <p>CN123</p>	<p>968</p>

MP	Function and input	Software version and behavior	Page
MP7444	Delay time for the change signals (M/S/T) Input: 0, 1, 2 0: Change signal after complete filter run time (previous behavior) 1: Change signal if the nominal feed rate reached the value 0 before the actual value (formed by the filters) did. 2: Change signal if the nominal feed rate reached the value 0 before the actual value (formed by the filters) did, and an additional delay until the actual value reaches the time window the first time.	PLC RUN	1249
MP7450	Offsetting the tool change position from MP951.x in block scan Format: %xxxxxxxxxxxxxxxxxxxx Input: Bits 0 to bit 17 represent axes 1 to 18 0: Do not offset 1: Offset	PLC RUN	1246
MP7451.x	Feed rate for returning to the contour for axes 1 to 18 Input: 10 to 1 000 000 [mm/min]	PLC RUN	1246
MP7460.x	Reserved	PLC RUN CN123	–
MP7461.x	Reserved	PLC RUN CN123	–
MP7470	Maximum contouring feed rate at 100 % override Input: 0 to 300 000 [mm/min] 0: No limitation	PLC RUN CN123	–
MP7471	Maximum velocity of the principal axes during compensating movements through M128 or TCPM Input: 0 to 300 000 [mm/min]	PLC RUN CN123	771
MP7475	Reference for datum table Input: 0: Reference is workpiece datum 1: Reference is machine datum (MP960.x)	PLC RUN CN123	1409

MP	Function and input	Software version and behavior	Page
MP7480 MP7480.0 MP7480.1	Output of the tool or pocket number With TOOL CALL block Input: 0: No output 1: Tool number output only when tool number changes 2: Tool number output for every TOOL CALL block 3: Pocket number and tool number output only when tool number changes 4: Output of the pocket number and tool number for every TOOL CALL block 5: Pocket number and tool number output only when tool number changes. Pocket table is not changed. 6: Pocket number and tool number output for every TOOL CALL block. Pocket table is not changed. With TOOL DEF block Input: 0: No output 1: Tool number output only when tool number changes 2: Tool number output for every TOOL DEF block 3: Pocket number and tool number output only when tool number changes 4: Pocket number and tool number output for every TOOL DEF block	PLC RUN	1610



MP	Function and input	Software version and behavior	Page
MP7481.x	Sequence for new and returned tool when changing tools Format: %xxxxxxx 0: First, output the pocket of the tool to be returned 1: First, output the pocket of the new tool Input: Bit 0: New tool from magazine 1 Bit 1: New tool from magazine 2 Bit 2: New tool from magazine 3 Bit 3: New tool from magazine 4 Bit 4: New tool from magazine 5 Bit 5: New tool from magazine 6 Bit 6: New tool from magazine 7 Bit 7: New tool from magazine 8 Bit 8: New tool from magazine 9 Bit 9: New tool from magazine 10 Bit 10: New tool from magazine 11 Bit 11: New tool from magazine 12 Bit 12: New tool from magazine 13 Bit 13: New tool from magazine 14 Bit 14: New tool from magazine 15 Bit 15: New tool from magazine 16	PLC RUN	1614
MP7481.0	Tool from magazine 1 to be returned		
MP7481.1	Tool from magazine 2 to be returned		
MP7481.2	Tool from magazine 3 to be returned		
MP7481.3	Tool from magazine 4 to be returned		
MP7481.4	Tool from magazine 5 to be returned		
MP7481.5	Tool from magazine 6 to be returned		
MP7481.6	Tool from magazine 7 to be returned		
MP7481.7	Tool from magazine 8 to be returned		
MP7481.8	Tool from magazine 9 to be returned		
MP7481.9	Tool from magazine 10 to be returned		
MP7481.10	Tool from magazine 11 to be returned		
MP7481.11	Tool from magazine 12 to be returned		
MP7481.12	Tool from magazine 13 to be returned		
MP7481.13	Tool from magazine 14 to be returned		
MP7481.14	Tool from magazine 15 to be returned		
MP7481.15	Tool from magazine 16 to be returned		



MP	Function and input	Software version and behavior	Page
MP7482	Pocket coding of the tool magazines Format: %xxxxxxx 0: Variable pocket coding 1: Fixed pocket coding Input: Bit 0: Magazine 1 Bit 1: Magazine 2 Bit 2: Magazine 3 Bit 3: Magazine 4 Bit 4: Magazine 5 Bit 5: Magazine 6 Bit 6: Magazine 7 Bit 7: Magazine 8	PLC RUN	1604
MP7483	Tool name/number for TOOL CALL / TOOL DEF Input: 0: Names and numbers are permitted (as before) 1: Only names are permitted 2: Only numbers are permitted		1611
MP7484.x	Search sequence in tool magazines Input: 0 to 15 [Index aus MP7261] -1: Cancel		1591
MP7485	Add usage time for tool selection Input: 0 to 100 [%] Default setting: 10		1597
MP7490	Functions for traverse ranges Format: %xxxx Input: Bit 0 – 0: Display one traverse range with MOD 1: Display three traverse ranges with MOD Bit 1 – 0: Each traverse range has its own datum (and 3 memories for the positions of the swivel head) 1: One datum for all traverse ranges Bit 2 – Calibration data: Touch probe for workpiece measurement: 0: One set of calibration data for all traverse ranges 1: Every traverse range has its own set of calibration data Bit 3 – Calibration data: Touch probe for tool measurement: 0: One set of calibration data for all traverse ranges 1: Every traverse range has its own set of calibration data	PLC RUN	628, 1541, 1552
MP7492.x	Number of axis in which the same datum is to be set during Datum Setting (with active preset table) Input: 0 to 17 -1: Do not set a datum	PLC RUN	1208
MP7492.0	Datum set in the first axis to		
MP7492.17	Datum set in the 18th axis		

MP	Function and input	Software version and behavior	Page
MP7493	Maximum deviation of the current tool orientation relative to the tool axis when setting a reference point with M114 Input: 0.0000 to 30.0000 [degrees] Default: 0.005	PLC RUN	763
MP7494	Axes for which an exact stop is to occur after positioning Format: %xxxxxxxxxxxxxxxxxxxx Input: Bits 0 to 17 represent axes 1 to 18 0: No exact stop 1: Exact stop	PLC RUN	–
MP7500 (is set via the kinematics table)	Reserved		
MP7502	Functionality of M144/M145 Input: %xxx Bit 0 - 0: M144/M145 not active 1: M144/M145 active Bit 1 – M144/M145 in the automatic modes 0: M144/M145 active 1: M144 is activated automatically at the start of an NC program. It can only be deactivated with M145 during an NC program. Bit 2 – M144/M145 in the manual modes 0: M144/M145 not active 1: M144/M145 active	PLC RUN	775
MP7503	Virtual tool axis – Reapproaching the contour and manual traverse in the current tool-axis direction (FCL2 upgrade function) Input: 0: Inactive 1: Active	PLC RUN	776
MP7506	Selection of kinematics at booting of the control Input: 0 to 999 -1: Function inactive	PLC RUN	720



MP	Function and input	Software version and behavior	Page
MP7507	Selecting the kinematics for the operating mode Input: %xxx Bit 0 0: Kinematics cannot be selected in Editing operating modes 1: Kinematics can be selected in Editing operating modes for simulation in Test Run mode Bit 1 0: Kinematics cannot be selected in Program Run operating modes 1: Kinematics of the real machine can be selected in Program Run operating modes Bit 2 0: 3D ROT soft key is not available in Test Run mode 1: 3D ROT soft key is available in Test Run mode Bit 3 0: Code number KINEMATIC permitted 1: Code number KINEMATIC disabled	PLC RUN	1442
MP7510 (only possible via the old kinematics table)	Reserved		
MP7520 (only possible via the old kinematics table)	Reserved		
MP7530 (only possible via the old kinematics table)	Reserved		
MP7550 (only possible via the old kinematics table)	Reserved		



4.4.15 Hardware

MP	Function and input	Software version and behavior	Page
MP7602	PLC cycle time PLC cycle time is a multiple of the HSCI cycle time (3 ms). Minimum possible value is therefore 12 ms. Input: 0 to 60 [ms] 0 to 12: 12 ms		1087
MP7610.x	Reserved		



MP	Function and input	Software version and behavior	Page
MP7620	Feed-rate override and spindle speed override Format: %xxxxxxx Input: Bit 0 – Feed-rate override if rapid traverse key is pressed in Program Run mode. 0: Override not effective 1: Override effective Bit 1 – No function Bit 2 – Feed-rate override if rapid traverse key and machine direction button are pressed in Manual Operation mode 0: Override not effective 1: Override effective Bit 3 – for TEs without rapid traverse override: 0: Linear characteristic curve for spindle override and feed rate override; unit depends on bit 4 0..150 or 0..15000 1: Non-linear characteristic curve for spindle override and feed rate override, unit 0..15000 Bit 4 – for TEs without rapid traverse override: 0: Spindle override and feed rate override with unit 0 to 150 1: Spindle override and feed rate override with unit 0..15000 Bit 3 – for TEs with rapid traverse override: 0: Linear characteristic curve for spindle-speed and feed-rate override, unit 0..15000 1: Non-linear characteristic curve for spindle override and feed rate override, unit 0..15000 Bit 4 – for TEs with rapid traverse override: No effect Bit 5 – Rapid traverse override instead of spindle override 0: Potentiometer is used for spindle override 1: Potentiometer is used for rapid-traverse override Bit 6 – Feed-rate smoothing 0: Not active 1: Active Bit 7 – Only with functional safety: Adjust potentiometer to F-LIMIT 0: Do not adjust potentiometer to F-LIMIT 1: Adjust potentiometer to F-LIMIT, 100 % corresponds to active SLS limit Bit 8 – Informational text if feed-rate or rapid-traverse override is set to 0 % 0: Informational text inactive 1: Informational text active Bit 9 – HSCI keyboard with three potentiometers 0: Keyboard unit with two potentiometers 1: Keyboard unit for HSCI with three potentiometers	PLC RUN	358, 963, 1156
MP7621	Reserved		–
MP7630	Recovery time after emergency stop test can be configured Input: 1 to 999 [ms] 0: 200 ms	PLC RUN	936

MP	Function and input	Software version and behavior	Page
MP7640.x	<p>Handwheel type MP7640.0 for X23 of the last machine operating panel in the HSCI chain, and MP7640.4 for X29 of the MC main computer</p> <p>Input: 0: No handwheel 1: Reserved 2: HR 130 3: Reserved 4: Reserved 5: Up to three HR 150 via HRA 110 6: HR 410 7, 8: Reserved 9: HR 410 FS 10: Reserved 11: HR420/HR520 without LED activation 12: HR 550FS handwheel with wireless transmission 13: HR520 with LED activation</p>	PLC RUN	1506



MP	Function and input	Software version and behavior	Page
MP7641	<p>Handwheel settings</p> <p>Format: %xxxxxxxxxxxxxx</p> <p>Input:</p> <p>Bit 0 – HR 130, HR 330, HRA 110: Input of the subdivision factor</p> <p>0: Through iTNC keyboard 1: Through PLC Module 9036</p> <p>Bit 1 – HR 420/HR 5xx: With detent positions</p> <p>0: Without detent positions 1: With detent positions</p> <p>Bit 2 – HR 420/HR 5xx: Axis direction keys and rapid traverse</p> <p>0: By the NC 1: By the PLC</p> <p>Bit 3 – HR 420/HR 5xx: NC start / NC stop</p> <p>0: By the NC 1: By the PLC</p> <p>Bit 4 – Handwheel superimposition in the active tool-axis direction</p> <p>0: Behavior as before 1: VT axis can be selected</p> <p>Bit 5 – Inactive behavior of HR 420/HR 5xx</p> <p>0: Report the keys of the HR to the PLC only when the HR is active 1: Report the keys of the HR to the PLC even if the HR is not active</p> <p>Bit 6 – Selecting and traversing auxiliary axes with HR 420/HR 5xx</p> <p>0: Traversing auxiliary axes not possible 1: Traversing auxiliary axes is possible</p> <p>Bit 7 – Teach-In button on HR 5xx</p> <p>0: By the NC 1: By the PLC</p> <p>Bit 8 – CTRL button on HR 5xx</p> <p>0: By the NC 1: By the PLC</p> <p>Bit 9 – PLC soft keys with active HR 420/HR 5xx</p> <p>0: PLC soft keys are not active when HR is active 1: PLC soft keys are active when HR is active</p>	<p>PLC</p> <p>RUN</p>	1506
MP7642.x	<p>Handwheel name</p> <p>MP7642.0 for X23 of the last machine operating panel in the HSCI chain, and MP7642.4 for X29 of the MC main computer</p> <p>Format: String</p> <p>Input: 31 arbitrary characters</p>	<p>60642x-03</p> <p>PLC</p> <p>RUN</p>	1506

MP	Function and input	Software version and behavior	Page
MP7645.x	Initialization parameter for handwheel at X23 of the last machine operating panel in the HSCI chain	PLC	1512,
MP7645.0	Assignment of the keys on handwheel HR 410 Input: 0: Evaluation of the keys by NC, including LEDs 1: Evaluation of the keys by PLC	RUN	1524, 401
MP7645.0	Assignment of a third handwheel via axis selector switch S2, when MP7645.2 = 0 Input: 0: Switch position 1 (at the left stop) 3rd handwheel axis Z Switch position 2 3rd handwheel axis IV Switch position 3 3rd handwheel axis V 1: Switch position 1 3rd handwheel axis X Switch position 2 3rd handwheel axis Y Switch position 3 3rd handwheel axis Z Switch position 4 3rd handwheel axis IV Switch position 5 3rd handwheel axis V 2: Switch position 3 3rd handwheel axis Z Switch position 4 3rd handwheel axis IV Switch position 5 3rd handwheel axis V		
MP7645.1	Fixed assignment of third handwheel if MP7645.2 = 1 Input: 1: X axis 2: Y axis 4: Z axis 8: Axis IV (MP410.3) 16: Axis V (MP410.4)		
MP7645.2	Assignment of a third handwheel via axis selector switch or MP7645.1 Input: 0: Assignment by axis selection switch according to MP7645.0 1: Assignment by MP7645.1		
MP7645.3	HRA 55xFS: Behavior of the permissive buttons Input: 0: Permissive buttons not pressed -> Relay contacts A and B are open, but not cross-circuit proof 1: Handwheel permissive buttons are cross-circuit proof		
MP7645.4-7	No function		

MP	Function and input	Software version and behavior	Page
MP7646.x	Reserved	60642x-03	
MP7647.x	Reserved	60642x-03	
MP7648.x	Reserved	60642x-03	



MP	Function and input	Software version and behavior	Page
MP7649.x	Initialization parameter for handwheel at X29 of the MC main computer	60642x-03	1512,
MP7649.0	Assignment of the keys on handwheel HR 410 Input: 0: Evaluation of the keys by NC, including LEDs 1: Evaluation of the keys by PLC	PLC RUN	1524, 401
MP7649.0	Assignment of a third handwheel via axis selector switch S2, when MP7649.2 = 0 Input: 0: Switch position 1 (at the left stop) 3rd handwheel axis Z Switch position 2 3rd handwheel axis IV Switch position 3 3rd handwheel axis V 1: Switch position 1 3rd handwheel axis X Switch position 2 3rd handwheel axis Y Switch position 3 3rd handwheel axis Z Switch position 4 3rd handwheel axis IV Switch position 5 3rd handwheel axis V 2: Switch position 3 3rd handwheel axis Z Switch position 4 3rd handwheel axis IV Switch position 5 3rd handwheel axis V		
MP7649.1	Fixed assignment of third handwheel if MP7649.2 = 1 Input: 1: X axis 2: Y axis 4: Z axis 8: Axis IV (MP410.3) 16: Axis V (MP410.4)		
MP7649.2	Assignment of a third handwheel via axis selector switch or MP7649.1 Input: 0: Assignment by axis selection switch according to MP7645.0 1: Assignment by MP7645.1		
MP7649.3	HRA 55xFS: Behavior of the permissive buttons Input: 0: Permissive buttons not pressed -> Relay contacts A and B are open, but not cross-circuit proof 1: Handwheel permissive buttons are cross-circuit proof		
MP7649.4-7	No function		



MP	Function and input	Software version and behavior	Page
MP7650.x	Handwheel counting direction (for HRA 110: for each axis) MP7650.0 for X23 of the last machine operating panel in the HSCI chain, and MP7650.4 for X29 of the MC main computer Input: Bit 0 0: Negative counting direction 1: Positive counting direction Axis-specifically only for HRA 110: Bits 0 to 17 represent axes 1 to 18 0: Negative counting direction 1: Positive counting direction	PLC RUN	1507
MP7660	Sensitivity for electronic handwheel Input: 0 to 65 535 [increments]	PLC RUN	1507
MP7670 MP7670.0 MP7670.1 MP7670.2	Subdivision factor for handwheel Input: 0 to 10 Subdivision factor for slow speed Subdivision factor for medium speed (only HR 410) Subdivision factor for fast speed (only HR 410)	PLC RUN	1507, 1512
MP7671 MP7671.0 MP7671.1 MP7671.2	Handwheel feed rate in the Handwheel operating mode with HR 410 Input: 0 to 1000 [% of MP1020] Slow speed Medium speed (only HR 410) Fast speed (only HR 410)	PLC RUN	1512
MP7672 MP7672.0 MP7672.1 MP7672.2	HR 410, distance per handwheel step Input: 0.0000 to 1.0000 [mm] Slow speed Medium speed Fast speed		1512
MP7674.x	Handwheel, axis-specific subdivision factor Input: 1 to 10 0: No limitation	PLC RUN	1507
MP7675.x	Handwheel, axis-specific maximum path Input: 0.0001 to 10.0000 [mm] 0: No limitation	PLC RUN	1507

MP	Function and input	Software version and behavior	Page
MP7680	<p>Machine parameter with multiple function</p> <p>Format: %xxxxxxxxxxxxxxxx</p> <p>Input: Bit 0 – Memory function for axis-direction keys with M4562</p> <p>0: Not saved 1: Saved if M4562 is set</p> <p>Bit 1 – Returning to the contour</p> <p>0: Not active 1: Active</p> <p>Bit 2 – Block scan</p> <p>0: Not active 1: Active</p> <p>Bit 3 – Interruption of block scan for STOP or M06</p> <p>0: Interruption 1: No interruption</p> <p>Bit 4 – Inclusion of programmed dwell time during the block scan</p> <p>0: Include the dwell time 1: Do not include the dwell time</p> <p>Bit 5 – Start of calculation for block scan</p> <p>0: Start from block with cursor 1: Start from beginning of program</p> <p>Bit 6 – Tool length in blocks with normal vectors</p> <p>0: Without R2 from tool table (south pole) 1: With R2 from tool table (center of sphere)</p> <p>Bit 7 – Inserting a defined rounding arc or spline</p> <p>0: Defined rounding arcs are always inserted 1: Defined rounding arcs are always inserted if the acceleration from MP1060.x or MP1070 was exceeded</p>	<p>PLC RUN</p>	<p>892, 893, 1217, 1243, 1246, 1404, 1595, 1160</p>



MP	Function and input	Software version and behavior	Page
MP7680	<p>Machine parameter with multiple function</p> <p>Bit 8 – Insertion of rounding arc or cubic spline 0: Rounding arc is inserted 1: A cubic spline is inserted instead of a rounding arc</p> <p>Bit 9 – Constant jerk on spline (bit 8 = 1) 0: No constant jerk 1: Constant jerk</p> <p>Bit 10 – Cutter-radius-compensated outside corners 0: Insertion of a circular arc 1: Insertion of a spline curve</p> <p>Bit 11 – Behavior of M116 0: Rotary axis is parallel to linear axis 1: Any position of rotary axis to linear axis</p> <p>Bit 12 – Behavior of Cycle 28 0: Standard behavior 1: The slot wall is approached and departed tangentially; at the beginning and end of the slot a rounding arc with a diameter equal to the slot width is cut</p> <p>Bit 13 – Behavior during program interruption with axis movement 0: Automatic activation of APPROACH POSITION 1: Do not automatically activate APPROACH POSITION</p> <p>Bit 14 – Behavior of NC start after NC stop and internal stop 0: NC start permitted 1: NC start only permitted after block scan (GOTO)</p> <p>Bit 15 – NC Start if program is aborted 0: NC start permitted 1: NC Start not permitted (message window)</p> <p>Bit 16 – Behavior of Cycle 39 0: Approach/departure movement on an arc is active 1: Approach/departure movement on an arc is not active</p> <p>Bit 17 – Behavior of Markers M4175 and M4176 0: Clear M4175 and M4176 upon internal stop 1: M4176 remains set during RUNCANCEL macro</p> <p>Bit 18 – Manual measurement of basic rotation with inclined rotary axes 0: Error message "Axis angle not equal to tilt angle" is active 1: Error message not active</p>	PLC RUN	892, 893, 1217, 1243, 1246, 1404, 1595, 1160

MP	Function and input	Software version and behavior	Page
MP7681	<p>M/S/T/Q transfer to the PLC during block scan</p> <p>Format: %xxxx</p> <p>Input: Bit 0 –</p> <p>0: Transfer M functions to the PLC during block scan</p> <p>1: Collect M functions and transfer them to the PLC after block scan</p> <p>Bit 1 –</p> <p>0: Transfer T code to the PLC during block scan</p> <p>1: Transfer last T code to the PLC after block scan</p> <p>Bit 2 –</p> <p>0: Transfer S or G code to the PLC during block scan</p> <p>1: Transfer S or G code to the PLC after block scan.</p> <p>Bit 3 –</p> <p>0: Transfer FN19 outputs to the PLC during block scan</p> <p>1: Transfer last FN19 outputs to the PLC after block scan.</p> <p>Bit 4 – MP subfiles during block scan</p> <p>0: MP subfiles are not activated during block scan</p> <p>1: MP subfiles are activated during block scan</p>	<p>PLC</p> <p>RUN</p>	1248



MP	Function and input	Software version and behavior	Page
MP7682	<p>Machine parameter with multiple function</p> <p>Format: %xxxxxxxxxxxxxxxx</p> <p>Input: Bit 0 – Incremental block after TOOL CALL</p> <p>0: With length compensation 1: Without length compensation</p> <p>Bit 1 – Reference value for calculating the preset during datum setting</p> <p>0: Actual value is calculated 1: Nominal value is calculated</p> <p>Bit 2 – Traverse path of rotary axes with modulo display</p> <p>0: Positioning without passing over zero 1: Positioning on the shortest path</p> <p>Bit 3 – Reserved, enter 0</p> <p>Bit 4 – Tolerance for compensating movements with tilting axes (M114)</p> <p>0: Tolerance will be included 1: Tolerance will not be included</p> <p>Bit 5 – Feed rate with M128 or TCPM</p> <p>0: Feed rate refers to tool tip 1: Feed rate from interpolation of all axes involved</p> <p>Bit 6 – Behavior with TOOL DEF strobe</p> <p>0: Depending on the NC program, the TOOL DEF strobe must be acknowledged by the PLC (TOOL DEF within a continuous contour)</p> <p>1: TOOL DEF strobe must always be acknowledged by the PLC</p> <p>Bit 7 – Block elements TOOL CALL and S in ISO blocks</p> <p>0: Machine as programmed 1: Machine at beginning of block (block display does not change)</p> <p>Bit 8 – Behavior of M8 at the end of Cycles 202 and 204</p> <p>0: At the end of Cycles 202 and 204, the status of M8 is restored to that before the cycle call (behavior until now).</p> <p>1: At the end of Cycles 202 and 204, the status of M8 is not restored automatically.</p>	PLC RUN	1148, 1149, 1216, 769, 1604

MP	Function and input	Software version and behavior	Page
MP7682	<p>Bit 9 – Loading of "Tilted working plane" status 0: The "tilted working plane" status is not applied to the Manual Operation mode after a program interruption (behavior until now). 1: The "Tilted working plane" status is loaded into the Manual Operation mode after a program interruption</p> <p>Bit 10 – Peripheral milling active/inactive 0: Peripheral milling allowed 1: Peripheral milling inactive</p> <p>Bit 11 – Reserved</p> <p>Bit 12 – Error message "Tool radius too large" is suppressed if $R2 > R$ 0: Error message is displayed 1: Error message is suppressed</p> <p>Bit 13 – No program interruption upon invalid TOOL DEF 0: Error message and NC stop upon invalid TOOL DEF 1: Only warning upon invalid TOOL DEF</p> <p>Bit 14 – No insertion of the tool if TIME2 has expired 0: Tool is inserted even if TIME2 has expired 1: TOOL CALL for a tool where TIME2 has expired leads to an error message and NC stop</p> <p>Bit 15 – Reserved = 0 Bit 16 – Reserved = 0 Bit 17 – Reserved = 0</p>		1264, 1595



MP	Function and input	Software version and behavior	Page
MP7683	<p>Executing pallet tables and NC programs</p> <p>Format: %xxxxxxxx</p> <p>Input: Bit 0 – No function</p> <p>Bit 1 – Program Run, Full Sequence mode</p> <p>0: NC start executes a complete NC program.</p> <p>1: NC start executes all NC programs up to the next pallett.</p> <p>Bit 2 – Program Run, Full Sequence mode</p> <p>0: As defined in bit 1</p> <p>1: All NC programs and pallets up to the end of the table are executed.</p> <p>Bit 3 – When the end of the table is reached, the process begins again with the first line.</p> <p>0: Function is not in effect</p> <p>1: Function is effective (bit 2 = 1)</p> <p>Bit 4 – Editing the active pallet table</p> <p>0: Active pallet table cannot be edited.</p> <p>1: In the Program Run, Full Sequence and Program Run, Single Block modes, the current pallet table can be edited.</p> <p>Bit 5 – AUTOSTART soft key</p> <p>0: Do not display soft key</p> <p>1: Display soft key</p> <p>Bit 6 – Display of pallet table and NC program</p> <p>0: Both simultaneously in a split screen</p> <p>1: Pallet table or NC program individually</p> <p>Bit 7 – AUTOSTART function</p> <p>0: AUTOSTART function by NC</p> <p>1: AUTOSTART function by PLC</p> <p>Bit 8 – Procedure for tool-oriented machining in the Program Run operating modes</p> <p>0: NC start machines all workpieces on the pallet until the next tool change</p> <p>1: NC start executes all NC programs until the end of the pallet</p>	PLC RUN	1215, 1496
MP7683	<p>Executing pallet tables and NC programs</p> <p>Bit 9 – EDIT PALLET soft key</p> <p>0: EDIT PALLET soft key is not displayed</p> <p>1: EDIT PALLET soft key is displayed</p>	PLC RUN	1215, 1496

MP	Function and input	Software version and behavior	Page
MP7684	<p>Nominal position value filter (bit 0 to bit 4) and path control with M128 or TCPM (bit 5 to bit 7)</p> <p>Format: %xxxxxxx</p> <p>Input: Bit 0 – Nominal position value filter</p> <p>0: Include acceleration</p> <p>1: Do not include the acceleration</p> <p>Bit 1 – Nominal position value filter</p> <p>0: Include the jerk</p> <p>1: Do not include the jerk</p> <p>Bit 2 – Nominal position value filter</p> <p>0: Include the tolerance</p> <p>1: Do not include the tolerance</p> <p>Bit 3 – Nominal position value filter</p> <p>0: Include the radial acceleration</p> <p>1: Do not include the radial acceleration</p> <p>Bit 4 – Nominal position value filter</p> <p>0: Include jerk and tolerance limit at changes in the curvature</p> <p>1: Do not include jerk and tolerance limit at changes in the curvature</p> <p>Bit 5 – Reserved</p> <p>Bit 6 – Feed rate with M128 or FUNCTION TCPM</p> <p>0: Feed rate profile improved. The feed rate at the TCP is kept constant while maintaining the parameters for acceleration, jerk and tolerance</p> <p>1: Feed rate profile like up to and including 60642x-01.</p> <p>Bit 7 – Detection of curvature changes</p> <p>0: Detection of curvature changes improved.</p> <p>1: Detection of curvature changes like up to and including 60642x-01.</p> <p>Bit 8 – Reserved</p>	<p>PLC</p> <p>RUN</p>	829



MP	Function and input	Software version and behavior	Page
MP7684	Bit 9 – Accelerated 5-axis machining with M128 with many rotary axis motions that are less than 2° per positioning block (not with handwheel superimpositioning with M118) 0: Inactive 1: Active Bit 10 - Modification of the calculation of the contouring feed rate at the beginning of a contour element 0: Active 1: Inactive		829
MP7690	Evaluation of the electronic ID labels Input: %xxx Bit 0 – HEIDENHAIN power modules 0: Active 1: Inactive Bit 1 – HEIDENHAIN synchronous motors 0: Active 1: Inactive Bit 2 – HEIDENHAIN power supply units 0: Active 1: Inactive		1319



4.4.16 Second spindle

MP	Function and input	Software version and behavior	Page
MP13010 to MP13530	Machine parameter block for the second spindle Input: Function and input range are identical with MP3010 to MP3530.		985



5 Modules, markers and words

5.1 Overview of modules

Module	Function	SW version	Page
9000/ 9001	Copy in the marker or word range		1849
9002	Read all inputs of a PLC input/output unit	Not with HSCI	
9003	Read the analog input of the MC and of the PL 4xxB	Not with HSCI	
9004	Edges of PLC inputs		1528
9005	Update all outputs of a PLC input/output unit	Not with HSCI	
9006	Set and start PLC timer		1750
9007	Diagnostic information of the PL 4xxB	Not with HSCI	
9008	Read certain inputs of a PL 4xxB	Not with HSCI	
9009	Update certain outputs of a PL 4xxB	Not with HSCI	
9010/ 9011/ 9012	Read in the word range		1850
9019	Size of the processing stack		1756
9020/ 9021/ 9022	Write in the word range		1851
9031	Overwrite machine parameters		484
9032	Read machine parameters		485, 1739
9033	Select machine parameter file		492
9034	Load a machine parameter subfile		493
9035	Read status information		622, 623, 658, 1162, 1531,
9036	Write status information		1508, 1530
9038	Read general axis information		621
9040	Read the axis coordinates (format = 0.001 mm)		1150
9041	Read the axis coordinates (format = 0.0001 mm)		1151

Module	Function	SW version	Page
9042	Read the spindle coordinates (format = 0.001°)		949
9044	Read the spindle coordinates (format = 0.0001°)		949
9045	Read the 3-D ROT data		764
9046	Copy controller data to the PLC memory	60642x-03	1096
9047	Reset or initialize controller algorithms	60642x-03	695
9048	Interrogate the operating states of axes		929
9050	Conversion from binary numbers → ASCII		1852
9051	Conversion from binary numbers → ASCII		1853
9052	Conversion from ASCII → binary		1854
9053	Conversion from ASCII → hexadecimal		1854
9054	Conversion from ASCII → hexadecimal binary		1855
9055	Local time		1175
9060	M function status		1255
9061	Status of non-modal M functions		1255
9063	Collision monitoring, activating/deactivating		1451
9064	Status of collision monitoring		1452
9065	Status of the commissioning function		1116
9066	Status of HEIDENHAIN supply unit		922
9066	Status of software settings		146
9070	Copy a number from a string		1828
9071	Find the string length		1829
9072	Copy a byte block into a string		–
9073	Copy a string into a byte block		–
9074	Load texts from error/dialog files into a PLC string		1285
9075	Write to symbolic PLC operand	606 42x-02	1745
9076	Read symbolic PLC operand	606 42x-02	1746
9080	Clear the small PLC window		1355
9081	Interrogate the status of the small PLC window		1355
9082	Display a string in the small PLC window		1356
9083	Display a moving-bar diagram in the small PLC window		1357
9084	Display PLC error messages with additional data		–
9085	Display PLC error message		1181
9086	Clear PLC error message		1181
9087	Status of PLC error message		1182

Module	Function	SW version	Page
9088	Display the M functions		1158
9089	Control in operation		1159
9090	Select a line in the pallet table		1497
9091	Find the line number of a tool in the tool table		1596
9092	Search for an entry in the tables selected for execution (.T/.D/.TCH)		1572
9093	Read data from tables selected for program run (.T/.D/.TCH)		1575
9094	Write data into a tool and datum table		1576
9095	Select active line in configuration file		657
9096	Delete a line from the tool table		1577
9097	Select the geometry description		732
9098	Find the active geometry description		731
9100	Assign data interface		1918
9101	Release data interface		1919
9102	Status of data interface		1919
9103	Transmit string through data interface		1920
9104	Receive string through data interface		1921
9105	Transmit binary data through data interface		1922
9106	Receive binary data through data interface		1923
9107	Read from receiving buffer		1924
9110	Transmit a message via LSV2		1925
9111	Receive a message via LSV2		1926
9112	Transmit ASCII characters via data interface		1927
9113	Receive ASCII characters via data interface		1928
9120	Start PLC axis		631
9121	Stop PLC axis		631
9122	Status of PLC axis		632
9123	Traverse the reference marks of PLC axes		633
9124	Feed rate override for PLC axis		634
9125	Stop PLC axis at next Hirth grid position		634
9128	Torque limiting by the PLC		878
9129	Status of torque limiting by the PLC		879
9130	Output of an analog voltage		1561
9132	Selection of override potentiometer		206
9133	Hardware information of the MC 42x(B)		904 907
9135	Switch on the 3-D touch probe		1538

Module	Function	SW version	Page
9136	Switch the touch probe on/off		1538
9137	Diagnostic information of the PL 510	Not with HSCI	
9138	Read analog input of the PL 510	Not with HSCI	
9139	Reset short-circuit monitoring of the outputs on the PLD 16-8	Not with HSCI	
9140	Set axis-specific feed-rate limit		–
9141	Read axis-specific feed-rate limit		–
9143	Activate the brake test		928
9144	Activate the emergency stop test		940
9145	Actual-to-nominal value transfer		847
9146	Save and reestablish actual position values		989
9147	Assign a reference value to an axis		798
9148	Use nominal value as actual value		1533
9149	Set field angle/Read via PLC		1118
9150	Axis-specific reading of axis traverse limits		625
9151	Select traverse range and axis designation		626
9152	Select traverse range, axis display and axis designation		627
9153	Switch the touch probe axis		1540
9155	Axis switchover from closed loop to open loop		989
9156	Axis switchover from open loop to closed loop		990
9157	Drive controller status		864
9158	Maximum torque		876
9159	Advance status message: Drives will be switched off		864
9160	Status request for temperature monitoring and I ² t monitoring		913
9161	Enable the drive controller		865
9162	Status request of the drive controller		865
9163	Switching the operating modes		980
9164	Read the actual speed value of the motor		849
9165	Sample the current motor temperature		905
9166	Momentary utilization of the drive motor		919
9167	Supply voltage monitoring		903
9168	Interrogate the commissioning status		1026
9169	Axes for which I32 does not switch off the drives	Not with HSCI	
9170	Find the current torque		920

Module	Function	SW version	Page
9171	Oriented spindle stop		971
9173	Speed-dependent wye/delta switchover		983
9174	Spindle status regarding wye/delta switchover		984
9175	Switch the spindle		986
9179	Status information about spindle(s)		986
9180	Simulation of NC keys		1393
9181	Disable individual NC keys		1393
9182	Re-enable individual NC keys		1394
9183	Disable groups of NC keys		1394
9184	Re-enable groups of NC keys		1395
9185	Disable touchpad/Interrogate status		1395
9186	Call a soft-key function		1395
9187	Status of a soft-key function call		1396
9188	Call a soft-key function		1396
9189	Shut down the control		1262
9190	Start the operating times		1171
9191	Stop the operating times		1171
9192	Read the operating times		1172
9193	Set the operating times		1172
9194	Alarm when operating time exceeded		1173
9195	System time		1174
9196	Find the PLC cycle time		1666
9197	Start cyclic timers		1750
9200	Display/delete PLC soft-key row		1390
9201	Display/delete PLC soft key		1391
9202	Select/deselect PLC soft keys and PLC windows		1391
9203	Activate PLC soft-key menu		1380
9204	Update the PLC soft keys		1381
9205	Set the word for acknowledgment of PLC soft keys		1382
9206	Change the settings of the PLC soft keys		1383
9207	Replace PLC soft keys		1384
9208	Status information of the PLC soft keys		1385
9210	Open or clear screen mask for the PLC window		1369
9211	Status of the large PLC window		1371
9215	Activate a PLC pop-up window		1193
9216	Pop-up window with tool selection list		1583

Module	Function	SW version	Page
9217	Pop-up window for messages		1194
9220	Renewed traversing of the reference marks		804
9221	Start a PLC positioning movement		641
9222	Status request of PLC positioning movement		642
9223	Free rotation		1153
9224	Stop PLC positioning movements		643
9225	Compensation value for the reference mark		797
9227	Position auxiliary axes and NC axes		644
9228	Coupling function for auxiliary axes		637
9229	Status of coupling function		639
9230	Datum shift		1534
9231	Compensation of thermal expansion		660
9232	Movement in the direction of the virtual tool axis	606 42x-02	773
9240	Open a file		1419
9241	Close a file		1420
9242	Position in a file		1421
9243	Read from a file line by line		1422
9244	Write to a file line by line		1423
9245	Read a field from a table		1411
9246	Write to a field in a table		1413
9247	Search for a condition in a table		1414
9248	Rename, copy or delete files		1408
9250	Start PLC editor for tables		1417
9251	End the PLC editor for tables		1418
9252	Position the cursor in the PLC editor		1418
9255	Read a field from a table as an integer value		1412
9256	Write to a field in a table		1414
9260	Receive events and wait for events		1837
9261	Send events		1839
9262	Context change between spawn processes		1840
9263	Interrupt a spawn process for a defined time		1840
9270	Read a code word		1708
9271	Write a code word		1708
9272	Write code number into PLC string	606 42x-02	1266
9275	Write ASCII data into the log		1296



Module	Function	SW version	Page
9276	Write operand contents into the log		1297
9277	Write data into the OEM log		1297
9279	Control reset		1262
9280	Start the NC macro (Run pallet entry)		1501
9281	Select a line in the pallet table		1498
9282	Tool usage test for pallet table		1598
9283	Generate tool-usage file	60642x-03	1600
9285	Set the access level		1168
9286	Select the VNC client and control of the focus assignment	60642x-03	1882
9287	General status requests regarding VNC clients	60642x-03	1883
9288	Status requests regarding a VNC client	60642x-03	1885
9290	Select a file		1407
9291	Call an NC macro		1711
9300	Lock/release the pocket table		1589
9301	Find the number of an entry in the pocket table		1591
9302	Search for a vacant pocket in the tool magazine		1591
9304	Copy columns P1 to P5 to the pocket table		1584
9305	Tool exchange in the pocket table		1590
9306	Exchange tools between tool magazines		1592
9310	Read the machine parameter from the run-time memory		486
9311	Dynamically change values for friction compensation		1107
9312	Change machine parameters in the current machine parameter file		487
9313	Read machine parameters from current machine parameter file		489
9314	Activate/Deactivate machine parameters		489
9320	Status of the NC program end		1213
9321	Find the current block number		1210
9322	Information of the current NC program		1211
9340	Search for a pocket depending on magazine rules		1585
9341	Edit a pocket table depending on magazine rules		1586
9342	Find magazine and pocket number		1587
9343	Compile and activate magazine rules		1588
9350	Read data from the tool table		1565
9351	Write data to tool table		-

Module	Function	SW version	Page
9390	Open the online help window with the control's browser		1203
9391	Display an error message with additional offset		1204
9392	Display PLC error message with help offset		1182
9392	PET column for PLC notification	606 42x-02	1184



5.2 Overview of markers and words

A list of PLC operands with brief description in English and German (GLB_NC_en.DEF, GLB_NC_de.DEF) is on the control under PLC:\JHV.

Operand		Description	Set	Reset	SW version	Page
M	1900 - 1999	Decoded M function if M4571 is set	NC	NC		1254
M	4000	Spindle in position	NC	NC		970
M	4001	Nominal speed command signal of the spindle not in the ramp	NC	NC		954
M	4002	Nominal speed value = 0	NC	NC		954
M	4003	Nominal speed value output analog or digital (MP3010 = 3 to 8)	NC	NC		952
M	4004	Impermissible speed was programmed	NC	NC		958
M	4005	Status display and nominal speed value output for M03	PLC	PLC		957, 1157
M	4006	Status display and nominal speed value output for M04	PLC	PLC		957, 1157
M	4007	Status display M05 and spindle stop	PLC	PLC		957, 1157
M	4008	Disable speed output for spindle	PLC	PLC		958, 1157
M	4009	Counterclockwise spindle rotation (for gear change)	PLC	PLC		960
M	4010	Clockwise spindle rotation (for gear change)	PLC	PLC		960
M	4011	Activate rotational speed MP3520.0 and direction of rotation from M4013	PLC	PLC		973
M	4012	Open the spindle control loop	PLC	PLC		970, 1544
M	4013	Direction for spindle orientation from a standstill (M03 = 0; M04 = 1)	PLC	PLC		973
M	4014	Reverse the direction of spindle rotation	PLC	PLC		957
M	4015	Renewed evaluation of the spindle reference mark	PLC	NC		970
M	4016	Cycle 13 is executed	NC	PLC		973
M	4017	Servo-controlled spindle in motion	NC	NC		970
M	4018	Reference mark for spindle not yet traversed	NC	NC		970
M	4019	Reverse the counting direction of the position encoder on the spindle	PLC	PLC		957
M	4030	Cycle 2 or Cycle 17 active	NC	NC		975, 979
M	4031	Cycle 17 or Cycle 18 active	NC	NC		979

Operand		Description	Set	Reset	SW version	Page
M	4040	Status display M07, M08, and M09 highlighted	PLC	PLC		1157
M	4041	Status display M07, M08, M09, MK	PLC	PLC		1157
M	4042	Status display M07, M08, M09, MK	PLC	PLC		1157
M	4050	Touch probe not ready (ready signal is missing)	NC	NC		1537
M	4051	Stylus deflected before start of probing cycle	NC	NC		1537
M	4052	Stylus is deflected, probing process is completed	NC	PLC		1537
M	4053	Probing process has been completed or canceled	NC	NC		1537
M	4054	Battery voltage too low (battery warning at touch probe connection); evaluated only during the probing process	NC	NC		1537
M	4055	Enable the probing process	NC	PLC		1537
M	4056	NC stop in all operating modes if stylus is deflected	PLC	PLC		1537
M	4057	Touch probe cycle active (FN17: ID990 NR2)	NC	NC		–
M	4060	Cycle for tool measurement started	NC	NC		1560
M	4061	0: Measure the tool 1: Check the tool	NC	NC		1560
M	4062	0: Wear tolerance not exceeded 1: Wear tolerance exceeded	NC	NC/ PLC		1560
M	4063	0: Breakage tolerance not exceeded 1: Breakage tolerance exceeded	NC	NC/ PLC		1560
M	4065	Workpiece dimensions are OK	NC	PLC		1545
M	4066	Workpiece must be reworked	NC	PLC		1545
M	4067	Workpiece is scrap	NC	PLC		1545
M	4070	Strobe signal for gear code	NC	NC		960
M	4071	Strobe signal for S code	NC	NC		965
M	4072	Strobe signal for M function	NC	NC		1254
M	4073	Strobe signal T code (P code) with TOOL CALL	NC	NC		1611, 1629
M	4074	Strobe signal T code (P code) with TOOL DEF	NC	NC		1611, 1629
M	4075	Transfer active with FN19	NC	NC		1713
M	4090	Acknowledgment of "gear change completed"	PLC	PLC		960
M	4091	Acknowledgment of S code	PLC	PLC		965
M	4092	Acknowledgment of M function	PLC	PLC		1254

Operand		Description	Set	Reset	SW version	Page
M	4093	Acknowledgment of T code (P code) with TOOL CALL	PLC	PLC		1611, 1629
M	4094	Acknowledgment of T code (P code) with TOOL DEF	PLC	PLC		1611, 1629
M	4095	Acknowledgment of transfer with FN19	PLC	PLC		1713
M	4120 - 4128	PLC positioning axis 1 to 9 active	NC/PLC	NC/PLC		646
M	4130	Activation of spindle orientation, or spindle orientation has been started with Module 9171	NC/PLC	NC		973
M	4131	Activation of Q-parameter transfer to the NC; data from D258, Q number from W516	PLC	NC		1714
M	4132	Activate datum shift from D528 to D544, or call Module 9230	PLC	NC		1534
M	4133	Start and stop the free rotation function	PLC	NC		1154
M	4134	Activation of a gear range and speed through the PLC	PLC	NC		960
M	4135	Strobe marker for selecting the traverse range	PLC	NC		624
M	4150	Operating mode: Manual Operation	NC	NC		1165
M	4151	Operating mode: Electronic Handwheel	NC	NC		1165
M	4152	Operating mode marker ■ Positioning with Manual Data Input ■ RUN ACTIVE UNIT	NC	NC		1165
M	4153	Operating mode marker ■ Program Run, Single Block ■ RUN SINGLE UNITS	NC	NC		1165
M	4154	Operating mode marker ■ Program Run, Full Sequence ■ RUN ALL UNITS	NC	NC		1165
M	4155	Operating mode: Traversing the reference marks	NC	NC		1165
M	4156	MANUAL TRAVERSE soft key pressed	NC	NC		1246
M	4157	Returning to the contour (MOVE TO POSITION) is active	NC	NC		1246
M	4158	Block scan active	NC	NC		1246
M	4159	PLC editor: END key or soft key pressed	NC	NC/PLC		1417
M	4160	Pallet table selected	NC	NC		1496
M	4161	M/S/T/Q transfer after block scan	NC	NC		1248
M	4162	DNC mode (0 = DNC inactive, 1 = DNC active)	NC	NC		
M	4163	Alternative operating mode smarT.NC is active	NC	NC		1165

Operand		Description	Set	Reset	SW version	Page
M	4170	END PGM, M02 or M30 was executed	NC	NC		1244
M	4172	First PLC scan after power on	NC	NC		–
M	4173	First PLC scan after interruption of the PLC program	NC	NC		–
M	4174	First PLC scan after editing the MPs (MP Edit was exited and the MPs were altered)	NC	NC		–
M	4175	Program interruption, control-in-operation symbol is blinking	NC	NC		1160
M	4176	Control is in operation, control-in-operation symbol is on or is blinking	NC	NC		1160
M	4177	Clearable error message displayed	NC	NC		932
M	4178	Error message EXTERNAL emergency stop is displayed	NC	NC		932
M	4179	Control is being shut down	NC	NC		1261
M	4180	Rapid traverse programmed (FMAX)	NC	NC		1156
M	4181	NC program selected	NC	PLC		1212
M	4182	AUTOSTART active	NC	NC		1215
M	4183	Time from AUTOSTART expired	NC	NC		1215
M	4185	Internal stop performed	NC	PLC		1213
M	4186	NC program is active in the Test Run mode	NC	NC		1165
M	4188	Compilation process of the PLC project active	NC	NC		1683
M	4189	Emergency-stop test or self-test of the control has been concluded	NC	NC		939
M	4190	Control is ready for emergency-stop test or self-test, or test is active	NC	NC		939
M	4191	Control is ready	NC	NC		939
M	4192	Request for machine control voltage ON	NC	NC		939
M	4194	I/O force list active	NC	NC		–
M	4195	Press the SHUTDOWN soft key	NC	NC	60642x-03	1526
M	4197	Cancel the shutdown of the control	PLC	NC	60642x-03	1526
M	4196	Pallet machining stopped for lack of a valid job	NC	PLC		1496
M	4200	Overflow during multiplication	NC	PLC		1789, 1802, 1831
M	4201	Division by 0	NC	PLC		1790, 1802, 1831
M	4202	Incorrectly executed modulo	NC	PLC		1791, 1802, 1831
M	4203	Error status for PLC module	NC	NC/ PLC		1802, 1831



Operand		Description	Set	Reset	SW version	Page
M	4204	Reserved for errors that the PLC programmer would like to catch	NC	NC		1831
M	4210	Error from Python script with F stop active	NC	NC		
M	4211	Error from Python script with NC stop active	NC	NC		
M	4212	Error from Python script with emergency stop active	NC	NC		
M	4213	Error from Python script with NC Cancel active	NC	NC		
M	4223	Error from PET table with NC Cancel active	NC	NC		1179, 1191
M	4220	Error from PET table with F stop active	NC	NC		1179, 1191
M	4221	Error from PET table with NC stop active	NC	NC		1179, 1191
M	4222	Error from PET table with emergency stop active	NC	NC		1179, 1191
M	4223	Error from PET table with NC Cancel active	NC	NC		1179, 1191
M	4225	Alternative error reaction active	PLC	PLC		1179
M	4227	PLC error message with priority 0 (error)	NC	NC		1179
M	4228	PLC error message with priority 1 (warning)	NC	NC		1179
M	4229	PLC error message with priority 2 (info)	NC	NC		1179
M	4230	NC start via LSV2	NC	NC		1404
M	4231	NC stop via LSV2	NC	NC		1404
M	4260	Emergency stop is active (ES.A, ES.B, ES.A.HW and ES.B.HW signals become active)	NC	NC		932
M	4300 - 4315	Value from MP4310.0	NC	NC		1738
M	4316 - 4331	Value from MP4310.1	NC	NC		1738
M	4332 - 4347	Value from MP4310.2	NC	NC		1738
M	4348 - 4363	Value from MP4310.3	NC	NC		1738
M	4364 - 4379	Value from MP4310.4	NC	NC		1738
M	4380 - 4395	Value from MP4310.5	NC	NC		1738
M	4396 - M4411	Value from MP4310.6	NC	NC		1738
M	4520	Additional T code (P code) follows with TOOL CALL	NC	NC		1612, 1629
M	4521	Tool number zero programmed	NC	NC		1611

Operand		Description	Set	Reset	SW version	Page
M	4522	Tool with pocket number programmed is in effect with MP7480.0 = 3 or 4 and TOOL CALL	NC	NC		1613
M	4523	Tool without pocket number programmed is in effect with MP7480.0 = 3 or 4 and TOOL CALL	NC	NC		1613
M	4524	Special tool called, TOOL CALL	NC	NC		1613, 1629
M	4525	TOOL CALL after expiration of tool life	NC	NC		1613
M	4526 - 4534	Axis 1 to axis 9 is the tool axis	NC	NC		622
M	4538	Geometry of the tool from W264	PLC	NC		1246, 1611
M	4539	Tool number highlighted in the status display	PLC	PLC		–
M	4540	Sequence of tool number or pocket number transfer (M4520 = 1)	PLC	PLC		1613, 1629
M	4541	Special tool in original pocket in spite of variable pocket coding	PLC	PLC		1593, 1613, 1629
M	4542	Do not update pocket number in the pocket table	PLC	PLC		1246, 1613
M	4543	Tool life 1 expired (TIME1 in the tool table)	NC	NC/ PLC		1595
M	4546	Tool life 2 expired (TIME2 in the tool table)	NC	NC/ PLC		1595
M	4547	T and G strobes with TOOL CALL	NC	NC		960, 1611
M	4560	NC stop (0: stop)	PLC	PLC		1404
M	4561	Rapid traverse	PLC	PLC		1404
M	4562	Memory function for axis direction keys (MP7680 bit 0 = 1)	PLC	PLC		1404
M	4563	Feed-rate enable for all axes	PLC	PLC		846
M	4564	NC start	PLC	PLC		1404
M	4570	Unit of measure for transfer with FN19	NC	NC		1713
M	4571	Activation of decoded M-code transfer in M1900 to M1999	PLC	PLC		1254
M	4574	Select the traverse range (with M4575)	PLC	PLC		624, 1541
M	4575	Select the traverse range (with M4574)	PLC	PLC		624, 1541
M	4576	Disable the handwheel	PLC	PLC		1507
M	4577	Disabled key was pressed	NC	PLC		1392
M	4579	INCREMENT OFF/ON soft key	NC	NC		1530
M	4580	Suppress emergency stop, open all position control loops, NC stop	PLC	PLC		844, 932

Operand	Description	Set	Reset	SW version	Page	
M	4581	Open all position control loops, NC stop, activate "Approach position"	PLC	PLC		844
M	4586	Enable AUTOSTART	PLC	NC/ PLC		1215
M	4587	Rescind feed rate limit above F MAX	PLC	PLC		842
M	4589	Activate datum management via preset table	NC	NC		1209
M	4590	Status of fast PLC input from MP4130.2	NC	PLC		1753
M	4591	Status of fast PLC input from MP4130.3	NC	PLC		1753
M	4592	Status of fast PLC input from MP4130.4	NC	PLC		1753
M	4593	Status of fast PLC input from MP4130.5	NC	PLC		1753
M	4600	Faulty internal communication between HEROS and Windows 2000	NC	NC		–
M	4620	Enable LIFTOFF function	PLC	PLC		–
M	4622	Delay NC macro with RESETINIT = from NCMACRO.SYS	PLC	PLC		803
M	4623	Disable starting of DNC mode (LSV2 access)	PLC	PLC		–
M	4624	Changed axis-traverse limits	NC	PLC		624
M	4625	Disable NC axes when velocity semifeedforward control is active	PLC	PLC		838
M	4626	Disable all key inputs of the TE keyboard unit, including the soft keys	PLC	PLC		1516
M	4627	Trigger condition for integrated oscilloscope fulfilled	NC	PLC		1002
M	4628	Recording of integrated oscilloscope ended	NC	PLC		1002
M	4641	HSCI bus free of error	NC	NC		1321
M	4660	HR 420/HR 5x0 assumes control	NC	NC		1516
M	4661	NC start on HR 420/ HR 5x0	NC	NC		1516
M	4662	NC stop on HR 420/ HR 5x0	NC	NC		1516
M	4663	Rapid traverse key on HR 420/HR 5x0	NC	NC		1516
M	4664	Spindle start on HR 420/HR 5x0	NC	NC		1516
M	4665	Spindle stop on HR 420/HR 5x0	NC	NC		1516
M	4666	Plus (+) key on HR 420/HR 5x0	NC	NC		1516
M	4667	Minus (–) key on HR 420/HR 5x0	NC	NC		1516
M	4668	CTRL key on HR 420/HR 5x0	NC	NC		1516
M	4670	Potentiometer on HR 420/HR 5x0 active	NC	NC		1516
M	4680	Disable activation of the HR 420/HR 5x0	PLC	PLC		1516

Operand		Description	Set	Reset	SW version	Page
M	4751	Write errors from PLC modules in the PLC log	PLC	PLC	60642x-03	1286
M	4752	Recording of transfer and return parameters of PLC modules that are called in the cyclic PLC program	PLC	PLC	60642x-03	1286
M	4753	Recording of transfer and return parameters of PLC modules that are called in submit/spawn processes	PLC	PLC		1286
M	4754	Write diagnostic information in MYDEBUG.LOG	PLC	PLC		1287



	Marker	Description	Set	Reset	SW version	Page
W	256	Gear code	NC/ PLC	NC/ PLC		960
W	258	S code	NC	NC		965
W	260	Code for M function	NC	NC		1254
W	262	Tool pocket number	NC	NC		1611, 1629
W	264	Tool number	NC	NC		1611, 1629
W	266	Index number of a programmed indexed tool	NC	NC		1596
W	268	Tool magazine number	NC	NC		1590
W	270	Line number in help file	NC	NC		1190
W	272	Operating mode	NC	NC		1165
W	274	Code of the depressed key	NC	NC		1392
D	276	Code of the code number last entered via MOD	NC	NC		1265
D	280	First numerical value from FN19	NC	NC		1713
D	284	Second numerical value from FN19	NC	NC		1713
W	302	Number of the horizontal PLC soft key that was pressed	NC	NC		1389
W	304	Number of the vertical PLC soft key that was pressed	NC	NC		1380
W	320	Nominal speed value [min^{-1}]	NC	NC		952
W	322	Actual speed value [min^{-1}]	NC	NC		952
W	336	Setting of the AFC soft key	NC	NC/ PLC		1235
W	342	Value from column PLC in table AFC.TAB	NC	NC/ PLC		1225
W	348	Current AFC status (0=inactive, 1=learn, 2=control)	NC	NC/ PLC		1223
W	350	Error from AFC that led to NC stop	NC	NC/ PLC		1224
D	356	Programmed speed [0.001 min^{-1}]	NC	NC		952, 960
D	360	Programmed feed rate	NC	NC		842
D	364	Nominal speed value [min^{-1}]	NC	NC		952
D	368	Actual speed value [min^{-1}]	NC	NC		952
D	372	Maximum spindle speed including spindle speed override [min^{-1}]	NC	NC		952
D	388	Current contouring feed rate [mm/min]	NC	NC		842
W	480-484	Analog input at X48 [0.1 V]	NC	NC		1529
W	486 - 490	Temperature input at X48 [0.5 °C]	NC	NC		1529
W	492	Percentage for spindle override (NC to PLC)	NC	NC		963
W	494	Percentage for feed-rate override (NC to PLC)	NC	NC		1156

	Marker	Description	Set	Reset	SW version	Page
W	516	Q no. 0-7 for numerical data transfer PLC to NC	PLC	PLC		1714
B	518	Definition of the free rotation function	PLC	PLC		1154
B	519	Traverse direction for free rotation	PLC	PLC		1154
W	522	Enabling the fast PLC inputs	PLC	PLC		1753
W	524	Open the position control loop, deactivate the monitoring functions if drive enabling via PLC input from MP4132 is missing	PLC	PLC		863
D	528	Double word with multiple function; here data for transfer from PLC to NC	PLC	PLC		1714
D	528 - 544	Target position for PLC positioning	PLC	PLC		646
D	528 - 544	Datum shift for axes 1 to 5	PLC	PLC		1534
W	560 - 568	Feed rate for PLC positioning	PLC	PLC		646, 1154
W	576 - 584	Lag-tracking axis error compensation	PLC	PLC		659
D	592	Nominal position for spindle orientation	PLC	PLC		973
D	596	Max. feed rate from PLC [mm/min]	NC/ PLC	PLC		842
D	604	Maximum possible spindle speed	PLC	NC/ PLC		952
W	632	Alternative control input variable for AFC	PLC	NC/ PLC		1222
W	754	% function for feed-rate override for free rotation	PLC	PLC		1154
D	756	Programmed rotational speed or rotational speed from the PLC [0.001 min ⁻¹]	NC/ PLC	NC/ PLC		960
D	760	Offset in tilting axes touch probe center offset [1/10 000°]	PLC	PLC		1544
W	764	Percentage for spindle speed override (PLC to NC)	NC/ PLC	NC/ PLC		963
W	766	Percentage for feed-rate override (PLC to NC)	NC/ PLC	NC/ PLC		1156
D	768 - 956	Value from MP4210.0 to MP4210.47	NC	NC		1737
W	960 - 968	Value from MP4220.0 to MP4220.4	NC	NC		1738
W	976 - 994	Value from MP4310.0 to MP4310.9	NC	NC		1738
W	1002	Last PLC run-time error that occurred	NC	NC		1754
W	1004	Error number of the last NC error message that occurred	NC	NC		-
W	1008	S code for minimum speed	NC	NC		965
W	1016	PLC module that was last processed incorrectly	NC	NC		-
W	1018	Number of files opened by the PLC	NC	NC		1408
W	1020	Number of all open files	NC	NC		1408

	Marker	Description	Set	Reset	SW version	Page
W	1022	Error status of the last called PLC module	NC	NC		–
W	1024/1100 ^a	Axis enabling	NC	NC		844
W	1026/1104 ^a	Axes in position	NC	NC		900
W	1028/1108 ^a	Axes in motion	NC	NC		901
W	1030/1112 ^a	Current direction of traverse	NC	NC		609
W	1032/1116 ^a	Reference marks not yet traversed	NC	NC		803
W	1034/1120 ^a	Positive software limit switch was approached	NC	NC		628
W	1036/1124 ^a	Negative software limit switch was approached	NC	NC		628
W	1038/1128 ^a	Prepare to open the position control loop	PLC	PLC		845
W	1040/1132 ^a	Axis-specific opening of the position control loop	PLC	PLC		845
W	1042/1136 ^a	Deactivation of monitoring functions	PLC	PLC		894
W	1044/1140 ^a	Actual-to-nominal value transfer	PLC	PLC		847
W	1046/1144 ^a	Manual traverse in positive direction	PLC	PLC		1404
W	1048/1148 ^a	Manual traverse in negative direction	PLC	PLC		1404
W	1050/1152 ^a	Incremental jog positioning in positive direction	PLC	PLC		
W	1052/1156 ^a	Incremental jog positioning in negative direction	PLC	PLC		
W	1054/1160 ^a	Reference end position	PLC	PLC		803
W	1056/1164 ^a	Lubrication pulse: Value in MP4050.x exceeded	NC	NC		629
W	1058/1168 ^a	Reset the accumulated distance	PLC	PLC		629
W	1060/1172 ^a	Axis-specific feed-rate enable	PLC	PLC		846
W	1062/1174 ^a	Lock the handwheel for specific axes	PLC	PLC		1507

a. Depending on the active PLC module interface MP4020 bit 14



6 Configuring the axes and spindle

6.1 Control loops

6.1.1 Selecting the axes

With MP10 you define which machine axes are to be operable. The bits may be changed during the run-time without a control reset. However, the bits to be changed must have been set before the control was switched on. Changing bits that had not been set leads to a control reset.

MP10	Active axes
Format:	%xxxxxxxxxxxxxxxxxxx
Input:	Bits 0 to 17 represent axes 1 to 18 0: Axis not active 1: Axis active

Screen display

You can define how the axes are shown on the screen:

- ▶ In MP100.x, assign a designation to each logical axis.
- ▶ Define in MP7291.x the screen line in which the axis is to be displayed.

Rules for the display:

- NC axes are designated with uppercase letters.
- PLC axes are designated with lowercase letters.
- Axes that are not present are given a hyphen "-".

MP100	Designation of axes
Format:	-wvucbazyxwvucbazyx
Input:	Characters 1 to 18 from the right represent axes 1 to 18
MP100.3	Designation of axes for traverse range 1
MP100.3	Designation of axes for traverse range 2
MP100.2	Designation of axes for traverse range 3

MP7291	Display of axes on the screen
Format:	SXYZABCUVWxyzabcuvw-
Input:	Characters 1 to 18 from the right represent lines 1 to 18 At far left is the spindle "S," which is always output in the lowest line.
MP7291.0	Display in traverse range 1
MP7291.1	Display in traverse range 2
MP7291.2	Display in traverse range 3

Assignment of axis keys IV, V, VI and T

On the keyboard unit and the HR 4xx and HR 5xx handwheels, you can assign the axis keys IV and V as desired.

The HR 5xx handwheels feature six instead of five axis-address keys. You can also enter the axis "T" in MP410.x for the virtual tool axis VT in the SIK option "Global Settings" if this option is available.

MP410 Assignment of axis keys IV, V and VI

Input:	Axis labels A/B/C/U/V/W/T with SW option 44
MP410.3	Axis key IV
MP410.4	Axis key V
MP410.5	VI axis key (only HR 5xx)

Demo operation for NC axes

Operation of NC axes with MP12 enables the simulation of axis motors that are not present on the machine. This permits you to put into operation and test axes that have not yet been mounted on the machine.

The following is required for this:

- The axis to be simulated must be **completely** configured in the machine parameters.
- For safety reasons, the PLC may not output any "drive and axis group enables" for the axes to be simulated while in demo operation. This might necessitate very comprehensive changes to the PLC program.

MP12 Axis-specific demo operation for NC axes

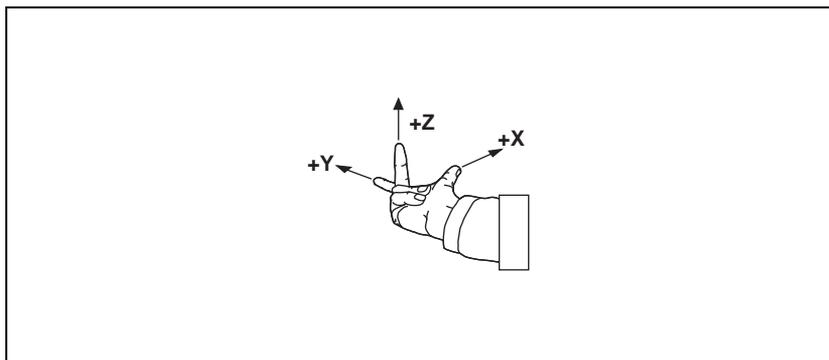
Format:	%xxxxxxxxxxxxxxxx
Input:	Bits 0 to 17 represent axes 1 to 18
	0: Demo operation not active
	1: Demo operation active

6.1.2 Axis designation

Principal axes X, Y, Z

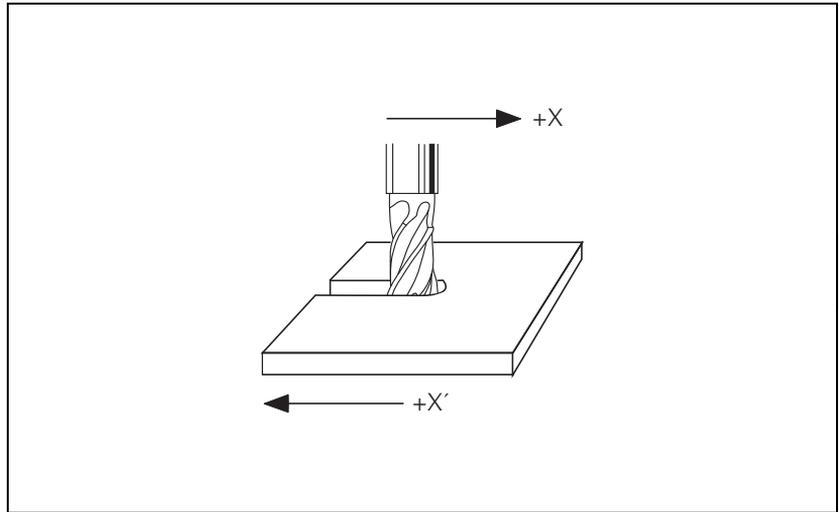
The coordinate axes and their directions of motion are defined in the international standard ISO 841.

An easy way to remember this system is to use the "right-hand rule":



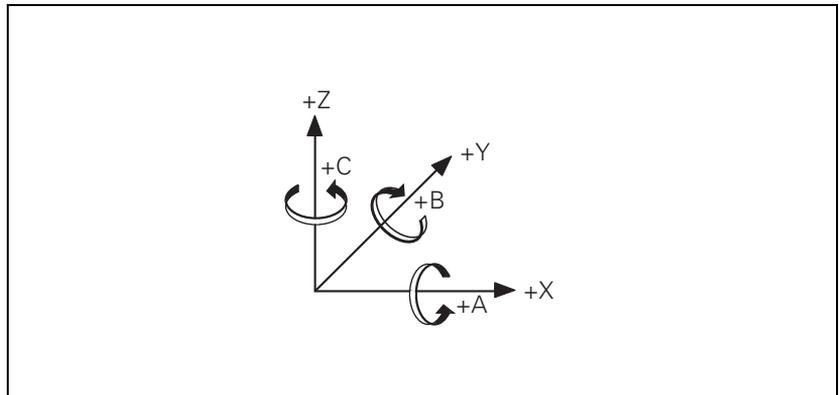
Algebraic signs of the axes

When the programmer writes an NC program, he always assumes that the tool (not the workpiece) is in motion. If the machine moves its workpiece-holding element (table) in a particular axis instead of the tool, then the direction of actual motion is opposite to the direction of axis motion. In this case the direction of motion is designated with the same algebraic sign as the axis direction, but with an apostrophe: $+X'$, $+Y'$ and $+Z'$:



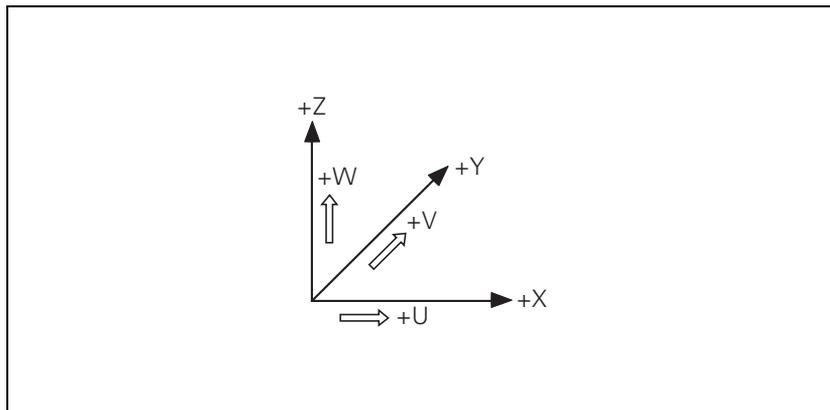
Rotary axes A, B, C

The directions of the rotary axes A, B and C follow the "right-fist rule." The fingers of the closed right hand point in the proper rotational direction of an axis when the thumb points in the direction of the associated linear axis:



Secondary linear axes

The secondary linear axes U, V and W are parallel to the principal axes X, Y and Z.



6.1.3 Encoders

Position encoders report positions and movements of the machine to the control. The iTNC 530 operates with incremental and absolute encoders with EnDat interface.

Signal period

For any given distance the position encoder supplies a fixed number of signal periods. In addition, the signal is interpolated in the control.

To calculate the signal period, the control requires the following data:

- ▶ In MP331.x, enter for each axis the length required for the number of signal periods given in MP332.x.
- ▶ In MP332.x, enter for each axis the number of signal periods for the length given in MP331.x.



Note

Large input values in MP331.x and MP332.x cannot be read by the PLC!

The iTNC calculates the quotient:

$$\text{Signal period} = \frac{\text{MP331.x}}{\text{MP332.x}}$$



Note

Digital axes:

If no position encoder (MP110.x = 0) is connected, the data of the speed encoder must be entered in MP331.x and MP332.x. This also applies to speed encoders with EnDat interface, since the incremental track of the speed encoder is used for position feedback control.

HEIDENHAIN offers incremental linear encoders with **distance-coded reference marks**. The nominal increment between two fixed reference marks depends on the encoder being used:

- ▶ In MP334.x, enter for each axis the nominal increments between two fixed reference marks.

If the number of grating periods between the reference end position and the first reference mark exceeds the value from MP334.x, the error message **Ref mark <axis>: incorrect spacing** appears. This monitoring is turned off with MP334.x = 0.

Example:

LS 486C:

Incremental linear encoder with distance-coded reference marks, grating period 20 µm (= one signal period covers 0.02 mm), nominal increment between reference marks is 20 mm.

MP331.x = 0.02

MP332.x = 1

$$\text{MP334.x} = \frac{20 \text{ mm}}{0.02 \text{ mm}} = 1000 \text{ (or 0)}$$

MP331.x Distance for the number of signal periods in MP332

Input: 0.0001 to +1.797693135E+308 [mm] or [°]

MP332.x Number of signal periods for the distance in MP331

Input: 1 to +1.797693135E+308

MP334.x Nominal increment between two fixed reference marks on encoders with distance-coded reference marks

Input: 1 to 65 535

0: 1000

External interpolation

If you connect encoders with TTL signals and an external interpolation unit through the TTL/1 V_{PP} adapter to the control:

- ▶ In MP340.x, enter the interpolation factor of the external interpolation unit.

MP340.x Interpolation factor for external interpolation

Input: 0 to 99

0 = 1: No external interpolation



Encoder signals

The CC 6xxx can be connected with incremental 1 V_{pp} and absolute position encoders with EnDat interface:

- ▶ Use MP118.x to configure the position encoder inputs of the axes
- ▶ Use MP119.x to configure the position encoder inputs of the spindles

When using an external Y cable, you need to deactivate the terminating resistor of the position encoder input via MP118.x/MP119.x bit 1.

MP118.x Position encoder input for axes

Format: %xxxx

Input: Bit 0: Position encoder input 1 V_{pp}

 0: 1 V_{PP}

 1: Reserved

 Bit 1: Terminating resistor for encoder signal line

 0: Active

 1: Deactivated

 Bit 2: Input frequency of the position encoder input at 1 V_{pp}:

 0: 27 kHz

 1: 400 kHz

 Bit 3: Analog or digital position encoder input

 0: Analog encoder signal control (1 V_{pp})

 1: Digital encoder signal control (EnDat 2.2)

 Bit 4: Reserved

 Bit 5: Reserved

MP119.x Position encoder input for spindles

Format: %xxxx

Input: Bit 0: Position encoder input 1 V_{pp}

 0: 1 V_{PP}

 1: Reserved

 Bit 1: Terminating resistor for encoder signal line

 0: Active

 1: Deactivated

 Bit 2: Input frequency of the position encoder input at 1 V_{pp}:

 0: 27 kHz

 1: 400 kHz

 Bit 3: Analog or digital position encoder input

 0: Analog encoder signal control (1 V_{pp})

 1: Digital encoder signal control (EnDat 2.2)

 Bit 4: Reserved

 Bit 5: Reserved



Direction of traverse

With MP210 and MP1040 you define the direction of traverse of the axes. The counting direction depends on the position in which the encoders are mounted. Configuration errors in these parameters provoke the error message **MOVEMENT MONITORING ERROR IN <AXIS>**. Through W1030/W1112 the NC informs the PLC of the direction in which the axes traverse.

If the speed encoder is also used for position measurement, MP210 must be set for the speed encoder. Configuration errors in these parameters provoke the error message **Standstill monitoring <in axis>**.

MP210 Counting direction of position encoder output signals

Format: %xxxxxxxxxxxxxxxxxxxx

Input: Bits 0 to 17 represent axes 1 to 18

0: Positive

1: Negative

MP1040 Analog axes: Polarity of nominal value voltage Digital axes: Algebraic sign of the nominal speed value

Format: %xxxxxxxxxxxxxxxxxxxx

Input: Bits 0 to 17 represent axes 1 to 18

0: Positive

1: Negative

	Set	Reset
W1030/1112 Current direction of traverse	NC	NC
Bits 0 to 17 represent axes 1 to 18		
0: Positive traverse direction		
1: Negative traverse direction		



Note

The counting direction of the speed encoder signals is defined in the motor table (DIR column). If the error message **C3B0 Motor <AXIS> does not rotate** appears, you must change this entry.

Encoder monitoring

HEIDENHAIN contouring controls monitor the signal transmission from the encoders. With machine parameters MP20.x and MP21.x you activate the monitoring function for the position encoders. The following criteria are checked:

Criterion	Error message
Absolute position with distance-coded reference marks	Encoder <AXIS> DEFECTIVE
Amplitude of encoder signals	Encoder AMPLITUDE TOO LOW <AXIS>
Edge separation of encoder signals	Encoder <AXIS>: FREQUENCY TOO HIGH

MP20

Monitoring functions for the axes

Format: %xxxxxxxxxxxxxxxxxxxx

Input: Bits 0 to 17 represent axes 1 to 18

0: Monitoring not active

1: Monitoring active

MP20.0 Absolute position of distance-coded reference marks

MP20.1 Amplitude of encoder signals

MP20.2 Edge separation of encoder signals

MP21

Monitoring functions for the spindle

Format: %xx

Input: Bit 0 – Spindle 1

0: Monitoring not active

1: Monitoring active

Bit 1 – Spindle 2

0: Monitoring not active

1: Monitoring active

MP21.0 Absolute position of distance-coded reference marks

MP21.1 Amplitude of encoder signals

MP21.2 Edge separation of encoder signals



Note

Please note:

- For digital axes, the speed encoders are always monitored if the drive is switched on via PLC Module 9162
- For more information on error messages from speed encoders, see the "Error Messages" section.



Monitoring for encoders with EnDat interface:

In the event of a disturbance, the error message **EnDat defective <error code> <axis>** will appear.

The error code is shown in hexadecimal notation. Error codes may also appear combined, in which case they are added together.

There are two possible types of errors:

- The encoder reports an error.
- Access to the encoder via the EnDat interface is faulty.

Codes for errors reported by the encoder:

Error code	Meaning
0x00000001	Light source defective
0x00000002	Signal amplitude too low
0x00000004	Incorrect position value
0x00000008	Overvoltage
0x00000010	Undervoltage
0x00000020	Overcurrent
0x00000040	Replace the battery
0x00000080	Reserved
0x00000100	Reserved
0x00000200	Reserved
0x00000400	Reserved
0x00000800	Reserved
0x00001000	Reserved
0x00002000	Reserved
0x00004000	Reserved
0x00008xxx	EnDat could not be read. Possible causes: <ul style="list-style-type: none">■ Encoder defective■ Check the wiring (cable and EnDat amplifier)■ Encoder not connected■ Encoder connected to wrong connector■ Motor and position encoder switched■ Check cable lengths

Error codes if the access to the encoder via the EnDat interface is faulty:

Error code	Meaning
0x7F	If a motor with a newer generation multiturn encoder is attached to a CC 424 with a software less than 340 49x-02 SP7, the EnDat error 0x7F is reported. In this case, the NC software must be updated to 340 49x-SP7, 340 49x-03, or higher.
0x80010000	Delete the alarm bit
0x80020000	Read the alarm status
0x80040000	Read the number of pulses
0x80080000	Read the number of signal periods
0x80100000	Read the number of differentiable revolutions
0x80200000	Read the measuring steps
0x80400000	Read the serial number
0x80800000	Read the type of encoder
0x81000000	Read the position value
0x82000000	Reserved
0x84000000	Reserved
0x88000000	Read the checksum
0x90000000	Alarm bit remains set
0xA0000000	Timeout while waiting for data signal "high"
0xC0000000	Timeout while waiting for data signal "low"
0x80000000	Error during access to EnDat interface

Speed encoder

The iTNC 530 uses the **Type of encoder** entry in the "motor.mot" motor table. If an encoder with Z1 track is entered in the motor table, the message **C310 Z1 track error** appears in the event of an error. If an encoder with EnDat interface is entered in the motor table, the control attempts to communicate with the encoder. If this fails, the error message **C3F0 EnDat not found <axis>** appears.



Attention

If you use the HEIDENHAIN standard motor table motor.mot and motors with EnDat encoders, you might have to change the entry for the motor in the SYS column (type of encoder) of the motor table or enter a new motor.

- SYS = 1: Incremental rotary encoder with Z1 track
- SYS = 2: Absolute speed encoder with EnDat interface

EnDat 2.2 encoders

Encoders with EnDat 2.2 interface can be operated without analog encoder signals. The position is evaluated purely digitally via a serial data protocol.

For **motor encoders with EnDat 2.2 interface**, the control requires the following values for calculating the signal period:



Note

Encoders with EnDat 2.2 interface do not supply any incremental signals. Only a purely digital measured value is transferred to the control. The number of increments per encoder revolution (MP332.x) must therefore be set to the value "1".

The controller unit assumes that the rotary encoder has only one line per encoder revolution. Enter the following data:

- ▶ M331.x: For each axis, enter the distance traversed per motor revolution.
- ▶ MP332.x: Enter the value "1".
- ▶ MP1054.x: For each axis, enter the distance traversed per motor revolution.
- ▶ Enter the value "1" in the **STR** column (line count of the motor encoder) of the motor table.

If the temperature signal of the EnDat2.2 encoder is transmitted via the additional lines for a KTY in the cable to the speed input of the control, and not via the EnDat2.2 protocol, this can be configured with MP2222 bit 2 = 1. If you use an external temperature sensor, you always need to use a KTY84-130.



Note

Large input values in MP331.x and MP332.x cannot be read by the PLC!



Note

HEIDENHAIN recommends:

The optimum manufacturing precision can only be attained with linear encoders. A control without linear encoders finds the axis position through the pitch of the ball screw. The problem is, the ball screw gets hot during machining and expands. And the result is a position measurement that deviates from the actual position, causing error. With linear encoders, the control always determines the correct slide position. This means that your machine positions its axes with constantly high precision.

More information is available on the Internet at:

<http://www.heidenhain-shows-the-way.com>

MP2222.x Bit 2 – Temperature signal from KTY input, and not from EnDat 2.2 protocol

Format: %xxxxxxxxxxxxxxxx

Input: 0: Inactive

1: Temperature signal from KTY input

Encoders with multiturn information

MP1358 is used to activate axis-specific evaluation of the multiturn information on the respective encoders.

The reference position of the axis can be read out as usual with PLC Module 9041. If $MP1358.x = 1$, the return value is the position of the encoder plus the multiturn information (e.g. 730 degrees) plus the value in $MP960.x$. As this return value of the module now also includes the multiturn information of the encoder, this needs to be considered when defining $MP960.x$ and $MP1356.x$.

MP1358.x Evaluate multiturn information

Format: %000000000000000000000000

Input: 0: Do not evaluate multiturn information

1: Evaluate multiturn information

6.1.4 Assignment for axes

With the following machine parameters you assign the position and speed encoder inputs, the speed command output and the machine parameter block of the current and speed controller to the individual logic axes:

- ▶ In $MP108.x$, enter the number of the controller's motherboard.
The HSCI address of the respective drive-control motherboard is entered in $MP108.x$. The HSCI address is obtained from the position of the drive-control motherboard in the HSCI system. However, the HSCI address to be entered only depends on the drive-control motherboards in the system. I/O units (PLs) and machine operating panels (MBs) are not taken into account. This means that for the first drive-control motherboard, you have to enter the address 0 in $MP108$, regardless of whether I/O units or machine operating panels are located before the CC in the HSCI chain.
 $MP120.x$ is used for further assignment of the axes to the outputs of the respective drive-control motherboard.
- ▶ In $MP110.x$ you enter the number of the position encoder input. An error message appears if an invalid number is entered.
- ▶ In $MP112.x$ you enter the number of the speed encoder input.
- ▶ In $MP120.x$ you enter the number of the speed command output.
- ▶ In $MP130.x$ you enter index number y of machine parameter block $MP2xxx.y$ of the current and speed controller. This way different machine parameter blocks $MP2xxx.y$ can be used for the axis and spindle in C-axis operation.



With the CC 61xx controllers, the MP120.x is used to assign the outputs to the axes. Entering X51 to X56, X80 to X85 in MP120.x also provides all further connector assignments to the axis, because the assignment between speed encoder, PWM output (power module output on the UEC) and position encoder is a permanent assignment:

When using a CC 61xx in conjunction with HEIDENHAIN inverters

Input of MP120.x/MP121.x: PWM output	51	52	53	54	55	56
Input: Speed encoder	X15	X16	X17	X18	X19	X20
Input: Position encoder (input in MP110.x/MP111.x)	X201	X202	X203	X204	X205	X206

When using a UEC:

Input in MP120.x: Speed output: PWM output	80	81	82	83	84	85
Input: Speed encoder	X15	X16	X17	X18	X19	X20
Input: Position encoder	X201	X202	X203	X204	X205	X206



Note

Depending on the maximum spindle speed, it might no longer be possible to use all PWM outputs (see "Specifications" on page 1067).

If MP120.x = 0, then the axis will only be displayed.

Digital axes: If MP110.x = 0, then the speed encoder (with or without EnDat interface) is also used for position control.

MP108.x Assignment of axes to drive-control motherboards

Input: 0 to 4: Number of the drive-control motherboard in the HSCI chain

MP110.x Assignment of position encoder inputs to the axes

Input: 0: No position encoder
201 to 206: Position encoder inputs X201 to X206

MP112.x Assignment of speed encoder inputs to the axes

Input: 0: No speed encoder input
15 to 20: Speed encoder inputs X15 to X20

MP120.x Nominal speed command outputs of the axes

Input: 0: No servo-controlled axis
1 to 8: Analog servo-controlled axis over CMA-H
51 to 56: Digital speed command outputs X51 to X64 of the CC
80 to 85: Digital speed command output outputs X80 to X85 of the UEC

MP130.x Y index of the machine parameters MP2xxx.y for the axes

Input: 0 to 17



6.1.5 Assignment for spindles

With the following machine parameters you assign the position and speed encoder inputs, the speed command output and the machine parameter block of the current and speed controller to the spindle/spindles:

- ▶ In MP109.x, enter the number of the controller's motherboard.
The HSCI address of the respective drive-control motherboard is entered in MP109.x. The HSCI address is obtained from the position of the drive-control motherboard in the HSCI system. However, the HSCI address to be entered only depends on the drive-control motherboards in the system. I/O units (PLs) and machine operating panels (MBs) are not taken into account. This means that for the first drive-control main board, you have to enter the address 0 in MP109, regardless of whether I/O units or machine operating panels are located before the CC in the HSCI chain.
MP121.x is used for further assignment of the spindles to the outputs of the respective drive-control motherboard.
- ▶ In MP111.x you enter the number of the position encoder input. An error message appears if an invalid number is entered.
- ▶ In MP113.x you enter the number of the speed encoder input.
- ▶ In MP121.x you enter the number of the speed command output.
- ▶ In MP131.x and MP132.x you enter index number y of machine parameter block MP2xxx.y of the current and speed controller.

With the CC 61xx controllers, the MP121.x is used to assign the outputs to the spindles. Entering X51 to X56, X80 to X85 in MP121.x also provides all further connector assignments to the spindle, because the assignment between speed encoder, PWM output (power module output on the UEC) and position encoder is permanent.

When using a CC 61xx in conjunction with HEIDENHAIN inverters

Input of MP120.x/MP121.x: PWM output	51	52	53	54	55	56
Input: Speed encoder	X15	X16	X17	X18	X19	X20
Input: Position encoder (input in MP110.x/MP111.x)	X201	X202	X203	X204	X205	X206

When using a UEC:

Input in MP121.x: Speed output: PWM output	X80	X81	X82	X83	X84	X85
Input: Speed encoder	X15	X16	X17	X18	X19	X20
Input: Position encoder	X201	X202	X203	X204	X205	X206

C-axis operation or a central drive is an exception where you can configure a second position encoder input for a drive, see page 987.

MP109	Assignment of spindle(s) to drive-control motherboards
Input:	0 to 4: Number of the drive-control motherboard in the HSCI chain
MP111	Position encoder input for the spindle/spindles
Input:	0: No position encoder 201 to 206: Position encoder inputs X201 to X206
MP111.0	Position encoder input for the first spindle
MP111.1	Position encoder input for the second spindle
MP113	Speed encoder for the spindle/spindles
Input:	0: No speed encoder input 15 to 20: Speed encoder inputs X15 to X20
MP113.0	Speed encoder for the first spindle
MP113.1	Speed encoder for the second spindle
MP121	Nominal speed command output of the spindle/spindles
Input:	0: No servo-controlled spindle 1 to 8: Analog servo-controlled spindle over CMA-H 51 to 56: Digital speed command outputs X51 to X56 of the CC 80 to 84: Digital speed command output outputs X80 to X84 of the UEC
MP121.0	Nominal speed command output of the first spindle
MP121.1	Nominal speed command output of the second spindle
MP131	Y index of the machine parameters MP2xxx.y for the spindle(s) in operating mode 0
Input:	0 to 17
MP131.0	Y index of first spindle
MP131.1	Y index of second spindle
MP132	Y index of the machine parameters MP2xxx.y for the spindle(s) in operating mode 1
Input:	0 to 17
MP132.0	Y index of first spindle
MP132.1	Y index of second spindle

6.1.6 Specifications of the controller unit:

Machine interfacing		iTNC 530	
Regulation with CC 6xxx / UEC 1xx			
Position control resolution	$\frac{\text{Signal period}}{4096}$	or encoder resolution (EnDat 2.2 interpol.)	
Path interpolation	3 ms		
Fine interpolation	Cycle time for fine interpolation = Cycle time of position controller		
Cycle time of current controller	PWM frequency	Cycle time if MP2182.x = 0:	
	3333 Hz	150 μ s	
	4000 Hz	120 μ s	
	5000 Hz	100 μ s	
	6666 Hz	75 μ s (option 49)	
	8000 Hz	60 μ s (option 49)	
	10000 Hz	50 μ s (option 49)	
Cycle time of speed controller	<ul style="list-style-type: none"> ■ Speed controller cycle time = 2 · current controller cycle time ■ Unless PWM frequency \leq 5 kHz and double-speed performance, then: Speed controller cycle time = Current controller cycle time 		
Cycle time of position controller	Position controller cycle time = Speed controller cycle time		



Machine interfacing	iTNC 530
Maximum motor speed	$n_{\max} = \frac{f_{\text{PWM}} \cdot 60000 \text{ min}^{-1}}{p \cdot 5000 \text{ Hz}}$ <p> n_{\max}: Maximum motor speed [min^{-1}] f_{PWM}: PWM frequency [Hz] p: Number of pole pairs </p> <p>The following PWM frequencies are available: 3333 Hz, 4000 Hz, 5000 Hz With option 49: 6666 Hz, 8000 Hz, 10000 Hz</p>
or by way of:	<p>Maximum signal frequency of motor encoder = 400 kHz or 800 kHz (depending on MP2561.x bit 0)</p> <p>Calculation of maximum motor speed:</p> $n_{\max} = \frac{f_{\max} \cdot 60000 \text{ [s/min]}}{\text{STR}}$ <p> n_{\max}: Maximum motor speed [min^{-1}] f_{\max}: Maximum signal frequency of motor encoder [kHz] STR: Encoder line count </p> <p>Example 1:</p> $n_{\max} = \frac{400 \text{ kHz} \cdot 60000 \text{ [s/min]}}{2048}$ $n_{\max} = 11718.75 \text{ min}^{-1}$ <p>Example 2:</p> $n_{\max} = \frac{800 \text{ kHz} \cdot 60000 \text{ [s/min]}}{2048}$ $n_{\max} = 23437.5 \text{ min}^{-1}$ <p>Switching the maximum input frequency for motor encoders through MP 2561.x bit 0 to 800 kHz may be necessary for high-speed motors or spindles, depending on the resolution of the motor encoder. The precondition is that the encoder be suitable for this frequency range and still be able to provide a sufficient signal amplitude at 800 kHz.</p>

Machine interfacing	iTNC 530
Maximum feed rate by way of:	<p>Maximum signal frequency of the position encoder:</p> <ul style="list-style-type: none"> ■ At 1 V_{pp}: 27 kHz or 400 kHz ■ At 11 μA_{pp}: 27 kHz or 140 kHz <p>Calculation of the maximum feed rate F_{max}:</p> $F_{\max} = SP \cdot f_{\max} \cdot 60^{-3} \left[\frac{\text{m} \cdot \text{s}}{\mu\text{m} \cdot \text{min}} \right]$ <p>F_{max}: Maximum feed rate [m/min] f_{max}: Maximum signal frequency of position encoder [kHz] SP: Signal period of the position encoder [kHz]</p> <p>Example:</p> $F_{\max} = 20 \mu\text{m} \cdot 400 \text{ kHz} \cdot 60^{-3} \left[\frac{\text{m} \cdot \text{s}}{\mu\text{m} \cdot \text{min}} \right]$ <p>F_{max} = 480 m/min</p>

6.1.7 Reading axis information

Module 9038 Read general axis information

With Module 9038 you can interrogate the general status information of the axes. You can interrogate the status of a specific axis or of all axes at once. Bits 0 to 14 represent axes 1 to 15. Bit 15 represents the spindle (depends on MP4020 bit 14). If status information is read for only one axis, only bit 0 is changed. The following table shows the meanings of the return codes:

Status information	Meaning
0	0: Axis (spindle) not active (MP10 or MP3010 or no encoder) 1: Axis (spindle) active
1	Depending on the current traverse range: 0: NC axis or not active 1: PLC axis
2	0: No closed-loop axis (spindle), only display or not active 1: Closed-loop axis (spindle)
3	Maximum temperature of the motor [°C]
4	0: Not a Hirth axis 1: Hirth axis (MP420)
5	Hirth grid [1/10 µm] (MP430)
6	Modulo value (MP810)
7	0: Linear axis or not active 1: Rotary axis in at least one of the traverse ranges
8	0: Analog axis (spindle) or not active 1: Digital axis (spindle)
9	0: Not a slave axis 1: Slave axis

Call:

PS B/W/D/K <Axis>
Axis-specific: Dependence on MP4020 bit 14
Bit-encoded output for all axes: -1

PS B/W/D/K <Status information>
See table above

CM 9038

PL B/W/D <Information>

Error code:

Marker	Value	Meaning
M4203	0	Information was read
	1	Error code in W1022
W1022	1	Status information not available on this iTNC
	2	Axis not found

Current tool axis

You can define the current tool axis in two ways in the NC block:

- In the HEIDENHAIN conversational dialog with TOOL CALL
- In ISO programming with G17 to G20

In the PLC you can interrogate the current tool axis via markers (only axis 0 to 9) or via module: This information is only available after each tool call with "TOOL CALL," and is then updated after each tool call.

		Set	Reset
M4526	Axis 1 is tool axis	NC	NC
M4527	Axis 2 is tool axis	NC	NC
M4528	Axis 3 is tool axis	NC	NC
M4529	Axis 4 is tool axis	NC	NC
M4530	Axis 5 is tool axis	NC	NC
M4531	Axis 6 is tool axis	NC	NC
M4532	Axis 7 is tool axis	NC	NC
M4533	Axis 8 is tool axis	NC	NC
M4534	Axis 9 is tool axis	NC	NC

Module 9035 Read status information (current tool axis)

Call:

PS B/W/D/K <100>

CM 9035

PL B/W/D <Current tool axis>

0 to n: Axes 1 to n+1

-1: No information exists about the current tool axis

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Status information invalid



6.1.8 Traverse ranges

You can divide the working range of the machine into three traverse ranges, e.g. one for each workpiece. Each traverse range is limited by a software limit switch.

For the software limit switch of a traverse range:

- The datum is the machine datum (MP960.x).
- Software limit switches for tilting axes must be activated with MP812 when $MP810.x \neq 0$
- The traverse range can be limited further through the MOD function.
- If a software limit switch is activated, the error message **LIMIT SWITCH <AXIS>** appears.
- Software limit switches can be overwritten with **FN17:SYSWRITE**, e.g. for automatic tool change. This function is effective only until the next GOTO command (GOTO key or FN9 to FN12) or the end of the program.

Determining range of traverse

- ▶ You can determine the current range of traverse with Module 9035

Module 9035 Read status information

Call:

```
PS   B/W/D/K <27>
CM   9035
PL   B/W/D   <Range of traverse>
           0 to 2: Traverse ranges 1 to 3
```

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Status information invalid
	20	Call was not in a submit or spawn job

Selecting the traverse range

You can switch the range of traverse in two ways:

- ▶ Select the traverse range with Module 9151 or 9152. With Module 9152 you can change the axis display at the same time.

or

- ▶ Select the traverse range with M4574 and M4575 according to the table below.
- ▶ In all operating modes you must activate the traverse range with strobe marker M4135.

You may only use the traverse range switching function via M4574 and M4575 during an M/S/T/Q strobe in all operating modes (except for **Manual Operation** and **E1. Handwheel**).

M4574	M4575	Traverse range/Datum
0	0	Range 1
1	0	Range 2
0	1	Range 3

If the axis traverse limits are changed by the user (after pressing the MOD key), marker M4624 is set. The marker must then be reset by the PLC program.

		Set	Reset
M4574	Select the traverse range (with M4575)	PLC	PLC
M4575	Select the traverse range (with M4574)	PLC	PLC
M4135	Strobe marker for selecting the traverse range	PLC	NC
M4624	Strobe marker for selecting the traverse range	NC	PLC



Module 9150 Axis-specific reading of axis traverse limits

PLC module 9150 enables you to read the axis traverse limits that can be defined (after pressing the MOD key).

If the axis traverse limits are set by the user (after pressing the MOD key), marker 4624 is set. The marker must then be reset by the PLC program.

Condition:

- The default values entered in MP91x.x are the maximum possible values for the axis traverse limits.

Call:

PS B/W/D/K <Axis>

PS B/W/D/K <Traverse range>

0: Traverse range 1

1: Traverse range 2

2: Traverse range 3

-1: Current traverse range

PS B/W/D/K <Mode>

0: Axis traverse limits that were changed with the MOD key

CM 9150

PL D <Positive axis traverse limit>

PL D <Negative axis traverse limit>

Error code:

Marker	Value	Meaning
M4203	0	Axis traverse limit determined
	1	Error code in W1022
W1022	1	Invalid value for mode
	2	Invalid value for axis or traverse range

Module 9151 Select traverse range and axis designation

With Module 9151 you can select the traverse range and the axis designation in one step. The axis designations in MP100.x are overwritten and cannot be activated until the module has received the axis designation -1, a traverse range has been activated with M4135, MP100.x is edited, or the control has been reset.

When the module is called it sets M4135. After switchover the NC resets M4135.

The PLC module 9151 can also be called when the user has stopped the machining process with **External Stop**, and has then activated the MANUAL TRAVERSE function with the appropriate soft key. When RESTORE POSITION is activated, the system checks whether the traverse range has been reset to the original value. If this is not the case, an error message is output and the RESTORE POSITION function is not executed. You can, however, cancel the machining process with **External Stop** or Emergency Stop.

If the traverse range is switched when the MANUAL TRAVERSE function is active (activated by soft key), an active M128 function is automatically deactivated. After switching back with RESTORE POSITION, M128 is reactivated.

Call:

PS B/W/D/K/S<String with axis designation>

Format: XYZABCUVWxyzabcuvw

Characters 1 to 9 represent axes 1 to 9

With -1 the axis designations from M100.x are valid

PS B/W/D/K <Traverse range>

0 to 2: Range of traverse

-1: Do not change range of traverse

CM 9151

Error code:

Marker	Value	Meaning
M4203	0	Traverse range/axis designation switched over
	1	Error code in W1022
W1022	2	Invalid value for traverse range
	3	For the axis assignment, neither a string nor -1 was transferred
	21	The module was called during a running part program or without an M/S/T/Q strobe



Module 9152 Selecting traverse range, axis display and axis designation

With Module 9152 you can select the traverse range, the axis designation, and the axis display. The axis designations in MP100.x and the axis display in MP7291 are overwritten and cannot be activated until the module has received -1 for the axis designation and axis display, a traverse range has been activated with M4135, MP100.x or MP7291 have been edited, or the control has been reset.

When the module is called it sets M4135. After switchover the NC resets M4135.

Call:

PS B/W/D/K/S<String with IV and V key configuration>

Format: AB

The first character represents the IV key, the second character represents the V key

With -1 the key configuration from MP410 is valid

PS B/W/D/K/S<String with axis display>

Format: SWWUCBAZYXwwucbazyx

Characters 1 to 9 (from the right) represent lines 1 to 9

Character 10 = S always in line 10

With -1 the axis display from MP7291 is valid

PS B/W/D/K/S<String with axis designation>

Format: XYZABCUVWxyzabcuvw

Characters 1 to 9 represent axes 1 to 9

With -1 the axis designations from M100.x are valid

PS B/W/D/K <Traverse range>

0 to 2: Range of traverse

-1: Do not change range of traverse

CM 9152

Error code:

Marker	Value	Meaning
M4203	0	Traverse range, axis designation and axis display are switched
	1	Error code in W1022
W1022	2	Invalid value for traverse range, or string for axis configuration, axis display or key configuration is too long
	3	For the axis assignment, axis display or key configuration, neither a string nor -1 was transferred
	21	The module was called during a running part program or without an M/S/T/Q strobe

Setting the software limit switches

With the following machine parameters, you can set the software limit switches for the various ranges of traverse. The position values are with respect to the machine datum. Ranges of traverse 2 and 3 do not become effective until they are activated by M4574, M4575 and M4135 or with Module 9151 or 9152.



Note

The values for MP910.x, MP911.x, MP912.x, MP920.x, MP921.x and MP922.x can be transferred with the actual-position-capture key.

MP910.x Positive software limit switches, traverse range 1 (default setting after power on)

Input: -99 999.9999 to +99 999.9999 [mm] or [°]

MP911.x Positive software limit switches, traverse range 2

Input: -99 999.9999 to +99 999.9999 [mm] or [°]

MP912.x Positive software limit switches, traverse range 3

Input: -99 999.9999 to +99 999.9999 [mm] or [°]

MP920.x Negative software limit switches, traverse range 1 (default setting after power on)

Input: -99 999.9999 to +99 999.9999 [mm] or [°]

MP921.x Negative software limit switches, traverse range 2

Input: -99 999.9999 to +99 999.9999 [mm] or [°]

MP922.x Negative software limit switches, traverse range 3

Input: -99 999.9999 to +99 999.9999 [mm] or [°]

MP7490 Functions for traverse ranges

Format: %xxxx

Input: Bit 0 = 0: Display one traverse range via MOD
 Bit 0 = 1: Display three traverse ranges via MOD
 Bit 1 = 0: Each traverse range has its own datum (and 3 memories for the positions of the swivel head)
 Bit 1 = 1: One datum for all traverse ranges

The NC reports the activation of limit switches to the PLC in words W1034/W1120 and W1036/W1124:

	Set	Reset
W1034/1120 Positive software limit switch was approached Bits 0 to 17 represent axes 1 to 18	NC	NC
W1036/1124 Negative software limit switch was approached Bits 0 to 17 represent axes 1 to 18	NC	NC



Special function M150

If NC blocks were used in the Positioning with Manual Data Input, Program Run Single Block, and Program Run Full Sequence operating modes to program positions that are outside of the traverse ranges, then normally the blocks containing this violation are not performed, and an error message is output.

With M150 the block is traversed to at least shortly before the limit of the traverse range, despite this programming violation. Positioning is performed as close to the limit of the traverse range as possible. For example, if the limit is –600.000 and the programmed position is –700.000, M150 traverses to –599.999. This means that the limit switch information for the PLC via W1034/W1120 and W1036/W1124 is not set, since the limit switch is not traversed to.

6.1.9 Lubrication pulse

You can define the traverse distance for each axis after which the PLC commands lubrication:

- ▶ In MP4050.x you define the traverse distance at which the lubrication pulse is to be output. The NC reports in W1056/W1164 when the entered distance in an axis has been exceeded.
- ▶ With W1058/W1168 you reset the distance counter to 0 after lubrication.

The summation of the path traversed always occurs in MP4050.x, regardless of the operating mode and how the axis was moved.

MP4050.0-8 Path-dependent lubrication of axes 1 to 9

Input: 0 to 99 999.999 [m or 1000°]

	Set	Reset
W1056/1164 Lubrication pulse: Value in MP4050.x exceeded Bits 0 to 17 represent axes 1 to 18	NC	NC
W1058/1168 Reset the accumulated distance Bits 0 to 17 represent axes 1 to 18	PLC	PLC

6.2 PLC axes

You can assign the controlled axes individually to the PLC.

Remember that:

- PLC axes can be operated with following error (also called lag) or with velocity feedforward control. The axis-specific jerk (MP1097.x and MP1098.x) is accounted for.
- You can start more than one axis simultaneously. However, the axes are not interpolated with each other.
- PLC axes are positioned by the shortest path if you enter a modulo value in MP810.x.
- Up to 20 commands (such as positioning, override settings, etc.) for PLC axes can be executed per run-through of the PLC program.

With MP100.x you define for every traverse range which axes the PLC controls and which the NC controls. Uppercase letters represent NC axes, and lowercase letters represent PLC axes. To indicate axes that are not present, mark them with a hyphen (-).

Furthermore, based on the settings in MP810.x, the control detects whether the PLC axis is a rotary axis.

MP100	Designation of axes
Format:	XYZABCUVWxyzabcuvw-
Input:	Bits 0 to 17 represent axes 1 to 18
MP100.0	Traverse range 1
MP100.1	Traverse range 2
MP100.2	Traverse range 3



Module 9120 Start PLC axis

This module starts the positioning of a PLC axis regardless of other processes in the control.

Conditions:

- Status changes through a PLC positioning command are not detected until the next PLC scan.
- The axis must be activated in MP10 and identified in MP100 as a PLC axis.
- Traverse over the software limit switches is not checked.
- The axis must be stationary before positioning. Interrupt a running positioning movement with Module 9121.
- Feed-rate override is disabled. To change the feed rate, use Module 9124.
- If no reference mark has been traversed, the positioning process builds on the counter value as it was upon switch-on.

Call:

PS B/W/D/K <Axis>
0 to 17 represent axes 1 to 18

PS B/W/D/K <Target position>
Input unit: [0.0001 mm]

PS B/W/D/K <Feed rate>
Input unit: [mm/min]

PS B/W/D/K <Mode>
Bit 0: Type of target position input
0: Absolute, i.e. relative to the machine datum
1: Incremental

CM 9120

PL B/W/D <Error code>
0: No error. Positioning was started.
1: Axis does not exist
2: Not a PLC axis
3: Axis is already being positioned
4: Absolute position is outside of modulo range
5: Programmed axis not in closed loop
6: Feed rate not permitted

Module 9121 Stop PLC axis

Stops a running PLC positioning process in an axis.

Condition:

- Status changes through a PLC positioning command are not detected until the next PLC scan.

Call:

PS B/W/D/K <Axis>
0 to 17 represent axes 1 to 18

CM 9121

PL B/W/D <Error code>
0: Positioning is canceled
1: Axis does not exist
2: Not a PLC axis
3: Axis was already stationary

Module 9122 Status of PLC axis

Request for PLC positioning status.

Module 9122 was expanded by bit 7. Bit 7 is only supported in positioning jobs that were started with PLC Module 9227. During positioning with Module 9120, limit-switch monitoring as with Module 9227 bit 1 = 1 is not active.

Condition:

- Status changes through the PLC positioning command are not detected until the next PLC scan.

Call:

PS	B/W/D/K	<Axis> 0 to 17 represent axes 1 to 18
CM	9122	
PL	B/W/D	<Status> Bit 0 – PLC axis? 0: NC axis or not active 1: PLC axis Bit 1 – Reference mark 0: Reference mark not yet traversed 1: Reference mark traversed Bit 2 – Positioning 0: Inactive 1: Active Bit 3 – Direction of motion 0: Positive 1: Negative Bit 4 – Positioning error 0: No positioning errors occurred 1: Positioning error Bit 5 – Closed-loop or open-loop axis 0: Closed-loop axis was programmed 1: Axis programmed which was switched to open-loop in Module 9155 Bit 6 – Target position reached? 0: Target position not yet reached 1: Target position reached Bit 7 – Axis in limit-switch range? 0: Axis not in limit-switch range 1: Axis in limit-switch range



Module 9123 Traverse the reference marks of PLC axes

Traverse the reference marks as for NC axes. This module is not suitable for encoders with distance-coded reference marks (use Module 9220).

- You can use the same procedure to traverse a reference mark for PLC axes as for NC axes. Use Module 9123 only if no conventional procedure is possible.
- Module 9123 moves the axis in the given direction until the reference mark has been traversed. The axis stops next to the reference mark, offset by the braking path.

Call:

PS B/W/D/K <Axis>
Axis-specific: Dependence on MP4020 bit 14

PS B/W/D/K <Feed rate>
Input unit: [mm/min]

PS B/W/D/K <Mode>
Bit 0: Direction of traverse
0: Positive
1: Negative

CM 9123

PL B/W/D <Error code>
0: No error. Positioning was started.
1: Axis does not exist
2: Not a PLC axis
3: Axis is already being positioned
5: Programmed axis not in closed loop
6: Feed rate not permitted

Module 9124 Feed rate override for PLC axis

Enter a default feed-rate override for a PLC axis if positioning was started with Module 9120 or 9123.

Conditions:

- After interruption of a PLC program, the override value is set to 100%.
- When a positioning is started, the last defined override value is in effect.
- The override value can also be changed during a positioning movement.

Call:

PS B/W/D/K <Axis>
0 to 17 represent axes 1 to 18

PS B/W/D/K <Override>
Input unit: 0 to 10 000, corresponds to 0 to 100 % in 0.01 % steps.

CM 9124

PL B/W/D <Error code>
0: No error, override value was set
1: Axis does not exist
2: Not a PLC axis
3: Override value incorrect

Module 9125 Stop PLC axis at next Hirth grid position

Stop an already started PLC positioning movement of an axis at the next Hirth grid position.

Call:

PS B/W/D/K <Axis>
0 to 17 represent axes 1 to 18

CM 9125

PL B/W/D <Error code>
0: Positioning is canceled
1: Axis does not exist
2: Not a PLC axis
3: Axis was already stationary
4: Axis is not a Hirth axis (MP420.x)

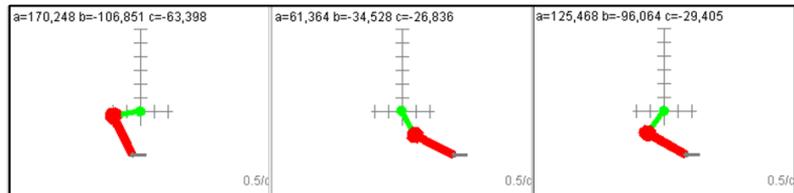


6.2.1 Coupling function for PLC axes or auxiliary axes

The coupling function makes it possible to control PLC axes in dependency on other PLC axes or an NC axis. This enables you to realize complex simultaneous movements of several PLC axes. The coupling function is activated and deactivated through PLC module 9228. PLC module 9229 supplies the status information on the coupling functions.

The mutual dependence of the axes must be defined in mathematical formulas. The following example explains how to proceed:

A robot arm consists of three limbs, each of which can be controlled as three PLC axes. The arm is to move only in such a way that the third limb and its joint (shown in gray) move horizontally at a constant height.



The example is based on the following assumptions:

- At the 0° position the arm extends horizontally to the right and all joints are extended.
- The length of the second arm (marked in red) is 1.6 times the length of the first arm (marked in green).
- The third limb and its joint (shown in gray) must move horizontally at a constant height.
- a, b, c: Angles of the respective joints
- ma, mb, mc: Length of the respective components

The following condition must be fulfilled for the third joint to remain at a constant height h:

- Height of the first joint = 0
- Inclination of the first arm (green) = a
- Height of the second joint = $\sin(a) * ma$
- Inclination of the second arm (red) = a + b
- Height of the third joint = $\sin(a) * ma + \sin(a+b) * mb$
- This results in height $h = \sin(a) * ma + \sin(a+b) * mb$

To determine the angle b as a function of a, you need the definition of b, which is derived from the formula for h as follows:

- $h = \sin(a) * ma + \sin(a+b) * mb$
- $b = -a + \arcsin((h/mb - \sin(a) * ma/mb))$

To keep the third arm (gray) horizontal, a + b + c must be equal to 0 (a + b + c = 0). Therefore:

- $c = -a - b$
- and: $c = -\arcsin((h/mb - \sin(a) * ma/mb)) - a$

Realization

For the coupling functions (PLC module 9228), the resolution of the nominal positions in the PLC module is 0.0001° . The trigonometric functions $\sin()$, $\cos()$, $\tan()$, etc. require data input in radians ($360^\circ = 2 * \pi$).

The inverse trigonometry functions supply the result in radians. To obtain the radian values from the nominal positions in internal-resolution form, the position value of the axis must be multiplied by $\pi / 1800\,000$ or in shorter form by $\pi/18e5$.

Hence, the value of an inverse trigonometry function must be multiplied by $18e5/\pi$ to obtain the internal coordinate resolution.

The constants h , ma and mb result from the given mechanical design.

The example using the values $h = 1.6$, $ma=1$, $mb=1.6$ results in the synchronous function being called in the PLC program (the error handling is missing here) as follows:

```
LBL MF49      ;activate the coupling function
L S" -a + asin(1 - 1/1.6*sin(a*pi/18e5))*18e5/pi"
= S0
PS K0         ;string number
PS K4         ;axis
PS K1         ;mode: Switch-on
CM 9228
PLX

;coupling function for additional picker axis
L S" - asin(1 - 1/1.6*sin(a*pi/18e5))*18e5/pi "
= S0
PS K0         ;string number
PS K5         ;axis
PS K1         ;mode: Switch-on
CM 9228
PLX
```

EM

The following process must be realized by the PLC program:

- Positioning the PLC axis to a suitable starting position
- Activating the coupling function
- Moving the first PLC axis appropriately (by means of positioning movements through the PLC), the other PLC axes are automatically corrected.
- Deactivating the coupling function when the respective positioning movement has been completed



Note

For the coupled axis (axes), the software limit switches are not monitored!

When activating the coupling function you should set the software limit switches of the principal such that also the coupled axes that are not monitored cannot exceed the actual limit position. When you rescind the coupling function, you must set the default values of the software limit switches again.



Module 9228 Coupling function for auxiliary axes

Module 9228 is used to enable/disable the coupling of auxiliary axes with other auxiliary or NC axes for positioning movements. The resulting motions must be defined in a formula (coupling function) for each coupled axis. This formula is to be transmitted to the module as a PLC string.

Condition:

- The arithmetic operations and trigonometric functions that can be used in a formula are described in "Special case: Entering a formula" in the Machine Parameters Chapter of the Technical Manual.
- The coupling-function data is not forwarded until the PLC cycle has been completed.
- The status of a coupling function (readable with Module 9229) is available in the subsequent PLC cycle at the earliest.
- As long as the coupling function in the interpolator has not taken effect, status module 9229 reports the status "0" for the coupling axis.
- The coupling function can only be activated at standstill.
- The axes must initially be positioned in such a way that the activation of a formula does not result in a nominal-value step. Module 9229 (mode 1, current nominal position with the help of a formula) is used to determine whether the initial position of a coupling axis must be changed.
- For the coupling functions (PLC module 9228), the resolution of the nominal positions in the PLC module is 0.0001° or 0.0001 mm. The trigonometric functions $\sin()$, $\cos()$, $\tan()$, etc. require data input in radians ($360^\circ = 2 * \pi$). The inverse trigonometry functions supply the result in radians. To obtain the radian values from the nominal positions in internal-resolution form, the position value of the axis must be multiplied by $\pi / 180\,000$ or in shorter form by $\pi/18e5$. Hence, the value of an inverse trigonometry function must be multiplied by $18e5/\pi$ to obtain the internal coordinate resolution.
- The evaluation of W1022 is not necessary, because the error codes in the return value make a detailed evaluation possible (see call interface).
- The first PS during the module call must contain the string number of the coupling function with the leading axis (NC/plc letter).
- The second PS during the module call must indicate the axis index of the auxiliary axis to be coupled.
- Limit switches are not monitored for the coupled axis.
- If there is an emergency stop and the machine is then switched on again, the coupling is deactivated. The cancelation can be seen via status module 9229, feedback value 0 in mode 0.



Note

Limit switches are not monitored for the coupled axis. The machine tool builder must ensure that the coupled axis only moves within the permissible area.

Module 9229 Status of coupling function

Module 9229 is used to determine the status and error conditions of Module 9228 (coupling functions). The module can calculate the current position value to avoid nominal-position steps when axes are coupled, and to be able to pre-position the coupling axes appropriately.

Condition:

- The status or error code is available no sooner than during the next PLC scan after PLC module 9228 has been called. (Exception: status/error 4, error code in W1022)

Call:

PS K/B/W/D <Coupling function>
 The parameter is evaluated only if mode = 1.

PS K/B/W/D <Coupling axis>
 The parameter is evaluated only if mode = 0.

PS K/B/W/D <Mode>
 0: Status/Error feedback, possible states/errors: 0, 1, 2, 3, 4 after Module 9228 has been called.
 1: One-time position calculation (only possible in submit/spawn), possible errors: 3, 4

CM 9229
 PL B/W/D <Status/Error>
 If the status/error is 0 to 3, M4203 is not set:
 0: No coupling function active for programmed axis
 1: Coupling function is active for programmed axis
 2: Coupling function canceled (see additional info)
 3: Syntax error in coupling function of Module 9228 or 9229 (see additional info)
 4: See error codes in W1022

PL B/W/D <Additional info>
 Mode 0:
 If additional info on status/error = 2 (cancellation):
 0: Canceled by emergency stop
 1: Canceled due to computing error during run time
 If additional info on status/error = 3 (syntax error):
 0: Not yet supported

Mode 1:
 Position value or additional info in the event of an error:
 0x80000000 (largest negative value)

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022 (also see error code above)
W1022	1	Invalid mode programmed
	2	Invalid axis number programmed
	11	Invalid PLC string programmed as formula
	20	Call from the cyclic PLC program
	45	Canceled due to error message from operating system

6.3 PLC positioning

You can position the axes directly through the PLC. For PLC positioning of the main spindle, see page 944.

Prerequisites

The following constraints apply to a PLC positioning command:

- It is possible in the Manual and Handwheel modes only while there is no positioning movement.
- Possible in the other modes of operation only with an M/S/T/Q strobe or if no part program is started.
- If the NC is positioning an axis, you can position additional axes only if they have already been defined as PLC axes. "PLC axes" on page 630.

Programming

You start a PLC positioning movement with Module 9221 or 9227, interrogate the status with Module 9222, and stop it ahead of time with Module 9224. After Module 9221 or 9227 has been called, markers M4120 to M4128 are set. If you reset these markers directly or program Module 9224 accordingly, positioning is canceled at the desired point in time. This is necessary if you would like to change a parameter, such as the feed rate, during positioning.

The following conditions apply to a PLC positioning command:

- If more than one axis is moved simultaneously, the axes will be interpolated.
- If you start another axis during a PLC positioning movement,
 - the first positioning command will be canceled and
 - the resulting positioning movement will be executed in all axes.
- Tool compensation is not included. Before a PLC positioning command you must end any tool compensation.
- A PLC positioning movement is not displayed in the test graphics.
- PLC positioning is done without nominal position value filter.

The NC cancels a PLC positioning movement under the following conditions:

- If in the Manual or Handwheel modes there is an NC STOP
- If in the automatic operating modes there is an NC STOP and "internal stop"
- If there is an emergency stop
- An error message that results in a STOP
- A reset of Markers M4120 to M4128
- Stopping of the positioning with Module 9224 (all axes)



Module 9221 Start a PLC positioning movement

Starts a PLC positioning movement in one axis.

With module 9221, you can deactivate collision monitoring for a specific PLC positioning movement. This means that it is also possible to position an axis when DCM is active and the reference marks have not yet been traversed in all axes.

A PLC positioning command can only be executed if collision monitoring is deactivated for all axes involved in the positioning movement. The deactivation of collision monitoring for the PLC positioning movement does not affect the status information provided by Module 9064.

Call:

PS B/W/D/K <Axis>
0 to 17 represent axes 1 to 18

PS B/W/D/K <Target position>
Input unit: 0.0001 mm

PS B/W/D/K <Feed rate>
Input unit: mm/min

PS B/W/D/K <Mode>
Bit 0 – Definition of the target position
0: Absolute, i.e. relative to the machine datum
1: Incremental
Bit 1 – Software limit switches
0: Inactive
1: Active
Bit 2 – Reserved
Bit 3 – PLC positioning with collision monitoring deactivated

CM 9221

PL B/W/D <Error code>
0: Positioning is being started
1: Axis is not in a closed loop or is an auxiliary axis
2: Inadmissible values for the feed rate
3: Axis has not traversed the reference mark
4: No M/S/T/Q strobe during started part program
5: Programmed axis not in closed loop
6: PLC positioning already started

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Axis does not exist or status interrogation faulty
W1022	2	Positioning will be started; open-loop axis programmed or impermissible values programmed for the feed rate
	9	PLC positioning already started
	21	No M/S/T/Q strobe during started NC program
	33	Axis has not traversed the reference mark

Module 9222 Status request of PLC positioning movement

With this module you can interrogate the status of a PLC positioning movement for an individual axis or bit-encoded.

The status of an axis or all axes remains until a new status is set when the next PLC positioning of one or more axes occurs.

Call:

PS B/W/DK <Axis>

Interrogation of an axis:

0 to 17 represent axes 1 to 18

Interrogation of all axes:

- 1: Target position reached
- 2: PLC positioning was started
- 3: PLC positioning was canceled
- 4: PLC positioning reached limit switch
- 5: PLC positioning not possible
- 6: PLC positioning temporarily stopped
- 7: PLC positioning not started due to collision monitoring

CM 9222

PL B/W/D <Status>

Interrogation of an axis:

- 0: No PLC positioning was started
- 1: Target position reached
- 2: PLC positioning was started
- 3: Due to cancelation, target position not reached
- 4: Target position is outside of traverse range
- 5: Positioning not possible (e.g. due to "free rotation")
- 6: Positioning temporarily stopped (e.g. stop in Automatic operating modes)
- 7: Positioning not started due to collision monitoring

Interrogation of all axes:

Status of PLC positioning movement bit-encoded in word format

Error code:

Marker	Value	Meaning
M4203	0	Status was transferred
	1	Axis does not exist or status interrogation faulty
W1022	1	Invalid status information was requested
	2	The status of an open-loop axis, auxiliary axis or slave axis is being interrogated



Module 9224 Stop PLC positioning movements

Individual PLC positioning movements can be stopped with Module 9224.

- If M4120 to M4128 are already set, they are reset.
- It is still possible to stop PLC positioning movements by resetting markers M4120 to M4128.
- Positioning movements of axes 10 to 14 can be stopped only with this module.

Call:

PS B/W/D/K <Axis>

PS B/W/D/K <Mode>

Reserved: 0 transferred

CM 9224

PL B/W/D <Error>

0: Positioning is canceled

1: Invalid axis was programmed

2: This axis is not positioned by the PLC

Error code:

Marker	Value	Meaning
M4203	0	PLC positioning was stopped
	1	Error code in W1022
W1022	2	Invalid axis (invalid axis number, slave axis, auxiliary axis, or uncontrolled axis)
	9	Axis is not being positioned by the PLC

Module 9227 Position auxiliary axes and NC axes

The module starts the positioning of an NC axis (PLC positioning like with Module 9221) or the positioning of an auxiliary axis (like with PLC Module 9120). With Module 9227, acceleration and jerk can be programmed in addition to the target position and feed rate parameters. During PLC positioning movements via PLC Module 9227, bit 1 in the Mode parameter (software limit switches active) is now also effective on auxiliary axes.

Further modes can be programmed for positioning NC axes (like with Module 9221):

- Rapid-traverse feed rate (only for operating panels with rapid traverse override)
- Active limit switch interrogations
- Collision monitoring deactivated

See also documentation for PLC Modules 9120 and 9221

Conditions:

- The machine parameter configuration of an axis to be started with Module 9227 determines whether the positioning movement of an auxiliary axis or an NC axis is started.

Positioning of NC axes (PLC positioning):

- The same constraints apply as for starting a PLC positioning movement with Marker 4120, etc. In particular, the module may only be called if no program is running, or if an M/G/S/T/T2/Q strobe is pending. No axis direction key may be pressed in the Manual operating mode. The entered positions are referenced to the machine datum. For modulo rotary axes (MP810), positioning is by the shortest path.
- The NC axes can be started by setting the activation markers 4120 to 4128 or with the module call. The module call has priority.
- When the module is called, the corresponding activation marker is set automatically.
- The PLC positioning of this axis can be interrupted by resetting an activation marker.
- If you wish to change a parameter (e.g. target position, feed rate) of a positioning command already in progress, you must first abort positioning, then change the parameter and start again.
- A simultaneous PLC positioning movement of several axes is interpolated. If you start an additional axis while already positioning another, the first movement is aborted, and then all the programmed axes (e.g. X, Y and Z) are positioned together.
- As soon as a PLC positioning with rapid traverse is active (bit 2 is set), all active PLC positioning movements are at rapid traverse, and instead of the feed-rate override the rapid-traverse override is effective.
- A deactivation of collision monitoring (bit 3) only has an effect, if all axes positioned by the PLC at the same time were started with collision monitoring inactive.
- The deactivation of collision monitoring (bit 3) for the PLC positioning movement does not affect the status information provided by Module 9064 (collision monitoring status). With active DCM the module still reads "Monitoring active".
- Error code 7 not possible.



Positioning auxiliary axes:

- Axes with automatic reduction (modulo value in MP810.x) are always moved to the target position in the direction of the shortest traverse, unless the target position was given as an incremental value.
- The axis must be stationary. Any positioning movement must be interrupted beforehand with Module 9121.
- The feed-rate override is not offset.
- If the axis was in the "search for reference mark" state before, this state is canceled. The positioning movement always starts from the current counter value.
- If Modules 9120, 9227 (Start Auxiliary Axis), 9121 (Stop Auxiliary Axis) and 9122 (Pass over Reference in Auxiliary Axis) are called several times for the same axis during a PLC scan, only the latest command is followed.
- The system checks for limit switch overshoot!
- Error codes 3 and 4 not possible

Call:

PS B/W/D/K <Axis>

PS B/W/D/K <Target position/Increment>
in [0.0001 mm], ref system

PS B/W/D/K <Feed rate>
in [mm/min]

PS B/W/D/K <Acceleration>
in [mm/s²]
0: Value from MP1060 is used

PS B/W/D/K <Jerk>
in [mm/s³]
0: Value from MP1088 is used

PS B/W/D/K <Mode>
NC and auxiliary axes:
Bit 0 = 0: Absolute positioning
Bit 0 = 1: Incremental positioning
Bit 1 = 1, software limit switch active (now also for auxiliary axes)

Only for NC axes,
to be programmed for PLC axes = 0:
Bit 2 = 1: Rapid traverse override effective
Bit 3 = 1: DCM collision monitoring is deactivated

CM 9227

PL B/W/D <Status>
0: Function performed
1: Invalid axis was programmed
2: Invalid mode, feed rate, acceleration or jerk programmed
3: Axis has not traversed the reference mark
4: Program run, no NC-strobe output
5: Axis is not in a closed loop (Module 9155)
6: Axis is already being positioned
7: Absolute position is outside of modulo range (W1022 = 2)
(error code possible only for auxiliary axes)

Error code:

Marker	Value	Meaning
M4203	0	Function performed
	1	Error code in W1022
W1022	1	Invalid axis programmed
	2	Invalid mode, feed rate, acceleration or jerk programmed, axis is in an open loop (Module 9155) or absolute position is outside of modulo range
	9	Axis is already being positioned
	21	Program run, no NC-strobe output
	33	Axis has not traversed the reference mark

PLC positioning through markers and words

To ensure compatibility, a PLC positioning command is permissible for axes 1 to 9 with M4120 to M4128, D528 to D544 and W560 to W568.



Attention

Software limit switches are ignored!

Programming:

- ▶ Enter the target position in the double words D528 to D544 in the unit [0.0001 mm].
- ▶ Enter the feed rate in words W560 to W568 [mm/min].
- ▶ To start the PLC positioning movement: Set markers M4120 to M4124 for the desired axis.

		Set	Reset
D528-544	Target position for PLC positioning	PLC	PLC
W560-568	Feed rate for PLC positioning	PLC	PLC
M4120	PLC positioning axis 1 active	NC/PLC	NC/PLC
M4121	PLC positioning axis 2 active	NC/PLC	NC/PLC
M4122	PLC positioning axis 3 active	NC/PLC	NC/PLC
M4123	PLC positioning axis 4 active	NC/PLC	NC/PLC
M4124	PLC positioning axis 5 active	NC/PLC	NC/PLC
M4125	PLC positioning axis 6 active	NC/PLC	NC/PLC
M4126	PLC positioning axis 7 active	NC/PLC	NC/PLC
M4127	PLC positioning axis 8 active	NC/PLC	NC/PLC
M4128	PLC positioning axis 9 active	NC/PLC	NC/PLC



6.4 Axis-error compensation

The iTNC can compensate the following mechanical axis errors:

- Backlash
- Linear axis errors
- Nonlinear axis errors
- Thermal expansion
- Reversal spikes during circular movements
- Friction, stick-slip
- Torsion compensation

Per axis you can activate either the linear or the nonlinear axis error compensation. All other types of compensation are nonexclusive.

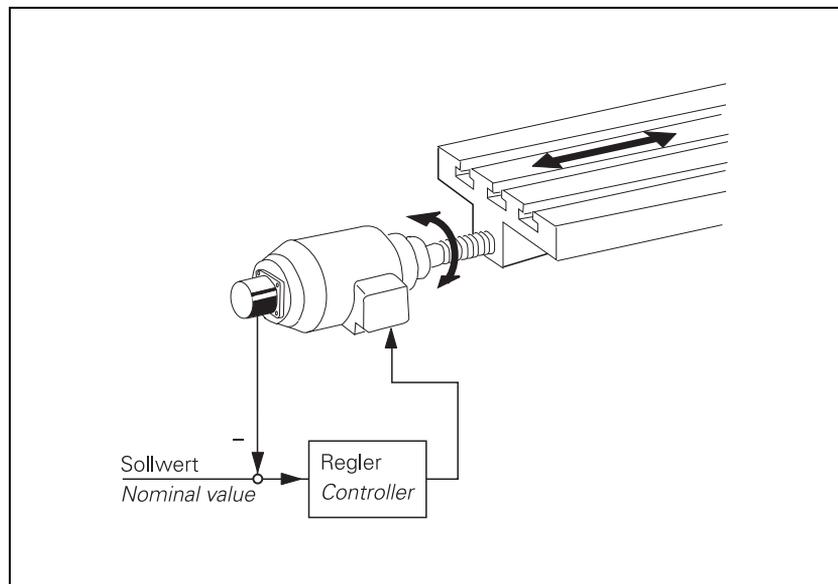
6.4.1 Backlash compensation

Cause outside of the control loop

During a reversal in axis direction, there is often a little play between the rotary encoder and table. This play is referred to as backlash.

Positive backlash: The rotary encoder reading is ahead of the table. The table traverse is too short.

Negative backlash: The rotary encoder reading is behind the table. The table traverse is too long.

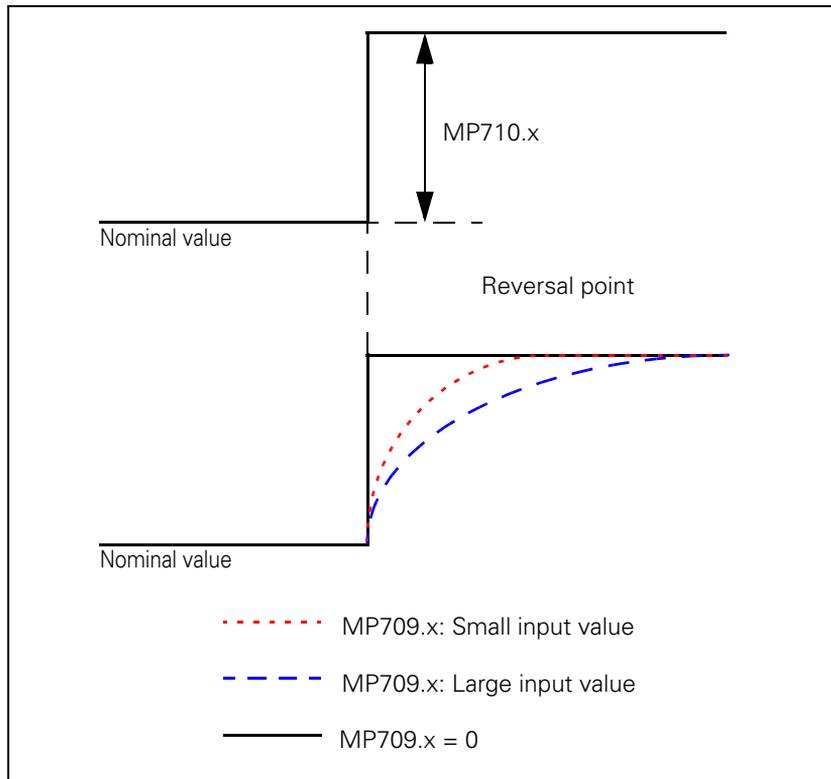


Compensation:

- ▶ Enter the backlash in MP710.x.
- ▶ In MP709.x, enter the time in which the distance to be compensated should be traversed.

The value of the backlash is added to the position value at every reversal of direction (even if it results from a nonlinear axis-error compensation, for example) and corrected by the position controller. The value of the k_V factor therefore influences the settling time for backlash compensation.

In special cases, such as long ball screws with narrow diameters, additional tension in the system can lead to undesirable dynamic side effects. In order to compensate for this tension at the reversal point, enter in MP709.x the time in which the distance to be compensated (from MP710.x) should be traversed.



Acceleration-dependent backlash compensation

In the backlash compensation, the time in which the distance to be compensated should be traversed is entered in MP709.x. The time entered is a fixed value and is used independently of axis acceleration.

Acceleration-dependent backlash compensation that is entered in machine parameter MP708.x is a different feature. It reduces the following error after direction reversal that is independent of speed, acceleration and the traversed contour path. In acceleration-dependent backlash compensation, you define a distance within which the backlash is compensated. The resulting duration for backlash compensation therefore depends on the current acceleration. Backlash compensation takes longer at low than at high acceleration.

In the machine parameter MP708.x, you enter the distance within which the backlash is to be compensated. You also need to enter the backlash in MP710.x or in the compensation table for variable backlash. In MP709.x, you need to enter the value 0 to activate acceleration-dependent backlash compensation.

MP708.x is adjusted by incrementally changing the machine parameter value in the circular interpolation test (reversal spikes).

If you are using a non-linear axis-error compensation table or KinematicsComp, you should use machine parameter MP709.x instead of MP708.x.

The compensation values for backlash and static friction can be calculated dynamically depending on the acceleration, or by using fixed time constants. The new parameter MP2690 can be used to prevent the compensation values from changing too rapidly at very high acceleration. A minimum time constant is entered to limit the bandwidth of the change of the compensation values.

Acceleration-dependent backlash compensation

- ▶ Enter the backlash in MP710.x.
- ▶ Enter the input value 0 in MP709.x.
- ▶ In MP708.x, enter the traverse distance within which the backlash is to be compensated.

MP710.x Backlash compensation

Input: -9.9999 to +9.9999 [mm] or [°]

MP709.x Time constant for backlash compensation

Input: 0: Previous behavior of MP710.x or MP708.x is active
1 to 1000 [ms]

MP708.x Traverse distance for acceleration-dependent backlash compensation

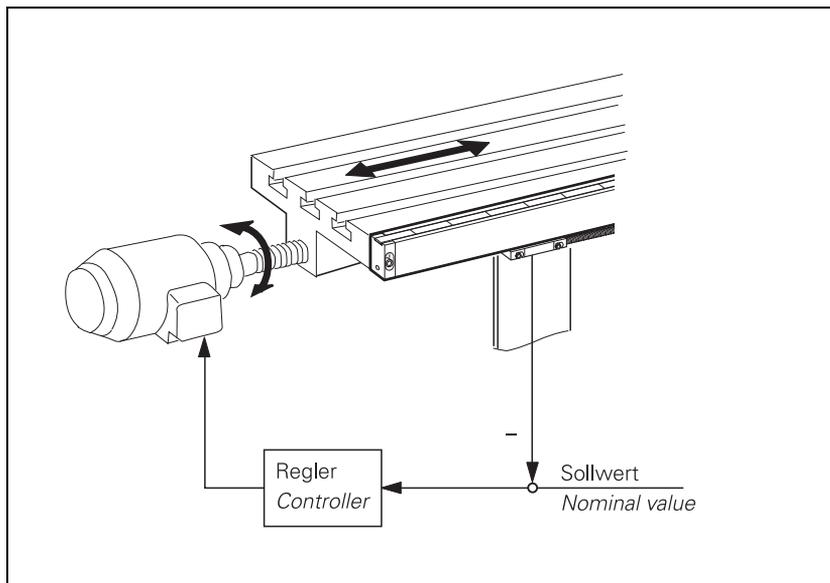
Input: 0: Function inactive
0.0001 to 1.000 [mm]

MP2690.x Band limitation of the compensation values

Input: 0.0001 to 1.0000 [seconds]
0 = Not active

Cause within the control loop

If axis movement is measured with a linear encoder, the iTNC can compensate the play between the motor and the table. At the same time, the reversal spikes during circular movements are compensated: machine parameters MP715 to MP716 for "Compensation of reversal spikes" are **not** necessary.



Compensation:

- ▶ In MP750.x, enter the reversal error in mm.
- ▶ In MP752.x, enter the time in which the distance to be compensated should be traversed.

MP750.x Backlash in axes 1 to 9

Input: -9.9999 to +9.9999 [mm] or [°]

MP752.x Compensation time for reversal error

Input: 0 to 1000 [ms]

Example:

MP750.x: 0.03 mm

MP752.x: 15 ms

For every change in direction, a nominal speed command signal is output for 15 ms, which corresponds to a feed rate of 120 mm/min:

$$\frac{0.03 \text{ mm}}{15 \text{ ms}} = 120 \text{ mm/min}$$

The reversal error is compensated dynamically depending on the acceleration. The reversal error is compensated more rapidly at high acceleration, and more slowly at low acceleration. MP708 is used to define the distance within which the complete reversal error is to be compensated. This dynamic compensation is active only if no time constant is defined in MP752. The bandwidth of the compensation value can be limited in MP2690.

6.4.2 Linear axis error compensation



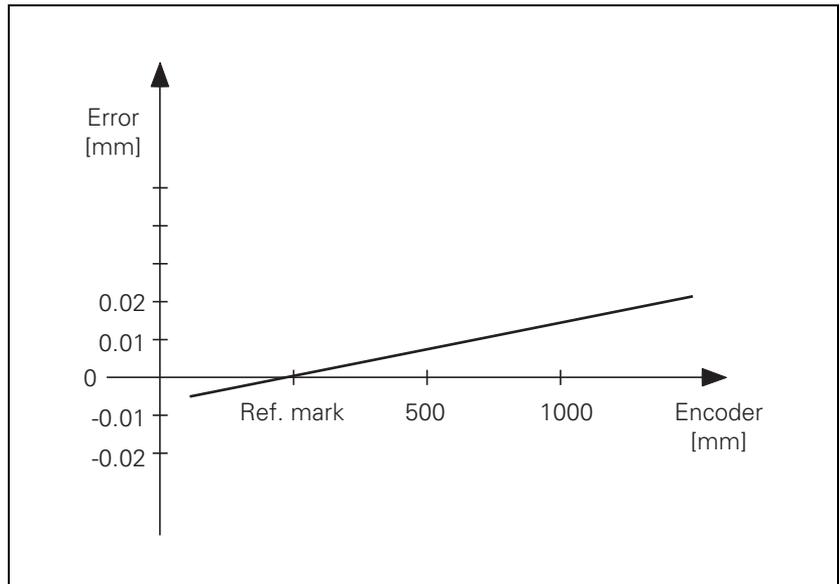
Note

Linear axis error compensation is not available for rotary axes!

For every linear axis you can compensate a linear axis error.

Positive linear axis error: The table moves too far.

Negative linear axis error: The table moves short.



Compensation:

- ▶ In MP720, enter the axis error in [mm/m].
- ▶ With MP730, activate the linear axis error compensation.

MP720.x Linear axis error compensation

Input: -1.000 to +1.000 [mm/m]

MP730 Selection of linear/nonlinear axis error compensation

Format: %xxxxxxxxxxxxxxxxxxxx

Input: 0 to 17 represent axes 1 to 18

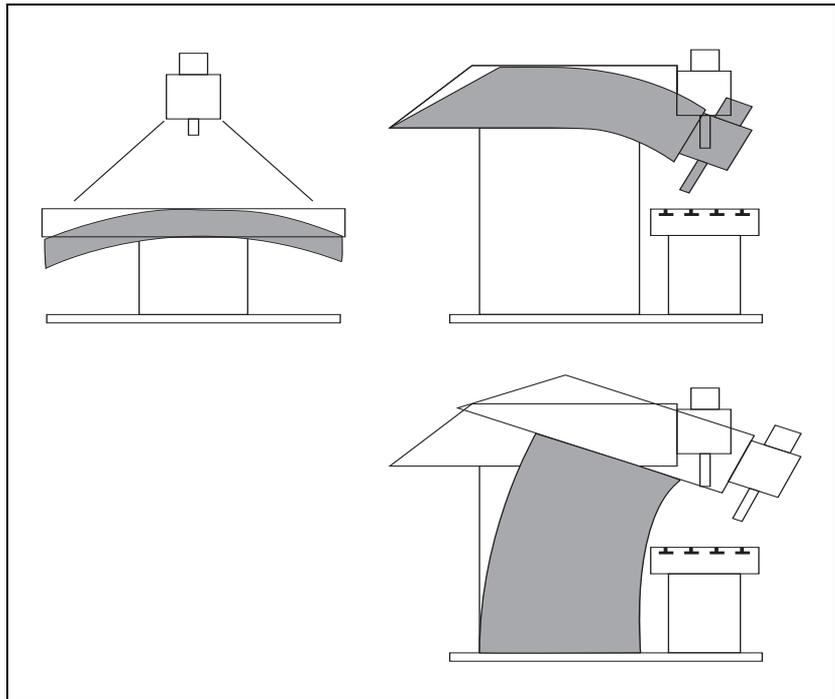
0: Linear axis error compensation

1: Nonlinear axis error compensation



6.4.3 Nonlinear axis error compensation

Errors in machine geometry (e.g. an error in one axis caused by the sagging of another axis) or external influences (e.g. temperature) can cause nonlinear axis errors. These graphics show typical nonlinear axis errors:



The best way to measure nonlinear axis error is with a comparator measuring system such as the HEIDENHAIN VM 101.



Note

The iTNC can compensate screw-pitch error and axis sag simultaneously.

Nonlinear axis error compensation is also effective for an open loop. In this case the compensation value is considered when the control loop is closed.

Entering compensation values

Enter the compensation values into the *.COM tables:

- ▶ Ascertain the error trace with a comparator measuring system.
- ▶ To create a compensation-value table: Press the MOD key and enter the code number 807667; for each axis that is to be compensated for, use the program manager to create a compensation-value table with the name *.COM.
- ▶ Using soft keys (e.g. **1 OFF/ON**), activate columns in the compensation value table only for the axes whose positions affect the error of the compensated axis.
- ▶ Begin your entry with the soft key **HEAD LINE**: Enter the datum for the compensation values as a distance from the machine datum (MP960.x).
- ▶ Enter a value for the spacing of the compensation points as a power to the base of 2. The input value is ± 999.999999 (six decimal places, 1 nm).
- ▶ Exit the header by pressing END.
- ▶ With the soft key **APPEND N LINES**, enter the number of compensation points:
 - Up to 4000 lines
 - Maximum of 24 columns in all active compensation value tables
 - Total maximum of $24 \cdot 4000 = 96000$ compensation points
- ▶ To enter compensation values: Enter only the break points of the error trace. The iTNC interpolates linearly between the break points.

If another *.COM file is to be used for the negative direction of traverse, the file name must end with a minus character (-). The difference between two direction-sensitive compensation-value tables is corrected immediately at a reversal of direction.

Example:

Entry in the *.CMA file: **Axis_X**. The compensation value table Axis_X.COM is used. If the compensation value table Axis_X-.COM exists, it will be used for the negative traverse direction.



Example

The following dependencies apply for axes 2 = Y and 3 = Z:

- Ball screw pitch error in Z and Y: Z = F(Z) and Y = F(Y)
- Axis sag in Z depending on Y
- Range of traverse: Z = 800 mm, Y = 500 mm
- Datum for compensation values: Z = -200 mm, Y = -90 mm
- Desired spacing of compensation points: 7 mm

Number of compensation points:

$$\frac{500 \text{ mm}}{7 \text{ mm}} = 71 \text{ Compensation points in Y}$$

$$\frac{800 \text{ mm}}{7 \text{ mm}} = 114 \text{ Compensation points in Z}$$

<div style="border: 1px solid black; padding: 5px;"> <p>Manual operation Compensation value table Compensation value?</p> <table border="1" style="width: 100%; border-collapse: collapse; font-family: monospace;"> <thead> <tr> <th>NR</th> <th>F(Y) = Z COH</th> <th>F(Z) = Y REF (Z)</th> <th>DATUM: -90</th> <th>DISTANCE: +7</th> </tr> </thead> <tbody> <tr><td>0</td><td>-90</td><td>03</td><td></td><td>-0.01</td></tr> <tr><td>1</td><td>-93</td><td>+0</td><td></td><td>+0</td></tr> <tr><td>2</td><td>-76</td><td>-0.005</td><td></td><td>-0.02</td></tr> <tr><td>3</td><td>-69</td><td>+0</td><td></td><td>-0.02</td></tr> <tr><td>4</td><td>-62</td><td>+0</td><td></td><td>-0.025</td></tr> <tr><td>5</td><td>-55</td><td>+0</td><td></td><td>-0.02</td></tr> <tr><td>6</td><td>-48</td><td>+0</td><td></td><td>-0.02</td></tr> <tr><td>7</td><td>-41</td><td>+0</td><td></td><td>+0</td></tr> <tr><td>8</td><td>-34</td><td>+0</td><td></td><td>+0</td></tr> <tr><td>9</td><td>-27</td><td>-0.01</td><td></td><td>+0</td></tr> <tr><td>10</td><td>-20</td><td>+1</td><td></td><td>-0.035</td></tr> <tr><td>11</td><td>-13</td><td>+0</td><td></td><td>+0</td></tr> <tr><td>12</td><td>-6</td><td>+0</td><td></td><td>-0.035</td></tr> <tr><td>13</td><td>+1</td><td>+0</td><td></td><td>+0</td></tr> <tr><td>14</td><td>+8</td><td>+0</td><td></td><td>+0</td></tr> <tr><td>15</td><td>+15</td><td>+0</td><td></td><td>+0</td></tr> </tbody> </table> <div style="display: flex; justify-content: space-between; margin-top: 5px;"> BEGIN ↑ END ↓ PAGE ↑ PAGE ↓ INSERT LINE DELETE LINE NEXT LINE HEAD LINE </div> </div>	NR	F(Y) = Z COH	F(Z) = Y REF (Z)	DATUM: -90	DISTANCE: +7	0	-90	03		-0.01	1	-93	+0		+0	2	-76	-0.005		-0.02	3	-69	+0		-0.02	4	-62	+0		-0.025	5	-55	+0		-0.02	6	-48	+0		-0.02	7	-41	+0		+0	8	-34	+0		+0	9	-27	-0.01		+0	10	-20	+1		-0.035	11	-13	+0		+0	12	-6	+0		-0.035	13	+1	+0		+0	14	+8	+0		+0	15	+15	+0		+0	<p>Axis Y: Ball screw-pitch error in column 2 = F(), Sag error in column 3 = F()</p>
NR	F(Y) = Z COH	F(Z) = Y REF (Z)	DATUM: -90	DISTANCE: +7																																																																																		
0	-90	03		-0.01																																																																																		
1	-93	+0		+0																																																																																		
2	-76	-0.005		-0.02																																																																																		
3	-69	+0		-0.02																																																																																		
4	-62	+0		-0.025																																																																																		
5	-55	+0		-0.02																																																																																		
6	-48	+0		-0.02																																																																																		
7	-41	+0		+0																																																																																		
8	-34	+0		+0																																																																																		
9	-27	-0.01		+0																																																																																		
10	-20	+1		-0.035																																																																																		
11	-13	+0		+0																																																																																		
12	-6	+0		-0.035																																																																																		
13	+1	+0		+0																																																																																		
14	+8	+0		+0																																																																																		
15	+15	+0		+0																																																																																		
<div style="border: 1px solid black; padding: 5px;"> <p>Manual operation Compensation value table Compensation value?</p> <table border="1" style="width: 100%; border-collapse: collapse; font-family: monospace;"> <thead> <tr> <th>NR</th> <th>F(Z) = Z COH</th> <th>F(Y) = Y REF (Z)</th> <th>DATUM: -200</th> <th>DISTANCE: +7</th> </tr> </thead> <tbody> <tr><td>0</td><td>-200</td><td>03</td><td></td><td></td></tr> <tr><td>1</td><td>-193</td><td>+0</td><td></td><td></td></tr> <tr><td>2</td><td>-186</td><td>+0</td><td></td><td></td></tr> <tr><td>3</td><td>-179</td><td>+0</td><td></td><td></td></tr> <tr><td>4</td><td>-172</td><td>+0</td><td></td><td></td></tr> <tr><td>5</td><td>-165</td><td>+0.005</td><td></td><td></td></tr> <tr><td>6</td><td>-158</td><td>+0</td><td></td><td></td></tr> <tr><td>7</td><td>-151</td><td>+0.005</td><td></td><td></td></tr> <tr><td>8</td><td>-144</td><td>+0.007</td><td></td><td></td></tr> <tr><td>9</td><td>-137</td><td>+0</td><td></td><td></td></tr> <tr><td>10</td><td>-130</td><td>+0</td><td></td><td></td></tr> <tr><td>11</td><td>-123</td><td>+0.001</td><td></td><td></td></tr> <tr><td>12</td><td>-116</td><td>+0</td><td></td><td></td></tr> <tr><td>13</td><td>-109</td><td>+0</td><td></td><td></td></tr> <tr><td>14</td><td>-102</td><td>+0</td><td></td><td></td></tr> <tr><td>15</td><td>-95</td><td>+0</td><td></td><td></td></tr> </tbody> </table> <div style="display: flex; justify-content: space-between; margin-top: 5px;"> BEGIN ↑ END ↓ PAGE ↑ PAGE ↓ INSERT LINE DELETE LINE NEXT LINE HEAD LINE </div> </div>	NR	F(Z) = Z COH	F(Y) = Y REF (Z)	DATUM: -200	DISTANCE: +7	0	-200	03			1	-193	+0			2	-186	+0			3	-179	+0			4	-172	+0			5	-165	+0.005			6	-158	+0			7	-151	+0.005			8	-144	+0.007			9	-137	+0			10	-130	+0			11	-123	+0.001			12	-116	+0			13	-109	+0			14	-102	+0			15	-95	+0			<p>Axis Z: Ball screw pitch error in column 3 = F()</p>
NR	F(Z) = Z COH	F(Y) = Y REF (Z)	DATUM: -200	DISTANCE: +7																																																																																		
0	-200	03																																																																																				
1	-193	+0																																																																																				
2	-186	+0																																																																																				
3	-179	+0																																																																																				
4	-172	+0																																																																																				
5	-165	+0.005																																																																																				
6	-158	+0																																																																																				
7	-151	+0.005																																																																																				
8	-144	+0.007																																																																																				
9	-137	+0																																																																																				
10	-130	+0																																																																																				
11	-123	+0.001																																																																																				
12	-116	+0																																																																																				
13	-109	+0																																																																																				
14	-102	+0																																																																																				
15	-95	+0																																																																																				



Activating error compensation

The appropriate machine parameter must be set for nonlinear axis error compensation, and the compensation value table must be registered in a configuration file:

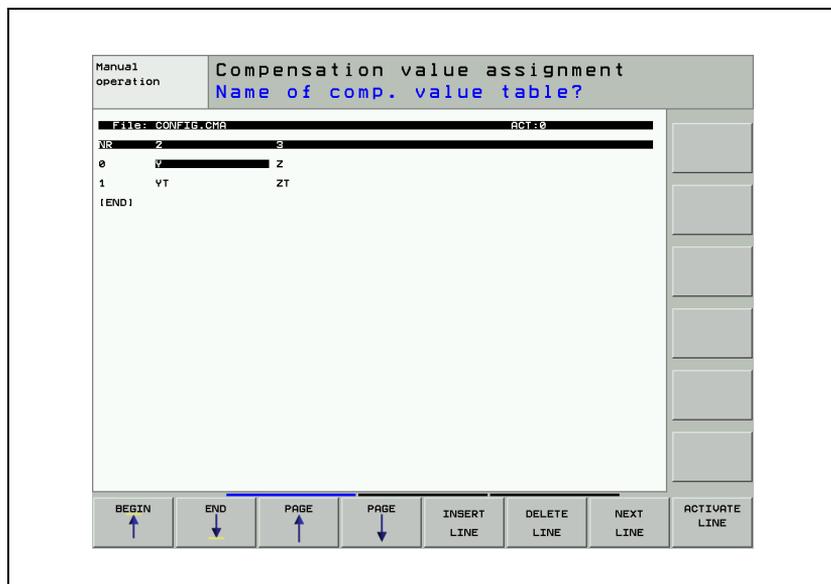
- ▶ With MP730, activate for each axis the nonlinear axis error compensation.
- ▶ To create a configuration file: Press the MOD key and enter the code number 807667; with the program manager, create a configuration file with the extension .CMA.
- ▶ Use soft keys (e.g. **1 OFF/ON**) to activate columns for the axes for which you have created compensation value tables.
- ▶ Enter the compensation value table: You can assign more than one compensation value table to each axis; however, only one table at a time can be active. Enter the file names of the compensation value tables in the respective lines. You can select the active line either with the soft key **ACTIVATE LINE** or with Module 9095. With Module 9035 you can interrogate the active line.
- ▶ Enter the complete name of the configuration file of the *.CMA type in the system file OEM.SYS with the **TABCMA =** command.

Example

Entry in the configuration file for axes 2 = Y and 3 = Z:

Compensation value tables valid for 20 °C = Y.COM and Z.COM

Compensation value tables valid for 35 °C = YT.COM and ZT.COM



MP730 Selection of linear/nonlinear axis error compensation

Format: %xxxxxxxxxxxxxxxxxxxx
 Input: 0 to 17 represent axes 1 to 18
 0: Linear axis error compensation
 1: Nonlinear axis error compensation

Module 9095 Select active line in configuration file

Call:
 PS B/W/D/K <Active line>
 CM 9095
 PL B/W/D <Error code>
 0: No error
 1: Entered line does not exist
 2: Compensation value table does not exist
 3: Compensation value table > 256 entries
 4: Maximum total number of compensation points exceeded
 5: Too many compensation value tables
 6: CMA file does not exist
 7: Call was not from a submit job
 8: Call during running program without strobe
 10: CMA file is protected

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Indicated line does not exist, the compensation-value table has > 256 entries, or maximum number of compensation points exceeded
	20	Call was not in a submit or spawn job
	21	Call during already started program without strobe
	36	.CMA file is protected
	56	Compensation-value table does not exist



Module 9035 Read status information

Call:

PS B/W/D/K <19>

CM 9035

PL B/W/D <Active line number>
0: Line number
-1: No .CMA file active

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Status information invalid
	20	Call was not in a submit or spawn job

Special case: Rotary axis

With MP732, the non-linear axis-error compensation can be projected onto the actual traverse range for a specific axis, not only from 0° to 360°. This means that traverse ranges from -90° to +90°, for example, can be entered for rotary axes.

MP732

Nonlinear axis-error compensation for rotary axes

Format:

%xxxxxxxxxxxxxxxx

Input:

Bits 0 to 17 represent axes 1 to 18

0: Not active (conventional compensation, only 0° to 360°)

1: Active (mapped to traverse range)

6.4.4 Compensation of thermal expansion

To compensate thermal expansion, exact measurements of machine thermal behavior as a function of temperature (e.g., the center of axis expansion, the amount of the expansion) are necessary.

The temperatures measured by the Pt100 thermistors are copied into the PLC words from IOconfig. Since the thermal expansion of the axes is largely proportional to the temperature: you can directly determine the amount of expansion by multiplying the temperature value by a certain factor.

Compensation:

- ▶ Transfer the distance to be compensated to Module 9231. At the same time, "lag tracking" becomes active. This means that the actual position is offset by a certain value per PLC cycle until the complete value is compensated.
- ▶ In MP4070, enter the value for the offset per PLC cycle.

For gantry axes, the compensation value must be transferred separately for each axis.

Heat compensation when using tilting axes is defined through machine parameters or via the **TEMPCOMP** column of the kinematics table. See "Temperature compensation with tilting axes" on page 741.

The actual value display does not change during compensation. As an alternative, for axes 1 to 5 you can enter the value to be corrected in W576 to W584.

MP4070 Compensation amount per PLC cycle for lagged-tracking axis error compensation

Input: 0.0001 to 0.5000 [mm]

	Set	Reset
W576 - 584 Lag-tracking axis-error compensation	PLC	PLC
For axes 1 to 5		
Input: -32768 to +32767 [1/10 µm]		

Module 9231 Compensation of thermal expansion

With Module 9231, thermal expansion can be compensated by transferring the axis number and a compensation value.

Call:

PS B/W/D/K <Axis>

Axis 0 to 17

PS B/W/D/K <Compensation value>

Range: -30000 to +30000 [1/10 µm]

CM 9231

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Invalid axis number
	2	Invalid compensation value
	24	The module was called in a spawn job or submit job



6.4.5 Compensation of reversal spikes during circular traverse

With the iTNC 530 HSCI you compensate the reversal spikes through a friction compensation in the area of the speed controller (MP2610.x to MP2620.x). "Compensation of sliding friction (only for digital axes)" on page 662.

6.4.6 Compensation of stick-slip friction

On guideways with high stick-slip friction (static friction), a following error can occur at low feed rates during operation with velocity feedforward control. This error can be compensated by the iTNC. You can measure following error by using, for example, the integrated oscilloscope of the iTNC.

With the iTNC 530 HSCI you compensate the reversal spikes with the machine parameters MP2610.x to MP2620.x. "Stick-slip friction compensation at quadrant transitions" on page 1106.

Limiting the integral factor

In machines with very high static friction, a position deviation at standstill can lead to the accumulation of a very high integral factor. This can lead to a jump in the position value when the axis "tears loose." In such cases you can limit the integral-action component of the speed controller with MP2512.x.

The limitation is effective as soon as the axis is in position (W1026/W1104). If high utilization occurs when an axis is at standstill (e.g. due to static friction), the current is also limited.

MP2512.x Limit of integral factor of the speed controller

Input: 0.000 to 30.000 [s] (realistically: 0.1 to 2.0)

6.4.7 Compensation of sliding friction (only for digital axes)

Sliding friction is compensated within the range of the speed controller:

- ▶ With the integrated oscilloscope of the iTNC, define the nominal current value (I NOMINAL) at a very low speed of approx. 10 min^{-1} .
- ▶ Enter the value for current in MP2610.x. At every change in direction, this amount is fed forward to the speed controller to compensate the sliding friction at low speeds.
- ▶ Measure the nominal value for current (I NOMINAL) at rated speed and enter it in MP2620.x. Depending on the nominal speed value, a certain current is fed forward to the speed controller and causes a sliding friction that depends on the speed.

When the traverse direction is reversed at high feed rates, the sliding friction might be overcompensated. In a circular interpolation test, such overcompensation appears in the form of reversal spikes that jut inward. With MP2612.x you can prevent overcompensation by delaying the compensation.

MP2610.x Friction compensation at low speeds (effective only with velocity feedforward control)

Input: 0 to 100.0000 [A]
0: No friction compensation (or axis is analog)

MP2612.x Delay of the friction compensation (effective only with velocity feedforward control)

Input: 0.0000 to 1.0000 [s] (typically: 0.015 s)
0: No friction compensation (or axis is analog)

MP2620.x Friction compensation at rated speed

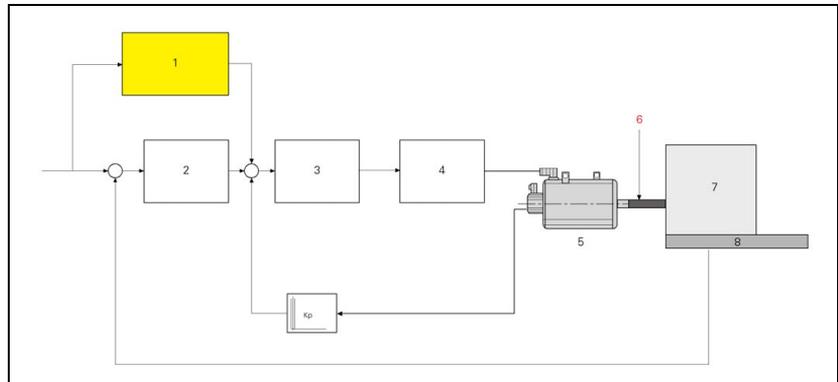
Input: 0 to 100.000 [A]
0: No friction compensation (or axis is analog)



6.4.8 Torsion compensation

With MP2640.x you can perform a torsion compensation between the position and speed measuring systems. The torsion compensation regulates the difference in position that results from the elasticity between the motor (rotary encoder) and the position measuring system. An additional torsion motion is added to the speed controller. In MP 2640.x you enter a drive-specific factor for the elastic constant. HEIDENHAIN recommends using MP 2640 as a replacement for MP 2606, since MP 2640 can be used to take additional effects into account.

The block diagram shows how the torsion compensation works:



- 1: Torsion compensation
- 2: Position controller
- 3: Speed controller
- 4: Current controller – power module
- 5: Motor
- 6: Elastic coupling
- 7: Machine
- 8: Linear encoder

Adjustment of MP2640.x is only worthwhile once all other feedforward-control parameters MP2600.x to MP2630.x have been adjusted. Furthermore, stick-slip friction compensation via MP2640.x should not be used if stick-slip friction compensation only takes place via MP2612.x (MP2614.x=0).

Torsion compensation via machine parameter MP2640.x is now as of 340 49x-06 also added in machines without position encoder if a value that is not equal to 0 is defined in MP2640.x. The adjustment remains unchanged, i.e. a position encoder must be used. However, the compensation values can now also be used for dimensionally identical machines without position encoder. Moreover, the compensation value is filtered before it is added. This improves the behavior of the axis after direction reversal, for example.

MP2640.x Torsion compensation
Input: 0.001 to 100.000 [$\mu\text{m}/\text{A}$]
 0: Not active

6.4.9 KinematicsComp

Introduction

Increasingly stringent requirements on workpiece tolerances constantly increase demands placed on a machine tool. However, machine tools are inevitably also subject to errors arising both from the production process and the machine design. For example, according to ISO 230 each axis can have six types of error. These include not only familiar errors, such as positioning and angular errors, but also pitch, yaw and roll as well as the wobble of a rotary axis.

This means in the end that the more axes a machine has, the larger the number of error sources. The use of mechanical means to cope with these problems requires considerable effort, particularly in the field of 5-axis machining, or if large machines with parallel axes are involved. The axes are also subject to drift, usually resulting from inhomogeneous temperature distribution in the machine components. The drift becomes apparent in the form of both linear movements (translations) and, quite often, rotational movements (rotations).

The new KinematicsComp function (software option 52) of the iTNC 530 enables the machine manufacturer to save a comprehensive description of the machine errors in the control. In the kinematics model, the manufacturer describes the machine's degrees of freedom and the positions of the rotary axes. Until now it had only been possible to define the nominal geometry of the machine. Now the actual behavior of all axes can be integrated in this original kinematics model.

KinematicsComp can even be used to define position-dependent temperature compensation. The required data is supplied by several sensors located at relevant points on the machine.

KinematicsComp is capable of significantly improving the manufacturing and contour accuracy, especially on large machines. The long distances traversed and the large moving masses cause relatively large errors. Because the use of mechanical means to reduce these errors requires a lot of effort, the economic benefits achieved by KinematicsComp are especially significant for these applications.

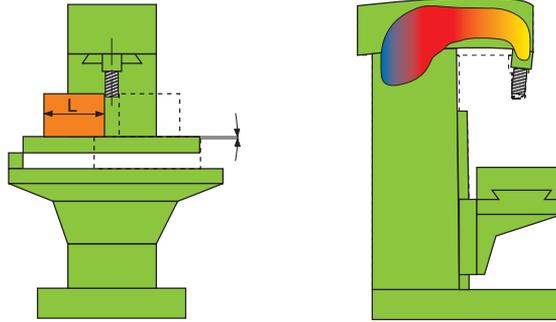
KinematicsComp requires a kinematics model of the machine, which describes the real behavior of the machine as accurately as possible. The ideal kinematics model of the machine must be described, as previously, in the **KEY**, **AXIS** and **COORD** columns in the kinematics table. For KinematicsComp, the kinematics table was extended by the column **COMPTAB**. In the lines of the kinematics table with the '**Trans**' **KEY** entry, constants or links to axis-error compensation tables can be entered in the new column **COMPTAB** in order to describe deviations from the ideal kinematics model.

Summary:

- The KinematicsComp software option 52 makes geometrical compensation of comprehensive axis-dependent error sources possible.
- Only the position of the TCP Tool Center Point is corrected. This prevents undesired movements of the rotary axis.
- Exact determination of the errors places high demands on metrology. Laser interferometers, especially laser tracers, provide very exact measurement of the errors of linear axes.

Error classes

The following error classes can occur and can be compensated for with KinematicsComp:



- Geometry errors (static) – (nonlinear) deviations between machine coordinate system and CNC depend on:
 - Accuracy of measuring systems
 - Straightness, angularity of the machine
 - Accuracy of the kinematics model
 - Axis offset and positioning errors
 - Friction, backlash and Abbé error
- Thermally induced errors (quasi-static) – inhomogeneous temperature distribution in the machine parts:
 - Spindle
 - Feed drives
 - Cooling
 - Symmetry of machine frame

Geometry errors of axes

There are basically two types of error – Component error and position error:

Component errors – deviations from the ideal movement – are identified by three characters (e. g. EAX) according to the standard ISO 230-1. The characters designate position functions of the axis under consideration.

- 1st character (E): Identifier of the component error
- 2nd character (e.g. A): Error direction
- 3rd character (e.g. X): Axis under consideration

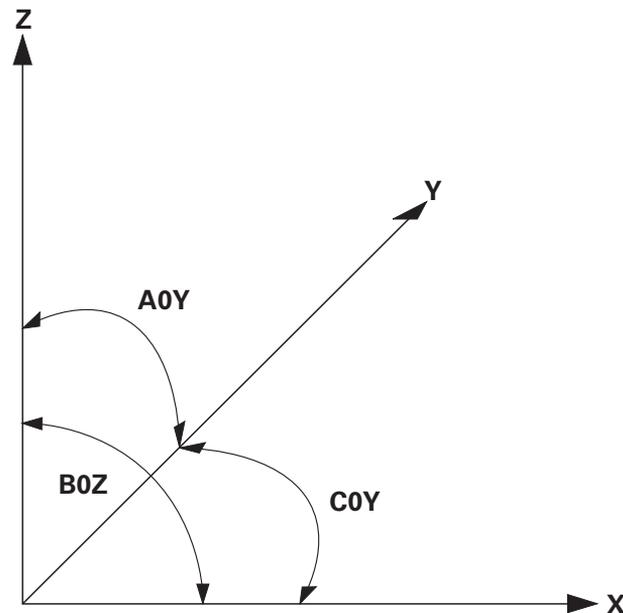
Position errors – deviations from the ideal axis position – are also identified by three characters ((e. g. X0C) according to the standard ISO 230-1. These characters designate constants.

- 1st character (e.g. X): Error direction
- 2nd character (0): Identifier of the position
- 3rd character (e.g. X): Axis under consideration

Angular errors of linear axes

Position error (angle) between linear axes. Example:

- Principal axis X
- 1st minor axis Z
- 2nd minor axis Y



**Component errors
of linear axes**

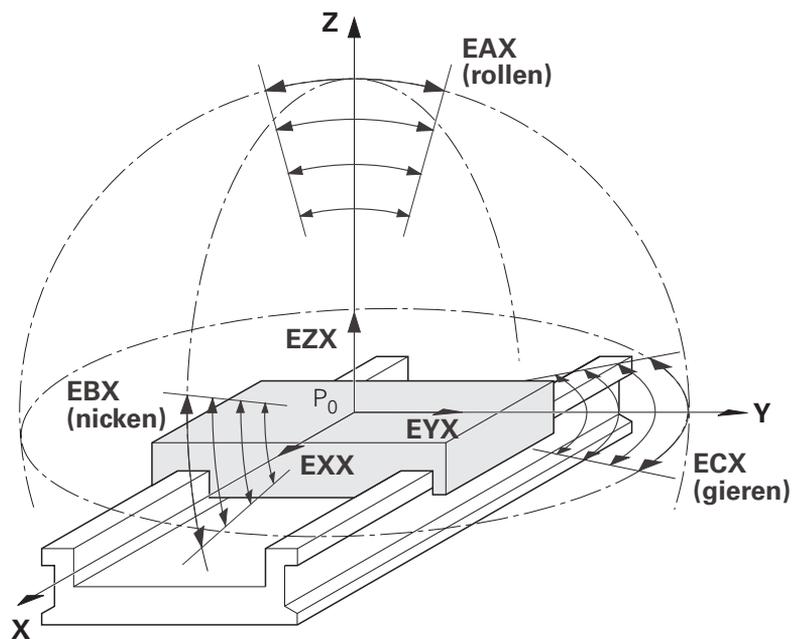
Example: Horizontal motion axis X

Translational deviations:

Designation	Error
E _{XX}	Positioning error
E _{YX}	Straightness error
E _{ZX}	Straightness error

Rotational deviations:

Designation	Error
E _{AX}	Roll
E _{BX}	Pitch
E _{CX}	Yaw



E_{XX} , E_{YX} , E_{ZX} , E_{AX} , E_{BX} and E_{CX} are functions of X.

Errors of rotary axes

Example: Vertical motion axis C

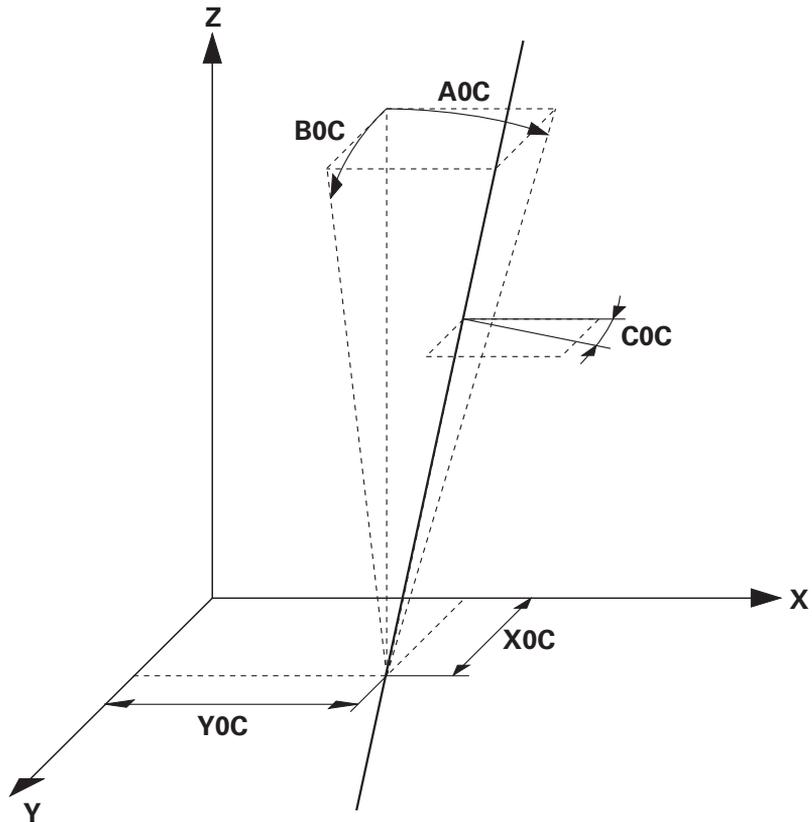
Position errors – Static errors (home position) of rotary axes

Translational deviations:

Designation	Error
X0C	Position error
Y0C	Position error

Rotational deviations:

Designation	Error
A0C	Angular error
B0C	Angular error
C0C	Angular error (offset)



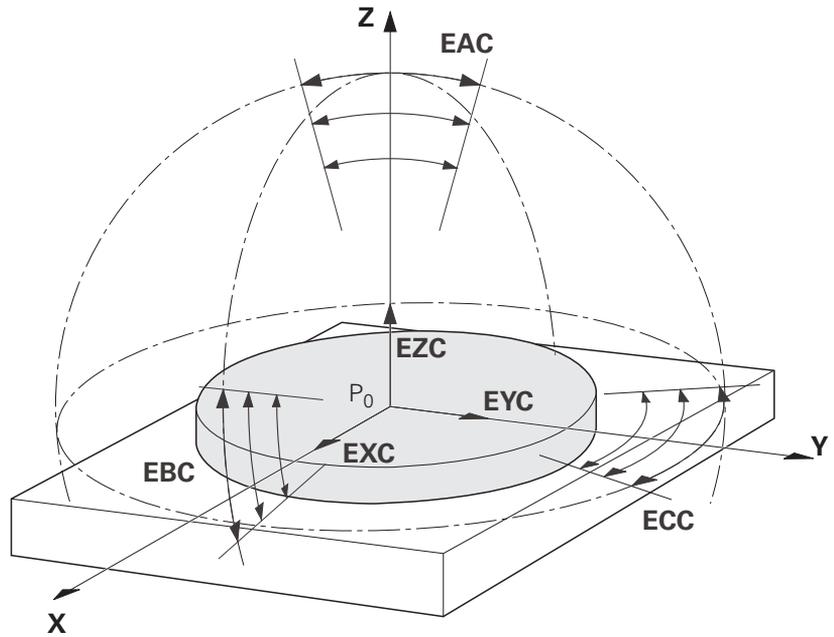
Component errors – Errors during the movement of rotary axes

Translational deviations:

Designation	Error
EXC	Radial error in X
EYC	Radial error in Y
EZC	Axial error

Rotational deviations:

Designation	Error
EAC	Wobble relative to X (roll)
EBC	Wobble relative to Y (pitch)
ECC	Positioning error



EXC, EYC, EZC, EAC, EBC and ECC are functions of C.

Number of errors

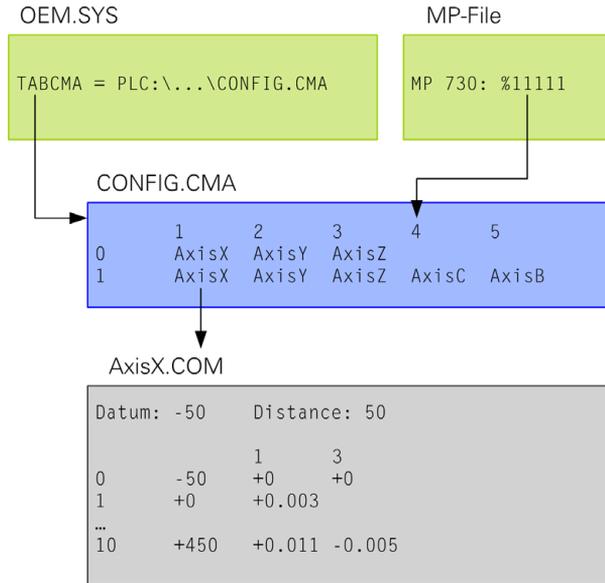
A 5-axis machine with three linear and two rotary axes can have a total of 43 possible errors (6 errors per linear axis, 11 errors per rotary axis, and 3 angular errors of the linear axes). The spindle is not included.

Previous kinematics description

The kinematics model of the machine must always be described in the new table format (columns: KEY, AXIS, COORD, ...). In this kinematics model, the machine axes are described and the positions of the rotary axes are defined. An ideal model is described, in which only the positions of the rotary axes are adapted to the actual conditions.

NR	KEY	AXIS	COORD
	...		
	Trans	X	0.016
	Trans	Z	250.051
	MachAxis	B	
	MachAxis	Y	
	MachAxis	X	
	MachBase		
	MachAxis	Z	
	Trans	X	-298.923
	Trans	Y	200.172
	MachAxis	C	
	...		
	[END]		

The axis errors are compensated via compensation value tables of the *.CMO type. Each individual axis can be used to correct another axis with respect to the respective individual axis. These nonlinear compensation entries make adequate correction of positioning errors possible. If angular errors occur, however, it is only possible to correct the positioning error with respect to a distance.



Enhanced kinematics description

In the enhanced kinematics description, the transformation chain is linked with the nonlinear compensation entries. The geometry errors can be described in the column **COMPTAB**.

- The effective directions are determined by the transformation chain.
- The compensation function corrects the position of the TCP Tool Center Point.
- The compensation rate depends on MP4070.

NR	KEY	AXIS	COORD	COMPTAB
	...			
	Trans	X	0.016	
	Trans	Z	250.051	
	MachAxis	B		
	Trans	A	0	0.0015
	MachAxis	Y		
	MachAxis	X		
	MachBase			
	MachAxis	Z		
	Trans	X	-298.923	
	Trans	Y	200.172	
	MachAxis	C		
	...			
	[END]			

Constants or links to a nonlinear relationship can be entered in the column **COMPTAB**.

Entering a constant

Invariable deviations (not dependent on axis position or temperature) can be entered directly in the column **COMPTAB**. The unit is [mm] or [°].

Example: The B axis is rotated by x in the machine coordinate system (error AOB).

NR	KEY	AXIS	COORD	COMPTAB
	...			
	Trans	X	0.016	
	Trans	Z	250.051	
	MachAxis	B		
	Trans	A	0	0.0015
	MachAxis	Y		
	MachAxis	X		
	MachBase			
	MachAxis	Z		
	Trans	X	-298.923	
	Trans	Y	200.172	
	MachAxis	C		
	...			
	[END]			



Links to axis-error compensation tables

Variable deviations can be described as a function of axis positions (referred to below as compensation function). The compensation function is described in the form of a table, in which compensation points are entered. The path between the individual compensation points is interpolated linearly (corresponds to the behavior of tables for nonlinear axis error compensation). The compensation points can be entered as discrete values (in [mm] or [°], the decimal character is a point), or they can also be defined as indiscrete data through PLC words. If PLC words are used, the compensation points will be updated by the contents of the respective PLC words every second. The contents of the PLC words are interpreted as 1/10000 mm or 1/10000 degrees.

The compensation functions are defined in the tables available for nonlinear axis error compensation (*.COM). To make this possible, the tables were extended by the six columns C1 to C6. The compensation function is assigned to a **Trans** line of the kinematics table by an identifier in the column **COMPTAB**. The first part of the identifier refers to the axis, the second part refers to the column of the error table *.COM belonging to that axis.

Example: **A4C1** means that the compensation function of the 4th axis in column C1 is selected. The compensation value table is assigned to an axis as previously via the active line of the selected *.CMA file.

Each compensation value table can contain a maximum of six compensation functions (= columns C1 to C6), which can be assigned to the **Trans** lines of the kinematics table. Only one compensation value table per machine axis is permissible (as previously).

If an axis position is changed, a new collective compensation value is calculated in the X, Y and Z axes. This compensation value is the result of a re-calculation of the entire kinematics chain, including the current compensation values. The position of the TCP Tool Center Point is corrected by adding the collective compensation value to the axis positions.

Machine parameter MP4070 is used to limit the change rate of the compensation (maximum compensation value per PLC cycle).

Compensation is not shown in the position display of the control. The compensation values can be read with PLC module 9040 or 9041 (transfer value 8). The module supplies the values to be added to the ACTUAL position of the axis. This means that the algebraic sign must be neglected in order to attain the compensation values that are traversed by the machine axes.

The DCM collision monitoring function of the iTNC 530 does not consider the compensation. All collision bodies are enlarged by MP1292 by the value entered there in order to ensure a reliable collision protection.

Example:

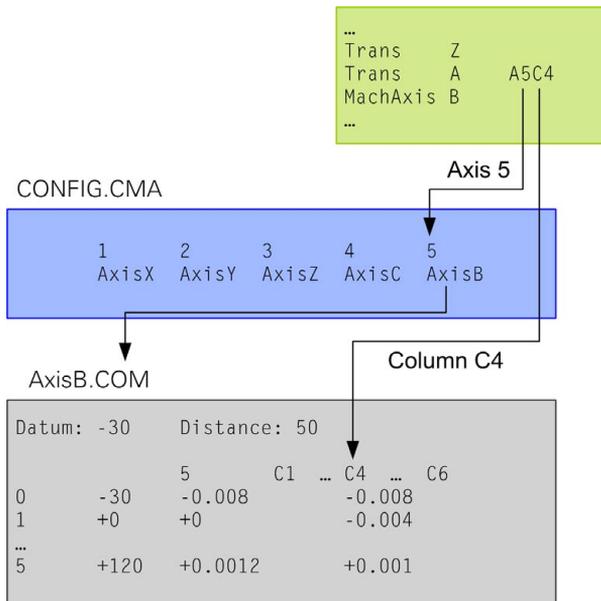
Entry of the link "Dependence on axis 5, compensation column C4": The B axis wobbles relative to X during movement (error EAB)

NR	KEY	AXIS	COORD	COMPTAB
	...			
	Trans	X	0.016	
	Trans	Z	250.051	
	Trans	A	0	A5C4
	MachAxis	B		
	MachAxis	Y		
	MachAxis	X		
	MachBase			
	MachAxis	Z		
	Trans	X	-298.923	
	Trans	Y	200.172	
	MachAxis	C		
	...			
[END]				

The *.COM files have been extended by the columns C1 to C6. These columns are addressed by a corresponding link in the transformation chain.

MP732 must be activated for rotary axes, because the kinematics compensation (C1 to C6) is only applied to the traverse range of the axis.

In this example, there is a link to "A5C4". This links to axis 5 or to column 5 of the CMA file, and to column C4 in the COM file saved there:



Constants or variables (PLC words, max. 64) can be entered in columns C1 to C6. For example, variables can be used to compensate for thermally induced drift as a function of position.

AxisB.COM

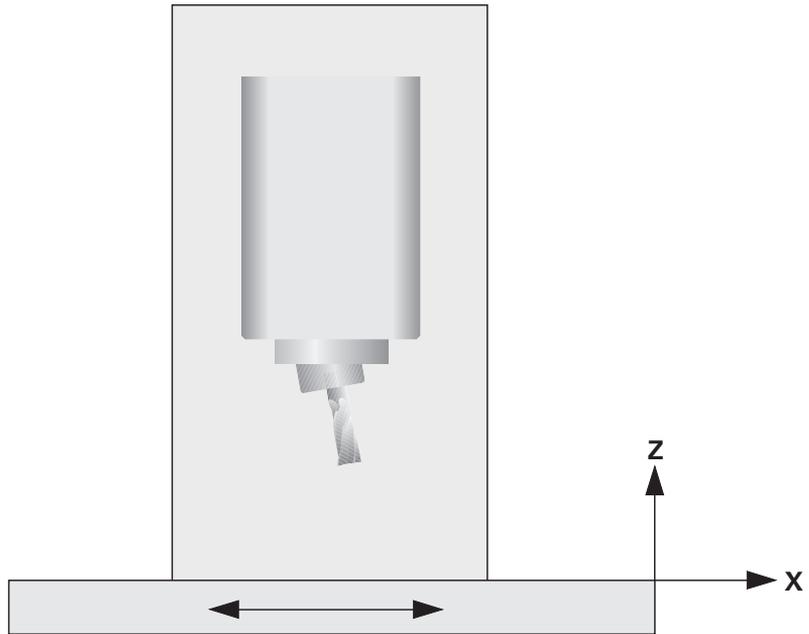
	Datum: -30	Distance: 50			
		5	C1 ... C4 ... C6		
0	-30	-0.008			-0.008
1	+0	+0			-0.004
...					
5	+120	+0.0012			+0.001

AxisB.COM

	Datum: -30	Distance: 50			
		5	C1 ... C4 ... C6		
0	-30	-0.008			0
1	+0	+0			W6002
...					
5	+120	+0.0012			W6010

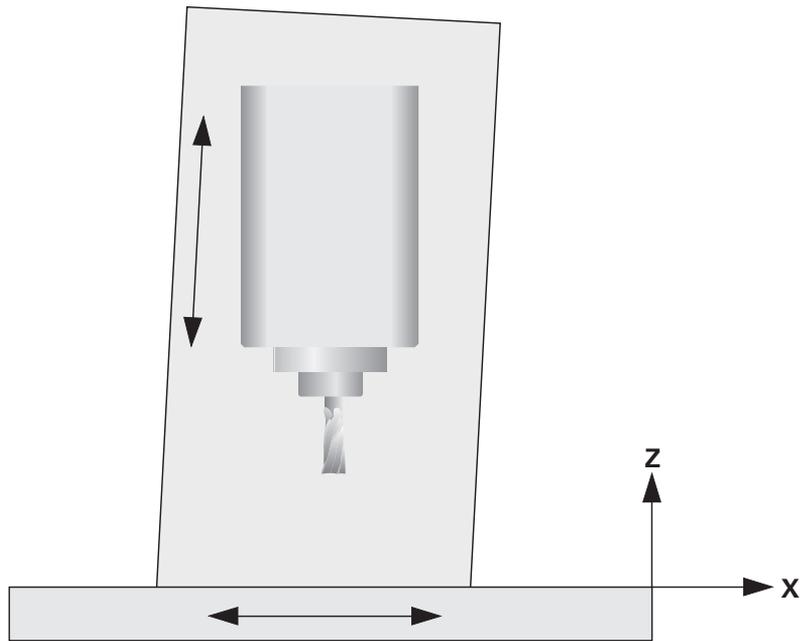
Examples

First example: Position error (angle) of spindle (A0S and B0S):



NR	KEY	AXIS	COORD	COMPTAB
	...			
	ToolAxis	Z		
	Trans	A		+0
	Trans	B		+0.0015
	MachAxis	Z		
	MachBase			
	MachAxis	Y		
	MachAxis	X		
	...			
	[END]			

Second example: Position error (angle) of a linear axis (A0Z and B0Z):



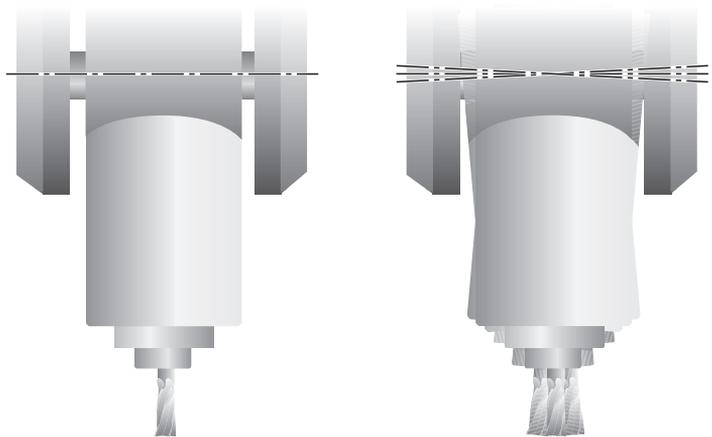
NR	KEY	AXIS	COORD	COMPTAB
	...			
	Trans	A		-0
	Trans	B		+0.0002
	MachAxis	Z		
	MachBase			
	Trans	B		-0.0002
	Trans	A		+0
	MachAxis	Y		
	MachAxis	X		
	...			
	[END]			

Third example: Position error (angle) of the spindle and the A axis of a fork head (B0S, B0A and C0A):



NR	KEY	AXIS	COORD	COMPTAB
	...			
	ToolAxis	Z		
	Trans	B		+0.003
	Trans	Y	-0.021	
	Trans	Z	+300.018	
	Trans	B		-0.0015
	Trans	C		+0.0005
	MachAxis	A		
	Trans	C		-0.0005
	Trans	B	0	+0.0015
	Trans	X	+0.025	
	Trans	Y	+0.017	
	Trans	Z	+199.936	
	MachAxis	C		
	MachAxis	Z		
	MachAxis	Y		
	MachAxis	X		
	...			
	[END]			

4th example: Component error of the A axis of a fork head:



NR	KEY	AXIS	COORD	COMPTAB
	...			
	Trans	X		A4C1
	Trans	Y	-0.021	A4C2
	Trans	Z	+300.018	A4C3
	Trans	B		A4C4
	Trans	C		A4C5
	MachAxis	A		
	Trans	C		-xxx
	Trans	B		-xxx
	Trans	Z		-xxx
	Trans	Y		-xxx
	Trans	X		-xxx
	Trans	X	+0.025	
	Trans	Y	+0.017	
	Trans	Z	+199.936	
	MachAxis	C		
	MachAxis	Z		
	MachAxis	Y		
	MachAxis	X		
	...			
	[END]			

Recommendation

KinematicsComp enables you to compensate for many types of errors. There are different possibilities for implementing the compensation. We recommend maintaining a certain configuration scheme in order to optimize the reproducibility of compensation.

HEIDENHAIN recommends a scheme, which links the types of errors (ISO 230-1) with the arrangement in the transformation chain (kinematics description) and the compensation file (*.COM). The errors that occur are always defined before the affected axis. This procedure assumes that the ideal coordinate system is equivalent to the machine table system. If necessary, the system can be reduced according to ISO 230 to improve the transparency of the transformation chain.

Linear axes (e.g. X axis) – Recommendation for COM file:

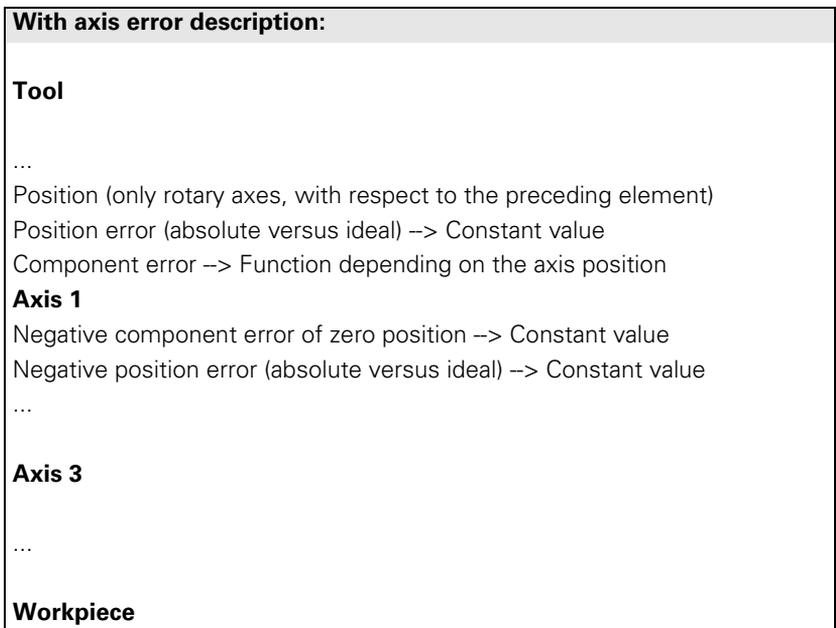
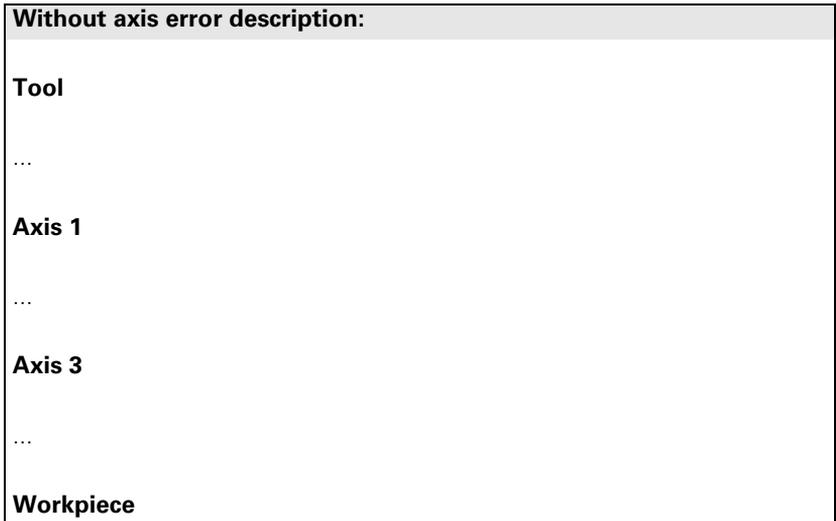
	Columns 1 - 18	Column C1	Column C2	Column C3	Column C4	Column C5	Column C6
Positioning error (EXX)	(X)	X					
Straightness error (EYX)	(X)		X				
Straightness error (EZX)	(X)			X			
EAX					X		
EBX						X	
ECX							X

Rotary axes (e.g. A axis) – Recommendation for COM file:

	Columns 1 - 18	Column C1	Column C2	Column C3	Column C4	Column C5	Column C6
Radial or axial error (EXA)		X					
Radial or axial error (EYA)			X				
Radial or axial error (EZA)				X			
Positioning error EAA	X						
EBA						X	
ECA							X

**Kinematics model
according to
ISO 230**

The kinematics description starts with the tool, continues with the axes and ends with the workpiece. The axis error descriptions must then be included in this description. A transformation chain with a complete axis error description looks like this:



Example of a linear axis (e.g. X axis) – Recommendation for transformation chain:

NR	DOC	KEY	AXIS	COORD	COMPTAB
		...			
	BOX	Trans	B		xxx
	COX	Trans	C		xxx
	EXX	Trans	X		A..C1
	EYX	Trans	Y		A..C2
	EZX	Trans	Z		A..C3
	EAX	Trans	A		A..C4
	EBX	Trans	B		A..C5
	ECX	Trans	C		A..C6
		MachAxis	X		
	-ECX where X=0	Trans	C		xxx
	-EBX where X=0	Trans	B		xxx
	-EAX where X=0	Trans	A		xxx
	-EZX where X=0	Trans	Z		xxx
	-EYX where X=0	Trans	Y		xxx
	-EXX where X=0	Trans	X		xxx
	-COX	Trans	C		-xxx
	-BOX	Trans	B		-xxx
		...			
[END]					



Example of a rotary axis (e.g. C axis) – Recommendation for transformation chain:

NR	DOC	KEY	AXIS	COORD	COMPTAB
		...			
	XOC	Trans	X	xxx	
	YOC	Trans	Y	xxx	
	AOC	Trans	A		xxx
	BOC	Trans	B		xxx
	EXC	Trans	X		A..C1
	EYC	Trans	Y		A..C2
	EZC	Trans	Z		A..C3
	EAC	Trans	A		A..C4
	EBC	Trans	B		A..C5
	ECC	Trans	C		A..C6
		MachAxis	C		
	-ECC where C=0	Trans	C		xxx
	-EBC where C=0	Trans	B		xxx
	-EAC where C=0	Trans	A		xxx
	-EZC where C=0	Trans	Z		xxx
	-EYC where C=0	Trans	Y		xxx
	-EXC where C=0	Trans	X		xxx
	-BOC	Trans	B		-xxx
	-AOC	Trans	A		-xxx
	-Position and	Trans	Y	-xxx	
	-YOC				
	-Position and	Trans	X	-xxx	
	-XOC				
		...			
[END]					



Reduced error model using a position error of a linear axis (e.g. X axis) – Recommendation for transformation chain:

NR	DOC	KEY	AXIS	COORD	COMPTAB
		...			
	BOX	Trans	B		xxx
	COX	Trans	C		xxx
		MachAxis	X		
	-COX	Trans	C		-xxx
	-BOX	Trans	B		-xxx
		...			
[END]					

The position errors calculated and averaged from several axis positions are entered here.

The position error X0X cannot be corrected in machine parameter MP960. Inverse transformation is necessary to be able to compensate one specific axis (if information about all axes is not available, for example). Describing this axis therefore restores the ideal kinematics model.

Reduced error model using a component error (with integrated position errors) of a linear axis (e.g. X axis) – Recommendation for transformation chain:

NR	DOC	KEY	AXIS	COORD	COMPTAB
		...			
	EYX	Trans	Y		A..C2
	EZX	Trans	Z		A..C3
	EAX and AOX	Trans	A		A..C4
	EBX and BOX	Trans	B		A..C5
	ECX and BOX	Trans	C		A..C6
		MachAxis	X		
	-ECX where X=0	Trans	C		-xxx
	-EBX where X=0	Trans	B		-xxx
	-EAX where X=0	Trans	A		-xxx
	-EZX where X=0	Trans	Z		-xxx
	-EYX where X=0	Trans	Y		-xxx
		...			
[END]					

The error EXX can be corrected conventionally in the COM file. The position errors BOX and COX that are relative to the home position of the X axis are integrated in the compensation values of the COM file (component errors). These values must be re-transformed to be able to return to the ideal model after the definition of the axis (MachAxis X). This reduced error model has the disadvantage that the position errors cannot be shown separately from the component errors.



Reduced error model using a position error of a rotary axis (e.g. C axis) – Recommendation for transformation chain:

NR	DOC	KEY	AXIS	COORD	COMPTAB
		...			
	XOC	Trans	X	xxx	
	YOC	Trans	Y	xxx	
	EAC	Trans	A		xxx
	EBC	Trans	B		xxx
		MachAxis	C		
	-EBC	Trans	B		-xxx
	-EAC	Trans	A		-xxx
		...			
[END]					

The position errors calculated and averaged from several axis positions are entered here.

The position error C0C can be corrected in machine parameter MP960.

Inverse transformation is necessary to be able to compensate one specific axis (if information about all axes is not available, for example). Describing this axis therefore restores the ideal kinematics model.

Reduced error model using a component error (with integrated position errors) of a linear axis (e.g. X axis) – Recommendation for transformation chain:

NR	DOC	KEY	AXIS	COORD	COMPTAB
		...			
	XOC and EXC	Trans	X		A..C1
	YOC and EYC	Trans	Y		A..C2
	EZC	Trans	Z		A..C3
	EAC	Trans	A		A..C4
	EBC	Trans	B		A..C5
		MachAxis	C		
	-EBC where C=0	Trans	B		-xxx
	-EAC where C=0	Trans	A		-xxx
	-EZC where C=0	Trans	Z		-xxx
	-YOC and -EYC where C=0	Trans	Y		-xxx
	-XOC and -EXC where C=0	Trans	X		-xxx
		...			
[END]					

The error ECC must be corrected conventionally in the COM file (columns 1 - 18). The position errors A0C and B0C that are relative to the home position of the C axis are integrated in the compensation values of the COM file (component errors). These values must be re-transformed to be able to return to the ideal model when the axis is in home position.



The following number of transformation entries results for a five-axis machine:

- Per linear axis: 10 entries in **COMPTAB**
- Per rotary axis: 10 entries in **COMPTAB**, 2 entries in **COORD**
- Total $5 \times 10 = 50$ in **COMPTAB**, $2 \times 2 = 4$ entries in **COORD**

All position and component errors of the axes of a five-axis machine tool (three linear and two rotary axes) can be described separately with 50 transformation entries. This means that the compensation of one axis has no effect on the other axes. The positioning errors (e.g. EXX, ECC) are corrected as usual in columns 1-18 of the COM files, the offsets are corrected in machine parameter MP960.

When determining only position errors, the size of the transformation chain is reduced:

- Per linear axis: 4 entries in **COMPTAB**
- Per rotary axis: 4 entries in **COMPTAB**, 2 entries in **COORD**
- Total $5 \times 4 = 20$ in **COMPTAB**, $2 \times 2 = 4$ entries in **COORD**

The isolated description is absolutely necessary if information about other axes is missing. If information about other axes is available, inverse transformations might be omitted or combined with the transformations of the subsequent axes.

The limitation of a maximum of 50 transformation entries (**Trans** and **MachAxis**) currently applies for the total description of the actual machine kinematics.

For error compensation with KinematicsComp, a maximum of 100 additional entries in the column **COMPTAB** is allowed. However, all **MachAxis** entries of the kinematics description must first be subtracted from the maximum of 100 entries, because the **MachAxis** entries are included in the calculation of the compensation (e.g. if five **MachAxis** entries exist, only 95 entries remain for use in the **COMPTAB** columns).

No more than 50 of these entries are allowed in the **COORD** column.



Position errors of spindle

Error model using a position error of the spindle in tool axis direction Z without rotary head axes – Recommendation for transformation chain:

NR	DOC	KEY	AXIS	COORD	COMPTAB
		TOOLFILE			
		ToolAxis	Z		
	AOS	Trans	A		xxx
	BOS	Trans	B		xxx
		...			
	[END]				

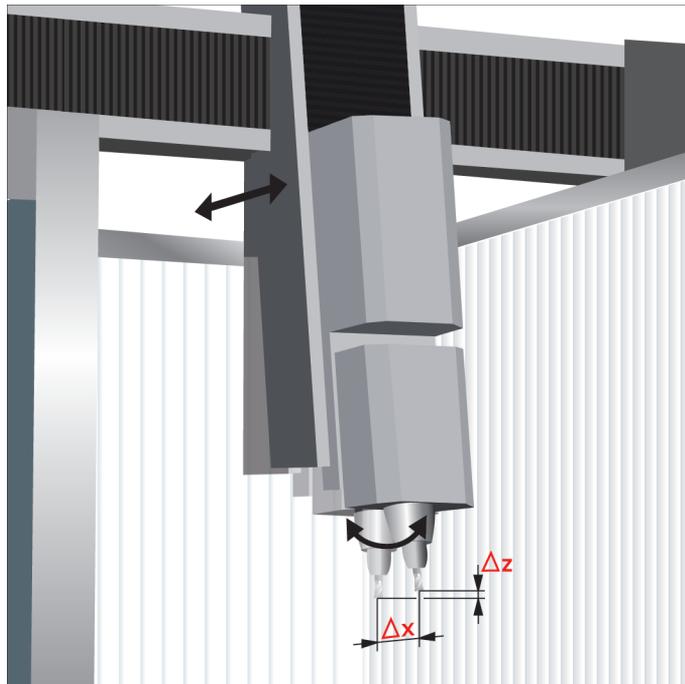
Error model using a position error of the spindle in tool axis direction Z with rotary head axes – Recommendation for transformation chain:

NR	DOC	KEY	AXIS	COORD	COMPTAB
		TOOLFILE			
		ToolAxis	Z		
	AOS	Trans	A		xxx
		Trans	...		
		...			
		MachAxis	B		xxx
		...			
	[END]				



6.4.10 Software option 141—Cross Talk Compensation (CTC)

To accelerate the axes of a machine tool, motor forces or torques must be applied. These forces can cause deformations of machine parts and thus lead to deviations at the Tool Center Point (TCP). These deviations can occur both in the direction of the accelerated axis and in other directions (cross talk).



If the dynamic deviations depending on the axis acceleration are known, they can be compensated with the CTC software option and thus be minimized.

Often, the resulting error at the TCP depends not only on the acceleration but also on the position or orientation of the axes in the working space. CTC can also take this into account.

The CTC software option permits compensations in the following format:

$$\Delta s_i = k_{ij}(s_k) * a_j$$

Where $k_{ij}(s_k)$ can depend linearly on the axis position k .

Δs : Error values

a : Acceleration

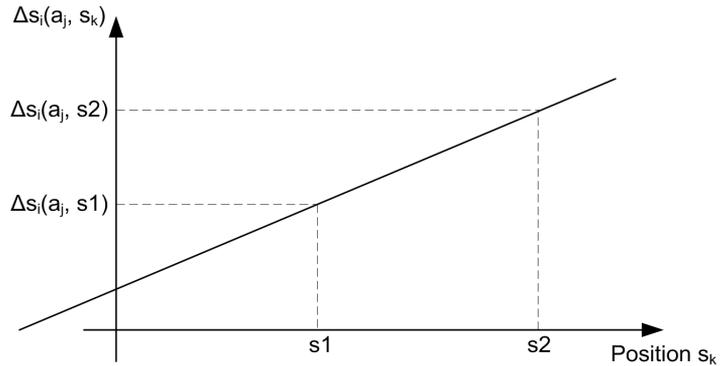
s : Axis position

i : Index of the axis to be compensated

j : Index of the accelerated axis

k : Index of the axis in relation to which there is a position dependency

The figure below shows the relationships for a specific acceleration of the axis j . Axis i is to be compensated, where the compensation value depends on the position of the axis k :



Considerations before using CTC

Considerations before using CTC:

- What kind of cross talk is to be expected?
- Which dependencies in the workspace (position) concerning cross talk are to be expected?

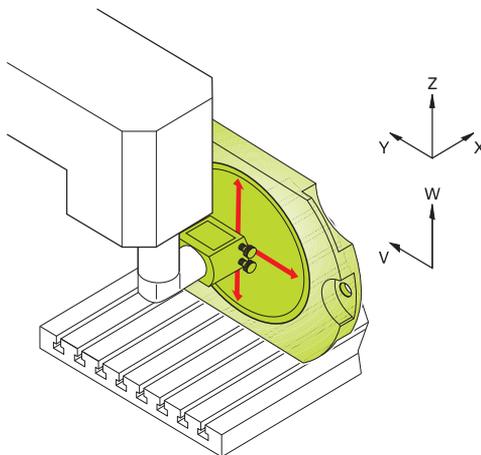
The cross talk to be expected must then be measured.

Only axes that are controlled by the same DSP can be compensated with each other for software option 141 (CTC). This means that such axes cannot be distributed to drive-control motherboards.

Measurement and parameterization

The compensation values can be ascertained via the control, by setting up a pure display axis or with an appropriate external encoder. Preparations for determining the compensation values for CTC:

- ▶ Create a suitable measurement program:
Linear movement in one axis at very high feed rate for high acceleration
Example for Y axis:
L Y100 F10000
L YO F10000
- ▶ Set up the required encoders and configure them.
Example: KGM grid encoder setup for measurement of cross talk in Y/Z plane
- ▶ Configure the axes of the grid (W in the example) as display axis.

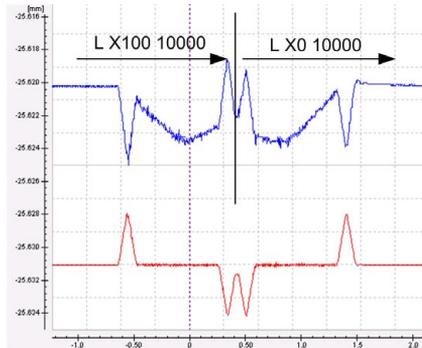


Determine the compensation values:

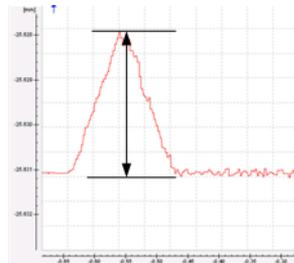
- Select the signals that are to be recorded with the integrated oscilloscope or TNCScope. The signals to be recorded must be differentiated depending on the cross talk:
 - Measure the deviation perpendicular to the accelerated axis (cross talk):
 - Actual acceleration of the exciting axis
 - Position value of the TCP in the direction of the axis to be compensated (transverse direction)
 - Determine the deviation in the direction of the accelerated axis:
 - Actual acceleration of the exciting axis
- Two signals must be recorded to determine the deviation:
- Signal 1: Position value of the TCP in the direction of the axis to be compensated (in this case this is the exciting axis)
 - Signal 2: Position value of the position encoder of the axis to be compensated
 - Deviation $\Delta s = \text{signal 1} - \text{signal 2}$

Example of determining the cross talk from Y to Z with the setup described above:

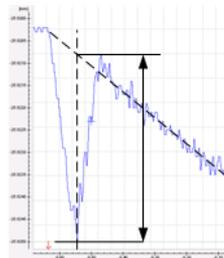
- ▶ Select the signals that are to be recorded with the integrated oscilloscope or TNCScope:
 - Actual acceleration of the exciting axis: Y a_actl
 - Position value of the TCP in the direction of the axis to be compensated (transverse direction): W s_actl
- ▶ Start the measurement program and record the signals
- ▶ Determine the cross talk between the axes:
 - Example of measured values
 - Blue curve = s_actl (W) without CTC, Red curve = a_actl (Y)



- ▶ Determine the maximum acceleration: $a_{max} = 1.58 \text{ m/s}^2$



- ▶ Determine the maximum amplitude:
 - For the determination of the cross talk, only the dynamic portion of the amplitude must be considered. If there is a long-range error, it must be subtracted. In this example, the amplitude would be approx. 0.0043 mm.



- ▶ Enter the respective values in TNCOpt under: Expanded Compensations / CTC tab

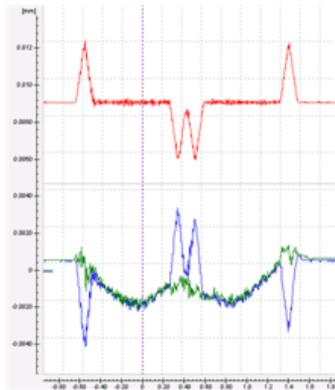
- ▶ Perform the measurement at different locations.
To determine a possible position dependence (e.g. dependence of height from the Z position), the measurement must be repeated at another position (second height).
- ▶ Enter all values in TNCopt and transfer to control.
- ▶ CTC must then be enabled for all machining axes via MP2700 bit 0.

This function is described in detail in the Help function of TNCopt 6.0. Available as of the end of 2011.

Checking the compensation

To verify the compensation values, the measurement program must then be run with active CTC. For test purposes you can activate and deactivate CTC axis-specifically with MP2700 bit 0.

Blue curve = $s_{actl}(W)$ without CTC, Green curve = $s_{actl}(W)$ with CTC



6.4.11 Software option 142—Position Adaptive Control (PAC)

The specifics of a machine's kinematics cause a unique position of the axes' center of gravity in the working space. This results in a variable dynamic behavior of the machine, which can adversely affect the stability of the servo-control significantly depending on the axis positions. To exploit the machine's dynamic possibilities, you can use the PAC option (Position Adaptive Control) to change machine parameters depending on position.

This makes it possible to assign respectively optimal loop gain to defined support points. Additional position-dependent filter parameters can be defined in order to further increase control loop stability. Besides the position-dependent adjustment of machine parameters, the PAC option also provides a way to change machine parameter values depending on other input quantities such as velocity or acceleration of an axis.

Up to five controller parameters can be changed by each of the two options PAC and MAC. A controller parameter must be changed either by PAC or by MAC, and not by both PAC and MAC.

The position-dependent parameter settings can be determined at various positions by using TNCopt (for example, the Bode diagram). As of version 6.0, TNCopt provides you with appropriate input forms for the parameter settings that can be transferred to the control.

Considerations before using PAC

Considerations before using PAC:

Is there a significant position dependence concerning the controller parameters? This becomes apparent:

- by using TNCopt if resonances vary considerably at different positions.
- if there is a position-dependent oscillation tendency of the axis. In this case there may also be audible oscillations at certain positions.

Preparation and activation

Activating PAC:

- ▶ Activate PAC via TNCopt in the "Expanded compensations" menu item.
- ▶ Adjust the axis at different positions (e.g. Bode diagram, step response).
- ▶ In the TNCopt form, enter the parameters you have determined for up to four support points. You can define up to five machine parameters and up to five position-dependent filters (with two support points). The parameters are interpolated linearly between the support points.
- ▶ PAC must then be enabled for all machining axes via MP2700 bit 1.

This function is described in detail in the Help function of TNCopt 6.0. Available as of the end of 2011.

6.4.12 Software option 143—Load Adaptive Control (LAC)

The dynamic behavior of machines with rotary tables can vary significantly depending on the mass moment of inertia of the fixed workpiece. The LAC option (Load Adaptive Control) enables the control to automatically ascertain the workpiece's current mass moment of inertia and the friction forces.

In order to optimize changed control behavior at differing loads, adaptive feedforward controls can exploit data on acceleration, holding torque, static friction and friction at high shaft speeds. In this case the parameter values are adapted using a relatively large time constant in order to ensure high resistance to disturbances during traverse.

Using LAC in cases where the friction in the system may change considerably and/or quickly, such as on machines with sliding guides, LAC can be used to activate an adaptive friction model that adapts the correct parameter for the friction current. In this case the friction model adapts to the currently effective conditions that may vary depending on the current traversing speed, aging of the machine or the lubrication cycles.

Adaptive feedforward control can be activated with TNCopt as of version 6.0. It is not necessary to save compensation values or tables.

Considerations before using LAC

Considerations before using LAC:

- Is it a moving table? The characteristics of rotary tables, tilting tables or movable tables change with the load. The mass of the workpiece can vary significantly during machining.
- Can the friction of a tool axis change depending on the load or other mechanical influences?

Preparation and activation

Activating LAC:

- ▶ Activate LAC via TNCopt in the "Expanded compensations" menu item.

Depending on the application, you can activate different feedforward options for the adaptive feedforward control of LAC:

- ▶ Holding torque (MP2630):
Should be activated if the holding current changes with the weight of the workpiece. This is the case with tables functioning as vertical axes.
- ▶ Basic friction (MP2610):
Should be activated if the friction changes with the weight of the workpiece. This occurs frequently with rotary tables.
- ▶ Friction at high shaft speed (MP2620):
If the friction changes at high shaft speeds, this feedforward control can be adapted.
- ▶ Acceleration feedforward control (MP2600):
If the weight of a table changes due to its load, the feedforward control is adapted accordingly.
- ▶ LAC must then be enabled for all machining axes via MP2700 bit 2.

This function is described in detail in the Help function of TNCopt 6.0. Available as of the end of 2011.

Checking the compensation

After a load change, the changed characteristics are determined during machining. The adaptation speed can be entered in TNCopt by using a gain. The default gain yields very good results in almost all cases. The gain should be adjusted in the following cases, however:

- If the gain is too low, the learning phase is very long. If this is the case you can see in the internal control oscilloscope that during the phase of acceleration or constant traversing speed, there is a discrepancy between the integral current and nominal current that decreases only very slowly.
- If the gain is too high, the adaptation algorithm might become unstable or the feedforward values might carry a lot of noise. The internal control oscilloscope shows a lot of noise in the nominal current.

Module 9047 Reset or initialize controller algorithms

Module 9047 can reset controller algorithms or reinitialize them.

Constraints:

- The module must not be called in the cyclic PLC program part.

Call:

```
PS    B/W/D/K  <Mode>
                0: Commands to the CC (reset/reinitialize)
PS    B/W/D/K  <Axis number>
                -1: All axes
                0 to n: Dependence on MP4020 bit 14
PS    B/W/D/K  <CC command>
                0: Reset Load Adaptive Control (SW option 143)
                1: Deactivate all adaptive controller parameters
                2: Activate all adaptive controller parameters
CM    9047
PL    B/W/D    <Error code>
                0: No error. Reinitialization successful
                1: Call was not in a submit/spawn job
                2: NC/CC does not support the desired function
                3: Invalid value in the <Mode> parameter
                4: Invalid value in the <Axis number> parameter
                5: Invalid value in the <CC command> parameter
                100: Programmed CC data is not available for programmed
                    axis
```

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Axis does not exist or status interrogation faulty
W1022	1	Invalid value for Mode parameter, CC code
	2	Invalid axis number programmed
	20	Call was not in a submit/spawn job
	45	Internal error
	51	Programmed CC command is not available for this axis on the CC, or the PLC module is not supported by this NC software
	61	Internal error



6.4.13 Software option 144—Motion Adaptive Control (MAC)

The MAC option provides a way to change machine parameters depending on motion-dependent input quantities (e.g velocity or following error). This makes it possible, for example, to realize a velocity-dependent adaptation of the kV factor on motors whose stability changes through the various traversing velocities.

Up to five controller parameters can be changed by each of the two options PAC and MAC. A controller parameter must be changed either by PAC or by MAC, and not by both PAC and MAC.

Considerations before using MAC

Considerations before using MAC:

Does the machine have a pronounced dependence on velocity or acceleration regarding the controller parameters?

This becomes apparent in:

- The internal control oscilloscope if resonances occur at different velocities or accelerations, and therefore the controller variables (actual speed, nominal current) are subject to oscillations.
- Audible oscillations at certain velocities and/or accelerations, or at a certain velocity threshold.

Is there a need to change any controller settings or feedforward functions depending on the velocity or acceleration?

This may be required for:

- Master-slave torque drives whose tensioning is to be reduced for rapid traverse (high acceleration), for example.

Preparation and activation

Activating MAC:

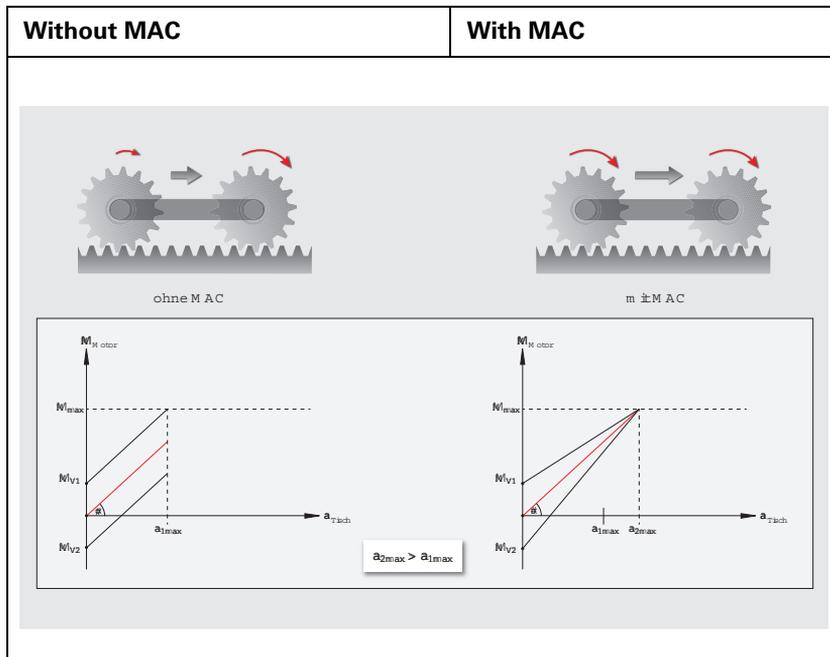
- ▶ Activate MAC via TNCopt (using the "Expanded compensation types" menu item).
- ▶ Set the controller parameters of the axis for different velocities or accelerations.
- ▶ Use TNCopt to enter the parameters you have determined for up to four support points. You can define up to five machine parameters and up to five position-dependent filters (with two support points). The parameters are interpolated linearly between the support points.
- ▶ Following error, jerk, acceleration and velocity are available as input quantities, to which the parameters are adjusted. Optionally, the input quantity can be considered unsigned. This is useful in symmetrical situations, for example to obtain the same adjustment for negative and positive accelerations.
- ▶ MAC must then be enabled for all machining axes via MP2700 bit 3.

This function is described in detail in the TNCopt 7.0 online help. Available as of the beginning of 2013.

Example

One possible application is the acceleration-dependent change of the tensioning torque between master axis and slave axis for master-slave torque control.

With the MAC option, this arrangement makes it possible to attain a significantly higher maximum acceleration at rapid traverse, for example through parameterized reduction of the tensioning torque with increasing acceleration.



6.4.14 Software option 145—Active Chatter Control (ACC)

Strong forces come into play during heavy cutting (power milling). These forces are transmitted between the milling spindle and the machining axes via the tool and the workpiece. Depending on the tool spindle speed, the resonances in the machine tool and the removal volume (metal-removal rate during milling), the tool can sometimes begin to "chatter." This chattering places heavy strain on the machine, and causes ugly marks on the workpiece surface. The tool, too, is subject to heavy and irregular wear from chattering. In extreme cases it can result in tool breakage.

To reduce a machine's inclination to chattering, HEIDENHAIN offers software option 145 Active Chatter Control (ACC). The use of this option is particularly advantageous regarding the attainable metal removal rate during heavy cutting. Depending on the type of machine, the metal removal rate can be increased significantly within the same machining time. Proceed as follows to be able to use ACC:

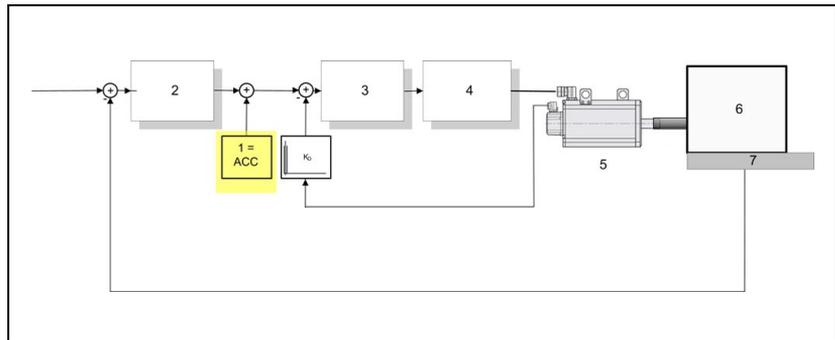
- ▶ Enable software option 145.
- ▶ ACC must then be enabled for all machining axes via MP2700 bit 4. The machining axes are usually the axes X, Y, and Z.
- ▶ For all tools that are used for heavy cutting, enter "1" in the column **ACC** of the tool table.
- ▶ For all tools that are used for heavy cutting, enter the number of tool teeth in the column **CUT** of the tool table.

If you then insert a tool with active ACC function, a damping signal reducing the chatter tendency is automatically added in the control loop.

You have the additional possibility of using machine parameter MP2710 to influence the weighting of the damping signal between the X, Y, and Z axes. In this machine parameter you can select the axis to which the higher-weighted damping signal is to be applied.

Besides this automatic function, you have the option of creating an ACC parameter file with TNCopt 7.0 and transfer it to the control. Since every tool influences the chatter characteristics of the cutting process, an individual file can be created for each tool. Alternatively you can also create a machine-specific file. The respective chatter frequency range and the intensity of the damping signal can be set using the parameter files. If a tool with active ACC function is inserted, the NC first searches for a tool-specific parameter file. If no such file is available, it searches for a machine-specific file. If this is not available either, ACC operates in automatic mode.

The block diagram shows how ACC works:



- 1: Active Chatter Control (ACC)
- 2: Position controller
- 3: Speed controller
- 4: Current controller – power module
- 5: Motor
- 6: Machine
- 7: Linear encoder

MP2700.x Activation of expanded functions (SW options)

Format: %0000000000000000

Input: Bit 0: CTC compensation
 0: Not active
 1: Active
 Bit 1: PAC compensation
 0: Not active
 1: Active
 Bit 2: LAC compensation
 0: Not active
 1: Active
 Bit 3: MAC compensation
 0: Not active
 1: Active
 Bit 4: ACC compensation
 0: Not active
 1: Active

MP2710.x ACC: Principle of function of automatic mode

Input: 0 to 2
 0: Robust standard mode with an average application of a damping signal (recommended setting)
 1: Chatter damping is increased for the selected axis; this can sometimes be advisable for one axis or two axes (e.g. vertical axes) and results in slightly reduced control accuracy (following error increases slightly)
 2: Chatter damping is significantly increased for the selected axis; this is usually advisable for no more than one axis and results in slightly reduced control accuracy (following error increases)



6.5 Tilting and swivel axes

Swivel heads and tilting tables are often used on milling machines for 5-axis machining and to machine workpieces from several sides.

The NC programs are written with a CAD system or directly at the iTNC using the **Tilt working plane** (Plane, TCPM) function. The user programs the part program in the X/Y plane and the iTNC interpolates the proper axes. All path functions, cycles, "datum setting" and "probing" can be applied in the transformed working plane.

6.5.1 Determining the mechanical offset

As an example, we will show how to determine the mechanical offset of a 45° double swivel head and of a forked swivel head.



Note

The 3-D ROT function must be inactive during the entire measuring process.

**Double swivel head
45°**

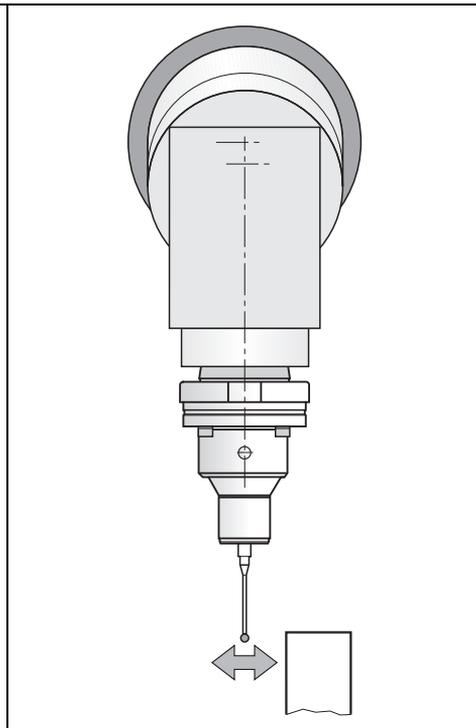
In this example, the mechanical offset of a double swivel head is determined by using a 3-D touch probe.

Input values for mechanical offset (via description table in the kinematics, ## are the values to be determined):

NR	KEY	AXIS	COORD	ON/OFF	FILE	DOC
0	Trans	Z	##			Shift in Z axis
1	Trans	X	##			Shift in X axis
2	Trans	A	##			Rotate around A axis
3	Trans	Z	##			Shift in Z axis
4	MachAxis	C				Free tilting axis C
5	Trans	A	##			Rotate around A axis
6	Trans	X	##			Shift in X axis
7	MachAxis	B				Free tilting axis B
:						
[END]						

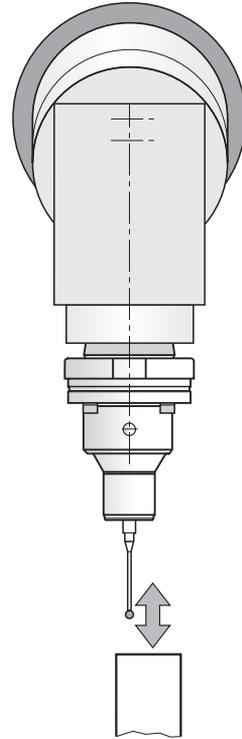
Step 1a

- Bring tilting axes B and C into 0° position
- Probe surface X1
- Set X = 0



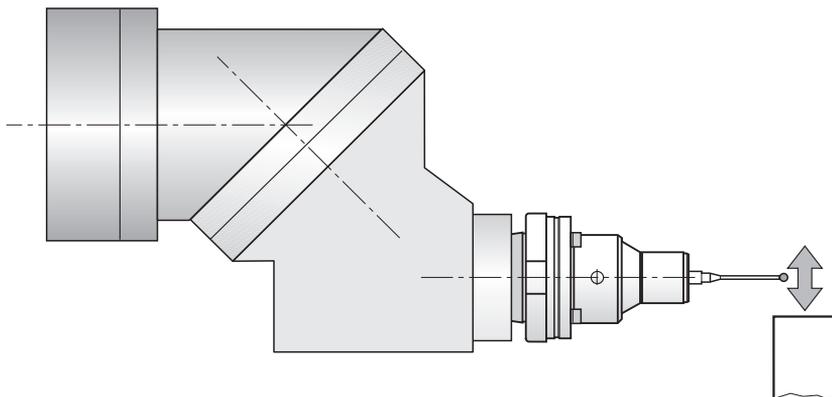
Step 1b

- Probe surface Z
- Set Z = 0



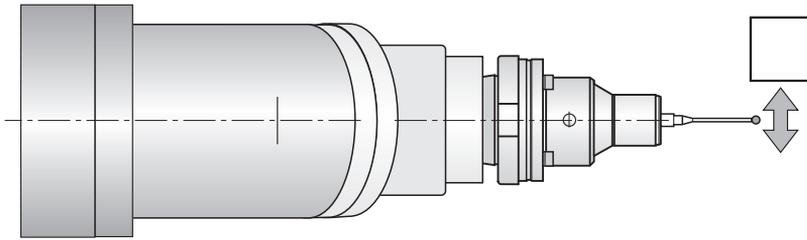
Step 2

- Position C to 180
- Probe surface Z
- Entry for the kinematics
 - Via table: **<Line no. [0] Column COORD>** = –determined value – probe length + ball radius



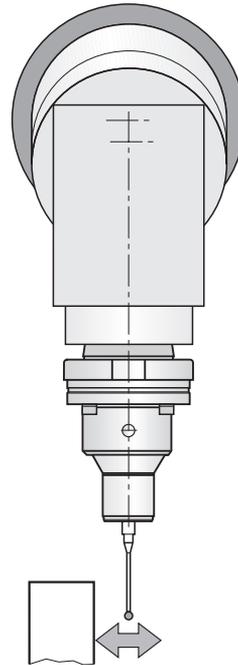
Step 3

- Probe surface X1
- Entry for the kinematics
 - Via table: <Line no. [1] Column COORD> = $-0.5 * \text{determined value}$



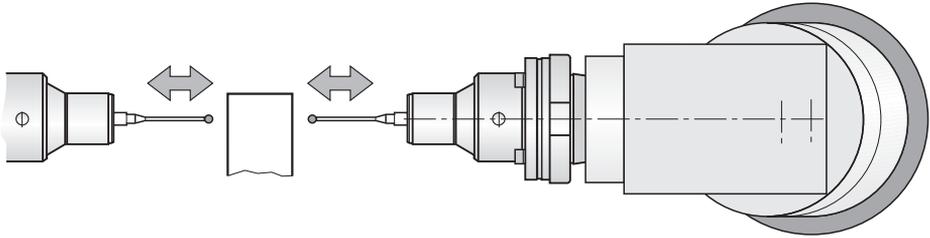
Step 4a

- Position C to 0
- Probe surface X2
- L = determined value



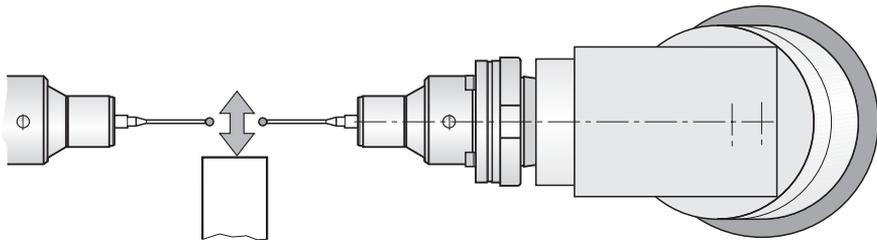
Step 4b

- Position B to -90
- Probe surface X1
- $\Delta X1$ = determined value
- Position B to +90
- Probe surface X2
- $\Delta X2$ = determined value
- Entry for the kinematics
 - Via table: <Line no. [3] Column COORD> = $\{[0.5 * (\Delta X2 - \Delta X1 - L - 2*(\text{probe length}) + 2*(\text{ball radius}))] - \text{MP7530.0}\} / \cos 45^\circ$



Step 4c

- Position B to -90
- Probe surface Z
- Set Z = 0
- Position B to +90
- Probe surface Z
- Entry for the kinematics
 - Via table: <Line no. [6] Column COORD> = $(-0.5 * \text{determined value}) - \text{<Line no. [1] Column COORD>}$

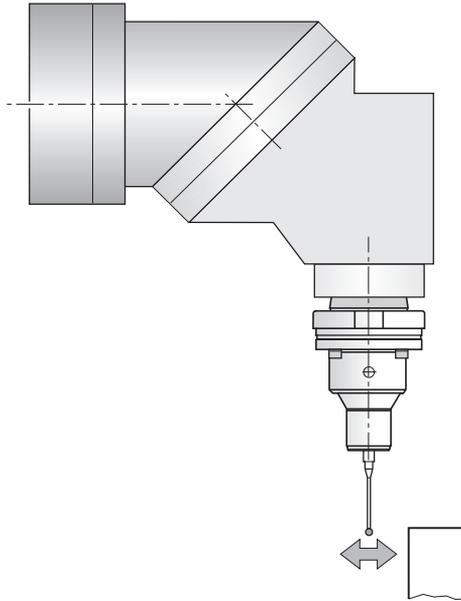


Step 5

- Probe surfaces X1, X2, Y2 and Z in a tilted working plane with the tilting angles $B = -90$, $B = +90$ and $C = 180$
- If there are differences between the individual tilting angles, the offsets from line numbers [0] and [3] in the kinematics table should be determined with a different process (steps 6 to 8), and the averages from both processes should be entered in line numbers [0] and [3] in the kinematics table.

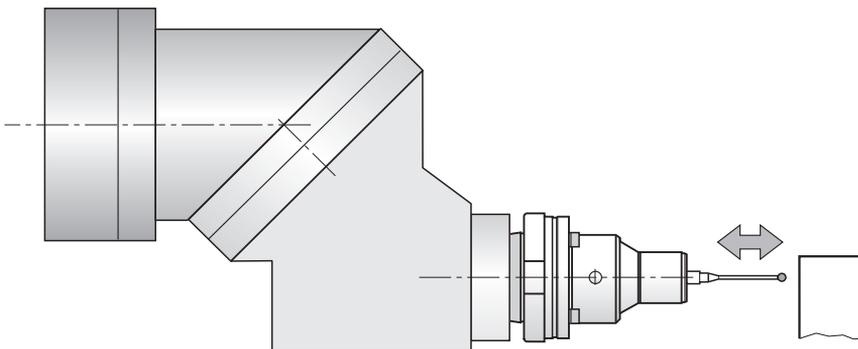
Step 6

- Probe surface Y2
- Set Y = 0



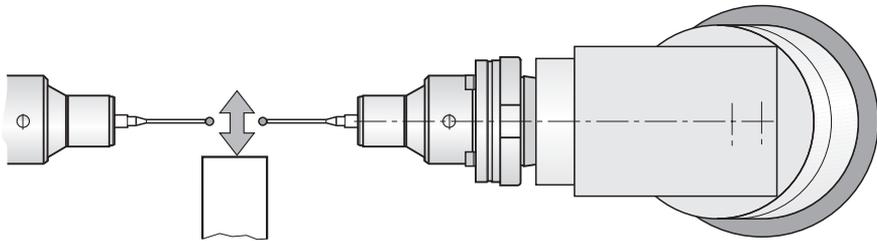
Step 7

- Position C to 180
- Probe surface Y2
- $\Delta Y2$ = determined value
- $Z1 = \Delta Y2 - \text{probe length} + \text{ball radius}$
- Entry for the kinematics if the results differ for
 - **<Line no. [0] Column COORD>** and $Z1$, then $\text{COORD} = 0.5 * (\text{<Line no. [0] Column COORD>} + Z)$



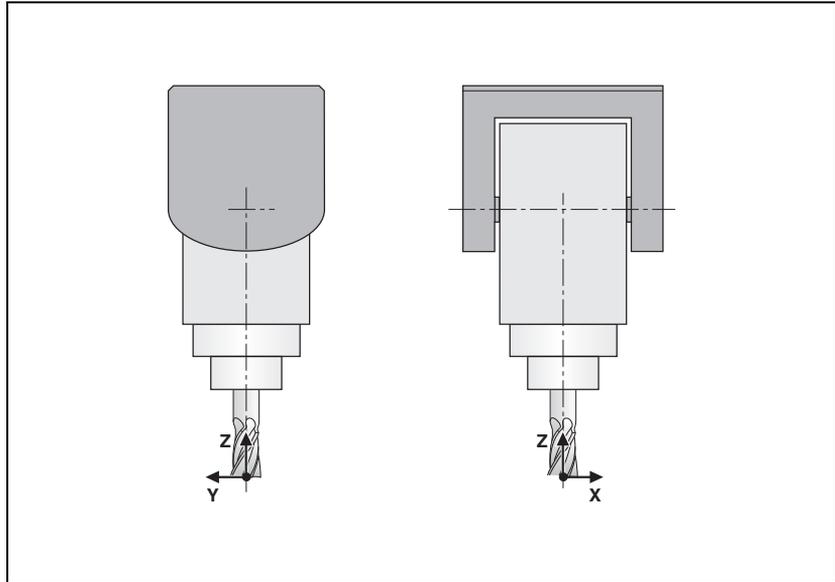
Step 8

- Position C to 0
- Position B to -90
- Probe surface Z
- ΔZ = determined value
- Position B to +90
- Probe surface Z
- $Z = \{[-0.5 * (\Delta Z + \text{determined value}) - \text{probe length} + \text{ball radius}] - Z1\} / \cos 45^\circ$
(Z1 see step 7)
- Entry for the kinematics if the results differ for
 - <Line no. [3] Column COORD> and Z, then $\text{COORD} = 0.5 * (\text{<Line no. [3] Column COORD>} + Z)$



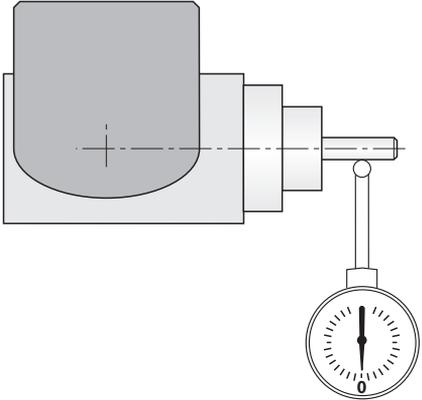
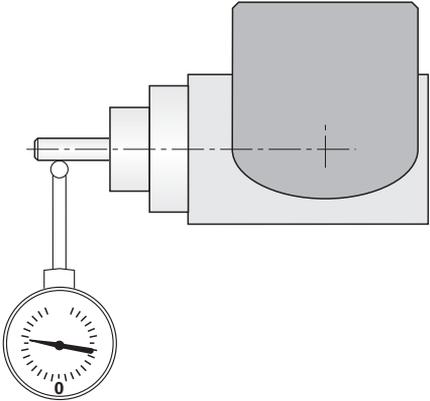
Forked swivel head

In this example, the mechanical offset of a forked swivel head is determined with a dial gauge and a cylinder with a known diameter.



Input values for mechanical offset (via description table in the kinematics, ## are the values to be determined):

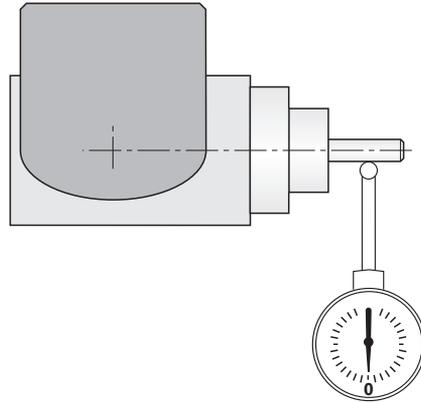
NR	KEY	AXIS	COORD	DOC
0	Trans	Z	##	Shift in Z axis
1	Trans	Y	##	Shift in Y axis (Y1: Offset of spindle to A axis)
2	MachAxis	A		Free tilting axis A
3	Trans	X	##	Shift in X axis
4	Trans	Y	##	Shift in Y axis (Y2: Offset of fork to C axis)
5	MachAxis	C	##	Free tilting axis C
:				
	[END]			

<p>Step 1a</p> <p>Determining the Y1 offset:</p> <ul style="list-style-type: none"> ■ Position A = -90 ■ Set the dial gauge to 0 	
<p>Step 1b</p> <p>Determining the Y1 offset:</p> <ul style="list-style-type: none"> ■ Position A = +90 ■ Offset = 0.5 * determined value ■ If the determined value > 0, then <Line no. [1] Column COORD> = - offset ■ If the determined value < 0, <Line no. [1] Column COORD> = - offset 	
<p>Step 1c</p> <p>Checking the settings for the Y1 offset:</p> <ul style="list-style-type: none"> ■ Position A = +90 ■ Activate 3-D ROT ■ Set the dial gauge to 0 ■ Set datum Y = 0 ■ Position A = -90 ■ Probe same position again ■ Display and dial gauge must read 0 	

Step 2a

Determining the Z offset:

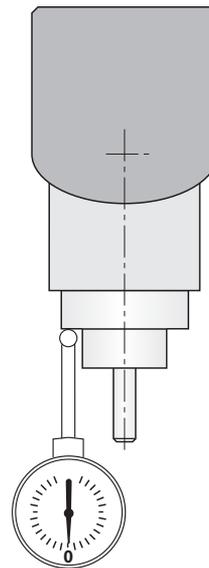
- Deactivate 3-D ROT
- Position A = -90
- Set the dial gauge to 0
- Set display Z = 0

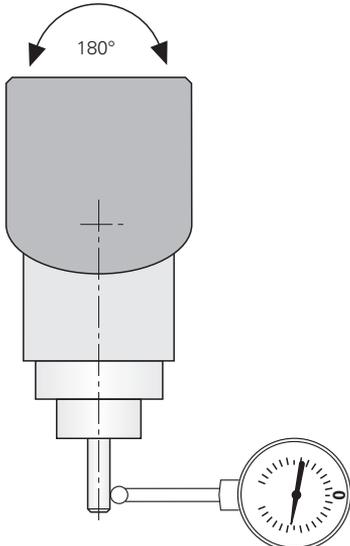
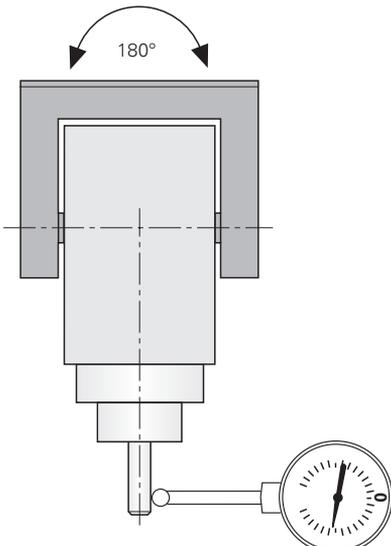


Step 2b

Determining the Z offset:

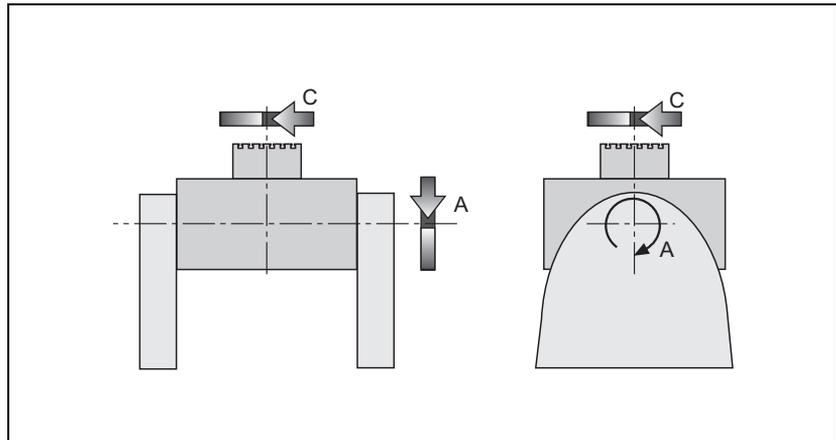
- Position A = 0
- Move the Z axis until the dial gauge reads 0 at the spindle tip
- <Line no. [0] Column **COORD**> = "determined value" – cylinder radius



<p>Step 3</p> <p>Determining the Y2 offset:</p> <ul style="list-style-type: none"> ■ Position A = 0 ■ Position C = 0 (basic setting) ■ Set the dial gauge to 0 ■ Position C to 180 ■ Read Y offset from the dial gauge ■ <Line no. [4] Column COORD> = $(0.5 * \text{determined value}) - \text{<Line no. [1] Column COORD>}$; <Line no. [1] Column COORD> = Offset Y1) 	
<p>Step 4</p> <p>Determining the X offset:</p> <ul style="list-style-type: none"> ■ C = Position to -90 ■ Set the dial gauge to 0 ■ Position C = 270 ■ Read X offset from the dial gauge ■ <Line no. [3] Column COORD> = $0.5 * \text{determined value}$ 	

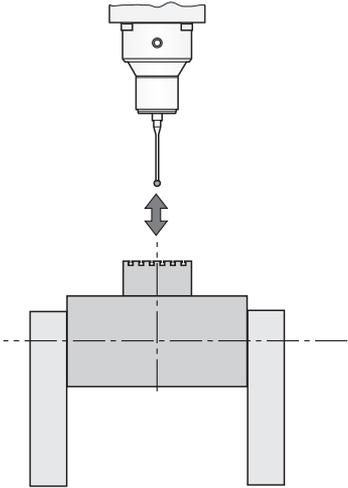
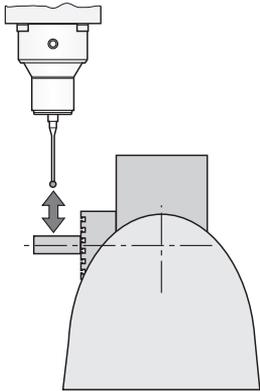
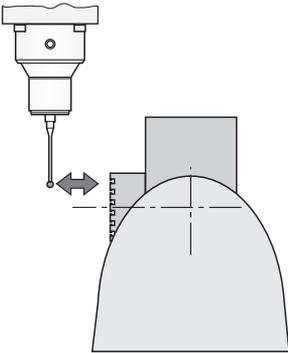
Tilting table / rotary table

In this example, the mechanical offset of a tilting or rotary table is determined by using a 3-D touch probe. In the A axis the table can only be tilted by -90° .



Input values for mechanical offset (via description table in the kinematics):

NR	KEY	AXIS	COORD	DOC
	MachAxis	X		X axis
	MachAxis	Y		Y axis
	MachAxis	Z		Z axis
	Trans	X	[X1]	Shift in X axis to the center of rotation of the C axis
	Trans	Y	$[Y1]+0.5 * (\Delta Z - \Delta Y)$	Shift in Y axis to the A axis
	Trans	Z	$[Z1]+0.5 * (\Delta Z + \Delta Y)$	Shift in Z axis to the A axis
	MachAxis	A		Free tilting axis A
	Trans	Y	$0.5 * (\Delta Z - \Delta Y)$	Shift in Y axis
	Trans	Z	$0.5 * (\Delta Z + \Delta Y)$	Shift in Z axis
	MachAxis	C	##	Free rotary axis C
	[END]			

<p>Step 1</p> <p>Ascertain the center point of the center of rotation of the C axis using the 3-D touch probe (= X1 and Y1); set X = 0 and Y = 0.</p>	
<p>Step 2</p> <ul style="list-style-type: none"> ■ Probe surface Z (= Z1) ■ Set Z = 0 	
<p>Step 3</p> <p>Determining the Z offset:</p> <ul style="list-style-type: none"> ■ Position A = -90 ■ Probe at test mandrel ■ ΔZ = radius of test mandrel + Z display 	
<p>Step 4</p> <p>Determining the Y offset:</p> <ul style="list-style-type: none"> ■ Probe surface Z ■ ΔY = Display Y 	
<p>Step 4</p> <ul style="list-style-type: none"> ■ $0.5 * (\Delta Z - \Delta Y)$ ■ $0.5 * (\Delta Z + \Delta Y)$ ■ Entry of the values in the description table as indicated above 	

6.5.2 Description of the kinematics configuration



Note

The description of the kinematics is only taken into account for tilting functions, such as TCPM or the Tilt Working Plane function.

The kinematics description consists of a transformation chain, starting from the tool reference point, to the swivel head, the linear axes and the tilting table, up to the center of the rotary table.

Seen as a whole, the new kinematics description defines one transformation chain, consisting of

- fixed lengths (machine dimensions)
- variable lengths (linear axes)
- fixed rotations (machine conditions)
- variable rotations (rotary axes)

starting from the tool reference point (e.g. spindle housing, swivel head, machine base, linear axes, machine frame, tilting table, rotary table). Each translation is described.

The difference to the previous kinematics description is that here, no coding is specified for the head or table rotary axes, or for the "ref. coordinates." This means that the sequence and algebraic signs of the table elements are reversed compared to the old description.

The following machine equation results for a closed transformation chain (tool reference point is at the center point of the rotary table):

$$\begin{array}{l} T_{\text{Head}} + T_{\text{Table}} + T_{\text{New}} - T_{\text{Ref}} = 0 \\ \Rightarrow \\ T_{\text{New}} = T_{\text{Ref}} - T_{\text{Head}} - T_{\text{Table}} \end{array}$$

T_{Head} : Translation for swivel head

T_{Table} : Translation for tilting table (algebraic sign according to the new kinematics description)

T_{New} : New translation (sum of the remaining translations after the tilting head to before the rotary table)

T_{Ref} : "Ref. coordinates" (previous description of the center point of the rotary table)



Kinematics tables

In order to describe the kinematics of a machine, various tables are necessary depending on the application and the software option (option 40 "DCM – Dynamic Collision Monitoring"). Depending on the type of content, these tables are structured according to the following hierarchy when using the new kinematics description.

Table overview	Brief description
<ul style="list-style-type: none"> ■ Assignment table <ul style="list-style-type: none"> ■ Description table 1 (FILE) ■ Description table 2 (FILE) <ul style="list-style-type: none"> ■ Tool-carrier table (TOOLFILE) ■ Partial description table (SUBFILE1, SUBFILE2) ■ Definition table for collision-monitored objects (CMO) 	List of selectable kinematics, is called in OEM.SYS
	Contains the transformation chain of the kinematics description, is referenced in the assignment table
	Contains the kinematics description of a specific tool carrier, is called in the description table and referenced via the tool table
	Contains the kinematics description of an exchangeable machine component that is included in the kinematics (e.g. swivel head), is called in the description table and referenced in the assignment table
	Contains descriptions of objects located within the machine work envelope, and for which the possibility of collision with other machine objects cannot be ruled out. CMO ("Collision Monitored Object")



Note

Definition tables, and therefore collision-monitored objects, are processed and therefore considered only with software option 40 "DCM – Dynamic Collision Monitoring" (ID 526 452-01).

Specify the machine kinematics as well as the resulting transformation model, starting from the machine reference point (REF 0, e.g. traverse block with M91).

- ▶ For machines with exchangeable components (e.g. adapter spindles, angle heads), specify partial kinematics as necessary. These can be included separately, in order to additionally increase the flexibility of the machine kinematics. Partial kinematics of this type are connected with the appropriate path in columns **SUBFILE1** or **SUBFILE2** of the **assignment table**.
- ▶ For tools with special tool-carrier kinematics, consider the possibility of connecting their additional kinematics descriptions via the tool table. Additional kinematics descriptions for tool carriers are connected in the **KEY** column of the **assignment table** via the **TOOLFILE** entry.
- ▶ Position the axes to the machine datum (transformation model with consideration of MP960).
- ▶ Now, starting from the tool reference point, define in sequence the shifts or rotations via the linear, rotary and tilting axes to a center point of a (rotary) table.
- ▶ Enter the shifts and rotations in the **description table**.

As a rule, the control takes changes in the mechanical offset into account, meaning that these changes do not have to be compensated with a PLC datum shift.



Note

- File paths should be entered as absolute paths. If only the file name is entered, the kinematics path from OEM.SYS is valid (e.g. KINEMATIC = PLC:\Kinemat\Kinelist.TAB).
- Another **SUBFILE** or **TOOLFILE** cannot be called from a **SUBFILE** or **TOOLFILE**, respectively
- File names can be entered with paths and the extension (.TAB) in the SUBFILE1, SUBFILE2 and FILE columns. If the path is missing, the path of the assignment table is used. If the file name extension is missing, *.TAB is amended (internally).

Assignment table

The assignment table "manages" the kinematics descriptions, and is the foundation for the selection dialog box called via the **KINEMATIC** keyword. The operator selects the kinematics description necessary for the current machine configuration from this dialog box. Normally only one assignment table is needed on a control.

In addition, by connecting partial description tables (**SUBFILE1** and **SUBFILE2**), the assignment table makes it possible to easily manage different machine configurations in just one assignment table. The iTNC ignores the missing entry of a SUBFILE in the assignment table, even if a **SUBFILE1** or **SUBFILE2** is entered in the description table. By combining the SUBFILE entries in the assignment table, you can cover four configurations with just one description table.

This makes the following combinations possible in the assignment table with one description table (here **KINEMAT1.TAB**):

NR	MP7500	FILE	MPFILE	SUBFILE1	SUBFILE2
0	1	PLC:\KINEMAT1.TAB			
1	7	PLC:\KINEMAT1.TAB	PLC:\MP\KINE...	PLC:\HeadA	
2	3	PLC:\KINEMAT1.TAB	PLC:\MP\KINE...	PLC:\HeadA	PLC:\HeadB
3	%11	PLC:\KINEMAT1.TAB	PLC:\MP\KINE...		PLC:\HeadB
[END]					

Specification of the assignment table:

- ▶ Switch to the **Programming and Editing** operating mode, press the MOD key and enter the code number 807667.
- ▶ Choose the file PLC:OEM.SYS from within program management.
- ▶ Enter the code word **KINEMATIC=**, followed by the file name with its complete path from the assignment table.
(e.g., **KINEMATIC= PLC:\KINEMAT\KINELIST.TAB**)
- ▶ Leave OEM.SYS by pressing the END key.

Creation of the assignment table:

- ▶ In program management, switch to the directory already selected via OEM.SYS, and enter the name of the assignment table specified in OEM.SYS, followed by the extension .TAB.
- ▶ Choose the table format with the MP7500, FILE, MPFILE, SUBFILE1, SUBFILE2 and DOC fields.
- ▶ In each line, enter the value from MP7500.x (integer or bit-encoded) for the information for tilting the working plane.
- ▶ In the **FILE** column, enter the absolute path to the description table.
- ▶ If necessary, enter in the **MPFILE** column an appropriate machine parameter subfile.
- ▶ If necessary, enter in the **SUBFILE1** or **SUBFILE2** columns a partial description file.

- ▶ Enter a brief descriptive text for the kinematics selection dialog. The texts displayed for selecting the kinematics can be created as language-sensitive texts (even with UTF8 character set). The window for selecting the kinematics is opened after the keyword KINEMATIC is entered. In the **DOC** column of the kinematics table, you enter a string that consists of the following elements: "**#<line number of DIALOG.A>**". Depending on MP7230.x, the texts is fetched from the specified line in DIALOG.A, and displayed in the selection window. The line numbers of DIALOG.A are identified by the preceding character #. At present, utf8 characters cannot be displayed in the selection window. If the entry in the **DOC** column is used as a keyword to make changes to the kinematics by using **WRITE TO KINEMATIC**, the "**#<line number of DIALOG.A>**" can be entered as KEY.

Example:

WRITE TO KINEMATIC AT COLUMN "HEIGHT" CAPTURE "DOC" KEY "#143" = 22

Each line of the assignment table contains information for referencing a specific kinematics description. Format of the assignment table:

Column	Input	Description
NR	0, 1, 2 ...	Automatic line numbering
MP7500	Integer (e.g. "3") or bit-encoded (e.g. %11)	Option selection for tilting the working plane (depends on the activation of the preset table). See the description of the "Tilt Working Plane" feature.
FILE	Path and file name ^a	Referencing of a description table for one machine kinematics
MPFILE	Path and file name ^a	Referencing of a machine parameter subfile, which is to be activated when these machine kinematics are selected. Relative paths can also be used when entering MP subfiles in the MPFILE column. The relative path always starts with the folder in which the assignment table is saved.
SUBFILE1 SUBFILE2	Path and file name ^a	Referencing of two possible partial description tables for each (selected) kinematics description. These tables are activated via the KEY column in the description table and the SUBFILE1 or SUBFILE2 keyword.
DOC	Text	Brief descriptive text for selection dialog of the kinematics model via code number

- File paths should be entered as absolute paths. If only the file name is entered, the kinematics path from OEM.SYS is valid (e.g. KINEMATIC = PLC:\Kinemat\Kinelist.TAB).

Example of an assignment table:

NR	MP7500	FILE	MPFILE	SUBFILE1	DOC
0	1	PLC:\KINEMAT1.TAB			Standard
1	7	PLC:\KINEMAT2.TAB	PLC:\MP\KINEMAT2.MP		B Head
2	3	PLC:\KINEMAT3.TAB	PLC:\MP\KINEMAT3.MP	PLC:\DoubleHeadTyp1	Double Swivel H1
3	%11	PLC:\KINEMAT4.TAB	PLC:\MP\KINEMAT4.MP	PLC:\DoubleHeadTyp2	Double Swivel H2
[END]					

With MP7506 you can make an advance selection of which entry in the kinematics table is to become active when the control is booted. Selection is made by entering the line number of the desired kinematics in **KINELIST.TAB**.

MP7506 Selection of kinematics at booting of the control

Input: 0 to 999
 -1: Function not active



Note

Please note that with this feature, the same kinematics are always active whenever the control is booted. If you change the actual kinematics of the machine before booting, this is not taken into account. The kinematics set via MP7506 are still active after booting even after this change.

Description table

The description table contains the actual definition of the kinematics. The transformations are described here, subkinematics (SUBFILE1, SUBFILE2, TOOLFILE) are connected if necessary, and if software option 40 (DCM - "Dynamic Collision Monitoring") is available, objects for collision monitoring are included. Any number of description tables can be used.



Note

- The number of kinematics entries in a description table (**Trans, MachAxis**) is limited to 50. This also includes the entries in **TOOLFILE, SUBFILE1** and **SUBFILE2**. However, references to collision bodies (**CM0**) do not count as kinematics entries.
- For a correct description of the kinematics, the machine axes X, Y and Z must form a Cartesian coordinate system.
- HEIDENHAIN offers the KinematicsDesign software tool. KinematicsDesign is a program for interactive creation of control kinematics, fixture templates, completed fixtures and tool carrier kinematics.



Creating/activating/changing an assignment table:

- ▶ In program management, switch to the desired directory and enter the name of the description table, including the extension .TAB.
- ▶ Choose the table format with the KEY, AXIS, COORD, ON/OFF, FILE, DONTTEST, TEMPCOMP and DOC fields.
- ▶ With the aid of the table format for a description table shown below, enter the description of the machine kinematics.
- ▶ Activate the description table by transferring the row numbers from the assignment table
 - from the PLC with Module 9097
 - from the NC with **FN17: SYSWRITE ID290 NR1 = ...** (line number)
- ▶ You can ascertain the active description table in two ways:
 - With the PLC you can use Module 9098 to ascertain the name of the description table or the line number in the assignment table.
 - With the NC you can use **FN18: SYSREAD Q1 = ID290 NR1** to ascertain the line number in the assignment table.



Note

The active description table is indicated with the status "M" in program management.

Time at which changes to the descriptions become effective:

- the iTNC is restarted
- the kinematics are selected via the code number **KINEMATIC**
- the kinematics are activated with **FN17: SYSWRITE ID290 NR1 = ...**
- the kinematics are selected with PLC Module 9097
- the description table is changed with **WRITE TO KINEMATIC**
- the description table is changed with **WRITE KINEMATICS**

The description table contains the actual kinematics description. Format of the description table:

Column	Input	Description
NR	0, 1, 2 ...	Automatic line numbering
KEY	■ TOOLFILE	The additional kinematics description referenced in the tool table via the entry in the KINEMATIC column, such as a tool carrier, is included in the calculations.
	■ SUBFILE1 ■ SUBFILE2	The additional description is included in a partial kinematics table, which is referenced via the file in the SUBFILE1 or SUBFILE2 column of the assignment table. Basically, the same content conditions apply for subfiles as for the description tables.
	■ MachAxis	Defines the point at which a machine axis becomes effective in the kinematics sequence. From this entry onward, movements by the indicated axis (AXIS column) change the positions of previously defined axes or objects relative to subsequent axes or objects. The start of the sequence is always the tool reference point.
	■ MachBase	Defines the stationary part of the machine (machine base) for the graphic simulation in KinematicsDesign. All movements of a machine are simulated around the stationary part of the machine. MachBase only affects graphic simulation in KinematicsDesign. However, HEIDENHAIN recommends entering "MachBase" at the respective position.
	■ CMO	Defines a monitorable collision body in the current kinematics sequence. This machine element is described by the file (table) referenced in the FILE column.
	■ ToolAxis	Entering an axis (X, Y, Z, U, V, W) defines the tool axis. Only this axis can be selected with a TOOL CALL , see page 728.
	■ Trans	Here a transformation of the indicated axis (AXIS column) in the current kinematics sequence by the value entered in the COORD column is performed. This can be a linear translation as well as a rotation about an axis.
AXIS	X, Y, Z, A, B, C ...	Entry of the axis designation for function given in the KEY column (valid for the MachAxis and Trans parameters in the KEY column).



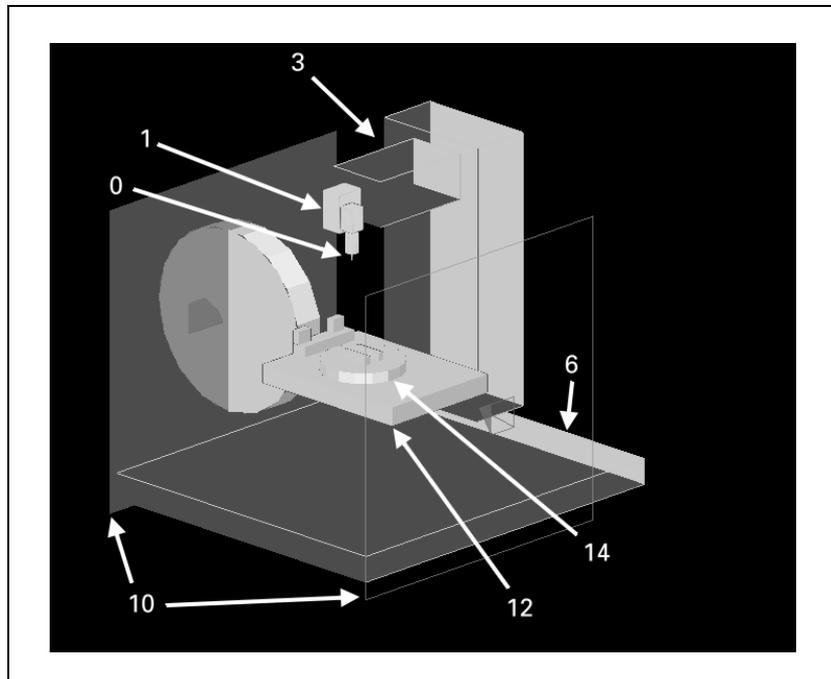
Column	Input	Description
COORD	E.g. 47.092 [mm] or 45.05 [°]	Entry of the transformation value for linear axes (X, Y, Z, ...) in mm or for rotary axes (A, B, C) in degrees. The units are not entered. The iTNC infers the units from the axis designation entered (valid for the Trans parameter in the KEY column).
ON/OFF	0 or 1 (1=CMO inactive)	Activate or deactivate a definition table for a collision-monitored object (valid for the CMO parameter in the KEY column). This makes it possible, for example, for a tool-change macro to switch off monitoring of the cabinet.
FILE	Path and file name ^a	Entry of the path and file name of a definition table for a collision-monitored object (valid for the CMO parameter in the KEY column).
DONTTEST	Path and file name ^a	Entry of the path and file name of a collision-monitored object (CMO), which is not to be monitored for collision with the CMO referenced in the FILE column (valid for the CMO parameter in the KEY column).
TEMPCOMP	Formula, e.g. $3.45e-3 * (W486 - 20)$	Entry of a formula for a permanently effective temperature compensation for the translation given in the line. For temperature compensation, see page 659.
DOC	Comment	Entry of a comment

- a. File paths should be entered as absolute paths. If only the file name is entered, the kinematics path from OEM.SYS is valid (e.g. KINEMATIC = PLC:\Kinemat\Kinelist.TAB).

The following example of a description table is intended to show the procedure for determining the kinematics of a machine.

NR	KEY	AXIS	COORD	ON/OFF	FILE	DONTTEST	TEMPCOMP
0	TOOLFILE						
1	SUBFILE1						
2	MachAxis B						
3	CMO				CMO_Portal		
4	MachAxis X						
5	MachAxis Y						
6	CMO				CMO_FloorSection	PLC:\Kinemat\...	
7	Trans	X	470.092				1.73e-3*(W486-20)
8	Trans	Y	-282.405				0.82e-3*(W488-20)
9	Trans	Z	-900				2.3e-3*(W490-20)
10	CMO			1	CMO_Cabin	PLC:\Kinemat\...	
11	MachAxis Z						
12	CMO				CMO_LiftTable	CMO_Cabin	
13	MachAxis C						
14	CMO				CMO_TurnTable	CMO_LiftTable	
[END]							

Representation of the machine by the description table above



Tool-carrier kinematics table (TOOLFILE)

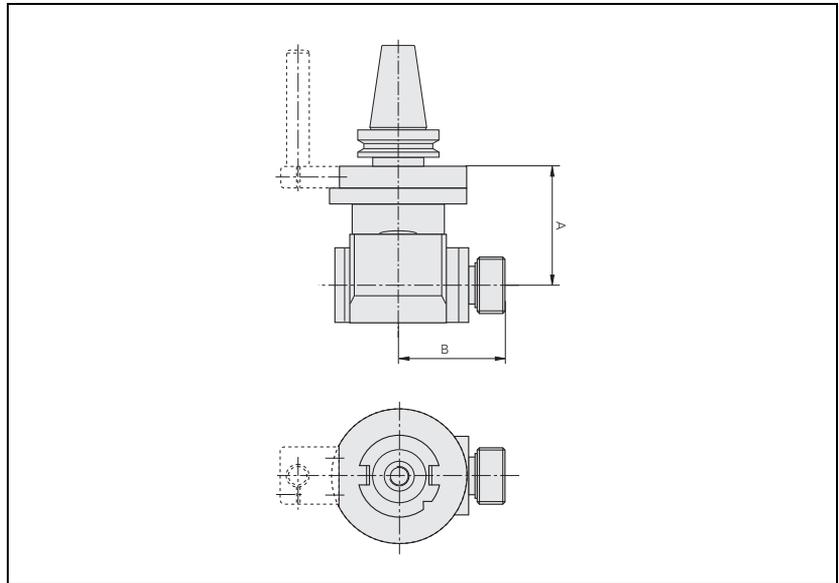
The **TOOLFILE** table is used to automatically activate specific tool-carrier kinematics when a tool-change procedure occurs. The inclusion of a **TOOLFILE** table adds an additional shift of the tool-reference point (reference point for tool dimensions) to the subsequent description of the kinematics. This occurs dynamically with the tool change.

For example, if the following **TOOLFILE** is included for an angular milling head (tool is aligned in the X direction) in the **KINEMATIC** column of the tool table, then the previous tool-reference point is shifted in the X direction by 96 mm and in the Z direction by -105 mm, and the tool alignment is rotated in the B direction by -90°. This additional kinematics description is added to the existing kinematics. This corresponds to the offset that this tool carrier would cause in the current kinematics representation.

NR	KEY	AXIS	COORD	ON/OFF	FILE
0	Trans	B	90		
1	Trans	X	-96		
2	Trans	Z	105		
3	CMO				CMO_Toolcarrier
[END]					

Example:

A = 105 mm, B = 96 mm, tool alignment in X axis



The tool holder of the tool carrier stated above (tool reference point) is now the initial point for viewing the kinematics with the included **TOOLFILE**. The subsequent kinematics description is based on this. This means that the algebraic signs of the shift are now determined by the new tool reference point.

The tool dimensions (e.g. tool length) in the tool table are automatically included in machining from this point on.

Transformations can be entered in the **TOOLFILE** table. If software option 40 has been activated (as in the table above), then collision objects (CMO) can also be entered.

The structure of these tables is identical to that of the description tables, but with the restriction that **TOOLFILES** and **SUBFILES** cannot be included in them.

The kinematics description of a **TOOLFILE** becomes active at the point from which it is included within the description table. Since the tool is at the beginning of the transformation chain, a **TOOLFILE** is only useful if it is included in the first line of the description table.

If no tool-carrier kinematics are referenced in the tool table (no entry in the **KINEMATIC** column), then the **TOOLFILE** entry in the description table has no effect.

In the following representation, the **TOOLFILE** referenced via the **KINEMATIC** column of the tool table is included in the description table.

The screenshot displays three overlapping windows in a CNC control environment:

- Top Window: Tool table editing**

Additional kinematic description

File: TOOL1

NR	KEY	AXIS	COORD	FILE
0	N		+0	+0
1	N		+0	+0
2	N		+0	+0
3	N		+0	+0
4	N		+0	+0
- Middle Window: Program table editing**

File: KINEMAT5.TAB

NR	KEY	AXIS	COORD	FILE
0	TOOLFILE			
1	SUBFILE1			
2	MachAxis B			
3	CMO			CMO_Portal
4	MachAxis X			
5	MachAxis Y			
6	CMO			CMO_FloorSection
7	Trans X		470.092	
- Bottom Window: Program table editing**

File: TOOLDRB1.TAB

NR	KEY	AXIS	COORD	FILE
0	CMO			CarrierBottom
1	Trans X		0.03	
2	Trans Z		22	
3	CMO			CarrierTop
4	Trans X		2.5	
5	Trans Z		1.75	



Partial description table (SUBFILE)

The **SUBFILE** makes it possible to integrate the kinematics of entire machine components (such as swivel heads and machine tables) from an external description table. The structure of these tables is identical to that of the description tables, but with the restriction that **TOOLFILES** and **SUBFILES** cannot be included in them.

Transformations, machine axes and collision objects (CMO) can be entered in the **SUBFILE**.

The kinematics description of a **SUBFILE** becomes active at the point from which it is included within the description table. However, please note that the **SUBFILE** is already referenced in the assignment table via the **SUBFILE1** or **SUBFILE2** column.

If no partial description table is entered in the SUBFILE1 or SUBFILE2 columns of the assignment table, then any **SUBFILE1** / **SUBFILE2** entry in the description table has no effect.

In the following representation, the **SUBFILE1** referenced via the assignment table is included in the description table.

The screenshot displays a multi-windowed software interface for program table editing. The main window shows a table with columns for line number (NR), key (KEY), axis coordinate (AXIS COORD), and file (FILE). A secondary window is open over the main one, showing a detailed view of a specific file path: PLC:\Kinemat\Heads\DoubleHeadType1. A third window is also visible, showing a table with columns for line number (NR), key (KEY), axis coordinate (AXIS COORD), and file (FILE). The interface includes a toolbar on the right with icons for M, S, T, and DIAGNOSE, and a bottom toolbar with buttons for BEGIN, END, PAGE, INSERT LINE, DELETE LINE, NEXT LINE, LIST, and FORH.

NR	KEY	AXIS COORD	FILE
0	SUBFILE1		
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			

NR	KEY	AXIS COORD	FILE
0	SUBFILE1		
2	MachAxis B		
3	CMO		CMO_Porta1
4	MachAxis X		
5	MachAxis Y		
6	CMO		CMO_FloorSection
7	Trans X	470.092	
8	Trans Y	-202.405	
9	Trä Manual		
10	CMO operation		
11	Mac		
12	CMO		
13	Mac		
14			

NR	KEY	AXIS COORD	FILE
0	CMO		PLC:\Kinemat\Heads\CMO_DoubleHeadA
1	Trans X	0.03	
2	Trans Z	170	
3	MachAxis A		
4	CMO		PLC:\Kinemat\Heads\CMO_DoubleHeadB
5	Trans X	100	
6	Trans Z	130	

Specifying the tool axis in the kinematics

It is possible to specify a permitted tool axis in the kinematics table. This way you can permit only a certain tool axis to be selected with a **TOOL CALL**.

A tool axis is defined in the kinematics table with the **ToolAxis** keyword in the **KEY** column and by indicating an axis (X, Y, Z, U, V, W). As an alternative, the tool axis can also be defined in a SUBFILE or TOOLFILE.

If a different tool axis is selected with a **TOOL CALL**, the **Tool axis not allowed** error message is output and an NC stop is triggered. The error message is generated as well if no tool axis is programmed in the **TOOL CALL** and no tool axis has been defined modally.

So that the kinematics can be switched, the tool axis is not checked during the switch. This makes it possible to change the tool axis by switching the kinematics. The axes can be traversed without a subsequent **TOOL CALL** to activate the new, permitted tool axis, but the entry under **ToolAxis** and the active tool axis are not checked.

When an NC program is started in the Test Run operating mode, a default tool axis is also set. It is taken from the **ToolAxis** entry (permitted tool axis) of the kinematics active in the test run.



Attention

The following must be kept in mind when specifying a tool axis:

- After the kinematics have been switched, checking of the tool axis must be activated with a **TOOL CALL**.
- A tool axis may only be defined once via **ToolAxis** in the machine kinematics. If **ToolAxis** is defined more than once, then only the first instance is used.
- If the **ToolAxis** is to be switched via the kinematics tables, then the definition of the tool axis should be performed in the various SUBFILES. This way you avoid multiple definitions in one kinematics description.



The iTNC enables you to append a 3-D basic rotation to the end of the kinematics description, e.g. for chucking equipment.

When a rotation is appended to the kinematics, the coordinate system of the machine table is rotated with respect to the coordinate system of the machine axes. The preset vector from the preset table is always interpreted as a vector of the coordinate system of the machine table. As a result, the rotations must first be appended to the kinematics, and then the reference point must be determined. The working plane must be correspondingly tilted for the determination of the reference point.

The following entries must be appended to the end of the kinematics description (e.g. after the description of the rotary table):

Trans C ...

Trans B ...

Trans A ...

Then you can use the WRITE TO KINEMATIC function to enter the coordinates of the rotation in the kinematics table. The rotation becomes effective when the PLANE function is activated.

Use Cycle 431 "Measure plane" to determine the spatial angles for the rotation. Then the coordinates of the rotation must be entered in the kinematics description in the sequence described above. The coordinates of the rotation only correspond to the spatial angles of the kinematics description if they have been entered in the described sequence. Please also note that the first two measurement results of Cycle 431 define the orientation of the principal axis (principal axis X if tool axis Z). It may be necessary to align the workpiece in the plane before determining the rotation with Cycle 431. To do this, use Touch Probe Cycle 403.

The READ KINEMATICS and WRITE KINEMATICS functions are used to read and modify elements of the currently active kinematics in the NC program. In the NC program you cannot define this function (soft keys: program functions, kinematic functions) until you have entered the code number 555 343. With the iTNC 530, transformations and geometry axes are elements of the transformation chain.

READ KINEMATICS

Syntax:

READ KINEMATICS NOTE_QS0 MODE Q0 RESULT_Q1 KEY "KEY;AXIS" TO QS1

- **NOTE:**
The function saves a "pointer" in this string variable. You need this pointer if you want to use WRITE KINEMATICS to describe these transformation elements again.
- **MODE:**
0: Increment read-pointer (next line)
1: Set the read-pointer to the beginning of the transformation chain (first line)
- **RESULT:**
In "Result", the function provides information about possible errors.
0: No error
1: Error
- **KEY:**
The key words whose entries are to be read are transferred in "Key." With the iTNC 530, the key words match the column names of the transformation table. The individual key words are separated by a semicolon.
- **TO:**
Keys are returned, separated by a semicolon. The sequence depends on the sequence of the "KEY" key words.

WRITE KINEMATICS

Syntax:

WRITE KINEMATICS NOTE_QS0 RESULT_Q1 KEY "COORD" = QS50

- **NOTE:**
Pointer to a certain transformation element. This pointer must first be determined with READ KINEMATICS.
- **RESULT:**
In "Result", the function provides information about possible errors.
0: No error
1: Error
- **KEY:**
The key words whose entries are to be overwritten are transferred in "Key." The individual key words are separated by a semicolon.
- **=:**
A string that contains the new entries is to be transferred as value. The entries are separated by a semicolon. The sequence depends on the sequence of the "KEY" key words.



**Kinematics tables
via PLC**

Module 9098 Find the active geometry description

Module 9098 can find the name of the active description table and/or line number in the assignment table.

Call:

PS B/W/D/K <String number for table name>
(line number is also determined)
-1: Find only line number, no name

CM 9098

PL B/W/D <Line number in the assignment table>
-1: Line number not found

Error code:

Marker	Value	Meaning
M4203	0	Name and/or line number found
	1	Error code in W1022
W1022	2	Incorrect parameter for string number
	20	Module was not called in a spawn or submit job

Module 9097 Select the geometry description

A geometry description from an assignment table can be chosen with Module 9097. The module can be called in a running NC program only in connection with a strobe. The module must be called in a submit job or spawn job, and cannot be canceled with the CAN command.

Call:

PS B/W/D/K <Line number in the assignment table>

PS B/W/D/K <Mode, reserved>
Transferred value must be 0

CM 9097

PL B/W/D <Error condition>
0 = Geometry description was selected
1 = Invalid mode
2 = Line was not found in the assignment table
3 = Assignment table is not defined
4 = Description table does not exist
5 = Description table is incomplete
6 = Module was not called in a spawn job or submit job
7 = Call during running NC program without strobe
8 = No **KINEMATIC=** entry in the OEM.SYS file
9 = Error in the **MPFILE** column
10 = Error in the **MP7500** column
11 = Error in the machine parameter subfile
12 = Reserved
13 = Names of collision objects (CMO) not unique

Error code:

Marker	Value	Meaning
M4203	0	Geometry description was selected
	1	Error code in W1022
W1022	2	Invalid mode; or line was not found in the assignment table; or description table was not defined, does not exist or is incomplete; or there is no KINEMATIC= entry in the OEM.SYS file.
	9	Error in the MPFILE column
	10	Error in the MP7500 column
	11	Error in the machine parameter subfile
	20	Module was not called in a submit or spawn job
	21	Call was made during a running NC program without a strobe



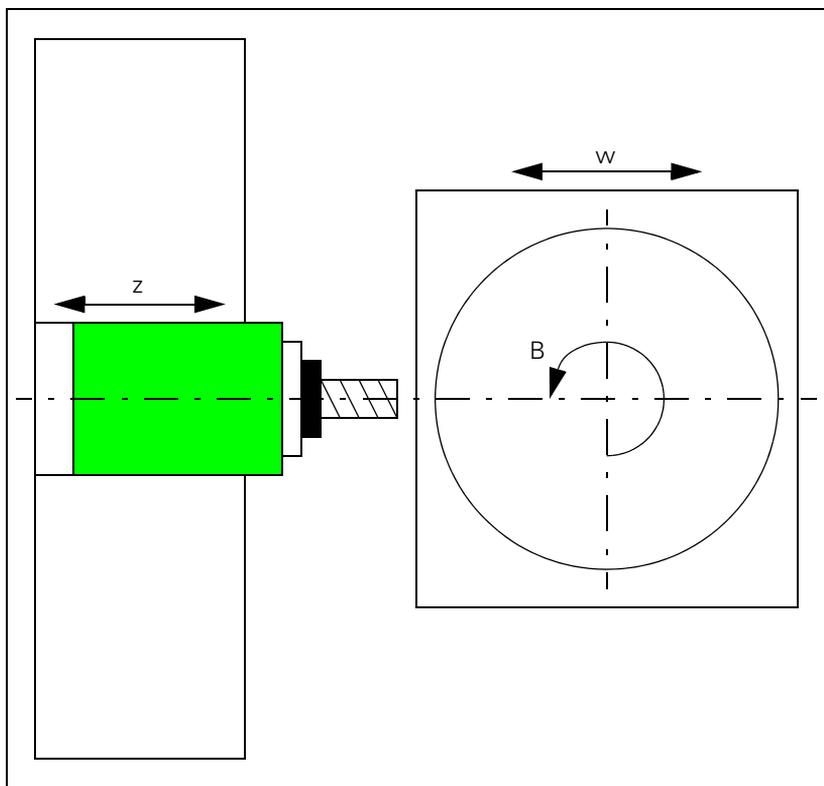
Overwriting the geometry description

Individual cells of the description table can be overwritten from within a machining program.

- ▶ Enter the code number 555343.
- ▶ Press the **ENTRY IN KINEMATIC TABLE** soft key.
- ▶ With the **WRITE TO KINEMATIC AT COLUMN <column to be written to> CAPTURE <searched column> KEY <keyword> =<value>** command you can overwrite individual cells in the active kinematics description in the Program Run operating modes. The line is selected by searching the **CAPTURE** column for the entry **KEY**. The column **COLUMN** is overwritten with **<value>**.

Example 1:

Vertical machining center, B-rotary table in the W axis, machine quill in Z



The position of the W axis must be considered in the kinematics table in order to use the tilting functions. For example, an NC macro can be programmed before the tilting function. This macro includes the current position of the W axis in the calculations of the kinematics table.

The following requirements apply to the correct calculation of the W axis:

- The kinematics table is determined with a defined position of the W axis.
- The Z position to be corrected is marked in a newly created column, **DOC** for example.

Example:

Deactivation of an active COM table so that collision monitoring is deactivated for a certain action, such as a tool change.

Table in the kinematics description before overwriting:

NR	KEY	AXIS	COORD	ON/OFF	FILE	DONTTEST	TEMPCOMP
:							
2	MachAxis	B					
3	CMO				CMO_Portal	PLC:\Kinemat\...	
4	MachAxis	X					
:							
[END]							

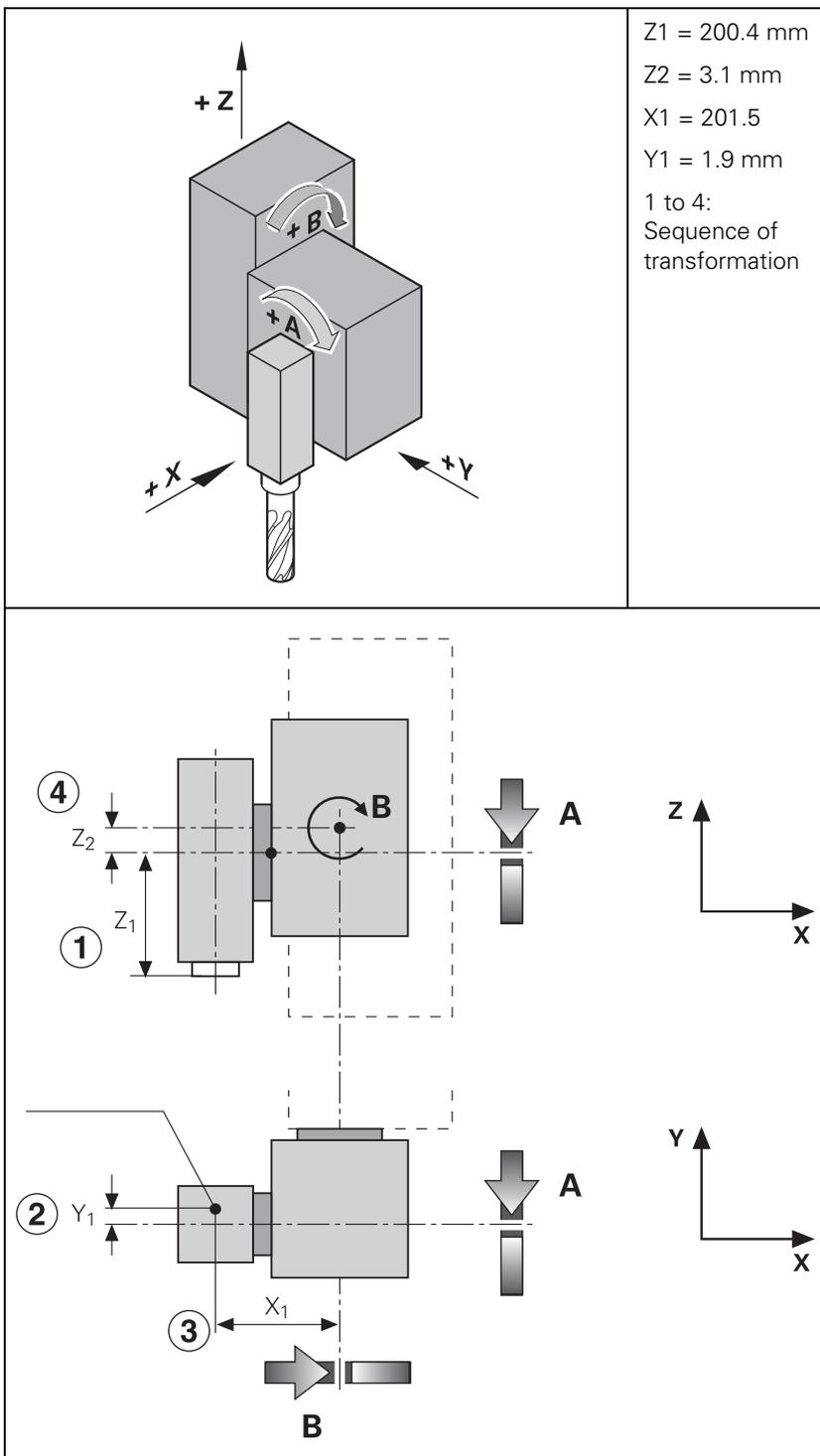
<pre> 0 BEGIN PGM N545TCM MM : 4 WRITE TO KINEMATIC AT COLUMN "ON/OFF" CAPTURE "FILE" KEY "CMO_Portal" = 1 : </pre>	<p>Comment:</p> <p>Overwrite cell in the kinematics table</p>
---	---

Table after overwriting with collision monitoring switched off for the **CMO_Portal** object:

NR	KEY	AXIS	COORD	ON/OFF	FILE	DONTTEST	TEMPCOMP
:							
2	MachAxis	B					
3	CMO			1	CMO_Portal	PLC:\Kinemat\...	
4	MachAxis	X					
:							
[END]							



**Example 1:
Rectangular double
swivel head**



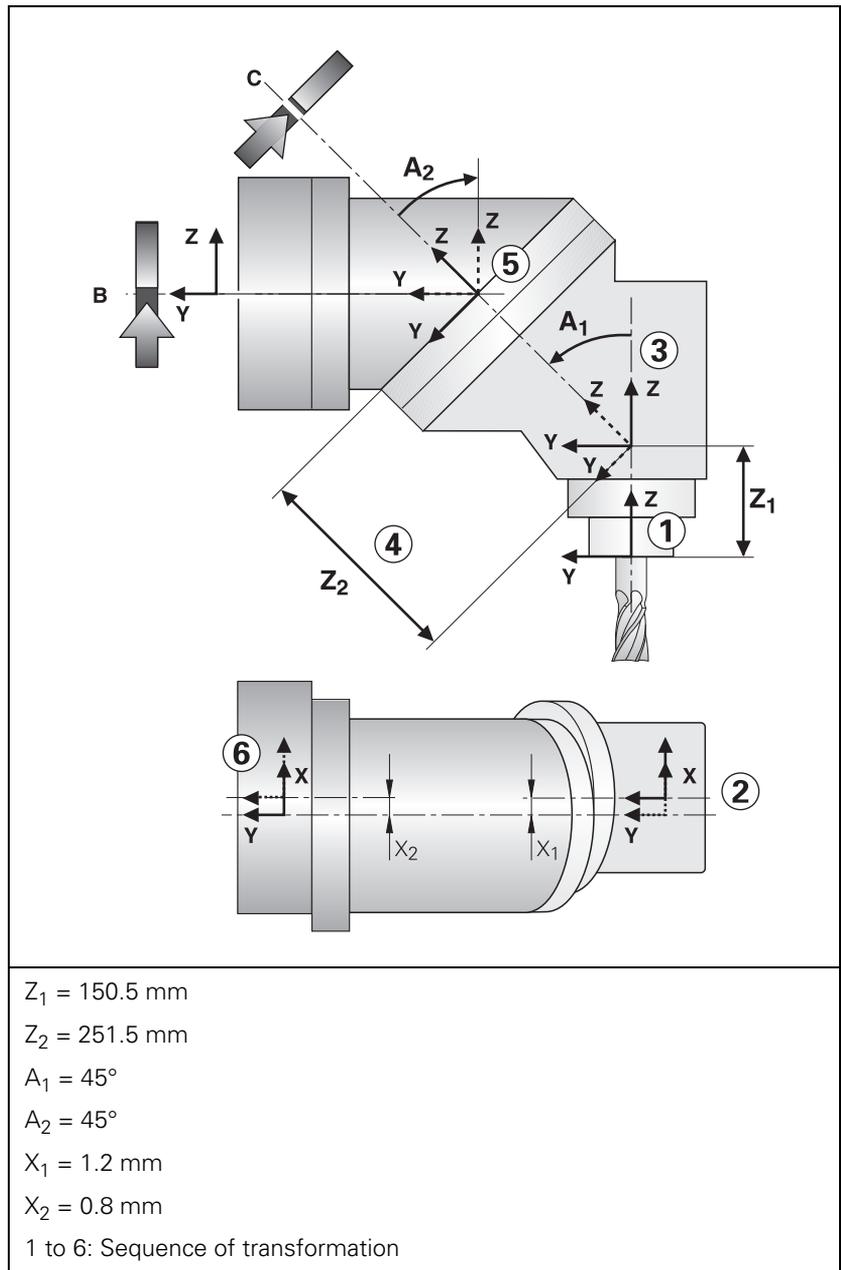
$Z_1 = 200.4 \text{ mm}$
 $Z_2 = 3.1 \text{ mm}$
 $X_1 = 201.5$
 $Y_1 = 1.9 \text{ mm}$
 1 to 4:
 Sequence of
 transformation

Description of the swivel head (without elements of the collision monitoring):

NR	KEY	AXIS	COORD	ON/OFF	FILE	DOC
0	Trans	Z	200.4			(Z1)
1	Trans	Y	-1.9			(Y1)
2	MachAxis	A				
3	Trans	X	201.5			(X1)
4	Trans	Z	3.1			(Z2)
5	MachAxis	B				
6	:					
:	:					
[END]						



Example 2: Double swivel head 45°

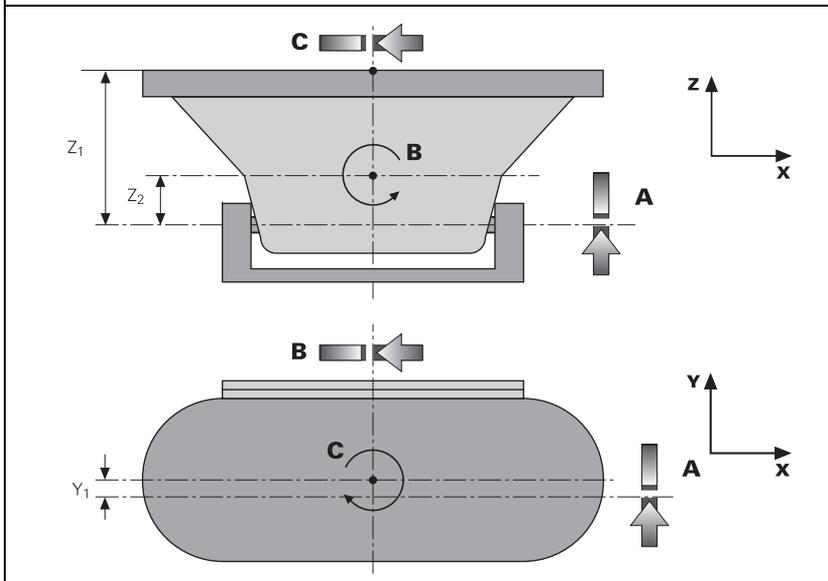
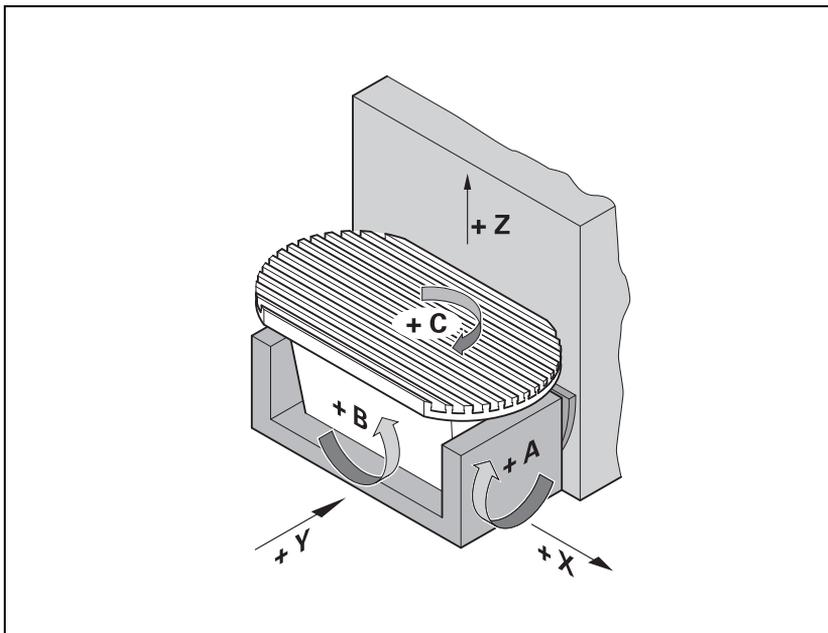


Description of the swivel head (without elements of the collision monitoring):

NR	KEY	AXIS	COORD	ON/OFF	FILE	DOC
0	Trans	Z	150.5			(Z1)
1	Trans	X	-1.2			(X1)
2	Trans	A	-45			(A1)
3	Trans	Z	251.5			(Z2)
4	MachAxis	C				Free tilting axis C
5	Trans	A	45			(A2)
6	Trans	X	0.8			(X2)
7	MachAxis	B				
	:					
	[END]					



Example 3:
Universal table
 (pitch, tilt, rotation)



$Y_1 = 2.7 \text{ mm}$
 $Z_1 = 331.3 \text{ mm}$
 $Z_2 = 125.9 \text{ mm}$

Coordinates (with respect to machine datum) of the center of rotation of table C when all tilting axes are in their home position:

$XR = 420.0 \text{ mm}$
 $YR = 151.2 \text{ mm}$
 $ZR = -395.4 \text{ mm}$

Description of the rotary table (without elements for collision monitoring) with assumed tool reference point:

NR	KEY	AXIS	COORD	DOC
0	Trans	X	420	Translation in X direction
1	Trans	Y	148.5	Translation in Y direction
2	Trans	Z	-190	Translation in Z direction
3	MachAxis	Z		
4	MachAxis	Y		
5	MachAxis	X		
6	MachAxis	B		
7	Trans	Z	-125.9	Translation in Z direction
8	MachAxis	A		
9	Trans	Y	2.7	Translation in Y direction
10	Trans	Z	331.3	Translation in Z direction
11	MachAxis	C		End of translation
[END]				

Selecting a geometry description in case of an error

In order for the machine operator to be able to select another geometry description in case of an error, abbreviations for the geometry descriptions can be entered in the **DOC** column of the assignment table.

To select another geometry description in case of an error:

- ▶ While in the **Programming and Editing** operating mode, press the MOD key.
- ▶ Enter the code number **KINEMATIC**.

A pop-up window appears with the abbreviation from the **DOC** column in the assignment table:

- ▶ Use the arrow keys to select the appropriate geometry description.
- ▶ Press the ENT key

The control resets and activates the selected geometry description.

6.5.3 Temperature compensation with tilting axes

A change in temperature always causes a change in length. For tilting axes, thermal growth of the spindle head must be compensated in the X, Y and/or Z axes.

There are two possibilities for temperature compensation:

- Permanently effective temperature compensation
 - by entering a formula in the **TEMPCOMP** column of the description table

In most cases, the formula to be used will be the formula for calculating a change in length: $\Delta l = l \cdot \Delta T \cdot \alpha$

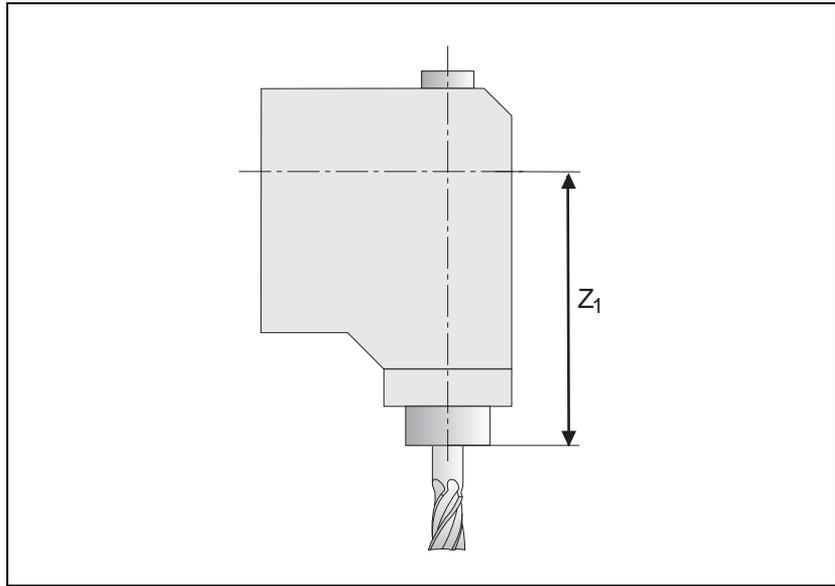
Δl : Change in length

l : Length

ΔT : Change in temperature

α : Coefficient of expansion (steel: $11.5 \cdot 10^{-6}$ 1/K)

Example:



$Z_1 = 300 \text{ mm}$ (at $20 \text{ }^\circ\text{C}$)

$\alpha_{\text{steel}} = 11.5 \cdot 10^{-6} \text{ 1/K}$ (coefficient of expansion of steel)

Wxxx: Temperature measured by a Pt100 thermistor

"TEMPCOMP" = $300 \cdot 11.5 \cdot 10^{-6} \cdot (W_{xxx} - 20)$

Better would be:

"TEMPCOMP" = $3.45 \cdot 10^{-3} \cdot (W_{xxx} - 20)$

If the front part of the spindle gets warmer by 40 K, it results in a spindle length growth of

$$\Delta l = 300 \text{ mm} \cdot 40 \text{ K} \cdot 11,5 \cdot 10^{-6} \frac{1}{\text{K}} = 0,138 \text{ mm}$$

Constraints on the entry of a formula

- Maximum length of a formula: 31 characters
- Up to 64 variables in total
- Mathematical operations in lowercase letters, variables in uppercase letters
- The following operations are permitted in a formula:
 - Addition +
 - Subtraction –
 - Multiplication *
 - Division /
 - Logarithm to the base of 10 log10
 - Exponent ^
 - Parentheses ()
 - Sine sin
 - Cosine cos
 - Tangent tan
 - Arc sine asin
 - Arc cosine acos
 - Arc tangent atan
 - Square root sqrt



Note

A faulty syntax of the formula is not detected in the kinematic tables (**TEMPCOMP** column).

Permanent temperature compensation

The permanent temperature compensation with the formula from the **TEMPCOMP** column is only effective if the description table is active. The algebraic sign of the compensation must match that of the axis error compensation in Words W576 to W584.

The variables are monitored every second and changes are reported to the position controller. The position controller uses the formula in the **TEMPCOMP** column and the current angle of the rotary axes to calculate the compensation values.

For temperature compensation, only rows with an entry in the **TEMPCOMP** column are considered, including the entry **0**. Any missing entry (NO ENT) interrupts temperature compensation at this point, meaning that the subsequent rows will not be taken into account. If rows 1 to 7 in the following example contained no entries instead of the entry **0**, the temperature compensation would only be applied to the Z axis. **The tilting axes would not be considered.** You can use this interruption of the temperature compensation to reduce processing time if, for example, a swivel head and tilting table follow each other in a table, but the temperature compensation is only to be applied to the swivel head. You would then make no entry for the tilting table in the **TEMPCOMP** column.

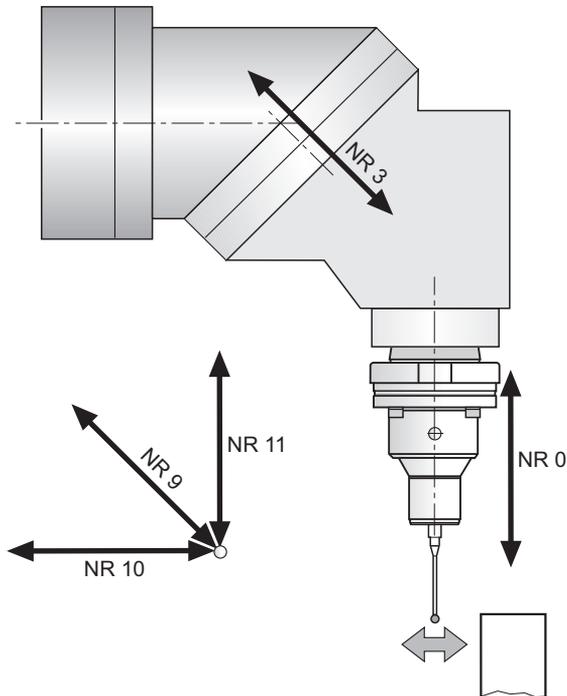
With Module 9040 or 9041, transfer value 8, the value of the temperature compensation can be determined.

The formula is only entered for the transformations where compensation is to occur.

Example of a description table with permanent temperature compensation for a 45° double swivel head (also applies to the **TEMPCOMP** column in the description tables in the new format):

NR	MP7510	MP7520	MP7530	TEMPCOMP	MP7550
0	4	0	+150,5	1.73e-3*(W486-20)	0
1	1	0	-1.2	0	0
2	8	0	-45	-45	0
3	4	0	+251,5	0.0115*(W486-20)	0
4	32	0	0	0	0
5	8	0	+45	+45	0
6	1	0	+0,8	0	0
7	16	0	0	0	0
8	0	0	0	0	0
9	1	0	0	0.0345*(W486-20)	0
10	2	0	0	0.0230*(W486-20)	0
11	4	0	0	0.0173*(W486-20)	0
[END]					

The arrows in the figure show the effective directions of the entries in the column **TEMPCOMP**. NR x shows the respective line number of the entries:



6.5.4 Changing the milling heads

In order to change the milling heads, some machine parameter values must be defined along with the tilting axis geometry (e.g. in the assignment table via the **SUBFILE1** or **SUBFILE2** entries). For this purpose a machine-parameter subfile can be entered in the **MPFILE** column. The machine parameter subfile contains the new axis configuration, the new axis motors, the assignments of the PWM outputs and the encoder inputs, etc. In this machine-parameter subfile there must be no machine parameters that provoke a control reset:

- ▶ In the standard machine parameter file, the bits in MP10 need to be set to 1 for all possible axes.



Note

If an axis deactivated in the standard machine-parameter file becomes activated in the machine-parameter subfile, the axis remains deactivated.

- ▶ Switch off drives for the affected axes.
- ▶ Choose a row in the assignment table in which a machine-parameter subfile is activated which sets the bits in MP10 to 0 for the affected axes. If the encoder of a **digital** axis needs to be disconnected, then the appropriate bit must also be set in MP20.x bit x = 0. The motor or the encoder must be defined in MP2200.x of the standard machine parameter file. This entry is retained even if the axis is deactivated.



Note

Encoders with EnDat interface can be disconnected and reconnected to the CC 6xxx during operation. It is not necessary to reboot the control.

- ▶ Change the milling head.
- ▶ Choose a row in the assignment table which contains a machine-parameter subfile for the new axes to set the bits in MP10 to 1 for the new axes. If an encoder on a **digital** axis was disconnected and reconnected, then MP20.x must be entered again correctly.
- ▶ Switch on drive for the new axis/axes.
- ▶ After the drive has been switched on, the affected motors should make at least one revolution.

6.5.5 KinematicsOpt

Accuracy requirements are becoming increasingly stringent, particularly in the area of 5-axis machining. Complex parts are required to be manufactured with precision and reproducible accuracy even over long periods.

KinematicsOpt (software option 48) is an important component that helps you meet these complex requirements: A 3-D touch probe cycle measures the rotary axes on your machine fully automatically, regardless of whether they are used in rotary-table or head configurations. A calibration sphere is fixed at any position on the machine table, and measured with a resolution that you define. In the cycle definition, you only have to define for each rotary axis the area that you want to measure.

From the measured values, the TNC calculates the static tilting accuracy. The software minimizes the spatial error arising from the tilting movement and, at the end of the measurement process, automatically saves the machine geometry in the respective machine constants of the kinematics table.

When using the cycles for KinematicsOpt, please also refer to the Touch Probe Cycles User's Manual.

Functions

- Testing the static tilting accuracy by probing different points on a calibration sphere with a 3-D workpiece touch probe.
- Optimizing the static tilting accuracy by adjusting the kinematics description.
- Backing up and restoring the kinematics data that can be changed with this software.

Prerequisites

- The software options 48 (KinematicsOpt) and 8 (software option 1) must be enabled.
- Feature Content Level (FCL) 3 or higher must be enabled.
- The 3-D touch probe used for the measurement must be calibrated.
- A calibration sphere with an exactly known radius and sufficient rigidity must be attached to the machine table.
- The kinematics must be described in the new table format (columns: **KEY**, **AXIS**, **COORD**, ...)
- The kinematics description of the machine must be complete and correct. The transformation values must be entered with an accuracy of approx. 1 mm.
- All machine axes involved must have adequate positioning accuracy.
- The geometry of the machine must have been measured.
- The machine datum (MP960.x) must be defined for the rotary axes (is not measured in the cycle).
- The machine parameters MP6600 to MP6602 must be defined.
- The cycles use the value entered in MP420 or MP430 to determine whether the respective axis is a rotary axis or a Hirth axis.
- Option 52 KinematicsComp, if rotational position errors of the rotary axes are to be compensated



Kinematics description

In the optimizing mode, the linear offsets (compensation translations) are adjusted to the kinematics description. Please note that two compensation transformations defining the position (not the angular position) of the rotary axis must be defined for every rotary axis. The TNC can find a suitable compensation translation even beyond two rotary axes. As soon as a translation is not parallel to the rotary axis to be optimized it can be accepted as a compensation translation.

If further axes lie between the rotary axis to be measured and the translation, then the translation must be parallel to the other rotary axes.

Transformations that do not have an entry in the COORD column are not changed by KinematicsOpt.

Example of a kinematic description with a B head and C table:

NR	KEY	AXIS	COORD
xx	Trans	X	+100.014
xx	Trans	Z	+299.951
xx	MachAxis	B	
xx	Trans	Z	-600.142
xx	Trans	X	+300.021
xx	Trans	Y	-251.191
xx	MachAxis	C	

This description is also possible:

NR	KEY	AXIS	COORD
xx	Trans	X	+100.014
xx	Trans	Y	-251.191
xx	Trans	Z	+299.951
xx	MachAxis	B	
xx	Trans	Z	-600.142
xx	Trans	X	+300.021
xx	MachAxis	C	

Compensation translation for C

Example of a kinematic description with a B head and optional C table:

NR	KEY	AXIS	COORD		COORD (compensations)
0	TOOLFILE			-->	No changes
2	CMO			-->	No changes
3	Trans	X	+0.01	-->	+0.006
4	Trans	Y	+0	-->	+0
6	Trans	Z	+250.02	-->	+250.034
7	MachAxis	B		-->	No changes
8	MachAxis	Z		-->	No changes
9	MachAxis	Y		-->	No changes
10	MachAxis	X		-->	No changes
11	SUBFILE1	X		-->	No changes
12	Trans	X	+0	-->	No changes
[END]					

SUBFILE1:

NR	KEY	AXIS	COORD		COORD (compensations)
0	Trans	Z	-655.045	-->	No changes
1	Trans	X	+440.01	-->	+440.006
2	Trans	Y	+250.02	-->	+250.034
3	CMO			-->	No changes
4	MachAxis	C		-->	No changes
5	CMO			-->	No changes
[END]					

Tolerance limit

In MP6600 you can enter the maximum permissible amount of change for the cycles used in KinematicsOpt. The amount of change prevents you from accidentally changing the machine kinematics too much with KinematicsOpt. Whenever optimization requires a change greater than what is permitted in MP6600, a message appears and the change must be confirmed with NC start.



Sphere radius

In MP6601 you can enter the maximum permissible deviation of the calibration-sphere radius. The sphere radius is measured and checked with KinematicsOpt. This monitoring feature also detects incorrect probing caused by contamination.



Danger

After optimization the position of the presets relative to the workpiece may have changed. It is essential that you check or reset the datum points.

MP6600

KinematicsOpt: Amount of change

Input: 0.010 to 1.000 [mm]
Default: 0.05

MP6601

KinematicsOpt: Radius deviation of the calibration sphere

Input: 0.010 to 0.100 [mm]
Default: 0.02

PLC positioning

Some rotary axes cannot be moved by the NC, e.g. Hirth axes that can be positioned only by the PLC.

In MP 6602, you can also enter an M number that is used for positioning all rotary axes. This M function will be executed instead of the positioning movements through the NC. The rotary-axis angles required for this are saved in the Q parameters Q120 to 122. As a result, a positioning movement must always be initiated by an M macro. This is a very simple example of how this M macro may look like:

```
BEGIN PGM MM
```

```
FN 19: PLC =+4 / Q120 ;the PLC positions the A axis
```

```
FN 19: PLC =+6 / Q122 ;the PLC positions the C axis
```

```
END PGM MM
```



Note

Only local Q parameters (QL) are allowed in the M macro and in the NC programs called by the M macro. Otherwise, there is a great potential danger of collision during the further execution of the cycle.

The cycles cannot verify that the correct position has been reached.

Be sure to remember while programming that the rotary axes must be pre-positioned to 0° degrees before the execution of the cycles if you use a PLC positioning movement. Since, in this case, there is also a great danger of collision if automatic presetting is used, this prerequisite must be confirmed by the operator.

This macro is called from all KinematicsOpt cycles when rotary axes are positioned. If an M function macro is specified, the target coordinates, not the rotary-axis positioning movement, are saved in the Q parameters 120 to 122 (axis angles) and the M function is called. The M function must ensure the correct positioning of the rotary axes to the nominal positions, as defined in Q120 to Q122.

Please keep in mind that you must not perform any transformations in the macro, and that you may only use QL parameters (no QS parameters from 0 to 99)!

MP6602 KinematicsOpt: M-function macro for positioning the rotary axes

Input: 0 to 999
-1: Function inactive

Deselect rotary axis

Through measuring points 0 – 12 it is possible to "hide" a rotary axis (0 = axis). The angle of incidence of this axis is considered nevertheless.



Note

Please note that all transformations are effective incrementally. Compensation values for one axis lead to a corresponding shift in all subsequent positions of the axes. By deselecting an axis, it is completely removed from the optimization with KinematicsOpt.

Positioning direction

The positioning direction of the rotary axis to be measured is determined from the start angle and the end angle. Because $+270^\circ$ is identical to -90° for example, the same angles can result in different measuring positions.

Examples:

Start angle = -90°

End angle = $+90^\circ$

Measuring points = 4

Angle increment = (end angle – start angle) / (measuring points – 1)
= $(90^\circ - -90^\circ) / 3$
= $+60^\circ$

Therefore: P1 = -90° ; P2 = -30° ; P3 = $+30^\circ$; P4 = $+90^\circ$

Start angle = $+270^\circ$

End angle = $+90^\circ$

Measuring points = 4

Angle increment = (end angle – start angle) / (measuring points – 1)
= $(90^\circ - 270^\circ) / 3$
= -60°

Therefore: P1 = -90° ; P2 = -120° ; P3 = -210° ; P4 = -270°



Backlash and positioning errors

You can activate backlash measurement for all rotary axes in the input parameter Q432 during programming. You must enter an angular value that is greater than the actual backlash of the rotary axes.

Since the accuracy of this information depends on the measuring circle radius, the text log indicates the measurement uncertainty in degrees per 1 μm of system uncertainty:

The system uncertainty includes at least the position error of the linear axes and the scatter of the touch probe. However, this data is not available to the measuring cycle, and must therefore be estimated by the operator (see example below).

Additionally, the positioning error of the rotary axis is always determined. The calculated positioning error is also subject to this measurement uncertainty. In order to realistically measure the positioning accuracy, it is recommended that the angle increment when measuring the rotary axis is defined to be clearly below 90° (recommendation: less than 45°). If the angle increment is too large, then the positioning error is not calculated.

As a rule, the measurement uncertainty increases as the measuring circle radius increases.

Example for calculating the measurement uncertainty (the greater the calculated value of the measurement uncertainty, the less informative the measurement is):

- Position uncertainty of each linear axis: 10 μm
- Uncertainty of touch probe: 2 μm
- Measurement uncertainty: 0.0002 degrees/ μm
- System uncertainty = $\text{SQRT}(10^2+10^2+10^2+10^2) = 17.44 \mu\text{m}$
- Measurement uncertainty = $0.0002 \text{ }^\circ/\mu\text{m} * 17.44 = 0.0034^\circ$

Log function

The Cycles 451 and 452 save log files in the PLC:\ directory, from which you can see the results of the individual measuring points. TNCscope can be used for a graphic evaluation (import the table). On the Extras -> Info tab, you can display additional information about the measurement and a short legend of the individual measured data.

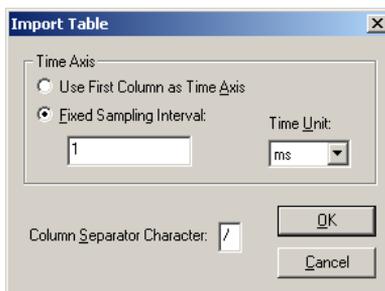
The files are saved under PLC: as TCHPR45x_<axis name>.A. The columns created may vary depending on the conditions for measurement.

Data with the designation **REF** are to be understood with respect to the machine coordinate system. Data with the designation **ERROR** are measured errors in the coordinate system of the rotary axes. Data with the designation **OPT** are the errors to be expected in the coordinate system of the rotary axes after an optimization.

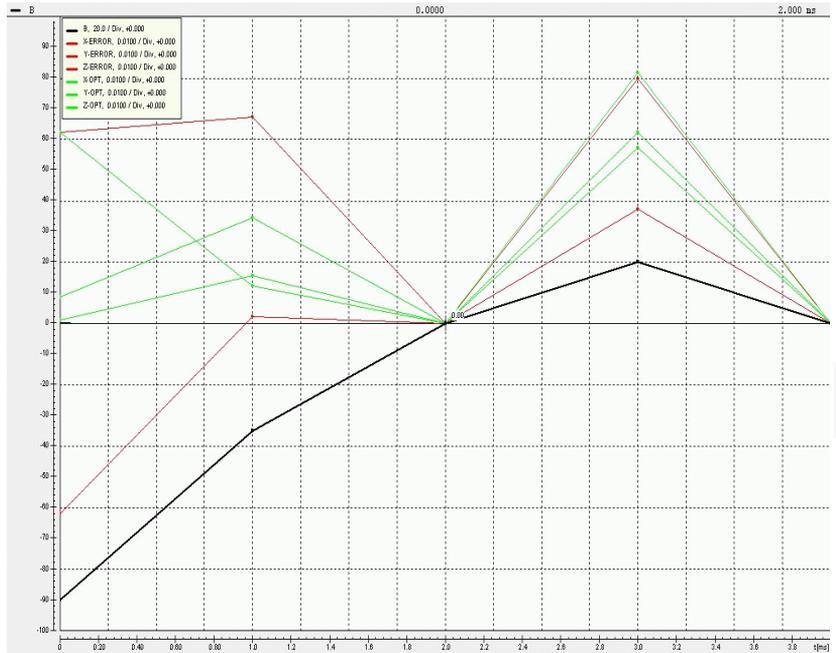
The data can be imported for evaluation (e.g. in TNCscope, Excel).

TNCscope: Open the file in TNCscope by selecting:

- ▶ File – **Import Table**. Then select the file to be opened.
- ▶ The **Import Table** dialog box opens.
- ▶ Select **Fixed sampling interval** and enter the value 1 ms in the Time unit field.
- ▶ Select "/" as the column separator and confirm your entry with OK.



Measured and optimized deviations (simulated measured data / displayed with TNCscope):



Optimized deviations as a function of the position of the rotary axis (simulated measured data / displayed with TNCscope):



■ Programming station

The cycle recognizes whether a programming station is being used. Because it is not possible to measure real data on a programming station, the measured data are simulated.

Touch probe cycle 450 (SAVE KINEMATICS)

Use this cycle to save the values of the active kinematics description in order to be able to restore the active kinematics description if required. For a detailed description of Cycle 450 SAVE KINEMATICS, refer to the Touch Probe Cycles User's Manual.



Note

- As a machine tool builder, you can save the shipping data for example. You enter the standard PLC code number (807667) and save the data to any memory location.
- When an NC software version is installed, the backup file of the kinematics description that was created with Cycle 450 is overwritten. After installing the software, you must save the kinematics description again with Cycle 450 if required.

Meanwhile, all axis and transformation entries containing numerical values are saved. Along with the entries in the column **COORD**, now also numerical entries in the optional column **COMPTAB** are saved and restored. When restoring the COMPTAB files, the compensation limit from MP6600 is considered, independently of whether it is a translation or a rotation.

A link to a column in a compensation file can be entered in the column **COMPTAB**. These non-numeric entries are neither saved nor restored. If, after backing up, such an entry is manually replaced by a numerical value, then this manually entered value is not changed when the old data are restored (the manual change is kept).

The text log also contains the COORD entries that were backed up. Upon restoration, the text log shows the previous and the new, restored entries together. The entries in the **COMPTAB** column do not appear in the log.



Touch probe cycle 451 (MEASURE KINEMATICS)

The MEASURE KINEMATICS cycle enables you to check and, if required, optimize the kinematics of your machine. Use the 3-D TS touch probe to measure any calibration sphere that you have attached to the machine table.

Angle optimization

In mode 1, only the position of a rotary axis is adapted. In mode 2, the position and angular orientation are adapted.

The general angular position of a rotary axis can be calculated if a minimum of two measuring points were probed per axis. Make sure that the measuring circle radius is not too small. Otherwise, the repeatability of the system can have a major influence on the result. The optimization of the angular position of a rotary axis only makes sense if the angles within the desired traverse path of the linear axes are relatively constant. If there are major changes in the angles at different positions of the linear axes due to component errors of the linear axes, the optimization of the angular position of the rotary axes is not reasonable. An adjustment of the angular position can contribute to improving the accuracy, especially on small and compact machines.

The ideal coordinate system is determined by the machine base (linear axes). The angles are calculated based on this system. The viewing direction depends on whether the rotary axis considered is a head axis or a table axis.

KinematicsOpt can perform a compensation to adjust the angular position of the rotary axes. This is only possible in conjunction with software option 52 KinematicsComp, and if the kinematics model has been prepared appropriately. This is done by changing the values of the rotation entries in the COMPTAB column. The values are only changed if the field already contains a numerical value (floating-point number). This prevents possible entries of a component-error compensation through KinematicsComp from being affected.

Angular-position compensations do not only influence the measured rotary axis, but also the complete transformation chain. After the affected rotary axis has been changed, KinematicsOpt therefore tries to undo the changes made to angles, which is desirable in some cases. This is possible within the scope determined by the correspondingly defined tolerances for the transformation chain.

Example of AC table:

NR	DOC	KEX	AXIS	COORD	COMPTAB
0		ToolFile			
1		ToolAxis	Z		
2		MachAxis	Z		
3		MachAxis	X		
4		MachAxis	Y		
5		MachBase			Ideal coordinate system
6	Y0A	Trans	Y	-300.025	
7	Z0A	Trans	Z	-449.881	
8	B0A	Trans	B		+0.00131
9	C0A	Trans	C		-0.01005
10		MachAxis	A		
11	-C0A	Trans	C		+0.01005
12	-B0A	Trans	B		-0.00131
13	Y0C	Trans	X	+350.152	
14	Y0C	Trans	Y	-49.982	
15	Table surface	Trans	Z	+50	
16	A0C	Trans	A		-0.00115
17	B0C	Trans	B		-0.00281
18		MachAxis	C		

After the A axis has been measured, its angular position is determined and optimized. Adjustments are only possible through compensations in the **COMPTAB** column if a numerical value is entered in the corresponding lines. This is the case in lines 8 and 9 of this example, where the position errors B0A and C0A can be defined. After the A axis has been defined, Cycle 451 tries to cancel the rotations, which have just been performed. This is done by the compensations defined in lines 11 and 12 immediately afterwards. The two rotations are arranged inversely to the way they are arranged in lines 8 and 9. This simplifies traceability, but it is not absolutely necessary. After the angular position has been adjusted, a measurement is performed, in which the position of the rotary axis (Y0A and Z0A) is determined. The position is corrected in lines 6 and 7.



Then the C axis is measured and its angular position is optimized. The position errors A0C and B0C are corrected through the compensations in lines 16 and 17. Because no further elements that make an inverse transformation of the rotations possible are listed after the C axis, the workpiece coordinate system remains rotated from the ideal system (machine base).

Example of AC head:

NR	DOC	KEX	AXIS	COORD	COMPTAB
0		ToolFile			
1		ToolAxis	Z		
2	YOA	Trans	Y	+50.013	
3	ZOA	Trans	Z	+300.815	
4		MachAxis	A		
5	B0A	Trans	B		+0.0030
6	C0A	Trans	C		-0.0009
7	X0C	Trans	X	+0.053	
8	Y0C	Trans	Y	-0.092	
9	-B0C	Trans	B		+0.00337
10	-C0C	Trans	C		+0.0103
11		MachAxis	C		
12	C0A	Trans	C		-0.0103
13	B0A	Trans	B		-0.00337
14		MachAxis	Z		
15		MachAxis	X		
16		MachAxis	Y		
17		MachBase			Ideal coordinate system

During measurements in mode 2, the angular errors B0A and C0A in lines 5 and 6 of the COMPTAB column are compensated. Inverse transformation of the two angular errors B0A and C0A is not possible, because no corresponding rotations are defined. This means that the tool coordinate system is rotated from the ideal system. After the angular position has been adjusted, a measurement is performed, in which the position of the rotary axis (YOA and ZOA) is determined. The position is corrected in lines 2 and 3.

Then the C axis is measured and its angular position is optimized. The position errors A0C and B0C can be corrected in lines 12 and 13 through the compensations in the COMPTAB column. An inverse transformation of both angular errors is possible in lines 9 and 10. As a result, further changes to other axes (also A) can be ruled out.

Example of B head (45°) and C table:

NR	DOC	KEX	AXIS	COORD	COMPTAB
0		ToolFile			
1		ToolAxis	Z		
2	X0B	Trans	X	+50.013	
3	Z0B	Trans	Z	+300.815	
4		Trans	A	+45	
5		MachAxis	B		
6	A0B	Trans	A		-0.0027
7	C0B	Trans	C		+0.0009
8		Trans	A	-45	
9		MachAxis	Z		
10		MachAxis	X		
11		MachBase			Ideal coordinate system
12		MachAxis	Y		
13	X0C	Trans	X	+349.217	
14	Y0C	Trans	Y	-451.012	
15	Table surface	Trans	Z	-650.514	
16	A0C	Trans	A		-0.0005
17	B0C	Trans	B		-0.0011
18		MachAxis	C		

During measurements in mode 2, the angular errors A0B and C0B in lines 6 and 7 of the COMPTAB column are compensated. Inverse transformation of the two angular errors A0B and C0B is not possible, because no corresponding rotations are defined. This means that the tool coordinate system is rotated from the ideal system. After the angular position has been adjusted, a measurement is performed, in which the position of the rotary axis (X0B and Z0B) is determined. The position is corrected in lines 2 and 3.

Then the C axis is measured and its angular position is optimized. The position errors A0C and B0C can be corrected in lines 16 and 17 through the compensations in the COMPTAB column. An inverse transformation is not defined, and therefore the workpiece coordinate system is rotated from the ideal system.

For a detailed description of Cycle 451 MEASURE KINEMATICS, refer to the Touch Probe Cycles User's Manual.



Touch probe cycle 452 (PRESET COMPENSATION)

Touch probe cycle 452 can be used to optimize the kinematic transformation chain of the machine. Then the TNC corrects the workpiece coordinate system in the kinematics model in such a way that the current preset is in the center of the calibration sphere after optimization.

Cycle 452 corrects the transformations up to the last active rotary axis (like Mode 1 of Cycle 451).

After the last measured rotary axis, it is attempted to set the preset at the center of the calibration sphere via the kinematics transformations. The next possible transformations for this compensation are searched for after the last measured rotary axis of the transformation chain.

An error message is output if the preset cannot be completely compensated.

Cycle 452 can be used, for example, to compensate specific thermal behavior, and also to adapt the kinematics of tool changer heads.

Example 1 for the application of Cycle 452:
Adjust tool changer head (A axis)

SUBFILE: Head 1		SUBFILE: Head 2	
Trans	Y	Trans	Y
Trans	Z	Trans	Z
MachAxis	A	MachAxis	A
Trans	X	Trans	X
Trans	Y	Trans	Y
Trans	Z	Trans	Z
MAINFILE			
MachAxis	C		
MachAxis	Y		
MachAxis	X		
MachAxis	Z		

Once a head has been optimized (A and C axis, SUBFILE Head 1 and MAINFILE), it must be used to set the preset in the center of the calibration sphere (Cycle 451). Then Cycle 452 is simply used in combination with the second head (SUBFILE: Head 2) to optimize the A axis with preset compensation.

The transformations shown in green are recognized as compensation transformations, and the preset is thus placed at the sphere center.

Example 2 for the application of Cycle 452:
Compensation of thermal behavior

Trans	X
Trans	Y
Trans	Z
MachAxis	Y
MachAxis	X
MachAxis	Z

A present compensation is performed in this 3-axis machine. In the Y and Z directions the preset is returned to the position in the machine coordinate system at which it was located for setting the reference point of the calibration sphere.

The compensation translations need not be aligned according to the machine coordinate system. The three translations only need to make possible a shift in any desired spatial direction.

Example of angle head:

NR	DOC	KEX	AXIS	COORD	COMPTAB
0		ToolFile			
1		ToolAxis	Z		
2		Trans	X	+50.013	Translations for preset compensation
3		Trans	Y	+0.024	
4		Trans	Z	+300.815	
5		Trans	B	-90	Fixed rotation
6		MachAxis	Z		
7		MachAxis	Y		
8		MachAxis	X		
9		MachBase			Ideal coordinate system

Example of non-perpendicular preset compensation

NR	DOC	KEX	AXIS	COORD	COMPTAB
0		ToolFile			
1		ToolAxis	Z		
2		Trans	X	+0.032	Translations for preset compensation do not result in a Cartesian (perpendicular) coordinate system
3		Trans	Z	+299.941	
4		MachAxis	B		
5		Trans	Z	+200.018	
6		Trans	B	+33	
7		Trans	Z	+50	
8		Trans	Y	0	
9		Trans	B	-33	
10		Trans	X	-0.026	
11		Trans	Y	-0.012	
12		MachAxis	C		
13		MachAxis	Z		
14		MachAxis	Y		
15		MachAxis	X		
16		MachBase			Ideal coordinate system



6.5.6 "Tilt working plane" feature

The user defines the position of the working plane in Cycle 19 "Tilted Working Plane." Then the iTNC performs a coordinate transformation.

With the 3D ROT soft key you can activate the tilted working plane separately for the MANUAL and PROGRAM RUN operating modes.

If bit 2 is set in MP7507, the 3D ROT soft key will be available in the Test Run mode after you have pressed the MOD key. Here you can enter (in the same manner as in the Manual Operation mode) the tilt angles in the axes A, B and C, and you can activate/deactivate tilting for the test run. Additionally, a soft key is available to load the status of the machine (active kinematics, tilt angles + active in program run). If bit 2 is set in MP7507, tilting is not deactivated anymore when a new program is selected in Test Run mode. The axis that is set for manual probing will be used as the tool axis until the first tool call. This can cause the error message "Angle cannot be calculated" if the tool axis and the selected kinematics model cannot be used together.

With MP7500 you can define the function of the tilted working plane cycle. With **FN18: SYSREAD ID290 NR2** you can request the values of the individual bits from MP7500.

Assignment of input values (Cycle 19)

With MP7500 bit 1 you define whether the input applies to the position of the tilted axes (bit 1 = 0) or the position of the working plane (bit 1 = 1).

If the input value applies to the position of the working plane, the iTNC calculates the position of the tilting axes and saves the coordinates in Q parameters:

- Q120: Coordinate of the A axis
- Q121: Coordinate of the B axis
- Q122: Coordinate of the C axis

With **FN17:SYSWRITE ID990 NR5 IDX5** you can determine if a principal axis is shown on top of another principal axis in an untilted coordinate system due to a tilt motion.

Automatic positioning

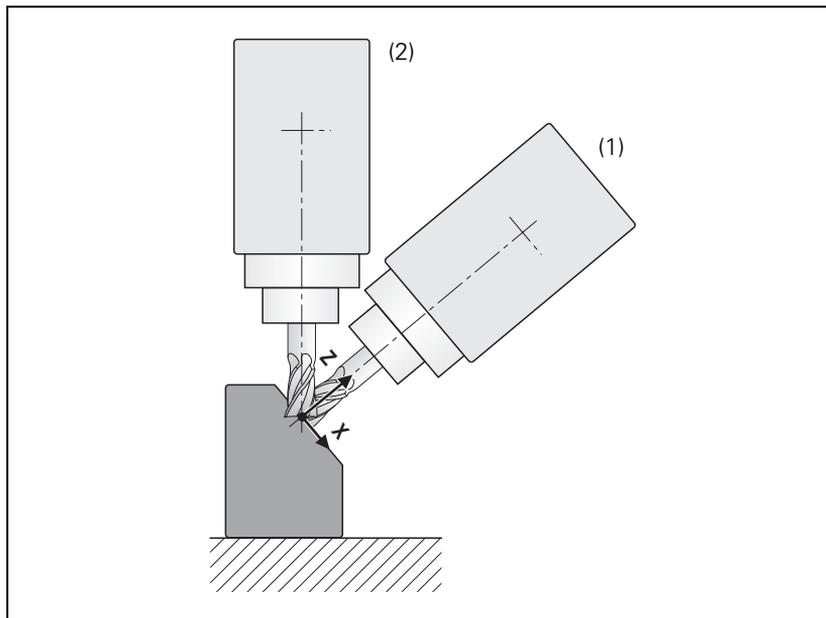
After the coordinate transformation, the Z axis remains parallel to the tool axis, perpendicular to the X/Y plane. With MP7500 bit 2 you define whether the "tilted working plane" function automatically positions the tilting axes (bit 2 = 1). In this case the user can enter the feed rate and setup clearance in the cycle.

The iTNC then moves automatically to the setup clearance and interpolates the swivel and principle axes so that the tool point remains in the same position in the tilted coordinate system.

With MP7682 bit 9 you control whether the status of the **Tilt working plane** function in an NC program is applied to the **Manual** operating mode during a program interruption.

Closed-loop axes:

During "datum setting" for X, Y and Z, the datum is recalculated back to the home position of the tilting element when "tilted working plane" is **active** (1). So when "tilted working plane" is **inactive** and the tilting element is in its home position, the tool is positioned at the datum set while "tilted working plane" was **active** (2).



Behavior during datum setting can also be influenced via MP7500 bit 5:

MP7500 bit 5 = 0 (is entered in the kinematics description table)

During datum setting in X, Y and Z with an **active** tilted working plane, the current rotary-axis coordinates are checked to see if they are correct for the tilt angles, and with an **inactive** tilted working plane the rotary axes are checked to see if they are at 0. For datum setting with an **active** tilted working plane, the corresponding angles must be entered under 3-D ROT.

MP7500 bit 5 = 1 (is entered in the kinematics description table)

It can happen with tilting elements with Hirth couplings that by locking the Hirth coupling, the actual value of the encoder will no longer exactly agree with the mechanical position of the tilting element. If this happens, the nominal values should be used to calculate the various presets (MP7682 bit 1). If problems continue to occur, MP7500 bit 5 should be set to 1. The checking described in MP7500 bit 5 = 0 does not take place. The tilt angles entered under 3-D ROT are used to calculate the datums in X, Y and Z.

With MP7682 bit 1 you define whether the nominal or the actual values are used to calculate the presets during "datum setting" (is valid for MP7500 bit 5). Also see "Special case: Reference-point setting with the PLANE function, Hirth axes and M114" on page 763.

Open-loop axes:

The user must enter the current positions of the tilting axes by using the 3-D ROT soft key.



Note

In the combination of coordinate transformation cycles, note the sequence of activation and deactivation.

Special case: Reference-point setting with the PLANE function, Hirth axes and M114

If rotary axes with Hirth coupling are positioned via PLC or NC, then as mentioned above, angles can only be entered according to a certain grid. Machining can be performed in any plane by tilting the plane with the PLANE function and programming the rotary axis grid coordinates with M114 (automatic compensation of the machine geometry when working with tilting axes).

It is possible to save and set reference points in this state.

The iTNC then uses the nominal positions of the rotary axes programmed in M114 to calculate the reference point. With MP7493 the machine manufacturer can enter a maximum deviation of the rotary axes based on the nominal position resulting from the orientation of the working plane. The control accepts this deviation when setting reference points and when measuring with M114. The default value for this machine parameter is 0.005. The iTNC uses the current ACTUAL/NOMINAL positions (MP7682, bit 1) of the rotary axes to calculate the reference point.



Note

Please note that in this procedure the tool might not be perpendicular to the tilted working plane.

MP7493 **Maximum deviation of the current tool orientation relative to the tool axis when setting a reference point with the PLANE function with M114**

Input: 0.0000 to 30.0000 [degrees]
Default: 0.005

Spatial angle C \neq 0

On machines with C tables and tool axis Z, the spatial angle C \neq 0 (with A = 0 and B = 0) can be realized through a rotation of the coordinate system or a rotation of the table:

- ▶ With MP7500 bit 6 = 0, the spatial angle C is realized through a rotation of the coordinate system.
- ▶ With MP7500 bit 6 = 1, the spatial angle C is realized through a rotation of the table. At the same time, the angle is saved in Q122. This makes it possible, for example, to machine a workpiece in the X/Y plane by always using the same axis for paraxial linear blocks.

Interrogating the 3-D ROT data via PLC

PLC Module 9045 makes the relevant data for the "Tilt working plane" function available to the integrated PLC as well. The following data are available:

- Tilt angles (A, B, C)
- Tilted axes (A, B, C)
- In which operating mode the "Tilt working plane" function is active

When the 3-D ROT data are interrogated via Module 9045, the data are entered in four sequential double words beginning from the given starting address [n]. The returned bit mask indicates which tilt angles are possible as a result of the currently active kinematics configuration.



Note

Ensure that the addresses are available on the control, and that the given target address is a double word address.

The data are output in the following format:

- D[n+0]: Tilt angle A (unit 0.0001°)
- D[n+4]: Tilt angle B (unit 0.0001°)
- D[n+8]: Tilt angle C (unit 0.0001°)
- W[n+12]: Currently tilted axes (bit-encoded)
Bit 0: Axis A tilted
Bit 1: Axis B tilted
Bit 2: Axis C tilted
- W[n+14]: "Tilt working plane" is active in operating mode
Bit 0: Tilting active in **Program Run** operating mode
Bit 1: Tilting active in **Manual** operating mode

Module 9045 Read the 3-D ROT data

Call:

PS B/W/D/K <Starting address as double-word number [n]>

CM 9045

Error code:

Marker	Value	Meaning
M4203	0	3-D ROT data read
	1	3-D ROT data was not read



Conditions and constraints

Conditions:

- The display position in the status window is referenced to the tilted coordinate system.
- In the combination of coordinate transformation cycles the sequence of activation must agree with the sequence of deactivation.
- The tool radius compensation in the working plane and the tool length compensation parallel to the tool axis is active.
- For machining with tilting tables, the coordinate system remains parallel to the machine coordinate system.
- If MP7500 bit 1=1 (spatial angle), Module 9045 shows all axes that will be tilted.
- If MP7500 bit 1=0 (axis angle), Module 9045 shows all axes that are actually tilted.

Constraints:

- PLC positioning movements are always parallel to an axis of the machine coordinate system (Cycle 19 has no influence).
- A datum shift via PLC also works with the "tilted working plane" function.
- The axis designations for the tilting axes are limited to A, B and C. Each designation can be used only once.
- With an active Cycle 19 "Tilted working plane," it is not possible to position with M91 or M92.
- Please note that the "Tilt working plane" function remains active even after the cancelation of a block scan, the end of an NC program or a reset, until it is explicitly deactivated. In these cases, the tilted plane should be specifically deactivated by means of a suitable Cancel macro in order to avoid machine movements that the user does not expect.

If the position of the working plane is entered, only the following tilting axes (with tool axis Z) are permissible:

- Double swivel head 45°: Axis sequence A fixed; B or C variable; A fixed; B or C variable
- Rectangular double swivel head: Axis sequence A or B variable; C variable
- Rotary or tilting table: Axis sequence C variable; A or B variable
- Swivel head and rotary table: Axis sequence A or B variable; C variable
- Tilting table 45°: Axis sequence C variable; A fixed; B variable; A fixed
- Rectangular double swivel head: Axis sequence A variable; B variable
- Universal swivel head: Axis sequence A fixed; B -90°; A variable; B +90°; A fixed; C variable
- Swivel head and rotary table: axis sequence B variable, A variable
- Swivel head and rotary table: Axis sequence C fixed, A fixed, B fixed -90°, A variable, B fixed +90°, A fixed, C fixed, C variable

With tool axis Y:

- Rotary or tilting table: Axis sequence B variable; A variable
- Double swivel head 45° and rotary table: Axis sequence A fixed; C variable; A fixed; B variable
- Rotary or tilting table: Axis sequence A or C variable; A or C variable
- Swivel head 45° and rotary table: Axis sequence A +45°, B variable, A -45°, C variable

With tool axis X:

- Universal swivel head: Axis sequence B fixed; A variable; B fixed; C variable



MP7500

Format:

Input:

“Tilt working plane” (inactive preset table)

%xxxxxxxx

Bit 0 – Switch-on "tilted working plane" function

0: Off

1: On

Bit 1 –

0: Angles correspond to the position of the tilting axes of the head/table

1: Angles correspond to the spatial angle (the iTNC calculates the position of the tilted axes of the head/table)

Bit 2 –

0: The tilting axes are not positioned with Cycle 19

1: The tilting axes are positioned with Cycle 19

Bit 3 –

0: The current tilting-axis position is taken into account with respect to the machine datum

1: The 0° position is assumed for the first rotary axis

Bit 4 –

0: Compensate mechanical offset during exchange of the spindle head when calling M128, M114, TCPM or "tilted working plane"

1: Compensate mechanical offset during PLC datum shift

Bit 5 –

0: The current tilting-axis position is taken into account with respect to the machine datum

1: The tilting-axis position that was entered with the 3-D ROT soft key applies

Bit 6 –

0: Spatial angle C is realized through a rotation of the coordinate system

1: Spatial angle C is realized through a rotation of the table

Bit 7 –

0: The current tilting-axis position is taken into account with respect to the machine datum

1: The active tilting-axis position is

a) derived from the tilting angles in the 3D ROT window if manual tilting is active

b) derived from the reference coordinates of the rotary axes if tilting is inactive

Bit 8 –

0: The tilting axis positioning is considered depending on bit 3, bit 5 and bit 7

1: If manual tilting is active, the datum to be set for the principal axes X, Y and Z is recalculated back to the home position of the tilting element

MP7500 "Tilt working plane" (active preset table)

As of SW version:340 422-01, 340 480-02

Format: %xxxxxxx

Input: Bit 0 – Switch-on "tilted working plane" function
0: Off
1: On
Bit 1 –
0: Angles correspond to the position of the tilting axes of the head/table
1: Angles correspond to the spatial angle (the iTNC calculates the position of the tilted axes of the head/table)
Bit 2 –
0: The tilting axes are not positioned with Cycle 19
1: The tilting axes are positioned with Cycle 19
Bit 3 – No function
Bit 4 – No function
Bit 5 – Test of the tilting axis during "datum setting" in X, Y and Z
0: Current tilting-axis position must fit to the defined tilting angles
1: No test
Bit 6 –
0: Spatial angle C is realized through a rotation of the coordinate system
1: Spatial angle C is realized through a rotation of the table
Bit 7 – No function
Bit 8 – No function

MP7510 Transformed axis

Format: %xxxxxx

Input: 0: End of the transformation sequence
Bit 0 corresponds to axis X
Bit 1 corresponds to axis Y
Bit 2 corresponds to axis Z
Bit 3 corresponds to axis A
Bit 4 corresponds to axis B
Bit 5 corresponds to axis C

MP7510.0–14Transformation 1 to transformation 15

MP7520 Additional code for transformation

Format: %xx

Input: Bit 0 – Tilting axis
0: Swivel head
1: Tilting table
Bit 1 – Type of dimension in MP7530.x
0: Incremental dimension for swivel head
1: Absolute with respect to the machine datum for tilting table

MP7520.0–14Transformation 1 to transformation 15



MP7530 Type of dimension for transformation

Input: Entry of a formula is possible, see page 479

0: Free tilting axis

MP7530.0–14 Transformation 1 to transformation 15



Note

MP7530 cannot be overwritten with Module 9031 (overwrite machine parameters), since the MP contains a string, but the module transfers an integer value.

MP7550 Home position of the tilting element

Input: –99 999.9999 to +99 999.9999

MP7550.0 A axis

MP7550.1 B axis

MP7550.2 C axis

MP7682 Machine parameter with multiple function

Input: Bit 1 – Reference value for calculating the preset during "datum setting"

0: Actual value is calculated

1: Nominal value is calculated

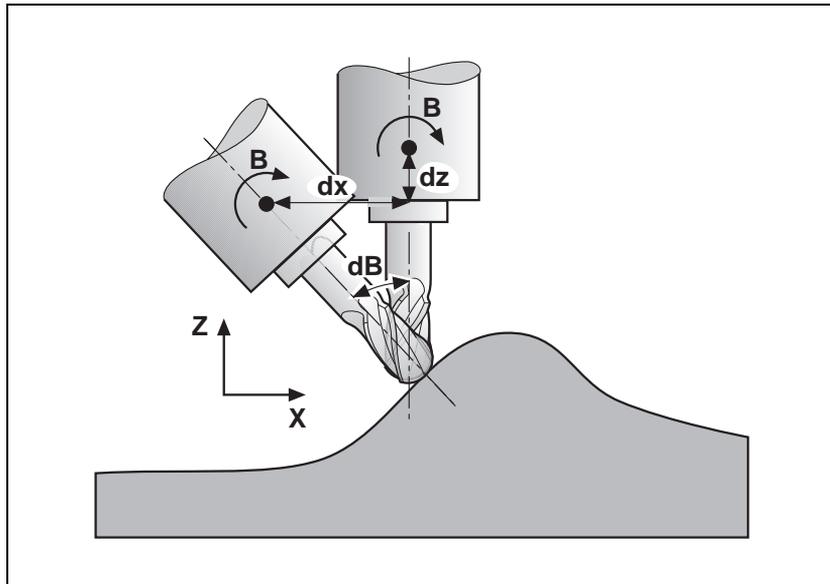
Bit 9 – Apply tilting of the working plane

0: The status of the **Tilt working plane** function is not applied to the **Manual** operating mode during a program interruption (behavior until now).

1: The status of the **Tilt working plane** function is applied to the **Manual** operating mode during a program interruption.

6.5.7 Automatic compensation of offset for tilting axes

Unlike the "tilted working plane," here the coordinate system is not tilted. With M114, M128 or TCPM, the iTNC compensates the offset of the tool that results from tilting the axes. The tool tip is always located on the programmed nominal coordinates.



The iTNC can run a 3-D length compensation; the radius compensation must be performed by the CAD system or the postprocessor. If the iTNC compensates the tool length, then the programmed feed rate refers to the tool point. Otherwise it refers to the tool datum.

Miscellaneous function M114

Automatic compensation with M114:

- Linear and rotational movements are superimposed. The resulting contour deviations depend on the length of the linear interpolation.
- When the table is rotated, the coordinate system is rotated against the machine coordinate system. **The iTNC does not take this into account.**
- M114 can be used with non-controlled tilting axes or PLC tilting axes. In this case, the current tilting angle and the tilting axis are entered in the NC block behind M114.
- M114 is effective locally in cycles, i.e. the function is canceled before the return to the main program. If you want M114 to also be effective in the main program, you must use **FN17: SYSWRITE ID420 NRO IDX0 = 0** (globally effective coordinate transformation).

Miscellaneous function M128, TCPM

Automatic compensation with M128 or TCPM:

- Linear and rotational movements are superimposed. The resulting contour deviations are compensated.
- When the table is rotated, the coordinate system is rotated against the machine coordinate system. **The iTNC takes this into account.**
- M128 and TCPM remain in effect even after a change in operating modes. This means that the axis can be moved with the compensated machine geometry in Manual mode with the axis direction keys, or in the **E1 . Handwheel1** mode.
- With the miscellaneous function M118, the handwheel positioning movements can be superimposed on the program run movements. The iTNC automatically performs the compensating movements in the principal axes.

A transitional element is inserted at non-tangential contour transitions when positioning with rotary axes. However, W1026/W1104 (axes in position) is not set and axes will not be clamped. This problem can be solved with M134 (exact stop at non-tangential contour transitions when positioning with rotary axes):

- ▶ Program M134 in the NC program or set MP7440 bit 6 = 1.

When M128 or TCPM are used, the principal axes make compensating movements:

- ▶ In MP7471, define the maximum velocity of the principal axes during compensating movements. The speed for compensating movements with M128 can also be limited via the PLC and the **FMAX** soft key, in addition to MP7471.

MP7440 Output of M functions

Format: %xxxxxxx

Input: Bit 6 – Automatic activation of M134
 0: M134 must be activated in the NC program
 1: M134 is automatically activated when an NC program is selected

MP7471 Maximum velocity of the principal axes during compensating movements through M128 or TCPM

Input: 0 to 300 000 [mm/min]

Virtual tool axis (VT)

With the iTNC 530, handwheel superimposed traverse in the active tool axis system "Virtual Tool Axis VT" is possible when TCPM is active. This function is available in the Global Program Settings (option 44). When the HR 5xx is connected, the position of the virtual tool axis is displayed separately. It can be used in automatic mode when the HR 5xx/HR 410 is connected.

Requirements that must be met for using the Virtual Tool Axis VT:

- TCPM (Tool Center Point Management) function must be active.
- DCM (Dynamic Collision Monitoring) must be deactivated.
- MP7641 bit 4 "Handwheel superimposition in the active tool-axis direction" must be set.
- MP7503 "Virtual tool axis" must be set.
- MP7682 bit 9 "Load tilted working plane" should be set for the tilting values to be loaded into the **Manual Operation** mode during a program interruption.
- T must be entered in MP410.x for the virtual tool axis (VT).
- Global Program Setting software option (option 44) must be activated.
- In the Global Program Settings you must define the permissible traverse range for the VT axis (as with M118).
- Marker M4576 "Disable handwheel pulses" must not be set.

Associated new functions:

- In the Global Program Settings for handwheel superimposition, the axis VT can also be activated for the active tool-axis direction.
- If an HR 5xx is used, the axis VT can be selected for it in the axis menu.
- When an HR 410 handwheel is connected, the axis VT can be selected via PLC module 9036 (selection parameter 6 (handwheel axis); axis value -1).
- If selection parameter 10 (handwheel axis) is selected, PLC Module 9035 supplies the value \$8000000 (bit 31 is set) if the VT axis is active.
- If the handwheel traverses an offset in this axis, the tool length compensation will change (additional DL).
This additional length compensation is retained after a tool change.



Module 9232 Movement in the direction of the virtual tool axis

The new PLC Module 9232 makes it possible to trigger a movement in the direction of the virtual tool axis. The direction of movement is determined by the algebraic sign in the <Feed rate> parameter. The maximum traverse path can be limited via the <Path limitation> parameter in Automatic operating modes. The function can be used for retracting via the PLC program, for example. The global program settings must be set for this purpose (as described under Constraints).

Constraints:

- The <Path limitation> parameter is only effective if an Automatic or Program Run mode is active during the module call. Path limitation is not active in the Manual or Handwheel modes of operation.
- The module must be called cyclically as long as movements are to be performed.
- For safety reasons, the NC deactivates the function after the first cycle in which the module was not called anymore.
- Path limitation is not active in Manual operating modes; therefore, status 7 is not possible.
- With status 2 (interpolator reports an error), the PLC program can only try to start the functionality again after the PLC run-time system has deactivated the function. This can be done by interrupting the cyclic calls.
- The distance traversed (third return value) is retained even after the deactivation and after changes of operating modes.
- No traverse path value is returned in the Manual operating modes because there is no contour reference. The Manual Traverse soft key in Automatic operating modes has the same effect for the module as one of the Manual operating modes.
- The Global Program Settings option (#44) is required.
- The module uses the interfaces of handwheel superimpositioning in the virtual tool axis (Global Program Settings).
- The handwheel has priority if it is activated for the virtual tool axis (status 8).
- Once handwheel superimpositioning has been activated in the Global Program Settings, the module returns the distance traversed in Automatic operating modes.
- The distance traversed (third return value) is set to 0 when handwheel superimpositioning is deactivated in the Global Program Settings.
- When handwheel superimpositioning is active, the distance traversed with the module is retained (visible in the Global Program Settings).
- The path limitation active in the interpolator is the minimum from the global settings (column "Max. value", input field "VT") and the limitation of the module (third parameter).
- The error marker M4203 is only set with status 1 (PLC error) and status 2 (interpolator error). If errors occur during normal operation, the PLC run-time system deactivates the function as a consequence.

Call:

PS B/W/D/K <Mode>
 0: Movement in Automatic operating modes (Path Limitation parameter effective)
 1: Movement in Manual operating modes (Path Limitation parameter NOT effective)

PS B/W/D/K <Feed rate in [mm/min]>
 Direction of movement depending on the algebraic sign

PS B/W/D/K <Path limitation>
 Maximum traverse path in [0.1 µm]
 Only active if an Automatic operating mode is active during the module call!

CM 9232

PL B/W/D <Status>
 0: Function not active
 1: PLC error (see W1022)
 2: Interpolator error (see return value that follows)
 3: Waiting for response from interpolator (function not yet active)
 4: Function active, no movement (at feed rate 0)
 5: Function active, movement in positive direction
 6: Function active, movement in negative direction
 7: Function active, maximum traverse path reached (only in Automatic modes)
 7: Function active, handwheel is active

PL B/W/D <Interpolator error code with status 2>
 0: No error
 1: M128 not active
 2: Handwheel superimpositioning in Global Settings not active
 3: DCM is active during movement

PL B/W/D <Traverse path>
 Manual modes: 0
 Automatic modes and active handwheel: distance traversed via handwheel

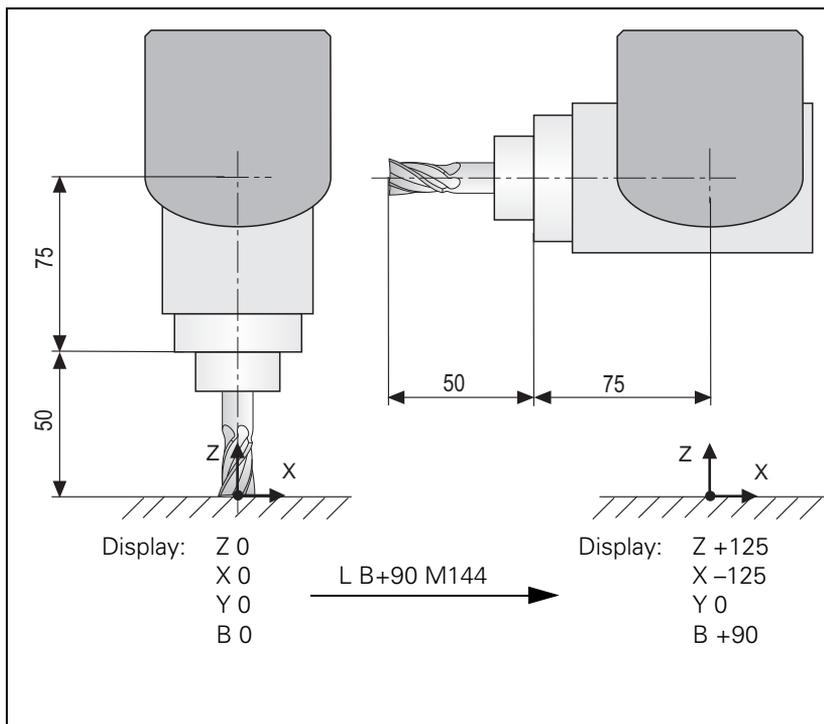
Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022 (also see return values of the module)
W1022	1	Invalid mode programmed
	2	The <Mode> parameter is invalid
	8	Call in incorrect operating mode
	24	The module was called from a SUBMIT/SPAWN job
	45	Module execution was canceled



**Miscellaneous
function M144/
M145**

M144 considers the kinematics in the display when a swivel axis is moved. However, the other axes do not perform any compensating movements. M144 is deactivated with M145.



FN18: SYSREAD ID310 NR144 can determine whether M144 is active or inactive.

MP7502 Functionality of M144/M145

Input:

%xxx

Bit 0

0: M144/M145 not active

1: M144/M145 active

Bit 1 – M144/M145 in the automatic modes

0: M144/M145 active

1: M144 is activated automatically at the start of an NC program.

It can only be deactivated with M145 during an NC program.

Bit 2 – M144/M145 in the manual modes

0: M144/M145 not active

1: M144/M145 active

6.5.8 Virtual tool axis

General information

When moving the axes in the manual operating modes, and when re-approaching the contour after a program interruption, you can select via the **3D ROT** and **TOOL AXIS** soft keys the coordinate system of the tool. The axes can be moved in the current tool axis direction even before traversing the reference marks. To accomplish this, the last position of the rotary axes before power-off is stored in non-volatile memory for incremental encoders. These non-volatile axis values are shown in a pop-up window when the control is started, and must be acknowledged by the machine operator with **ENT** (confirmation of the values) or **NO ENT** (values deviate, reference run must be performed first).

With the iTNC 530, handwheel superimposed traverse in the active tool axis system "Virtual Tool Axis VT" is possible when TCPM is active, see page 772.

Activation

MP7503 Virtual tool axis – Reapproaching the contour and manual traverse in the current tool-axis direction

Input: 0: Inactive
1: Active

6.5.9 Tilting functions with open-loop rotary axes

Inclined milling with open-loop rotary axes (counter axes)

Machines with rotary axes that only have open-loop axes (counter axes, $MP120.x = 0$, are not controlled) can be used for inclined milling. The following procedure is necessary:

- ▶ With M128 inactive (TCPM: the position of the tool tip remains constant when positioning tilting axes), the machine operator sets the rotary axes to the required nominal values.
- ▶ When M128 is activated, the control assumes the actual values of the non-controlled rotary axes, and uses them to calculate the changed position on the tool center point (TCP).
- ▶ The display of axes X/Y/Z is updated with the newly calculated values, and the compensating movement is performed with the next positioning.
- ▶ As long as M128 is active, the positions of the non-controlled rotary axes are monitored. If the positions of these axes deviate by more than the values defined in MP1110.x (standstill monitoring), then an error message is output and the momentary machining is interrupted.
- ▶ This function is even permitted after a program interruption with the **MANUAL TRAVERSE** soft key. In this case the new compensating movement is determined after switching with the **APPROACH POSITION** soft key.



6.5.10 Cylindrical surface

Cycles 27 and 28, "Cylinder Surface," enable the user to machine a contour on a cylindrical surface (see the User's Manual).

Prerequisites of the previous kinematics description:

- In MP7510 to MP7530, the center of rotation of a rotary axis must be defined (see example 3).
- If PLC datum compensation is used, the same home position must apply in the description of the machine geometry in MP7510.x to MP7530.x as in the datum shift.
- After a change in MP7510.x or MP7530.x, the datum must be reset.

Prerequisites of the new kinematics tables:

- A complete description of the kinematics, from the tool reference point to the center of the rotary axis, must exist.
- If the axis geometry is changed (**MachAxis** ...), the datum must be reset.

6.6 Gantry/synchronized axes

Conventions

For synchronized axes:

- The slave axis cannot be moved separately.
- The nominal value display of the slave axis shows the nominal value of the master axis.
- The PLC program must ensure that the synchronized axis does not move until the master and slave axes are ready (clamping, feed-rate enable).
- For the slave axis, the bits for "traverse direction" in W1030/W1112 and "axis in motion" in W1028/W1108 are **not** set.
- Linear and nonlinear axis error compensation as well as temperature compensation must be entered separately for each axis.
- An axis cannot be both master and slave at the same time.
- The values for rapid traverse, acceleration, jerk, software limit switches, feed rate for reference mark traverse, and manual feed rate are also taken over from the input values of the master axis for the slave axis.
- The axes must be either both analog or both digital.
- Master and slave axes can be linear or rotary axes.
- When operating with following error, the k_v factors for master and slave must be the same.
- For gantry axes, one position encoder is sufficient.
- The nonlinear axis-error compensation can be used separately for master and slave axes.
- For the nonlinear axis-error compensation, master and slave axes may be dependent on each other.
- The number of slave axes in master-slave torque control and gantry combinations is restricted by the controller unit on which the master axis is configured. The slave axes and the master axis must be configured on the same drive-control motherboard (DSPs). Thus, up to five slave axes maximum are possible (e.g. with CC 6106).

6.6.1 Gantry axes

In gantry axes, tandem tables, etc., two servo-controlled axes are coupled so that they can move only simultaneously. The main axis is referred to as the master, and the tracking axis as the slave. The number of control loops available also determines the number of Gantry axes possible.

Because a gantry coupling is configured exclusively over the slave axis, it is possible to assign more than one slave to the same master. However, a gantry slave axis cannot at the same time serve as a gantry master axis (e.g. up to five slave axes are possible for a CC for six axes).



Note

The master and slave axes must be on the same drive-control main board, which means that both axes need to be controlled by the same DSP. In this way, you can flexibly assign the master and slave axes to the PWM outputs X51 to X56 of a CC 6106. On controller units with more than one drive control board (CC 6108, CC 6110), the master axis and the slave axis both need to be assigned to either the A connections or the B connections. Mixed connection is not permitted.

The function is effective during control both with following error and with velocity feedforward.

Activating synchronized axes:

- ▶ Assign a slave axis to a master axis.

MP850.x	Synchronized axes
Input:	0: Master axis
	1: Slave axis to axis 1
	2: Slave axis to axis 2
	3: Slave axis to axis 3
	4: Slave axis to axis 4
	5: Slave axis to axis 5
	6: Slave axis to axis 6
	7: Slave axis to axis 7
	8: Slave axis to axis 8
	9: Slave axis to axis 9

Example

Axis 4 is slave to axis 1:

- MP850.0 = 0
- MP850.1 = 0
- MP850.2 = 0
- MP850.3 = 1
- MP850.4 = 0
- MP850.5 = 0
- MP850.6 = 0
- MP850.7 = 0
- MP850.8 = 0

Master-slave position deviation

The iTNC monitors the synchronism of the coupled axes. If the master and slave axes deviate from each other by the difference of the following errors, the iTNC displays the slave axis with the message **EXCESSIVE SERVO LAG IN <AXIS>**. The LAG display shows the current difference in position.

- ▶ In MP855.x of the slave axis, enter the maximum permissible difference in positions between the master and slave.

Machine parameter MP855.x (synchronization monitoring) is effective for all slave axes of a synchronized drive (if a value > 0 has been entered), regardless of whether it is a gantry or master-slave combination.

If an offset is caused in the axes through an emergency stop, they will be synchronized after the emergency stop.

Datum at position after switch-on (MP860.x = 0)

Entry for the slave axis

With MP860.x you can select whether the position after switch-on should be used as a synchronization reference. Master and slave axes must be at identical positions. If the defined datums are to be reproduced, then only the master needs to be moved over the reference mark.

Monitoring of synchronized axes begins immediately upon switch-on.

Datum at reference marks (set MP860.x bit 1)

Entry for the slave axis

With MP860.x you can select whether the position should be ascertained by traversing the reference marks. After crossing over the reference marks, the master and slave axes are positioned to the same value. The default setting can be corrected with MP960.x (machine datum). In order for MP960.x to be set, the axes must traverse the reference marks with MP860.x = 0, so that no compensation movements are made. An offset in the axes is corrected after both reference marks are traversed. Reference mark traverse is ended as soon as a reference mark is traversed in both axes. The monitoring function is not active until after the compensation movement. The monitoring function is not active before the reference marks are traversed.

Conditions:

- The same type of reference mark traverse must be set for both the master and slave axes (MP1350.x).
- The velocity with which an offset (after traversing a reference mark or emergency stop) is compensated for is defined in MP1330.x for the slave axis.
- In the sequence for traversing the reference marks (MP1340.x), the master axis must be defined before the slave axis.
- The compensation movement **cannot** be stopped with an NC stop (only with an emergency stop).
- The compensation movement is **not** considered in the following words:
 - W1026/W1104 (axes in position)
 - W1028/W1108 (axes in motion)
- If the master axis has traversed the reference mark at the time of an NC stop or an emergency stop, but the slave axis has not yet crossed it, then the slave axis can only be moved across it by using the axis-direction keys.
- Using a linear encoder: it is sufficient if the master axis has one reference end position.
- Using the speed encoder for linear measurement: One reference end position is enough, but the NC needs a reference end position signal for both axes (W1054/D1160).

Conventions

The following applies for a gantry combination:

- For a gantry combination, HEIDENHAIN recommends using a separate position encoder for each axis. This is the only way that the benefits of an independent position control of each axis of the gantry combination can be used.
- Linear and nonlinear axis error compensation as well as temperature compensation must be entered separately for each axis.
- Nonlinear axis-error compensation can be entered separately for master and slave axes.
- For the nonlinear axis-error compensation, master and slave axes may be dependent on each other.
- When operating with following error, the same k_v factors for master and slave must be entered.

MP855.x Synchronization monitoring

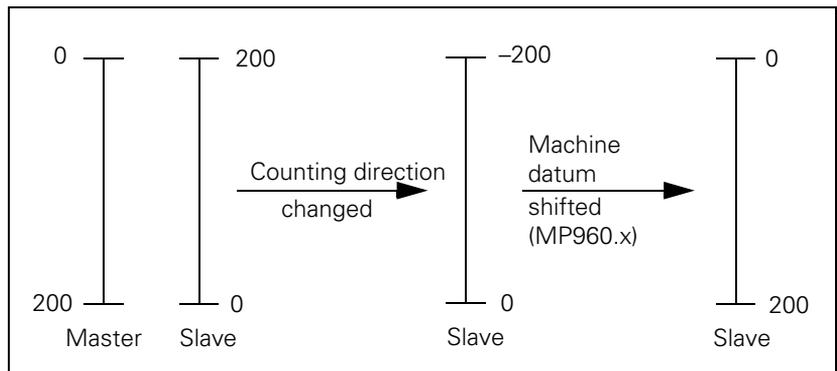
Input: 0 to 100.0000 [mm]
0: Monitoring not active

MP860.x Datum for synchronous control

Input: %xxx
Bit 0 – Datum for synchronous control
0: Datum at position after switch-on
1: Datum at reference marks (machine datum)
Bit 1 – Master-slave torque control
0: Not a torque slave axis
1: Axis is torque-slave axis
Bit 2 – Brake test for synchronized i.e. gantry axes
0: Test the brakes of a synchronized axis simultaneously
1: Test the brakes of a synchronized axis successively

Example

- Gantry axes with two position encoders
- Position encoder of the slave axis is mounted mirror-inverted.



6.6.2 Master-slave torque control

In master-slave torque control, two motors (master and slave) are mechanically coupled. Because of the coupling, only one position encoder is required. The motor to which the position encoder is assigned is the master.

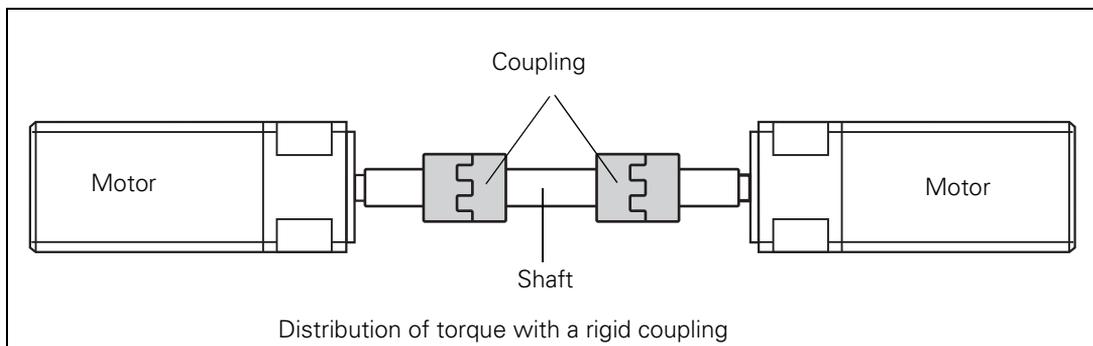
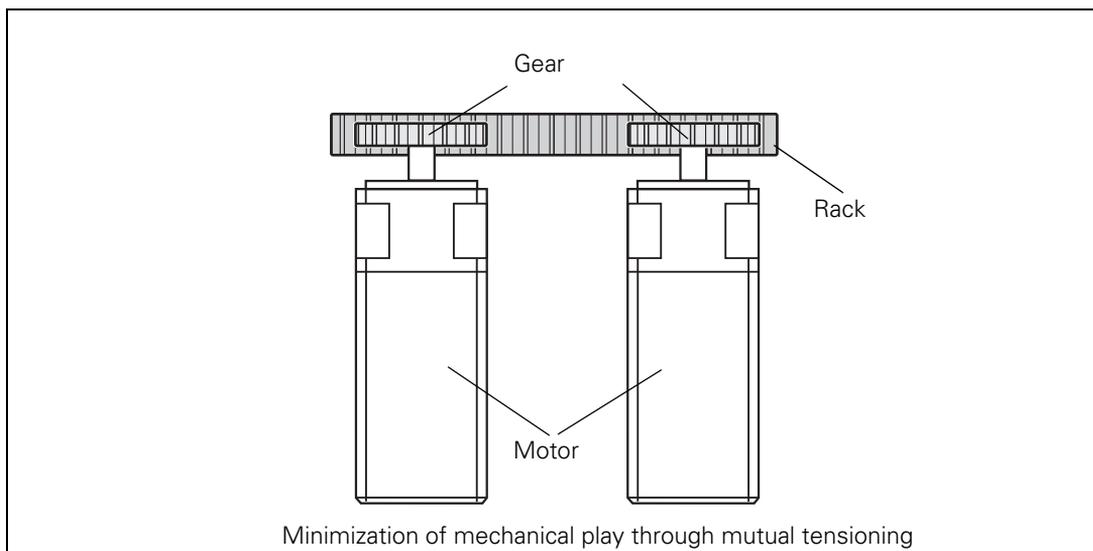


Note

The master and slave axes must be on the same drive-control main board, which means that both axes need to be controlled by the same DSP. In this way, you can flexibly assign the master and slave axes to the PWM outputs X51 to X56 of a CC 6106. On controller units with more than one drive control board (CC 6108, CC 6110), the master axis and the slave axis both need to be assigned to either the A connections or the B connections. Mixed connection is not permitted.

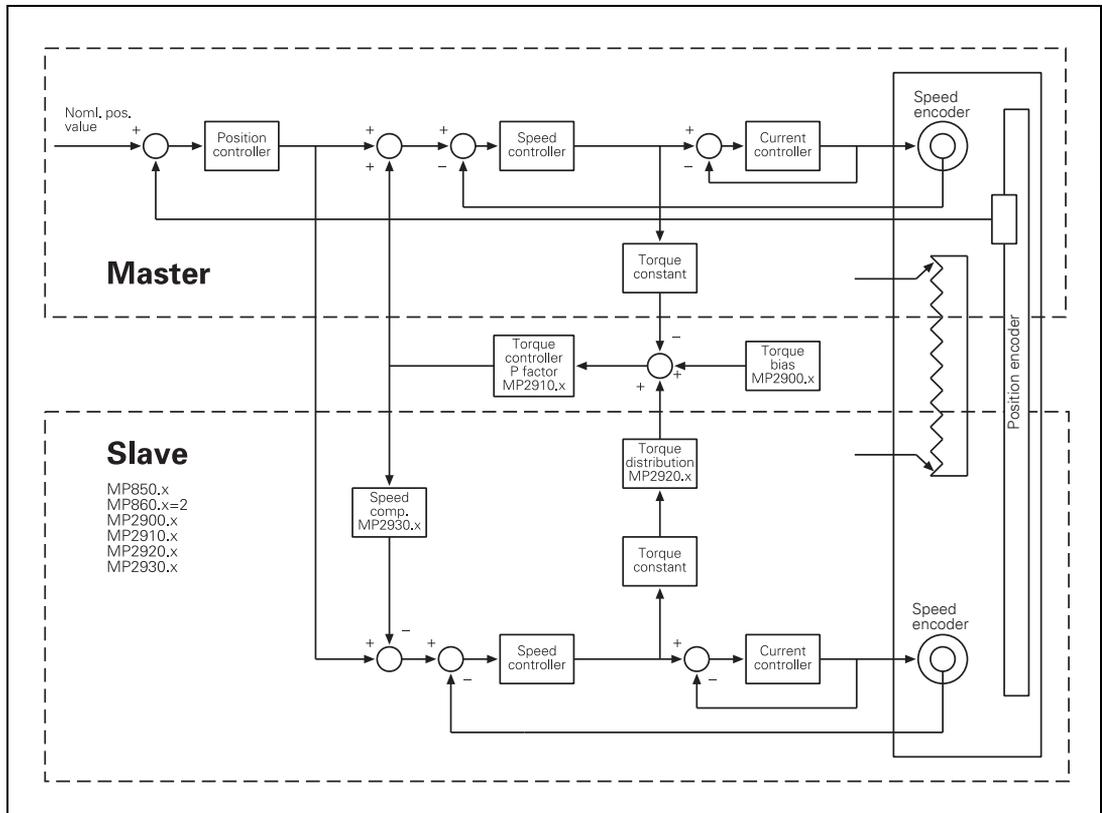
In principle there are two applications:

- Minimization of mechanical play through mutual tensioning
- Distribution of torque with a rigid coupling



Method of function

Position control is deactivated in the slave axis. The nominal velocity of the master axis is at the same time the nominal velocity of the slave axis. The speed controllers of both axes remain independent. The manipulated variables coming from the speed controllers, i.e. the nominal torque current values, are weighted with the torque constants of the motors and compared with each other. In addition, a tensioning torque (MP2900.x) can be introduced at this comparison point. MP2192.x makes a distribution of the drive torque possible. The output of the torque balancing controller (= speed compensation value) can be distributed either evenly to master and slave (MP2912.x = 0) or exclusively to the slave (MP2912.x = 1).



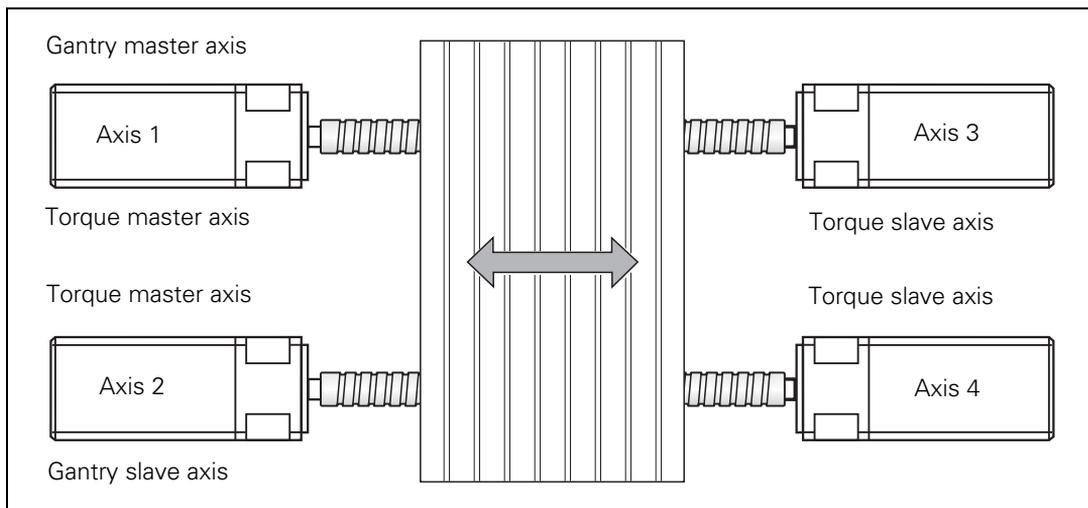
Conventions

The following applies for a master-slave combination:

- For a master-slave combination, no more than one common position encoder is required.
- Linear and nonlinear axis error compensation as well as temperature compensation is not possible separately for both axes.
- Nonlinear axis-error compensation cannot be entered separately for master and slave axes. Here the values of the master axis are always used.
- When operating with following error, the k_v factor for the master axis must be used.

Gantry axes in master-slave torque control

It is possible to run gantry axes in master-slave torque control. The gantry master and gantry slave axes are at the same time torque master axes and have one torque slave axis each.



Example for the MP entries:

MP850.0 = 0

Axis 1 is master axis

MP850.1 = 1

Axis 2 is slave to axis 1

MP850.2 = 1

Axis 3 is slave to axis 1

MP850.3 = 2

Axis 4 is slave to axis 2

MP860.0 = 0 or 1

Axis 1: Datum for synchronous control

MP860.1 = 0 or 1

Axis 2: Datum for synchronous control

MP860.2 = 2

Axis 3 is torque slave axis

MP860.3 = 2

Axis 4 is torque slave axis

Activation of master-slave torque control

- ▶ Activate the master and slave axes with MP10.
- ▶ In MP110.x, define the position encoder for the master.
- ▶ Enter MP110.x = 0 for the slave.
- ▶ In MP850.x, define the master axis as the main axis and the slave axis as the tracking axis.
- ▶ Activate the master-slave torque control by entering MP860.x bit 1 for the slave axis.

Axes for which master-slave torque control is active can be switched by the PLC to single-axis operation during operation by overwriting MP850.x.

MP860.x	Datum for synchronous control
Input:	%xxx
	Bit 0 – Datum for synchronous control
	0: Datum at position after switch-on
	1: Datum at reference marks (machine datum)
	Bit 1 – Master-slave torque control
	0: Not a torque slave axis
	1: Axis is torque-slave axis
	Bit 2 – Brake test of synchronized axes
	0: Test the brakes of a synchronized axis simultaneously
	1: Test the brakes of a synchronized axis successively

Setting the master-slave torque control for minimizing mechanical play

- ▶ For the master and slave axes you must select in MP1040 the same or the opposite direction of rotation, depending on the application (MP210 has no effect on the slave).
- ▶ Adjust the current controller for the master and slave axes "Commissioning" on page 1004.
- ▶ Enter the following temporary values in the machine parameters for the slave axis:
MP2900.x = approx. 20% to 25% of the rated torque of the motor
MP2910.x = 0.2
MP2912.x = 0
- ▶ If you use a position encoder, enter 1 in MP2912.x for the slave axis; if you do not use a position encoder, enter the value 0 in MP2912.x for the slave axis.
- ▶ Enter MP2510.x (I factor of speed controller) = 50 or, if you have one, an empirical value for your motor.
- ▶ Adjust the P and I factors of the speed controller for the master and slave axes at the same time, see "Commissioning" on page 1004. It is not permissible to commission the master and slave axes separately, since the motors must be tensioned during commissioning.
- ▶ If you do not reach the desired rise time (approx. 10 ms), you can increase the P factor with the aid of a filter. Here the band-rejection filter is preferable to the low-pass filter.
- ▶ To find the center frequency for the band-rejection filter, slowly increase the P factor to the oscillation limit and find the frequency with the integrated oscilloscope.



Note

For low-frequency oscillations (< approx. 200 Hz) you should not use a filter, because it may have a negative influence on the dynamics of the control. For the mid-range frequency (approx. 200 Hz to approx. 400 Hz) ensure that you do not excite any low-frequency oscillation.

The higher the frequency of the oscillation (> approx. 400 Hz), the less negative will be the influence of high damping on the dynamics.

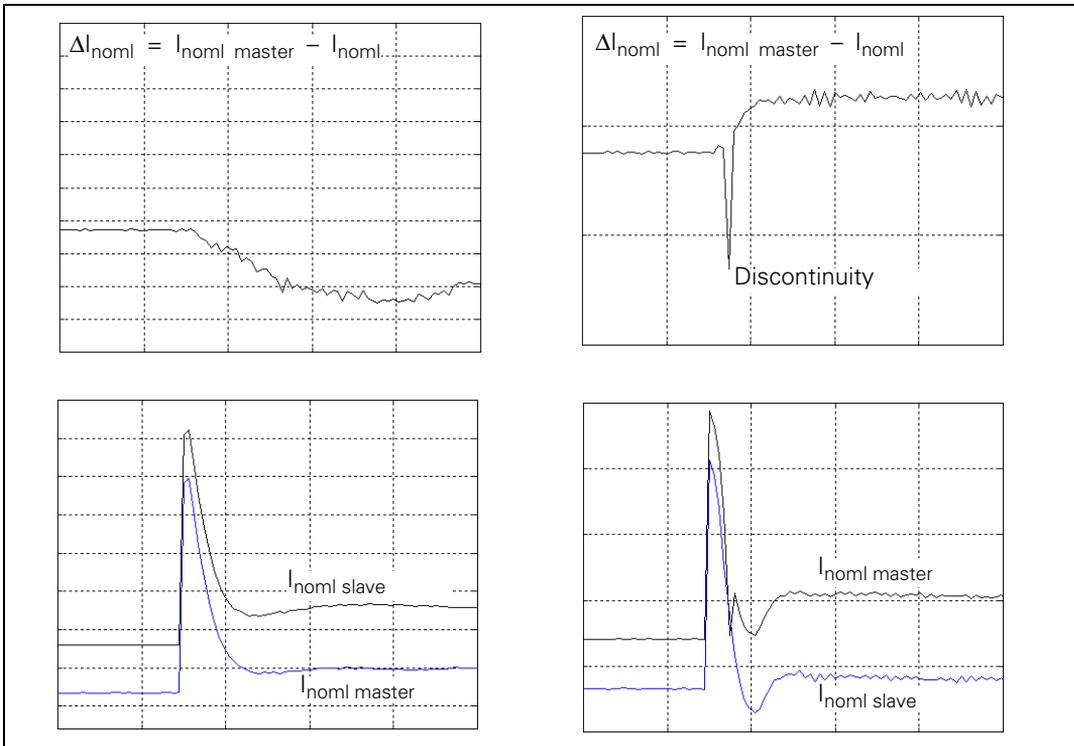


Note

For identical motors, the factors of the speed controller should be identical to ensure identical dynamic behavior.

Test the tensioning torque:

- ▶ With the integrated oscilloscope, record the nominal current (I_{noml}) of the master and the slave axes at standstill.
- ▶ Send a step to the speed controller and, with the integral oscilloscope, record the nominal current of the master and slave axes.
- ▶ If there is a discontinuity in the course of the nominal current, increase the tensioning torque for the slave axis in MP2900.x.





Note

The lower the ratio of the total mass moment of inertia (transmission, machine table, etc.) to the motor mass moment of inertia, the smaller the required tensioning torque is (MP2900.x).

Test the P factor of the torque controller:

- ▶ With the integrated oscilloscope, record the actual speed value V (N ACTL).
- ▶ Increase the P factor in MP2910.x for the slave axis up to the oscillation limit.
- ▶ Enter in MP2910.x for the slave axis 50% of the resulting value.

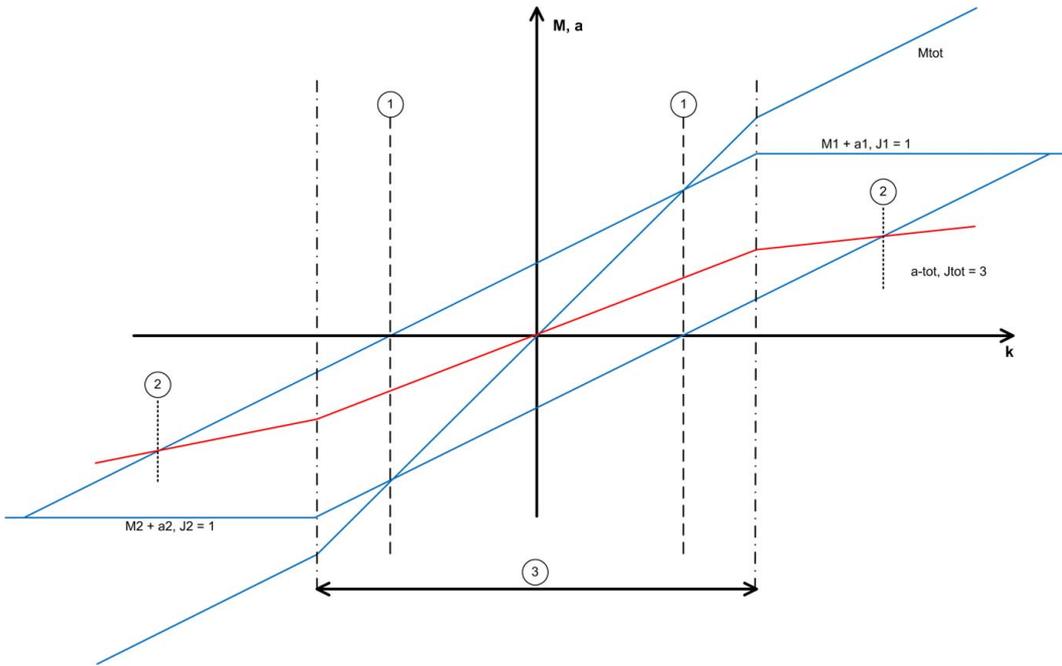
Notes on the tensioning torque in MP2900.x:

- If a tensioning torque of e.g. 10 Nm is entered in MP2900.x of the slave axis, the slave axis and the master axis must each generate a torque of 10 Nm.
- The actual tensioning torque corresponds to the entry in MP2900.x, i.e. 10 Nm.

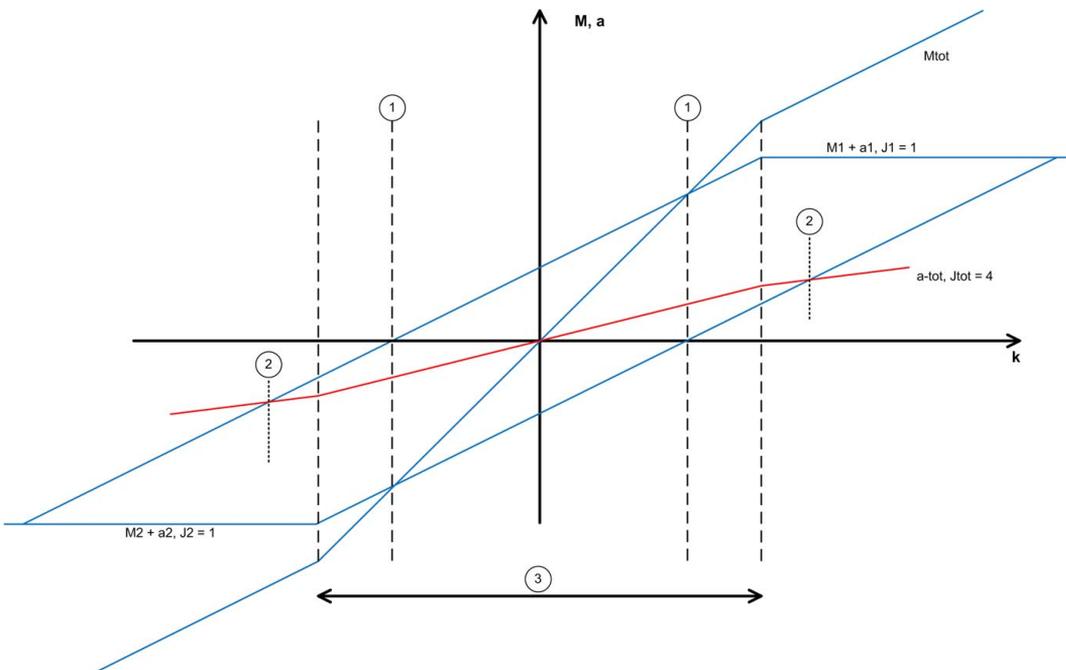
The figures below show the correlation between torque, acceleration and tooth-flank change for master-slave torque control. The figures show the behavior for different moments of inertia or accelerations using the following points in the characteristic curves:

- 1: Operating point at which a static tooth-flank change takes place ($a_{\text{tot}} = 0$)
- 2: Operating point at which a dynamic tooth-flank change takes place. This point depends on the ratio of the motor's moment of inertia to the total mass moment of inertia.
- 3: Linear operating range of master-slave drive.

First figure illustrating a moment of inertia of $J_{tot} = 3$, $J_{Rack} = J_{Mot}$



Second figure illustrating a moment of inertia of $J_{tot} = 4$, $J_{Rack} = 2 \times J_{Mot}$



Remember that:

- M1: Torque of motor 1
- M2: Torque of motor 2
- Mtot: Total torque $M1 + M2$
- a1: Acceleration of motor 1
- a2: Acceleration of motor 2
- J1: Inertia of motor 1, $J1 = J_{Mot}$
- J2: Inertia of motor 2, $J2 = J_{Mot}$
- JRack: Moment of inertia of rack
- Jtot: Total moment of inertia $J1 + J2 + JRack$



Setting the master-slave torque control for torque distribution in a rigid design

- ▶ For the master and slave axes you must select in MP1040 the same or the opposite direction of rotation, depending on the application (MP210 has no effect on the slave).
- ▶ Adjust the current controller for the master and slave axes see "Commissioning" on page 1004.
- ▶ Enter the following temporary values in the machine parameters for the slave axis:
MP2900.x = 0
MP2910.x = 0.2
MP2912.x = 0
- ▶ If you use a position encoder, enter 1 in MP2912.x for the slave axis; if you do not use a position encoder, enter the value 0 in MP2912.x for the slave axis.
- ▶ Enter MP2510.x (I factor of speed controller) = 50 or, if you have one, an empirical value for your motor.
- ▶ Deactivate the slave axis in MP10.
- ▶ For the master axis, adjust the P and I factor of the speed controller. See see "Commissioning" on page 1004.
- ▶ If you do not reach the desired rise time (approx. 10 ms), you can increase the P factor with the aid of a filter. Here the band-rejection filter is preferable to the low-pass filter.
- ▶ To find the center frequency for the band-rejection filter, slowly increase the P factor to the oscillation limit and find the frequency with the integrated oscilloscope.



Note

For low-frequency oscillations (< approx. 200 Hz) you should not use a filter, because it may have a negative influence on the dynamics of the control. For the mid-range frequency (approx. 200 Hz to approx. 400 Hz) ensure that you do not excite any low-frequency oscillation. The higher the frequency of the oscillation (> approx. 400 Hz), the less negative will be the influence of high damping on the dynamics.

- ▶ Deactivate the master axis in MP10.
- ▶ Set MP850.x and MP860.x to 0 for the slave axis.
- ▶ Set the speed controller and the filter parameters for the slave axis in the same manner as for the master axis.



Note

For identical motors, the factors of the speed controller should be identical to ensure identical dynamic behavior.

Test the P factor of the torque controller:

- ▶ In MP10 reactivate the master and slave axes.
- ▶ With the integrated oscilloscope, record the actual speed value V (N ACTL).
- ▶ Increase the P factor in MP2910.x for the slave axis up to the oscillation limit.
- ▶ Enter in MP2910.x for the slave axis 50% of the resulting value.



Note

The tensioning torque entered in machine parameter MP2900.x applies for the master and slave axis. If a torque of 10 Nm is entered, this results in a total tensioning torque of 20 Nm between the master and slave.

MP2900.x Tensioning torque between master and slave for master-slave torque control (entry for the slave axis)

Input: -100.00 to +100.00 [Nm]

MP2910.x P factor of the torque controller for master-slave torque control (entry for the slave axis)

Input: 0.00 to 999.99 [1/(Nm · min)]

MP2920.x Factor for variable torque distribution for master-slave torque control (entry for the slave axis)

Input: 0.000 to 100.000

1: Master and slave axes have identical motors

MP2912.x Speed compensation ratio for master-slave torque control (entry for the slave axis)

Input: 0: The output of the torque-adjustment controller (= speed compensation value) is distributed evenly to master and slave.
1: The output of the torque-adjustment controller (= speed compensation value) is only distributed to the slave.

6.6.3 Brake test for synchronized axes

The brake test is activated (> 0) or deactivated ($= 0$) separately for each servo motor via MP2230.x. During the brake test, an additional test torque from MP2230.x, defined via a multiplier, is applied to the motor stall current. This test torque exerts additional load on the holding brake of the servo drive. The axis is prevented from moving during the brake test and the brake test is considered to have been passed only if the brake withstands this load.

The iTNC 530 runs the brake test for synchronized axes as follows:

Behavior during the brake test of synchronized axes:

The function reads the machine configuration to detect which servo drives are operated together as a synchronized axis and which must therefore be handled separately in the brake test. Machine parameter MP850.x is used to configure the servo drives as a synchronized axis. The brakes and servo drives of the synchronized axis are tested simultaneously. It is ensured, however, that the same algebraic sign is used for the test torque of all servo drives. The sign is determined for all servo drives of the synchronized axis based on the entry for the holding torque of the master in MP2630.x. If no value is entered in MP2630.x, the current holding torque of the master is used.

There are two possibilities for starting the brake test. In both cases the function described above is used to test the brakes of synchronized axes simultaneously:

- Automatic brake test

The brake test takes place automatically during the power-up test of the control, as soon as all servo drives of the respective synchronized axis have been switched on.

- Brake test via PLC module

The brake test can be activated axis-specifically by the PLC program with PLC module 9143. What is new in the service packs listed below is that the slave drives of a synchronized axis are tested simultaneously with the master. Previously it was not possible to perform the brake test of slave drives via the PLC module. As a prerequisite for the brake test of a synchronized axis, all servo drives of the axis must be switched on and the brakes must be open.

Alternative behavior:

Testing the brakes of a synchronized axis successively

MP860 bit 2 can activate a changed brake test sequence. If MP860 bit 2 = 1, the servo drives of a synchronized axis are tested successively rather than simultaneously. As a result, the brake test is performed individually for all servo drives of a synchronized axis.

All brakes and servo drives of the synchronized axis are tested successively, one after the other, at the specified test current. For the brakes and servo drives that are not part of the momentary test, but are configured as connected to the servo drive to be tested, the current is set during the test so that the servo drive is not moved. The brakes of these servo drives must be open for this purpose. This way, each time only the brake of an individual servo drive is tested, without the other servo drives or brakes of the synchronized axis having an effect on the test.

To be able to use this brake test sequence, the individual brakes of the synchronized axis must not be combined. It must be possible to control the brakes individually.

There are also two possibilities for starting the brake test. In both cases the function described above is used to test the brakes of synchronized axes sequentially:

- Automatic brake test
The brake test takes place automatically during the power-up test of the control, as soon as all servo drives of the respective synchronized axis have been switched on.
- Brake test via PLC module
The brake test can be activated axis-specifically by the PLC program with PLC module 9143. What is new in the below-mentioned service packs is that the slave drives of a synchronized axis are tested successively after the master. Previously it was not possible to perform the brake test of slave drives via the PLC module. As a prerequisite for the brake test of a synchronized axis, all servo drives of the axis must be switched on and the brakes must be open.

Constraints for the brake test:

As a prerequisite for the brake test of a synchronized axis, all servo drives of the axis must be switched on and the brakes must be open. The test can only be performed if all relevant servo drives are switched on.



Note

Before performing the brake test, ensure via the PLC program that all servo drives of a synchronized axis are switched on and the holding brakes are open.

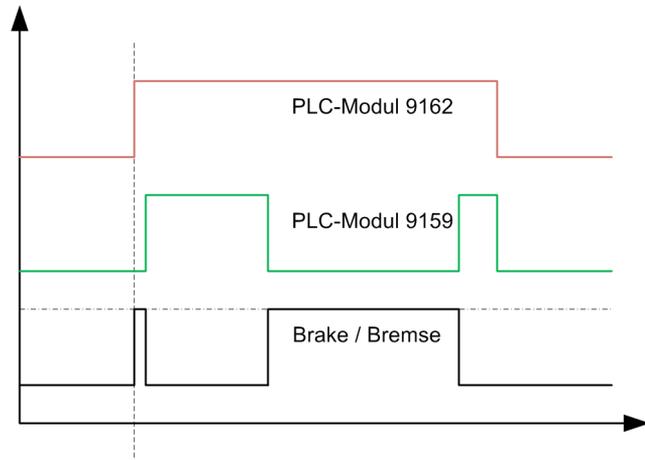
For slave drives for which the brake test has been disabled via MP2230.x, the current is adjusted so that the servo drive is not moved while the other servo drives of the synchronized axis are being tested.

Since the algebraic sign of the test torque cannot be determined until the drives are feedback-controlled and the brakes are open, an appropriate waiting time must be specified for the start of the brake test of synchronized axes. The time set in MP2309.x is used for this. The value for MP2309.x must equal the time that passes until the brake is really open after the controller has been switched on. The same time must be entered in MP2309.x for all servo drives of a synchronized axis. For CC 424 and CC 6xxx controller units the time in MP2309.x is taken into account each time the controller is switched on. For CC 422 controller units the time in MP2309.x is only taken into account the first time the controller is switched on, in order to delay the brake test by this time.

In general, the following applies to the brake control: If the brakes are controlled by the PLC, and not by the inverters, the PLC module 9159 (drive controllers are switched off) transmits the status message to the PLC program regarding the closing of the brakes during the brake test. However, also remember the relationship with PLC Module 9162. The status signal of PLC Module 9162 returns a value of "1" if the drive is ready.



The diagram shows the status signals of the two PLC Modules 9159 and 9162, as well as the trigger signal for the brake from the HEIDENHAIN inverter (0 = brake is closed). The diagram shows the basic sequence during a switch-on process with brake test, as well as a switch-off process with overlap time for braking. A value of "1" from PLC Module 9162 indicates that the drive is ready. A value of "1" from PLC Module 9159 at the beginning of the diagram indicates the duration of the brake test. A value of "1" from PLC Module 9159 at the end of the diagram corresponds to the overlap time for braking (typically 200 ms).



Note

HEIDENHAIN recommends the sequential brake test of synchronized axes for all machines on which a brake test for synchronized axes is made (MP860 bit 2 = 1).

- Test the behavior of the PLC program and the brake test on the machine.

6.7 Reference marks

6.7.1 Definition

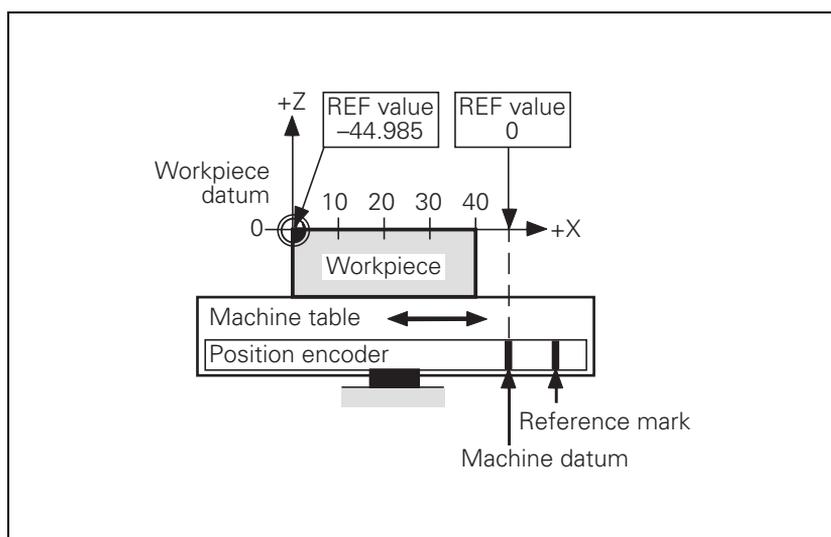
The position value (the coordinates) of an axis position is defined with respect to a freely chosen datum. When the axes are moved, the ACTUAL position is calculated incrementally. If there is an interruption in power, the reference between the axis position and the position value is lost.

Reference marks

HEIDENHAIN linear encoders are designed with one or more reference marks. The reference marks identify an axis position at a known distance from the machine datum. The position of the freely selectable datum is defined with respect to the machine datum.

The datum and the actual position can be reproduced as soon as the reference marks are traversed.

HEIDENHAIN recommends position encoders with distance-coded reference marks. With distance-coded reference marks, the position value can be reestablished after traverse of a short distance over any two reference marks.



6.7.2 Traversing the reference marks

The reference marks must be traversed after any interruption in power:

- ▶ Press the machine START button: The reference marks are automatically traversed. The sequence of axes is predetermined.

or:

- ▶ Press the machine axis-direction button. The user determines the sequence of the axes.

After the reference marks have been traversed:

- The software limit switches are activated.
- The most recently saved datum and machine datum are reproduced.
- PLC positioning and positioning with M91 and M92 become possible.
- The counter is set to zero for axes in an open loop.

Distance between the scale reference point and the machine datum

For position encoders with distance-coded reference marks, the machine datum is defined with respect to the scale reference point, which is at the first reference mark after the beginning of the measuring length. On angle encoders, the scale reference point is marked.

- ▶ In MP960.x, enter the distance between the scale reference point and the machine datum.

For position encoders without distance-coded reference marks but with more than one reference mark, the distance between the reference mark to be traversed and the scale reference point must also be entered:

- ▶ With Module 9225, enter the distance between the reference mark to be traversed and the scale reference point.

Module 9225 Compensation value for the reference mark

With Module 9225 you define the distance between the reference mark to be traversed and the scale reference point for the NC and PLC axes.

Call:

```
PS    B/W/D/K <Axis>
        0 to 8: Axes 1 to 9
        15: Spindle
PS    B/W/D/K <Compensation value in 0.1 µm>
        0: Reference mark to be traversed = scale reference point
CM    9225
PL    B/W/D <Error code>
        1: Axis does not exist
```

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Axis not found

Assigning a reference value

In some cases a new reference value may have to be assigned to an axis, e.g. if an axis is mechanically fixed and the encoder is moved. Since due to the mechanical fixing the position of the axis cannot be changed, you can assign it a new reference value.

- ▶ Enter the new reference value in Module 9147.

Module 9147 Assign a reference value to an axis

If a new reference value is assigned to an axis, the corresponding bit is reset in W1032/W1116.

Call:

```
PS    B/W/D/K <Axis number>
        0 to 8: Axes 1 to 9
PS    B/W/D/K <New reference value in 0.1 µm>
CM    9147
```

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Invalid axis number
	21	Missing strobe in M4176 = 1
	24	Module was called in a spawn or submit job

Defining the process of traversing the reference marks

In machine parameters, you define the process of traversing the reference marks:

- ▶ In MP1320.x and MP1330.x (for rotary encoders also in MP1331.x) you define the direction and velocity for traversing the reference marks.
- ▶ In MP1340.x you define the sequence of axes for traversing the reference marks.
- ▶ With MP1350.x you select the type of reference marks.

External reference pulse

If the reference mark of the encoder cannot be used, e.g. owing to an unfavorable transmission of motor and rotary axis, an external reference pulse may be evaluated.

- ▶ In MP4130.x, define the fast PLC input for the external reference pulse
- ▶ For the corresponding axis in MP1360.x, enter the number of the fast PLC input
- ▶ Enter MP1350.x = 6 for the corresponding axis

This can also be used for linear axes. Activation of the fast PLC inputs (MP4130.x) makes it possible to interrogate the inputs in the controller cycle. This controller cycle time and the traversing speed when crossing the reference mark mainly determine the accuracy of the reference run:

- Fast traverse speed → less accurate
- Longer controller cycle time → less accurate

It may be possible to achieve sufficient accuracy with two reference runs (first fast, then slow). Specify the logic for the fast PLC inputs with MP4131.2-5.

"Traverse Reference Points" operating mode

The NC uses W272 to report the "Pass Over Reference Point" operating mode to the PLC.

If you switch the operating mode before all reference marks are traversed, the **PASS OVER REFERENCE** soft key prompts you to traverse the remaining reference marks. In W1032/W1116 the PLC receives the information as to which axes have not yet been referenced.

In W1032/W1116, the bits for axes that are not to traverse the reference marks (MP1340.x = 0) remain set.

In the NCMACRO.SYS file, after the code number RESETINIT= you can enter the name (incl. path) of a macro that will be called when the Pass Over Reference Point mode of operation is exited. If the NC macro is terminated once with **END PGM** or **M02**, it will no longer be run when the **Pass Over Reference Point** mode is called and exited.

If M4622 is set during the first run of the PLC program, the message window **Waiting for M4622** appears after the reference marks have been traversed. The window does not disappear until you have reset M4622. In this way, you can delay the execution of the NC macro defined through RESETINIT = in the NCMACRO.SYS.

To synchronize the current machine status and the look-ahead calculation with an NC macro call, see "NCMACRO.SYS" on page 1709.

Reference end position

To prevent the axes from violating their traverse limits when traversing the reference marks, each axis requires a trip dog (at the reference end position). The trip dogs must be installed by the machine tool builder at the ends of the traverse range. The switch signals from the trip dogs are sent to free PLC inputs. The PLC program must gate these PLC inputs with W1054/D1160 for "reference end position."

The axis will only automatically be positioned to the software limit switch if

- It is beyond the positive software limit switch and is moving in the positive direction to the positive trip dog.
- It is beyond the negative software limit switch and is moving in the negative direction to the negative trip dog.

Activate the software limit switches before traversing the reference marks

The control saves the last axis positions to be traversed before switch-off. These positions allow you to check the traverse ranges of the axes when traversing the reference marks. This can be activated for a specific axis in MP1352.x and is switched on/off with the MONITOR SW LIMIT soft key. This soft key is available after the control has been switched on in the "reference mark traverse" operating mode.



Danger

- When approaching the reference marks again (e.g. after switching the axis), this monitoring function is not active.
- If fatal error messages occur, the position at shutdown may not be exactly identical to the position at switch-on.
- Please inform your customers for which axes you have enabled the activation of the software limit switches in MP1352, and for which axes you have not enabled it. The MONITOR SW LIMIT soft key, specifically for switching monitoring off, is available if MP1352 is set for at least one axis. The user, however, is not informed for which axes the software limit-switch monitoring function before reference-mark traverse is active.

Encoders with EnDat interface

Encoders with EnDat interface can be connected to the position and speed inputs of the controller unit CC 6xxx. With these encoders there is no need to traverse the reference marks. The position value is only read when the control is switched on. It cannot be read again.

When connecting a position encoder with an EnDat interface:

- ▶ Enter $MP1350.x = 5$.

When connecting a speed encoder with an EnDat interface:

- ▶ The iTNC automatically attempts to communicate with the encoder.

When connecting a speed encoder with an EnDat interface as a position encoder:

- ▶ Enter $MP1350.x = 5$.
- ▶ In $MP110.x$, enter 0 for the axis with the speed encoder with EnDat interface.



Note

If use of multiturn encoders with EnDat interfaces results in overruns, the corresponding information is entered in the system file NCDATA.SYS. For a control exchange, this file must be transferred or $MP960.x$ must be readjusted.

Setting $MP960.x$ (machine datum) when using rotary encoders with EnDat interface:

- ▶ Move the axis (encoder) to the nominal datum.
- ▶ Enter the displayed value of the deviation in $MP960.x$. Reverse the algebraic sign.

or:

- ▶ Set the cursor in the active machine-parameter file to the corresponding $MP960.x$.
- ▶ Press the actual-position-capture key.
- ▶ Enter "0" in the datum field and press Enter. The current actual value is then entered automatically in $MP960.x$.

$MP1350.x = 8$ enables you to reference incremental encoders with EnDat 2.2 protocol. With these encoders the absolute value is not available after switch-on, so that reference marks must be traversed. If this type of encoder is connected to the position input, bit 3 of $MP118.x$ must be set. This type of encoder cannot be used at the speed encoder input yet.

Double reference run

During the double reference run, the absolute position is first output through the EnDat interface of the speed encoder. If at a later time the reference mark of the position encoder is traversed, the control continues to work with this reference.

- ▶ Set the corresponding bits in MP1355 to 1 for the axes for which the double reference run is to be used.

The distance between the speed encoder and the position encoder must be entered in MP1356.x. When the reference mark of the position encoder is **first** traversed, the message **Set MP1356.<axis number> to <value>** appears.

- ▶ Enter this value in MP1356.x.
- ▶ Define the behavior of W1032/W1116 in MP1357.x.

MP960.x Machine datum

Input: -1.79769313486E+308 to
+1.79769313486E+308 [mm] or [°]
Values with respect to the scale reference point

MP1320 Direction for traversing the reference marks

Format: %xxxxxxxxxxxxxxxxxxxx
Input: Bits 0 to 17 represent axes 1 to 18
0: Positive
1: Negative

MP1330.x Velocity for traversing the reference marks

Input: 80 to 1 000 000 [mm/min]

MP1331.x Velocity for leaving the reference mark end position for axes 1 to 18 (only for rotary encoders MP1350 = 2)

Input: 10 to 1 000 000 [mm/min]

MP1340.x Sequence for traversing the reference marks

Input: 0: No evaluation of reference marks
1 to 18: Axis 1 to 18

MP1350.x Sequence for finding the reference mark

Input: 0: Linear encoder with distance-coded reference marks (old routine)
1: Position encoder with one reference mark
2: Special type (length measurement with ROD)
3: Linear encoder with distance-coded reference marks (new routine)
4: Same as 3 except that two reference marks are evaluated
5: Encoder with EnDat interface
6: Reference pulse via fast PLC input
7: Speed encoder with EnDat interface at the position and speed encoder inputs
8: Incremental encoder with EnDat 2.2 interface, reference run required (possible only at the position encoder input)

MP1352 Activate the software limit switches before traversing the reference marks

Format: %xxxxxxxxxxxxxxxxxxxx
Input: 0 = Software limit switch not active
1 = Software limit switch active



- MP1355 Double reference run**
 Format: %xxxxxxxxxxxxxxxxxxxx
 Input: Bits 0 to 17 represent axes 1 to 18
 0: Reference run as defined in MP1350.x
 1: Double reference run
- MP1356.x Distance between speed and position encoder for double reference run.**
 Input: -99 999.999 to +99 999.999 [mm] or [°]
- MP1357.x W1032/W1116 for double reference run**
 Input: 0: Reset W1032/W1116 if the reference run has been over the EnDat interface of the speed encoder
 1: Reset W1032/W1116 if the reference mark was traversed with the position encoder
- MP1360.x Fast PLC input for reference pulse**
 Input: 0: No fast PLC input for reference pulse
 1 to 5: Fast PLC input for reference pulse (MP4130.x)

		Set	Reset
M4622	Delay NC macro with RESETINIT = from NCMACRO.SYS	PLC	PLC
W1032/1116	Reference marks not yet traversed Axis-specific: Dependence on MP4020 bit 14	NC	NC
W1054/1160	Reference end position Axis-specific: Dependence on MP4020 bit 14	PLC	PLC



Renewed traversing of the reference marks

Module 9220 Renewed traversing of the reference marks

With this module you start an NC or PLC axis or a servo-controlled spindle for traversing the reference mark. It is possible to repeat the reference mark traverse in an axis that has already been referenced. The module can be called in all operating modes. Software limit switches are not effective. The strobe marker must remain set for the entire duration of the reference-mark traverse.

Axis:

- The sequence of functions (MP1350.x) and the velocity for leaving the reference end position (MP1331.x) are defined by machine parameter.
- The velocity and the direction for traversing the reference marks are either taken from MP1330.x and MP1320.x or they are defined in the module.



Note

The direction of traverse should be defined in the module only in exceptional cases. Since the reference end positions are not considered in this case, the limits of the traverse range may be violated.

- If an axis is started for reference point traverse although the reference mark has already been traversed, the corresponding bit is set in W1032/W1116 and the reference mark is traversed again. The same constraints apply as when traversing the reference mark for the first time.
- An axis cannot be started for referencing until all other axes are in position.

Servo-controlled spindles:

- The speed for traversing the reference mark is defined in the module.
- The spindle must be started from a standstill to traverse the reference mark.
- If the spindle is started for reference mark traverse, marker M4018 is set.

Call:

PS B/W/D/K <Axis/spindle>
Axis-specific: Dependence on MP4020 bit 14

PS B/W/D/K <Feed rate/shaft speed>
0: Feed rate MP1330.x
>0: Feed rate in mm/min or shaft speed in 1/1000 min⁻¹

PS B/W/D/K <Direction of traverse>
-1: Negative direction
0: Direction from MP1320.x
1: Positive direction

CM 9220

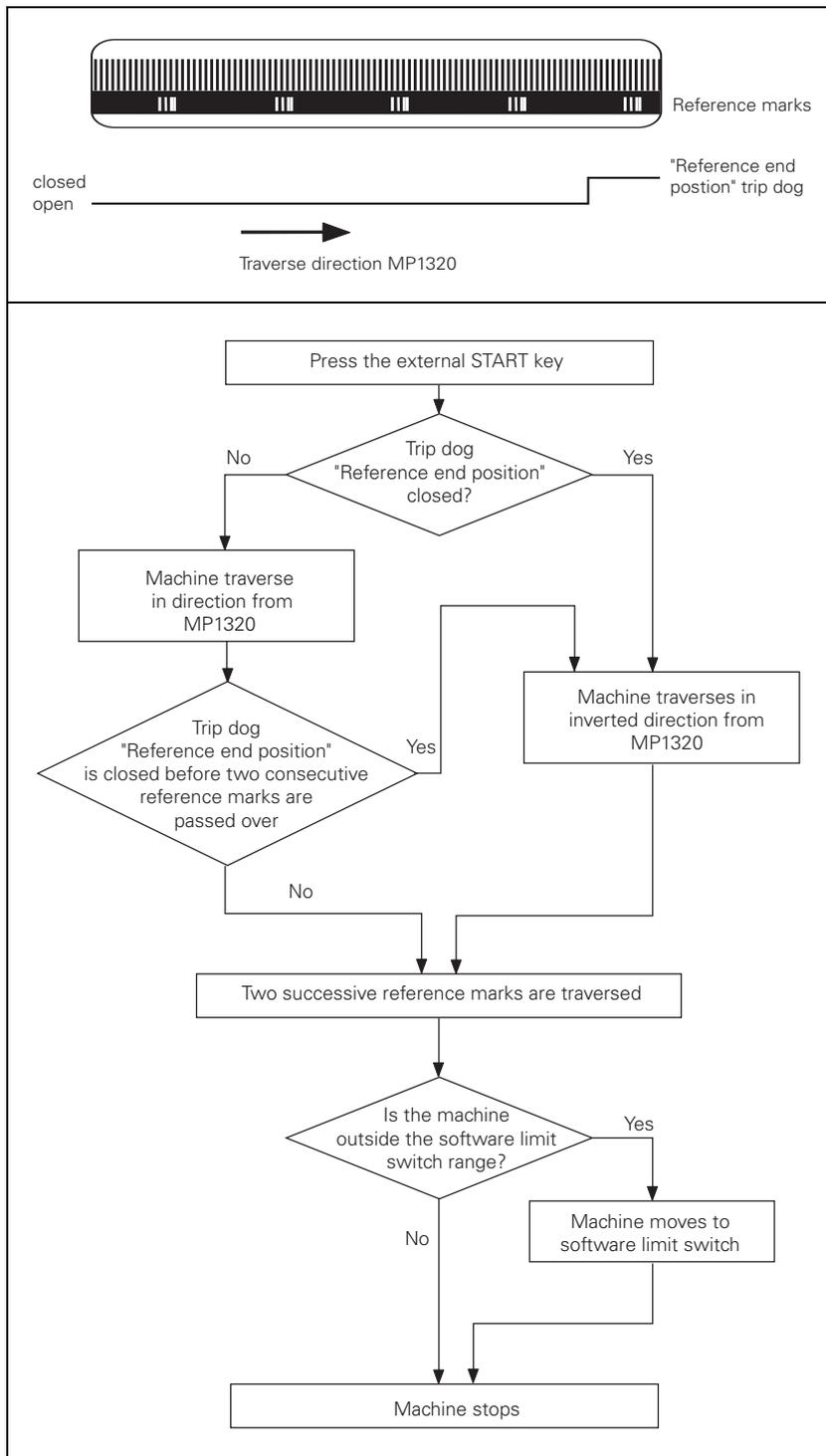
PL B/W/D <Error code>
0: Reference mark traverse is commanded
1: Non-existent axis or open-loop spindle
2: Inadmissible values for the feed rate / direction
4: Reference traverse not possible because reference traverse already started
5: Axis is already being positioned or the spindle is in motion
6: Other axis is already being positioned
7: PLC positioning has already been commanded
8: Programmed axis not in closed loop

Error code:

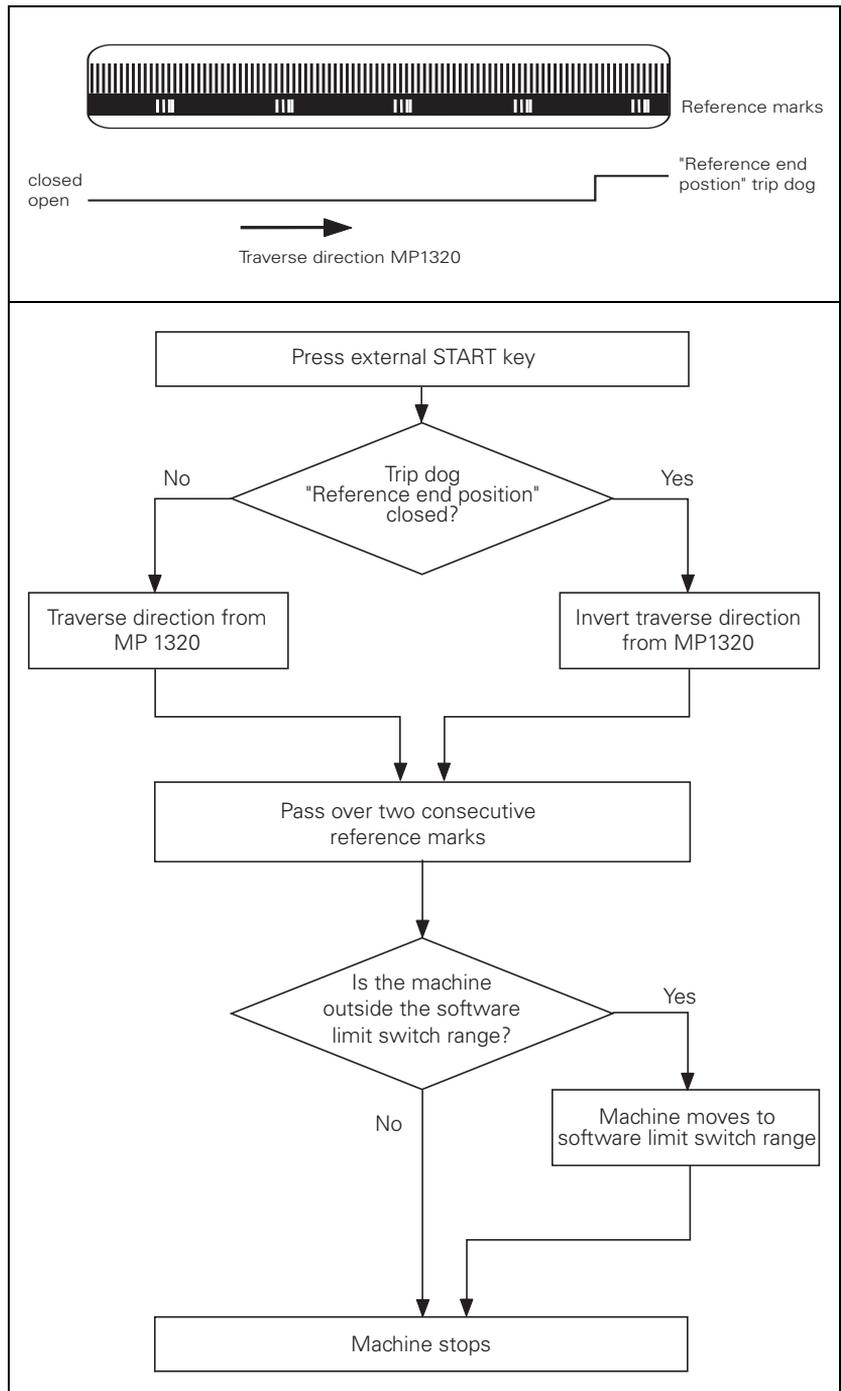
Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Non-existent axis or open-loop spindle or open-loop axis programmed
	2	Illegal values for feed rate/direction
	9	Axis is already being positioned or spindle in motion or PLC positioning already commanded
	28	Reference traverse not possible because reference traverse already started

Position encoder with distance-coded reference marks

Function when MP1350.x = 3



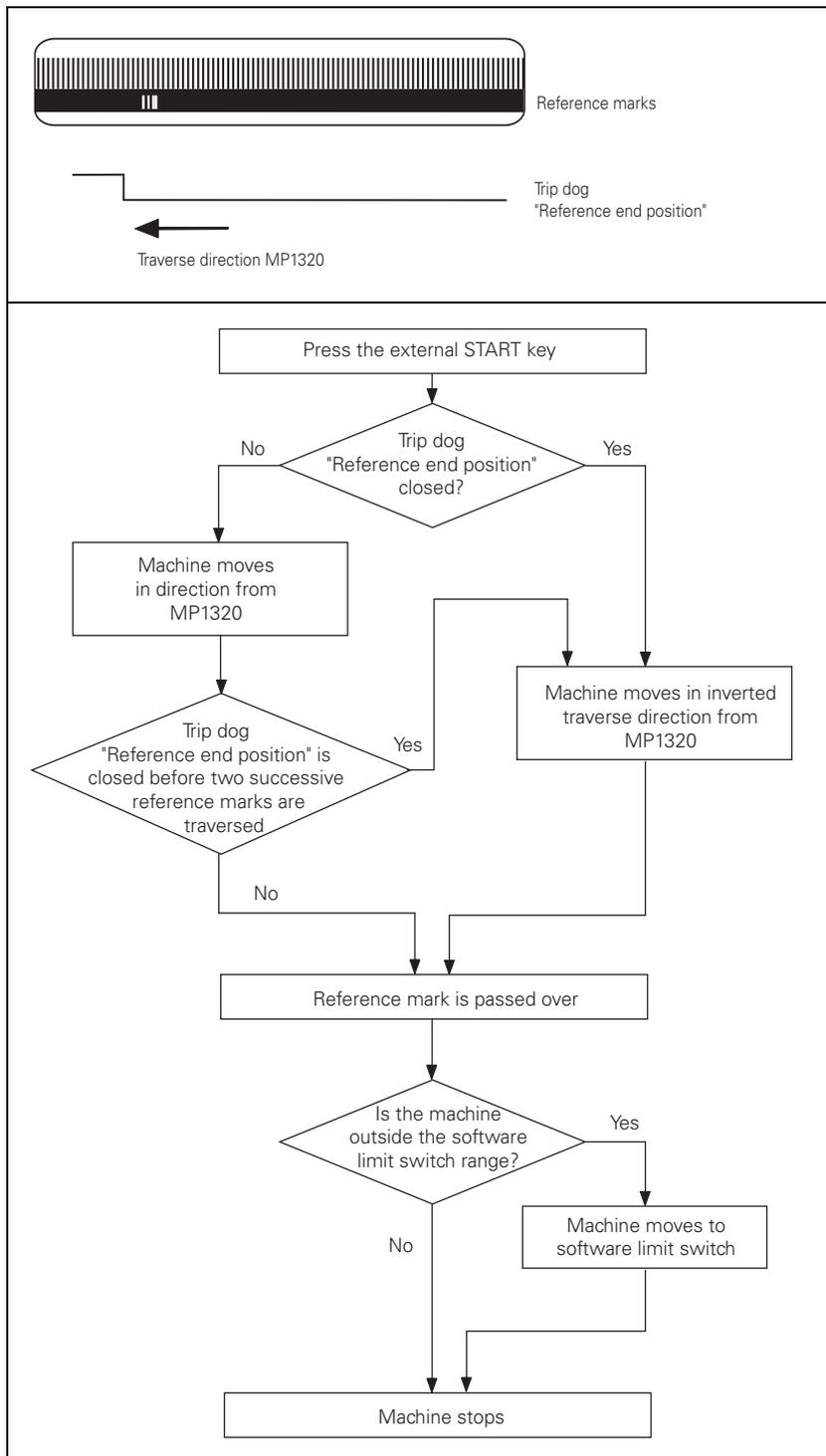
Function when MP1350.x = 0. This setting is used only to ensure compatibility. Do not use for new installations.



If during automatic referencing the trip dog is not closed until it is in the reference end position range, the contouring control will ignore this signal. It is therefore necessary that there are at least two reference marks in the range of the reference end position.

**Position encoder
with one reference
mark**

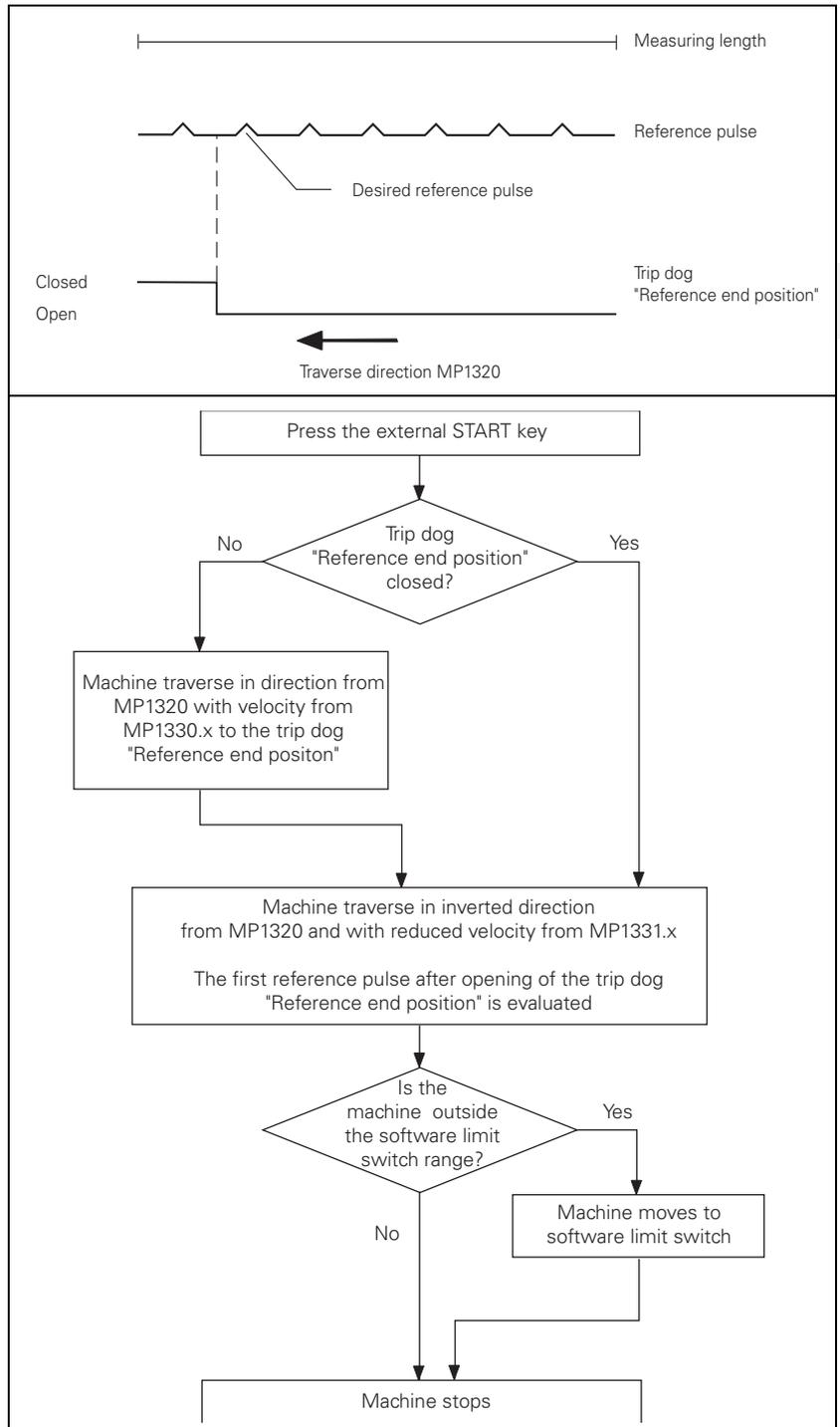
Function when MP1350.x = 1



Linear measurement through rotary encoder

Function when $MP1350.x = 2$

For linear measurement using a rotary encoder, a reference pulse is produced at each revolution of the encoder. Ensure that after machine switch-on the same reference pulse is always evaluated. This can be realized with the trip dog for reference end position.



6.8 The control loop

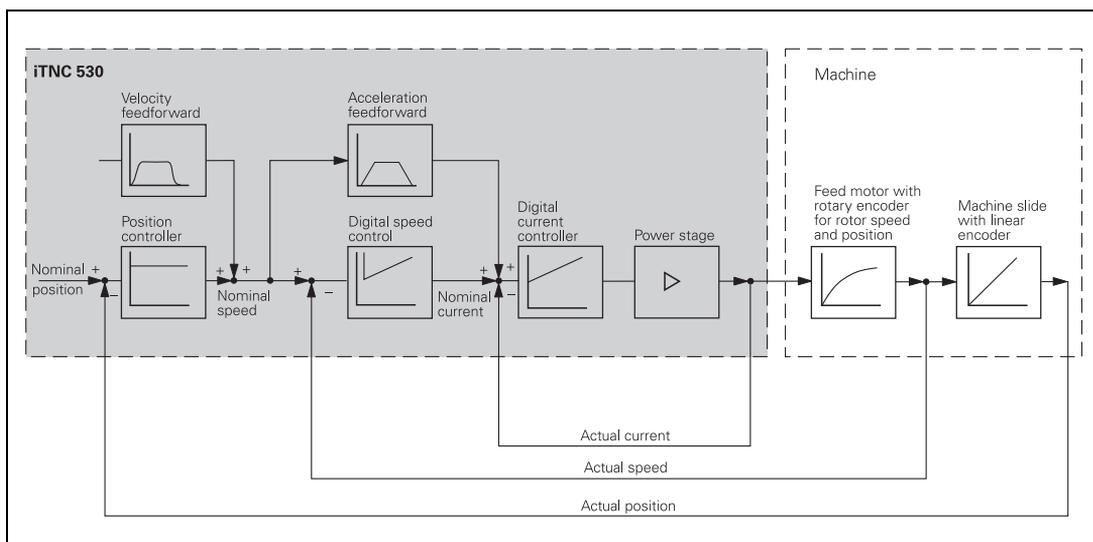
Machine tools normally function on the principle of cascade control. Here the position control loop is prior to the speed and current control loops.

Benefits of cascade control:

- Transparent structure of the individual control loops
- Disturbances can be compensated through the subsequent controllers. This relieves the prior controller.
- The respective outer control loop protects the inner control loop by limiting the command variable.
- Individual commissioning of each control loop, starting with the innermost loop.

The position, speed, and current controllers, and the power module are integrated in the iTNC. The power module is driven by the CC 6xxx through PWM signals (PWM = pulse width modulation).

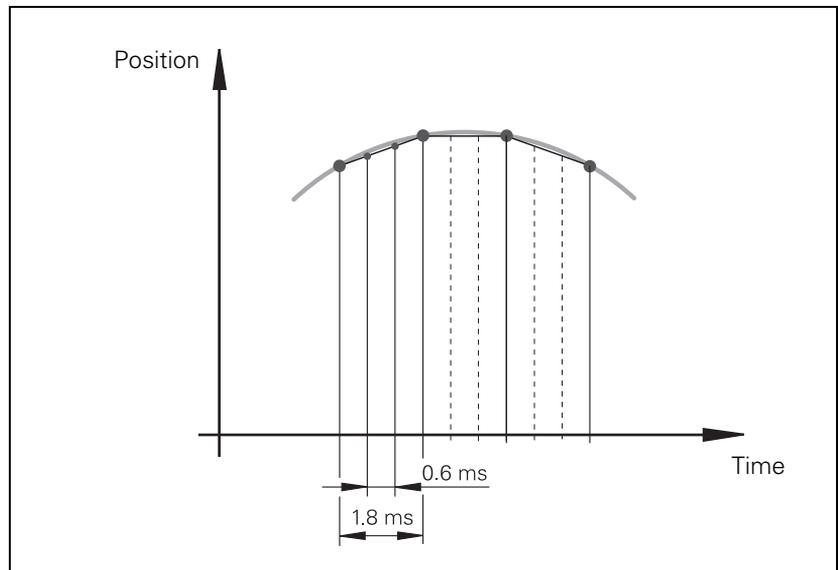
The iTNC 530 controls machines with up to 18 axes and 2 spindles. Spindle speeds up to $40\,000\text{ min}^{-1}$ for motors with two pole pairs are possible.



There is a separate time interval for each control loop:

- Position controller cycle time: Time interval during which the interpolation points on the path are calculated.
- Speed controller cycle time: Time interval in which the actual speed value is compared to the calculated nominal speed value.
- Current controller cycle time: Time interval in which the actual current value is compared to the calculated nominal current value.

The cycle times that apply vary depending on the CC used or the CC's settings.



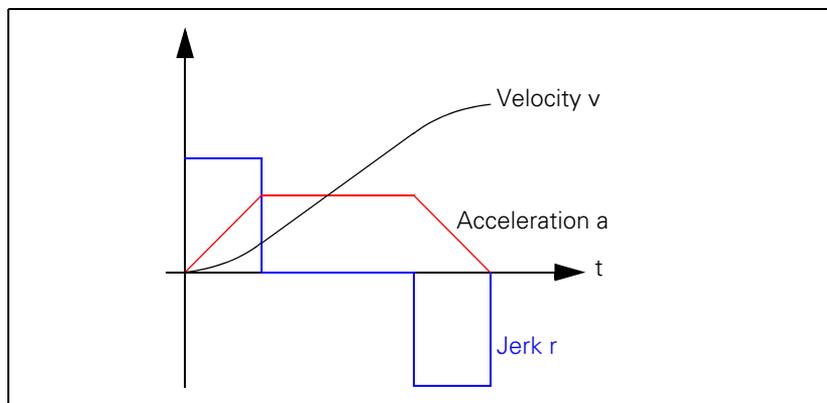
6.8.1 Relation between jerk, acceleration, velocity and distance

To ensure proper operation of an axis, the following two conditions must be fulfilled:

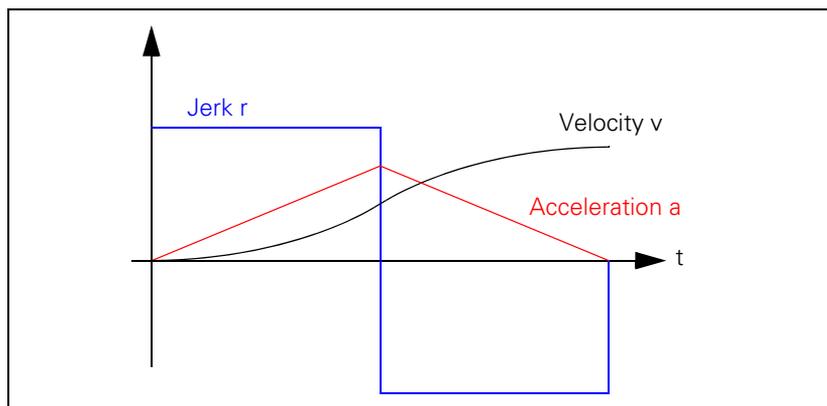
- The desired maximum speed v_{max} and maximum jerk r_{max} result in a maximum acceleration a_{max} .
- A minimum distance s_{min} must be traversed in order to attain the maximum speed v_{max} .

Maximum acceleration

Taking into account the motor and the power module, the machine should be designed in such a way that acceleration during the acceleration phase is as constant as possible. This ensures maximum utilization of the drive current.



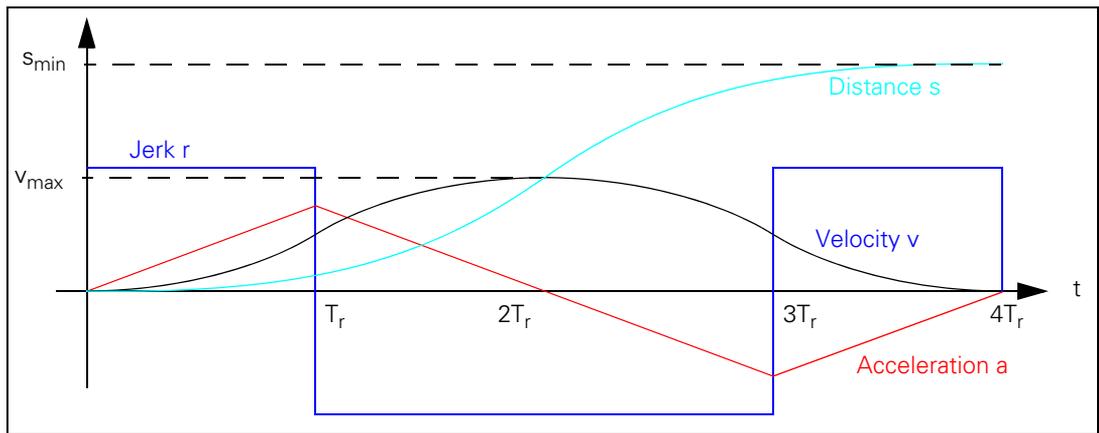
On the other hand, the machine should also be designed to fulfill the dynamic requirements. The jerk should be kept to a minimum and the jerk phase should be maximized in order to prevent the machine from oscillating. The result is no constant acceleration, but a short acceleration peak. If the maximum velocity and the maximum permissible jerk of the machine are preset, the maximum attainable velocity can be determined.



$$a_{max} = \sqrt{v_{max} \cdot r_{max}}$$

Minimum distance

To attain the maximum velocity, a minimum distance s_{min} must be traversed. If the traversed distance is greater than s_{min} , a movement with constant speed is inserted at the time $2T_r$. The minimum distance is:



$$s_{min} = 2 \cdot v_{max} \cdot \sqrt{\frac{v_{max}}{r_{max}}}$$

Example

Rapid traverse $v_{max} = 30\,000$ mm/min (= 0.5 m/s); MP1010.x = 30000
 Max. jerk with velocity $v > 20\,000$ mm/min (= 0.33 m/s) $r_{max1} = 70$ m/s³;
 MP1090.1 = 70, MP1092 = 20000
 Max. jerk $r_{max2} = 35$ m/s³ during machining; MP1090.0 = 35
 Maximum attainable acceleration a_{max1} during rapid traverse:

$$a_{max1} = \sqrt{v_{max} \cdot r_{max1}} = \sqrt{0,5 \frac{m}{s} \cdot 70 \frac{m}{s^3}} = 5,92 \frac{m}{s^2}$$

Maximum attainable acceleration a_{max2} during machining (v up to 20 000 mm/min):

$$a_{max2} = \sqrt{v_{max} \cdot r_{max2}} = \sqrt{0,33 \frac{m}{s} \cdot 35 \frac{m}{s^3}} = 3,40 \frac{m}{s^2}$$

Distance s_{min} required to attain rapid-traverse velocity:

$$s_{min} = 2 \cdot v_{max} \cdot \sqrt{\frac{v_{max}}{r_{max}}} = 2 \cdot 0,5 \frac{m}{s} \cdot \sqrt{\frac{0,5 \frac{m}{s}}{70 \frac{m}{s^3}}} = 0,085 \text{ m} = 85 \text{ mm}$$

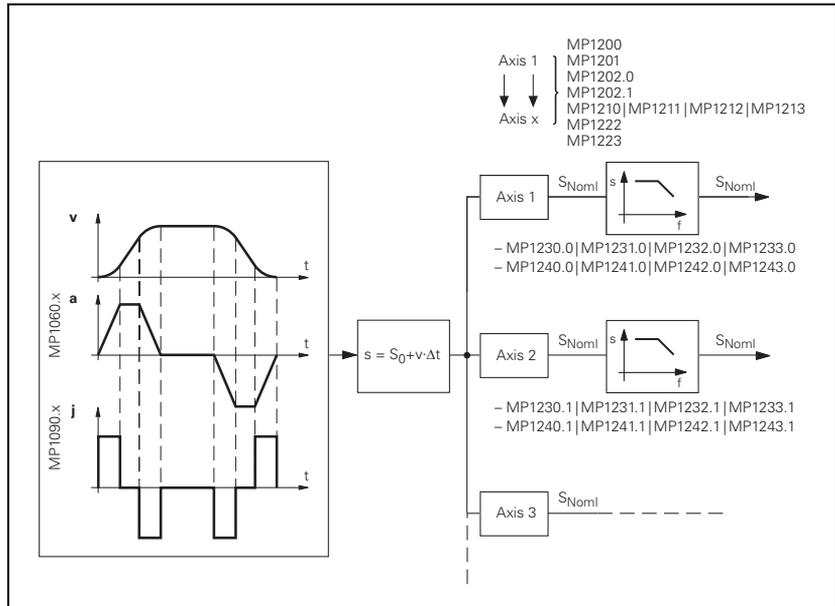


Note

The rectangular jerk curve is rounded through the use of a nominal position value filter (MP1096.x ≠ 0). As a result, acceleration is reduced and the minimum distance required for attaining the maximum velocity is increased.

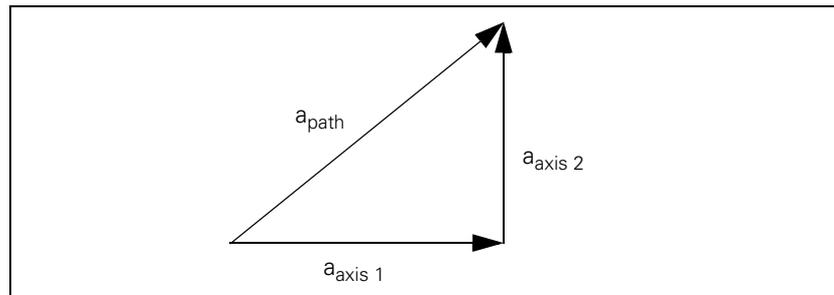
6.8.2 Interpolator

Schematic of the Interpolator:



The interpolator calculates a velocity from the programmed feed rate. In the CC 6xxx it makes this calculation every 3 ms. The value is also dependent on the acceleration curve and the end position.

If more than one axis is moved simultaneously, the path acceleration a_{path} is formed from the appropriate axis components. The same applies to rapid traverse in the path (see "Rapid traverse and feed rate limiting" on page 839).



If the inverter is not designed for such accelerations, you can limit the path acceleration:

- ▶ In MP1061.0, enter the maximum permissible path acceleration for the machining feed rate.
- ▶ In MP1061.1, enter the maximum permissible path acceleration for the rapid traverse.

This division enables you to enter a lower value for the acceleration at machining feed rate, which makes it possible to optimize the behavior of drives with regard to backlash compensation.

You must adjust the velocity feedforward value to the dynamics of the machine:

- ▶ With MP1060.x you define the acceleration or the steepness of the velocity curve.
- ▶ In MP1090.x, you limit the jerk for the **Program run full sequence** and **Program run single block** modes of operation. Just as the acceleration is the rate of change in velocity, the jerk is the rate of change in acceleration. The greater the entered value, the more the system will tend to oscillate.
- ▶ With MP1085.x you limit the axis-specific jerk for path contours in the **Program Run, Single Block, Program Run, Full Sequence** and **MDI** operating modes, as long as the feed rate is not equal to FMAX or is less than the value in MP1092. Until now, this value was set for all axes in MP1090.0.
The value in MP1090.0 will continue to be used as the maximum permissible jerk for motions in all axes. Even with interpolating axes this value is the maximum jerk for the entire machine.
The formula shown can be used to calculate a guide value for MP1085. The formula is conceived for the calculated jerk to be large enough that the acceleration of the axis is not impaired.
Input: 0.1 to 1000 [m/s³]

$$MP1085 \geq \frac{MP1060^2 \cdot 60000}{MP1010}$$

- ▶ With MP1086.x you limit the axis-specific jerk for single-axis motions at rapid traverse in the **Program Run, Single Block, Program Run, Full Sequence** and **MDI** operating modes, if the feed rate is FMAX or is greater than the value in MP1092. Until now, this value was set for all axes in MP1090.1.
The value in MP1090.1 will continue to be used as the maximum permissible jerk for rapid-traverse movements in all axes. Even with interpolating axes this value is the maximum jerk for the entire machine.
The formula shown can be used to calculate a guide value for MP1086. The formula is conceived for the calculated jerk to be large enough that the acceleration of the axis is not impaired.
Input: 0.1 to 1000 [m/s³]
0: Not active (then also for FMAX value in MP1085.x)

$$MP1086 \geq \frac{MP1060^2 \cdot 60000}{MP1010}$$



Note

HEIDENHAIN recommends entering the permissible jerk for each axis in **MP1085.x** and **MP 1086.x**. This way the jerk is based on the weakest axis participating in a motion. The value in MP1090.x should be chosen correspondingly greater than the largest axis-specific jerk (or the value 0 should be entered to omit limiting by MP1090.x) so that the control has the possibility of using the optimum jerk and therefore the optimum acceleration for interpolating axes.

- ▶ Use MP1087.x to limit the axis-specific jerk in **Manual mode**.
- ▶ For unfiltered positioning movements, such as PLC positioning movements or positioning movements in measuring cycles, MP1088.x can be used to define axis-specific jerk limitation.
- ▶ Use MP1089.x to limit the axis-specific jerk in the **Pass Over Reference Point** mode of operation. This is necessary if you want to brake or accelerate faster in this operating mode than in other operating modes.

Please note:

$$\text{Jerk} = \frac{a^2}{v}$$

At high feed rates (e.g. rapid traverse) a higher jerk is permitted than at low feed rates:

- ▶ Enter the jerk for low feed rates in MP1090.0, and for high feed rates in MP1090.1. MP1090.x is the jerk on the tool path. The input value is determined by the weakest axis.
- ▶ With MP1090.x you can limit the path jerk that results from MP1085.x or MP1086.x. MP1090.0 takes effect for movements not at FMAX or if the feed rate is less than the value in MP1092. MP1090.1 takes effect for movements at FMAX or if the feed rate is greater than the value in MP1092. This limitation can be switched off by entering the value 0 in MP1090.x.
- ▶ With MP1092 you specify a feed-rate threshold for the various jerk parameters in MP1085.x and MP1086.x. If the programmed feed rate is less than the value in MP1092, the parameters in MP1085.x apply. If the programmed feed rate is greater than the value in MP1092, the parameters in MP1086.x apply. This way you have the same behavior as previously for all axes via MP1090.0 and MP1090.1. Enter the value 0 in MP1092 to deactivate this feed-rate limit. Then the jerk parameters from MP1085.x apply to movements not at rapid traverse. The values in MP1086.x become active for movements at FMAX.



A nominal position value is acquired every 1.8 ms from the calculated velocity.
For linear interpolation:

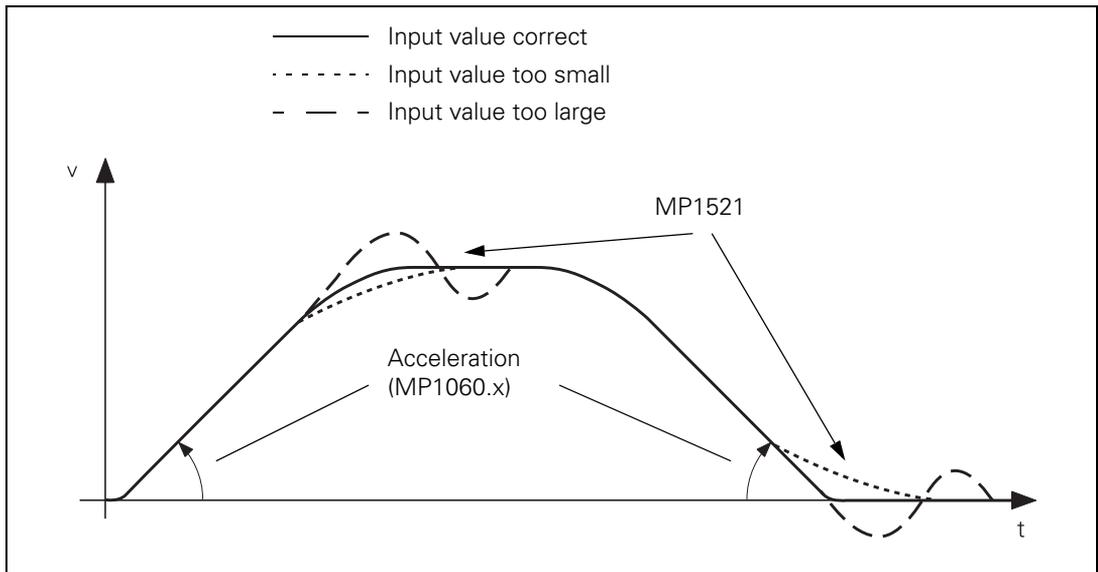
$$s = s_0 + v \cdot \Delta t$$

- s = nominal position value
- s₀ = previous nominal position value
- v = calculated velocity
- Δt = cycle time

The nominal position value is resolved into the individual axis components, depending on which axes have been programmed.

It may happen that the axes at first move past the target position and then oscillate onto it. This overshoot behavior during acceleration and braking can be influenced by a time constant:

- In MP1521, define the time constant for the overshoot behavior.



MP1060.x	Acceleration
Input:	0.001 to 500 [m/s ²] or [1000°/s ²]
MP1061.0	Limitation of the path acceleration for machining feed rate
Input:	0.001 to 500 [m/s ²] or [1000°/s ²]
MP1061.1	Limitation of the path acceleration for rapid traverse
Input:	0.001 to 500 [m/s ²] or [1000°/s ²]
MP1085.x	Axis-specific jerk for path movements in the operating modes Program Run Full Sequence, Program Run Single Block, and Positioning with Manual Data Input
Input:	0.0 to 9999.9 [m/s ³ or 1000° /s ³]
MP1086.x	Maximum permissible jerk during single-axis movements at rapid traverse for the operating modes Program Run Full Sequence, Program Run Single Block, and Positioning with Manual Data Input
Input:	0: Function inactive 0.0 to 9999.9 [m/s ³ or 1000° /s ³]
MP1087.x	Maximum permissible axis-specific jerk for Manual operating mode
Input:	0.1 to 1000.0 [m/s ³ or 1000°/s ³]
MP1088.x	Axis-specific jerk limiting for unfiltered positioning movements
Input:	0.1 to 9999.9 [m/s ³ or 1000° /s ³]
MP1089.x	Max. permissible axis-specific jerk for Pass Over Reference Point mode
Input:	0.1 to 1000.0 [m/s ³ or 1000°/s ³]
MP1090.x	Maximum permissible jerk on the tool path
Input:	0.0 to 9999.9 [m/s ³ or 1000° /s ³] 0: Not active
MP1090.0	With machining feed rate
MP1090.1	Beginning with feed rate from MP1092
MP1092	Feed rate threshold for jerk parameters from MP1085.x and MP1086.x
Input:	1 to 300 000 [mm/min] 0: Not active
MP1521	Transient response during acceleration and deceleration
Input:	1 to 255 [ms] 0: Function inactive



Feed-rate smoothing with MP1522

In addition to the nominal position value filters, you can activate feed-rate smoothing. This smooths jerks caused by a change in the feed rate. This reduces the machine's tendency to vibrate, without significantly increasing the machining time.

An effective possibility for use is with machine oscillations under approx. 25 Hz. The advantage is that the feed-rate smoothing with MP1522 can dampen the oscillation behavior complementary to the nominal position value filters, which makes it possible to raise the limits of the nominal position value filters again. This could result in shorter workpiece machining times.

It is principally recommended that the frequencies of the nominal position value filters be set in dependency of the feed-rate smoothing.

The following calculation can be used to find the damping achieved by MP1522:

Limit frequency (f) of the feed-rate smoothing

$$f_{g(6dB)} = \frac{1}{3} \times \frac{1000}{MP1522}$$

$$f_{g(12dB)} = \frac{1}{2} \times \frac{1000}{MP1522}$$

Feed-rate smoothing (MP1522) with known oscillation frequency

$$MP1522(6dB) = \frac{1}{3} \times \frac{1000}{f}$$

$$MP1522(12dB) = \frac{1}{2} \times \frac{1000}{f}$$

- ▶ In MP1522, define the time constant according to the formula above for the feed-rate smoothing.
- ▶ Begin the adjustment with a smoothing of the feed rate in the 6-dB range, and increase the value step by step.
- ▶ Check the following error with a suitable NC program (e.g. feedforward adjustment program in TNCopt) until it reaches a minimal value.

Typical input value:

MP1522 = Approx. 0.33 to $0.5 \cdot T_{Machine}$
($T_{Machine}$ = Period of the machine resonance frequency)

Example:

$f_{Machine} = 50$ Hz; $T_{Machine} = (1 / 20$ Hz) = 50 ms;

MP1522 = 0.5×50 ms = 25 ms (input value = 25)



Note

There can be an increase in the following error at corners or contour transitions (change of direction) of an axis. Therefore, do not choose too large a value for MP1522, and only use feed-rate smoothing in combination with the nominal position value filters.

MP1522 Feed-rate smoothing

Input: 0 to 60 [ms]
 0: Function inactive



Nominal position value filter

To attain a high machining velocity, the workpiece contour can be adapted to the machine dynamics by means of a nominal position value filter.

4 low-pass filters are available for limiting the bandwidth of the dynamics of nominal position and speed values.

- Single filter
- Double filter
- HSC filters
 - With Cycle 32 – mode 0 (HSC finishing filter)
 - With Cycle 32 – mode 1 (HSC roughing filter)
- Advanced HSC filters
 - With Cycle 32 – mode 0 (HSC finishing filter)
 - With Cycle 32 – mode 1 (HSC roughing filter)

Single and double filters

Single and double filters are classic low-pass filters that always smooth a contour towards the inside at changes in direction. They can be used when a very high surface quality is required, or when a high machining speed is required and larger tolerances are permitted.

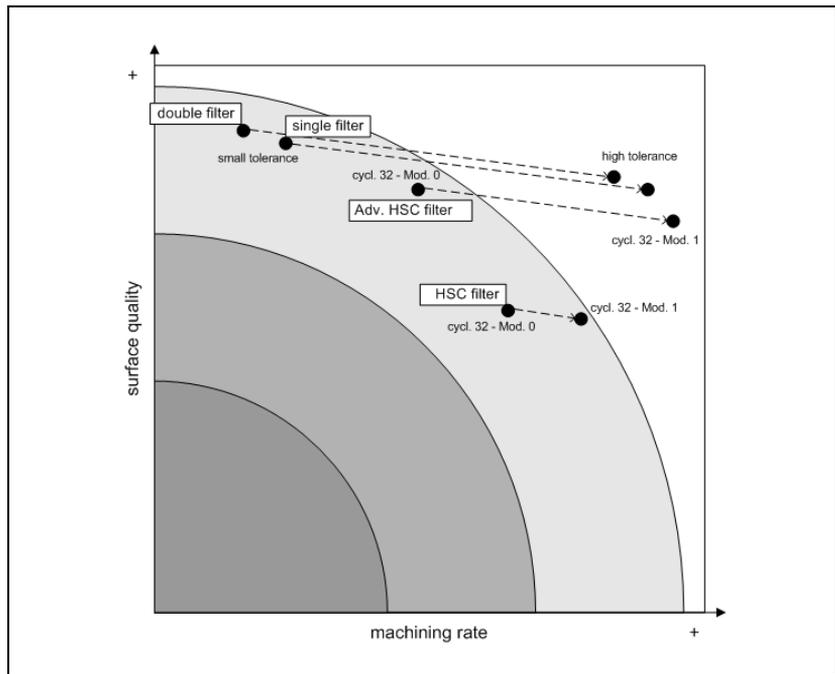
Compared to the double filter, the single filter permits a higher machining speed with a surface quality that is almost always comparable.

Depending on the cycle time, tolerance and axis-jerk limit values, the control automatically increases the filter frequency actually used (previously: filter order).

HSC filter and advanced HSC filter

The speed advantage of both HSC filters is especially large for circular contours. However, you must consider slight overshoots at corners and curvature transitions that are within the given tolerances (MP1202.x or Cycle 32). In general it is recommended that the advanced HSC filter be applied first. It has a more even effect on the speed and any possible overshoots.

The figure below gives a rough overview of when to use the nominal position value filters.



The nominal position value filters (MP1200) function in all operating modes (even in rapid traverse). For RIGID TAPPING (Cycle 17), the nominal position value filter is automatically switched off.

If the machine dynamics are adapted via a nominal position value filter, the iTNC always maintains

- the tolerance
 - from MP1202.0 (at the machining feed rate)
 - from MP1202.1 (at rapid traverse)
 - from the tolerance given in Cycle 32 – mode 0 (HSC finishing filter)
 - from the tolerance given in Cycle 32 – mode 1 (HSC roughing filter). Here the limit frequency is automatically lowered so that faster machining results for larger tolerances.
- the axis-specific jerk at corners (MP1230.x, MP1231.x, MP1232.x and MP1233.x, MP1250.x factor from MP123x.x for rapid traverse) depending on the filter type used
- the axis-specific jerk at curvature transitions (MP1240.x, MP1241.x, MP1242.x and MP1243.x) depending on the filter type used
- the axis-specific acceleration (MP1060.x)
- the path acceleration (MP1061)
- the radial acceleration (MP1070.x)
- the tolerance at curvature changes.

The tolerance consideration at curvature changes can be switched off with MP1222 for HSC filters and MP1223 for advanced HSC filters, which in most cases fully suffices for the accuracy, and has shown in the past that a significantly higher surface quality is achieved. These machine parameters are typically only used in combination with the HSC filter at resonant frequencies under 25 Hz when a high accuracy is required.

Selection criteria and settings for the nominal position value filters

The settings for the nominal position value filters mainly depend on the emphasis of the requirements for machining the workpiece. Speed and accuracy, in connection with clean and smooth surfaces, are the decisive criteria.

At the same time, the oscillation and resonance tendencies of the machining system (the machine tool) are to be considered, and taken into account in the settings for the nominal position value filters. The following recommendations can be made for the settings:

Clean surface

Definition of the term "surface":

- A clean and smooth surface has the highest priority
- Application: Finishing
- Oscillations in the axes must be damped, since following errors of 1 µm are still visible on the surface
- Tolerances are typically between 0.01 mm and 0.02 mm (may be slightly exceeded in order to achieve a better surface)

Settings guidelines for "surface":

- Preferential use of the advanced HSC filter (or single filter)
- Use M124 for a faulty NC program (from CAD system)
- Low jerk values in MP1090.0, MP123x.x
- Switch off tolerance consideration for curvature changes, MP1222 and MP1223 = 0.
- High jerk values for MP124x so that no limitations take effect (example: test MP124x.x up to a value of 1000).
- Radial acceleration MP1070 maximum values between 1 and 3 [m/s²].

Accuracy

Definition of the term "accuracy":

- Maintaining the tolerances has the highest priority.
- Slight oscillations can be seen on the surface.
- Tolerances typically between 0.005 and 0.01 mm.

Settings guidelines for "accuracy":

- Preferential use of HSC filter (MP1200 = 2) (with limit frequencies greater than 30 Hz).
- Lower jerk values in MP1090.0, MP1090.1, MP123x.x, MP124x.x than for "speed".
- Low radial acceleration values with MP1070 (typical values are $< 1.5 \text{ m/s}^2$). Ideally, circular paths should be checked with a KGM grid encoder from HEIDENHAIN. However, in many cases the circular interpolation test with the integrated oscilloscope or TNCopt suffices.
- MP122x = 1 (consideration of tolerance limits at curvature changes).
- The adjustment should be tested with suitable NC programs. The **TNCopt** software from HEIDENHAIN features the "Contour-Single.h" NC program for this. The advantage is that the speed and the contour deviations can be seen directly.

Speed

Definition of the term "speed":

- Surface quality is secondary; short machining times have the highest priority.
- Application: Roughing.
- Tolerances typically between 0.1 and 0.2 mm.

Settings guidelines for "speed":

- Filter selection
 - For large tolerances (greater than $50 \mu\text{m}$), preferential use of single filters (MP1200 = 0)
 - For small tolerances (less than $30 \mu\text{m}$), preferential use of HSC filters (MP1200 = 2). Setting for MP1222 = 0.
- High jerk values in MP1090.0, MP1090.1
- High jerk values in MP123x.x, MP124x.x (typical: factor 3 to 5 of the path jerk)
- High radial acceleration via MP1070 (input value should not exceed the values for linear axes from MP1060.x, maximum $5 \text{ [m/s}^2\text{])}$
- Use M124 for a faulty NC program (from CAD system)
- When using the HSC filter: Program a high tolerance for rotary axes and roughing filters (mode 1) via Cycle 32. Typical values for the rotary-axis tolerances are around 0.01° to 5° . This can achieve significant speed advantages. However, possible undercuts must be taken into account.
- The adjustment should be tested with suitable NC programs. The **TNCopt** software from HEIDENHAIN features the "Contour-Single.h" NC program for this. The advantage is that the speed and the contour deviations can be seen directly.



Settings of the nominal position value filters

- When setting the jerk and acceleration values, as well as selecting the suitable filters, take into account
 - the running noises of the machine
 - the mechanical load (wear)
 - the desired machining speed.
- ▶ Enter the permissible axis-specific jerk:
 - For single filter: MP1230.x (at corners), MP1240.x (at curvature changes, e.g. tangential transition from a line to an arc)
 - For double filter: MP1231.x (at corners), MP1241.x (at curvature changes)
 - For HSC filter: MP1232.x (at corners), MP1242.x (at curvature changes)
 - For advanced HSC filter: MP1233.x (at corners), MP1243.x (at curvature changes)
 - For rapid traverse at corners in MP1250.x: Here a factor of the value from the corresponding MP123x.x is entered. At transitions between rapid traverse motions, the jerk set in MP123x.x multiplied by this factor is active.
- ▶ In MP1202.0, define a tolerance for contour transitions with motions at the machining feed rate. This tolerance can be overwritten by the machine user with Cycle 32 "Tolerance."
- ▶ In MP1202.1, define a tolerance for contour transitions with motions at rapid traverse. This tolerance can **not** be overwritten by the machine user with Cycle 32 "Tolerance."
- ▶ When selecting the limit frequencies for single filters (MP1210), double filters (MP1211), HSC filters (MP1212) and advanced HSC filters (MP1213), take into account the lowest resonant frequency of your machine's axes and the desired damping at this frequency. If the limit frequency is set to 0, the filter is switched off.



Note

The tolerance (MP1202.x, Cycle32) always refers to the nominal value, meaning the servo lag also affects the contour accuracy. For example, if the servo lag $S = 5 \mu\text{m}$ and the tolerance $T = 10 \mu\text{m}$, then the total deviation is $15 \mu\text{m}$.

- ▶ With MP1200, select from the single, double, HSC and advanced HSC filters the nominal position value filters for the **Program Run, Single Block, Program Run, Full Sequence** and **Positioning with Manual Data Input** operating modes.
- ▶ With MP1201, select the nominal position value filters for the **Manual, Handwheel1, Jog Increment** and **Pass Over Reference Point** operating modes. Single and double filters are available here.
- ▶ With MP1222 and MP1223 you specify whether the tolerance from MP1202 or Cycle 32 is also considered for curvature changes. MP1222 affects the HSC filter, and MP1223 affects the advanced HSC filter.
- ▶ In order to achieve the optimum results for your machine or application, test the various filter settings with a test part consisting of short, straight paths.

Tips for setting the nominal position value filter

HSC filter (MP1212)

The HSC filter can be used to limit the bandwidth of the nominal command value. The corresponding limit frequency is entered in MP1212 for this.

If the HSC filter is used with limit frequencies less than 30 Hz, then ensure that the path jerk (MP1090) in dependency of the limit frequency is not set to too large a value. Otherwise, overshoots at the corners could exceed the prescribed tolerances. Independently of this, MP1090 should always be set in a manner such that the following error does not become too large during the jerk phase.

In order to maintain the accuracy at HSC frequencies under 30 Hz, the following table should also be considered:

CC 6xxx controller unit	Frequency to be damped [Hz]							
	10	12.5	15	17.5	20	22.5	25	27.5
MP1212	10	12.5	15	17.5	20	22.5	25	27.5
Max. path jerk for tolerance = 5 µm	18	18	18	18	23	45	90	150
Max. path jerk for tolerance = 10 µm	30	36	36	36	45	100	150	–
Max. path jerk for tolerance = 20 µm	45	45	64	68	90	180	–	–

If the HSC filter does not need to be used to limit the bandwidth, very brief machining times can be achieved by setting high limit frequencies with high jerk values. Of course, requirements for this are increased jerk (MP1232.x, MP1242.x, MP1250.x) and acceleration values (MP1060, MP1070) since starting at certain limit frequencies, the jerk and acceleration values increasingly limit the feed rate. The machining time is briefest at a certain value in MP1212. The following rule of thumb is used to find the optimum value:

$$MP1212 = 23 + MP1232 * 0.2 \text{ (however, MP1212 should be } \geq 30)$$

If the tolerance is increased, the optimum value for MP1212 is shifted to smaller frequencies.

Cutoff frequency for nominal position value filters or filters of rotary axes

Machine parameters MP1218.0 to .2 can be used to allow higher machining speeds in the rotary axes if Cycle 32 with a small angle tolerance (required for toroidal cutters, for example) is used. This also permits an axis-specific configuration with differing drive designs of the rotary axes. Cutoff frequencies for separate nominal-value filters for the A, B and C axes can be entered in these parameters. The filter type is the same as that for the linear axes (MP 1200).

If 0 (default value) is entered, the 32nd-order triangle filter is used for this axis (as before for all rotary axes).



List of the nominal position value filters

The settings for the nominal position value filters are listed below by the MP number.

MP1200	Nominal position value filter
Input:	0: Single filter 1: Double filter 2: HSC filter 3: Advanced HSC filter
MP1201	Nominal position value filter in the Manual, Handwheel, Jog Increment, and Pass Over Reference Point operating modes
Input:	0: Single filter 1: Double filter
MP1202	Predefined tolerance for Cycle 32
Input:	0.0000 to 3.0000 [mm]
MP1202.0	Tolerance at corners for movements at machining feed rate
MP1202.1	Tolerance at corners for movements at rapid traverse
MP1205	Reduction of the starting feed rate
Input:	0: Not active (fast, but somewhat less precise) 1: Active (slow, but likely more precise)
MP1210	Limit frequency for single filter
Input:	0.0 to 166.0 [Hz]
MP1211	Limit frequency for double filter
Input:	0.0 to 166.0 [Hz]
MP1212	Limit frequency for HSC filter
Input:	0.0 to 166.0 [Hz]
MP1213	Limit frequency for advanced HSC filter
Input:	0.0 to 166.0 [Hz]
MP1218.x	Cutoff frequency for nominal position value filters or filters of rotary axes (A, B, C)
Input:	0.0 to 166.0 [Hz] 0: Triangle filter (standard)
MP1222	Tolerance at curvature changes with HSC filter
Input:	(only effective if MP7640 bit 4 = 0) 0: Do not include the tolerance 1: Include the tolerance
MP1223	Tolerance at curvature changes with advanced HSC-filter
Input:	(only effective if MP7640 bit 4 = 0) 0: Do not include the tolerance 1: Include the tolerance

MP1230.x	Max. permissible axis-specific jerk at corners for single filter
Input:	0.1 to 1000.0 [m/s ³]
MP1231.x	Max. permissible axis-specific jerk at corners for double filter
Input:	0.1 to 1000.0 [m/s ³]
MP1232.x	Max. permissible axis-specific jerk at corners for HSC-filter
Input:	0.1 to 1000.0 [m/s ³]
MP1233.x	Max. permissible axis-specific jerk at corners for advanced HSC filter
Input:	0.1 to 1000.0 [m/s ³]
MP1240.x	Max. permissible axis-specific jerk at curvature transitions for single filter
Input:	0.1 to 1000.0 [m/s ³]
MP1241.x	Max. permissible axis-specific jerk at curvature transitions for double filter
Input:	0.1 to 1000.0 [m/s ³]
MP1242.x	Max. permissible axis-specific jerk at curvature transitions for HSC filter
Input:	0.1 to 1000.0 [m/s ³]
MP1243.x	Max. permissible axis-specific jerk at curvature transitions for advanced HSC filter
Input:	0.1 to 1000.0 [m/s ³]
MP1250.x	Max. permissible axis-specific jerk at curvature transitions for advanced HSC filter
Input:	0.1 to 1000.0 [m/s ³]



The following MP7684 should no longer be used in HSCI systems. The nominal position value filters should be adjusted via machine parameters 12xx.

MP7684 Nominal position value filter (bit 0 to bit 4) and path control with M128 or TCPM (bit 5 to bit 7)

Format: %xxxxxxxx

Input: Bit 0 - Nominal position value filter
0: Include acceleration
1: Do not include the acceleration
Bit 1 – Nominal position value filter
0: Include the jerk at corners
1: Do not include the jerk at corners
Bit 2 – Nominal position value filter
0: Include the tolerance
1: Do not include the tolerance
Bit 3 – Nominal position value filter
0: Include the radial acceleration
1: Do not include the radial acceleration
Bit 4 – Nominal position value filter (has priority over MP1222 and MP1223)
0: Jerk and tolerance limits at changes in the curvature are included
1: Jerk and tolerance limits at changes in the curvature are not included
Bit 5 – Reserved
Bit 6 – Feed rate with M128 or FUNCTION TCPM
0: Feed rate profile improved. The feed rate at the TCP is kept constant while maintaining the parameters for acceleration, jerk and tolerance
1: Feed rate profile like up to and including 60642x-01.
Bit 7 – Detection of curvature changes
0: Detection of curvature changes improved.
1: Detection of curvature changes like up to and including 60642x-01.
Bit 8 – Reserved
Bit 9 – Accelerated 5-axis machining with M128 with many rotary axis motions that are less than 2° per positioning block (not with handwheel superimpositioning with M118)
0: Inactive
1: Active
Bit 10 - Modification of the calculation of the contouring feed rate at the beginning of a contour element
0: Active
1: Inactive



HEIDENHAIN generally recommends using the MP12xx machine parameters for the nominal position value filters. The following table provides an overview of the parameters.

MP	Function	Possible input value
1200	Selection of the filter type	0 = Single filter 1 = Double filter 2 = HSC filter 3 = Advanced HSC filter
1201	Nominal position value filter in manual operation	0 = Single filter 1 = Double filter
1202.0 1202.1	Tolerance for contour transitions at corners With machining feed rate With rapid traverse	0.0000 to 3.0000 [mm] 0.0000 to 3.0000 [mm]
1210	Limit frequency for single filter	0.0 to 166.0 [Hz]
1211	Limit frequency for double filter	0.0 to 166.0 [Hz]
1212	Limit frequency for HSC filter	0.0 to 166.0 [Hz]
1213	Limit frequency for advanced HSC filter	0.0 to 166.0 [Hz]
1218.x	Cutoff frequency for single filters of rotary axes	0.0 to 166.0 [Hz]
1222	HSC filter: Tolerance for curvature transitions	0 = Do not include the tolerance 1 = Include the tolerance
1223	Advanced HSC filter: Tolerance for curvature transitions	0 = Do not include the tolerance 1 = Include the tolerance
1230.x	Max. permissible axis-specific jerk for single filter	0.1 to 1000.0 [m/s ³]
1231.x	Max. permissible axis-specific jerk for double filter	0.1 to 1000.0 [m/s ³]
1232.x	Max. permissible axis-specific jerk for HSC filter	0.1 to 1000.0 [m/s ³]
1233.x	Max. permissible axis-specific jerk for advanced HSC filter	0.1 to 1000.0 [m/s ³]
1240.x	Max. permissible axis-specific jerk at curvature transitions for single filter	0.1 to 1000.0 [m/s ³]
1241.x	Max. permissible axis-specific jerk at curvature transitions for double filter	0.1 to 1000.0 [m/s ³]
1242.x	Max. permissible axis-specific jerk at curvature transitions for HSC filter	0.1 to 1000.0 [m/s ³]
1243.x	Max. permissible axis-specific jerk at curvature transitions for advanced HSC filter	0.1 to 1000.0 [m/s ³]
1250.x	Factor for axis-specific jerk at corners at rapid traverse (from value in MP123x.x)	0.0000 to 30.0000
1262	Filter order used for HSC filters (only useful if LIFTOFF is used during powerfail with CC 6xxx)	15 to 31
1263	Filter order used for advanced HSC filters (useful only if LIFTOFF is used during powerfail with CC 6xxx)	15 to 31



Tolerance consideration with M128 or TCPM

When working with M128 or TCPM, the rotary axes necessitate compensating movements in the main axes. Since rotary axes mostly do not have the same dynamics that linear axes have, machining is slower when rotary axes are involved.

During program run with M128 or TCPM, the head dimensions are also included in the tolerance consideration (MP1202.x, Cycle 32).

A higher tolerance is usually permissible for rotary axes, since this does not affect the accuracy (when using a spherical cutter). This higher tolerance leads to shorter positioning times of the rotary axes and the better surfaces:

- ▶ Under **TA** in Cycle 32, program a separate tolerance for rotary axes.

Because of this tolerance, a separate nominal position value filter is used for rotary axes.

6.8.3 Position controller

Position controller cycle time

You can choose between two types of feedback control:

- Feedback control with following error (servo lag)
- Feedback control with velocity feedforward
- ▶ Select the type of control in the **Positioning with manual data input, Program run, single block** and **Program run, full sequence** operating modes with MP1392.
- ▶ Select the type of control in the **Manual** and **Handwheel** modes of operation with MP1391.



Note

The machine must always be adjusted for both types of control.

MP1392 Velocity feedforward in the operating modes Program Run Single Block, Program Run Full Sequence and Positioning with Manual Data Input

Format: %xxxxxxxxxxxxxxxxxxxx
Input: Bits 0 to 17 represent axes 1 to 18:
0: Operation with following error (lag)
1: Operation with velocity feedforward control



Note

M90 (following error mode: Constant contouring speed at corners) is effective only if operation with following error is selected for all axes (MP1392 = %000000000000000000).

MP1391.0 Velocity feedforward control in the MANUAL and HANDWHEEL operating modes

Format: %xxxxxxxxxxxxxxxxxxxx
Input: Bits 0 to 17 represent axes 1 to 18:
0: Inactive
1: Active

MP1391.1 Acceleration feedforward control in the MANUAL and HANDWHEEL operating modes

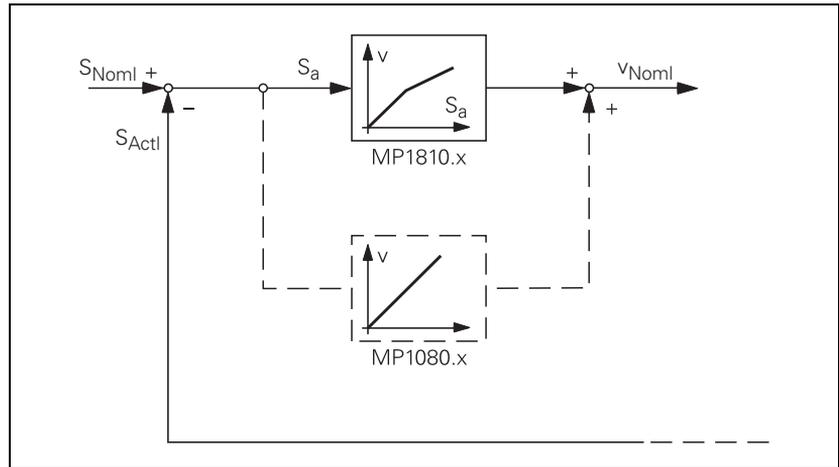
Format: %xxxxxxxxxxxxxxxxxxxx
Input: Bits 0 to 17 represent axes 1 to 18:
0: Inactive
1: Active



Feedback control with following error

Following error (also known as servo lag) is a gap that remains between the nominal position commanded by the NC and the actual position.

Simplified representation:



The nominal position value s_{noml} for a given axis is compared with the actual position value s_{actl} and the resulting difference is the following error s_a :

$$s_a = s_{noml} - s_{actl}$$

s_a = following error
 s_{noml} = nominal position value
 s_{actl} = actual position value

The following error is multiplied by the k_v factor and passed on as nominal velocity value:

$$v_{noml} = k_v \cdot s_a$$

v_{noml} = nominal velocity value

Analog axes (control via CMA-H):

For stationary axes, the integral factor has an additional effect (MP1080.x). It produces an offset adjustment.

Digital axes:

There is no offset. MP1080.x has no function.

k_v factor during control with following error

The control loop gain, known as the k_v factor, defines the amplification of the position control loop. You must find the optimum k_v factor by trial and error.

If you choose a k_v factor that is too large, the following error will become very small. However, this can lead to oscillations.

If you choose too small a k_v factor, the axis will move to a new position too slowly.

For axes that are interpolated with each other, the k_v factors must be equal to prevent contour deviations.

▶ In MP1810.x define a set of k_v factors for operation with following error.

You can selectively increase the contour accuracy with a higher k_v factor. This k_v factor is activated with the M function M105:

▶ In MP1815.x define a second set of k_v factors and activate them with M105.

M105 also influences compensation of reversal spikes during circular motion. With M106 you can switch back to the original set of k_v factors:

▶ Enable the M functions M105/M106 with MP7440, bit 3.

Interrelation of k_v factor, feed rate, and following error

The following formula shows the interrelation of k_v factor, feed rate, and following error:

$$k_v = \frac{v_e}{s_a} \quad \text{or} \quad s_a = \frac{v_e}{k_v}$$

k_v = loop gain [(m/min)/mm]

v_e = rapid traverse [m/min]

s_a = following error [mm]

MP1810.x k_v factor for control with following error

Input: 0.100 to 20.000 [(m/min)/mm]

MP1815.x k_v factor for control with following error effective after M105

Input: 0.100 to 20.000 [(m/min)/mm]

MP7440 Output of M functions

Format: %xxxxx

Input: Bit 3 – Switching the k_v factors with M105/M106

0: Function is not in effect

1: Function is effective

Feedback control with velocity feedforward

The nominal velocity value consists of an open-loop and a closed-loop component.

With velocity feedforward control, the machine-adjusted nominal velocity value is the open-loop controlled component. The closed-loop velocity component is calculated through the following error. The following error is small.

In most cases, machines are controlled with velocity feedforward, since it makes it possible to machine exact contours even at high speeds.

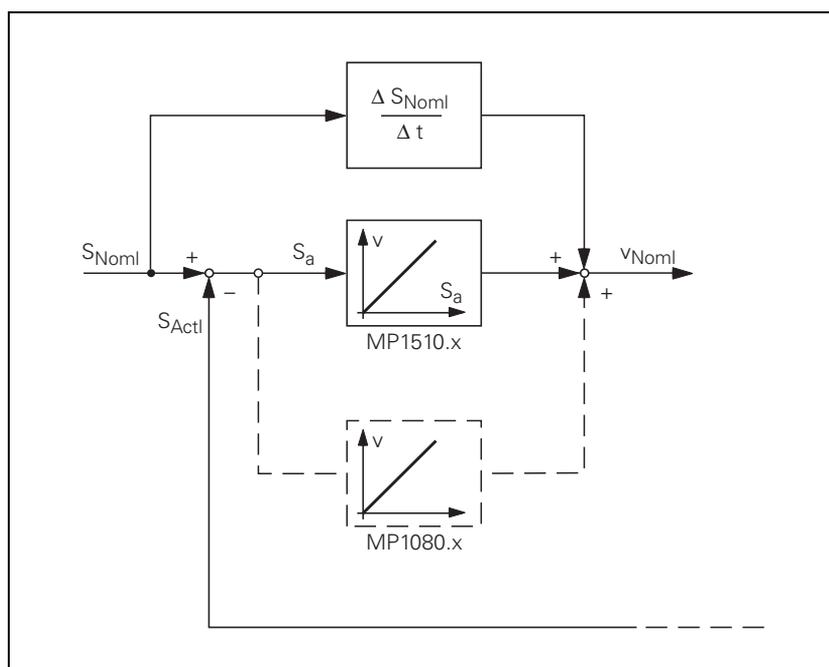
For the **Positioning with manual data input, Program run, single block** and **Program run, full sequence** operating modes:

- ▶ Switch on the velocity feedforward control with MP1392.

For the **Manual** and **E1. Handwheel** operating modes:

- ▶ Switch on the velocity feedforward control with MP1391.

Block diagram:



Analog axes (control via CMA-H):

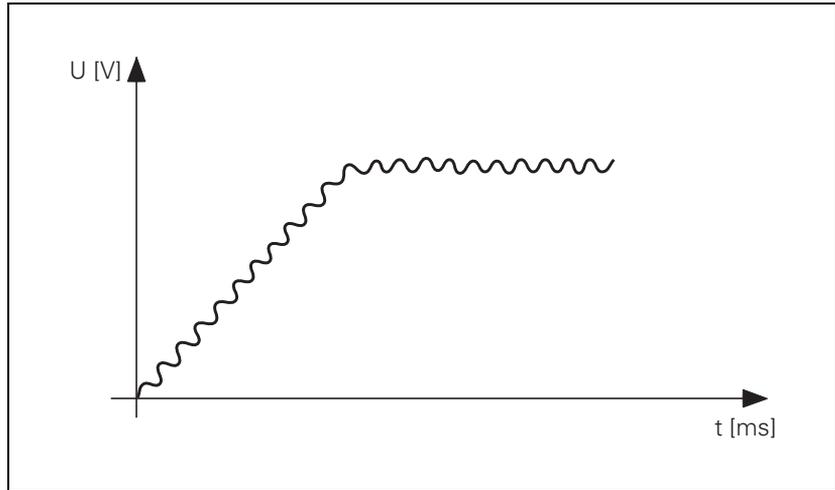
For stationary axes, the integral factor has an additional effect (MP1080.x). It produces an offset adjustment.

Digital axes:

There is no offset. MP1080.x has no function.

You can influence feedback control with velocity feedforward with the k_v factor:

- ▶ In MP1510.x, enter a k_v factor.



Attention

If the k_v factor that you select is too large, the system will oscillate around the forward-fed nominal velocity value.

Unlike operation with following error, you must also enter the optimum k_v factor for each axis when operating with interpolated axes.

You can selectively increase the contour accuracy with a higher k_v factor. This k_v factor is activated with M105:

- ▶ In MP1515.x, define a second set of k_v factors and activate them with M105.

M105 also influences compensation of reversal spikes during circular motion. With M106 you can switch back to the original set of k_v factors:

- ▶ Enable the M functions M105/M106 with MP7440, bit 3.

MP1510.x **k_v factor for velocity feedforward control**

Input: 0.100 to 1000.000 [(m/min)/mm]

MP1515.x **k_v factor for velocity feedforward control effective after M105**

Input: 0.100 to 1000.000 [(m/min)/mm]

MP7440 **Output of M functions**

Format: %xxxxx

Input: Bit 3 – Switching the k_v factors with M105/M106:

0: Function is not in effect

1: Function is effective



Feedback control with velocity semifeedforward

MP1396.x allows the operator to switch to velocity semifeedforward control. Normally, work will be carried out using velocity feedforward. Velocity semifeedforward is activated, for example, by an OEM cycle before roughing, in order to permit a higher following error and thereby a higher velocity, combined with a lowered accuracy, in order to traverse corners. Before finishing, another OEM cycle can be used to switch back to velocity feedforward, in order to finish with the highest accuracy possible.

In order to use velocity semifeedforward, a factor must be entered for every axis in MP1396.x, where values toward 0 control the following error more, and values toward 1 control the velocity feedforward more. The factor can be overwritten with FN17: SYSWRITE ID600. However, the factor from MP1396.x becomes valid again at the end of a program (MP7300 = 1) and whenever a new NC program is selected.

As soon as a factor between 0.001 and 0.999 has been entered in MP1396.x, the k_v factor from MP1516.x becomes effective.



Note

For axes that are interpolated with each other, the k_v factors must be equal. In this case the smaller k_v factor determines the input value for these axes.

The values for position monitoring are interpolated according to the factor in MP1396.x between the values for servo lag (MP1710.x, MP1720.x) and the values for velocity feedforward control (MP1410.x, MP1420.x).

Feedback control with following error (servo lag)	Feedback control with velocity semifeedforward	Feedback control with velocity feedforward
MP1391 bit x = 0 MP1392 bit x = 0 MP1396.x = nonfunctional	MP1391 bit x = 1 MP1392 bit x = 1 MP1396.x = 0.001 MP1396.x = 0.999	MP1391 bit x = 1 MP1392 bit x = 1 MP1396.x = 1
	←—————→	

If the PC Marker M4625 is set, the start of an NC axis for which operation with following error or velocity semifeedforward control is active is prevented, and the error message **Semifedforward not permitted** is displayed. The marker is only evaluated when an NC program is started or when the axes are traversed with axis-direction keys.



To use feedback control with velocity semifeedforward:

- ▶ Activate the velocity feedforward control with MP1391 and/or MP1392.
- ▶ Determine the k_v factor for velocity feedforward control (MP1510.x).
- ▶ Activate the velocity semifeedforward control by entering the desired factor in MP1396.x.
- ▶ Determine the k_v factor for velocity semifeedforward control (MP1516.x).
- ▶ Enter MP1396.x = 1 to return to velocity feedforward control.
- ▶ For example, you may now use FN17: SYSWRITE ID 600 in an OEM cycle to overwrite or reestablish the factor in MP1396.x.

MP1396.x Feedback control with velocity semifeedforward

Input: 0.001 to 0.999
1: Velocity feedforward control

MP1516.x kv Factor for velocity semifeedforward

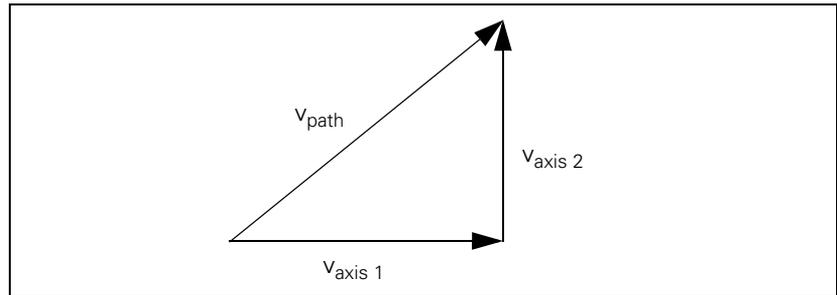
Input: 0.100 to 20.000 [(m/min)/mm]

		Set	Reset
M4625	Disable NC axes when velocity semifeedforward control is active	PLC	PLC



Rapid traverse and feed rate limiting

If more than one axis is moved simultaneously, the rapid traverse on the path v_{path} is formed from the appropriate axis components. The same also applies to the path acceleration (see "Interpolator" on page 814).



If there are problems with the inverter, e.g. because the energy being generated from the axes in motion cannot be dissipated, you can limit the rapid traverse on the path:

- ▶ Enter the maximum rapid traverse on the path in MP1011.
- ▶ Enter the inputs axis-specifically in MP1010.x.
- ▶ With MP1012.x you can enter a second axis-specific rapid traverse as an alternative to MP1010.x. Use the **FN17: SYSWRITE ID 20 NR 18 = <value>** function to switch between the two machine parameters. MP1012 can only be used in the **Program Run, Single Block, and MDI** operating modes.
<value> = 0: MP1010 and MP1011 active
<value> = 1: MP1012 active, MP1011 not active, meaning that the possible increase in the contouring feed rate is not limited.
The rapid traverse from MP1010 in connection with MP1011 is always active at program end and when a program is first selected.

The feed rate can be limited via the PLC:

- ▶ Enter the reduced maximum contouring feed rate in D596.
- ▶ Limit the feed rate axis-specifically with Module 9140. It can be read with Module 9141.

If the value in D596 is greater than MP1011, then MP1011 becomes effective. After the control is switched on, or after an interruption of the PLC run, D596 is preassigned with the value 300 000 so that MP1011 becomes effective.

Rapid traverse can be limited by the user with the F MAX soft key. This limitation is not effective if M4587 is set. In this case only limitation through D596 is effective. After M4587 is reset, both D596 and the last limit set via the F MAX soft key are effective again. Limitation of rapid traverse via the F MAX soft key can be rescinded with the RESET soft key. The value from MP1011 then becomes active again.

The feed rate is significantly lower for manual operation than for rapid traverse:

- ▶ Enter in MP1020 a feed rate for manual operation.

The programmed feed rate and the current path feed rate are saved in D360 and D388 in mm/min. In the manual operating modes, the highest axis feed of all axes is stored in D388.

The maximum possible feed rate depends on the encoder being used.

$$v_{\max} [\text{mm/min}] = P [\text{mm}] \cdot f_i [\text{kHz}] \cdot 60$$

v_{\max}	=	Maximum traverse speed
P	=	Signal period of the encoder
f_i	=	Input frequency of the encoder input, see "Encoder signals" on page 608 and see "Encoder connections" on page 296.

Digital axes:

For digital axes, the maximum feed rate also depends on the number of pole pairs in the drive motor and the pitch of the ball screw.

$$v_{\max} [\text{mm/min}] = \frac{60\,000}{\text{Number of pole pairs}} [\text{1/min}] \cdot \text{ball screw pitch} [\text{mm}] \cdot \frac{f_{\text{PWM}}}{5000}$$

Analog axes:

- ▶ In MP1050.x, enter the desired analog voltage for rapid traverse.
- ▶ Adjust the rapid traverse feed rate (v_{\max}) with the analog voltage at the servo amplifier.

Module 9140 Set axis-specific feed-rate limit

With Module 9140 you can set axis-specific feed-rate limits. The limits are saved in sequential double words for each axis. The address of the starting double word must be given. Along with a feed-rate value (≥ 0), the following limitations are possible in the double words:

- -1: Maximum feed rate (a previous limitation via Module 9140 is rescinded)
- -2: Axis-specific rapid traverse from MP1010.x (for multi-axis interpolation, the contouring feed rate is limited to the slowest respective value of the participating axes)
- -3: Axis-specific manual feed rate from MP1020.x (for multi-axis interpolation, the contouring feed rate is limited to the slowest respective value of the participating axes)



Invalid feed rates are set to 0, and M4203 is set. Other axes retain their limits.

Call:

PS B/W/D/K <Start double word>

PS B/W/D/K <Number of axes or double words>

CM 9140

Error code:

Marker	Value	Meaning
M4203	0	Feed-rate limitation set
	1	Error code in W1022
W1022	1	Invalid feed-rate value (< -3)
	2	Invalid number of axes or double words
	3	Invalid block length as of starting address
	5	Not a double-word address
	24	Module was called in a spawn job or submit job

Module 9141 Read axis-specific feed-rate limit

With Module 9141 you can read axis-specific feed-rate limits. The limits are saved in sequential double words for each axis. Along with feed-rate values (≥ 0), the limitations -1, -2 or -3 (see Module 9140) are also transferred.

Call:

PS B/W/D/K <Start double word>

PS B/W/D/K <Number of axes or double words>

CM 9140

Error code:

Marker	Value	Meaning
M4203	0	Feed-rate limitation set
	1	Error code in W1022
W1022	2	Invalid number of axes or double words
	3	Invalid block length as of starting address
	5	Not a double-word address
	24	Module was called in a spawn job or submit job

MP1010.x	Rapid traverse
Input:	10 to 1 000 000 [mm/min or °/min]
MP1011	Limit of rapid traverse on the path
Input:	10 to 1 000 000 [mm/min or °/min]
MP1012.x	Second axis-specific rapid traverse
Input:	10 to 1 000 000 [mm/min or °/min]
MP1020.x	Manual feed
Input:	10 to 300 000 [mm/min or °/min]
MP1050.x	Analog axes: Analog voltage at rapid traverse
Input:	1.000 to 9.000 [V] Digital axes: without function Input: 1

		Set	Reset
M4587	Rescind feed rate limit above F MAX	PLC	PLC
D596	Max. feed rate from PLC [mm/min]	NC/PLC	PLC
D360	Programmed feed rate	NC	NC
D388	Current contouring feed rate [mm/min]	NC	NC

Position loop resolution

The encoder signals are interpolated 1024-fold.

$$\text{Position loop resolution } [\mu\text{m}] = \frac{\text{Signal period } [\mu\text{m}]}{1024}$$

Analog axes

The iTNC outputs a voltage per position error. The 10 V analog voltage is subdivided with a D/A converter on the CMA-H. This results in a smallest voltage step of 0.328 mV.

Rapid traverse (MP1010.x) is attained at a certain voltage (MP1050.x). This results in the voltage DU per position error or following error s_a :

$$\Delta U = \frac{\text{MP1050.x [mV]}}{S_a [\mu\text{m}]}$$

If DU is divided by the smallest possible voltage step (0.328 mV), the result is the number n of the possible voltage steps per position error.

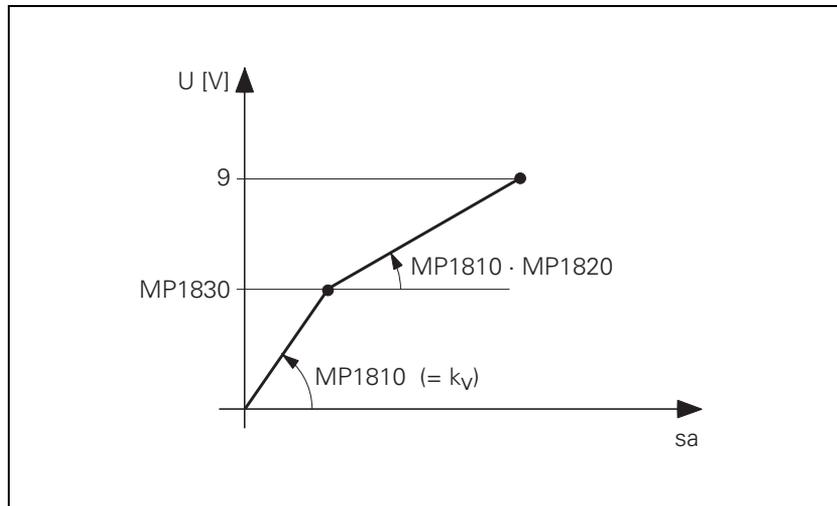


Characteristic curve kink point (for control with following error)

For machines with high rapid traverse, you can not increase the k_v factor enough for an optimum control response to result over the entire velocity range (from standstill to rapid traverse).

In this case you can define a characteristic curve kink point, which has the following advantages:

- High k_v factor in the low range
- Low k_v factor in the upper range (beyond the machining velocity range)
- ▶ Define the position of the characteristic kink with MP1830.x. In the upper range, the k_v factor is multiplied by the factor from MP1820.x.
- ▶ Enter a multiplier in MP1820.



The characteristic curve kink point must lie above the tool feed rate!

Calculating the lag (following error):

$$s_a = \left(\frac{\text{MP1830.x} [\%]}{100 [\%]} + \frac{100 [\%] - \text{MP1830.x} [\%]}{\text{MP1820.x} \cdot 100 [\%]} \right) \cdot \frac{v_e}{k_v}$$

MP1820.x Multiplier for the k_v factor

Input: 0.001 to 1.00000

MP1830.x Characteristic curve kink point

Input: 0.000 to 100.000 [%]

Opening the position control loop

If M4581 has been set, the control opens the loops of all axes and of the spindle, and then performs an NC stop. This makes it possible, for example, to open the position control loops and at the same time switch off the drives. As with an external emergency stop, position monitoring is shut off for the time defined in MP1150.1, and an actual-to-nominal value transfer is executed. After the time defined in MP1150.1 has expired, position monitoring is again active, for at least the time defined in MP1150.2.

If at a standstill or with an external stop, the PLC sets M4581 in the Program Run Full Sequence, Program Run Single Block, or Positioning with MDI operating modes, and then the axes are moved, the **Approach position** function will be activated when the PLC resets M4581.

If MP4020 bit 8 = 1, then, if the axes are moved after an emergency stop, "Approach position" is automatically activated.

If M4580 has been set, an external EMERGENCY STOP ("control is ready") **is not** reported to the NC, but rather the function is executed like M4581.

If the position control loop is opened, the axis release in W1024/W1100 is canceled.

		Set	Reset
M4580	Suppress emergency stop, open all position control loops, NC stop	PLC	PLC
M4581	Open all position control loops, NC stop, activate "Approach position"	PLC	PLC
W1024/1100	Axis enabling Axis-specific: Dependence on MP4020 bit 14 0: Position control loop open 1: Position control loop closed	NC	NC
MP1150.1	Time period for which the monitoring function is to remain off after the fast PLC input defined in MP4130.0 is set Input: 0 to 65.535 [s] Recommended: 0.2 to 0.5		
MP4020	PLC functions Input: Bit 8 – Behavior after an ext. emergency stop 0: "Approach position" is not automatically activated 1: "Approach position" is automatically activated		



Clamping the axes

After running an NC block you can clamp the axes.

- ▶ If necessary, define in MP7494 the axes for which an exact stop is to occur after positioning.
- ▶ Wait until "axis in position" is set in W1026/W1104.
- ▶ Clamp the axis or axes.
- ▶ Open the position control loop with W1040/W1132.
- ▶ With Module 9161, switch the drive off.

A waiting period is necessary between "axis clamping" and "position control loop opening."

- ▶ In W1038/W1128, set the bit for the corresponding axis.

The next NC block is not run until the positioning window has been reached and the position control loop has been opened with W1040/W1132.

If the position control loop is opened, the axis release in W1024/W1100 is canceled. You can link switching off the drives via Module 9161 with W1024/W1100.

If a clamped axis is to be repositioned, the NC cancels the "axis in position" message in W1026/W1104:

- ▶ With Module 9161, switch the drive on.
- ▶ Release the clamping.
- ▶ Close the position control loop with W1040/W1132.

MP7494 Axes for which an exact stop is to occur after positioning

Format: %xxxxxxxxxxxxxxxxxxxx

Input: 0: No exact stop
1: Exact stop

	Set	Reset
W1038/1128 Prepare to open the position control loop Axis-specific: Dependence on MP4020 bit 14 0: Not active 1: Active	PLC	PLC
W1040/1132 Axis-specific opening of the position control loop Axis-specific: Dependence on MP4020 bit 14 0: Do not open the position control loop 1: Open the position control loop	PLC	PLC

Feed-rate enable

To move the axes, you must first enable the feed rate through the PLC. Until "feed-rate enable" is set, the nominal velocity value zero is output. "F" is highlighted in the status display.

Feed-rate enable for all axes:

- ▶ Set M4563.

Axis-specific feed-rate enable:

- ▶ Reset M4563.
- ▶ In W1060/W1172, set the corresponding bits.

		Set	Reset
M4563	Feed-rate enable for all axes	PLC	PLC
W1060/1172	Axis-specific feed-rate enable	PLC	PLC
	Axis-specific: Dependence on MP4020 bit 14		
	0: No feed-rate enable		
	1: Feed-rate enable		



Actual-to-nominal value transfer

During actual-to-nominal value transfer, the current position is saved as the nominal position value. This becomes necessary, for example, if the axis has been moved when the position control loop is open.

There are two ways to turn the actual position into the nominal position:

- ▶ To transfer the actual position value in the **Manual** and **Electronic Handwheel** modes, set the corresponding bit in W1044/W1140.
- ▶ To transfer the actual position in all operating modes, use Module 9145.



Attention

Ensure that actual-to-nominal value transfer does not occur continually, since the position monitoring will not be able to detect any uncontrolled machine movements. In this case no safe machine operation would be possible.

	Set	Reset
W1044/1140 Actual-to-nominal value transfer	PLC	PLC
Axis-specific: Dependence on MP4020 bit 14		
0: No actual-to-nominal value transfer		
1: Actual-to-nominal value transfer		

Module 9145 Actual-to-nominal value transfer

An actual-to-nominal value transfer for NC axes, or for PLC and NC axes together, is possible only if the control is not active (M4176 = 0) or if an M/S/T/T2/G strobe is pending.

The module can always be called for an actual-to-nominal transfer only for PLC axes.

For a transfer via M strobe, MP7440 bit 2 must not be set. For a transfer via S/G strobe, MP3030 must not be set.

Call:

PS B/W/D/K <Axes bit-encoded>

CM 9145

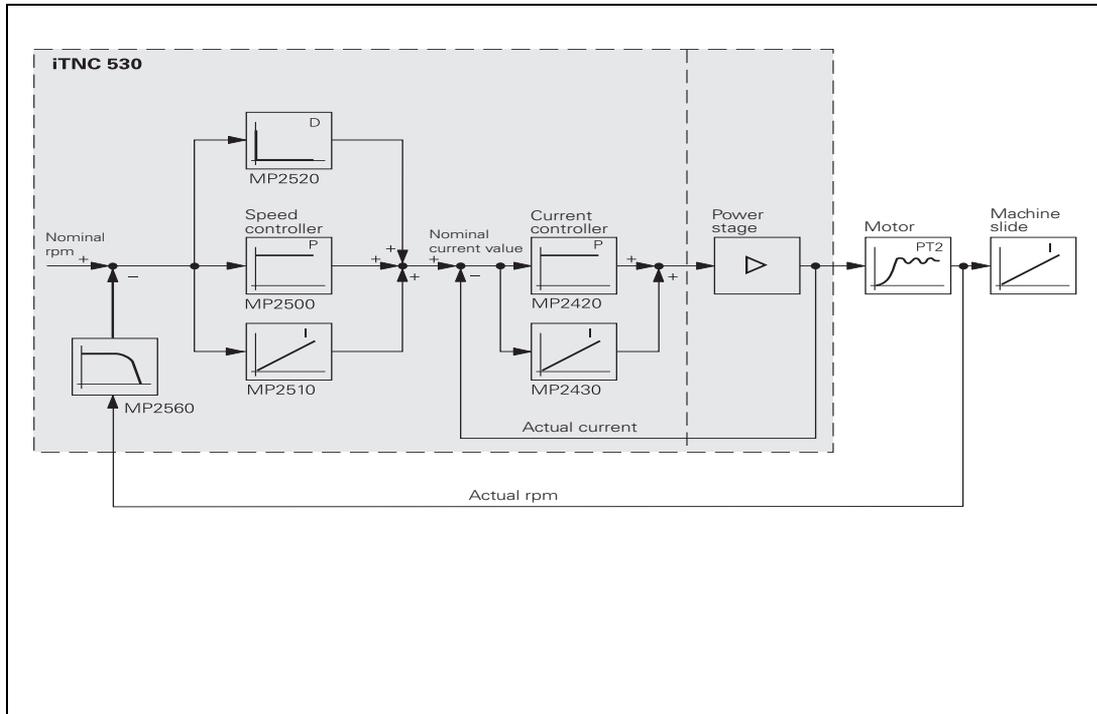
Error code:

Marker	Value	Meaning
M4203	0	Actual-to-nominal value transfer performed
	1	Error code in W1022
W1022	2	Invalid axis number
	21	Missing M/S/T/T2/G strobe in M4176 = 1
	24	Module was called in a spawn or submit job

6.8.4 Speed controller

Up to 20 digital speed controllers for the axes and spindle(s) are possible in the iTNC 530 HSCI:

The actual speed values are measured directly at the motors with HEIDENHAIN rotary encoders. The position controller provides the nominal speed value. The speed controller is driven by the difference between nominal and actual speed values. It provides the nominal current value as output.



"Commissioning" on page 1004.

With Module 9164 you can read the actual speed value of the motors.

You can adjust the step response of the speed controller:

- ▶ With the position controller switched off, enter with MP2500.x a proportional factor and with MP2510.x an integral factor for the speed controller. Adjust the step response so that only one overshoot is visible and the settling time t_{off} is as small as possible. Realistic values for the settling time: 3 ms to 15 ms.

MP2500.x Proportional factor of the speed controller

Input: 0 to 1 000 000.000 [As]

MP2510.x Integral factor of the speed controller

Input: 0 to 100 000 000 [A]

Module 9164 Read the actual speed value of the motor

The resolution of the actual speed value depends on the encoder being used:

$$\text{Resolution} = \frac{1}{\text{Line count} \cdot 1024} \cdot 100\,000 \text{ [min}^{-1}\text{]}$$

Call:

PS B/W/D/K <Axis>

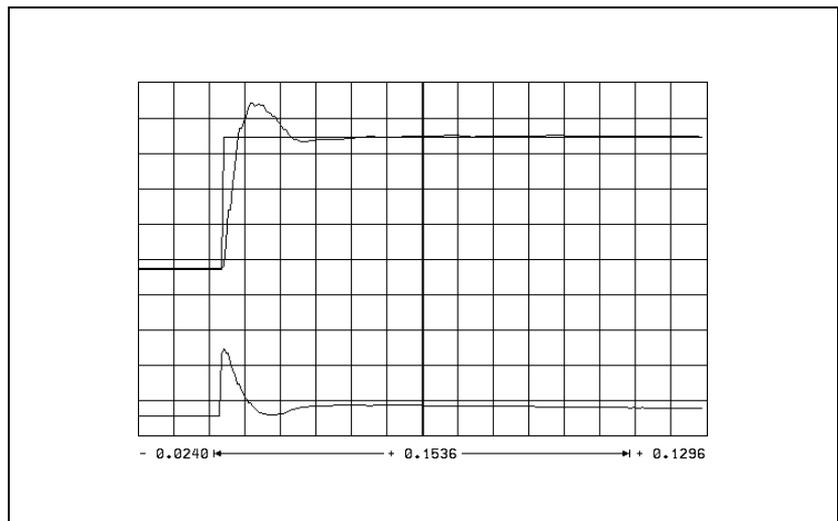
Axis-specific: Dependence on MP4020 bit 14

CM 9164

PL B/W/D <Actual speed value in the format 0.001 [min⁻¹]>

Error code:

Marker	Value	Meaning
M4203	0	Actual speed value was read
	1	Control has no integrated current controller



The step response illustrated above is idealized. In practice, interfering oscillations are superimposed on the step response.

You can reduce these interference oscillations with the differential factor, the PT_2 second-order time-delay element, the band rejection filter and the low-pass filter of the speed controller.

Differential factor

The differential factor can reduce low-frequency oscillations. However, it increases oscillations in the high frequency range.

- ▶ In MP2520.x, enter a differential factor.



Note

Ensure that the system is stable enough!

The differential factor is not recommended on machines with motors that have belt couplings. The influence of aging and temperature is too great.

Estimating the differential factor:

$$\text{MP2520.x} \approx \frac{T \cdot \text{MP2500.x}}{8}$$

MP2520.x: Differential factor of the speed controller [As²]

MP2500.x: Proportional factor of the speed controller

T: Period of the lowest interference frequency [s]

MP2520.x Differential factor of the speed controller

Input: 0 to 1.0000 [As²]

Low-pass filter

With the low-pass filter you can damp high-frequency oscillations (> approx. 450 Hz):

- ▶ Use the oscilloscope to find the fundamental frequency of the iTNC.
- ▶ Activate the 1st or 2nd order low-pass filter (see table).

Fundamental frequency of the interference oscillation	Filter order (MP2560.x)
450 Hz to 550 Hz (approx.)	First order (MP2560.x = 1)
> 550 Hz (approx.)	Second order (MP2560.x = 2)

If you cannot achieve satisfactory results with the low-pass filter, try the PT₂ element.

MP2560.x Low-pass filter of the speed controller

Input: 0: No low-pass filter
1: 1st-order low-pass filter
2: 2nd-order low-pass filter



PT₂ element of the speed controller

If the controlled system is insufficiently damped (e.g. direct motor coupling or roller bearings), it will be impossible to attain a sufficiently short settling time when the step response of the speed controller is adjusted. The step response will oscillate even with a low proportional factor:

- ▶ In MP2530.x, enter a value for damping high-frequency interference oscillations. If the value you choose is too high, the k_v factor of the position controller and the integral factor of the speed controller is reduced. Realistic input values: 0.0003 to 0.0020

MP2530.x PT₂ element of the speed controller (2nd-order delay)

Input: 0 to 1.0000 [s]

Band-rejection filter

With the band-rejection filter you can damp oscillations that you cannot compensate with the differential factor, the PT₂ element, or the low-pass filter:

- ▶ With the oscilloscope of the iTNC, find the fundamental frequency of the interference oscillations and enter them in MP2550.x.
- ▶ Increase MP2540.x incrementally until the interfering oscillation is minimized. If you set the damping too high, you will limit the dynamic performance of the control loop. Realistic input values: 3 to 9 [dB]

MP2540.x Band-rejection filter damping of the speed controller

Input: 0.0 to 18.0 [dB]

MP2550.x Band-rejection filter center frequency of the speed controller

Input: 0.0 to 999.9 [Hz]

Active damping of low-frequency oscillations

The active damping of low-frequency oscillations is suitable for damping noise oscillations of approx. 10 to 30 Hz. The damping factor is set in MP2607.x, and the damping time constant in MP2608.x. It can be calculated according to the following formula:

$$\text{MP2608.x} = \frac{k}{2 \cdot \pi \cdot f}$$

k: Factor from 0.8 to 1.0

f: Frequency to be damped (approx. 10 Hz < f < 30 Hz)



Note

The active damping should only be used if improvements actually occur, since the damping could also lead to lower and higher frequencies being fortified.

Activating the active damping:

- ▶ Ascertain the deepest resonant frequency (e.g. with the frequency diagram in TNCopt when adjusting the IPC and k_V factor).
- ▶ Set $\text{MP2607.x} = 1.5$.
- ▶ Calculate the damping time constant according to the above formula with $k = 0.9$, and enter this value in MP2608.x .
- ▶ Record **I (n int)** or **s diff** with the integrated oscilloscope, and move the axis with the axis-direction buttons.
- ▶ Vary the value of k up and down somewhat (between 0.8 and 1.0), calculate MP2608.x , and compare the oscilloscope recordings in order to find the correct value for MP2608.x (the value with the lowest amplitude).
- ▶ Vary MP2607.x , and compare with the recordings in the frequency diagram in TNCopt for the adjustment of the IPC and k_V factor.
- ▶ Select the value for MP2607.x by evaluating the advantages and disadvantages of the active damping.

MP2607.x Damping factor for active damping

Input: 0 to 30.000
0: No damping
1.5: Typical damping factor

MP2608.x Damping time constant for active damping

Input: 0 to 0.9999 [s]
0: No damping
0.005 to 0.02: Typical damping time constant

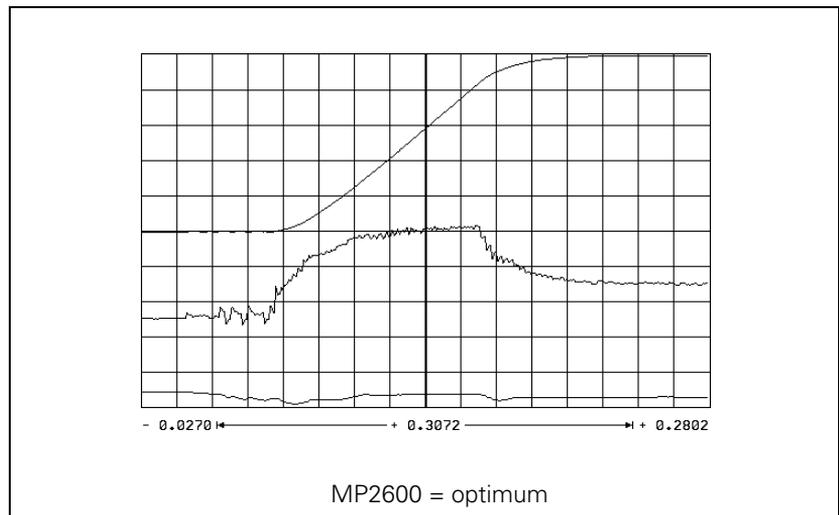
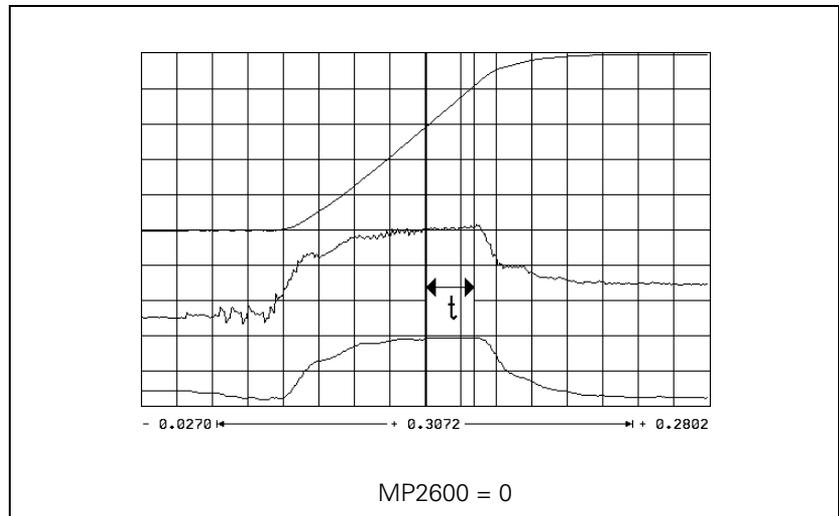


Acceleration feedforward control

Acceleration feedforward functions only in velocity feedforward control in parallel with the speed controller.

At every change in velocity, spikes of short duration appear in the following error. With acceleration feedforward control you can minimize these spikes:

- ▶ First adjust the friction compensation. Enter values in MP2610.x to MP2620.x.
- ▶ From the integral-action component of the nominal current value I (int rpm) calculate the input value for MP2600.x.
- ▶ Adjust the acceleration feedforward control with MP2600.x.



For calculation of the acceleration feedforward, the integral-action component of the nominal current value I (int rpm) is recorded with the internal oscilloscope. The actual speed value V (act rpm) and nominal current value (i nominal) are also recorded for better illustration.

$$MP2600.x = \frac{I (N INT) [A] \cdot t [s] \cdot 60 [s/min] \cdot MP1054.x [mm]}{\Delta V (N IST) [mm/min]}$$

I (int rpm)= integral-action component of the nominal current value

t = acceleration time in which I (int rom) remains constant

ΔV (ACT RPM)=actual-speed-value change while

MP1054.x= traverse distance per motor revolution

MP2600.x Acceleration feedforward control

Input: 0 to 100.0000 [A/(U/s²)]

Limiting the integral factor

In machines with a great deal of stiction, a high integral-action component can accumulate if there is a position error at standstill. This can result in a jump in position when the axis begins moving. In such cases you can limit the integral-action component of the speed controller:

- ▶ Enter a limit in MP2512.x.
Realistic input values: 0.1 to 2.0

MP2512.x Limit of integral factor of the speed controller

Input: 0.000 to 30.000 [s]

Integral phase compensation (IPC)

An I factor can be set in the speed controller of the iTNC (MP2510.x). This I factor is needed to attain a short setting time. However, the I factor has a negative influence on the position controller, i.e. the position controller tends to oscillate more easily, and it is often impossible to set the k_V factor (MP1510.x) high enough.

The IPC (Integral Phase Compensation) compensates the negative influence of the I factors on the speed controller, and makes it **possible** to increase the k_V factor (MP1510.x).

The IPC is beneficial on the following types of machines:

- Machine type 1: Machines with a dominant natural frequency between 15 Hz and 80 Hz, for which it is not possible to set a sufficiently high k_V factor.
- Machine type 2: Small-to-medium size machines that are driven directly.



Note

- The acceleration feedforward (MP2600.x) must already have been carefully adjusted for both types of machines.
- If after commissioning the IPC you wish to optimize the speed controller again, you must switch off the IPC beforehand, because the IPC influences the curve form.
- Use the same test program to commission the IPC as is used to measure the jerk and the k_V factor.

Machine type 1:

- ▶ The machine is commissioned as usual until the k_V factor is to be determined.
- ▶ Enter $MP2602.x = 1$ and $MP2604.x = 0$.
- ▶ Increase the k_V factor ($MP1510.x$) until you reach the oscillation limit.
- ▶ Starting value: $MP2604.x = \frac{2}{3} \cdot \frac{MP2600.x}{MP2500.x}$
- ▶ Change $MP2604.x$ until you have found the maximum k_V factor ($MP1510.x$).
If you cannot find a maximum k_V factor, use the default value in $MP2604.x$.
- ▶ Starting value: $MP2602.x = \frac{MP2600.x}{MP2500.x}$
- ▶ Increase $MP2602.x$ until you have found a maximum k_V factor ($MP1510.x$).
If the value found for $MP2602.x$ is significantly greater than the starting value ($>$ factor 2), you should adjust $MP2604.x$ again by enlarging and reducing it to find the optimum value.
- ▶ $MP1510.x = 0.65 \cdot \text{ascertained } k_V$

Machine type 2:

- ▶ The machine is commissioned as usual until the k_V factor is to be determined.
- ▶ Enter $MP2602.x = 1$ and $MP2604.x = 0$.
- ▶ Increase the k_V factor ($MP1510.x$) until you reach the oscillation limit.
- ▶ Starting value: $MP2604.x = \frac{2}{3} \cdot \frac{MP2600.x}{MP2500.x}$
- ▶ Change $MP2604.x$, normally by reducing it, until the following error is at its minimum.
- ▶ $MP1510.x = 0.65 \cdot \text{ascertained } k_V$

MP2602.x **IPC time constant T_1**
Input: 0.0001 to 1.0000 [s]
 0: IPC inactive

MP2604.x **IPC time constant T_2**
Input: 0.0001 to 1.0000 [s]
 0: IPC inactive

Optimizing the jerk feedforward control

In the same manner as the other feedforward controls in a control loop, the jerk feedforward control causes following errors to be reduced or compensated during a dynamic phase (in this case the phase of the change in acceleration).

When the feedforward control is initially configured, the necessary jerk feedforward control is calculated for the known control-loop factors, but the mechanical deformation of the transmission components (e.g. due to torsion of the ball screw or the slack sides and tight sides of belt drives) and the resulting following error are unknown. This can be minimized or compensated with the dimensionless multiplier MP2606, which affects the jerk feedforward control directly.

When optimizing the jerk feedforward control all other controller parameters must already have been adjusted.

With the CC 61xx you can use this function without active IPC.

Commissioning:

- ▶ Enter the following test program:

```
0 BEGIN PGM TEST MM
1 LBL 1
2 L X <maximum traverse> R0 FMAX
3 L X0 FMAX
4 CALL LBL1 REP 10/10
5 END PGM TEST MM
```

- ▶ Run the program at high speed.
- ▶ Use the integrated oscilloscope to record the following error.
- ▶ Adjust MP2606.x until a minimal following error occurs in the jerk phase. Here you can compensate the following error for the jerk phase. Typical input values: 0 to 0.5 (e.g. 0.01).



Note

Please note that values over 0.5 in MP2606.x do not make sense for the CC 61xx controller unit, and are therefore not permissible. Therefore, an error message will be issued for values greater than 0.5.

MP2606.x **Following error in the jerk phase**

Input: 0.000 to 0.800

Holding torque

The holding torque is the torque that is required to keep a vertical axis at a standstill.

The holding torque is given by the iTNC through the integral-action component of the nominal current value. In most cases the holding torque is constant. The required holding current can therefore be fed forward through MP2630.x. This relieves the speed controller.

- ▶ To prevent the effect of stiction, measure the current at low speed in both directions (e.g. 10 min⁻¹).
- ▶ Calculate the holding current from the mean of the measured current values and enter the result in MP2630.x.

Mean:

$$\text{MP2630.x} = \frac{I \text{ NOMINAL}_1 + I \text{ NOMINAL}_2}{2}$$



Note

- Please note that when reading the current via the internal oscilloscope, on the CC 6xxx you are seeing the effective value of the current.
- If the ready signal (RDY) is missing from the speed encoder inputs of vertical axes, the DSP error message **8B40 No drive< release>** appears. A vertical axis is defined with an entry in MP2630.x.

MP2630.x Holding current

Input: -100.000 to +100.000 [A]

6.8.5 Switching drives on and off

The procedure recommended by HEIDENHAIN for switching the drives on and off, including the correct triggering of the brakes, can be configured with the PLC basic program. HEIDENHAIN recommends using the PLC basic program.

Readiness of the drives

The drives can only be switched on once the drive system is ready. This includes:

- the readiness of the supply unit (RDY.PS signal at X69 pin 17a – interrogation via Module 9066 bit 5), the inverters (RDY signal at X51 to 64 pin 10b of the PWM outputs) and of the motors.
- the conclusion of the motor orientation (determination of the field angle).
- that the inputs for axis-specific drive enable are released. These entries are configured with MP4132.

Basic settings for "switching drives on"

Basic settings:

- ▶ Ensure that the clamping or the brakes activate when an emergency stop is effected.
- ▶ Ensure via Module 9159 (drive controllers are switched off) that the clamping or brakes activate before the drive controllers (current and speed controllers) are switched off.
- ▶ Interrogate the readiness of the HEIDENHAIN supply module with Module 9066 bit 5.
- ▶ Before switching off the drives, save the positions of axes with common encoders.
- ▶ Ensure that all control components are wired correctly. HEIDENHAIN recommends always wiring the system according to the basic circuit diagram of the iTNC 530 HSCI. Registered customers can download the current basic circuit diagram from the HESIS-Web Including Filebase on the Internet at <http://portal.heidenhain.de>.
- ▶ In MP2040.x, define the axis groups for the axis-specific drive enable and do the proper wiring for controlling the drive enable. Use Module 9157 to interrogate the status of the drive enable.

If drive enabling is missing for the PLC inputs specified in MP4132, the drive motor is **always** switched off and a report is sent to the position controller.

- ▶ Set W524 Bit 0 = 1, so that the control loop is opened and no error message appears when the drive is switched off.

If drive enabling of axis groups via MP4132 is used, the PLC program should perform a plausibility test to see if all axes used are defined in MP2040.x



Danger

If a bit is defined incorrectly in MP2040.x, the drive motor might not be switched off through the PLC inputs specified in MP4132!

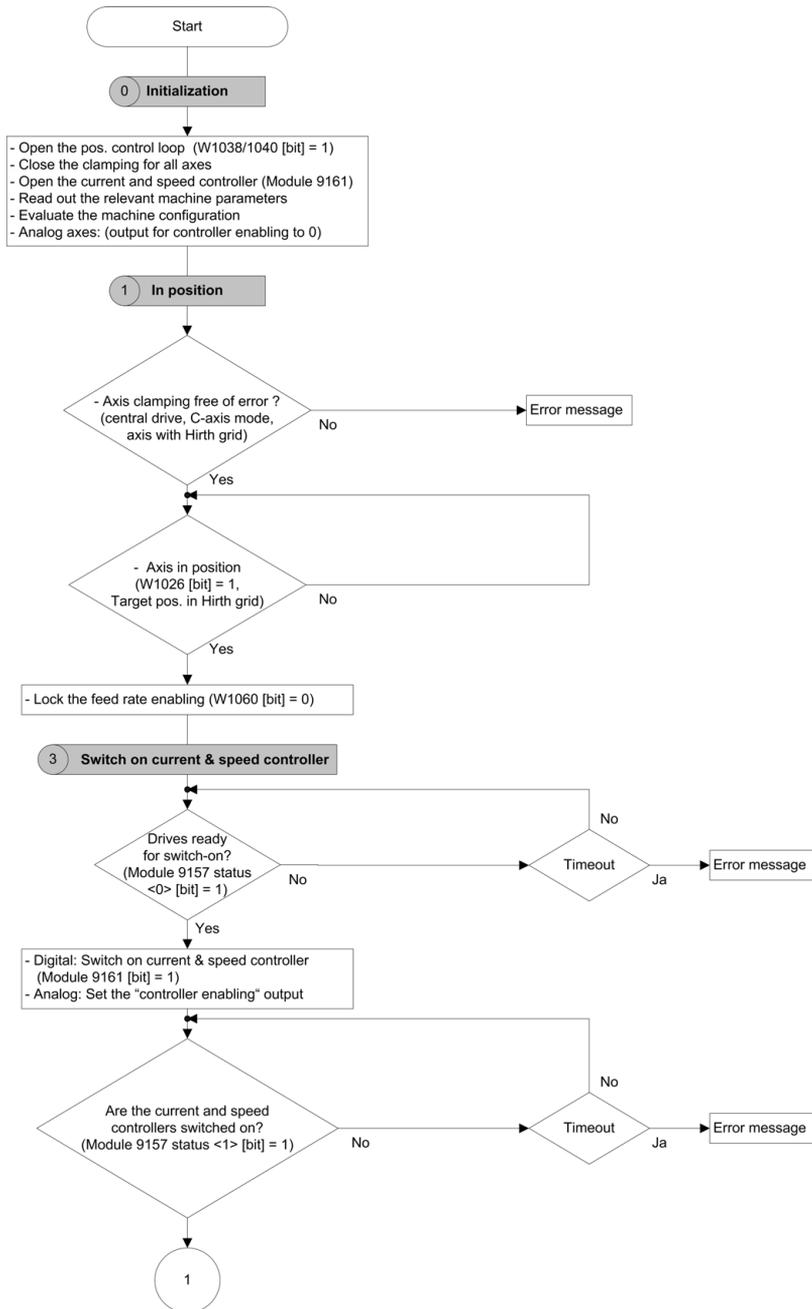


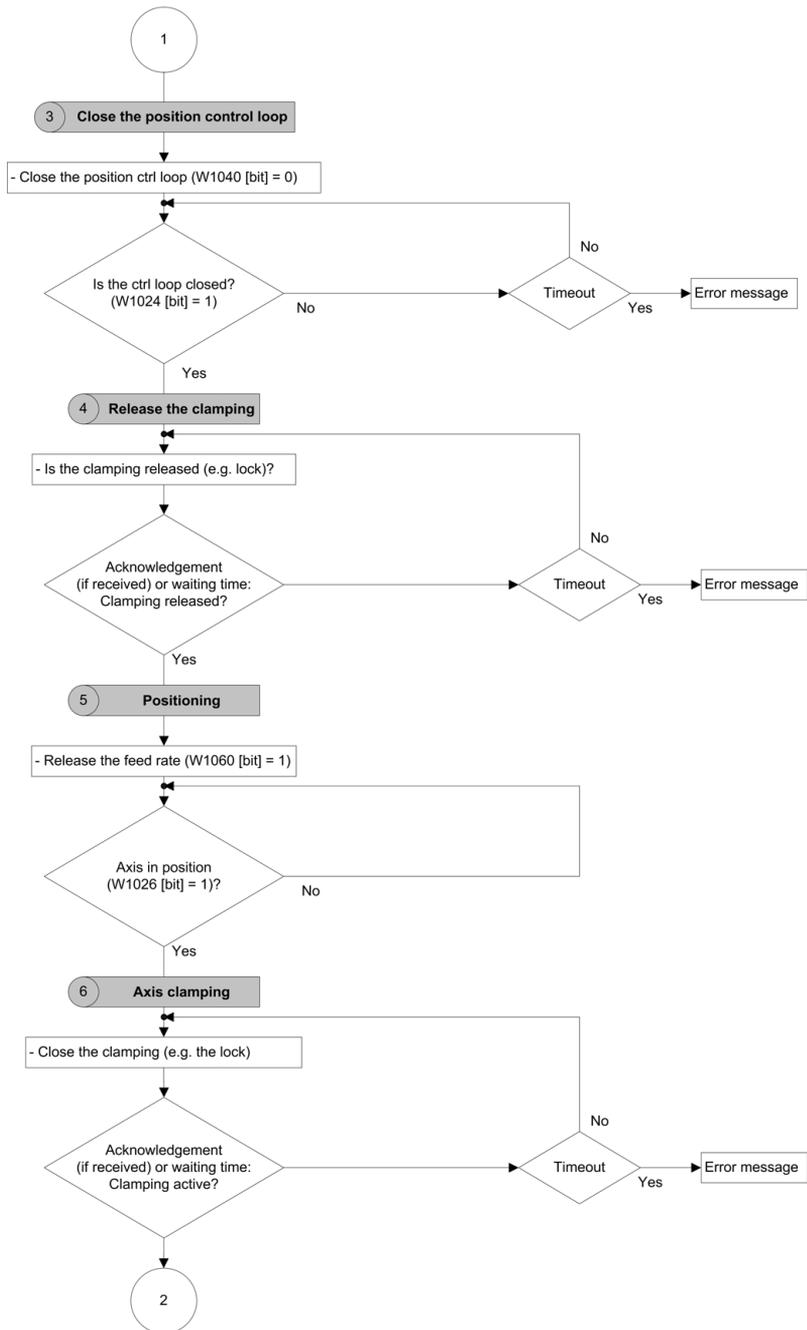


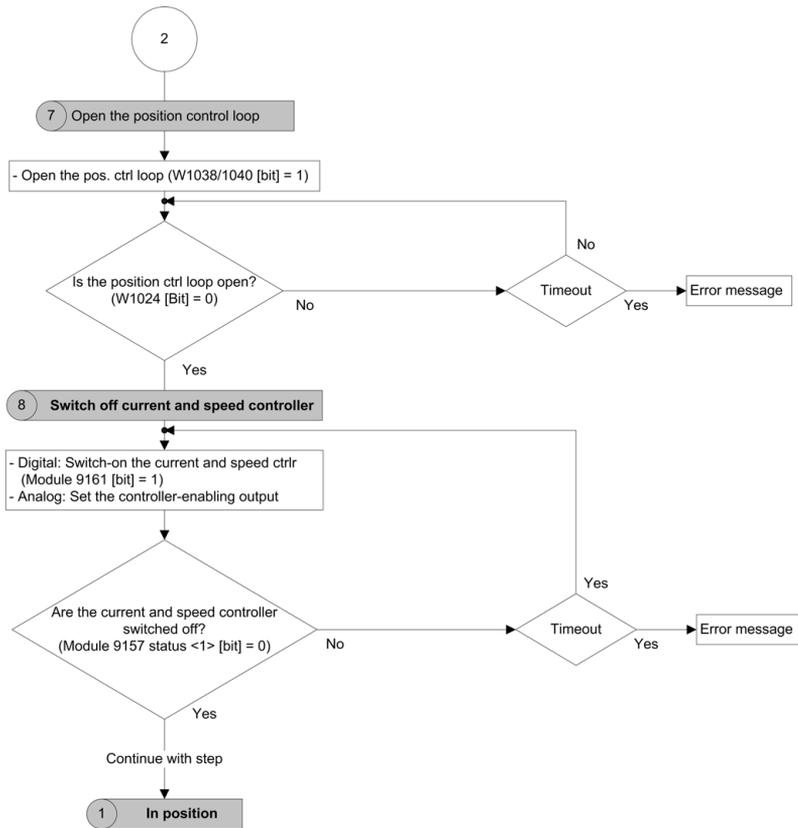
Note

- If drive enabling (MP4132, -ES.B) is missing, the error message **8B40 No drive release <axis>** appears.
- If you do **not** want to use drive enabling for axis groups, but rather just global drive enabling through -ES.B, set all bits in MP2040.x to %00000000000000000000000000000000 and in W524 to zero.
- The iTNC monitors the time between the switch-on of the drive controller (via Module 9161) and the READY signal of the power module (via the PWM cables). If the READY signal is missing after the waiting time has passed, the error message **8B40 No drive release <axis>** appears. In MP2170 you preset the permissible waiting time.
- If the readiness signal is reset at the PWM outputs, the drive controllers (current and speed controllers) are switched off. Normally, the error message **MOVEMENT MONITORING IN <AXIS> B** is output through the position control loop. Subsequently, the PWM signal release is switched off by the reset signal.
- The current controller is switched on within 50 ms of switching on the controller (Module 9161). This also delays the acknowledgment over Module 9162 at below 50 ms. The delay time depends on MP2221.x bit 7.
- If a vertical or rotary axis is switched on, the current that was active at the time of switch-off is used. This is to prevent the axis from sagging when the brake is released, even if the feedforward control for the axis is not optimally set.

Procedure for "switching drives on and off"







Machine parameters, markers, modules and PLC words

MP2040 Axis group (for axis-specific drive enable)

Format: %xx
(Spindle = bit 31)

Input: 0: Axis/spindle not assigned (disabling only through I32).
1: Axis/spindle assigned

MP2040.0-2 Axis groups 1 to 3

MP2040.3-7 Reserved, enter %00000000000000000000000000000000

MP2170 Waiting time between the switch-on of the drive and the drive's standby signal

Input: 0.001 to 4.999 [s]
0: 2 [s]

MP4132 Axis-specific drive enabling

Input: 0 to 20000
[Number of the PLC input or symbolic PLC operand]
-1: Function inactive

Bit 7 of MP2221.x enables you to deactivate the reduction of a drive's switch-on time when it is switched on with PLC module 9161. If spindles with wye/delta switchover but without evaluation of the status of the wye/delta contactor are used, it may occur that the drive is switched on again before the contactor has switched over to the appropriate mode. If this occurs, the current controller sets the voltage to the maximum possible value. If the contactor is then closed, this may lead to overcurrent, and as a result, to a switch-off (e.g. due to error message 8B60) of the drive.

MP2221.x Bit 7 – Switch-on time of the drive

Input: 0: Reduction of the switch-on time is active
1: Reduction of the switch-on time is not active

	Set	Reset
W524		
Open the control loop if drive enabling via MP4132 is missing	PLC	PLC
Bit 0 = 0: Position control loop stays closed		
Bit 0 = 1: Position control loop is opened		
Bit 1 to bit 15: Non-functional		



Module 9157 Drive controller status

Status information about the drive controller can be ascertained with this module.

Call:

PS B/W/D/K <Selection of status information>
0: Readiness of drives
Result 0: Drive is not ready and cannot be switched on
Result 1: Drive is ready and can be switched on
1: Drive controller status (as in Module 9162)
2: Through PLC inputs in axis enabled by MP4132
3: Signal to PLC inputs in MP4132
axis-specific: dependent on MP4020 bit 14
3: Spindle in mode 0 or mode 1
axis-specific: dependent on MP4020 bit 14

CM 9157

PL B/W/D <Axis status information bit-encoded>
Axis-specific: Dependence on MP4020 bit 14

Error code:

Marker	Value	Meaning
M4203	0	Status information was ascertained
	1	Error code in W1022
W1022	2	Invalid status information was programmed
	24	Module was called in a spawn job or submit job

Module 9159 Advance status message: Drives will be switched off

Call:

CM 9159

PL W/D <Drives, in bit code, that are switched off in the time defined in MP2308>
Axis-specific: Dependence on MP4020 bit 14



Module 9161 Enable the drive controller

With this module you can switch the drive controllers (speed and current controllers) on and off for specific axes. A nominal speed value is also output when the drive controller is not enabled.

The switch-on time for switching on a drive via PLC module 9161 is less than 50 ms. In individual cases this results in problems. You can deactivate the reduction of the switch-on time with bit 7 of MP2221.x.

Call:

PS W/D/K <Enabled axes>
Axis-specific: Dependence on MP4020 bit 14
0: No drive controller enabling
1: Drive controller enabling

CM 9161

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Control has no current controller, or the call is in a spawn or submit job

Module 9162 Status request of the drive controller

Call:

CM 9162
PL B/W/D <Drive is ready>
Axis-specific: Dependence on MP4020 bit 14
0: Not ready
1: Ready

Module 9169 Axes for which I32 does not switch off the drives

Call:

PS B/W/D/K <Axes bit-encoded>
Axis-specific: Dependence on MP4020 bit 14
CM 9169

		Set	Reset
M4563	Global feed-rate enable for all axes	PLC	PLC

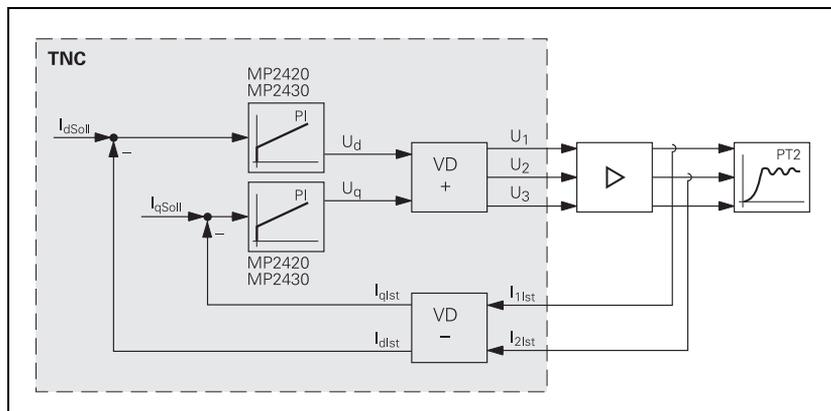
6.8.6 Current controller

Up to 20 digital current controllers for the axes and spindle(s) are possible in the iTNC 530:

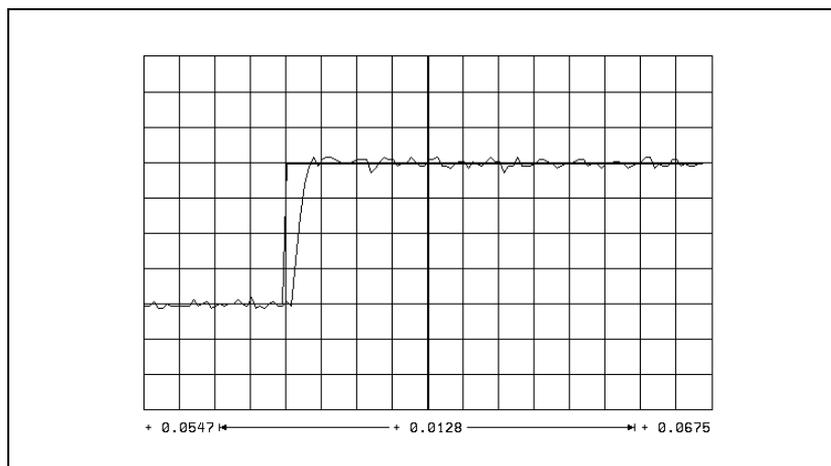
The nominal values for magnetizing current I_{dnom} and torque current I_{qnom} are divided into the PWM signals U_1 , U_2 and U_3 through a PI controller and vector rotator $VD+$, and are transferred to the power module through X51 to X60.

The actual current values I_{1act} and I_{2act} are determined by the power module and are transferred to vector rotator $VD-$ through X51 to X60. The vector rotator determines the actual values of magnetizing current I_{dAct} and torque current I_{qNom} .

Circuit diagram:



You adjust the current controller to attain the optimum result, with the position and speed controller switched off.



The step response is adjusted such that there is no overshoot and the rise time is as small as possible:

- ▶ In MP2420.x, define the P factor of the current controller.
Automatic calculation of the P factor for synchronous and asynchronous motors is also possible. However, automatic calculation is not to be used for linear synchronous and torque motors.
The calculated value is entered in the MP file. An * is appended to the calculated value to show that the parameter was determined automatically.
- ▶ In MP2430.x, define the I factor of the current controller.
Automatic calculation of the I factor for synchronous and asynchronous motors is also possible. However, automatic calculation is not to be used for linear synchronous and torque motors.
The calculated value is entered in the MP file. An * is appended to the calculated value to show that the parameter was determined automatically.

MP2420.x P factor of the current controller

Input: 0 to 9999.99 [V/A]
* = automatic calculation of the P factor

MP2430.x I factor of the current controller

Input: 0 to 9 999 999 [Vs/A]
* = automatic calculation of the I factor

In addition, the function of the current controller is monitored automatically. If the fault-free functioning of the current controller is not ensured, the brake of the respective drive is activated immediately and an emergency stop error message is triggered. This happens if the speed controller specifies the maximum current, but the actual current remains zero.

Expanded current controller bandwidth

For axes with rigidly connected mechanics it may make sense to adjust the current control loop using the maximum possible bandwidth to attain high gains in the superimposed velocity control loop. For the CC 61xx controller unit there are two procedures for increasing the bandwidth of the current control loop:

- Using machine parameter MP2450 to increase the controller factors in the current control loop,
- and proportionally differentiating feedforward (D feedforward) to optimize the command action of the current control loop (MP2440).

Increased controller factors in the current control loop:

The dead time in the current control loop is the decisive factor for the attainable controller factors, and therefore for the attainable bandwidth. With MP2450, you can activate compensation of this dead time. After activation, you have to repeat the current controller adjustment. Usually you can then set considerably higher current controller factors. The higher the controller factors, the larger the bandwidth of the current controller. The bandwidth of the current control loop can be evaluated with the Bode diagram function of the TNCopt commissioning and diagnostic software (see TNCopt User's Manual).

However, the noise in the current—and as a result, the loudness of the drive—increases when the current controller bandwidth is increased.

D feedforward for optimization of the current controller command action:

With **MP2440**, you activate feedforward with proportionally differentiating behavior for the torque-generating current. This way you achieve an optimized command action of the current controller.

The parameter **MP2440** indicates the cutoff frequency in Hertz [Hz] up to which feedforward is active. When this cutoff frequency is reached, feedforward is deactivated by a low-pass element. It is therefore also referred to as PDT_1 feedforward in the following.

The following table shows the maximum possible cutoff frequency depending on the PWM frequency of the drive. The higher the cutoff frequency, the larger the bandwidth of the current controller regarding its command action. At the same time, however, the noise in the current—and as a result, the loudness of the drive—increases. If you enter 0 Hz, feedforward is deactivated.

Maximum values for parameter **MP2440** as a function of the PWM frequency:

PWM frequency	Maximum value for MP2440
3333 Hz	800 Hz
4000 Hz	960 Hz
5000 Hz	1200 Hz
6666 Hz	1600 Hz
8000 Hz	1920 Hz
10000 Hz	2400 Hz

Adjustment of the current controller if increased current controller factors are active over MP2450:



Note

For standard drives, it is usually not necessary to increase the current controller bandwidth. In general, the increase is only useful if the drives are connected directly without an interconnected gear.



If PDT_1 feedforward or the increase of the current controller factors is activated, you must keep in mind that the noise in the current increases when the bandwidth is increased. You have to find a compromise between the

- increase of the current controller bandwidth and
- the development of noise in the drive.

To adjust the increased current controller factors, proceed as follows:

- ▶ To activate the function, enter $MP2450 = 1$ for the corresponding parameter of the axis.
- ▶ Readjust the current controller with TNCopt (Bode diagram function; see TNCopt User's Manual).



Note

If you activate the PDT_1 feedforward by entering a value in parameter **MP2440**, it is not necessary to readjust the current controller. The PDT_1 feedforward does not have a noticeable effect on the current control loop but rather only on the superimposed speed control loop. For axes with rigidly connected mechanics, you attain higher controller factors in the speed control loop if the PDT_1 feedforward is activated.

MP2440.x Cutoff frequency of the feedforward current controller

Input: 0.1 to 5000.0 [Hz]
0: Feedforward control not active

MP2450.x Dead-time compensation

Input: 0: Not active
1: Active

6.8.7 Braking the drives for an emergency stop and a power failure

In an emergency stop and power failure the spindle must be braked as quickly as possible. If the braking energy cannot be drawn off quickly enough, the DC-link voltage increases sharply. Under circumstances, the inverter could switch off and the spindle coast to a stop. A powerful braking of the spindle also leads to a high strain on the mechanics of the machine. Preferably the spindle should be braked in an emergency stop by limiting the braking power.



Note

Limiting the braking power is also effective when braking the spindle with M05, if the brake ramp in M05 (MP3411 and MP3412) is steeper than the brake ramp when limiting the braking power.

Normally, in case of an emergency stop, the axes are braked at the limit of current. This can create problems:

- With gantry axes a mechanical offset can occur between the master and slave axes.
- A gear between spindle and motor can be overloaded.

In such cases, the axes should preferably be braked in an emergency stop by adding an additional braking ramp.



Note

Both of the above braking strategies are possible for axes and spindles. If both braking strategies are activated for an axis or spindle, they do not exclude each other; this means that in case of an emergency stop, whichever strategy responds first takes effect.

Problems with inverters **without** regenerative power supplies (with braking resistors) during braking mostly arise if the inverter is switched off too early. The strain on the mechanics during braking is reduced, but can also be influenced with braking strategies.

Inverters **with** regenerative power supplies usually do not develop problems if they are switched off. The main concern here is for the mechanics of the machine. If the maximum regenerative power of the inverter is exceeded when braking the drives (during an emergency stop, for example), the axes and spindle(s) coast to a stop. In this case it is sensible to define separate maximum braking powers for each drive in MP2390.x. This will ensure that each drive is braked as quickly as possible.

Braking the axes by entering an additional braking ramp

In this strategy, the braking ramp to be used in an emergency stop is entered. Set the axis braking ramp for an emergency stop:

- ▶ Enter as a minimum value in $MP2590.x = \frac{MP1060.x \cdot 60}{MP1054.x}$.
- ▶ Use the emergency stop to brake the axis from rapid traverse.
- ▶ Increase the value entered in MP2590.x until the braking time is as short as possible and the mechanics of the axis are not overloaded.

If the value entered is too small, i.e. if braking is too slow, the axis brakes at the acceleration defined in MP1060.x.

Once the value for MP2590.x has been determined and entered, the braking process at this setting must be checked. You must ensure that simultaneous braking of all axes does not lead to a switch-off due to overload of the inverters. The nearer the determined braking ramp is to the emergency stop ramp, the greater the resulting load on the inverters is.



Note

The value entered in MP2590.x refers to the motor speed, meaning the ball screw pitch is not considered.

MP2590.x Braking ramp in an emergency stop

Input: 0.001 to 999.999 [min⁻¹/ms]
0: Function inactive

Braking the spindle/spindles by entering the braking power

In this strategy the maximum braking power for braking the spindle/spindles in an emergency stop or power failure is entered.

If power limiting (MP2392.x) is used in normal operation, then the maximum braking performance is limited to the lower of the two values from the power limiting and the braking power.

Example:

Function	Case 1	Case 2
Power limiting (MP2392.x)	10 kW	5 kW
Maximum braking power (MP2390.x, MP2394.x)	5 kW	10 kW
Limiting the braking performance to	5 kW	5 kW



Braking the spindle/spindles in an emergency stop

- ▶ For **inverters with regenerative power supply**, enter MP2390.x = 0 so as not to limit the braking power.
- ▶ Calculate for **inverters with braking resistors** the input value for MP2390.x from the following formula:

$$\text{MP2390.x} = \frac{U_Z^2}{R \cdot 1000}$$

R = Braking resistance [Ω]
U_Z = DC-link voltage [V]

The maximum braking power from MP2390.x can be greater than the power limit from MP2392.x.

Braking of the spindle during an emergency stop can be delayed by the time in MP3550. This avoids tool breakage during power milling, because low spindle speeds might result in the spindle coming to a standstill before the axes do. However, the spindle braking is delayed by no more than half a spindle revolution.

Braking the spindle/spindles during a power fail

During a power fail, the "SH1B" signal at X51 to X60 is maintained for 10 more seconds, in order to brake the spindle/spindles. At the same time, the control tries to reset the PLC outputs.

- ▶ If you are using an additional braking resistor (e.g. UP 110) in connection with an **inverter with regenerative power supply**, calculate the value to be entered in MP2394.x from the above formula.
- ▶ Calculate for **inverters with braking resistors** the input value for MP2394.x with the above formula.



Note

If after entry of a value in MP2390.x or MP2394.x the mechanics are overloaded by the braking process, lower the value in MP2390.x or MP2394.x until you have found an optimum between braking time and mechanical loading.

MP2390.x	Maximum braking power
Input:	0: No limiting of the braking power in an emergency stop 0.001 to 3000.000 [kW]
MP2394.x	Maximum brake power for power failure
Input:	0: No limiting of the braking power in a power failure 0.001 to 3000.000 [kW]
MP3550	Delay of emergency-stop reaction of spindles
Input:	0.001 to 0.100 [s] 0 = Delay not active

6.8.8 Power and torque limiting

You can limit the power of your spindle motor to achieve wider gear ranges. Wide-range motors are characterized by a larger speed range with higher torque at low speed.

One solution for bringing about this behavior is to use an oversized motor, and to limit the maximum power. However, power limiting does not reduce the high torque to the speed at which power limiting becomes effective. This high torque (until power limiting takes effect) can be reduced with torque limiting, in order to keep the mechanics of the machine from becoming overloaded.

The maximum spindle power will always be limited in order to avoid overloading the power supply module. If MP2392 is set to zero and a power supply module is defined in MP2198, the maximum spindle power is limited to the value S6-40 from the power supply module table (Supply.spy). For applications requiring a higher spindle power (i.e. the spindle should be able to take up the maximum power P-Max from Supply.spy), the desired maximum power must be entered in MP2393.

With torque limiting you can also limit the torque of the axis motors, in order to keep the mechanics of the machine from becoming overloaded. Power limiting is not useful for axis motors.

For **axes and spindles**, the torque is limited to the value taken from either the table of power modules or the motor table, whichever is lower.

On supply units where the $\overline{\text{ERR.IZ.GR}}$ signal is available, the power of the spindle is limited via MP2392.x in case of error (not for axes).

HEIDENHAIN recommends activating this monitoring function via MP2220.x bit 2 (not with UE 2xx).

The torque can be calculated for any speed:

$$M = \frac{P \cdot 60}{n \cdot 2 \cdot \pi}$$

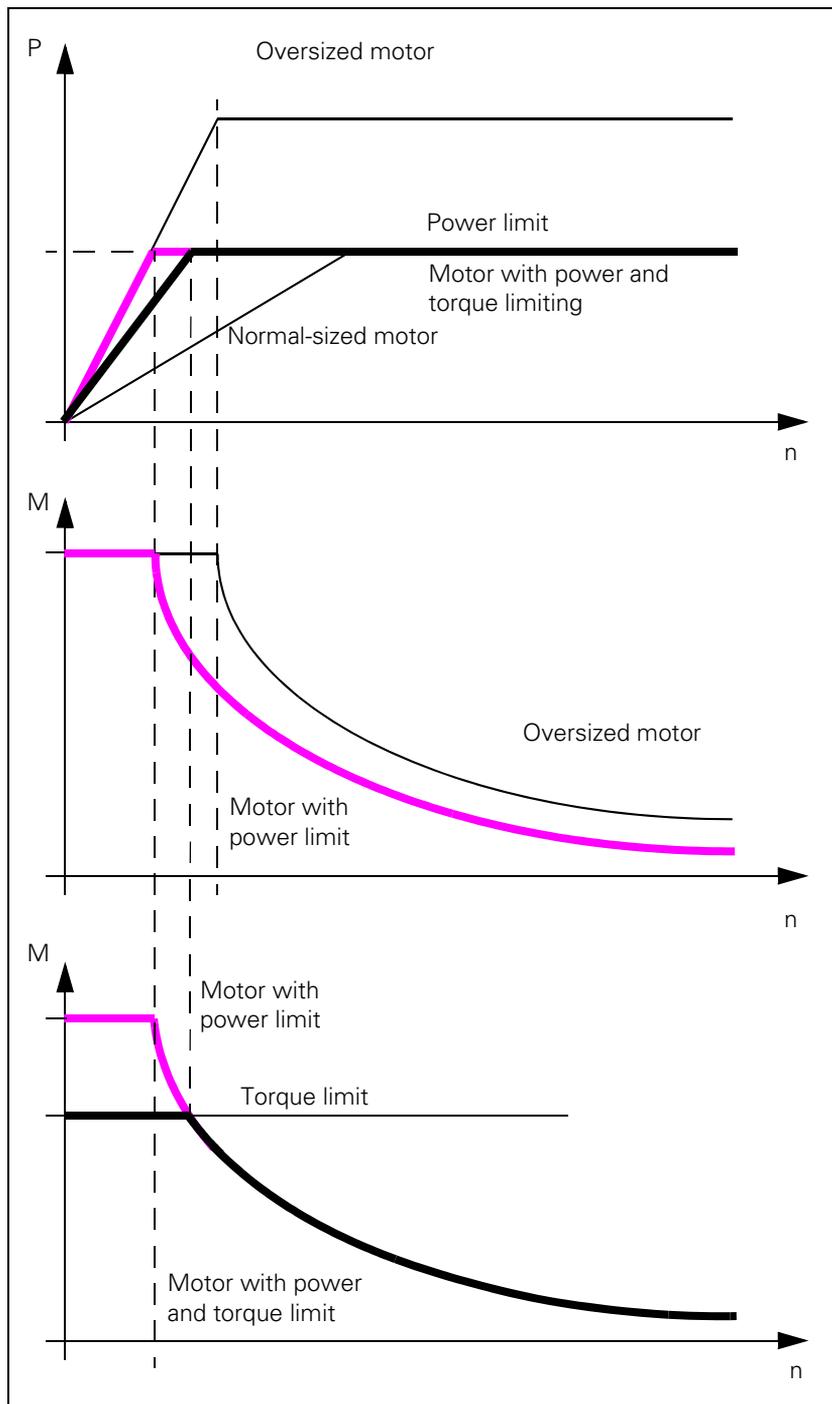
M: Torque [Nm]
P: Power [W]
n: Speed [min^{-1}]



Note

The power and torque limiting can have an effect on the braking of the spindle in an emergency stop.

- ▶ Enter the maximum power for the spindle in MP2392.x.
- ▶ Enter the maximum torque for the spindle or axis in MP2396.x.
- ▶ Activate the power limiting of the spindle at $\overline{\text{ERR.IZ.GR}}$ via MP2220.x bit 2 = 0 (not for UE 2xx)



MP2220.x Monitoring functions

Input: Bit 2 – Power limit of spindle at $\overline{\text{ERR.IZ.GR}}$ (only for HEIDENHAIN inverters, except UE 2xx)
0: Power limit active
1: Power limit inactive

MP2392.x Power limit

Input: 0: No power limiting
0.001 to 3000.000 [kW]

MP2396.x Maximum torque

Input: 0: No torque limiting
0.1 to 30 000.0 [Nm]

MP2223.x Special controller functions

Input: Bit 1 – If a torque is specified via PLC Modules 9128/9158 and MP2223 bit 1 is set, the NC automatically considers the holding current in MP2630 when limiting the current. This makes it possible to use the PLC modules to set a limit value that is below the holding torque.
0 = Inactive
1 = Consider MP2630 with PLC Modules 9128/9158

Power limiting of the spindle after PLC request

An additional maximum drive power rating can be defined with the machine parameter MP2393.x. This function must be activated via PLC Module 9314. This makes it possible to reduce power consumption for starting up the spindle. This can prevent the maximum power of the power supply module from being exceeded if an axis is positioned at the same time as the spindle is started.

The PLC knows (via M3/M4/M5) when the spindle is being started up and when the defined spindle speed has been reached. During the PLC positioning movements (tool change) you can switch to the power in MP2393. During machining (low power consumption of the axes, low traversing speeds) you can switch to the maximum spindle power (in MP2393, or no power limit).

MP2393.x Power limiting after PLC request

Input: 0: No power limiting
0.001 to 3000.000 [kW]

Module 9158 Maximum torque

With Module 9158 you can limit the torque of an axis or spindle. The maximum torque resulting from the data in the control's motor table cannot be exceeded. In this case the torque is limited to the value from the motor specifications. After the drive is switched off, the original torque becomes effective again.

If torque limiting is active, the standstill monitoring is inactive; only the motion monitoring remains active.

If a torque is specified via PLC Modules 9128/9158 and MP2223 bit 1 is set, the NC automatically considers the holding current in MP2630 when limiting the current. This makes it possible to use the PLC modules to set a limit value that is below the holding torque.

The torque-producing current required for the desired torque must be transferred to the module:

Synchronous motor	Asynchronous motor
$k_M = \frac{P_N}{I_N \cdot 2 \cdot \pi \cdot \frac{n_N}{60}}$ $I_q = \frac{M}{k_M}$	<ul style="list-style-type: none"> ■ Armature control range ($n < n_{FS}$) $I_q = \frac{M \cdot n_N \cdot 2 \cdot \pi \cdot \sqrt{I_N^2 - I_0^2}}{P_N \cdot 60}$ ■ Field weakening range ($n > n_{FS}$) $I_q = \frac{M \cdot n_N \cdot n \cdot 2 \cdot \pi \cdot \sqrt{I_N^2 - I_0^2}}{P_N \cdot n_{FS} \cdot 60}$
<p>I_q: Torque-producing current</p> <p>M: Desired torque</p> <p>k_M: Torque constant</p> <p>n_N: Rated speed (from motor table)</p> <p>I_N: Rated current (from motor table)</p> <p>P_N: Power rating (from motor table)</p>	<p>I_q: Torque-producing current</p> <p>M: Desired torque</p> <p>n_N: Rated speed (from motor table)</p> <p>n: Current speed</p> <p>I_N: Rated current (from motor table)</p> <p>I_0: No-load current (from motor table)</p> <p>P_N: Rated power output (from motor table)</p> <p>n_{FS}: Threshold speed for field weakening (from motor table)</p>



Danger

If Module 9158 is used, then certain monitoring functions regarding the drives must be switched off. Please note the following error messages and their possible consequences (see DSP error messages)

- **C380 Motor <axis> not controllable**
- **C3B0 Motor <axis> does not rotate**



Call:
 PS B/W/D/K/S< Axis or spindle>
 Axis-specific: Dependence on MP4020 bit 14
 PS B/W/D/K/S<Torque-producing current in mA>
 -1 = Torque given in motor specifications
 CM 9158

Error code:

Marker	Value	Meaning
M4203	0	Torque preset active
	1	Error code in W1022
W1022	1	0 Nm torque transferred
	2	Invalid axis number, open-loop axis, or uncontrolled axis
	24	Module was called in a spawn job or submit job



Module 9128 Torque limiting by the PLC

Module 9128 can be used to program a maximum torque for the programmed axis. The torque of the drive is limited to the programmed value. The value –1 cancels the torque limitation and the value from the motor data becomes effective again. The torque can be limited in [mA] or in [0.1 %] of the rated current.

Condition:

- The module is only executable in the cyclic PLC program.
- The programmed value for the maximum torque may not be higher than the value in the motor data. If the programmed value is higher than the value in the motor data, the value in the motor data is used as the limit.
- A torque value of 0 cannot be programmed.
- Programming a torque value of –1 cancels limitation. The original value from the motor data becomes effective again.
- The unit of the resulting torque is [mA].
- If a drive is switched off, the torque from the motor data becomes effective when it is switched on again.
- The notes for Module 9158 must be considered!
- If a torque is specified via PLC Modules 9128/9158 and MP2223 bit 1 is set, the NC automatically considers the holding current in MP2630 when limiting the current. This makes it possible to use the PLC modules to set a limit value that is below the holding torque.

Call:

PS B/W/D/K <Mode>
 0: Current in [0.1 %] of the rated current
 1: Current in [mA] (like Module 9158)

PS B/W/D/K <Axis number>
 Axis-specific: Dependence on MP4020 bit 14

PS B/W/D/K <Torque>
 –1: Cancel the torque limiting

CM 9128

Error code:

Marker	Value	Meaning
M4203	0	Torque limiting programmed
	1	Error code in W1022
W1022	1	Invalid value for torque
	2	Invalid value for axis number or mode, axis is an open-loop axis or is temporarily not a closed-loop axis
	24	Module was called in a spawn or submit job



Module 9129 Status of torque limiting by the PLC

Module 9129 is used to determine the current status of torque limiting for the programmed axis. The momentary maximum torque can be determined in [mA] or in [0.1 %] of the rated current.

Condition:

- The module is only executable in the cyclic PLC program.
- The greatest possible return value is the value resulting from the motor data.
- If torque limiting is not active, the maximum current can be determined from the motor data.

Call:

PS B/W/D/K <Mode>
0: 0 = Limiting inactive/ 1 = Limiting active
1: Current in [mA]
2: Current in [0.1 %] of the rated current

PS B/W/D/K <Axis number>
Axis-specific: Dependence on MP4020 bit 14

CM 9129

PL B/W/D <Status>
Mode 0: 1 = Limiting active / 0 = Limiting inactive
Mode 1: Current in [mA]
Mode 2: Current in [0.1 %] of the rated current

Error code:

Marker	Value	Meaning
M4203	0	Status ascertained
	1	Error code in W1022
W1022	2	Invalid value for axis number or mode, axis is an open-loop axis or is temporarily not a closed-loop axis
	24	Module was called in a spawn or submit job

6.8.9 Weakened field operation

General information

Asynchronous motors are usually operated with a weakened field. This operating mode can become necessary for synchronous motors if the present inverter voltage does not suffice for the required rotational speed. Typical applications for this operating mode are high-speed synchronous spindle motors and "high-speed" torque motors.

For synchronous drives the operation with a weakened field is automatically activated if MP2160.x = 1 has been entered, and the desired speed makes it necessary. Under certain conditions a voltage protection module must be used.

Because of their design, the HEIDENHAIN EcoDyn motors are treated as a special case. They must always be operated with a weakened field, but no voltage protection module is necessary.

For HEIDENHAIN EcoDyn motors, MP2160.x = 2 is to be entered.

The speed-torque characteristics from the drive manufacturer indicate whether weakened-field operation is necessary. Among other information, they show the possible speeds in combination with the necessary inverter voltage.



Note

Please note that there are other possible settings for weakened-field operation with the CC 61xx, see "Peculiarities in weakened-field operation" on page 1103.



Using the voltage protection module

If the power supply fails during weakened-field operation, and the synchronous drive is running at a high speed at the same time, then the DC-link voltage can rise rapidly (generator effect of the drive). If this voltage rises to over 850 V, then inverters and possibly the motor itself can become damaged. Reliable protection against this is offered by a voltage protection module, which short-circuits the motor phases when the trigger threshold of 850 V is exceeded. (for example SM 110 or SM 130; see the "Inverter Systems and Motors" Technical Manual).

Whether a voltage protection module is necessary for a drive depends on the desired speed, the nominal speed **N-N** and the no-load voltage **U0** of the drive. See the motor table for these values.

The following formula calculates the motor speed at which a voltage greater than 850 V would be induced by the generator effect. If the desired motor speed is greater than the one calculated, then the voltage protection module (SM 110 or SM 130) **must** be used!

$$N_{max} = \frac{850 \text{ V} \times N_{noml}}{U_0 \times \sqrt{2}}$$

Please refer to the "Inverter Systems and Motors" Technical Manual to see which voltage protection module is needed for which motor types (depending on the maximum phase current of the SM 110 or SM 130).



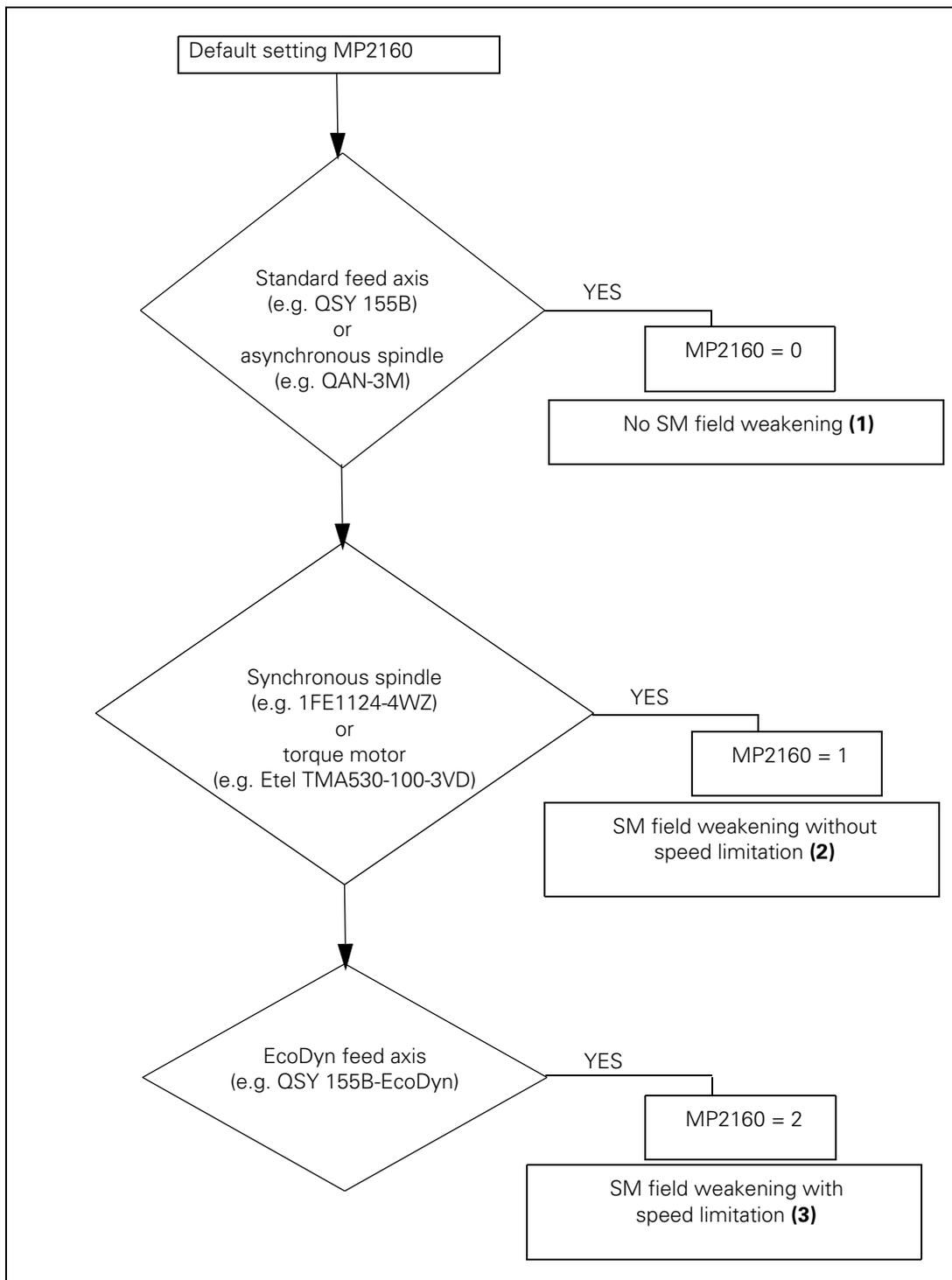
Attention

A braking resistor, such as PW xxx or UP 110, does not offer sufficient protection.

Setting the weakened-field operation

When setting MP2160 and MP2190 for synchronous motors (SM), proceed as shown in the diagrams for commissioning.

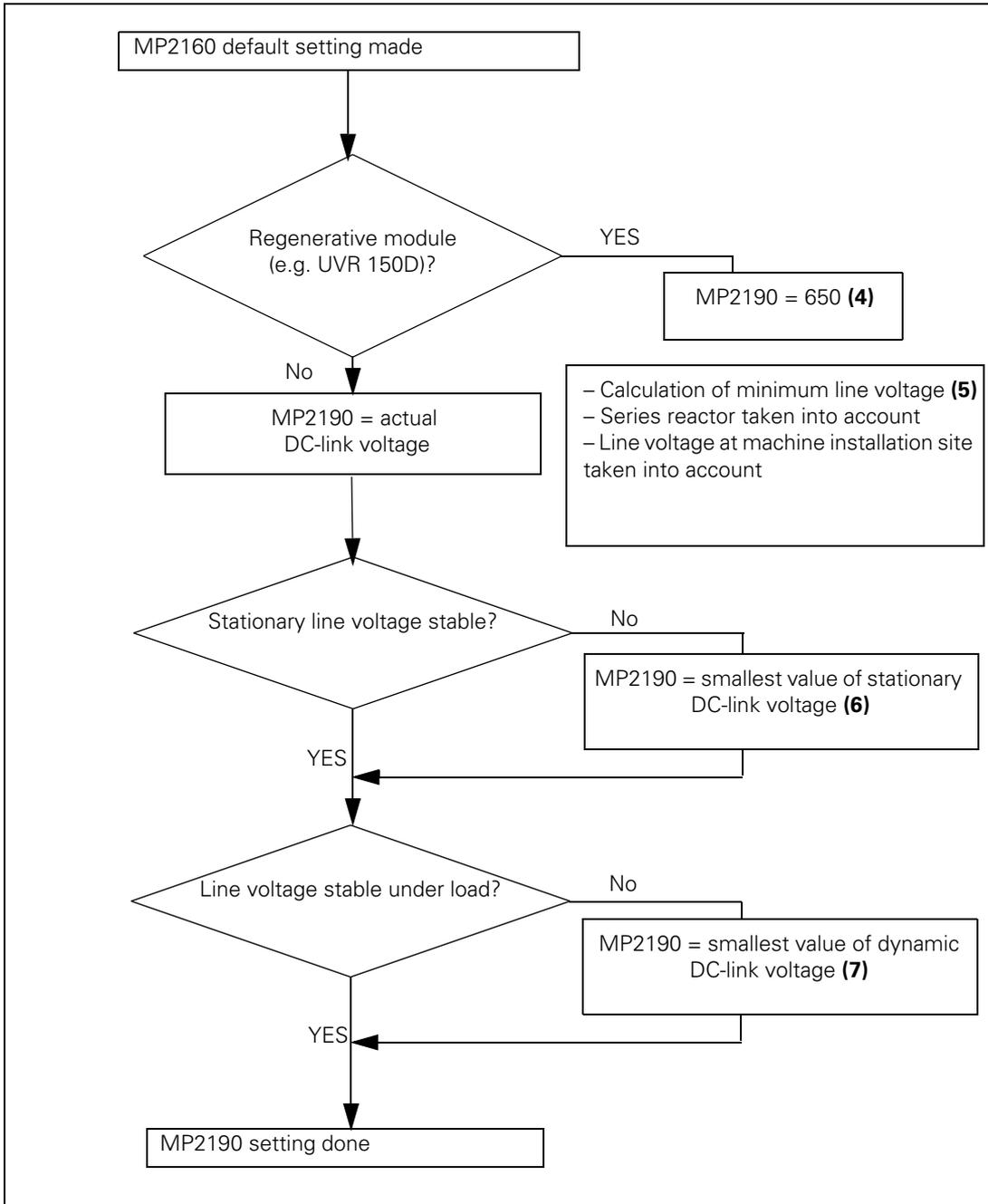
Default setting MP2160:



- ▶ Enter **MP2160.x = 0** if you want to operate synchronous motors **without** a weakened field. (In the commissioning flow chart = **(1)**).
Field weakening for synchronous motors is deactivated, i.e. the field weakening current is always 0, regardless of the speed and load. Use this entry also for all asynchronous motors. MP2160 is without effect on them.
- ▶ Enter **MP2160.x = 1** if you want to operate the synchronous motor with a weakened field (check whether a voltage protection module must be used. (In the commissioning flow chart = **(2)**).
This entry is intended for synchronous spindles and torque motors in turning mode. Here, field weakening is active in a wide speed range, i.e. the maximum speed n_{max} is much higher than the rated speed n_{rated} ($n_{max} \gg n_{rated}$). The internal voltage U_{emk} at maximum speed in the motor leads to a considerable rise of the DC-link voltage if the power supply fails, so that the connected inverters and also the motor can get damaged. It is therefore necessary to install a voltage protection module (e.g. SM 110, see the "Inverter Systems and Motors" Technical Manual) as a protective measure.
- ▶ Enter **MP2160.x = 2** if you use HEIDENHAIN EcoDyn synchronous motors. (In the commissioning flow chart = **(3)**).
This entry is intended for axis motors in EcoDyn operation. Here, field weakening is only active in a relatively small speed range, i.e. the maximum speed n_{max} is not significantly higher than the rated speed n_{rated} ($n_{max} > n_{rated}$). The control calculates a voltage-dependent, maximum speed n_{maxUz} . This speed ensures that the internal voltage U_{emk} in the motor only leads to a limited increase DC-link voltage if the power supply fails. Using a voltage protection module is therefore not necessary. Please note that the voltage-dependent, maximum speed n_{maxUz} can be lower than the maximum speed in the motor table. Please take this into consideration during project planning.
 U_0 = entry U_0 (no-load voltage) of the motor from the motor table
 n_{rated} = entry N-N (rated speed) of the motor from the motor table
 U_Z = 800 V (maximum DC-link voltage if the power supply fails)

$$n_{maxUz} = \frac{\frac{800V}{\sqrt{2}} \cdot n_{nenn}}{U_0}$$

Setting of MP2190:



Regenerative module (= (4) in commissioning flow chart)

The DC-link voltage for regenerative power modules is independent of the line voltage U_{Netz} . On regenerative HEIDENHAIN power modules it is 650 V (MP2190.x = 650) and on the regenerative Siemens modules 600 V or 625 V (MP2190.x = 600/625).

Actual DC-line voltage on non-regenerative power modules (= (5) in the commissioning flow chart)

Entering the DC-link voltage in MP2190 is also used by the control to ascertain the magnetizing current. At a line voltage of 400 V (interlinked), the rectified DC-link voltage is 565 V. If the fluctuation range of the line voltage is known (e.g. +/- 10 %), use the minimum value $U_{\text{Netz,Min}}$ of the line voltage to calculate the DC-link voltage according to the following equation:

$$U_z = \sqrt{2} \cdot U_{\text{NetzMin}}$$

If in addition you use a series reactor—for example to comply with EMC standard—then remember that the DC-link voltage can be reduced. So enter, if possible, the measured value of the DC-link voltage. If a measurement is not possible, then reduce the value calculated on the basis of the line voltage by approx. 3 %.

Example: Line voltage $U_{\text{Netz}} = 380 \text{ V} \Rightarrow$
MP2190 = $0.97 \cdot 537 = 521$.

If the line voltage in the machine's work area is less than the line voltage during commissioning, enter the value of the DC-link voltage at its later location during commissioning.

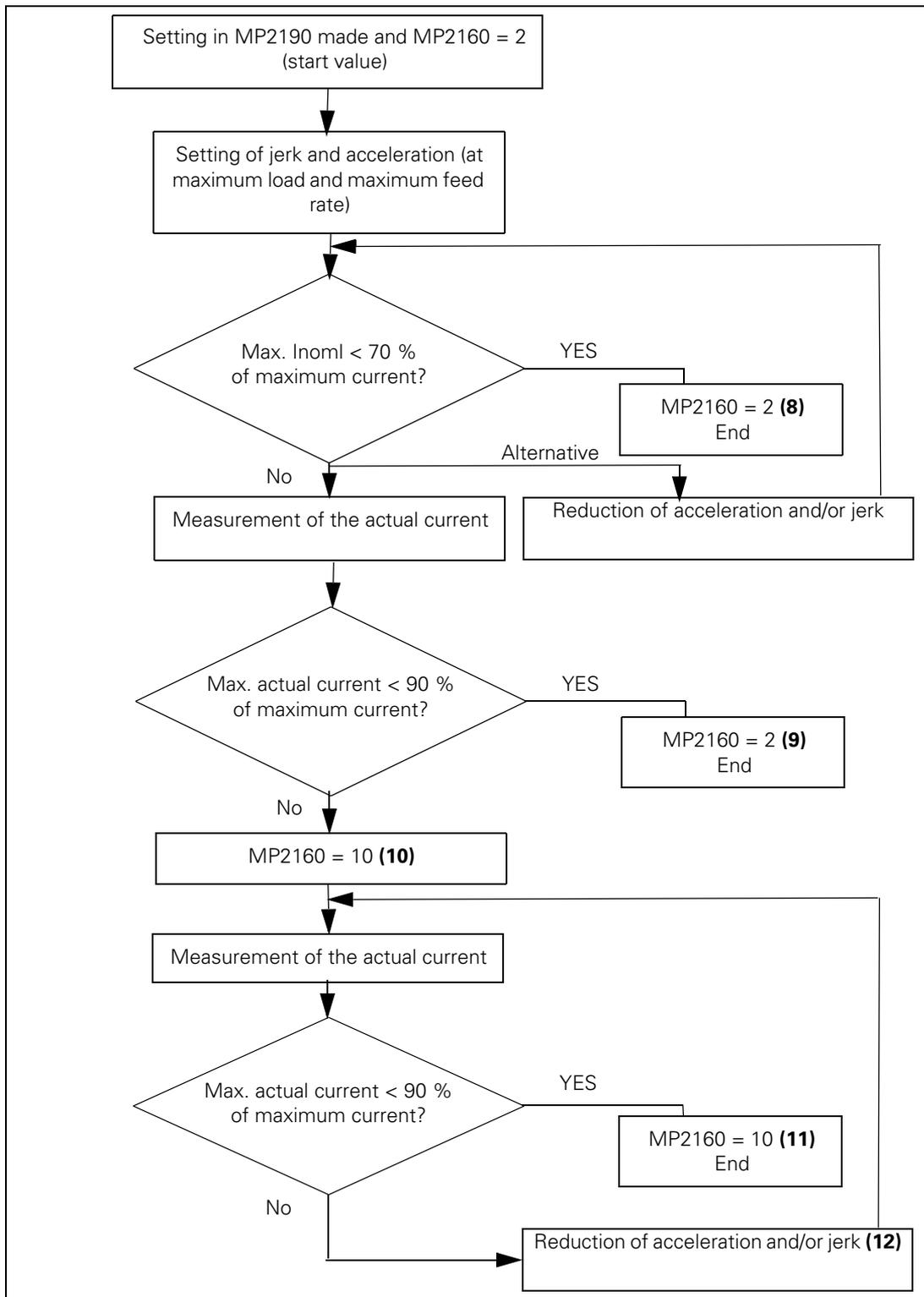
Stationary line voltage stable? (= (6) in the commissioning flow chart)

Any stationary fluctuating line voltage is to be accounted for by using the smallest occurring value for calculating the entry in MP2190, insofar as it has not already been accounted for through minimum line voltage. This ensures that the required magnetizing current is available, in particular during the acceleration phase. An excessively large DC-link voltage could lead to increased following error during acceleration and at high speeds.

Line voltage stable under load? (= (7) in the commissioning flow chart)

If you use an unstable power network on the machine, it can reduce the line voltage under load (when the axes are accelerated) and therefore also reduce the DC-link voltage. So it might be necessary to adjust the entry in MP2190 for the load. Compare the behavior of one axis during acceleration of only that axis (light load) with its behavior during acceleration of all axes, perhaps including the spindle (maximum load). If there are no differences in the following error or nominal current, the entry in MP2190 can remain unchanged. If these signals show anything unusual (surge in current or following error), reduce the entry in MP2190 accordingly.

Feed axis (EcoDyn operation):



Feed axis in EcoDyn mode—medium load (MP2160 = 2) ((8) in the commissioning flow chart)

After you have found the desired acceleration and jerk for the axis concerned, check the axis load during motion. To do so, reverse the axis up to maximum feed rate and record the torque current I_{nom1} on the internal oscilloscope. If the amount of torque current never exceeds approx. 70 % of the maximum current, there are enough reserves for the magnetizing components—the setting of MP2160 (=2) is completed. Please note: The maximum current is the lesser of maximum motor current and maximum power stage current.

Feed axis in EcoDyn mode—high load (MP2160 = 2) ((9) in the commissioning flow chart)

However, if the torque current is greater than 70 % of the maximum current, record the actual current I_{act1} on the internal oscilloscope to check whether it approaches the maximum current. The actual current contains both the torque current component as well as the magnetizing current component. If the amount of actual current never exceeds approx. 90 % of the maximum current, there are still enough reserves for the magnetizing components—the setting of MP2160 (=2) is completed here, too.

Feed axis in EcoDyn mode—high load (MP2160 = 10) ((10) in the commissioning flow chart)

With the entry MP2160 = 10 you can activate an alternative algorithm for determining magnetizing current. This can lower the magnetizing current. The total actual current in the field weakening range falls with it.

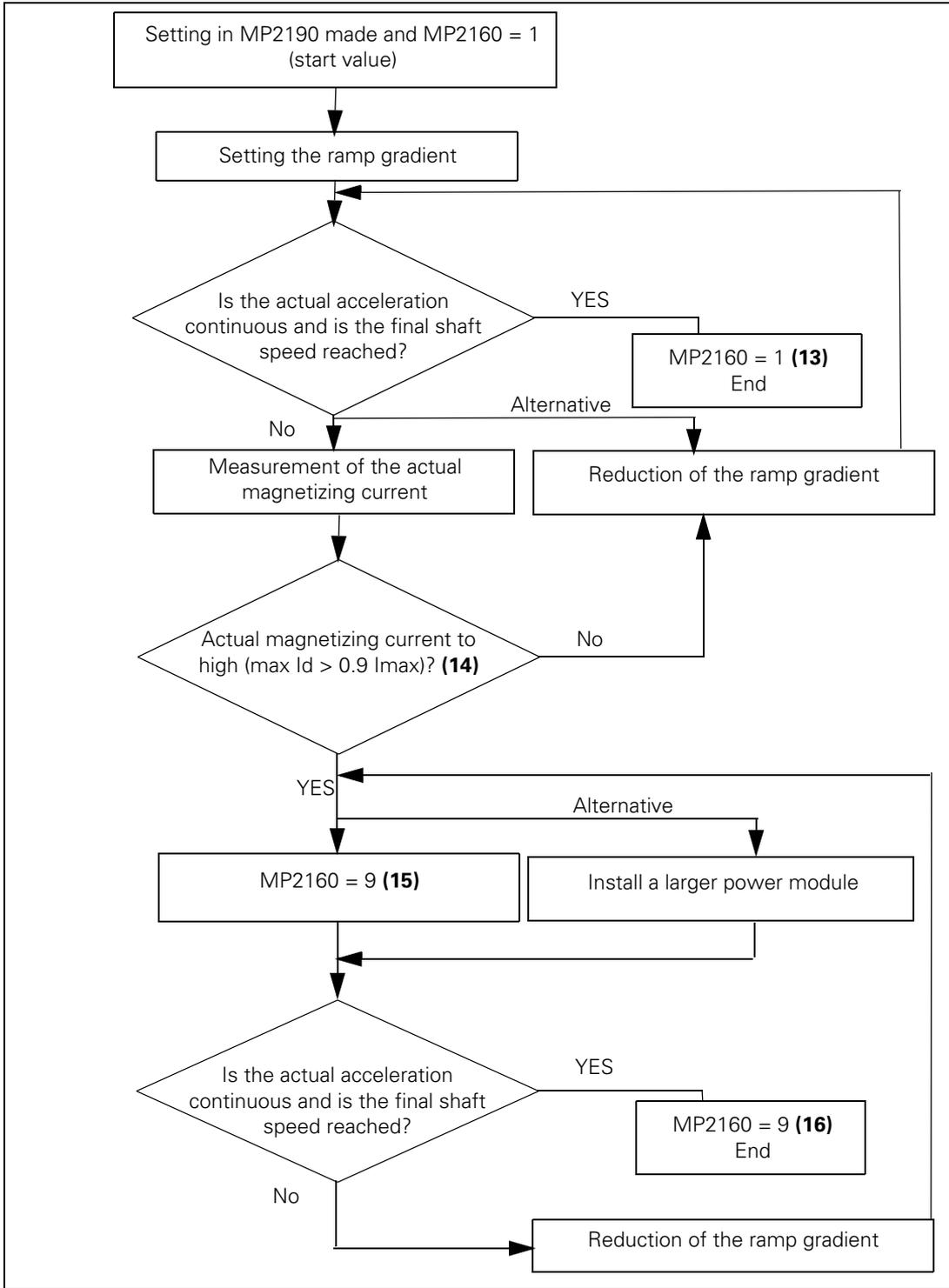
Feed axis in EcoDyn mode—high load (MP2160 = 10) ((11) in the commissioning flow chart)

If you succeed with MP2160 = 10 in limiting the actual current to less than 90 % of the maximum current, there are enough reserves again for the magnetizing component—the setting of MP2160 (=10) is completed here.

Feed axis in EcoDyn mode—high load (MP2160 = 10) ((12) in the commissioning flow chart)

If the actual current stays over 90 % of the maximum current, the load on the drive can be reduced through a reduction of the acceleration and/or jerk.

Synchronous spindle/torque motor:



Synchronous spindle/torque motor (MP2160 = 1) ((13) in the commissioning flow chart)

When you have set the desired ramp gradient (and rounding arc, if required) for the respective drive, test if the motor accelerates to the required final speed with a steady increase in acceleration. "Steady increase in acceleration" means that there are no drops in the acceleration curve. The actual acceleration usually declines with increasing speed, however.

Synchronous spindle/torque motor (MP2160 = 1) ((14) in the commissioning flow chart)

Drops in the acceleration curve indicate an excessive magnetizing-current consumption. You can see this if it exceeds 90 % of the maximum current of the drive. You can display the magnetizing current on the internal oscilloscope of the control.

Synchronous spindle/torque motor (MP2160 = 9) ((15) in the commissioning flow chart)

If the maximum current is limited by the power module (I_{max} of power module \ll I_{max} of motor) it can make sense to use a more powerful power module. As an alternative, you can use MP2160 = 9 to activate an algorithm for reducing the magnetizing current.

Synchronous spindle/torque motor (MP2160 = 9) ((16) in the commissioning flow chart)

If you succeed with MP2160 = 9 (or by installing a stronger power module) in attaining a continuous actual acceleration to the final shaft speed, there are enough reserves again for the magnetizing component—the setting of MP2160 (=9) is completed here.

The EcoDyn synchronous motors from HEIDENHAIN are operated with limited field weakening. No voltage protection module is necessary here.

- Select from the motor table the motors with the designation **QSY1xxx EcoDyn** or **QSY1xxx EcoDyn EnDat** for MP2200.x.

MP2160.x Field weakening for synchronous motors

Input:

- 0: No weakened-field operation
- 1: Weakened-field operation permitted (check if a voltage protection module is necessary)
- 2: Operation of a HEIDENHAIN EcoDyn motor
- 9: Weakened-field operation permitted, alternative to 1
- 10: Feed axis in EcoDyn mode, alternative to 2

6.9 Offset adjustment

Digital axes:

An offset adjustment at the output of the current controller is automatically compensated.

Analog axes:

The maximum permissible offset voltage in the control is 100 mV. If this voltage is exceeded, the error message **EXCESSIVE OFFSET IN <AXIS>** appears.

6.9.1 Offset adjustment with integral factor

With the integral factor you can adjust an offset automatically:

- ▶ Enter an integral factor in MP1080.x. Depending on the size of the factor the offset voltage will be reduced quickly or slowly.
- ▶ Play in the drives can result in instability in the control loop. In this case, enter the factor zero.

MP1080.x is effective only at a standstill.

MP1080.x Analog axes: Integral factor for offset adjustment

Input: Input 0 to 65 535

Digital axes: No function

Input: 0

6.9.2 Offset adjustment by code number

- ▶ Activate the offset adjustment with the code number 75 368.

The iTNC displays the offset values of the analog axes [mV] in the dialog line. The displayed offset value consists of the offset values that are generated in the motor controller and in the control.

If the values are to be automatically compensated:

- ▶ Press the ENT key or the CONTINUE soft key. The control outputs a corresponding countervoltage.

If nothing is to be changed:

- ▶ Press the END soft key.

If the offset adjustment is to be switched off again:

- ▶ Enter the code number 75 368 and press the NO ENT key or the CANCEL soft key.

The offset values are saved in the control and remain safe in the event of power interruption. After a control is exchanged, the offset adjustment must be repeated by means of the code number.

6.10 Contouring behavior

6.10.1 Radial acceleration

You can define the radial acceleration of axes in addition to the simple acceleration (MP1060.x):

► Define the radial acceleration in MP1070.

MP1070 limits the feed rate during circular movement according to the formula:

$$v = \sqrt{r \cdot \text{MP1070}}$$

v = feed rate during circular movement [m/s]

r = radius [m] (of the path of the tool center)

HEIDENHAIN recommends:

$$\text{MP1070} = 0,5 \dots 1 \cdot \text{MP1060}$$

If the programmed feed rate is less than that calculated above, then the programmed feed rate becomes effective.

MP1070 functions for operation with both following error and feedforward control.

MP1070	Radial acceleration
Input:	0.001 to 500 [m/s ²]

6.10.2 Contour velocity at corners

To comply with a defined tolerance, the iTNC can reduce the tool velocity before machining corners, line-to-arc transitions and arc-to-arc transitions. The control can react to a potential violation velocity tolerance up to 256 blocks in advance. This feature is known as "look-ahead":

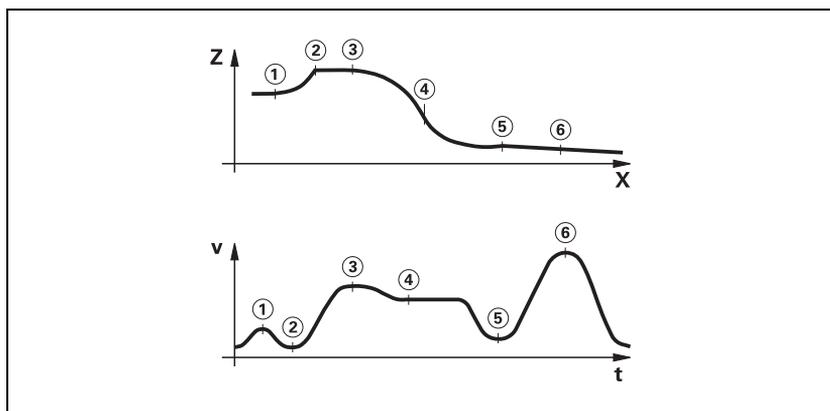
- ▶ Define the permissible tolerance for contour transitions in MP1096.x for movements at the machining feed rate and at rapid traverse. The larger the tolerance, the greater the tool velocity.

The user can overwrite this tolerance with Cycle 32 "Tolerance."

Jerk limitation (see "Interpolator" on page 814) and nominal-position-value filters enable the iTNC to machine 3-D surfaces at high speed. Prerequisite: The contour must be described with short line segments.

To ensure that cutter-compensated outside corners remain exact, a spline must be inserted into the cutter midpoint path instead of a transitional arc. The longer path of the spline (compared to the circle) results in an increased machining time. A spline also has the advantage of reducing the jerk:

- ▶ Enter bit 10 = 1 in MP7680.



Note

FN17: SYSWRITE ID1020 NR1 = <value> can be used from the NC program to activate machine parameter subfiles for various machining operations. See 1703.

MP1096 Tolerance for contour transitions at corners

Input: 0: No nominal position value filter
0.001 to 3.000 [mm]: Permissible tolerance at contour transitions

MP1096.0 With machining feed rate

MP1096.1 With rapid traverse

MP7680 Machine parameter with multiple function

Input: Bit 10 – Cutter-radius-compensated outside corners:

0: Insertion of a circular arc

1: Insertion of a spline curve

Proposed input value: %xx1xxxxxxxxxx

Bit 11 – Reserved



Rounding of corners

If you program M90, the tool velocity in following-error mode is kept constant at corners without radius compensation. This causes a corner rounding that varies with the feed rate (see the User's Manual).

If you program M112 or M124, defined arcs will be inserted at the corners regardless of the feed rate (see the User's Manual). The rounding arcs generate twice as many NC blocks, and the feed rate is now only limited by the radial acceleration.

The function M124 (contour smoothing filter) is considered to be valid syntax even without parameter T (minimum point interval). M124 alone (without parameter T) deactivates the filter function. M124 can thus be deactivated in an NC program or NC macro (e.g. at the beginning of a TOOLCALL macro).

- ▶ With MP7680 bit 7, specify whether the rounding arcs should always be inserted or only if the acceleration from MP1060.x or MP1070 has been exceeded at the corners.
- ▶ With MP7680 bit 8, specify whether a rounding arc or a cubic spline is to be inserted between lines during the M function M112. The feed rate is reduced enough to prevent any excessive jerk. This does not apply if F MAX is programmed. The cubic spline produces an additional jerk reduction. However, it requires a longer processing time than an inserted rounding, and due to the longer path of the spline (compared to the circle), the machining time also increases.

If you have set bit 8, you can specify with bit 9 whether the jerk will remain constant on the spline. The contour speed is adjusted for constant jerk.

MP7680

Machine parameter with multiple function

Input:

Bit 7 – Insertion of a defined rounding arc or spline:

0: Defined rounding arcs are always inserted

1: Defined rounding arcs are always inserted if the acceleration from MP1060.x or MP1070 was exceeded

Bit 8 – Insertion of a rounding arc or cubic spline:

0: Rounding arc is inserted

1: A cubic spline is inserted instead of a rounding arc

Bit 9 – Constant jerk on spline (Bit 8 = 1):

0: No constant jerk

1: Constant jerk

6.11 Monitoring functions

The NC monitors the axis positions and the dynamic response of the machine. If the fixed values are exceeded, it displays an error message and stops the machine.

With W1042/W1136 you can switch off the following types of monitoring for individual axes:

- Position monitoring
- Standstill monitoring
- Movement monitoring
- Nominal speed value monitoring

	Set	Reset
W1042/1136 Deactivation of monitoring functions	PLC	PLC
Axis-specific: Dependence on MP4020 bit 14		
0: Monitoring functions active		
1: Monitoring functions inactive		



Attention

Safe machine operation is not possible if the monitoring functions are switched off. Uncontrolled axis movements are not detected.

In the PLC program, PLC inputs specified in MP4132.x can be used axis-specifically to switch off the monitoring functions. With W524 bits 0 to 7 you can deactivate the monitoring functions for the axis group, or with W1042/W1136 for the individual axes. With W524 the position control loop is also opened.

If the reaction time of the PLC for switching off the monitoring functions is not sufficient, you must use a high-speed PLC input. The input defined with MP4130.0 is evaluated directly by the NC and is thus independent of the PLC cycle time. After transmission to the MC, the PLC input is processed in the position control loop cycle. The run times for PLC inputs also apply to this input, see page 1662.

- ▶ In MP4130.0, enter the number of the PLC input that is to be defined as high-speed PLC input. The number "0" is valid and means input number "0".



Note

Only the PLC inputs of a PLD in slots 1 to 4 of a PLB 6xxx can be used as fast PLC inputs.

- ▶ Define in MP4131.0 the activation criterion for the PLC input specified in MP4130.0.
- ▶ Enable MP4130.0 with W522 bit 0. As soon as the input is set, the monitoring functions are switched off, the axes stopped, and the drive is switched off. If the following error is greater than MP1030.x (positioning window), the actual value is saved as nominal value. The monitoring functions become active again if the high-speed PLC input is reset or MP4130.0 has been disabled with W522 bit 0.

MP1150.1 Time period for which the monitoring function is to remain off after the fast PLC input defined in MP4130.0 is set

Input: 0 to 65.535 [s]
Recommended: 0.2 to 0.5

MP4130.0 Number of the high-speed PLC input for switching off the monitoring functions

Input: 0 to 20 000 [no. of the PLC input]
-1: Function inactive

MP4131.0 Activation criterion for fast PLC input for switching off the monitoring functions

Input: 0: Activation at low level
1: Activation at high level

	Set	Reset
W522	PLC	PLC
Enabling the fast PLC inputs		
Bit 0: Fast PLC input is defined in MP4130.0 for switching off the monitoring functions		



6.11.1 Position monitoring

The axis positions are monitored by the iTNC as long as the control loop is closed.

The input values for position monitoring depend on the maximum possible following error (servo lag). Therefore the input ranges for operation with following error and velocity feedforward are separate.

For both modes of operation there are two range limits for position monitoring.

If the first limit is exceeded, the error message **EXCESSIVE SERVO LAG IN <AXIS>** appears. The machine stops.

You can clear this message with the CE key. An actual-to-nominal value transfer is then executed for the respective axes.

If the second limit is exceeded, the blinking error message **EXCESSIVE SERVO LAG IN <AXIS>** appears. The control-is-ready signal output is reset.

You cannot clear this message. You must restart the control to correct the error.

► In the machine parameters given below, define two range limits for position monitoring in each operating mode.

► Adjust the input values to the machine dynamics.

If blocked axes are the cause of the erasable error message **EXCESSIVE SERVO LAG IN <AXIS>**, a nominal velocity value may freeze, since the machine axes can no longer be moved:

► In MP1150.0, specify the time after which the nominal velocity value is to be deleted. After this time has expired, the actual position value is assumed as nominal position value. Before this time has expired, the error message cannot be cleared with the CE key. At this time the actual position value is assumed as nominal value, and the nominal velocity value is deleted.

MP1150	Position monitoring
MP1150.0	Delay time for deleting the nominal velocity value with the deletable error message: Excessive servo lag in <axis>
Input:	0 to 65.535 [s] Recommended: 0
MP1410.x	Position monitoring for operation with velocity feedforward control (erasable)
Input:	0.0010 to 30.0000 [mm] Recommended: 0.5 mm
MP1420.x	Position monitoring for operation with velocity feedforward control (emergency stop)
Input:	0.0010 to 30.0000 [mm] Recommended: 2 mm
MP1710.x	Position monitoring for operation with following error (erasable)
Input:	0.0000 to 300.0000 [mm] Recommended: 1.2 · following error
MP1720.x	Position monitoring for operation with following error (emergency stop)
Input:	0.0000 to 300.0000 [mm] Recommended: 1.4 · following error



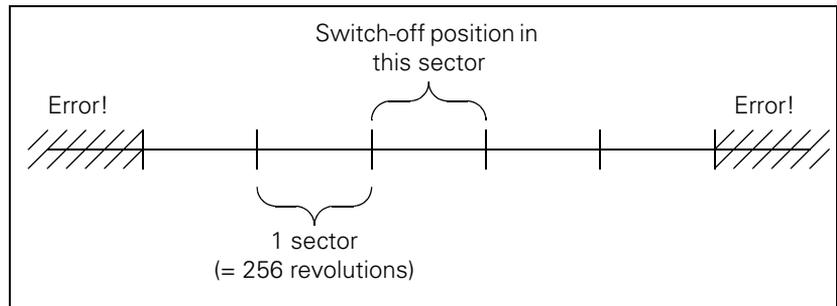
Difference between position at switch-on and shutdown

When the control is switched off, the actual position of the axes is saved with an absolute encoder. During switch-on it is compared with the position values read by the encoder.

If the positions differ by more than the difference defined in MP1146.x, a pop-up window appears with both positions. The new position must be confirmed with a soft key. If it is not confirmed, the error message **Check the position encoder <axis>** appears.

Special case: Absolute multiturn rotary encoder

The control stores an overflow (more than 4096 revolutions of the encoder) internally. Additionally, the number of traversed sectors (1 sector = 256 revolutions) is stored. After the drives are switched on, the current sector is compared to the stored sector.



If the motor is farther from the switch-off position than two complete sectors when it is switched on (further than the sector "after next"), the error message **Switch-off pos. <axis> unequal ENDAT** is displayed immediately after the drives are switched on.



Attention

The error must then be corrected!

After the control has been restarted, it is assumed that the number of revolutions is correct again.

The pop-up window may appear, stating that the positions at switch-on and shutdown differ by more than MP1146.x. If the motor is located at the correct position, you can confirm the message with the **YES** soft key.

MP1146.x Difference between the position at shutdown and the position read in via the EnDat interface

Input: 0.0000 to 300.0000 [mm] or [°]
0: No difference permitted

6.11.2 Nominal speed value monitoring

With the CC 61xx, nominal speed value monitoring is active as a rule, and applies to the following values:

- For axes:
The maximum feed rate is 15% greater than the maximum motor speed (N-MAX) multiplied by MP1054.
- For spindle(s):
The maximum spindle speed is 15% greater than the maximum motor speed (N-MAX) multiplied by the gear transmission ratio.

If monitoring responds, the error message **88C0 Max. nominal motor speed %s exceeded** is output and an emergency stop reaction is initiated.

When using an EcoDyn motor, the error is triggered when the maximum permissible voltage is exceeded.



6.11.3 Movement monitoring

Movement monitoring is possible during operation both with velocity feedforward and with following error.

During movement monitoring, the actual path traveled is compared at short intervals (several servo cycles) with the nominal path calculated by the NC. If during this period the actual path traveled differs from the calculated path, the flashing error message **MOVEMENT MONITORING IN <AXIS>** appears.

Analog axes:

An existing offset during a standstill may cause a potential at the analog output without any resulting positioning movement:

- ▶ In MP1140.x, enter a threshold from which the movement monitoring should go into effect.

Digital axes:

There is no offset.

- ▶ In MP1140.x, enter the speed from which the movement monitoring should go into effect.

For digital axes, in addition to the comparison of actual and nominal values, the calculated position from the pulses of the position encoder are compared with the pulses of the speed encoder:

- ▶ Enter in MP332.x the number of signal periods and in MP331.x the path for the number of signal periods (See "Encoders" on page 606).
- ▶ Enter the distance per motor revolution in MP1054.x. A formula can also be entered in MP1054.x.
- ▶ In MP1144.x, enter a limit value for this position difference. If you are not using a position encoder, you must enter 0 in MP1144.x as the position difference.

If the difference is greater than the input value from MP1144.x, the error message **MOVEMENT MONITORING IN <AXIS> B** appears.



Attention

If you enter the maximum value in MP1140.x or MP1144.x, no movement monitoring is active.
Safe machine operation is not possible without the movement monitoring function.

MP1140.x Threshold above which the movement monitoring functions

Input: Analog axes: 0.030 to 10.000 [V]
Digital axes: 0.030 to 10.000 [1000 min⁻¹]
Recommended: 0.030 [1000 min⁻¹]

MP1054.x Distance of one motor revolution [mm or °]

Input: Analog axes: Nonfunctional
Digital axes: Entry of a formula possible,
see "Special case: Entering a formula" on page 479

MP1144.x Motion monitor for position and speed

Input: Analog axes: Nonfunctional
Digital axes: 0 to 99 999.999 [mm]
0: No monitoring

6.11.4 Standstill monitoring

Standstill monitoring is effective during operation both with velocity feedforward and with following error, as soon as the axes have reached the positioning window.

If the position difference is greater than the value defined in MP1110.x, the blinking error message **STANDSTILL MONITORING IN <AXIS>** appears. The message also appears if, while moving to a position, an overshoot occurs that is larger than the input value in MP1110.x, or if the axis moves in the opposite direction when beginning a positioning movement:

- ▶ In MP1110.x, enter a threshold from which the standstill monitoring should go into effect.

MP1110.x Standstill monitoring
Input: 0.0010 to 30.0000 [mm]

6.11.5 Positioning window

The positioning window defines the limits within which the control considers a position to have been reached. After the position has been reached, the control begins running the next block. The position controller can correct a disturbance inside this window without activating the "Return to the Contour" function.

- ▶ In MP1030.x, define the size of the positioning window.

MP1030.x Positioning window
Input: 0.0001 to 2.0000 [mm]

Axes in position

Once the axes have moved into the positioning window, the corresponding bits are set in W1026/W1104. This also applies to the status after the machine control voltage is switched on. Axes that are not used are considered to be in position.

The NC resets the bits as soon as you start a positioning movement or traverse the reference marks.

In the **Electronic Handwheel** mode of operation the bit for the current handwheel axis is reset.

W1026/W1104 is not set for contours that can be machined with constant surface speed.

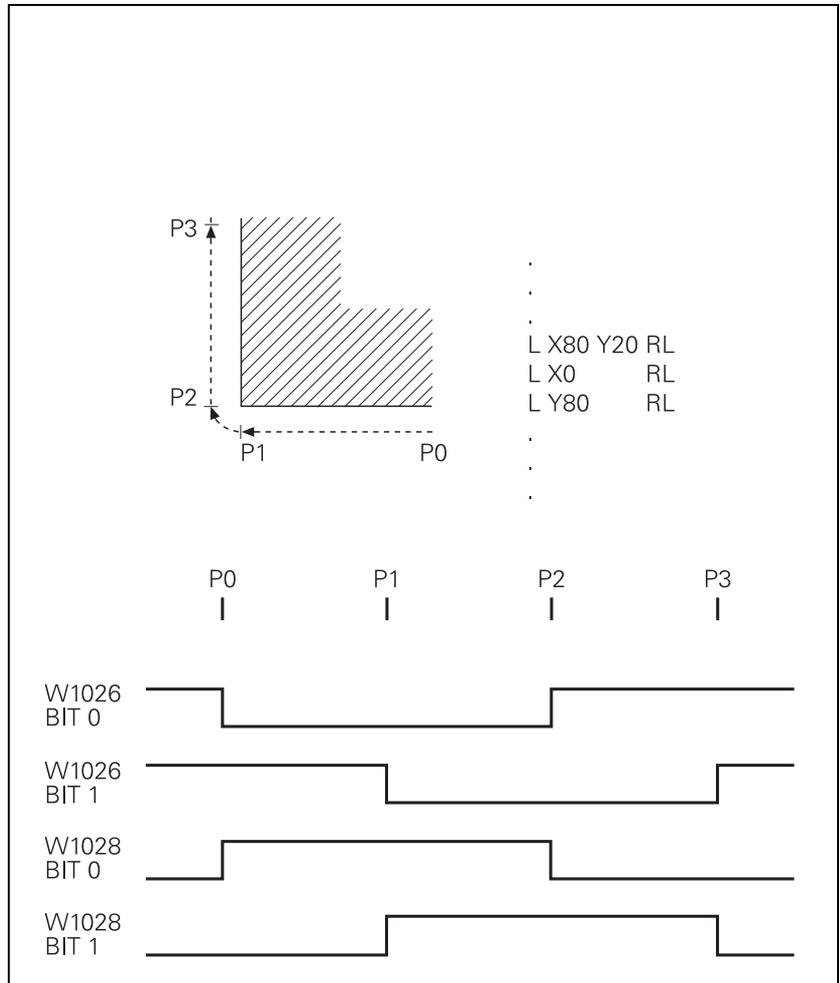
	Set	Reset
W1026/1104 Axes in position	NC	NC
Axis-specific: Dependence on MP4020 bit 14		
0: Axis not in positioning window		
1: Axis in positioning window		



Axes in motion

During axis movement, the NC sets the corresponding bits in W1028/W1108.

	Set	Reset
W1028/1108 Axes in motion	NC	NC
Axis-specific: Dependence on MP4020 bit 14		
0: Axis not in motion		
1: Axis in motion		



6.11.6 Monitoring of the power supply unit

The rectified supply voltage of the power supply unit is monitored. The supply voltage must lie within a defined range (400 V +/- 10 %). If this is not the case the power supply unit reports an AC fail (PF.PS.AC).

At the same time, the DC-link voltage is monitored:

- If approx. 800 V– with UM xxx, UR xxx or over approx. 830 V– with UE xxx, UxC is exceeded, the NC revokes the pulse release (reset) for the IGBT of the power module. The motors coast out of loop to a stop. No energy is returned to the dc link.
- Below approx. 300 V AC (UE, UxC) the power supply unit reports AC fail (PF.PS.AC).
- If the DC-link voltage falls below approx. 410 V– (UV, UM, UR, UE, UxC), the power supply unit reports a powerfail (signal PF.PS.ZK).
- Below approx. 200 V–, (UV, UR) the control is reset (signal $\overline{\text{RES.PS}}$). Immediately below it the power supply unit switches off.
- ▶ The evaluation of the two signals -PF.PS.AC and -PF.PS.DC is activated by MP2195 bit 7 and bit 8, respectively.

Inverter signal	Meaning
AC fail ($\overline{\text{PF.PS.AC}}$)	Failure of supply voltage for inverter
Power fail (PF.PS.ZK)	DC-link voltage failure

Since the AC fail is reported to the control before the powerfail, the control has more time to react to the subsequent DC-link voltage failure.



Note

Only certain HEIDENHAIN power supply units provide the AC-fail signal (see the Technical Manual for "Inverter Systems and Motors"). If you are using power supply units that do not provide this signal, you must not select the AC-fail signal in MP2195.

If a power fail is triggered on the control, all drives are brought to a controlled stop. The PLC outputs are switched off and the control freezes to ensure that the hard disk can no longer be accessed.

The control must be turned off and on again.

Also, the performance of each UV(R) supply unit is monitored. If the maximum power of a UV(R) is exceeded, the spindle power is reduced if the spindle is supplied by the same UV. This function can be deselected in MP2195 bit 10 = 1.

MP2195 Power monitoring of the UV(R)

Input: Bit 10
 0: Monitoring active
 1: Inactive

Module 9167 Monitoring of DC-link voltage

With this module you can switch the DC-link voltage monitoring for powerfail (U_z <approx. 385 V or 410 V) on and off.

If you don't call the module during the first PLC cycle, the supply voltage monitoring is automatically started after the first PLC cycle.

Call:

PS B/W/D/K <Command code>
 0: DC-link voltage monitoring off
 1: DC-link voltage monitoring on

CM 9167

PL B/W/D <Error code>
 0: Command executed
 -1: Transferred parameter invalid

Error code:

Marker	Value	Meaning
M4203	0	DC-link voltage monitoring on or off
	1	Error code in W1022
W1022	2	Transferred parameter invalid

6.11.7 Temperature monitoring

Temperature of the MC

The internal temperature of the MC is continuously monitored. At about 55 °C the temperature warning **TNC temperature warning** appears. If the temperature does not fall below 55 °C, the warning is reactivated after two minutes. Beginning at about 60 °C the error message **TNC temperature too high <temperature> °C** appears and an emergency stop is triggered. If the temperature does not fall below 60 °C when the machine is switched on again, the error message reappears after 10 to 20 seconds.

The temperature of the MC can be found with Module 9133.

Module 9133 Output of hardware information

Call:

PS B/W/D/K <Mode>
0: Internal temperature sensor in [°C]
1: Temperature CPU1 (basic PCB) in [°C]
2: Temperature CPU2 (additional PCB) in [°C]
3: Voltage of buffer battery in [mV]
4: 5 V supply voltage of main board
5: 3.3 V supply voltage
6: Rotational speed of the housing fan

CM 9133

PL B/W/D <Value>

Error code:

Marker	Value	Meaning
M4203	0	Value was determined
	1	Error code in W1022
W1022	2	Invalid number given
	8	No second CPU found (for code 2)

Motor temperature

To measure the motor temperature, a KTY 84 must be connected at pin 13 and pin 25 of X15 to X20. The temperature value is ascertained at least once per second. The maximum permissible motor temperature is taken from the motor table.

As soon as the given temperature is exceeded, the blinking error message **MOTOR TEMPERATURE <AXIS> TOO HIGH** appears and the drives are automatically switched off.

Module 9165 Sample the current motor temperature

Appropriate measures can be taken before the motor reaches the maximum temperature.

Call:

PS B/W/D/K <Axis>
Axis-specific: Dependence on MP4020 bit 14

CM 9165

PL B/W/D <Temperature>
Range: 0 to 255 °C

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Control has no current controller

Temperature measurement of linear and torque motors

Linear motors usually have a KTY and several PTC thermistors or thermostiches for temperature measurement.



Attention

The PTC thermistors or thermostiches must be **galvanically isolated** and evaluated by the PLC.

The KTY requires **double insulation** to the motor windings, which must be provided by the motor manufacturer. Otherwise, do **not** connect the KTY to the control!

The KTY is monitored by the control (NC). The temperature signal is conducted to the control together with the encoder signals (X15 to X20). If the KTY is not to be evaluated, this function must be deactivated over MP2220.x, MP2221.x (see "Monitoring functions" on page 894).

For linear and torque motors, the conductor for the temperature signal of the KTY is frequently in the motor power cable, which can cause interference. Since the conductor for the temperature signal is then led into the conductor of the speed encoder, the interference causes noise in the encoder signals. HEIDENHAIN therefore recommends conducting the temperature signals over the line drop compensator, so that the interference signals are filtered.



Note

HEIDENHAIN recommends the additional temperature monitoring of the PTC thermistors or thermo switches via the PLC, since these are distributed over the entire length (linear motors) or circumference (torque motors) (as opposed to the KTY, for which there are only spot measurements).

For example, PTC thermistors can be connected to a PLC input via the securely grounded 3RN1013-1BW10 thermistor motor-protection device from SIEMENS.

Temperature of the power module's heat sink

At X51 to X60 the temperature warning signal is available at pin 10a.

If the permissible temperature of the heat sink on the power module is exceeded, this signal is reset.



Attention

To avoid destroying the power module, the drives must be brought immediately to a standstill after a temperature warning.

Data on maximum permissible temperatures are available from the manufacturer of your power module.

The TEMPERATURE WARNING signal is not evaluated in the NC.

- Use Module 9160 or 9066 to interrogate the temperature warning, and take appropriate measures.

6.11.8 Internal power supply and housing fan

Via the PLC you can capture and evaluate the current values of the internal power supply and the speed of the housing fan.

Module 9133 Output of hardware information

Call:

PS B/W/D/K <Mode>
 0: Internal temperature sensor in [°C]
 1: Temperature CPU1 (basic PCB) in [°C]
 2: Temperature CPU2 (additional PCB) in [°C]
 3: Voltage of buffer battery in [mV]
 4: 5 V supply voltage of main board in [mV]
 5: 3.3 V supply voltage in [mV]
 6: Rotational speed of the housing fan in [min⁻¹]

CM 9133

PL B/W/D <Value>

Error code:

Marker	Value	Meaning
M4203	0	Value was determined
	1	Error code in W1022
W1022	2	Invalid number given
	8	No second CPU found (for code 2)

MP4020 PLC functions

Format: %xxxxxxxxxxxxxx

Input: Bit 13 – Monitoring the housing fan of the MC/CC

 0: Monitoring active

 1: Inactive

6.11.9 I²t monitoring

General information

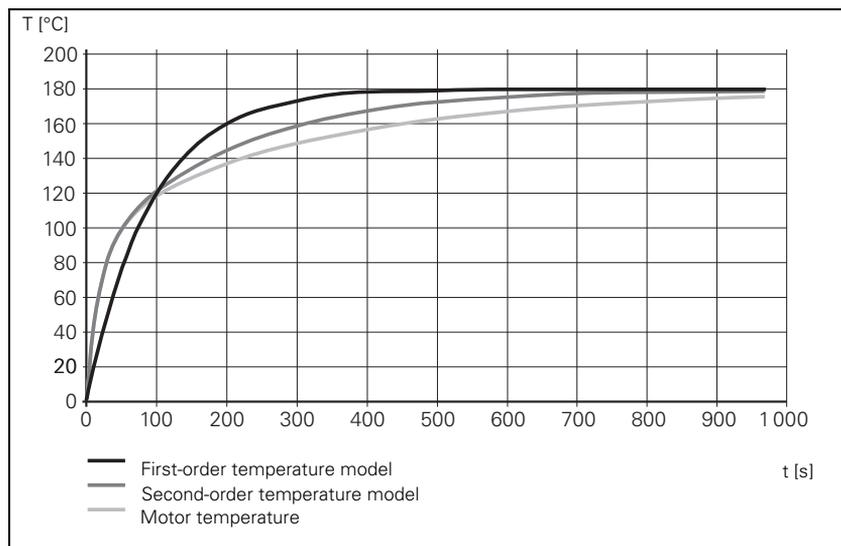
HEIDENHAIN inverter systems feature individual I²t monitors, one for each power module and motor.

Function

An I²t monitor calculates and supervises the temperature pattern in a thermal motor or power-stage model during operation, even if temperature measurement is also provided by the hardware.

The active current, the rated or stall current, (multiplied by MP2302.x for motors and by MP2304.x for power modules) and a device-specific temperature model are the basis for calculation. A first-order temperature module is available for monitoring power modules, first and second-order modules are available for motors. These models make it possible to permanently calculate the temperature of the stator winding in the motor or the semiconductor in the power module.

Temperature models in an example comparison (motor)



The I²t monitor responds if this calculated temperature exceeds a certain limit.

Because temperature increase and heat dissipation are uneven when the motor is stationary or moving slowly, the I²t monitor distinguishes between standstill and traversing mode.

This limit range is defined in a motor table or power module table. The following entries are important:

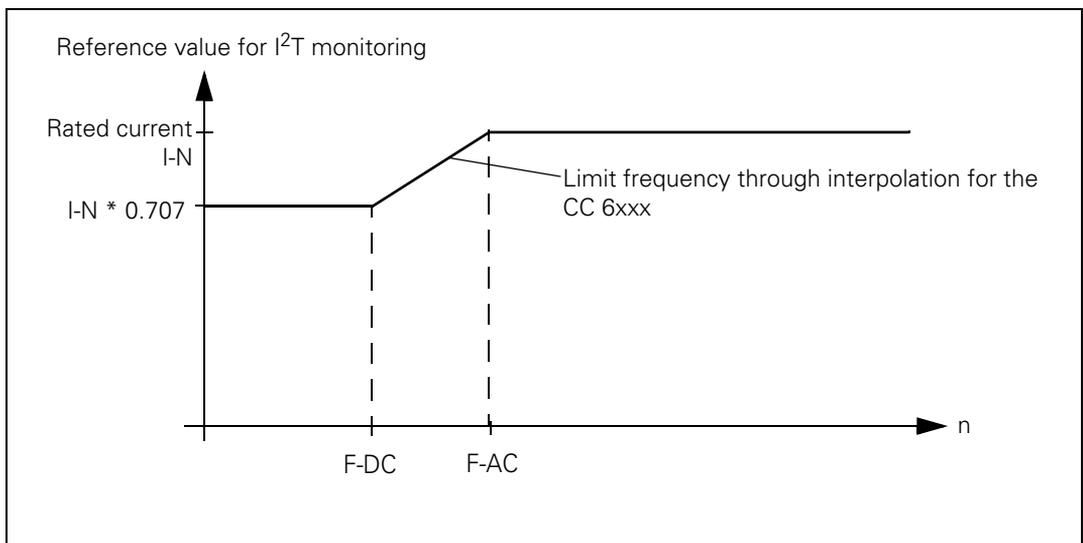
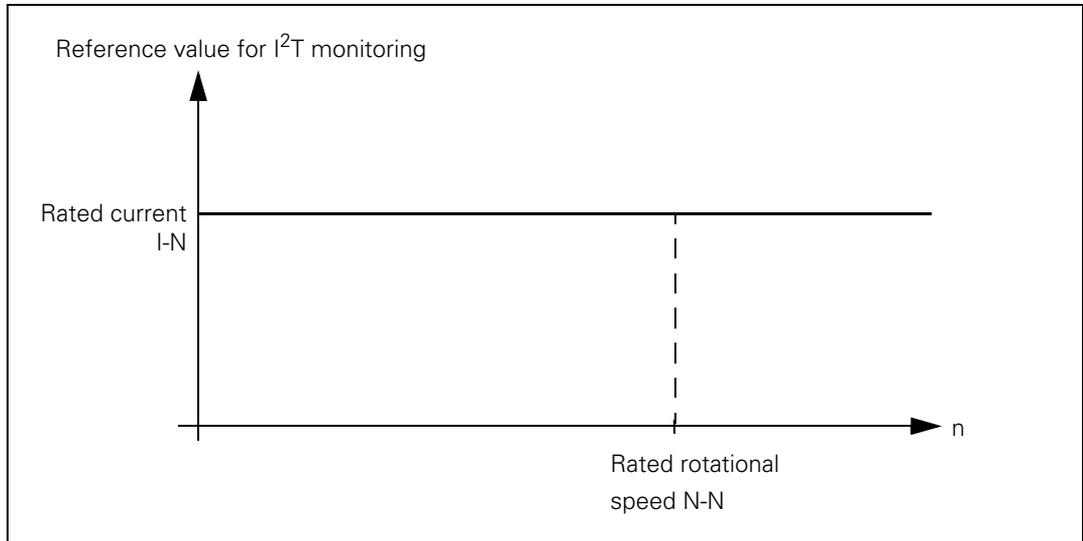
- F-AC (transition frequency in traversing mode [Hz])
- F-DC (transition frequency at standstill [Hz])

Fundamentals

The following graphics illustrate these parameters in relation to the reference voltage.

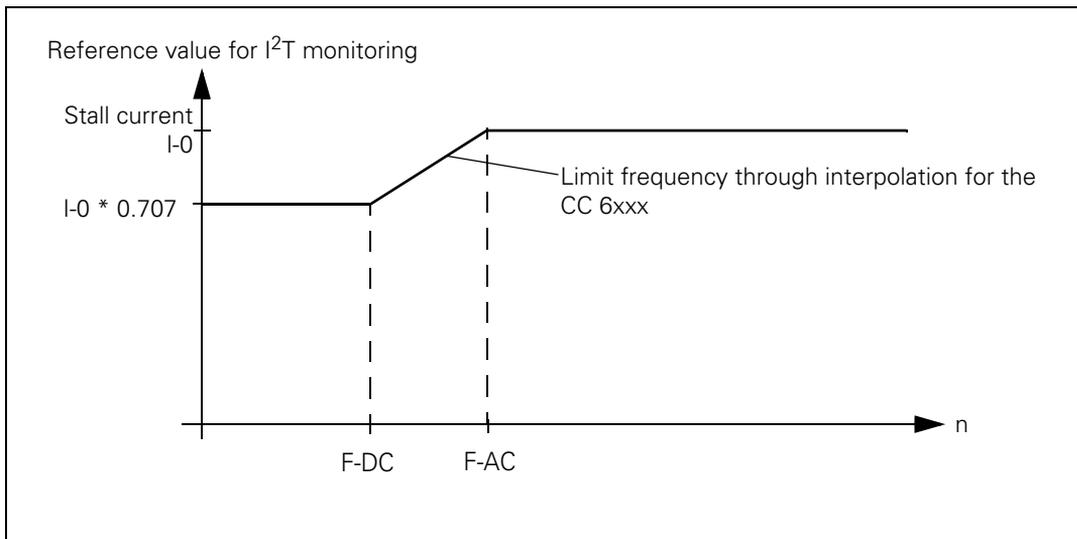
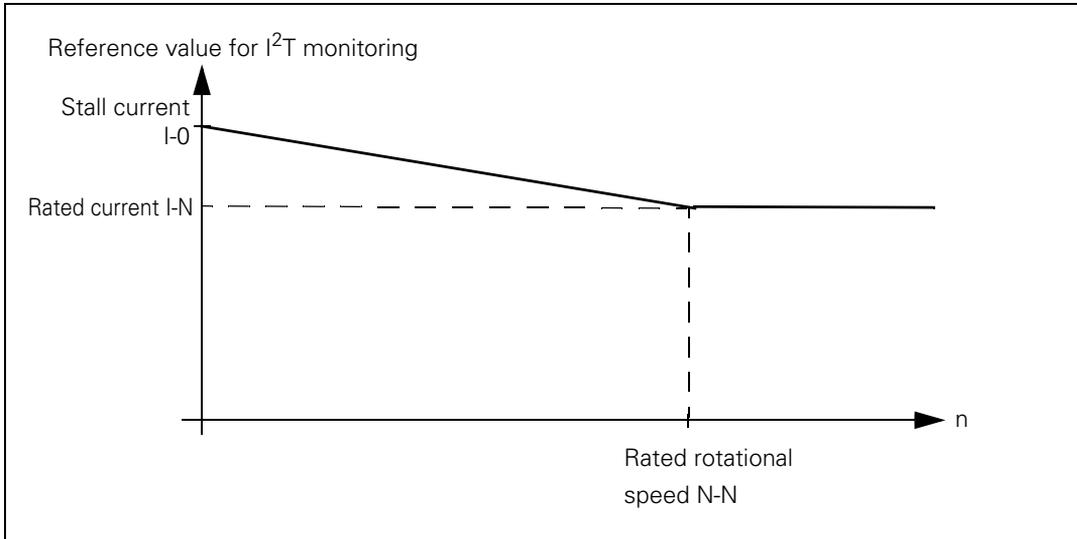
With the CC 6xxx it is possible to use an interpolated current range for the transition from standstill to traverse. This allows an exact calculation of the temperature model.

If there is no stall torque value given in the motor table, the following model of current (with respect to the rated current) is used to calculate the temperature in the motor. The factors for MP 2302.x and MP 2304.x are not yet taken into account.



If the stall current value is given in the motor table, the following model of current (with respect to the stall current) is used to calculate the temperature in the motor. This is only used for synchronous motors, however. For asynchronous motors, the entry I-0 is not the (thermal) continuous stall current but the magnetizing current, because it is not necessary to distinguish between continuous stall current and rated current on asynchronous motors. This differentiation is not necessary for asynchronous motors because with them, a slip frequency always occurs under load.

For synchronous motors, the factors from MP2302.x and MP2304.x are not yet taken into account in the following description.



Commissioning and evaluation

- ▶ In MP2302.x, enter the factor for the I^2t monitoring of the motor. The input value is a factor for the reference current (1 = 100% of the motor's standstill current or rated current). If you enter zero, the I^2t monitoring for the motor (not for the power module) is switched off. If you enter a value less than 1, then a motor standstill current or rated current (reference current) reduced by this factor is used for calculation. The actual current is rated higher by this factor relative to the reference current, and the I^2t value is increased more quickly. This means that the results for values less than 1 provide increased protection for the motor. Values greater than 1 provide less protection for the motor.
- ▶ In MP2304.x, enter the factor for I^2t monitoring of the power module. The input value is a factor of the power module's rated current (1 = 100%). If you enter zero, the I^2t monitoring for the power module (not for the motor) is switched off. If you enter a value less than 1, then a power module current or rated current (reference current) reduced by this factor is used for calculation. The actual current is rated higher by this factor relative to the reference current, and the I^2t value is increased more quickly. This means that the results for values less than 1 provide increased protection for the power module. Values greater than 1 provide less protection for the power module.
- ▶ All required entries for calculation of a temperature model have to be available in the motor table or power module table. See "Temperature models" on page 914.
- ▶ Use Module 9160 to interrogate the I^2t monitoring, see "Interrogation through PLC module" on page 913.

Limit values

The limit values for the I^2t function (measure of the permissible temperature inside the device [%]) are managed by the control's NC and are structured as follows:

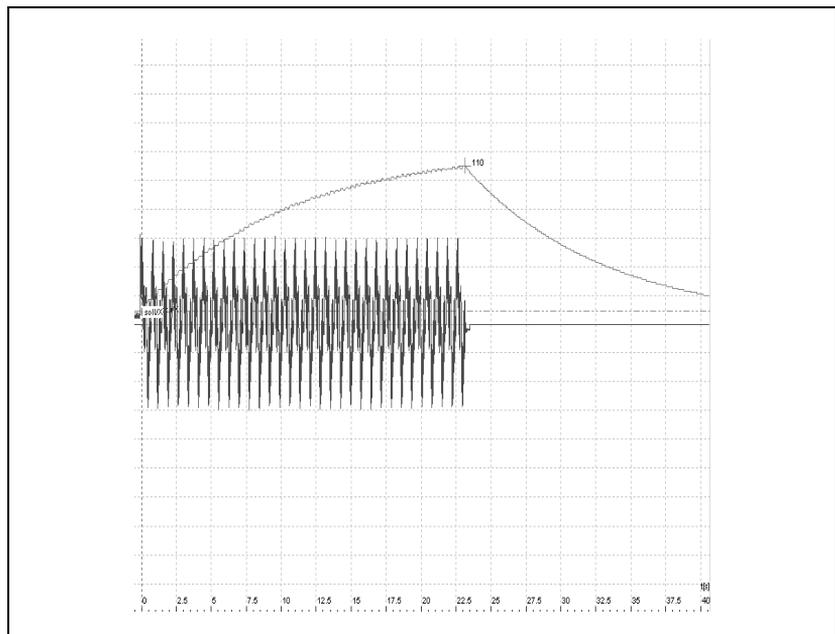
- Value exceeds 100%: An axis-specific I^2t early warning is sent to the PLC (for evaluation and possible countermeasures such as reduction of the feed rate with error message by PLC program with the aid of PLC Module 9160). If the value does not exceed 110% and falls below 90%, the axis-specific early warning is reset.
- 110%: An NC stop is triggered and the drives are switched off.



Note

In the oscilloscope you can display the current value of the I^2t monitoring of the motor and power module, as well as the current load of the drive.

Motor overload with I^2t monitoring



Machine parameters

MP2302.x Factor for I²t monitoring of motor
Input: 0 to 1000.000 [-: rated or stall current of the motor]
0: I²t monitoring of motor switched off
1: Rated or stall current is reference value

MP2304.x Factor for I²t monitoring of the power module
Input: 0 to 1000.000 [-: rated current of power module]
0: I²t monitoring of power module switched off
1: Rated current of power module is reference value

Interrogation through PLC module

Module 9160 Status request from temperature monitoring and I²t-monitoring

The I²t monitoring reported by the module is given with respect to the first I²t monitor response (power stage or motor), if both I²t monitors are activated (MP2302.x, MP2304.x). This early warning is withdrawn as soon as the limit for reset is reached. For the response behavior, see "Limit values" on page 912.

Call:

CM 9160

PL D <Temperature monitoring>

Bit 15876543210

Axis:Sxxxxx987654321

PL D < I²t monitoring I²t early warning>

Bit 15876543210 15 876543210

Axis Sxxxxx987654321 Sxxxxx987654321

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Control has no current controller

PLC Module 9160 (status interrogation of temperature and I²t monitoring) is a special case. This PLC module issues an error message when the PLC module interface is active (MP4020 bit 14) and can no longer be used. Please use PLC Module 9066 (status of HEIDENHAIN hardware) instead of PLC Module 9160.

Temperature models

The temperature model of the motor or power module is defined by the entries in the motor table or power module table, respectively (motor.mot, motor.amp). Remember that calculation of which temperature model to use depends exclusively on the availability or nonavailability of the parameters. In addition, the parameters for motors and power modules are to be evaluated separately.

These parameters are explained below using the respective temperature model for the calculation.

Temperature model, first order

The following values (entries in the motor table or power module table) are required for the first-order temperature model to calculate the temperature.

■ **F-DC** [Hz]:

Lower limit frequency for the transition of traverse to standstill with the CC 6xxx.

F-DC = 0 – Default value (0) is active

F-DC > 0 – Input value in Hz is active

■ **T-DC** [s]:

Thermal time constant for operation at standstill (not evaluated at present)

■ **F-AC** [Hz]:

Upper limit frequency for the transition from standstill to traverse.

F-AC = 0 – Default value (0) is active

F-AC > 0 – Input value in Hz is active

■ **T-AC** [s]:

Thermal time constant for the motor or power stage. Identifies the point in the temperature curve at which 63 % of the maximum temperature is reached.

T-AC = 0 – Default value: 10 s for axes, 150 s for ball screw

T-AC > 0 – Input value [s] for power modules. For motors, this input value is active if $Tth_2 = 0$.

■ Only for motors **Tth₂** [s]:

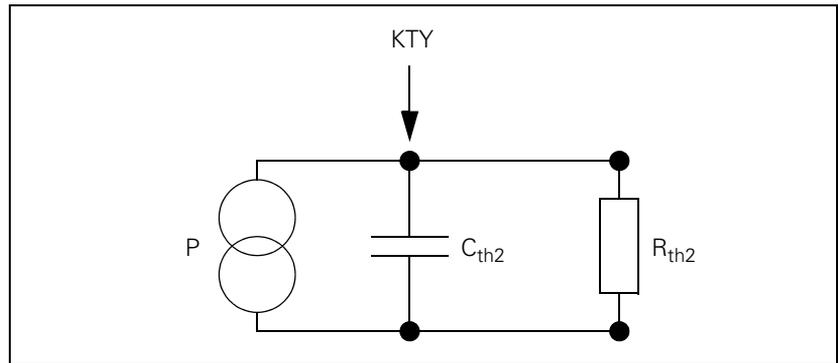
Thermal time constant for the motor. Identifies the point in the temperature curve at which 63 % of the maximum temperature is reached.

Tth₂ = 0 – Default value: 10 s for axes, 150 s for ball screw

Tth₂ > 0 – Input value in [s] for motors

F-AC and F-DC = 0.1 Hz can be used as standard value for commissioning.

First-order temperature model of the motor



- P: Heat output of the three phases
- KTY: KTY temperature sensor in the winding
- C_{th2} : Thermal capacity of the motor housing
- R_{th2} : Thermal resistance on the motor housing
- T_{th2} : Thermal time constant $R_{th2} \cdot C_{th2}$

Temperature model, second order

The following values (entries in the motor table) are required for the second-order temperature model to calculate the temperature (default values [axis/spindle] are valid for the entry "0"):

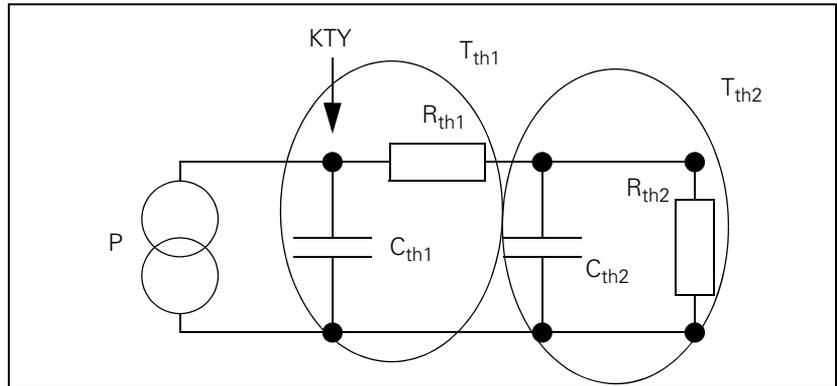
- **F-DC** [Hz]:
Lower limit frequency for the transition of traverse to standstill with the CC 6xxx.
F-DC = 0 – Default value (0 Hz) is active
F-DC > 0 – Input value in Hz is active
- **T-DC** [s]:
Thermal time constant for operation at standstill (not evaluated at present)
- **F-AC** [Hz]:
Upper limit frequency for the transition from standstill to traverse.
F-AC = 0 – Default value (0 Hz) is active
F-AC > 0 – Input value in Hz is active
- **Tth₁** [s]:
Thermal time constant for the transition from winding to housing
Tth₁ = 0 – Default value (0 s) is active
Tth₁ > 0 – Input value in [s] is active
- **Rth₁** [K/W]:
Thermal resistance for the transition from winding to housing.
Rth₁ = 0 – Default value: 0 K/W
Rth₁ > 0 – Input value in [K/W] is active
- **Tth₂** [s]:
Thermal time constant for the transition from housing to coolant
Tth₂ = 0 – Default value: 10 s for axes, 150 s for ball screw
Tth₂ > 0 – Input value in [s] is active
- **Rth₂** [K/W]:
Thermal resistance for the transition from winding to coolant
Rth₂ = 0 – Default value: 0 K/W
Rth₂ > 0 – Input value in [K/W] is active
- When the CC starts up, the current motor temperature (KTY sensor) is taken into the calculation model in order, for example, to compensate any excessive temperatures.



Note

- All parameters have to be entered for the model to become active. If a parameter is missing, the first-order temperature model becomes active, either with the thermal time constant "Tth2" or with "T-AC."

Second-order temperature model of the motor



- P: Heat output of the three phases
- KTY: KTY temperature sensor in the winding
- C_{th1} : Thermal capacity of the winding
- C_{th2} : Thermal capacity of the housing
- R_{th1} : Thermal resistance winding/housing
- R_{th2} : Thermal resistance housing/coolant
- $T_{th1} = R_{th1} \cdot C_{th1}$
- $T_{th2} = R_{th2} \cdot C_{th2}$

6.11.10 Momentary utilization of drive motors

Module 9166 provides the momentary utilization of the given drive motor as a percentage value. The module provides the percentage of the drive's mechanical power output that arrives at the tool.

Utilization means:

Speed range	$n_{actl} < n_{noml}$	$n_{act} \geq n_{noml}$
Asynchronous or synchronous motor	$\frac{ M_{curr} }{ M_{rated} }$	$\frac{ P_{curr} }{ P_{rated} }$

If $n_{actl} = n_N$, then:

$$\frac{|M_{curr}|}{|M_{rated}|} = \frac{|P_{curr}|}{|P_{rated}|}$$

M_{curr} and P_{curr} are ascertained cyclically by the current controller. This is based on the equivalent circuit data of the motor and the present motor current.

$P_{rated} = P-N$ from motor.mot

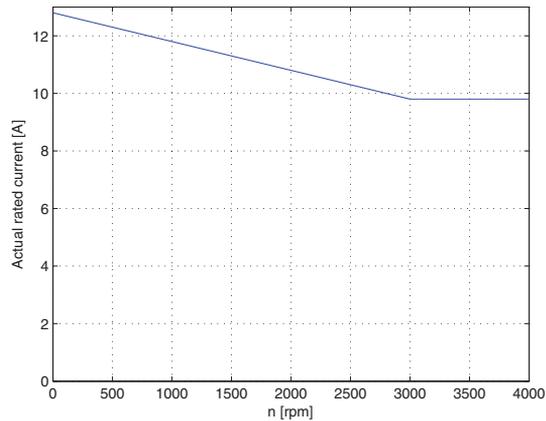
$N_{rated} = N-N$ from motor.mot

$$M_{rated} = \frac{P_{rated}}{2 \cdot \pi \cdot \frac{n_{rated}}{60}}$$

Consideration of I0 when using synchronous motors:

If I0 differs too much from I_{rated}, then the utilization display can be adapted via MP2312.x. In this case you must set $MP2312.x = I_{rated}/I_0$.

Example for the behavior of I0 and I_{rated} with an OSY155F-EcoDyn motor:



In the diagram you can see that:

$I_0 = 12.8 \text{ A}$, $I_{rated} = 9.8 \text{ A}$, $n_{Rated} = 3000 \text{ rpm}$

For the synchronous motor:

$$M_{actl} = kT \cdot I_{actl}$$

$$M_{rated} = kT \cdot I_{rated}$$

$$M_{rated} = f(n)$$

Module 9166 Momentary utilization of the drive motor

The evaluation through MP2312.x is already calculated in the utilization of the drive motor.

Call:

PS B/W/D/K <Axis>
Axis-specific: Dependence on MP4020 bit 14

CM 9166

PL B/W/D <Utilization of the drive in %>

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Control has no current controller

MP2312.x Factor for utilization of motors

Input: 0 to 1 000.000

0: Factor = 1

6.11.11 Determining the current torque of a drive

With Module 9170 you can determine the averaged, maximum and minimum torque of a drive.

The information about the torque utilization always refers to the values determined between two PLC cycles. The momentary torque is returned in tenths of per cent of the nominal torque.

Example:

For the first axis (e.g. X axis and QSY 190D EcoDyn with M_N of 23.0 Nm), an averaged torque of 281 is determined via Module 9170.

$$6.46 \text{ Nm} = \frac{23.0 \text{ Nm}}{1000} \times 281$$

$$M_{\text{act}} = \frac{M_{\text{nenn}}}{1000} \times \text{Value}_{\text{Modul9170}}$$

Module 9170 Find the current torque

Module 9170 makes the averaged, maximum and minimum load values of the drive of an axis available between two PLC cycles as a reference value for the rated torque value or rated power. One mode also supplies the algebraic sign of the averaged load.

Call:

PS	B/W/D/K	<Axis> Axis-specific: Dependence on MP4020 bit 14
PS	B/W/D/K	<Mode> 0: Torque value in per cent of the rated torque 1: Torque value in tenths of per cent of the rated torque 2: Torque value in tenths of per cent of the rated torque with algebraic sign; minimum and maximum torque will not be calculated (=0)
CM	9170	
PL	B/W/D	<Averaged torque value> Mode 0: In per cent Mode 1: In tenths of per cent Mode 2: In tenths of per cent with algebraic sign
PL	B/W/D	<Maximum torque value> Mode 0: In per cent Mode 1: In tenths of per cent Mode 2: 0
PL	B/W/D	<Minimum torque value> Mode 0: In per cent Mode 1: In tenths of per cent Mode 2: 0



Error code:

Marker	Value	Meaning
M4230	0	Torque ascertained
	1	Error code in W1022
W1022	1	Invalid mode
	2	Invalid axis number



6.11.12 Status of HEIDENHAIN power modules

Module 9066 is used to determine the status information of the HEIDENHAIN power supply unit.

Module 9066 Status of HEIDENHAIN supply unit

Call:

PS B/W/D/K <Code for hardware components>
0: HEIDENHAIN supply unit
1: Serial number of the SIK
2: I2t early warning
3: I2t monitoring
4: Error of analog part

CM 9066

PL B/W/D <Status information>
Code 0: HEIDENHAIN power supply unit
Bit 0: Not used
Bit 1: DC-link voltage too high
Bit 2: Heat sink temperature too high
Bit 3: Short-circuit of a motor phase with Uz
Bit 4: DC-link current too high
Bit 5: Power supply unit not ready
Bit 6: Leakage current too high
Code 1: Serial number of the SIK
Codes 2 to 4: Axis masks

Error code:

Marker	Value	Meaning
M4203	0	Status ascertained
	1	Error code in W1022
W1022	2	Invalid code
	24	Module was called in a spawn job or submit job

The HEIDENHAIN power supply units have several status signals which lead to error messages on the control. MP2195 is used to suppress the error message for each status signal.

HEIDENHAIN does not recommend suppressing the error messages from the power supply units. If you are using a UE 2xx, the signals must be suppressed because the UE 2xx compact inverter does not provide these signals.



Status signals that are already inactive during control power-up

The handling of status signals from HEIDENHAIN power supply units, which are already inactive during control start-up, varies depending on MP2195 bit 0:

MP2195 bit 0 = 0: Missing signals cannot be detected with Module 9066 and do not result in an error message when the drive is switched on.

MP2195 bit 0 = 1: After the PLC program has been compiled, missing signals can be detected with Module 9066 and trigger an error message when the drive is switched on. Signals that are not provided by the power supply unit must be suppressed with MP2195 (bit 1 to bit 6), because non-existent signals are always identified as errors.

The SUPPLY.SPY table for the power supply modules contains the STATUS-SIG column for the status signals of the power supply module. This column informs you, among other things, whether the PF.PS.AC and PF.PS.DC is supported by the power supply module. This information in SUPPLY.SPY is evaluated by the CC 61xx.

Bits 7 and 8 of MP2195 enable you to deactivate the error messages that may be triggered by the evaluation of these signals (PF.PS.AC, PF.PS.DC), without changing the SUPPLY.SPY table.



Note

- Signals that change their status during operation are always identified as errors.
- Errors of the supply module (e.g. excessive Uz or Uv temp.) now lead to an NC stop (if M148 then LIFTOFF) with a subsequent emergency stop.

MP2195 Handling of status signals from HEIDENHAIN power supply units

Input: Bit 0 – Status signals that are already active during control power-up.
0: Missing signals are ignored
1: Missing signals are evaluated
Bit 1 – $\overline{\text{ERR.UZ.GR}}$ signal
0: Error message is not suppressed
1: Error message is suppressed
Bit 2 – $\overline{\text{ERR.TMP}}$ signal
0: Error message is not suppressed
1: Error message is suppressed
Bit 3 – Reserved
Bit 4 – $\overline{\text{ERR.IZ.GR}}$ signal
0: Error message is not suppressed
1: Error message is suppressed
Bit 5 – RDY.PS signal
0: Error message is not suppressed
1: Error message is suppressed
Bit 6 – $\overline{\text{ERR.ILEAK}}$ signal
0: Error message is not suppressed
1: Error message is suppressed
Bit 7 - PF.PS.AC
0: Previous behavior; Error message is not suppressed
1: Error message is suppressed
Bit 8 - PF.PS.DC
0: Previous behavior; Error message is not suppressed
1: Error message is suppressed
MP2195 can also be overwritten by the PLC and the LSV2 protocol.



6.11.13 Controlling the motor brakes

The motor brakes are controlled with the $\overline{\text{BRK}}$ braking signal, which is transmitted to the **HEIDENHAIN inverters** via the PWM interface (X51 to X62). The corresponding outputs are activated there. See the basic circuit diagrams.

Control of the motor brakes via the PWM interface must be deactivated for non-HEIDENHAIN inverters that do not support this function:

- ▶ Set $\text{MP2230.x} = 0$

The motor brakes are opened no later than 50 ms after the speed controller is switched on. For safety reasons, the controller is not switched off until the braking signal has been output:

- ▶ Enter in MP2308.x the time (overlap time) after which the controller is to be switched off (after the braking signal has been output).

If the inverter sends the $\overline{\text{RES.PS}}$ reset signal, then the $\overline{\text{BRK}}$ braking signals are output immediately upon switch-off of the controllers, i.e. without any overlap time.

Activated brakes cause a change in the controlled system. The motor with the changed controlled system is controlled during the overlap time. This can lead to oscillations when the controller is switched off. These oscillations are suppressed by the NC software. MP2220 bit 3 can be used to deactivate the suppression of vibrations. HEIDENHAIN does not recommend switching off the suppression of the oscillations.

In MP2309.x you can define a time period in which the speed and position controller parameters are adjusted to values for controlling a closed brake when the drive is switched on. This parameter can be used to avoid oscillations in the drive during switch-on when the brake is still closed and the controller is already active.

MP2220 Monitoring functions

Input: Bit 3 – Switching off the controller when the motor brakes are activated
 0: Suppress oscillations
 1: Oscillations are allowed

MP2308.x Time between output of the braking signal $\overline{\text{BRK}}$ and switching off of the controller (overlap time)

Input: 0.001 to 5.000 [s]
 0: 0.200 s

MP2309.x Controller parameters adjusted to closed brake

Input: 0: Previous behavior; not active
 0.001 to 5.000 [s]

Automatic test of the motor brakes upon switch-on

You can carry out an automated functional test of the motor brake after switching on the drive, either before traversing the reference mark or through activation by PLC Module 9143.

The operation and testing of motor-holding brakes must comply with Information Sheet No. 005 "Gravity-loaded axes (vertical axes)" issued by the engineering technical committee (BGM (German Employer's Liability Association in the metal industry)).

This brake test only functions in combination with HEIDENHAIN inverter systems and only when using the brake output on X392/X393 if it is wired according to the basic circuit diagram from HEIDENHAIN.

For the period of one second, a torque (current) is applied while the brake is active. The path that the axis has moved is then measured. If the permissible path is exceeded, the error message **8130 Motor brake defective <axis>** appears, and the axis remains controlled. The test is carried out simultaneously for all affected axes.



Attention

In case of an error, the axis must be moved to a safe position, and physically supported, if necessary. Only then may the machine be switched off so that the defect can be corrected.

If no motor current flows while testing the motor brakes, the error message **8140 No current for brake test <axis>** appears.

- ▶ Enter in MP2230.x a factor for the motor stall current with which the motor brake test is to be carried out. If the test is not to be done, or for motors without brakes, enter MP2230.x = 0.
The brake output on X392/X393 is also controlled if MP2230.x = 0.

The reference value for the factor from MP2230.x is the stall current **I₀** entered in the motor table. If **I₀** in the motor table equals 0, then the rated current **I-N** from the motor table is used.

Recommended input value for MP2230.x:

$$\text{MP2230.x} \geq 1,3 \cdot \frac{M_L}{M_0}$$

M_L: Maximum load torque of the axis. In a standard case the holding torque of a vertical axis is used here. It is to be ensured via activation of the brake that a vertical axis does not fall down when the drive controllers are switched off.

M₀: Stall torque of motor



Always keep in mind:

- Torque for motor test $\geq 1.3 \cdot$ maximum load torque of the axis
- Stall torque of the motor \geq maximum load torque of the axis
- Holding torque of the motor brake \geq torque for the motor test



Note

- Please note that when reading the current via the internal oscilloscope, on the CC 6xxx you are seeing the effective value of the current.
- Please note that the test torque can only be generated with a certain factor of uncertainty. Factors of influence here are the accuracy of the current sensors and the torque constant of the motor.

- ▶ Enter in MP2232.x the permissible path that the motor is allowed to move against the brake. If the test is not to be carried out, or for motors without brakes, enter MP2232.x = 0.
The TNC monitors the maximum path during the brake test (MP2232.x) based on the information from the speed encoder (motor encoder).

MP2232.x must be $<$ MP1110.x so that the standstill monitoring does not activate!

Recommended input value for MP2232.x:

$$MP2232.x = 2 \cdot \alpha \cdot \frac{MP1054.x}{360^\circ}$$

α : Permissible braking angle: Backlash of the motor brake as per the manufacturer specifications (for HEIDENHAIN motors, $\alpha \leq 1^\circ$)

Example:

QSY 155B-EcoDyn: $M_0 = 13 \text{ Nm}$, $M_{Br} = 40 \text{ Nm}$

$M_L = 11 \text{ Nm}$

$$MP2230.x \geq 1,3 \cdot \frac{11 \text{ Nm}}{13 \text{ Nm}} = 1,1$$

MP1054.x (distance per motor revolution) = 20 [mm]

$\alpha = 1^\circ$

$$MP2232.x = 2 \cdot 1^\circ \cdot \frac{20 \text{ mm}}{360^\circ} = 0,111 \text{ mm}$$

MP2230.x Multiplier for motor current during test of motor brake

Input: 0.100 to 30.000 [\cdot motor stall current]

0: No test of motor brakes, or motor without brake. The inverter brake output is controlled nevertheless.

Recommended: $1.3 \cdot M_L / M_0$

MP2232.x Maximum permissible path during test of motor brakes

Input: 0 to 10.0000 [mm] or [°]

Module 9143 Activate the brake test

With this module an axis-specific brake test with the configuration from the machine parameters or with other values for MPs 2230 and 2232 can be started.

Constraints:

- Synchronized axes
For synchronized axes, only the brake test of the master can be configured and requested via the PLC module. If a brake test for an associated slave drive of the synchronized axis is configured via MP2230.x, then the slaves are automatically tested together with the master. The settings in the machine parameters are used for the brake test of the slave drives.
In order to start the brake test of synchronized axes via PLC Module 9143, all drives of a synchronized axis must be switched on via the PLC program before the brake test can be performed. If a servo drive involved is not switched on, the brake test is canceled with the error message **8330 Brake test was canceled**.
- Programming in a submit job blocks other submit jobs until the test is completed.
- The PLC module automatically passes the processing time to other spawn and submit processes.

Call:

PS	K/B/W/D	<Axis number> 0 = 1st axis, 1 = 2nd axis, etc.
PS	K/B/W/D	<Multiplier for motor stall current> Value in 1/1000 or 0: Default MP2230 (factor of nominal current)
PS	K/B/W/D	<Permissible traverse path> Value in 0.1 [um] or 0: Default MP2232
CM	9143	
PL	B/W/D	<Status/Error> 0: Brake OK 1: Brake defective 2: Invalid axis or negative values for rated current or traverse path 3: Call during running NC program or during other PLC jobs 4: Call was made from a cyclic PLC program 5: Error during data exchange 6: Not allowed for safe control 7: Drive not ready 8: Brake test was canceled (e.g. by emergency stop)



Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Invalid axis programmed (invalid axis number, not a closed-loop axis, axis currently open-loop axis or slave axis) or negative values for the traverse path or current are programmed
	8	Module is not allowed for control with functional safety
	20	Module was not called in a submit or spawn job
	21	Call during program run or during other active PLC jobs for the programmed axis
	40	Drive not ready
	45	Canceled due to error during data exchange or due to external influences (e.g. emergency stop)

Module 9048 Interrogate the operating states of axes

Module 9048 is used to interrogate the operating state of a certain axis or for all axes together.

Call:

PS B/W/D/K <Axis number>
 Axis number: Individual information for a programmed axis
 -1: Information for all axes, bit-coded as axis mask

PS B/W/D/K <Status information>
 0: Brake test active/inactive
 1: Free rotation active/inactive

CM 9048

PL B/W/D <Status>
 Interrogation of an individual axis: 1/0 = active/inactive
 Interrogation of all axes: Bit-encoded axis mask

Error code:

Marker	Value	Meaning
M4203	0	Status ascertained
	1	Error code in W1022
W1022	1	Invalid value for status information
	2	Invalid axis programmed: If status information 0 was transferred, then this error occurs if an invalid axis number, an open-loop axis or an axis that is temporarily not a closed-loop axis was selected. If status information 1 was transferred, then this error occurs if an invalid axis number, an open-loop axis, a slave axis or an NC axis that is not a rotary axis was selected.

6.11.14 Emergency stop monitoring

For the emergency stop routine of the iTNC 530 HSCI without functional safety (FS), the PLB has the PLC "control-is-ready" output X9/3a (-STO.A.G) and the two emergency stop inputs -ES.A (X9/7a) and -ES.B (X9/7b) (ES: Emergency Stop). The delayed control of the safety relay in the compact inverters or supply units (X71, X72) over the PLC output -STO.A.G is mandatory. Please read the corresponding basic circuit diagram.

The -STO.A.G output signalizes the readiness of the control and is = 1 until one of the following conditions is met:

- Controller unit reports fatal error
- Controller unit reports an error with an emergency stop as reaction
- Main computer reports fatal error
- Main computer reports an error with an emergency stop as reaction
- PLC, PLC program reports an error with an emergency stop as reaction
- No PLC program is run

As a result of these events, -STO.A.G becomes 0. When the error message is acknowledged, -STO.A.G become 1 again and the control is again ready for operation. To prevent that the control then immediately goes into closed-loop mode, the -STO.A.G signal must be wired into the emergency stop chain (-ES.A, -ES.B). Switching -STO.A.G off must result in a switch-off of the -ES.A and -ES.B inputs. This triggers additionally an external emergency stop. In addition, -STO.A.G = 0 must remove the latch circuit of the machine control voltage. This makes it possible to switch the machine back on by means of "Control voltage on."



Note

Switching STO.A.G off must result in a switch-off of the -ES.A and -ES.B inputs.

On the PLB, for example, the -ES.A and -ES.B are available for triggering an emergency stop on the iTNC 530 HSCI with functional safety (FS). However, the wiring of the safety relays in the compact inverters or power supply units via the PLC outputs STO.A.G and STOS.A.G is then optional. Please note the special documentation for controls with functional safety (FS) on the corresponding basic circuit diagram!

Internal emergency stop

If an **internal emergency stop** is triggered (e.g. due to standstill monitoring),

- the nominal speed value "null" is transmitted, braking the drives on the intended braking ramp (usually at the limit of current)
- the control-is-ready output is switched off (-STO.A.G on PL; responsible watchdog reacts after 10 ms at the latest)
- the inverter enabling is switched off (-STO.A.x over PWM cable; responsible watchdog reacts after the time set in MP2172 of 1 to 60 s), and the inverters are disconnected,
- markers M4177, M4178 and M4260 are set by the NC.

Machine parameter MP2172.x serves to monitor the braking of the drives. The monitoring time for the braking process is defined in MP2172.x. The time for switching off the pulses (entry in MP2172.x) must be greater than the maximum possible braking time of the axis/spindle that can occur through electrical braking. Especially for axes/spindles without mechanical braking, you must ensure that the pulses are not switched off until after the maximum possible braking time for the axis/spindle that can occur through electrical braking. Non-decelerated axes/spindles coast to a stop after pulse switch-off. In the worst case, this can cause damage to the machine. Specific operating conditions of the machine, such as maximum feed rate, overload on the axes, etc., must also be taken into account.

During an internal emergency stop, an error appears on the screen. Depending on its class (cause), it is possible that an error **cannot** be cleared with the CE key:

- ▶ Correct the error and restart the switch-on routine.

MP2172 Delay for the STO.A.x signal (inverter enable)

Input: 0 to 60 [s] as an integer
 0 = 3 seconds (default value)

External emergency stop -ES.A and -ES.B

For triggering an emergency stop, controls with HSCI have the emergency-stop inputs -ES.A and -ES.B (e.g. -ES.A (X9/7a) and -ES.B (e.g. X9/7b) on the PL). A "0" signal at -ES.A or -ES.B triggers an emergency stop reaction with the error message "external emergency stop." The PLC inputs I3 and I32 known up to now are being replaced in HSCI systems by -ES.A and -ES.B.

Both emergency stop inputs -ES.A and -ES.B must be wired. Both inputs must carry the same signal. This signal is safety related and must be generated by the appropriate devices (e.g. dual-channel emergency stop button, safety contactor combination).

If an **external emergency stop** is triggered,

- the nominal speed value "null" is transmitted, braking the drives on the intended braking ramp (usually at the limit of current)
- the **EMERGENCY STOP** error message is displayed, and
- markers M4177, M4178 and M4260 are set by the NC.

You can clear the error message with **CE** after switching the machine control voltage back on.

However, an external emergency stop (-ES.A, -ES.B) does not disconnect the output -STO.A.G.

If marker M4580 is set, then instead of the external emergency stop, the control loops of all axes and of the spindle are opened and an NC stop is performed. If the marker M4580 is deleted when an emergency stop is pending, the external emergency stop is released.

Maximum reaction times of the control until the response is triggered at the input after the signal change.

Input	Response period	Response
-ES.A = 0	5 ms	Emergency stop
-ES.B = 0		

Minimum required pulse width for detection by the control at the -ES.A, -ES.B inputs: pulse width > 3 ms

		Set	Reset
M4177	Clearable error message displayed	NC	NC
M4178	Error message EXTERNAL emergency stop is displayed	NC	NC
M4580	Suppress the emergency stop, open all position control loops, NC stop	PLC	PLC
M4260	Control-is-ready acknowledgement	NC	NC



Axis releases

At the same time HEIDENHAIN recommends using the axis-specific "axis releases" of the CC through the configuration in MP4132.x.

On controls with HSCI but without functional safety (FS), a pure axis enable, which up to now has been realized over I32, can only be realized through the function of the former X150. However, a CC 61xx does not have an X150 connector for axis-specific or axis-group-specific drive enabling.

But instead, you can now enter in machine parameters MP4132.0 to MP4132.7 eight numbers of those PLC inputs that simulate the inputs of X150. If one of these PLC inputs is switched off, the associated drives are decelerated on the provided braking ramp (usually at the limit of current). Over PLC module 9161 you have to switch off the axes of the affected axis groups. To do so, you can use PLC module 9157 to scan the corresponding PLC inputs.

MP2040.0 to MP2040.7 continue to be used for the assignment of axes to individual axis groups.

If the corresponding PLC inputs are switched off, the NC

- transmits the nominal speed value "null", braking the drives on the intended braking ramp (usually at the limit of current)
- detects the standstill of the drives after braking and switches off the pulses of the power stage over the PWM interface (-STO.B.x).

Machine parameter MP2173.x serves to monitor the braking of the drives. The monitoring time for the braking process is defined in MP2173.x. After the monitoring time has expired, the control checks whether the servo drive is at a standstill. If this is not the case, the control assumes that a serious error has occurred and switches off the pulses of the power stages via -STO.B.x (signal to PWM cable). This ensures that, after a request to switch off the servo drives (e.g. emergency stop, X150, PLC or alarm), the pulses are safely switched off (via -STO.B.x) at the latest after the time specified in MP2173.x expires. If the standstill is detected right before the time defined in MP2173.x expires, the active braking process is continued and the pulses are not switched off until after the overlap time.

The time for switching off the pulses (entry in MP2173.x) must be greater than the maximum possible braking time of the axis/spindle that can occur through electrical braking. Especially for axes/spindles without mechanical braking, you must ensure that the pulses are not switched off until after the maximum possible braking time for the axis/spindle that can occur through electrical braking. Non-decelerated axes/spindles coast to a stop after pulse switch-off. In the worst case, this can cause damage to the machine. Specific operating conditions of the machine, such as maximum feed rate, overload on the axes, etc., must also be taken into account.

Maximum reaction times of the control until the response is triggered at the input after the signal change.

Input	Response period	Response
PLC input on PLD in slots 1 to 4	18 ms	Stop axis group along the emergency braking ramp
PLC input on PLD in slots 5 to 8	21 ms	Stop axis group along the emergency braking ramp

Minimum required pulse width for detection by the control at PLC inputs on PLDs in slots 1 to 4: pulse width > 6 ms



Attention

The time for switching off the pulses via -STO.B.x, (entry in MP2173.x) must always be greater than the maximum possible braking time of the axis/spindle that can occur through electrical braking. However, do not set too large a value in MP2173. Otherwise the safety function of the machine parameter cannot be ensured.

MP2173.x Pulse switch-off of the power stage (STO.B.x)

Input: 0.2 to 100.000 [s]
0 = 3 seconds (default value)

PLC outputs

PLC outputs can be controlled by the PLC program and switched off in the event of an emergency stop. Additionally, the IOconfig configuration tool enables you to define the behavior of PLC outputs in the event of an emergency stop. Outputs that are defined as output type 3 (switch-off in case of an emergency stop) are switched off (= 0) if an external or internal emergency stop is triggered. In this case, the type 3 outputs remain switched off until the emergency stop is rescinded and the control voltage has been switched on again.



Note

Test the desired behavior of the PLC outputs in the event of an emergency stop!

Failure test

For test purposes, the behavior in the event of a failure of the MC main computer can be simulated in order to inspect the correct wiring of the holding brakes. Performing the FAILTEST means:

- There are no more software-controlled actions—neither by the NC nor by the PLC
- In HSCI systems, the control-is-ready output –STO.A.G is reset



Danger

Hanging axes must be supported before the test in order to prevent damage to the machine in case of error.

- ▶ Enter the code number **FAILTEST** under MOD.
- ▶ Acknowledge the message window with the YES soft key in order to carry out the test.

Connection diagram

In the event of an error, a drop-off of the control-is-ready output (-STO.A.G) must trigger an emergency stop. The control therefore checks this output every time that line power is switched on.

If you use safety protection combinations, this may lead to problems if the emergency stop test is also performed with these combinations. The reason for this is the recovery time of such assemblies after an emergency stop. In some cases it is greater than 200 ms.

In order to maintain these specifications, the "Control is ready" output (-STO.A.G) remains off for the time in MP7630 after detecting the 0 level at the "control-is-ready signal acknowledgment" input (-ES.A). This time for step 5 (see flowchart) can be configured in MP7630. This time is to be set so that the safety protection combinations can become ready again.



Note

The circuitry recommended by HEIDENHAIN is illustrated in the Basic Circuit Diagram.

Ensure that the control-is-ready acknowledgment occurs within 1 second.

MP7630 Recovery time after emergency stop test can be configured

Input: 0: 200 ms
1 to 999 [ms]

Emergency stop test

The emergency stop test was expanded for HSCI controls without functional safety (FS). After the emergency stop test or a self-test is started, internal signals of the HSCI participants are now tested for proper function in the first phase. In a further, second phase the emergency stop test and brake test are conducted with external signals (ES.A, ES.B, STO.A.G).

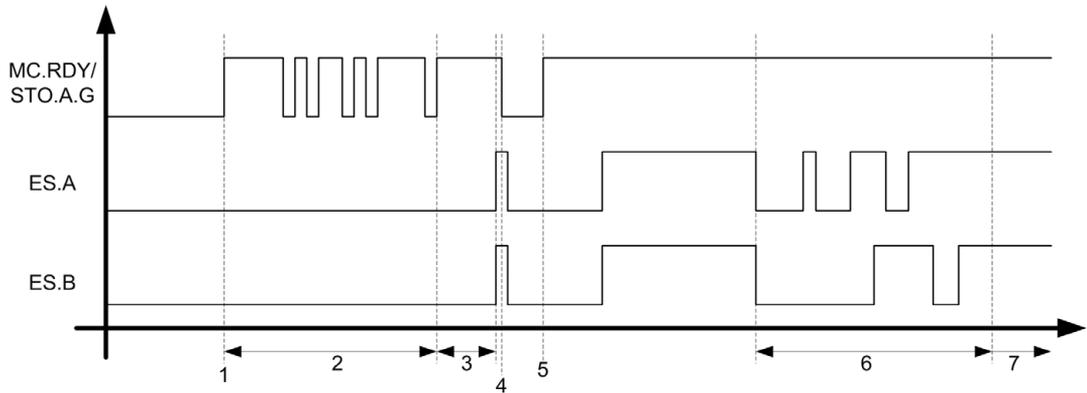
But it still applies, that, in the event of an error, a drop-off of the control-is-ready output (MC.RDY or STO.A.G) always triggers an emergency stop.



Note

The circuitry recommended by HEIDENHAIN is illustrated in the Basic Circuit Diagram.

Time diagram of essential signals after the control is booted and during the self test:



Step	Function	Screen display
1	Start of the self test, immediately after compiling of the PLC program	Pop-up window Self test
2	Phase 1 of the self-test: Triggering and detection of essential internal signals are tested. In this phase of the self-test, the signals STO.A.G (X9/3a) and STOS.A.G (X9/2a) are set and deleted several times.	HSCI components are tested
3	Waiting for machine control voltage	RELAY EXT. DC VOLTAGE MISSING
4	Detection of the control voltage and cut-off of the STO.A.G/STOS.A.G signal by the NC software. ES.A/ES.B must switch to zero within one second.	Emergency stop test
5	Switch-on of the STO.A.G/STOS.A.G signal by the NC software.	
6	Phase 2 of the self-test: Release and detection of the emergency stop signals ES.A and ES.B are tested internally. No hardware terminals are switched!	
7	Normal control operation Control voltage is on, STO.A.G/STOS.A.G output and ES.A/ES.B are at "1".	TRAVERSE REFERENCE POINTS

The following error messages can occur during the test:

■ **Timeout during self-test**

At least one HSCI participant has not answered a request or has not correctly detected a signal condition to be tested. Possible causes:

- HSCI participant/device is defective
- HSCI cabling is faulty
- The hardware components used and/or software are not compatible with each other.

■ **Error during self-test**

The sequence of the individual test steps and processes in the self-test do not fulfill the requirements. A signal condition to be tested is not in the required initial condition. Possible causes:

- HSCI participant/device is defective
- HSCI cabling is faulty
- The hardware components used and/or software are not compatible with each other.

■ **Error in self-test**

Has same causes as **Error during self-test**. However, after the cause of error is corrected (e.g. by closing the guard door), the test can be continued without the control having to restart.

Repetition of the power-up test

One of the requirements of EN ISO 13849-1 for the iTNC 530 HSCI is that the power-up test (emergency-stop test and braking test) must be repeated within no more than 168 hours. It must be ensured by the PLC program that this requirement is met. However, a running NC program can be run to its end even after the 168 hours have passed if the guard doors have been and remain closed the entire time. However, this may only be done if the guard doors remain closed until the power-up test has been repeated! If the guard doors are open and the 168 hours have expired, the PLC program must trigger an error with an emergency stop reaction. According to standards, once the 168 hours have passed it is no longer possible to rule out any faulty behavior of the safety functions.

The emergency stop test can be started directly through the PLC module 9144. The braking test is started through the PLC module 9143. The required timer for monitoring must be realized in the PLC program. When the control is booted after the main switch is on, the control automatically runs the power-up test.

The following PLC markers help you to realize the repeated power-up test:

■ **M4189 – Emergency-stop test completed**

The marker M4189 is set when the emergency-stop test of the control has been completed. The marker is also reset during the execution of PLC module 9144 and is not set again until the end of the emergency stop test.

■ **M4190 – Emergency-stop test active**

The marker M4190 is set when the control is ready for the emergency-stop test after it has been started up. The marker remains set until the emergency-stop test has been completed. During the execution of PLC module 9144 the marker also remains set until the end of the emergency-stop test.

■ **M4191 – Control is ready**

The marker M4191 contains the information of the "control-is-ready" output. This information is given before the hardware output reaches the corresponding physical status (MC.RDY or STO.A.G).

■ **M4192 – Request for "Control Voltage ON"**

The marker M4192 is set in controls without Functional Safety (FS) if there is an NC software request for the control voltage to be switched on. You can then automatically connect the machine control voltage again through a PLC output and corresponding external circuitry.

		Set	Reset
M4189	Emergency stop test completed	NC	NC
M4190	Emergency stop test	NC	NC
M4191	Control is ready	NC	NC
M4192	Request for control voltage ON	NC	NC

Module 9144 Safety self-test / Emergency stop test

PLC module 9144 is used to activate special functions regarding the safety self-test or emergency-stop test, as well as the functional safety (FS) of a HEIDENHAIN control system.

The test can be started directly through the PLC module. Also, a PLC soft key can be made available through the PLC program if all minimum requirements are fulfilled so that the user can start the emergency stop test directly by soft key.

It must be ensured by the PLC program that the following minimum requirements are met before the self-test is started:

- All guard doors must be closed and, if possible, locked.
- No active machining operation is allowed.

Further constraints may apply, depending on the control model used:

■ Safety self-test and emergency-stop test for controls with or without HSCI and without functional safety (FS)

The PLC program decides whether and at what time a repeated test is to be run or suggested after the control has been switched on. The PLC program can use Module 9144 to start the test immediately, taking the minimum requirements (mode 0) into account.

If the PLC program wants to test the brakes beforehand, it must command the brake test through PLC module 9143. Before the repeated self test begins, the PLC program must switch off the drives.

- The marker M4190 is set and the marker M4189 is reset when a test is started.
- The marker M4189 is set and the marker M4190 is reset when the entire test cycle has been completed.
- The marker M4192 is set if there is a request for the machine control voltage to be switched on.

Module 9144 and the PLC markers are used to start the repeated self-test through the PLC and to fully automate the self-test. The test is started by Module 9144 and at first triggers an external emergency stop through the signals MC.RDY and STO.A.G. Then the test continues up to the point at which the machine control voltage must be switched back on. This can be done by the operator on request by the NC, or the procedure can be automated such that marker 4192 is evaluated and the machine control voltage is switched on by the PLC. As soon as the control voltage has been switched back on, the self-test is continued up to completion.

Call:
 PS K/B/W/D <Mode>
 0: Start self test immediately
 1: Reserved
 10: Define the operating mode for functional safety
 11: Request for testing the axis position

PS K/B/W/D <Parameter 1>
 Mode 0: No evaluation, must be programmed
 Mode 1: No evaluation, must be programmed
 Modus 10:
 0: Operation through machine operating panel
 1: Operation through manual control unit
 2: Homing and testing of axes
 Mode 11: Number of axis to be tested

PS K/B/W/D <Parameter 2>
 Mode 0: No evaluation, must be programmed
 Mode 1: No evaluation, must be programmed
 Mode 10: No evaluation, must be programmed
 Mode 11: No evaluation, must be programmed

CM 9144
 PL B/W/D <Status/Error>
 Mode 0:
 0: Function is being executed
 1: Error according to W1022

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Invalid value programmed for mode or parameter
	28	Test already active
	43	This is not an HSCI system or a system with functional safety
	51	Function is not supported by this control
	58	Control without operating-mode group
	61	Function is not supported by this control



Module 9143 Activate the brake test

With this module an axis-specific brake test with the configuration from the machine parameters or with other values for MPs 2230 and 2232 can be started.

Constraints:

- Programming in a submit job blocks other submit jobs until the test is completed.
- The PLC module automatically passes the processing time to other spawn and submit processes.

Call:

PS K/B/W/D <Axis number>
0 = 1st axis, 2 = 2nd axis, etc.

PS K/B/W/D <Multiplier for rated current>
Value in 1/1000 or
0: Default MP2230 (factor of nominal current)

PS K/B/W/D <Permissible traverse path>
Value in 0.1 [um] or
0: Default MP2232

CM 9143

PL B/W/D <Status/Error>
0: Brake OK
1: Brake defective
2: Invalid axis or negative values for rated current or traverse path
3: Call during running NC program or during other PLC jobs
4: Call was made from a cyclic PLC program
5: Error during data exchange
6: Not allowed for safe control
7: Drive not ready
8: Brake test was aborted (e.g. by emergency stop)

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Invalid axis programmed (invalid axis number, not a closed-loop axis, axis currently open-loop axis or slave axis) or negative values for the traverse path or current are programmed
	8	Module is not allowed for control with functional safety
	20	Module was not called in a submit or spawn job
	21	Call during program run or during other active PLC jobs for the programmed axis
	40	Drive not ready
	45	Canceled due to error during data exchange or due to external influences (e.g. emergency stop)



6.11.15 EN 13849-1 on the iTNC 530 HSCI without functional safety (FS)

The safety functions provided for you vary depending on the hardware and software installed in the various HEIDENHAIN control systems.

On the iTNC 530 HSCI without functional safety (FS), no integrated safety functions, such as safe operating modes or safely limited speed, are available. The entire realization of these functions according to EN 13849-1 requires external safety components. TÜV Süd only inspected this control in regard of the safe switch-off of the drives.

The HEIDENHAIN safety system for the iTNC 530 HSCI without functional safety (FS) has a dual-channel structure. Safety contactor combinations (SCC) are central to the measurement and evaluation of all safety-relevant assemblies (e.g. keylock switch, emergency stop button). The SCCs generate output signals that are received as input signals by the two safety channels. The safety functions are realized over dual-channel enabling of the function.

One channel is realized through external wiring with the SCCs to the enable relays in the inverters. The second channel includes the control, which consists of the main computer (MC) and one or more drive controller modules (CC= controller computer). Safety-relevant errors always lead to safe stopping of all drives through defined stop reactions.

The basic circuit diagram of the respective control serves as the basis for this dual-channel design of cut-off paths. This is a non-binding proposal from HEIDENHAIN, and must be adapted to the requirements of the machine used. As a machine manufacturer, you are responsible for adhering to the relevant standards and safety regulations.

The documentation on this topic consists of a system description with application conditions from HEIDENHAIN. The requirements stated in this document must be adhered to; the document is available from your HEIDENHAIN contact person upon request.

The software used in the individual controls is not relevant for the dual-channel cut-off paths and was therefore not examined by TÜV Süd. Only the hardware, the inverters, and the external safety components used are decisive for the evaluation of the cut-off paths.

In addition, the control-is-ready output of the iTNC 530 HSCI has only a single-channel design and is therefore not a "safe" output. The control-is-ready output in the HEIDENHAIN controls, however, serves as single-channel acknowledgment signal to the SCCs and must be wired according to the basic circuit diagrams.

6.12 Spindle

Two spindles can be controlled alternately (see "Operating a second spindle" on page 985).

The main spindle/spindles are controlled with the PLC.

The programmed speed can be output as

- Code via PLC outputs
- Analog nominal speed command signal for an analog spindle
- Digital nominal speed value for a digital spindle

The spindle functions are of varying priorities. If several functions are output at the same time, the function with the highest priority is run, and the rest are deleted. The following spindle function priorities are valid:

First: Spindle orientation

Second: Transmission jog

Third: M3/M4

Fourth: M5

► Specify in MP3010 the speed output for the spindle.

MP3010 Output of speed, gear range

Input:	0:	No output of spindle speed
	1:	Speed code, if the speed changes
	2:	Speed code at every TOOL CALL
	3:	Nominal speed value always, G code if the gear shifts
	4:	Nominal speed value always, G code at every TOOL CALL
	5:	Nominal speed value always, no G code
	6:	Same as 3, but with servo-controlled spindle for oriented spindle stop
	7:	Same as 4, but with servo-controlled spindle for oriented spindle stop
	8:	Same as 5, but with servo-controlled spindle for oriented spindle stop

6.12.1 Position encoder of the spindle

Analog and digital spindles can be driven in a closed control loop. In this case the spindle needs its own position encoder:

- ▶ Define the position encoder input in MP111.x.
 - If you have a digital spindle and would like to use the speed encoder also as a position encoder, then you must set $MP111.x = 0$.
- ▶ Enter in MP3142 the line count of the encoder to be used. 1 V_{PP} signals undergo 1024-fold subdivision
- ▶ Enter in MP3142 how the position encoder is mounted on the spindle. Due to the higher required accuracy, the position encoder must be mounted directly on the spindle: **MP3143 = 0**

If design considerations make this impossible:

- ▶ Define the encoder-to-spindle transmission ratio in MP3450.x and MP3451.x for each gear stage.

In this case there will be several reference pulses per revolution. For example, with a transmission of 4:1 (motor to spindle), you will receive four reference pulses (every 90°) per spindle revolution.

- ▶ Evaluate the reference mark with Module 9220. "Renewed traversing of the reference marks" on page 804.

If **MP3143 = 1**, then X9 pin 14b is evaluated as the reference signal. If **MP3143 = 3**, then the **second** signal at X9 pin 14b is evaluated as the reference signal. This can be necessary if the spindle is located shortly before the switch for the reference signal. This might not be detected, and so only the next signal can be evaluated. If **MP3143 = 3**, the spindle performs two revolutions, and therefore the second signal at X9 pin 14b is used as the reference signal. In both cases the reference mark of the position and speed encoder is not evaluated. In this case the reference signal is generated externally and **must** be evaluated with Module 9220, see "Renewed traversing of the reference marks" on page 804.

During referencing of the spindle (with PLC Module 9220, special sequence with external sensor X9 pin 14b), as of software version 606 42x-02 the specified spindle braking ramp is used for braking when the reference mark or the external switch is reached. Up to now the spindle used to be stopped on the emergency stop ramp. The ramp gradient of the spindle with M03 and M04 for gear ranges 1 to 8 is defined in MP3411.0-7.



Attention

Due to its low accuracy, this solution is not recommended.

If **MP3143 = 2**, then a signal at X9 pin 14b is used to enable the reference pulse for the spindle position encoder. This ensures that the same reference mark of the encoder is always evaluated. As soon as a reference mark of the encoder is traversed, the pulses are counted and immediately after the switch is scanned the "correct" reference mark is calculated. This might mean that the spindle is no longer moved after the switch is scanned. The signal must be available for at least as long as the position controller cycle time + 2 ms. This means that the maximum speed for detecting the signal is:

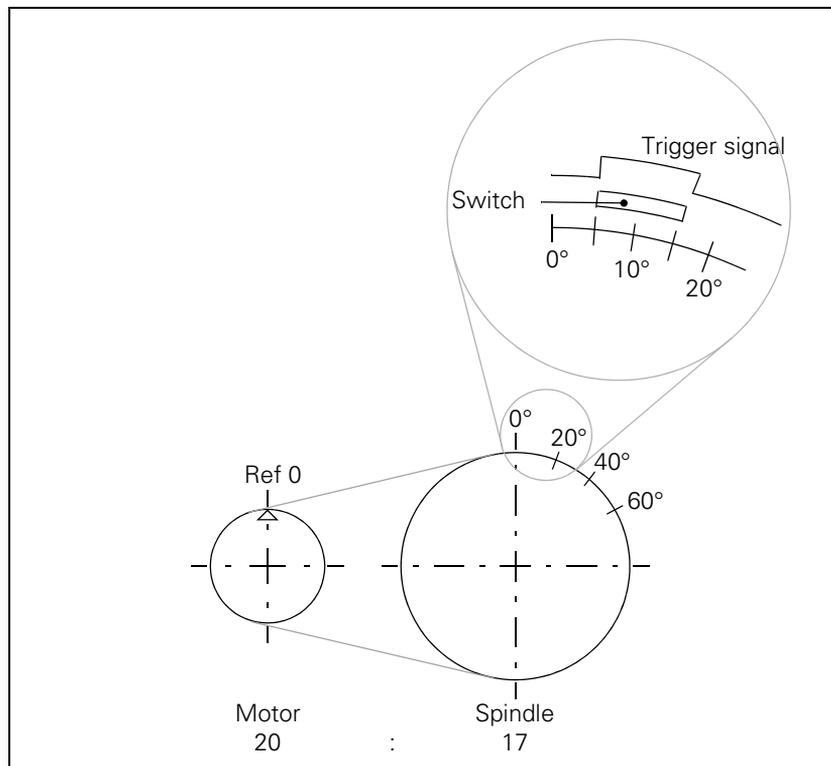
$$n_{\max} = \frac{s \cdot 60000}{360 \cdot (t_{\text{Pos.}} + 2 \text{ ms})}$$

s: Trigger window of the switch (10° in the example)

$t_{\text{Pos.}}$: Position controller cycle time

The example results in a maximum speed of:

$$n_{\max} = \frac{10^\circ \cdot 60000 \text{ ms/min}}{360^\circ \cdot (3 \text{ ms} + 2 \text{ ms})} = 333 \text{ min}^{-1}$$





Note

The switch for enabling of the reference mark signal must be adjusted precisely enough for the "correct" reference mark to be evaluated!

The transmission ratio (MP3450.x, MP3451.x) is evaluated only if the value in MP3143.x is > 0 .

Two additional input values 4 and 5 have been introduced for the machine parameter MP3143 for the type of mounting of the position encoder of the spindle:



MP111 Position encoder input for the spindles

Input: 0: No position encoder
201 to 206: Position encoder inputs X201 to X206

MP111.0 Position encoder input for the first spindle
MP111.1 Position encoder input for the second spindle

MP3142 Line count of the spindle position encoder

Input: 100 to 100 000 [lines]

MP3143 Mounting configuration of the spindle position encoder

Input: 0: Position encoder immediately on the first spindle
1: Position encoder via transmission (ratio in MP3450.x and MP3451.x); or X9 pin 14b : reference pulse
2: Position encoder via transmission (ratio in MP3450 and MP3451); or X9 pin 14b: reference pulse release
3: Same as input value 1, except that the second reference pulse is evaluated.
4: Reference-mark evaluation of the spindle via EnDat. The encoder must be mounted directly (without transmission). No reference pulse is necessary. A new reference-mark evaluation via marker M4015 may only be performed at standstill.
5: The position encoder is mounted directly (same as input value 0) and the transmission ratio is also evaluated. Entering the value 5 results in the same behavior as that of software version 340 49x-05 (rigid tapping will also be possible without a position encoder on the spindle)!

MP3450.0-7 Number of spindle position-encoder revolutions for gear ranges 1 to 8

Input: 0 to 65 535
0: No transmission

MP3451.0-7 Number of spindle position-encoder revolutions for gear ranges 1 to 8

Input: 0 to 65 535
0: No transmission



Module 9042 Read the spindle coordinates (format 0.001°)

The following coordinate values are saved in five successive double words beginning with the specified target address:

- Actual value
- Nominal value
- Actual value in reference system
- Following error (servo lag)
- Distance-to-go

The values for actual, nominal, and reference value are standardized at 0° to +360.000°.

The values for servo lag and distance-to-go are displayed between -2879.912° and +2879.912°. Format: 0.001°.

If MP3010 < 6 (no closed-loop spindle), then all coordinates are read as zero.

During operation under open-loop control (M03 / M04 active or M05 and open position control loop), the nominal value is considered to be the actual value. The servo lag and distance-to-go are considered to be zero.

Call:

PS B/W/D/K <Target address Dxxxx>

CM 9042

Error code:

Marker	Value	Meaning
M4203	0	Actual speed value was read
	1	Target address is too large or is not a double-word address

Module 9044 Read the spindle coordinates (format 0.0001°)

Call:

SEE MODULE 9042

6.12.2 Speed encoder of the spindle

Digital speed control requires a shaft speed encoder:

- ▶ Define the speed encoder input in MP113.x. MP121.x is used to assign the outputs to the spindles with the CC 61xx controllers.

The iTNC 530 monitors the reference mark of the speed encoder. The monitor checks whether the line count for one revolution from reference mark to reference mark is equal to the line-count entry in the motor table.

If differences occur, the DSP error message **C3A0 Incorrect reference position S** appears. If this happens, check the speed encoder, encoder cable, and whether you have selected the correct motor.

With a gear wheel encoder, even if it is properly installed, monitoring can result in this error message due to its inherent inaccuracy:

- ▶ In this case, switch the monitoring off with MP2220 bit 0 = 1.

The iTNC 530 monitors the direction of rotation. If the nominal value of current exceeds the limit value for a certain time, the DSP error message **C380 Motor <spindle 1/2, axis> not controllable** appears.

In MP113.0 and MP 113.1 you can set different inputs for the speed encoder of the spindle, even if the PWM output for the spindle drive stays the same. The PWM output and the two encoder inputs must be on the same drive-control motherboard (DSP) of a CC, however. This means that they must be configured on CC connectors with the same letter A or B.

This may be necessary for applications in which you switch between two spindles or spindle motors that are operated with the same UM power module.

At lower speeds, high-frequency spindles only have a low amount of torque. If such a spindle is having its speed controlled, the tool changer may slightly twist the spindle, causing the limit of current to be exceeded. This leads to the above error message:

- ▶ In this case, switch the monitoring off with MP2221 bit 1 = 1.



Attention

For axes, monitoring of the rotational direction (MP2220 bit 1) must **not** be deactivated. An error (e.g. one motor phase interchanged with another or incorrect entry in the **DIR** column of the motor table) might cause uncontrolled acceleration of the motor in one direction if the monitoring function for the rotational direction is deactivated.

This also applies to spindles. For spindles, however, an incorrect acceleration in one direction is less dangerous than for axes.

Monitoring of the direction of rotation (MP2220 bit 1) for synchronous motors (entry **SM** in the column **TYPE** in the motor table) cannot be switched off.



MP113

Input:

MP113.0

MP113.1

Speed encoder for the spindle/spindles

0: No speed encoder

15 to 20: Speed encoder inputs X15 to X20

Speed encoder for the first spindle

Speed encoder for the second spindle

MP2220

Input:

Monitoring functions

Bit 0 – Monitoring the reference mark

0: Monitoring active

1: Monitoring inactive

Bit 1 – Monitoring the rotational direction

0: Monitoring active

1: Monitoring inactive

6.12.3 Analog and digital closed-loop spindle control

For both analog and digital output of the nominal speed command you can program speeds of from 0 to 99 999.999 min⁻¹.

The maximum controllable spindle speed is: $\frac{80\,000}{\text{No. of pole pair}}$

If the load increases, the spindle speed is corrected until the maximum current is attained. If the load continues to increase in spite of the maximum current, the spindle speed is reduced. For the maximum current, the value from either the motor table or the power-module table of the drive (whichever is lower) applies.

If in MP3010 you have selected the output of the nominal speed value, M4003 is set. The programmed speed is saved in D356, the nominal speed value in W320 and the actual speed value in W322. In addition, the nominal speed value is saved in D364 and the actual speed value in D368, since speeds above 32 767 min⁻¹ cannot be represented in words W320 and W322.

The current spindle speed is already available in the PLC word W322 or D368 before the control voltage is switched on. This makes it possible to identify a spindle coasting to a stop after a restart.

The PLC double word D372 makes the maximum spindle speed including the spindle potentiometer available to the PLC program. This makes it possible, for example, to already acknowledge within the ramp the M functions for switching on the spindle by comparing the actual and maximum speeds.

With D604 you can limit the possible spindle speed through the PLC. To ensure compatibility, D604 is preassigned with 99 999 999 after control switch-on or after an interruption in the PLC scan.

Analog spindles:

The nominal speed value of the motor is output as an analog dc voltage of ±10 V at connection X8 or X9.

Digital spindles:

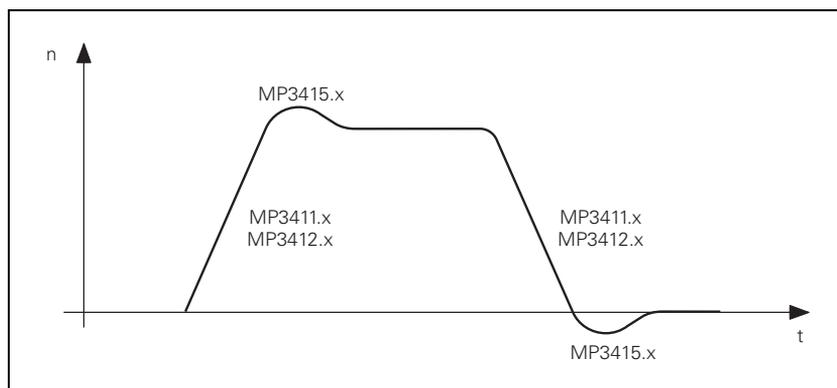
The nominal speed value is transferred to the internal speed controller.

		Set	Reset
M4003	Nominal speed value output analog or digital (MP3010 = 3 to 8)	NC	NC
D356	Programmed speed [0.001 min⁻¹]	NC	NC
D364	Nominal speed value [min⁻¹]	NC	NC
W320	Nominal speed value [min⁻¹]	NC	NC
D368	Actual speed value [min⁻¹]	NC	NC
D372	Maximum spindle speed including spindle override [min⁻¹]	NC	NC
W322	Actual speed value [min⁻¹]	NC	NC
D604	Maximum possible spindle speed	PLC	NC/PLC



Nominal speed value

- ▶ In MP3411.x, define the ramp gradient for the nominal speed value at M03 and M04 for each gear range.
- ▶ With MP3412.0, specify a multiplication factor for MP3411.x, for
 - M05 (MP3412.0)
 - SPINDLE ORIENTATION (MP3412.1)
 - TAPPING (with floating tap holder) (MP3412.2)
 - RIGID TAPPING (without floating tap holder) (MP3412.3)The same factor applies for all gear ranges.
- ▶ Set MP3411 for M03, M04 and M05 such that the motor accelerates and brakes within the current limit.
- ▶ With MP3415, define the overshoot behavior for every operating mode when the spindle is switched on with M4011. Set MP3415.0 so that only one overshoot is visible.



If the nominal speed value is in the acceleration or deceleration ramp, then M4001 is reset. This also applies if the speed is changed with the override potentiometer.

If the nominal speed value is output as zero, M4002 is set.

In machine parameter MP7440 bit 7 = 1 you can define for the M function M0 that the internal geometry status is not to change to M05. The status M03/M04 is retained in the geometry. The spindle status in Q110 is also not influenced by M0 if bit 7 is set. Thus the processing of cycles does not result in the error message "Spindle ?" after an NC start.



Danger

MP7440 bit 7 may only be set if after an M00, an NC start is absolutely required by the PLC program to start the spindle again. Otherwise it is possible that the spindle starts when the user does not expect it.

MP3411.0-7 Ramp gradient of the spindle with M03 and M04 for gear ranges 1 to 8

Input: Analog axes: 0 to 1.999 [V/ms]
Digital axes: 0 to 1.999 [(1000 min⁻¹)/ms]

MP3415 Overshoot behavior of the spindle with M03, M04 and M05

Input: 0 to 1000 [ms]
MP3415.0 With M03, M04 and M05
MP3415.1 For spindle orientation
MP3415.2 For tapping
MP3415.3 For rigid tapping

MP3412 Multiplication factor for MP3411.x

Input: 0.000 to 1.999
MP3412.0 With M05
MP3412.1 With spindle orientation
MP3412.2 For tapping with floating tap holder
MP3412.3 For rigid tapping

MP7440 Output of M functions

Input: Bit 7 – Internal spindle status of the geometry
0: With M00 the internal spindle status switches to M05
1: With M00 the internal spindle status does not switch to M05

		Set	Reset
M4001	Nominal speed command signal of the spindle not in the ramp	NC	NC
M4002	Nominal speed value = 0	NC	NC

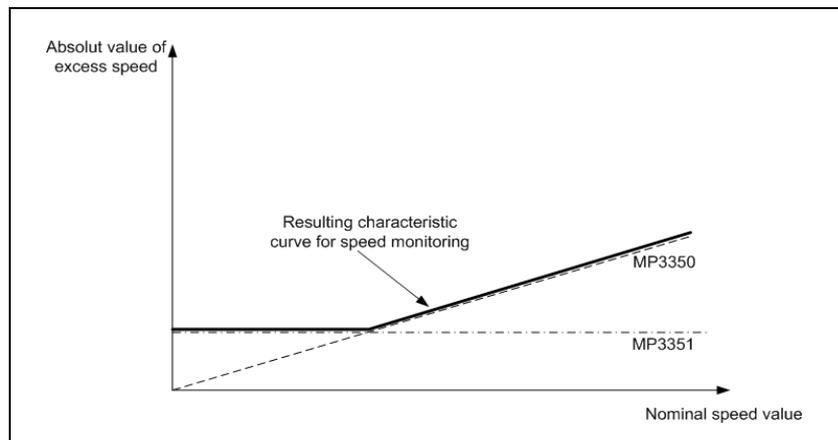


Monitoring the spindle speed

With MP3350 and MP3351 you set the maximum permitted excessive spindle speed. An NC stop with a subsequent emergency stop is triggered if actual spindle speed > than the nominal spindle speed + the permitted excess speed. With this function you can also have the spindle be monitored for unexpected start-up.

In MP3350 you enter a relative value in percent, and in MP3351 an absolute value, for the permissible excessive speed. The absolute value in MP3351 is only used if the absolute value that results from MP3350 is less than the value in MP3351.

If you want to use monitoring of the spindle speed for spindles with gear stages, then you should be more generous with the tolerance that results from MP3350 and MP3351, since the spindle speed can vary significantly when switching between the gear stages.



A percentage value for the permissible shortfall of the nominal spindle speed can be entered in MP3540. You can thus use this machine parameter for monitoring whether the spindle speeds falls below a specific spindle speed limit. (e.g. nominal spindle speed = 500 rpm, MP3540 = 20 % → the monitoring function will respond if the spindle speed falls below 400 rpm). Monitoring is useful for power milling where the workpiece is to be machined at a high feed rate and a low constant spindle speed. If the monitoring function responds, an EMERGENCY STOP reaction is triggered in order to protect the machine from possible damage, for example if the spindle comes to a standstill while still engaged in the material. The error message "8BF0 Actual speed value too small" is displayed. Monitoring does not become active until the spindle speed exceeds the value defined in MP 3542.

Monitoring for falling below the lower spindle speed limit (defined in MP3540) does not become active until the actual spindle speed exceeds the value defined in MP3542. This threshold is required in order to avoid activating the monitoring function before a reasonable shaft speed is reached (for example, if the noise of encoders at the actual value of the spindle speed is of subordinate importance, or in order to prevent monitoring from becoming active when the tool plunges into the material).

In an EMERGENCY STOP, braking of the spindle is delayed compared to braking of the axes by the time in MP3550. This avoids tool breakage during power milling, because low spindle speeds might result in the spindle coming to a standstill before the axes do. However, the spindle braking is delayed by no more than half a spindle revolution.

MP3350 Maximum permissible overshoot of the spindle speed in percent of the nominal spindle speed

Input: 0 to 100 [%]

MP3351 Entry of an absolute value for the permissible overshoot of the spindle speed

Input: 0.001 to 99 999.999 [rpm]
0: Monitoring off

MP3540 Permissible spindle speed shortfall

Input: 1 to 99 [%]
0 = Monitoring not active

MP3542 Minimum spindle speed as of which the monitoring in MP3540 becomes active

Input: 0 to 100 000 [rpm]

MP3550 Delay of emergency-stop reaction of spindles

Input: 0.001 to 0.100 [s]
0 = Delay not active

Increased spindle power for roughing

With MP3530 and MP13530 you can avoid a reduction in the maximum spindle power during roughing due to changes in load when the tool's teeth are engaged in the material.

Here the spindle speed is increased higher than previously when there is no load. It is now increased above the nominal speed, which was not the case previously. Energy is stored in the inertia of the spindle, and is then available when the teeth engage the material. This increases the mean power, and the mean speed corresponds to the nominal speed.

If MP3530 or MP13530 are set, an overshoot can occur when the nominal speed is reached. If this overshoot impairs the performance, then you may have to deactivate the increase of the spindle power.

MP3530 Increased spindle power for roughing

Input: 0 = Not active
1 = Increased spindle power for roughing

MP13530 Increased spindle power for roughing (second spindle)

Input: 0 = Not active
1 = Increased spindle power for roughing



Direction of spindle rotation

- ▶ With MP3130, define the polarity of the nominal speed value
- ▶ In MP3140, enter the counting direction of the position encoder signals.

As soon as you set M4005 for M03, or M4006 for M04, the nominal speed value is output. With M4007 for M05, the nominal speed value zero is output (spindle stop).

M4005 to M4007 also controls the miscellaneous functions in the status window.

If more than one marker is set at the same time, the error message **PLC: M4005, M4006, M4007 INCORRECT** appears.

With M4014 you can reverse the direction of rotation, in order to adjust the transmission for horizontal or vertical spindles, for example. The polarity of the nominal spindle speed is inverted.

M4019 reverses the counting direction of the spindle.

MP3130 Polarity of the nominal spindle speed

Input: 0: M03 positive, M04 negative
1: M03 negative, M04 positive
2: M03 and M04 positive
4: M03 and M04 negative

MP3140 Counting direction of spindle position encoder output signals

Input: 0: Positive counting direction with M03
1: Negative counting direction with M03

		Set	Reset
M4005	Status display and nominal speed value output for M03	PLC	PLC
M4006	Status display and nominal speed value output for M04	PLC	PLC
M4007	Status display M05 and spindle stop	PLC	PLC
M4014	Reverse the direction of spindle rotation	PLC	PLC
M4019	Reverse the counting direction of the position encoder on the spindle	PLC	PLC

Disable speed output for spindle

With M4008 you can block the speed output for the spindle. At the same time, M03, M04 or M05 are highlighted. The nominal speed value is zero.

		Set	Reset
M4008	Disable speed output for spindle	PLC	PLC

Gear ranges

You can define up to eight gear ranges:

- ▶ In MP3510.x, enter for each gear range the rated speed for "S-override 100%." Enter the value zero for unnecessary gear ranges.
- ▶ In MP3210.x, enter for every gear range the S analog voltage or motor revolutions at rated speed.
- ▶ In MP3240.1, define the minimum nominal speed value for the motor.
- ▶ In MP3120, define whether zero is permitted as a programmed speed.

If an impermissible speed is programmed, M4004 is set and the error message **WRONG RPM** is displayed.



Note

The gear range from W256 is output when the spindle speed is 0.

MP3510.0-7 Rated speed for the gear ranges 1 to 8

Input: 0 to 99 999.999 [min^{-1}]

MP3210.0-7 Analog nominal spindle voltage at rated speed for the gear ranges 1 to 8

Input: 0 to 100.000 [V]

MP3210.0-7 Digital spindle motor revolutions at rated speed for the gear ranges 1 to 8

Input: 0 to 100.000 [1000 min^{-1}]

MP3240.1 Analog spindle: Minimum nominal value voltage

Input: 0 to 9.999 [V]

MP3240.1 Digital spindle: Minimum motor speed

Input: 0 to 9.999 [1000 min^{-1}]

MP3120 Zero speed permitted

Input: 0: S = 0 permitted
1: S = 0 not permitted

		Set	Reset
M4004	Impermissible speed was programmed	NC	NC

Gear shifting

You control the gear shifting through PLC outputs. The NC enters the current gear range according to the programmed speed in W256. The gear range is calculated with MP3510.x. The output of the gear range is defined in MP3010. MP3030 bit 1 determines whether the speed should be reduced to 0 when shifting between gears.



Attention

- On machines with transmission or wye/delta switchover, the RPM FACTOR (Q403) function of Cycle 209 (Tapping with chip breaking) must not be used with a value other than the default value. Switching the gear range or wye/delta switchover during retraction with the RPM FACTOR function may cause damage to the machine.
- Please inform the customers who are using these types of machines that the RPM FACTOR (Q403) function of Cycle 209 (Tapping with chip breaking) must not be used with a value other than the default value.

When the gear range is changed, the NC sets the G strobe (M4070). As soon as you confirm the gear shift with M4090, the program resumes and the G strobe (M4070) is reset by the NC.

If a TOOL CALL block is followed by the output of a T strobe and G strobe, then M4547 is set by the output of the T strobe and reset by output of the G strobe. If there is no output of either the T or G strobe, M4547 is not set.

In the PLC program you can change the programmed speed and the gear range that is calculated by the NC. The programmed speed is saved by the NC in D356 and D756. The following procedure is fundamentally possible, but not in connection with mid-program startup.

- ▶ Enter a speed in D756 and a gear range in W256. The speed must lie within the speed range of the gear.
- ▶ With M4134, activate your entries in D756 and W256.
- ▶ After the NC has reset M4134, change the gear and report with M4090 that the gear shift has been completed.

So that no problems arise during mid-program startup, or when horizontal/vertical spindles are in use, gear shifting should be realized for the spindle via various sets of parameters.

A changing nominal speed value ramp can be output to shift gears by alternately setting and resetting M4009 and M4010. This can be realized by interrogating the timers in the PLC program. This function also works if you have used M4008 to disable the speed output for the spindle:

- ▶ In MP3240.2, define the nominal speed value that is output with M4009/M4010 to the spindle motor.

MP3030 Behavior of the spindle

Input: Bit 1 – Zero spindle speed when shifting to another gear range
 0: Reduce speed to 0
 1: Do not reduce speed to 0

MP3240.2 Analog spindle: Spindle jog voltage for gear shifting (M4009/M4010)

Input: 0 to 9.999 [V]

MP3240.2 Digital spindle: Motor speed for gear shifting (M4009/M4010)

Input: 0 to 9.999 [1000 min⁻¹]

		Set	Reset
W256	Gear code	NC/PLC	NC/PLC
D356	Programmed speed [0.001 min⁻¹]	NC	NC
D756	Programmed speed or speed from PLC [0.001 min⁻¹]	NC/PLC	NC/PLC
M4009	Counterclockwise spindle rotation (for gear change)	PLC	PLC
M4010	Clockwise spindle rotation (for gear change)	PLC	PLC
M4070	Strobe signal for gear code	NC	NC
M4090	Acknowledgment of "gear change completed"	PLC	PLC
M4134	Activation of a gear range and speed through the PLC	PLC	NC
M4547	T and G strobes with TOOL CALL	NC	NC



Spindle override

You can change the spindle speed within certain limits with the spindle override potentiometer.

- ▶ Define the limits in MP3310.x.
- ▶ In MP3515.x, enter for every gear range a maximum attainable speed which must not be exceeded with the spindle override.

The percentage adjusted with the spindle override is entered by the NC in W492 and W764. You can change the percentage through the PLC:

- ▶ Enter the desired percentage in W764.
The NC adopts the new value as soon as it is entered.

Machine parameter MP7620 bit 3 and bit 4 is used to configure the characteristic curve and the behavior of the spindle-speed override and feed-rate override. If bit 3 is not set (= 0), the feed-rate and spindle potentiometers are now available as a linear characteristic curve. On TE keyboard units without rapid-traverse override, bit 4 serves to additionally select the resolution used for the evaluation of the overrides.

Behavior of MP7620 bit 3 on TEs without rapid-traverse override:

0: Linear characteristic curve for spindle override and feed rate override; unit depends on bit 4 0..150 or 0..15000

1: Non-linear characteristic curve for spindle override and feed rate override, unit 0..15000

Behavior of MP7620 bit 4 on TEs without rapid-traverse override:

0: Spindle override and feed rate override with unit 0 to 150

1: Spindle override and feed rate override with unit 0..15000

Behavior of MP7620 bit 3 on TEs with rapid-traverse override:

0: Linear characteristic curve for spindle-speed and feed-rate override, unit 0..15000

1: Non-linear characteristic curve for spindle override and feed rate override, unit 0..15000

Behavior of MP7620 bit 4 on TEs with rapid-traverse override:

No effect

The spindle override functions either in 1% steps or according to a nonlinear characteristic curve:

- ▶ With MP7620, bit 3, select the mode of the override.

Value range in W492 and W764:

- 1% steps: 1 to 150
- Nonlinear characteristic curve: 0 to 15 000
In the lowest range, 0.01% steps are available. Beginning with a value of 2.5%, the step is 0.75%

If bit 5 of MP7620 is set, the spindle override potentiometer is used as rapid traverse override potentiometer. The value set with the potentiometer will then be entered by the NC in words W496 and W752 (if bit 5 = 1). The value in word W752 can be changed by the OEM and will be used by the NC. The entry in MP3310 for limiting the spindle override also applies for the rapid traverse override. The value range for W496 and W752 is independent of the setting in MP 7620 bit 3 for the "nonlinear curve." The potentiometer designation on the keyboard unit or the machine operating panel must match the setting in MP7620 bit 5 (adhesive plate for rapid traverse override ID 350 430-02)!

The OEM must then use vacant keys on the machine operating panel for the spindle override, for example. The OEM can specify a value for the spindle override in word W764 of the PLC. The value is used by the NC.



Danger

Only use MP7620 bit 5 in accordance with the keyboard unit or the machine operating panel used. The potentiometer designation must match the function you set in MP7620 bit 5.

An incorrect or missing identification of the potentiometers can lead to errors during operation and even to property damage or personal injury!

If bit 8 of MP7620 is set, the error message "Override potentiometer" will be displayed if the feed-rate or rapid-traverse override is set to 0% at the start of an axis movement and the axis can therefore not be moved.

MP3310.0-1 Limit for spindle override

Input: 0 to 150 [%]

MP3310.0 Upper limit

MP3310.1 Lower limit

MP3515.0-7 Maximum spindle speed for gear ranges 1 to 8

Input: 0 to 99 999.999 [min⁻¹]

MP7620**Feed-rate override and spindle speed override**

Input:

%xxxxxxx

Bit 3 – for TEs without rapid traverse override:

0: Linear characteristic curve for spindle override and feed rate override; unit depends on bit 4 0..150 or 0..15000

1: Non-linear characteristic curve for spindle override and feed rate override, unit 0..15000

Bit 4 – for TEs without rapid traverse override:

0: Spindle override and feed rate override with unit 0 to 150

1: Spindle override and feed rate override with unit 0..15000

Bit 3 – for TEs with rapid traverse override:

0: Linear characteristic curve for spindle-speed and feed-rate override, unit 0..15000

1: Non-linear characteristic curve for spindle override and feed rate override, unit 0..15000

Bit 4 – for TEs with rapid traverse override:

No effect

Bit 5 – Rapid traverse override instead of spindle override

0: Potentiometer is used for spindle override

1: Potentiometer is used for rapid-traverse override

Bit 8 – Informational text if feed-rate or rapid-traverse override is set to 0 %

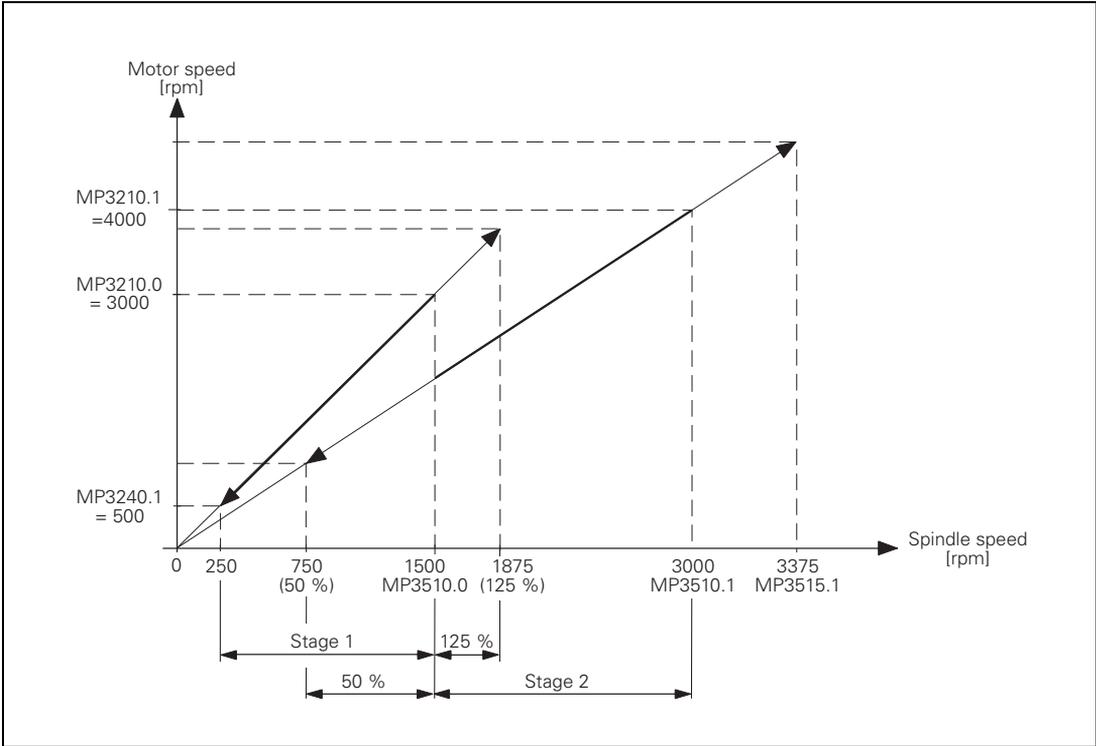
0: Informational text inactive

1: Informational text active

		Set	Reset
W492	Percentage for spindle override (NC to PLC)	NC	NC
W764	Percentage for spindle override (PLC to NC)	NC/PLC	NC/PLC

Example: Two gear ranges for a digital spindle

- Gear range I: Spindle 1500 min⁻¹ with motor 3000 min⁻¹
(MP3210.0 = 3000; MP3510.0 = 1500)
- Gear range II: Spindle 3000 min⁻¹ with motor 4000 min⁻¹
(MP3210.1 = 4000; MP3510.1 = 3000)
- Upper limit for spindle override : 125% (MP3310.0 = 125)
- Lower limit for spindle override: 50% (MP3310.1 = 50)
- Maximum possible output speed for gear range II: 3375 min⁻¹
(MP3515.1 = 3375)
- Minimum motor speed: 500 min⁻¹ (MP3240.1 = 500)



6.12.4 Coded output of spindle speed

If you have selected speed-code output in MP3010 (entry 1 or 2), an S code is entered in W258. You must output the speed code to the spindle drive through PLC outputs.

If the speed code is changed, the NC sets the S strobe (M4071). If you acknowledge the S code with M4091, the NC program is continued and the S strobe (M4071) is reset by the NC.

If required, the programmed spindle speed is rounded off to the next standard value by the NC and given in S code as per ISO 6983 (see S-code table). Speeds of 0 to 9000 min⁻¹ are possible:

- ▶ Specify in MP3010 the speed range and the speed increment. The S code for the minimum speed is stored in W1008.

Example:

Minimum speed = 1 min⁻¹ (S code 20)
Maximum speed = 1000 min⁻¹ (S code 80)
Speed increment = 2:

MP3020 = 20802
W1008 = 20

MP3020 Speed range for S code output

Format: xxyz
 xx: S code for minimum speed
 yy: S code for maximum speed
 z: Speed increment
Input: 0 to 99 999

		Set	Reset
W258	S code	NC	NC
M4071	Strobe signal for S code	NC	NC
M4091	Acknowledgment of S code	PLC	PLC
W1008	S code for minimum speed	NC	NC

S code table

S code	min ⁻¹
S 00	0
S 01	0.112
S 02	0.125
S 03	0.14
S 04	0.16
S 05	0.18
S 06	0.2
S 07	0.224
S 08	0.25
S 09	0.28
S 10	0.315
S 11	0.355
S 12	0.4
S 13	0.45
S 14	0.5
S 15	0.56
S 16	0.63
S 17	0.71
S 18	0.8
S 19	0.9
S 20	1
S 21	1.12
S 22	1.25
S 23	1.4
S 24	1.6
S 25	1.8
S 26	2
S 27	2.24
S 28	2.5
S 29	2.8
S 30	3.15
S 31	3.55
S 32	4
S 33	4.5
S 34	5
S 35	5.6
S 36	6.3
S 37	7.1
S 38	8
S 39	9
S 40	10

S code	min ⁻¹
S 41	11.2
S 42	12.5
S 43	14
S 44	16
S 45	18
S 46	20
S 47	22.4
S 48	25
S 49	28
S 50	31.5
S 51	35.5
S 52	40
S 53	45
S 54	50
S 55	56
S 56	63
S 57	71
S 58	80
S 59	90
S 60	100
S 61	112
S 62	125
S 63	140
S 64	160
S 65	180
S 66	200
S 67	224
S 68	250
S 69	280
S 70	315
S 71	355
S 72	400
S 73	450
S 74	500
S 75	560
S 76	630
S 77	710
S 78	800
S 79	900
S 80	1000
S 81	1120

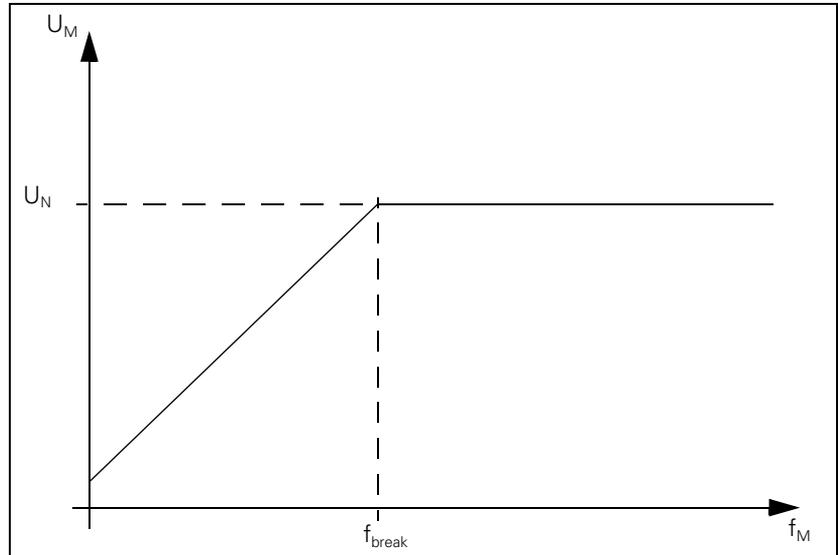
S code	min ⁻¹
S 83	1400
S 84	1600
S 85	1800
S 86	2000
S 87	2240
S 88	2500
S 89	2800
S 90	3150
S 91	3550
S 92	4000
S 93	4500
S 94	5000
S 95	5600
S 96	6300
S 97	7100
S 98	8000
S 99	9000



6.12.5 Volts-per-hertz control mode

In volts-per-hertz control mode (U/f control mode), the motor is speed-controlled in an open loop. The motor voltage increases in proportion to frequency up to the break (= threshold rpm for field weakening). Then the motor voltage remains constant (= rated voltage of motor); only the frequency continues to increase.

If the spindle reaches the maximum current due to excessive load, the error message **C380 Motor <Spindle 1/2> not controllable** appears. For the maximum current, the value from either the motor table or the power-module table of the drive (whichever is lower) applies.



The maximum speed in the volts-per-hertz control mode corresponds to the maximum speed in closed loop operation.

To drive a motor in the volts-per-hertz control mode :

- ▶ In the motor table, enter for your motor in the column **Motor model** (TYPE) UASM, in the column **Encoder line count** (STR.) the value 0, in the column **Type of encoder** (SYS) the value 0 and in the column **Maximum temperature** [°C] T-MAX the value 255.
- ▶ The machine parameters for current controller (MP24xx.y) and speed controller (MP25xx.y, MP 26xx.y) are nonfunctional.
- ▶ The acceleration and braking ramp (MP341x) must be set so that the maximum current is not exceeded.

Since during volts-per-hertz (U/f) control mode no speed encoder is used, W322 = 0 (actual speed value) supplies the value 0:

- ▶ Module 9164 can determine the actual speed value while the spindle is running, but not during the acceleration and braking phases.



Note

The oscilloscope shows the actual current instead of the nominal current (I NOML), since there is no nominal current with U/f components.

6.12.6 Oriented spindle stop

For spindle orientation the spindle must be in a closed control loop:

- ▶ Mount a position encoder for the spindle.
- ▶ With MP3010 (input value 6 to 8), specify whether the control provides for spindle orientation.

In the NC's touch probe cycles and rigid tapping cycle, the NC orients the spindle directly. In these cases, the NC sets M4017. You must reset M4012 in the PLC.

To orient the spindle to a specific angle in an NC program, use FN17: SYSWRITE ID990 NR8. The conditions above must be followed. The NC program resumes after the spindle is in position (M4000). You can ascertain the current spindle angle with FN18: SYSWRITE ID990 NR8.

If the spindle orientation is started with an M function (e.g. M19), you must activate the oriented spindle stop in the PLC.

In MP7442, enter the number of the M function (e.g., 19) which will trigger the oriented spindle stop during the machining cycles. If MP7442 = 0 (no oriented spindle stop), the error message **ORIENTATION not permitted** appears when a cycle which uses oriented spindle stop is called.

The spindle orientation runs asynchronously to the NC positioning commands. You may only acknowledge the orientation once the spindle is in position (M4000).

The NC starts orienting the spindle only if the drive is switched on with Module 9161.

There are three ways to orient the spindle in the PLC:

- Module 9171
- Marker M4130
- Via initiator with marker M4011

MP7442 Number of the M function for spindle orientation in the fixed cycles

Input: 1 to 999: Number of the M function
 0: No oriented spindle stop
 -1: Oriented spindle stop by the NC



Oriented spindle stop with Module 9171

The spindle speed is reduced in open-loop control along the ramp from MP3412.1 to the speed for spindle orientation (MP3520.1). As soon as this speed is reached, the control loop closes. The spindle is oriented in feedback control along the ramp from MP3412.1 to the nominal position. As long as the spindle moves in a closed loop, M4017 remains set:

- ▶ In MP3440.x, assign each gear range a k_v factor for adjusting the gear ranges.
- ▶ In MP3415.1, define the overshoot behavior of the first spindle during spindle orientation.
- ▶ Define the positioning window in MP3420. As soon as the spindle is in the positioning window, M4000 is set.

If the spindle should not remain in the position control loop after it reaches the nominal position, then you must set M4012. Once the marker is set the spindle is no longer operated with position feedback control.

If M4012 always remains set, the control loop opens after every oriented spindle stop as soon as the positioning window is reached.

You can compensate a maladjustment resulting from mounting the rotary encoder:

- ▶ In MP3430, enter the offset between the nominal and actual position of the reference mark. The offset is then compensated during orientation.

After the spindle is switched on, the NC evaluates the reference mark, even if the position control loop is not closed. M4018 is set until the reference mark is evaluated. For special applications you can evaluate the reference mark again by setting M4015. The NC resets M4015 once the reference mark has been evaluated.

- ▶ With MP7291, select the display mode for the spindle position. If M03 and M04 are not active, the display returns to zero every 360 degrees (modulo function).

MP3412.1	Multiplier for MP3411 during spindle orientation
Input:	0 to 1.999
MP3415.1	Spindle overshoot behavior during orientation
Input:	0 to 1000 [ms]
MP3420	Spindle positioning window
Input:	0 to 360.0000 [°]
MP3430	Deviation of the reference mark from the desired position (spindle preset)
Input:	0 to 360 [°]
MP3440.0-7	k_v factor for spindle orientation for gear ranges 1 to 8
Input:	0.1 to 10 [(1000°/min) /°]
MP3520.1	Spindle speed for oriented stop
Input:	0 to 99 999.999 [min ⁻¹]

		Set	Reset
M4000	Spindle in position	NC	NC
M4012	Open the spindle control loop	PLC	PLC
M4015	Renewed evaluation of the spindle reference mark	PLC	NC
M4017	Servo-controlled spindle in motion	NC	NC
M4018	Reference mark for spindle not yet traversed	NC	NC

With Module 9171 you can specify the speed, nominal position and direction of rotation for spindle orientation.

M4130 is set as long as the positioning movement lasts.



Module 9171 Oriented spindle stop

The module functions only in the cyclic PLC program. If you call the module while the spindle is rotating, the transferred direction will be ignored. The spindle will be oriented in the direction of spindle rotation.

If the values 2 to 4 are transferred as direction of rotation, the spindle will be oriented to the angle last defined in CYCL DEF 13. The transferred angle is added to the value from CYCL DEF 13.

Call:

PS B/W/D/K <Angle [1/10 000 °]>
or additional preset if there is a value from CYCL DEF 13

PS B/W/D/K <speed [1/1000 min⁻¹]>
0: MP3520.1 is assumed

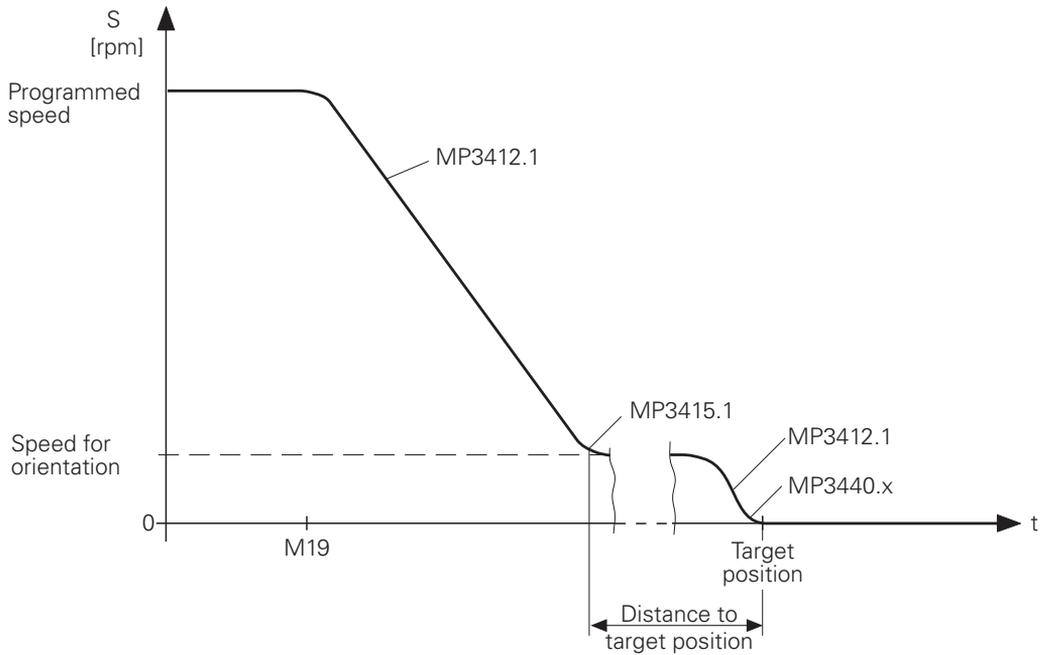
PS B/W/D/K <Direction of rotation>
-1: Negative direction (M04)
0: Direction of the shortest path
1: Positive direction (M03)
2: Same as -1 but angle from CYCL DEF 13
3: Same as 0 but angle from CYCL DEF 13
4: Same as +1 but angle from CYCL DEF 13
5: Incremental procedure with the entered angle.
Depending on the algebraic sign given, orientation occurs in either positive or negative direction (+/-).

CM 9171

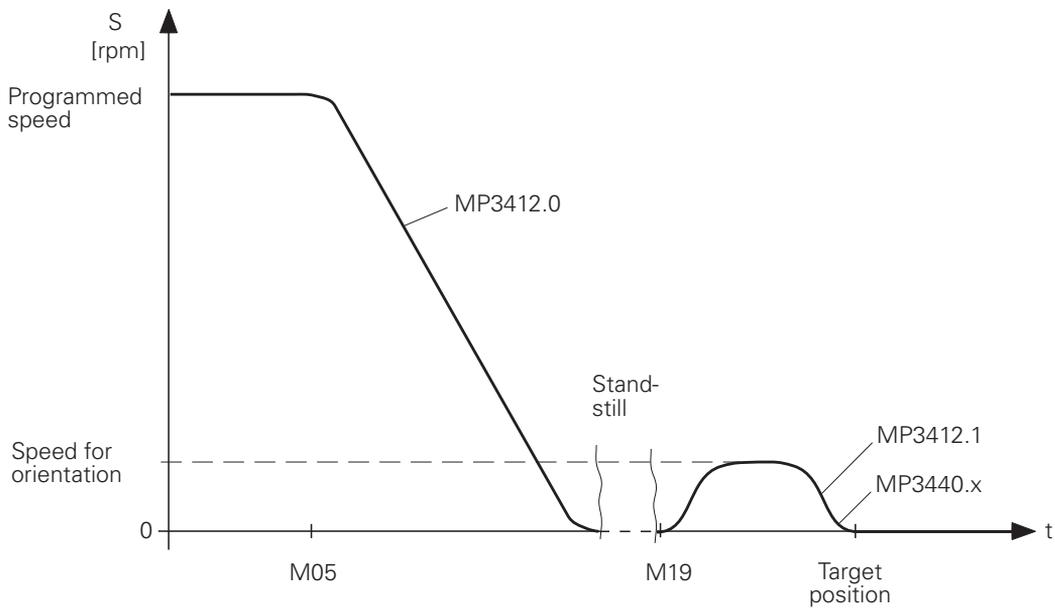
Error code:

Marker	Value	Meaning
M4203	0	Spindle is brought to an oriented stop
	1	Error code in W1022
W1022	1	Incorrect value for direction of rotation or rotational angle
	2	Incorrect speed
	19	No feedback-controlled spindle
	24	The module was called in a spawn job or submit job
	27	A spindle orientation is already running

Orientation of a moving spindle:



Orientation of a stationary spindle:



Oriented spindle stop with M4130

You can start the spindle orientation with M4130. The nominal position is taken from D592 and the speed from MP3520.1. The nominal position is with respect to the reference point.

For example, the nominal position can be transferred with MP4210.x or taken from the oriented spindle stop cycle (CYCL DEF 13). If the value is taken from the cycle, you must set the MSB of D592 to 1 and the other bits to 0. M4016 is set during execution of Cycle 13.

From a standstill, the spindle is oriented on the shortest path.

Prerequisite: At the start, the distance between the nominal and actual position must not be greater than the positioning window (MP3420). If the distance is greater than the positioning window, the spindle is positioned according to M4013 with M03 or M04.

		Set	Reset
D592	Nominal position for spindle orientation	PLC	PLC
M4013	Direction for spindle orientation from a standstill (M03 = 0; M04 = 1)	PLC	PLC
M4016	Cycle 13 is executed	NC	PLC
M4130	Activation of spindle orientation, or spindle orientation has been started with Module 9171	NC/PLC	NC

MP4210.0-47 Setting a number in the PLC (D768 to D956)

Input: -99 999.9999 to +99 999.9999

Oriented spindle stop via proximity switch with M4011

The spindle can be oriented through a proximity switch:

► Set M4011.

Then the spindle is moved in the direction from M4013 and at the speed from MP3520.0. The spindle is stopped as soon as you reset M4011. The current positioning value is shown in the status window.

MP3520.0 Speed activation through marker M4011

Input: 0 to 99 999.999 [min⁻¹]

		Set	Reset
M4011	Activate rotational speed MP3520.0 and direction of rotation from M4013	PLC	PLC

Offset compensation (only analog spindles)

After spindle orientation the offset is compensated automatically. In order to give the spindle enough time to settle to a stop, the offset compensation is delayed until the spindle has been in position for at least two seconds. The offset is then compensated in intervals of 0.152 mV per second. The spindle turns slowly due to the offset voltage.

6.12.7 Tapping with floating tap holder and nominal speed output

For tapping with floating tap holder, the position control loop is open. M4030 is set during the tapping cycle. After the spindle is switched on with M03, this is acknowledged with M4092. The nominal spindle speed must be reached before infeed begins.

During switch-on, the spindle follows the ramp in MP3411.x. During switch-off, it follows the ramp in MP3412.2:

- ▶ In MP3412.2, enter a multiplier for MP3411 during tapping.
- ▶ In MP3415.2, define the overshoot behavior of the spindle during tapping.
- ▶ Acknowledge the output of the M functions. An NC stop cannot be executed until a previous M function is acknowledged.

If the feed-rate and spindle ramps have differing gradients, the spindle follows the slower ramp.

Example:

Shaft speed $s = 1000 \text{ min}^{-1}$

MP3411.x = 0.025 [1000 $\text{min}^{-1}/\text{ms}$]

$$\frac{1000 \text{ min}^{-1}}{0,025 \cdot [1000 \text{ min}^{-1}/\text{ms}]} = 40 \text{ ms}$$

In this example the spindle was braked 40 ms before reaching the hole depth.

Delay times permit an optimum adjustment of the floating tap holder. You can delay the switch-off:

- ▶ In MP7120.2 enter a spindle slow-down time.

The delay cannot last longer than 30 ms before reaching the hole depth. Values above 30 ms are ignored (see the diagram).

You can delay a subsequent spindle start with M04:

- ▶ In MP7120.0 enter a dwell time. The ramp follows MP3412.2

You can delay restarting the infeed:

- ▶ Change the programmed dwell time in the cycle.

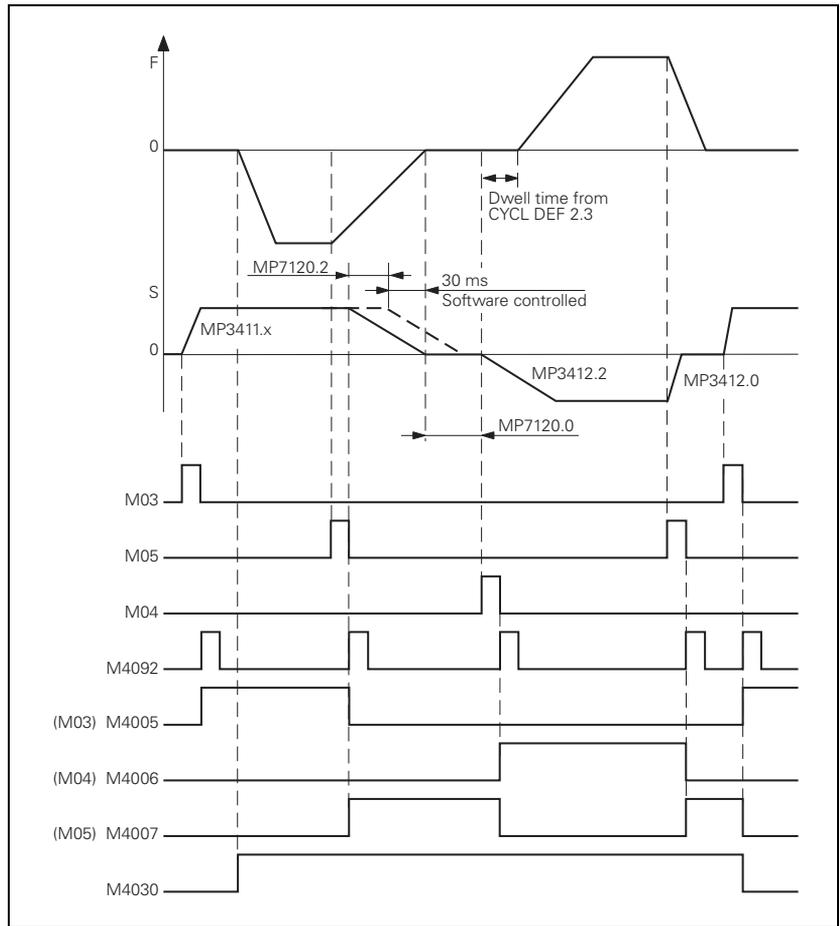
The NC uses M05 to switch off the spindle. The switch-off ramp follows MP3412.0. Then the spindle is switched back on with M03.

The feed rate override for tapping must be limited. Otherwise the floating tap holder may be damaged:

- ▶ Enter a limit in MP7110.x.



The following diagram shows the time sequence of the cycle:

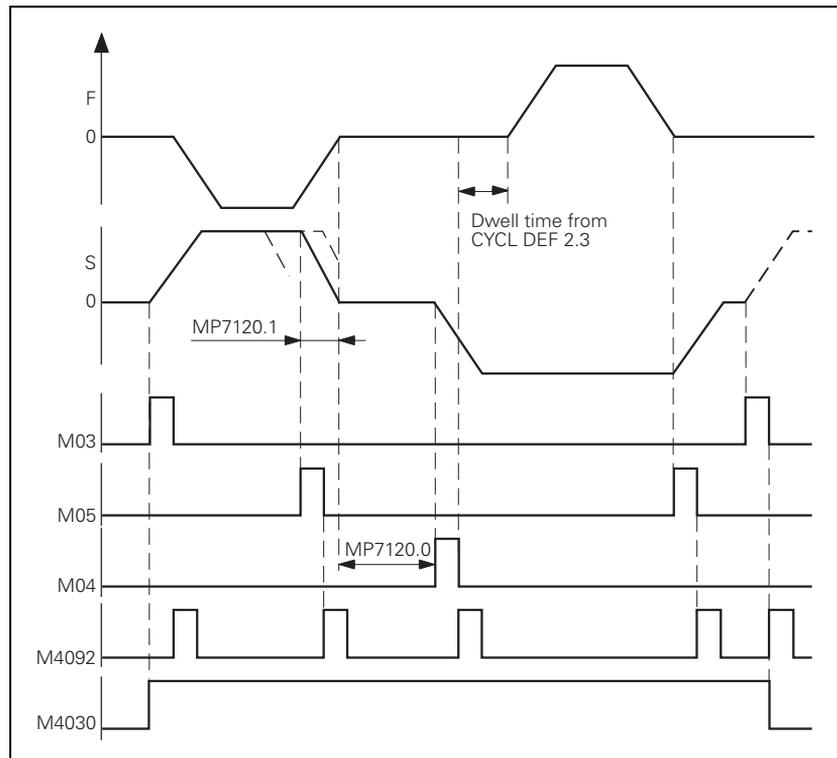


		Set	Reset
M4030	Cycle 2 or Cycle 17 active	NC	NC
MP3412.2	Multiplier for MP3411 during tapping		
Input:	0 to 1.999		
MP3415.2	Overshoot behavior of the spindle during tapping		
Input:	0 to 1000 [ms]		
MP7110.0	Minimum for feed-rate override during tapping		
Input:	0 to 150 [%]		
MP7110.1	Maximum for feed-rate override during tapping		
Input:	0 to 150 [%]		
MP7120.0	Dwell time for reversal of spindle rotational direction		
Input:	0 to 65.535 [s]		
MP7120.2	Spindle slow-down time after reaching the hole depth		
Input:	0 to 65.535 [s]		

6.12.8 Tapping with floating tap holder and coded spindle-speed output

The dwell time for rotational direction reversal (MP7120.0) and the programmed dwell time have the same effect as the nominal speed value output.

The following diagram shows the time sequence of the cycle:



MP7120.1 Advanced switching time of the spindle during tapping with coded spindle-speed output

Input: 0 to 65.535 [s]

6.12.9 Rigid tapping

Cycle 17

- ▶ Define the rigid tapping process in the NC program with Cycle 17. While Cycle 17 is running, the iTNC automatically switches the tool axis to velocity feedforward mode.
- ▶ Define the dynamic response of the spindle and the machine tool axes in machine parameters. See "The control loop" on page 810 and „Spindle" on page 944.

With Cycle 17 the spindle can also be feedback-controlled. This results in a better speed curve:

- ▶ Set MP7160 bit 2 = 1 to drive the spindle under position feedback control with Cycle 17.

The tool axis can track the spindle or it can be interpolated with the spindle. Interpolation can result in higher speed stability of the tool axis. The path jerk (spindle and tool axis) can be set via MP3415.3:

$$r = \frac{a}{\text{MP3415.3}}$$

Whichever value is smaller from this formula and from MP1090.0 is valid.

- ▶ In MP7160, set bit 4 = 1 to interpolate the tool axis with the spindle.

With small thread depths and excessive spindle speeds it is possible that the programmed spindle speed may not be attained. The immediate transition from the acceleration phase to the braking phase can diminish the quality of the thread:

- ▶ Set MP7160 bit 1 = 1 in order to limit the spindle speed so that the spindle runs for about 1/3 of the tapping time at a constant speed.

During tapping, the position of the tool axis tracks the actual position of the spindle.

Please note that the use of acceleration feedforward control for the tool axis makes the tool axis sensitive to fluctuations in spindle speed caused, for example, by gear transmission. If this happens, the tool axis starts to run rough:

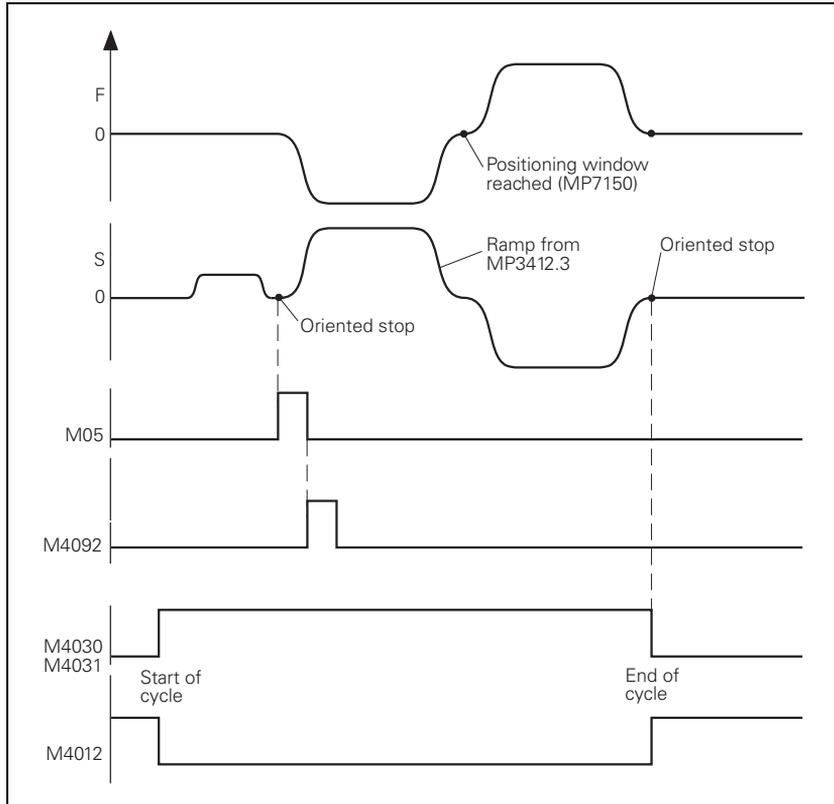
- ▶ In MP7160, set bit 3 = 1 to switch off acceleration feedforward control for Cycle 17.

Before tapping, the axes (e.g. Z and S) are synchronized through an oriented spindle stop, i.e., every Z position is assigned to a certain spindle angle. The NC orients the spindle. The NC sets M4017. The position control loop must be closed (M4012). Also see „Oriented spindle stop" on page 968.

Synchronization makes it possible to cut the same thread more than once. The assigned spindle angle depends on the thread pitch entered in the cycle. You can deselect this function to save machining time:

- ▶ Set MP7160 bit 0 = 1
In this case you cannot cut the thread more than once.

M4031 and M4030 are set while the cycle runs.

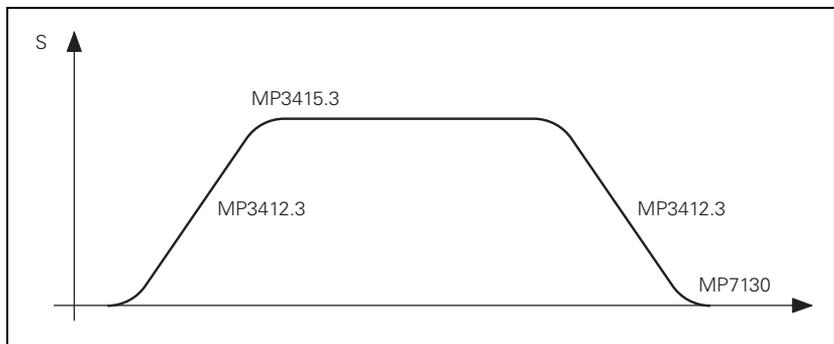


While Cycle 17 is running, the positioning window from MP7150 applies to the tool axis:

- ▶ Enter a value in MP7150 less than or equal to MP1030.x.

Define the acceleration and braking process of the spindle during rigid tapping:

- ▶ In MP3412.3 enter a multiplier for MP3411.x.
- ▶ With MP3415.3, define the overshoot behavior of the spindle.
- ▶ With MP7130, define the run-in behavior of the spindle.



MP3412.3	Multiplier for MP3411.x for rigid tapping
Input:	0 to 1.999
MP3415.3	Overshoot behavior of the first spindle during rigid tapping
Input:	0 to 1000 [ms]
MP7130	Run-in behavior of the spindle during rigid tapping
Input:	0.001 to 10 [°/min]
MP7150	Positioning window of the tool axis during rigid tapping
Input:	0.0001 to 2 [mm]
MP7160	Spindle response during Cycles 17, 207 and 18
Format:	%xxx
Input:	Bit 0 – Oriented spindle stop with Cycles 17 and 207 0: Oriented spindle stop before execution of the cycle 1: No oriented spindle stop before execution of the cycle Bit 1 – Spindle speed 0: Spindle speed is not limited 1: Spindle speed is limited so that it runs with constant speed approx. 1/3 of the time Bit 2 – Spindle in position feedback control 0: Spindle operated without position feedback control 1: Spindle operated with position feedback control Bit 3 – Acceleration feedforward 0: Active 1: Not active Bit 4 – 0: Tool axis tracks the spindle 1: Tool axis and spindle interpolated

		Set	Reset
M4030	Cycle 2 or Cycle 17 active	NC	NC
M4031	Cycle 17 or Cycle 18 active	NC	NC

Cycle 18

With Cycle 18 the tool axis tracks the actual position of the spindle. The starting position is the actual position. The target position is the hole depth:

► Program the approach and departure separately.

M4031 is set while Cycle 18 is running. M4012 must be reset for the cycle to be executed.

MP3412.3, MP3415.3, MP7130, MP7150 and MP7160 bit 1, bit 2 and bit 3 function as for Cycle 17.

6.12.10 Switching the modes of operation

For a spindle motor, two parameter sets with the same name can be saved in the motor table. This may be necessary if

- Another parameter set applies to a spindle motor at the higher speed range.
- A wye/delta connection switchover is carried out for a motor.

The switchover can be carried out during standstill or with a revolving spindle.



Danger

The contactor for the wye/delta switchover must not be switched under load or as long as a magnetizing current flows!

As soon as the operating mode is switched with Module 9163, the NC switches the drive controller of the spindle off and activates the parameter set from the motor table and the machine parameters. You can check this with Module 9162. After the operating mode has been switched, you must reactivate the drive controller of the spindle with Module 9161.

To use the operating-mode switchover:

- ▶ Enter the two parameter blocks of your spindle motor with the same name in the motor table. Identify parameter block 1 by entering 0 in the **MODE** column, and parameter block 2 by entering 1.
- ▶ Switch between the two operating modes with Module 9163.
- ▶ With Module 9161, reactivate the drive controller.

For the two operating modes, you can use different machine parameters for the current and speed controllers:

- ▶ In MP131.x you enter the y index of machine parameters MP2xxx.y for the current and speed controller in operating mode 0.
- ▶ In MP132.x you enter the y index of machine parameters MP2xxx.y for the current and speed controller in operating mode 1.

Module 9163 Switching the operating modes

Call:

```
PS   B/W/D/K <Control loop>
      Axis-specific: Dependence on MP4020 bit 14
PS   B/W/D/K <Type of connection>
      0: Operating mode 0
      1: Operating mode 1
```

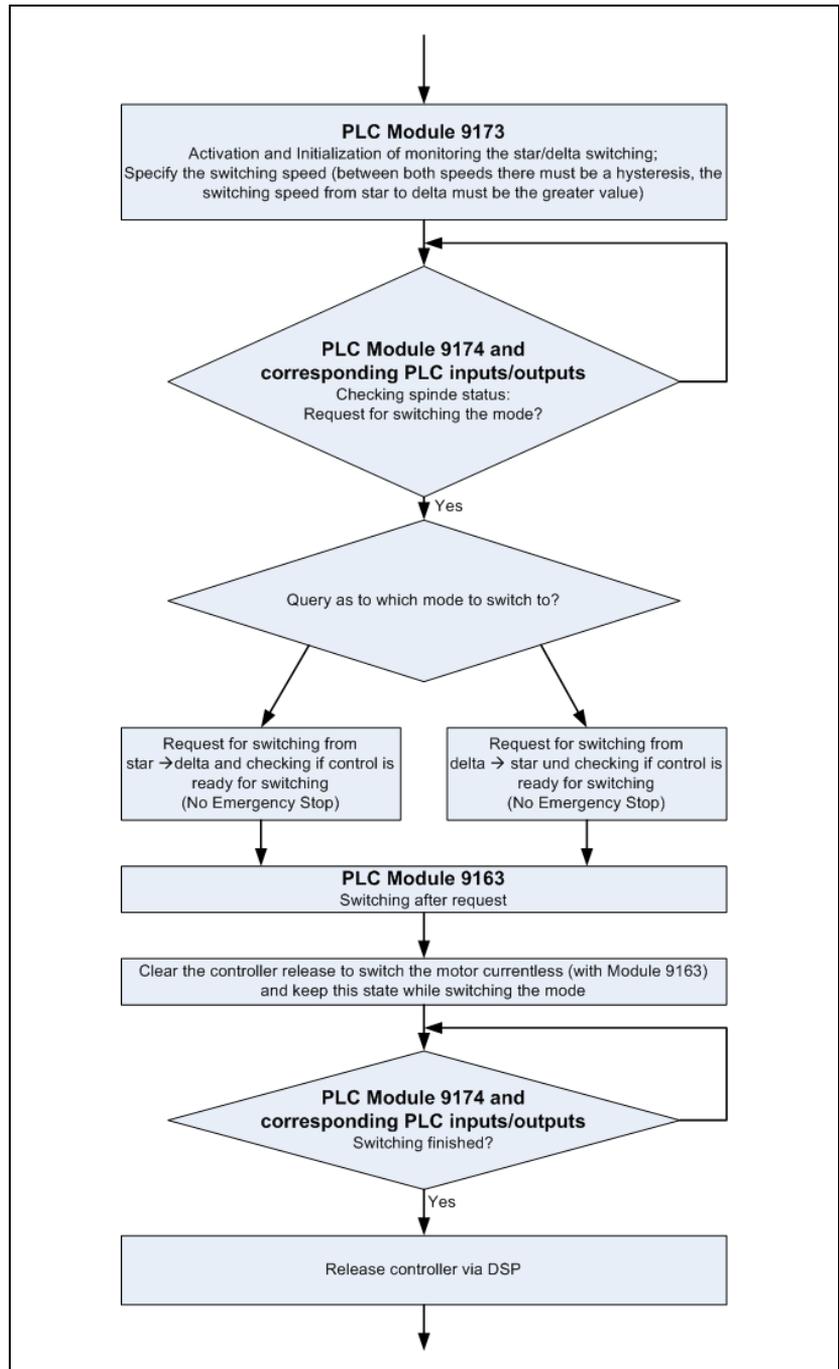
CM 9163

Error code:

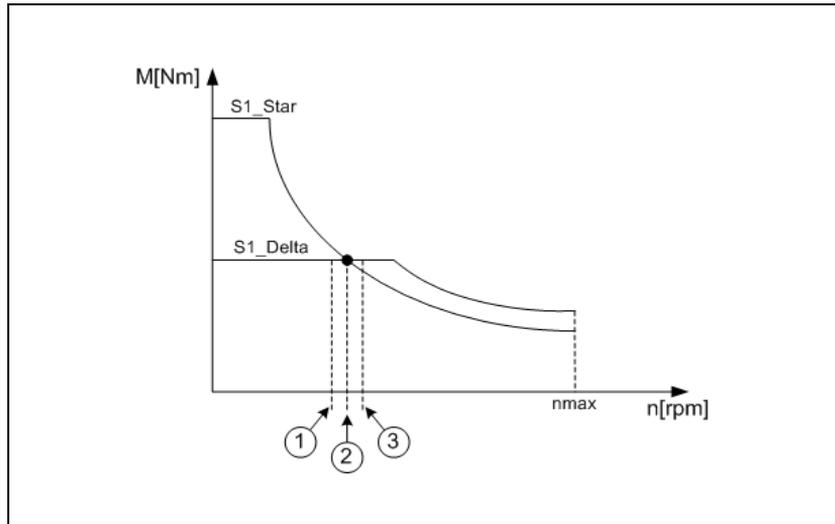
Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Switching not possible for this control loop
	2	Incorrect operating mode or incorrect control-loop number

Wye/delta switchover

It is possible to switch between wye and delta operation of the spindle without having to stop the spindle. However, only one gear stage can be used in order to use this direct switchover. This must be configured correspondingly via MP3010, MP3210 and MP351x. The flowchart shows the principle procedure for direct switchover of the modes via the PLC:



The diagram shows in principle how you can determine the switchover speeds for the wye/delta switchover:



- 1: Lower switching speed = $n(\text{intersection}) - 2\% \text{ to } 4\% \text{ of } n_{\text{max}}$
- 2: Speed at which the two characteristic curves intersect:
 $n(\text{intersection}) \text{ at } n(S1_{\text{wye}}) = n(S1_{\text{delta}})$
- 3: Upper switching speed = $n(\text{intersection}) + 2\% \text{ to } 4\% \text{ of } n_{\text{max}}$



Danger

The contactor for the wye/delta switchover must not be switched under load!



Attention

- On machines with transmission or wye/delta switchover, the RPM FACTOR (Q403) function of Cycle 209 (Tapping with chip breaking) must not be used with a value other than the default value. Switching the gear range or wye/delta switchover during retraction with the RPM FACTOR function may cause damage to the machine.
- Please inform the customers who are using these types of machines that the RPM FACTOR (Q403) function of Cycle 209 (Tapping with chip breaking) must not be used with a value other than the default value.
- Synchronous motors may only be switched at standstill. Switching a rotating synchronous motor is not possible.

MP2223.x Bit 3—Wait for readiness of current controller
 Format: %xxxxxxxxxxxxxxxxxx
 Input: 0: Wait for readiness of current controller
 1: Do not wait for readiness of current controller



Module 9173 Speed-dependent wye/delta switchover

With Module 9173, a monitoring function dependent on the speed can be realized for the wye/delta switchover of the spindle. A switchover request is detected by Module 9174, which supplies the current status of the wye/delta operation. If a switchover is necessary, this is still done by Module 9163. The actual speed is assumed during the switchover. The readiness of the current controller is automatically rescinded when Module 9163 is called. Bit 7 of MP2223 can be set as an alternative.

Condition:

- The module should not be called cyclically. A single call is enough for activation, deactivation or changing.

Additional information: Useful combinations for <Mode> parameter:

- xx0 Deactivate monitoring
- 001 Monitoring on, Switchover compatible
- 011 Monitoring on, Accelerated but safe switchover, Wait for readiness of current controller
- 111 Monitoring on, Fastest switchover possible

Call:

PS B/W/D/K <Mode>

Bit 0:

0 =Monitoring off (function deactivated)

1 =Monitoring on (automatic assumption of the actual speed during switchover)

Bit 1:

0 =Switchover compatible to previous switchover; switchover same as first switch-on of spindle

1 =Switchover accelerated

Bit 2:

0 =Wait for readiness of current controller

1 =Do not wait for readiness of current controller

PS B/W/D/K <Switchover speed from wye to delta operation>

PS B/W/D/K <Switchover speed from delta to wye operation>

CM 9173

Error code:

Marker	Value	Meaning
M4203	0	Speed-dependent monitoring active
	1	Error code in W1022
W1022	1	Invalid speeds (switchover speed of wye operation \geq switchover speed of delta operation), or negative speed

Module 9174 Spindle status regarding wye/delta switchover

Module 9174 supplies the current spindle status for wye/delta operation. In order to receive one of the two "Request switchover x -> x" status values, the speed-dependent wye/delta switchover monitoring in Module 9173 must be active.

Call:

CM 9174

PL B/W/D/K <Status>

0: Spindle in wye operation

1: Request for wye -> delta switchover

2: Spindle in delta operation

3: Request for delta -> wye switchover

Special procedure for asynchronous motors

Special procedure for wye/delta switchover on asynchronous motors.

- Activate the power module and the current controller simultaneously. By setting MP2222 bit 10, the sequence upon restart of the drive after a wye/delta switchover is optimized. HEIDENHAIN recommends setting bit 10 = 1 when using asynchronous motors.
- Demagnetize the motor during switch-off. By setting MP222 bit 11, the motor is demagnetized during switch-off of the drive before the switching process. HEIDENHAIN recommends setting bit 11 = 1 only if the residual magnetism of the motor leads to problems during switching (e.g. "overcurrent" error message of the power module during switching).

MP2222.x Special controller functions

Format: %xxxxxxxxxxxxxxxx

Input: Bit 10 – Optimized sequence for wye/delta switchover on asynchronous motors: Activate the power module and the current controller simultaneously.

0: Not active

1: Active

Bit 11 – Special sequence for wye/delta switchover on asynchronous motors: Demagnetize the motor during switch-off.

0: Not active

1: Active

6.12.11 Operating a second spindle

With the iTNC 530 you can operate two spindles alternately, i.e., only one spindle can be active at a given time.

Both spindles can be operated as analog (over CMA-H) or digital spindles. If one spindle is to be operated as a digital spindle and the other one as an analog spindle, the first spindle must be operated as a digital spindle.

Assignment of encoder input and speed command output

The second spindle is driven instead of an axis, i.e., there are fewer axes available. An exception is analog operation of the second spindle without a position encoder. In this case all axes remain available. The assignment of position and speed encoder inputs as well as of speed command outputs is entered in MP111.x, MP113.x and MP121.x. See "Assignment for axes" on page 614.



Note

If the speed encoder (with active reference mark monitoring, MP2220 bit 0) is disconnected and reconnected, the reference mark must be reevaluated (M4015) after the drive has been switched on again, otherwise the error message **Incorrect reference position** appears.

Switching between the spindles

You can switch between the two spindles through the PLC:

- ▶ Enter MP4020 bit 5 = 1 to activate double spindle operation.
- ▶ With Module 9175, switch between spindle 1 and spindle 2.

Module 9179 is used to determine the active spindle.

Commissioning the second spindle

- ▶ Digital second spindle: In MP10, deactivate one axis.
- ▶ Digital second spindle: Set MP110.x, MP112.x and MP120.x of the deactivated axis to zero.
- ▶ Machine parameters MP13010 to MP13520 are available for the second spindle. In their functions and input ranges, these parameters are identical with MP3010 to MP3520 for the first spindle. "Spindle" on page 944.
- ▶ Current and speed controller: For commissioning, use the machine parameters MP2040.x to MP2912.x. Determine the x index to be used for the second spindle with MP131.1 (for operating mode 0) and with MP132.1 (for operating mode 1).

MP4020 PLC functions

Format: %xxxxxxxxxxxxxx

Input: Bit 5 – Single- or double-spindle operation

0: Single-spindle operation

1: Double-spindle operation

MP13010 bis MP13520 Machine parameter block for the second spindle

Input: Function and input range are identical to MP3010 to MP3520

Module 9175 Switch the spindle

With this module you can switch between spindle 1 and spindle 2. When switching via an M strobe, MP7440 bit 2 must not be set. When switching via an S or G strobe, MP3030 or MP13030, respectively, must not be set. The module only needs to be called once. Switching is only possible if

- the control is not in operation (M4176 is not set),
- the control is in operation (M4176 is set) and an M/S/T/T2/G strobe is pending, or
- the machine is not currently approaching a contour. (M4157 not set)

Call:

PS B/W/D/K <Spindle number>
0: First spindle
1: Second spindle

CM 9175

Error code:

Marker	Value	Meaning
M4203	0	Indicated spindle activated
	1	Error code in W1022
W1022	2	Invalid spindle number
	6	M4157 = 1 (RESTORE POSITION active)
	20	Module was called in a spawn or submit job
	21	Missing strobe in M4176 = 1

Module 9179 Status information about spindle(s)

Status information about the spindles can be ascertained with Module 9179.

Call:

PS B/W/D/K <Number of spindle>
PS B/W/D/K <Code for status information>
0: Active spindle ("Number of spindle" is not evaluated)

CM 9179

PL B/W/D <Status information>

Error code:

Marker	Value	Meaning
M4203	0	Status information has been ascertained
	1	Error code in W1022
W1022	1	Invalid code for status information
	2	Invalid spindle number



6.12.12 C-axis operation

In C-axis operation, an axis and a spindle are driven by the same motor. Digital or analog operation of axis and spindle is possible. Axis and spindle may each be equipped with one position encoder. Because the speed encoder is built into the motor, it measures both the axis and the spindle.

As of software version 606 42x-01 SP 02, it is possible to run a C axis with the CC 61xx. The position encoder inputs must be on the same drive-control motherboard in the CC 61xx. This way, you can configure a second position encoder input for a drive for C-axis operation or a central drive. It is not possible to distribute the position encoder inputs to various drive-control motherboards.

Example:

Spindle on X15, X51, X201

B axis with additional position encoder on X15, X51, X202

This configuration is possible, but the spindle position encoder must be connected with the position input belonging to the PWM output.

Assignment of encoder inputs and speed command outputs to the axis and spindle:

- ▶ Enter in MP110.x the position encoder input of the axis (if present).
- ▶ Enter in MP111.x the position encoder input of the spindle (if present).
- ▶ Enter MP112.x = 0 for the axis (it uses the speed encoder of the spindle motor).
- ▶ Enter in MP113.x the speed encoder input of the spindle.
- ▶ Enter the same speed command output in MP121.x for the spindle and in MP120.x for the axis.

Commissioning of the axis and the spindle:

- ▶ The current and speed controllers are commissioned only for the spindle.
- ▶ The position controllers **must** be commissioned separately for the axis and spindle.



Note

The axis position controller should be commissioned in the gear range that is actually used for positioning.

If possible, use the lowest gear range to ensure optimum control.

C axis operation must be deselected for commissioning the spindle, meaning that no identical PWM outputs may be entered in MP120.x and in MP121.x.

If you use only one position encoder for both the spindle and the axis, the axis display keeps running while the spindle is in operation:

- ▶ Before switching from the axis to the spindle, save the actual position value of the axis with Module 9146. This ensures that the axis display remains at the last value, even when the spindle is rotating.
- ▶ Before switching from the spindle to the axis, recover the actual position value of the axis with Module 9146.

If you save the actual position value with Module 9146 and then close the position control loop, or if the position control loop is closed and the actual position value is then saved with Module 9146, the error message **Actual position value saved <Axis>** appears. The error message triggers an emergency stop.

Switching from **spindle to axis**:

- ▶ Stop the spindle.
- ▶ Switch to the gear range required for axis operation.
- ▶ Switch the spindle motor to the axis.
- ▶ With Module 9156, switch the axis from the open-loop to the closed-loop (servo-controlled) state.
- ▶ Enable the current and speed controls via Module 9161 with the corresponding bit for the axis.
- ▶ Release the axis clamping.
- ▶ Close the position control loop of the axis by setting the corresponding bit in W1040/W1132.
- ▶ Begin axis operation.

Switching from **axis to spindle**:

- ▶ Stop the axis.
- ▶ Clamp the axis.
- ▶ Open the position control loop of the axis by resetting the corresponding bit in W1040/W1132.
- ▶ With Module 9155, switch the feedback control for the axis off.
- ▶ Switch the spindle motor from the axis back to the spindle.
- ▶ With Module 9161 bit 15, release the current and speed controllers.
- ▶ Shift back to the original gear range.
- ▶ Start spindle operation.



Module 9146 Save and reestablish actual position values

Module 9146 saves and later reestablishes the actual position values of axes. If the actual position values were saved, the last value displayed remains until they are reestablished.

Call:

PS B/W/D/K <Axes bit-encoded>

PS B/W/D/K <Mode>

0: Save actual position values

1: Reestablish actual position values

CM 9146

Error code:

Marker	Value	Meaning
M4203	0	Actual position values saved or reestablished
	1	Error code in W1022
W1022	1	Invalid mode
	2	Invalid axes
	24	Module was called in a spawn job or submit job

Module 9155 Axis switchover from closed loop to open loop

With Module 9155 you can switch an axis from the closed-loop to the open-loop state. Now the bit can be transferred to the axis.

Call:

PS B/W/D/K <Axes bit-encoded>

CM 9155

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Invalid axis number
	21	Missing strobe or M4176 = 1
	24	Module was called in a spawn or submit job

Module 9156 Axis switchover from open loop to closed loop

With Module 9156 you can switch an axis from the open-loop to the closed-loop state. An automatic actual-to-nominal value transfer is executed. Now the bit can be transferred to the axis.

Call:

PS B/W/D/K <Axes bit-encoded>

CM 9156

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Invalid axis number
	21	Missing strobe or M4176 = 1
	24	Module was called in a spawn or submit job



6.12.13 Software option 96—Advanced spindle interpolation

To be able to use additional functions for an interpolated spindle, you need option 96. The prerequisite is that the spindle can be operated and positioned as an axis. The PLC program and machine parameters must permit C-axis operation for this. To activate the C-axis mode, the spindle is oriented to the last saved position and then switched to axis mode. When positioning is finished, the control loop opens and the current position freezes. If you want to use this option, HEIDENHAIN recommends using the current PLC basic program. The use of this option is already prepared as of version 55.

In combination with this option and the new Cycle 290 INTERPOLATION TURNING, the iTNC530 can create a rotationally symmetric step, which is defined by the starting and end point, in the active working plane. The center of rotation is the tool location in the working plane at the time the cycle is called. The rotational surfaces can be inclined or rounded relative to each other.

In interpolation turning, the cutting edge of the tool moves on a circle, with the cutting edge always oriented to the center of the circle. By varying the circle radius and the axial position, any rotationally symmetric objects can be produced in any working plane.

This cycle can only be used for finishing. Roughing operations with multiple steps are not possible. The machining strategy can be chosen flexibly: from the outside in or vice versa, and also from top to bottom or vice versa. This results in four different machining strategies, which are distributed over the four quadrants.

6.13 Integrated oscilloscope

The iTNC features an integrated oscilloscope.

This oscilloscope has six channels, of which no more than four can be used for signals from the current and speed controller. If more than four channels of the current and speed controller are to be displayed, the error message **Channel <number> cannot be displayed** appears.

The following signals marked with an "x" can be recorded, depending on the controller being used:

Signal	Meaning	CC 6xxx, CC 424	CC 422
Saved	The signal last recorded is displayed	x	x
s actual	Actual position [mm]	x	x
s nominal	Nominal position [mm]	x	x
s diff	Following error of the position controller [mm]	x	x
Volt.analog	Analog axis/spindle: Analog voltage = nominal velocity value [mV]	x	x
v actual	Actual value of the axis feed rate [mm/min]. Calculated from position encoder.	x	x
v nominal	Nominal value of the axis feed rate [mm/min]. Axis feed rate calculated from the difference from the nominal position values. The following error is not included.	x	x
Feed rate	Contouring feed rate [mm/min]	x	x
Position: A	Signal A of the position encoder	x	x
Position: B	Signal B of the position encoder	x	x
v (act rpm)	Shaft speed actual value [mm/min]; Calculated from rotary speed encoder and standardized with MP1054.	x	x
v (nom rpm)	Nominal velocity value [mm/min]: Output quantity of the position controller	x	x
I (int rpm)	Integral-action component of the nominal current value [A]; CC 6xxx: Effective value	x	x
I nominal	Nominal current value [A] that determines torque; CC 6xxx: Effective value	x	x
I actual	Actual current value [A]	x	-
Actl. Id	Actual magnetizing current value [A]	x	-
Max. Iq	Maximum torque current [A]	x	-
Compensat.	Position compensation value (composed of temperature compensation, axis-error compensation, backlash compensation, etc.)	x	-
PLC	The PLC operands (B, W, D, I, O, T, C) are recorded. Enter the operand in the text box next to PLC.	x	x
a nominal	Nominal acceleration value (m/s ²)	x	x
r nominal	Nominal jerk value [m/s ³]	x	x
Pos. diff.	Difference between position and speed encoder [mm]	x	x
a actual	Actual acceleration value [m/s ²]. Calculated from position encoder.	x	x

Signal	Meaning	CC 6xxx, CC 424	CC 422
r actual	Actual jerk value [m/s ³]. Calculated from position encoder.	x	x
I ² -t (mot.)	Current value of the I ² t monitoring of the motor [%]	x	x
I ² -t (pow. module)	Current value of the I ² t monitoring of the power module [%]	x	x
Utilization	Current utilization of the drive [%]	x	x
Block no.	Block number of the NC program	x	x
Gantry Diff	Difference between synchronous axes [mm]	x	x
U nominal	Nominal voltage [V]	x	-
P mech.	Mechanical power [W]	x	-
P elec.	Electrical power [W]	x	-
M actual	Actual torque value [Nm]	x	x ^a
S noml (f.)	Nominal position as per nominal position value filter [mm]	x	x
DSP debug	Diagnosis function for internal purposes	x	-
Contour deviat.	Circular form test, contour deviation (mm)	x	x
F TCPM	Feed rate at the tool tip with TCPM	x	x
Int. diagn.	Reserved for internal purposes	x	-
DC-link P ^b	If MP2198.x and MP2199.x are configured, the DC-link power of the power supply module for the respective axis can be displayed.	x	-
Amplitude	Amplitude of the position encoder	x	x
Motor A	Signal A of the speed encoder	x	-
Motor B	Signal B of the speed encoder	x	-
CC DIAG	The signal is axis-specific and connected with an additional input box on the right side of the oscilloscope screen. In the input box you can enter a number for selecting a channel of the DSP. See the following table for the available channels and signals.	x	-
SPLC	For a control with functional safety (FS): value of an operand of the SPLC program of the MC	x	-
SPLC-CC	For a control with functional safety (FS): value of an operand of the SPLC program of the CC	x	-
KinComp	Value of the kinematic temperature compensation (function of KinematicsComp)	x	x
REF	Axis position in ref. coordinates	x	x

- a. With CC 422 only for synchronous motors without field weakening, not for asynchronous motors
- b. Only available with a modular HEIDENHAIN inverter system

The oscilloscope provides additional functions for commissioning the current controller. See "Commissioning" on page 1004.

The recorded data remain stored until you start recording again or activate another graphic function.

Number	Meaning
101	Status of emergency stop input
201	Masking the reference pulse via W1054/D1160
202	Activating the fine interpolation filter (nominal position value filter is active)
203	Position controller is closed W1040/W1132
204	Position control is fed forward (MP1392, MP 1392)
205	Deactivation of monitoring functions via W1042/W1136
206	Position control in semifeedforward mode (MP1396)
207	Rapid switch-off of speed controller (activated in MP4130)
208	DC-link voltage monitoring can be activated via PLC module 9167. The current status can be read via this channel.
300	Ready signal of power module (green LED on the power module)
301	Error signal of power module (triggers an error message)
302	Output for controlling the brake at the power module (X 344)
400	UV ready signal of the inverter
401	DC-link voltage too high
402	DC-link current too high
403	Excessive temperature of UV
404	Leakage current detected by UV
405	AC fail signal is active
600	I2T early warning threshold of drive was exceeded
601	Axis enabled (PLC module 9157)
602	Axis is active (PLC module 9162)
603	Commutation angle of motor was measured using the reference mark or EnDat
604	TNCopt function is currently being performed

Colors

- ▶ In MP7365.x, define the colors for the oscilloscope.

Setup

- ▶ Activate the oscilloscope with the DIAGNOSIS, DRIVE DIAGNOSTICS and OSCI soft keys.



Note

A step response is not possible. In order to make a step response possible, the keyword 688379 must have already been entered.

The setup menu appears:

- ▶ Choose the parameters to be entered with the cursor keys.

Manual operation	Oscilloscope
Mode of op.	YT
Sample time	3.0ms
Output	Ramp Feed rate 200
Channel 1	X v actual
Channel 2	<input checked="" type="checkbox"/> v nominal
Channel 3	Off
Channel 4	Off
Channel 5	Off
Channel 6	Off
Trigger	Channel 1
Trigger threshold	+0
Slope	+
Pre-trigger	0%

OSCI SAVE SCREEN RESTORE SCREEN MP EDIT END

The oscilloscope can be operated with reduced functions starting from the **Power interrupted** message. If you enter the code number **DSP123** before calling the oscilloscope, then starting from the **Power interrupted** message the data are recorded and can be displayed via the oscilloscope.



Note

Please note that the control has not finished booting at this time, and that the displayed values can therefore deviate from the correct or expected values.

Operating mode:

- ▶ Select the desired setting or choose the circular interpolation test
 - YT: Chronological depiction of the channels
 - XY: X/Y graph of two channels
 - CIRC: Circular interpolation test

Sample time:

- ▶ Set the time interval for recording the signals.
Possible entries: 0.6 ms, 1.8 ms and 3.6 ms
4096 samples are stored. The signals are therefore stored for the following duration:
 - $0.6 \text{ ms} \cdot 4096 = 2.4576 \text{ s}$
 - $1.8 \text{ ms} \cdot 4096 = 7.3728 \text{ s}$
 - $3.6 \text{ ms} \cdot 4096 = 14.7456 \text{ s}$

Output:

- ▶ Select whether the nominal speed value is to be issued as a step or ramp.
 - If you select ramp output, then the programmed feed rate, k_V factors, and acceleration values that you have specified with machine parameters go into effect.
 - If you select step output, a step will be output as nominal velocity value when you press the axis-direction buttons in the **Manual operating mode**. During output, the position control loop is open. A step can be output only if code number 688379 or 807667 has been entered.

Feed rate:

- ▶ Enter the height of the step for the nominal velocity value (in mm/min). This entry has no effect for ramp output.

Channel 1 to channel 6:

- ▶ Assign the channels of the recorded signals to the respective axes.

Trigger:

- ▶ Define the type of recording.
You have the following possibilities:
 - **Free run:** The recording is started and ended by soft key. When you press the STOP soft key, the last 4096 events are saved.
 - **Single shot:** When you press the START soft key, the next 4096 events are saved.
 - **Channel 1 to channel 6:** Recording begins as soon as the trigger threshold for the set channel is crossed.



Trigger threshold:

- ▶ Enter the trigger threshold (you will find the appropriate units in the signals table on 992):

Slope:

- ▶ Define whether recording will be triggered with the rising (positive) or falling (negative) edge.

Pre-trigger:

Recording begins at a time preceding the trigger time point by the value entered here

- ▶ Enter a value.

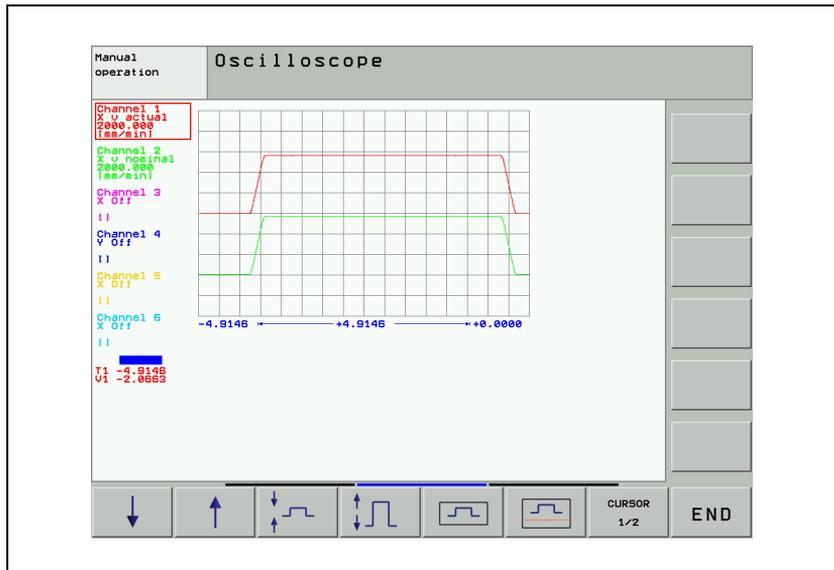
Delta trigger:

You can specify a second trigger threshold in the oscilloscope, which enables you to use a value range to define the event triggering a recording. Depending on the trigger edge and the first trigger value, you can determine whether a trigger signal is output when the value range is reached or exceeded. The inverse of the edge of the first trigger value is always selected as the active trigger edge of the delta trigger. The delta trigger is given as a value relative to the first trigger threshold. If a value of zero (0) is entered for the delta trigger (default setting), then the delta trigger is off:

- ▶ Enter the second trigger threshold (you will find the appropriate units in the signals table on 992):

Oscilloscope display:

- ▶ Press the OSCI soft key.



During recording, the selected signals are continuously displayed. After recording ends, the memory contents are displayed. In addition, the signal type and the resolution are displayed for each channel. The length of the recorded range, with respect to the entire memory content, is shown as a bar in the status field.

- ▶ Move the cursor with the arrow keys to select the channel. The amplitude of the selected channel as well as the time (with reference to start of recording) are shown in the status field.
- ▶ Activate a second cursor by pressing the **CURSOR 1/2** soft key. The oscilloscope displays the amplitude and time of this cursor. The time [s] of the second cursor is shown with respect to the time point of the first cursor. This function allows you to measure the acceleration time of an axis.



Note

The cursor is located on the trigger point only if it has not been moved after the measurement. If the cursor has been moved, it will remain at the point of the time axis to which it has been moved. The cursor does not return to the trigger point until a trigger parameter has been changed.

Meaning of the soft keys:

Meaning of the soft keys:	
	Hide/show grid lines.
	Hide/show lines between measured points.
	Start recording. Recording is ended either by the trigger condition or with the STOP soft key.
	Move the signal down.
	Move the signal up.
	Decrease the vertical resolution.
	Increase the vertical resolution.
	Optimum vertical resolution; The signal is centered in the middle of the screen.
	Optimum vertical resolution; The signal is referenced to the datum line.
	Switch to second cursor.
	Move the signal to the left.
	Move the signal to the right.
	Decrease horizontal resolution.
	Increase horizontal resolution.
	Invert the signal.
	Exit the oscilloscope function.

Saving and loading a recording

You can display the signal last recorded for a channel again by selecting the **Saved** signal.

With the **SAVE SCREEN** soft key in the Setup menu you can save the recorded signals with all settings in a file on the hard disk. The file must have the extension *.DTA. This file can

- be recalled with the PLC development software PLCdesign
- be read back into the control.

In order to read a *.DTA file back into the control:

- ▶ Press the **RESTORE SCREEN** soft key in the Setup menu.
- ▶ Enter the complete file name and path of the *.DTA file.
- ▶ Press the ENT key.
- ▶ Press the **OSCI** soft key to displays the signals from the *.DTA file.

As of software 340 49x-06, it is possible to operate the integrated oscilloscope from the NC program. This makes it possible, for example, to automate certain machine analyses. When the oscilloscope is operated from the program, it behaves in the same way as during manual operation. No additional locking takes place: For example, active recording can be stopped with the STOP soft key, restarted, or the settings can be changed. Remote access to the internal oscilloscope during recording can also disturb the recording process. **FUNCTION SCOPE START** reserves the internal oscilloscope for the respective function. The oscilloscope is released automatically with **FUNCTION SCOPE STOP**, or at the end of the program (PGM END) and selection of a new NC program. To call the new functions, enter the code number 555343, press the FUNCTION SCOPE soft key and then the PROGRAM FUNCTIONS soft key:

■ **FUNCTION SCOPE START JOB "INI file" RESULT "RES file"
FUNCTION SCOPE START JOB QS1 RESULT QS2**

This syntax starts the oscilloscope. The file specified in **JOB** initializes the oscilloscope (channels, measurands to be recorded, trigger conditions, sample time, ...). A *.DTA file is used for the initialization. This **INI file** can be created automatically. To do this, set the desired configuration in the internal oscilloscope and save the settings in a *.DTA file. Use the SAVE CONFIG. soft key in the oscilloscope to save the configuration. The file then only contains the current oscilloscope settings without the measured data. The path to which the oscilloscope is to save the measured data is indicated in **Result**. Both parameters are mandatory. However, you can specify them independently of each other as explicit text or as a QS reference. The INI file and the RES file must be entered with absolute paths. Both files must have the extension *.DTA for oscilloscope recordings. If an INI file already contains oscilloscope data, this data will be ignored during evaluation.

This function reserves the internal oscilloscope, and therefore the oscilloscope cannot be started with the OSCI soft key.

■ **FUNCTION SCOPE STOP Q1
FUNCTION SCOPE STOP**

This syntax switches the oscilloscope off again. If the execution of an NC program is to vary depending on whether a trigger condition has been met, a Q parameter can be transferred with the shutdown function. The oscilloscope management describes the value of the transferred parameter in order to give a feedback on the recording process. If the NC program is structured appropriately, and especially if a default value other than 1 is assigned to the parameter, this makes it possible for the user to derive whether a trigger condition has been met and to select appropriate branches that, for example, restart the oscilloscope or write the oscillograms to the RES file. The following parameter values can occur:

0 = Recording had already been stopped before the **SCOPE STOP** function was triggered. The trigger condition had not been met since **SCOPE START**.

1 = Recording was stopped by the **SCOPE STOP** function. The trigger condition had not been met since **SCOPE START**.

2 = Recording had already been stopped before the **SCOPE STOP** function was triggered. The trigger condition had been met since **SCOPE START**.

3 = Recording was stopped by the **SCOPE STOP** function. The trigger condition had been met since **SCOPE START**.

■ FUNCTION SCOPE STORE

If you call this function that does not contain parameters, the storage of the data in the buffer of the oscilloscope is forced from within the NC program. If a trigger condition has been fulfilled before the **STORE** function is called, the results for the trigger condition are written to the file indicated in **RESULT**. You need to stop recording before you can save the data. This function releases the internal oscilloscope, and therefore the oscilloscope can be started again with the OSCI soft key.

		Set	Reset
M4627	Trigger condition for integrated oscilloscope fulfilled	NC	PLC
M4628	Recording of the integrated oscilloscope ended	NC	PLC



Circular test

A circular interpolation test can be run in the oscilloscope.

- ▶ Choose the **CIRC** operating mode in the oscilloscope.
- ▶ Select the **Contour dev.** setting for the appropriate axes in the two channels.
- ▶ Start recording.
- ▶ Start an NC program in which a circle is programmed. The circle center point must be at the origin of both axes.
- ▶ Stop recording.

Below the grid, the recording time relative to the trigger time point is displayed.

Example of a circular interpolation test with the integrated oscilloscope:

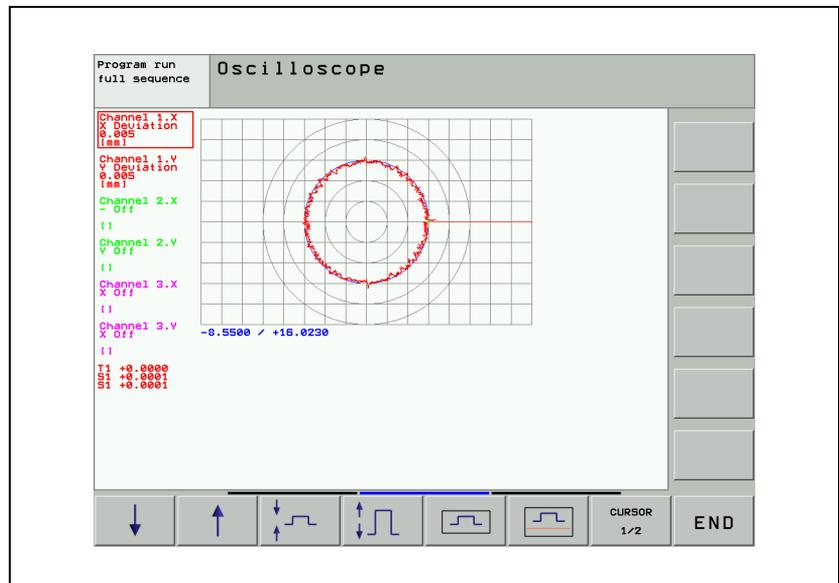
Actual position:

X +30

Y +0

NC program:

```
0 BEGIN PGM Circular interpolation test MM
1 CC X+0 Y+0
2 CP IPA+360 DR+ F1000
3 M30
4 END PGM Circular interpolation test MM
```



6.14 Commissioning

6.14.1 Tables for power modules, supply modules and motors

In the machine parameter editor you select the installed motors, power modules and supply modules.

- ▶ Call a list of power modules, motors or supply modules with the corresponding soft key.
In the list of motors, the type of motor (synchronous, asynchronous, or linear motor) and the operating mode are displayed in addition to the motor designation.

Meaning of the soft keys:	
	Call a list of power modules.
	Jump to the beginning of the list of power modules.
	Jump to the end of the list of power modules.
	Scroll one page forward in the list of power modules.
	Scroll one page backward in the list of power modules.
	Select a power module with the arrow keys and transfer it with the corresponding STORE MP2100.X soft key.
	Open the table of power modules in write-protected mode and jump to the selected power module.
	Open the table of power modules in editing mode and jump to the selected power module.
	Return to the machine parameter editor.

Meaning of the soft keys:	
	Call the list of motors.
	Jump to the beginning of the list of motors.
	Jump to the end of the list of motors.
	Scroll one page forward in the list of motors.
	Scroll one page backward in the list of motors.
	Select a motor with the arrow keys and transfer it with the corresponding "STORE MP2200.X" soft key.
	Open the table of motors in write-protected mode and jump to the selected motor.
	Open the table of motors in editing mode and jump to the selected motor.
	Return to the machine parameter editor.
	Call the list of supply modules.
	Jump to the beginning of the list of supply modules.
	Jump to the end of the list of supply modules.
	Scroll one page forward in the list of supply modules.
	Scroll one page backward in the list of supply modules.
	Select a supply module with the arrow keys and transfer it with the corresponding STORE MP2198.X soft key.
	Open the table of supply modules in write-protected mode and jump to the selected supply module.
	Open the table of supply modules in editing mode and jump to the selected supply module.
	Return to the machine parameter editor.

After you have selected the motor and the power module, the models are automatically entered in MP2100.x, MP2200.x and MP2198.x.

A search function supports you in selecting motors, power stages and power supply modules for the machine parameters and also supports a text search function. The FIND soft key is available in the motor.mot, inverter.inv and supply.spy tables. In the "Find" pop-up window you can enter the complete or partial name of the device you want to search for.

If you use motors, power modules or supply modules that are not listed in the menus, please contact HEIDENHAIN.

You can overwrite standard data or add other models to the tables. If you change the table of the motors, power modules or supply modules, the changed tables are saved separately in the PLC partition, where they are used by the iTNC:

- **PLC:\MP\MOTOR.MOT** (motor table)
- **PLC:\MP\INVERTER.INV** (power module table)
- **PLC:\MP\SUPPLY.SPY** (supply module table)
- **PLC:\MP\MOTOR.AMP** (old power module table)

If at any time you want to use the HEIDENHAIN standard tables again, you must erase the above-mentioned tables in the PLC partition.



Note

Due to reasons of compatibility, the iTNC still supports the **MOTOR.AMP** power-module table. However, it is not possible to use the higher currents of the HEIDENHAIN 1xx D inverters at the PWM frequencies 3.33 kHz and 4.0 kHz with this table.

The iTNC follows a specific sequence when searching for a valid table.

1. **PLC:\mp\inverter.inv**
2. **PLC:\mp\motor.amp**
3. **PLC:\mp\supply.spy**
4. **inverter.inv** HEIDENHAIN standard table for power modules on the system partition
5. **motor.amp** Previous HEIDENHAIN standard table for power modules on the system partition
6. **supply.spy** HEIDENHAIN standard table for supply modules on the system partition

If you use a motor that appears in the motor table, but only the data for the speed encoders differs, you can overwrite this data in the motor table with MP2202.x, MP2204.x and MP2206.x. The motor table is not actually changed. The changes only take place in the working memory.



Note

The original entry from the motor table is used only when MP2202.x = *, MP2204.x = * and MP2206.x = *.



After the commutation angle has been determined successfully, the **8150 – Field angle ascertainment successful** error message is now output. The drive remains controlled.

In MP2198.0 to MP 2198.4 you enter the types of power supply modules being used. This makes it possible to calculate the DC-link power for display in the oscilloscope.

In MP2199.x you assign the respective axis or spindle to one of the two power supply modules defined in MP2198.x.

MP2100.x Type of axis power modules

Input: Name from file <Motor.amp>

MP2196.x Identifier for power supply module in MP2198.x

Input: P, Q, R, T

MP2198.x Type of power supply module

Input: Name from file <supply.spy>
Default setting: Empty string

MP2199.x Assignment of the axes to the supply modules

Input: 0: The axis/spindle is assigned to the UV in MP2198.0
1: The axis/spindle is assigned to the UV in MP2198.1

MP2200.x Motor model

Input: Name of the selected motor (entered by the iTNC)

MP2202.x Overwrite "Line count" from the motor table

Input: *: Entry from the motor table active
0: No speed encoder (volts-per-hertz control mode)
1 to 999 999

MP2204.x Overwrite "Counting direction" from the motor table

Input: *: Entry from the motor table active
+: Positive counting direction
–: Negative counting direction

MP2206.x Overwrite "Type of encoder" from the motor table

Input: *: Entry from the motor table active
0: No speed encoder (volts-per-hertz control mode)
1: Incremental rotary encoder with Z1 track
2: Absolute rotary encoder with EnDat interface (aligned)
3: Absolute linear encoder with EnDat interface
4: Linear motor with one reference mark (CC424(B))
5: Absolute rotary encoder with EnDat interface (not aligned)
6: Incremental rotary encoder without Z1 track
7: Incremental rotary encoder with distance-coded reference marks (nonaligned)
8: Incremental linear encoder with distance-coded reference marks (not aligned)
9: Aligned rotary encoder with EnDat 2.2 interface
10: Nonaligned rotary encoder with EnDat 2.2 interface
11: Linear encoder with EnDat 2.2 interface
12: EnDat2.2 incremental, position encoder (e.g. LIP 200)
13 bis 18: Reserved
19: EnDat 2.2 incremental, speed encoder
(e.g. ERM + EIB)

**Entries in the power
module table
(inverter.inv)**

- **NAME:** Designation of the power module
- **PWM:** PWM frequency in [Hz] at which the power module is driven
- **S:** Switch position of the current sensor. Is required for the D series of HEIDENHAIN inverters (UM 1xx D) in order to use the higher currents named above at the lower PWM frequencies.
Input 0 or 1
- **I-MAX:** Maximum current of the inverter output in [A]
- **I-N:** Rated current of the inverter output in [A]
- **U-IMAX:** Current sensor voltage in [V] at I-MAX
- **I-N-DC:** Permissible continuous current in stationary rotating field or until F-DC is reached in [A]
- **T-DC:** Time constant, how long maximum current can be applied to a stationary synchronous motor in [s]
- **F-DC:** Lower motor cutoff frequency down to which the motor can be loaded with I-N-DC in [Hz]
- **T-AC:** Cycle duration for the duty cycle S6-40% in [s]
- **F-AC:** Motor frequency from which I-MAX is permissible in [s]
- **T-IGBT:** Protection time of the IGBTs in [s]
- **R-SENSOR:** Resistance of the current sensor. The entries in this column are maintained by HEIDENHAIN and need not be changed by the machine tool builder.
- **F-Limit:** Lower frequency limit

**Entries in the power
module table
(motor.amp)**

- Designation of power module (NAME)
- Maximum current (I-MAX) in A
- Rated current (I-N) in A at a PWM frequency of 5 kHz
- Current sensor voltage (U-IMAX) in V at I-MAX
- Permissible continuous current in stationary rotating field or until F-DC is reached (I-N-DC) in A
- Time constant, how long maximum current can be applied to a stationary synchronous motor (T-DC) in seconds
- Lower motor base frequency down to which the motor can be loaded with I-N-DC (F-DC) in Hz
- Cycle duration for the duty cycle S6-40%(T-AC) in seconds
- Motor frequency from which I-MAX is permissible (F-AC) in seconds
- Protection time of the IGBTs (T-IGBT) in seconds
- Rated currents with PWM frequencies of 3333 Hz, 4000 Hz, 5000 Hz, 6666 Hz, 8000 Hz and 10000 Hz (I-N-AC-3333, I-N-AC-4000, I-N-AC-5000, I-N-AC-6666, I-N-AC-8000, I-N-AC-10000) [A]



Entries in the motor table (motor.mot)

A value that is not equal to 0 in a corresponding machine parameter replaces the entry in the motor table.

- Motor model (TYPE)
 - UASM = Uncontrolled asynchronous motor (volts-per-hertz control mode)
 - SM = synchronous motor
 - ASM = asynchronous motor
 - LSM = linear motor
- Designation of motor (NAME)
- Operating mode (MODE)
- Rated current (I-N) in A
- Rated voltage (U-N) in V
- Rated speed (N-N) in min^{-1}
- Rated frequency (F-N) in Hz
- No-load voltage (U0) in V
- No-load current (I0) in A
- Stator resistance cold (R1) in $\text{m}\Omega$
- Rotor resistance cold (R2) in $\text{m}\Omega$
- Stator leakage reactance (XStr1) in $\text{m}\Omega$
- Rotor leakage reactance (XStr2) in $\text{m}\Omega$
- Magnetizing reactance (XH) in $\text{m}\Omega$
- Upper speed X-H characteristic (N-XH) in min^{-1}
- Threshold speed for field weakening (N-FS) in min^{-1}
- Maximum speed (N-MAX) in min^{-1}
- Factor for X-H characteristic (%-XH)
- Factor for stalling torque reduction (%-K)
- Number of pole pairs (PZ)
- Temperature coefficient (TK) in Ω/K
- Line count of the motor encoder (STR)
- Encoder being used (SYS)
 - 0 = No speed encoder (volts-per-hertz control mode)
 - 1 = Incremental rotary encoder with Z1 track
 - 2 = Absolute rotary encoder with EnDat interface (aligned¹)
 - 3 = Absolute linear encoder with EnDat interface
 - 4 = Incremental linear encoder
 - 5 = Absolute rotary encoder with EnDat interface (not aligned¹)
 - 6 = Incremental rotary encoder without Z1 track (one reference mark)
 - 7 = Incremental rotary encoder with distance-coded reference marks (not aligned¹)
 - 8 = Incremental linear encoder with distance-coded reference marks (not aligned)

1. See "Field orientation" on page 1019.

- Counting direction of the motor encoder (DIRECT.)
- Maximum temperature (T-MAX) in °C
- Maximum current (I-MAX) in A
- Rated power output (P-N) in W
- Motor mass moment of inertia (J) in kgm²
- Inductance of the series reactor (L) in μH
- Thermal time constant for direct current (T-DC) in seconds
- Lower thermal limit frequency (F-DC) in Hz
- Thermal time constant for alternating current (T-AC) in seconds
- Upper thermal limit frequency (F-AC) in Hz; above this frequency, the maximum current I-MAX applies
- Thermal time constant for winding (Tth1) in s
- Thermal resistance between winding and lamination (Rth1) in K/W
- Thermal time constant for lamination (Tth2) in s
- Thermal resistance between lamination and coolant (Rth2) in K/W
- Stall torque (M0) in [Nm]
The entry is currently not yet evaluated, however.
- Torque (Mmax) at Imax in [Nm]
The entry is currently not yet evaluated, however.
- Direct inductance (Ld) in [mH]
The entry is currently not yet evaluated, however.
- Quadrature inductance (Ld) in [mH]
The entry is currently not yet evaluated, however.
- Recommended PWM frequency of the motor (Fpwm) in [Hz]
The entry is currently not yet evaluated, however.
- Current controller factor (Kp) in [V/A] calculated from the motor data
The entry is currently not yet evaluated, however.
- Current controller factor (Ki) in [V/A/s] calculated from the motor data
The entry is currently not yet evaluated, however.
- Type of temperature sensor (TempSens)
The entry is currently not yet evaluated, however.
- Iron loss resistance [Rfe] in [ohms]
The entry is currently not yet evaluated, however.



Table of power supply modules (supply.spy)

- Designation (NAME)
- Type of power supply module (E-R)
 - 0: No regenerative module
 - 1: Regenerative module
- Rated power output (P-N) in W
- Peak power (S6-40) (P-S6-40) in W
- Peak power for (0.2 s) (P-MAX02) in W
- DC-link voltage (UZ) in V
- Ratio of measuring voltage / UZ (UZ-AN) in V/V
- Ratio of measuring voltage / IZ (IZ-AN) in V/V
- Status signals (STATUS-SIG)
 - Bit 0:AC-FAIL
 - Bit 1:POWERFAIL
 - Bit 2:TEMP
 - Bit 3:READY
 - Bit 4:Reserved
 - Bit 5:Reserved
 - Bit 6: Reserved
 - Bit 7: Reserved
- Proportional factor of D controller (P-D)
- Integral factor of D controller (P-D)
- Proportional factor of Q controller (P-Q)
- Integral factor of Q controller (I-Q)
- PWM frequency (FREQ) in HZ

Reasons for using a series reactor:

Asynchronous motor

- Minimum leakage inductance of drive

The total leakage inductance L_{Total} of the drive is formed from the leakage inductance of the motor $L_{LeakMotor}$ and the inductance of the series reactor $L_{Reactor}$

$L_{Total} = L_{LeakMotor} + L_{Reactor}$ with:

$$L_{LeakMotor} = \frac{(X_{Str1} + X_{Str2}) \cdot 1000}{2 \cdot \pi \cdot f_N}$$

- X_{Str1} : Stator leakage reactance (see X_{Str1} in motor.mot) [mΩ]
- X_{Str2} : Rotor leakage reactance (see X_{Str2} in motor.mot) [mΩ]
- f_N : Rated frequency of the motor (see F-N in motor.mot) [Hz]
- $L_{LeakMotor}$: Leakage inductance of the motor [μH]

The total leakage inductance L_{Total} must not fall below a specified minimum value. This also depends on the PWM frequency f_{PWM} , the magnetizing current I_0 of the motor, and the DC-link voltage U_Z , and is calculated as follows:

For $I_0 < 26$ A:

$$L_{Total} = \frac{700 \mu\text{H} \cdot 5000 \text{ Hz} \cdot U_Z}{f_{PWM} \cdot 600}$$

For $I_0 \geq 26$ A:

$$L_{Total} = \frac{700 \mu\text{H} \cdot 5000 \text{ Hz} \cdot U_Z \cdot 26 \text{ A}}{600 \cdot I_0 \cdot f_{PWM}}$$

- L_{Total} : Total leakage inductance of the drive [μH]

This way the value for the series reactor can be determined as:

$$L_{Reactor} = L_{Total} - L_{LeakMotor}$$

If the result is a negative value, HEIDENHAIN does not require the use of a series reactor. It is possible, however, that there are other reasons for using a series reactor, such as the motor manufacturer's specifications.

- Impermissible temperature of the bearings

High harmonic losses in the rotor lead to excessively high bearing temperatures. One possible remedy, besides increasing the PWM frequency, is the use of a series reactor. Here there is no universally valid formula.

The amount of inductance results from the measurement of the bearing temperature.

- Limitation of dU/dt

Because of the strain placed on the motor insulation, the motor manufacturer can prescribe the use of a series reactor. The strain placed on the motor insulation results from the voltage build-up dU/dt at the inverter output.

For HEIDENHAIN inverters the voltage build-up is $dU/dt = 6000 \text{ V}/\mu\text{s}$.

Synchronous motor

- Limit of current during field weakening operation

To make sure that the demagnetization current in the field weakening range does not exceed a permissible value of I_{DMax} , the total leakage inductance $L_{Total} = L_{Motor} + L_{Reactor}$ must have the following value:

$$L_{Total} = 10^6 \cdot \frac{\frac{U_0}{\sqrt{3}} \cdot \frac{n_{Max}}{n_N} - \frac{UZ}{\sqrt{6}}}{\frac{\pi}{30} \cdot I_{DMax} \cdot p_Z \cdot n_{Max}}$$

- U_0 : No-load voltage of the motor (see U_0 in motor.mot) [VeffL-L]
- n_N : Rated speed of the motor (see N-N in motor.mot) [min-1]
- n_{Max} : Maximum speed of the motor (see N-MAX in motor.mot) [min-1]
- UZ : DC-link voltage (see MP2190) [V]
- p_Z : Number of pole pairs (see PZ in motor.mot)
- I_{DMax} : Maximum demagnetization current [Aeff]
- L_{Total} : Total leakage inductance of the drive [μ H]
- L_{Motor} : Total leakage inductance of the motor [μ H]
- I_N : Reactor current of the motor [Aeff]

The maximum demagnetization current is the no-load current of the motor during maximum speed, where $I_{DMax} \leq I_N$.

If the maximum demagnetization current is not known, it can be approximated by:

$$L_{DMax} = I_0 + \frac{I_N - I_0}{n_N} \cdot n_{Max}$$

- I_N : Rated current of the motor [Aeff]
- I_0 : Continuous standstill current of the motor (see I_0 in motor.mot) [Aeff]

and $I_0 \geq I_N$

- Impermissible temperature of the bearings

High harmonic losses in the rotor lead to excessively high bearing temperatures. One possible remedy, besides increasing the PWM frequency, is the use of a series reactor.

Here there is no universally valid formula.

The amount of inductance results from the measurement of the bearing temperature. Furthermore it must be noted that, especially with synchronous spindles, the motor manufacturer requires a minimum PWM frequency and minimum inductance of the series reactor depending on this PWM value.

In this context it is important that the heating of (and possible damage to) the bearing and/or rotor cannot be detected by monitoring the motor winding temperature



Note

If required, use the speed-dependent PWM frequency switchover in order to minimize harmonic loss, see page 1082

■ Compliance with the maximum permissible short-circuit current

The short-circuit current I_K is calculated from:

$$I_K = \frac{U_0}{\sqrt{3} \cdot (X_{Str1} + X_H + 2 \cdot \pi \cdot f_N \cdot L_{Drossel})}$$

- U_0 : Mesh no-load voltage (see U_0 in motor.mot) [VeffL-L]
- X_{Str1} : Stator leakage reactance (see X_{Str1} in motor.mot) [Ω]
- X_H : Magnetizing reactance (see X_H in motor.mot) [Ω]
- f_N : Rated frequency of the motor (see f_N in motor.mot) [Hz]

This must not exceed the maximum current: $I_K \leq I_{Max}$

Summary: Advantages and disadvantages of the series reactor:

■ Advantages

- Reduction of current harmonics
- Reduction of losses in the motor (important for bearing temperature)
- Reduction of the voltage build-up dU/dt at the motor winding
- Reduction of the demagnetization current with synchronous motors in the field weakening range
- Reduction of the short-circuit current with synchronous motors

■ Disadvantages

- Reduction of motor power at the voltage limit (field weakening range)
- Heating of the series reactor
- Space requirement



Note

If a series reactor is installed later, the current controller must be readjusted.

If a series reactor is used with another inductance than that shown in the motor table, the inductance value of the series reactor must be entered in MP2208.

The series reactor must fulfill the following specifications:

- The calculated inductance is required per phase.
There are manufacturers who do not indicate the inductance per phase, but rather the mesh inductance (calculated from V_{effL-L} and I_{eff}).
- The series reactor must be designed for the requirements of the motor. (I_N at f_N and I_N (NMax) at f_{Max} must be possible)
- The inductance must comply over the motor's operating range

MP2208.x Inductance of the series reactor

Input: * = Entry from the motor table active
Value of the series reactor in $[\mu H]$

Determining data for synchronous motors

The motor data for synchronous motors are entered in the motor table after some conversions using the values from the motor data sheet of the respective manufacturer (here using the example of a SIEMENS motor).

Values in the HEIDENHAIN motor table	Values from the motor data sheet
TYPE: SM	Permanently excited synchronous motor
NAME: 1FT6044-4AF7	1FT6044-4AF7
MODE: 0	
Rated current I-N in [A _{eff}] winding I-N: 2.9	Data sheet value I _{rated} (100 K) I _N = 2.9 A
Rated voltage U-N in [V _{eff}] interlinked U-N: 341	Calculation from data sheet values n _{rated} , k _E , R _{Str} , I _{rated} (100 K), L _D : $U-N = \sqrt{3} \cdot \sqrt{(U_e + U_r)^2 + U_x^2}$ $U_e = (n_{rated} / 1000) \cdot (k_E / \sqrt{3})$ $U_e = (3000 / 1000) \cdot (108 / \sqrt{3})$ $U_e = 187.06 \text{ V}_{eff L,N}$ $U_r = R_{Str} \cdot I_{rated} (100 K)$ $U_r = 3.05 \cdot 2.9$ $U_r = 8.85 \text{ V}_{eff L,N}$ $U_x = 2 \cdot \pi \cdot (n_{rated} / 60) \cdot PZ \cdot (LD / 1.5) \cdot I_{rated} (100 K)$ $U_x = 2 \cdot \pi \cdot (3000 / 60) \cdot 2 \cdot (0.016 / 1.5) \cdot 2.9$ $U_x = 19.44 \text{ V}_{eff L,N}$ $U-N = \sqrt{3} \cdot \sqrt{(187,06 + 8,85)^2 + 19,44^2}$ $U-N = 341 \text{ V}_{eff L,L}$
Rated speed N-N in [min ⁻¹] N-N: 3000	Data sheet value n _{rated} N-N = 3000 min ⁻¹
Rated frequency F-N in [Hz] F-N: 100	Calculation from data sheet value n _{rated} F-N = (n / 60) · PZ F-N = (3000 / 60) · 2 F-N = 100 Hz
No-load voltage at rated speed U0 in [V _{eff}] interlinked U0: 324	Calculation from data sheet value n _{rated} and k _E U0 = (n _{rated} / 1000) · k _E U0 = (3000 / 1000) · 108 U0 = 324 V _{eff L,L}
No-load current I0 in [A _{eff}] winding I0: 3	Data sheet value I ₀ (100 K) I0 = 3 A _{eff}



Values in the HEIDENHAIN motor table	Values from the motor data sheet
Stator resistance at 20 °C R1 in [mΩ] at 20° C R1: 3050	Data sheet value R_{Str} R1 = 3050 mΩ
Rotor resistance at 20 °C R2 in [mΩ] at 20° C R2: 0	
Stator leakage reactance at F-N Xstr1 in [mΩ] Xstr1: 0	If nothing given, then zero.
Rotor leakage reactance at F-N Xstr2 in [mΩ] Xstr2: 0	
Magnetizing reactance XH for F-N at rated conditions XH in [mΩ] XH: 10052	Calculation from data sheet value L_D , n_{rated} $XH = 2 \cdot \pi \cdot (n_{rated} / 60) \cdot PZ \cdot L_D$ $XH = 2 \cdot \pi \cdot (3000 / 60) \cdot 2 \cdot 0.016$ XH = 10052 mΩ
Desaturation speed N-XH in [min ⁻¹] N-XH: 0	
Rotational speed of beginning field weakening range N-FS [min ⁻¹] N-FS: 0	
Maximum speed (mechanical) N-MAX in [min ⁻¹] N-MAX: 7700	Data sheet value n N-MAX = 7700 min ⁻¹
Saturation factor %-XH in % %-XH: 100	
Stalling torque reduction factor %-K in % %-K: 100	
No. of pole pairs (half pole no. of motor) PZ PZ: 2	From data sheet value or model designation
Temperature coefficient of the stator winding TK in 1/K TK: 0.004	
Line count of the speed encoder: STR STR: 2048	From the mounted speed encoder
Type of encoder SYS: 1	Aligned incremental encoder with one reference mark: 1 Aligned absolute encoder with EnDat interface: 2
Counting direction DIRECT. DIRECT.: +	

Values in the HEIDENHAIN motor table	Values from the motor data sheet
Max. temperature of motor at temperature sensor T-MAX in [°C] T-MAX: 150	
Maximum motor current I-MAX in [A _{eff}] winding I-MAX: 7.5	Data sheet value I _{max} I-MAX = 7.5 A _{eff}
Rated power P-N in [W] P-N: 1351	Calculation from data sheet value n _{rated} and M _{rated} (100 K) P-N = $2 \cdot \pi \cdot (n_{\text{rated}} / 60) \cdot M_{\text{rated}}$ P-N = $2 \cdot \pi \cdot (3000 / 60) \cdot 4.3$ P-N = 1351 W
Motor mass moment of inertia J in [kgm ²] J: 0.00051	Data sheet value J _{mot} J = 0.00051 kgm ²
Inductance of the series reactor L in [mH] L: 0	
Thermal time constant for direct current T-DC in [s] T-DC: 2400	Calculation from data sheet value T _{th} T-DC = T _{th} · 60 T-DC = 40 · 60 T-DC = 2400
Lower thermal cutoff frequency F-DC in [Hz] F-DC: 0	
Thermal time constant for alternating current T-AC in [s] T-AC: 2400	Calculation from data sheet value T _{th} T-AC = T _{th} · 60 T-AC = 40 · 60 T-AC = 2400
Upper thermal cutoff frequency F-AC in [Hz] F-AC: 0	

6.14.2 Field orientation

Field orientation with absolute encoder (EnDat)

As soon as the absolute position of the encoder has been read, the absolute position and determined field angle are assigned to each other. The field angle is assigned to the zero position of the encoder.

Field orientation via encoder with Z1 track

After switching on the control, the motor orients itself (rough orientation) via the Z1 track of the encoder. The drive is ready for operation after this procedure. The field angle is determined and assigned as soon as the reference mark is traversed during the first motor motion.

Field orientation via the PLC

If a synchronous drive is used along with an encoder, and the field angle cannot be determined directly via the encoder, there is no assignment between the encoder and rotor magnets.

The Field Orientation function, which must be carried out once per switch-on procedure, is used by the iTNC 530 to determine the association between the encoder and rotor magnets (field angle), and stores this information internally.



Note

Please pay attention to the following notes for determining the field angle:

- HEIDENHAIN generally recommends using either encoders with absolute values or encoders with Z1 tracks. With these encoders the field angles are available immediately after switch-on, so no additional efforts are necessary to determine them.
- The field angle can be determined only if the current controller is already adjusted!

Regarding the **motor.mot** motor table, the field orientation must be performed for the following drives:

- Linear motor with absolute encoder with EnDat interface (**SYS = 3**)
- Synchronous or torque motor with non-aligned rotary encoder with EnDat interface (**SYS = 5**)
- Synchronous or torque motor with incremental rotary encoder without Z1 track (**SYS = 6**); one reference mark per revolution
- Synchronous or torque motor with incremental rotary encoder with distance-coded reference marks (**SYS = 7**)
- Linear motor with incremental linear encoder with distance-coded reference marks (**SYS = 8**)

Methods of the iTNC for determining the field angle

The description of the field angles by the iTNC 530 HSCI, see "Field orientation" on page 1108.

6.14.3 Preparation

Proceed as follows:

- ▶ Check the wiring against the grounding diagram and the safety concept (See "Basic Circuit Diagram" at the end of Chapter 3).
- ▶ Check the control-is-ready function.
See "Emergency stop monitoring" on page 930.
- ▶ Check the emergency stop circuit by pressing the emergency stop buttons and the emergency stop limit switch.
- ▶ Select the current machine parameter file. Determine input values using the documentation on hand. Enter temporary values for machine parameters that must be optimized during commissioning.
- ▶ Create a PLC program for interfacing the control to the machine (use the PLC development software PLCdesign).
- ▶ Ensure that in the system file OEM.SYS the instruction **PLCMAIN=** refers to the current PLC program.

DC-link voltage

The iTNC 530 uses the DC-link voltage to calculate the maximum motor voltage:

- ▶ Enter in MP2190.x the level of the DC-link voltages at the power modules. Machine parameters 2190.x can be used to overwrite the entries of the SUPPLY.SPY table. The table is stored in the PLC:\mp\ directory.

MP2190.x DC-link voltage U_z

Input: 0 to 3000 [V]
 *: Entry from the power supply module table
 HEIDENHAIN inverters:
 Regenerative: 650 V
 Non-regenerative: 565 V

Temporary input values

- ▶ Enter the following temporary input values when you begin commissioning:

MP	Temporary input value	Meaning
MP20.0	%00000000000000	Monitoring the absolute position of the distance-coded reference marks
MP1030.x	0.01	Positioning window
MP1054.x	?	Linear distance of one motor revolution (depends on the machine)
MP1090.x	1	Maximum permissible jerk on the tool path
MP1092	<greater than rapid traverse>	Feed rate threshold from which MP1090.1 becomes effective
MP1095	0	Single filter
MP1096.x	0	Position nominal value filter off
MP1099.0	5	Minimal filter order for single filters
MP1099.1	3	Minimal filter order for double filters
MP1110.x	2.0	Standstill monitoring



MP	Temporary input value	Meaning
MP1140.x	0.03	Movement monitoring (for digital axes the minimum value is entered)
MP1144.x	0	Motion monitor for position and speed
MP1340.x	0	No evaluation of reference marks
MP1396.x		Feedback control with velocity semi-feedforward
MP1410.x	0.5	Position monitoring in operation with velocity feedforward control (erasable)
MP1420.x	2	Position monitoring during operation with velocity feedforward control (emergency stop)
MP1510.x	1	k_v factor for velocity feedforward control
MP1521.x	0	Transient response during acceleration and deceleration
MP1710.x	50	Position monitoring in operation with following error (erasable)
MP1720.x	50	Position monitoring during operation with following error (emergency stop)
MP1810.x	1	k_v factor for control with following error
MP1820.x	1	Multiplier for the k_v factor
MP1830.x	100	Characteristic curve kink point
MP2220.x	%0000100	Monitoring functions
MP2420.x	0	Proportional factor of the current controller
MP2430.x	0	Integral factor of the current controller
MP2500.x	0.5	Proportional factor of the speed controller
MP2510.x	20	Integral factor of the motor speed controller (for axes with holding torque, e.g. vertical axes, the value 1 must be entered because otherwise the axis drifts away)
MP2512.x	0	Limit of integral factor of the speed controller
MP2520.x	0	Differential factor of the speed controller
MP2530.x	0	PT ₂ element of speed controller
MP2540.x	0	Band-rejection filter damping
MP2550.x	0	Band-rejection filter for center frequency
MP2600.x	0	Acceleration feedforward control
MP2602.x	0	IPC time constant T ₁
MP2604.x	0	IPC time constant T ₂
MP2606.x	0	Following error in the jerk phase
MP2610.x	0	Friction compensation at low speed
MP2612.x	0	Delay of friction compensation
MP2620.x	0	Friction compensation at rated speed
MP2630.x	0	Holding current

Additional temporary input values for the spindle

- ▶ Enter the following additional temporary input values when you begin commissioning the spindle:

MP	Temporary input value	Meaning
MP3010.x	3 to 8	Output of speed, gear range
MP3020	991	Speed range
MP3411.x	1.999	Ramp gradient
MP3412.x	1	Multiplier for MP3411.x
MP3415.x	0	Overshoot behavior
MP3420	1	Positioning window
MP3440.x	1	k_v factor



Note

C-axis operation must be deselected for commissioning, meaning that no identical PWM outputs may be entered in MP120.x and in MP121.x.

Operating mode switchover

During commissioning you can switch between operating mode 0 and operating mode 1 with the CONNECT STAR DELTA soft key. With Module 9168 you can interrogate the current settings in the PLC. You can switch the motor using PLC outputs and activate the corresponding machine parameters with Module 9163:

- ▶ Perform the adjustment for operating mode 0 and operating mode 1.
 - If you do not use operating mode 1, set the corresponding machine parameters to zero.

6.14.4 Commissioning digital control loops with TNCopt

In order to commission digital control loops with TNCopt, you must carry out preparations on the control as described in this chapter. Also pay attention to the notes in the documentation for TNCopt. Functions not supported by TNCopt must be commissioned manually (see "Commissioning of digital axes" on page 1026 and see "Commissioning the digital spindle" on page 1055).

The PC software TNCopt will also be in continuous development. You will find a description of this software and a function overview in the help files of the software.

When commissioning with TNCopt, the machine axes must be moved. For safety purposes, a function of this type should be enabled on the control. The entry **TNCOPT.LOCKSOFTKEYVISIBLE = YES** in OEM.SYS makes the soft key **TNCOPT OFF ON** visible after pressing the MOD key. It is used to enable such functions. As a default, the soft key is always set to **OFF** when the control is started up.

Current controller

- ▶ Switch on the control.
- ▶ Do **not** acknowledge the **Power Interrupted** message. In the **Programming and Editing** mode of operation, use the MOD key to enter the code number 688379 or 807667 (followed by the DIAGNOSIS and DRIVE DIAGNOSTICS soft keys) to switch to the **Drive Diagnostics** mode of operation.
- ▶ Press the I CONTROL soft key.
- ▶ Acknowledge the **Power Interrupted** message in the **Manual** mode of operation.
- ▶ Switch on the control voltage.
- ▶ Switch to the **Oscilloscope** mode of operation.
- ▶ Press the START STEP soft key.
- ▶ Commission the current controller with TNCopt.

Speed controller

- ▶ Position the axis or spindle to be optimized at a location where it can be commissioned safely.
- ▶ Set `MP1340.x = 0` to deselect evaluation of the reference marks.
- ▶ Ensure that the loaded PLC program fulfills the following conditions:
 - Position control loop is opened, because the NC opens the position control loop only during the step function. If the position controller is not optimized, error messages appear if the position control loop is closed.
 - Enable the drive controller
 - NC stop inactive
 - Axis direction buttons active
 - Axes clamped
- ▶ In the **Programming and Editing** mode of operation, use the MOD key to enter the code number 688379 to switch to the **Drive Diagnostics** mode of operation.
- ▶ Press the OSCI soft key.
- ▶ Set the following values in the **Oscilloscope**:
 - Output: Step**
 - Feed rate: 100**
 - Channel 1: I nominal**
 - Trigger: Free run**
- ▶ Press the OSCI soft key to switch the curve representation.
- ▶ Press the START soft key to start recording.
- ▶ Set the feed-rate override potentiometer to 100%.
- ▶ Commission the speed controller with TNCopt.

Feedforward

- ▶ Position the axes to a location where the feedforward functions can be commissioned safely.
- ▶ Set the datum for the affected axes at this location to "0".
- ▶ In MP1060.x, set the acceleration to 0.5.
- ▶ In the **Program Run, Full Sequence** mode of operation, select the NC program `FF_*.H` (* = axis to be optimized) from the `TNC:\TNCOPT` folder.



Note

TNCopt generates the NC programs `FF_*.H` with the feed-rate values 6000 and 200. The larger feed-rate value should equal the machine's highest machining feed rate. Adjust them if necessary. The lower feed-rate value must not be changed.

- ▶ Set the feed-rate override potentiometer to 100%.
- ▶ Commission the feedforward functions with TNCopt.



Reversal spikes

- ▶ Position the axes to a location where the reversal-spike compensation can be commissioned safely.
- ▶ Set the datum for the affected axes at this location to "0".
- ▶ In MP1060.x, set the acceleration to 0.5.
- ▶ In the **Program Run, Full Sequence** mode of operation, select the NC program **CIR_*.H** (* = axis to be optimized; # = second control loop, defines the plane) from the **TNC:\TNCOPT** folder.
- ▶ Set the feed-rate override potentiometer to 100%.
- ▶ Commission the reversal-spike compensation with TNCopt.

IPC and k_V factor

- ▶ Position the axes to a location where the IPC and k_V factor can be commissioned safely.
- ▶ Set the datum for the affected axes at this location to "0".
- ▶ Set the k_V factors in MP1510.x to 1.
You can also start with a higher value for MP1510, as long as the value is safely below the oscillation limit.
- ▶ In the **Program Run, Full Sequence** mode of operation, select the NC program **IPC_*.H** (* = axis to be optimized) from the **TNC:\TNCOPT** folder.
- ▶ Set the feed-rate override potentiometer to 100%.



Note

Ensure that the machine parameters for the IPC (MP2602.x, MP2604.x and MP2606.x) have been set to 0.

- ▶ Adjust the IPC and k_V factor with TNCopt.

6.14.5 Commissioning of digital axes

The iTNC must be adjusted in sequence for the:

- Current controller
- Speed controller
- Position controller

The signals that you need are recorded with the integral oscilloscope.

Current controller

Use the integrated oscilloscope to adjust the current controller. The speed and position control loops are open when you adjust the current controller. You must therefore activate a special PLC commissioning program:

- ▶ Enter the name of this PLC program in the OEM.SYS file with the **PLCPWM =** instruction.

It suffices to program an EM (end module).

The drive must be enabled externally and the iTNC needs the "ready" signal.

As soon as the PLC program defined with **PLCPWM =** is active, you can use Module 9168 to interrogate the commissioning status.

Module 9168 Interrogate the commissioning status

Call:

CM 9168

PL D <Status>

-1: Commissioning not active or as yet no axis is selected

Axis-specific: Dependence on MP4020 bit 14

Spindle – spindle operating mode

0: Operating mode 0

1: Operating mode 1

PLC module 9168 is a special case. This PLC module issues an error message when the PLC module interface is active (MP4020 bit 14) and can no longer be used. Please use PLC Module 9065 (status of HEIDENHAIN power supply unit) instead of PLC Module 9168.

Adjusting the current controller:

- ▶ Switch on the control.
- ▶ Do **not** acknowledge the **Power Interrupted** message. In the **Programming and Editing** mode of operation, use the MOD key to enter the code number 688379 or 807667 (followed by the DIAGNOSIS and DRIVE DIAGNOSTICS soft keys) to switch to the **Drive Diagnostics** mode of operation.
- ▶ Press the I CONTROL soft key.
- ▶ In the **Manual** mode of operation, acknowledge the **Power Interrupted** message.
- ▶ Use the CHOOSE AXIS soft key in the oscilloscope to select the axis to be adjusted.
- ▶ With the FACTOR P/I soft key, select the I factor and set MP2430.x = 0.
- ▶ With the FACTOR P/I soft key, select the P factor.
- ▶ Calculate the starting value of the P factor with the following formula:

$$\text{Starting value} = \frac{100\,000 \cdot L}{T_a}$$

T_a	$f_{\text{PWM}} \text{ (MP2180.x)}$
150	3 333 Hz
120	4 166 Hz
100	5 000 Hz
75	6 666 Hz
60	8 333 Hz
50	10 000 Hz

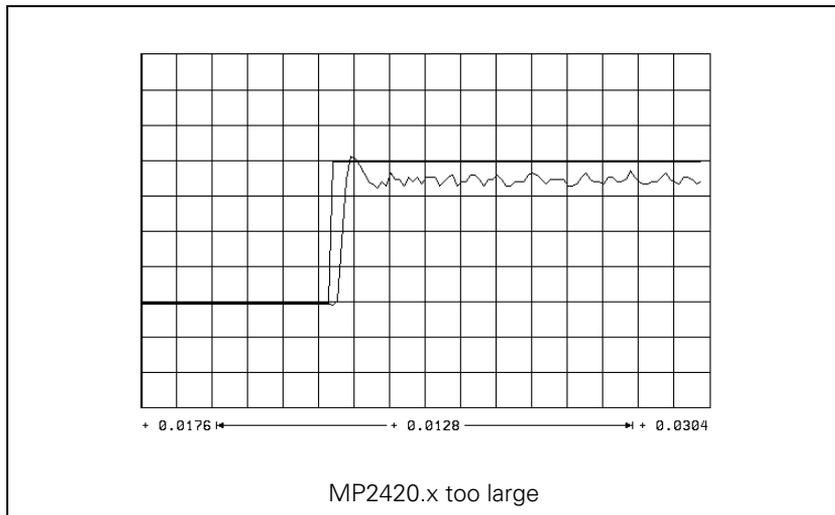
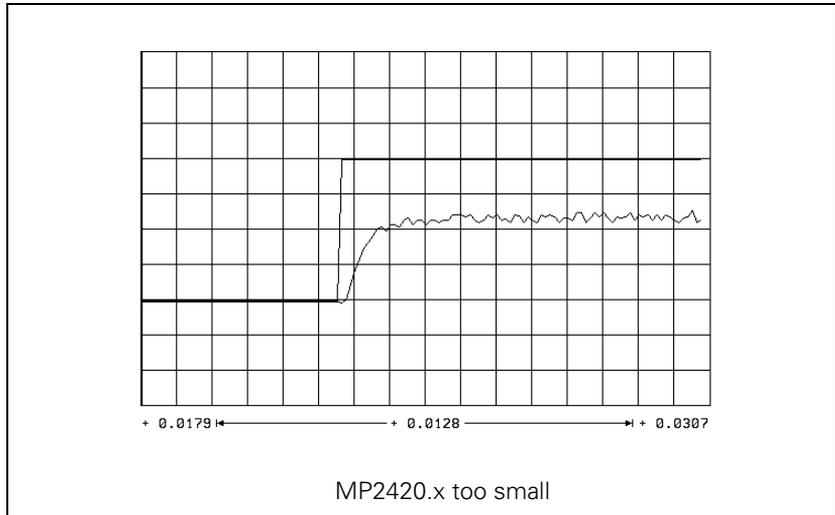
$$\text{Synchronous motor: } L = \frac{XH}{2 \cdot \pi \cdot (F-N) \cdot 1000}$$

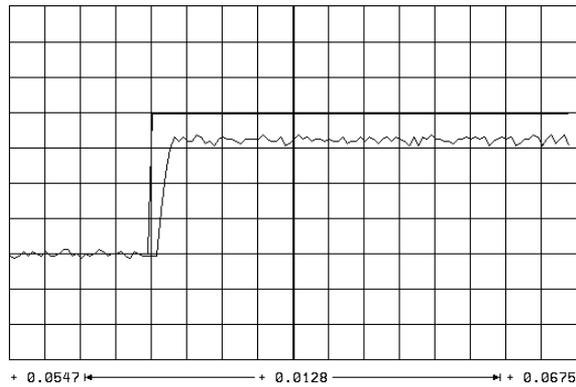
$$\text{Asynchronous motor: } L = \frac{X\text{Str1} + X\text{Str2}}{2 \cdot \pi \cdot (F-N) \cdot 1000}$$

The values for XH (magnetizing reactance), F-N (rated frequency), XStr1 (stator leakage reactance) and XStr2 (rotor leakage reactance) can be found in the motor table. Switch to the editing mode of the motor table (APPEND MOTOR soft key).

The values for XH, XStr1 and XStr2 are specified in [mΩ] in the motor table. Use these values in the formulas. The formula already contains the conversion factor.

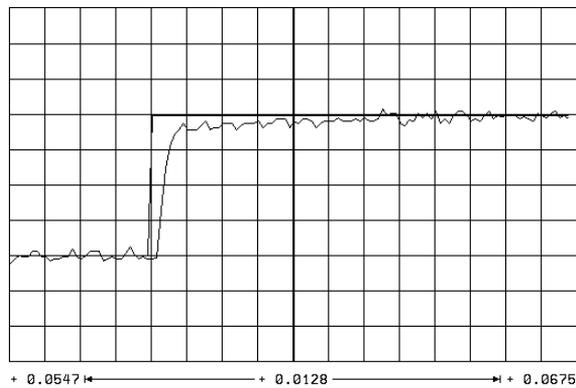
- ▶ Set this P factor (MP2420.x) with the ↑ soft key.
- ▶ Press the START STEP soft key.
This sends a step function to the current controller and measures the step response. The height and length of the step function are automatically calculated by the iTNC.
- ▶ With the ↑ soft key, increase the P factor (MP2420.x) step by step until just barely no undershoot is visible.



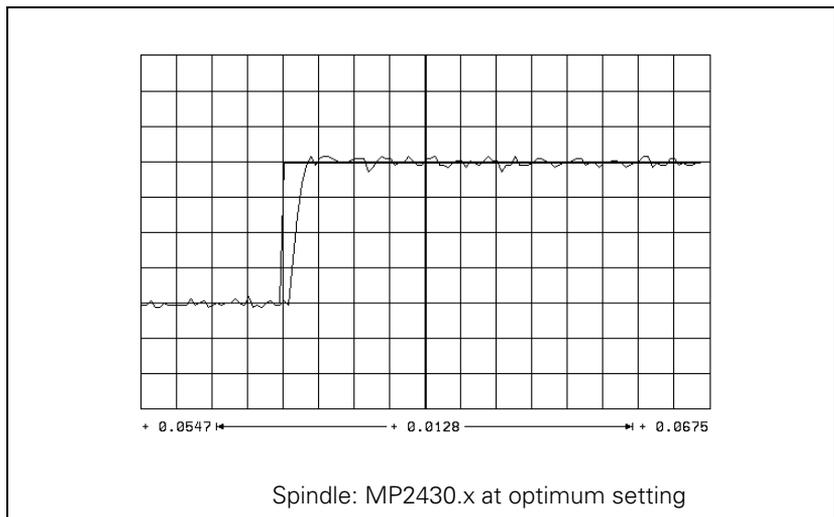
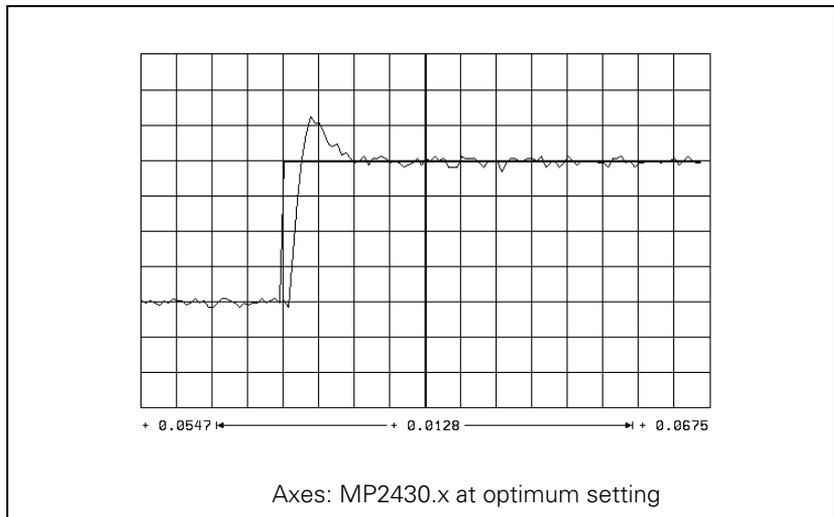


MP2420.x at optimum setting

- ▶ Save this value with the STORE MP2420.x soft key.
- ▶ With the FACTOR P/I soft key, select the I factor.
- ▶ With the ↑ soft key, increase the I factor (MP2430.x) step by step until
 - Spindle: Just barely no overshoot is visible, and so that the nominal value is reached as quickly as possible (short rise time).
 - Axes: You see an overshoot but no undershoot.



MP2430.x too small



- ▶ Save this value with the STORE MP2430.x soft key.
- ▶ Press the END key to exit the **Commission Current Controller** mode.

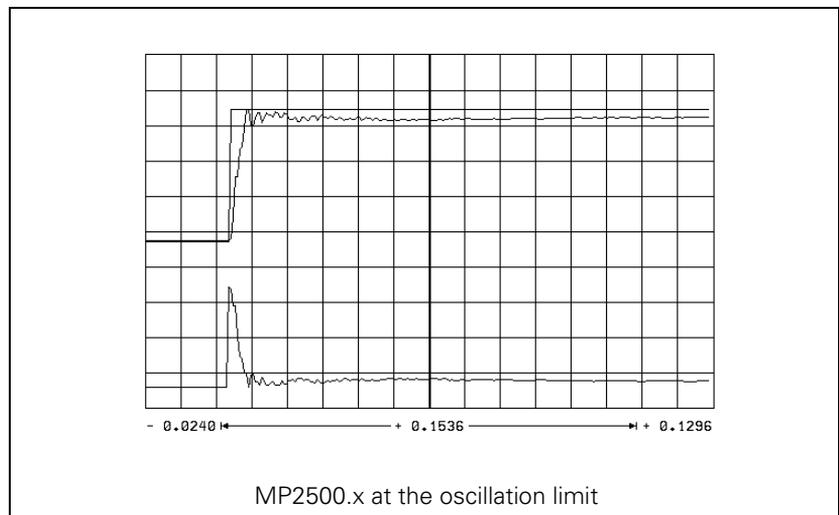
If the overall result of commissioning a servo drive or an axis is unsatisfactory, these recommendations do not need to be strictly followed. This may mean that

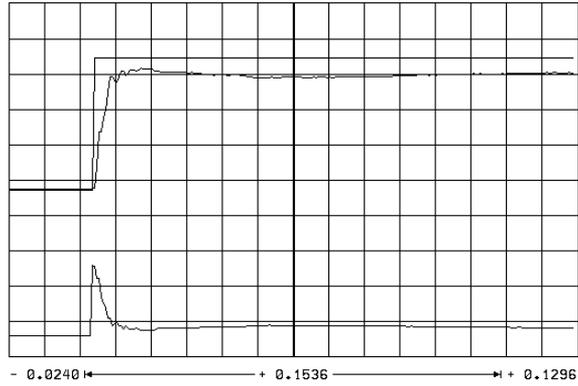
- Adjustment with overshoot is used to increase the current controller bandwidth.
- A higher PWM frequency is selected to increase the current controller bandwidth.
- The current controller bandwidth is reduced.

Speed controller

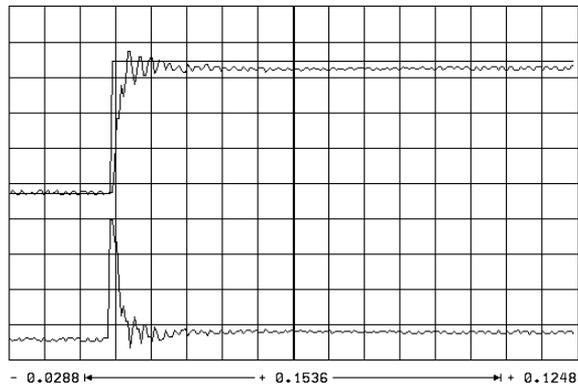
Adjusting the speed controller:

- ▶ Deselect "Pass over reference point" by setting MP1340 = 0.
- ▶ Ensure that the loaded PLC program fulfills the following conditions:
 - Position control loop is opened (W1038/W1128/W1040/W1132), because the NC opens the position control loop only during the step function. If the position controller is not optimized, error messages appear if the position controller is closed.
 - Servo drive controller is enabled (Module 9161)
 - NC stop is inactive, MP4560 = 1
 - Axis direction buttons active
 - Axes clamped
- ▶ In the MANUAL mode, use the oscilloscope function to select a step function (approx. 500 mm/min) that will not overdrive the speed controller, i.e. that does not limit I NOMINAL.
Display the nominal velocity value V (NOM RPM), the actual speed value V (ACT RPM) and the nominal current value (I NOMINAL).
- ▶ Activate the step function with the axis direction buttons.
- ▶ To change the machine parameters, press the MP EDIT soft key in the setup menu.
- ▶ Increase MP2500.x (P factor) until the oscillation limit is reached.





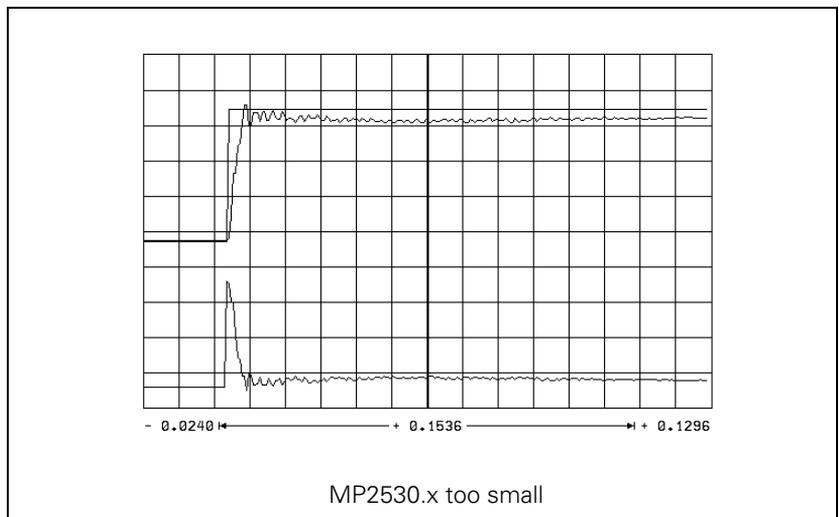
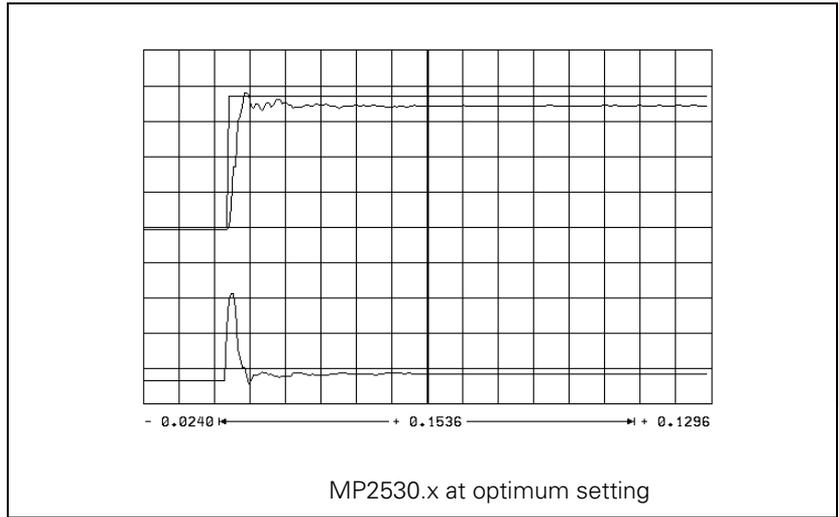
MP2500.x too small

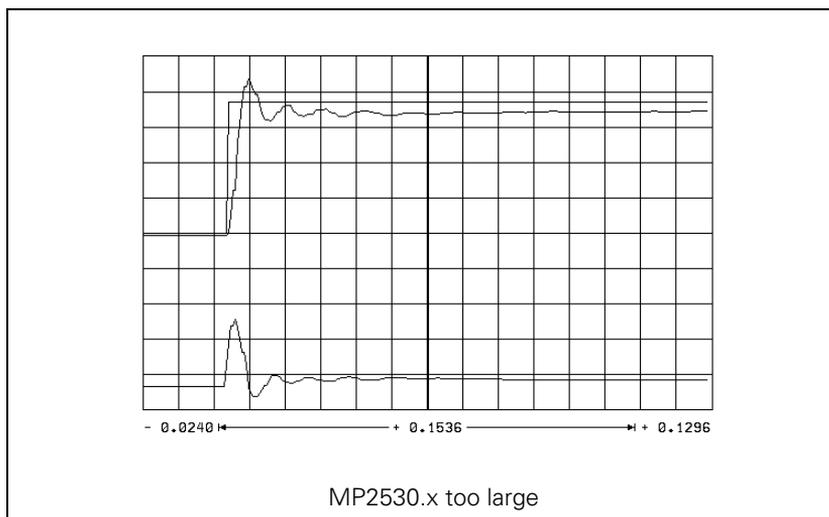


MP2500.x too large

► Input value for MP2500.x = <detetermined value> · 0.6

- ▶ Compensate high-frequency interference oscillations (> 400 Hz) with MP2530.x or MP2560.x.





Attention

You can use MP2520.x to compensate low-frequency oscillation (< 100 Hz) on axes with mechanical problems.

However, HEIDENHAIN recommends that you avoid using MP2520.x if possible.

Do not use for axes with belt drive!

You can also compensate disturbance oscillations with the band-rejection filter:

- ▶ Calculate the frequency of the oscillation and enter it in MP2550.x.
- ▶ Increase the band-rejection filter damping in MP2540.x until the interfering oscillation is minimized. Realistic input values: 3 to 9 dB.

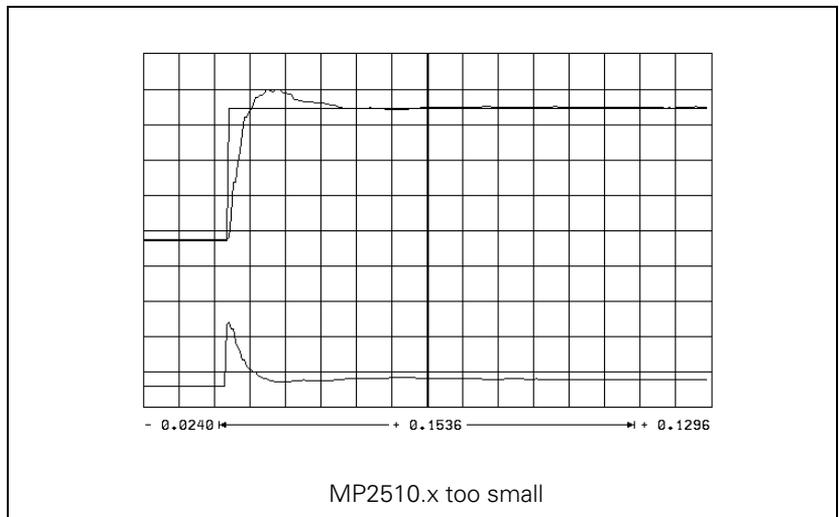
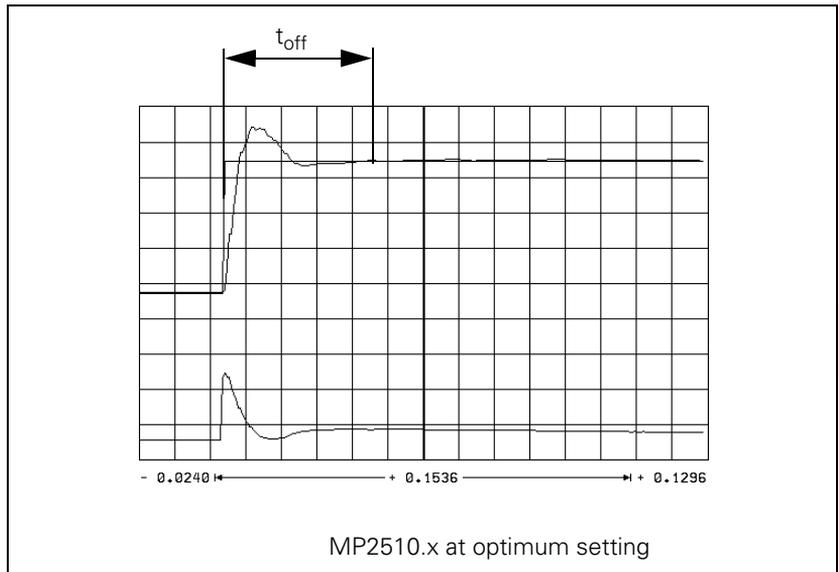


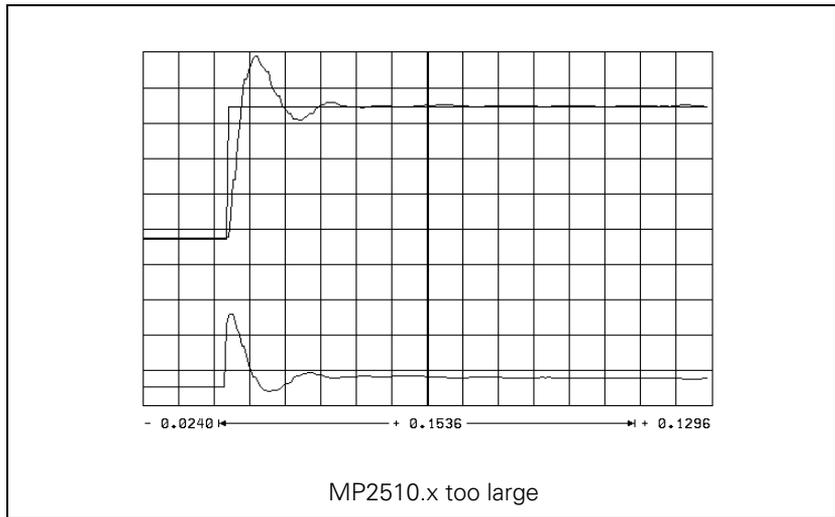
Note

The compensation dampens the control loop. Try first to remove the mechanical causes of the disturbance oscillations.

To reduce the occurrence of disturbance oscillations, HEIDENHAIN recommends the use of motor couplings with a low tendency to oscillate (e.g. from the Rotex Company).

- Increase MP2510.x (I factor) until you see one overshoot followed by a slight undershoot and the settling time t_{off} is as small as possible (realistic value: 3 ms to 15 ms).





Determining the acceleration

- ▶ Load the machine with the maximum permissible weight.
- ▶ Enter the rapid traverse rate as step height
- ▶ During the step response, record the step response of the nominal velocity value V (NOM RPM), the actual speed value V (ACT RPM), and the nominal current value I (I NOMINAL). It is permissible to limit the nominal current value I NOMINAL during acceleration.
- ▶ From the step response of the speed controller you determine the maximum possible acceleration (incl. 10% safety margin).

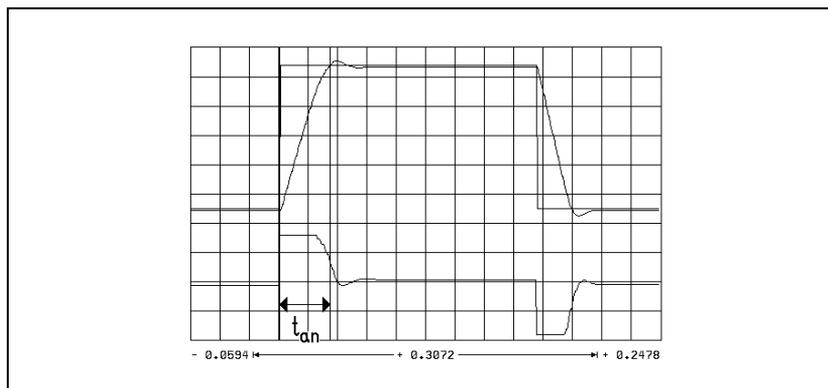
$$a = \frac{F_{\max}}{t_{\text{an}} \cdot 66\,000}$$

a: Acceleration [m/s²]

F_{max}: Maximum machining feed rate (MP1010.x) [mm/min]

t_{an}: Rise time [s]

- ▶ Enter the maximum possible acceleration in MP1060.x.



Checking the counting direction

- ▶ On the oscilloscope, set TRIGGER to FREE RUN.
- ▶ Start recording.
- ▶ Change to MANUAL operating mode.
- ▶ Press the axis direction buttons.
- ▶ Check the counting direction on the display and if necessary, correct it with MP210.x.

Position controller

Adjusting the position control loop:

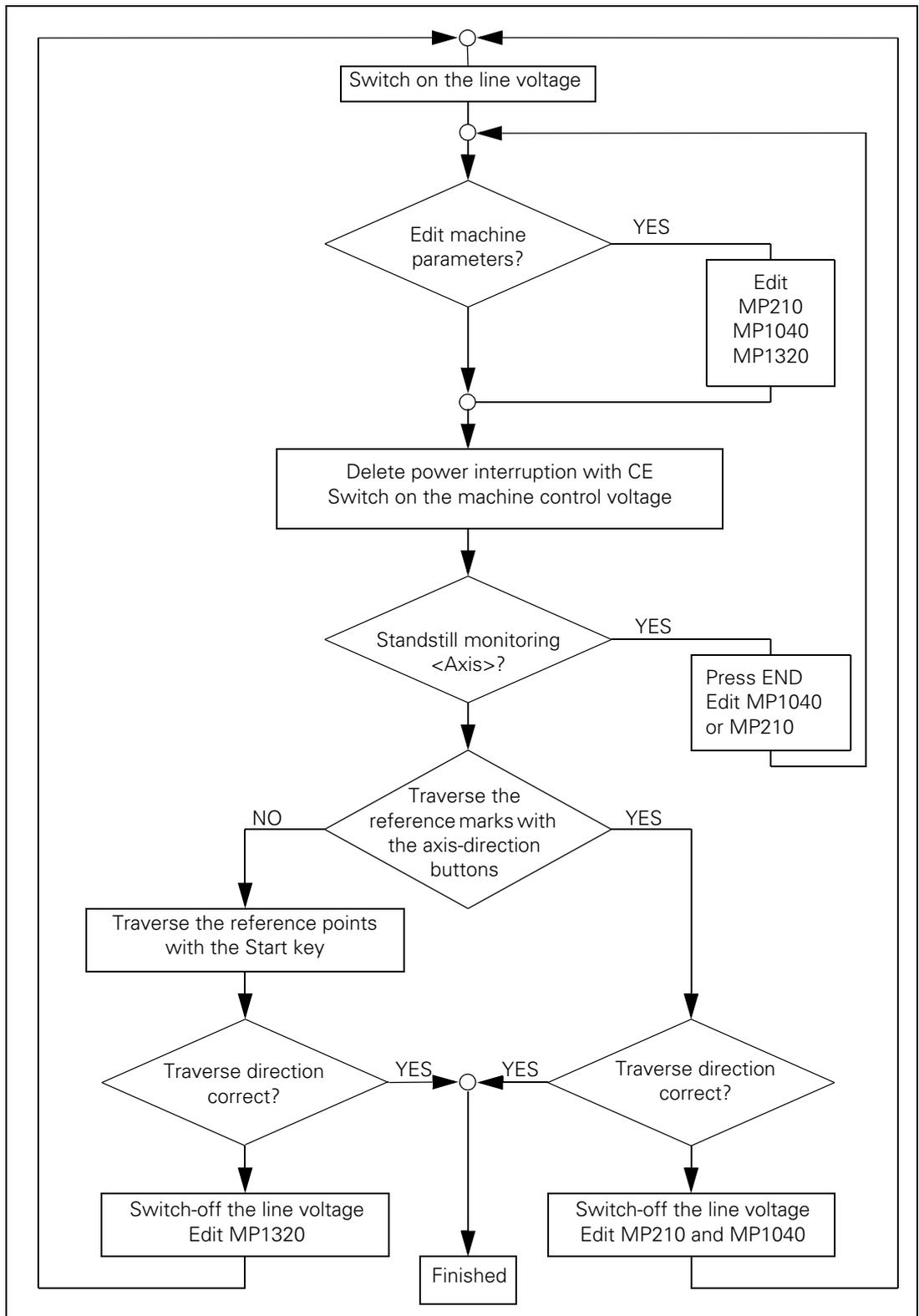
- ▶ Activate a PLC program that is adapted to the machine.
- ▶ Ensure that the position control loop is closed (W1038/W1128/W1040/W1132) and all inputs/outputs are properly operated.
- ▶ Optimize the position control loop in the following 12 steps:

If the position controller still oscillates after optimization, check the I factor (may be too high).

Step 1: Check the traversing direction (see flowchart)

- ▶ In MP1340.x, enter the sequence in which the reference points are to be traversed.





Step 2: Set the traverse range

You can enter up to three traverse ranges.

"Traverse ranges" on page 623. Define the software limit switches as follows:

- ▶ In the MANUAL operating mode, press the MOD key to select the REF display. The position displays show the distance to the machine datum (MP960.x).
- ▶ With the axis direction buttons or the handwheel, move all axes in positive and negative direction until they almost reach the emergency stop limit switches. Write down the displayed positions with algebraic sign.
- ▶ Enter the noted values in MP91x.x and MP92x.x.
- ▶ Press the MOD key and select the ACTL display.

Step 3: In MP1391 or MP1392, select the type of control

For control with velocity feedforward:

- ▶ Enter the temporary input values.

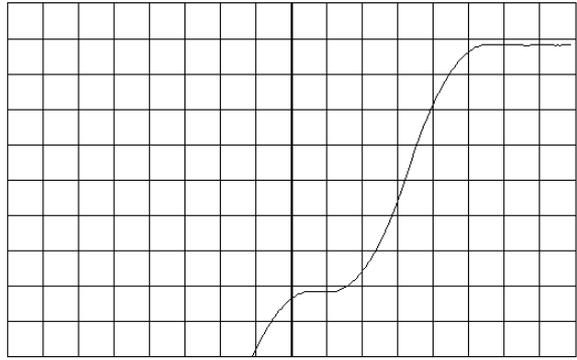
Machine parameters	Temporary input value
MP1391 or MP1392	Set to velocity feedforward control
MP1090.0	Enter a very small jerk, e.g. 1 (dependent on the machine)
MP1090.1	
MP1092	<greater than rapid traverse>
MP1095	0
MP1096.x	0
MP1099.0	5
MP1099.1	3
MP1396.x	1
MP1521.x	0

- ▶ Enter the following test program:

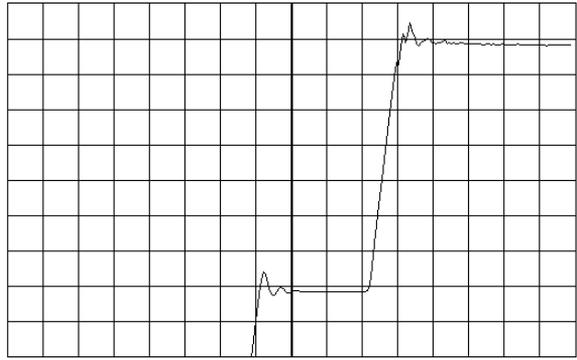
```
LBL 1  
L X <maximum traverse>  
R0 FMAX  
LX0 FMAX  
CALL LBL1 REP 100/100
```

- ▶ Display the actual speed (act. speed) with the integrated oscilloscope and, if necessary, also show the following error (lag).
- ▶ Start the test program with feed-rate override = 100%.
- ▶ In MP1090.0 increase the jerk until the overshoot just disappears.

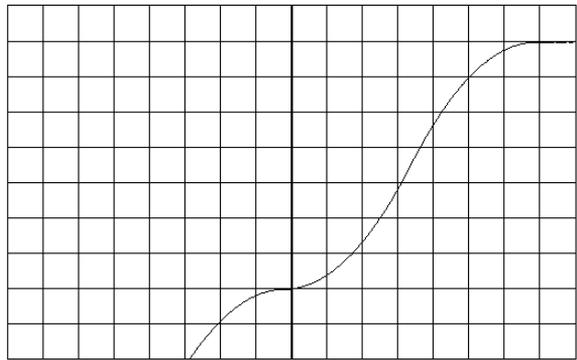




MP1090.x at optimum setting

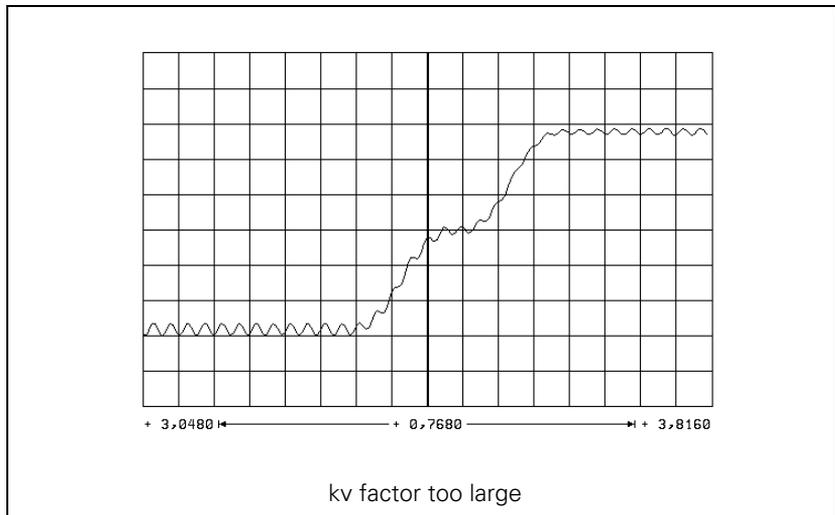
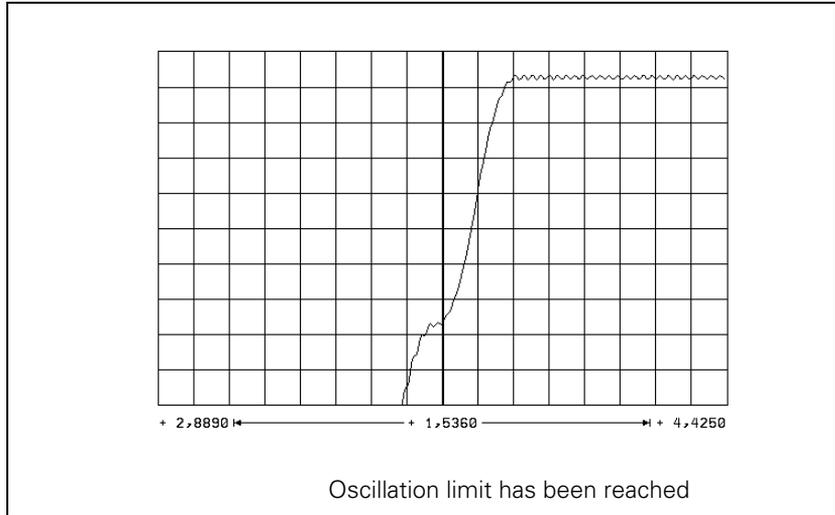


MP1090.x too large



MP1090.x too small

- ▶ Enter the jerk determined from MP1090.0 in the axis-specific parameters MP1241.x and MP1242.x (previously MP1097.x and MP1098.x), and also in MP1086.x, MP1087.x and MP1089.x if required.
- ▶ Increase the k_v factor until the oscillation limit is reached.
- ▶ Calculate MP1510:
 $MP1510.x = \langle \text{determined value} \rangle \cdot 0.6$



Unlike in operation with following error, you can also enter the optimum k_v factor for interpolated axes. You can save a number of different k_v factors in the iTNC and activate them with M functions (see "The control loop" on page 810). MP1090.x applies to all axes. The worst axis determines the input value.

Procedure:

- ▶ Load the axis-specific values in MP1241.x and MP1242.x (or MP1097.x and MP1098.x if using MPMODE=340422 in OEM.SYS).
- ▶ Reduce the adjusted jerk (MP1090.0) depending on the mechanical design of the machine. Do not set the jerk lower than necessary, however, because this strongly reduces the dynamic performance.
 - If at optimized jerk the maximum acceleration is not reached during the acceleration phase, enter the maximum machining feed rate in MP1092. In this case, define a higher jerk for high feed rates (> MP1092) to increase acceleration at these feed rates.

To select the nominal position value filter:

- ▶ Run a test program of short line segments.
- ▶ Use the oscilloscope to record the following error for each axis.
- ▶ Determine for each axis the oscillations on the following error. If you cannot find any oscillations, increase the jerk for the test in order to excite oscillation in the axes. Remember after the test to reset the jerk for each axis to its original value.
- ▶ Select the type of suitable filter characteristic according to the description of the nominal position value on page 821. Consider the lowest determined frequency and the desired damping at this frequency.
- ▶ With MP1200 and MP1201 (previously MP1094 and MP1095) you select the single, double, HSC or advanced HSC filter.
- ▶ Test the three filter settings using a test part made of short line segments.
 - Single filter
 - Double filter
 - HSC filters
 - Advanced HSC filters



Note

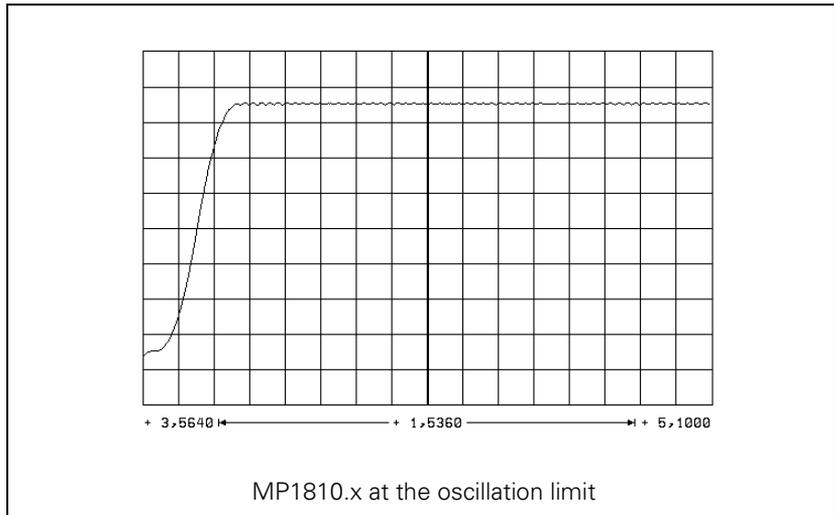
If you have selected the best nominal position value filter for your application, please note that your input value from MP1096.0 can be overwritten by the machine user through Cycle 32. If you have switched off the nominal position value filters (MP1096.x = 0), the machine user can also switch them on using Cycle 32.

Control with following error (servo lag):

The adjusted maximum jerk works during operation with following error.
MP1090 is not changed.

Procedure:

- ▶ Check the temporary input values for the machines parameters
- ▶ Specify the k_v factor for the machining feed rate:
- ▶ Enter the following test program:
LBL1
L X<maximum traverse> R0 F<machining feed rate>
L X0 R0 F<maximum machining feed rate>
CALL LBL1 REP 100/100
- ▶ Display the actual feed rate (v actual) with the internal oscilloscope.
- ▶ Start the test program with feed-rate override = 100 %.
- ▶ Increase the value in MP1810.x up to the oscillation limit.
- ▶ Calculate MP1810.x:
MP1810.x = <determined value> · 0.6



For axes that are interpolated with each other, the k_v factors must be equal.
The axis with the smallest k_v factor defines the input value for all axes.

You can save a number of different k_v factors in the iTNC (MP1815.x) and activate them with M functions (see "The control loop" on page 810).

Procedure for defining a characteristic curve kink point:

- ▶ k_v factor for rapid traverse (characteristic curve kink point):

$$\text{MP1830.x} = \frac{\text{Max. contouring feed rate} \cdot 100 \%}{\text{Rapid traverse}}$$

$$\text{MP1820.x} = 1$$



Set to operation with following error.

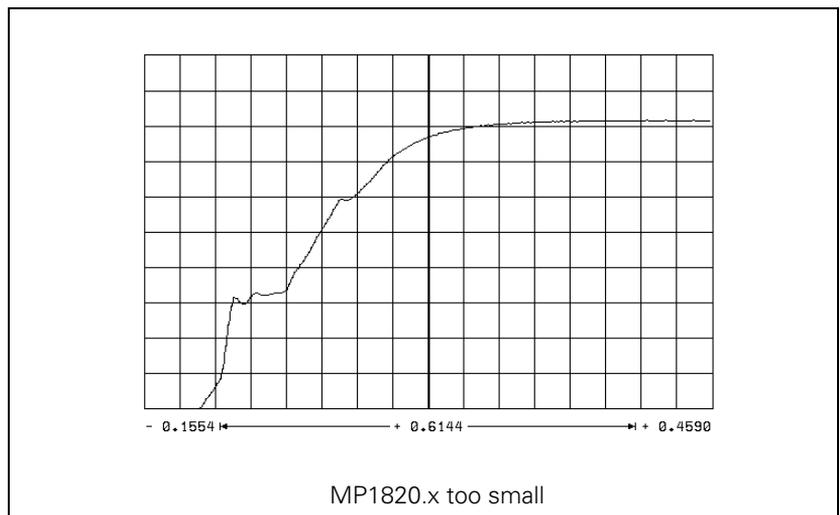
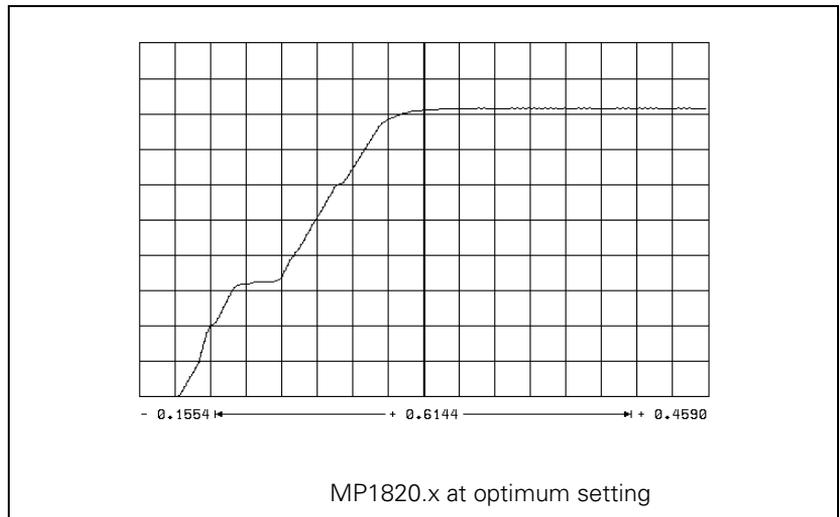
► Enter the following test program:

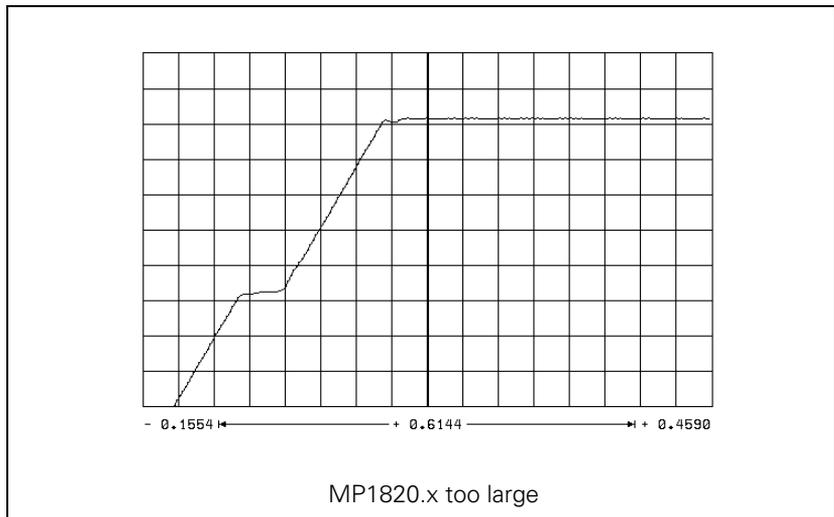
```
LBL2  
L X <maximum traverse> R0 FMAX  
L X0 R0 FMAX  
CALL LBL1 REP 100/100
```

► Start the test program.

► Display the actual feed rate (v actual) with the internal oscilloscope:

- If no oscillations are recognizable, no kink point is required.
- If oscillations are visible, you must reduce MP1820.x until the oscillations have disappeared.





Step 4: Switch-on the nominal position value filter

► In MP1096.x, enter a defined tolerance (e.g. 0.02 mm).

Step 5: Activate the monitoring functions



Note

To ensure that the monitoring functions become effective at the right moment, you must enter meaningful values.

HEIDENHAIN recommends the following input values. You must change these values slightly to adapt them to the design of the machine.

MP	Temporary input value	Meaning
MP1030.x	0.01 mm	Positioning window
MP1110.x	2 · MP1030.x	Standstill monitoring
MP1140.x	0.03 [1000 min ⁻¹]	Movement monitoring
MP1144.x	0.5 mm	Motion monitor for position and speed
MP1410.x	0.5 mm	Position monitoring in operation with velocity feedforward control (erasable)
MP1420.x	2 mm	Position monitoring during operation with velocity feedforward control (emergency stop)
MP1710.x	1.2 · following error in rapid traverse	Position monitoring in operation with following error (erasable)
MP1720.x	1.4 · following error in rapid traverse	Position monitoring during operation with following error (emergency stop)

Step 6: Compensate the backlash

If the cause of the backlash is outside of the control loop:

- ▶ Enter the backlash in MP710.x.

If the cause of the backlash is inside of the control loop:

- ▶ Enter the following test program:

```
LBL 1  
L X100 R0 F10  
L X0  
CALL LBL 1 REP 100/100
```

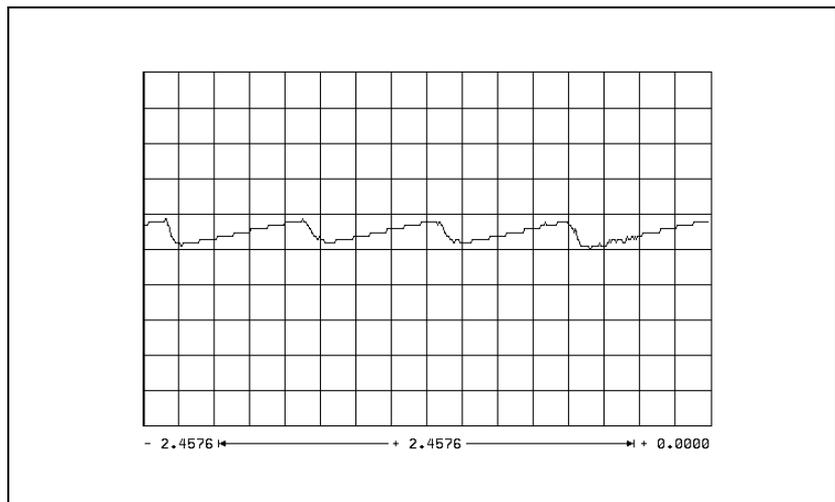
- ▶ Use the internal oscilloscope to record V ACTUAL and V (ACT RPM). At the reversal point the actual feed rate follows the actual shaft speed by the time delay t.
- ▶ Set the machine parameters:
 - $MP750 = t \cdot \Delta V \text{ ACTUAL}$ (keep in mind the units for t and $\Delta V \text{ ACTUAL}$)
 - $MP752 = \text{approx. } 20 \text{ ms}$ (determined in test)

$$\Delta V \text{ ACTUAL} = |V \text{ ACTUAL} - V \text{ (ACT RPM)}|$$

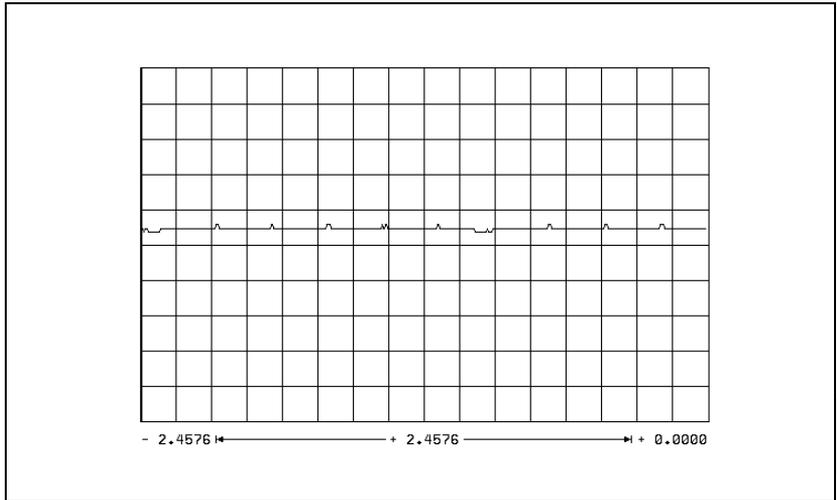
Step 7: Compensate the static (stick-slip) friction

- ▶ Enter the backlash, if any exists.
- ▶ Enter the following test program (static friction in the Y axis):

```
LBL 1
L X+400 IY+0.5 R0 F200
L X0 IY+0.5 R0
CALL LBL1 REP 100/20
```
- ▶ Set the machine parameters:
 - MP1511.x = 0
 - MP1512.x = 20
 - MP1513.x = 0
- ▶ With the integrated oscilloscope, display the following error in the Y axis (Y SDIFF).
- ▶ Start the program and adjust the feed-rate override so that the following error caused by static friction becomes visible.



- ▶ Increase the feed rate until the following error is no longer measurable.
- ▶ From the current contouring feed rate, calculate the feed rate specific to the Y axis and enter the value in MP1513.1.
- ▶ Adjust the feed rate until the following error is measurable again.
- ▶ Increase MP1511.x in increments of 10 000 until the following error is no longer measurable.



If the machine oscillates at a standstill:

- ▶ Decrease MP1512.x.

Step 8: Limit the integral factor of the shaft speed controller

Very high static friction can cause an axis to jerk loose and "jump" around the target position.

- ▶ Increase MP2512.x until the axis remains stationary.

Step 9: Adjust the holding moment

- ▶ Enter the following test program (static friction in axis Z):

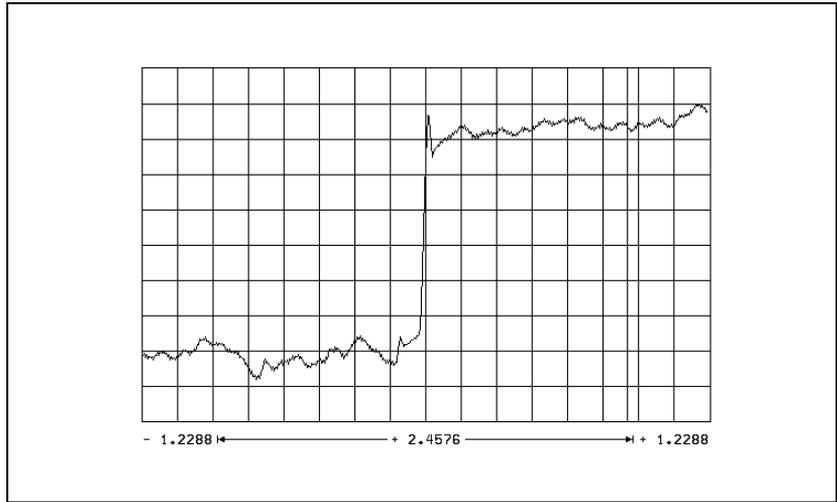
```

LBL 1
L Z+2 RO F50
L Z-2 RO F50
CALL LBL 1/10

```

- ▶ Use the integrated oscilloscope to record the actual shaft speed (ACT RPM) and the nominal current value (I NOMINAL).
- ▶ Start the program.
- ▶ With the feed rate override knob, adjust the motor speed to $\pm 10 \text{ min}^{-1}$ (MP1054.x).
- ▶ Determine the current (I NOMINAL) in both directions of rotation.



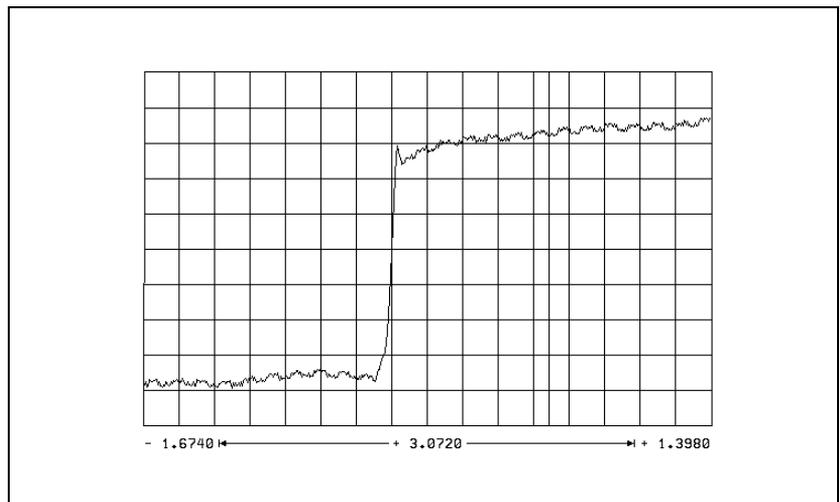


► Calculate MP2630.x:

$$\text{MP2630.x} = \frac{|\text{NOMINAL}_1 + |\text{NOMINAL}_2|}{2}$$

Step 10: Compensate the sliding friction

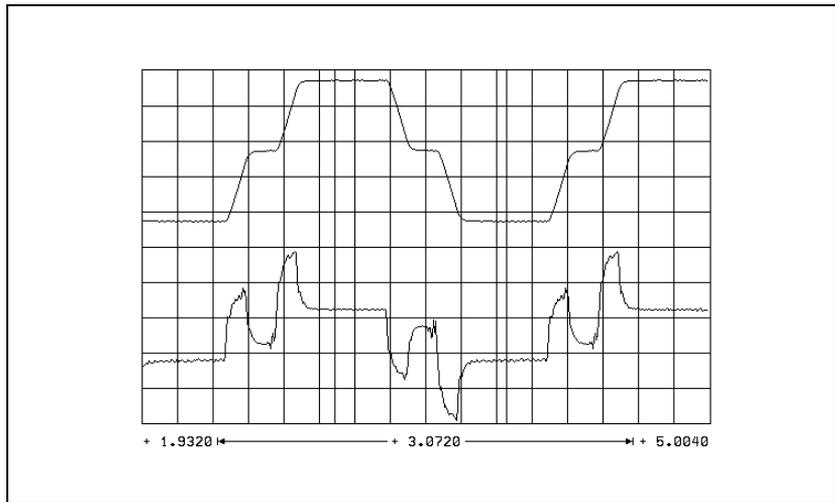
- ▶ Select operation with velocity feedforward control.
- ▶ Enter the following test program (sliding friction in the X axis):
LBL 1
L X+2 RO F50
L X-2 RO F50
CALL LBL 1/10
- ▶ Use the integrated oscilloscope to record the actual shaft speed (ACT RPM) and the nominal current value (I NOMINAL).
- ▶ Start the test program.
- ▶ With the feed rate override knob, adjust the motor speed to 10 min^{-1} (MP1054.x).
- ▶ Determine the current (I NOMINAL) in both directions of rotation.



- ▶ Calculate MP2610.x:

$$\text{MP2610.x} = \frac{I \text{ NOMINAL}_1 + I \text{ NOMINAL}_2}{2}$$

- ▶ Change the test program so that the motor rotates at its rated speed.
- ▶ Restart the test program.
- ▶ Determine the current (I NOMINAL) for the rated shaft speed.



► Calculate MP2620.x:

$$MP2620.x = \frac{I_{NOML1} - I_{NOML2}}{2}$$

In the event that the motor cannot be driven at the rated speed:

► Measure $I_{NOMINAL}$ at maximum speed (rapid traverse) and calculate the current at rated speed as follows:

$$MP2620.x = \frac{(I_{max} - MP2610.x) \cdot \langle \text{rated speed} \rangle}{n_{max}} + MP2610.x$$

I_{nmax} : Current at rapid traverse

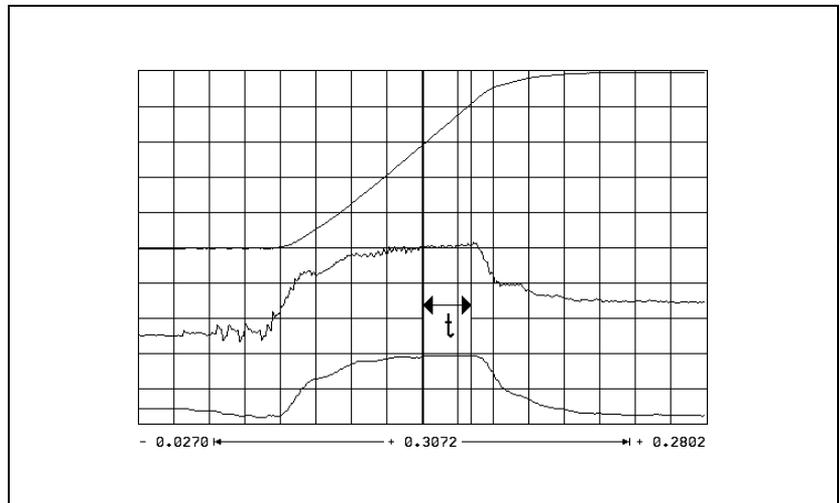
n_{max} : Shaft speed at rapid traverse

Step 11: Check the acceleration feedforward

- ▶ Select operation with velocity feedforward control.
- ▶ Enter the following test program:

```

LBL 1
L X+100 RO F5000
L X-100 RO F5000
CALL LBL 1/10
    
```



- ▶ Use the integrated oscilloscope to record the actual shaft speed (ACT RPM), the nominal current value (I NOMINAL), and the integral-action component of the nominal current value I (INT RPM).
- ▶ Start the test program.
- ▶ Adjust the speed with the feed rate override knob so that I NOMINAL is not limited.
- ▶ Measure the gradient of the acceleration ramp in the part in which I(INT RPM) remains constant.
- ▶ Calculate MP2600.x:

$$MP2600.x = \frac{I (INT RPM) [A] \cdot t [s] \cdot 60 [s/min] \cdot MP1054.x [mm]}{\Delta V (ACT RPM) [mm/min]}$$

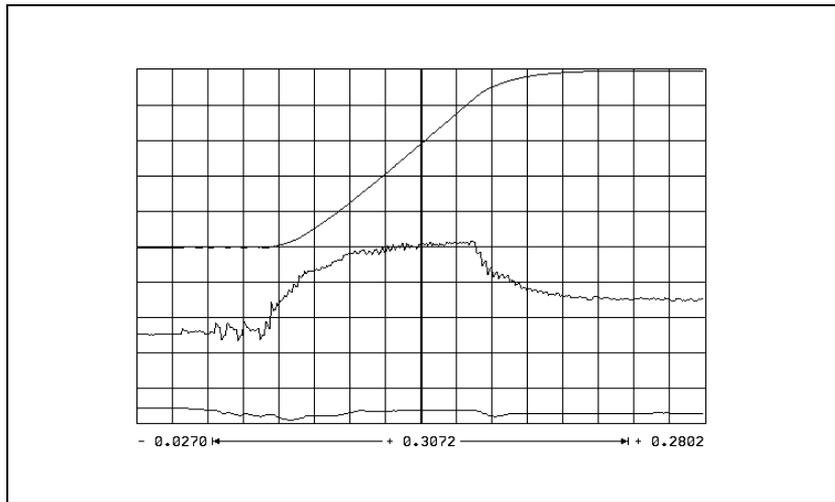
I (N INT): Integral-action component of the nominal current value

t: Acceleration time in which I (INT RPM) remains constant

ΔV (ACT RPM): Change of actual rpm during t

MP1054.x: Traverse distance per motor revolution

- ▶ Repeat this measurement to check the input value of MP2600.x. I (INT RPM) must have approached zero.



Step 12: Run the circular test

With the circular test you can check the exact input values for compensating sliding friction:

- ▶ Determine the radial acceleration:
 $MP1070 = 0.7 \cdot MP1060.x$
 $MP1060.x$ represents the smallest acceleration in the working plane.
- ▶ At mid-range feed rate (approx. 500 mm/min) check the parameter MP2610.
 At the optimum setting the reversal spikes are at a minimum.

At feed rates greater than approx. 6000 min^{-1} the reversal spikes might point inward as a result of overcompensation:

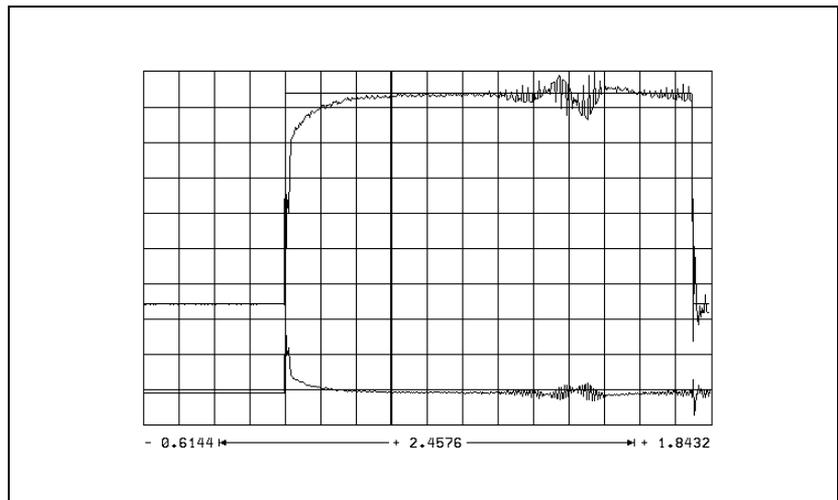
- ▶ In this case, increase $MP2612.x$ until the reversal spikes no longer point inward.

6.14.6 Commissioning the digital spindle

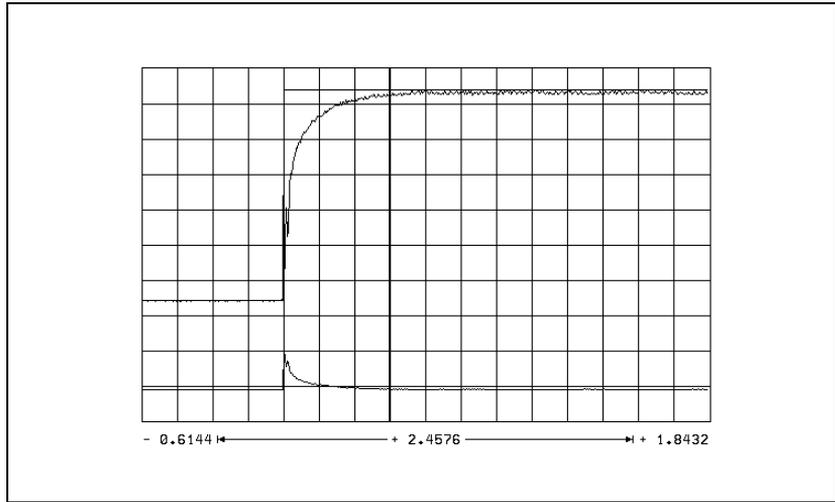
Current controller Same procedure as for digital axes.

Speed controller Define the step function:

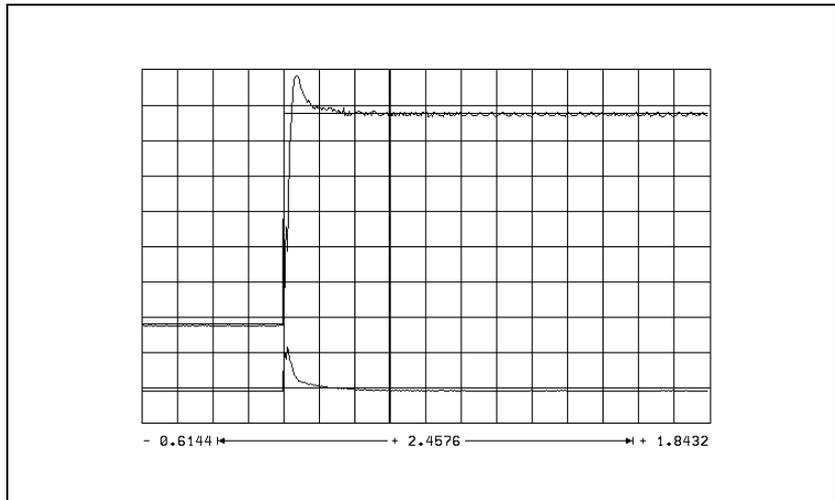
- ▶ In MP3411.x, enter the maximum acceleration and start the step by switching the spindle on.
- ▶ Activate a spindle speed from the highest gear range.
- ▶ With the integrated oscilloscope, record the nominal velocity value V (NOM RPM), the actual speed value V (ACT RPM), and the nominal current value (I NOMINAL).
- ▶ Output a step by activating the Spindle On function (M03/M04).
- ▶ Choose the height of the step function for a very low speed so as not to overload the speed controller, i.e. so that I NOMINAL is not limited.
- ▶ Increase the P factor (MP2500.x) until the system oscillates or no change is visible. To edit machine parameters, press the MP EDIT soft key in the Setup menu.



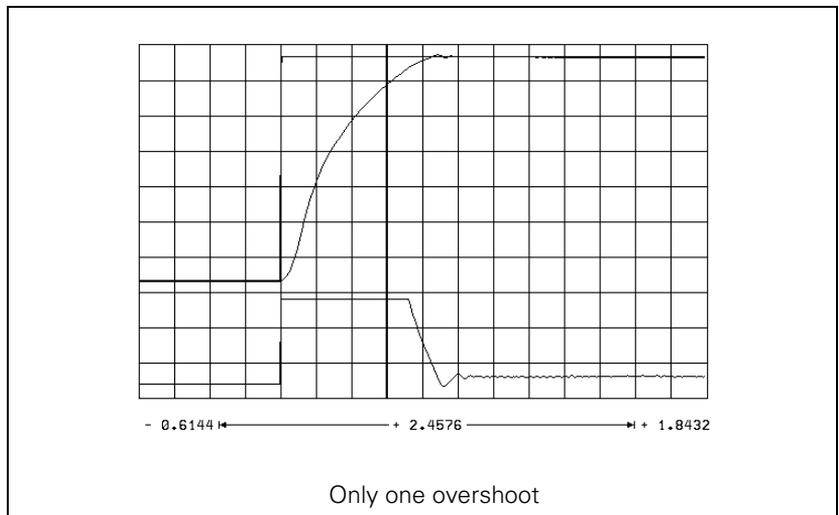
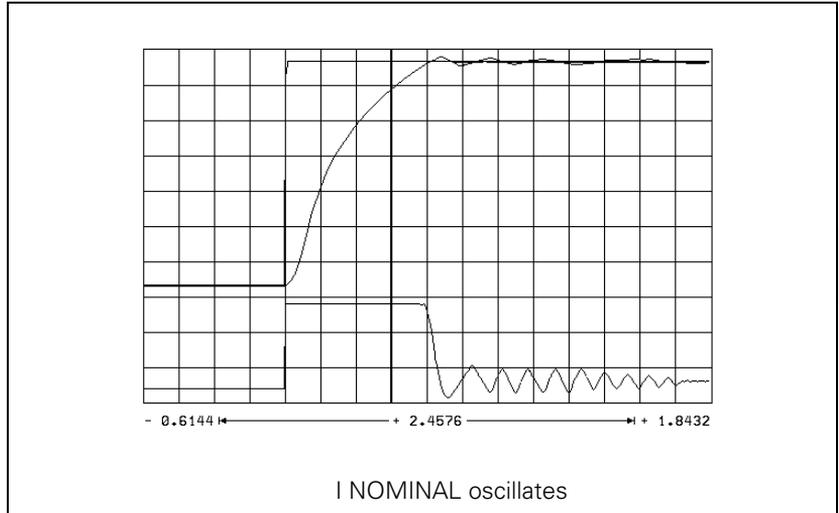
- ▶ Calculate MP2500.x: $MP2500.x = MP2500.x \cdot 0.6$



► Increase the I factor (MP2510.x) until you see one overshoot followed by a slight undershoot.

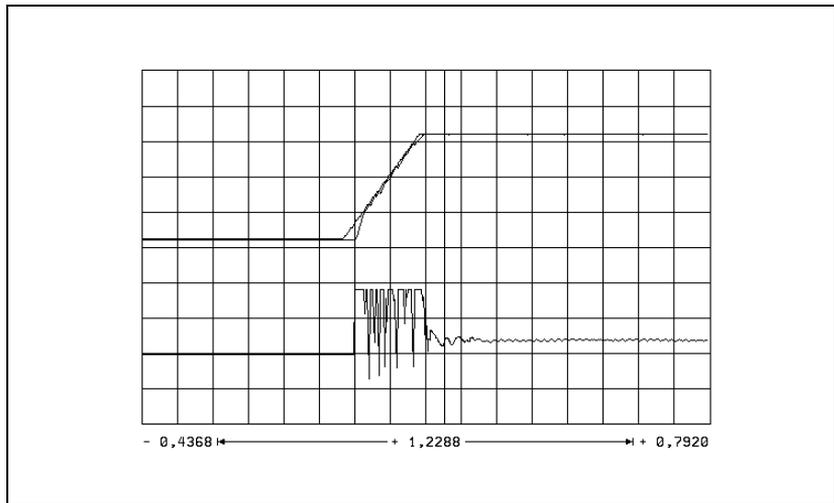


- ▶ Output the step with maximum shaft speed. I NOMINAL is within the limitation during acceleration. I NOMINAL must not oscillate after reaching the maximum speed. If I NOMINAL oscillates:
 - Reduce MP2500.x and MP2510.x evenly until the overshoots are minimized.

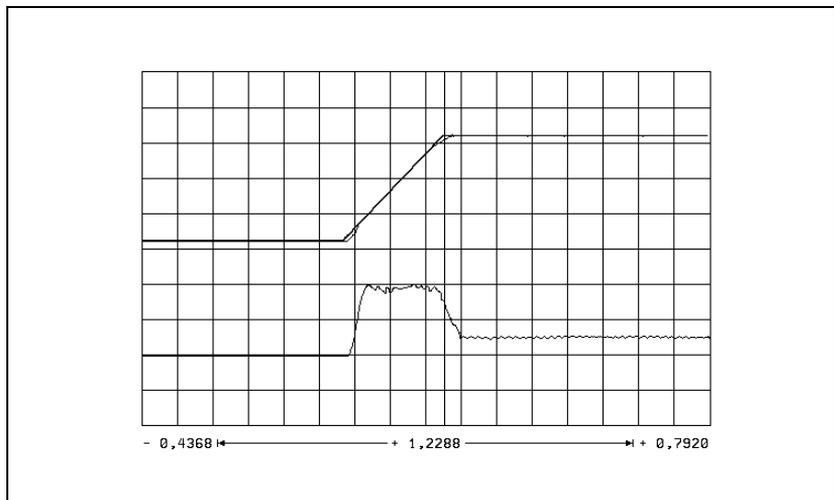


Optimize the acceleration:

- ▶ Optimize the acceleration individually for each gear range.
- ▶ Choose a ramp gradient at which the motor almost reaches the electrical current limit, and set it with MP3411.x.



- ▶ In MP3412.0, enter a factor for MP3411.x that becomes effective in the braking ramp with M05. It is the electrical current limit that is braked.



In the TAPPING and SPINDLE ORIENTATION modes, I NOMINAL must not reach the limit for acceleration:

- ▶ In MP3412.x, enter a factor for MP3411.x for these operating modes.
- ▶ With MP3415.x, specify an individual overshoot behavior for every spindle operating mode. Adapt the nominal value trace to the actual trace.

Check the direction of rotation.

You can check the direction of rotation of the spindle when M03 is output. If the spindle does not rotate in clockwise direction:

- ▶ Modify MP3130.

Position controller

The position control loop of the spindle is closed only during the spindle orientation:

- ▶ Close the position control loop of the spindle. "Oriented spindle stop" on page 968.
 - If the error message "Nominal speed value S too high" appears, you must modify MP3140.
- ▶ Optimize the k_v factor (MP3440.x for each gear range). A TOOL CALL must be run to transfer the modified gear-specific MPs.

6.14.7 Commissioning an analog axis

Temporary input values

► Enter the following temporary input values when you begin

MP	Temporary input value	Meaning
MP1030.x	0.01	Positioning window
MP1090.x	1	Maximum permissible jerk on the tool path
MP1092	<Maximum rapid traverse>	Feed rate threshold from which MP1090.1 becomes effective
MP1110.x	2.0	Standstill monitoring
MP1140.x	10	Movement monitoring
MP1410.x	0.5	Position monitoring in operation with velocity feedforward control (erasable)
MP1420.x	2	Position monitoring in operation with velocity feedforward control (emergency stop)
MP1510.x	1	k_v factor for velocity feedforward control
MP1710.x	50	Position monitoring in operation with following error (erasable)
MP1720.x	50	Position monitoring in operation with following error (EMERGENCY STOP)
MP1810.x	1	k_v factor for control with following error
MP1820.x	1	Multiplier for the k_v factor
MP1830.x	100	Characteristic curve kink point

Adjusting the servo amplifier

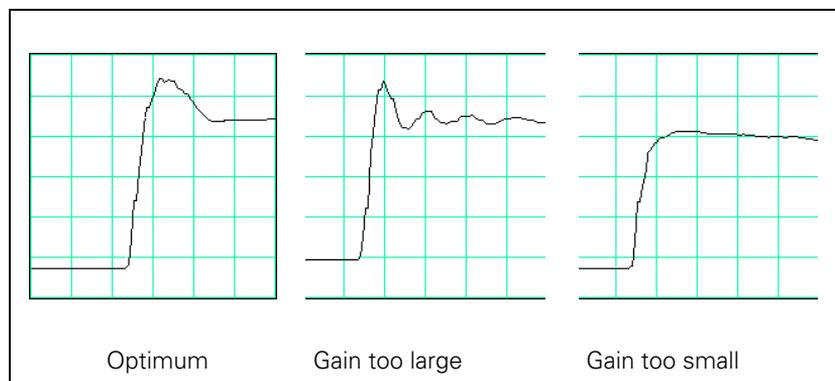


Note

For analog axes, you must adjust the servo amplifier before you optimize the position controller.

Procedure:

- ▶ Disconnect the nominal-value connection between the servo amplifier and CMA-H.
- ▶ Short-circuit the nominal value input on the servo amplifier. The input must have a 0 V voltage.
- ▶ Activate control enabling at the servo amplifier.
- ▶ Connect the power supply to the servo amplifier.
- ▶ Perform a coarse offset adjustment:
 - If the axis moves in spite of the short-circuited nominal value input, you must adjust the offset potentiometer until the axis stops moving.
- ▶ Remove the jumper at the nominal value input and establish a nominal-value connection to the CMA-H.
- ▶ Perform a coarse velocity adjustment:
 - Set MP1010.x (rapid traverse) and MP1050.x (analog voltage at rapid traverse).
 - With the internal oscilloscope functions, output the nominal value step at the height for rapid traverse.
 - Record UANALOG and check the voltage.
 - Use a tachometer to measure the rotational speed of the motor and a tacho potentiometer at the servo amplifier to adjust the nominal speed for rapid traverse.
 - Connect an oscilloscope to the tachometer of the motor.
 - Measure the step response on the tachometer during the step output.
- ▶ Adjust the proportional (P) component and the integral-action (I) component of the speed controller at the servo amplifier.



Determining the acceleration

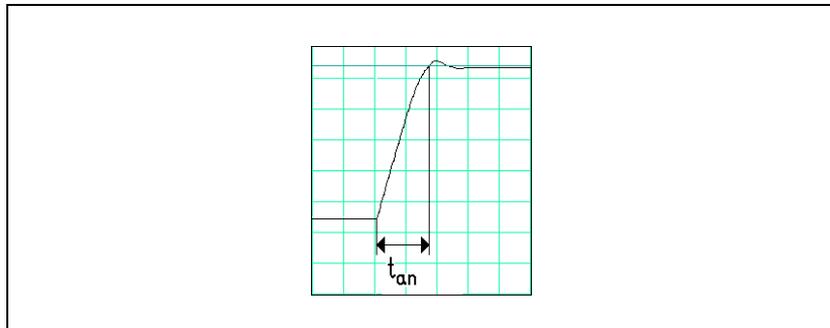
After adjusting the servo amplifier, you can determine from the step response the maximum possible acceleration:

$$a = \frac{F_{\max}}{t_{\text{an}} \cdot 60\,000}$$

a: Acceleration [m/s²]

F_{max}: Maximum machining feed rate (MP1010.x) [mm/min]

t_{an}: Rise time [s]



- ▶ Enter the maximum possible acceleration in MP1060.x.

Position controller

Please note:



Note

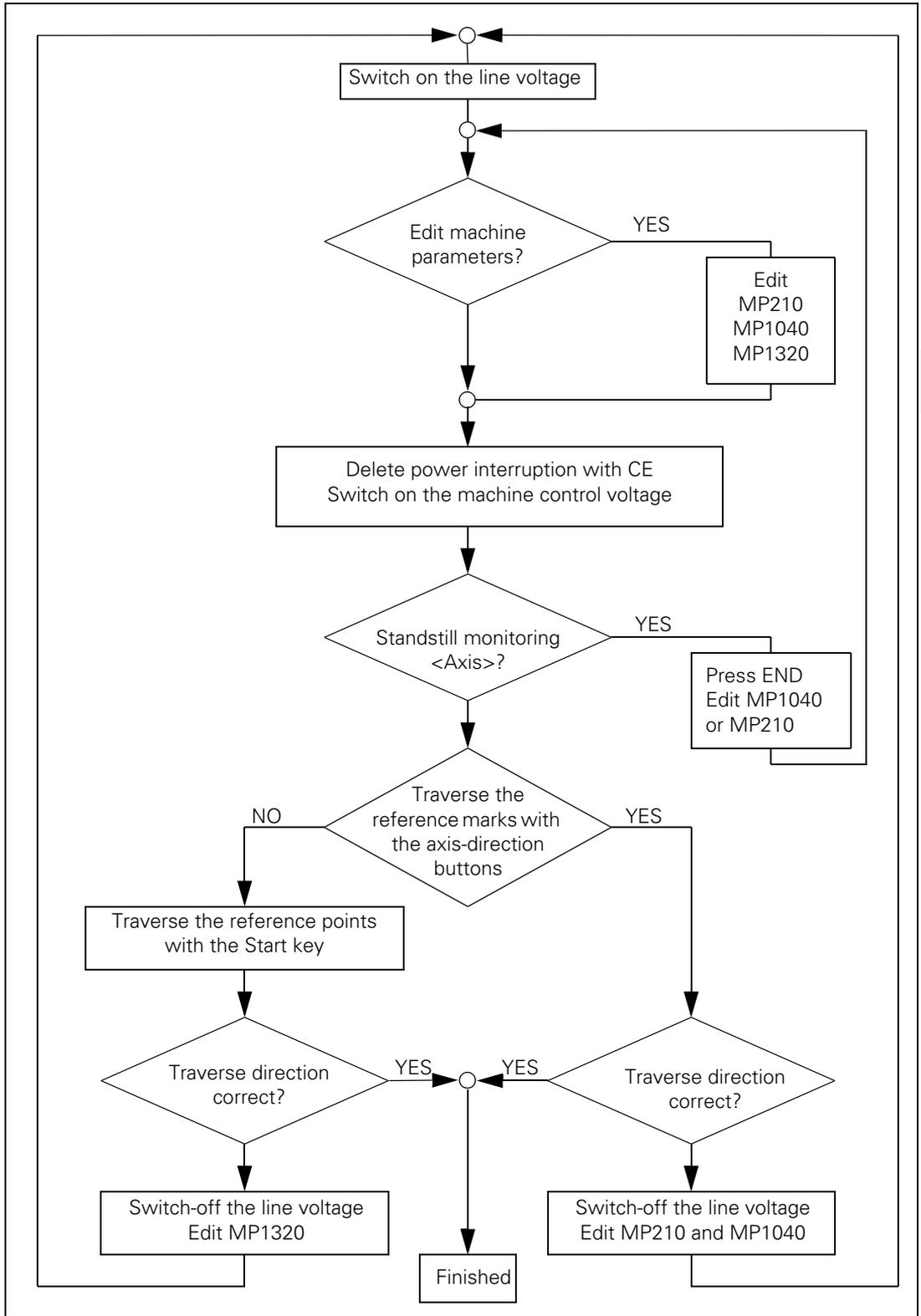
For analog axes, you must adjust the servo amplifier before you optimize the position controller.

Adjusting the position control loop:

- ▶ Activate a PLC program that is adapted to the machine.
- ▶ Ensure that the position control loop is closed (W1038/W1128/W1040/W1132) and all inputs/outputs are properly operated.
- ▶ To optimize the position control loop take the following steps:

Step 1: Check the direction of traverse

(see flowchart)



Step 2: Set the traverse range

Same procedure as for digital axes.

Step 3: Specify the type of control

For control with following error, same procedure as for digital axes.

For control with velocity feedforward control, same procedure as for digital axes.

Step 4: Perform an offset adjustment

At the iTNC: "The control loop" on page 810.

Step 5: Activate the monitoring functions

- ▶ Enter the following temporary input values when you begin: see "Commissioning of digital axes" on page 1026

Step 6: Compensate the backlash

Same procedure as for digital axes.

Step 7: Compensate the static (stick-slip) friction

Same procedure as for digital axes.

6.14.8 Commissioning the analog spindle**Adjusting the servo amplifier**

Same procedure as for analog axes.

Acceleration

Same procedure as for digital spindle. You measure the signals directly at the servo amplifier with an external oscilloscope.

Direction of rotation

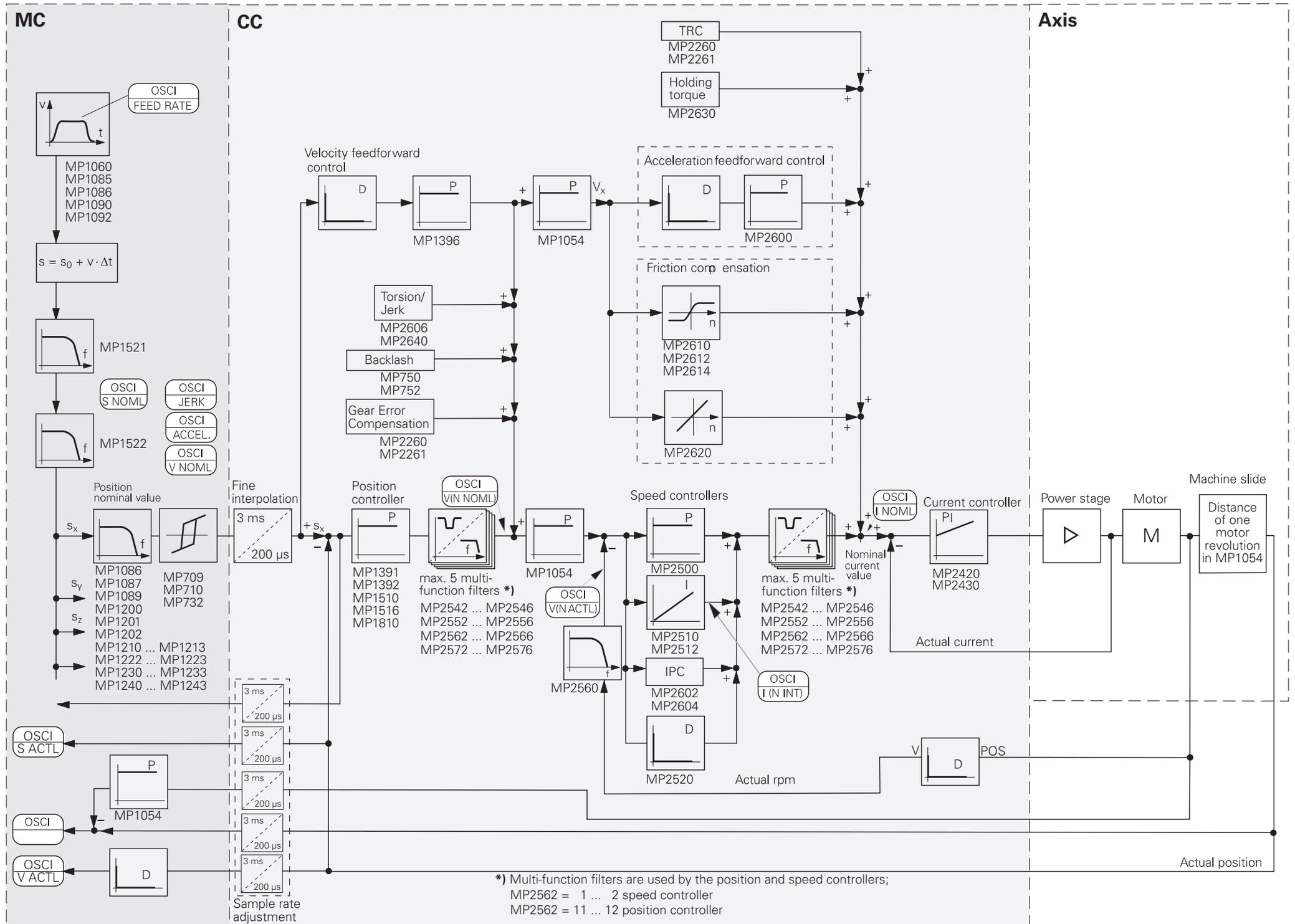
Same procedure as for digital spindle.

Position controller

Same procedure as for digital spindle.



6.15 Block diagram for iTNC 530 with CC 61xx





7 CC 61xx / UEC 1xx controller units

7.1 Specifications

Machine interfacing		iTNC 530 HSCI
Regulation with CC 6xxx / UEC 1xx		
Interpolation factor of encoder signals	4096	
Position control resolution	$\frac{\text{Signal period}}{4096}$ or encoder resolution (EnDat 2.2 interpol.)	
Path interpolation	3 ms	
Fine interpolation	Cycle time for fine interpolation = Cycle time of position controller	
Cycle time of current controller	PWM frequency	Cycle time if MP2182.x = 0:
	3333 Hz	150 μ s
	4000 Hz	120 μ s
	5000 Hz	100 μ s
	6666 Hz	75 μ s (option 49)
	8000 Hz	60 μ s (option 49)
	10000 Hz	50 μ s (option 49)
Cycle time of speed controller	<ul style="list-style-type: none"> ■ Speed controller cycle time = 2 · current controller cycle time ■ Unless PWM frequency \leq 5 kHz and double-speed performance, then: Speed controller cycle time = Current controller cycle time 	
Cycle time of position controller	Position controller cycle time = Speed controller cycle time	

Machine interfacing	iTNC 530 HSCI
<p>Maximum motor speed</p> <p>or by way of:</p>	$n_{\max} = \frac{f_{\text{PWM}} \cdot 60000 \text{ min}^{-1}}{p \cdot 5000 \text{ Hz}}$ <p>n_{\max}: Maximum motor speed [min⁻¹] f_{PWM}: PWM frequency [Hz] <p>p: Number of pole pairs</p> <p>The following PWM frequencies are available: 3333 Hz, 4000 Hz, 5000 Hz With option 49: 6666 Hz, 8000 Hz, 10000 Hz</p> <hr/> <p>Maximum signal frequency of motor encoder = 400 kHz or 800 kHz (depending on MP2561.x bit 0)</p> <p>Calculation of maximum motor speed:</p> $n_{\max} = \frac{f_{\max} \cdot 60000 \text{ [s/min]}}{\text{STR}}$ <p>n_{\max}: Maximum motor speed [min⁻¹] f_{\max}: Maximum signal frequency of motor encoder [kHz] STR: Encoder line count</p> <p>Example 1:</p> $n_{\max} = \frac{400 \text{ kHz} \cdot 60000 \text{ [s/min]}}{2048}$ $n_{\max} = 11718.75 \text{ min}^{-1}$ <p>Example 2:</p> $n_{\max} = \frac{800 \text{ kHz} \cdot 60000 \text{ [s/min]}}{2048}$ $n_{\max} = 23437.5 \text{ min}^{-1}$ </p>
<p>Maximum feed rate by way of:</p>	<p>Maximum signal frequency of the position encoder:</p> <ul style="list-style-type: none"> ■ At 1 V_{pp}: 27 kHz or 400 kHz ■ At 11 μA_{pp}: 27 kHz or 140 kHz <p>Calculation of the maximum feed rate F_{max}:</p> $F_{\max} = \text{SP} \cdot f_{\max} \cdot 60^{-3} \left[\frac{\text{m} \cdot \text{s}}{\mu\text{m} \cdot \text{min}} \right]$ <p>F_{\max}: Maximum feed rate [m/min] f_{\max}: Maximum signal frequency of position encoder [kHz] SP: Signal period of the position encoder [kHz]</p> <p>Example:</p> $F_{\max} = 20 \mu\text{m} \cdot 400 \text{ kHz} \cdot 60^{-3} \left[\frac{\text{m} \cdot \text{s}}{\mu\text{m} \cdot \text{min}} \right]$ $F_{\max} = 480 \text{ m/min}$



7.2 Connecting the encoders

7.2.1 General information

Encoders report positions and movements of the machine to the iTNC 530 HSCI. HEIDENHAIN contouring controls operate with incremental encoders with a 1 V_{PP} signal and absolute encoders with EnDat interface.

Please use only HEIDENHAIN encoder cables, connectors and couplings.

On the CC 61xx there is a **fixed assignment** between the input of the speed encoder, the PWM output and the input of the position encoder.

7.2.2 Position encoders

Incremental position encoders with 1 V_{PP} signals and absolute encoders with EnDat interface can be connected to the CC 61xx. In MP118.x, the position encoder input for the axes is configured for specific axes. In MP119.x, the position encoder input for the spindles is configured for specific axes.

- ▶ Set the encoder signal with MP118.x and MP 119.x bit 0.
- ▶ Set the maximum input frequency with MP118.x and MP 119.x bit 2.
- ▶ Set the encoder type (analog or digital) with MP118.x and MP 119.x bit 3.

MP118.x **Position encoder input for axes**

Format: %xxxx

Input: Bit 0: Position encoder input 1 V_{PP}

0: 1 V_{PP}

1: Reserved

Bit 1: Reserved

Bit 2: Input frequency of the position encoder input

With 1 V_{PP}:

0: 27 kHz

1: 400 kHz

Bit 3: Analog or digital position encoder input

0: Analog encoder signal control (1 V_{PP})

1: Digital encoder signal control (EnDat 2.2)

MP119.x **Position encoder input for spindles**

Format: %xxxx

Input: Bit 0: Position encoder input 1 V_{PP}

0: 1 V_{PP}

1: Reserved

Bit 1: Reserved

Bit 2: Input frequency of the position encoder input

With 1 V_{PP}:

0: 27 kHz

1: 400 kHz

Bit 3: Analog or digital position encoder input

0: Analog encoder signal control (1 V_{PP})

1: Digital encoder signal control (EnDat 2.2)

7.2.3 Speed encoders

Incremental position encoders with 1 V_{PP} signals and absolute encoders with EnDat interface can be connected to the CC 61xx.



7.3 Relationship between speed input, position input and PWM output

On the CC 61xx there is a **fixed assignment between speed encoder input, position encoder input and PWM output**. MP112.x is omitted for this reason (special case: MP113.x). The selection of the encoder inputs and PWM outputs is only via MP120.x/MP121.x.

- When using a CC 61xx in conjunction with HEIDENHAIN inverters

Input of MP120.x/MP121.x: PWM output	51	52	53	54	55	56
Input: Speed encoder	X15	X16	X17	X18	X19	X20
Input: Position encoder (input in MP110.x/MP111.x)	X201	X202	X203	X204	X205	X206

- When using a UEC

Input in MP120.x: Speed output: PWM output	80	81	82	83	84	85
Input: Speed encoder	X15	X16	X17	X18	X19	X20
Input: Position encoder	X201	X202	X203	X204	X205	X206

It is also possible with the CC 61xx to operate open-loop axes (counter axes) or analog controlled axes (MP120 input values 1 to 8) via the position encoder inputs.

Special case: MP113.x

MP113.x is used to switch the speed encoder inputs for the operation of a second spindle. The two speed encoder inputs must be on the same drive-control motherboard.

If only one spindle is used, MP113.x = 0, and the assignment from the above table is valid.

If, for example, two spindle motors are driven with one PWM output (and the same power module) and the power can be switched over through a contactor, the speed encoder input can be selected with MP113.x.

PWM output for the spindle	Speed encoder input	MP113.x input value
X51	X15 or X17	(0 or 15) or 17
X52	X16 or X18	(0 or 16) or 18
X53	X15 or X17	15 or (0 or 17)
X54	X16 or X18	16 or (0 or 18)

The error message **C2A0 Encoder input <axis>** appears if an invalid entry is given for MP113.x.

7.4 Single-speed, double-speed, PWM frequency

7.4.1 General information

Structure of the new CC 61xx and UEC 11x controller units

There is no backplane between the CC and MC for the CC 61xx and MC 6xxx. The CCs continue to be supplied via X69 and X74. The MCs of the new generation are supplied with +24 V NC voltage via X101, independently of the CCs.

A new DSP processor is used on the new CC 61xx and the UEC 11x. A single one of these DSPs can regulate up to six control loops, with the same controller performance as the CC 424 (B).



Note

Due to the new structure of the CC 61xx, not all of the axis configurations of the CC 424 can directly be implemented on the CC 61xx! Please note in particular the changes in the available double-speed control loops of a CC 61xx.

The CC 61xx family has a modular structure. The addressed DSP is on the motherboard of the controller. Every drive-control motherboard is an HSCI participant in the HSCI system. These drive-control motherboards have two control loops, and can therefore control two axes. The same applies to each drive-control expansion board, which does not have its own DSP, however. A drive-control expansion board has two control loops, but can only be used in combination with a drive-control motherboard (DSP).

As a result, a maximum of two drive-control expansion boards can be controlled by one drive-control motherboard (DSP). This configuration results in a maximum of six axes that a single DSP can control. If more than six axes are to be controlled, then another drive-control motherboard is necessary, providing another DSP. All axes that are controlled by a DSP belong to a controller group.

If two drive-control motherboards (two DSPs) are housed in one CC (this means two HSCI participants in one housing), then the inputs and outputs of the CC are uniquely assigned to one of the two controller groups (DSPs) via the letters A and B.

Index A means that these inputs and outputs are controlled by the first DSP (first drive-control motherboard). All inputs/outputs that are assigned to a DSP belong to a controller group (here: controller group A). Index B means that these inputs/outputs are controlled by the second DSP (second drive-control motherboard), and therefore belong to controller group B.

For configuring the controller units via machine parameters, the HSCI address of the respective drive-control motherboard must be entered in MP108.x or MP109.x. The HSCI address is obtained from the position of the drive-control motherboard in the HSCI system. The first drive-control motherboard after X500 of the MC is given the address 0, etc. However, the HSCI address to be entered only depends on the drive-control motherboards in the system. I/O units (PLs) and machine operating panels (MBs) are not taken into account. This means that for the first drive-control motherboard, you have to enter the address 0 in MP108, regardless of whether PLs or MBs are located before the CC in the HSCI chain.



There is one LED on each drive-control motherboard indicating its HSCI address relevant for MP108.x and MP109.x by a blink code.



Also, on the CC 61xx the inputs and outputs have permanent assignments to each other. Switching of the inputs and outputs, as with the CC 424, is not possible here.

Example:

X51, X15, X201 are permanently assigned to each other.

X53A, X17A, X203A are permanently assigned to each other.

etc.

See the table for the assignments. The assignment within each row is permanent. Switching between the rows is not possible with the CC 61xx.

Speed output: PWM output (CC 61xx / UEC 11x)	Input: Speed encoder	Input: Position encoder
X51(A/B) / X81	X15(A/B)	X201(A/B)
X52(A/B) / X82	X16(A/B)	X202(A/B)
X53(A/B) / X83	X17(A/B)	X203(A/B)
X54(A/B) / X84	X18(A/B)	X204(A/B)
X55(A/B) / X85	X19(A/B)	X205(A/B)
X56(A/B) / X86	X20(A/B)	X206(A/B)

7.4.2 Single-speed and double-speed (software option 49)

Controller performance

Performance of the CC 61xx

Depending on the required machine or controller performance, it may be necessary to drive an axis with increased computing power of the controller: this is then referred to as a double-speed axis. On the CC 61xx controllers, there is a DSP on every drive-control motherboard. The UEC 11x controller also has a DSP, with which it controls all the axes of the UEC 11x.

A DSP has sufficient computing power to control up to six single-speed axes or three double-speed axes. A double-speed axis requires the computing power of two single-speed axes. This enables you, for example, to achieve excellent results regarding the workpiece surface quality or speed optimization.

Single-speed control loops are used for:

- Spindles
- Conventional axes

With the CC 61xx, single-speed control loops can normally also be used for:

- Linear motors
- Torque motors

Double-speed control loops are used for:

- High-frequency spindles
- "Axes that are difficult to control"

The Technical Manual of your control describes the conditions under which a double-speed control loop must be used. All control loops are defined as single-speed control loops in the software default settings.

With the CC 61xx controller generation, single-speed or double-speed axes are no longer configured on the specific DSP (CC 424), but individually in MP2000.x.



Note

CC 61xx double-speed control loops require software option 49!

With the CC 61xx, it is possible to distribute double-speed axes controlled by one DSP over any of the outputs of the controller groups. The only restriction is the maximum controller performance of the parent DSP, which must not be exceeded. For example, with a CC 6106 up to any three outputs may be used for double-speed axes.

Double-speed control loops are needed particularly for linear drives in order to attain proper speed/position control factors required by high-efficiency mechanical systems.

High-speed spindle motors requiring PWM frequencies greater than 5 kHz with short current controller cycle times can be operated only with double-speed control loops.

Please note that the maximum number of available control loops may be reduced through the configuration of double-speed control loops.

This leads to the following possible configurations for a CC 6106:

Composition of the CC 6106 controller unit:

Controller group (one DSP) for max. six single-speed (SS) axes or three double-speed (DS) axes					
Drive-control motherboard (CB)		Drive-control expansion board 1 (CE_1)		Drive-control expansion board 2 (CE_2)	
X51 (SS)	X52 (SS)	X53 (SS)	X54 (SS)	X55 (SS)	X56 (SS)

Possible configurations for the CC 6106:

Number of single-speed axes:	Number of double-speed axes:
6 single-speed	0 double-speed
4 single-speed	1 double-speed
2 single-speed	2 double-speed
0 single-speed	3 double-speed

Composition of the CC 6108 controller unit:

Controller group A (one DSP) for max. four SS or three DS				Controller group B (one DSP) for max. four SS or three DS			
CB A		CE 1		CB B		CE 1	
X51A (SS)	X52A (SS)	X53A (SS)	X54A (SS)	X51B (SS)	X52B (SS)	X53B (SS)	X54B (SS)

Possible configurations for the CC 6108, each for controller group A and B:

Number of single-speed axes:	Number of double-speed axes:
4 single-speed	0 double-speed
3 single-speed	1 double-speed
2 single-speed	2 double-speed
1 single-speed	2 double-speed
0 single-speed	3 double-speed

Composition of the CC 6110 controller unit:

Controller group A (one DSP) for max. four SS				Controller group B (one DSP) max. six SS					
CB A		CE 1		CB B		CE 1		CE 2	
X51A (SS)	X52A (SS)	X53A (SS)	X54A (SS)	X51B (SS)	X52B (SS)	X53B (SS)	X54B (SS)	X55B (SS)	X56B (SS)

Possible configurations for the CC 6110 for controller group A:

Number of single-speed axes:	Number of double-speed axes:
4 single-speed	0 double-speed
3 single-speed	1 double-speed
2 single-speed	2 double-speed
1 single-speed	2 double-speed
0 single-speed	3 double-speed

Possible configurations for the CC 6110 for controller group B:

Number of single-speed axes:	Number of double-speed axes:
6 single-speed	0 double-speed
5 single-speed	0 double-speed
4 single-speed	1 double-speed
3 single-speed	1 double-speed
2 single-speed	2 double-speed
1 single-speed	2 double-speed
0 single-speed	3 double-speed



Composition of the UEC 11x:

DSP on drive-control motherboard for max. five single-speed (SS) axes					
Controller group 1 (on drive-control motherboard CB)		Controller group 2 (controller expansion board 1 CE 1)		Controller group 3 (controller expansion board 2 CE 2)	
X80 (SS) Spindle	X81 (SS) Axis	X82 (SS) Axis	X83 (SS) Axis	X84 (SS) Axis	Vacant

Useful configurations of UEC 11x:

Number of single-speed axes:	Number of double-speed axes:
5 single-speed	0 double-speed
4 single-speed (X80, X81, X82, X83)	1 double-speed (X84)
3 single-speed axes (X82, X83, X84)	1 double-speed spindle (X80)

MP2000.x Performance of control loop (software option 49)

Input: 0: Single-speed axis
1: Double-speed axis



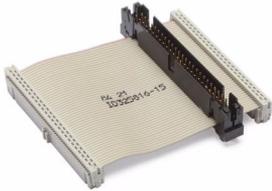
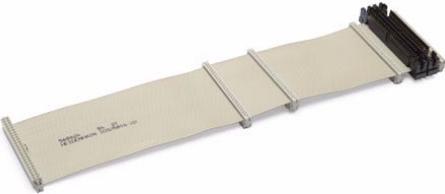
7.4.3 Combination of several CC 61xx controller units

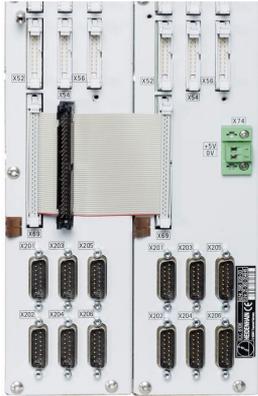
Combination of several CC 61xx controller units

Two drive-control motherboards are housed in one CC 6108 or CC 6110. Both boards must be connected to the supply bus via the connectors X69A and X69B. A short ribbon cable (ID 325816-15) with three connectors for X69 is therefore included with either of the two CCs. Two of the connectors are connected to X69A and X69B of the CC, respectively. The third connector of the cable serves to establish the connection to X69 of the power supply module.

The ribbon cable with ID 325816-24 (100 mm) must be used for connecting two CC 6106 controller units.

However, if you want to arrange two CC 61xx controller units other than the CC 6106 controller unit next to each other in one row, you need to use the ribbon cable with ID 325816-22. This cable is required for the following configurations: 2 x CC 6108, 2 x CC 6110 or CC 6110 with CC 6108. The four connectors X69 of the CCs are then connected via the additional accessory cable. The fifth connector of the cable is used to establish the connection to X69 of the power supply module.

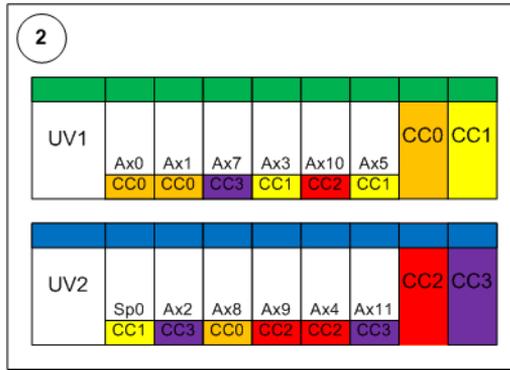
Ribbon cable	
ID 325816-15	 A short, beige ribbon cable with three connectors. The top connector is a 24-pin D-sub connector, and the bottom connector is a 5-pin D-sub connector. The cable is labeled "ID 21 325816-15".
ID 325816-24	 A short, beige ribbon cable with two connectors. Both the top and bottom connectors are 24-pin D-sub connectors. The cable is labeled "ID 21 325816-24".
ID 325816-22	 A long, beige ribbon cable with five connectors. The top connector is a 24-pin D-sub connector, and the bottom connector is a 5-pin D-sub connector. The cable is labeled "ID 21 325816-22".

Ribbon cable	
<p>ID 325816-24 connecting two CC 6106</p>	 <p>The image shows a ribbon cable (ID 325816-24) connected to two CC 6106 controller units. The cable is plugged into connectors X55 and X56 on the left unit and X53 and X56 on the right unit. A green power connector X74 is visible on the right unit. Various other connectors like X201, X202, X203, X204, X205, X206, X207, X208, X209, X210, X211, X212, X213, X214, X215, X216, X217, X218, X219, X220, X221, X222, X223, X224, X225, X226, X227, X228, X229, X230, X231, X232, X233, X234, X235, X236, X237, X238, X239, X240, X241, X242, X243, X244, X245, X246, X247, X248, X249, X250 are labeled on the units.</p>
<p>ID 325816-22 connecting two CC 6108</p>	 <p>The image shows a ribbon cable (ID 325816-22) connected to two CC 6108 controller units. The cable is plugged into connectors X53A and X53B on the left unit and X53A and X53B on the right unit. A green power connector X74 is visible on the right unit. Various other connectors like X201A, X201B, X202A, X202B, X203A, X203B, X204A, X204B, X205A, X205B, X206A, X206B, X207A, X207B, X208A, X208B, X209A, X209B, X210A, X210B, X211A, X211B, X212A, X212B, X213A, X213B, X214A, X214B, X215A, X215B, X216A, X216B, X217A, X217B, X218A, X218B, X219A, X219B, X220A, X220B, X221A, X221B, X222A, X222B, X223A, X223B, X224A, X224B, X225A, X225B, X226A, X226B, X227A, X227B, X228A, X228B, X229A, X229B, X230A, X230B, X231A, X231B, X232A, X232B, X233A, X233B, X234A, X234B, X235A, X235B, X236A, X236B, X237A, X237B, X238A, X238B, X239A, X239B, X240A, X240B, X241A, X241B, X242A, X242B, X243A, X243B, X244A, X244B, X245A, X245B, X246A, X246B, X247A, X247B, X248A, X248B, X249A, X249B, X250A, X250B are labeled on the units.</p>

If several CC 61xx controller units are connected via a cable with ID 325816-xx, the +5 V power must usually only be supplied by connector X74 of the rightmost CC controller unit. The +5 V supply must be tested, however, see page 235.

Two drive-control motherboards are housed in one CC 6108 or CC 6110. The HSCI bus must be forwarded from one board to the other via the connectors X500A and X502B. A short HSCI cable (approx. 20 cm) is therefore included with either of the two CCs.

Figure 2: Impermissible parameterization



- HEIDENHAIN recommends using the MS 11x installation kits for a multi-row configuration with several UV(R) supply modules, see page 122.
- All UV(R) power supply modules must be connected to the grounding conductor with a cross section of at least 10 mm².
- The signal-ground connections of the CC 61xx should always be connected to the housing if only one CC 61xx per row is used.
- If one or more than one UP is used, the UPs must always be connected to the DC-link of the respective inverter. HEIDENHAIN recommends connecting the UPs to the left of the CC 61xx.

7.4.4 PWM frequency

The control loops can be assigned different PWM frequencies in MP2180.x.

There are three fundamental PWM frequencies: 3333 Hz, 4000 Hz and 5000 Hz.

A control loop can be operated at the fundamental PWM frequency or at double this frequency.

PWM frequencies of the CC 61xx

The same PWM frequency must be set for both PWM outputs of a drive-control motherboard or a drive-control expansion board in MP2180.x.

- Drive control board 1: X51, X52
- Drive control board 2: X53, X54
- Drive control board 3: X55, X56

PWM frequencies of the UEC 1xx

The same PWM frequency must be set for both PWM outputs of a drive-control motherboard or a drive-control expansion board in MP2180.x.

- Drive control board 1: X80, X81
- Drive control board 2: X82, X83
- Drive control board 3: X84

If you have set different fundamental PWM frequencies for the two outputs, the error message **C013 PWM frequency error** will appear.

- ▶ In MP2180.x, set the PWM frequency (either the fundamental or double the fundamental PWM frequency) for every pair of outputs.

The cycle time of the current controller depends on the PWM frequency:

$$T_I = \frac{1}{2 \cdot f_{PWM}}$$

Can be set via MP2180.x (switchable during operation); the calculation of the current-controller cycle time must be adapted via MP2182.x

Switching the PWM frequency during operation

It may sometimes be necessary to increase the PWM frequency at high speeds, for example, if problems with the spindle temperature occur. With the CC 61xx, the PWM frequency can be switched during operation. You can switch only between the fundamental PWM frequency and double the fundamental PWM frequency. Since the current controller cycle time cannot be changed during operation, the entry MP2182.x = 2 is required. This means that at the same time that you enter double the fundamental PWM frequency in MP2180.x, you must set MP2182.x = 2.

Example:

Fundamental PWM frequency of 5000 Hz; at high rotational speeds, a control loop is to be operated at double the fundamental PWM frequency.

Default setting	Machine parameters to be edited	
	Single-speed control loop	Double-speed control loop
MP2180.x = 5000	MP2180.x = 10000	MP2180.x = 10000
MP2182.x = 0	MP2182.x = 2	MP2182.x = 2

MP2186.x and MP2188.x can be used to switch between simple and double PWM frequency, depending on the rotational speed.

Set MP2182.x = 2 if you want to use automatic PWM frequency switching. The PWM frequency of a power module is then doubled at a certain speed, based on a PWM frequency f_{PWM} 5 kHz. In MP2186.x and MP2188.x you define the speeds at which switching occurs. The current controller cycle time is always (if MP2182.x = 2) based on the high PWM frequency.

MP2180.x PWM frequency

Input: 0: $f_{PWM} = 5000$ Hz
 3200 to 3999: $f_{PWM} = 3333$ Hz
 4000 to 4999: $f_{PWM} = 4000$ Hz
 5000 to 5999: $f_{PWM} = 5000$ Hz
 6000 to 7999: $f_{PWM} = 6666$ Hz
 8000 to 9999: $f_{PWM} = 8000$ Hz
 10000: $f_{PWM} = 10000$ Hz

MP2182.x Cycle time of current controller at double the fundamental PWM frequency

Input: 0 = Standard case:
 MP2180 = [3333 Hz to 5000 Hz] with single-speed axes
 MP2180 = [3333 Hz to 10000 Hz] with double-speed axes
 (current controller cycle time = $1 / (2 * MP2180)$)
 2 = Special case CC 61xx:
 MP2180 = [3333 to 5000 Hz] with speed-dependent doubling (MP2186, MP2188) of the fundamental PWM frequency from MP2180 with double-speed axes
 (current controller cycle time = $1 / (4 * MP2180)$)

Switching the PWM frequency depending on the speed

This function is used with high-speed spindle drives. This switchover is only possible for double-speed control loops.

In MP2186.x and MP2188.x, a speed-dependent hysteresis for switching the PWM frequency is specified. It only takes effect if the value in MP2188 is less than the value in MP2186.

This function is connected with MP2182.x and MP2180.x. Only if MP2182.x has the value 2 and MP2180.x is ≤ 5 kHz, does the switching of the PWM frequency take effect. Please note that the adjustment of the current controller (MP2420, MP2430) is based on the lower PWM frequency ≤ 5 kHz. Adaptation of the current-controller parameters and consideration of the power-module derating are performed automatically.

Using this function provides several benefits:

- At lower speeds and therefore a lower PWM frequency, the power module provides a comparatively high current. This results in a relatively high maximum motor torque.
- On the one hand, losses due to harmonics in the motor become more important as the speed increases, and on the other hand, the relationship between the electrical frequency and the PWM frequency worsens. These two disadvantages can be counteracted by increasing the PWM frequency. The resulting reduction of the current normally is insignificant, since in part due to motor characteristics, very high motor currents are mostly no longer possible or needed at higher speeds.



Note

The speed-dependent switching of the PWM frequency is only permitted with power modules from HEIDENHAIN.



Danger

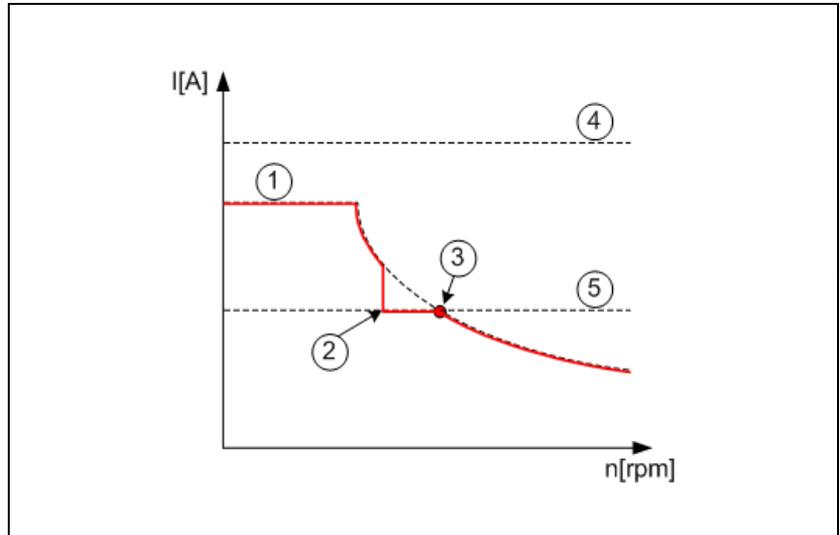
Speed-dependent switching of the PWM frequency with non-HEIDENHAIN power modules can lead to malfunctions, and possibly to damage of the power modules. Therefore, only use this function with power modules from HEIDENHAIN.

When determining the optimum switching speed for the PWM frequency, you should consider that the maximum motor current decreases as the speed rises, due to the finite DC-link voltage.

The current for the drive depends on the maximum permissible motor current and power-module current. The smaller of the two values limits the current for the drive. The value of the maximum power-module current is reduced by approx. 30 % because of the derating when doubling the PWM frequency. During derating the rated current I_N and the maximum current I_{MAX} are reduced by the same factor.



The optimum switching speed results from the intersection of the maximum current curve of the motor with the maximum current curve of the power module for the high PWM frequency. You can determine the maximum current curve of the spindle motor by using the TNCscopeNT software. Record the current $I_{noml/S}$ in dependency of the speed. The spindle must be accelerated to the maximum speed so that the derating behavior can be seen in the curve. The figure shows the behavior when the values entered for the switching speed (MP2186.x, MP2188.x) are too low. This then results in a speed range where the current for the motor is less than the permitted and maximum current, resulting in inconsistencies in the motor's torque behavior.



- 1: Red line: Maximum current for the drive, resulting from the entries in MP2186.x and MP2188.x
- 1: Broken line: Maximum possible current for the drive (I_{max} of motor)
- 2: Switching point for the PWM frequency set too low
- 3: Optimum switching point for the PWM frequency
- 4: Maximum power-module current at low PWM frequency
- 5: Maximum power-module current at high PWM frequency (I_{max} of power module)

Please note that switching the PWM frequency always influences both PWM outputs of a drive-control board. Switching a single PWM output is not possible. This means that only one of the two outputs can be used. Besides, a PWM output with a PWM frequency > 5 kHz requires increased performance as well as software option 49, see page 1074.

Summary:

- The speed is switched at the intersection of the two current curves (I_{\max} of motor, I_{\max} of power module) so that no inconsistencies in the torque behavior of the motor occur.
- For better controllability (no harmonics at higher PWM frequencies), it might already make sense to switch at lower speeds.
- The best speed to switch at must be determined by experimenting. The value above should serve as an initial value.

MP2186.x Speed-dependent switching of the PWM frequency

Input: 0 to 100 000 [rpm]
MP2186.x specifies the shaft speed at which the PWM frequency is switched to twice the PWM frequency

MP2188.x Speed-dependent switching of the PWM frequency

Input: 0 to 100 000 [rpm]
MP2188.x specifies the shaft speed at which the original PWM frequency is returned to from the doubled PWM frequency (as the result of MP2186.x).
MP2188.x must be < MP2186.x



7.5 PLC cycle time

On the CC 61xx, the PLC cycle time is a multiple of the HSCI cycle time (3 ms). Minimum possible value is therefore 12 ms.

► Enter the PLC cycle time in ms in MP7602.

MP7602	PLC cycle time
Input:	0 to 60 [ms]
	0 to 12: 12 ms

7.6 Monitoring functions

The KTY temperature sensor of the motors is monitored by the control for excessive and insufficient temperatures. If the KTY is not to be evaluated (e.g. because the temperature sensor is not doubly isolated), this function must be deactivated with MP2220.x bit 4 = 1.

The CC 61xx monitors the input frequency of the speed encoders. If this monitoring leads to problems (e.g. unjustified responses), it can be deactivated with MP2220.x bit 7 = 1. The following error messages can appear:

- Speed encoder: **8860 Input frequency from speed encoder <axis>**
- Position encoder: **8870 Input frequency from position encoder <axis>**

Only one encoder is used for capturing the speed and position when linear and torque motors are being used. If such an axis is removed from the closed-loop control and later reintroduced, a mechanical offset can occur. This offset is not fixed "in one blow," but instead is adjusted by gradually raising the k_V factor from 0 to the original value. This function is deactivated with MP2220 bit 8 = 1.

MP2220.x Monitoring functions

Input:	Bit 4 – Monitoring for excessive temperature
	0: Active
	1: Inactive
	Bit 5 – Monitoring for insufficient temperature
	0: Active
	1: Inactive
	Bit 6 – Reserved
	Bit 7 – Monitoring of encoder input frequency
	0: Active
	1: Inactive
	Bit 8 – Adjust mechanical offset by gradually increasing the k_V factor
	0: Active
	1: Inactive

7.7 Special functions

7.7.1 Multifunction filter

With the CC 61xx, you can influence the manipulated variable of the speed controller (= nominal current) and the position controller (= nominal speed) by means of up to five freely definable filters per axis. These filters are multifunctional filters, which means that the filter type of each individual filter order can be selected as desired. They are also effective for the spindle(s).

Objective of the filters

The first objective when adjusting a machine is the optimization of the control loop in the current and speed controller. The increase of the P component of the control loops in order to raise the dynamics of the machine is the main aspect of this. If a control loop is at the oscillation limit, these oscillations can be damped with filter functions, so that the P components can be increased again.

The second objective when adjusting a machine is the optimization of the position controller. Here it is attempted to increase the k_V factor in the position controller, in order to simultaneously increase the machine's performance (the acceleration behavior, for example). The procedure is always to increase the k_V factor to the oscillation limit, damp these oscillations with the filters, and then increase the k_V factor again.

Types of filters

Three different types of filters per axis are available for selection:

■ PT2 low-pass

- Use:
 - Oscillations in the upper frequency range (typically: from 500 Hz)
 - High-frequency noises on axes (such as during switch-on)

■ Band-rejection filter

- Use:
 - Oscillations in the middle frequency range (typically: between 100 Hz and 2.5 Hz)
- Typical settings:
 - Damping from 6 dB to 9 dB
 - Bandwidth: equal to the center frequency, constant from 500 Hz
- Disadvantage:
 - These can strengthen oscillations in the lower frequency range

■ Phase increase

- Use:
 - Oscillations in the lower to middle frequency range, which occur because of an insufficient phase reserve
 - Oscillations in the lower frequency range, for which band rejection would excessively decrease the amplitude
- Typical settings:
 - Phase from 20° to 80°
 - Center frequency: Frequencies from 3 Hz to 400 Hz
 - Bandwidth: Equal to the center frequency (oscillation frequency)
- Disadvantage:
 - The control-loop gain above the center frequency is increased. The increased use of band-rejection filters can become necessary, or the P component might need to be reduced.
- Note: After the settings have been made, the stability of the control loop must be checked again (P and I component)

Recommended types of filters

Experience has shown that the band rejection of the multifunction filters is to be used for damping oscillations in the **speed controller**.

On the other hand, the tendency of the **position controller** to oscillate should be counteracted with IPC (Integral Phase Compensation). Only if this adjustment does not lead to the desired result can the multifunction filters such as the phase increase (better, since it does not facilitate oscillations at lower frequencies as much) or the band-rejection filter be used.

Since the ambient conditions can be so different, the use of the filters must be checked separately in every case. The TNCopt PC software from HEIDENHAIN should always be used, so that the sequence of the adjustment matches the ideal case. This manual can only present recommended guidelines and procedures.

Possible settings for multifunction filters

	Filter 1	Filter 2	...	Filter 6
Selection of filter type 0 = Filter not active 1 = PT2 low-pass (speed ctrl.) 2 = Band-rejection (speed ctrl.) 3 = Phase increase (speed ctrl.) 11 = PT2 low-pass (position ctrl.) 12 = Band-rejection (position ctrl.) 13 = Phase increase (position ctrl.)	MP2562.x	MP2563.x	...	MP2567.x
<ul style="list-style-type: none"> ■ PT2 low-pass: No effect ■ Band rejection: Damping [dB] ■ Phase increase: Phase [0- 90°] 	MP2542.x	MP2543.x	...	MP2547.x
<ul style="list-style-type: none"> ■ PT2 low-pass: Cutoff freq. [Hz] ■ Band-rejection: Center freq. [Hz] ■ Phase increase: Center freq. [Hz] 	MP2552.x	MP2553.x	...	MP2557.x
<ul style="list-style-type: none"> ■ PT2 low-pass: No effect ■ Band-reject.: Bandwidth [Hz] ■ Phase increase: Bandwidth [Hz] 	MP2572.x	MP2573.x	...	MP2577.x

The filters can be used in the position or speed controller in any combination, i.e. even if the first filter is deactivated for the 1st axis,

- MP2542.0
- MP2552.0
- MP2562.0 = 0
- MP2572.0

the second filter, for example (in this case phase increase in the speed controller), can be activated for the first axis:

- MP2543.0 = 40
- MP2553.0 = 120
- MP2563.0 = 3
- MP2573.0 = 120

Filters in the position controller

The k_V factor can be increased by using the filters in the position controller. After the k_V factor has been increased up to the oscillation limit, a band-rejection filter can be defined for the oscillation frequency so that the k_V factor can be further increased.

MP2542.x Damping/phase increase for filter 1

Input: 0 to 40.0 [db]

MP2543.x Damping/phase increase for filter 2

Input: 0 to 40.0 [db]

MP2544.x Damping/phase increase for filter 3

Input: 0 to 40.0 [db]

MP2545.x Damping/phase increase for filter 4

Input: 0 to 40.0 [db]

MP2546.x Damping/phase increase for filter 5

Input: 0 to 40.0 [db]

MP2547.x Damping/phase increase for filter 5

Input: 0 to 40.0 [db]

MP2552.x Center/cutoff frequency for filter 1

Input: 0 to 30000.0 [Hz]

MP2553.x Center/cutoff frequency for filter 2

Input: 0 to 30000.0 [Hz]

MP2554.x Center/cutoff frequency for filter 3

Input: 0 to 30000.0 [Hz]

MP2555.x Center/cutoff frequency for filter 4

Input: 0 to 30000.0 [Hz]

MP2556.x Center/cutoff frequency for filter 5

Input: 0 to 30000.0 [Hz]

MP2557.x Center/cutoff frequency for filter 5

Input: 0 to 30000.0 [Hz]

MP2562.x **Filter type for filter 1**
Input: 0: No filter
 1: PT2 low-pass filter (speed controller)
 2: Band-rejection filter (speed controller)
 3: Phase increase (speed controller)
 11: PT2 low-pass filter (position controller)
 12: Band-rejection filter (position controller)
 13: Phase increase (position controller)

MP2563.x **Filter type for filter 2**
Input: 0: No filter
 1: PT2 low-pass filter (speed controller)
 2: Band-rejection filter (speed controller)
 3: Phase increase (speed controller)
 11: PT2 low-pass filter (position controller)
 12: Band-rejection filter (position controller)
 13: Phase increase (position controller)

MP2564.x **Filter type for filter 3**
Input: 0: No filter
 1: PT2 low-pass filter (speed controller)
 2: Band-rejection filter (speed controller)
 3: Phase increase (speed controller)
 11: PT2 low-pass filter (position controller)
 12: Band-rejection filter (position controller)
 13: Phase increase (position controller)

MP2565.x **Filter type for filter 4**
Input: 0: No filter
 1: PT2 low-pass filter (speed controller)
 2: Band-rejection filter (speed controller)
 3: Phase increase (speed controller)
 11: PT2 low-pass filter (position controller)
 12: Band-rejection filter (position controller)
 13: Phase increase (position controller)

MP2566.x **Filter type for filter 5**
Input: 0: No filter
 1: PT2 low-pass filter (speed controller)
 2: Band-rejection filter (speed controller)
 3: Phase increase (speed controller)
 11: PT2 low-pass filter (position controller)
 12: Band-rejection filter (position controller)
 13: Phase increase (position controller)

MP2567.x **Filter type for filter 5**
Input: 0: No filter
 1: PT2 low-pass filter (speed controller)
 2: Band-rejection filter (speed controller)
 3: Phase increase (speed controller)
 11: PT2 low-pass filter (position controller)
 12: Band-rejection filter (position controller)
 13: Phase increase (position controller)



MP2572.x **Bandwidth for filter 1**
Input: 0 to 30000.0 [Hz]

MP2573.x **Bandwidth for filter 2**
Input: 0 to 30000.0 [Hz]

MP2574.x **Bandwidth for filter 3**
Input: 0 to 30000.0 [Hz]

MP2575.x **Bandwidth for filter 4**
Input: 0 to 30000.0 [Hz]

MP2576.x **Bandwidth for filter 5**
Input: 0 to 30000.0 [Hz]

MP2577.x **Bandwidth for filter 5**
Input: 0 to 30000.0 [Hz]



7.7.2 Filter order for separate low-pass filter in the speed controller

Application

If a low-pass filter is used with the CC 61xx, the filter order of the low-pass filter can be set in MP2560.x. However, MP2560.x = 0 should be entered under standard conditions.

Spindle: High-frequency spindles often cause considerable current noise (shown by **I_{nom}** or **Utilization** on the oscilloscope). The optimization goal is as little current noise as possible at high spindle speeds. MP2560.x = 10 to 20 should be used as a starting value.

Axes: The low-pass filter should be used if the actual speed has "a lot of noise," for example MP2560.x = 1 or 2. If there is no improvement, set MP2560.x = 0.

For adjustment of the filters, see "Adjustment of the speed controller" on page 1133.

Machine parameter

MP2560.x

Filter order of the low-pass filter

Input:

0 to 20

Recommended input value: 0

Recommended input value if much current noise from high-frequency spindles: 10 to 20

7.7.3 Dynamic determination of load

General information

During traverse motions, each machine axis behaves differently depending on the various loads, such as the differing inertias of the workpieces.

In connection with the CC 61xx controller unit, the iTNC offers you the possibility of optimizing the machine according to load classes. Before the machining process, the optimum settings for a workpiece with a certain load class can be entered.

Determining the load

The effects of the inertia of the workpieces can be detected with a measuring cycle created by the OEM. A measurement that determines the maximum occurring torque (target value in % of the rated torque without decimal places) between two points in time is started. At the same time, the machine is optimized for this load class, and the machine parameters determined are saved in a machine-parameter subfile.

The load class is determined for each workpiece before machining starts (via manual entries or by determining the load with an NC macro), and the corresponding, optimized machine-parameter subfile is selected.

Procedure

The following procedure is recommended for determining and using a load class.

By the OEM: Determining the load class and creating the machine-parameter subfiles

- ▶ Optimizing the machine and saving the results without load
- ▶ Loading the machine with a mass appropriate for the load class to be determined
- ▶ Starting the measurement, traversing the axis to be measured, and stopping the measurement

Example of a corresponding NC program:

```
0  BEGIN PGM measure_a MM
1  L C+0 RO FMAX
2  FN 17: SYSWRITE ID 621 NRO IDX4 = +0
3  L IA+180 RO FMAX
4  L IA-180 FMAX
5  FN 18: SYSREAD Q1 = ID621 NRO IDX4
6  END PGM measure_a MM
```

- ▶ Optional: Saving the measured results in a variable:
FN 17 SYSWRITE ID 590 NR 2 IDX <axis> (IDX = 1 to 30 (is not deleted during program selection)
FN 17 SYSWRITE ID 590 NR 3 IDX <axis> IDX = 1..30 (protected during a power failure)
- ▶ Optional: Reading the saved value
FN 18 SYSREAD ID 590 NR <2 or 3> IDX <axis> (IDX = 1 to 30)
- ▶ Optimizing the machine and saving the results in a machine-parameter subfile for this load class
- ▶ Performing the measurements for each load class as described above



By the customer: Selecting the machining method optimized for the load

- ▶ Entering the load class (or mass) before machining, or determining the load class with the NC macro from the OEM as described above.
- ▶ Activating the correct machine-parameter subfile depending on the measured result

FN 17 SYSWRITE ID 1020 NR 1 = <line number> (when using the keyword **MPFRAGMENTFILE = <path/file name>** in OEM.SYS)

- ▶ Deactivating the machine-parameter subfile

FN 17 SYSWRITE ID 1020 NR 2

within an NC macro, which is run when a program is concluded with M02, M30 or END PGM (keyword RUNENDPGM in NCMACRO.SYS). In the Manual operating mode, this macro must also be saved in the M function table under M02 and M30.



Note

- Retention forces and friction forces are included in the measured value
- Deviations in the utilization due to oscillations or torque ripple are included in the measured value
- The override of the feed-rate potentiometer is taken into account
- The determined value depends on the axis parameters. Use axis parameters with stable behavior for the determination

Module 9046 Copy controller data to the PLC memory

Module 9046 is used to copy controller data (data of the CC) to the PLC memory via configured channels.

Constraints:

- The module must not be called in the cyclic PLC program part.
- The module should only run once for each channel. Once a channel has been configured successfully, the programmed CC data is copied to the PLC memory at the PLC clock rate.
- If a channel that has already been configured must be reconfigured (other CC data or other axis), the channel must be reset first. Direct reconfiguration leads to a module error.

Call:

PS	B/W/D/K	<Mode> 0: Configuration of cyclic data channels from the CC to the PLC
PS	B/W/D/K	<Axis number> 0 to n: Dependence on MP4020 bit 14
PS	B/W/D/K	<Code for CC data> Mode 0: -1: Reset channel (default setting) 0: Mass moment of inertia of the table [kg*m2] 1: Mass of the table [kg] 2: Total mass moment of inertia (table and motor) [kg*m2] 3: Total mass (table and motor) [kg] 4: Total power [W] 5: Axis power [W] 6: Torque-producing nominal current (with sign) [mA] 7: Total actual current (with sign) [mA]
PS	B/W/D/K	<Data channel> Mode 0: 0 to 15, up to 16 channels can be programmed
PS	B/W/D/K	<Target address in PLC memory> Mode 0: Double-word address to which the value of the programmed data channel is copied
CM	9046	
PL	B/W/D	<Error code> 0: No error. Data channel was (already) configured 1: Call was not in a submit/spawn job 2: NC/CC does not support the desired function 3: Invalid value in the <Mode> parameter 4: Invalid value in the <Axis number> parameter 5: Invalid value in the <Code for CC data> parameter 6: Invalid value in the <Channel number> parameter 7: Invalid double-word address in the <Target address in PLC memory> parameter 8: Programmed data channel is already being used 100: Programmed CC data is not available for programmed axis



Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Axis does not exist or status interrogation faulty
W1022	1	Invalid value for Mode parameter, CC code
	2	Invalid axis number programmed
	3	Invalid value for Channel parameter programmed
	4	Memory range for the programmed double-word address is too small or does not exist
	5	Programmed address is not a double-word address (not divisible by 4)
	6	Programmed data channel is already being used
	20	Call was not in a submit/spawn job
	45	Internal error
	51	Programmed CC data is not available for this axis on the CC, or the PLC module is not supported by this NC software
	61	Internal error

7.7.4 LIFTOFF function

General information

During an NC stop, a lift-off movement in the direction of the tool axis is superimposed over the deceleration along the path. This movement for protecting the workpiece is referred to as LIFTOFF.

LIFTOFF is possible in case of:

- NC stop if this is configured in M4620. Distance by which the tool is lifted off: 0.1 mm.
- Power failure and other error reactions by the CC intended for this purpose if the constraints described below are maintained. The distance by which the tool is lifted off can be configured in MP1160.
- PLC error message, for which the reaction NC-LIFTOFF is contained in the PET table. Distance by which the tool is lifted off: 0.1 mm.



Note

- If there is an emergency stop, no LIFTOFF is performed. The servo drives will then be decelerated along the emergency braking ramp as quickly as possible.

The following conditions must be met for a LIFTOFF to be performed:

- LIFTOFF is enabled by NC program via M148 = active
- Tool is enabled in the tool table via column **LIFTOFF** = Y
- LIFTOFF is enabled by PLC via M4620 = 1

The velocity at which the tool is lifted off depends on the retraction path. The necessary movement is always performed in a time of 0.1 seconds.

LIFTOFF in case of power failure

If the power fails and LIFTOFF is enabled (M148 must be active, column **LIFTOFF** in the tool table = Y, PLC: M4620=1), an attempt is made to lift the tool from the contour by the distance given in MP1160 with the help of the remaining energy of the DC-link.

Certain conditions must be maintained before and during LIFTOFF:

- The 24 V power supply must be maintained for at least 1 second (UPS for 24 V, or buffer capacity or capacitor). HEIDENHAIN offers the PSL 13x or CML 110 (Capacitor Module for Low Voltage) for this.
- The current and speed controllers may not be switched off (e.g. via PLC Module 9161)
- AC-fail monitoring (MP2195 Bit 7 = 0) and/or Uz monitoring are/is active (MP2195 Bit 9 = 0). Both monitoring functions lead to LIFTOFF if they are activated and respond. HEIDENHAIN recommends activating both monitoring functions to provide the quickest reaction possible in case of a power failure or a failure of the power supply module.
- The wye-delta contactor combination may not fail, otherwise the spindle would not be controllable during LIFTOFF.



With the CC 61xx and the UEC 1xx, the LIFTOFF function can also be performed if regenerative or non-regenerative UV power supply modules are being used. In this case, the controller unit itself must measure the Uz dc-link voltage. This results in the following voltage thresholds for LIFTOFF: If the DC-link voltage falls below the threshold of 450 V, the spindle is decelerated to feed energy into the DC-link. Until now the spindle was accelerated again once the DC-link voltage exceeded the threshold of 450 V. Now the spindle is decelerated until the DC-link voltage has risen above 500 V. Then the spindle speed is maintained. This process can be repeated as often as necessary if the DC-link voltage falls below the threshold of 450 V. At a voltage of 400 V the initialization of LIFTOFF is reported to the MC and the spindle is retracted from the workpiece, at a voltage of 350 V the spindle is decelerated systematically in order to maintain the DC-link voltage, at a voltage of 300 V the powerfail signal is set by the controller unit.



Note

- LIFTOFF only functions with HEIDENHAIN inverters.



Note

Perform the LIFTOFF function on the machine to check whether the tool lifts off the complete distance defined in MP1160 without problems. Due to the dynamics of a machine, it may occur that the LIFTOFF (if values > 10 mm are defined in MP1160) cannot be executed completely and is aborted with an error message.



Danger

- Limit-switch monitoring is not active during LIFTOFF.
- LIFTOFF must be deactivated while exchanging tools.

MP1160 **LIFTOFF in case of power failure**
Input: 0 to 30.0000 [mm]
 Default: 0.1 [mm]

Additional settings

Since the residual energy of the DC-link is only available for a controlled LIFTOFF for a very short time, under certain circumstances the prescribed distance may not be reached. Since even times on the scale of milliseconds can be of use here, a special possibility was created in order to react to a power failure more quickly. The starting point is the processing time for the HSC nominal position value filters, which, due to the number of filter coefficients (filter order) to be calculated, lead to a delay in the reaction time. This number of filter coefficients to be calculated can be reduced with MP1262 (filter order for HSC filters) and MP1263 (filter order for advanced HSC filters). The starting point is the filter order of 31 recommended by HEIDENHAIN, which is entered as a default value and should suffice for the LIFTOFF function in most cases.

If this is not the case, then you can reduce the reaction time via the filter order. Since the coefficients are calculated in the interpolator clock cycle of usually 3.0 ms, the reaction time decreases correspondingly per decrement of the filter order.



Note

A modification of the filter order means a **basic change** in the filter characteristics of the nominal position value filters set, and can therefore have significant influences on the behavior of the machine. The effects must be checked individually, and the nominal position value filters must be re-adjusted if necessary.

Starting from the following settings for MP1262 and MP1263, checking and re-adjustment of the nominal position value filters is strongly recommended:

- $MP1262 \leq 1.1 / (MP1212 * \text{interpolator cycle time})$
- $MP1263 \leq 0.67 / (MP1213 * \text{interpolator cycle time})$

MP1262 Filter order used for HSC filters

Input: 0 to 31 [filter order]
Default: 31

MP1263 Filter order used for advanced HSC filters

Input: 0 to 31 [filter order]
Default: 31

The CC 424(B) controller board was used to define voltage thresholds in MP2192 and MP2194, which were used to configure the LIFTOFF function. These voltage thresholds are now defined by the HSCI CC 61xx controller unit itself, depending on the power supply module being used. The associated machine parameters MP2192 and MP2194 are no longer required. If you do not need the LIFTOFF function, you can use bit 9 in MP2195 to deactivate monitoring of the DC-link voltage.

The CC 61xx makes the following distinction: It takes into account whether the power supply module is:

- a regenerative power supply module with servo-controlled DC-link voltage, or
- a nonregenerative power supply module where, due to changes in the line voltage and load on the DC link, voltage fluctuations must be considered.

MP2195 Monitoring of the DC-link voltage

Input: Bit 9
0 = Uz monitoring and LIFTOFF function are active
1 = Uz monitoring and LIFTOFF function are not active



7.7.5 TRC – Torque Ripple Compensation

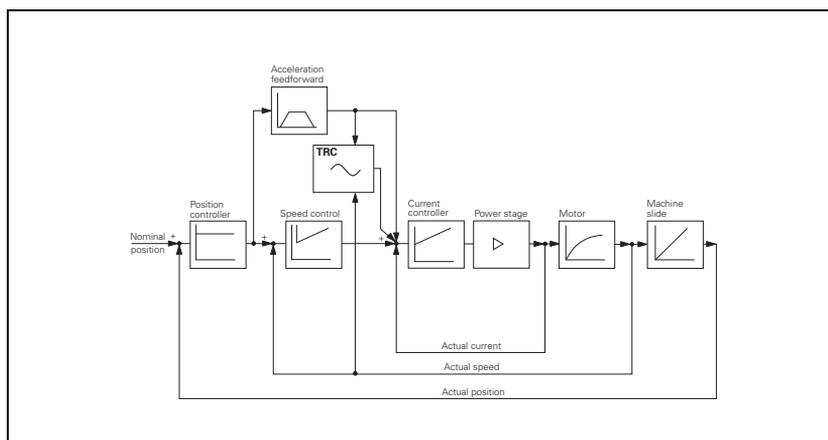
General information

Certain motors with permanent magnets (linear, torque and some synchronous motors) have an increased, position-dependent variation of the motor torque (not QSY motors from HEIDENHAIN). This can be the result of two things:

- During idle running, the cogging torque due to attractive forces of the permanent magnets
- When under load, the torque ripple from higher harmonics, resulting from the opposing electromotive forces (generator effect)

In practice, both causes always occur together, i.e. the torque of the motor is subject to periodic oscillations. This can have a negative effect on the controllability of the motor, which can result in a greater following error, and under circumstances, lower surface quality of the workpiece.

To compensate for the **cogging torque**, a compensation current ascertained specifically for each motor can now be added.



Activating TRC

TRC can only be activated via a special compensation file. The settings in this file can only be made with the TNCopt commissioning software from HEIDENHAIN (as of version 2.3). Please refer to the TNCopt documentation. The compensation current is determined with a special method for measurement, and the parameters for calculating this compensation are stored in a compensation file on the control. The iTNC 530 then takes these parameters into account when calculating the controller parameters.

- Directory: **PLC:\MP\TRC**
- File name: **xx_<Motor_name>.TRC** or new **xx_<Motor_name>.CMP**
 - **xx**: Index of the axis (e.g. 00 = 1st axis, X axis)
 - **<Motor_name>**: Name of the motor from the motor table (max. 29 characters)
 - **.CMP** (old .TRC): File extension for "Torque Ripple Compensation"

An entry in MP2260.x specifies whether torque ripple compensation is to be performed for an axis. If MP2260.x is followed by a blank line (no entry), no compensation current is calculated for this axis.

Example:

MP2260.0: **00_MotNameFromMotTab;Motor of 1st axis**

MP2260.1: **;Motor of 2nd axis, no compensation**

With MP2261.x, compensation values contained in the compensation file (*.cmp, defined in MP2260) can be deactivated for specific axes. For master-slave torque control, a separate TRC file can now be defined for the master axis and the slave axis in order to compensate gear-transmission errors separately.



Note

- The TRC function can only be used with PWM frequencies up to 5 kHz.
- A TRC file can only be used on the control on which the adjustment has been made.
- A TRC file must be re-created if the motor, rotary encoder or angle encoder is exchanged.
- A TRC file can only be generated for synchronous motors or for linear or torque motors.

MP2260.x Name of the file for "torque ripple compensation" (TRC)

Input: xx_<MotorNameFromMotorTable>.CMP (generated in TNCopt). If no .CMP is found, .TRC is accessed
No input: No compensation

MP2261.x Deactivate compensation

Input: %0000000000000000
1 = Compensation not active
Bit 0: Torque ripple compensation
Bit 1: Gear error compensation



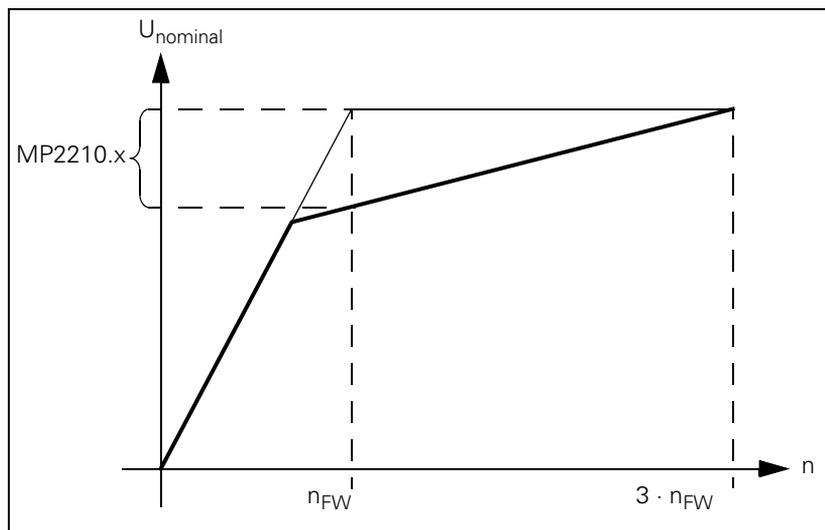
7.7.6 Peculiarities in weakened-field operation

Reduction of the magnetization current

Some of the asynchronous spindle motors require a high magnetizing current at low speeds ($n < n_{\text{field weakening}}$). This magnetizing current can, however, lead to thermal motor problems at the rpm for field weakening n_{FW} when there is no load.

The nominal voltage (and so also the nominal magnetization current) at the rpm for field weakening n_{FW} when there is no load can be reduced with MP2210.x. The entered reduction results in a nominal voltage curve as shown in the figure below. The maximum nominal voltage is reached when $n = 3 \cdot n_{\text{field weakening}}$. The nominal voltage can be reduced by max. 60 % (MP2210.x = 60).

If a load is placed on the drive, then the nominal voltage is increased again automatically in order to improve the dynamics.



MP2210.x Reduction of the nominal voltage at the rpm for field weakening when there is no load

Input: 0 to 60 [%]
0: Function inactive



Note

Please note that the reduction of the magnetization current can lead to a limitation of the drive's dynamics. However, this depends on the drive, and must be judged separately in each case.

Setting the reduction of the magnetization current

If thermal problems arise on an asynchronous spindle with no load during weakened-field operation, and these problems cannot be traced to an overload or other mechanical problems, then please reduce the magnetization current as follows:

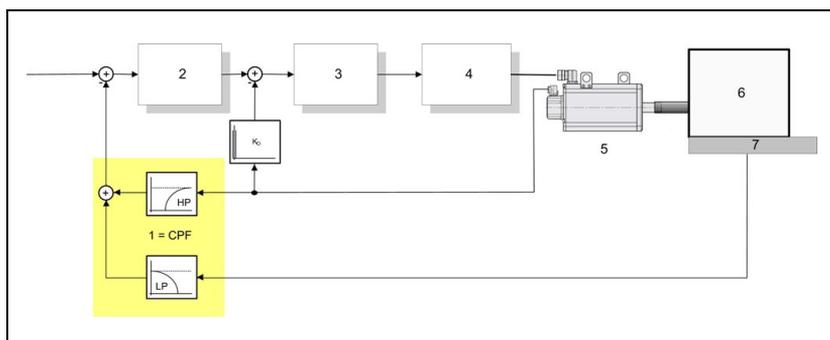
- ▶ Reduce the nominal voltage in steps of 10 % in MP2210.x.
- ▶ Reduce MP2210.x until a stable temperature within the motor specifications is reached when there is no load.

7.7.7 Crossover Position Filter (CPF)

The CPF can be used only in dual-encoder systems with rotary encoder and position encoder. In such systems, the CPF is used to passively dampen certain machine vibrations caused by the drive components in the drive chain. In order to stabilize the position control loop in such systems with resonances, the position signal from the position encoder, which is filtered through a low-pass filter, and the position signal from the motor speed encoder, which is filtered through a high-pass filter, are added and made available to the position controller as actual position value. The filter frequency (separation frequency) is set in MP2609 for this. The typical filter frequency values in MP2609 are between 20 Hz and 80 Hz.

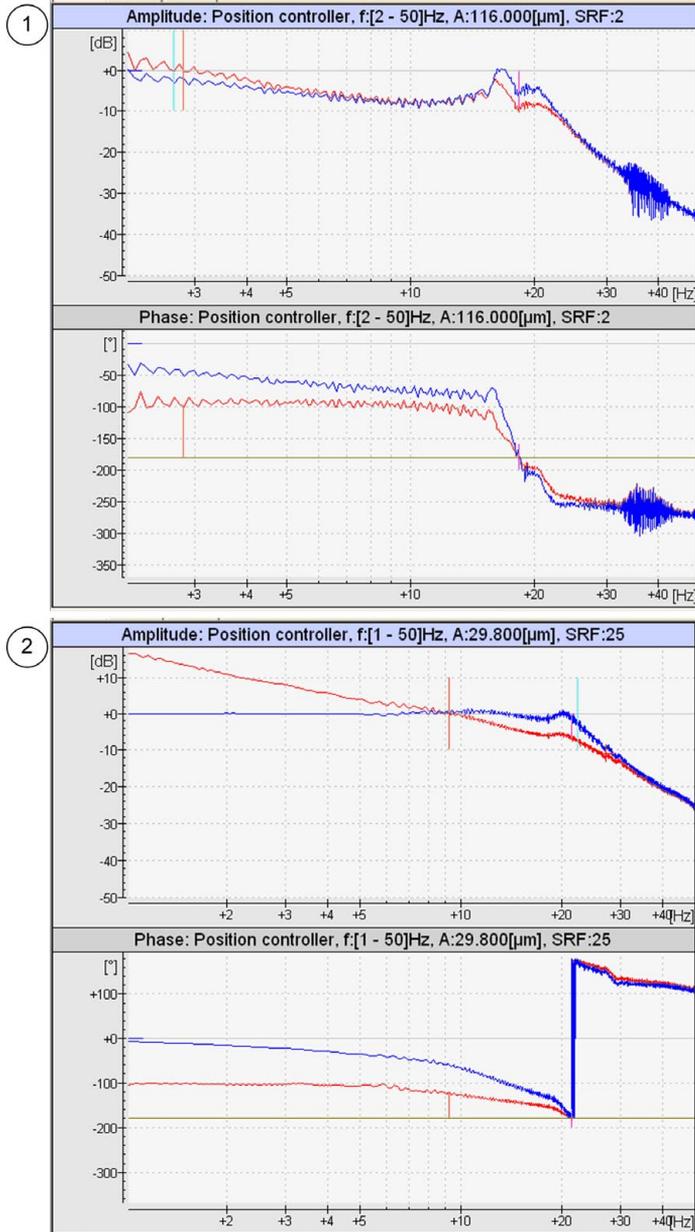
The separation frequency can be determined with the position-controller Bode diagram of the TNCopt PC tool. The frequency at which an excessive resonance occurs is to be selected as the value for MP2609. After the separation frequency has been defined, the kV factor can be increased.

The block diagram shows how the crossover position filter works:



- 1: Crossover position filter (CPF) with high-pass (HP) and low-pass (LP)
- 2: Position controller
- 3: Speed controller
- 4: Current controller – power module
- 5: Motor
- 6: Machine
- 7: Linear encoder

The two Bode diagrams illustrate the procedure for setting the separation frequency for the CPF. Diagram 1 shows that the excessive resonance is reached at 15 Hz. Diagram 2 shows the frequency response after 15 Hz has been entered in MP2609.



MP2609.x Filter frequency for CPF filter
 Input: 0 to 300.0 [Hz]

7.8 Stick-slip friction compensation at quadrant transitions

With the CC 61xx, the stick-slip friction compensation already begins before the zero crossover of the speed.

With the CC 61xx, the parameters MP2612.x and MP2614.x now function with respect to distance (unit: [mm]).

This makes it possible to compensate quadrant transitions independently of velocity, acceleration, and diameter.

If stick-slip compensation is used at the same time that an HSC filter is being used, then in rare cases unwanted movements of the machine can occur (brief movement in wrong/reversed direction). You can avoid this effect by using the triangle filter instead of the HSC filter.

MP2610.x Friction compensation at low motor speed

Input: 0 to 30.0000 [A] (effective value)
0: No friction compensation

MP2612.x Distance before the reversal point from which a reduction of the current from MP2610.x is to go into effect

Input: 0 to 1.000 [mm] or [°]
0: No friction compensation
0.1: Typical input value

MP2614.x Distance after the reversal point from which the current from MP2610.x is to go into effect again

Input: 0 to 1.000 [mm] or [°]
0: Friction compensation same as CC 422
0.1: Typical input value



Module 9311 Dynamically change values for friction compensation

Module 9311 is used at run-time to prescribe other values for the friction compensation. The original values from MP2610.x, MP2612.x and MP2614.x are temporarily overwritten in the DSP. The MP file remains unchanged.

Conditions:

- This function is supported as of the DSP hardware CC 424(B) and CC 61xx.

Call:

PS B/W/D/K <Axis number>

PS B/W/D/K <Current in [mA]>

0 to 30000 replaces the value in MP2610.x

PS B/W/D/K <Path in [0.1 μm]>

0 to 10000 replaces the value in MP2612.x

PS B/W/D/K <Path in [0.1 μm]>

0 to 10000 replaces the value in MP2614.x

CM 9311

Error code:

Marker	Value	Meaning
M4203	0	New values assumed for axis number
	1	Error code in W1022
W1022	1	Invalid value as replacement for machine parameter
	2	Invalid axis number programmed
	19	Function is not supported
	24	Call was not from a cyclic program

7.9 Field orientation

General information

If a linear, torque or synchronous motor is used with an incremental encoder without a Z1 track or an unaligned encoder with EnDat interface, there is no assignment between the encoder and the rotor magnets. The field angle must be determined before this motor can be moved.

The iTNC 530 uses the "field orientation" function to determine the field angle for the motors listed above. The assignment between the encoder and the rotor magnet (field angle) is determined and stored.



Note

The "field orientation" function can be performed only if the current controller is already adjusted!

Regarding the **motor.mot** motor table, the field orientation must be performed for the following drives:

- Linear motor with absolute encoder with EnDat interface (**SYS = 3**)
- Synchronous or torque motor with unaligned rotary encoder with EnDat interface (**SYS = 5**)
- Synchronous or torque motor with incremental rotary encoder without Z1 track (**SYS = 6**); one reference mark per revolution
- Synchronous or torque motor with incremental rotary encoder with distance-coded reference marks (**SYS = 7**)
- Linear motor with incremental linear encoder with distance-coded reference marks (**SYS = 8**)

Absolute encoder with EnDat interface	Incremental encoders
As soon as the absolute position of the encoder has been read, the absolute position and determined field angle are assigned to each other. The field angle is assigned to the zero position of the encoder.	After switching on the drive, the motor orients itself (rough orientation; the message Finding the field angle appears). The drive is ready for operation after this procedure. The field angle is determined and assigned as soon as the reference mark/s is/are traversed during the first motor motion.



Danger

If the speed encoder is exchanged, the Field Orientation function must be rerun.

Field orientation with absolute encoder (EnDat)

As soon as the absolute position of the encoder has been read, the absolute position and determined field angle are assigned to each other. The field angle is assigned to the zero position of the encoder.

Field orientation via encoder with Z1 track

After switching on the control, the motor orients itself (rough orientation) via the Z1 track of the encoder. The drive is ready for operation after this procedure. The field angle is determined and assigned as soon as the reference mark is traversed during the first motor motion.

General information about encoders for direct drives

- An absolute encoder with EnDat interface should be used, since the absolute position value is available directly after switch-on, and the field angle can be assigned immediately. This means that the motor can be controlled immediately.
- The encoder should have a high line count. This leads to better controllability.
- With incremental encoders the motor must first be moved a "minimum" distance in order to determine a field angle with which the motor can be moved until the reference mark. Only after the reference mark has been traversed can the field angle determined during commissioning be assigned.
- If excessive clamping of the axis prevents the "minimum" motion for determining the field angle, then no field angle can be determined and the axis cannot be controlled. In this case the clamping must be undone for the field angle to be determined. If this is not possible, because the axis would fall down, then an absolute encoder with EnDat interface must be used.

7.9.1 Possibilities for determining the field angle

There are various possibilities for determining the field angle:

- The field angle is determined automatically when the drive is switched on, without any motion of the motor. The method of determination is set in MP2250.x. The field angle is stored after it has been determined. This field angle is used when the motor is switched on again. The **FIELD ORIENT.** soft key has no function.
- By pressing the **FIELD ORIENT.** soft key once while the motor is being commissioned. The soft key appears in the **Commissioning Current Controller** operating mode. After pressing it, the motor moves. The field angle is determined and stored during this motion. This field angle is used when the motor is switched on again. A plausibility test is run during the field angle determination.



Attention

This method cannot be used for hanging axes (with 100 % weight compensation), since the brakes are not applied and the monitoring functions are deactivated!

- ▶ Select the method for field angle determination in MP2254.x.

HEIDENHAIN recommends using MP2254.x = 2 when commissioning new drive systems (such as machine prototypes), because the plausibility tests will be run. After successful commissioning, MP2254.x = 0 can be used to save time (such as for series production of the machine).

In certain cases it can be of advantage to determine the field angle via MP2254.x = 3. This mode can be used if

- there are no brakes,
- in the **Commissioning Current Controller** mode of operation, where the brakes are always open,
- the user ensures that the brakes can be opened manually or with the PLC.

MP2254.x Determining the field angle

Input: 0: Field angle is determined during operation; soft key has no function (without plausibility test)
2: Field angle determination via soft key, motor motion is permitted (with plausibility test)
3: Same as 2, but the drive must no longer be switched on by the PLC. The drive is moved immediately after the **FIELD ORIENT.** soft key is pressed.



Note

For synchronous spindles, the field angle should be determined via the **FIELD ORIENT.** soft key (MP2254.x = 2), since this is a more exact determination.

Plausibility test

This tests several machine parameters and parts of the circuitry for their plausibility:

- Encoder line count
- Number of pole pairs
- Rotational direction of the electrical field
- Traverse distance per electrical revolution



Note

This method for determining the field angle is recommended for commissioning, new designs, and other similar situations.

The following messages can appear during the plausibility test:

■ **8630 Field orient. successful**

Indicates that the field angle was successfully determined and stored in MP2256.x.

■ **8B10 Wrong traverse direction**

Indicates that the rotational direction of the electrical field does not match the counting direction of the encoder.

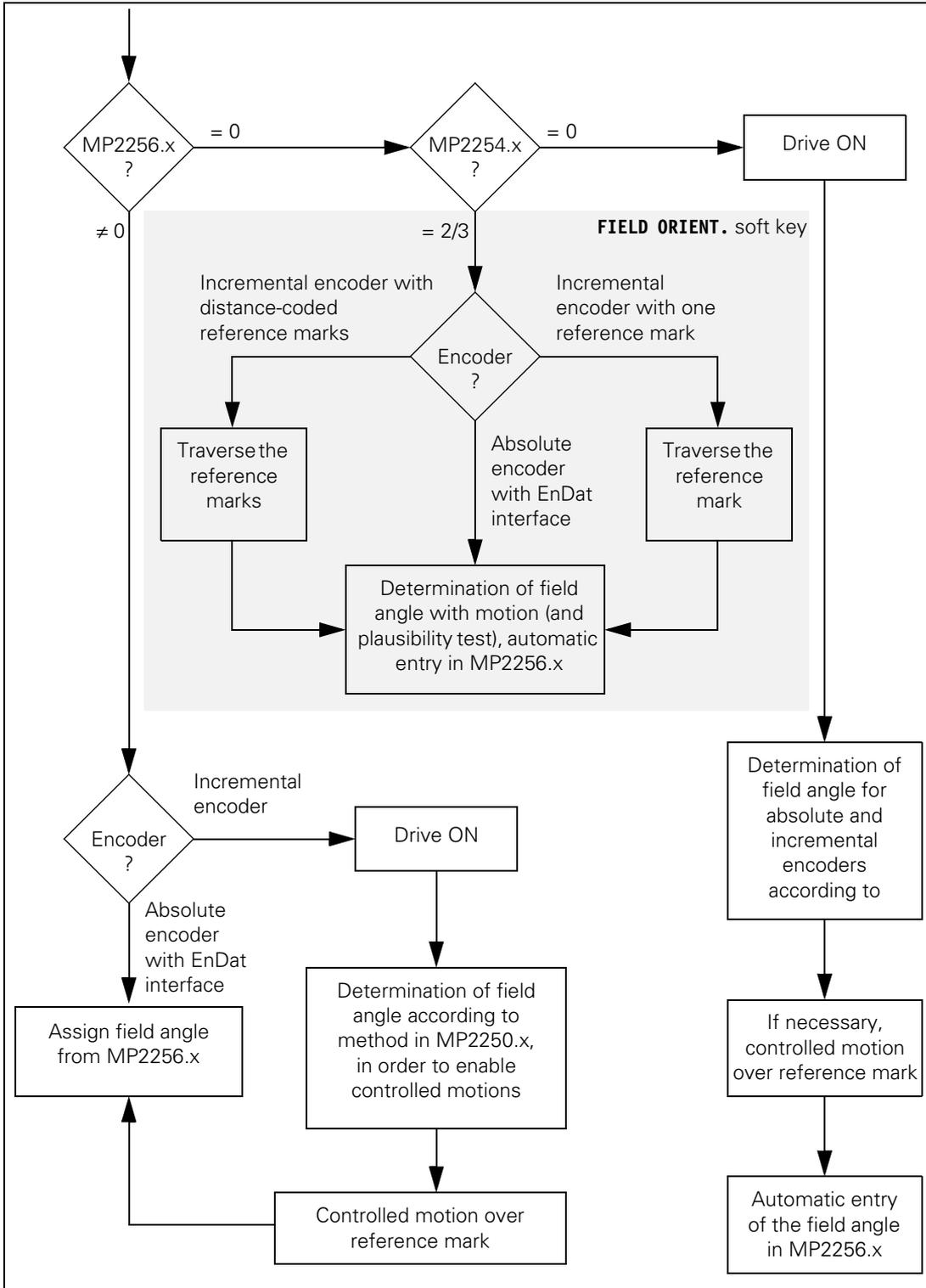
Error fix: Change the entry in MP2204.x.

■ **8B20 Error field orientation**

Indicates that no usable measurement results could be determined.

A common cause is incorrect values in MP331.x, MP332.x and MP1054.x. A further cause could be that the motor is moving against a resistance (e.g. brake is still active, bellows, limit stop) or that the mechanics are too stiff.

Overview of field orientation



7.9.2 Determination of the field angle without motor motion (MP2254.x = 0)

A distinction must be made if you intend to determine the field angle without motor motion:

- Commissioning: No field angle has been determined yet (MP2256.x = 0)
- A field angle has been already determined (MP2256.x ≠ 0)

The field angle is determined automatically after switching on the drive. This process lasts approximately 4 to 6 seconds (the PLC program must not rescind the drive enabling during this time). The **Finding field angle** progress bar appears.

If the power module is not active before the determination of the field angle begins, the error message **8B40 No drive release <axis>** appears. If the power module switches off during the determination, **8B50 Axis module <axis> not ready** appears.

Field angle not yet determined (MP2256.x = 0)

If the field angle on this machine was not yet determined after the control was started (MP2256.x = 0), the determination starts automatically. The method for determining the field angle is stored in MP2550.x. The determined field angle is stored in MP2556.x.

Field angle already determined (MP2256.x ≠ 0)

If the field angle on this machine was already determined after the control was started (MP2556.x ≠ 0), a distinction must be made:

- If an absolute encoder with EnDat interface is being used:
The absolute position of the encoder is read immediately after the control has been started. The field angle from MP2256.x is assigned to this position. Therefore, the first motor motion already occurs with the determined field angle.
- If an incremental encoder is being used:
Immediately after the control has been started and the control voltage has been switched on, then depending on MP2250.x a field angle is determined with which the motor can be traversed over the reference mark. After traversing the reference mark, the field angle from MP2256.x is assigned. The subsequent motor motions utilize the field angle from MP2256.x.

Determining the field angle

There are three methods for determining the field angle without motor motion:

- Method 2: Current pulses are output with the brakes applied, and the absolute rotor position is determined from the reaction. A "minimum" movement of the motor must be possible when the brakes are applied.
 - Method 3: Method 3 functions in the same manner as Method 2, but with the difference that the motor brakes are not applied. Therefore, this method is not suitable for hanging axes. However, this method can lead to more exact results than Method 2, so it should be used for synchronous spindles. Minimal spindle movements can occur during field angle determination.
 - Method 4: This new method was introduced for determining the field angle so that the relationship between the position of the incremental encoder and the position of the rotor magnets can be established even if there is considerable noise in the encoder signals.
- ▶ In MP2250.x, select the method to be used for determining the absolute rotor position.



Note

Standstill monitoring is active while determining the field angle. If it responds for motors without motor brakes, increase the threshold in MP1110.x. Afterwards, reset MP1110.x to the original value.

MP2250.x Determining the field angle without motor motion

Input: 0: Same as input value 2
1: Reserved
2: Method 2 (brakes applied)
3: Method 3 (same as Method 2, but motor brake is not applied)
4: Method 4 (if there is a lot of noise in the encoder signals)

MP2252.x Reserved

Input: Enter 0

7.9.3 Determination of the field angle with motor motion (MP2254.x = 2/3)

Since the motor moves in a certain direction while finding the field angle, it should be near the midpoint of the traverse path before the field angle is determined.

Axes with linear/torque motors can be slid "by hand" if the brakes are not applied.

While the field angle is being found, the speed controller and position controller are open and the drive controller is active. This means that the motor is moved (approx. 2 pole pairs) and the brake must be open until the field angle is determined.



Danger

**Hanging axes need 100 % weight compensation.
Please contact HEIDENHAIN if this is not the case.**



Attention

Limit switches are ignored!
If axes move into an illegal area, press the emergency stop button!



Note

When using incremental encoders with distance-coded reference marks, MP334.x (nominal increment between two fixed reference marks) must be set correctly.

MP2254.x = 2

The PLC initial servicing program, whose name and path are entered in the OEM.SYS file after the **PLCPWM** = entry, must ensure that the inverters are ready after "Switch on external dc voltage," but that the motor brakes are only open while determining the field angle. Alternately, the motor brakes can be opened manually for the duration of the field angle determination.

MP2254.x = 3

Under certain conditions, determination of a field angle with the help of the PLC is not necessary or desired. Here the motor is moved immediately after the **FIELD ORIENT**. key is pressed, and the field angle is determined. This mode can be used if

- there are no brakes,
- in the **Commissioning Current Controller** mode of operation, where the brakes are always open,
- the user ensures that the brakes can be opened manually or with the PLC.

Module 9065 Status of the commissioning function

Module 9065 is used to interrogate status information of commissioning functions dealing with the determination of the field angle, and with the commissioning of the current controller of an axis.

Conditions for the determination of the field angle:

- Synchronous, linear and torque motors determine the field angle each time the control is started if no EnDat or Z1-track encoders are used. For the duration of determining the field angle (about 5 to 7 seconds), <Mode> 1 returns bit-encoded the axes for which field-angle determination is active.
- Module 9162 reports that the speed controller is not ready while the field angle is being determined. A PLC error message can be suppressed if determining is active.

Conditions for commissioning the current controller:

- If no commissioning function is active, the value -1 is returned.

Call:

PS B/W/D/K <Mode>
 Determining the field angle:
 0: Axes for which the field angle is being determined with a
 commissioning aid and an internal oscilloscope
 1: Axes for which automatic determination of the field angle
 is active
 Commissioning the current controller:
 2: Axes for which the commissioning of the current
 controller is active
 3: Interrogation whether the spindle is in delta operation
 during the commissioning of the current controller (bit
 0 = 1)

CM 9065
PL B/W/D <Axes bit-encoded>
 Depends on MP4020 bit 14

Error code:

Marker	Value	Meaning
M4203	0	Axes have been determined
	1	Error code in W1022
W1022	1	Invalid value for mode



Before determining the field angle (**FIELD ORIENT.** soft key not yet pressed) the inverter must be in the following mode of operation:

- Green "READY" LED on
- Red "STO A" LED (previously SH1) off
- Red "STO B" LED (previously SH2) on (drive controller not ready, brakes closed)

As soon as the drive enable comes from the PLC, the **Finding field angle** message appears, otherwise **8B40 No drive release** appears.

The motor moves and the field angle is determined. Limit switches are not taken into account.

In order to avoid a possible error message resulting from standstill monitoring, the threshold for standstill monitoring during field orientation can be configured in MP1120.x.

- ▶ Switch the control on.
- ▶ Do **not** acknowledge the **Power Interrupted** message. In the **Programming and Editing** mode of operation, use the MOD key to enter the code number **688379**. The oscilloscope is started.
- ▶ Press the **I CONTROL** soft key.
- ▶ In the **Manual Operation** mode of operation, acknowledge the **Power Interrupted** message.
- ▶ Use the **SELECT AXIS** soft key in the oscilloscope to select the corresponding axis.
- ▶ Press the **FIELD ORIENT.** soft key.

The PLC must

- switch the drive on/off
- release and lock the brakes

The motor moves back and forth. The field angle is determined for the reference mark or datum, and is stored automatically. The **Finding field angle** progress bar appears. Then another message appears (see page 7 – 1111).

- ▶ Press the END soft key.

The control carries out a reset. If the message **8630 Field orient. successful** appeared, then the field angle was assigned and is available.

7.9.4 Reading or setting the field angle via the PLC

For axes without encoders with Z1 tracks, which cannot be moved when switched off (e.g. due to Hirth coupling), or if it has been ensured that they cannot move when switched off, Module 9149 can be used to read out the field angle after positioning. This axis-specific field angle must be stored via the PLC in non-volatile memory.

It is also important that the saved field angle is cleared (set to 0) before each positioning, so that the axis (motor) is not started with an incorrect field angle after a power failure. This way an incorrect writing of the field angle is prevented if a power failure occurs.

The nonvolatile field angle must be set via Module 9149 before the drives of the affected axis are switched on. This means that it is not necessary to determine the field angle again.



Danger

Please note the following items when setting/reading the field angle via the PLC:

- An incorrectly set field angle can lead to undesirable reactions of the motor, including uncontrollability. It might even move in the wrong direction!
- If the axis position is moved again after the field angle has been read, then the determined field angle may no longer be used.
- The commutation angle may be set only after you have ensured that the stored commutation angle corresponds to the current position (e.g. due to Hirth coupling).
- The module is suitable only for synchronous, torque, or linear motors in conjunction with nonaligned encoders without EnDat interface.
- The module responds with a value only if the reference mark has been traversed.
- The commutation angle for an axis can be set only once after the control is switched on and before the drives are first switched on.

Module 9149 Set/Read commutation angle

Call:

PS B/W/D/K <Axis>

PS B/W/D/K <Commutation angle in 0.001°>
1 to 720000

PS B/W/D/K <Mode>

0: Read commutation angle

1: Set commutation angle

CM 9149

PL B/W/D <Error code>

0: Commutation angle set/read

1: Module was not called in a spawn or submit job

2: Invalid mode

3: Invalid axis number

4: Invalid commutation angle

Error code from controller when mode 0 is active (read commutation angle):

100: Unknown reference position

Error code from controller when mode 1 is active (set commutation angle):

200: Invalid motor type (no synchronous or linear motor)

201: Invalid encoder type (not "non-aligned")

202: Invalid commutation angle

203: Commutation angle already set

PL B/W/D <Commutation angle>

Error code:

Marker	Value	Meaning
M4203	0	Commutation angle set/read
	1	Error code in W1022
W1022	1	Invalid mode
	2	Invalid axis number
	20	Module was not called in a spawn or submit job
	45	Error code from controller

7.9.5 Saving the determined field angle

The determined field angle is automatically entered in MP2556.x.

For purposes of reliability and redundancy, either the serial number of the encoder (only for EnDat interface) or a unique control ID is entered as identification in MP2257.x.

If the current identification does not match the entry in MP2257.x, an error message appears:

- When using an encoder with EnDat interface, the error message **8830 EnDat: no field angle <axis>** appears. In any case the field angle must be determined anew, since the encoder does not match the field angle from MP2256.x
- When using an incremental encoder, the error message **MP2257.<index> incorrect (ID=\$<identification>)** appears. The field angle from MP2256.x and the new identification (**ID=\$<identification>**) for MP2257.x can only be assumed after determining that the same drive is meant (e.g. after changing controls).



Danger

In all other cases the field angle must be determined anew, since otherwise uncontrolled drive motions could occur!



Note

You can force a new field angle determination by entering MP2256.x = 0 (for example, after exchanging a motor or encoder).

MP2256.x **Determined field angle**

Input: 0: Field angle does not need to be determined, or has not been determined

MP2257.x **Control or encoder identification for the field angle from MP2256.x**

Input: 0: Field angle does not need to be determined, or has not been determined

7.9.6 Definition of the field angle

The following applies to the determined field angle, which is entered in MP2556.x:

The field angle is determined for 360° electrical rotation of the motor, not for mechanical rotation.

The motor is moved with external power in the positive direction (when viewing the shaft, the shaft rotates clockwise). The voltages U_{1-Y} (phase 1 to star point) and U_{2-Y} (phase 2 to star point) are measured. The positive peak value of U_{1-Y} corresponds to a field angle of 90°.

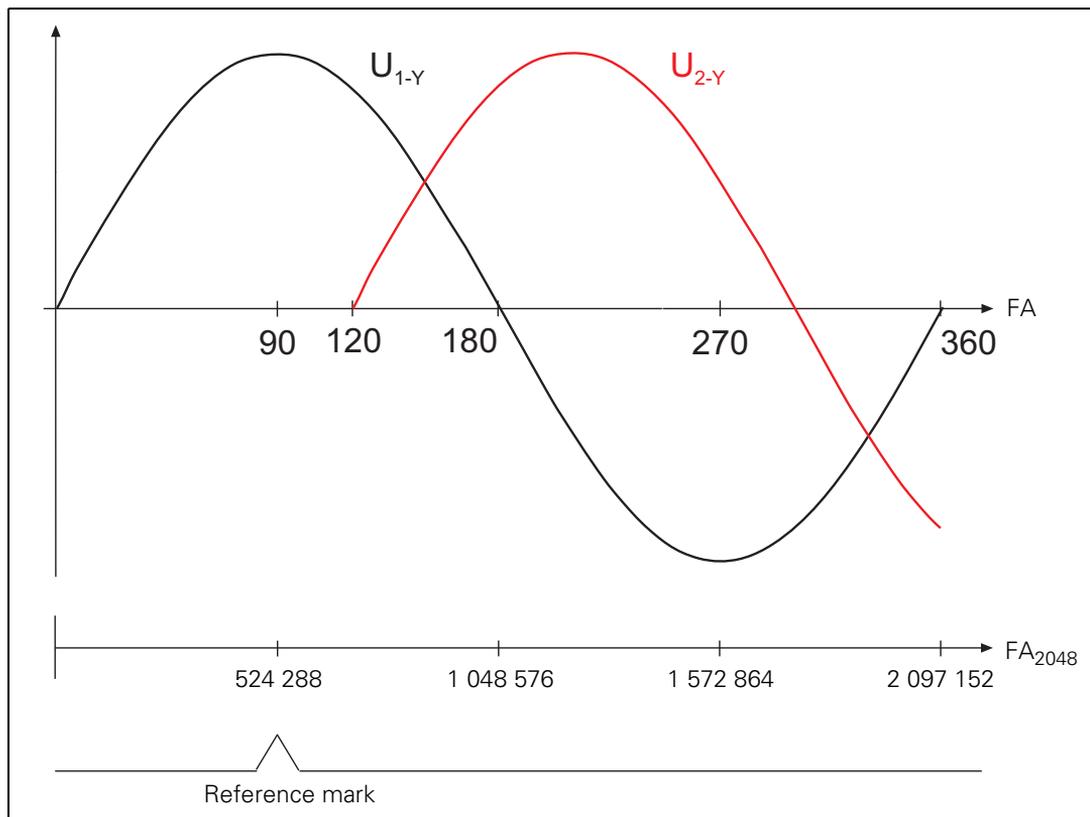
The field angle at the reference mark is saved in increments in MP2256.x. An increment is formed from $\langle \text{line count} \rangle \cdot \langle \text{interpolation factor} \rangle$.

Therefore:

$$\text{MP2256.x} = (\langle \text{field angle at reference mark} \rangle \cdot \langle \text{line count} \rangle \cdot 1024) / 360^\circ$$

In the example below, the reference mark is at the field angle 90°, i.e. $\text{MP2256.x} = (90^\circ \cdot 2048 \cdot 1024) / 360^\circ = 524288$.

On a "standard" synchronous motor (with aligned speed encoder), the reference mark is at the field angle 0°. If the field angle were to be determined for this motor, the result would be $\text{MP2256.x} = \text{approx. } 2097152$.



- U_{1-Y} : Motor voltage between phase 1 and star point
- U_{2-Y} : Motor voltage between phase 2 and star point
- FA: Field angle in degrees
- FA_{2048} : Field angle in increments for an encoder with 2048 lines and 1024-fold interpolation ($2048 \cdot 1024 = 2\,097\,152$)

7.10 Adjustment of linear and torque motors

7.10.1 General information

The internal oscilloscope of the iTNC can operate at a maximum resolution of only as fine as 600 μs , but the feedback control can operate at resolutions as fine as 100 μs .

Frequencies higher than $1/(600 \mu\text{s} \cdot 2) = 833 \text{ Hz}$ result in undersampling. This can result in misinterpretation of the oscilloscope image. High frequencies are mirrored downward. For example, a 1000 Hz oscillation appears as a 833 Hz - 167 Hz = 666 Hz oscillation.

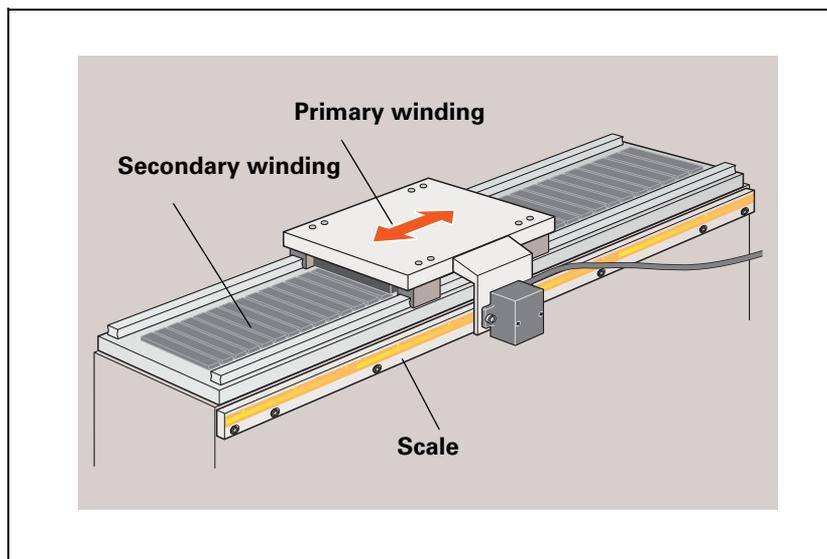


Note

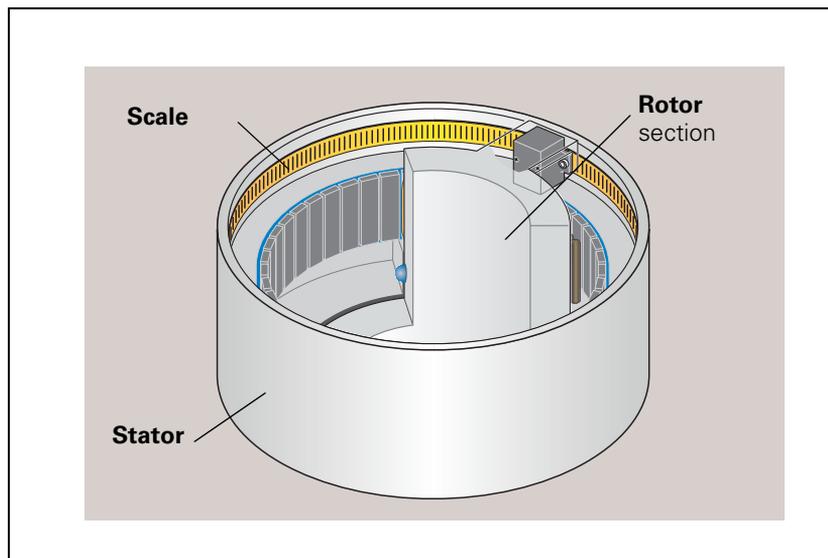
In order to avoid misinterpreting the oscilloscope image, **TNCopt** should be used when searching for oscillation frequencies or optimizing the controller.

For the CC 61xx, the internal oscilloscope and TNCopt display effective values.

Linear motor setup



Torque motor setup

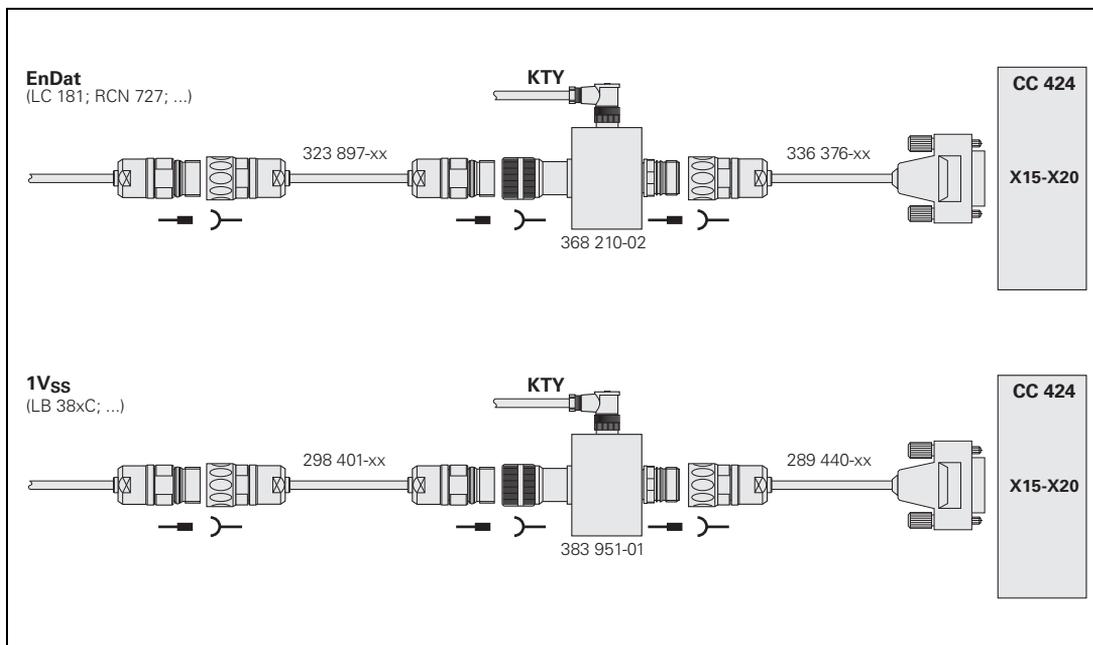


Position and speed encoders

For linear and torque motors, position encoders are used as speed encoders. Therefore, they must be connected to the speed encoder inputs (X15 to X20).

In order to adapt the pin layouts for absolute encoders with EnDat interface (e.g. LC, RCN), you must use the line drop compensator with ID 336697-03, and for incremental encoders with 1 V_{PP} signals (e.g. LB, ROD) you must use the line drop compensator with the ID 383951-01.

The temperature sensor (KTY 84) can also be connected to both line drop compensators.



Temperature sensor

Linear motors usually have a KTY and several PTC thermistors or thermostats for temperature measurement.



Attention

The PTC thermistors or thermostats must be **galvanically isolated** and evaluated by the PLC.

The KTY requires **double insulation** to the motor windings, which must be provided by the motor manufacturer. Otherwise, do **not** connect the KTY to the control!

The KTY is monitored by the control (NC). The temperature signal is conducted to the control together with the encoder signals (X15 to X20). If the KTY is not to be evaluated, this function must be deactivated over MP2220.x bit 4 = 1 (see "Monitoring functions" on page 1087).

For linear and torque motors, the conductor for the temperature signal of the KTY is frequently in the motor power cable, which can cause interference. Since the conductor for the temperature signal is then led into the conductor of the speed encoder, the interference causes noise in the encoder signals. HEIDENHAIN therefore recommends conducting the temperature signals over the line drop compensator, so that the interference signals are filtered.



Note

HEIDENHAIN recommends the additional temperature monitoring of the PTC thermistors or thermostats via the PLC, since these are distributed over the entire length (linear motors) or circumference (torque motors) (as opposed to the KTY, for which there are only spot measurements).

For example, PTC thermistors can be connected to a PLC input via the securely grounded 3RN1013-1BW10 thermistor motor-protection device from SIEMENS.

7.10.2 Safety precautions for linear and torque motors



Note

Comply strictly with the warnings and safety precautions printed in this chapter. They help to avoid accidents and to prevent damage to material through improper handling!



Danger

Linear and torque motors are equipped with strong magnets and exercise strong magnetic forces of attraction!

This can endanger health directly (e.g. for people with pacemakers), or indirectly (e.g. through fast motor movements and high thrust force).



Danger

- **Please comply with all safety regulations of the motor manufacturer!**
- **Never put your hands in the traverse range of a machine that is switched on.**
- Always switch off the machine before working within the traverse range (machine must be free of potential).
- Ensure free traverse for the axis.
- Before switch-on, check the commutation.
- Monitor the end positions.
- Keep the motor area free of chips.
- Watch out for unusual noises.
- Ensure proper function of the motor coolant system.
- Check at regular intervals the primary and secondary surfaces on the side toward the air gap.
- Check for mechanical stability. The integrated buffers must be able to absorb the energy at V_{max} in case of a fault.



Attention

Please comply with the following instructions during all servicing work:

- Mounting, servicing for initial operation, and maintenance are to be performed only by trained personnel.
- Wear work gloves during installation and maintenance work.
- Persons with pacemakers must not service the machine.
- Keep clocks and magnetized data media (e.g. credit cards, floppy disks, etc.) at a distance.
- Do not allow heavy metallic objects to contact the secondary winding of the motor.
- Never allow magnetic surfaces to contact metal.
- Never place the primary winding directly onto the secondary winding.
- Keep a good grip on steel tools and bring them to the secondary winding only slowly and from the side.

7.11 Commissioning linear and torque motors

In this example a linear motor (Siemens 1FN3900-4WB0) and a torque motor (ETEL TMA0360-070-3UC) are adjusted.

If the motors are not yet found in the motor table, enter them in the table using the data sheet values and the conversion rules (see "Determining entries for motor tables" on page 1140), or contact HEIDENHAIN.

On a CC 61xx, the input value 4 can be entered in MP 2206.x for linear motors with one reference mark. In this case the commutation angle must not be determined by using the current controller window. It is determined automatically when the drive is switched on (MP2254.x = 0). When the reference mark is traversed for the first time, it is saved in MP 2256.x.

7.11.1 Machine parameters for linear motors

The following machine parameters can be defined for the 1FN3900-4WB0 linear motor (rotational speed and position measurement through LC 181):

- MP331.x = 0.016 [mm] (when using an LC 181)
- MP332.x = 1
- MP1350.x = 5 (position encoder with EnDat interface)
- MP1054.x = 46 [mm/rev]
- MP2100.x = HEIDENHAIN-UM114
- MP2200.x = 1FN3900-4WB0
- MP2202.x = * (distance traveled per electrical period as in the motor table)
- MP2204.x = * (counting direction as in the motor table; if the rotating field does not match the counting direction, enter a minus sign)
- MP2206.x = * (encoder as in the motor table)
- MP2220.x bit 4 = 1 (no monitoring for excessive temperature, since the temperature sensor only has a single insulation layer)

If speed and position are measured by an LB 3xx C, the following machine parameters change:

- MP331.x = 0.004 [mm] (when using an LB 3xx C)
- MP332.x = 1
- MP334.x = 2000
- MP1350.x = 4 (position encoder with distance-coded reference marks)
- MP2204.x = * (counting direction as in the motor table; if the rotating field does not match the counting direction, enter a minus sign)
- MP2206.x = 8 (incremental linear encoder with distance-coded reference marks (not aligned))

7.11.2 Machine parameters for torque motors

The following machine parameters can be defined for the ETEL TMA0360-070-3UC torque motor (rotational speed and position measurement through RCN 723):

- MP331.x = 360 [°]
- MP332.x = 32768 (when using an RCN 723)
- MP1350.x = 5 (position encoder with EnDat interface)
- MP1054.x = 360
- MP2100.x = HEIDENHAIN-UM114
- MP2200.x = ETEL TMA0360-070-3UC
- MP2202.x = * (line count as in the motor table)
- MP2204.x = * (counting direction as in the motor table; if the rotating field does not match the counting direction, enter a minus sign)
- MP2206.x = * (encoder as in the motor table)



7.11.3 Adjustment of the current controller

- ▶ Enter as many machine parameters as possible.
- ▶ Assign empirical values to machine parameters that must still be determined (see "Preparation" on page 1020).



Danger

During current controller adjustment of linear and torque motors, the rotor position of the motor is not yet known.

For this reason, if the motor brakes are not active, the motor might move slightly when the current pulses switch on. In other words, it might oscillate about a preferred position. It is also possible to position "manually" to the preferred position. Do not do this, however, during a measurement.

Use the integrated oscilloscope or TNCopt to adjust the current controller.

During adjustment of the current controller the speed controller and position controller are open. During output of the current pulses the drive controller becomes active.

The PLC initial servicing program, whose name and path are entered in OEM.SYS after the **PLCPWM** = entry, must ensure that the inverters are ready after "Switch on external dc voltage," but that the motor brakes are not opened.

Before and after the output of the current pulses (**START STEP** soft key is not pressed) the inverter must be in the following operating mode:

- Green "READY" LED on
- Red "STO A" LED off
- Red "STO B" LED on (drive controller not ready)

During output of the current pulses:

- Green "READY" LED on
- Red "STO A" LED off
- Red "STO B" LED off (drive controller ready)



During current adjustment, proceed as follows:

- ▶ Switch the control on.
- ▶ Do not acknowledge the **Power Interrupted** message. In the **Programming and Editing** mode of operation, use the MOD key to enter the code number **688379**. The oscilloscope is started.
- ▶ Press the **I CONTROL** soft key.
- ▶ In the **Manual Operation** mode of operation, acknowledge the **Power Interrupted** message and switch the control voltage on.
- ▶ Use the **SELECT AXIS** soft key in the oscilloscope to select the axis to be adjusted.
- ▶ With the **P/I FACTOR** soft key, select the I factor and set $MP2430.x = 0$.
- ▶ With the **P/I FACTOR** soft key, select the P factor.
- ▶ As a starting value for the P factor first enter the value 0, select the step increment **SLOW**, and press the \uparrow soft key about 5 to 10 times.
- ▶ Press the **START STEP** soft key.
This sends multiple step functions to the current controller and measures the step responses. The height and length of the steps are automatically calculated by the control.

Applying a voltage causes a brief humming noise.



Danger

If the brakes are not active, the motor can move somewhat (in the direction of a preferred position)!
The error message **Standstill monitoring <axis>** might be displayed. It can be deleted.

- ▶ With the \uparrow soft key, increase/decrease the P factor (MP2420.x) step by step just enough so that no overshoot is visible.
- ▶ Save this value with the **STORE MP2420.x** soft key.
- ▶ With the **P/I FACTOR** soft key, select the I factor.
- ▶ As a starting value for the I factor first enter the value 0, select the step increment **SLOW**, and press the \uparrow soft key about 5 to 10 times.
- ▶ With the \uparrow soft key, increase/decrease the I factor (MP2430.x) step by step just enough so that no overshoot is visible.
- ▶ Save this value with the **STORE MP2430.x** soft key.
- ▶ Press the END key to exit the **Commission Current Controller** mode.

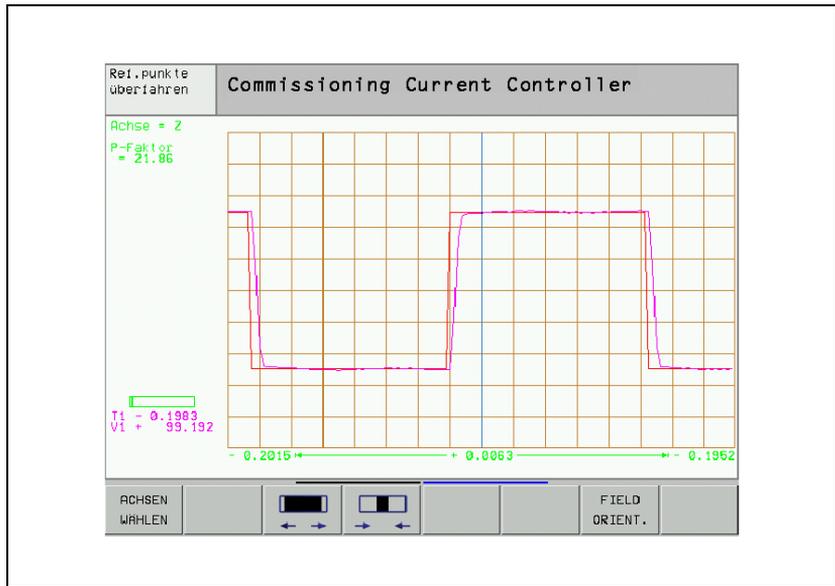
The control carries out a reset.



Note

For **linear and torque motors**, adjust the P and I component of the current controller so that no overshoots are visible in the step response. This reduces motor noise.

Properly adjusted step response:



7.11.4 Adjustment of the speed controller

The speed controller should be adjusted to largely suppress oscillation frequencies but to permit short rise times.

The following table shows approximate input values for commissioning the speed controller so that axes can be moved:

Machine parameter	Value
MP1510.x	1 to 5; k_v factor for velocity feedforward control
MP2500.x	1 to 50 for linear motors (starting value 1); 50 to 5000 for torque motors (starting value 50)
MP2510.x	10 to 2000 for linear motors (starting value 10); 1000 to 800 000 for torque motors (starting value 1000)
Increase positioning window and position monitoring	see "Commissioning" on page 1004
MP2562.x to MP2566.x	0 (deactivate all filters)
MP2602.x, MP2604.x	0 (without IPC)

The current controller must have been adjusted and the field angle ascertained (see "Adjustment of the current controller" on page 1130 and see "Field orientation" on page 1108).

Preparation on the control

- ▶ Position the axis or spindle to be optimized at a location where it can be commissioned safely.
- ▶ Ensure that the loaded PLC program fulfills the following conditions:
 - Enable the drive controller
 - NC stop inactive
 - Axis direction buttons active

Setting the filters in the speed controller

- ▶ To determine the oscillation frequencies of the speed controller, activate the **Optimization/speed controller step response** function in TNCopt.

The oscillation frequencies of the speed controller for linear and torque motors must be determined at slow feed rate in positive and negative traverse direction, and at the midpoint and end of the traverse range.

Later you enter the ascertained frequencies, bandwidths and damping in the corresponding machine parameters (MP2542.x to MP2566.x). Normally no more than three filters are required.

Experience recommends:

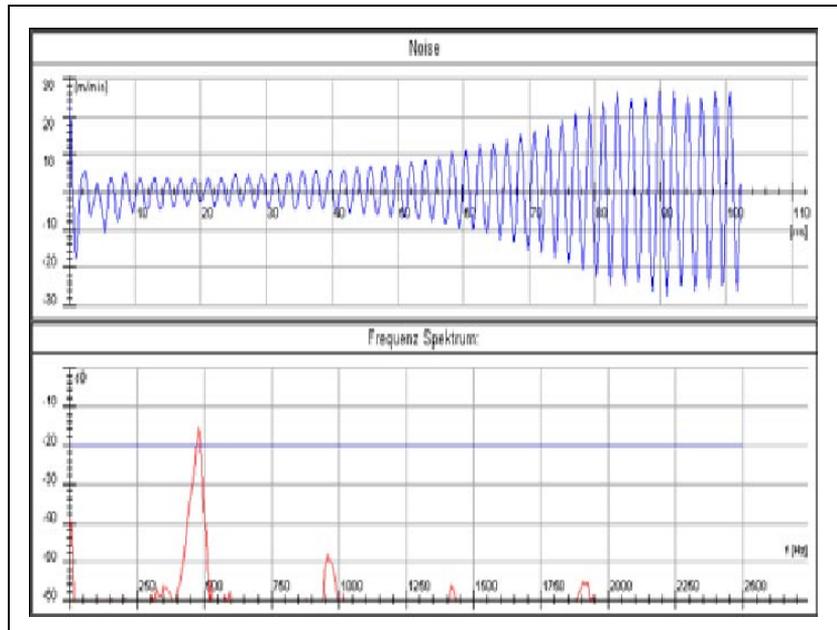
- At very small frequencies (< 200 Hz) use no filters
- Use very little damping (3 dB to 6 dB) at low frequencies (200 Hz to 400 Hz)
- The bandwidth should be approximately at center frequency
- Damping values are normally between 3 dB and 18 dB. Damping above 18 dB usually brings no further improvement.
- The use of a PT2 filter on linear motors does not bring positive results. For torque motors it is sometimes necessary to activate a PT2 filter (not below approx. 400 Hz).



Note

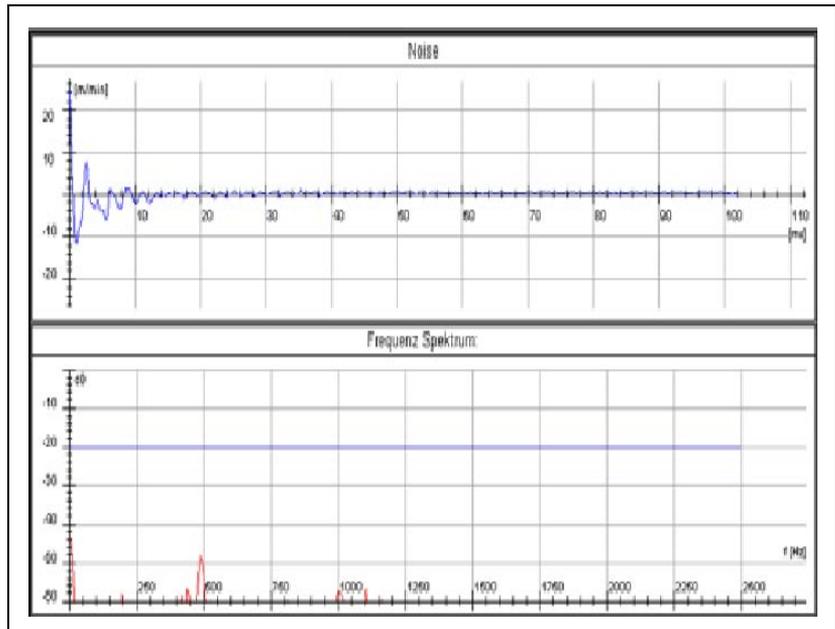
Only use the filters in the speed controller if they are necessary, i.e. if the P factor can be increased noticeably.

- ▶ For P and I factors, set low values at first:
(for linear motor: P factor = 1, I factor = 10;
for torque motor: P factor = 50, I factor = 100).
- ▶ Increase the P factor up to the oscillation limit in 10 % steps
(see the TNCopt User's Manual).



- ▶ Find the characteristic values of the first filter (F1):
Enter the oscillation frequency (e.g. at 479 Hz),
Enter the bandwidth (e.g. 479 Hz, center frequency),
Enter the damping (3 dB).
- ▶ Examine the step response.
- ▶ Check whether a constant P factor decreases the excessive value.
- ▶ Increase the damping in 3-dB steps (damping greater than approx. 18 dB normally has no benefit).

- ▶ Examine the step response. The oscillations should be greatly decreased.

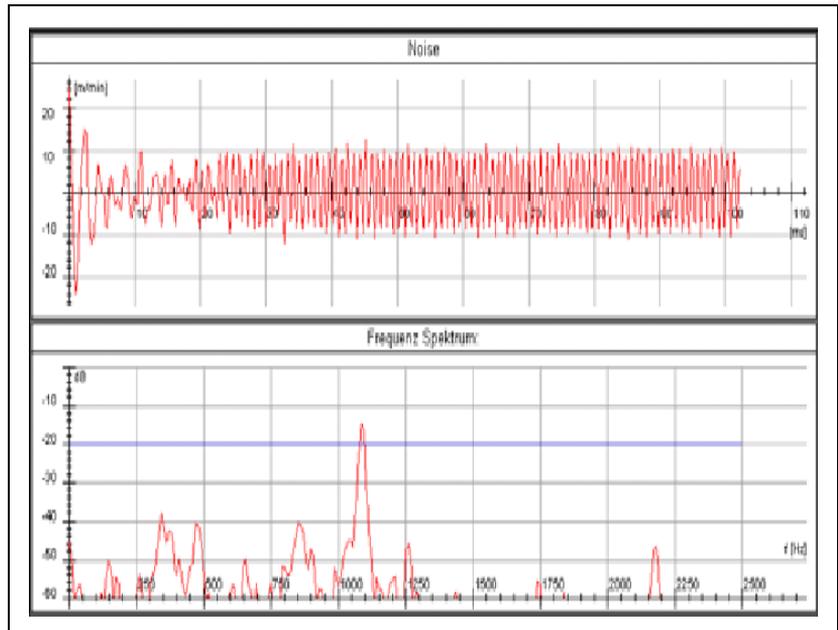


- ▶ Increase the P factor further.

This may cause new oscillations.

When selecting filters, watch whether your selection causes or aggravates negative characteristics. If it does, do not activate any further filters.

Use filters only if the P factor can be significantly increased!



After finding the oscillation frequencies, several filters can be activated.

- On the control, enter the determined filter type (MP2562.x to MP2566.x), the determined center frequency (MP2552.x to MP2556.x), the determined bandwidth (MP2572.x to MP2576.x) and the determined damping (MP2542.x to MP2546.x).

Adjusting synchronous and asynchronous motors

Synchronous and asynchronous motors are adjusted as described in the TNCopt User's Manual.

If a sufficiently short enough rise time is not achieved (approx. 3 ms), use a filter (see "Setting the filters in the speed controller" on page 1134). Using a filter also results in an increased P factor. The I factor must then be redetermined.

In most cases automatic adjustment of synchronous and asynchronous motors determine rise times that are too short (< 2 ms). Rise times that are too short result in irregular control responses. In this case both the P and I factors must be reduced (until a rise time of approx. 2 ms is achieved).

Adjusting linear and torque motors

- ▶ First determine the oscillation frequencies and adjust the filters (see "Setting the filters in the speed controller" on page 1134).
- ▶ To adjust linear and torque motors, activate the **Optimization/Linear motor adjustment** function in TNCopt.

This makes it possible to automatically find the controller parameters in the sequence P factor, k_V factor, I factor (also see the TNCopt User's Manual). An NC program must be started in which the motor is moved back and forth.

Because inhomogeneity of magnets, changing the air gap, and other factors can cause linear motors to oscillate more strongly at certain positions, HEIDENHAIN recommends adjusting the motor over a large range of traverse (approx. 300 mm to 500 mm).

During adjustment, TNCopt automatically increases the newly adjusted controller factor after each reversal point until it detects an oscillation limit. The respective current value is displayed in TNCopt. Usually you can already hear the oscillation limit being reached. In this case the oscillation limit is set per command button (see the TNCopt User's Manual). The factors are reduced after detection of the oscillation limit. Then they are considered to be optimized values.

Particularly the optimization of the I factor often results in very low frequency oscillations that the controller does not recognize as oscillations. Here the oscillation limit must be set by command button.

- ▶ Move the axis back and forth.

The P factor, k_V factor or I factor is automatically increased by 10 % of the current value.

- ▶ As soon as you hear an oscillation in the axis (also visible on the iTNC's oscilloscope with **I nom1**), set the oscillation limit via command button (see the TNCopt User's Manual).

The determined P factor, k_V factor or I factor is reduced and assumed. TNCopt automatically switches to finding the next factor.

- ▶ Also repeat the finding of the k_V factor and I factor.

The P, I and k_V factors were found.

- ▶ Enter the determined values at the control in MP2500.x, MP2510.x and MP1510.x.

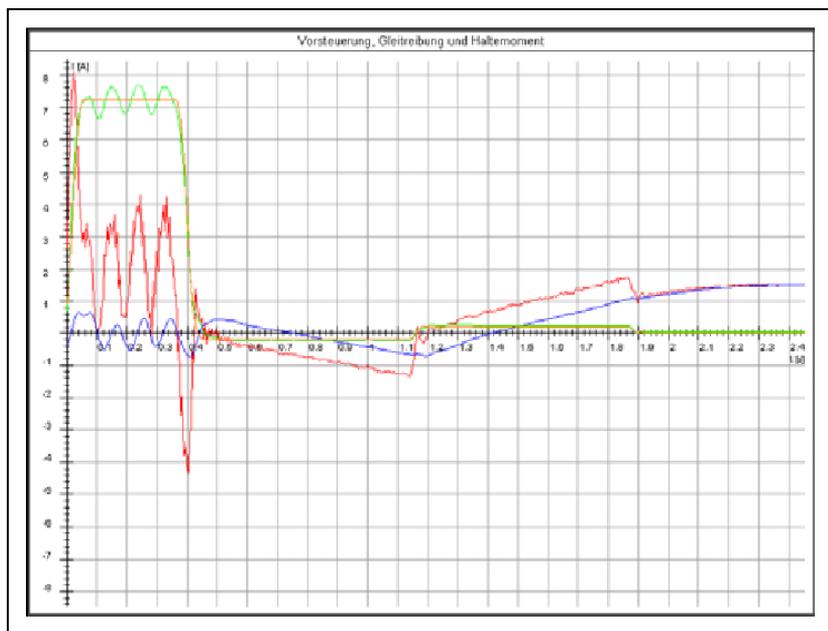
Finding the feedforward values

Automatic adjustment of feedforward values in TNCopt often fails for linear motors. The reason is the often extreme ripple of the nominal current due to the pronounced torque ripple. But even if the automatic adjustment does not work, TNCopt can still be used for manual adjustment.

The proposed curves are sometimes not the optimum for linear motors. It can therefore be better for manual adjustment if you increase the traverse paths and the feed rate (by approximately doubling it, for example) and adjust the acceleration in MP1060 to about 1 m/s^2 to 2 m/s^2 .

If the integral current should approach the nominal current only very slowly, reduce the k_V factor and increase the I factor. However, these changes should be undone after determining the feedforward values.

The image below shows an adjustment of a highly critical linear motor with a distinctly inhomogeneous magnetic field. This can be seen from the nominal current, which fluctuates very strongly depending on the position:



7.12 Determining entries for motor tables



Note

If you use a linear or torque motor that is not listed in the current HEIDENHAIN motor table, please contact HEIDENHAIN. As an alternative, you can also determine the input values yourself.

7.12.1 Determining data for linear motors

The motor data for linear motors are entered in the motor table after some conversions using the values from the motor data sheet of the respective manufacturer (here using the example of a Siemens motor).

Values in the HEIDENHAIN motor table	Values from the motor data sheet
TYPE: LSM	Permanently excited linear synchronous motor
NAME: 1FN3900-4WB0	1FN3900-4WB0
MODE: 0	
Rated current I-N in [A _{eff}] winding I-N: 49.4	Data sheet value I _N I _N = 49.4 A
Rated voltage U-N in [V _{eff}] interlinked U-N: 394	Calculation from data sheet values FN, I _N , RP.120, LP $U-N = \sqrt{3} \cdot \sqrt{(U_e + U_r)^2 + U_x^2}$ $U_e = N-N / (1000 \text{ mm/m}) \cdot FN / (I_N \cdot 3)$ $= (2.66 \text{ m/s}) \cdot 8100 \text{ N} / (49.4 \text{ A} \cdot 3)$ $= 145.4 \text{ V}_{\text{eff L-N}}$ $U_r = RP.120 \cdot I_N$ $= 0.8 \Omega \cdot 49.4 \text{ A}$ $= 39.5 \text{ V}_{\text{eff L-N}}$ $U_x = 2 \cdot \pi \cdot F-N \cdot LP \cdot I_N$ $= 2 \cdot \pi \cdot 57 \text{ Hz} \cdot 0.0075 \text{ H} \cdot 49.4 \text{ A}$ $= 132.69 \text{ V}_{\text{eff L-N}}$ $U-N = 394 \text{ V}_{\text{eff (L-L)}}$
Rated speed N-N in [mm/s] N-N: 2666	Calculation from data sheet value v _{MAX, FN} $N-N = (1000 \text{ mm/m}) \cdot v_{\text{MAX, FN}} / (60 \text{ s/min})$ $= (1000 \text{ mm/m}) \cdot (160 \text{ m/min}) / (60 \text{ s/min})$ $= 2666 \text{ mm/s}$ Note: Other meaning than for rotating motors!
Rated frequency F-N in [Hz] F-N: 57	Calculation from V-N, distance per electrical period (see entry in STR column) $F-N = N-N \cdot (1000 \mu\text{m/mm}) / (\text{distance per elec. period } [\mu\text{m}])$ $F-N = (2666 \text{ mm/s}) \cdot (1000 \mu\text{m/mm}) / (46\,000 \mu\text{m})$ $= 57 \text{ Hz}$



Values in the HEIDENHAIN motor table	Values from the motor data sheet
No-load velocity at rated velocity U0 in [V _{eff}] interlinked U0: 252	Calculation from V-N and data sheet value $k_{F,3}$ $U0 = N \cdot N / (1000 \text{ mm/m}) \cdot (k_{F,3} / 3) \cdot \sqrt{3}$ $= 2.66 \text{ m/s} \cdot (164 \text{ N/A} / 3) \cdot \sqrt{3}$ $= 252 \text{ V}_{\text{eff}} \text{ L,L}$
No-load current I0 in [A _{eff}] winding I0: 0	
Primary winding resistance at 20 °C R1 in [mΩ] at 20° C R1: 600	Calculation from data sheet value $R_{P,20}$ $R1 = R_{P,20} \cdot 1000$ $= 0.6 \Omega \cdot 1000$ $= 600 \text{ m}\Omega$
Rotor resistance at 20 °C R2 in [mΩ] at 20° C R2: 0	
Primary winding leakage reactance at F-N Xstr1 in [mΩ] Xstr1: 0	If nothing given, then zero.
Rotor leakage reactance at F-N Xstr2 in [mΩ] Xstr2: 0	
Magnetizing reactance XH for F-N at rated conditions XH in [mΩ] XH: 2685	Calculation from F-N and data sheet value L_p $XH = 2 \cdot \pi \cdot F \cdot N \cdot L_p$ $= 2 \cdot \pi \cdot 57 \text{ Hz} \cdot 7.5 \text{ mH}$ $= 2685 \text{ m}\Omega$
Desaturation velocity N-XH in [min ⁻¹] N-XH: 0	
Rotational speed of beginning field weakening range N-FS [min ⁻¹] N-FS: 0	
Max. velocity (mechanical) N-MAX in [mm/s] N-MAX: 2666	Calculation from data sheet value $v_{\text{MAX, FN}}$ $N \cdot \text{MAX} = 160 \text{ m/min} \cdot (1000 \text{ mm/m}) / (60 \text{ s/min})$ $= 2666 \text{ mm/s}$ <p>Note: Other meaning than for rotating motors!</p>
Saturation factor %-XH in % %XH: 100	
Stalling torque reduction factor %-K in % %-K: 100	
Number of pole pairs (PZ) = 1 per definition PZ: 1	
Temperature coefficient of the primary winding TK in 1/K TK: 0.004	

Values in the HEIDENHAIN motor table	Values from the motor data sheet
Distance per electrical period STR STR: 46 000	Calculation from data sheet value τ_M Distance= $\tau_M \cdot 2 \cdot (1000 \mu\text{m}/\text{mm})$ = $23 \text{ mm} \cdot 2 \cdot (1000 \mu\text{m}/\text{mm})$ = $46\,000 \mu\text{m}$ Note: Other meaning than for rotating motors!
Type of encoder SYS: 3	Incremental encoder (e.g. LB): 8 Absolute encoder with EnDat interface (e.g. LC): 3
Counting direction DIRECT. DIRECT.: +	
Max. temperature of motor at temperature sensor T-MAX in [°C] T-MAX: 120	Data sheet value $T_{P, \text{MAX}}$ $T_{P, \text{MAX}} = 120 \text{ °C}$
Maximum motor current I-MAX in [A_{eff}] winding I-MAX: 138.9	Data sheet value I_{MAX} $I_{\text{MAX}} = 138.9 A_{\text{eff}}$
Rated power P-N in [W] P-N: 21600	Calculation from data sheet values $v_{\text{MAX, FN}}$, F_N $P\text{-N} = v_{\text{MAX, FN}} \cdot (1 \text{ min}/60 \text{ s}) \cdot F_N$ $P\text{-N} = 160 \text{ m}/\text{min} \cdot (1 \text{ min}/60 \text{ s}) \cdot 8100 \text{ N}$ $P\text{-N} = 21600 \text{ W}$
Mass of primary winding J in [kg] J: 56.2	Data sheet value m_P $m_P = 56.2 \text{ kg}$ Note: Other meaning than for rotating motors!
Inductance of the series reactor L in [mH] L: 0 μH As long as $(X1\text{str}+X_h) / (2 \cdot \pi \cdot (N\text{-N} / 60) \cdot P_Z)$ is greater than 700 μH , no series reactor is required.	
Thermal time constant for direct current T-DC in [s] T-DC: 0	
Lower thermal cutoff frequency F-DC in [Hz] F-DC: 0	
Thermal time constant for alternating current T-AC in [s] T-AC: 0	
Upper thermal cutoff frequency F-AC in [Hz] F-AC: 0	



7.12.2 Determining data for torque motors

The motor data for torque motors are entered in the motor table after some conversions using the values from the motor data sheet of the respective manufacturer (here using the example of an ETEL motor).

Values in the HEIDENHAIN motor table	Values from the motor data sheet
TYPE: SM	Permanently excited synchronous motor
NAME: Etel-TMA0360-070-3UC	TMA0360-070-3UC
MODE: 0	
Rated current I-N in [A _{eff}] winding I-N: 32.6	Data sheet value I _N I _N = 32.6 A
Rated voltage U-N in [V _{eff}] interlinked U-N: 252	Calculation from data sheet values F _N , I _N , R _{P.120} , L _P $U-N = \sqrt{3} \cdot \sqrt{(U_e + U_r)^2 + U_x^2}$ $U_e = 2 \cdot \pi \cdot (N-N / 60) \cdot T_{cw105} / l_{cw105} / 3$ $U_e = 2 \cdot \pi \cdot (180 / 60) \cdot 485 / 32.6 / 3$ $U_e = 93.48 V_{eff L,N}$ $U_r = (R_{105L;L} / 2) \cdot l_{cw105}$ $U_r = (1.92 / 2) \cdot 32.6$ $U_r = 31.30 V_{eff L,N}$ $U_x = 2 \cdot \pi \cdot (n / 60) \cdot (2p / 2) \cdot (L_{1L,L} / 2) \cdot l_{cw105}$ $U_x = 2 \cdot \pi \cdot (180 / 60) \cdot (66 / 2) \cdot (0.00738 / 2) \cdot 32.6$ $U_x = 74.83 V_{eff L,N}$ $U-N = 252.0 V_{eff L,L}$
Rated speed N-N in [min ⁻¹] N-N: 180	Data sheet value n N-N = 180 min ⁻¹
Rated frequency F-N in [Hz] F-N: 99	Calculation from data sheet value n and 2p $F-N = (n / 60) \cdot (2p / 2)$ $F-N = (180 / 60) \cdot (66 / 2)$ $= 99 \text{ Hz}$
No-load voltage at rated speed U ₀ in [V _{eff}] interlinked U ₀ : 171	Calculation from data sheet value n and K _t $U_0 = 2 \cdot \pi \cdot (n / 60) \cdot (K_t / 3) \cdot \sqrt{3}$ $U_0 = 2 \cdot \pi \cdot (180 / 60) \cdot (15.7 / 3) \cdot \sqrt{3}$ $U_0 = 170.9 V_{eff L,L}$
No-load current I ₀ in [A _{eff}] winding I ₀ : 0	
Stator resistance at 20 °C R ₁ in [mΩ] at 20 °C R ₁ : 670	Calculation from data sheet value R _{20_{L-L}} $R_1 = R_{20_{L-L}} / 2$ $= 1.34 \Omega / 2$ $= 670 \text{ m}\Omega$
Rotor resistance at 20 °C R ₂ in [mΩ] at 20 °C R ₂ : 0	

Values in the HEIDENHAIN motor table	Values from the motor data sheet
Stator leakage reactance at F-N Xstr1 in [mΩ] Xstr1: 0	If nothing given, then zero.
Rotor leakage reactance at F-N Xstr2 in [mΩ] Xstr2: 0	
Magnetizing reactance XH for F-N at rated conditions XH in [mΩ] XH: 2295	Calculation from data sheet value L_{1L-L} , n and 2p $XH = 2 \cdot \pi \cdot (n / 60) \cdot (2p / 2) \cdot (L_{1L-L} / 2)$ $= 2 \cdot \pi \cdot (180 / 60) \cdot (66 / 2) \cdot (0.00738 / 2)$ $= 2295 \text{ m}\Omega$
Desaturation speed N-XH in [min ⁻¹] N-XH: 0	
Rotational speed of beginning field weakening range N-FS [min ⁻¹] N-FS: 0	
Maximum speed (mechanical) N-MAX in [min ⁻¹] N-MAX: 180	Data sheet value n N-MAX = 180 min ⁻¹
Saturation factor %-XH in % %-XH: 100	
Stalling torque reduction factor %-K in % %-K: 100	
No. of pole pairs (half pole no. of motor) PZ PZ: 33	From data sheet value 2p PZ = 2p/2 PZ = 66 / 2 = 33
Temperature coefficient of the stator winding TK in 1/K TK: 0.004	
Line count of the speed encoder: STR STR: 32768	
Type of encoder SYS: 5	Incremental encoder with Z1 track: 1 Aligned absolute encoder with EnDat interface: 2 Unaligned absolute encoder with EnDat interface: 5 Unaligned incremental encoder with distance-coded reference marks: 7
Counting direction DIRECT. DIRECT.: +	
Max. temperature of motor at temperature sensor T-MAX in [°C] T-MAX: 120	
Maximum motor current I-MAX in [A _{eff}] winding I-MAX: 53	Data sheet value I _p I-MAX = 53.0 A _{eff}



Values in the HEIDENHAIN motor table	Values from the motor data sheet
Rated power P-N in [W] P-N: 9142	Calculation from n and Tcw105 $P-N = 2 \cdot \pi \cdot (n / 60) \cdot T_{cw105}$ $P-N = 2 \cdot \pi \cdot (180 / 60) \cdot 485 \text{ Nm}$ P-N = 9142 W
Motor mass moment of inertia J in [kgm ²] J: 0.157	Data sheet value J J = 0.157 kgm ²
Inductance of the series reactor L in [mH] L: 0 μH As long as $(X_{1str} + X_h) / (2 \cdot \pi \cdot (N-N / 60) \cdot P_Z)$ is greater than 700 μH, no series reactor is required.	
Thermal time constant for direct current T-DC in [s] T-DC: 60	
Lower thermal cutoff frequency F-DC in [Hz] F-DC: 0.25	
Thermal time constant for alternating current T-AC in [s] T-AC: 400	
Upper thermal cutoff frequency F-AC in [Hz] F-AC: 0.25	

With MP2209.x you can overwrite the value for "Mass moment of inertia J" in the motor table. This makes it possible to react to additional, rigidly coupled inertias.

MP2209.x **Mass moment of inertia of a drive motor**
 Input: * = Entry from the motor table active
 Value of the mass moment of inertia in [kgm²]



8 Machine interfacing

8.1 Display and operation

You can modify the display and operating modes of the iTNC by editing the machine parameters.

The display screen is divided into separate windows. The user can select the operating functions by soft key. Refer to the User's Manual.

8.1.1 Position and status display

The status display shows the status of the control.

With a soft key you can activate an additional status display in the graphic window instead of the graphic. The status display is shown in forms that are divided into tabs. You switch between the tabs via soft keys. The STATUS PGM, STATUS POSITIONS, STATUS TOOL and STATUS COORDINATE TRANSFORMATIONS tabs can be selected directly, and the other status displays are selected with soft keys.

This information includes:

- Axis positions
- Tools
- Feed rate
- M functions
- Active spindle (S1 or S2)
- Current time
- The most important status information is collected on an overview page.
- In the PGM status a moving-bar graphic shows the current program run time as a percentage. In addition, the actual and programmed feed rate are shown as numerical values.
- The active machining cycle and the active parameters programmed in Cycle 32 Tolerance are shown in the CYC status.
- The active basic rotation is also shown in the TRANS status.
- Only the status information important for test run is shown in the Test Run operating mode.
- If the **Global Program Settings** software option is available, two additional tabs containing information about the activated functions are shown.
- If the **Adaptive Feed Rate Control** software option is available, the additional AFC tab containing information about the feed-rate control is shown.

Position display step

To define the position display step for axis and spindle positions:

- ▶ Enter the desired display step for the axes in MP7290.x and for the spindle in MP7289.

The position loop resolution is not influenced by this parameter.

MP7290.x Position display step for the axes

Input: 0: 0.1 mm or 0.1°
1: 0.05 mm or 0.05°
2: 0.01 mm or 0.01°
3: 0.005 mm or 0.005°
4: 0.001 mm or 0.001°
5: 0.0005 mm or 0.0005°
6: 0.0001 mm or 0.0001°

MP7289 Position display step for the spindle

Input: 0: 0.1°
1: 0.05°
2: 0.01°
3: 0.005°
4: 0.001°
5: 0.0005°
6: 0.0001°

Position display of the tool axis

The tool length can be offset in the position display of the tool axis. If it is, the displayed position value then refers to the tool point:

- ▶ With MP7285, select whether the tool length should be offset.

The behavior of an incremental block after a **TOOL CALL** can be specified:

- ▶ With MP7682 bit 0, select whether the tool length should be offset.

MP7285 Tool length offset in the tool-axis position display

Input: 0: Tool length is not offset
1: Tool length is offset

MP7682 Machine parameter with multiple function

Input: Bit 0 – Incremental block after TOOL CALL
0: With length compensation
1: Without length compensation

Position display for rotary axes and PLC auxiliary axes

For these axes you can set the modulo value for the counting mode (i.e. the value after which the axis display returns to zero). The software limit switches become inactive when the modulo counting mode is activated.

However, for the modulo counting mode you can activate software limit switches for a traverse-range limit (e.g. 10 to 180 in MP91x.x or 92x.x) via MP812. This traverse-range limit cannot be restricted additionally in the manual operating mode.

When using encoders with distance-coded reference marks (MP1350), you can define a traverse range of more than 360° for rotary axes with limit switches (e.g. ± 190 degrees). When the machine is switched off, the position is stored by the control. This enables the control to ascertain whether the axis is, for example, at a position of -1° or +359° after the machine is switched back on.



Note

Please note that the traverse-range limits set earlier in the manual operating mode also become active in the modulo counting mode when MP812.x is set. However, these can no longer be edited or viewed in the manual operating mode.

- ▶ Select the display mode with MP810.x.
- ▶ Activate or deactivate the software limit switches with MP812.

Rotary axes with modulo display can be positioned either without crossing zero or always along the shortest path:

- ▶ Select the type of positioning with MP7682.
 - For bit 2=0: Programming with M126
 - For bit 2=1: You need not program with M126.

Furthermore, based on the settings in MP810.x, the control detects whether the PLC axis is a rotary axis:

- MP810.x ≠ 0: PLC axis is a rotary axis

MP810.x Display mode for rotary axes and PLC auxiliary axes

Input: 0.0000 to 99 999.9999 [°]
0: Display +/-99 999.9999
≠ 0: Modulo value for display

MP812 Activate software limit switches for rotary axes with modulo display, M94 and encoders with EnDat interface

Input: %xxxxxxxxxxxxxxxxxxxxx
Bits 0 to 17 represent axes 1 to 18
0: Software limit switch not active
1: Software limit switch active

MP7682 Machine parameter with multiple function

Input: Bit 2 – Traverse path of rotary axes with modulo display
0: Positioning without passing over zero
1: Positioning on the shortest path

Reading of axis coordinates

- ▶ Read the axis coordinates with Module 9040 or Module 9041.

The values are saved in double words beginning at the given address. Enough double words must be reserved. The number of required addresses is according to the **AXISNUMBER** = entry in OEM.SYS. If this entry does not exist, the number is according to the enabled control loops.

To define the coordinate value of an axis, the reference point of the axis must first be traversed.

Module 9040 Read axis coordinates (format 0.001 mm)

Module 9040 reads the axis coordinates from the control loop for all NC axes. Different values can be loaded: Actual value, nominal value, servo lag, distance-to-go, the deflection of a triggering touch probe, temperature compensation values, and the distance counter (counts the distance traveled by an axis).

The datum system equals the reference system shifted by a datum compensation. All values, except for the distance counter, are saved in the format 1/1000 mm in double words beginning at the programmed target address. The number of double words that are written to varies depending on the configuration.

Conditions:

SEE MODULE 9041.

Call:

SEE MODULE 9041.



Module 9041 Read axis coordinates (format 0.0001 mm)

Module 9041 reads the axis coordinates from the control loop for all NC axes. Different values can be loaded: Actual value, nominal value, servo lag, distance-to-go, the deflection of a triggering touch probe, temperature compensation values, and the distance counter (counts the distance traveled by an axis).

The datum system equals the reference system shifted by a datum compensation. All values, except for the distance counter, are saved in the format 1/10000 mm in double words beginning at the programmed target address. The number of double words that are written to varies depending on the configuration.

Condition:

- Regardless of whether individual axes have been excluded through MP10, the coordinate values are always loaded for all axes. The values for excluded axes remain undefined.
- The coordinate value of an axis remains undefined until the reference point has been traversed.
- In a tilted working plane, the axis coordinates are referenced to the non-tilted coordinate system.
- The tool length is ignored for the actual values in the datum system.
- The coordinate type "Actual values in the datum system" must only be read when the working plane is NOT tilted.
- Distance counter in [mm]:
Updated once a minute if a change of at least 10 mm has occurred. The maximum value of a counter corresponds to the value from MP4050.x and is cleared with W1058/W1168 (resetting the accumulated distance).

Call:

PS K/B/W/D <Target address Dxxxx>

PS K/B/W/D <Type of coordinate>

0: Actual values

1: Nominal values

2: Actual values in the reference system

3: Following error

4: Distance-to-go

5: Deflection (measuring touch probe)

6: Actual values in the shifted reference system (datum shift)

7: Reference values with backlash compensation offset from MP710.x

8: Temperature compensation from the description tables of the tilting-axis geometry

9: Distance counter in [mm]

10: Reserved

11: Reserved

12: KinematicsComp compensation values

CM 9040 OR

CM 9041

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Invalid coordinate type, target address too large, or given target address is not a double-word address

Free rotation

Free rotation means that the rotary axis rotates as often as required (with a display range of 0 to 360°) without being affected by software limit switches. You can define the free rotation function through words (axis 4 and 5) or with Module 9223.

No uncontrolled axes can be used for this function, but PLC axes (auxiliary axes) are possible along with main axes.

The maximum feed rate is 300 000 °/min or the limiting value in MP1010.x. The feed rate is not shown in the status window. You can change the feed rate with the override potentiometer (W754) continuously, for example by copying W494 (active feed rate override) to W754. MP7620 bit 3 is accounted for.

Use Module 9048 to inquire whether free rotation is active for a specific axis.

The following conditions apply to the free rotation function:

- Collision monitoring (DCM) does not work in combination with the "free rotation" PLC function, since this special function for the defined "free" rotary axis runs asynchronously to the system. For the work envelope this means that all CMOs moved with the "free" rotary axis cannot be monitored by DCM.
- All monitoring functions, such as for position, following error, signal amplitude of the encoder, etc., remain active.
- All settings for the rotary axes via machine parameters, such as jerk, acceleration, etc., remain valid.
- The axis that is in free rotation behaves asynchronously to (independently of) all other axes. For example, an NC program can be run while the "free rotation" axis rotates independently. This axis is not considered when calculating the path.
- The axis for free rotation must be a rotary axis that can rotate over 360° without any restrictions.
- The free rotation function can be used not only for principal axes (NC axes), but also for auxiliary axes (PLC axes).



Free rotation defined by words

- W566 Feed rate in axis 4 for free rotation
- W567 Feed rate in axis 5 for free rotation
- W754 Feed rate override percentage for free rotation
- B518 Defining the free rotation function
- B519 Traverse direction for free rotation
- M4133 Starting and stopping the free rotation function
If you set M4133, the NC takes the information from B518 and B519 and resets M4133.

Free rotation with Module 9223

If a program has been started, the module may be called only in conjunction with an M/S/T/Q strobe. Auxiliary axes (e.g. rotary axes that are configured as auxiliary axes) are an exception; for them the PLC module can also be used without active strobe signal during a running NC program.

Module 9223 Free rotation

When the module is called, M4133 is set (start and stop).
The feed-rate override in W754 remains in effect.

Call:

```
PS    B/W/D/K  <Axis>
        0 ... 17

PS    B/W/D/K  <Feed rate [°/min]>
PS    B/W/D/K  <Mode>
        0: Stop
        +1: Start in positive direction
        -1: Start in negative direction

CM    9223
PL    B/W/D    <Error code>
        0: No error: Positioning is started/stopped
        1: No rotary axis transferred
        2: Impermissible feed rate
        3: Axis has not traversed the reference mark
        4: No M/S/T/Q strobe during running program
        5: Programmed axis not in closed loop
        6: Other traversing command active for axis at start of
           function
        7: Free rotation not active for axis at stop of function
```

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Faulty execution of module
W1022	2	No rotary axis transferred, or impermissible feed rate or open-loop axis programmed
	9	Other traversing command active for axis at start of function
	21	No M/S/T/Q strobe during started program
	33	Axis has not traversed the reference mark

		Set	Reset
M4133	Start and stop the free rotation function	PLC	NC
B518	Definition of the free rotation function 0: Cancel the function 8: Free rotation for axis 4 16: Free rotation for axis 5	PLC	PLC
B519	Traverse direction for free rotation 0: Axis 4 and axis 5 = + 8: Axis 4 = - , axis 5 = + 16: Axis 4 = +, axis 5 = - 24: Axis 4 and axis 5 = -	PLC	PLC
W754	% function for feed-rate override for free rotation	PLC	PLC
W566 - 568	Feed rate for free rotation Axis 4 to axis 5	PLC	PLC



Feed rate display

The programmed contour feed rate is displayed in the **Program run, single block** and **Program run, full sequence** operating modes. With the feed rate potentiometer you can change the feed rate from 0 to 150%.

If rapid traverse was programmed, **FMAX** is displayed and M4180 is set.

The percentage adjusted with the feed rate override is entered by the NC in W494 and W766.

You can change the percentage through the PLC:

- ▶ Enter the desired percentage in W766. The NC immediately takes over the new value.

Machine parameter MP7620 bit 3 and bit 4 is used to configure the characteristic curve and the behavior of the spindle-speed override and feed-rate override. If bit 3 is not set (= 0), the feed-rate and spindle potentiometers are now available as a linear characteristic curve. On TE keyboard units without rapid-traverse override, bit 4 serves to additionally select the resolution used for the evaluation of the overrides.

The rapid-traverse override cannot be configured and is always evaluated with a non-linear characteristic curve in steps from 0 to 15000.

Behavior of MP7620 bit 3 on TEs without rapid-traverse override:

0: Linear characteristic curve for spindle override and feed rate override; unit depends on bit 4 0..150 or 0..15000

1: Non-linear characteristic curve for spindle override and feed rate override, unit 0..15000

Behavior of MP7620 bit 4 on TEs without rapid-traverse override:

0: Spindle override and feed rate override with unit 0 to 150

1: Spindle override and feed rate override in steps from 0 to 15000

Behavior of MP7620 bit 3 on TEs with rapid-traverse override:

0: Linear characteristic curve for spindle-speed and feed-rate override, in steps from 0 to 15000

1: Non-linear characteristic curve for spindle override and feed rate override, in steps from 0 to 15000

Behavior of MP7620 bit 4 on TEs with rapid-traverse override:

No effect

The feed-rate override is effective either in 1% steps or according to a nonlinear characteristic curve:

- ▶ With MP7620 bit 3, select the mode of the override.

Value range in W494 and W766:

- 1% steps: 1 to 150
- Nonlinear characteristic curve: 0 to 15 000

In the lowest range, 0.01% steps are available. Beginning with a value of 2.5%, the step is 0.75%



Note

If bit 3 of MP7620 is set, then, due to the nonlinear characteristic curve, the labeling of the override only matches the actual value at the 0% and 100% positions.

In the manual modes of operation the axis feed rate is shown instead of the contouring feed rate.

You can choose between two types of display:

- The axis feed rate is shown after you press an axis-direction key. If two keys are pressed simultaneously, no feed rate is displayed.
- If no key is pressed, the smallest axis feed rate is always shown. The PLC axes are not included in the selection of the smallest feed rate. If more than one key is pressed simultaneously, a feed rate is also displayed.

► Define the type of display in MP7270.

MP7270 Feed rate display in the operating modes MANUAL OPERATION and ELECTRONIC HANDWHEEL

Input: 0: Display of axis feed rate through pressing an axis direction key (axis-specific feed rate from MP1020)
 1: Display of axis feed rate also before an axis direction key is pressed (smallest value from MP1020 for all axes)

MP7620 Feed-rate override and spindle speed override

Format: %xxxxxxx
 Input: Bit 0 – Feed-rate override if rapid-traverse key is pressed in Program Run mode:
 0: Override not effective
 1: Override effective
 Bit 1 – Non-functional
 Bit 2 – Feed-rate override if rapid-traverse key and machine-direction button are pressed in Manual mode:
 0: Override not effective
 1: Override effective
 Bit 3 – for TEs without rapid traverse override:
 0: Linear characteristic curve for spindle override and feed rate override; unit depends on bit 4 0..150 or 0..15000
 1: Non-linear characteristic curve for spindle override and feed rate override, unit 0..15000
 Bit 4 – for TEs without rapid traverse override:
 0: Spindle override and feed rate override with unit 0 to 150
 1: Spindle override and feed rate override in steps from 0 to 15000
 Bit 3 – for TEs with rapid traverse override:
 0: Linear characteristic curve for spindle-speed and feed-rate override, unit 0..15000
 1: Non-linear characteristic curve for spindle override and feed rate override, unit 0..15000
 Bit 4 – for TEs with rapid traverse override:
 No effect

		Set	Reset
W494	Percentage for feed rate override (NC to PLC)	NC	NC
W766	Percentage for feed rate override (PLC to NC)	NC/PLC	NC/PLC
M4180	Rapid traverse programmed (FMAX)	NC	NC



Feed rate for rotary axes

The iTNC interprets the programmed feed rate for a rotary axis in degrees per minute. The contour feed rate depends on the distance of the tool center from the center of the rotary axis.

With the M116 function the contouring feed rate can be converted to mm/min. In this way the feed rate is independent of the distance from the tool center to the center of axis rotation:

- ▶ Define the rotation center of the rotary axis with MP75xx. Also see page 701.

Display of the M functions

The following functions are displayed in the status window:

- M03, M04, M05: Miscellaneous functions for spindle control
- M07, M08, M09: Miscellaneous functions for coolant control

You can control the display of these functions through the PLC.

- M4005, M4006: Status display, M03 and M04 change the direction of rotation of the spindle.
- M4008: Blocks the speed output for the spindle. The programmed spindle speed continues to be displayed. At the same time, M03, M04 or M05 are highlighted. The nominal speed value is zero.

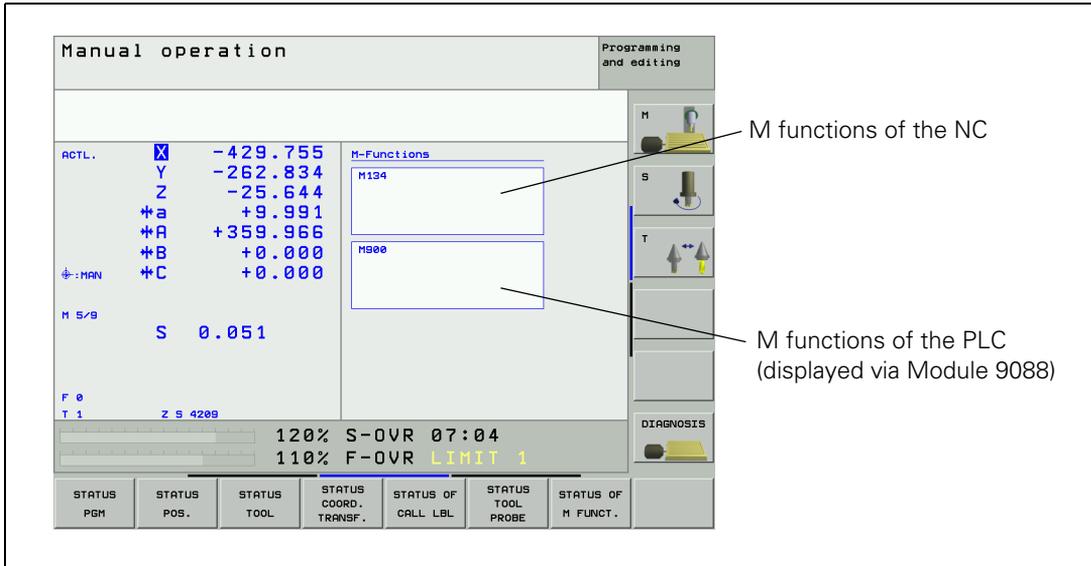
Other M functions of the NC are shown in the status window.

		Set	Reset
M4005	Status display and nominal speed value output for M03	PLC	PLC
M4006	Status display and nominal speed value output for M04	PLC	PLC
M4007	Status display M05 and spindle stop	PLC	PLC
M4008	Disable speed output for spindle	PLC	PLC
M4040	Status display M07, M08, and M09 highlighted	PLC	PLC
M4041	Status display M07, M08, M09, MK	PLC	PLC
M4042	Status display M07, M08, M09, MK	PLC	PLC

M4041	M4042	Display
0	0	M09
1	0	M07
0	1	M08
1	1	MK

M functions of the PLC can be displayed in the status window:

- ▶ With Module 9088 you can display M functions in the status window or delete them.



Module 9088 Display the M functions

Call:

PS B/W/D/K <Number of the M function to be displayed>

PS B/W/D/K <Mode>

-1: Delete all M functions in the status window

0: Delete M function

1: Display M function

CM 9088

Error code:

Marker	Value	Meaning
M4203	0	M function displayed or deleted
	1	Error code in W1022
W1022	1	Invalid M-function number
	2	Invalid mode number

Control in operation

If the control is at work, e.g. executing a positioning movement or an M function, a symbol resembling an asterisk (*) is shown in the status window.

If a running NC program is interrupted with an external stop key, the control-in-operation symbol blinks in the status display window.

In the **Positioning With Manual Data Input, Program Run, Single Block** and **Program Run, Full Sequence** operating modes, these conditions are reported to the PLC with M4175 and M4176.

The marker M4175 (control-in-operation symbol is blinking) and the marker M4176 (control-in-operation symbol is on or blinking) are not cleared simultaneously upon an internal stop. The marker M4176 remains set during a subsequent RUNCANCEL macro, whereas marker M4175 is already reset. This shows that the control is still in operation during the RUNCANCEL macro. The marker M4176 is not cleared until execution of the RUNCANCEL macro has been completed.

If—due to the resetting of marker M4176 (control-in-operation symbol is on or blinking) while the RUNCANCEL macro is running—impermissible PLC reactions (e.g. range switchover, PLC positioning movements, overwriting of machine parameters) are triggered in the PLC program, the fatal error message **PLC strobe not allowed (M4193)** will be issued. If the error message **PLC strobe not allowed (M4193)** appears, HEIDENHAIN recommends setting bit 17 of MP7680 and adapting the PLC program.

Also, PLC Marker M4193 (RUNCANCEL macro active) is available. The marker is set if a RUNCANCEL macro is active.

To delete or display the control-in-operation symbol through the PLC:

- ▶ Ensure that the control-in-operation symbol is not already blinking or being displayed by the NC.
- ▶ Enter the command code in Module 9089.

Module 9089 Control in operation

The control-in-operation symbol can be set only if it is not already being displayed by the NC, since the NC has priority over the PLC. If the symbol is being displayed by the NC, it cannot be erased.

M4176 is not influenced by the display of the control-in-operation symbol through the PLC.

Call:

PS	B/W/D/K	<Command code>
		0: Clear the control-in-operation symbol
		1: Display the control-in-operation symbol
CM	9089	
PL	B/W/D	<Error code>
		0: Control-in-operation symbol was cleared/displayed
		1: Incorrect error code
		2: Control-in-operation symbol is already being displayed by the NC
		3: Control-in-operation symbol is blinking
		4: Control-in-operation symbol was not cleared because it is already being displayed by the NC

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Incorrect transfer parameter

		Set	Reset
M4175	Program interruption, control-in-operation symbol is blinking	NC	NC
M4176	Control is in operation, control-in-operation symbol is on or is blinking	NC	NC
MP7680	Bit 17 – Behavior of Markers M4175 and M4176		
Input:	0: Clear M4175 and M4176 upon internal stop 1: M4176 remains set during RUNCANCEL macro		



Erasing the status information

To erase the status information, tool data and contents of the Q parameters:

- ▶ Select the conditions with MP7300.

Status information is all programmed values, such as scaling factor, datum shift, feed rate, tolerance from Cycle 32, etc. These are all reset, and the tolerance programmed in Cycle 32 is erased (is only effective again with MP1096.x). The Q parameters and tool data are set to zero.

MP7300 Erasing the status information, tool data and Q parameters

- Input:
- 0: Erase the status information, Q parameters and tool data if a program is selected.
 - 1: Erase the status information, Q parameters and tool data if a program is selected and in the event of M02, M30, and END PGM.
 - 2: Erase the status information and tool data if a program is selected.
 - 3: Erase the status information and tool data if a program is selected and in the event of M02, M30, END PGM.
 - 4: Erase the status information and Q parameters if a program is selected.
 - 5: Erase the status information and Q parameters if a program is selected and in the event of M02, M30, END PGM.
 - 6: Erase the status information if a program is selected.
 - 7: Erase the status information if a program is selected and in the event of M02, M30, END PGM.

Input	Erase with PGM MGT	Erase with M02, M30, END PGM	Status information	Tool data	Q parameters
0	x	–	x	x	x
1	x	x	x	x	x
2	x	–	x	x	–
3	x	x	x	x	–
4	x	–	x	–	x
5	x	x	x	–	x
6	x	–	x	–	–
7	x	x	x	–	–

Interrogating the status display through the PLC

Module 9035 Read status information

With this module you can interrogate the status display or read the status information. You transfer a number indicating the desired information. In order to receive information about the status of the alternative **smarT.NC** operating mode, please use the operating mode markers or the operating mode word (see "Operating modes" on page 1165).

Transferred number	Return code
0 Editor mode in foreground	0: Programming and Editing 1: Test Run
1 Machine mode in foreground	0: Traverse reference points 1: Manual operation 2: Electronic Handwheel 3: Positioning with Manual Data Input 4: Program Run, Single Block 5: Program Run, Full Sequence
2 Editor mode in background	0: None (main operating mode active) 1: MOD active 2: Program management/external interfaces active 3: MP editor active 4: PLC programming active
3 Machine mode in background	0: None (main operating mode active) 1: MOD active 2: Program management/external interfaces active 3: Tool table selected 4: Pocket table selected 5: Other table editor active
4 Displayed screen window	Bit encoded Bits 0 to 7: Editing screen Bit 0=1: Editing screen is displayed Bit 1=1: Operating-mode window active Bit 2=1: Block display/program select/setup window active Bit 3=1: Position display active Bit 4=0: Always 0 / Not possible Bit 5=1: Status/Graphics window active Bits 6/7: Reserved Bits 8 to 15: Machining screen Bit 8=1: Machining screen is displayed Bit 12=1: PLC status window active in machining screen Bits 9 to 15: Reserved
5 Selected file in Programming and Editing or Test Run	0: No file 1: *.H (conversational NC PGM) 2: *.I (ISO NC PGM) 3: *.T (tool table) 4: *.D (datum table) 5: *.P (pallet table) 6: *.A (ASCII file) 7: *.TCH (pocket table)
6 Selected file in Program Run, Full Sequence and Program Run, Single Block	0: No file 1: *.H (conversational NC PGM) 2: *.I (ISO NC PGM)



Transferred number		Return code
7	Selected axis for actual position capture in Programming and Editing mode	0 to 17: Axis 1 to 18
8	Selected axis for actual position capture in Positioning with MDI mode	0 to 17: Axis 1 to 18
9	Handwheel axis	-1: None or more than one 0 to 2: X, Y, Z 3 to 17: Axis 4 to 18
10	Handwheel axis, bit-encoded	Bit 0: X axis Bit 1: Y axis Bit 2: Z axis Bit 3 to bit 17: Axis 4 to 18
	Handwheel subdivision factor	
11	X key	0 to 10
12	Y key	
13	Z key	
14	IV key	
15	V key	
16	Input format of the \$MDI file	0: *.H (conversational NC PGM) 1: *.I file (ISO NC PGM)
17	Display format	0: mm 1: inches
18	Tilting the working plane	Bit 0=1: Tilting is active (manual and/or during program run) Bit 1=1: Tilting is active in manual operation Bit 2=1: Tilting is active in program run
19	Active line in the *.CMA file	>=0: Line number -1: No *.CMA file
20	HR 410 speed	0: Slow 1: Medium 2: Fast
21	Control model	0: TNC 310 1: TNC 370 2: TNC 410 3: TNC 426 CA/PA 4: TNC 426 CB/PB/M or TNC 430 CA/PA/M 5: iTNC 530 6: iTNC 530 (with Windows 2000/XP) 20: ATEK M
22	Status of M128	0: M128 not active 1: M128 active
23	Handwheel superimposition with M118	0: M118 not active Bit 0 to 17: Axis 1 to 18
26	Jog increment	
27	Traverse range	
28	Query of the superimposed operating mode "Table editing" in the "Machine" operating mode	0: No table in editing mode 3: Tool table in editing mode 4: Pocket table in editing mode

Transferred number		Return code
	Handwheel subdivision factor	
31	Axis 1	0 to 10
32	Axis 2	
33	Axis 3	
34	Axis 4	
35	Axis 5	
36	Axis 6	
37	Axis 7	
38	Axis 8	
39	Axis 9	
	Tool change	
50	Tool change sequence (see FN18: SYSREAD ID61 NRO)	
51	Pocket number for reserve	
52	Magazine number for reserve	
53	Pocket number for insertion	
54	Magazine number for insertion	
	Handwheel	
70	Active handwheel	
71	Standard handwheel	
100	Number of the tool axis	
1000	Table editor (only in a spawn job or submit job)	>= 0: Active line in the table editor -1: Table editor not active
1001	Pallet table (only in a spawn job or submit job)	>= 0: Active line in the pallet table -1: Pallet table not active
1002	Status of pallet machining	-1: Main program is not a pallet table 0: Machining was not started 1: NC program is selected but not started 2: NC program was started 3: Pallet-change macro was started 4: Macro from the PALEPILOG entry in NCMAKRO.SYS was started 5: Pallet-change macro was started by the PLC (Module 9280)

Call:

PS B/W/D/K <Number of the desired status information>

CM 9035

PL B/W/D <Status information>

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Status information invalid
	20	Call was not in a submit or spawn job



8.1.2 Operating modes

The iTNC features main operating modes, background operating modes, and the alternative **smarT.NC** operating mode. The background operating modes are each called from the main operating modes.

Each operating mode and the varied corresponding information can be interrogated and evaluated via the PLC.

The "Machine" and "Editor" operating modes can be interrogated via markers, W272 and a PLC module (see "Module 9035 Read status information" on page 1162), and the alternative **smarT.NC** operating mode can be interrogated via Marker M4163 in combination with W272.

		Set	Reset
M4150	"Manual Operation" main operating mode is active	NC	NC
M4151	Electronic Handwheel main operating mode is active	NC	NC
M4152	Operating mode marker <ul style="list-style-type: none"> ■ Positioning with Manual Input main operating mode (if smarT.NC inactive → M4163 = 0) ■ RUN ACTIVE UNIT smarT.NC main operating mode (if smarT.NC active → M4163 = 1) 	NC	NC
M4153	Operating mode marker <ul style="list-style-type: none"> ■ Program Run, Single Block main operating mode (if smarT.NC inactive → M4163 = 0) ■ RUN SINGLE UNITS smarT.NC main operating mode (if smarT.NC active → M4163 = 1) 	NC	NC
M4154	Operating mode marker <ul style="list-style-type: none"> ■ Program Run, Full Sequence main operating mode (if smarT.NC inactive → M4163 = 0) ■ RUN ALL UNITS smarT.NC main operating mode (if smarT.NC active → M4163 = 1) 	NC	NC
M4155	Traversing the Reference Points main operating mode is active	NC	NC
M4163	Alternative operating mode "smarT.NC" is active	NC	NC
M4186	NC program was started in the Test Run mode	NC	NC

	Set	Reset
W272	NC	NC
Operating mode word		
<ul style="list-style-type: none"> ■ If smarT.NC inactive → M4163 = 0 <ul style="list-style-type: none"> 1: MANUAL OPERATION 2: ELECTRONIC HANDWHEEL 3: POSITIONING WITH MANUAL DATA INPUT 4: PROGRAM RUN, SINGLE BLOCK 5: PROGRAM RUN, FULL SEQUENCE 7: TRAVERSING THE REFERENCE POINTS ■ If smarT.NC active → M4163 = 1 <ul style="list-style-type: none"> 3: RUN ACTIVE UNIT 4: RUN SINGLE UNITS 5: RUN ALL UNITS 		

"Machine" operating modes

- "Machine" main operating modes
 - **Traverse reference points** (not with absolute encoders)
 - **Manual Operation**
 - **El. Handwheel**
 - **Positioning with Manual Data Input**
Machine individual NC blocks without a special program context
 - **Program Run, Single Block**
Machine an NC program block by block. Each NC block must be started manually.
 - **Program Run, Full Sequence**
Completely machine the NC program
- "Machine" background operating modes (cannot be called from every main operating mode)
 - **MOD**
Basic settings and information
 - **Program Management/Interfaces**
File manager and configuration of external interfaces
 - **Edit Tool Table**
 - **Edit Pocket Table**
 - **Edit Other Tables**

"Editor" operating modes

- "Editor" main operating modes
 - **Programming and Editing**
 - **Test Run**
Virtual machining of a workpiece with nearly real-time graphic simulation
- "Editor" background operating modes
 - **MOD**
General settings and settings via code number
 - **Program Management/Interfaces**
 - **MP Editor**
 - **PLC programming**



smarT.NC alternative operating mode

Overview of the smarT.NC operating modes

- **smarT.NC: Programming** operating mode
Program creation. This smarT.NC operating mode is comparable to **Programming and Editing** of the Editor main operating mode.
- **smarT.NC: File Management** operating mode
- **smarT.NC: Testing** operating mode
This smarT.NC operating mode is comparable to **Test Run** of the Editor main operating mode.
- **smarT.NC: Machining** operating mode
Program run. This smarT.NC operating mode is comparable to **Program Run, Single Block** and **Program Run, Full Sequence** of the Editor main operating mode.
It is divided into the following sub-operating modes:
 - **„RUN SINGLE UNITS**
Machine individual units without a special program context
 - **„RUN ALL UNITS**
Completely machine the NC program
 - **„RUN ACTIVE UNIT**
Machine an NC program unit by unit. Each unit must be started manually.

Disabling operating modes

PLC Module 9285 can be used to disable operating modes and other functions.

Module 9285 Set the access level

PLC Module 9285 locks/enables the following predefined functions. If the bit is set in the mask, the function is locked. If the bit is set to 0, the function is enabled:

Group 0: Disabling operating modes	Bit mask
Manual Operation operating mode	0x01
Electronic Handwheel operating mode	0x02
Positioning with Manual Data Input operating mode	0x04
Program Run, Single Block operating mode	0x08
Program Run, Full Sequence operating mode	0x10
smarT.NC operating mode	0x20

Group 1: Disabling write-access to files	Bit mask
NC programs (* .H, *.I, *.HU, *.HP, *.HC, *.DXF)	0x01
Tool table	0x02
Pocket table	0x04
Preset table	0x08
Pallet table	0x10

Group 2: Disabling other functions	Bit mask
Manual probing	0x01
Code numbers	0x02
Fixture management	0x04

Call:

PS B/W/D/K <Group number>
0: Disable operating modes
1: Disable write-access to tables
2: Disable other functions

PS B/W/D/K <Bit mask>

CM 9285

PL B/W/D <Status>
0: Function performed
1: Illegal group number
2: Incorrect parameterization via bit mask
20: Module was not called in a spawn job or submit job



Error code:

Marker	Value	Meaning
M4203	0	Function performed
	1	Error code in W1022
W1022	1	Invalid group number
	2	Invalid value for bit mask
	20	Module was not called in a spawn job or submit job

8.1.3 Operating times

The iTNC can measure up to 16 operating times and store them in a file in the SYS partition:

Operating time	Meaning
TNCTIME	Control on
MACHINETIME	Machine on
PROGTIME	Program run
PLCTIME0 to PLCTIME12	Definable times of the PLC

For all operating modes except **Programming and Editing** and **Test Run**:

- ▶ Press the MOD key and press the **MACHINE TIME** soft key.
- ▶ With MP7237.x, specify the times that can be reset with the code number 857282, and the PLC operating times that you wish to display.
- ▶ In MP7238.x, define the dialog messages to be displayed for the individual operating times.

The time is measured in seconds and is updated every minute during the run time. When the control is switched off, no more than one minute is lost.

The NC measures the time for **Control on**, **Machine on** and **Program run**.

For PLC operating times 1 to 13:

- ▶ Start with Module 9190.
- ▶ Stop with Module 9191.

Except for **Control on**, all operating times are saved during a hard-disk backup. see "Data backup" on page 179.

With the following modules you can evaluate and change the operating times:

- Module 9190: Starting the operating times
- Module 9191: Stopping the operating times
- Module 9192: Reading the operating times
- Module 9193: Setting the operating times
- Module 9194: Alarm when operating times are exceeded

MP7237 Display and reset the operating times

Format: %xxxxxxxxxxxxx

MP7237.0 Displaying PLC operating times

Input: Bits 0 to 12 represent PLC operating times 1 to 13
0: Do not display
1: Display

MP7237.1 Reset PLC operating times with code number 857282

Input: Bits 0 to 12 represent PLC operating times 1 to 13
0: Do not reset
1: Reset

MP7237.2 Reset PLC operating times with code number 857282

Input: Bit 0 – Nonfunctional
Bit 1 – "Machine on" operating time
Bit 2 – "Program run" operating time
0: Do not reset
1: Reset

MP7238.0-12 Dialog messages for PLC operating times 1 to 13

Input: 0 to 4095
Dialog no. from the file **PLCDIALOG=** (OEM.SYS)



Module 9190 Start the operating times

You start one or more operating times.

Call:

PS B/W/D/K <PLC operating time>

Bits 0 to 12 represent PLC operating times 1 to 13

CM 9190

Error code:

Marker	Value	Meaning
M4203	0	PLC operating time started
	1	Incorrect parameter

Module 9191 Stop the operating times

You stop one or more operating times.

Call:

PS B/W/D/K <PLC operating time>

Bits 0 to 12 represent PLC operating times 1 to 13

CM 9191

Error code:

Marker	Value	Meaning
M4203	0	PLC operating time started
	1	Incorrect parameter

Module 9192 Read the operating times

You read the current value of an operating time. The current value is transferred in seconds. If the value is greater than 2 147 483 648 (approx. 69 years), a negative number will be transferred.

Call:

PS B/W/D/K <Number of the operating time>
-3: **Control on**
-2: **Machine on**
-1: **Program run**
0 to 12: PLC operating times 1 to 13

CM 9192

PL B/W/D <Current time [s]>
-1: Error

Error code:

Marker	Value	Meaning
M4203	0	PLC operating time started
	1	Incorrect transfer value, or module was not called in a spawn or submit job

Module 9193 Set the operating times

You overwrite the current value of the operating time. The old value is lost irretrievably.

The time for **Control on** cannot be overwritten.

Transfer all values greater than 2 147 483 648 (approx. 69 years) as negative numbers.

Constraint:

- So as not to impair the behavior of the overall system, the module may only be used for re-initializing the machine times, and may not be called too often. The module (and therefore the calling submit or spawn job) waits for acknowledgment.

Call:

PS B/W/D/K <Number of the operating time>
-2: **Machine on**
-1: **Program run**
0 to 12: PLC operating times 1 to 13

PS B/W/D/K <New time [s]>

CM 9193

Error code:

Marker	Value	Meaning
M4203	0	Operating time was overwritten
	1	Incorrect transfer value, or module was not called in a spawn or submit job



Module 9194 Alarm when operating time exceeded

You define a marker that is set when a certain threshold is exceeded. The marker is set every minute after the threshold is passed. The marker can be delayed by max. 59 s the first time it is set. All values greater than 2 147 483 648 (approx. 69 years) must be transferred as negative numbers.

If you enter the value zero as the alarm threshold, the function is deactivated.

Call only in a submit job or spawn job.

Call:

PS B/W/D/K <Number of the operating time>
-3: **Control on**
-2: **Machine on**
-1: **Program run**
0 to 12: PLC operating times 1 to 13

PS B/W/D/K <Alarm threshold [s]>

PS B/W/D/K <Number of the alarm marker>

CM 9194

Error code:

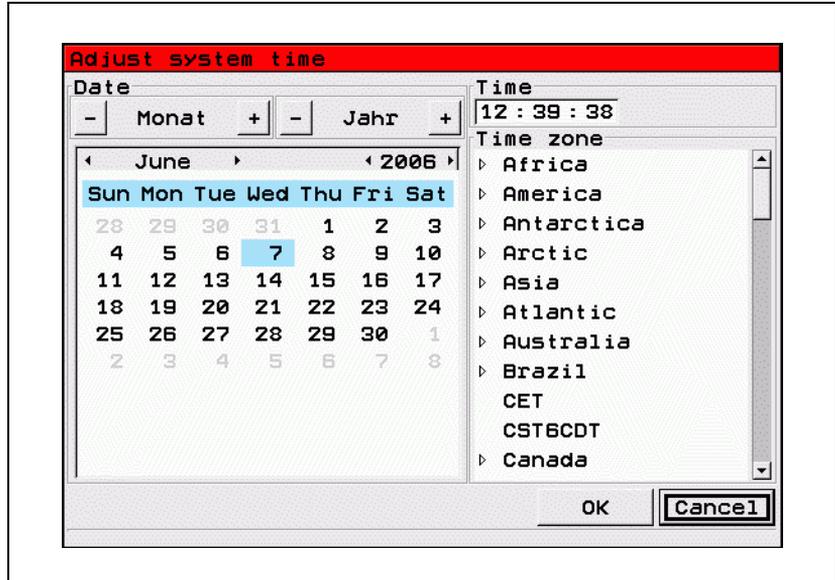
Marker	Value	Meaning
M4203	0	Alarm function activated
	1	Incorrect transfer value, or module was not called in a spawn or submit job

System time

Setting the local time via soft key (only for single-processor control)

Press the MOD key in the Programming and Editing mode of operation to display the SET DATE/TIME soft key. After pressing this soft key, the following window in which you can enter the current local time appears. If you make any changes to these settings, the control must be restarted.

Set the time zone for the control to automatically switch between standard and daylight-saving time.



Module 9195 System time

At the factory, the system time of the iTNC is set to Central European Time or Central European Summer Time in the BIOS. The iTNC internally operates with UNIX system time. The UNIX system time contains the number of seconds accumulated since 0:00 hours on January 1, 1970. The iTNC calculates from the system time (BIOS) to the UNIX system time.

Enter the current local time via the SET DATE/TIME soft key so that the program manager time matches the local time.

With Module 9195 you can read the current value of the UNIX system time. The value read with Module 9195 is independent of the indicated local time and always refers to Universal Time (Greenwich time).

Call:

CM 9195

PL D <System time>

Number of seconds since 0:00 hours on January 1, 1970.

MP7235 Time difference to Universal Time (Greenwich Mean Time)

Input: -23 to +23 [hours]

0: Universal Time (Greenwich Mean Time)

1: Central European Time (CET)

2: Central European daylight-saving time

Module 9055 Local time

With Module 9055 you can convert the value read with Module 9195 into a legible ASCII format. Module 9055 corrects the transferred value to local time.

Call:

```
PS    B/W/D/K <System time>
      Number of seconds since 0:00 hours on January 1, 1970.
PS    B/W/D/K <String number for the result>
PS    B/W/D/K <Format>
      0: DD.MM.YYYY hh:mm:ss
      1: D.MM.YYYY h:mm:ss
      2: D.MM.YYYY h:mm
      3: D.MM.YY h:mm
      4: YYYY-MM-DD- hh:mm:ss
      5: YYYY-MM-DD- hh:mm
      6: YYYY-MM-DD h:mm
      7: YY-MM-DD- h:mm
      8: DD.MM.YYYY
      9: D.MM.YYYY
     10: D.MM.YY
     11: YYYY-MM-DD
     12: YY-MM-DD
     13: hh:mm:ss
     14: h:mm:ss
     15: h:mm
```

CM 9055

Error code:

Marker	Value	Meaning
M4203	0	String was created
	1	Incorrect transfer value

8.1.4 Error messages

Error messages are displayed by the NC or PLC below the operating mode:

- ▶ If the error message blinks, switch the machine off and correct the error. If the error message does not blink, M4177 is set.

You can call PLC error messages with Module 9084 or 9085, or by activating a marker (M4800 to M4999):

- ▶ Define your PLC error messages in the *.PET table.
- ▶ Assign the markers to the error messages.

With Module 9086 you can delete PLC error messages, and with Module 9087 you can interrogate the current status of the error message.



Note

A *.PET table is absolutely mandatory, since without it the PLC program cannot be compiled or activated.

If a *.PET table contains more than 2048 error messages, the excessive messages are ignored and the error message **PET table: Too many lines** appears.

- ▶ In the OEM.SYS file, use the command **PLCERRTAB=** to enter the name of the *.PET table.
- ▶ You can automatically generate the entry by calling COMPILE.

If more than one PLC error message is activated at once:

- ▶ Press the CE key to read the error messages in succession. With the special command **ERRQUE=** you can display the list of active error messages in the large PLC window.
- ▶ Press the ERR key to have a list of all current error messages displayed. These can be selected and cleared in a targeted manner.



Error list

The columns of the error list (ERR key) have the following meanings:

- **Number:** Error number (–1: no error number defined), assigned by HEIDENHAIN or the machine tool builder
- **Class:** Error class. Defines how the iTNC processes this error:
 - **ERROR:** Collective error class
The error reaction depends on the status or current operating mode of the control
 - **FEED HOLD:** The feed-rate entry is cleared
 - **PGM HOLD:** Program run is interrupted (control-in-operation symbol blinks)
 - **PGM ABORT:** Program run is canceled (internal stop)
 - **EMERG. STOP:** An emergency stop is triggered
 - **RESET:** iTNC performs a reset
 - **WARNING:** Warning message, program run is continued
 - **INFO:** Informational message, program run is continued
- **Group:** Error source. Shows the cause of the error:
 - **GENERAL:** Other error
 - **OPERATING:** Error during machining
 - **PROGRAMMING:** Error during editing
 - **PLC:** Error by the PLC
- **Error message:** Error text displayed by the iTNC

PET table

The PLC error message table (*.PET) consists of the following columns, to which you can assign special attributes:

- **NR**
Line number in the table. The modules select the PLC error message by assigning the line number.
- **ERROR**
With the HELP key you can display information on the error messages. (see "Help" on page 1189).
There are three ways to specify the error text:
 - Direct entry of the error text (max. 32 characters)
 - Line number of the PLC error text file (# <line no.>) specified with **PLCERROR=** in OEM.SYS.
 - Number of the string memory containing the error text (# <string no.>)
- **MARKER**
The PLC error message can be activated without module call by setting the marker defined here (M4800 to M4999). The marker is also set if the error message was activated through Module 9085.
Entry 0: No error marker
- **RESET**
0: No NC reset upon activation of the error message. Error display does not blink.
1: NC reset upon activation of the error message. Error display flashes.
- **NC-STOP**
0: No NC stop upon activation of the error message
1: NC stop upon activation of the error message (M4221 is set)

■ **NC-CANCEL**

0: No NC stop with subsequent INTERNAL STOP upon activation of the error message

1: NC stop with subsequent INTERNAL STOP upon activation of the error message (M4223 is set)

■ **F STOP**

0: Feed-rate enable is not influenced

1: Feed rate-enable is reset upon activation of the error message (M4220 is set)

■ **EMER.STOP**

0: No emergency stop upon activation of the error message

1: Emergency stop upon activation of the error message (M4222 is set)

■ **AE**

You can use marker M4225 to activate an alternative error reaction, provided that it is available in the PET table. The column **AE** (Alternative error reaction E/C/S/F) of the PET table is evaluated for this. However, the error reactions that are active at the time an error message occurs will become effective. This means that error messages that were initiated before marker M4225 was set, but have not been displayed yet, will trigger the original error reactions. The following entries can be made in the **AE** column:

- **E = EMER.STOP**
- **C = NC-CANCEL**
- **S = NC-STOP**
- **F = F-STOP**
- **N = No reaction**
- **X = Error will be ignored**

■ **CE**

0: Error message can be cleared with the CE key

1: Error message cannot be cleared with the CE key

■ **PRIOR**

A priority of 0 to 2 can be entered for the error message. Priority 0 (error) is the highest priority, then priority 1 (warning) and priority 2 (info). PLC error messages triggering an emergency stop receive the highest priority (independent of the priority from the *.PET table). Therefore, these error messages always appear at the first position in the error list. The error message **External emergency stop** has a lower priority, but still a higher priority than the top PLC priority. This means that PLC error messages triggering an emergency stop always appear at the first position in the error list. They are followed by **external emergency stop** and then by further PLC error messages, depending on their priority. Depending on their priority, M4227 to M4229 are set for active error messages. Different colors can be set for the priorities (see "Color settings" on page 1267).

■ **MType**

This column is reserved for future applications. Enter E.

■ **ONL-NAME**

Name of a help file (*.CHM).

■ **ONL-NUMMER**

Help number within the help file under ONL-NAME.

■ **SF**

0: No service file is generated upon an error message.

1: A service file is generated upon an error message.



■ REACT

This attribute results in an NC stop at block end if the error marker is set at this time:

0: No NC stop at block end upon activation of error message

1: NC stop or INTERNAL STOP at block end upon activation of the error message



Note

With the attribute **REACT**, note that

- Cycles are executed completely before an NC stop is triggered.
- With smarT.NC units, the current block is executed completely before an NC stop is triggered.
- An NC stop is triggered only if the error marker is (still) set at the end of the block or the end of the cycle.
- Together with the PLC error message, the NC sets Markers W272 = 4 and M4153 = 1. Thus, an operating mode that is not the one being displayed is reported in the PLC.

■ LIFTOFF

The reaction is: immediate NC stop, lift-off and emergency stop after standstill. If an error message of this class was generated, this can be evaluated in the PLC program via the PLC Marker 4224.

■ PLC

A bit pattern, which can be read out by the PLC program, can be saved. The hexadecimal input range in the PLC column is \$0 to \$FFFFFF. The PET column is read out with Module 9393.

		Set	Reset
M4220	Error from PET table with F stop active	NC	NC
M4221	Error from PET table with NC stop active	NC	NC
M4222	Error from PET table with emergency stop active	NC	NC
M4223	Error from PET table with NC Cancel active	NC	NC
M4225	Alternative error reaction active	PLC	PLC
M4227	PLC error message with priority 0 (error)	NC	NC
M4228	PLC error message with priority 1 (warning)	NC	NC
M4229	PLC error message with priority 2 (info)	NC	NC

Module 9084 Display PLC error messages with additional data

With Module 9084 you can display PLC error messages with additional data. You can insert placeholders (%s, %d, %f) at any position of the error messages. The placeholders are assigned the data from the module at run time. Only those placeholders that are defined in the PLC error message will be replaced. %s is replaced by the string or the string content. The first occurrence of %d or %f in the PLC error message is replaced by the content of variable 1, and the second occurrence of %d or %f is replaced by the content of variable 2. %d is an integer, %f is a floating point number with three decimal places. Alternatively, you can define the number of decimal places with %.1f to %.6f. If the module is called several times with the same line number of the *.PET table, the error message is entered only once in the queue. A maximum of 32 PLC error messages can be entered in the queue. If necessary, the error marker assigned is set. If the *.PET table or the line number is not found, the following error message appears:

PLC-ERROR <line number>.

Machine parameters for the PLC are stored in freely definable tables (tool change positions, service feed rate / rpm...). The name of the file varies depending on the machine type or number and is defined in the OEM.SYS. e.g. MP_DATA_TAB = PLC:\MP\MMP_DATA_FILE.TAB
PLC Module 9084 can also resolve a logical name (>OEM.name) for the additional text. Restriction: This is only possible if the module was called from a submit/spawn job.

Call:

```
PS    B/W/D/K <Line number of the *.PET table>  
      0 to 2047: Line number  
PS    B/W/D/K/S<Data for %s>  
PS    B/W/D/K <Data for %d or %f; variable 1>  
PS    B/W/D/K <Data for %d or %f; variable 2>  
CM    9084
```

Error code:

Marker	Value	Meaning
M4203	0	PLC error message with additional data displayed
	1	Error code in W1022
W1022	1	Line number missing
	11	Invalid string in data for %s
	23	Overflow of PLC error message queue



Module 9085 Display PLC error message

Up to 32 error messages can be placed in the queue, of which up to eight can be from the string memory.

Blinking error message: Is displayed without entry in the queue.

Error number -1: Blinking error message **EMERGENCY STOP PLC** is displayed, even if no *.PET table was defined.

Error number not equal to -1 and no *.PET table selected:

Blinking error message **PLC: ERROR TABLE MISSING**

Call:

PS B/W/D/K <Line number of the *.PET table>
0 to 2047: Line number
-1: Blinking error message **EMERGENCY STOP PLC**

CM 9085

Error code:

Marker	Value	Meaning
M4203	0	Error message displayed or in queue
	1	Error code in W1022
W1022	1	Line number missing
	8	Incorrect operating mode, compatibility error marker set
	23	Overflow of PLC error message queue, or too many error messages from string memory

Module 9086 Clear PLC error message

With this module you can clear all set PLC error messages or a specific (nonblinking) error message in the queue.

Call:

PS B/W/D/K <Line number of the *.PET table>
0 to 2047: Line number
-1: Clear all PLC error messages

CM 9086

Error code:

Marker	Value	Meaning
M4203	0	Error message displayed or in queue
	1	Error code in W1022
W1022	1	Line number missing
	8	Incorrect operating mode, compatibility error marker set

Module 9087 Status of PLC error message

Call:

PS B/W/D/K <Line number of the *.PET table, status code>
0 to 2047: Line number
-1: PLC error message, general
-2: Number of the active PLC error message
-3: Number of error messages in the *.PET table

CM 9087

PL B/W/D <Status/error code>
For code 0 to 2047:
0: No error message with the number, or message cleared
-1: Line number not found (see W1022)
Bit 0 – PLC error message is displayed
Bit 1 – PLC error message in queue
For code -1:
0: No PLC error message
2: PLC error message in queue
For code -2:
≥ 0: Number of the displayed error
-1: No error in the *.PET table
For code -3:
≥ 0: Number of errors in the *.PET table

Error code:

Marker	Value	Meaning
M4203	0	Status information was read
	1	Error code in W1022
W1022	1	Invalid line number of status code

Module 9392 Display PLC error message with help offset

With Module 9392, you can display a PLC error message with an additional help offset and further parameters. The error message must be defined in the .PET table.

The behavior of the module corresponds to the behavior of PLC Module 9084 used in conjunction with Module 9391.

The wild cards %s, %d and %f can be defined at the appropriate places in the error texts. The individual module parameters are assigned to the wild cards (as described in the module interface), and are entered in the error texts at run time. Only the wild cards that occur in an error text are replaced. The wild cards %.1f ... %.6f can be used in order to show the places after the decimal point. If %f is entered, three decimal places are used.



The module can be called from a cyclic PLC program or from a spawn or submit job.

An offset is added to the value for the help number in the .PET file in order to generate the actual help number. This way a group error number can be defined for an (OEM) device in the .PET table. The error number (used as an offset) supplied by the device in case of error then leads to the appropriate help text.

Condition:

- The .CHM help file must be stored language-sensitive in the TNC:\tncguide\de directory, or in TNC:\tncguide\en etc.
- An OEM-specific OEMx.CHM file is necessary.
- If no OEMx.CHM file is indicated, the online help is not called. All other reactions that are saved for this error in the .PET table are performed.

Call:

```
PS    K/B/W/D  <Line number in the .PET error table>>
PS    K/B/W/D  <Additional text from S0 to S99 or constant string>
PS    K/B/W/D  <Variable 1>
PS    K/B/W/D  <Variable 2>
PS    K/B/W/D  <Offset for the help number in .PET>
CM    9392
```

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Transferred parameter outside of value range or error number is not in .PET table
	2	Invalid parameter
	3	String address outside of value range
	8	Incorrect operating mode, compatibility error marker set
	23	Overflow of PLC error message queue

Module 9393 PET column for PLC notification

The column 'PLC' was added to the PET table. A bit pattern, which can be read out by the PLC program, can be saved. The hexadecimal input range in the PLC column is \$0 to \$FFFFFF. The PET column is read out with Module 9393.

Call:

```
PS   B/W/D/K <PET line number>
      Error number of the PLC
PS   B/W/D/K <Mode>
      0: PLC column
CM   9393
PL   B/W/D   <Value of the PLC column>
      Bit pattern from PLC column, in case of error = 0
```

Error code:

Marker	Value	Meaning
M4203	0	Function performed
	1	Error code in W1022
W1022	1	Invalid PET line number programmed
	2	Invalid mode programmed

Example:

```
PS   K10                ; Error 10 from .PET
PS   S"ERROR-ERROR"    ; %s
PS   K9000              ; %d
PS   W100               ; %.1f
PS   K2                 ; Offset for the help number in .PET
CM   9392
```

Assumption: Text in .PET "%s error in module %d, W100 = %.1f" and W100 = 1234

Therefore, the following output text results:
"ERROR-ERROR error in module 9000, W100 = 123.4"



8.1.5 Service files

Data relevant for service purposes can be saved in a .zip file. Creating a service file is generally recommended should an error occur with the iTNC 530 HSCI. This file contains useful information, especially for the HEIDENHAIN Service department during troubleshooting.

The appropriate data from the NC and PLC are saved in the file **TNC:\service\service<xxxxxxx>.zip**. The name of the file is generated automatically, whereby <xxxxxxx > is the system time shown as an unambiguous character string in hexadecimal code.

You can generate a service file in the following ways:

- Press the ERR key and then the SAVE SERVICE FILES soft key.
- It can be provoked by certain configured errors. Here a "1" must be entered in the SF column (**Create service file**) of the PET table. A service file is generated no more than once per minute.
- Crash of the NC software due to a fatal error
- Request via TNCremoNT

After a service file has been created, an informational text is displayed. This text indicates that the service file also contains your and your customer's NC data (NC programs, etc.). You and your customers must be aware of this when you give the service file to third parties.

The informational text must be confirmed with the OK button or the ENTER key.

When creating the service file via telemaintenance, the informational text is not displayed. You and your customers must be aware that telemaintenance enables third parties to access your data anyway.

The following data (and other information) is saved in the service file:

- Log
- PLC log
- Selected files (*.H/*.I/*.T/*.TCH/*.D) of all operating modes
- *.SYS files
- Machine parameters
- Information and log files of the operating system (can be partially activated via HE Logging settings)
- Contents of PLC memory
- NC macros defined in PLC:\NCMACRO.SYS
- Information about the hardware
- Error outputs and configuration files of the PLC compiler
- Current Feature Content Level (FCL) and active software options, including the option designations through the file PLC:\SIK.INFO
- The changes to machine parameters via the PLC, LSV2 and NC programs through the MPSEVER.TXT file.

Please remember that OEM logs are not automatically recognized and saved by HEIDENHAIN. However, you can store your own

PLC:\oemservicefiles.sys control file, in which you can enter additional files to be saved.

Behavior when saving the HEIDENHAIN logs:

The size of the log data saved is always between 5 MB and 15 MB.

- Log \geq 5 MB
→ Only the current log is saved.
- Log $<$ 5 MB
→ The current and the last archived log are automatically saved.



Note

HEIDENHAIN recommends that you encrypt your passwords in order to protect your passwords in *.SYS files when a service file is generated.

When a service file is generated, the *.SYS files are also saved in the .zip file. As a result, however the passwords in the *.SYS files can be accessed by other persons. If you also want to protect your passwords from unauthorized access when the service file is generated, proceed as follows:

- ▶ Open the *.SYS file in which you have stored passwords (e.g. OEM.SYS).
- ▶ Place the cursor on the line containing a password (e.g. PLCPASSWORD, MPPASSWORD).
- ▶ Press the ENCRYPT PASSWORD soft key.
This encrypts the password and places it after the keyword.

In the default setting, the data are automatically converted to ASCII format when being saved to the .zip file. If however the conversion to ASCII fails, the binary version of the file is saved in the .zip file.



Note

Files that are saved on the encrypted PLC partition **PLCE:** cannot be written to the service file.

You can also store your own control file **PLC:\oemservicefiles.sys**, in which you can enter additional files to be saved.

In addition, your service agency can help the end user save another control file **TNC:\service\userfiles.sys**.



Control files have the following syntax:

■ **FOLDER** <folder> [<fmask>] [-a=<n><u>] [-s=<n><u>] [-b]

Saving all files of a directory. The effect of the command can be influenced by various parameters. The individual parameters must be entered in the described sequence. If more than one criterion is specified, all of the criteria must apply for the file to be saved.

Event	Function
<folder>	Directory that is to be searched for files to be saved. Only the specified directory is searched for files, and not the subdirectories.
<fmask>	Optional parameter as a selection criterion for the file name. The wild cards * can be used for any number of characters, and the character ? can be used for exactly one character. Example: *.TXT, PGM12?.H If no parameter is entered, all files of the directory are saved.
[-a=<n><u>]	Optional parameter for specifying the age of the files to be saved. Files that are older than the specified period of time will not be saved. <n> for the time period: ■ Input of a number <u> for the unit of time: ■ s = seconds ■ m = minutes ■ h = hours ■ d = days Examples: -a=30s, -a=7d, -a=24h
[-s=<n><u>]	Optional parameter for specifying the size of the files to be saved. Files exceeding the specified size will not be saved. The parameter is to prevent that an NC program with 1 GB is saved, for example. <n> for the size: ■ Input of a number <u> for the unit of size: ■ B = byte ■ k = kilobyte ■ m = megabyte Examples: -s=100k, -s=1m
[-b]	Binary files are not converted to ASCII files. All binary files are saved in binary format.



■ **SETUPFILE <file> <token> [-a=<n><u>] [-s=<n><u>] [-b]**

Saving a file whose name is saved in a system file. You can use the same parameters that are used for FOLDER.

Event	Function
<file>	Specify one of the following keywords for the system files that are to be searched for the token: <ul style="list-style-type: none"> ■ OEM (= PLC:\OEM.SYS) ■ NCPATH (= SYS:\NCPATH.SYS) ■ NCMACRO (= PLC:\NCMAKRO.SYS)
<token>	Enter the token under which the file to be saved is entered in the specified system file. Example: SETUPFILE OEM MPFILE, SETUPFILE NCMACRO PALETT

■ **FILE <fname> [-a=<n><u>] [-s=<n><u>] [-b]**

Saving an individual file. You can use the same parameters that are used for FOLDER.

Event	Function
<filename>	Complete path and name of the file to be saved.

■ **INCLUDE <file>**

When a file is specified, it is evaluated as another control file. In order to avoid that a file is specified twice, the INCLUDE statement must not be nested more than three levels.

Event	Function
<file>	Complete path and name of the additional control file.



8.1.6 Help

Help soft key in MOD

With the help file you can display help text, useful information or machine commands:

- ▶ Ensure that a help file of the type *.HLP is defined in the system file OEM.SYS with the **MODEHELP** = command.
- ▶ Press the MOD key.
- ▶ Press the HELP soft key.

You can edit the help file in the PLC editor:

- ▶ Press the MOD key and enter the code number 807667.
- ▶ Create a help file with the extension *.HLP.

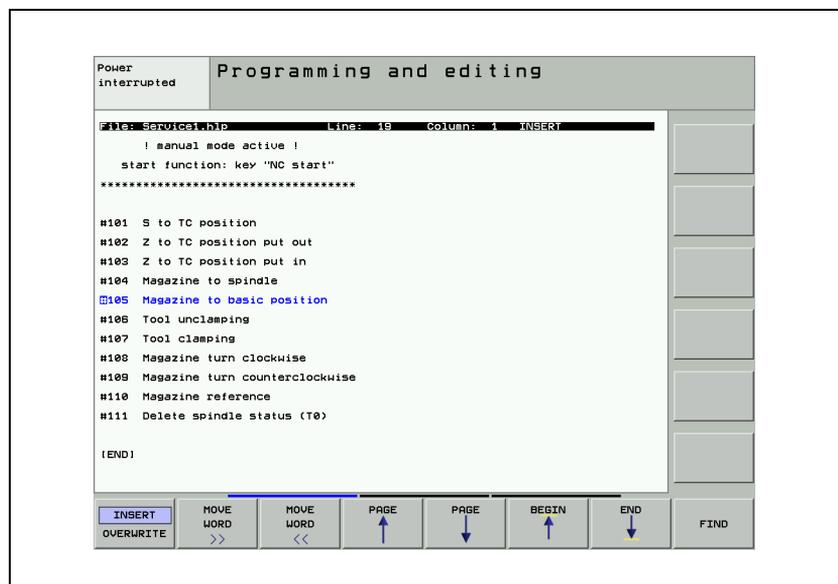
With machine commands:

- ▶ Define at the beginning of the line a numerical value in the format #xxxx.

If the user moves the cursor to a line with a numerical value, this number is displayed in W270. In the PLC program you can interrogate W270 and execute the command.

If the cursor is moved to a line without a valid numerical value, the value -2 is entered in W270.

If no HELP file is selected, W270 contains the value -1.



You can create several help files:

- ▶ Select the conversational language for help files with MP7230.3. The entry behind **MODEHELP** = in OEM.SYS is overwritten with the language-specific path (**PLC:\LANGUAGE\<<Language>**).

The user selects a file:

- By pressing the HELP soft key
- By pressing the PGM MGT soft key

If a HELP file is selected in the foreground and background operating mode, the error message **PARALLEL OPERATION NOT POSSIBLE** appears.

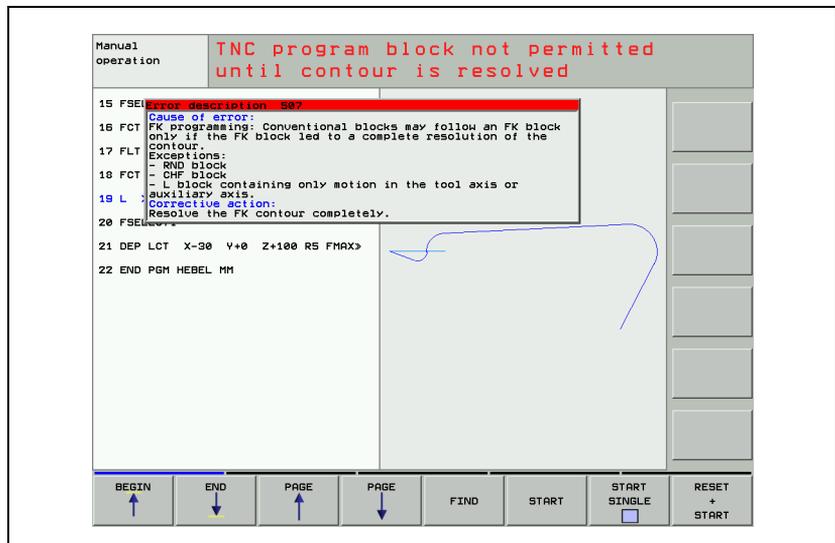
HELP files are saved externally with the identifier J.

	Set	Reset
W270	Line number in help file	NC
	-1: No help file selected	NC
	-2: Not a valid numerical value	
	0 to 9999: Line number	

Help window with HELP key

If an NC error message appears:

- ▶ Press the HELP key.
You will receive information on the cause and correction of the error.



To make such information available for PLC error messages as well, you must save the texts in files:

- ▶ Create two files:
 - Texts under the heading "Cause of Error" in **REASON.A**, for example.
 - Texts under the heading "Corrective Action" in **FIX.A**, for example.
- ▶ Define the names of both files in the system file OEM.SYS with keywords:
 - Cause of error: **PLCERRREASON = REASON.A** (for example)
 - Corrective action: **PLCERRFIX = FIX.A** (for example)
- ▶ Save the files in the corresponding language directories (**PLC:\LANGUAGE\<1 language>**).
- ▶ With MP7230.3, select the active language (file).

The files are divided into text blocks. Each text block can contain up to 10 lines, each with 60 characters. It is ended with <FF>. On the iTNC you can enter an <FF> with the key combination SHIFT + RET. Through the error number (line number) in the PET table, the TNC finds the associated text block in the "error correction" and "error cause" files. Error number zero is the first text block. The error number is also shown in the heading of the help window.

With the PLC programming software PLCdesign, you can generate a PET table and text files, and then transfer them to the iTNC.

In the PLC you can use markers M4220 to M4223 to interrogate the activity of an error from the PET table.

		Set	Reset
M4220	Error from PET table with F stop active	NC	NC
M4221	Error from PET table with NC stop active	NC	NC
M4222	Error from PET table with emergency stop active	NC	NC
M4223	Error from PET table with NC Cancel active	NC	NC



8.1.7 PLC pop-up window

The PLC pop-up (i.e. superimposed) window is shown in the following operating modes:

- **Manual Operation**
- **Positioning with Manual Data Input**
- **Program Run, Single Block**
- **Program Run, Full Sequence**

▶ Activate a pop-up window with Modules 9215 or 9217

If you transfer file names without paths, the iTNC looks for the file in the language-specific directory PLC:\LANGUAGE\

▶ With MP7230.3, select the active language (file).

When the PLC pop-up window is called, other pop-up windows (such as the help window) are moved to the background and become active again after the PLC pop-up window has been closed. The PLC pop-up window can be displaced to the background by another pop-up window.

The modules do not return until the pop-up window is closed. They must be called in a spawn job, not in a submit job, because otherwise the subsequent submit jobs will not be run until the pop-up window is closed.

A pop-up window called with Module 9215 or 9217 can be ended with Module 9261.



Module 9215 Activate a PLC pop-up window

The module uses the following events:

- \$01 000 000 Window build-up
- \$00 010 000 Closing the pop-up window

The event for window build-up is generated internally and must not be started externally. If the event for closing the pop-up window is transferred, the module closes without waiting for keyboard input.

The user can make his selections from a list by using the cursor keys and the ENTER key, or the hot keys. The module then returns the line number of the selected menu item. (Line 1 = number 0). You separate the individual entries with <LF>.

Call only in a submit job.

Call:

PS B/W/D/K/S	<String with window heading>
PS B/W/D/K/S	<String with file name of the list> [complete path or only file name]
PS B/W/D/K	<Start position> [line in which the highlight is located]
PS B/W/D/K	<Mode> Bit 0/Bit 1 – Character size 00: Automatic 01: Small characters 10: Medium characters 11: Large characters Bit 2 – Frame 0: With frame 1: Without frame Bit 3 – Heading 0: Display heading 1: Do not display heading Bit 4 – Hot keys 0: No hot keys 1: Hot keys (0 to 9 and A to Z) before menu items Bit 5 – Vertical / Horizontal 0: Vertical arrangement 1: Horizontal arrangement (bit 4 = 0)
CM 9215	
PL B/W/D	<Selected line> 0 to n: Line number from list -1: No selection made (END, NOENT) 2: For error see W1022

Error code:

Marker	Value	Meaning
M1022	0	Incorrect mode transferred
	3	Not a valid string for file name or heading
	6	Window cannot be displayed (internal error, e.g. problems with memory or operating system)
	20	Module was not called in a spawn job
	28	Another PLC pop-up window is open
	36	File with the list could not be found

Module 9217 Display pop-up window for messages

Module 9217 can be used in the Machine operating modes to display a pop-up window with the text of a file (max. 16 KB) and various soft-key rows. The pressed key or soft key is reported. Starting at a certain number of lines (up to 500 lines possible), a scroll bar is displayed automatically. Use the arrow keys or the PgUp and PgDn keys to scroll through the pop-up window. The keyboard is assigned to the pop-up window. The width of the window is determined by the longest line.

A displayed window can be closed via Module 9261 (sending of events) through event \$010000.

Call:

```

PS      S          <Window title>
PS      S          <ASCII file with message text>
PS      B/W/D/K   <Mode>
                        0: No soft-key row; clear window with CE
                        1: Soft-key row with OK soft key
                        2: Soft-key row with YES and NO soft keys
                        3: Soft-key row with YES, NO and END soft keys

CM      9217
PL      B/W/D     <Pressed key or soft key>
                        -1: Error
                        0: CE key
                        1: OK soft key
                        2: YES soft key
                        3: NO soft key
                        4: END soft key

```

Error code:

Marker	Value	Meaning
M4203	0	Pop-up window displayed
	1	Error code in W1022
W1022	1	Invalid mode
	6	No connection to display server
	11	Invalid string for title
	20	Module was not called in a submit or spawn job
	28	Pop-up window already active
	36	.ASCII file with message text does not exist



8.1.8 TNCguide – context-sensitive help system for the iTNC 530 (user documentation)

The help system was integrated on the basis of the CHM help format known from Windows. The CHM format was introduced by Microsoft in 1997 with the HTML help system, and is now used by many Windows programs. It is a collection of individual HTML files that are collected in a single compressed file.

Mozilla Firefox is used as the browser on the iTNC 530.

In principle, the iTNC 530 will be able to open and display any CHM files, including those prepared by the OEM. HEIDENHAIN will generally supply the following documentation in the form of CHM files together with the iTNC software:

- Conversational Programming User's Manual
- Touch Probe Cycles User's Manual
- smarT.NC Pilot
- Collection of all NC error messages

The User's Manual for the programming station is also supplied with the iTNC 530 programming station. Context-sensitive entry points are defined for these files, meaning that pressing the HELP key brings the user directly to the appropriate location in the documentation. If no context-sensitive entry point is available, the TNC opens the parent **main.chm** book file, in which all CHM files in the respective (language-sensitive) help directory are shown. The user navigates to the desired entry with the mouse or the arrow keys.

TNCguide can also be opened with the mouse: after clicking the help symbol, shown to the right directly above the soft-key row, the mouse pointer changes to a question mark. If the user then clicks a soft key, the TNC starts the help system and displays the description of this soft key, if a context-sensitive entry point exists. If no context-sensitive entry point is available, the control opens the parent **main.chm** book file, as described above.



Basic conditions for the OEM

So that CHM files created by OEMs can be shown in the **main.chm** book file, the conditions listed below must be followed:

- The OEMx.CHM files must be stored in the **TNC:\tncguide\de** directory, or in **TNC:\tncguide\en**, etc. When the dialog language is switched (with MP7230), the iTNC searches the corresponding language subdirectory when the help system is called.
HEIDENHAIN recommends placing an OEMx.CHM file in English in each language subdirectory if you do not translate your documentation into every language. This ensures that online help is available for all topics, regardless of the language settings on the control (MP7230). If no OEMx.CHM file exists in the language subdirectory, no OEM-specific help is shown when the online help is called.
- HEIDENHAIN has already defined the names for the CHM files created by the OEM, so that these files can be displayed as books (if they exist) within the parent **main.chm** book file:

Name of CHM file	Help-number range from	to
OEM1.chm	10 000 000	10 999 999
OEM2.chm	11 000 000	11 999 999
OEM3.chm	12 000 000	12 999 999
OEM4.chm	13 000 000	13 999 999
OEM5.chm	14 000 000	14 999 999
OEM6.chm	15 000 000	15 999 999
OEM7.chm	16 000 000	16 999 999
OEM8.chm	17 000 000	17 999 999
OEM9.chm	18 000 000	18 999 999
OEM10.chm	19 000 000	19 999 999

- The help-number range shows the context-sensitive entry points that are permanently defined for each file in order to simplify entry via the parent main.chm book file.
- The following links contain useful information about HTML help as well as software for downloading:
<http://msdn.microsoft.com/library/en-us/htmlhelp/html/vsconHH1Start.asp>
<http://www.helpware.net/>
- HEIDENHAIN recommends using Mozilla Firefox 1.0.x to check how the HTML pages are displayed, since it is used on the control. The view differs somewhat from the view in Internet Explorer, especially regarding the page layout. However, this does not replace a thorough test of the CHM file on the control.



Using PLC modules to call CHM files created by the OEM:

- PLC Module 9391 is introduced, with which a PLC error message is displayed, and in addition an offset is added to the value of the error number (= ONL Number) in the .PET table in order to generate the actual help number. This way a group error number can be defined for an (OEM) device in the .PET table, and the error number (used as an offset) supplied by the device in case of error then leads to the appropriate help text. This requires an OEM-specific *.CHM file, which must be indicated in the .PET table (= ONL Name).
- Additionally, PLC Module 9390 is also introduced, with which a help window can be opened directly by the PLC.

Every possible way of calling your OEM help file offers you the possibility of showing the entire directory (including HEIDENHAIN help files) in the directory tree, or just the directory of your OEM help file. This selection is made when calling the help file. If you enter **main.chm** as the help file in the call (via PLC module, *.PET file, soft key or NC error), then the entire directory is shown. If you enter your OEM help file OEMx.CHM as help file in the call, then only the directory of your help file is shown.

Files and structure of the help system

All online help available on a control can be called separately as well as within the online help system. In order for the call within the entire system to function, the conditions described below must be followed:

The help system is structured as follows:

- main.chm – Welcome page of the help system
 - jh1.chm
 - jh2.chm
 - ...
 - oem1.chm
 - oem2.chm
 - oem3.chm
 - ...

The HTML pages of the individual online help topics exist completely independently of each other. Special entries combine the tables of contents and collate the index entries (see below).

Table of contents

The table of contents of main.chm contains "merge" objects:

```
<OBJECT type="text/sitemap">  
  <param name="Merge" value="oem1.chm:/merged.hhc">  
</OBJECT>
```

HEIDENHAIN has specified here the name of the CHM file and the file name **merged.hhc** of the table of contents contained therein. When the welcome page of the help system (main.chm) is called, all existing help files linked with merge commands are included. Each table of contents is only displayed if it exists in a file named **merged.hhc**. This is why it is essential that you include the directory structure of your help as **merged.hhc** when you generate your **OEMx.CHM** files.

The first level of this table of contents should have exactly one entry: the title of the help file. This title then appears as a "book" in the entire table of contents, and can be opened by the user in order to show the subordinate headings.

Along with the actual HTML pages, the following files must be included when generating the OEMx.CHM files:

- *.hhc file
In this file you describe the structure and format of your help system. If you call your help file via main.chm, then this file must be named **merged.hhc**. If your help file is to be called directly, without **main.chm**, then the directory tree in the *.hhc file that you indicated as content file when you created the CHM file is used.
- *.hhk file
In this file you list all entries that are to be shown in the index later, and create the links to the corresponding HTML pages. At least one entry is necessary here in order for your help file to be displayed.
- *.hhp file
This file is the project file that is necessary for generating an OEMx.CHM file.
- *.h file (only necessary for context-sensitive help)
In this file you use the **#define** command to establish connections between error numbers, from the respectively valid ranges of error numbers, and any variables.
Example:
#define IDH_OEM1_CHAP110000000
#define IDH_OEM1_Page1_110000100
- *.txt file (only necessary for context-sensitive help)
In this file you establish the connection between the variables and the corresponding HTML pages, which are then called.
Example:
IDH_OEM1_CHAP1=chapter1.htm
IDH_OEM1_Page1_1=page11.htm



INDEX

Note: "Merging" of the indexes only functions if each file involved contains at least one index entry, meaning at least one index entry is also necessary for the OEM help file. In addition, the entry "Binary Index=Yes" must be set in the project file (*.hhp).

The project file of main.chm contains the following entries:

■ [MERGE FILES]

- jh1.chm
- jh2.chm
- ...
- oem1.chm
- oem2.chm
- oem3.chm
- ...

This collates and displays the index entries of all present and named help files when the index of main.chm is called.

Context-sensitive call

In a context-sensitive call of the OEM help, the index display starts from the OEM help, and here the index entries can only be collated if all other help files *except* the current OEM file itself are entered in the project file of the OEM help.

HEIDENHAIN makes a complete list of the file names available.

Summary for OEM help

- Project file: Binary Index=Yes
- Project file: [MERGE FILES] with current list *except* its own file name!
- Table of contents: "merged.hhc" – Title of the help, can be opened as a "book."
- Index: At least one index entry.

Designing soft keys for context- sensitive help

In order to establish context sensitivity in combination with soft keys, the additional HELPID and HELPFILe soft keys must be entered in the descriptions of the soft keys. Use HELPID to assign to a soft key an ID (= error number) from the respectively valid range of error numbers. With the HELPFILe attribute you indicate in which *.chm file the error number can be found (e.g. OEM1.chm) or via which file the error number is to be searched for (e.g. main.chm). The assigned error number, in combination with the *.txt and *.h files, is used to call the appropriate HTML page when the help is called. If you enter main.chm as the HELPFILe, the entire directory structure with the HEIDENHAIN help files is shown. If you enter only one file (e.g. OEM1.CHM), then only the directory of this file is shown. The OEMx.CHM files must be stored in the **TNC:\tncguide\de** directory, or in **TNC:\tncguide\en**, etc. When the dialog language is switched (with MP7230), the iTNC searches the corresponding language subdirectory when the help system is called. You can enter HELPFILe for each soft key, or once for all soft keys.

Example:

PLC SOFTKEY Project File - Version 1.0

```
; Path for the soft-key help file  
HELPFILe 'TNC:\tncguide\de\oem1.chm'
```

...

```
; here the assignment of a HELPID to the soft keys  
; without indication of a *.chm file. This automatically links to  
; the help file indicated above.
```

```
ACTION Action2_Softkey HELPID:10000000
```

```
PULSE Pulse1_Softkey HELPID:10000100
```

```
BLANK  
ENDSKMENU
```

or:

PLC SOFTKEY Project File - Version 1.0

...

```
; here the assignment of a HELPID and a HELPFILe to  
; the soft keys. This automatically links to the  
; indicated help file.
```

```
ACTION Action2_Softkey HELPID:10000000
```

```
HELPFILe:TNC:\tncguide\de\oem1.chm
```

```
PULSE Pulse1_Softkey HELPID:10000100
```

```
HELPFILe:TNC:\tncguide\de\oem1.chm
```

```
BLANK  
ENDSKMENU
```



Enhanced error notification

- PLC error messages
For each entry (i.e. error message) in the PET file, the machine manufacturer can enter the name of a help file (*.CHM) and a help number within this *.CHM file. This is done with the two new columns in the PET table, ONL Name and ONL Number. The called help files must—as mentioned earlier—be present and language-sensitive in the TNC:\tncguide\de, TNC:\tncguide\en, etc. directories. When a PLC error message is current and the user then presses the ERR key and the HEIDENHAIN TNCguide soft key, the appropriate chapter from this file is shown (context-sensitive call). If a help number but no *.CHM file is indicated, the main.chm file is automatically shown. If the OEM has adhered to the permanently defined help-number ranges for the respective files (see above for the help-number ranges), then the correct, context-sensitive help page is shown in this case as well.
- NC error messages
The machine manufacturer has the possibility of calling his own additional error descriptions for NC errors. This is also done with a separate CHM file. With the OEMERRORCHM.FILE and OEMERRORCHM.OFFSET entries in OEM.SYS, context-sensitive links to any page in the OEM file can be created. Under OEMERRORCHM.FILE you must enter the *.CHM file in which the error number is to be searched. If main.chm is entered under OEMERRORCHM.FILE, the error number is searched for from there and the entire directory tree of **main.chm** is displayed. If the error number is not found, the start page of main.chm is shown. Under OEMERRORCHM.OFFSET you must enter an error number that is used as an offset. The number of the NC error is added to the number entered here. The resulting error number is then searched for.
Example:
OEMERRORCHM.FILE = **main.chm**
OEMERRORCHM.OFFSET = **10000000**
If, for example, the NC error message **Key non-functional** is shown (NC error number 938), the resulting error number = 10000938.
If you press the ERR key now, two soft keys are offered.
If you press the HEIDENHAIN TNCguide soft key, the HEIDENHAIN error description appears. If you press the MACHINE MANUFACTURER soft key, the resulting error number (e.g. 10000938) is searched for in the given CHM file. The appropriate page is opened if the error number exists, otherwise the start page of the given file opens.
The help files to be called must—as mentioned earlier—be present and language-sensitive in the TNC:\tncguide\de, TNC:\tncguide\en, etc. directories.
The MACHINE MANUFACTURER soft key is only displayed if one of the two entries named exists in OEM.SYS.

Including an OEM-specific online help file

Once you have created a valid *.chm file, proceed as follows in order to display your own OEM-specific help file in the HEIDENHAIN TNCguide:

- ▶ You may need to rename your *.chm file. You must use one of the names reserved by HEIDENHAIN for OEM help files.
e.g. **OEM1.CHM**
- ▶ Use TNCremo to transfer the help file to the control.
- ▶ Store your help file in the appropriate language directory: **TNC:\tncguide\de**, **TNC:\tncguide\en**, etc. If you have created only an English help file, HEIDENHAIN recommends placing it in the other language directories as well.
- ▶ Press the HELP key to call the TNCguide. Your help file should now automatically be included in the TNCguide directory tree.

National languages

CHM files will not be available for all possible TNC dialog languages at the time the NC software is released. We are planning on delivering the German and English languages with the software, and the CHM files in other languages will be made available for free downloading (for unregistered users as well, of course) from HESIS-Web including Filebase. The user then simply downloads the appropriate file(s) for the respective language(s), and stores them in the directory defined by us on the TNC hard disk: **TNC:\tncguide\de** or the appropriate language subdirectory.

A **readme.a** file is located in the respective language directories on the TNC hard disk under **TNC:\tncguide**. This file describes the procedure for loading the CHM files from HESIS-Web including Filebase.

Online help files and TNCremoNT

The online help files (with file name extension .chm) can be managed with TNCremoNT:

■ Transfer of *.chm files:

Online help files are binary files. If TNCremoNT is updated to version 2.5, the file extension .chm is automatically added to the list of binary file types. Otherwise the list of binary file types must be amended manually under Extras > Configuration on the Mode tab in order to transfer them correctly.

■ Performing a backup of the TNC via TNCremoNT:

Online help files are automatically untagged during creation of the scan list used for the backup. This also applies to CHM files that the machine manufacturer has saved on the control.

Reason:

The *.chm files saved on the control require a large amount of memory, and do not need to be backed up, since they are freely available from the HEIDENHAIN homepage.

Please note that only online help files from HEIDENHAIN are available here. This setting can be applied to other file types as well in TNCbackup under Edit > Settings, if necessary.

Module 9390 Open the online help window with the control's browser

With Module 9390 you open an online help window with the control's browser. The module can be called from a cyclic PLC program or from a spawn job or submit job. The extension .CHM can be given in the call, but is not necessary.

Constraints:

- The .CHM help file must be stored language-sensitive in the **TNC:\tncguide\de** directory, or in **TNC:\tncguide\en** etc.
- If an empty string is given as file name, the **main.chm** file is used. Depending on the link in **main.chm** the context number then branches to the **OEMx.CHM** specified there.
- If 0 was given as the help number, or if the given number could not be found in the help system, the start page of the given help file is opened.
- If the help system could not be started (e.g. the .CHM file is missing), an NC error message with a reference to this PLC module is generated.

Call:

PS SXX <File name>
 from string number S0/S1/... or S "file" or S"
 PS K/B/W/D <Help number of the HTML page to be opened from the
 respectively valid help-number range>
 CM 9390

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Invalid parameter or parameter does not exist
	22	Help system could not be started



Module 9391 Display an error message with additional offset

With Module 9391 you can display with an additional offset a PLC error message from the .PET table. Module 9391 corresponds in behavior to Module 9085. The module can be called from a cyclic PLC program or from a spawn job or submit job.

An offset is added to the value for the help number in the .PET file in order to generate the actual help number. This way a group error number can be defined for an (OEM) device in the .PET table. The error number (used as an offset) supplied by the device in case of error then leads to the appropriate help text.

Two new columns were added to the .PET table for this. In the column ONL-Name you enter the file name of the help file that is to be called additionally in connection with an offset. In the ONL-Number column you enter the ID (=help number) of the HTML page in the help file to be opened. The offset is added to this ID (=help number) during the module call, and is given by the module. This makes it possible to refer to specific HTML pages in addition to the more general PLC error message.

Constraints:

- The .CHM help file must be stored language-sensitive in the **TNC:\tncguide\de** directory, or in **TNC:\tncguide\en** etc.
- An OEM-specific **OEMx.CHM** file is necessary.
- If no **OEMx.CHM** file is indicated, the online help is not called. All other reactions that are saved for this error in the .PET table are performed.

Call:

PS K/B/W/D <Line number in the .PET error table>

PS K/B/W/D <Offset for the help number in .PET>

CM 9391

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Invalid parameter or parameter does not exist



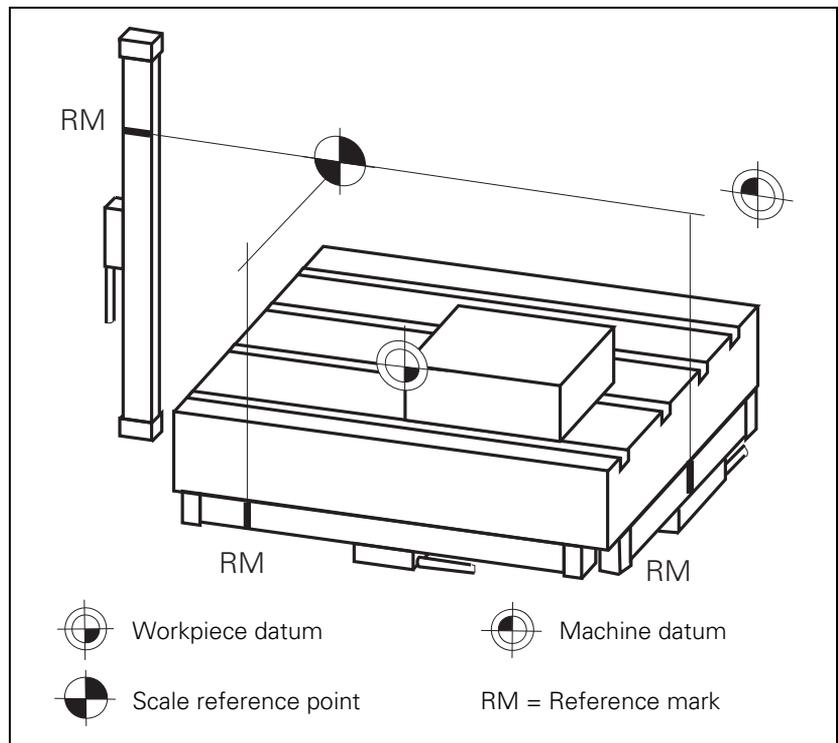
8.1.9 Machine datum

The machine is built with a fixed **machine datum**. All referenced displays and positioning blocks refer to this machine datum.

In the **Manual** and **Electronic Handwheel** operating modes you can define a **workpiece datum** with the "datum setting" function. NC programming blocks are entered with respect to this defined datum.

Other datums:

- If the user programs M91 in an NC program, the NC programming block refers to the machine datum.
- ▶ In MP960.x, enter the distance between the machine datum and the scale reference point.
- All NC programming blocks are referenced to the machine datum.
- ▶ With MP7295, disable the "datum setting" function for specific axes.
- NC programming block values are defined with respect to fixed positions of the machine if the user programs M92 in the NC program.
- ▶ In MP950.x, enter the distance between the machine datum and the machine-referenced position. The values for MP950.x can be assumed with the "actual position capture" key.



Note

M91 and M92 are active only in the block in which they are programmed.

Datum setting

The datum can be set either by only the "Datum setting" soft key or by the soft key plus the axis keys:

- ▶ With MP7296, define how the datum should be set.

To change the datum in the OEM cycles:

- ▶ Press the MOD key.
- ▶ Enter the code number 555 343.
- ▶ Enter: **FN25: PRESET <axis>/<value to be converted>/<new datum>**.

<Axis>	Axis for which the datum is to be set
<Value to be converted>	Coordinate in the active coordinate system to which the datum is to be set, or the number of the Q parameter that contains this coordinate
<New datum>	Desired value of the datum



Note

The currently effective datum, but not the corresponding value in the preset table, is overwritten with **FN25: PRESET**.

If two axes are being operated paraxially, for example with two alternating spindle motors, you can specify that the datum being set in one axis with "Datum setting" should also apply to the other axis.

- ▶ Enter in MP7492.x the number of the axis that is to also use the same datum. The index of MP7492.x defines the axis whose datum is to be applied.

Example:

Axes 0 and 1 are parallel axes. When setting the datum for Axis 0, the datum is also to apply to Axis 1. In this case you must set MP7492.0 = 1.

MP950.x Datum for positioning blocks with M92
Input: -99 999.9999 to +99 999.9999 [mm] or [°]
Values with respect to the machine datum

MP960.x Machine datum
Input: -1.79769313486E+308 to
+1.79769313486E+308 [mm] or [°]
Values with respect to the scale reference point

MP7295 Disable "Datum setting"
Format: %xxxxxxxxxxxxxxxxxxxxxx
Input: Bits 0 to 17 represent axes 1 to 18
0: Not disabled
1: Disabled

MP7296 "Datum setting" through axis keys
Input: 0: Datum can be set by axis keys and soft key
1: Datum can be set only by soft key

MP7492.x Number of axis in which the same datum is to be set during datum setting (with active preset table)
Input: 0 to 17
-1: Do not set a datum
MP7492.0 Datum set in the first axis
to
MP7492.17 Datum set in the 18th axis



Datum management via preset tables

Presets are managed with the TNC:\PRESET.PR preset table.

- ▶ With MP7294, disable the "datum setting" function in the preset table for specific axes.

Each traverse range uses its own preset table. The preset table of the current traverse range can be viewed in the **Manual** operating mode. **FN17: SYSWRITE** and **FN18: SYSREAD** refer to the active preset table.

FN17: SYSWRITE ID503 is used for entering a value directly (i.e. without conversion) into the preset table. This way a defined state can be set as a default value in the preset table.

FN17: SYSWRITE ID502 and **FN18: SYSREAD ID502** are used for entering a value with conversion into the active coordinate system or preset table, or for reading it.



Note

The currently effective datum, but not the corresponding value in the preset table, is overwritten with **FN25: PRESET**.

The **PR.LINESLOCKED =** entry in OEM.SYS is used to write-protect lines in the preset table, such as for the machine datum. Separate each line number with a comma, and connect line ranges with a dash: **PR.LINESLOCKED = 1,4-8,22**. Write-protection can only be assigned to the first 255 lines. Write-protected lines are shown in a different color. The active line and line 0 are always write-protected. The manually set datum, which was set by the user with the axis keys, is automatically written into line 0 of the preset table.

When the control is started up, new cycle data is generated (with or without support from the preset table).

It is also possible to take the presence or absence of the preset table into account in OEM cycles. See page 1704.

MP7268.x is used to hide the columns of the preset table, or to sort them as desired. MP7268.x is used to define the sequence of the columns. If MP7268.x = 0, the respective column is not displayed and is not saved during external data backup.

The values in preset tables are shown in inches if the control was set to inches. When the files are output, the data will be saved without unit, i.e. converted files must always be interpreted in mm format.



You can write-protect lines of the preset table. A write protection for individual lines or entire areas in the preset table can now be set by an entry of **PR. LINESLOCKED** = (e.g. **PR. LINESLOCKED = 1,4-8,22**). Write-protection can be activated only for the first 255 lines of the preset table. Protected lines are indicated in the editor by a different color.

		Set	Reset
M4589	Activate datum management via preset table	NC	NC
MP7294	Disable axis-specific "Datum setting" in the preset table		
Format:	%xxxxxxxxxxxxxxxxxxxx		
Input:	Bits 0 to 17 represent axes 1 to 18 0: Not disabled 1: Disabled		
MP7268	Sorting/hiding columns of the preset table		
Input:	1 to 99 = Order of the columns 0 = Do not display the column		



8.1.10 NC program

Setting "look-ahead"

In order to adapt the feed rate to the workpiece machining process, the iTNC 530 precalculates the geometry. This way changes in directions (corners, curvatures, and changes in curvatures) are detected in time, and the participating NC axes can be braked or accelerated accordingly.

You can set the number of NC blocks to be used for precalculation. The greater the number of blocks for advance calculation, the higher the possible feed rate. Shorter machining times can be achieved.

This improved effect is especially noticeable with many short traverse blocks in the micron range, which are exported from CAD/CAM files, for example. However, the default setting of 256 blocks usually suffices.

MP7400 **Setting look-ahead**

Input: 0: 256 [blocks] (default)
 1: 512 [blocks]
 2: 1024 [blocks]

Block number increment for DIN/ISO programs

► In MP7220 enter the block number increment for DIN/ISO programs.

MP7220 **Block number increment for DIN/ISO programs**

Input: 0 to 250

Finding the block number

Module 9321 Find the current block number

Please use Module 9322 if possible!

The current block number is ascertained with Module 9321.

Call:

PS B/W/D/K <String number (reserved for future applications)>
CM 9321
PL B/W/D <Current block number>

Error code:

Marker	Value	Meaning
M4203	0	Block number has been found
	1	Error

Module 9322 Information of the current NC program

With Module 9322, you can determine the current block number of the active NC program. If the module is called from the cyclic PLC program, only the block number of the current NC main program is read in real time. If the module is called from a spawn job or submit job, the path of the current NC program is determined in addition to the block number (from the block scan).

Constraints:

- Because of the geometry look-ahead, the call from the cyclic PLC program only supplies the block number in real time, but no information about the NC program.
- Call from the cyclic PLC program:
For all traverse blocks that are not generated from an NC program, a cycle or an NC macro, block number -1 is read.
- Call from the cyclic PLC program:
After the NC program is canceled or after the end of the NC program, the last block number executed is returned.

Call:

PS B/W/D/K <Mode>

When called from a cyclic PLC program, the setting of <Mode> is without effect. The block number of the active NC program is always returned.

When called from a spawn job or submit job:

0: String / block number and path refer only to the active NC (sub)program. Block number from block scan.

1: String / block number and path refer to the active NC (sub)program or cycle. Block number from block scan.

2: Only the name of the NC main program without information about the block number (block number is set to 0 when executed correctly).

PS B/W/D/K <String number for path of the NC program or cycle>

Call from a cyclic PLC program: Without effect.

CM 9322

PL B/W/D <Block number of current NC program or cycle>

-1: Error if error marker is set

-1: Block number in certain cases, if call was from a cyclic PLC program

Error code:

Marker	Value	Meaning
M4203	0	Information of current NC program has been read
	1	Error code in W1022
W1022	1	Invalid mode
	2	Invalid string number

NC program selected

With marker M4181 it is possible to interrogate whether an NC program is selected in the **Program Run, Full Sequence** or **Program Run, Single Block** mode of operation. The marker is not set if an NC program is selected from a pallet table.

		Set	Reset
M4181	NC program selected	NC	PLC

Display of the NC program

The NC program can be displayed in various layouts:

- ▶ Set MP7281 = 0 to show all blocks completely.
- ▶ Set MP7281 = 1 to show only the current block completely. All other blocks are shown as only one line.
- ▶ Set MP7281 = 2 to show all blocks as one line. A block is shown completely only when it is created or edited.

MP7281 Depiction of the NC program

Input:

- 0: All blocks completely
- 1: Current block completely, others line by line
- 2: All blocks line by line; complete block when editing

Deleting lines from an NC program

Individual blocks of an NC program can be deleted without a confirmation question.

- ▶ Set MP7246 bit 1=1 to have to the confirmation question **Really delete NC block?** and the soft keys YES, NO and END appear when the DEL key is pressed.

MP7246 Machine parameter with multiple function

Input:

- Bit 1 – Clear with DEL key
- 0: Does not need confirmation
- 1: Must confirm via soft key

Checking the NC program during editing

The control checks an NC program while editing it. The duration depends on the number of lines to be checked. Errors that occur after the defined line number are not recognized during editing.

- ▶ Enter in MP7229.0 the line number to which the NC program is to be checked (LBL and TOOL DEF blocks).
- ▶ Enter in MP7229.1 the line number to which FK blocks are permitted. If FK blocks do not appear until after this line number, they are not checked.

MP7229 Properties of the NC program

MP7229.0 Line number for program testing
Input: 100 to 9999
MP7229.1 Program length to which FK blocks are allowed
Input: 100 to 9999



**Status information
about the end of an
NC program**

Module 9320 Status of the NC program end

Module 9320 can ascertain status information on the termination of the NC program.

Call:

CM 9320

PL B/W/D <Cause of NC program end>
 1: Emergency stop
 2: Positioning error
 3: Programmed stop (stop, M00)
 4: Normal end
 5: Geometry error
 6: END PGM, M02
 7: Internal stop
 8: RS-232-C transmission error

PL B/W/D <Error class>
 Bit 0: Reserved
 Bit 1: Control loop
 Bit 2: Probing
 Bit 3: Limit switch
 Bit 4: Error from FN14
 Bit 5: Tool management
 Bit 6: Programming error
 Bit 7: Program selection/preparation
 Bit 8: Pallet administration
 Bit 9: Emergency stop

PL B/W/D <Help number>
 -1: No help number

PL B/W/D <Block number in the NC program>

PL B/W/D <Additional information>
 In the "tool management" error class:
 Tool number

		Set	Reset
M4185	Internal stop performed	NC	PLC



Access to data from the NC program

The NC blocks DATA ACCESS READ and DATA ACCESS WRITE enable you to access specific NC or PLC data from within the NC program. The same interface is used as for data access with Python. The subscription to and unsubscription from data take place automatically. The access rights are determined by the location where the executing NC program is stored (TNC:\, PLC:\).

Example:

```
0 BEGIN PGM DATA_RW MM
1 ;Read tool name of T2 from TOOL.T
2 DATA READ QS8 = "\TABLE\TOOL\T\2\NAME"
3 ;Tool radius of T2 from TOOL.T
4 DATA READ Q8 = "\TABLE\TOOL\T\2\R"
5 ;Write tool name of T5 in TOOL.T
6 DATA WRITE "\TABLE\TOOL\T\5\NAME" = QS8
7 ;Write tool radius of T5 in TOOL.T
8 DATA WRITE "\TABLE\TOOL\T\5\R" = Q8
9 ;Coordinate from line 5 of the KINEMATO.TAB
10 DATA READ QL1 = "\TABLE\'TNC:\KINEMATO.TAB'\NR\5\COORD"
11 ;File name from line 1 of the KINEMATO.TAB
12 DATA READ QS15 = "\TABLE\'TNC:\KINEMATO.TAB'\NR\1\FILE"
13 ;Write coordinate to line 7 of the KINEMATO.TAB
14 DATA WRITE "\TABLE\'TNC:\KINEMATO.TAB'\NR\7\COORD" = QL1
15 ;Write file name to line 2 of the KINEMATO.TAB
16 DATA WRITE "\TABLE\'TNC:\KINEMATO.TAB'\NR\2\FILE" = QS15
17 ;Read PLC word 4000
18 DATA READ QR1 = "\PLC\memory\W\4000"
19 ;Write to PLC word 4002
20 DATA WRITE "\PLC\memory\W\4002" = QR1
21 END PGM DATA_RW MM
```

Canceling an NC program

An NC macro can be called automatically if an NC program was canceled by an error message or an external or internal stop. You can use it to exchange information between the NC and the PLC. This NC macro may not contain any positioning commands, or the error message **Program data erroneous** will appear.

- ▶ In NCMACRO.SYS enter the name (and path) of the NC macro after the code word **RUNCANCEL =**.



Automated NC program start

NC programs and pallet tables can be started by the iTNC automatically at a date and time set by the user. To use the Autostart function:

- ▶ Use MP7683 bit 5 to show the AUTOSTART soft key.
- ▶ With MP7683 bit 7, specify whether the NC program should be started by the NC or the PLC after the expiration of the programmed time. The two following markers can be used if you want the program to be started by the PLC:
 - M4182 indicates whether the AUTOSTART function was activated.
 - M4183 indicates whether the time programmed by the user has expired.
- ▶ Switch to **Program Run, Full Sequence** mode and use **PGM MGT** to activate the NC program or pallet table to be processed. No current error messages are allowed.
- ▶ Press the AUTOSTART soft key.
- ▶ Enter the date and time at which the machine is to be switched on.
- ▶ Set M4586 for the PLC to enable the Autostart function. If the PLC does not enable the function, the error message **Autostart not enabled** appears.
- ▶ Activate the autostart function with the AUTOSTART ON soft key. **Active** blinks in the window.

		Set	Reset
M4182	AUTOSTART active	NC	NC
M4183	Time from AUTOSTART expired	NC	NC
M4586	Enable AUTOSTART	PLC	NC/PLC

MP7683 Executing pallet tables

Input:

Bit 5 – AUTOSTART soft key

0: Do not display soft key

1: Display soft key

Bit7 – AUTOSTART function by PLC

0: AUTOSTART function performed by the NC

1: AUTOSTART function is performed by the PLC. The NC does not trigger an NC start.

Sorted machining of block elements

The DIN/ISO editor of the iTNC 530 does not automatically sort block elements. This means that the block elements TOOL CALL and S in DIN/ISO blocks are machined at the location they were programmed at. If in NC programs (such as from a postprocessor) these block elements are not programmed until the end of the block, then they won't be machined until the end of the block.

- ▶ Set MP7682 bit 7 for the block elements TOOL CALL and S in DIN/ISO blocks to automatically be machined at the beginning of the block. The display of the block elements does not change.

MP7682 Machine parameter with multiple function

Input: Bit 7 – Block elements TOOL CALL and S in DIN/ISO blocks
0: Machine as programmed
1: Machine at beginning of block (block display does not change)

Behavior during program interruption

If an NC program is interrupted and subsequently an axis is moved, then before continuing the NC program the **RESTORE POSITION** function must be executed in order to return the axes to the positions current after the last processed NC block. This function is automatically activated by the TNC.

- ▶ Set MP7680 bit 13 so that the **RESTORE POSITION** function will **not** be activated automatically.

If the spindle is stopped by an NC stop and an internal stop, and does not restart automatically with the NC start, you can force a mid-program startup:

- ▶ Set MP7680 bit 14 to permit an NC start only with mid-program startup or a GOTO command



A message window can be displayed automatically if NC start is pressed after program cancellation.

Locked against contin. machining

The NC program was canceled and locked from continued machining with NC Start.
Do you want to unlock it?
You can also unlock it by using GOTO block number or selecting the program.

The machine operator must then decide whether an NC start is permissible:

- ▶ Set MP7680 bit 15 to permit an NC start only after acknowledging a message, renewed program selection or a GOTO command.

MP7680

Machine parameter with multiple function

Input:

Bit 13 – Behavior during program interruption with axis movement

0: Automatic activation of RESTORE POSITION

1: Do not automatically activate RESTORE POSITION

Bit 14 – Behavior of NC start after NC stop and internal stop

0: NC start permitted

1: NC start only permitted after mid-program startup or GOTO

Bit 15 – NC Start if program is aborted

0: NC start permitted

1: NC Start not permitted (message window)

Retract the tool automatically from the contour

Through the **LIFTOFF** column of the tool table, a function can be activated that retracts the tool from the workpiece by 0.1 mm in tool-axis direction after an NC stop.

- ▶ Enable the LIFTOFF function via M4620.
- ▶ The function is also activated in the NC program with M148. M149 deactivates M148.

		Set	Reset
M4620	Activate LIFTOFF function	PLC	NC/PLC

8.1.11 Adaptive feed control (AFC)

With adaptive feed control (AFC, software option 45), the contouring feed rate is regulated depending on the respective spindle power in percent. This is done with the help of the feed-rate override factor, which is normally determined in the PLC by the setting of the override potentiometer. If AFC is active, then this factor is no longer formed from the potentiometer, but rather from the spindle power and other process data, and the contouring feed rate is calculated. The iTNC then automatically changes the value of the feed-rate override factor, with the goal of maintaining the spindle power at a constant reference value.

The main benefit of this process control, along with ensuring the quality of the machining procedure (e.g. detection of cutter breakage or wear), is the optimization of the machining time, which is intended to ensure or improve productivity and efficiency. Changes in the material (harder sections), deviations in oversizes and tool wear lead to the danger of overloading the spindle. AFC can be effective in counteracting this.

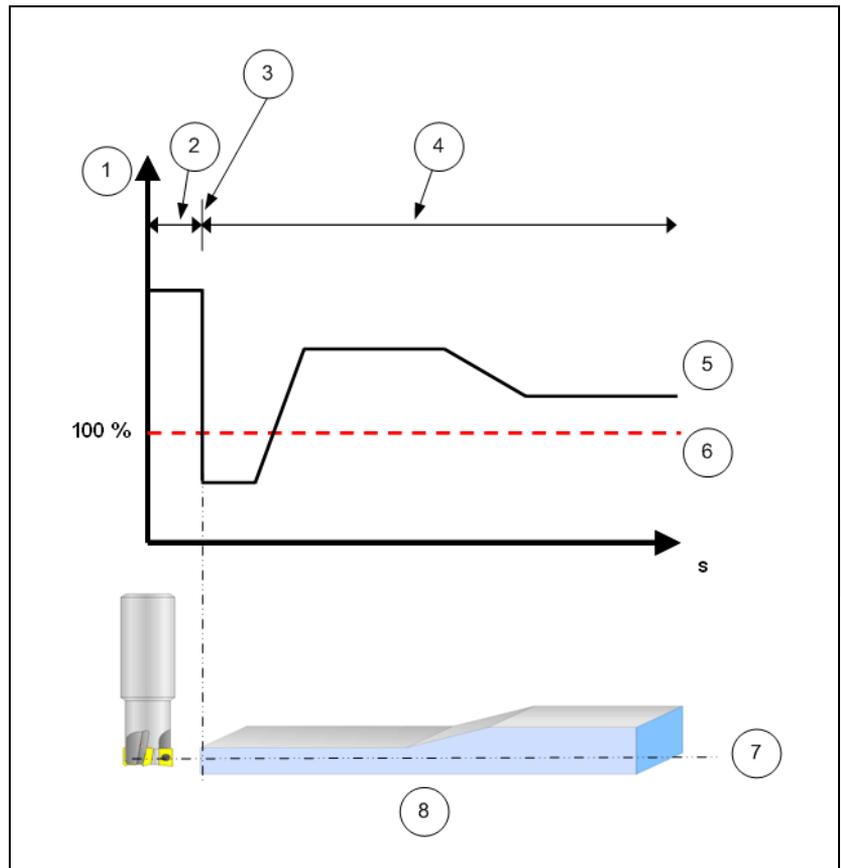
Benefits of adaptive feed control:

- **Optimizing the machining time:**
By regulating the feed rate, the attempt is made to maintain the maximum spindle power learned over the entire machining time.
The total machining time is reduced by increasing the feed rate in machining zones where less material is being removed.
- **Tool monitoring:**
The tool life of a tool is increased by reducing the feed rate when the maximum spindle power learned is exceeded, until the time when this reference spindle power is reached again.
If it is detected that the maximum spindle power was exceeded and at the same time the minimum feed rate could not be maintained, a programmed switch-off reaction is performed. This avoids subsequent damage due to cutter breakage or wear.
- **The machine's mechanical elements are protected**
Damage to the machine due to overload can be avoided by reducing the feed rate ahead of time and with appropriate switch-off reactions.
- **Documentation by capturing and saving the learning and process data.**

Limitations regarding the use of AFC:

- AFC cannot be used with analog spindles.
- AFC cannot be used in volts-per-hertz control mode.

The following graphic schematically shows the principle of adaptive feed control:



- 1: Feed rate
- 2: Air cut
- 3: Beginning of machining
- 4: Feed-rate calculation during machining
- 5: Feed rate calculated by AFC
- 6: Feed rate programmed in the NC program
- 7: Cutting depth
- 8: Workpiece

Operating states of AFC

The two main operating states of adaptive feed control are **Learning** and **Controlling and Monitoring**.

General information about learning

A learning phase must be completed before AFC can be activated. This learning phase serves to ascertain the corresponding reference power of the spindle **PREF** for each individual machining step. The learning phase is divided into one or more learning sections. The beginning of a learning section or machining section is defined by the output of M function M03 or M04 (Spindle on right/left) in an NC program after a **TOOL CALL** has been processed. The end of the cut is defined by the spindle stop (M function M05). The NC macros M3.H, M4.H and M5.H must be assigned to these spindle M functions in the **MFUNCT.TAB** table.

At the beginning of cutting with M3 or M4, the FN17 block:

FN 17:SYSWRITE ID 622 NR0 IDX 0.0 =+1 must be programmed in the NC program or M function macro for starting a cut after the M function.

At the end of cutting with M5, the FN17 block

FN 17:SYSWRITE ID 622 NR0 IDX 0.0 =+0 for ending a cut must be programmed before M5. A new cut is started with each FN 17:SYSWRITE ID 622 NR0 IDX 0.0 =+1 and is ended with FN 17:SYSWRITE ID 622 NR0 IDX 0.0 =+0.

This way AFC can be used without having to change the existing NC program. The only restriction is that no M-function macros can be executed with active radius compensation. Before calling the macros, the radius compensation must be deactivated, and reactivated after the macro if necessary.

The beginning of another section can be defined for the same tool with another call of M03 or M04. In order to exclude a tool from AFC (e.g. touch probe, tap, ...), the AFC column must be undefined (empty) in the **TOOL.T** tool table. Otherwise in this column you must enter the identifier from the global control parameter table **AFC.TAB**. This identifier selects the control parameter block from the table **AFC.TAB**.

If a cutting data table **.CDT** exists and is used, the basic control setting in the AFC column of this table is read. This means that the entry in the cutting data table has precedence over the entry in the tool table. The data ascertained during each learning section of the reference machining, especially the maximum spindle power, are saved with a generated cutting number in a table. They form the foundation for the subsequent controlled machining with active AFC.

As an alternative to the generated cutting number, the value in NR can be used as the cutting number by programming discrete FN17 cutting blocks. However, the programmer must then ensure that the numbers are entered correctly and sequentially. The learning phase can consist of an entire machining process with multiple tools, but it also can also be stopped after a certain time in which the important parameters for the respective tools have been determined. The learning phase can be ended either manually via a soft key or automatically after a programmed time for the machining step has expired.



With **FN 17: SYSWRITE ID 622 NR0 IDX1.0 = <value>** you can transfer a **<value>** for the time in seconds after which the teach-in cuts are ended automatically. The function behaves like the pressing of the **EXIT LEARNING** soft key after the appropriate time. This function is deactivated again by programming **<value> = 0**.

At any time during the teach-in cut, you can change the machining feed rate with the feed-rate override potentiometer any way you want, and so influence the ascertained reference load. The file generated in this step is given the name of the selected NC program and receives **.AFC.DEP** as the file extension. **.DEP** is appended because this file depends on an existing **.H** file. E.g. an NC program with the name **MACHINING.H** results in a file with the name **MACHINING.H.AFC.DEP**. This file, and the global parameter table **AFC.TAB** as well, are set up as freely definable tables, and form view is also possible (switch via soft key). However, the format is prescribed and cannot be changed.

Details of the learning phase:

- The spindle power is only recorded once the nominal speed has been reached, since the spindle is operated at a constant speed, and acceleration and braking phases are not included in the machining.
- Saving of the spindle power in air (idle power) in order to later evaluate only the power used during the actual machining process.
- Detection of entry into the material by the cutter when the idle load is exceeded by 2%.
- Saving of the maximum value of the spindle power of a learning section as reference for the control.
- Saving of the machining time of each learning section.
- Monitoring for stationary spindle or deviation from the nominal speed during active feed not at FMAX.



Note

- Below a certain minimum tool diameter (e.g. 5 mm), in some circumstances it may not make sense to use AFC, since the actual data may become too small for safe control. This can be specified in the tool or cutting data table by leaving the AFC column empty. The entry in the cutting data table makes it possible for the basic setting to depend on the pairing of the tool and workpiece materials.
- If it becomes necessary in other cases for AFC to abort a machining step, there are two possibilities:
 - By leaving the AFC column empty in the tool or cutting-data table.
 - By deactivating AFC via targeted switch-off with **FN 17:SYSWRITE ID 622 NR0 IDX 0.0 =+0** in the NC program or a corresponding M-function macro. If you use the M-function macros for M3 and M4, then the FN 17 block must follow the call of the M-function macro. This way you end the cut for AFC before machining actually takes place.

If the learning process is ended during a learning step, either via the **EXIT LEARNING** soft key or because the programmed time (FN17 - ID622) has expired, the control immediately begins regulating the feed rate adaptively. The value determined up to that point is used as the spindle reference value, and is entered in **<NAME>.H.AFC.DEP.** This is useful if it can be seen that the spindle power will not increase significantly any further in this step.

General information about controlling

The maximum spindle power was determined for each learning and machining step. By changing the feed-rate factor within programmable limits (e.g. 70% to 130%), the software continuously attempts to match the momentary spindle power to the reference value during machining.

This way the feed rate can be increased for areas where only a small amount of material is being cleared, and a slower feed rate can be used when there is more material than usual or the cutter is becoming dull. This protects the spindle drive and the tool from overload. If a disturbance in the process occurs during machining, a programmable switch-off reaction is performed. This occurs if the feed rate falls below the minimum feed-rate factor (e.g. 70%) at the reference spindle load. You can then assume that the cutter has become dull or has broken. If the feed rate falls below the defined value of 30% of the programmed feed rate, an NC stop is performed. This defined error reaction cannot be influenced, and occurs independently of the programmed overload reaction.

Controlling can be switched off and on again via soft key at any point during machining. Controlling is also always switched off if the value set for the override potentiometer is manually reduced by more than 10%. The potentiometer is then effective again instead of AFC. Adaptive feed control must then be switched on again via soft key. A potentiometer value less than or equal to 50% is also always effective, i.e. AFC is inactive then. If the potentiometer value exceeds these 50% while AFC is on, then AFC becomes active again. The maximum spindle power that occurs while under control is ascertained and saved for evaluation later. The machining time of each machining section is also saved.

The events AFC ON, AFC OFF, potentiometer manually changed -> AFC OFF are entered in the log of the iTNC.

If a change to another tool (not replacement tool) takes place within an active cut, the error message "Faulty assignment of tool/cut no." is issued and an NC stop is initiated.

If AFC detects at the beginning of a cut that the tool and the associated cut number do not match, it stops instead of executing an automatic teach-in cut (NC stop) and the error message "Faulty assignment of tool/cut no." is issued. After NC start has been pressed again, a teach-in cut is carried out with the new tool.

In both cases the user can either cancel with internal stop and correct the cause of the error message, or continue with the actual data by pressing NC start.

Tasks of AFC in each machining section:

- Detection of entry into the material by the cutter when the idle load is exceeded by 2%. Regulate to entry speed until a path corresponding to the cutter radius has been cleared, but at most 600 ms.
- Calculation of the optimum feed-rate factor.
- Monitoring for overload and switch-off reaction.
- Saving the maximum value of the spindle power for the evaluation.
- Detection of exit from the material by the cutter when the power is below the programmed exit load, and regulation to the exit speed until a path corresponding to the cutter radius has been traversed.
- After detection of traverse in air, regulate to the programmed idle feed rate.
- AFC is not in effect when machining at rapid traverse, since FMAX is used in the IPO, and at times override for FMAX.

As an alternative method for controlling with AFC, you can use the value in PLC word W632 [value in tenths of percent] as control input variable, instead of using the spindle power in percent. A spindle power of 50% corresponds to the value 500 to be entered in W632 (0.1 % corresponds to the value 1). This may be necessary for individual applications (e. g. pecking).

The control input variable in W632 must be calculated in the PLC program, which can best be done with a fast PLC cycle (e.g. 15 ms). Also, when word W632 is used, no idle load for the spindle is determined and no transitional period for the tool entering or leaving the workpiece is considered.

This function must be activated by entering **AFC.PLCCONTROL = ON** in the OEM.SYS.

However, the function only works if a value is entered in W632.

Automatic selection of the Learning / Controlling status

If an NC program is selected for machining and started, and if AFC is active, then it is first checked whether an associated **<name>.AFC.DEP** file exists in the same directory as the NC file. If this is the case, then the data in this file are immediately used as control parameters, and machining is performed with adaptive feed control. If this file does not exist, the first machining run is used as the learning phase and the file is created. If the AFC file is determined to be incomplete during machining, the missing steps are automatically performed in the Learning mode, thereby generating the missing data.

When Learning is deactivated (M05, manually via soft key, or with FN 17), the learned data are saved and the status of this data block is set to "Controlling." During Controlling, the data blocks are read in the same sequence they were learned, the step numbers are evaluated and other parameters (tool number, index) are checked to see if they match.

The current status or mode of AFC is stored in PLC Word W348 (inactive = 0, learning phase = 1, controlling = 2).

- W348 = 0: AFC inactive (OFF)
- W348 = 1: AFC in learning phase
- W348 = 2: AFC is controlling

Additional FN functions

Further FN 17 functions are available for software option 45 "AFC - Adaptive Feed Control":

- Definition of reference power. The learning phase can therefore be omitted. The reference power is specified in percent with respect to the rated power of the spindle. Example:
FN 17:SYSWRITE ID 622 NRO IDX 3.0 = 85
- Direct switchover from the learning phase to adaptive feed control. The prerequisite for this is that a valid reference load has been reached in the meantime. Example:
FN 17:SYSWRITE ID 622 NR1 IDX 3.0 = 0
- Activating cut-based tool-wear monitoring. The monitoring limit is specified in percent of the reference power. If the spindle power exceeds the monitoring limit, the PLC marker M4510 will be set. In this way, the PLC can initiate the desired reactions, such as setting the tool life of a tool to "expired". Example:
Monitoring limit at 20% with a reference power of 50%. In this way, 60% of the rated power output of the spindle is monitored:
FN 17:SYSWRITE ID 622 NRO IDX 4.0 = 20
- Activating cut-based spindle-load monitoring. The monitoring limit is specified in percent of the reference power. If the spindle power exceeds the monitoring limit, an NC stop is triggered. Monitoring limit at 40% with a reference power of 50%. In this way, the system monitors for 70% of the rated power output of the spindle:
FN 17:SYSWRITE ID 622 NRO IDX 5.0 = 20

The PLC can read in word W350 why the NC stop was triggered:

- W350 = 1 or 2:
NC stop due to falling below the permissible minimum power
- W350 = 3:
NC stop due to detected cutter breakage (FN17 function ID 622 IDX 6.0)
- W350 = 4:
NC stop because the spindle load was exceeded (FN17 function ID 622 IDX 5.0)

■ FN17 – ID 622 IDX 6.0

The function **FN 17: SYSWRITE ID 622 NR<> IDX 6.0** is used to activate spindle load monitoring independently of the cut in order to detect cutter breakage. The monitoring limit is specified in percent with respect to the rated power of the spindle. If the value exceeds or falls below the monitoring limit, an NC stop is triggered.

A filter constant must be specified in parameter **NR<>** in order to avoid an NC stop when the tool leaves the material. Only abrupt changes in spindle power that are typical of tool breakage may be evaluated. The filter constant can be between 1 and 60 ms. If a constant of 0 is entered, the default value 10 ms will be used.

The values for the filter constant and the tolerance limit must be determined for each individual application. Example:

Filter constant of 15 ms, tolerance limit for cutter breakage at 10% of the rated power output of the spindle:

FN 17:SYSWRITE ID 622 NR15 IDX 6.0 = 10

The PLC can read in word W350 why the NC stop was triggered:

- W350 = 1 or 2:
NC stop due to falling below the permissible minimum power
- W350 = 3:
NC stop due to detected cutter breakage (FN17 function ID 622 IDX 6.0)
- W350 = 4:
NC stop because the spindle load was exceeded (FN17 function ID 622 IDX 5.0)



Note

Due to the difficult conditions surrounding setting the values for **FN 17: SYSWRITE ID 622 NR<> IDX 6.0**, you should only use this function after consultation with HEIDENHAIN.

Associated files

Files for machining

The following requirements are mandatory for controlling to be activated:

- An NC program must have been selected (**<name>.H**).
- The column AFC must be enabled in the tool table via MP7266.40.
- If you are working with a **<name>.CDT** cutting table, then you must create an AFC column and enter any desired control strategy there. If an entry in the cutting data table intersects with an entry in the tool table, the entry in the cutting data table has priority.
- M-function macro files M3.H, M4.H, M5.H (and possibly more).
- **MFUNCT.TAB** macro definition.

The three files of adaptive feed control:

File of the global control parameters: **AFC.TAB**

In the **AFC.TAB** table, which must be saved in the **TNC:** root directory, you enter the basic control settings with which the TNC is to perform the feed-rate control.

The data in this table are default values that are copied during the teach-in cut to the file associated with the respective machining program, where they serve as the basis for controlling.

Control parameters for the subsequent standard control strategy are already saved there. Other settings can be programmed as desired.

The static control parameters that must be programmed in **AFC.TAB** for each cutting-data set are:

Column	Function
NR	Consecutive line number in the table (has no further functions)
AFC	Name of the control strategy. You must enter this name in the AFC column of the tool table. It specifies the assignment of control parameters to the tool.
FMIN	Minimum feed rate at which the TNC is to perform an overload reaction if the feed rate falls below this value for one second. Enter the value in percent of the programmed feed rate.
FMAX	Maximum feed rate in the material up to which the TNC can automatically increase the feed rate. Enter the value in percent of the programmed feed rate.
FIDL	Feed rate for traverse when the tool is not cutting (feed rate in the air). Enter the value in percent of the programmed feed rate.
FENT	Feed rate for traverse when the tool moves into or out of the material. Enter the value in percent of the programmed feed rate.
OVLD	Desired reaction of the iTNC to overload: <ul style="list-style-type: none">■ M: Perform an NC-function macro defined by the OEM■ S: Immediately perform an NC stop■ F: Perform an NC stop once the tool has retracted■ E: Only display an error message on the screen■ -: Do not perform an overload reaction The iTNC conducts a shutdown response if the maximum spindle power is exceeded for more than one second and at the same time the feed rate falls below the minimum you defined.
POUT	Spindle power at which the TNC is to detect tool exit from the workpiece. Enter the value in percent of the learned reference load. Recommended input value: 8%
SENS	Sensitivity (aggressiveness) of regulation. A value between 50 and 200 can be entered. 50 is for slow control, 200 for a very aggressive control. An aggressive control reacts quickly and with strong changes to the values, but it tends to overshoot. Recommended value: 100



Column	Function
PLC	The value entered here is written to word W342 at the beginning of a cut. Word W342 can be evaluated in a PLC program. Depending on this, any necessary fine adjustments to the adaptive feed control can be performed from the PLC program. Maximum input range: 3 digits.

Example of an **AFC.TAB**:

NR	AFC	FMIN	FMAX	FIDL	FENT	OVLD	POUT	SENS	PLC
0	Standard	70	125	100	80	-	5	100	0

Proceed as follows to create the AFC.TAB file (only necessary if the file does not yet exist):

- ▶ Select the **Programming and Editing** operating mode.
- ▶ Select the file manager: Press the PGM MGT key
- ▶ Select the **TNC:** directory.
- ▶ Create the new **AFC.TAB** file and confirm with the ENT key. The TNC displays a list with table formats.
- ▶ Select the **AFC.TAB** table format and confirm with the ENT key. The TNC creates the table with the **Standard** control strategy.

Settings file with the control parameters: **<name>.H.AFC.DEP**

This file is automatically created during the learning phase, and filled with values. It contains all control information for the cuts that occur in the NC program. You can manually change all entries, except TNR and IDX, at any time in order to adapt the control parameters specifically to the NC program. Call the **<name>.H.AFC.DEP** via the **SETTINGS TABLE** soft key. It contains the following additional information:

Column	Function
NR	Number of the machining step
TOOL	Number or name of the tool with which the machining step was made (not editable)
IDX	Index of the tool with which the machining step was made (not editable)
PREF	Reference load of the spindle. The TNC measures the value in percent with respect to the rated power of the spindle.
N	Code whether a tool number (0) or tool name (1) was programmed in the NC program.
POUT	Exit load of the spindle. The TNC uses the value as a percent of the reference load PREF of the spindle.
ST	Status of the machining step <ul style="list-style-type: none"> ■ L: (= Learning) In the next program run, a teach-in cut is recorded for this machining step. The iTNC overwrites any existing values in this line. ■ C: (= Controlling) The teach-in cut was performed successfully. The next time it is machined, AFC will control the feed rate.
AFC	Name of the control strategy used. If the control strategy given in the tool table is not in AFC.TAB, then the entry default is in this column. Default settings are then used for adaptive feed control.

Example of a **<name>.H.AFC.DEP** file after a successful teach-in cut:

NR	TOOL	IDX	FMIN	FMAX	FIDL	FENT	OVLD	POUT	PREF	SENS	ST	PLC	AFC
0	2	0	60	140	150	70	-	5	64	100	C	0	Standard

Proceed as follows to select (and edit) the **<name>.H.AFC.DEP** file.

- ▶ Select the **Program Run, Full Sequence** operating mode
- ▶ Shift the soft-key row
- ▶ Select the table of AFC settings with the **SETTINGS TABLE** soft key
- ▶ If necessary, enter any improvements. Edited entries are identified by an * before the name of the control strategy.



The DELETE LINE soft key is also available in the Programming and Editing mode of operation. However, only use this function if you specifically want to delete an incorrectly created line from the <NAME>.H.AFC.DEP table for control settings.

This may be reasonable when lines were created in the table, which would not have been created during normal program run. Such lines might, for example, be produced when program run is canceled, or by a GOTO command to a previous beginning of a cut. You must confirm deletion in a confirmation request, because deleting a line moves up the following lines with their control parameters.



Note

Please note that the control parameters will be assigned to the wrong cuts if you delete lines incorrectly.

If you are not sure which line may be deleted, then perform the learning phase anew for the complete NC program.

Protocol file with the actual data of the control: <name>.H.AFC2.DEP

The iTNC stores various pieces of information for each machining step of a teach-in cut in this file. During control, the iTNC updates the data and makes various evaluations.

Call the <name>.H.AFC2.DEP file via the **EVALUATE TABLE** soft key.

Column	Function
NR	Number of the machining step
TOOL	Number of the tool with which the machining section was performed (cannot be edited)
IDX	Index of the tool with which the machining step was made (not editable)
SNOM	Nominal spindle speed [rpm]
SDIF	Maximum difference of the spindle speed in % of the nominal speed
CTIME	Machining time for the control cut
FAVG	<p>Mean value of the feed rate during cutting in %.</p> <p>This allows you to determine the time saved during cutting.</p> <p>If the FAVG is more than 100 %, time was saved because 100 % is defined by the teach-in cut.</p> <p>If Learning/Controlling is mixed, the average feed rate can be used to calculate the real time difference between learning and controlling. If the average feed rate is clearly below 100 %, this is also a meaningful indication of tool wear.</p> <p>At the end of the program the total time saving calculated is entered in % into the last line next to the total machining time.</p>

Column	Function
FMIN	Mean value of the feed-rate minimum values during cutting in %. Based on the mean value you obtain a meaningful tendency for F-min (tool wear), and not only an individual instantaneous value.
PMAX	Maximum recorded spindle power during machining. The TNC shows the value as a percent of the spindle's rated power.
PREF	Reference load of the spindle. The TNC shows the value as a percent of the spindle's rated power.
OVLD	Reaction by the iTNC to overload: <ul style="list-style-type: none"> ■ M: Perform an NC-function macro defined by the OEM ■ S: Immediately perform an NC stop ■ F: Perform an NC stop once the tool has retracted ■ E: Only display an error message on the screen ■ -: Do not perform an overload reaction <p>The iTNC conducts a shutdown response if the maximum spindle power is exceeded for more than one second and at the same time the feed rate falls below the minimum you defined.</p>
BLOCK	Block number at which the machining cut begins.

Example of a `<name>.H.AFC2.DEP` file after a cut with active AFC:

NR	TOOL	IDX	SNOM	SDIFF	LTIME	CTIME	TDIFF	PMAX	PREF	OVLD
0	2	0	1000	3.5	00:00:40	00:00:32	-20.0	70.2	64	-

Proceed as follows to select the `<name>.H.AFC2.DEP` file.

- ▶ Select the **Program Run, Full Sequence** operating mode
- ▶ Shift the soft-key row
- ▶ Select the table of AFC settings with the **SETTINGS TABLE** soft key
- ▶ Select the protocol file with the **EVALUATE TABLE** soft key



NC-function macro at overload

The entry **M** in the **OVL** column of **AFC.TAB** can be used to call an NC-function macro for each control strategy as a reaction to an overload. The AFC detects that the spindle load has been exceeded if the actual feed rate is below the value in **FMIN** for more than one second. No more than one minute after the overload has been detected is the NC macro performed. The path for the NC macro must be entered in the **PLC:\ncmacro.sys** file via the keyword AFC as follows:

```
AFC = <path><ncmacro>.H  
e.g: AFC = PLC:\NC_MACRO\AFC_STOP.H
```

If this entry is not in the ncmacro.sys file or the NC macro does not exist, then the NC program is stopped with the corresponding error message. The FN17 function SYSWRITE ID990 NR11 can also be used for the AFC NC macro. It makes it possible to perform a tool change with subsequent positioning. This way replacement tool can be inserted in a tool-change procedure as part of an AFC NC macro. Use the following NC macro for this:

```
1 ;Read current tool number: Q1  
2 FN 18: SYSREAD Q1 = ID20 NR1  
3 ;Read maximum tool life of the tool (at TOOL CALL): Q2  
4 FN 18: SYSREAD Q2 = ID50 NR10 IDXQ1  
5 ;Set current tool life of the tool to maximum tool life  
6 FN 17: SYSWRITE ID50 NR11 IDXQ1 = Q2  
7 ;Insert the replacement tool using repositioning logic  
8 FN 17: SYSWRITE ID990 NR11 = 0
```

The FN function FN 17: SYSWRITE ID990 NR11 cannot be used with active radius compensation if the tool change is performed in a change macro. This new FN function works with a tool change only with PLC positioning.

As an alternative to block number 8 in the macro FN 17: SYSWRITE ID990 NR11 = 0, an automatic tool call with M101 (On) and M102 (Off) can be performed at defined points in the NC program. For this, you must delete block 8 from the macro and enter M101/102 at the possible tool change points in the NC program. If the machining run-time between M101/102 pairs is shorter than one minute, it is ensured that a possible tool change will be performed one minute after the overload message M has occurred.

Otherwise, the time until the tool change will be extended. Depending on the duration of the longest individually occurring NC block, the change time may always be extended by the duration of this block.

Example:

An NC block with a duration of 40 seconds, and five blocks, each with a duration of two seconds, are executed in a loop.

The look-ahead times (limited to a machining time of one minute) result in: $40 + 2 + 2 + 2 + 2 + 2 = 50$ seconds. Because the result is < 1 minute, the long block of the next loop is added. The result is $50 + 40 = 90$ seconds of look-ahead calculation. The tool change macro can be inserted at this point, if required.

If the machining process is at the beginning of the long block when the overload occurs, the remaining time of the long block will also be added. This can result in a time of 90 seconds + remaining time, e.g. 35 seconds = 125 seconds plus a further delay through feed rate reduction by AFC (e.g. 80% = $125 * 1.25 = 156$ seconds) from the occurrence of the overload to the execution of the tool change macro.



Default settings for using AFC

Adaptive feed control is a software option (option 45), and must therefore be enabled separately for each machine. The following settings must then be made on the control:

- Create the **TNC:\AFC.TAB** file, and define one or more control strategies.
- Set MP7246 bit 3 so that the <NC name>.H.AFC.DEP table for control settings can be generated for adaptive feed control. A separate file is created for each NC program in the learning phase.
- Use MP7266.40 to enable the **AFC** column in the tool table, or to create the **AFC** column in a ***.CDT** cutting data table.
- Enter the name of the control strategy appropriate for each tool in the **AFC** column of the tool table or cutting data table.
If you do not enter a control strategy for the tool used in the NC program, AFC will not be in effect for this tool.
If you are in the **AFC** column in the tool table, you can use the SELECT AFC CONTROL SETTING soft key to select a control strategy from **AFC.TAB**.
- Set the macro execution for the M functions M3, M4, M5 and any other M functions in the **PLC:\MFUNCTION\MFUNCTION.TAB** table.
- Create the M-function macros M3.H, M4.H, M5.H and any others in the **PLC:\MFUNCTION** directory. In order to edit **FN17**, the appropriate code number (555 343) must first be entered.



Note

The M-function macros cannot be called if radius compensation is active in the NC program. Before calling the macros, the radius compensation must be deactivated, and reactivated after the macro if necessary.

M-function macros with which M3 or M4 automatically define the beginning of a cut for AFC, and M5 defines the end of a cut:

```
0 BEGIN PGM M3 MM
1 M3
2 FN 17:SYSWRITE ID 622 NRO IDX 0.0 =+1;after M3
3 END PGM M3 MM

0 BEGIN PGM M4 MM
1 M4
2 FN 17:SYSWRITE ID 622 NRO IDX 0.0 =+1;after M4
3 END PGM M4 MM

0 BEGIN PGM M5 MM
1 FN 17:SYSWRITE ID 622 NRO IDX 0.0 =+0;before M5
2 M5
3 END PGM M5 MM
```





Note

The M-function macros for M3, M4 and M5 (as listed above) are used to write the control settings for the various cuts sequentially, in the same order as the cuts in the NC program, into the **<NC name>.H.AFC.DEP** file. You cannot use these M-function macros to assign a permanent number to the individual cutting data blocks.

This means that you cannot use the **Program Run, Single Block** operating mode to perform a teach-in cut for just a single cut.

You must always run the entire NC program so that AFC can update the correct lines in **<NC name>.H.AFC.DEP**.

However, the FN 17:SYSWRITE ID 622 function offers you the possibility of numbering all the cuts of an NC program, making a permanent assignment possible. This is considered in the **Program Run, Single Block** operating mode, and you can retroactively carry out the learning process for individual cuts.

You must deactivate the execution of the given M-function macros for this. The definition of the beginning and end of a cut via the FN 17:SYSWRITE ID 622 function must then occur separately for each cut in the NC program.



Note

You cannot use just any numbers when numbering the cuts in the NC program. The beginning of the first cut must be started after M3 or M4 with FN 17:SYSWRITE ID 622 NR0 IDX 0.0 =+1 and be ended before M5 with FN 17:SYSWRITE ID 622 NR0 IDX 0.0 =+0. **NR** must be incremented for each further cut. Targeted access to individual cutting data blocks is only possible with sequential numbering.

Through a changed algorithm, the CC 61xx achieves a lower drive utilization for synchronous and linear motors, provided that the stall current (I-0) is higher than the rated current (I-N) of the motor in the motor table. As a result, the use of automatic feed control (AFC) attains higher milling power. Compatibility mode:

If AFC reference values that were recorded with the CC 424(B) or CC 422 are to be used with the CC 61xx, then you can switch to the old algorithm over MP2221 bit 10.

MP7246 Machine parameter with multiple function

Input: Bit 3 – Settings file for AFC
 0: Do not generate settings file for AFC
 1: Generate settings file for AFC

MP2221.x Handling of linear and synchronous motors to attain higher milling power with AFC

Input: 0: Handling for attaining higher milling power is active
 1: Handling for attaining higher milling power is not active



Initial learning and controlling with AFC

After you have entered the settings described above, proceed as follows to carry out a learning phase for an NC program:

- ▶ In the **Program Run, Full Sequence** operating mode, load the NC program for which the learning procedure is to be carried out, e.g. **<NC name.H.>**
- ▶ Activate AFC with the **ADAPTIVE FEED CONTROL** and **FEED CONTROL ON** soft keys. This creates and displays the **<NC name.H.AFC.DEP>** table of control settings.
- ▶ Start the NC program.
- ▶ The learning procedure is performed automatically for each individual machining cut.
- ▶ The spindle reference power determined with the teach-in cut until this time is displayed in a pop-up window.



Note

You do not have to run the entire machining step in the learning mode. If the cutting conditions do not change significantly, you can switch to the control mode immediately. Press the **EXIT LEARNING** soft key, and the status changes from **L** to **C**.

At any time during the teach-in cut, you can change the machining feed rate with the feed-rate override potentiometer any way you want, and so influence the ascertained reference load.

You can repeat a teach-in cut as often as desired. Manually change the status from **ST** back to **L**. It may be necessary to repeat the teach-in cut if the programmed feed rate is far too fast, and forces you to sharply decrease the feed rate override during the machining step.

The **PREF RESET** soft key enables you to reset the spindle reference power determined by the teach-in cut until this moment, and to restart reference-value determination.

You can teach any number of machining steps for a tool. A machining step always starts with **M3** or **M4**, and ends with **M5**. For machining steps in which the tool remains the same, but it appears useful to define multiple cuts for AFC, you can repeat the corresponding FN 17 function. A new cut is started with each FN 17:SYSWRITE ID 622 NR0 IDX 0.0 =+1 and is ended with FN 17:SYSWRITE ID 622 NR0 IDX 0.0 =+0.

- ▶ All control settings for the individual cuts of an NC program are saved as a table in the **<NC name>.H.AFC.DEP** file (**SETTINGS TABLE** soft key), where they are used for future machining.
The status of each cutting data block changes from L (learning) to C (controlling) after a successful teach-in cut.
- ▶ The data determined in the learning phase is saved in the **<NC name>.H.AFC2.DEP** file (**EVALUATION TABLE** soft key).



Once the NC program has been run, the learning procedure is finished for adaptive feed control. The status in the **<NC name>.H.AFC.DEP** file (**SETTINGS TABLE** soft key) must have switched from L (learning) to C (controlling) for each cut. This way a valid settings file exists for adaptive feed control. As long as the status is not set to L (learning) and adaptive feed control has been deactivated with the **AFC OFF** soft key, the NC program is run with AFC.

- ▶ If you start the NC program again, the feed-rate override is changed by the adaptive feed control. The control tries to maintain the spindle power as constantly as possible at the reference value **PREF** of the spindle power. If the AFC is active, the spindle potentiometer is fixedly set to a factor of 100 % for the Program Run operating modes right from the start of cutting.
- ▶ The data in the **<NC name>.H.AFC2.DEP** file (**EVALUATION TABLE** soft key) determined in the learning phase is compared with the values from machining with AFC.

Activating/ deactivating AFC

Proceed as follows to activate or deactivate adaptive feed control:

- ▶ Select the **Program Run, Full Sequence** operating mode.
- ▶ Shift the soft-key row.
- ▶ Activate adaptive feed control: Set the **AFC** soft key to **ON**.
- ▶ Deactivate adaptive feed control: Set the **AFC** soft key to **OFF**.

The setting of the AFC OFF/ON soft key is saved, and even if the control is reset, an emergency stop occurs, or the machine is switched off via the main switch, the momentary setting of the soft key is stored.

The setting of the AFC OFF/ON soft key is also mapped in PLC word W336.

- W336 = 0: Soft key is set to OFF
- W336 = 1: Soft key is set to ON

If the AFC soft key is set to ON, the spindle potentiometer is fixedly set to a factor of 100 % for the Program Run operating modes right from the start of cutting.



Note

If adaptive feed control is active, the TNC assumes the functions of the feed-rate override. If you reduce the feed-rate override by more than **10%** of the current setting, then the TNC switches AFC off. In this case the TNC displays a window, which you can acknowledge with the CE key.

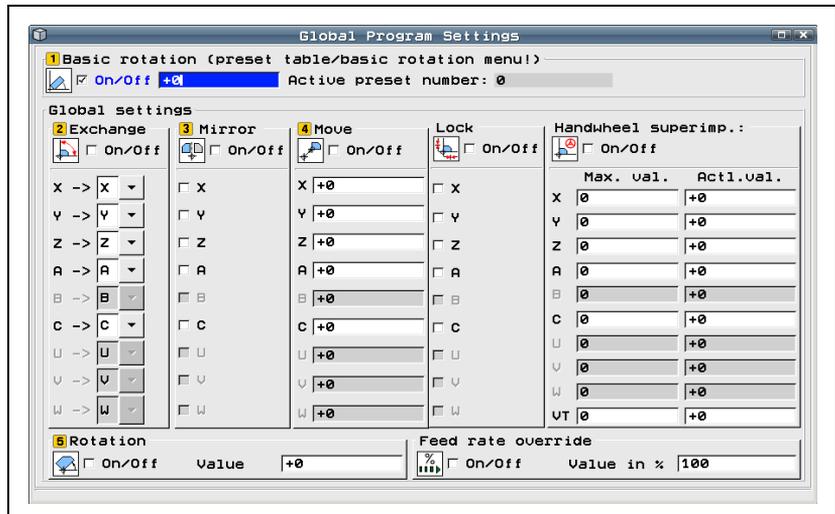
If you set the feed-rate override to less than **50%**, the value of the override potentiometer takes effect. However, AFC is not automatically switched off here. If the value of the override potentiometer later exceeds the 50% again, AFC assumes control again.

The TNC shows various pieces of information in the additional status display when adaptive feed control is active. Please refer to the User's Manual of your control.

8.1.12 Global program settings (PGM)

Global PGM Settings (software option 44) offers you the possibility of superimposing various coordinate transformations and settings in the program-run modes of operation.

For a detailed description of the individual functions, see the iTNC 530 User's Manual.





Attention

The following must be observed when using the **global PGM settings**:

- Basically, all settings in the Global PGM Settings remain active when executing cycles and NC programs. However, when executing macros or PLC positioning movements, all Global PGM Settings are automatically deactivated.
- If you disable one or more axes via Global PGM Settings, then this disabling is only canceled within a macro or during the PLC positioning. However, if you transfer axis position values to the PLC (e.g. with FN19) within a cycle or NC program and then perform a PLC positioning, then this positioning really is performed with all axes. However, if the axis position values transferred from a cycle or an NC program are not approached, due to the Global PGM Settings, then this can lead to a crash during the subsequent PLC positioning.
- M118 cannot be deselected in cycles if handwheel superimpositioning has been activated in the Global PGM Settings. However, handwheel superimpositioning is automatically deactivated in the Global PGM Settings when executing a macro.
- If you read the position values of an axis via the PLC, then when you interrogate the nominal values, the nominal value programmed in an NC program is returned, and not the actual value of the axis. If further actions derive from the nominal value, then this can lead to problems, since these value do not necessarily reflect the actual position values of the axes. If you interrogate the actual value of an axis, then the actual position is returned.

FN17 and FN18 functions for Global Program Settings

You can use the FN 17 SYSWRITE functions to reset global program settings to standard values (ID 330), set them inactive (ID 331) or set values (ID 332).

The FN 18 SYSREAD functions are used to check global program settings for activity (ID 330 and ID 331) and read the values (ID 332).

ID 330 NR 0

Resets everything to default values and checks for active global program settings.

Examples:

- **FN 18: SYSREAD Q1 = ID 330 NRO IDX0**
Returns 1 if a global program setting is active; otherwise 0
- **FN 17: SYSWRITE ID 330 NRO IDX0 = 0**
Resets everything to default values (like SET STANDARD VALUES soft key)

Sets a global program setting inactive/active and checks for active global program settings.

To be able to activate a global program setting with **FN 17: SYSWRITE ID 331 NRx IDXO = 1**, software option 44 "Global PGM Settings" must be set; otherwise the NC program is canceled at this point. Setting a global program setting to inactive, e.g. with **FN 17: SYSWRITE ID 331 NRO IDXO = 0**, also works without software option 44, however.

Examples:

FN 18 functions:

- **FN 18: SYSREAD Q1 = ID 331 NRO IDXO**
Returns 1 if a setting is active, otherwise 0 (identical to ID 330 NR 0)
- **FN 18: SYSREAD Q1 = ID 331 NR1 IDXO**
Returns 1 if a "basic rotation" is active; otherwise 0
- **FN 18: SYSREAD Q1 = ID 331 NR2 IDXO**
Returns 1 if "axis swapping" is active; otherwise 0
- **FN 18: SYSREAD Q1 = ID 331 NR3 IDXO**
Returns 1 if "mirror axes" is active; otherwise 0
- **FN 18: SYSREAD Q1 = ID 331 NR4 IDXO**
Returns 1 if "shifting" is active; otherwise 0
- **FN 18: SYSREAD Q1 = ID 331 NR5 IDXO**
Returns 1 if "rotation" is active; otherwise 0
- **FN 18: SYSREAD Q1 = ID 331 NR6 IDXO**
Returns 1 if "feed rate factor" is active; otherwise 0
- **FN 18: SYSREAD Q1 = ID 331 NR7 IDXO**
Returns 1 if "disable axes" is active; otherwise 0
- **FN 18: SYSREAD Q1 = ID 331 NR8 IDXO**
Returns 1 if "handwheel superimpositioning" is active; otherwise 0

FN 17 functions:

- **FN 17: SYSWRITE ID 331 NRO IDXO = 0**
Sets all global program settings to inactive (like GLOBAL SETTINGS INACTIVE soft key)
- **FN 17: SYSWRITE ID 331 NRO IDXO = 1**
Sets all global program settings to active
- **FN 17: SYSWRITE ID 331 NR1 IDXO = 0**
Sets "basic rotation" to inactive
- **FN 17: SYSWRITE ID 331 NR1 IDXO = 1**
Sets "basic rotation" to active
- **FN 17: SYSWRITE ID 331 NR2 IDXO = 0**
Sets "axis swapping" to inactive
- **FN 17: SYSWRITE ID 331 NR2 IDXO = 1**
Sets "axis swapping" to active
- **FN 17: SYSWRITE ID 331 NR3 IDXO = 0**
Sets "mirror axes" to inactive
- **FN 17: SYSWRITE ID 331 NR3 IDXO = 1**
Sets "mirror axes" to active
- **FN 17: SYSWRITE ID 331 NR4 IDXO = 0**
Sets "shifting" to inactive
- **FN 17: SYSWRITE ID 331 NR4 IDXO = 1**
Sets "shifting" to active
- **FN 17: SYSWRITE ID 331 NR5 IDXO = 0**
Sets "rotation" to inactive



- **FN 17: SYSWRITE ID 331 NR5 IDX0 = 1**
Sets "rotation" to active
- **FN 17: SYSWRITE ID 331 NR6 IDX0 = 0**
Sets "feed rate factor" to inactive
- **FN 17: SYSWRITE ID 331 NR6 IDX0 = 1**
Sets "feed rate factor" to active
- **FN 17: SYSWRITE ID 331 NR7 IDX0 = 0**
Sets "disable axes" to inactive
- **FN 17: SYSWRITE ID 331 NR7 IDX0 = 1**
Sets "disable axes" to active
- **FN 17: SYSWRITE ID 331 NR8 IDX0 = 0**
Sets "handwheel superimpositioning" to inactive
- **FN 17: SYSWRITE ID 331 NR8 IDX0 = 1**
Sets "handwheel superimpositioning" to active

ID 332 NR1 to
ID 332 NR 9

Sets or reads values from the global program settings

For axis-specific program settings, the axis index (e.g. 1 = X axis, 2 = Y axis, ... 9 = W axis) must also be entered. Axis-specific are: ID 332 NR2, NR3, NR4, NR7, NR8 and NR9. For ID 332 NR8 and ID 332 NR9 you can select the VT axis with IDX10. For program settings that are not axis-specific (NR1, NR5, NR6), IDX 0 must be entered.

When writing with **FN 17: SYSWRITE ID 332 NRx IDX y = Q1**, software option 44 must be set, otherwise the NC program is canceled with an error message at this point. During axis swapping (ID 332 NR 2), a successful SYSWRITE call leads to two lines of the exchange table being changed.

Example:

- **FN 17: SYSWRITE ID 330 NR 0 = 0**
Set everything to default values
- **FN 17: SYSWRITE ID 331 NR 2 = 1**
Activate axis swapping
- **FN 17: SYSWRITE ID 332 NR 2 IDX 1 = 2**
Replace X axis by Y axis
Please note that at the same time, the Y axis is replaced by the X axis because every axis letter can only appear once on each side of the exchange table.

FN 18 functions:

- **FN 18: SYSREAD Q1 = ID 332 NR 1 IDX0**
Returns the value of the basic rotation
- **FN 18: SYSREAD Q1 = ID 332 NR 2 IDX1**
Axis swapping: returns the index of the axis that the X axis was exchanged for (e.g. 7 for the U axis).
- **FN 18: SYSREAD Q1 = ID 332 NR 3 IDX1**
Mirror axes: returns 1 if the X axis is mirrored in the global program settings, otherwise 0
- **FN 18: SYSREAD Q1 = ID 332 NR 4 IDX2**
Returns the shift value of the Y axis
- **FN 18: SYSREAD Q1 = ID 332 NR 5 IDX0**
Returns the value of the rotation
- **FN 18: SYSREAD Q1 = ID 332 NR 6 IDX0**
Returns the feed rate override
- **FN 18: SYSREAD Q1 = ID 332 NR 7 IDX3**
Returns 1 if the Z axis is disabled in the global settings

- **FN 18: SYSREAD Q1 = ID 332 NR 8 IDX1**
Returns the max. value of handwheel superimpositioning for the X axis
- **FN 18: SYSREAD Q1 = ID 332 NR 9 IDX10**
Returns the actual value of handwheel superimpositioning for the VT axis

FN 17 functions:

- **FN 17: SYSWRITE ID 332 NR 1 IDX0 = 45**
Sets the basic rotation to 45 degrees
- **FN 17: SYSWRITE ID 332 NR 2 IDX3 = 9**
Performs Z-axis movements with the W axis
- **FN 17: SYSWRITE ID 332 NR 3 IDX2 = 1**
Sets "mirroring" for the Y axis
- **FN 17: SYSWRITE ID 332 NR 4 IDX1 = -10**
Sets -10 as the shift value in X
- **FN 17: SYSWRITE ID 332 NR 5 IDX0 = -45**
Sets the rotation to -45 degrees
- **FN 17: SYSWRITE ID 332 NR 6 IDX0 = 200**
Sets the feed rate override to 200 %
- **FN 17: SYSWRITE ID 332 NR 7 IDX3 = 0**
Undoes disabling of the Z axis
- **FN 17: SYSWRITE ID 332 NR 8 IDX2 = 100**
Sets the max. value of the Y axis handwheel superimpositioning to 100
- **FN 17: SYSWRITE ID 332 NR 9 IDX10 = 5**
Sets the actual value of the VT axis handwheel superimpositioning to 5



8.1.13 Cycles

HEIDENHAIN contouring controls feature standard fixed cycles (e.g. peck drilling, tapping, pocket milling), which can be called in the NC program. In addition to the standard HEIDENHAIN cycles, you can program Original Equipment Manufacturer (OEM) cycles (see the Cycle Design User's Manual). You can influence the function of many HEIDENHAIN standard cycles through machine parameters.

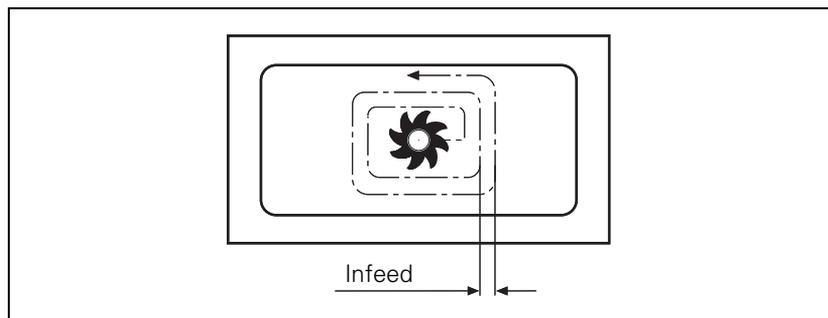
For more information on the tapping cycle and the oriented spindle stop cycle, see page 944.

See page 1535 for more information on the touch probe cycles.

Pocket milling

Cycles 4 and 5:

- ▶ In MP7430, enter the overlap factor for roughing out a rectangular or circular pocket.



$\text{Infeed} = (\text{MP7430}) \cdot \text{cutter radius}$

MP7430 is only effective for Cycles 4 and 5.

MP7430 **Overlap factor for pocket milling**

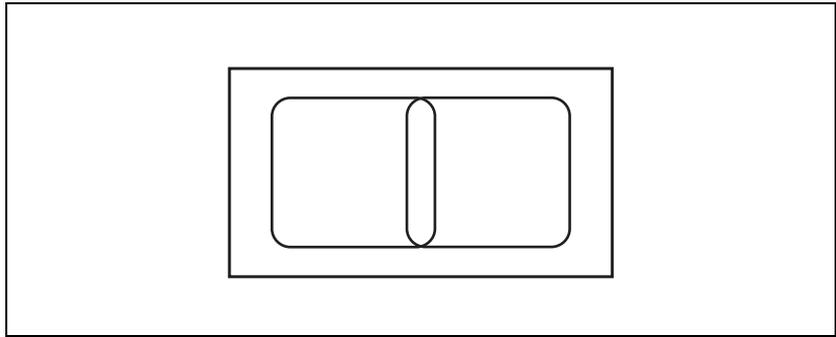
Input: 0.001 to 1.414

Cycles for milling pockets with combined contours

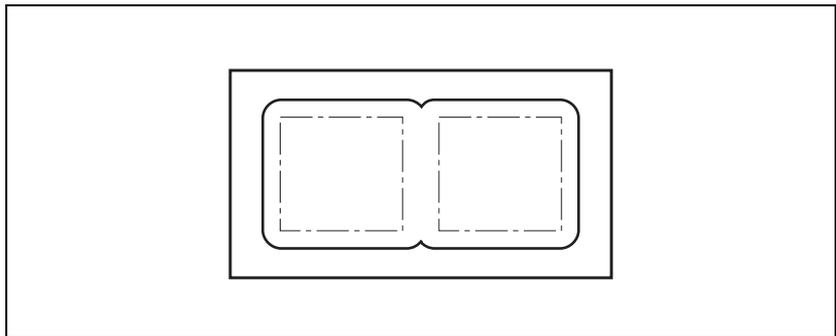
Cycles 6, 14, 15, 16:

- ▶ With MP7420, specify:

- Bit 0: The milling direction for channel milling
- Bit 1: The sequence for rough-out and channel milling
- Bit 2: The conditions under which programmed pockets should be merged (see graphics below)
- Bit 3: Whether each process (channel milling or pocket clearing) is to be completed for all pecking depths before performing the other process, or whether both are to be performed alternately for each pecking depth
- Bit 4: Position after completion of the cycle

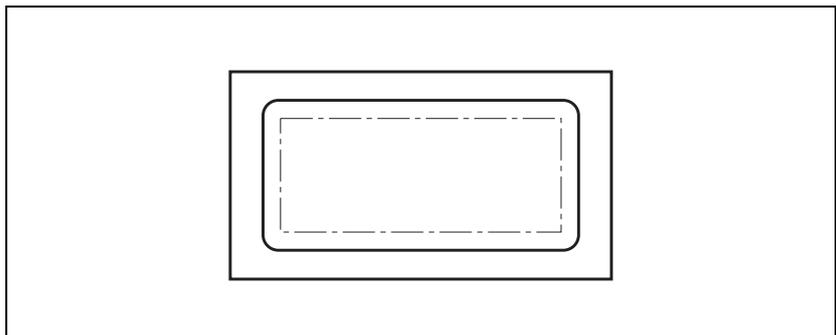


The programmed contours of two pockets intersect slightly.



MP7420 bit 2 = 0:

The control clears the pocket separately because the paths of the tool center do not intersect. Material will remain at inside corners.



MP7420 bit 2 = 1:

The control clears the pockets together because the programmed contours intersect. No material will remain at inside corners.

MP7420**Cycles for milling pockets with combined contours**

Format: %xxxxx

Input:

Bit 0 – Milling direction for channel milling

0: Counterclockwise for pockets, clockwise for islands

1: Clockwise for pockets, counterclockwise for islands

Bit 1 – Sequence for rough-out and channel milling (only for SL 1):

0: First channel milling, then pocket rough-out

1: First pocket rough-out, then channel milling

Bit 2 – Merging of listed contours

0: Contours are merged only if the tool-center paths intersect

1: Contours are merged if the programmed contours intersect

Bit 3 – Rough-out and channel milling to pocket depth or for every infeed

0: Each process uninterrupted to pocket depth

1: Both processes for each pecking depth before proceeding to the next depth

Bit 4 – Position after completion of the cycle

0: Tool moves to the same position as before the cycle was called

1: TNC retracts the axis to the "clearance height"

Scaling factor

Cycle 11:

- ▶ In MP4710, enter the effective range of the scaling factor.

MP7410**Scaling cycle in two or three axes**

Input:

0: Scaling cycle is effective in all three principal axes

1: Scaling cycle is effective only in the working plane

Cylindrical surface

With Cycles 27, 28 and 39 you can machine a contour on a cylindrical surface (see the User's Manual).

In standard approach/departure movements of Cycle 39, the tool approaches/ departs the contour tangentially on an arc that depends on the tool radius. If contours are close to each other, the approach/departure movements might damage adjacent contours.

- ▶ Define the rotation center of the rotary axis with MP75xx. Also see page 701.
- ▶ Define the behavior of Cycle 28 with MP7680 bit 12.
- ▶ Define the behavior of Cycle 39 with MP7680 bit 16.

MP7680**Machine parameter with multiple function**

Input:

Bit 12 – Behavior of Cycle 28

0: Standard behavior

1: The slot wall is approached and departed tangentially; at the beginning and end of the slot a rounding arc with a diameter equal to the slot is cut

Bit 16 – Behavior of Cycle 39

0: Standard behavior

1: Approach/departure movement on an arc is not active

8.1.14 End of program run

If the program end is reached in the operating modes **Program Run, Single Block** and **Program Run, Full Sequence**, the NC sets M4170. This marker is reset with the next program start.

You can evaluate the information "Program end" during program run with pallet changers, for example.

		Set	Reset
M4170	END PGM, M02 or M30 was executed	NC	NC

8.1.15 Returning to the contour

With HEIDENHAIN contouring controls you can resume an interrupted NC program at a specified block number by scanning the previous blocks (see "Mid-Program Startup" in the User's Manual).



Note

- M functions M142 and M143 are not permitted with mid-program startup.
- If you perform a mid-program startup in a program containing M128, then the TNC performs any compensation movements necessary. The compensation movements are superimposed over the approach movement.

If the program is aborted due to a power outage, and **power-fail monitoring is active**, then after the **Power interrupted** message is acknowledged, the **NC program canceled** message appears. The point of interruption is remembered, and is offered for the mid-program startup.



Note

- You must enable these functions through machine parameters.
- You must prepare the PLC program accordingly.

Markers inform the PLC about individual conditions during mid-program startup (block scan). Depending on these markers you can enable certain functions such as the axis-direction buttons for **MANUAL TRAVERSE**.

M4156 is set if the **MANUAL TRAVERSE** soft key is pressed.

M4157 is set if the **RESTORE POSITION** soft key is pressed ("Return to Contour").

M4158 is set if the **RESTORE POS. AT** soft key is pressed. M4158 is reset if the **RESTORE POSITION** or **INTERNAL STOP** soft key is pressed.



During the block scan, PLC positioning commands are included in calculation only if they are also executed. The TOOL CALL block normally initiates PLC positioning commands for tool change. If you want these positioning commands to be calculated in the block scan:

- ▶ In MP951.x, enter the absolute position with respect to the machine datum. The values for MP951.x can be assumed with the "actual position capture" key.
- ▶ Activate the calculation for the specific axes with MP7450.

With flexible tool-pocket coding in the central tool file (see "Tool changer" on page 1565), the change of pocket number in the tool file must be prevented during block scan if the TOOL CALL blocks are not collected:

- ▶ Set M4542.

The block scan can be interrupted by a programmed STOP or with M06, whereby you can have the programmed dwell time included:

- ▶ With MP7680, bits 3 and 4, select the parameters for the block scan.
- ▶ With MP7451.x, define the feed rate for returning to the contour.

If an NC program block is interrupted in **Single Block mode** or by a **STOP** block and the positions of NC axes are changed, the NC program can be restarted at the changed positions. If in OEM.SYS **STRICTREPOS = YES**, the function for restoring the position is activated (see "OEM.SYS" on page 1701).



Note

If you change the tool data in the PLC, update them with M4538, or change or update them with an NC macro, then the new tool data cannot be correctly offset in the block scan.

MP951.x Simulated tool-change position for TOOL CALL during mid-program startup (block scan)

Input: -99 999.9999 to +99 999.9999 [mm] or [°]

MP7450 Offsetting the tool change position from MP951.x in block scan

Format: %xxxxxxxxxxxxxxxxxxxxxx

Input: Bits 0 to 17 represent axes 1 to 18

0: Do not offset

1: Offset

MP7451.x Feed rate for returning to the contour

Input: 10 to 1 000 000 [mm/min]

MP7680 Machine parameter with multiple function

Input: Bit 1 – Returning to the contour

0: Not active

1: Active

Bit 2 – Block scan

0: Not active

1: Active

Bit 3 – Interruption of block scan for STOP or M06

0: Interruption

1: No interruption

Bit 4 – Inclusion of programmed dwell time during the block scan

0: Include the dwell time

1: Do not include the dwell time

Bit 5 – Start of calculation for block scan

0: Start from block with cursor

1: Start from beginning of program

		Set	Reset
M4156	MANUAL TRAVERSE soft key pressed	NC	NC
M4157	Returning to the contour (RESTORE POSITION) is active	NC	NC
M4158	Block scan active	NC	NC
M4538	Geometry of the tool from W264	PLC	NC
M4542	Do not update pocket number in the pocket table	PLC	PLC



M/S/T/Q transfer during block scan

The PLC can collect the M/S/T/Q signals during the block scan in order to output them after the block scan ends:

- ▶ Set MP7681 to a value other than zero so that after a block scan the message **RESTORE MACHINE STATUS** is displayed and output with the M/S/T/Q signals.
- ▶ With the external start key, activate the output of the displayed signals. As long as these signals are being output, M4161 remains set.
- ▶ In the system files **PLC:\MGROUPS.SYS** and **PLC:\MSPLIT.SYS**, define the M functions to be output after a block scan.

The M/S/T/Q signals are output in this sequence (exception: see the **ORDER=PRIO** instruction):

- 1. M function that was defined with **MFIRST**
- 2. M/S/T/Q signals in the programmed sequence
- 3. M function that was defined with **MLAST**

After **RESTORE MACHINE STATUS**, the control checks whether the status set by the PLC agrees with the status calculated by the NC. No error message appears if this is the case; if for example another traverse range is selected in a tool change macro but the original traverse range is set at the end of the macro. If the NC status and PLC status do not match, the error message **PLC function not permitted** appears.

In order to run the above named functions on machines that have executed them through the PLC, leading to the **PLC function not permitted** error message, as an alternative there are functions that can be executed via an NC macro:

- **FN17: SYSWRITE ID20 NR13**, to switch between two spindles
- **FN31: RANGE SELECT**, to switch the traverse range (**RANGE**), the axis assignment (**ASSIGNED**), and the axis display (**DISPLAYED**)
- **FN32: PLC PRESET**, to execute a PLC datum shift

The functions are only visible with code number 555343.

Function	NC	PLC
Spindle switch-over	FN17: SYSWRITE ID20 NR13	Module 9175
Range of traverse, axis assignment, axis display	FN31: RANGE SELECT	Module 9152
Datum shift	FN32: PLC PRESET	Module 9230

Canceling block scan

If block scan is canceled, it is possible for the NC status and PLC status not to match. A macro can be entered in NC MACRO.SYS after the keyword **STARTUPCANCEL** = for this. This macro is always called when block scan is not ended with **RESTORE MACHINE STATUS**. This macro brings the NC into concordance with the actual condition of the machine (traverse range, spindle, etc.).

MP7681 bit 4 enables you to activate an MP subfile during block scan. The MP subfile is activated with FN17 ID1020.



Danger

If bit 4 is set and the block scan is canceled nevertheless, the respective changes will remain effective!

If required, you have to undo these changes in order to restore a consistent state.

You can use a suitable NC macro for this, for example.

MP7681

Format:

Input:

M/S/T/Q transfer to the PLC during block scan

%xxxx

Bit 0:

0: Transfer M functions to the PLC during block scan

1: Collect M functions and transfer to PLC after block scan.

Bit 1:

0: Transfer T code to the PLC during block scan

1: Transfer last T code to the PLC after block scan

Bit 2:

0: Transfer S or G code to the PLC during block scan

1: Transfer last S or G code to PLC after block scan

Bit 3:

0: Transfer FN19 outputs to the PLC during block scan

1: Transfer last FN19 outputs to the PLC after block scan

Bit 4:

0: MP subfiles are not activated during block scan

1: MP subfiles are activated during block scan

		Set	Reset
M4161	M/S/T/Q transfer after block scan	NC	NC



Delay time for the change signals (M/S/T)

MP7444 allows you to define the time for the output of the change signals M/S/T after a positioning movement. This enables you to reduce the machining time by avoiding filter run times. It is not always necessary to wait until the filter run times have been completed (e.g. switch-on of cooling water). The delay time can be deactivated in MP7444, if required.

MP7444 Delay time for the change signals (M/S/T)

Input: 0: Previous behavior, change signal after complete filter run time
1: Change signal if the nominal feed rate reached the value 0 before the actual value (formed by the filters) did.
2: Change signal if the nominal feed rate reached the value 0 before the actual value (formed by the filters) did, and an additional delay until the actual value reaches the time window the first time.

Instructions in MGROUPS.SYS

GROUP =

You divide M functions into groups. After a block scan, the last programmed M function in a group is transferred to the PLC. An M function can be entered in up to eight groups. During block scan, the M function is collected for all groups in which it occurs, and is then considered during restoration. However, this could lead to such M functions being executed more than once when restoring the machine status. This function only makes sense if the **ORDER = PRIO** option has not been entered.

Example: **GROUP = M3,M4,M5**

SPECIAL =

You define all M functions that are not defined in a group, and that should be sent to the PLC after a block scan.

Example: **SPECIAL = M19,M22,M25**

MFIRST =

MLAST=

You define two M functions to be sent to the PLC at the start and end of an output sequence after a block scan. This enables the PLC program to recognize that a sequence of M/S/T/Q strobes that was collected during the block scan is being transferred. You can omit these instructions if you do not need this information.

Example: **MFIRST = M80, MLAST = M81**

REMAIN = OUTPUT

All M functions that are not defined in MGROUPS.SYS are transferred during the block scan to the PLC. If you do **not** enter this instruction such M functions are ignored.



Note

You must use **REMAIN = OUTPUT** in the following functions:

- Datum shift with M4132
- PLC positioning except with TOOL CALL
- Traverse range switchover with M4135, if MP7490 = 1
- Switchover spindle 1/spindle 2 with Module 9175

If the PLC shifts the datum or switches the traverse range, the error message **PLC function not permitted** appears.

ORDER = PRIO

The M functions are transferred in the sequence in which they are entered in the MGROUPS.SYS file. If you do **not** enter this instruction, the M functions are transferred after a block scan in the sequence in which they were programmed.

HEIDENHAIN recommends that you **avoid** using this instruction.

TOOLGROUP =, TDEFGROUP =, SPINDLEGROUP =, FN19GROUP =

In conjunction with **ORDER = PRIO**, the output sequence of the S/T/Q strobes after a block scan is specified in the MGROUPS.SYS file. HEIDENHAIN recommends that you **avoid** using these instructions.

NCMACRO = TC, M

With an M function or during a TOOL CALL, you can also call an NC program instead of a T strobe (see "Tool changer" on page 1565 and "Calling an NC macro with an M function" on page 1258).

NC macros are always run during the block scan. The included M functions are collected if they are listed in a group of the MGROUPS.SYS file. The entry **NCMACRO =** now causes the complete tool change macro (TC) and the M function macros (M), respectively, to be run when the machine status is restored.

Instructions in MSPLIT.SYS

M functions that are effective in several groups are divided in the MSPLIT.SYS file into function components.

Example: **M13=M3, M8**



8.1.16 M functions

In the iTNC you can program miscellaneous functions, also known as M functions. The code of an M function is transferred to the PLC before or after execution of the NC block.

M89 to M299 are reserved for the NC, and several M functions between M00 and M88 have fixed meanings for the NC. The other M functions are freely available.

Effective at A = beginning of block
 E = end of block

M function	Meaning	Effect
M00	Program STOP/Spindle STOP/Coolant OFF	E
M01	Optional program STOP	E
M02	Program STOP/Spindle STOP/Coolant OFF/possible clearing of the status display ^a /go to block 1	E
M03	Spindle ON clockwise	A
M04	Spindle ON counterclockwise	A
M05	Spindle STOP	E
M06	Tool change/Program STOP ^b /Spindle STOP	E
M07		A
M08	Coolant ON	A
M09	Coolant OFF	E
M10		E
M11		A
M12		E
M13	Spindle ON clockwise/Coolant ON	A
M14	Spindle ON counterclockwise/Coolant ON	A
M15 - M18		A
M19		E
M20 - M29		A
M30	Same as M02	E
M31		A
M32 - M35		E
M36 - M51		A
M52 - M54		E
M55 - M59		A
M60		E
M61		A
M62		A
M63 - M70		E
M71 - M88		A

M function	Meaning	Effect
M89	Vacant miscellaneous function or cycle call, modally effective ^b	E
M90	Operation with following error: Constant feed rate in corners	A
M91	Within the positioning block: Coordinates are referenced to machine datum	A
M92	Within the positioning block: Coordinates are referenced to a position defined by the machine tool builder, such as tool change position	A
M93		A
M94	Reduce the rotary axis display to a value below 360°	A
M95 - M96	Approach behavior at the starting point of the contour	E
M97	Machine small contour steps	E
M98	Machine open contours completely	E
M99	Blockwise cycle call	E
M100		E
M101	Automatic tool change with replacement tool if maximum tool life has expired	A
M102	Reset M101	E
M103	Reduce feed rate during plunging to factor F	A
M104	Reactivate most recently set datum	A
M105	Machining with second k_V factor	A
M106	Machining with first k_V factor	A
M107	Suppress error message for replacement tools with oversize	A
M108	Reset M107	E
M109	Constant contouring speed on the tool cutting edge (increasing and decreasing the feed rate)	A
M110	Constant contouring speed on the tool cutting edge (only decreasing the feed rate)	A
M111	Reset M109/M110	E
M112	Insert rounding radius between nontangential straight lines	A
M113	Reset M112	E
M114	Automatic compensation of machine geometry when working with tilted axes	A
M115	Reset M114	E
M116	Feed rate for rotary axes in mm/min	A
M117	Reset M116	E
M118	Superimpose handwheel positioning during program run	A
M119		
M120	Pre-calculate the radius-compensated contour (LOOK AHEAD)	A
M121 - M123		

M function	Meaning	Effect
M124	Ignore points for calculating the rounding arc with M112	A
M125		
M126	Shortest-path traverse of rotary axes	A
M127	Reset M126	E
M128	Retain position of tool tip when positioning tilting axes (TCPM)	A
M129	Reset M128	E
M130	Within the positioning block: Points are referenced to the non-tilted coordinate system	A
M131		A
M132		A
M133		E
M134	Exact stop at nontangential contour transitions when positioning with rotary axes	A
M135	Reset M134	E
M136	Feed rate F in mm per spindle revolution	A
M137	Reset M136	E
M138	Selection of tilted axes	A
M139		A
M140	Retraction from the contour in the positive tool axis direction	A
M141	Suppress touch probe monitoring	A
M142	Delete modal program information	A
M143	Delete basic rotation	A
M144	Compensating the machine's kinematics configuration for ACTUAL/ NOMINAL positions at end of block	A
M145	Reset M144	E
M146	Save the current geometry information in a temporary file (tool-oriented pallet machining)	A
M147		A
M148	Retract the tool automatically from the contour at NC stop	A
M149	Reset M148	E
M150	Utilize the traverse range completely in NC blocks	A
M151 - M199		
M200	Laser cutting: Direct output of the programmed voltage	A
M201	Laser cutting: Output of voltage as a function of distance	A
M202	Laser cutting: Output of voltage as a function of speed	A
M203	Laser cutting: Output of voltage as a function of time (ramp)	A
M204	Laser cutting: Output of voltage as a function of time (pulse)	A
M205 - M299		A
M300 - M999		

- a. depends on MP7300
- b. depends on MP7440

- ▶ In the PLC, evaluate the M functions that have no fixed meaning for the NC.

When an M function is transferred to the PLC, the code of the M function is saved in W260 and the strobe marker M4072 is set:

- ▶ Set M4092 in order to report the execution of the M function. The next NC block is run. M4072 is reset by the NC.

The Mfunctions M00 to M99 can also be transferred decoded to the markers M1900 to M1999:

- ▶ Activate this function with M4571.



Note

M functions greater than 99 but less than 299 are not transferred to the PLC. They have a fixed meaning for the NC to activate certain functions.

		Set	Reset
W260	Code for M function	NC	NC
M4072	Strobe signal for M function	NC	NC
M4092	Acknowledgment of M function	PLC	PLC
M4571	Activation of decoded M-code transfer in M1900 to M1999	PLC	PLC
M1900 - M1999	Decoded M function if M4571 is set	NC	NC



Status of M functions

- ▶ With Module 9060 you can ascertain the status of M functions M100 to M199.
- ▶ With Module 9061 the status of the non-modal M functions M94, M142, M143 and M146 can be ascertained.

Module 9060 M function status

Module 9060 can determine whether an M function between M100 and M199 is active.

Call:

PS B/W/D/K <Number of the M function (100 to 199)>

CM 9060

PL B/W/D <Status>

0: M function inactive

1: M function active

Error code:

Marker	Value	Meaning
M4203	0	Status was ascertained
	1	Error code in W1022
W1022	1	Invalid number of M function

Module 9061 Status of non-modal M functions

With Module 9061 the status of the non-modal M functions M94, M142, M143 and M146 can be interrogated. The status of the interrogated M function is then set until the module is called again, even if the NC program has finished.

Call:

PS B/W/D/K <Number of the M function (90 to 199)>

CM 9061

PL B/W/D <Status>

0: M function was not active

1: M function was active

Error code:

Marker	Value	Meaning
M4203	0	Status was ascertained
	1	Error code in W1022
W1022	1	Invalid number of M function

Program stop with M functions

In the **Program Run, Single Block** and **Program Run, Full Sequence** operating modes the next NC block is not run until you have reported execution of the M function:

- ▶ Set M4092 in order to report the execution of the M function.

For special machines you can deselect the program stop (see "Special functions for laser cutting machines" on page 1561).

- ▶ Select the program stop with MP7440, bit 2.

Program stop with M06

According to ISO 6983, the M function M06 means "tool change."

- ▶ With MP7440 bit 0, select program stop when M06 is transferred to the PLC. After the program stop and the tool change, the NC program must be restarted through an NC start or by the PLC.

Modal cycle call M89

You can use the M function M89 to program a modal cycle call. The possibilities for calling a cycle are:

- NC block **CYCL CALL**
- Miscellaneous function M99. M99 is non-modal, i.e. it must be specially programmed each time it is to be executed.
- Miscellaneous function M89. M89 depends on MP7440 bit 1. M89 is effective modally as a cycle call, i.e. in each subsequent positioning block, the fixed cycle last programmed is called. M89 is canceled by M99 or by a CYCL CALL block.

If M89 is not defined as a modal cycle call, it is transferred to the PLC as a normal M function at the beginning of the block.

Reduced feed rate of tool axis with M103

With M103 F.. you can reduce the contouring feed rate for motion in the negative direction of the tool axis. The tool axis share of the feed rate is limited to a value that the iTNC has calculated from the most recently programmed feed rate.

$$F_{\max} = F_{\text{prog}} \cdot F_{\%}$$

F_{\max} = maximum feed rate in the negative direction of the tool axis

F_{prog} = most recently programmed feed rate

$F_{\%}$ = programmed factor behind M103 as a percentage

M103 F.. is canceled by re-entering M103 without a factor.

- ▶ Enable the M103 F.. function with MP7440 bit 2.

Automatic activation of M134

In the standard setting, a transition element is inserted for positioning with rotary axes at non-tangential transitions (depending on the acceleration, jerk and tolerance). With M134, a precision stop is performed at these transitions:

- ▶ Enable the automatic activation of M134 with MP7440 bit 6.

MP7440

Output of M functions

Format:

%xxxxxxx

Input:

Bit 0 – Program stop with M06:

0: Program stop with M06

1: No program stop with M06

Bit 1 – Modal cycle call M89:

0: Normal code transfer of M89 at beginning of block

1: Modal cycle call M89 at end of block

Bit 2 – Program stop with M functions:

0: Program stop until acknowledgment of the M function

1: No program stop. Acknowledgment is not waited for.

Bit 3 – Switching of k_v factors with M105/M106:

0: Function is not in effect

1: Function is effective

Bit 4 – Reduced feed rate in the tool axis with M103

0: Function is not in effect

1: Function is effective

Bit 5 – Reserved

Bit 6 – Automatic activation of M134

0: M134 must be activated in the NC program

1: M134 is automatically activated when an NC program is selected

Error messages during cycle call

Before execution of a fixed cycle, the spindle must be started with M3 or M4. If this is not the case, the error message **Spindle ?** appears. If you are using a high speed cutting (HSC) spindle that is started by its own M function (not M3 or M4):

- ▶ Suppress the error message **Spindle ?** with MP7441 bit 0.

If a positive depth is programmed in machining cycles, the error message **Enter depth as negative** appears:

- ▶ Suppress the error message **Enter depth as negative** with MP7441 bit 2.

MP7441

Error message during cycle call

Format:

%xxx

Input:

Bit 0 –

0: Error message **Spindle ?** is not suppressed

1: Error message **Spindle ?** is suppressed

Bit 1 – Reserved, enter 0

Bit 2 –

0: Error message **Enter depth as negative** is suppressed

1: Error message **Enter depth as negative** is not suppressed

Auxiliary cycles

Cycles 18 (thread cutting) and 33 (thread on taper) are auxiliary cycles. You cannot use them alone, but you can use them for your OEM cycles (see the User's Manual).

► Set MP7245 = 1 to enable the auxiliary cycles.

MP7245 Disable auxiliary cycles

Input: 0: Auxiliary cycles disabled
 1: Auxiliary cycles permitted

Calling an NC macro with an M function

The M functions M0 to M88 and M300 to M999 can call an NC macro in all operating modes. First the table **PLC:\MFUNCT\MFUNCT.TAB** must exist. Line number 0 represents M0, line number 1 represents M01, etc. The NC macros must be entered in the directory **PLC:\MFUNCT** with the name of their M function (e.g. **M301.H**).

M functions that call an NC macro are not sent to the PLC. If an M function defined in **MFUNCT.TAB** is programmed in a macro, the M function is reported to the PLC.

To synchronize the current machine status and the look-ahead calculation with an NC macro call, see "NCMACRO.SYS" on page 1709.

For behavior during a block scan, see "Instructions in MGROUPS.SYS" on page 1249.

With **FN17: SYSWRITE ID420 NRO IDXO = 0**, all coordinate transformations (e.g. cycles 7, 8, 10, 11, 19) performed in the NC macro become globally effective. Without this block, they remain locally effective (only in the NC macro). If a coordinate transformation status is to be globalized, FN17: ID420 must be programmed after the nominal condition has been established.

If, for example, an M-function macro that executes (globalizes) e.g. PLANE RESET is called from an OEM cycle, resetting the PLANE function is only effective one level higher. When the cycle has been executed, the last PLANE function is active again. This means: If PLANE is to be reset permanently, the globalization must also be carried out in the OEM cycle.

Explanation of the columns in the table MFUNCT.TAB:

Column name	Description	Input
NR	Number of the M function	–
MACRO	Is the macro present?	YES: "Y", "y" or "1"
EFFECTIV	Is the M function effective at the beginning or end of block (only M0 to M88 and M300 to M999)?	NO: "N", "n" or "0"
WAIT	Only for M functions that do not call NC macros: Should the NC wait for acknowledgment from the PLC (only for M0 to M88 and M300 to M999)? If there is no entry in this column, MP7440 bit 2 is valid.	
MANLOCK	Is the M function locked in the manual operating modes?	
NONESTED	May the macro not be called by another macro?	
RK	Should the macro stored for the M function also be used for active radius compensation (RR/RL)? Until now, the error message Cancel radius comp. before PLC pos. was displayed. A macro that can also be run during active radius compensation should only contain NC functions that do not influence radius compensation in the calling program.	



Note

A maximum of 30 NC programs can be nested (subprograms, cycles, macros).

8.1.17 Powering up and shutting down the control

Powering up the control

While the control is starting, a customer-specific company logo can be displayed instead of the HEIDENHAIN logo. If a service pack is installed, then a different logo may appear instead of the message window.

Requirements of the logo:

- The logo must be a bitmap file (*.BMP) with a color resolution of 16 or 24 bits. It is displayed in 16 bits.
- Maximum picture size:
 - BF 250: 1024 x 768
 - BF 260: 1280 x 1024



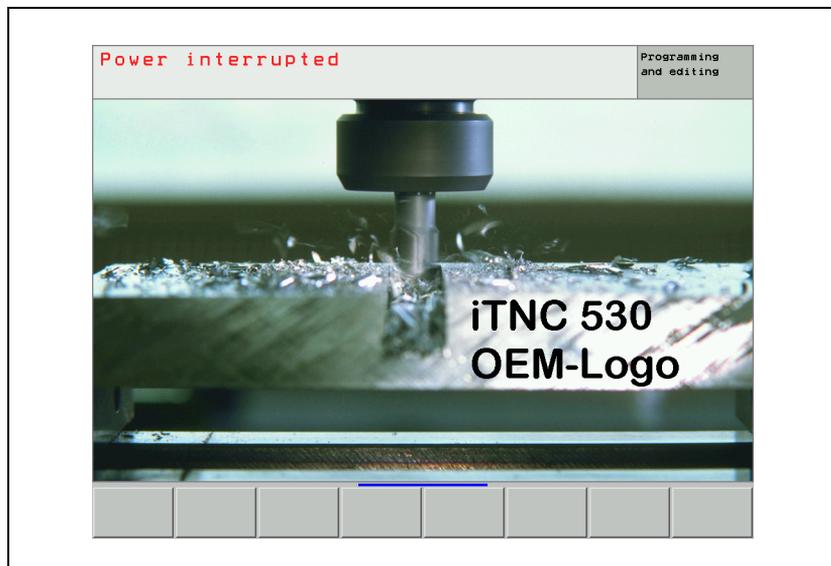
Note

If the picture is larger than the window, it will be cropped symmetrically. If the picture is smaller than the window, it will be displayed centered.

- The logo must be designed to remain recognizable even with a reduced window:

While the control is powering up, the complete logo is displayed. If power is interrupted, only a reduced window remains available:

 - BF 250: 1020 x 681
 - BF 260: 1142 x 848



- ▶ Standard logo: Enter the keyword **LOGO** = in OEM.SYS, followed by the complete path of the logo, e.g. **LOGO = PLC:\LOGO\OEM-LOGO.BMP**
- ▶ Logo with service pack installed: Enter the keyword **LOGOSP** = in OEM.SYS, followed by the complete path of the logo, e.g. **LOGOSP = PLC:\LOGO\SP-LOGO.BMP**
- ▶ On the TNC, create a new directory, for example **PLC:\LOGO**.
- ▶ Transfer the logo to this directory

Shutting down the control

The control must be shut down before it can be switched off. This is done either with the soft key in **Manual mode** or by the PLC with Module 9279 or 9189. If the control is shut down (either with a PLC module or with the soft key), M4179 is set, the ready signal of the drives is removed, and the hard disk is set to sleep mode.

If a PLC output is to be set after shutting down the control, this function must be activated with MP4040 = 1 or 2. Shutting down via Module 9279 must be done with Mode 2. After the control has been shut down and after the time from MP4041 has expired, the PLC output specified in MP4044 is set.

PLC outputs of a PL 62xx or an MB machine operating panel can be used for this function. However, the following must be kept in mind:

- The maximum current of an output on a machine operating panel is limited to 150 mA (500 mA with the PL 62xx).
- Galvanic isolation of the PLC supply must be ensured. The machine operating panels and their outputs are supplied with +24 V NC.

		Set	Reset
M4179	Control is being shut down	NC	NC
MP4040	Set PLC output after shutdown		
Input:	0: Do not set a PLC output 1: Only after shutdown via Module 9279 2: After shutdown via Module 9279 or soft key		
MP4041	Time after shutdown until setting of the PLC output from MP4044		
Input:	0 to 1000 [s]		
MP4044	PLC output to be set after shutdown		
Input:	Symbolic name of PLC output 0: Cannot be evaluated, results in an error message		



Module 9279 Shut down control (configurable)

PLC Module 9279 carries out the reset function of the NC. Depending on the <Mode> parameter, configuration parameters concerning the shutdown process can be taken into account.

Constraints:

- The functionality of the shutdown delay with PLC markers M4195 and M4197 is not available when shutting down via this module.

Call:

PS B/W/D/K <Mode>
 0: Shut down the control without evaluating the MP configuration for shutting down (MP4040 to MP4042)
 1: Shut down and restart the control
 2: Shut down the control depending on the MP configuration for shutting down (MP4040 to MP4042)

CM 9279

Error code:

Marker	Value	Meaning
M4203	0	Control reset was carried out
	1	Error code in W1022
W1022	2	Mode parameter is invalid
	20	Call was not in a submit/spawn job

Module 9189 Shut down the control

Module 9189 shuts down the control. The PLC is not executable after shutdown. The message windows, which appear during shutdown via soft key, do not appear.

Call:

CM 9189

Error code:

Marker	Value	Meaning
M4203	0	Control was shut down
	1	Error code in W1022
W1022		Module was not called in a spawn job or submit job



Message for power interruption

After the control powers up, the **Power interrupted** message appears.

- ▶ Press the CE key to acknowledge this message and compile the PLC program.

With MP7212 you can suppress this message, e.g. for unattended operation.

MP7212 **Power interrupted message**

Input: 0: Acknowledge the **Power interrupted** message with CE key
 1: **Power Interrupted** message does not appear

8.1.18 Arc end-point tolerance

The iTNC uses the entered NC data to calculate the deviation of the arc radius between the beginning and end of the arc:

- ▶ Enter a tolerance value in MP7431.
If the entered tolerance is exceeded, the error message **CIRCLE END POS. INCORRECT** appears.

MP7431 **Arc end-point tolerance**

Input: 0.0001 to 0.016 [mm]

8.1.19 Limit-switch tolerance for M140 / M150

With MP7432 you can enter a tolerance for the limit switches. The limit switches can then be traversed with M140/M150 by this tolerance without an error message.

Retraction with M140/M150 can still be performed if a limit value has been entered and the axes are still within the tolerance. If the axes are still within the tolerance, they can even be moved outside of the traverse range, but only to another limit switch.

If one or more axes that are already slightly outside the limit-switch range are to be moved to the same limit switch, the axes are not moved. No limit-switch error message appears if the axes are still within the tolerance of the limit switch.

MP7432 **Limit-switch tolerance for M140 / M150**

Input: 0.0001 to 1.0000 [mm]
 0: Limit-switch tolerance off

8.1.20 Radius compensation

A path to be traversed can be increased or decreased by the tool radius by entering "R+" or "R-."

The input dialog is not initiated with the "L" key but directly with the orange axis-direction key. For reasons of compatibility, this function has been retained for point-to-point and straight cut controls.

Example: X + 20 R+ Conversational programming
 G07 X + 20 G49 ISO programming

Paraxially compensated positioning blocks (R+/R-) and radius-compensated positioning blocks (RR/RL) must not be entered in succession.

To avoid erroneous entries:

- ▶ Disable the generation of paraxial positioning blocks by setting MP7246 bit 0.

You can switch off the peripheral milling function (3-D radius compensation with tool orientation) by setting MP7682 bit 10. As a result, only 2-D radius compensation is active even if M128 with RR/RL radius compensation is active.

MP7246 Machine parameter with multiple function

Input: Bit 0 – Paraxial positioning blocks
 0: Generation permitted
 1: Generation disabled

MP7682 Machine parameter with multiple function

Input: Bit 10 – Peripheral milling
 0: Peripheral milling allowed
 1: Peripheral milling inactive

8.1.21 User parameters

You can provide the machine tool operator with easy access to up to 16 machine parameters known as user parameters. He can then call them through the MOD function by simply pressing the USER PARAMETER soft key.

- ▶ In MP7330.x, enter the numbers of the machine parameters that you wish to make available.

Example:

If MP7230.1 should be the first available user parameter:

- ▶ Enter the input value 7230.01 in MP7330.0.

If the user selects a user parameter, a message appears on the screen. You can specify this message:

- ▶ In the system file OEM.SYS, enter the name of the PLC dialog message file with the command **PLCDIALOG=**.
- ▶ In MP7340.x enter the line number of the PLC dialog message to be displayed.

MP7330.0-15 Specification of user parameters 1 to 16

Input: 0 to 99999.00 (no. of the user parameter)

MP7340.0-15 Dialog messages for user parameters 1 to 16

Input: 0 to 4095 (line number of the PLC dialog message file)

8.1.22 Code numbers

You can enter certain code numbers through the MOD function. With these code numbers you can activate certain functions.

The following code numbers have a fixed meaning:

Code number	Function
95148	Select machine parameter file
807667	Select the PLC mode
857282	Resets the operating times
75368	Automatic offset adjustment
123	Calls machine parameters that are accessible to the user.
531210	Delete M0 to M999 and B0 to B127
688379	Oscilloscope
555343	FN17: Overwrite system data FN25: Overwrite datum
NET123	Ethernet settings
LOGBOOK	Read out the log
FAILTEST	Simulate the internal emergency stop
SIK	Option menu
KINEMATIC	Selection window for the machine kinematics
SETUP	Update the NC software

The code of the entered code number is entered in double word D276. You can evaluate this code and define your own functions for code numbers, or disable fixed code numbers.

		Set	Reset
D276	Code of the code number last entered via MOD	NC	NC

Module 9272 Code number in PLC string

The module copies the most recently entered code number as a string into a PLC string.

Constraints:

- The NC sets Marker M4629 when a code number is entered. The marker needs to be reset by the PLC program.
- Purely numerical, unsigned code numbers continue to be entered into word D276 and are additionally available to the module as strings.
- If an empty string or not only numbers are entered, the value -1 is entered into word D276. The code number is then only available via Module 9272.
- If no code number has ever been entered after a start-up, a string with a length of 0 is returned.

Call:

PS B/W/D/K <Mode>
0: Read code number in PLC string

PS B/W/D/K <PLC string number>

PS B/W/D/K <Additional parameters>
No evaluation as yet, but must be programmed

CM 9272

Error code:

Marker	Value	Meaning
M4203	0	Function performed
	1	Error code in W1022
W1022	2	Invalid mode programmed
	3	Invalid PLC string address programmed



8.1.23 Programming station mode

You can switch the control into programming-station mode with MP7210. This way the control can be used as a simple programming station. No drives are enabled.

In this setting NC programs cannot be executed. You can only create and test NC programs.

You can select whether the PLC should be active. In addition, you can activate the emergency-stop loop if the PLC is active. The emergency-stop loop must already be connected correctly (X41/34 and X42/4) in order to switch the control on correctly.

If a programming-station mode is active (MP7210 > 0), this is shown in the first line of the main menu of the **PLC Programming** operating mode via "**MP7210 = x.**"

MP7210	Programming station
Input:	0: Controlling and programming
	1: Programming station with PLC active
	2: Programming station with PLC inactive
	3: Programming station with PLC and emergency stop both active (X41/34 and X42/4)

8.1.24 Color settings

The colors of the display unit can be defined by machine parameters.

The following color settings **cannot** be changed:

- HEIDENHAIN logo after machine switch-on (standard color)
- Error message for invalid machine parameters (red)
- Blinking error message (red)
- Plan view in the graphic display (blue)
- Cursor (inverse)

You define the desired color by mixing the basic colors red, green and blue. Every basic color has 256 difference stages of intensity. The input values for color settings are byte-oriented. HEIDENHAIN recommends hexadecimal input.

Color	Red		Green		Blue	
Adjustment	Rough	Fine	Rough	Fine	Rough	Fine
HEX ranges	0 to F					
Input for yellow: \$0...	3	9	3	9	0	0

The colors can also be poorly adjusted (e.g. red error message on red background). HEIDENHAIN therefore supplies the controls with a standard color setting, which is suggested by the control during creation of the MP list.

In controls with HSCI, the screen saver is configured using the XFCE window manager or through the screen saver settings in the taskbar at the bottom of the screen:

- ▶ Make the taskbar at the bottom of the screen visible by moving the mouse pointer over it.
- ▶ Press the green HEIDENHAIN button to open the JH menu.
- ▶ Select the Settings menu item.
- ▶ Select the Screen Saver menu item.

If a (new) error message occurs, the screen saver will be exited automatically.

MP7350 Window frames

MP7351 Error messages

- MP7351.0 Priority 0 (error)
- MP7351.1 Priority 1 (warning)
- MP7351.2 Priority 2 (information)

MP7352 "Machine" operating mode display

- MP7352.0 Background
- MP7352.1 Text for operating mode
- MP7352.2 Dialog

MP7353 "Programming" operating mode display

- MP7353.0 Background
- MP7353.1 Text for operating mode
- MP7353.2 Dialog

MP7354 "Machine" program text display

- MP7354.0 Background
- MP7354.1 General program text
- MP7354.2 Active block
- MP7354.3 Comments and unused machine parameters in the machine parameter file
- MP7354.4 Background of inactive windows

MP7355 "Programming" program text display

- MP7355.0 Background
- MP7355.1 General program text
- MP7355.2 Active block
- MP7355.3 Comments and unused machine parameters in the machine parameter file
- MP7355.4 Background of inactive windows



MP7356	Status window and PLC window
MP7356.0	Background
MP7356.1	Axis positions in the status display
MP7356.2	Status display other than axis positions
MP7357	"Machine" soft-key display
MP7357.0	Background
MP7357.1	Text color
MP7357.2	Inactive soft-key row
MP7357.3	Active soft-key row
MP7358	"Programming" soft-key display
MP7358.0	Background
MP7358.1	Text color
MP7358.2	Inactive soft-key row
MP7358.3	Active soft-key row
MP7360	Graphics: 3-D view
MP7360.0	Background
MP7360.1	Top surface
MP7360.2	Front face
MP7360.3	Text display in the graphics window
MP7360.4	Lateral face
MP7360.5	Lowest point of blank form
MP7360.6	Highest point of blank form (below surface)
MP7361	Graphics: Projection in three planes
MP7361.0	Background
MP7361.1	Top view
MP7361.2	Front and side view
MP7361.3	Axis cross and text in the graphic display
MP7361.4	Cursor
MP7362	Additional text display in the graphic window and pocket calculator
MP7362.0	Background of graphic window and pocket calculator
MP7362.1	Background of status display and keys of the pocket calculator
MP7362.2	Status symbols and symbols of the pocket calculator (c in "cos")
MP7362.3	Status values and texts of the pocket calculator (os in "cos")
MP7362.4	Color of the unselected tabs in the graphics window
MP7362.5	AFC tab – Background color
MP7362.6	AFC tab – Color of actual override factor
MP7362.7	AFC tab – Color of actual spindle load
MP7363	Programming graphics
MP7363.0	Background
MP7363.1	Resolved contour
MP7363.2	Subprograms and frame for zooming
MP7363.3	Alternative solutions
MP7363.4	Nonresolved contour
MP7363.5	Rapid traverse movements

- MP7364 Color of the help illustrations for cycles**
 MP7364.0-6 Colors 1 to 7 of the graphic program used
 MP7364.7 Line color (color 8 of the graphic program)
 MP7364.8 Color for highlighted graphic elements if defined in the help illustration
 MP7364.9 Background
- MP7365 Oscilloscope**
 MP7365.0 Background
 MP7365.1 Grid
 MP7365.2 Cursor and text
 MP7365.3 Selected channel
 MP7365.4-9 Channel 1 to 4
- MP7366 Pop-up window (HELP key, pop-up menus etc.)**
 MP7366.0 Background
 MP7366.1 Text or foreground
 MP7366.2 Active line
 MP7366.3 Title bar
 MP7366.4 Scroll-bar field
 MP7366.5 Scroll bar
 MP7366.6-14 Reserved
- MP7367 Large PLC window**
 MP7367.0 Background
 MP7367.1-7 Colors 1 to 7 (Color 8: MP7350)
 MP7367.8-14 Colors 9 to 15
- MP7368 Calculator**
 MP7368.0 Background
 MP7368.1 Background of displays and keys
 MP7368.2 Key texts ("os" in "cos")
 MP7368.3 Key symbols
- MP7369 Directory tree in PGM MGT**
 MP7369.0 Text background
 MP7369.1 Text
 MP7369.2 Text background of the active folder
 MP7369.3 Line color of the tree structure
 MP7369.4 Folders
 MP7369.5 Drives
 MP7369.6 Text background of the heading in the browser window
- MP7370 Small PLC window**
 MP7370.0 Background
 MP7370.1-15 Colors 1 to 15
- MP7371 Status window and PLC window**
 MP7371.0 Background of window
 MP7371.1 Background of elements
 MP7371.2 Values except positions
 MP7371.3 Positions



MP7375	smarT.NC
MP7375.0	Background: Forms
MP7375.1	Background: Unselected tab
MP7375.2	Background: Tree view and input fields
MP7375.3	Background: Inactive input field
MP7375.4	Background: Help graphic
MP7375.5	Cursor: Tree view and background of current field
MP7375.6	Cursor: Tree view if the input focus is in the form
MP7375.7	Text color: Inactive input field
MP7375.8	Text color: Active input field
MP7375.9	Text color: Radio buttons
MP7375.10	Text color: Inactive label
MP7375.11	Background: Radio/check buttons during "mouse-over"
MP7375.12	PREDEF global data: Background
MP7375.13	PREDEF global data: Text
MP7375.14	Changed global data: Background
MP7375.15	Changed global data: Text
MP7375.16	Tool tip: Background
MP7375.17	Tooltip: Text
MP7375.18	Dialog title: Background
MP7375.19	Dialog title: Text
MP7375.20	Pattern generator: Points of the same height
MP7375.21	Pattern generator: Currently active points
MP7375.22	Pattern generator: Deleted points
MP7375.23	Pattern generator: Hidden points
MP7375.24	Pattern generator: Rectangle for zoom

The standard color setting is shown in the following list:

Machine parameters	Standard setting
MP7350	\$0808080
MP7351.0	\$0FF2020
MP7351.1	\$000FF00
MP7351.2	\$00000FF
MP7352.0	\$0ECECEC
MP7352.1	\$0000000
MP7352.2	\$00000FF
MP7353.0	\$0C0C0C0
MP7353.1	\$0000000
MP7353.2	\$00000FF
MP7354.0	\$0FFFFFF
MP7354.1	\$0000000
MP7354.2	\$00000FF
MP7354.3	\$0A0A0A0
MP7354.4	\$0C8C8C8
MP7355.0	\$0FFFFFF
MP7355.1	\$0000000
MP7355.2	\$00000FF
MP7355.3	\$0A0A0A0
MP7355.4	\$0C8C8C8
MP7356.0	\$0ECECEC
MP7356.1	\$00000FF
MP7356.2	\$00000FF
MP7357.0	\$0C0C0C0
MP7357.1	\$0000000
MP7357.2	\$0000000
MP7357.3	\$00000FF
MP7358.0	\$0C0C0C0
MP7358.1	\$0000000
MP7358.2	\$0000000
MP7358.3	\$00000FF
MP7360.0	\$0AAAAAA
MP7360.1	\$08888F0
MP7360.2	\$00011AA
MP7360.3	\$0FFFFFF
MP7360.4	\$00000FF
MP7360.5	\$0550000
MP7360.6	\$0FFFFFF

Machine parameter	Standard setting
MP7361.0	\$0AAAAAA
MP7361.1	\$00000E8
MP7361.2	\$00000E8
MP7361.3	\$0FF0000
MP7361.4	\$0FF00FF
MP7362.0	\$0ECECEC
MP7362.1	\$0FFFFFF
MP7362.2	\$00000FF
MP7362.3	\$00000FF
MP7362.4	\$0C0C0C0
MP7362.5	\$0ECECEC
MP7362.6	\$000FF80
MP7362.7	\$000AAFF
MP7363.0	\$0ECECEC
MP7363.1	\$00000FF
MP7363.2	\$0FF00FF
MP7363.3	\$000EC00
MP7363.4	\$0FF0000
MP7364.0	\$0DBD3DB
MP7364.1	\$0FF0000
MP7364.2	\$0202020
MP7364.3	\$0000000
MP7364.4	\$00000FF
MP7364.5-6	\$00000FF
MP7364.7	\$0AA0000
MP7364.8	\$000EEEE
MP7364.9	\$0DBD3DB
MP7365.0	\$0FFFFFF
MP7365.1	\$0808080
MP7365.2	\$00000FF
MP7365.3	\$0FF0000
MP7365.4	\$0C08030
MP7365.5	\$000FF00
MP7365.6	\$0FF00FF
MP7365.7	\$00000FF
MP7365.8	\$0FFCF00
MP7365.9	\$000CFFF



Machine parameter	Standard setting
MP7366.0	\$0ECECEC
MP7366.1	\$0000000
MP7366.2	\$00000FF
MP7366.3	\$0FF0000
MP7366.4	\$0FFFFFF
MP7366.5	\$0FF0000
MP7366.6	\$0000000
MP7366.7	\$0202020
MP7366.8	\$0404040
MP7366.9	\$0606060
MP7366.10	\$0808080
MP7366.11	\$0A0A0A0
MP7366.12	\$0C0C0C0
MP7366.13	\$0E0E0E0
MP7366.14	\$0FFFFFF
MP7367.0	\$0ECECEC
MP7367.1	\$0000000
MP7367.2	\$0D0D0D0
MP7367.3	\$0A0A0A0
MP7367.4	\$0808080
MP7367.5	\$0404040
MP7367.6	\$0202020
MP7367.7	\$000FF9C
MP7367.8	\$0FFFF7C
MP7367.9	\$0FF0000
MP7367.10	\$000FF00
MP7367.11	\$0001C5C
MP7367.12	\$000FFFF
MP7367.13	\$00000FF
MP7367.14	\$0FF00FF

Machine parameter	Standard setting
MP7368.0	\$0ACACAC
MP7368.1	\$0FFFFFF
MP7368.2	\$00000FF
MP7368.3	\$0FF0040
MP7369.0	\$0ECECEC
MP7369.1	\$0000000
MP7369.2	\$00000FF
MP7369.3	\$0000000
MP7369.4	\$0FF6000
MP7369.5	\$0FF0040
MP7369.6	\$0FF0000
MP7370.0	\$0C0C0C0
MP7370.1	\$0000000
MP7370.2	\$0D0D0D0
MP7370.3	\$0A0A0A0
MP7370.4	\$0808080
MP7370.5	\$0404040
MP7370.6	\$0202020
MP7370.7	\$000FF9C
MP7370.8	\$0FFFF7C
MP7370.9	\$0FF0000
MP7370.10	\$000FF00
MP7370.11	\$0001C5C
MP7370.12	\$000FFFF
MP7370.13	\$00000FF
MP7370.14	\$0FF00FF
MP7370.15	\$0FFFFFF



Machine parameter	Standard setting
MP7371.0	\$0ECECEC
MP7371.1	\$0FFFFFF
MP7371.2	\$00000FF
MP7371.3	\$00000FF
MP7375.0	\$0E8E8E8
MP7375.1	\$0C0C0C0
MP7375.2	\$0FFFFFF
MP7375.3	\$0D0D0D0
MP7375.4	\$0D2D2D2
MP7375.5	\$00022FF
MP7375.6	\$0A0E0FF
MP7375.7	\$0000000
MP7375.8	\$0FFFFFF
MP7375.9	\$00000FF
MP7375.10	\$0A0A0A0
MP7375.11	\$0FF0000
MP7375.12	\$000FF00
MP7375.13	\$0000000
MP7375.14	\$0FF0000
MP7375.15	\$0000000
MP7375.16	\$0FFFFC0
MP7375.17	\$0000000
MP7375.18	\$0FF0000
MP7375.19	\$0000000
MP7375.20	\$000FA00
MP7375.21	\$00000C8
MP7375.22	\$0800000
MP7375.23	\$0FF6432
MP7375.24	\$07B342D



8.1.25 Graphic display

In the graphics window you can view the following graphics:

- Test run graphics
- Parallel graphics
- Programming graphics
- Help graphic

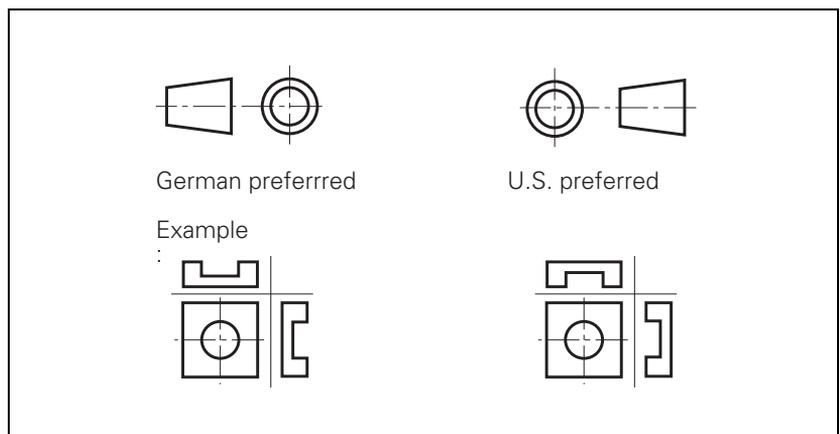
For the test graphics and parallel graphics you can choose one of three display modes:

- Projection in three planes
- Plan view
- 3-D view

Projection in three planes

The display in three planes can be shown in 1st-angle projection as preferred in Germany or in the American-style 3rd-angle projection:

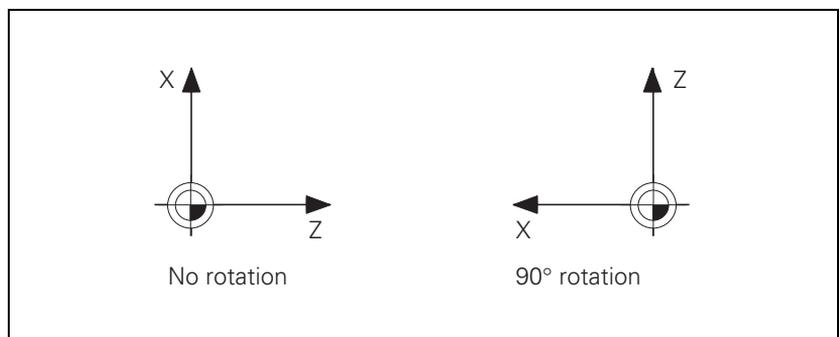
- ▶ Select the type of projection with MP7310, bit 0.



Rotation of the coordinate system

You can rotate the coordinate system for graphic display by $+90^\circ$ if, for example, the Y axis is defined as tool axis.

- ▶ Select the angle of rotation with MP7310, bit 1.



Graphic display for datum shift

In an NC program you can program several BLK forms in succession.

After datum shift with Cycle 7, the shift can be interpreted to apply also to subsequent blank forms:

- ▶ Define the BLK form shift in bit 2 of MP7310.

Position of the cursor

In the display in three planes you can display the position of the cursor:

- ▶ Switch this function on with MP7310, bit 3.

MP7310

Graphic display mode

Format: %xxxxxxx

Input:

Bit 0 – Projection in three planes:

0: German-preferred projection

1: US-preferred projection

Bit 1 – Rotating the coordinate system in the working plane by 90°

0: No rotation

1: Rotation by +90°

Bit 2 – BLK form after datum shift:

0: Shifted

1: Not shifted

Bit 3 – Display of the cursor position:

0: Not displayed

1: Displayed

Bit 4 – Reserved

Bit 5 – 3-D graphics during program test

0: 2.5-D and 3-D

1: 2.5-D

Bit 6 – Stock removal with an inclined tool

0: Not active

1: Active

Bit 7 – Exact evaluation of the column LCUTS (cutting length) from the TOOL.T table in order to display special tools (e.g. saw blade).

0: Free evaluation

1: Exact evaluation for special tools

8.1.26 Special characters

To enter special characters, use the following key combinations:

Key combination	Special characters
SHIFT + "	'
SHIFT + &	@
SHIFT + ([
SHIFT +)]
SHIFT + -	-
SHIFT + /	\
SHIFT + !	
SHIFT + ^	~
SHIFT + #	ESC
SHIFT + RET	<Form feed>
SHIFT + SPACE	Switch between uppercase and lowercase



8.1.27 iTNC character set

Small characters

No.	Character
01 - 1D	
1E	
1F	
20	<SPACE>
21	!
22	"
23	#
24	\$
25	%
26	&
27	'
28	(
29)
2A	*
2B	+
2C	,
2D	-
2E	.
2F	/
30	0
31	1
32	2
33	3
34	4

No.	Character
54	T
55	U
56	V
57	W
58	X
59	Y
5A	Z
5B	[
5C	\
5D]
5E	^
5F	_
60	'
61	a
62	b
63	c
64	d
65	e
66	f
67	g
68	h
69	i
6A	j
6B	k

No.	Character
AD	
AE	®
AF	
B0	°
B1	
B2	²
B3	³
B4	
B5	μ
B6	
B7	
B8	,
B9	¹
BA	^e
BB	»
BC	¼
BD	½
BE	¾
BF	¿
C0	À
C1	Á
C2	Â
C3	Ã
C4	Ä

No.	Character
E3	ã
E4	ä
E5	å
E6	æ
E7	ç
E8	è
E9	é
EA	ê
EB	ë
EC	ì
ED	í
EE	î
EF	ï
F0	ð
F1	ñ
F2	ò
F3	ó
F4	ô
F5	õ
F6	ö
F7	
F8	ø
F9	ù
FA	ú

Medium characters

No.	Character
01 - 1D	
1E	
1F	
20	<SPACE>
21	!
22	"
23	#
24	\$
25	%
26	&
27	'
28	(
29)
2A	*
2B	+
2C	,
2D	-
2E	.
2F	/
30	0
31	1
32	2
33	3
34	4
35	5
36	6
37	7

No.	Character
54	T
55	U
56	V
57	W
58	X
59	Y
5A	Z
5B	[
5C	\
5D]
5E	^
5F	_
60	'
61	a
62	b
63	c
64	d
65	e
66	f
67	g
68	h
69	i
6A	j
6B	k
6C	l
6D	m
6E	n

No.	Character
AD	
AE	®
AF	
B0	°
B1	
B2	²
B3	³
B4	
B5	μ
B6	
B7	
B8	„
B9	¹
BA	ª
BB	»
BC	¼
BD	½
BE	¾
BF	¿
C0	À
C1	Á
C2	Â
C3	Ã
C4	Ä
C5	
C6	Æ
C7	Ç

No.	Character
E3	ã
E4	ä
E5	å
E6	æ
E7	ç
E8	è
E9	é
EA	ê
EB	ë
EC	ì
ED	í
EE	î
EF	ï
F0	ð
F1	ñ
F2	ò
F3	ó
F4	ô
F5	õ
F6	ö
F7	
F8	ø
F9	ù
FA	ú
FB	û
FC	ü
FD	ý

Large characters

No.	Character
01 - 1F	
20	<SPACE>
21	!
22	"
23	#
24	\$
25	%
26	&
27	'
28	(
29)
2A	*
2B	+
2C	,
2D	-
2E	.
2F	/
30	0
31	1
32	2
33	3
34	4

No.	Character
35	5
36	6
37	7
38	8
39	9
3A	:
3B	;
3C	<
3D	=
3E	>
3F	?
40	@
41	A
42	B
43	C
44	D
45	E
46	F
47	G
48	H
49	I
4A	J
4B	K
4C	L

No.	Character
4D	M
4E	N
4F	O
50	P
51	Q
52	R
53	S
54	T
55	U
56	V
57	W
58	X
59	Y
5A	Z
5B	[
5C	\
5D]
5E	^
5F	_
60	
61	a
62	b
63	c
64	d

No.	Character
65	e
66	f
67	g
68	h
69	i
6A	j
6B	k
6C	l
6D	m
6E	n
6F	o
70	p
71	q
72	r
73	s
74	t
75	u
76	v
77	w
78	x
79	y
7A	z
7B - FF	

8.1.28 Conversational language

The TNC is delivered with all NC-dialog human languages already loaded:

- ▶ In MP7230.0 select the conversational language in which you wish to work.

If the NC dialog messages for the selected language are not on the hard disk, the error message **LANGUAGE LOAD ERROR** appears. You can continue working in the default language English.

You can write your own dialog messages and save them in several languages:

- ▶ Save your dialog messages in permanently defined directories in the PLC partition.

These directories are:

```
PLC:\LANGUAGE\ CHINESE\
                CZECH\
                DANISH\
                DUTCH\
                ENGLISH\
                ESTONIA\ (option 41 – ID 530 184-01)
                FINNISH\
                FRENCH\
                GERMAN\
                HUNGARIA\
                ITALIAN\
                KOREAN\ (option 41 – ID 530 184-01)
                LATVIAN\ (option 41 – ID 530 184-011)
                LITHUANIAN\ (option 41 – ID 530 184-01)
                NORWEGIAN\ (option 41 – ID 530 184-01)
                POLISH\
                PORTUGUE\
                ROMANIAN\ (option 41 – ID 530 184-01)
                RUSSIAN\
                SLOVAK\ (option 41 – ID 530 184-01)
                SLOVENIAN\ (option 41 – ID 530 184-01)
                SPANISH\
                SWEDISH\
                TURKISH\ (option 41 – ID 530 184-01)
```

- ▶ With MP7230.1–3, switch to the desired language.

You can store PLC dialog message files, PLC error message files, and help files with identical file names in the different languages and in UNICODE (e.g. with PLCtext V3.1):



Note

If texts from these files are also used in the PLC window, they must use UTF8-coding.

- ▶ In the system file OEM.SYS, enter only the file names with the commands **PLCDIALOG =** and **PLCERROR =**. The NC looks for the paths given in MP7230.1 or MP7230.2. The entry behind **MODEHELP =** is overwritten with the selected path whenever MP7230.3 is changed.

The PLC texts for PLC soft keys, FN14 error messages (UTF8 code in ERROR.A or FN14 file) and PLC mask files (ERROR.A or DIALOG.A) can also be displayed as UTF8-encoded texts (e.g. for Asian languages). For the FN14 error messages and the PLC mask files, the UTF8 code must then exist in the files stated.

PLC texts that are to be displayed in the PLC status window may also be UTF8-encoded.

Dialogs for user parameters and soft-key descriptions for the HR 420 cannot be displayed in Asian languages. If such dialogs are used, the English language is displayed instead.

The ASCII characters of the values 0xEF, 0xBB and 0xBF serve to indicate the UTF8 code. These three characters must be in the correct sequence as the first three characters in the appropriate file, and only serve to indicate the UTF8 code.

MP7230 Switching the conversational language

Input:	0: English
	1: German
	2: Czech
	3: French
	4: Italian
	5: Spanish
	6: Portuguese
	7: Swedish
	8: Danish
	9: Finnish
	10: Dutch
	11: Polish
	12: Hungarian
	13: Reserved
	14: Russian (Cyrillic characters)
	15: Chinese (simplified)
	16: Chinese (traditional)
	Additional dialog language – 17: Slovenian (option 41)
	18: Norwegian (option 41)
	19: Slovak (option 41)
	20: Latvian (option 41)
	21: Korean (option 41)
	22: Estonian (option 41)
	23: Turkish (option 41)
	24: Romanian (option 41)
	25: Lithuanian (option 41)
MP7230.0	NC conversational language
MP7230.1	PLC conversational language (user parameters), soft keys for OEM cycles
MP7230.2	PLC error messages
MP7230.3	Help files



Module 9074 Load texts from error/dialog files into a PLC string

Module 9074 is used to copy texts from error files and dialog files (ERROR.A, DIALOG.A) to a PLC string directly and without coding.

Condition:

- Due to the file access the module must be called from a submit job or spawn job.
- The module returns the number of loaded characters (string length). With UTF-8 texts this value may be larger than the number of visible characters. Also, with UTF-8 texts the indexed access to individual characters in the string may not be correct in some cases.

Call:

```
PS    K/B/W/D  <PLC string>
        Source or constant string (e.g. S"D012")
PS    K/B/W/D  <PLC string>
        Target
CM    9074
PL    B/W/D    <Number of characters in target string>
```

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	11	Invalid address of the source string or target string
	20	Module was not called in a spawn job or submit job

Example: Transmission of text 13 from the error file to PLC string 54

```
PS    S"#E013"           ; Source: Line 13 from error.a
PS    K9000              ; Target string: S54
CM    9074
PL    W_STRLEN           ; Number of char in the string
L     M4203
IFT
      L     NP_W1022_Module_Error status
      =     Wxxx
ENDI
```

Additional conversational languages

HEIDENHAIN offers additional conversational languages, which can only be selected in combination with option 41 (ID 530 184-01). If option 41 is activated via **Programming and Editing** > **MOD** key and the keyword **SIK**, then the "additionally" declared dialog language is activated.

Decimal point

- ▶ With MP7280 you specify whether the decimal point will be a comma or a period.

MP7280 Decimal character

Input: 0: Decimal comma
1: Decimal point

8.1.29 Logs

PLC log

In order to make debugging in PLC programs easier, the following markers can be used to record additional PLC module calls and their parameters. The data are written to the log file PLC:\JHPLCLOG\PLCMODULES.LOG.

If one of the three markers is set, this results in the one-time NC error message "Debug functions of the PLC are active." The other log files PLCDEBUG.LOG, MYDEBUG.LOG, VERSION.TXT, VERSION.LOG and MakeCycle.log are contained in the directory PLC:\JHPLCLOG\.

The marker should be set only for debugging purposes. Otherwise, unnecessary access to the hard disk will put a strain on the system. The following events are entered:

- Start of the PLC after switch-on.
- Start and stop of the PLC.
- Errors from PLC modules (this entry is only made if M4753 = 1).
- Run-time errors

		Set	Reset
M4751	Recording of transfer and return parameters of PLC modules that are called in the cyclic PLC program	PLC	PLC
M4752	Recording of transfer and return parameters of PLC modules that are called in submit/spawn processes	PLC	PLC
M4753	Writing errors from PLC modules to the PLC log (PLC:\JHPLCLOG\PLCMODULES.LOG)	PLC	PLC





Note

If the PLC program Basic_54 is used and the message "Debug functions of the PLC are active" occurs, you need to correct the PLC project with respect to M4753 or set MP4020 bit 17, because M4753 is set permanently in this version of the PLC basic program.

More information on the use of M4751, M4752 and M4753:

- The markers may be used for debugging only temporarily and must not be set in the factory default setting of a PLC.
- M4753 is/was set permanently by the PLC basic program Basic_54. MP4020 bit 17 must be set in order to suppress the message without having to modify the PLC program.
- A maximum of 1 MB of data can be written to the file PLCMODULES.LOG; any data exceeding this limit will be copied to .OLD.
- A maximum of 2 MB is available for both the file PLCMODULES.LOG and the file PLCMODULES.LOG.OLD. With the PLC basic program for example, the 2 MB are exhausted some seconds after the power interrupted message has been acknowledged if all debug markers are set.
- During debugging in the PLC program, it is advisable to position the markers close (if possible) to the code position to be examined, and not for the complete PLC scan.
- On OEM machines, PLC debug code that has been output for debugging must afterwards be replaced by the original code again.

Diagnostics log

If M4754 is set, internal diagnostic information is entered in the log MYDEBUG.LOG. The marker should be set only for debugging purposes. Otherwise, unnecessary access to the hard disk will put a strain on the system.

		Set	Reset
M4754	Writing diagnostic information in MYDEBUG.LOG	PLC	PLC





Note

Writing to OEM logs must only take place in worthwhile intervals, since under circumstances the processing time could be affected negatively, and the hard disk written to unnecessarily.

Module 9277 Write data into the OEM log

With Module 9277 the PLC can write data into a specific OEM log. Up to eight OEM logs can be used at the same time. The module can be called from a cyclic PLC program or from a spawn job or submit job. The string for the log entry may contain two wildcards (data1 and data2). Only wildcards that occur are replaced. The output format is controlled through the entry %d for integers or the entry %f for floating point numbers with three decimal places.

Alternatively, you can define the number of decimal places with %.1f to %.6f.

Example of a string for the log entry: **S"data1: %.2f data2: %d"**

If the maximum log size of 1 MB is exceeded, the log is copied to <name>.LOG.OLD and a new log with the same name is created. Once the logs have been called, they remain open until the control is shut down.

The transferred OEM log can be closed with the switch for additional entries (bit 2). This makes it possible to save log files on the hard disk. In this case, all parameters other than <Path with file name> are meaningless. Attempts to close a log multiple times will not lead to a reaction or error message.

Call:

```
PS   B/W/D/K/S<Path with file name (without .LOG extension)>
PS   B/W/D/K/S<String with placeholder for log entry>
PS   B/W/D/K  <Value for data1>
PS   B/W/D/K  <Value for data2>
PS   B/W/D/K  <Switch for additional entries>
      Bit 0 = 0/1: Entry without/with time stamp
      Bit 1 = 0/1: Entry without/with PLC cycle counter
      Bit 2 = 1: Log (= path with file name) is being closed
```

CM 9277

Error code:

Marker	Value	Meaning
M4203	0	Data written into OEM log
	1	Error code in W1022
W1022	2	Invalid string number or invalid string
	22	Message cannot be transmitted



Standard log

The standard log serves as a troubleshooting aid. There are 4 MB of memory available for this purpose.

All entries in the log are marked with the momentary date and time.

You can read out the standard log in two ways:

- ▶ After entering the code word LOGBOOK, enter the path and name of an ASCII file and the time and date from which the log should record, as well as the time and date up to which it should record. After that, an ASCII file is generated and opened with the log entries.
- ▶ The PC software programs PLCdesign, TNCremo, and TNCremoNT offer you several functions for reading out the log.

Entry		Description
RESET		Powering up the control
BERR		Blinking error message
BREG		Register contents with a blinking error message
ERR		Error messages <ul style="list-style-type: none"> ■ P: PLC error message with the line number in the PLC error text file ■ N: NC error message with number ■ Power fail interrupt: Control was switched off by a POWERFAIL ■ Result of the file system test (in case the control was not properly shut down previously)
ERR	dosfsck -a	Check of the PLC: and TNC: hard-disk partitions if the control was not shut down correctly
INFO	"xxx"	"xxx": Name of the control process that enters the information in the log <ul style="list-style-type: none"> ■ PLC ■ SYS ■ MAIN ■ REMO ■ CTRL ■ SMARTNC ■ SMARTNC_LDF ■ BDEHAN ■ SOKY ■ GEO ■ PYTHON ■ FIXTURE ■ FILEMAN
INFO	CTRL KINEMATIC	Active tool with tool number, radius (R =) and length (L =)
INFO	CTRL collision	Status of collision monitoring in the Manual and Automatic operating modes
INFO	CTRL REG AXIS xx START REF=yy	Position yyyy of axis xx at start of movement

Entry		Description
INFO	CTRL REG AXIS xx STOP REF=yy	Position yyyy of axis xx at stop of movement
INFO	MAIN ERRCLEARED	Acknowledgment of an error message
INFO	MAIN ERR_RECURED	Error message entered several times
INFO	MAIN ENDAT	Entry for the position upon switch-on, if the switch-off and switch-on positions of an EnDat encoder do not match
INFO	MAIN RUNPROC	Status information about the current process
KEY		Key strokes
INFO	MAIN SOFTKEY	Path with associated image file of a pressed soft key
STIB ^a	ON	Control-in-operation on
	OFF	Control-in-operation off
	BLINK	Control-in-operation symbol blinking
INFO	MAIN START	Type of control, NC software and valid Feature Content Level (FCL)
INFO	MAIN FILE DEL	Faulty files on the hard disk, to be erased during booting
INFO	MAIN HDD	Hard disk designation
INFO	MAIN DSP	ID number of the active controller software
INFO	MAIN CYCLES	Test results for fixed cycles and touch probe cycles
INFO	MAIN KEYSOURCE	Source of the keystrokes <ul style="list-style-type: none"> ■ KEYBOARD ■ PLC ■ PLCNCSTART ■ HANDWHEEL ■ LSV2
INFO	MAIN KINEMATIC	Listing of the definition tables with collision objects that are monitored for collision with option 40, DCM.

a. Control-in-operation symbol = " * " in the screen display

Entry		Description
INFO	MAIN PGM	Started NC program or NC macro
INFO	MAIN LINE	Line number of the running NC program or NC macro
INFO	MAIN PGMEND	<p>Information about the program end in program run</p> <p>Byte 0/1 00 01 Emergency stop 00 02 Positioning error 00 03 Programmed stop 00 04 Block end in single block</p> <p>mode</p> <p> 00 05 Geometry error 00 06 END PGM, M02 00 07 TNC STOP button 00 08 Data transmission error (RS-422/RS-232)</p> <p>Byte 2/3 xx xx Internal error class Bytes 4 ... 7 xx xx xx xx Internal error code Bytes 8 ... 11 xx xx xx xx Line number Byte 12 ... 15 xx xx xx xx Reserved</p> <p>In addition, when a program is stopped by an error message, the following information is entered in the log: NC program, line number, actual position, datum, datum shifts, tool number</p>
INFO	MAIN MACEND	<p>Information about the end of an NC macro</p> <p>Byte 0/1 00 01 Emergency stop 00 02 Positioning error 00 03 Programmed stop 00 04 Block end in single block</p> <p>mode</p> <p> 00 05 Geometry error 00 06 END PGM, M02 00 07 TNC STOP button 00 08 Data transmission error (RS-422/RS-232)</p> <p>Byte 2/3 xx xx Internal error class Byte 4...7 xx xx xx xx Internal error code</p>



Entry		Description	
INFO	MAIN PATH	PLCEDIT	File for PLC editor
		NCEDIT	File for NC Editor
		RUNPGM	Main program for program run
		RUNPALET	Pallet table for program run
		RUNDATUM	Datum table for program run
		RUNTOOL	Tool table for program run
		RUNTCH	Pocket table for program run
		SIMPGM	Main program for program test
		SIMDATUM	Datum table for program test
		SIMTOOL	Tool table for program test
		RUNBRKPGM	Stopping point for block scan
		SIMBRKPGM	Stopping point for program test
		RUNPRINT	Path for FN15: PRINT for program run
		SIMPRINT	Path for FN15: PRINT for program test
		MDIPGM	File for positioning with manual data input
		NCFMASK	Mask for file management in the NC area
		PLCFMASK	Mask for file management in the PLC area
		EASYDIR	Paths for standard file management
		TCHPATH	Datum table for manual measurement
		SIMTAB	Freely definable table in program test
RUNTAB	Freely definable table in program run		
KINTAB	Active kinematics table		
INFO	MAIN NCEVENT	Entries via FN38: SEND from the Program Run, Full Sequence or Program Run, Single Block operating modes	
	MAIN NCTEVENT	Entries via FN38: SEND from the Test Run operating modes	
INFO	MAIN BUTTON	Recording of mouse movements/buttons <ul style="list-style-type: none"> ■ "x": P = Press, R = Release ■ "y": L = Left button, R = Right button ■ "z": Key pressed simultaneously N = None, S = SHIFT, C = CTRL, A = ALT, W = Windows, L = Left button, M = Middle button, R = Right button ■ "a": Position of the mouse pointer in X ■ "b": Position of the mouse pointer in Y 	
	MOUSE "x" "y" "z" "a" "b"		
INFO WARNING ERROR	PLC <log identifier>	Entries through PLC modules 9275 and 9276	
INFO	SYS	SHUTDOWN	Control was shut down
		REBOOT-TNC	Control was rebooted (automatically)
		REBOOT-BIOS	Control was rebooted (automatically)

Entry		Description	
INFO ^a	REMO A_LG	Log in with LSV2 protocol	
	REMO A_LO	Log out with LSV2 protocol	
	REMO Delete	Deletion of a file via the LSV2 protocol	
	REMO Receive	Reception of a file via the LSV2 protocol	
	REMO C_LK	LSV2 protocol: Locking and releasing the keyboard; the key codes between locking and releasing are sent via LSV2 protocol	
INFO	SOKY	KEYSOURCE:	Source of information on a key: <ul style="list-style-type: none"> ■ KEYBOARD ■ PLC ■ PLCNCSTART ■ HANDWHEEL ■ LSV2 ■ KEYLOGGER
		PROCESS:	Name of the target process to which the key information is sent.
		IDENT:	Control-internal name of the soft key
		SOFTKEY:	Name of the BMX image file of the soft key
		OVERLAY:	Current overlay number of the soft key
		Autorepeat ...	Status information on the Autorepeat function (start, stop, waiting times, ...)
		Key Logger:	Status information on key recording (start, stop, repeat, ...)
INFO	SYS WINEVENT FILEMAN.STARTUP.READY	File manager (PGM MGT) started	
ERROR	"xxx": "yyy"	"xxx": Name of the control process that enters the information in the log <ul style="list-style-type: none"> ■ PLC ■ SYS ■ MAIN ■ REMO ■ CTRL ■ SMARTNC ■ SMARTNC_LDF ■ BDEHAN ■ SOKY ■ GEO ■ PYTHON ■ FIXTURE ■ FILEMAN "yyy": Optional information: Name of the process causing the error	

- a. For test purposes, all LSV-2 telegrams can be entered in the log. This function must be enabled with the LSV-2 TELEGRAM OFF/ON soft key.

Writing to the log

The log can be written to by the PLC or from an NC program.

Write data **from an NC program** into the log:

- ▶ Use the **FN38: SEND** function in the NC program. It is available after you have entered the code number 555343.

Example for programming a block with **FN38: SEND**:

```
...  
FN38: SEND /"Q-Parameter Q1: %f Q2: %f" /+Q1 /+Q2  
...
```

Entry in the log from the **Program Run, Single Block** or **Program Run, Full Sequence** operating mode:

```
INFO:          MAIN NCEVENT          <Date and time>  
          Q parameter Q1: <Value Q1> Q2: <Value Q2>
```

Entry in the log from the **Test Run** operating mode:

```
INFO:          MAIN NCTEVENT        <Date and time>  
          Q parameter Q1: <Value Q1> Q2: <Value Q2>
```

Write data for diagnostic purposes **from the PLC** into the log:

- ▶ With Module 9275 you can write ASCII data into the log.
- ▶ With Module 9276 you can write the contents of the operands into the log.



Note

Do not use Modules 9275 and 9276 in the PLC program as shipped. Instead, use them only for debugging. Otherwise the processing times could be increased and the hard disk could be written to unnecessarily, so that the log can no longer fulfill its function of recording keystrokes and error messages.

Additional log

A second log is automatically created parallel to the standard log. It is in the battery-buffered RAM, and so enables access to the log data on the control even without a hard disk. Enter the code number **LOGBOOK1** to output it and save it as an ASCII file, just as the standard log.

This log is not available for the programming station.

Module 9275 Write ASCII data into the log

With Module 9275 you can write ASCII data into the log. For later editing the entry can be given an identifier.

Call:

```
PS   B/W/D/K/S<Log entry>
      -1: No entry
PS   B/W/D/K/S<Log identifier>
      -1: No entry
PS   B/W/D/K <Priority>
      0: Information
      1: Warning
      2: Error
```

CM 9275

Error code:

Marker	Value	Meaning
M4203	0	Entry was written
	1	Error code in W1022
W1022	1	Invalid priority
	2	Invalid string number or invalid immediate string
	12	No string end identifier
	20	Module was not called in a spawn or submit job



Module 9276 Write operand contents into the log

With this module you can write the contents of operands (inputs, outputs, markers, bytes, words, double words, timers, counters) into the log. For later editing the entry can be given an identifier.

Call:

PS B/W/D/K <Identifier for operand name>

0: M (marker)

1: I (input)

2: O (output)

3: C (counter)

4: T (timer)

5: B (byte)

6: W (word)

7: D (double word)

PS B/W/D/K <Address of the first operand>

PS B/W/D/K <Number of operands>

PS B/W/D/K/S<Log identifier>

-1: No entry

PS B/W/D/K <Priority>

0: Information

1: Warning

2: Error

CM 9276

Error code:

Marker	Value	Meaning
M4203	0	Entry was written
	1	Error code in W1022
W1022	1	Invalid priority
	2	Invalid identifier for operand name
	3	Invalid first operand address
	4	Sum of first operand address and number of operands invalid
	5	Address is not a word/double-word address
	12	No string end identifier
	20	Module was not called in a spawn or submit job
	36	Entry in the log was truncated after 210 characters

8.1.30 Diagnostic functions

The iTNC 530 provides the diagnostic functions for error diagnosis, which are available after pressing the DIAGNOSIS soft key (see table below).

To call the diagnostic functions:

- ▶ While in the **Programming and Editing** operating mode, press the MOD key.
- ▶ Press the DIAGNOSIS soft key.

The following diagnostic functions are available:

Soft key	Soft key	Function
		After pressing this soft key, you can test various HSCI and Profibus settings, provided that you are using one of the two bus systems.
		After pressing this soft key, you can test various Profibus settings, provided that you are using a Profibus system.
		After pressing this soft key, you can test various HSCI settings, provided that you are using an HSCI system (, see page 1320).
		Various drive diagnosis functions can be selected after pressing this soft key. Before selecting the diagnostic function, under Supply unit you must select the power supply unit being used, so that signals that are not present are not interpreted as errors.
		The integrated oscilloscope is started.
		The diagnosis tool DriveDiag is opened, see page 1299.
		The following soft keys appear only if the Power Interrupted message was not acknowledged, and if the code number 688379 or 807667 was entered.
		The integrated oscilloscope for commissioning the current controller is opened.
		The file TNC:\herosdiagnose.txt is created after pressing this soft key. HEIDENHAIN uses this file for diagnosis of the operating system.

DriveDiag

The iTNC 530 also features the diagnostic tool "DriveDiag".

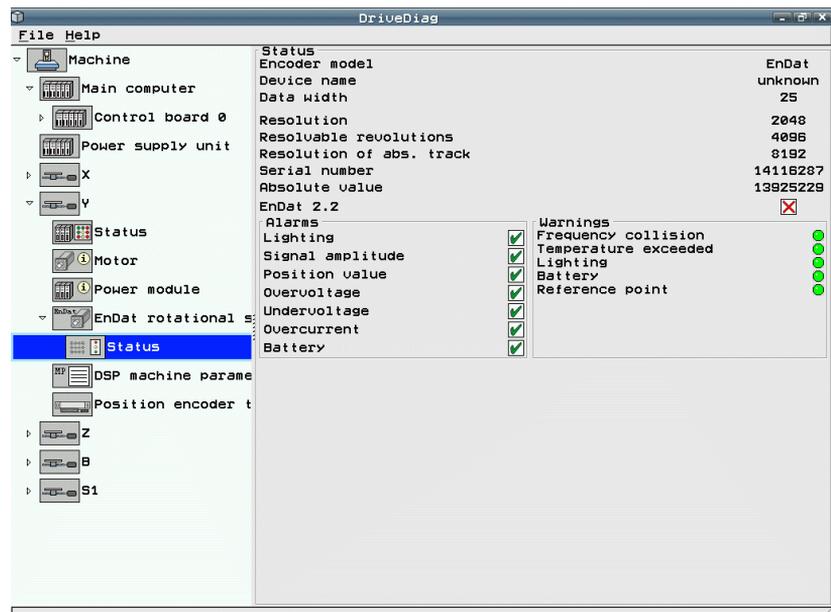
To start DriveDiag:

- ▶ While in the **Programming and Editing** operating mode, press the MOD key.
- ▶ Press the DIAGNOSIS soft key.
- ▶ Press the DRIVE DIAGNOSIS soft key.
- ▶ Press the DRIVE DIAG soft key.

This opens an additional task for the DriveDiag diagnosis tool.

The structure of the drive system is displayed on the left side of the screen in a tree structure. Detailed information on the currently selected component is displayed on the right side of the screen.

To navigate within the tree structure (left side of screen) and the tabs (right side of screen), use the arrow keys. Use the ENT key to move from the left side of the screen to the right side of the screen. Use the END key to move from the right side of the screen back to the left side of the screen. If the cursor is already located on the left side of the screen, you can exit DriveDiag with the END key.



To switch back to the screen of the control or to DriveDiag, press the screen switchover key.

Icon	Icon	Function
		Machine: Higher-level folder containing all components of the complete machine ■ Connection: Internal IP address (127.0.0.1)
		Main computer , see page 1300
		Power supply unit , see page 1306 ■ Information about ID label and status
		Axis (e.g. X , Y , Z , etc.), see page 1308 Higher-level folder containing all axis components

Main computer

Icon	Icon	Function
		Main computer Version tab: ■ Control model: Type of control ■ NC software: Installed NC software version ■ PLC software: PLC software being used
		Main computer Status tab: ■ Status information about various signals, see page 1302

Icon	Icon	Function
		<p>Drive control board x</p> <p>Version tab:</p> <ul style="list-style-type: none"> ■ Speed controller software: Installed speed-controller software version ■ Current-controller software: Installed current-controller software version ■ Hardware code: Additional information for identifying the hardware (only CC 422) ■ Version of additional info: Date and time of installation ■ Degree of support: Degree of support for controller software ■ SG software available: Information about whether the installed software supports functional safety ■ SG software active: Functional safety is activated/deactivated <hr/> <p>Drive control board x</p> <p>Voltages and currents tab:</p> <ul style="list-style-type: none"> ■ Supply voltage +5 V: Current value of voltage [V] ■ DC-link voltage: Current value of voltage [V] ■ DC-link current: Current value of current [A] ■ DSP processor board temp.: Current temperature value on the DSP computer board [°C] ■ Supply voltage +15 V: Current value of voltage [V] ■ Supply voltage -15 V: Current value of voltage [V] ■ Supply voltage +3.3 V: Current value of voltage [V] ■ Auxiliary voltage UL: Current value of voltage [V] ■ Auxiliary voltage UH: Current value of voltage [V] <hr/> <p>Drive control board x</p> <p>Status tab:</p> <ul style="list-style-type: none"> ■ Status information about various signals, see page 1304

Meanings of the signals under "main computer / status"

Signal	Meaning	Colors
External enabling signals		
Acknowledgment: Control is ready (-NE1)	The $\overline{NE1}$ signal (emergency stop input 1, MC) is active if a 0 level is present (low active). For the iTNC 530 the corresponding input is at connector X42/I3 (PLC input), and is looped to the MC as a hardware line.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Signal is not active, enable ■ Red: Signal is active, no enabling
External signals from MC		
Ref. signal of spindle (X30)	For more detailed information, refer to "Input: Spindle Reference Signal" in the Technical Manual.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Signal is active ■ Red: Signal is not active
Trigger signal of TS touch probe (X12)	For more detailed information, refer to "X12: Connecting the Touch Probe" in the Technical Manual.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Signal is active ■ Red: Signal is not active
Battery of TS touch probe (X12)	For more detailed information, refer to "X12: Connecting the Touch Probe" in the Technical Manual.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Signal is not active, warning ■ Red: Signal is active, no warning
TS touch probe is ready (X12)	For more detailed information, refer to "X12: Connecting the Touch Probe" in the Technical Manual.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Signal is active ■ Red: Signal is not active
Trigger signal of TT probe (X13)	For more detailed information, refer to "X13: Connecting the Touch Probe" in the Technical Manual.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Signal is active ■ Red: Signal is not active
TT touch probe ready (X13)	For more detailed information, refer to "X13: Connecting the Touch Probe" in the Technical Manual.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Signal is active ■ Red: Signal is not active



Signal	Meaning	Colors
MC internal signals and status		
"Commissioning Current Controller" mode	The signal is active if the control is in the operating mode for adjusting the current controller.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Operating mode for adjusting the current controller is active ■ Red: Operating mode for adjusting the current controller is not active
Power interruption acknowledged	The signal is active if the Power interrupted message has been acknowledged with the CE key.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Power interruption was acknowledged ■ Red: Power interruption was not acknowledged

Meanings of the signals under "drive control board / status"

Signal	Meaning	Colors
External enabling signals		
Drive enable (-NE2)	The $\overline{NE2}$ signal (emergency stop input 2, CC) is active if a 0-level is present (low active). For the iTNC 530 the corresponding input is at connector X42/I32 (PLC input), and is looped to the CC as a hardware line.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Signal is not active, enable ■ Red: Signal is active, no enabling
Powerfail	The \overline{PF} signal shows the status of the "effective" powerfail signal for the drive controller. The signal is the result of gating the $\overline{PF.PS.ZK}$ (DC-link powerfail) and $\overline{PF.PS.AC}$ (AC fail) signals. The gating process can be defined in the machine parameters and in the PLC.	<ul style="list-style-type: none"> ■ Gray: No information available ■ Green: Enabled, \overline{PF} is inactive (1-level) ■ Red: \overline{PF} is active (0-level): the DC-link voltage has decreased below a permissible (inverter-specific) level or the phase monitoring responded; no enabling
X50 Machine On	No longer relevant in the current HEIDENHAIN controls.	<ul style="list-style-type: none"> ■ Gray: No information available
MC is ready (-WD)	This signal shows that the MC is ready for control. This signal is a possible reason that the power module was switched off via SH1.	<ul style="list-style-type: none"> ■ Gray: No information available ■ Green: Enabled \overline{ME} not active (1-level) ■ Red: Not enabled: $\overline{WD1}$ is active (0-level), the MC's watchdog is not retrigged. This signal is relayed to the inverter as $\overline{SH1}$ ($\overline{SH1}$ also has other signal sources).
Powerfail (DC)	The signal is generated at the inverter, and is led via the supply bus to the drive controller. The input at the drive controller is displayed. Depending on the wiring, either this signal or Powerfail (AC) is relayed on the controller PCB to the powerfail signal.	<ul style="list-style-type: none"> ■ Gray: No information available ■ Green: Enabled: $\overline{\text{Powerfail (DC)}}$ is inactive (1-level) ■ Red: Not enabled: $\overline{\text{Powerfail (DC)}}$ is active (0-level): the DC-link voltage has decreased below a permissible (inverter-specific) level.
Powerfail (AC)	The signal is generated at the inverter, and is led via the supply bus to the drive controller. The input at the drive controller is displayed. Depending on the wiring, either this signal or Powerfail (ZK) is relayed on the controller PCB to the powerfail signal. Powerfail (AC) does not exist for all supply units (e.g. not for UV 130).	<ul style="list-style-type: none"> ■ Gray: No information available ■ Green: Enabled: $\overline{\text{Powerfail (DC)}}$ is inactive (1-level) ■ Red: Not enabled: $\overline{\text{Powerfail (AC)}}$ is active (0-level), phase monitoring responded, at least one power supply phase failed



Signal	Meaning	Colors
Internal enabling signals		
Switch-off (speed 0)	This signal causes all axes to be switched off or locked at 0-level. Thus the drives of all control boards are switched off or locked.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Enabled: –N0 inactive (1-level) ■ Red: Not enabled: –N0 active (0-level); all drives are locked
CC controller ready	If no error is present in the drive controller and the controller unit was started, "ready for control" is reported.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Enabled: CC is ready for control ■ Red: No enabling
Clearable DSP error	Clearable DSP errors are 2nd class errors (such as motor temperature). The CC can only resume control after the error has been cleared (by pressing the CE key).	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Enabled: There is no (clearable) 2nd class error. ■ Red: No enabling: A 2nd class error is present
Watchdog current controller	This signal is activated by the current controller's watchdog. It affects $\overline{SH2}$ on the power module via the PWM interface.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Enabled: Current controller watchdog OK ■ Red: No enabling: Current controller watchdog is active (0-level). No pulse release from the current controller via the PWM interface

Power supply unit

Icon	Icon	Icon	Function
	Power supply unit ID Label , see page 1319 tab: <ul style="list-style-type: none"> ■ Plug and Play: Whether the device is plug-and-play capable ■ Device: Designation of the device ■ Model: Designation of the device type ■ Serial number: Serial number of the device ■ ID number: HEIDENHAIN ID of the device 		
	Power supply unit Status tab: <ul style="list-style-type: none"> ■ Status information about various signals, see page 1302 		

Meanings of the signals under "supply unit / status"

Signal	Meaning	Colors
DC-link voltage	Current value of voltage in [V]	
DC-link current	Current value of current in [A]	
Powerfail	The $\overline{\text{PF}}$ signal shows the status of the "effective" powerfail signal for the drive controller. The signal is the result of gating the $\overline{\text{PF.PS.ZK}}$ (DC-link powerfail) and $\overline{\text{PF.PS.AC}}$ (AC fail) signals. The gating process can be defined in the machine parameters and in the PLC.	<ul style="list-style-type: none"> ■ Gray: No information available ■ Green: Enabled, $\overline{\text{PF}}$ is inactive (1-level) ■ Red: $\overline{\text{PF}}$ is active (0-level): the DC-link voltage has decreased below a permissible (inverter-specific) level or the phase monitoring responded; no enabling
Powerfail (DC)	The signal is generated at the inverter, and is led via the supply bus to the drive controller. The input at the drive controller is displayed. Depending on the wiring, either this signal or Powerfail (AC) is relayed on the controller PCB to the powerfail signal.	<ul style="list-style-type: none"> ■ Gray: No information available ■ Green: Enabled: $\overline{\text{Powerfail (DC)}}$ is inactive (1-level) ■ Red: Not enabled: $\overline{\text{Powerfail (DC)}}$ is active (0-level): the DC-link voltage has decreased below a permissible (inverter-specific) level.
Powerfail (AC)	The signal is generated at the inverter, and is led via the supply bus to the drive controller. The input at the drive controller is displayed. Depending on the wiring, either this signal or Powerfail (DC) is relayed on the controller PCB to the powerfail signal. Powerfail (AC) does not exist for all supply units (e.g. not for UV 130).	<ul style="list-style-type: none"> ■ Gray: No information available ■ Green: Enabled: $\overline{\text{Powerfail (DC)}}$ is inactive (1-level) ■ Red: Not enabled: $\overline{\text{Powerfail (AC)}}$ is active (0-level), phase monitoring responded, at least one power supply phase failed

Signal	Meaning	Colors
DC-link voltage>>	The signal reports the status of the DC-link voltage: Either it is OK or too high. This signal also switches off all power modules (via the unit bus). Possible error causes: <ul style="list-style-type: none"> ■ Missing or faulty braking resistor ■ Excessive braking power 	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: DC-link voltage OK ■ Red: DC-link voltage too high
Temperature	The signal reports the status of the heat sink temperature in the inverter: Either it is OK or too high.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Temperature OK ■ Red: Temperature too high
DC-link current >>	The signal reports the status of the DC-link current: Either it is OK or too high. Both the positive and the negative DC-link currents are evaluated.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: DC-link current OK ■ Red: DC-link current too high
Power supply unit ready	The signal reports the ready status of the supply unit: supply unit ok, main contactor on, or supply unit not ready.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Supply unit OK, main contactor on ■ Red: Power supply unit not ready
Ground fault	The signal reports the status of the leakage current monitoring: Either it is OK or too high or there is a ground fault.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Leakage current OK ■ Red: Leakage current too high or ground fault
Reset from CC to UV	Signal –RES.LE. Reset from controller unit to supply module. Resets the error memory in the supply module.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Reset signal is not active ■ Red: Reset signal is active

X, Y, Z, etc. (axis)

Icon	Icon	Icon	Function
	X, Y, Z, etc. (axis) Higher-level folder containing all axis components		
		Status Speed Controller tab: <ul style="list-style-type: none"> ■ Status information about various signals, see page 1311 	
		Status Position Controller tab: <ul style="list-style-type: none"> ■ Status information about various signals, see page 1314 	
		Status PLC tab: <ul style="list-style-type: none"> ■ Status information about various signals, see page 1315 	
		Motor ID Label , see page 1319 tab: <ul style="list-style-type: none"> ■ Plug and Play: Whether the device is plug-and-play capable ■ Device: Designation of the device ■ Model: Designation of the device type ■ Serial number: Serial number of the device ■ ID number: HEIDENHAIN ID of the device ■ Brake present: Whether the motor has a brake 	
		Motor Motor Data tab: <ul style="list-style-type: none"> ■ Information about motor-specific data from the motor table 	
		Motor Result tab: <ul style="list-style-type: none"> ■ Information about motor data derived from the motor table or from the machine parameter file <ul style="list-style-type: none"> • Motor constant [Nm/A]: Internal reference value that indicates the ratio of torque to current (= torque constant). • Voltage constant: Internal reference value that indicates the ratio of no-load voltage to rated speed. • Field angle [°]: Value converted to [°] for the corresponding entry from the MP file (MP2256.x). 	

Icon	Icon	Icon	Function
			<p>Power module</p> <p>ID Label, see page 1319 tab:</p> <ul style="list-style-type: none"> ■ Plug and Play: Whether the device is plug-and-play capable ■ Device: Designation of the device ■ Model: Designation of the device type ■ Serial number: Serial number of the device ■ ID number: HEIDENHAIN ID of the device
			<p>Power module</p> <p>Status tab:</p> <ul style="list-style-type: none"> ■ Status information about various signals, see page 1317
			<p>EnDat rotational speed encoder</p> <p>Depending on the functions provided by the encoder, the submenus status, diagnosis and temperature are available.</p>
			<p>EnDat 2.2 rotational speed encoder</p> <p>Depending on the functions provided by the encoder, the submenus status, diagnosis and temperature are available.</p>
			<p>EnDat position encoder</p> <p>Depending on the functions provided by the encoder, the submenus status, diagnosis and temperature are available.</p>
			<p>EnDat 2.2 position encoder</p> <p>Depending on the functions provided by the encoder, the submenus status, diagnosis and temperature are available.</p>
			<p>Status</p> <ul style="list-style-type: none"> ■ Status information about various signals, see page 1316
			<p>Diagnosis</p> <p>The function reserves are evaluated, which provide an exact overview of the current status of the encoder. The three values displayed inform the user at a glance of the encoder performance.</p> <p>In the yellow range, fault-free operation is no longer guaranteed. The encoder should be checked. If required, assistance can be provided by your HEIDENHAIN service agency.</p>
			<p>Temperature</p> <p>The temperature information provided by the temperature sensors in the encoder and the motor is displayed.</p>

Icon	Icon	Icon	Function
			<p>Motor encoder test (available only after the code number 688379 or 807667 has been entered)</p> <p>Diagram tab: After the test has been completed, the result is displayed in a diagram. Two green lines in the diagram mark the minimum and maximum height of the speed encoder signals. The red line shows the measured signals.</p> <p>Motor encoder test</p> <p>Overview tab: Begin the test by pressing the START MEASUREMENT button.</p> <ul style="list-style-type: none"> ■ Status information about various signals, see page 1317 <p>The plain text displayed in the lower part of the screen provides additional information about the test results.</p>
			<p>Position encoder test</p> <p>Diagram tab: After the test has been completed, the result is displayed in a diagram. Two green lines in the diagram mark the minimum and maximum height of the speed encoder signals. The red line shows the measured signals.</p> <p>Position encoder test</p> <p>Overview tab: Begin the test by pressing the START MEASUREMENT button.</p> <ul style="list-style-type: none"> ■ Status information about various signals, see page 1317 <p>The plain text displayed in the lower part of the screen provides additional information about the test results.</p>
			<p>Drive test (available only after the code number 688379 or 807667 has been entered)</p> <p>Test tab: Begin the test by pressing the START MEASUREMENT button.</p> <ul style="list-style-type: none"> ■ Status information about various signals, see page 1318 <p>The plain text displayed in the lower part of the screen provides additional information about the test results.</p>

Meanings of the signals under "status / speed controller"

Signal	Meaning	Colors
MC enabling marker	The MC can accelerate the switch-off via this marker.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Enable ■ Red: No enabling
X150/X151 Drive enabling	The signal shows the enabling status for switch-off via "X150/X151" or via the PLC input from MP4132. The signal is formed from the status of the inputs X150/X151 or from MP4132 and the setting in MP2040.x.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Enabled: There is currently no switch-off commanded ■ Red: No enabling: The drive is currently switched off or locked via X150/X151 or via the PLC input from MP4132
Drive enabling from speed controller	This signal is a group signal for all hardware enabling signals from the speed controller.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Enable ■ Red: No enabling
Drive enabled by software	This signal is a group signal for all hardware enabling signals from the software.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Enable ■ Red: No enabling
Internal drive status	Gating of all ready signals (external, internal and software)	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Enable ■ Red: No enabling
Power module active (-SH2)	The signal shows the status of the $\overline{SH2}$ line to the power module. The CC activates/deactivates this line for switching off the power module.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Enable: The $\overline{SH2}$ signal is inactive ■ Red: No enabling: The $\overline{SH2}$ signal is active
Current controller active	The signal shows the status of the current controller. The current controller is either switched on (in control) or switched off.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Enable: Current controller is on (in control) ■ Red: Do not enable: Current controller is off
Speed controller is active	The signal shows the status of the speed controller. The speed controller is either switched on (in control) or switched off.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Enable: Speed controller is on (in control) ■ Red: Do not enable: Speed controller is off

Signal	Meaning	Colors
Rotor position captured	<p>This signal gives information about determining the field angle:</p> <p>Drive is not oriented:</p> <ul style="list-style-type: none"> ■ Motor with rotary encoder without Z1 track (incl. linear motors) before the first "Drive on" status ■ Non-aligned rotary encoder with EnDat interface (incl. linear motors), if the field angle has not yet been determined <p>Drive is roughly oriented:</p> <ul style="list-style-type: none"> ■ Motor with rotary encoder without Z1 track (incl. linear motors) after the first "Drive on" status ■ Motor with rotary encoder with Z1 track after it has been read <p>Drive is oriented:</p> <ul style="list-style-type: none"> ■ Motor with rotary encoder with Z1 track after traversing the reference mark ■ Aligned rotary encoder with EnDat interface immediately after switch-on ■ Non-aligned rotary encoder with EnDat interface immediately if the field angle has already been determined ■ Motor with rotary encoder without Z1 track after traversing the reference mark if the field angle has already been determined 	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Field angle has been determined ■ Yellow: Field angle has been roughly determined ■ Dark gray: Field angle has not been determined
Switching on speed controller	<p>This signal indicates whether the speed encoder has been switched on since the last time the Power interrupted message was active. Status changes are saved.</p>	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Dark gray: Speed controller has not been switched on yet (since power interruption) ■ Green: Speed controller is currently being switched on ■ Yellow: Speed controller was switched on once (this does not mean, however, that the speed controller is currently on)



Signal	Meaning	Colors
Switching off speed controller	This signal indicates whether the speed encoder has been switched off since the last time the Power interrupted message was active. Status changes are saved.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Dark gray: Speed controller has not been switched off yet (since power interruption) ■ Green: Speed controller is currently being switched off ■ Yellow: Speed controller was switched off once (this does not mean, however, that the speed controller is currently off)
Brake released	This signal shows the status of the motor brake signal on the PWM bus. This signal is led from the power module via a relay to the motor.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Brake released ■ Red: Brake engaged
I²t warning	This signal shows the current and stored status of the I ² t monitoring.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: No I²t warning up to now ■ Yellow: There has already (since switch-on) been an I²t warning, but there is no current warning ■ Orange: There is a current I²t warning
Torque ripple	The signal indicates the status of torque-ripple compensation.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Compensation is active ■ Red: Compensation is not active
Acceleration feedforward control	The signal shows the status of acceleration feedforward control (MP1392 or 1391.1, depending on the selected mode of operation).	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Feedforward control is active ■ Red: Feedforward control is not active

Meanings of the signals under "status / position controller"

Signal	Meaning	Colors
Position control loop closed	Corresponds to PLC word W1024/W1100 for a specific axis. Information about whether the position control loop is closed or open.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Position control loop closed ■ Red: Position control loop open
Position-controlled	The signal indicates whether the axis is operated with position feedback control or speed feedback control.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Axis is operated with position feedback control ■ Red: Axis is operated with speed feedback control
Nominal position value filter	The signal indicates whether the nominal position value filters are active.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Nominal position value filter is active ■ Red: Nominal position value filter is not active
Position controller limit	The signal indicates whether "eliminate following error" is active.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Output limit is active ■ Red: Output limit is not active
CC monitoring active	The signal indicates whether following-error monitoring is active (can be deactivated by PLC, W1042/W1136).	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Following-error monitoring is active ■ Red: Following-error monitoring is not active
Touch probe is active	The signal indicates whether probing is active.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Probing is active ■ Red: Probing is not active
Reference pulse active	The signal indicates whether the MC expects a reference pulse (host command).	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Expectation of reference pulse is active ■ Red: Expectation of reference pulse is not active
Velocity feedforward control	<p>The signal indicates whether the position controller is working with feedforward control or with following error.</p> <p>Acceleration feedforward control can be switched on/off cyclically by the MC.</p> <p>Corresponds to parameter MP1392 or MP1391.0 in the MP file, depending on the current mode of operation.</p>	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Operation with feedforward control ■ Red: Operation with following error



Meanings of the signals under "status/PLC"

Signal	Meaning	Colors
Axis positioned (PLC)	If the axes have reached the positioning window after a movement, the status is shown in W1026/W1104.	<ul style="list-style-type: none"> ■ Green: Axis in position ■ Yellow: Axis not in position
Position control loop closed (PLC)	Position control loop closed (W1040/W1132 inverted). By setting W1040/W1132 in the PLC, the position control loop is opened by the PLC program.	<ul style="list-style-type: none"> ■ Green: Position control loop closed ■ Yellow: Position control loop open
Axis enabled (PLC)	W1024/W1100 shows if the position control loop is open or closed, and if the axis has been enabled.	<ul style="list-style-type: none"> ■ Green: Axis not enabled ■ Yellow: Axis enabled
Axis in motion (PLC)	During axis movement, the NC sets the bits in W1026/W1104.	<ul style="list-style-type: none"> ■ Green: Axis in motion ■ Yellow: Axis at standstill



Meanings of the signals under "EnDat/status"

Signal	Meaning	Colors
Encoder model	Designation of the encoder model	
Device name	Name of the encoder	
Data width	Number of bits of position value	
Resolution	Speed encoder Number of signal periods per revolution Linear encoder: Signal period in nanometers Value 0: The encoder is a purely serial encoder.	
Distinguishable revolutions	Information about multiturn encoders. Number of max. distinguishable revolutions.	
Resolution of abs. track	Speed encoder Number of measuring steps per revolution Linear encoder: Number of measuring steps per nanometer	
Serial number	Serial number of encoder	
Absolute value	Current absolute value	
EnDat 2.2	Information about whether the encoder is EnDat 2.2-compatible.	
Alarms	Evaluation of different types of error information from the encoder. If the encoder reports an encoder error, fault-free encoder function is no longer guaranteed. The encoder must be checked and replaced if necessary. If required, assistance can be provided by your HEIDENHAIN service agency.	<ul style="list-style-type: none"> ■ Green: No error ■ Red: Encoder reports an error
Warnings	Evaluation of different types of status information from the encoder. If the encoder issues a warning, it means that the encoder still functions properly, but one of the functions is operating at its limits. The encoder must be checked and replaced if necessary. If required, assistance can be provided by your HEIDENHAIN service agency.	<ul style="list-style-type: none"> ■ Green: No warning. ■ Yellow: Encoder issues a warning



Meanings of the signals under "power module / status"

Signal	Meaning	Colors
Power module ready (LT-RDY)	The power module is ready: <ul style="list-style-type: none"> ■ Safety relay is on ■ Main contactor is on ■ SH1 (MC) is "High" ■ No error from the power module 	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Power module is ready ■ Red: Power module reports "not ready"
Switch-off power module (IGBT)	The signal shows that the IGBT in the power module has been switched off.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: No power module switch-off (IGBT) ■ Red: Power module switch-off (IGBT)
Power module temperature	The signal reports the status of the heat sink temperature in the power module: Either it is OK or too high.	<ul style="list-style-type: none"> ■ Gray: No information about the signal available ■ Green: Temperature of power module OK ■ Red: Temperature of power module too high

Meanings of the signals under "speed controller / overview"

Information	Meaning	Colors
Direction (only available in motor encoder test)	Results of the test for the direction of rotation.	<ul style="list-style-type: none"> ■ Green: Direction of counting and rotation are the same (OK) ■ Red: Direction of counting and rotation differ (error)
Amplitude	Result of the test of the speed encoders' signals	<ul style="list-style-type: none"> ■ Green: Signal amplitude within the tolerance (OK) ■ Red: Signal amplitude outside the tolerance (error)

Automated testing of drives

Information	Meaning	Colors
Phase 1, Phase 2, Phase 3	Status of the phases in the motor and the supply line	<ul style="list-style-type: none"> ■ Green: Phase is OK ■ Red: Phase error
Leakage current	Information on a short circuit to ground in the motor, the power module or in the supply line	<ul style="list-style-type: none"> ■ Green: No ground fault ■ Red: Ground fault detected
Power module	Status of the power module	<ul style="list-style-type: none"> ■ Green: Power module is OK ■ Red: Power module error
Supply unit	Status of the supply unit	<ul style="list-style-type: none"> ■ Green: Supply unit OK ■ Red: Supply unit error
Connection	Status of the connection control ↔ power module	<ul style="list-style-type: none"> ■ Green: Connection is OK ■ Red: Connection error

Faults caused by excessive leakage current or ground fault are detected only by the UV(R) power supply unit, although the fault can also be triggered by one of the connected power modules. The power modules must not be disconnected for fault finding, however. It is sufficient to deactivate one axis after the other using MP10 in order to find the axis or the power module causing the leakage-current fault.



Electronic ID label

HEIDENHAIN inverter components of type D, as well as HEIDENHAIN synchronous motors with absolute encoders with EnDat interface, are equipped with an electronic ID label.

The product name, the ID number and the serial number are saved in this ID label. These devices are automatically detected when the control is started.

- ▶ Load the displayed component to the corresponding machine parameter automatically with the **SELECT** soft key.

During every further control restart, the control checks whether the connected units with electronic ID label match the entries in MP2100.x or MP2200.x. If not, an error message might appear and the connected encoder must be transferred to the corresponding machine parameters by soft key.

In exceptional cases, the evaluation of the electronic ID label can be deactivated with MP7690.

MP7690

Evaluation of the electronic ID labels

Input:

%xx

Bit 0 – HEIDENHAIN power modules

0: Active

1: Inactive

Bit 1 – HEIDENHAIN synchronous motors

0: Active

1: Inactive

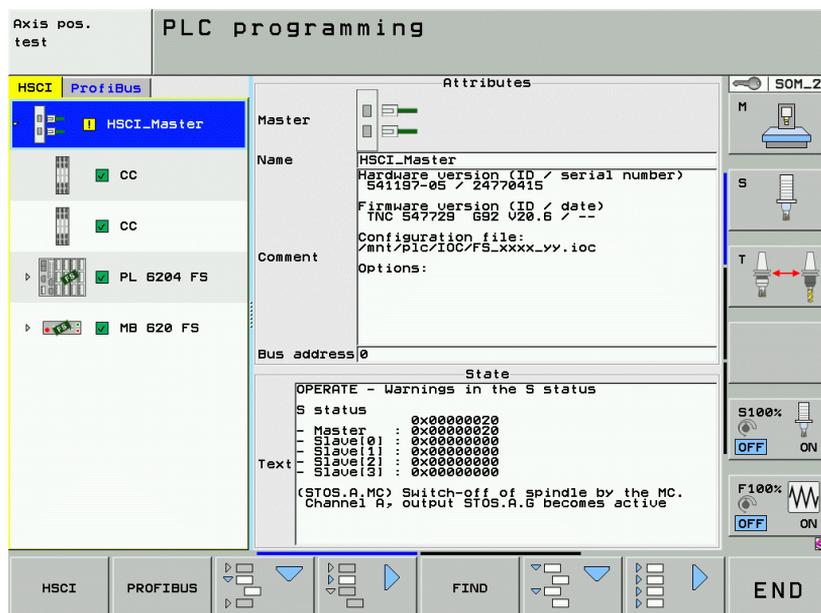
8.1.31 BUS diagnosis

The control features functions for diagnosis of the HSCI or PROFIBUS system.

You can access the functions as follows:

- ▶ Switch to the **Programming** mode of operation
- ▶ Press the MOD key
- ▶ Press the DIAGNOSIS soft key
- ▶ Press the BUS DIAGNOSIS soft key.

Example screen:



In Diagnosis mode, the structure of the HSCI/PROFIBUS system as well as the details of the HSCI/PROFIBUS components can be displayed.

The **HSCI** and **PROFIBUS** tabs above the left window are only shown if both bus systems are connected to the control. In this case you can switch between the bus systems by pressing the HSCI and PROFIBUS soft keys.

The detected, actually connected hardware at the HSCI bus (actual configuration) is displayed. The control also knows from the IOC file the previously configured hardware configuration to be expected (nominal configuration). The HSCI bus diagnosis compares the actual configuration to the nominal configuration and reveals deviations. At HSCI bus positions where a deviation was found, the actually configured hardware (**IOC:**) is displayed behind the detected connected hardware (dialog, e.g. **/IOC: MB 620**). This provides you with a direct comparison of the actual configuration and the nominal configuration in order to support you in finding errors. HSCI components that are missing from the actual configuration (e.g. due to an interruption in the HSCI bus) are identified by a corresponding symbol.

		Set	Reset
M4641	If the marker is set, the HSCI bus operates without error	NC	NC



Soft keys for HSCI/ PROFIBUS diagnostics

The following soft keys are available in the main menu for bus diagnosis:

Soft key	Submenu	Function
HSCI		Select the HSCI or PROFIBUS bus system
PROFIBUS		
		Open the HSCI/PROFIBUS tree
		Shrink the HSCI/PROFIBUS tree
		Move the separating line (enlarge/reduce the window size)
		
MORE FUNCTIONS		Open the submenu with additional functions
		Move the state window up or down
		
		Open/Close HSCI/PROFIBUS slaves
		
	FIND	Find an HSCI/PROFIBUS component
		Return to the previous soft key row
END		Exit the BUS diagnostics

Navigation with the arrow keys of the operating panel:

- ↓, ↑ Select an HSCI/PROFIBUS component
- → (or + key) Open the HSCI/PROFIBUS component for the connected modules or terminals to appear
- ← (or – key) Close the HSCI/PROFIBUS component

Screen contents

"HSCI/PROFIBUS" window (at left)

- Arrangement and designation of the HSCI/PROFIBUS components and terminals
- Status of the HSCI/PROFIBUS components and terminals:



OK



Error (further information in the text window)



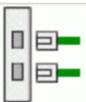
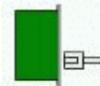
Warning (further information in the text window)



Undefined state (further information in the text window)

"Attributes" window (at upper right)

Configuration parameters of the selected HSCI/PROFIBUS component (see table).

Field	Function	Formed from...
Designation and symbol of the HSCI/PROFIBUS component		
	HSCI master 	
	PROFIBUS master 	
	Slave 	
	(Slave) modules, AS-i slave 	
	Terminal 	
Name	Name of the slave, slave module or terminal	<ul style="list-style-type: none"> ■ "Name" entered for the slave/slave module ■ "Symbolic name" entered for the terminal

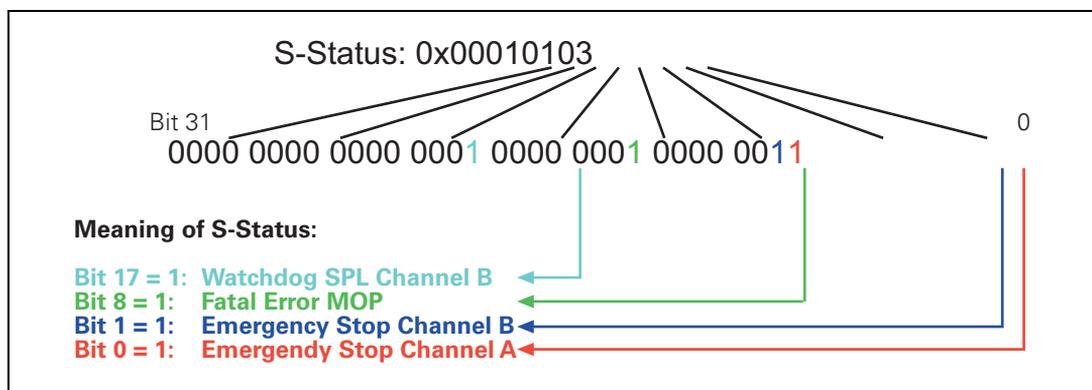
Field	Function	Formed from...
Comment	Name of the slave (for slaves and slave modules) or terminal description	<ul style="list-style-type: none"> ■ Master entries: <ul style="list-style-type: none"> • "Configuration file" ■ Master/slave entries: <ul style="list-style-type: none"> • "Name" • "Hardware ID" • "Hardware serial number" • "Firmware ID" • "Firmware timestamp" • "Configuration file" ■ Terminal entries: <ul style="list-style-type: none"> • "Comment" • "Pin name" • "Plan page" • "Order number"
Option	Designation of the option	Slave / slave module entered in "block"
Bus address	HSCI/PROFIBUS address	Slave entered in "HSCI/PROFIBUS address"
Slot	Slot number of the slave module	
Terminal	Terminal number	Determined from the terminals entered in "I/O offset" and "I/O bit"
Operand	Operand designation	Terminal entered in "PLC operand address"
Order no.	Order number	"Order number" entered for the terminal
Plan page	Wiring diagram page	"Page" entered for the terminal
Pin name	Terminal name	"Pin name" entered for the terminal

**"State" window
(at lower right)**

Current status of selected HSCI/PROFIBUS component.

- **Value:** Operand value of the terminal in decimal and hexadecimal code
- **Text:** Error texts that appear when an error occurs; otherwise, no display. The texts displayed are error-specific texts from the HDD/GSD file or the error information configured in IOconfig.
- Display of S status (31 bits) in HEX code. The meaning is displayed in abbreviated form below the S status.

You can use the following tables to look up the meaning of the individual S status bits. To do this, you need to convert the displayed HEX code to a binary number. Example:



Safe status bit	Signal	Meaning
0	-ES.A	Emergency stop channel A The control has initiated the SS1 alarm reaction.
1	-ES.B	Emergency stop channel B The control has initiated the SS1 alarm reaction.
2	-ES.A.HW	Emergency stop channel A, handwheel; no function in controls without functional safety. The control has initiated the SS1 alarm reaction.
3	-ES.B.HW	Emergency stop channel B, handwheel; no function in controls without functional safety. The control has initiated the SS1 alarm reaction.
4	-STO.A.MC.WD	Watchdog of MC software, switch-off of inverters, A channel (with functional safety (FS): switch-off of FS outputs). The control has initiated the SS1 alarm reaction.
5	-STOS.A.MC	Spindle is switched off by the MC, A channel, STOS.A.G is initiated (CC: switch-off of spindle); no function in controls without functional safety (FS)



Safe status bit	Signal	Meaning
6	-STO.B.CC.WD	Watchdog of CC software, switch-off of inverters, B channel The control has initiated the SS1F alarm reaction.
7	-SMC.A.WD	"Fast" watchdog of MC software; alarm on CC, which initiates the deceleration of the axes. The control has initiated the SS1 alarm reaction.
8	-SPL.WD	With FS: Multi-channel watchdog of SPL firmware (A/B channel); serious error of PL. Without FS: Single-channel watchdog of PL firmware. The control has initiated the SS1F alarm reaction.
9	-SMOP.WD	With FS: Multi-channel watchdog of SMOP firmware (A/B channel); serious error of MOP machine operating panel (SS1F). Without FS: Single-channel watchdog of MOP firmware (machine operating panel)
10	-PF.PS.AC	Power supply of inverter too low (parameterized LIFT OFF function in some cases).
11	-PF.PS.DC	DC-link voltage U_Z too low The control has initiated the SS1 alarm reaction.
12	-PF.BOARD	Fault in the supply voltage of the respective module. The control has initiated the SS1F alarm reaction.
13	-N0	Internal safe status bit The control has initiated the SS1 alarm reaction.
14	-REQ.SS2	The control has initiated the SS2 alarm reaction. Possible causes include: <ul style="list-style-type: none"> ■ Speed of MC fan or CC fan outside the tolerance ■ Temperature of MC, CC, UEC, UMC, PL or MB outside the tolerance ■ CC has detected an internal fault
15	-	Reserved

The following additional status bits are available for an external PL:

Safe status bit	Signal	Meaning
16	–SPL.A.WD	SPL watchdog, channel A
17	–SPL.B.WD	Only in controls with functional safety (FS): SPL watchdog, channel B
18	PGOOD.NC	Voltage monitoring of NC reports a fault
19	PGOOD.PLC	Voltage monitoring of PLC reports a fault
20	–INT	Internal interrupt
21..31	1	Reserved

The following additional status bits are available for an external MB machine operating panel:

Safe status bit	Signal	Meaning
16	–SMOP.A.WD	SMOP watchdog, channel A
17	–SMOP.B.WD	Only in controls with functional safety: SMOP watchdog, channel B
18	PGOOD.A	Voltage monitoring of channel A reports a fault
19	PGOOD.B	Voltage monitoring of channel B reports a fault
20	1	Reserved
21..31	1	Reserved

Overview of HSCI/ PROFIBUS I/O configuration

The number of inputs and outputs transmitted via HSCI/PROFIBUS are listed in the HSCI/PROFIBUS diagnosis.

It can thus, for example, be tested whether further Profibus slaves can be connected without exceeding the maximum number of 252 input bytes and 252 output bytes.

In order to see this overview, select the master to be displayed in the HSCI/PROFIBUS diagnosis.

The screenshot displays the 'Bus diagnosis' window with the following components:

- Manual operation** tab selected.
- HSCI_Master** selected in the tree view.
- Attributes** window showing:
 - Master:
 - Name: HSCI_Master
 - Configuration file: 0:\plc\hsci\node11_192_168_21_B2.ioc
 - Active Options:
 - Bus address: 0
- State** window showing:


```

OPERATE with warning
BUS-STATUS : 0x0002303
Master     : 0x0000000
Slave10   : 0xFFFFFFFF
Slave11   : 0xFFFFFFFF
Slave12   : 0xFFFFFFFF
      
```
- Text** window (empty).
- Bottom navigation bar:** HSCI, PROFIBUS, navigation arrows, MORE FUNCTIONS, END.



8.1.32 Expanded PLC module interface for 18 axes

Up to 18 NC axes are available in an HSCI system. This was done by increasing the greatest possible entry in the AXISNUMBER token of OEM.SYS from 14 to 18. Up to now, the PLC interface also supported only 14 axes. In order to use the increased number of axes, you must set bit 14 of MP4020.



Note

Please keep in mind that the parameters of PLC modules with bit-encoded axis masks or constants for axes and spindles are set and interpreted in different ways, depending on MP4020 bit14.

If the expanded PLC module interface is active, the spindle bit moves to bit 31 and the axis number (constant) for the spindle to K 31 for the following PLC modules:

PLC module	Function	If MP4020 bit 14 = 0	If MP4020 bit 14 = 1
9038	Read general axis information	K 15	K 31
9065	Status of the commissioning function	Bit 15	Bit 31
9158	Maximum torque	K 15	K 31
9128	Torque limiting by the PLC	K15	K31
9129	Status of torque limiting by the PLC	K15	K31
9149	Set field angle/Read via PLC	K15	K31
9157	Drive controller status	Bit 15	Bit 31
9159	Advance status message: Drives will be switched off	Bit 15	Bit 31
9161	Enable the drive controller	Bit 15	Bit 31
9162	Status request of the drive controller	Bit 15	Bit 31
9163	Switching the operating modes	K15	K31
9164	Read the actual speed value of the motor	K15	K31
9165	Sample the current motor temperature	K15	K31
9166	Momentary utilization of the drive motor	K15	K31
9168	Interrogate the commissioning status	Bit 15	Bit 31
9169	Axes for which I32 does not switch off the drives	Bit 15	Bit 31
9170	Find the current torque	K15	K31

PLC module	Function	If MP4020 bit 14 = 0	If MP4020 bit 14 = 1
9220	Renewed traversing of the reference marks	K15	K31
9314	Limitation of drive power	Bit 15	Bit 31

Modules like 9147 and 9120 are not affected by this change, since these modules only have axis numbers as transfer parameters, and do not return any information about spindles.

Also, some PLC words are affected by the conversion to the expanded PLC module interface. For the PLC words concerned (e.g. W1024), the expansion to 32 bits makes a conversion to double words (e.g. W1100) necessary so that all pieces of information can be included. The previous words continue to be valid, but they do not contain all of the 32 bits. The previous words will therefore be copied to a double word with 32 bits.

16-bit word W	32-bit double word D	Description
1024	1100	Axis enabling
1026	1104	Axes in position
1028	1108	Axes in motion
1030	1112	Current direction of traverse
1032	1116	Reference marks not yet traversed
1034	1120	Positive software limit switch was approached
1036	1124	Negative software limit switch was approached
1038	1128	Prepare to open the position control loop
1040	1132	Axis-specific opening of the position control loop
1042	1136	Deactivate monitoring functions
1044	1140	Actual-to-nominal value transfer
1046	1144	Manual traverse in positive direction
1048	1148	Manual traverse in negative direction
1050	1152	Incremental jog positioning in positive direction
1052	1156	Incremental jog positioning in negative direction
1054	1160	Reference end position
1056	1164	Lubrication pulse: Value in MP4050.x exceeded



16-bit word W	32-bit double word D	Description
1058	1168	Reset the accumulated distance (lubrication)
1060	1172	Axis-specific feed-rate enable
1062	1176	Lock the handwheel for specific axes

The PLC Modules 9160 (status interrogation of temperature and I2t monitoring) and 9168 (interrogate commissioning status) are further special cases. If the expanded PLC module interface is active, these two PLC modules generate an error message and can no longer be used in this case. Please use PLC Module 9066 (status of HEIDENHAIN hardware) instead of PLC Module 9160, and PLC Module 9065 (status of commissioning function) instead of PLC Module 9168. The corresponding functions were added to PLC Modules 9066 and 9065 with software version 340 49x-05.

A sufficient number of double words—matching the maximum possible number of machine axes (entry AXISNUMBER in the OEM.SYS)—must be reserved for saving the axis coordinates for PLC Modules 9040/9041 (read axis coordinates). If you do not reserve sufficient memory area for the axis coordinates, the subsequent memory area might be overwritten by Modules 9040 and 9041.

As the expanded PLC module interface also requires memory space for at most 32 double words, you can simply reserve the memory area for 32 double words as an alternative to the entry AXISNUMBER in order to prevent data from being lost by overwriting the memory area.

MP4020 Bit 14 – Expanded PLC module interface for 18 axes

Input: 0: PLC module interface for 14 axes plus spindle
 (axes = bit 0 to bit 14, spindle = bit 15)
 1: PLC module interface for 18 axes plus spindle
 (axes = bit 0 to bit 17, spindle = bit 31)



8.1.33 Window manager

The XFCE window manager is used to configure the control's graphical interface. The XFCE window manager is a standard window manager for UNIX-based operating systems. It provides the following functions:

- Optional taskbar for switching between various applications (user interfaces).
- Additional desktop on which the machine tool builder's applications can be run.
- Controlling the focus between NC-software applications and those of the machine tool builder.
- The size and position of pop-up windows can be changed. It is also possible to close, minimize and restore the pop-up windows.

Basic configuration

The window manager is supplied with two default configurations (FULL and SIMPLE). The configuration of the window manager can be changed at any time by using the XFCE configuration dialog.

The SIMPLE configuration neither has a taskbar, nor a background image for the third desktop. As a result, the TNC software with window manager differs only very slightly from the NC software version without window manager. The FULL configuration has a taskbar, however.

HEIDENHAIN recommends using the taskbar of the window manager only on controls equipped with a mouse or a touchpad.

If a software version with the window manager is installed on a control for the first time, the **PLC:\WINDOWMANAGER** directory does not exist. It is created when the NC software is started up for the first time, and the SIMPLE configuration is active.

If you delete the **PLC:\WINDOWMANAGER** directory, the directory is created again the next time the NC software is started and the SIMPLE configuration is used.

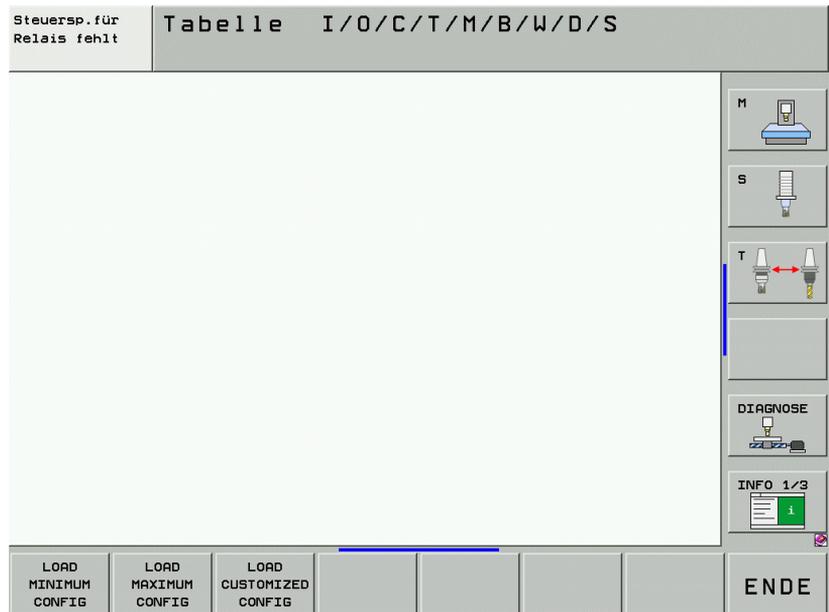
If TNCremoNT is used to transfer the PLC partition between the programming station and a single-processor or dual-processor version, the configuration of the Window Manager (**PLC:\WINDOWMANAGER** directory) is also transferred. The configuration is portable, except for the position of the XFCE taskbar (also called TNC taskbar).

On a dual-processor control, the TNC taskbar is moved to the opposite side of the screen if it displayed at the same position as the Windows taskbar, which is usually at the bottom of the screen.

If a window manager configuration exists, it is retained when the NC software is reinstalled (software update).



To call the configuration of the window manager, press the ADVANCED SETUP soft key and then the CONFIGURE WINDOW-MANAGER soft key.



- If you press the LOAD MINIMUM CONFIG soft key and confirm the following confirmation prompt, the current configuration of the window manager will be overwritten with the SIMPLE configuration.
- If you press the LOAD MAXIMUM CONFIG soft key and confirm the following confirmation prompt, the current configuration of the window manager will be overwritten with the FULL configuration.
- If you do not want to load any configuration, press the LOAD CUSTOMIZED CONFIG soft key. This starts the XFCE configuration dialog in which you can edit the current configuration.

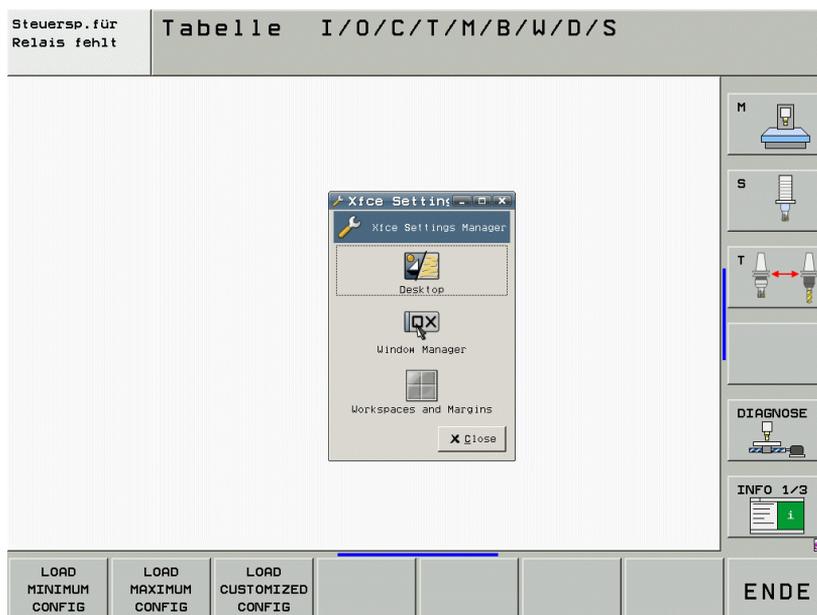
Configuration with XFCE

If you want to create your own configuration containing the TNC taskbar, you should first activate the FULL configuration. If you do not want your configuration to contain the TNC taskbar, first select the SIMPLE configuration.

Do not start the XFCE configuration until you have activated the respective basic configuration.

The two supplied basic configurations SIMPLE and FULL are only changed temporarily when you configure the window manager using the LOAD CUSTOMIZED CONFIG soft key. The files containing the two basic configurations are stored in the control and remain unchanged. This means that you can call the basic configurations at any time by pressing the LOAD MINIMUM CONFIG soft key or the LOAD MAXIMUM CONFIG soft key.

Direct saving of your own configurations is not possible. However, you can save your own configurations by saving the complete **PLC:\WINDOWMANAGER** directory via TNCremONT, for example. If you copy the **PLC:\WINDOWMANAGER** directory to other controls, the configuration saved in the directory will become active.



The XFCE configuration dialog provides the following configuration options:

■ **Desktop – Appearance**

The DESKTOP button and the Appearance tab of the XFCE configuration dialog enable you to define a background color and background images, which do only become effective on the third desktop (machine tool builder) or the desktops after the third one.

The background image from OEM.SYS is shown on the first two desktops (Edit, Machine).



Note

Additional background images (i.e. images that are not supplied) should always be saved in the **PLC:\WINDOWMANGER\BACKDROPS** directory. Saving the background images in this directory ensures that the path of the selected background image is adjusted to the configuration when backup/restore functions are executed between the programming station and dual-processor or single-processor controls.

■ **Desktop – Behavior**

This function is not yet supported.

In the future, the DESKTOP button and the Behavior tab of the XFCE configuration dialog will enable you to change the size of the desktop icons.

■ **Window manager – Style**

Press the window manager button and go to the Style tab of the XFCE configuration dialog to influence the layout of the windows. You can edit the font size of the window titles, the orientation of the window headers, the design of the window frames, etc.

■ **Window manager – Advanced**

Press the window manager button and go to the Advanced tab of the XFCE configuration dialog to influence the behavior of the windows.

■ **Workspaces and Margins**

To define the number of possible desktops and their names, use the Workspaces and the Margin buttons of the XFCE configuration dialog.

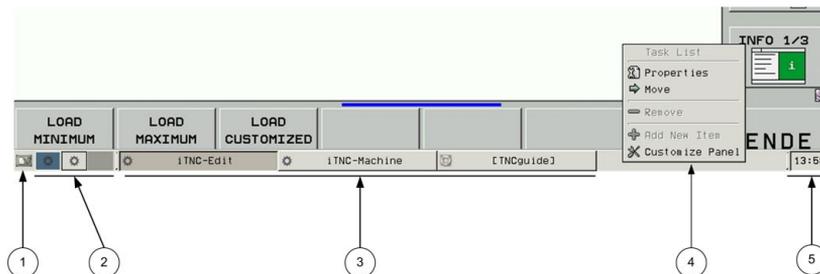


Note

A minimum of two desktops (Edit and Machine) must be defined in Number of Workspaces.

Configuration of the TNC taskbar

The TNC taskbar appears when you move the mouse pointer to the position of the taskbar (standard setting: at bottom of screen). When the mouse pointer is located over the TNC taskbar, press the right mouse key to open the context menu for the configuration.



When the FULL configuration is active, the TNC taskbar contains the following element buttons (from left to right):

- 1: Show desktop button
- 2: Pager for switching between the desktops and for moving windows between the desktops with "drag and drop".
- 3: Task list (shows all open windows)
- 4: Context menu for configuring the TNC taskbar
- 5: Time

To configure the TNC taskbar, proceed as follows:

- ▶ Press the ADVANCED SETUP soft key in the PLC operating mode.
- ▶ Press the CONFIGURE WINDOW MANAGER soft key.
- ▶ Move the mouse pointer over the position of the TNC taskbar until it appears.
- ▶ Place the mouse pointer over the TNC taskbar and press the right mouse key. To edit the properties of an element in the taskbar, place the mouse pointer over the respective button in the taskbar. This opens the context menu for configuring the TNC taskbar and the selected element.

The context menu provides the following configuration options:

- **Properties**
Change the properties of the selected element.
- **Move**
Change the sequence of the buttons in the taskbar.
- **Customize Panel**
Change the properties of the complete TNC taskbar.

Screen saver

In controls with HSCI, the screen saver is configured using the XFCE window manager or through the screen saver settings in the taskbar at the bottom of the screen:

- ▶ Make the taskbar at the bottom of the screen visible by moving the mouse pointer over it.
- ▶ Press the green HEIDENHAIN button to open the JH menu.
- ▶ Select the Settings menu item.
- ▶ Select the Screen Saver menu item.

If a (new) error message occurs, the screen saver will be exited automatically.

If the NC start key and an NC start key that is simulated via LSV2 are pressed, the screen saver is ended and the NC start executed immediately. Please note that the PLC program must always ensure that, while the guard doors are open, an NC start or axis motions can only be triggered if a permissive button is pressed at the same time. All of the other keys only cancel the screen saver the first time they are pressed, without executing the actual function.

The backlight of the screen can also be switched off together with the screen saver. The necessary settings can be made in the screen saver settings in the "Complex" tab. The option "Activate energy management" must be set.

Third desktop

The third desktop can be used primarily for additional applications that are to be displayed in addition to the actual control operating modes. With the Windows key you can always switch to the third desktop. You can also use the taskbar to switch to any of the desktops.

The NC start key is also effective on the third desktop (OEM screen) and is transferred to the NC (start of an NC program, if required). Furthermore, the ERR or HELP key opens the error list and/or help text window on the third desktop.

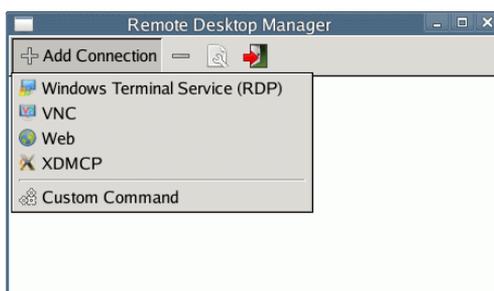
8.1.34 Software option 133—Remote Desktop Manager

Software option 133 enables you to display external computer units on the TNC screen that are connected via Ethernet and to operate them over the TNC. You can also start programs specifically under HeROS or display web pages of an external server.

HEIDENHAIN offers you the IPC 6341 as a Windows computer unit. With the IPC 6341 Windows computer you can start and operate Windows-based applications directly from the TNC, see page 1931.

Configure the Remote Desktop Manager as follows, after you have enabled software option 133:

- ▶ Make the taskbar at the bottom of the screen visible by moving the mouse pointer over it.
- ▶ Press the green HEIDENHAIN button to open the JH menu.
- ▶ Select the Remote Desktop Manager menu item.



Menu item	Function
Add Connection	Configure a new connection
Delete Connection	Delete a connection
Edit Connection	Edit a connection
End	Exit the Remote Desktop Manager

With the Remote Desktop Manager you can create and configure connections for individual external applications or computers (e.g Web or VNC). You can also define for each connection whether it is to be established right away when the control is booting, and on which control desktop it is to be displayed. Five basic connection types are available for the configuration of individual connections,

The following connection types are available in the **Add Connection** menu item. You can configure individual connections of the types listed there. The programs listed below are preinstalled on the control for the individual connection types:

- **Windows Terminal Service (RDP)**
Displays the desktop of a remote Windows computer on the control.
Program: Rdesktop Client 1.7.0
- **VNC**
Connection to an external computer (e.g. IPC 6341). Displays the desktop of a remote Windows or Unix computer on the control.
Program: Realvnc VNCViewer 4.1.3
- **Switch-off/restart of a computer**
Remote-controlled switch-off or reboot of a remote Windows® or Unix computer.
Program: net from Samba 3.3.1
- **Web**
Connection to an external HTTP server. Opens the Firefox browser. Shows the websites of remote computers by indicating the URL (e.g. also web interface for remote control of devices).
Program: Firefox 3.6.19
- **XDMCP**
Connection to an external Unix computer via X session.
Program: Xnest from X.Org XServer 1.9.2
- **User-defined connection**
The user's own applications or programs that should not be started using the NC software, but with HeROS 5.

Note: RDP and Net Shutdown are embedded programs of the Windows-Distribution (at least available since Windows XP).

The versions of these programs may change with future software releases, when newer versions of the respective program packages are available. The versions listed here are therefore meant as "at least version xxx".

When using the Remote Desktop Manager software option, please note the following:

- HEIDENHAIN only assures a functioning connection between HeROS 5 and the IPC 6341 (VNC, RDP and Net Shutdown protocols).
- HEIDENHAIN cannot guarantee the correct function of any other combinations or connections to external devices.
- If changes to preinstalled programs or data make service visits from HEIDENHAIN necessary, the service costs will be invoiced.
- To avoid loss of data or damage to an externally connected system, you should configure this computer to be shut down automatically when the the TNC is shut down.

General information on the connections

Once a connection has been configured, it is shown as an icon in the Remote Desktop Manager window. Click the connection icon with the right mouse key to open a menu in which the display can be started and stopped. With the Windows key you can always switch to the third desktop, which by default is used for the display. You can also use the taskbar to get to this desktop.

If the desktop of the external connection or the external computer is active, all inputs from the mouse and the keyboard are transmitted there.

All connections are canceled automatically when the HeROS 5 operating system is shut down. Please note, however, that only the connection is canceled, whereas the external computer or the external system is not shut down automatically.

Windows Terminal Service (RDP) connection

After selecting a connection type, you can configure a new connection with **Add Connection** or change an existing connection with **Edit Connection**:

Edit the connection

General

Connection name:

Restarting after end of connection:

Automatic starting upon login

Add to favorites

Move to the following workspace

Details

Computer:

User name:

Password:

Windows domain:

Full-screen mode

User-defined window size: x

Expanded options

"Computer" must be indicated

Menu item		Function
General information		General settings
	Connection name:	Name of the connection in the Remote Desktop Manager
	Restarting after end of connection	Restart a canceled connection: <ul style="list-style-type: none"> ■ Always restart Always restart if the connection ends ■ Never restart Never restart ■ Always after an error Restart if the connection ends due to an error ■ Ask after an error Display a message and ask after an error.
	Automatic starting upon login	Connection is established automatically when the control boots
	Add to favorites	A connection icon is displayed in the task bar. <ul style="list-style-type: none"> ■ Double-click Double-clicking the icon starts an inactive connection and/or switches to the desktop of this connection. ■ Single click Switches to the desktop of this connection. ■ Right click Shows a menu in which the connection can be established or canceled.
	Move to the following workspace	Desktop on which the connection is to be opened. Desktops 0 and 1 are reserved for the NC software and should not be used for other connections. The default connection is desktop 3. Also the Windows key on the keyboard unit automatically opens desktop 3.

Menu item		Function
Details		
	Computer:	Host name or IP address of the external computer. In the recommended configuration of the IPC 6341, it is the IP address 198.168.254.3.
	Password:	The default password for connection to the VNC server is "123". Please change this password on the IPC and TNC during commissioning.
	Windows domain	Domain of the target computer, if required.
	Full-screen mode	Show display of the external computer in full-screen mode. Only useful, if the resolution of the external computer matches the resolution of the control screen (BF).
	User-defined window size	Definition of the window size for the display of the external computer or connection. As a rule, the window size should correspond to the resolution of the external computer or connection, as long as it is lower than the resolution of the control screen.
Expanded options		Advanced settings that are usually not necessary. Only in consultation with HEIDENHAIN.

For the connection of the IPC 6341, HEIDENHAIN recommends using the RDP connection. With RDP, the screen of the external computer is not mirrored, as with VNC, but an individual desktop is opened for it. The desktop that is active on the external computer when the connection is established is then locked and the user is logged off. This prevents operation from two sides. The user "TNC" should be defined right from the start in the settings of the connection to the IPC 6341.

Also, with an RDP connection it is not possible to completely shut down the external computer via the Windows menu. It is only possible to cancel the connection or log off the user.

The RDP connection also adjusts the screen resolution automatically.

Please note that an RDP connection is closed again automatically after 30 seconds if no user name and password were configured for it or if the user has not logged on within this time. The **Automatic starting upon login** function therefore only makes sense if a user with user name and password was configured.



VNC connection

After selecting a connection type, you can configure a new connection with **Add Connection** or change an existing connection with **Edit Connection**:

Edit Connection

General

Name:

Reconnect Policy: **Prompt For Reconnect After Error**

Autostart at login time

Add to Favorites

Start at Workspace:

Details

Host:

Password:

Fullscreen Window

Custom Window Size: x

Share with other clients

View only

▸ **Advanced Options**

Host is a required field

Menu item		Function
General		General settings
	Connection name:	Name of the connection in the Remote Desktop Manager
	Restarting after end of connection	Restarting after end of connection: <ul style="list-style-type: none"> ■ Always restart Always restart if the connection ends ■ Never restart Never restart ■ Always after an error Restart if the connection ends due to an error ■ Ask after an error After an error, display a message and prompt for reconnection
	Automatic starting after end of connection	Connection is established automatically when the control boots
	Add to favorites	A connection icon is displayed in the task bar. <ul style="list-style-type: none"> ■ Double-click Double-clicking the icon starts an inactive connection and/or switches to the desktop of this connection. ■ Single click Switches to the desktop of this connection. ■ Right click Shows a menu in which the connection can be established or canceled.
	Move to the following workspace	Desktop on which the connection is to be opened. Desktops 0 and 1 are reserved for the NC software and should not be used for other connections. The default setting is desktop 3. Also the Windows key on the keyboard unit automatically opens desktop 3.

Menu item		Function
Details		
	Computer:	Host name or IP address of the external computer. In the recommended configuration of the IPC 6341, it is the IP address 198.168.254.3.
	Password:	The default password for connection to the VNC server is "123". Please change this password on the IPC and TNC during commissioning.
	Full-screen mode	Show display of the external computer in full-screen mode. Only useful, if the resolution of the external computer matches the resolution of the control screen (BF).
	User-defined window size	Definition of the window size for the display of the external computer or connection. As a rule, the window size should correspond to the resolution of the external computer or connection, as long as it is lower than the resolution of the control screen.
	Permit further connections	Also grant other VNC connections access to the VNC server.
	View only	Display-only mode. The mouse and keyboard inputs are not transferred to the external connection or external computer.
Advanced options		Advanced settings that are usually not necessary. Only in consultation with HEIDENHAIN.

HEIDENHAIN recommends using the VNC connection if a program requiring graphics acceleration (e.g. programs to play movies) is installed on the external computer. RDP does not support this.

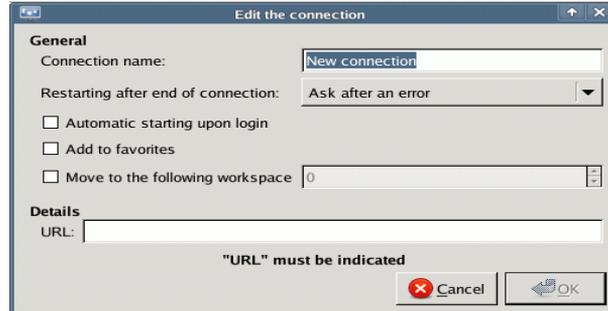
With VNC, the screen of the external computer is mirrored directly. The active desktop on the external computer is not locked automatically.

With a VNC connection it is also possible to completely shut down the external computer via the Windows menu. As the computer cannot be rebooted over a connection, it must actually be switched off and on again.

With the VNC connection, the screen resolution is not adjusted automatically during mirroring. If the resolutions of the external computer and the control desktop do not match, this can have the effect, for example, that a scrollbar appears on the screen. HEIDENHAIN therefore recommends activating the full-screen mode.

WEB connection

To configure a connection that opens a web page via the Firefox browser, select the connection type **WEB** and click **Add Connection** to set up a new connection or **Edit Connection** to change an existing one:



Menu item		Function
General		General settings
	Connection name:	Name of the connection in the Remote Desktop Manager
	Restarting after end of connection	Not necessary with this connection type
	Automatic starting upon login	Not necessary with this connection type
	Add to favorites	A connection icon is displayed in the task bar. <ul style="list-style-type: none"> ■ Double-click Double-clicking the icon starts an inactive connection and/or switches to the desktop of this connection. ■ Single click Switches to the desktop of this connection. ■ Right click Shows a menu in which the connection can be established or canceled.
	Move to the following workspace	Not active with this connection type

Menu item		Function
Details		
	URL:	Enter a valid web address or website. The contents to be displayed must be suitable for the Firefox browser. Addresses of local web servers can also be entered with indication of the port.

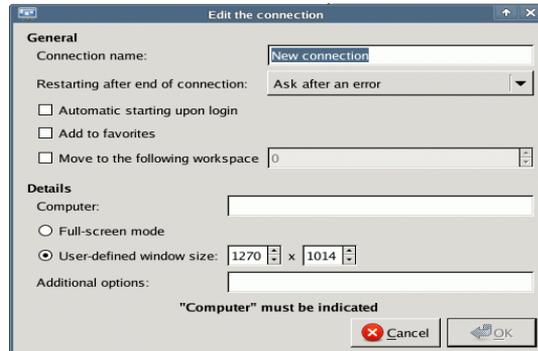
Every browser of a WEB connection is started with its own browser profile, with the name of the connection. Browser profiles save all browser settings, such as the starting page, connection type and history. Please note the following:

- Every new connection is configured with the default settings.
- These settings have to be adapted for the individual settings to some extent (e.g. proxy settings, users, password).
- If the connection name is changed, the existing settings are lost, too. The default settings become active.

If a browser is opened manually via the taskbar, this browser is opened with the configured settings. If another browser is opened manually via the taskbar, however, this browser, as well, takes the settings of any WEB connection that is already open. This can cause problems if different browser profiles have been saved. HEIDENHAIN therefore recommends starting further WEB accesses by opening a new tab in the currently active browser or via the "in new window" function.

XDMCP connection

To configure a new connection, select the XDMCP type of connection and click **Add Connection**:

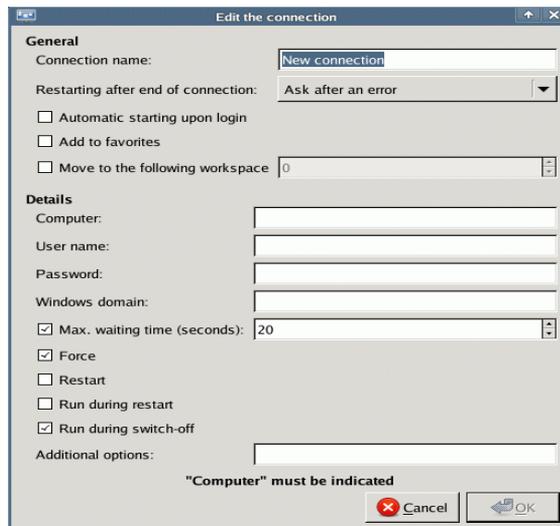


Menu item		Function
General		General settings
	Connection name:	Name of the connection in the Remote Desktop Manager
	Restarting after end of connection	Restart a canceled connection: <ul style="list-style-type: none"> ■ Always restart Always restart if the connection ends ■ Never restart Never restart ■ Always after an error Restart if the connection ends due to an error ■ Ask after an error Display a message and ask after an error.
	Autostart	Connection is established automatically when the control boots
	Add to favorites	A connection icon is displayed in the task bar. <ul style="list-style-type: none"> ■ Double-click Double-clicking the icon starts an inactive connection and/or switches to the desktop of this connection. ■ Single click Switches to the desktop of this connection. ■ Right click Shows a menu in which the connection can be established or canceled.
	Move to the following workspace	Desktop on which the connection is to be opened. Desktops 0 and 1 are reserved for the NC software and should not be used for other connections. The default setting is desktop 3. Also the Windows key on the keyboard unit automatically opens desktop 3.

Menu item		Function
Details		
	Computer:	Command to be executed
	Full-screen mode	Show display of the external computer in full-screen mode. Only useful, if the resolution of the external computer matches the resolution of the control screen (BF).
	User-defined window size	Definition of the window size for the display of the external computer or connection. As a rule, the window size should correspond to the resolution of the external computer or connection, as long as it is lower than the resolution of the control screen.

Shutting down or rebooting an external computer

To set up a connection with which you can configure the automatic shutdown of a Windows computer, select **Switch-off/restart of a computer** and click **Add Connection** to configure a new connection or **Edit Connection** to change an existing one:



Note

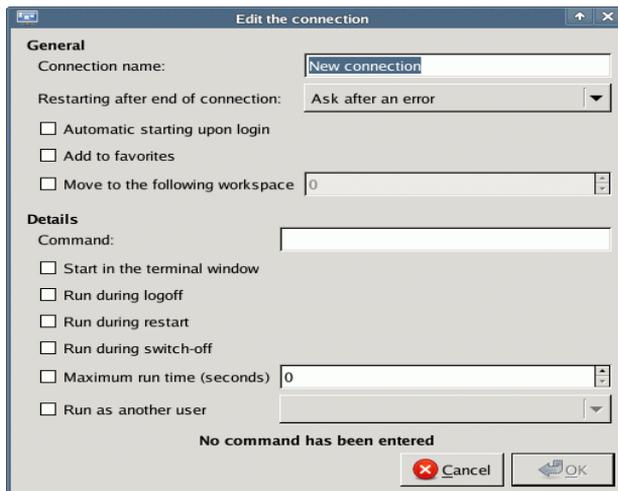
To avoid loss of data or damage to the Windows system, you should configure an automatic shutdown of the Windows computer together with the shutdown of the TNC.

Menu item		Function
General		General settings
	Connection name:	Name of the connection in the Remote Desktop Manager
	Restarting after end of connection	Not necessary with this connection type
	Automatic starting upon login	Not necessary with this connection type
	Add to favorites	<p>A connection icon is displayed in the task bar.</p> <ul style="list-style-type: none"> ■ Double-click Double-clicking the icon starts an inactive connection and/or switches to the desktop of this connection. ■ Single click Switches to the desktop of this connection. ■ Right click Shows a menu in which the connection can be established or canceled.
	Move to the following workspace	Not active with this connection type

Menu item		Function
Details		
	Computer:	Host name or IP address of the external computer. In the recommended configuration of the IPC 6341, it is the IP address 198.168.254.3.
	User name:	User name with which the connection should log in.
	Password:	The default password for connection to the VNC server is "123". Please change this password on the IPC and TNC during commissioning.
	Windows domain	Domain of the target computer, if required.
	Timeout (sec.):	When the TNC shuts down, it commands the Windows computer to shut down. Before the TNC displays the message "You may switch off now", it waits for <Timeout> seconds. If the Windows computer is switched off before the <Timeout> seconds are over, the TNC does not wait any longer.
	Force	All programs on the Windows computer are shut down automatically, even if dialogs of these programs, for example "Save file yes/no" are still open. If "Force" is not set, Windows waits for up to 20 seconds. This delays the shutdown process or the Windows computer is switched off before Windows has shut down.
	Reboot	Reboot the Windows computer.
	Run during restart	Reboot the Windows computer when the TNC reboots. Only effective if the TNC is rebooted with the Shutdown icon in the taskbar at bottom right or if a reboot is performed due to changes to the system settings (e.g. network settings).
	Run during switch-off	Switch off the Windows computer when the TNC is shut down (no reboot). This is the standard scenario. The END key will then no longer trigger a reboot, either.
Additional options		Advanced settings that are usually not necessary. Only in consultation with HEIDENHAIN.

User-defined connection

To set up a connection with which you can configure the automatic shutdown of a Windows computer, select **Switch-off/restart of a computer** and click **Add Connection** to configure a new connection or **Edit Connection** to change an existing one:



Menu item		Function
General		General settings
	Connection name:	Name of the connection in the Remote Desktop Manager
	Automatic starting upon login	Not necessary with this connection type
	Add to favorites	A connection icon is displayed in the task bar. <ul style="list-style-type: none"> ■ Double-click Double-clicking the icon starts an inactive connection and/or switches to the desktop of this connection. ■ Single click Switches to the desktop of this connection. ■ Right click Shows a menu in which the connection can be established or canceled.
	Move to the following workspace	Not active with this connection type



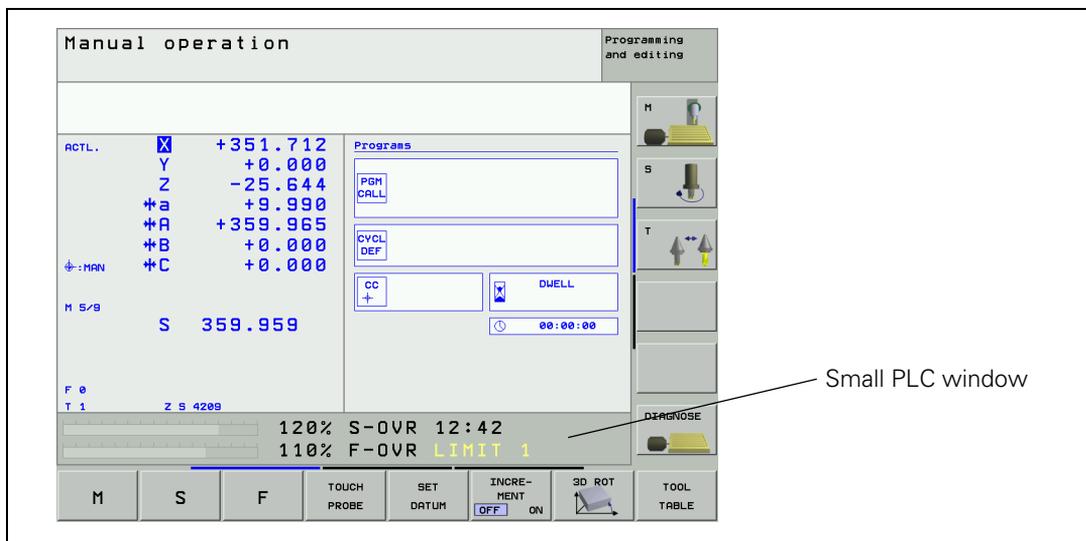
Menu item		Function
Details		
	Command:	Any program can be executed with the command.
	Start in the terminal window:	The command or program is started in a terminal window. This can be necessary on programs without a graphic user interface to display the program outputs in the terminal or to be able to make inputs via the terminal.
	Run during logoff:	Starts the command or program with every logoff from HeROS. The function is effective every time the HeROS interface is closed, which can also be done via the task bar without completely shutting down the control, for example.
	Run during restart:	Starts the command or program every time HeROS is restarted.
	Run during switch-off:	Starts the command or program every time HeROS is shut down.
	Max. run time (seconds):	Run-time restriction for the started program. When the specified time has expired, the Remote Desktop Manager terminates the program automatically. 0 = No run-time restriction
	Run as another user	The command or program is started with a different user. By default, all programs are started as "TNC" user. This user has only limited rights, e.g. no access to the PLC:. If "Run as another user" is set, any defined user can be selected. In addition to this, the user password must be entered.
Additional options		Advanced settings that are usually not necessary. Only in consultation with HEIDENHAIN.

8.2 PLC window

8.2.1 Small PLC window

The small PLC window is shown in the following operating modes:

- Manual Operation
- E1. Handwheel
- Positioning with Manual Data Input
- Program Run, Single Block
- Program Run, Full Sequence



Any ASCII text can be shown in two lines, with a limited number of characters. In the left half of the line a bar diagram can be shown optionally or additionally. Maximum number of ASCII characters per line:

- BF 250: 39 characters
- BF 260: 50 characters
- ▶ Specify the colors of the small PLC window in MP7370.x (see "Color settings" on page 1267).
- ▶ Configure the window display in the PLC program with Modules 9080 to 9083.
 - 9080: Clear small PLC window
 - 9081: Interrogate status of the small PLC window
 - 9082: Display a string in the small PLC window
 - 9083: Display a bar diagram in the small PLC window

Modules 9080, 9082 and 9083 must be called in a submit or spawn job.

Modules 9080, 9082 and 9083 are also in effect if the selected screen contains no PLC window (e.g. large graphic display) or the PLC window is in the background.

Do not interrupt processing of the modules through a CAN command!

Module 9080 Clear the small PLC window

With this module you can clear the contents of the small PLC window.

Call:

CM 9080

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Call was not in a submit or spawn job

Module 9081 Interrogate the status of the small PLC window

With this module you can ascertain whether a small PLC window is being displayed.

Call:

CM 9081

PL B/W/D

<Status of the small PLC window>

Bit 0=1: A small window is in the selected screen
(background or foreground)

Bit 1=1: Small PLC window in the foreground

Module 9082 Display a string in the small PLC window

The string is designated with a string number or is transferred as an immediate string and ends with the ASCII character <NUL>. It is shown in the small PLC window in line 0 or 1, with each character in the color given for it. Text strings with a length of max. 128 characters can be transferred to PLC module 9082 for display in the small PLC window. Depending on the size of the window and the screen being used, the NC automatically limits the text string to the maximum number of displayable characters. In the event of error, no string is shown.

Column	063																
Line 0																	
Line 1																	

References to PLC dialogs or PLC error messages are executed:

- Entered dialog or error number not found:
Is replaced by the ASCII character "@".
- Non-displayable character in the text (except string end):
Is replaced by the ASCII character "^".

The character size is oriented to the size of the screen window or by the current operating mode, and cannot be influenced.

If the specified color number is zero, the text is shown in the same color as the character last shown. If the first character of a line is specified as zero, the color is undefined and can change from one display line to another.

Call:

- PS K/B/W/D <Line number>
0 or 1
- PS K/B/W/D <Column number>
0 to 63
- PS K/B/W/D <Number of the color>
0 to 15
- PS K/B/W/D/S<String number or string>
- CM 9082

Error code:

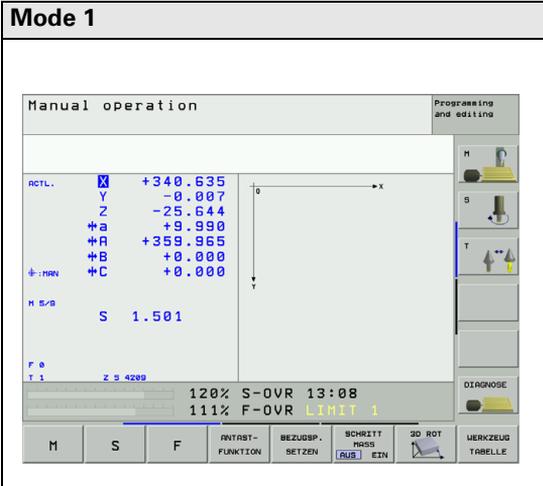
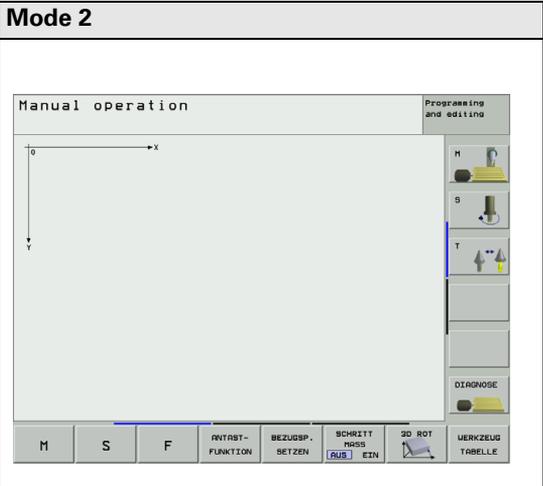
Marker	Value	Meaning
M4203	0	No error
	1	Call was not in a submit or spawn job or line less than zero or greater than 1 or Wrong string number or no end of strings or the last characters of the string can no longer be displayed in the screen window.



8.2.2 Large PLC window

Depending on the display mode, the large PLC window can be shown instead of the graphic/status window, or even over the entire screen. The PLC window can be combined with the PLC soft keys. See page 1372.

- ▶ Select the display mode with the screen management key or with Module 9202.

Mode 1		Mode 2	
			
SMALL	17 lines, 39 columns $0 = y16/x8$	SMALL	27 lines, 79 columns $0 = y16/x8$
MEDIUM	11 lines, 19 columns $0 = y24/x8$	MEDIUM	18 lines, 39 columns $0 = y24/x8$
LARGE	5 lines, 9 columns $0 = y48/x8$	LARGE	9 lines, 19 columns $0 = y48/x8$

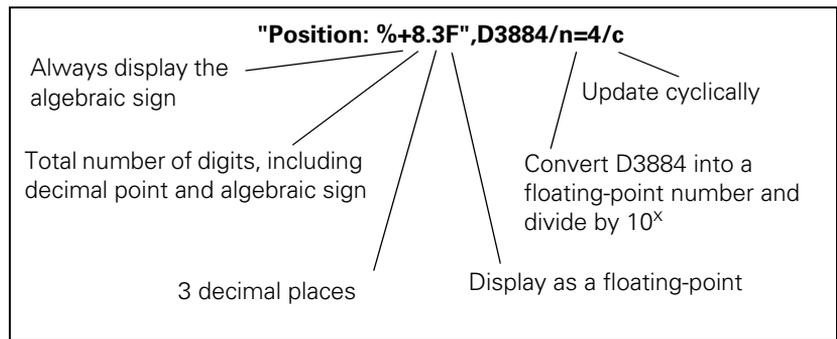
- ▶ Define the character size with the special command **charsize =** (see "Special commands:" on page 1363). The specified position refers to the lower left corner of the first character.

The content of the PLC window is defined in a screen mask – an ASCII file containing format instructions and special commands:

- ▶ Enter in Module 9210 the name of the screen mask to activate the PLC window, or use the functions of the soft-key project file for display (see "PLC soft keys" on page 1372).

Format instructions Format instructions are enclosed in quotation marks (""). Variables are transferred as parameters. Symbolic operands can also be used.

Example:



Variable types specified in the format instruction can be written in lowercase or uppercase letters (e.g. %D). The variable types of the format instruction must agree with the specified variables.



Note

Integer variables in the iTNC have a length of 32 bits.

PLC variables can be displayed as a number with decimal places:

- With the variable switch **/n=x** you convert from integer to double.

Variable names:

Variable names	Meaning
B <address>	PLC bytes, integer
W <address>	PLC words, integer
D <address>	PLC double words, integer 0/1
M <address>	PLC markers, integer 0/1
I <address>	PLC inputs, integer 0/1
O <address>	PLC outputs, integer 0/1
T <address>	PLC timers, integer 0/1
C <address>	PLC counters, integer 0/1
S <address>	PLC strings, string [128]
S#D <Number>	PLC dialogs, string
S#E <Number>	PLC error texts, string
TIME [0] to TIME [15]	System time as in Module 9055, char
AXISCHAR [Number]	Code letters for NC axis, char
MP <number>	Machine parameters, notation: MP910.1 Input value: Decimal places: double Hex or binary: integer Text: char

Time		
HOUR	int	No. of hours from real-time clock
MIN	int	No. of minutes from real-time clock
SEC	int	No. of seconds from real-time clock
DAY	int	Day from real-time clock
MONTH	int	Month as no. from real-time clock
YEAR2	int	Two-digit year no. from real-time clock
YEAR4	int	Four-digit year from the real-time clock

Settings for the tool touch probe		
TT.RAD	double	Calibrated radius of TT
TT.CENTER [3]	double	Calibrated center of TT
TT.PNT1 [3]	double	Calibrated touch point 0 of TT
TT.PNT2 [3]	double	Calibrated touch point 1 of TT
TT.PNT3 [3]	double	Calibrated touch point 2 of TT
TT.PNT4 [3]	double	Calibrated touch point 3 of TT

Settings for RS-232		
RS232.FEBAUD	string	Baud rate FE on RS-232
RS232.EXT1BAUD	string	Baud rate EXT1 on RS-232
RS232.EXT2BAUD	string	Baud rate EXT2 on RS-232
RS232.LSV2BAUD	string	Baud rate LSV2 on RS-232
RS232.MODE	string	RS-232 mode

Settings for simulation		
SIMU.ENAPRESET	string	Preset enable
SIMU.ENALIMIT	string	Limit switch enable
SIMU.LIMITPL [5]	double	Positive limit switch
SIMU.LIMITMI [5]	double	Negative limit switch
SIMU.PRESET [5]	double	Preset values

Settings of the machine		
MACHINE.LIMIT1PL [5]	double	Pos. limit switch group 1
MACHINE.LIMIT1MI [5]	double	Neg. limit switch group 1
MACHINE.LIMIT2PL [5]	double	Pos. limit switch group 2
MACHINE.LIMIT2MI [5]	double	Neg. limit switch group 2
MACHINE.LIMIT3PL [5]	double	Pos. limit switch group 3
MACHINE.LIMIT3MI [5]	double	Neg. limit switch group 3
MACHINE.PRESET1 [5]	double	Preset values 1
MACHINE.PRESET2 [5]	double	Preset values 2
MACHINE.PRESET3 [5]	double	Preset values 3
MACHINE.POSINC [5]	double	
MACHINE.DRVOFFSET [5]	double	
MACHINE.HANDW_FACTOR [5]	double	Handwheel subdivision factor

Settings for transformation of the machine reference system	
MATRANS.PRESX1_ABC[3]	double
MATRANS.PRESY1_ABC[3]	double
MATRANS.PRESZ1_ABC[3]	double
MATRANS.PRESAXIS1	string
MATRANS.PRESX2_ABC[3]	double
MATRANS.PRESY2_ABC[3]	double
MATRANS.PRESZ2_ABC[3]	double
MATRANS.PRESAXIS3	string
MATRANS.PRESZ3_ABC[3]	double
MATRANS.TRLPRES1[6]	double
MATRANS.TRLPRES2[6]	double
MATRANS.TRLPRES2[6]	double
MATRANS.TRLPRES3[6]	double
MATRANS.MANUAL	string
MATRANS.PGRMRUN	string
MATRANS.ANGLE[3]	double

Settings for the display	
DISPLAY.AXIS1	string
DISPLAY.AXIS2	string
DISPLAY.SCREEN	int
DISPLAY.FORMAT	string for MM/INCH conversion

Settings for the oscilloscope	
OSC.AXIS1	string
OSC.TIMEBASE	string
OSC.MODE	string
OSC.TRGCHAN	string
OSC.TRGTHRES	double
OSC.SLOPE	string
OSC.PRETRIG	string
OSC.FEED	double

Miscellaneous	
MISC.MDI	string
MISC.OUTPRECISION	string
MISC.TEACHINAXIS	int

Special characters:**\n**

Newline: Shifts the cursor to the left edge of the window. At the same time, it moves downward by the preset distance defined with **LINEDIST**.

\f

Formfeed: Functions like "**\n**". In addition, a page break is performed if the cursor moves past this special character. Otherwise the window is scrolled.

\xYY<SPACE>

Special character: YY is the hexadecimal number of the 8-bit ASCII code of the desired character, followed by a blank space, e.g. "\x23 " = "#". Keep in mind that certain characters could also be interpreted as system commands, e.g. "\25 " = "%" (see "iTNC character set" on page 1278).

Switches for variables:**/n=x**

For B/W/D. The integer can be converted to a floating point number with x places after the decimal point (e.g. to display a position in the format 0.1 m in mm instead).

/mi

For B/W/D. Ensure that the inch conversion is active. The number can be converted to a value in inches.

/e

For B/W/D/M/S. Define the field length in the format string. The current contents of the variables are displayed and can be changed.

/i

For B/W/D/M/S. Define the field length in the format string. A value can be entered in the empty field.

/c

For B/W/D/M/S/TIME. Define the field length in the format string, otherwise the subsequent text may be overwritten if the field length varies due to a change in the numerical value. The field content is updated cyclically.

Special commands:**/*<Comment>*/**

You can enter a comment between the asterisks.

MMINCH

Converts variables that contain an /mi switch (or a position) into inches. Select a default setting under Mode. Usual default setting: "No conversion"

POS=xpix, ypix

Writes the next text or graphic at the designated position. Default setting: Writing begins at the upper edge, then progresses line by line.

ypix = Distance in pixels from the upper edge of the current page

xpix = Distance in pixels from the left edge of the window

IPOS=xpix, ypix

Writes the next text or graphic at a position offset from the present position by the specified number of pixels.

xpix = Distance in pixels from the current X position

ypix = Distance in pixels from the current Y position

CPOS=column, line

Writes the next text or graphic at the designated position. The width of a column is calculated from the current character set. The height of a line is preset and can be changed with **LINEDIST**.

line = Line on the current page

column = Column from the left edge of the window

ICPOS=column, line

Writes the next text or graphic at a position offset from the present position by the specified number of lines and columns. The width of a column is calculated from the current character set. The height of a line is preset and can be changed with **LINEDIST**.

line = Distance in lines from the old line

column = Distance in spaces from the old space

LINEDIST=ypix Defines the line spacing. The default setting depends on the character size and is reset with every call of **CHARSIZE**.

COLOR=[f] or COLOUR=[f]

Sets the foreground color. Value range for f: 1 to 15

Default setting: Color 11

The colors are defined in MP7367.

(see "Color settings" on page 1267)

CURSOR=ON/OFF

Switches the inversion (highlighting) on and off.

Default setting: OFF

CHARSIZE=SMALL/MEDIUM/LARGE/AUTO

Defines the character size.

Split screen: **SMALL**

Large PLC window: **MEDIUM**

AUTO: Character size depends on the window size.

Default setting: **AUTO**

With every call of **CHARSIZE**, the value of **LINEDIST** is overwritten by a default setting that depends on the character set.

Default spacings:

Line	Column
SMALL	168 pixels
MEDIUM	2416 pixels
LARGE	4832 pixels



GRAPHICS= <filename> [,<Layer>] [/c]

Links a graphic into the window. The following graphic formats can be displayed:

- *.HE files (*.DXF files converted by PLCdesign)
- *.BMP
- *.BMX

With **POS**, **IPOS** or **ICPOS**, enter a position. For *.HE files the lower left corner is set to the current position; whereas for *.BMP and *.BMX files the upper left corner is set.

<filename>: File name with path or just the file name. In this case the path is completed according to MP7230.3 (language for Help files). If **%GraphicsResolution%** is entered in the path, then this text is replaced by the current screen resolution (1024x768 or 640x480).

<Layer> (optional): Layer of a BMX file to be displayed. If no entry is made, the basic image (layer 0) is displayed.

/c (optional): The graphic is cyclically refreshed. The cycle time is defined via **REFRESH = .** If it is a BMX file, the layers are displayed cyclically, beginning with 0 or **<Layer>**.

Screen	Max. resolution of *.BMP, *.BMX
BF 250 (position/program + PLC)	442 x 431 (16 or 24-bit color depth)
BF 250 (only PLC)	886 x 593 (16 or 24-bit color depth)

Images that are too large are truncated.

Example: **GRAPHICS = PLC:\Images\%GraphicsResolution%\Test.BMX**

The file Test.BMX is searched for in the paths PLC:\Images\1024x768\ or PLC:\Images\640x480\.

TEXTFILE= <filename>

Links a text file into the PLC window. The text begins at the current position. Every additional line begins at the same X position, but offset downward by **LINEDIST**. The line break automatically adapts itself to the available space. Characters such as "Line Feed," "Carriage Return," "Horizontal Tab" and "Vertical Tab" are converted to spaces.

The backslash "\" is used as a special symbol. It is used to initiate the following functions:

"\n", "\N"

Insert manual line feed (end of paragraph)

"\f", "\F"

Insert page feed (division into more than one screen page).

"\\"

Shows the "\" character in the text.

<filename>

Contains a file name with path, or the file name only. In this case the path is completed according to MP7230.3 (language for Help files).

ERRQUE=n [/c] [/e] [/l] [/n] [/s]

Links a table with the messages waiting in the PLC error queue.

n: Number of table lines

/c: Table is updated cyclically.

/e: Paging in tables, message can be acknowledged with CE.

/l: Alternative to **/n**. Line number before the error text. Position in the error source is displayed.

/n: Alternative to **/l**. **/l** has priority.

Error number before the error text.

Line number of the .PET table is displayed.

/s: Three-digit status field with the following information:

C: CE possible

S: Message causes a stop

E: Message causes an emergency stop

F: Resets the feed-rate enabling

0 to 2: Priority

REFRESH=n

Time interval in [ms]

All variables with the **/c** switch are checked and, if required, redisplayed.

Value range: 100 to 100 000 [ms]

Default setting: 400 ms

KBD

This command is needed only if relatively long texts are to be moved with the cursor keys. It assigns the keyboard to the PLC window as long as it is visible on the screen.

If the page limits were defined with **\f**, it is possible to scroll and to page up and down with the arrow keys.

If the mask contains elements with the **/e** or **/i** switch, the keyboard is automatically assigned to the PLC window. In this case the arrow keys jump from input field to input field.

Soft keys, screen switch-over keys, operating mode keys, special function keys (MODE, PGM-MGT, CALC) always remain assigned to the NC.

LINE=xpix, ypix

Draws a line from the current position to the designated position. The designated position is then assumed as the current position.

xpix = Distance in pixels from the left edge of the current page

ypix = Distance in pixels from the upper edge of the window

ILINE=xpix, ypix

Draws a line from the current position to a position that is offset by **xpix**, **ypix**.

xpix, ypix = Line lengths in x, y.

Then the current position is corrected by **xpix, ypix**.

LINestyle=SOLID/DASH/LDASH

Defines the line type for the **LINE/ILINE** command.

SOLID = Solid line

DASH = Dashed line (interrupted line)



LDASH = Dot-and-dash line

Default setting: **SOLID**

The width of the line is one pixel and cannot be changed.

FILE= <table name>

Opens a table for access with **tableread**.

You cannot open more than one table at a time. If the **FILE** command is called more than once, the previously opened table is closed. At the end of the mask the table is closed automatically.

tableread (line,column)

Reads field contents from the table that has been opened with **FILE=**. With the **/c** switch you can show the field contents of a table and update them cyclically.

Example:

CHARSIZE = SMALL;

LINESTYLE = SOLID;

FILE = TNC:\P_PLATZ.P;

COLOR=1;

"%s", tableread(0, "P-NR");

"%s", tableread(2, "P-NR");

Mathematical expressions for screen positions

If a numerical value is expected as a parameter for a special function, then a mathematical expression can be written in integer arithmetic.

The operators and priority rules used in the C programming language apply.

Available functions: +, -, *, /, %, &, |, ^.

The mathematical expression may have the following variables:

Variable	Meaning
PAGE	Number of the current page, beginning from zero
XPOS	X position of the cursor in pixels
YPOS	Y position of the cursor pixel
LINEDIST	Currently defined line spacing in pixels
ROWDIST	Currently defined character spacing, width of an ASCII character
XSIZE	Width of the screen window in pixels
YSIZE	Height of the screen window in pixels

Input fields

With the switches **/e** and **/i** you can assign input fields to the variables:

/e: Shows the current value, which can be overwritten.

/i: Shows an empty field in which a new value can be entered.

In addition, both switches **/e** and **/i** can be given an identifier **xxx** (**/e = xxx**, **/i = xxx**), where **xxx** is a positive whole number. With Module 9211 you can then ascertain whether the cursor is located in this field.

The switch **/s = xxx** is used to create a field in which no entries can be made. By entering the identifier **xxx** it is possible to ascertain with Module 9211 whether the cursor is located in this field.

If the switches **/e**, **/i** or **/s** are used, the cursor keys function as jump commands from input field to input field. The current page is scrolled if necessary. Any text between the input fields might no longer be shown.



Note

Do not edit any text before the first input field or after the last.

The C command "printf" requires a format that defines the length of the numerical field:

- ▶ Save this format in the mask file. Otherwise the length of the input field depends on the coincidental content of the associated variable.

For the input function this format instruction is converted internally into a form suitable for the C command "scanf":

printf: %[flags][digits1][.[digits2]][1]conversion_char

scanf: %[digits1] [size] conversion_char



Note

Special characteristics

- %d, %e
The size information "1" can be omitted. Floating-point variables are of the double type and automatically add to this information.
- %g
Do not use. Causes errors.
- %i
Do not use. Any number entered with leading zeros would be interpreted as an octal number.
- %u
Works correctly only in the definition range for the respective variables.
- The size indicator **h** (short integer) of the "scanf" function cannot be written. All integer variables are automatically expanded to 32 bits for input and output.



You can enter data in the input field through the ASCII keyboard and the numerical keys.

The following keys have special functions:

Function	Meaning
CE	Clears the input field or the displayed error message
ENT	Transfers the input value to the variable and sets the highlight to the next input field. If the input value is syntactically incorrect or exceeds the numerical range of the assigned variable, the error message ENTRY VALUE INCORRECT appears.
NOENT	Shows the original content of the field again and sets the highlight to the next input field.
-/+	If the input value begins with the algebraic sign – or +, the sign is switched.
<x	If the field was already edited, the last character of the entry is deleted. Otherwise the displayed value is assumed into the editing memory and the cursor is set to the end of the input value.

Opening or clearing screen mask for the PLC window

- ▶ Define the file names and path of the screen mask in one of the string memories or in an immediate string.
If no path name is specified, the path for the language indicated in MP7230.3 (help files) is used.

You can also use the functions from the soft-key project file to display a large PLC window (see "PLC soft keys" on page 1372).

Module 9210 Open or clear screen mask for the PLC window

With this module you can activate or clear the display in the large PLC window.

If you use the TEXTFILE command to show a text file in PLC mask files and this file does not exist in the set language (MP7230.3), the NC software now also searches the English language directory (PLC:\LANGUAGE\ENGLISH) for the text file.

Call:

PS B/W/D/K/S<Clear no. of string memory/file name/PLC window>
-1: Clear PLC window

CM 9210

PL B/W/D <Status/Error>
0: Mask opened / mask cleared
1: PLC window not yet ready again
-1: Error

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	11	The transferred value is outside of the permissible range (-1 to 3), or an invalid constant string was transferred.

If a faulty mask file was activated, an error message appears in the PLC window.

Error message	Meaning
COMMAND LIMITER ";" MISSING	End of command not found
UNKNOWN PARAMETER TOKEN	Unknown keyword
UNKNOWN COMMAND	Unknown command
STRING FORMAT ERROR	Impermissible format instruction
STRING NOT CLOSED	String end missing
TOKEN TOO LONG (>32 CHAR)	Variable name is too long
PARAMETER INDEX MISSING	Index is missing Closing bracket "]" is missing
SOURCE FILE NOT OPENED	Source file is not opened
TEMPORARY FILE NOT OPENED	Temporary target file is not opened
TOO FEW PARAMETERS	Too few parameters for format instruction
WRONG COMMAND PARAMETER	Parameter does not fit the format
WRONG PARAMETER SWITCH	Incorrect switch



Module 9211 Status of the large PLC window

With this module you can interrogate the status of the large PLC window.

Number	Return code
0: Status	0: No screen mask activated 1: Screen mask was activated 2: Screen mask is being activated 3: Screen mask could not be activated
1: Horizontal size	0: No PLC window displayed > 0: Number of pixels
2: Vertical size	0: No PLC window displayed > 0: Number of pixels
3: Displayed page	Displayed page of the screen mask
4: Current field	0: No cursor or the cursor is not located in a field identified with /s = xxx, /e = xxx or /i = xxx. >0: Return of the value xxx of a field identified with /s= xxx, /e = xxx or /i = xxx.

Call:

PS B/W/D/K <Number>

CM 9211

PL B/W/D <Status information>

-1: Error

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Invalid number of the status information

8.3 PLC soft keys

You can display your own soft keys on the iTNC and on the HR 420 through the PLC in all operating modes. You can easily define the soft-key projects with PLCdesignNT as of version 2.3 and MenuDesign (included in the PLCdesignNT package as of version 2.3)

When a PLC soft key is pressed the NC enters the soft-key number in W302 (horizontal soft-key row), W304 (vertical soft-key row) or W306 (handwheel soft keys). On the rising edge of the keystroke it enters the soft-key number; on the falling edge it enters -1. The PLC can enter -1 itself after recognizing the soft-key number.

8.3.1 Soft-key project file for screen



Note

The vertical soft keys can only be used with a BF 150.

The PLC soft-key structure is defined in a soft-key project file using various keywords. The number of submenus depends only on the iTNC memory.

In the OEM.SYS file, the PLC soft-key project file *.SPJ is entered through the keyword **SOFTKEYPROJECT =**. After acknowledgment of **Power interruption** the resource file with the same name and the extension .SYS is generated from this file. The results of this evaluation are stored in an ASCII file with the name **<Name of the soft-key project file>.SYS.LOG**.

The soft-key structure is displayed immediately. The PLC soft keys can be influenced with Modules 9205 to 9207. Module 9204 refreshes the PLC soft keys, which is necessary after Modules 9203 and 9207 are executed. Module 9208 determines the status information of the PLC soft keys.

In the standard setting the soft-key number is transferred to the PLC via W302/W304 after the NODE, BACK and ACTION soft keys have been pressed. When the key is released, -1 is confirmed. Operands can also be coupled directly to soft keys in the project file or with Module 9206. With Module 9205, you can also select a word address other than W302/W304.

BLANK soft keys are not reported to the PLC.



Note

The states of the assigned operands (STATUS markers or word) of the ACTION, PULSE, CHECK and RADIO soft keys and the display are checked cyclically. If these operands change, the display is adjusted accordingly. For example, if the marker of a CHECK soft key changes from 0 to 1, the display of the soft key is changed to "pressed."

If the value of the word address of a soft-key group, in which a group code is saved, is changed by the PLC, then this value is checked. If the value is valid, then the display of the group is adjusted accordingly.

**Miscellaneous
keywords in the
soft-key project file**

Entry in *.SPJ	Meaning
;	Comment
SKPATH	Path of the soft-key graphic files
SOFT KEY	<p>Soft-key definition. The name of the soft-key graphic file and the name of the soft key must be specified. The maximum permissible length of the soft-key name is 23 characters.</p> <p>First entry = soft-key number 0, Second entry = soft-key number 1, etc.</p> <p>With ACTION soft keys, the soft-key number is confirmed via W304 (unless changed by Module 9205) to the PLC.</p> <p>You can use the BMX and BMP formats. Properties:</p> <ul style="list-style-type: none"> ■ Color depth: 16 or 24 bits ■ Soft key for BF 2xx: 120 x 74 <p>You will find information on creating *.BMX files in the online help for PLCdesignNT.</p>

**Keywords in the
soft-key menus**

Entry in *.SPJ	Meaning
;	Comment
#include	An additional soft-key project file can be included. The name and path of this file must be entered.
Keywords for menu structuring:	
SKMENU	Beginning of the definition of a soft-key menu. The name of the menu must be specified. The soft keys are automatically assigned to the correct menu rows. Also note the additional parameters for this keyword on page 1376.
ENDSKMENU	End of the definition of a soft-key menu
NODE	Soft key jumps to a submenu. Is confirmed via W302/W304 to the PLC. The soft-key name and the name of the submenu must be indicated. Also note the additional parameters for this keyword on page 1376.
BLANK	Empty soft key. You can also specify a soft-key name.
BACK	Soft key jumps to a submenu. Is confirmed via W302/W304 to the PLC. The soft-key name and the name of the submenu must be indicated. Also note the additional parameters for this keyword on page 1376.

Entry in *.SPJ	Meaning
END	Closes an open pop-up menu. The soft key is reported to the PLC via W302/W304 for the duration that the key is pressed. Also note the additional parameters for this keyword on page 1376.
Keywords for function soft keys:	
ACTION	Function soft key. Is confirmed via W302/W304 to the PLC. The soft-key name must be indicated. Also note the additional parameters for this keyword on page 1376.
PULSE	The soft key is reported to the PLC via W302/W304 for the duration of the PLC cycle. A soft-key name must be indicated. Also note the additional parameters for this keyword on page 1376.
CHECK	A coupled marker is set the first time it is pressed, and is reset the next time. The soft key is reported to the PLC via W302/W304 for the duration that the key is pressed. A soft-key name must be indicated. Also note the additional parameters for this keyword on page 1376.
RADIO	From any group of these soft-key types, no more than one soft key can be pressed. Assigned status markers for the PLC are also set uniquely. This means that one status marker is always set for the PLC, but not more than one marker can be set at the same time (1-out-of-n rule). However, this does not apply to status markers that do not change within the PLC program (set or reset). If assigned status markers within the PLC program are changed, then the 1-out-of-n rule can be broken for one PLC cycle. It is corrected again during the next cycle. The soft key is reported to the PLC via W302/W304 for the duration that the key is pressed. It is also possible to define more than six RADIO soft keys to one group. A soft-key name must be indicated. Also note the additional parameters for this keyword on page 1376.
STATE	Multiple states can be managed for this soft key. The active state depends on the number of times the soft key was pressed. In order to display these states, the BMX file of the soft key must include the corresponding number of possible states, and a PLC word memory must be assigned. If a PLC bit memory is assigned, only two states can be managed. In the following example, a value is assigned to the PLC word W1000 via a BMX soft key with five possible states. Values from 0 to 4 are assigned here: SOFTKEY spindle_attr.bmx State_Softkey STATE State_Softkey STATUS:W1000 STATES:5 Also note the additional parameters for this keyword on page 1376.



Entry in *.SPJ	Meaning
Global optional keywords:	
TREEFILE	Declares an alternate system-file name
DEFFILE	Declares an alternate definition-file name
HELPPFILE	Declares an alternative help-file name for the context-sensitive help
MODOMAIN	Declares an alternative file name for language-sensitive text soft keys
MOPATH	Declares the location of the text files for language-sensitive text soft keys



Additional parameters for the keywords

Entry in *.SPJ	In connection with	Meaning
VR00T	SKMENU	The menu for the vertical soft-key row is defined in the header of the main menu.
HR00T	SKMENU	The menu for the horizontal soft-key row is defined in the header of the main menu.
EMODE	SKMENU	The menu for the programming operating modes is defined in the header of the main menu.
MMODE	SKMENU	The menu for the machine operating modes is defined in the header of the main menu.
ENABLE:<Marker>	NODE, BACK, END, ACTION, PULSE, CHECK, RADIO, STATE	Depending on the marker status, the soft key is either locked (marker = 0) or enabled (marker = 1). A locked marker is shown "inactive."
HIDE:<Marker>	NODE, BACK, END, ACTION, PULSE, CHECK, RADIO, STATE	The parameter HIDE assigns a marker to a soft key. If this marker receives the value 1, then that soft key is hidden and exchanged for an "empty" soft key.
STATUS:<marker or word>	NODE, BACK, END, ACTION, PULSE, CHECK, RADIO, STATE	An operand is assigned to the soft key (in addition to W302/W304). If a marker is indicated and the soft key is pressed, the marker is set. If a word is indicated, the soft key number is entered (index number from the *.sys file).
REPEATDELAY: <Delay time in ms> REPEATINTERVAL: <Interval in ms>	ACTION, PULSE	An "autorepeat function" can be realized for a soft key by combining these two parameters. The function of the soft key is performed after a delay (REPEATDELAY) at a specified interval (REPEATINTERVAL).
STATES:<Marker>	STATE	The STATES parameter is used to tell a soft key how many different states it can assume as the result of repeated pressing. The assigned soft-key image file must support the number of states (see the BMXdesign software from HEIDENHAIN for more information)
POPUPMENU:<Menu name>	NODE, ACTION, PULSE, CHECK, RADIO	The menu Menu name is shown in the other soft-key row (either vertical or horizontal).
CLOSEPOPUPMENU	NODE, BACK	A menu opened with the POPUPMENU:<Menu name> parameter is closed again.
LARGEWINDOW: <Mask file for PLC window>	NODE, ACTION, PULSE, CHECK, RADIO	A large PLC window with the given mask file is displayed across the entire screen.



Entry in *.SPJ	In connection with	Meaning
SMALLWINDOW: <Mask file for PLC window>	NODE, ACTION, PULSE, CHECK, RADIO	A large PLC window with the given mask file is opened instead of the graphics/status window. However, the user can extend it over the entire screen.
CLOSEPLCWINDOW	NODE, ACTION, PULSE, CHECK, RADIO	A large PLC window opened with the LARGEWINDOW: or SMALLWINDOW: parameter is closed again.
FirstInGroup	ACTION, PULSE, CHECK, RADIO	<p>Defines a soft-key group. This keyword identifies the first soft key of the respective soft-key group. A group consists of at least two soft keys with the same name, which can even be collected over multiple soft-key rows (menus in the same level). The functionalities listed above are set depending on the soft-key keyword.</p> <p>Evaluation via group code: So that the operator entry can be evaluated via a group code, the STATUS: <word> parameter must be entered for the first soft-key in the group. No further STATUS: <word> definitions may be entered within this group.</p> <p>ACTION, PULSE and CHECK soft keys: The number of the soft key (within a soft-key group) is transferred bit-coded in the dual system in the given word, for example: 1st soft key (value = 1) 2nd soft key (value = 2) 3rd soft key (value = 4) 4th soft key (value = 8) ... e.g. 9th soft key (value = 256)</p> <p>For groups with CHECK soft keys, more than one soft key can be pressed at the same time. Here the respective value of the group code is summed, and the result is saved in the given word.</p> <p>RADIO soft keys: If this parameter is used in combination with a RADIO soft-key group, the value of the group code is saved as follows: 1st soft key (value = 0) 2nd soft key (value = 1) 3rd soft key (value = 2) 4th soft key (value = 3) ...</p>

The token **SOFTKEYPRESS** in **OEM.SYS** can be used to influence the display and behavior of the PLC soft key types CHECK and RADIO. If **SOFTKEYPRESS = ON** is entered, the soft-key graphics and the states of the PLC markers already change when the soft key is pressed. If this entry is missing from the OEM.SYS, the changes listed below are performed as previously. This means that all changes listed in the table are not performed until the soft key is released.

The behavior of the following types of soft keys can be influenced with **SOFTKEYPRESS = ON**:

SK type	Resulting behavior
CHECK	If pressing the key causes the soft key to switch on,
	the change to another soft-key graphics (if BMX file is available) is performed when the soft key is pressed.
	the change is performed in the PLC status marker when the soft key is pressed.
	If pressing the key causes the soft key to switch off,
	the change to another soft-key graphics (if BMX file is available) is performed when the soft key is released.
	the change is performed in the PLC status marker when the soft key is released.
RADIO	the change of the active RADIO soft key is performed within the group when the soft key is pressed.
	the change to another soft-key graphics (if BMX file is available) is performed when the soft key is pressed.
	the change is performed in the PLC status marker when the soft key is pressed.

The system parameters (see "Available system parameters:" on page 1686) for conditional compilation can also be used in the soft-key project file.



Example of a soft-key project file

```

Soft-key          PLC SOFTKEY Project File - Version 1.0
number
(confirmed to    SKPATH 'PLC:\SK\1024x768\'
the PLC):
  0 SOFTKEY 'BACK.BMX'      BACK_SK
  1 SOFTKEY 'MAG.BMX'      MAG_SK
  2 SOFTKEY 'MAG_CCW.BMX'  MAG_CCW_SK
  3 SOFTKEY 'MAG_CW.BMX'   MAG_CW_SK
  ... ..

```

Definition of the soft keys with names for graphics and soft keys

```

  SKMENU RootMenuVM VROOT MMODE
  NODE MAG_SK Magazine
  BLANK
  BLANK
  BLANK
  BLANK
  BLANK
  ; Row 2
  CHECK DIAG_SK POPUPMENU:Diag_Mnu
  ACTION CHIP_BACK_SK
  BLANK
  BLANK
  RADIO SPI_OFF_SK
  RADIO SPI_ON_SK
  ENDSKMENU

```

Main menu, vertical soft-key row, machining modes

```

  SKMENU RootMenuHM HROOT MMODE
  ...
  BLANK
  ENDSKMENU

```

Main menu, horizontal soft-key row, machining modes

```

  SKMENU Magazine
  ACTION MAG_CCW_SK STATUS: MG_SK_TC_mag_left_ccw
  ACTION MAG_CW_SK STATUS: MG_SK_TC_mag_right_cw
  BACK BACK_SK
  ENDSKMENU

```

Entry of an active marker

```

  SKMENU Diag_Mnu
  ACTION DIAG_T_SK
  BLANK
  BLANK
  BLANK
  CHECK MACH_SK LARGEWINDOW:Machine.A
  CHECK MFUNC_SK SMALLWINDOW:MFUNCT.A
  ENDSKMENU

```

Submenu "Diagnosis" Display PLC window with mask file

```

  SKMENU row_machine
  ACTION SK_NC_start FIRSTINGROUP STATUS:NP_W310_GRO
  ACTION SK_NC_stop
  ACTION SK_axes_plus FIRSTINGROUP STATUS:NP_W312_GR
  ACTION SK_axes_minus
  ...

```

Group with number saved in W310
Group with number saved in W312

		Set	Reset
W302	Number of the horizontal PLC soft key that was pressed	NC	NC
W304	Number of the vertical PLC soft key that was pressed	NC	NC

Module 9203 Activate PLC soft-key menu

The PLC soft-key project can be defined in such a way that it is effective only for specific modes of operation. This might be useful, for example, if you want to use different PLC soft-key projects for the machine and programming modes of operation.

The module supplies the resource handle for Modules 9204 to 9208.

Call:

```

PS   D           <Resource handle>
          0 when it is called for the first time, otherwise
          Resource handle from Module 9203

PS   B/W/D/K/S<0>
          0 or "" is given (is no longer evaluated)

PS   B/W/D/K     <Number of PLC soft-key menu>
          Number of the soft-key menu
          Value < 0:
            - Returns the current resource handle

PS   B/W/D/K     <Mode>
          1: Vertical PLC soft-key row

PS   B/W/D/K     <Mode of operation>
          0: Programming modes of operation
          1: Machine modes of operation
          2: Programming and machine modes of operation

CM   9203
PL   D           <Resource handle>
          0: Error code in W1022

```

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Resource-handle overflow, incorrect resource handle, incorrect mode, incorrect operating mode or number of PLC soft-key root menu negative.
	3	Incorrect string number or incorrect string
	20	Module was not called in a spawn or submit job
	44	Error in the resource file



Module 9204 Update the PLC soft keys

If you want to restructure the PLC soft keys, you must call Module 9204. This is necessary each time you have called Modules 9203 and 9207.

Call:

PS D <Resource handle>
Resource handle from Module 9203

CM 9204

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Incorrect resource handle
	20	Module was not called in a spawn or submit job

Module 9205 Set the word for acknowledgment of PLC soft keys

Module 9205 can be used to define another word, in addition to W302/W304, in which the pressing of PLC soft keys is acknowledged. This can be done for the complete project file, individual PLC soft-key menus or individual PLC soft keys. Changes in individual PLC soft keys affect the entire project file.

Call:

PS D <Resource handle>
Resource handle from Module 9203

PS B/W/D/K <Mode>
0: Complete project file
1: Individual menu
2: Individual PLC soft key

PS B/W/D/K <Number of PLC soft-key menu/Number of PLC soft key>
If "complete project file": Non-functional, transfer 0

PS D <Reserved>
0 transferred

PS B/W/D/K <PLC word address for PLC soft keys>
-1: W304

CM 9205

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Invalid PLC word address
	2	Incorrect resource handle or incorrect mode
	3	Invalid PLC label address
	20	Module was not called in a spawn or submit job
	44	Error during setting of setup parameters



Module 9206 Change the settings of the PLC soft keys

With Module 9206, the settings of individual PLC soft keys in the PLC soft-key structure can be changed:

- PLC soft keys can be locked and unlocked. Locked PLC soft keys cannot be used.
- PLC soft keys can be coupled to new operands. This way the status of the PLC soft key is directly available in the PLC program.
- Couple PLC soft-keys to operands that
 - Unlock the PLC soft keys in a set state
 - Lock the PLC soft keys in a reset state

If a locked PLC soft key is pressed, it sets the marker M4577.

Call:

```
PS   D           <Resource handle>
                        Resource handle from Module 9203
PS   B/W/D/K     <Soft-key number>
PS   B/W/D/K     <Function>
                        0: Lock soft key
                        1: Unlock soft key
                        2: Decouple soft key from assigned operand
                        3: Couple new operand to soft key
                        4: Decouple the soft-key unlocking/locking from the
                           assigned operand
                        5: Couple the soft-key unlocking/locking with the operand
PS   B/W/D/K     <Operand address>
                        Only for function 3 and 5, otherwise transfer 0
PS   B/W/D/K     <Operand type>
                        Only for function 3 and 5, otherwise transfer 0
                        0: Marker M
                        1: Input I
                        2: Output O
                        3: Counter C
                        4: Timer T
```

CM 9206

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Incorrect resource handle or incorrect function
	20	Module was not called in a spawn or submit job
	44	Error during setting of setup parameters

Module 9207 Replace PLC soft keys

With Module 9207, individual PLC soft-keys can be replaced by another PLC soft key. The change can be applied to the entire project file or only to an individual menu. If a soft key is to be replaced in the entire project file, the source menu is excepted from it so that the soft key can remain to make it possible to reverse the replacement.

Call:

```
PS    D           <Resource handle>
                Resource handle from Module 9203
PS    B/W/D/K    <PLC soft-key number source>
PS    B/W/D/K    <Number of PLC soft-key menu source>
PS    B/W/D/K    <PLC soft-key number target>
PS    B/W/D/K    <Number of soft-key menu target>
                -1: Entire resource file except source menu
PS    B/W/D/K    <Function>
                0: Replace soft key
CM    9207
```

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Incorrect resource handle or incorrect function
	20	Module was not called in a spawn or submit job
	44	Error during setting of setup parameters



Module 9208 Status information of the PLC soft keys

Call:

PS D <Resource handle>
Resource handle from Module 9203
PS B/W/D/K <Function>
0: Number of the current soft-key menu
PS B/W/D/K <Reserved>
0 transferred
CM 9208
PL D <Status information>

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Incorrect resource handle or incorrect function
	20	Module was not called in a spawn or submit job
	44	Error finding the status information

8.3.2 Soft-key project file for HR 420, HR 5xx

Via the PLC, the iTNC can also manage freely-definable soft keys on the HR 420 and HR 5xx handwheels. The corresponding entries are made in the general soft-key resource file (*.spj) of the PLC project, in which the soft keys for the vertical and horizontal soft-key rows of the iTNC screen are defined. It is possible to switch from the basic menu of the HR, via the **FCT** (Function) soft key, to a freely-definable submenu structure, each of which contains 5 soft keys, whose description and functions are specified in the soft-key resource file.

This menu is displayed in the fourth row of the HR 420/HR 5xx . Four ASCII characters are available for each soft key. However, if each soft key uses all four characters, then there is no empty space between the soft-key designations.

In addition, you can optionally have a menu title with 20 characters displayed in the third row of the HR display for the main menu and each submenu. The scope of function of the soft keys on the handwheel is limited somewhat compared to the vertical and horizontal soft keys of the iTNC screen.

The following definitions are possible:

Entries for the menu definition and type of soft key in the HR 420/HR 5xx menu:

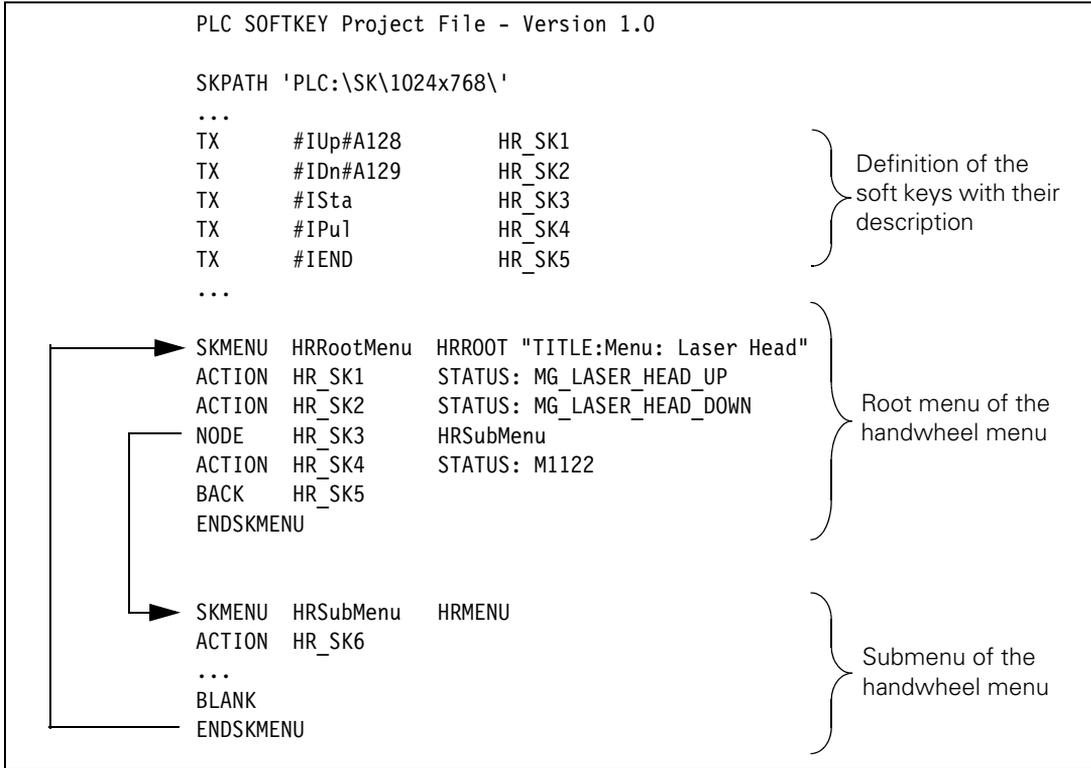
Entry	Parameter	Description
;		Comment
SKMENU <menu name> ENDSKMENU		Beginning or end of the definition of a soft-key menu. The name of the menu must be given for SKMENU (e.g. "HRRootMenu"—see the example). The soft keys are aligned on the HR in the sequence in which they appear in the file. Also note the additional parameters for this keyword.
	HRROOT	Freely-definable root menu when called from the basic handwheel menu via the FCT (Function) soft key
	HRMENU	Freely-definable submenu, called via the keyword NODE ...
	TITLE: <name>	Menu title: Freely definable text in the third line of the HR 420/HR 5xx The parameter TITLE: can also be surrounded by quotation marks. This permits blank spaces in the menu title.
NODE <submenu name>	Soft key jumps to a submenu. Is confirmed via W306 to the PLC. The soft-key name and the name of the submenu must be indicated.	
BACK	Soft key jumps to a submenu. Is confirmed via W306 to the PLC. The soft-key name and the name of the submenu must be indicated.	
BLANK	Empty soft key, is shown as "..". You can also specify a soft-key name.	
ACTION <soft-key name>		Function soft key. Is confirmed via W306 to the PLC. The soft-key name must be indicated.
	STATUS: <marker or word>	An operand is assigned to the soft key (in addition to W306). If a marker is indicated and the soft key is pressed, the marker is set. If a word is indicated, the soft key number is entered (index number in the *.sys file, e.g. Softkey.sys).
PULSE <soft-key name>	The soft key is reported to the PLC via W306 for the duration of the PLC cycle. A soft-key name must be indicated.	
The soft-key types RADIO , CHECK and STATE may not be used. Other parameters, such as ENABLE , HIDE , REPEAT , etc. are not yet available.		



Additional keywords and parameters for the definition of soft keys in the HR 420/HR 5xx menu:

Keyword	Parameter	Description
TX	#I<text>	Language-neutral text for the description of a soft key (up to 4 ASCII characters)
	#A<ASCII value>	Input of an ASCII value as a possibility for displaying special characters. This value must be entered in decimal notation as three digits. Special characters include - 128 = Arrow up - 129 = Arrow down - 133 = Return symbol in the menu
	A combination of the parameters named above is possible, e.g. #IUp#128 to display Up↑	

The following example shows the configuration of a soft-key menu for an HR 420. These settings must be fully integrated in the soft-key project file of the iTNC 530.



8.3.3 Soft-key rows

With Module 9200 you can display entire soft-key rows. With Module 9201 you can show individual soft keys. With Module 9202 you can switch to the display with PLC soft keys and PLC windows.

Display/delete PLC soft-key row

- ▶ In the system file PLC:\PLCSOFTK.SYS, enter the names and path of the required soft-key files. With the sequence of your entries you specify the soft-key number: Line 0 = soft-key number 0, etc. One soft-key level can consist of up to four soft-key rows, i.e. 32 soft keys per level.
- ▶ When calling the module, indicate the row to be displayed first.
- ▶ With the transfer parameter, specify how the soft keys should be displayed:
 - After the screen management key is pressed, i.e. after the PLC window is selected.
 - In the current operating mode:
In this case the NC soft keys are overwritten.
- ▶ Specify whether the NC soft keys should be overwritten or whether the PLC soft keys should be appended to the NC soft keys. If you append the PLC soft keys, a separate list is opened. Only one PLC soft-key row can be appended.

		Set	Reset
W302	Number of the horizontal PLC soft key that was pressed	NC	NC

Module 9200 Display/delete PLC soft-key row

The soft keys to be activated are specified in a constants field by their line numbers. If there is no PLCSOFTK.SYS file, or if the lines indicated in the constants field do not exist, no soft-key row is generated.

Can only be called from the sequential program.

Call:

PS B/W/D/K/KF<Select or delete soft-key address>

1: Delete soft-key level

KF: Address of soft-key selection

PS B/W/D/K <Soft-key row>

0 to 3: Soft-key row to be displayed

PS B/W/D/K <Soft-key mode>

0: Soft-key row for displayed PLC window

1: Soft-key row in current operating mode

2: Append soft-key row to NC soft keys

CM 9200

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Incorrect transfer parameters (e.g. KF address not in address range of the PLC code)
	2	Line nr. < 0 (not -1) in the constants field
	24	Module was called in a spawn job or submit job
	25	More than 32 elements in the constants field

Displaying/deleting PLC soft keys

► Procedure for displaying/deleting a PLC soft key

Module 9201 Display/delete PLC soft key

If no PLCSOFTK.SYS file exists, or if the specified line does not exist, no soft key is generated. In an existing PLC soft-key level, the soft key is displayed/deleted at the specified position.

Can only be called from the sequential program.

Call:

PS B/W/D/K <Soft-key number (line no./delete)>

0: Line no.

-1: Delete soft key

PS B/W/D/K <Position no.>

0 to 31

PS B/W/D/K <Soft-key mode>

0: Soft key for displayed PLC window

1: Soft key in current operating mode

2: Append soft key to NC soft key

CM 9201

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Transfer parameter out of value range
	2	Line nr. < -1
	24	Module was called in a submit job

Selecting/deselecting PLC soft keys and PLC windows

Module 9202 Select/deselect PLC soft keys and PLC windows

With Module 9202 you activate the display with PLC windows or the PLC soft-key display. This module works like the screen management key.

Call:

PS B/W/D/K <Display mode>

0: PLC soft key/window deselected

1: Small PLC soft key/window deselected

2: Large PLC soft key/window deselected

3: Large PLC soft key/window selected while table editor is active

CM 9202

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Transfer parameter out of value range

8.4 Keystroke simulation

HEIDENHAIN contouring controls have two control panels:

- iTNC control panel
- The machine operating panel from the machine tool builder

The control and operating panels are attached to the MC 6xxx via USB interfaces.

The key code of the iTNC keyboard unit is evaluated directly by the NC.

The machine operating panel has its own PLC inputs and outputs. These are transmitted via HSCI. You must configure the inputs and outputs using IOconfig, and evaluate them in the PLC.

8.4.1 iTNC control panel

The key code of the iTNC keyboard unit is evaluated by the NC. The key code is displayed in W274 while a key is being pressed. (see "Codes for keystroke simulation" on page 1400).

If you press a disabled key, marker M4577 is also set.

The following modules can influence keys and soft keys:

- Module 9180: Simulate NC keys
- Module 9181: Disable individual NC keys
- Module 9182: Re-enable individual NC keys
- Module 9183: Disable groups of NC keys
- Module 9184: Re-enable groups of NC keys
- Module 9186: Call a soft-key function
- Module 9187: Status of a soft-key function call
- Module 9188: Call a soft-key function

With MP4020 bit 9, specify whether a simulated key should only be transmitted to the NC, or also to an active PLC window.

With MP4020 bit 10, specify whether a disabled key should be disabled only for the active PLC window, or for the active PLC window and for the NC.

		Set	Reset
W274	Code of the depressed key	NC	NC
M4577	Disabled key was pressed	NC	PLC

MP4020 PLC compatibility

Input:

Bit 9 – Behavior of a simulated key

0: Simulated key is transferred immediately to the NC

1: Simulated key is processed first by an active PLC window before being transferred to the NC

Bit 10 – Behavior of a disabled key

0: Locked key only works on the active PLC window

1: Locked key works on neither the active PLC window nor on the NC

Module 9180 Simulation of NC keys

With this module you can simulate the activation of NC keys and soft keys. You transfer the code of the desired key.

If you transfer the code value zero, the number of occupied elements in the keystroke queue is returned. In this case there is no keystroke simulation.

Call:

PS B/W/D/K <Key code>

CM 9180

PL B/W/D <Number of occupied elements / error status>

0: Key code was transferred, key queue is empty

1 to 16 : Key code was not yet simulated, max. 16 entries in the keystroke queue are possible

-1: For error see W1022

Error code:

Marker	Value	Meaning
M4203	0	NC key was simulated
	1	Error code in W1022
W1022	1	Transferred parameter > maximum value
	2	Transferred parameter invalid
	22	Keystroke queue overflow

Module 9181 Disable individual NC keys

With this module you can disable individual NC keys.

If you press a disabled key, marker M4577 is set.

Call:

PS B/W/D/K <Key code>

CM 9181

PL B/W/D <Error status>

0: NC key disabled

-1: For error see W1022

Error code:

Marker	Value	Meaning
M4203	0	NC key was disabled
	1	Error code in W1022
W1022	1	Transferred parameter > maximum value
	2	Transferred parameter invalid

Module 9182 Re-enable individual NC keys

With this module you cancel the effect of Module 9181.

Call:

PS B/W/D/K <Key code>

CM 9182

PL B/W/D <Error status>

0: NC key enabled

-1: For error see W1022

Error code:

Marker	Value	Meaning
M4203	0	Disabling was canceled
	1	Error code in W1022
W1022	1	Transferred parameter > maximum value
	2	Transferred parameter invalid

Module 9183 Disable groups of NC keys

The key-group codes are:

- 0: All keys
- 1: ASCII
- 2: Horizontal soft keys, Page Up/Down
- 3: Cursor, ENT, NOENT, DEL, END, GOTO
- 4: Numbers, algebraic signs, decimal point, actual position capture
- 5: Operating modes
- 6: Block opening keys
- 7: Vertical soft keys, switch-over key

Call:

PS B/W/D/K <Key-group code>

CM 9183

PL B/W/D <Error status>

0: Group of NC keys disabled

-1: Transferred parameter > maximum value

Error code:

Marker	Value	Meaning
M4203	0	The group of NC keys was disabled
	1	Error code in W1022
W1022	2	Transferred parameter invalid

Module 9184 Re-enable groups of NC keys

With this module you cancel the effect of Module 9183.

Call:

PS B/W/D/K <Key-group code>
CM 9184
PL B/W/D <Error status>
0: Group of NC keys enabled
-1: Transferred parameter > maximum value

Error code:

Marker	Value	Meaning
M4203	0	Disabling was canceled
	1	Error code in W1022
W1022	2	Transferred parameter invalid

Module 9186 Call a soft-key function

With this module you can call certain soft-key functions in the machine operating modes.

Do not call a new function until the previous one is completed. You can interrogate this condition with Module 9187.

For a soft-key function to be simulated it must be displayed either in the foreground or background operating mode. Otherwise calling the module has no effect. Module 9187 reports the error.

Call:

PS B/W/D/K <Number of the soft-key function>
0: INTERNAL STOP
1: M output
2: S output
3: PROBE FUNCTION
4: PASS OVER REFERENCE MARK
5: RESTORE POSITION
6: INCREMENTAL JOG
7: Feed-rate limitation F MAX
8: MANUAL TRAVERSE

CM 9186

Error code:

Marker	Value	Meaning
M4203	0	Soft-key function was called
	1	Error code in W1022
W1022	1	Parameter out of value range
	28	Previous call not ended

Module 9187 Status of a soft-key function call

Immediately after Module 9186 is called, the status 1 (soft-key function not yet completed) is set—regardless of whether the function can be run in the current operating mode. Module 9186 cannot be called again until status 0 or 2 is set. The error status 2 is erased if Module 9186 is called again or if power is switched on.

Call:

CM 9187

PL B/W/D <Status>

0: Soft-key function completed or none called

1: Soft-key function not yet completed

2: Error: Soft-key function cannot be completed because soft key is not available or operating mode is incorrect

Module 9188 Call a soft-key function

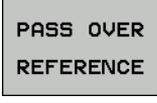
Module 9188 enables you to simulate soft-key functions. The soft-key designation is used for programming.

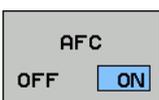
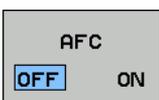
Conditions:

- For a soft-key function to be simulated the soft key must be contained in the soft-key row that is displayed on the screen or in one of the soft-key rows that are in the same level but on the following screen pages.
- The parameter **Attribute** defines whether a function should be activated although the soft key is not displayed in the soft-key row on the screen, but is contained in one of the following soft-key rows.
- As opposed to module 9186, the status can be determined with error code W1022 as soon as the module is called.



The following soft keys can be simulated using the designators listed below:

Soft key	Soft-key designator
	SK_NC_Manual_operation_TOUCH_PROBE
	SK_NC_Manual_operation_PRESET_TABLE
	SK_NC_TOOL_TABLE
	SK_NC_Manual_operation_INCREMENT_OFF
	SK_NC_Manual_operation_INCREMENT_ON
	SK_NC_Manual_operation_PASS_OVER_REFERENCE
	SK_NC_Manual_operation_SET_DATUM
	SK_NC_Program_run_TOOL_USAGE_TEST
	SK_NC_Program_run_F_MAX
	SK_NC_Program_run_AUTOSTART
	SK_NC_Program_run_AUTOSTART_ON

Soft key	Soft-key designator
 <p>AUTOSTART OFF ON</p>	SK_NC_Program_run_AUTOSTART_OFF
 <p>ON OFF</p>	SK_NC_Program_run_SLASHED_BLOCKS_NOT_ACTIVE
 <p>ON OFF</p>	SK_NC_Program_run_SLASHED_BLOCKS_ACTIVE
 <p>GLOBAL SETTINGS</p>	SK_NC_Program_run_GLOBAL_SETTINGS
 <p>AFC SETTINGS</p>	SK_NC_Program_run_AFC_ENTER_TABLES
 <p>TABLE EVALU- ATION</p>	SK_NC_Program_run_AFC_RESULTS
 <p>TABLE SETTINGS</p>	SK_NC_Program_run_AFC_SETTINGS
 <p>AFC OFF ON</p>	SK_NC_Program_run_AFC_ON
 <p>AFC OFF ON</p>	SK_NC_Program_run_AFC_OFF
 <p>SET STANDARD VALUES</p>	SK_NC_Program_run_GLOBAL_SETTINGS_SET_STANDARD_VALUES

Soft key	Soft-key designator
GLOBAL SETTINGS INACTIVE	SK_NC_Program_run_GLOBAL_SETTINGS_SET_INACTIVE
CANCEL CHANGE	SK_NC_Program_run_GLOBAL_SETTINGS_UNDO
INTERNAL STOP	SK_NC_Program_run_INTERNAL_STOP
RESTORE POSITION	SK_NC_Program_run_RESTORE_POSITION
MANUAL TRAVERSE	SK_NC_Program_run_MANUAL_TRAVERSE

Call:

PS B/W/D/K <Mode>

0 = Simulate the soft key using the soft-key designator

PS B/W/D/K/S<Soft-key designator>

PS B/W/D/K <Attribute>

Bit 0:

0 = Soft-key is only simulated if it is in the main row

1 = Soft key is simulated regardless of whether it is in the main or subordinate row

CM 9188

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	See error code
W1022	2	Invalid value for mode
	11	String as a soft-key designator is invalid
	20	Call was not in a submit or spawn job
	45	Internal operating system error
	61	Soft key is neither available in the main soft key row nor in the following soft-key rows
	62	Programmed soft-key name does not exist

Codes for keystroke simulation

Code	Key	Group
\$00	No key	
\$08	BACKSPACE	ASCII
\$09	TAB	ASCII
\$0A	RET	ASCII
\$09	TAB	ASCII
\$1B	ESC	ASCII
\$20	SPACE	ASCII
\$21	!	ASCII
\$22	"	ASCII
\$23	#	ASCII
\$24	\$	ASCII
\$25	%	ASCII
\$26	&	ASCII
\$28	(ASCII
\$29)	ASCII
\$2A	*	ASCII
\$2B	+	ASCII
\$2C	,	ASCII
\$2D	-	ASCII
\$2E	. (ASCII DOT)	ASCII
\$2F	/	ASCII
\$30	0	Numbers
\$31	1	Numbers
\$32	2	Numbers
\$33	3	Numbers
\$34	4	Numbers
\$35	5	Numbers
\$36	6	Numbers
\$37	7	Numbers
\$38	8	Numbers
\$39	9	Numbers
\$3A	:	ASCII
\$3B	;	ASCII
\$3C	<	ASCII
\$3D	=	ASCII
\$3E	>	ASCII
\$3F	?	ASCII
\$41	A	ASCII
\$42	B	ASCII
\$43	C	ASCII
\$44	D	ASCII



Code	Key	Group
\$45	E	ASCII
\$46	F	ASCII
\$47	G	ASCII
\$48	H	ASCII
\$49	I	ASCII
\$4A	J	ASCII
\$4B	K	ASCII
\$4C	L	ASCII
\$4D	M	ASCII
\$4E	N	ASCII
\$4F	O	ASCII
\$50	P	ASCII
\$51	Q	ASCII
\$52	R	ASCII
\$53	S	ASCII
\$54	T	ASCII
\$55	U	ASCII
\$56	V	ASCII
\$57	W	ASCII
\$58	X	ASCII
\$59	Y	ASCII
\$5A	Z	ASCII
\$5E	^	ASCII
\$150	Soft key 0	Vertical soft key left
\$151	Soft key 1	Vertical soft key left
\$152	Soft key 2	Vertical soft key left
\$153	Soft key 3	Vertical soft key left
\$154	Soft key 4	Vertical soft key left
\$155	Soft key 5	Vertical soft key left
\$156	Soft key 6	Vertical soft key left
\$157	Soft key 7	Vertical soft key left
\$158	Soft key 8	Vertical soft key left
\$159	Soft key 9	Vertical soft key left
\$160	Soft key 0	Vertical soft key right
\$161	Soft key 1	Vertical soft key right
\$162	Soft key 2	Vertical soft key right
\$163	Soft key 3	Vertical soft key right
\$164	Soft key 4	Vertical soft key right
\$165	Soft key 5	Vertical soft key right
\$166	Soft key 6	Vertical soft key right
\$167	Soft key 7	Vertical soft key right
\$168	Soft key 8	Vertical soft key right

Code	Key	Group
\$169	Soft key 9	Vertical soft key right
\$17D	FNEXT (vertical)	Vertical soft key right
\$180	Soft key 0	Horizontal soft key
\$181	Soft key 1	Horizontal soft key
\$182	Soft key 2	Horizontal soft key
\$183	Soft key 3	Horizontal soft key
\$184	Soft key 4	Horizontal soft key
\$185	Soft key 5	Horizontal soft key
\$186	Soft key 6	Horizontal soft key
\$187	Soft key 7	Horizontal soft key
\$19C	FBACK	Horizontal soft key
\$19D	FNEXT (horizontal)	Horizontal soft key
\$19E	FNEXT-UP	Horizontal soft key
\$1A0	C-UP	Cursor
\$1A1	C-DOWN	Cursor
\$1A2	C-LEFT	Cursor
\$1A3	C-RIGHT	Cursor
\$1A8	ENTER	Cursor
\$1A9	NO-ENTER	Cursor
\$1AB	DEL	Cursor
\$1AC	END BLOCK	Cursor
\$1AD	GOTO	Cursor
\$1AE	CE	
\$1B0	X	
\$1B1	Y	
\$1B2	Z	
\$1B3	IV	
\$1B4	V	
\$1B8	POLAR	
\$1B9	INCREMENT	
\$1BA	Q	
\$1BB	ACTPOS	Numbers
\$1BC	-	Numbers
\$1BD	.	Numbers
\$1C0	MANUAL	Operating mode
\$1C1	TEACH-IN	Mode of operation
\$1C2	SINGLE	Mode of operation
\$1C3	AUTO	Mode of operation
\$1C4	EDIT	Mode of operation
\$1C5	HANDWHEEL	Mode of operation
\$1C6	TEST	Mode of operation
\$1C7	MOD	

Code	Key	Group
\$1CB	PGM MGT	
\$1D0	PGM-CALL	Block opening
\$1D1	TOOL DEF	Block opening
\$1D2	TOOL CALL	Block opening
\$1D3	CYCL DEF	Block opening
\$1D4	CYCL CALL	Block opening
\$1D5	LBL SET	Block opening
\$1D6	LBL CALL	Block opening
\$1D7	L	Block opening
\$1D8	C	Block opening
\$1D9	CR	Block opening
\$1DA	CT	Block opening
\$1DB	CC	Block opening
\$1DC	RND	Block opening
\$1DD	CHF	Block opening
\$1DE	FK	Block opening
\$1DF	TOUCH-PROBE	Block opening
\$1E0	STOP	Block opening
\$1E1	APPR/DEP	Block opening
\$1EA	DIA	
\$1EB	FIG	
\$1EC	Screen switch-over	
\$1ED	HELP	
\$1EE	INFO	
\$1EF	CALC	
\$1F0	NC START	

8.4.2 Machine operating panel

The machine operating panel has its own PLC inputs and outputs. These are transmitted via HSCI. You must configure the inputs and outputs using IOconfig, and evaluate them in the PLC.

You can activate specific functions by linking the PLC inputs with the corresponding markers and words.

You can store the pressing of an axis-direction button:

- ▶ With MP7680 bit 0, enable the memory function.
- ▶ Use M4562 to save a depressed axis direction key. This means that the axis will move until there is an NC STOP.

If the LSV2 connection is active, the NC Start and NC Stop commands can be transmitted. PLC Marker M4230 functions like NC Start, and M4231 like NC Stop, unless the PLC markers are reset by the PLC program. The two markers for NC Start (M4564, M4230) are OR-gated in the PLC run-time system. The two markers for NC Stop (M4560, M4231) are AND-gated (if one marker = 0, then an NC Stop is triggered). After evaluating the states, the two markers of the LSV2 connection are set to their original state again (M4230=0, M4231=1).

MP7680 Machine parameter with multiple function

Input: Bit 0 – Memory function for axis-direction keys with M4562:
 0: Not saved
 1: Saved if M4562 is set

	Set	Reset
W1046/1144 Manual traverse in positive direction	PLC	PLC
Bits 0 to 17 represent axes 1 to 18:		
0: Do not move axis		
1: Move axis		

	Set	Reset
W1048/1148 Manual traverse in negative direction	PLC	PLC
Bits 0 to 17 represent axes 1 to 18:		
0: Do not move axis		
1: Move axis		

		Set	Reset
M4230	NC start via LSV2	NC	NC
M4231	NC stop via LSV2	NC	NC
M4560	NC stop (0: stop)	PLC	PLC
M4561	Rapid traverse	PLC	PLC
M4562	Memory function for axis direction keys (MP7680 bit 0 = 1)	PLC	PLC
M4564	NC start	PLC	PLC



8.4.3 Touchpad on USB port

In order to prevent accidental entries via the touchpad, you can lock it via the PLC with Module 9185.

This module is used to disable or enable entries via screen pointing devices that are attached to the USB port. The module does this by deactivating the mouse pointer on the screen. The status of the locking can also be interrogated.

Module 9185 Touchpad status

Call:

PS B/W/D/K <Mode>
0: Activate mouse pointer (UNLOCK)
1: Deactivate mouse pointer (LOCK)
2: Status request

CM 9185
PL B/W/D <Status>
-1: Status not defined
0: Mouse pointer active
1: Mouse pointer inactive

Error code:

Marker	Value	Meaning
M4203	0	Function was performed
	1	Error code in W1022
W1022	2	Invalid mode

8.5 Files

The iTNC enables you to edit various file types. File types are identified by an extension after the file name.

Disable soft keys for file types

With the **SELECT TYPE** soft key you can display a soft key for each file type:

- ▶ Select MP7224.0 to disable soft keys of specific file types.

Disabling file types for editing

Protected files cannot be edited or changed. They are displayed in the file overview with the color defined in MP7354.1 or MP7355.1, and if the EDIT ON OFF soft key is pressed, the **Protected file!** error message appears:

- ▶ Choose with MP7224.1 the file type that you want to protect.
- ▶ Choose with MP7224.2 the file type whose **EDIT ON OFF** soft key is to be disabled.

MP7224

Lock specific file types

Input:	Bit 0 – HEIDENHAIN programs *.H
	Bit 1 – ISO programs *.I
	Bit 2 – Tool tables *.T
	Bit 3 – Datum tables *.D
	Bit 4 – Pallet tables *.P
	Bit 5 – Text files *.A
	Bit 6 – Reserved
	Bit 7 – Point tables *.PNT
	0: Do not disable
	1: Disable
MP7224.0	Disable soft keys for file types
MP7224.1	Protect file types
MP7224.2	Disable the EDIT ON/OFF soft key

You can also set access levels with PLC Module 9285 to protect file types from being edited, see page 1168.



Selecting a file

If you are in the **Program Run, Single Block** or **Program Run, Full Sequence** operating modes, you can select a file via the PLC. W1018 returns the number of files opened by the PLC. W1020 returns the number of all open files.

► With Module 9290, transfer the name of the file to be selected.

Module 9290 Select a file

You can select a file in the **Program Run, Single Block** or **Program Run, Full Sequence** operating modes.

Call:

PS B/W/D/K/S<String number or file>

CM 9290

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Invalid string was transferred
	8	Control is not in the Program Run, Single Block or Program Run, Full Sequence operating mode
	20	Module was not called in a spawn or submit job
	29	Selected file is invalid or does not exist

Copying, renaming or deleting a file

Module 9248 is used to copy, rename and delete files. This module can be used to access all mounted drives.

Module 9248 Copy, rename and delete files

Any necessary file conversions (ASCII → binary, binary → ASCII) are performed automatically during the copy process.

Conditions:

- The file names must contain drive information (e.g. PLC:) and file types.
- The file types are used to determine whether a conversion is necessary.
- If a conversion must be performed, then the file types must be identical.
- The file types are used to determine whether renaming is permissible. If the file type remains the same during renaming, then there are no limitations. If the file type is changed, then renaming is only permitted between certain file types.
- Dependencies with other files are not considered when deleting files.

Call:

PS B/W/D/K/S <Name of the source file>

PS B/W/D/K/S<Name of the target file>

Only with mode 0 or 1

PS B/W/D/K <Mode>

0: Copy

1: Rename

2: Delete

CM 9248

Error code:

Marker	Value	Meaning
M4203	0	Successful execution of module
	1	Error code in W1022
W1022	2	Module was called in an invalid mode setting
	3	Invalid source string or target string
	7	Error during file conversion, invalid source or target string, error during copying without file conversion, or source file does not exist
	20	Module was not called in a spawn or submit job
	29	Copying or renaming not allowed for the current file types
36	Error during renaming or deletion. More information about the error can be requested with Module 9149, immediately after Module 9248 has been called. Or an error occurred during a copying process without conversion.	

		Set	Reset
W1018	Number of files opened by the PLC	NC	NC
W1020	Number of all open files	NC	NC



8.5.1 Datum tables (*.D)

You can define up to 255 different datums in a datum table:

- ▶ In MP7226.1, define the size of the table.

With Cycle 7 **DATUM SHIFT**, you can enter the new datum with absolute coordinates or specify a line number from the datum table (see the User's Manual).

With Modules 9092 to 9094 you can use the PLC to read from and write to the current datum table (see "Tool and pocket table" on page 1565).

With **FN17: SYSWRITE** and **FN18: SYSREAD** you can read and overwrite values in the datum table (OEM cycles). See pages 1725 and 1737.

MP7226.1 **Size of the datum table**

Input: 0 to 255 [lines]

Reference for values in the datum table

The values from the datum table can be interpreted with respect to the workpiece datum or to the machine datum (MP960.x):

- ▶ Enter the datum in MP7475.

MP7475 **Reference for datum table**

Input: 0: Reference is workpiece datum
 1: Reference is machine datum (MP960.x)

8.5.2 Freely definable tables

You can adapt tables to suit your own applications:

- ▶ Define the number and names of the fields as prototypes.

You can interrogate and edit the entries through PLC modules or through the FN functions **FN26: TABOPEN**, **FN27: TABWRITE** and **FN28: TABREAD** (see the User's Manual).

With the exception of pallet tables and cutting data tables, freely definable tables are given the file name extension .TAB.

Creating a prototype

- ▶ Switch to PLC mode.
(See "Selecting the PLC mode" on page 9 – 1667).
- ▶ In the **PLC:\PROTO** directory, create a table with the extension **.TAB**.
If you have not yet defined a prototype, a standard prototype will be offered.
If you have defined more than one prototype, a menu will appear when you create a table:

- ▶ Select an existing prototype and change the format by using the soft key **EDIT FORMAT**.

If you have selected a prototype, the structure commands of the individual columns are displayed:

- **NAME:** Heading of the column. Maximum 8 characters, no longer than **WIDTH**. Do not use any blanks.
- **TYPE:**
 - **N** = Numerical input (with "\$" in hexadecimal and "%" in binary format)
 - **C** = Alphanumeric input
 - **X** = Date-stamp function; if the line in the table is changed, then depending on the length defined in **WIDTH**, the current time (**WIDTH=8**) or the current time and date (**WIDTH=19**) can be entered in the field.
hh:mm:ss dd.mm.yyyy is the valid format here.
The timestamp function is active in combination with write-accesses by the PLC (via module) and from NC programs (FN function TABWRITE, WRITE TO KINEMATIC).
 - **L** = Entry of an integer value (no decimal places); special type for quicker access for machining, such as for position requests.
- **WIDTH:** Width of the column. For **TYPE = N** it includes the algebraic sign, decimal point and decimal places.
- **DEC:** Number of decimal places; = 0 for hexadecimal or binary format input. Has no meaning for **TYPE = C**.
- **ENGLISH to RUSSIAN:** Language-specific messages that are shown in the dialog line during editing of the column. Maximum 32 characters per language. Dialog entry is optional.
- ▶ Press the "Insert line" soft key and enter your structure commands in the respective column.
- ▶ With the END key you exit the display of the structure definition. The table you have just created is displayed with the newly defined columns.



Note

A table can have a maximum of 30 columns and a maximum width of 500 characters.

It is also possible to show dialogs with the UTF8 character set in freely definable tables (*.TAB, *.P). This makes it possible to use all conversational languages available on the control.

Data transfer

Valid for tables with the file name extensions .TAB, .P and .CDT:

If a freely definable table is transferred through a data interface, in the externally saved file the structure definition is saved between the lines **#STRUCTBEGIN** and **#STRUCTEND**. The contents of the table are after the line **#STRUCTEND**.

Reading and editing table fields in the PLC

You can read and overwrite table fields in the PLC by using modules. You can give the user access to parts of tables for editing.



Note

The following modules must be called in a submit job or spawn job.

When entering the column names, pay attention to the case of the letters (whether they are small or capital).

Module 9245 Read a field from a table

This module is used to read data from any freely definable table. Module 9245 and the **tableread** function (PLC screen masks) can be used to read data, including strings, from system tables like the tool table and pocket table (*.T and *.TCH).

Open the desired table with the file name extension *.TAB, *.P, *.T or *.TCH with Module 9240, and not in the buffered mode. If an error occurs, the result is undefined.

The module provides the contents as a string.

Call:

```
PS   D           <File handle>
      from Module 9240
PS   B/W/D/K     <Line>
      0 to 65 535
PS   B/W/D/K/S   <String number, column name>
PS   B/W/D/K/S   <String number for the result>
CM   9245
```

Error code:

Marker	Value	Meaning
M4203	0	Field was read
	1	Error code in W1022
W1022	1	Line does not exist in table
	2	Incorrect "file handle" or table was opened in "buffered" mode
	3	Impermissible string numbers
	7	The table could not be read from
	20	Module was not called in a spawn job or submit job
	29	The opened file is not a valid table
	30	Field name does not exist in table

Module 9255 Read a field from a table as an integer value

Open the table with the file name extension .TAB or .P with Module 9240, and not in the buffered mode. If an error occurs, the result is undefined.

The module provides the contents as an integer value.

Call:

PS D <File handle>
from Module 9240

PS B/W/D/K <Line>
0 to 65 535

PS B/W/D/K/S<String number, column name>

CM 9255

PL B/W/D <Result>

Error code:

Marker	Value	Meaning
M4203	0	Field was read
	1	Error code in W1022
W1022		See Module 9245



Module 9246 Write to a field in a table

Open the table with the file name extension .TAB or .P with Module 9240, and not in the buffered mode.

The field defined by the column name and line number is overwritten.

If a line that does not yet exist is transferred, the file is filled with blank spaces up to the defined line.

The module transfers a string.

Call:

```
PS   D           <File handle>
      from Module 9240
PS   B/W/D/K     <Line>
      -1: Next vacant line
      0 to 65 535
PS   B/W/D/K/S   <String number, column name>
PS   B/W/D/K/S   <String number, contents to be written>
CM   9246
```

Error code:

Marker	Value	Meaning
M4203	0	Field was written to
	1	Error code in W1022
W1022	1	Line does not exist in table
	2	Incorrect "file handle" or table was opened in "buffered" mode
	3	Impermissible string numbers
	6	Table is write-protected
	7	Not a numerical field (Module 9256)
	11	The transferred value cannot be saved to the addressed field. Incorrect format
	20	NCMACRO.SYS does not exist
	29	The opened file is not a table with the extension .TAB or .P
	30	Column name not found

Module 9256 Write an integer value to a field in a table

Open the table with the file name extension .TAB or .P with Module 9240, and not in the "buffered" mode.

The field defined by the column name and line number is overwritten.

This module can be used only for an integer. Values with decimal places are written without the decimal point.

If a line that does not yet exist is transferred, the file is filled with blank spaces up to the defined line.

Call:

```
PS    D          <File handle>
        from Module 9240
PS    B/W/D/K   <Line>
        -1: Next vacant line
        0 to 65 535
PS    B/W/D/K/S<String number, column name>
PS    B/W/D/K   <Numerical value to be written>
CM    9256
```

Error code:

Marker	Value	Meaning
M4203	0	Field was written to
	1	Error code in W1022
W1022		See Module 9246

Module 9247 Search for a condition in a table

Open the table with Module 9240 not in the "buffered" mode.

The module searches for a field content that fulfills one or more conditions.

The conditions are formulated with the commands of the System Query Language (SQL) data bank language. Pay attention to the case of the letters (whether they are small or capital) in the commands and column names.

If you indicate a starting line, the module can search for several suitable field entries.

Permissible SQL commands:

Command	Meaning
+, -, *, /	Arithmetic operators
NOT, AND, OR	Logical operators
<, >, <=, >=, ==, <>	Comparisons
LIKE 'abc'	Text comparison
LIKE '_abc%'	Partial string
()	Parentheses
MIN(column name)	Minimal value from the column
MAX(column name)	Maximum value from the column



Example:

Search in a pallet table for the line with the NC program 1.H and the set datum X=-10.

String contents:

```
WHERE (PAL/PGM LIKE 'PGM') AND (NAME LIKE '1.H') AND (X== -10)
```

Call:

```
PS D <File handle>  
from Module 9240
```

```
PS B/W/D/K <Starting line>  
0 to 65 535
```

```
PS B/W/D/K/S <String number of condition or string with condition>  
CM 9247
```

```
PL B/W/D <Line that fulfills the condition>  
-1: Error code in W1022
```

Error code:

Marker	Value	Meaning
W1022	1	Start line does not exist in table
	2	Incorrect "file handle" or table was opened in "buffered" mode
	3	Impermissible string numbers
	7	Module could not be read from the table
	20	Module was not called in a spawn job or submit job
	29	Incorrect file format
	30	Column name not found
	31	Syntax error in the transferred condition
	32	No data record found that fulfills the condition



Starting the PLC editor for tables

In the machining modes a table editor can be started:

- ▶ Specify the lines and columns that are to be displayed.

You can provide the PLC editor only with tables with the file extensions .TAB or .P. A temporary file with the name **SYS:\TEMP\PLCTABED.TAB** is saved.

With Modules 9240, 9241, 9245 and 9247 you can check this temporary file before you place the edited data into the original table with Module 9251.

- ▶ Enter the editable columns in the sequence in which they are to be displayed.
Do not enter the line number! It is displayed automatically.
- ▶ Separate the individual columns by a space character. If you have transferred an empty string, all columns of the original table are displayed.
- ▶ Enter the first and last line to be displayed on the screen. Line numbering begins with zero. If you enter -1 as the last line, the table will be shown to its end. If you release all lines and columns for editing, you can choose:
 - Whether lines can be deleted and inserted
 - Whether the original table should be edited directly

If you edit the original table directly, you cannot cancel the changes with Module 9251.

If you do not edit the original table directly and the PLC program is recompiled while the PLC editor is open, the editor will be closed without transferring the changes to the original table.

If the END key or the END soft key is pressed while the PLC editor is opened, the NC sets M4159. The PLC editor is **not** closed by the NC. It must be closed by the PLC with Module 9251.

M4159 is reset when Module 9250 is called.

With Module 9035 you can request the active line in the PLC editor.



Module 9250 Start the PLC editor for tables

With this module you start a table editor in the machining modes (compare "Tool Tables").

Call only in a submit job or spawn job.

Call:

```
PS   B/W/D/K/S<String with table name>
      Complete path and name
PS   B/W/D/K/S<String with columns to be edited>
PS   B/W/D/K <First line >
      [0 to 65 535]
PS   B/W/D/K <Last line>
      [0 to 65 535]
PS   B/W/D/K <Mode>
      Bit 0=1: Lines can be inserted and deleted
              (if all lines and columns are selected)
      Bit 1=1: Edit in the original file
              (if all lines and columns are selected)
      Bit 2=0: Shown as a table
      Bit 2=1: Shown as a form
      Bit 3=1: Write-protection of the file to be opened is
              respected
```

CM 9250

Error code:

Marker	Value	Meaning
M4203	0	Editor was opened
	1	Error code in W1022
W1022	1	First and last line do not define a meaningful range, or incorrect value for mode was transferred
	3	Impermissible string number
	6	Write-protected file cannot be opened if write-protection is active (bit 3 = 1)
	7	The module could not read from the table or open the temporary file
	8	The module was called from an operating mode in which the table editor cannot be started.
	20	Module was not called in a spawn job or submit job
	28	PLC editor already open for another table
	29	The opened file is not a table (extension .TAB or .P)
	30	Column name not found
	36	Invalid file name or file type programmed

M4159	PLC editor: END key or soft key pressed	Set NC	Reset NC/PLC
--------------	--	------------------	------------------------

Module 9251 End the PLC editor for tables

With this module you end the PLC editor and specify whether the changes are to be put into the original table. The changed values are not checked automatically. Before calling Module 9251 you can read and check the temporary file in the PLC.

Call:

PS B/W/D/K <Mode>
0: Do not place changes into the original file
1: Place changes into the original file

CM 9251

Error code:

Marker	Value	Meaning
M4203	0	Editor was opened
	1	Error code in W1022
W1022	3	Incorrect value was transferred for mode
	6	Changes could not be saved in the original table
	20	Module was not called in a spawn job or submit job
	28	PLC editor had not been opened with Module 9250

Module 9252 Position the cursor in the PLC editor

With this module you place the cursor of the PLC editor on a specified line and in a specified column. The line is defined relative to the starting line of Module 9250. The designated column must be defined in Module 9250.

Call:

PS B/W/D/K/S<String with column name>
PS B/W/D/K <Line>
CM 9252

Error code:

Marker	Value	Meaning
M4203	0	Cursor was set
	1	Error code in W1022
W1022	1	Incorrect line number
	3	Incorrect string number
	20	Module was not called in a spawn job or submit job
	30	Incorrect column name
	35	PLC editor is not open (Module 9250)



8.5.3 PLC files

With modules you can create PLC files and read or write in them line-by-line. PLC files are in ASCII format and are used, for example, for saving data specific to the PLC.



Note

The following modules must be called only in a submit job or spawn job.

Module 9240 Open a file

You can open up to eight files simultaneously. They are accessed from the process in which they were opened (submit job or spawn job).

If you want to prevent the file from being opened by more than one process, use the "lock file" mode.

To ensure adequate speed, use the "buffered" mode to read from and write to ASCII files. In this mode a part of the file is buffered in the main memory. This mode is not permitted for tables.

When the process is ended (EM in the submit job or spawn job), all files opened for this process are closed.

After the file is opened, Module 9240 always transfers a "file handle." The file handle is a serial number that can be used to select this file again in other modules.

To append data to an existing file, set bit 0=1 (reading and writing) **and** bit 2=0 (record oriented).

Call:

```
PS    B/W/D/K <Mode>
      Bit 0 = 0: Read only
      Bit 0 = 1: Read and write
      Bit 1 = 0: Do not lock file
      Bit 1 = 1: Lock file
      Bit 2 = 0: Record oriented (tables)
      Bit 2 = 1: Buffered (ASCII files)
      Bit 3 is only evaluated if bit 2 = 0:
      Bit 3 = 0: Do not create file
      Bit 3 = 1: Create file if it does not exist

PS    B/W/D/K/S<String with file name>
      Complete path and file name

CM    9240
PL    D      <File handle>
      Number for use in other modules
      -1: Error code in W1022
```

Error code:

Marker	Value	Meaning
M4203	0	File was opened
	1	Error code in W1022
W1022	1	Impermissible mode
	3	Incorrect string number
	7	File could not be opened
	20	Module was not called in a spawn or submit job

Module 9241 Close a file

With this module you close a file that has been opened with Module 9240. You must close the file in the process (submit job or spawn job) in which you opened it.

Call:

PS D <File handle>
Number from Module 9240

CM 9241

Error code:

Marker	Value	Meaning
M4203	0	File was closed
	1	Error code in W1022
W1022	2	Incorrect file handle
	20	Module was not called in a spawn or submit job



Module 9242 Position in a file

With this module you change the position of the cursor in a file opened with Module 9240. The new position is provided as result from Module 9242.

If the file was opened in the "record oriented" mode (tables), the cursor is positioned line by line.

If the file was opened in the "buffered" mode, the cursor is positioned character by character.

If you indicate a position before the beginning or after the end of the file, the cursor is positioned to the beginning or end of the file, respectively. The addressing of the new position is relative to the beginning or end of the file, or to the current position. You can interrogate the current position by transferring the position value zero relative to the current position.

Call only in a submit job or spawn job.

Call:

```
PS   D           <File handle>
                        Number from Module 9240
PS   B/W/D/K     <Desired position>
PS   B/W/D/K     <Mode>
                        0: Position relative to the file beginning
                        1: Position relative to the current position
                        2: Position relative to the file end

CM   9242
PL   B/W/D/K     <New position>
                        -1: Error code in W1022
```

Error code:

Marker	Value	Meaning
M4203	0	Cursor was positioned
	1	Error code in W1022
W1022	1	Impermissible mode
	2	Incorrect file handle
	7	File system error
	20	Module was not called in a spawn job or submit job



Module 9243 Read from a file line by line

To read from a table, use Module 9245.

Open the file with Module 9240.

With Module 9243 you read line-by-line from an ASCII file.

The "buffered" mode provides faster access times. The result is saved in a string. The module reads up to the line break (LF); at most 126 characters.

Call:

```
PS   D           <File handle>
      Number from Module 9240
PS   B/W/D/K     <String number for the result>
CM   9243
PL   B/W/D       <Number of read bytes>
      >0: Line has been read
      0: File end has been reached
      -1: Error code in W1022
```

Error code:

Marker	Value	Meaning
M4203	0	Line was read
	1	Error code in W1022
W1022	2	Incorrect file handle
	3	Incorrect string number
	7	File system error
	20	Module was not called in a spawn job or submit job



Module 9244 Write to a file line by line

To write to a table, use Module 9246.

With Module 9244 you write line-by-line to an ASCII file.

Open the file with Module 9240.

If file is opened in "buffered" mode:

- Processing time is shorter.
- Files are saved to the hard disk only if more than 512 bytes are overwritten in several calls, or if the file is closed.
- The number of data specified in the transfer string is overwritten.

If file is opened in "record oriented" mode:

- Processing time is longer.
- The data is immediately saved to the hard disk.
- Exactly one line is overwritten. If there is a difference in length, the subsequent data is displaced by the difference.

Call:

```
PS   D           <File handle>
      Number from Module 9240
PS   B/W/D/K/S<String number, source data>
CM   9244
PL   B/W/D       <Number of written bytes (including LF)>
      -1: Error code in W1022
```

Error code:

Marker	Value	Meaning
M4203	0	Line was written
	1	Error code in W1022
W1022	2	Incorrect file handle
	3	Incorrect string number
	7	File system error
	20	Module was not called in a spawn job or submit job

8.6 DCM – Dynamic Collision Monitoring

8.6.1 DCM – monitoring the working space for collisions

General information

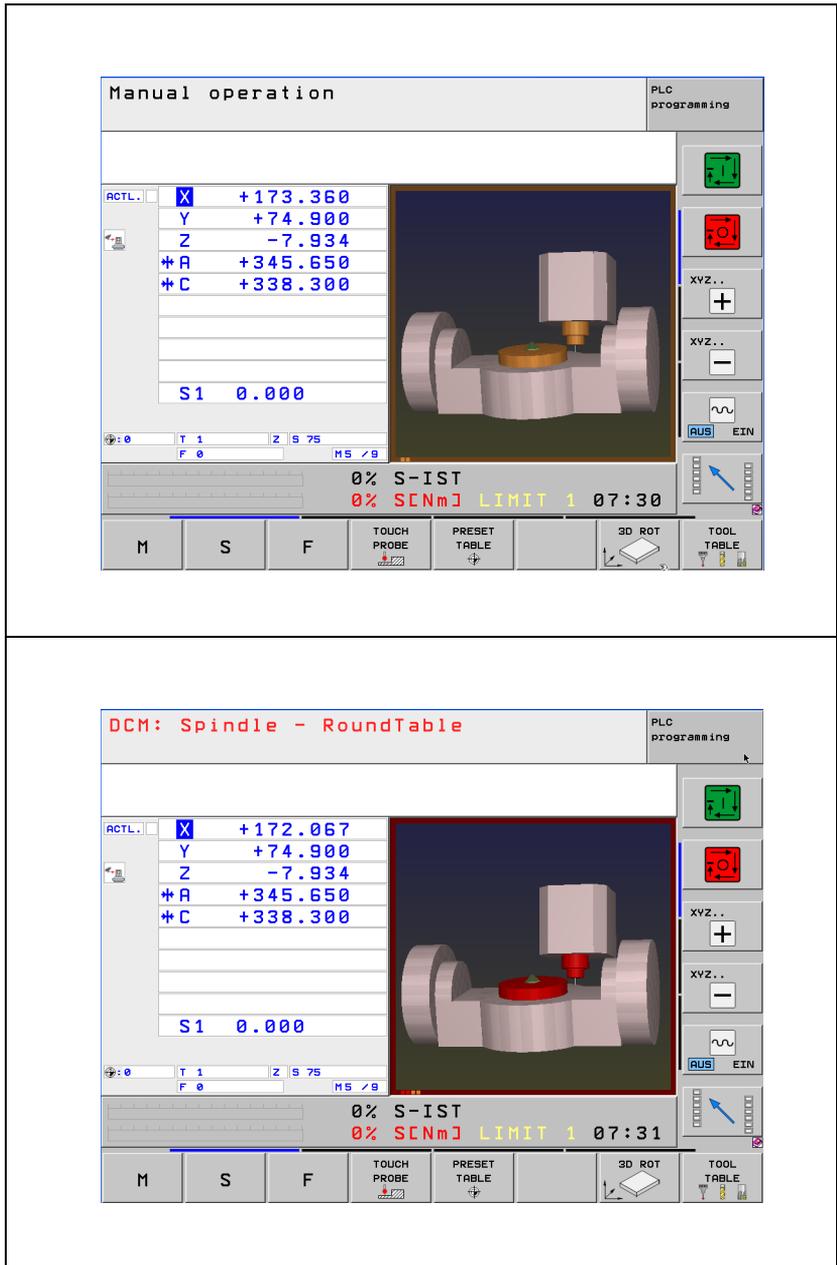
With the iTNC 530 HSCI, the "DCM – Dynamic Collision Monitoring" function (option 40, ID 526 452-01) can be used to integrate collision monitoring for various machine elements via the kinematics tables.

You use **DCM** to define objects (CMOs – "Collision Monitored Objects") within the kinematics description in relation to movable machine axes (X, Y, Z, A, B, C), as they appear on the machine as spatial objects or machine elements. These objects give a three-dimensional image of the machine.

These CMOs, including the **current tool**, an assigned, optional **tool-carrier kinematics description** and an optional **fixture** whose position and orientation have been ascertained, are then taken into account along with the motions of the machine. The active tool is automatically integrated at the tool reference point as a cylindrical object with the dimensions from the tool table. If there is a danger of collision between defined objects, or between the tool and defined objects, then the motions of the machine are stopped, the collision is avoided, and a collision warning or error message is output in the header of the iTNC530 screen. The machine's axes can then only be moved out of the danger zone by acknowledging the collision monitoring or by switching it off.

The Screen Layout key and the **PROGRAM + KINEMATIC** soft key enable you to display 3-D graphics of the collision objects defined in your machine in the Program Run operating modes. The viewer of the KinematicsDesign tool is used for this. However, the collision objects of the machine are only displayed and cannot be edited. Software option 40 is required for this feature.

The following figures illustrate the early warning and message regarding a possible collision:



Function and operation

After the software option has been enabled, collision monitoring is permanently active in the **Manual, Positioning with Manual Input, Program Run Single Block, Program Run Full Sequence, smarT.NC** and **E1. Handwheel** operating modes. You can switch collision monitoring off in the **Manual** operating mode in the "Collision Monitoring (DCM)" menu that appears when the **Collision** soft key is pressed. It is possible to activate/deactivate **DCM** separately here for the manual and the program-run modes of operation.

At the same time, individual objects can be deactivated at run-time (see "Activating and deactivating monitoring" on page 1439), or generally be excluded relative to other objects via the description table (see "Collision exclusions" on page 1436).

In the **Manual** and **E1. Handwheel** operating modes, the iTNC reduces the speed of the traverse motions when objects monitored for collision come into proximity, in order to avoid a collision. This can result in sudden reductions or increases of the feed rate to the maximum permitted speed in the respective manual operating modes. If the objects come within a specified distance of each other, the control stops the machine's motions and outputs an error message. The control makes a distinction between two different zones, which are indicated visually.

- Early warning
 - No message, axes can still be moved
 - Brown frame around the graphical depiction of the kinematics; the affected objects monitored for collision are colored brown
 - Cause: The iTNC reduces the traversing speed in order to safely prevent a collision
 - Solution: Remove the cause of the danger, or retract the axes from the danger zone.
- Error
 - Message example: **DCM: Spindle - RoundTable**
 - Meaning: Two objects monitored for collision are within approx. 4 mm of each other.
 - Cause: Distance is less than 4 mm [between 0 and 4 mm]
 - Solution: The axis can be moved again if the motion increases the distance between the collision objects (for example by pressing the axis-direction key for the opposite direction): First release the axis-direction key or stop the handwheel, and then press the appropriate axis-direction key or turn the handwheel in the appropriate direction. Acknowledging the error message with the **CE** key is then not necessary.



Machine parameter MP1294 can be set to attain higher traversing speeds in the **Manual Operation** and **Electronic Handwheel** operating modes when DCM is active. If you set MP1294 = 1, you can only move one axis in the **Manual Operation** and **El. Handwheel** operating modes at any one time. This simplifies the calculation of possible collisions and the axis can be moved at a higher feed rate.

MP1294 Increased traversing speed for DCM by moving only one axis

Input: 0: Function inactive
 1: Function active

If you want to execute a specific motion that reduces the distance between the two collision objects, you must deactivate DCM collision monitoring for the manual operating mode.

- ▶ Press the **Collision** soft key.
- ▶ In the "Collision Monitoring (DCM)" menu, set it to **Inactive** for manual operation.
- ▶ Press the **End** soft key.
- ▶ Move the axes manually (axis keys) without collision monitoring.

When finished, reactivate collision monitoring.

- ▶ Press the **Collision** soft key.
- ▶ Set collision monitoring in the "Collision Monitoring (DCM)" menu to **Active** again.
- ▶ Press the **End** soft key.

In the **Positioning with Manual Data Input, Program Run Single Block, smart.NC** and **Program Run Full Sequence** operating modes, the motion is considered block-by-block. For example, in the **Program Run, Full Sequence** mode, the NC program is stopped in the block in which the minimum safe distance calculated by the iTNC would be exceeded. This safe distance is calculated by the iTNC 530 dynamically. It results from the requirement that the traverse motion can be stopped in time at any time, taking the feed rate and direction into account. As soon as a traverse motion would result in a distance 5 mm less than the calculated safe distance, the motion is stopped or not performed before the traverse block is executed, and an error message is output.

In an NC program with many small traverse blocks and high traverse speeds, this would mean, for example: If two machine components are on a collision course, then after **DCM** has braked and stopped the motion, the components would be between 0 mm and 5 mm from each other.



Attention

Collision monitoring is not active on a machine during reference run (incremental encoders). On machines with absolute path encoders, **DCM** is active as soon as the machine is switched on.



Attention

The following limitations and conditions must also be considered for reliable operation and correct application of DCM:

- **DCM** helps to minimize the danger of collision. However, certain constellations cannot be considered during operation. Please pay attention to the following information.
- **DCM** can only protect machine components from collision if they have been dimensioned correctly, and if their positions in the machine coordinate system have been entered correctly.
- Once **DCM** has been configured, all possible collisions must be tested in the Manual operating mode in order to detect faulty configurations.
- Collisions of defined machine components or the tool with the workpiece are not detected.
- Please note that collision monitoring does not take any filters into account when calculating the path (e.g. nominal position value filters or angle tolerances for rotary axes in Cycle 32). The path deviation resulting from the nominal position value filters can be ignored. However, the machine manufacturer must calculate and define in MP1292 the deviations of the collision objects (CMOs) from the nominal path resulting from the angle tolerances for rotary axes (Cycle 32).
- **DCM** does not work in combination with the "Free Rotation" PLC function (page 8 – 1152), since this special function for the defined "free" rotary axis runs asynchronously to the system. For the work envelope this means that all CMOs moved with the "free" rotary axis cannot be monitored by **DCM**.
- In the cycles for "Tapping without Floating Tap Holder," you must note that **DCM** does not take into consideration the simple "tracking of the tool axis." In order to use **DCM** for these cycles, bit 4 of MP7160 must be set to 1 (activates the exact interpolation of the tool axis with the spindle).
- If a machine is operated in lag mode or in semi-feedforward mode, then it is assumed that the machine manufacturer takes the greatest possible path deviation into account and increases the size of the collision objects accordingly.
- Even if there are CMOs in the kinematics tables, there is no protection against collisions without software option 40 (DCM).
- When a tool is exchanged, the new tool data will not become effective until the strobe has been acknowledged by the PLC.





Attention

Furthermore, the following must be noted when editing kinematics or associated tables. Only if this is followed are changes to the kinematics detected by the collision monitoring:

- If kinematics or an associated table (e.g. CMO descriptions) are edited in an NC program or NC macro with **FN26: TABOPEN** and **FN27: TABWRITE**, then they must be closed after the changes are made. This occurs automatically at the end of a program or if **FN26: TABOPEN** is called again. Activation of the kinematics is permitted with **FN17: SYSWRITE ID290 NR1** here, since this FN17 function automatically closes all tables before activation. If a changed table is activated by the PLC via Module 9097 before it has been closed, the changes are not detected.
- If kinematics or an associated table are opened and edited by the PLC via Modules 9240 through 9246, they must be closed by the PLC after the changes have been made. This must be done before activation of the kinematics (e.g. via Module 9097), a tool call or a change of the tool data with M4538.
- If kinematics or an associated table are edited with the table editor, the editor must be exited with the END or PGM MGT keys. If the changed kinematics are activated (e.g. with PLC Module 9097 or **FN17: SYSWRITE ID290 NR1**) before the editor has been exited, the changes will not take effect.

Compensation of the angle tolerance

So that collision monitoring can function correctly when machining with Cycle 32 and angle tolerances, the maximum occurring angle tolerance must be limited with MP1290, and an oversize for collision objects must be specified with MP1292. The oversize (in MP1292) results from the maximum permissible angle allowance that the machine manufacturer must specify via MP1290 for Cycle 32, and from the dimensions of the swiveled or rotated machine components.

MP1290 and MP1292 are only effective if option 40 (DCM), M128 or TCPM Function and Cycle 32 with angle tolerance are active.

MP1290 Maximum angle tolerance for DCM

Input: 0.0000 to 3.0000 [°]
Default: 3 [°]

(in combination with option 40)

In connection with **DCM** (Dynamic Collision Monitoring), the machine manufacturer must enter a default maximum permissible angle tolerance via MP1290 (usually 0.1°). An angle tolerance programmed with Cycle 32 is then limited to this value if collision monitoring is active, i.e. the maximum effective angle tolerance is the value from MP1290. If **DCM** is switched off (via soft key or by switching to kinematics without collision-object definitions), the value programmed in Cycle 32 is in effect again.

The angle entered in MP1290, in combination with the machine kinematics, is the basis for the allowance defined in MP1292 for collision-object calculations by the control.

MP1292 Manual oversize for DCM

Input: 0 to 1000 [mm]
Default: 0 [mm]

(in combination with option 40)

For the DCM collision monitoring you enter here the necessary oversizes for the collision objects with a separate rotary axis filter.

The following should be considered for the oversizes:

- For the rotary axes, such as a rotary table, assume the largest possible radius during rotation (usually the table radius), or for a swivel head the distance from the tool tip (longest tool) to the most distant point of the swivel head.
- Use this information to calculate the non-considered offset of the rotary axis. Do so by using the maximum angle tolerance entered in MP1290 to calculate the offset in the following manner:

Example: Swivel head

With maximum tool length: 200 mm, head length: 480 mm,

angle tolerance: MP1290: 0.1 [°]

$$s_{\text{offset}} = r \cdot \sin(\text{MP1290})$$

$$s_{\text{offset}} = 680 \text{ mm} \cdot \sin(0.1^\circ)$$

$$s_{\text{offset}} = 1.19 \text{ mm}$$

Since the oversize entered in MP1292 is added to each length of the bodies in the calculation, the oversize to be entered in MP1292 must be halved:

$$\text{MP1292} = s_{\text{offset}} / 2$$

$$\text{MP1292} = 1.19 / 2$$

$$\text{MP1292} = \mathbf{1 \text{ [mm]}}$$
 (rounded up from 0.595 mm)

Since the resolution in MP1292 must be entered as an integer value in millimeters, the values input must be rounded up.





Note

Please note that when operating with multiple rotary axes, the offsets can summate, and so you must add the oversizes together.

Example:

Oversize_{Tilting table} = 0.595 [mm]

Oversize_{Rotary table} = 0.396 [mm]

MP1292 = **1 [mm]** (rounded up from 0.991 mm)

Integration of CMOs in the description table

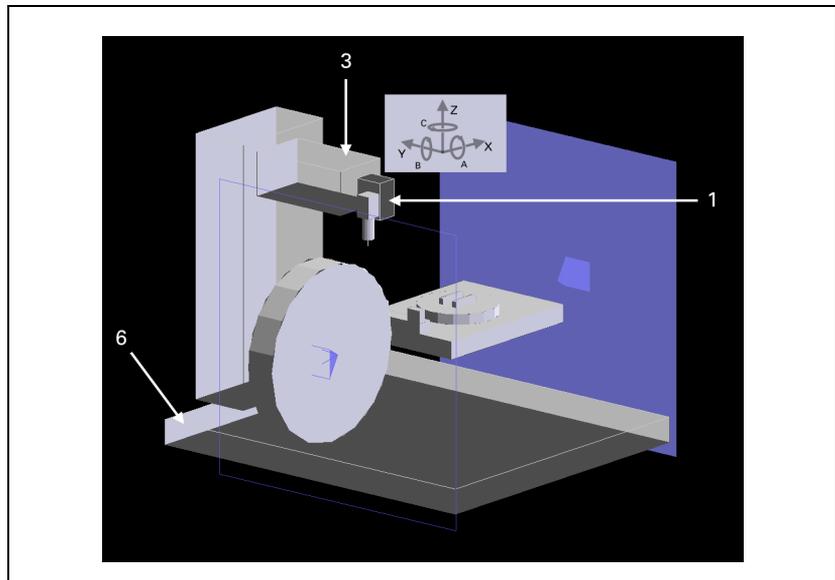
In order to use collision monitoring, the kinematics of the CMOs must be entered in the description table, see page 714.

The starting point for integrating the collision objects is always the tool reference point (e.g. spindle point) and the coordinate system (XYZ) valid for this point. It is then shifted via translations (e.g. X, Y or Z axes) and rotated or tilted via rotations (e.g. A, B or C axes). A collision object (CMO: cuboid, cylinder or plane) is inserted in the description table, taking the momentarily valid coordinate system into account.

The location at which the CMO is placed in the description table is important for the resulting machine and monitoring models. Please see the example below and the associated description table on the next page: The swivel head is defined (via **SUBFILE**) in line 1, and the B axis (**MachAxis B**) is defined in line 2. This means that when the B axis is traversed, all subsequent CMOs and of course the subsequent machine axes (including translations and transformations) will move in the B direction relative to previous CMOs.

In the lower example (machine figure), the portal (no. 3 in the figure) and all subsequent CMOs (no. 6 in the figure) rotate in the B axis direction relative to the double swivel head. The machine coordinate system shown in the figure below (illustrating the traverse directions) is valid.

In the description table, after the portal in line 3, the machine axes X and Y, followed by a machine floor (**CMO_FloorSection**) are defined. This means that when the X or Y axis is traversed, the machine floor (**CMO_FloorSection**) moves in X or Y direction relative to the previously defined portal (**CMO_Portal**). The coordinate system shown is again valid.



NR	KEY	AXIS	COORD	ON/OFF	FILE	DONTTEST
0	TOOLFILE					
1	SUBFILE1					
2	MachAxis	B				
3	CMO				CMO_Portal	
4	MachAxis	X				
5	MachAxis	Y				
6	CMO				CMO_FloorSection	
7	Trans	X	470.0L92			
8	Trans	Y	-282.405			
9	Trans	Z	-900			
10	CMO			1	CMO_Cabin	
11	MachAxis	Z				
12	CMO				CMO_LiftTable	CMO_Cabin
13	MachAxis	C				
14	CMO				CMO_TurnTable	CMO_LiftTable
	[END]					

Optimizing the DCM performance

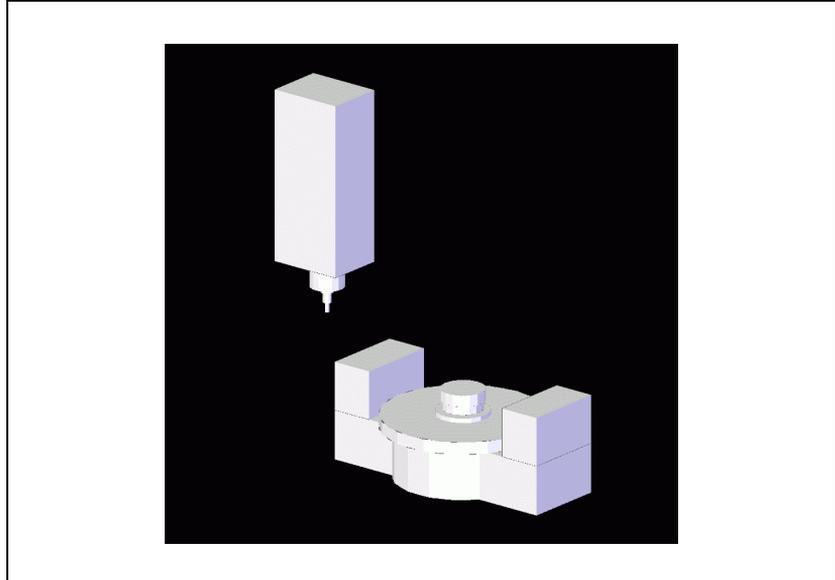
The calculation of the collision objects, as well as the monitoring of their relative positions, is very processor-intensive for the iTNC. This calculation may not consume too many resources, since they are essential to the actual functions.

In order to satisfy this demand, the number of collision objects must be defined such that no more than the permissible number of intersections need be calculated. The number of collision objects themselves must be defined in such a way that no more than the above mentioned number of intersection calculations are required. Otherwise the increased calculation efforts can cause increased block processing times. The maximum number of intersection calculations applies to the complete collision monitoring model, including the monitoring of tool-carrier kinematics and fixtures. You therefore need to save a sufficient number of intersection calculations when describing the actual machine so that the number of intersection calculations is not exceeded if you also want to monitor tool-carrier kinematics and fixtures. The number of collision objects defined (CYLINDERS and CUBOIDS in CMO files) is not definite, but rather the number of intersections calculated within the collision object model. The permissible number of intersection calculations depends on the MC computer being used:

- MC 62xx: Up to 200 intersection calculations
- MC 63xx: Up to 1000 intersection calculations

Deactivated collision objects (see "Activating and deactivating monitoring" on page 1439) and all definitions of excluded collisions (see "Collision exclusions" on page 1436) are not included in the calculations. It is therefore important that all necessary calculations of intersections be specified in advance.

Collision objects located in the same axis (CMOs between two "MachAxis..." machine-axis definitions), which due to physical realities can therefore not collide during traverse motions, are automatically **ignored** by the iTNC. This also applies if within two machine-axis definitions, collisions objects are separated by transformations (**TRANS [X, Y, Z, A, B, C]**). However, each of these objects can collide with the individual CMOs traversed in other axes.



In order to determine the number of possible intersections, proceed as follows:

- ▶ Ascertain the total number of collision objects defined (CYLINDERS and CUBOIDS) in the kinematics model with the most intersection calculations. Ascertain this number per axis (objects between two "MachAxis..." machine-axis definitions), and take into account any tool-carrier kinematics (TOOLFILE) which might be activated at run-time.
- ▶ Ascertain the number of possible intersections (collisions) by proceeding as shown in the following example (see figure above):

No.	Axis	No. of objects	Name
1	-	5	Tool, TOOLFILE and spindle with housing
2	X		
3	Y		
4	Z		
5	A		
6	-	3	Tilting table with shank
7	C		
8	-	4	Rotary table with chuck
9			

Calculation:

$$\text{No. of objects1} = 5 \cdot (3 + 4) = 35$$

$$\text{No. of objects2} = 3 \cdot 4 = 12$$

$$\text{Intersections} = \text{No. of objects1} + \text{No. of objects2} = \mathbf{47}$$

Collision exclusions

Definition of collision exclusions

In order to avoid unnecessary calculations, collisions which cannot occur due to physical realities can be excluded from each other in the description table by using the **DONTTEST** column. This is useful since it saves processing time and avoids unnecessary reductions in the feed rate in the manual operating modes.

Two different methods are available

1. Exclusions via existing definition tables of collision-monitored objects

You can switch off monitoring relative to each other with existing definition tables by using the **DONTTEST** column. One or more objects can be defined in a CMO table. Collision monitoring is switched off for all objects defined in this table relative to the objects defined for them in the **FILE** column.

In the description table below, this means that, for example, in line 12 the objects defined in "CMO_LiftTable" will not be monitored for collision with the objects defined in "CMO_Cabin."

NR	KEY	AXIS	COORD	ON/OFF	FILE	DONTTEST
0	TOOLFILE					
1	SUBFILE1					
2	MachAxis B					
3	CMO				CMO_Portal	
4	MachAxis X					
5	MachAxis Y					
6	CMO				CMO_FloorSection	
7	Trans	X	470.0L92			
8	Trans	Y	-282.405			
9	Trans	Z	-900			
10	CMO			1	CMO_Cabin	
11	MachAxis Z					
12	CMO				CMO_LiftTable	CMO_Cabin
13	MachAxis C					
14	CMO				CMO_TurnTable	CMO_LiftTable
	[END]					



2. Exclusion table

DCM also makes it possible to exclude the collision of an entire machine component (definition table for CMOs) against objects from various other machine components. Here the indication of complete definition tables alone does not always suffice.

From the **DONTTEST** column it is possible to call exclusion tables consisting of only a **DOC** column.

In this column you can enter all names of the various collision-monitored objects (= names in the **DOC** column of the objects in the definition tables). Collision monitoring is switched off for all objects named in this table relative to the objects defined for them in the **FILE** column.



Note

Unique names must be assigned to different CMOs in the **DOC** (iTNC 530) or "Description" (KinematicsDesign) column.

Table **ExcludeDef1**

NR	DOC
0	Rotary table
1	Clamper front
2	Clamper rear
4	Cabin left
5	Cabin right
6	Tool ^a
[END]	

- a. "Tool" is a name reserved for the tool by the system. See the special case below for the meaning.

In the description table below, this means that, for example, in line 3 the objects defined in **CMO_Portal** will not be monitored for collision with the objects defined in the name table **ExcludeDef1** above.

NR	KEY	AXIS	COORD	ON/OFF	FILE	DONTTEST
0	TOOLFILE					
1	SUBFILE1					
2	MachAxis	B				
3	CMO				CMO_Portal	ExcludeDef1
4	MachAxis	X				
5	MachAxis	Y				
6	CMO				CMO_FloorSection	
7	Trans	X	470.0L92			
8	Trans	Y	-282.405			
9	Trans	Z	-900			
10	CMO			1	CMO_Cabin	
11	MachAxis	Z				
12	CMO				CMO_LiftTable	
13	MachAxis	C				
14	CMO				CMO_TurnTable	
	[END]					

Special case: Tool with the predefined CMO name "Tool"

In some cases it can be necessary to exclude the tool from monitoring for collision with certain machine components (e.g. tool and tool carrier relative to portal for B heads). For mechanical reasons a collision between these components is impossible, but the tool is also permanently monitored for collision with all objects. For example, if the distance between the tool and a machine component with which it cannot collide becomes too small, this could already lead to a reduced feed rate. In this case the tool can be excluded according to the "Exclusion table" method described above by using the "Tool" name to remove this possibility of collision from monitoring.



Activating and deactivating monitoring

There are various methods for activating and deactivating collision monitoring.

■ Manual

In the **Manual** operating mode, press the **Collision** soft key to open the "Collision monitoring (DCM)" menu, and activate or deactivate DCM for the **Manual** and **Program Run** operating modes (active/inactive).

Depending on the settings for collision monitoring (DCM) via the COLLISION soft key and the actual operating situation, the collision monitoring behavior was sometimes ambiguous up to software version 60642x-03. If DCM was **active** for **Program Run** and **inactive** for **Manual Operation**, collision monitoring was inactive in the following cases:

1. If the program was interrupted in the Program Run operating modes by pressing NC stop and the MANUAL TRAVERSE soft key, for example.

Collision monitoring was not active when the axes were moved with the axis-direction keys or the handwheel.

2. If the handwheel was moved in the Program Run operating modes while the program was stopped, collision monitoring was inactive during M118 or handwheel superimposition via the Global Settings.

In order to provide optimum protection, as of software version 60642x-03 the collision monitoring function is active in the above-mentioned cases if it is enabled in the MOD dialog for Program Run OR Manual Operation.

■ PLC module or FN17

- It is always possible to create two kinematics tables with the same kinematics description on the control for each application. One of these tables would not contain any descriptions of collision objects (CMOs). Depending on your needs, you can activate the kinematics with or without collision monitoring via the PLC (Module 9097) or with **FN17: SYSWRITE ID290 NR1**.
- PLC Module 9063 is used to activate and deactivate collision monitoring (DCM) for the Program Run modes.
- Bit 3 of Module 9221 is used to deactivate collision monitoring for PLC positioning movements. This means that it is possible to position an axis when DCM is active and the reference marks have not yet been traversed in all axes. A PLC positioning command can only be executed if collision monitoring is deactivated for all axes involved in the positioning movement. The deactivation of collision monitoring for the PLC positioning movement does not affect the status information provided by Module 9064.

■ WRITE TO KINEMATIC

At the same time, monitoring of these CMOs can be switched off by entering "1" in the **ON/OFF** column of the **active** description table. This can also be during program run in the active description table using the **WRITE TO KINEMATIC** function.

Example:

Deactivation of an active COM table so that collision monitoring is deactivated for a certain action, such as a tool change.

Table in the new kinematics description format before overwriting:

NR	KEY	AXIS	COORD	ON/OFF	FILE	DONTTEST
:						
1	CMO				CMO_Head	PLC:\Kinemat\...
2	MachAxis	B				
3	CMO				CMO_Porta1	PLC:\Kinemat\...
4	MachAxis	X				
:						
[END]						

0 BEGIN PGM N545TCM MM	Comment:
:	
4 WRITE TO KINEMATIC AT COLUMN "ON/OFF" CAPTURE "FILE" KEY "CMO_Porta1" = 1	Overwrite cell in the kinematics table
:	

Table after overwriting with collision monitoring switched off for the **CMO_Porta1** object:

NR	KEY	AXIS	COORD	ON/OFF	FILE	DONTTEST
:						
1	CMO				CMO_Head	PLC:\Kinemat\...
2	MachAxis	B				
3	CMO			1	CMO_Porta1	PLC:\Kinemat\...
4	MachAxis	X				
:						
[END]						

**Note**

Ensure that no matter which method was used to deactivate collision monitoring, that it is switched on again after the desired action (without collision monitoring) has finished.



DCM in the Test Run mode

You can check a machining process for potential collisions in the **Test Run** operating mode in order to ensure that time-consuming machining processes are not canceled by collision messages. You specify the datum for the collision test performed in this operating mode in the "workpiece blank in working space" MOD function.

For more information on collision monitoring in **Test Run** mode, refer to the User's Manual for the iTNC 530.

Set MP7507 to define the operating modes (editing, simulation or program run modes of operation), in which the SELECT KINEMATICS soft key is to be displayed under MOD.

In order to select kinematics for the simulation:

- ▶ While in the Editing/Simulation operating mode, press the MOD key.
- ▶ Press the SELECT KINEMATICS soft key.

You can simulate NC programs that are adapted to a certain kinematics model or that switch the kinematics. The kinematics can also be displayed graphically in the simulation:

- ▶ Press the screen layout key.
- ▶ Press the **KINEMATIC** soft key.

In order to select kinematics for the real machine or for the machining operating modes:

- ▶ While in a machining operating mode, press the MOD key.
- ▶ Press the SELECT KINEMATICS soft key.

The previous method for selection, via the **KINEMATIC** code number, still exists. It activates the kinematics for both operating modes.

Further constraints:

- After the control has booted, the machine-parameter main file is in effect for the program run and editing/simulation modes of operation.
- The kinematics model selected for simulation (editing mode of operation) is saved in non-volatile memory.
- PLC switching of the kinematics, individual machine parameters or machine parameter subfiles will only affect the program run modes of operation.
- Changes by the NC program or an NC macro will always affect the current operation mode, in which the NC program or NC macro is running.
- Selecting the machine parameter main file (manually and via PLC) affects all operating modes.

MP7507

Select the kinematics for the operating mode

Input:

%xxx

Bit 0

0: Kinematics cannot be selected in Editing operating modes

1: Kinematics can be selected in Editing operating modes for simulation in Test Run mode

Bit 1

0: Kinematics cannot be selected in Program Run operating modes

1: Kinematics of the real machine can be selected in Program Run operating modes

Bit 2

0: 3D ROT soft key is not available in Test Run mode

1: 3D ROT soft key is available in Test Run mode

Bit 3

0: Code number KINEMATIC permitted

1: Code number KINEMATIC disabled

Configuration of the definition table

Definition table for collision-monitored objects (CMO table)

The CMO table contains the descriptions of cuboids, cylinders or planes, which are inserted as collision-monitored objects at points in the transformation sequence.

It is possible to use uniform kinematics on machines with and without software option 40.



Danger

This means that an error message does not appear if DCM is not available. Even if there are CMOs in the kinematics tables, there is no protection against collisions without software option 40 (DCM).

A definition table for one or more machine objects can be inserted at the following points in the transformation chain:

- Description table (main kinematics)
- In **SUBFILE1**, **SUBFILE2** (partial kinematics via assignment table) and in a **TOOLFILE** (dynamic tool-carrier kinematics during tool change).

Creation of a CMO table:

- ▶ In program management, switch to the desired directory and enter the name of the description table, including the extension .TAB.
- ▶ Choose the table format with the KEY, X, Y, Z, AXIS, RADIUS, HEIGHT and DOC fields.
- ▶ With the aid of the formatting table for a CMO table shown below, enter the description of the collision-monitored object.



Note

Unique names must be assigned to different CMOs in the **DOC** (iTNC 530) or "Description" (KinematicsDesign) column.

Format of the CMO table:

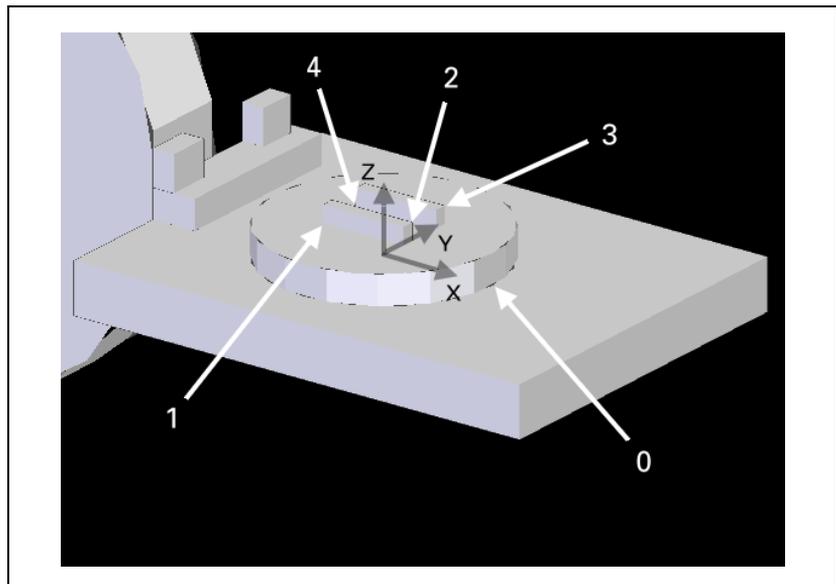
Column	Input	Description
NR	0, 1, 2 ...	Automatic line numbering
KEY	Cylinder	The following parameters for the X, Y, Z, AXIS, RADIUS , and HEIGHT columns define a cylindrical object, starting from the position of the previous translation, rotation or defined machine axis. Starting from this point, the cylinder is offset from X, Y and Z , and defined in the direction of the axis entered in AXIS , with radius RADIUS and height HEIGHT .
	Cuboid	The following parameters for the X, Y and Z columns define a rectangular object, starting from the position of the previous translation, rotation or defined machine axis. Starting from this point, the cuboid is defined from a minimum point in X, Y and Z , to a maximum point in the next line in X, Y and Z , (as with the BLK definition in conversational programming).
	LimitMin LimitMax	Here the columns X, Y and Z can be used to define limiting planes in the X, Y and Z directions.
AXIS	X, Y, Z	Entry of the axis designation for the direction of "Cylinder"-type objects in the KEY column
RADIUS	E.g. 40 [mm]	Entry of the radius for "Cylinder"-type collision-monitored objects in the KEY column
HEIGHT	E.g. 150 [mm]	Entry of the height for "Cylinder"-type collision-monitored objects in the KEY column
DOC	Name of the object ^a	Entry of a unique name for the object entered in the KEY column, so that in case of an imminent collision, a clear message regarding the danger of collision can be output in the header (e.g. Tool <-> Rotary table)
	TT_ENABLE_PR OBING	During tool measurement with a TT, this CMO, which protects the TT, can automatically be excluded from collision monitoring. The CMO will then automatically be excluded during the execution of touch probe cycles TCH PROBE 30/31/32/33 or 480/481/482/483, see page 1449.

- a. The "Tool" name is already assigned by the system, and may not be used twice.

Example of a CMO table ("CMO_TurnTable" object: no. 14 from the description table example):

NR	KEY	X	Y	Z	AXIS	RADIUS	HEIGHT	DOC
0	Cylinder	0	0	10	Z	270	70	Rotary table
1	Cuboid	-100	-70	70				Clamper front
2		100	-40	120				
3	Cuboid	-100	40	70				Clamper rear
4		100	70	120				
[END]								

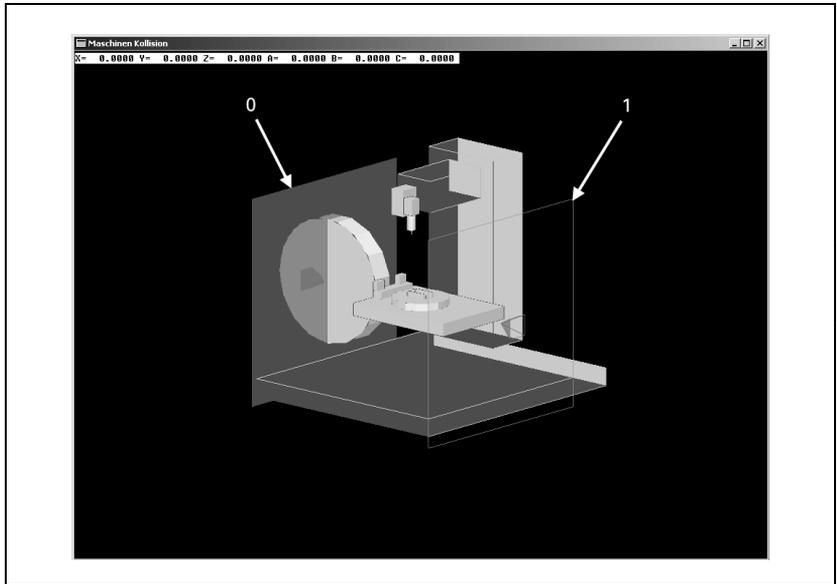
Representation of a collision-monitored object (CMO) with the CMO table above. The coordinate system shown indicates the transformed reference point, which is active in the description table. In this case it is the center point of the rotary table.



Example of a CMO table ("CMO_Cabin" object) as an example for limiting planes:

NR	KEY	X	Y	Z	AXIS	RADIUS	HEIGHT	DOC
0	LimitMin	-1200						Cabin left
1	LimitMax	-1000						Cabin right
	[END]							

Representation of limiting planes as in the CMO table above ("CMO_Cabin") with two planes, which were defined in the X direction



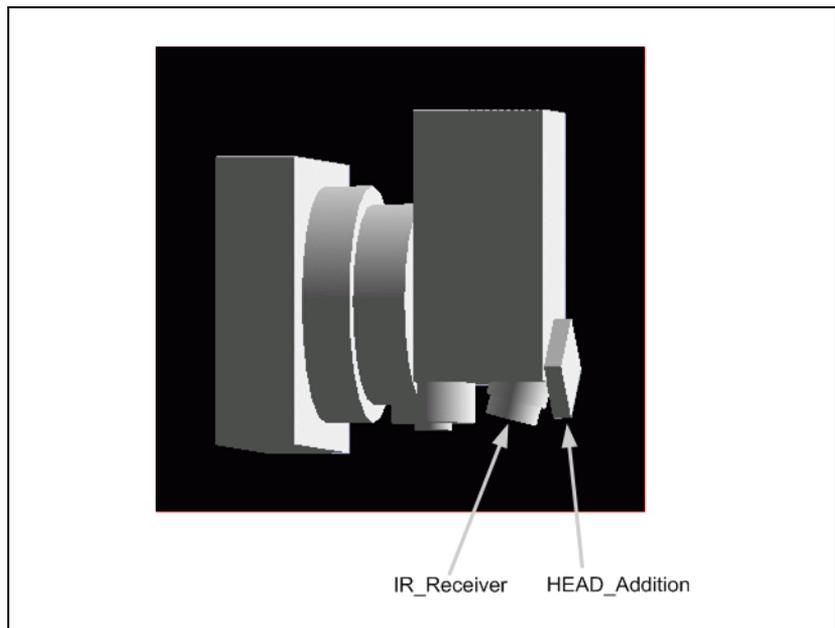
The coordinate transformations of collision monitored objects, abbreviated as CMO, can be entered directly in the object-description file of a CMO.

Shifts and rotations can be entered in a CMO definition table via the keyword **TRANS**, an axis (X, Y, Z, A, B or C) in the **AXIS** column and a value in the new **COORD** column. This functions like entering the description of a kinematics configuration. The coordinate transformations in a CMO definition table are only effective within the respective object-description file. There they are effective for all subsequent objects, however.

This makes it possible in just one object-description file

- to describe rotated and unrotated objects,
- to describe multiple objects after a rotation that then all have the same position, and
- to shift the datum and then perform a rotation.

The following kinematics section will be used as an example to show the changes in the description of the CMOs.



The description of the kinematics with CMOs can look like this:

NR	KEY	AXIS	COORD	ON/OFF	FILE	DONTTEST
0	TOOLFILE					
1	CMO				HEAD	
2	Trans	Z	67			
3	Trans	Y	-105			
4	Trans	A	15			
5	CMO				IR_Receiver	
6	Trans	A	-15			
7	Trans	Y	105			
8	Trans	Z	-67			
9	Trans	Y	-140			
10	Trans	Z	70			
11	Trans	B	135			
12	CMO				HEAD_Addition	
13	Trans	B	-135			
14	Trans	Z	-70			
15	Trans	Y	140			
16	Trans	X	0.025			
17	Trans	Z	145			
18	MachAxis	B				
19	CMO				Z-Portal	
20	MachAxis	Z				
21	MachAxis	X				
..	...					
	[END]					

The definition table for IR_Receiver:

NR	KEY	X	Y	Z	AXIS	RADIUS	HEIGHT	DOC
0	Cylinder	0	0	-35	Z	35	35	Connector IR_Receiver
1	Cylinder	0	0	-57	Z	30	22	IR_Receiver
	[END]							

The definition table for HEAD_Addition:

NR	KEY	X	Y	Z	AXIS	RADIUS	HEIGHT	DOC
0	Cuboid	-40	-20	-40				HEAD Addition
1		40	00	40				
	[END]							

The possibility of including the transformation of a CMO into the CMO's own definition table improves the clarity of the kinematics table. This also makes it easier to integrate CMOs, since there are no changes to the actual kinematics of a machine.

As a result, the description of the kinematics with CMOs can look like this:

NR	KEY	AXIS	COORD	ON/OFF	FILE	DONTTEST
0	TOOLFILE					
1	CMO				HEAD	
2	CMO				IR_Receiver	
3	CMO				HEAD_Addition	
4	Trans	X	0.025			
5	Trans	Z	145			
6	MachAxis	B				
7	CMO				Z-Portal	
8	MachAxis	Z				
9	MachAxis	X				
..	...					
	[END]					

Transformations that were only contained in the kinematics tables for the description of CMOs are omitted completely and instead integrated in the CMO definition table.

The new definition table for IR_Receiver:

NR	KEY	X	Y	Z	AXIS	COORD	RADIUS	HEIGHT	DOC
0	Trans				Z	67			
1	Trans				Y	-105			
2	Trans				A	15			Position1
3	Cylinder	0	0	-35	Z		35	35	Connector IR_Receiver
4	Cylinder	0	0	-57	Z		30	22	IR_Receiver
	[END]								



As an alternative, the **X**, **Y** and **Z** columns can be used for the description of transformations, but then no entry is permitted in the **AXIS** column. If there is an entry in the **AXIS** column, the **X**, **Y** and **Z** columns are ignored.

The new definition table for HEAD_Addition:

NR	KEY	X	Y	Z	AXIS	COORD	RADIUS	HEIGHT	DOC
1	Trans		-140	70					
2	Trans				B	135			
3	Cuboid	-40	-20	-40					HEAD Addition
4		40	00	40					
[END]									

Transformations in the CMO definition table must always occur before the definition of objects. These transformations only serve to describe the position and orientation of the CMOs, and do not need to be canceled in order to describe the further kinematics of the machine. If there is a transformation at the end of a definition table, the error message **Kinematic table defective** is output.

Excluding the TT for tool measurement

During tool measurement with a TT, the CMO protecting the TT can automatically be excluded from collision monitoring.

A CMO that has been identified correspondingly will then automatically be excluded during the execution of touch probe cycles TCH PROBE 30/31/32/33 or 480/481/482/483. The CMO for the touch probe is identified by entering **TT_ENABLE_PROBING** in the **DOC** column of the kinematics table.

The CMO identified in this way will be excluded from collision monitoring before the beginning of the touch probe cycle. At the end of the cycle or when the cycle is canceled, the CMO will automatically be included in collision monitoring again.

Another possibility for excluding an identified CMO from collision monitoring is to use the function **FN 17: SYSWRITE ID 990**. The function **FN 17: SYSWRITE ID 990 NR6 = 1** causes switching to input X13. The CMO is then excluded from collision monitoring for as long as X13 is active.

The following must be kept in mind:

- You can use **TT_ENABLE_PROBING** for only one CMO.
- If a touch probe cycle is canceled with the **EXTERNAL STOP** soft key, the tool (TT) should be retracted in the **Manual Operation** mode before pressing the **INTERNAL STOP** soft key. The **INTERNAL STOP** soft key results in the CMO being included in collision monitoring again, and the tool would then be immobilized by collision monitoring. DCM would trigger an error message reporting a distance < 2 mm. The tool (TT) can then not be retracted until collision monitoring has been switched off.
- After retraction from the measuring point, the tool (TT) must be located sufficiently far from the area for which protection is active again. After retraction, the distance must be > 5 mm. This safety clearance can be defined in MP6540.x.

Changing CMOs in NC programs

The NC function **WRITE TO KINEMATIC** can be used to adapt the position, orientation and size of CMOs to new situations even during execution of an NC program. It is now possible to search active CMO definition tables for keywords (**KEY**) and to overwrite individual cells in the table. When entering keywords in the **DOC** column, make sure that each keyword is only entered once.



Note

If you change the kinematics tables or CMO definition tables with **WRITE TO KINEMATIC**, then the values in the tables are overwritten. If you do not rescind the changes you made with **WRITE TO KINEMATIC** in the NC program, the new values remain in the table. The changes to the machine kinematics are then not just temporary.

Example for a change to a CMO definition table in an NC program:

	Comment:
0 BEGIN PGM N545TCM MM	
:	
5 WRITE TO KINEMATIC AT COLUMN "COORD" CAPTURE "DOC" KEY "Position1" = 25	Overwrite cell in the kinematics table
6 WRITE TO KINEMATIC AT COLUMN "HEIGHT" CAPTURE "DOC" KEY "IR_Receiver" = 45	Overwrite cell in the kinematics table
:	
43 WRITE TO KINEMATIC AT COLUMN "COORD" CAPTURE "DOC" KEY "Position1" = 15	Write original value to cell in table again
44 WRITE TO KINEMATIC AT COLUMN "HEIGHT" CAPTURE "DOC" KEY "IR_Receiver" = 22	Write original value to cell in table again
:	

DCM and handwheel superimpositioning

DCM can now also monitor handwheel superimposition (with M118 or Global Program Settings) for collisions (FCL function). During program run, however, the handwheel is not active.

Handwheel superimposition with active DCM is only possible when program run is stopped with **External Stop** or with the **Single block** operating mode. If an offset is defined for handwheel superimposition after the External Stop, this offset is considered by collision monitoring when the NC program run is continued. This behavior also applies to the virtual tool axis VT.



DCM for counter axes

The changes in position of manually-operated counter axes in the **Manual Operation** and **E1. Handwheel** operating modes are detected by the collision monitoring. However, counter axes and controlled axes may not be moved simultaneously if collision monitoring is to be effective.

In the **Positioning with MDI, Program Run – Single Block, Program Run – Full Sequence** and **smarT.NC** machining modes, the counter axes may only be moved at standstill, M strobe, etc., in order to ensure protection by DCM against collisions.

The NC axes can then also be moved in the machining modes while being protected against collision.



Danger

If you position manually-operated counter axes during execution of an NC program or during an individual NC block (during calculation of path interpolations), the new positions of the counter axes cannot be detected or considered by DCM. There is no protection against collisions.

PLC modules

Module 9063 Deactivate and activate collision monitoring

With PLC module 9063, you can influence the status of DCM collision monitoring in the program run modes of operation.

Condition:

- During program run, collision monitoring can only be deactivated during an NC strobe.
- PLC module 9064 enables you to interrogate the status of collision monitoring for the current operating mode.
- It may occur that more than one PLC cycle is executed between activation/deactivation and feedback from the PLC module 9064.

Call:

PS B/W/D/K <Mode>

0: Collision monitoring active/inactive during program run

PS B/W/D/K <Action>

0: Deactivate collision monitoring

1: Activate collision monitoring

CM 9063

Error code:

Marker	Value	Meaning
M4203	0	Function was performed correctly
	1	Error code in W1022
W1022	1	Invalid value for mode/action
	8	Call was during manual operation mode or during reference run
	21	Call was during program run without strobe signal
	45	Collision monitoring has already been programmed

Module 9064 Status information about collision monitoring

With Module 9064 you interrogate whether collision monitoring is active in the currently selected operating mode. The module was expanded by Mode 1.

Constraints:

- Collision monitoring is not active in the Reference Run mode.

Call:

PS B/W/D/K <Mode>

0: Configured status (depends on module 9063 and soft key)

1: Status in the current mode of operation (0 in Reference Run mode)

CM 9064

PL B/W/D <Status information>

0: Monitoring not active

1: Monitoring active

Error code:

Marker	Value	Meaning
M4203	0	Status ascertained
	1	Error code in W1022
W1022	1	Invalid value for mode



8.6.2 Fixture monitoring with DCM

DCM collision monitoring (software option 40) of the iTNC 530 also protects fixtures from collisions. Models of fixtures can be inserted into a machine's kinematics for this.

HEIDENHAIN provides parameterized descriptions of standard fixtures. The FixtureWizard for PCs, which is available free of charge, enables you to use these standard fixtures for describing your own fixtures, and to transfer the description to the iTNC 530.

In the Manual Operation mode, the fixture management function is used to set up the fixtures in the machine's work envelope. An interactive menu allows you to use the integrated measuring cycles to measure and automatically transfer the fixture data, and to define the variable input values, such as the jaw distance of a vise.

The iTNC provides a test program for checking the defined fixtures. In the Program Run, Full Sequence mode the iTNC moves to defined test points and evaluates them. The result is displayed on the screen, or is available as a log file.

That chapter on collision monitoring describes the fundamentals of collision monitoring and the creation of collision-monitored objects (CMOs), see "DCM – monitoring the working space for collisions" on page 1424. Kinematics descriptions that are used for collision monitoring may contain a maximum of 100 transformation entries (**Trans** and **MachAxis**). However, not all of the active CMOs must be considered. You only have to add the transformations of the CMO with the most transformations to the transformation entries of the actual machine kinematics.



Attention

Finally, the fixture model for collision monitoring and the behavior of the complete collision model must be tested on the machine as usual.

Using DCM to protect fixtures from collisions requires essentially three steps:

■ **1. Modeling fixtures with KinematicsDesign**

HEIDENHAIN makes fixture templates such as vises or jaw chucks, which were created with the PC program KinematicsDesign, available to you via the programming station version of the control. You can create additional fixture templates and provide them to your end users. The fixture templates have the file name extension *.cft.

■ **2. Defining fixtures with FixtureWizard**

With the FixtureWizard, the machine operator defines the exact dimensions of the fixture by entering parameter values in the fixture template. The FixtureWizard is available as a stand-alone PC tool and as a component of the iTNC fixture management feature. It generates an insertable fixture with concrete dimensions defined by you. Insertable fixtures have the file name extension *.cfx.

For information on operating the FixtureWizard, refer to Chapter 11.3 – "Fixture Monitoring" of the User's Manual for the iTNC 530.

■ **3. Inserting the fixtures in the kinematics model on the iTNC 530 and testing them**

In an interactive menu whose texts you create in KinematicsDesign, the TNC guides the user through the actual probing process. The probing process consists essentially of the performance of various probing functions on the fixture and entering variable quantities, such as the jaw gap of a vise. After you have inserted the fixture, you can have the TNC create a measuring program as needed with which you can have the actual position of the inserted fixture compared with the nominal position. If the deviations between the nominal and actual positions are too large, the TNC issues an error message.

The most important functions are described below by using a simple example fixture. Use the fixture templates from HEIDENHAIN as examples if you want to create more complex fixture templates. However, be sure not to change the original HEIDENHAIN templates.

Prerequisites

- The software option 40 (DCM - Dynamic Collision Monitoring) must be enabled.
- Your machine must have a 3-D touch probe for workpiece measurement. Otherwise you cannot place the fixture on the machine.
- **FIXTURE.FUNCTION = YES** must be entered in the OEM.SYS.
- Permissible insertion points for the fixtures must be defined in the kinematics description , see page 1456.



Files

The following files are involved in fixture management:

■ **Template for a fixture – *.cft**

Variable template for a fixture. This template is created with KinematicsDesign. Fixture templates from HEIDENHAIN are in the directory **TNC:\system\fixture\JH**. Please create a separate directory for your own fixture templates. Use the FixtureWizard to edit the created *.cft file and to generate a *.cfx file from it.

■ **Individual fixture – *.cfx**

Fixture that can be inserted in the machine kinematics, and have its position and orientation ascertained. It only contains variables of the **Manual input field** type that have to be defined when inserting the fixture. Once a *.cfx file has been inserted and the position and orientation ascertained, a test program can be generated from it, if necessary. The *.cfx files created with FixtureWizard are saved in the same default directory, from which the *.cft file was opened.

■ **Test programs for fixtures – *.H**

Once the fixture has been inserted and its position and orientation ascertained (*.cfx), the CREATE TEST PROGRAM soft key becomes available if the fixture contains test points, see page 1475. Press the soft key to create an NC program, which automatically probes the test points in the fixture kinematics. In this way, you can check the position and orientation of the respective fixture.

Test programs are identified by an index and are always saved in the directory **TNC:\system\Fixture\TpCheck_PGM**. The index is incremented automatically if several identical fixtures are inserted at the same insertion point (e.g. vise.001.H, vise.002.H, vise.003.H).

Please do not change or delete the test programs *.H. The fixture management needs all these files and usually deletes them automatically when the fixture is removed from the machine kinematics.

■ **System files for fixture monitoring**

Other files necessary for fixture monitoring are automatically saved in the directory PLC:\Fixture\. Please do not change or delete these files \$tpcheck\$.cfg, *.path, *.cfx and *.tab. The fixture management needs all these files and usually deletes them automatically when the fixture is removed from the machine kinematics.

Tools

The following programs are involved in fixture management:

■ **KinematicsDesign**

KinematicsDesign is a program for interactive creation of control kinematics, fixture templates, completed fixtures and tool carrier kinematics. The program is available for download from the HEIDENHAIN homepage, and is available on the iTNC 530 control.

■ **Fixture Wizard**

With the FixtureWizard, you use the fixture templates and the templates for the tool carrier kinematics to generate executable kinematics files for the control. These kinematics files can be used for DCM collision monitoring of the control. The program is available for PCs as a part of KinematicsDesign. It is also available on the control.

Defining insertion points

As OEM, you must define the location point or the location points of the machine kinematics at which a fixture can be inserted. The keyword **C1amp** was introduced for this purpose. The keyword **C1amp** in the column **KEY** of the kinematics table identifies the locations at which fixtures can be inserted. The entry in the corresponding **DOC** column is displayed as a dialog when the fixture is inserted.

Also, you must specify a file name for a *.tab file in the column **FILE**. In this file, the control saves a path to a *.tab file in the directory **PLC:\Fixture**. After the fixture has been inserted and its position and orientation ascertained, this file contains all data relevant for collision monitoring, if the user has assigned values to all open variables of the fixture.

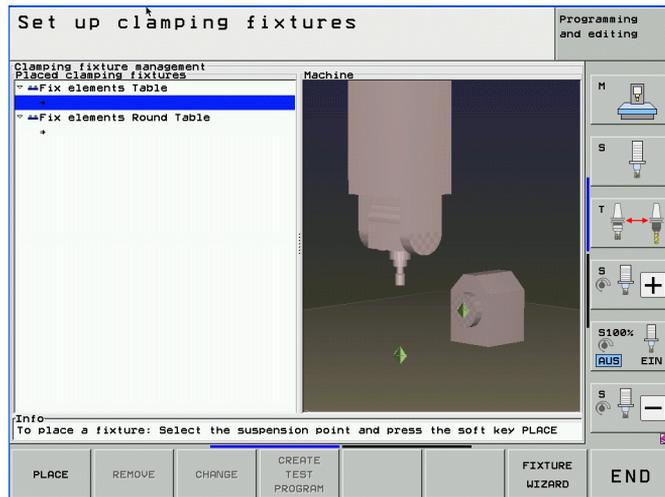
You must also create the tab file for the entry in **FILE**. The file must be of the prototype **CLAMP.TAB**.

Example of two insertion points in a kinematics table:

NR	KEY	AXIS	COORD	DOC	FILE
0	TOOLFILE			ToolCarrier	
1	CMO			Spindle	CMO\...
2	CMO			Head	CMO\...
3	Trans	X	+0	Transform X	
4	Trans	Z	+200	Transform Z	
5	MachAxis	B		Axis B	
6	CMO			Fork	CMO\...
7	Trans	X	+0	Transform X	
8	Trans	Y	+0	Transform Y	
9	MachAxis	C		Axis C	
10	CMO			CMO_SlideZ	CMO\...
11	MachAxis	Z		Axis Z	
12	MachAxis	Y		Axis Y	
13	MachAxis	X		Axis X	
14	MachBase			Base	
15	Trans	Z	-700	Transform Z	
16	CMO			Table	CMO\...
17	C1amp			Fixture 1	Fix1.tab
18	Trans	X	+500	Transform X	
19	Trans	Y	+0	Transform Y	
20	Trans	Z	+200	Transform Z	
21	CMO			RoundTableHouse	CMO\...
22	MachAxis	A		Axis A	
23	CMO			RoundTable	CMO\...
24	C1amp			Fixture 2	Fix2.tab
[END]					



The location points defined by you can be selected by the user after pressing the FIXTURE MANAGEMENT soft key in the 3rd soft-key row.



Several different or identical fixtures can be inserted at any insertion point in the description. The automatic assignment of an index to every fixture is used to distinguish between identical fixtures that are inserted at the same insertion point in the description. (e.g. vise.001, vise.002, vise.003)

Modeling fixtures

KinematicsDesign helps you to model the fixtures. You can define readily usable fixtures (file extension *.cfx), or you can create variable fixture templates (file extension *.cft).

For fixture templates (*.cft) you define the shape of the fixture by using variables for different dimensions of the fixture objects (CMOs). The fixtures are designed in the same way as the machine kinematics: CMOs, translations and transformations are used. Input parameters permit variable design of the fixture templates, and enable you to help the end user in the final definition, insertion and testing of the fixture.

The *.cft files for fixture templates are in a new table format in ASCII. The tables are an extension of the familiar columns of the kinematics tables *.tab of the machine kinematics, which were specially adapted for the modeling of fixtures. The tables can be opened with Microsoft Excel if the semicolon is selected as separator.

Fixture templates created by HEIDENHAIN are available on the HEIDENHAIN homepage at www.heidenhain.de under:

- ▶ Services and Documentation > Software > Fixtures
- ▶ Select the desired fixture: You will see a ZIP file with the appropriate files.
- ▶ Download the ZIP file and unzip it.
- ▶ Copy the unzipped files to the TNC in the **TNC:\system\Fixture\JH** directory.

Save only fixture templates from HEIDENHAIN in the **TNC:\system\Fixture\JH** directory, because this directory will be overwritten during a software update. Fixtures or fixtures templates that were created or changed by you should be saved in a separate directory, such as **TNC:\system\Fixture\OEMfixtures**, or at least under their own names in order to avoid losing data during a software update.



The following parameters or columns with the appropriate input options are available for modeling fixtures:

Format of the CFT/CFX table:

Column	Input	Description
NR	0, 1, 2 ...	Automatic line numbering
KEY	■ CMO	Defines a monitorable collision body in the current kinematics sequence. This machine element is described by the file (table) referenced in the FILE column. Fixtures can also contain individual CMOs that have already been defined, see page 1474.
	■ Trans	Here a transformation of the indicated axis (AXIS column) in the current kinematics sequence by the value entered in the COORD column is performed. This can be a linear translation as well as a rotation about an axis.
	■ Cylinder	The following parameters for the X, Y, Z, AXIS, RADIUS, and HEIGHT columns define a cylindrical object, starting from the position of the previous translation, rotation or defined machine axis. Starting from this point, the cylinder is offset from X, Y and Z, and defined in the direction of the axis entered in AXIS , with radius (RADIUS) and height (HEIGHT).
	■ Cuboid	The following parameters for the X, Y and Z columns define a rectangular object, starting from the position of the previous translation, rotation or defined machine axis. Starting from this point, the cuboid is defined from a minimum point in X, Y and Z, to a maximum point in the next line in X, Y and Z, (as with the BLK definition in conversational programming).

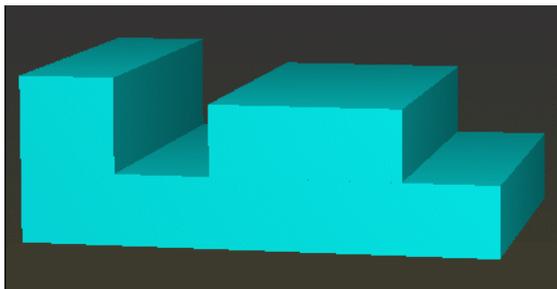
Column	Input	Description
		Only for fixtures (*.cft table):
KEY	■ Var	Page 8 – 1470
	■ VarParam	Page 8 – 1470
	■ VarTp	Page 8 – 1470
	■ VarEntry	Page 8 – 1470
	■ TpRef	Page 8 – 1464
	■ TpPos	Page 8 – 1464
	■ TpAng	Page 8 – 1464
	■ TpRefAng	Page 8 – 1464
	■ TpCheck	Page 8 – 1464
AXIS	X, Y, Z, A, B, C ...	Entry of the axis designation for function given in the KEY column (valid for the MachAxis , Trans and TPxxx parameters in the KEY column).
RADIUS	E.g. 40 [mm]	Entry of the radius for "Cylinder"-type collision-monitored objects in the KEY column.
HEIGHT	E.g. 150 [mm]	Entry of the height for "Cylinder"-type collision-monitored objects in the KEY column.
COORD	e.g. 47.092 [mm] or 45.05 [°]	Entry of the transformation value for linear axes (X, Y, Z, ...) in mm or for rotary axes (A, B, C) in degrees. The units are not entered. The iTNC infers the units from the axis designation entered (valid for the Trans parameter in the KEY column).
FILE	Path and file name ^a	Entry of the path and file name of a definition table for a collision-monitored object (valid for the CMO parameter in the KEY column).
DOC	Designator	Entry of a designator

Column	Input	Description
Only for fixtures (*.cft/*.cfx table):		
X, Y, Z	Position value	Entry of a position shift in X, Y, Z for "Cylinder" or "Cuboid"-type objects under "Position" in KinematicsDesign. Entry of the position at which the fixture is to be probed during probing cycles.
XSIZE, YSIZE, ZSIZE	Dimensions	Dimensions of a cube within a fixture file
NAME	Name	Name of a variable or a probing cycle
FUNC	Function	Function for determining the value of a variable from a manual probing cycle, see page 1469
VALUE	Value	Value of a variable, see page 1470
RANGE	Value range	Permissible value range of a variable with manual input field. e.g. "1...10"; "100..."; "...5, 10...15, -20"
SUBTYPE	Format	Format of a variable with manual input field, see page 1470.
DEFINCH	Value in inches	This is the default value for the variable when "Inch" is set. Only for linear input, see page 1470.
PICTURE	Help graphic	File with help graphic for a variable or a variable subfixture, see page 1470.
LABELID	Label text	Plain language or text ID for a variable, see page 1470.

- a. File paths in *.cft and *.cfx files must be entered relative to the active file.

Example

Example of modeling a simple fixture without variable, which consists of three individual bodies (CMOs).



To model a new fixture, proceed as follows:

- ▶ Start the KinematicsDesign PC tool.
- ▶ Select **File / New...** to create a new file.
- ▶ Select:
 - Fixture** to model a permanent fixture without variable input parameters for the user (*.cfx file), or
 - Fixture template** to model a variable fixture (*.cft), which will be given its final shape with the FixtureWizard tool.
- ▶ Enter any file name for the fixture.
- ▶ Design the desired fixture by inserting collision-monitored objects (CMOs) and the required transformations between the individual objects. The generation of the simple fixture shown above is described below in steps 1 to 5.
- ▶ KinematicsDesign saves the fixture as a *.cfx or *.cft file, depending on the selection.

Fixture templates *.cft and individual fixtures *.cfx enable the user to enter variable input parameters. Keep the following differences in mind for modeling:

- Design a fixture in the form of a fixture template *.cft in such a way that it can be changed with the FixtureWizard. The user then uses the FixtureWizard to adapt the incomplete fixture template to the actual fixture. This gives the user the possibility to change, for example, the length of the fixture or the jaw height, depending on the application (i.e. the real fixture).
- Design a readily usable fixture *.cfx, and give the user the possibility to change parameters of the otherwise complete fixture, which might not be constant quantities (such as the jaw distance) when ascertaining the position and orientation of the fixture. You enter these changes without using the FixtureWizard. During the process of ascertaining the position and orientation, the user is guided through a menu, in which the editable parameters can be defined.

For information on operating the FixtureWizard, refer to Chapter 11.3 – "Fixture Monitoring" of the User's Manual for the iTNC 530.

1

test2.cft [0] (Cuboid)

Basis

- Jaw_fix
- Trans X
- Trans Z
- Jaw_move

Bezeichnung: Basis

Position	Abmessung
X	+0.0000 200.0000
Y	+0.0000 150.0000
Z	+0.0000 30.0000

Achspostionen

Zurücksetzen

2

test2.cft [1] (Cuboid)

Basis

- Jaw_fix
- Trans X
- Trans Z
- Jaw_move

Bezeichnung: Jaw_fix

Position	Abmessung
X	+0.0000 40.0000
Y	+0.0000 150.0000
Z	+30.0000 40.0000

Achspostionen

Zurücksetzen

3

test2.cft [2] (Trans)

Basis

- Jaw_fix
- Trans X
- Trans Z
- Jaw_move

Bezeichnung: Trans X

Achse: X

Wert: +80.0000

Achspostionen

Zurücksetzen

4

test2.cft [3] (Trans)

Basis

- Jaw_fix
- Trans X
- Trans Z
- Jaw_move

Bezeichnung: Trans Z

Achse: Z

Wert: +30.0000

Achspostionen

Zurücksetzen

5

test2.cft [4] (Cuboid)

Basis

- Jaw_fix
- Trans X
- Trans Z
- Jaw_move

Bezeichnung: Jaw_move

Position	Abmessung
X	+0.0000 80.0000
Y	+0.0000 150.0000
Z	+0.0000 30.0000

Achspostionen

Zurücksetzen

When entering the positions of the collision bodies, the values should be entered in the **X**, **Y** and **Z** input fields under **Position** (as in the example in step 2). Additional transformations should not be used (as in this example using **Trans X** and **Trans Z** in steps 3 and 4). The additional transformations result in additional calculations during collision monitoring. This would unnecessarily require computing power.

Probe cycle

Along with the body of the fixture, you must also define the necessary probing points and probing cycles for the operator to clearly and unambiguously ascertain the position and orientation of the fixture. These points and cycles are defined in KinematicsDesign when you model the fixture. You can choose between various probing cycles (see format of the *.cft table).

Input possibilities for probing cycles:

Entry	Input	Description
Description (DOC)	Any text	Designation or name of the object, used in the fixture description.
Type (KEY)	■ TpRef	Probing cycle of type: Datum in plane Defines a manual probing cycle at this point in the fixture kinematics. The type prescribes which probing cycles are available for selection while ascertaining the position and orientation of the fixture.
	■ TpPos	Probing cycle of type: Datum in one axis Defines a manual probing cycle at this point in the fixture kinematics. The type prescribes which probing cycles are available for selection while ascertaining the position and orientation of the fixture.
	■ TpAng	Probing cycle of type: Basic rotation Defines a manual probing cycle at this point in the fixture kinematics. The type prescribes which probing cycles are available for selection while ascertaining the position and orientation of the fixture.
	■ TpRefAng	Probing cycle of type: Basic rotation with corner Defines a manual probing cycle at this point in the fixture kinematics. The type prescribes which probing cycles are available for selection while ascertaining the position and orientation of the fixture.
	■ TpCheck	Probing cycle of type: Test point Defines a manual probing cycle at this point in the fixture kinematics. These probing points are used when creating a test program, see page 1475.
Name (NAME)	Any text	Name for the probing cycle. This name can be used to access the results of the probing cycle.



Entry	Input	Description
Axis (AXIS)	Axis identifier	Indication of the main axis for the probing cycle.
X, Y, Z	Position value	Entry of the position at which the fixture is to be probed.
Label text (LABELID)	Any text	Entry of a plain-language name or text ID. Is displayed for the operator as an additional aid when inserting the fixture.
Tilt angles (SPA, SPB, SPC)	Tilting angles in the coordinate system of the fixture	Only for probing cycles of type: Test point (TpCheck). Entry of a tilting angle (spatial) in the coordinate system of the fixture. The tilting angle must be used if a test point cannot be probed without tilting. In that case, the test program automatically tilts the coordinate system for this test point. The tilt angle is only active for this test point. It is automatically reset after probing.

Using the probing cycle, the operator inserts the fixture at the prescribed insertion point (**CLAMP**) in the kinematics. Only once the exact position and orientation of the fixture have been ascertained can it then effectively be monitored for collisions.

Insertion of a corresponding probing cycle necessitates additional amendments to the fixture kinematics. These amendments are necessary in order for the results of the probing cycle to be included in the fixture kinematics. For example, variable translations or rotations that shift the fixture in the kinematics, depending on the results from the probing cycle, must be inserted.

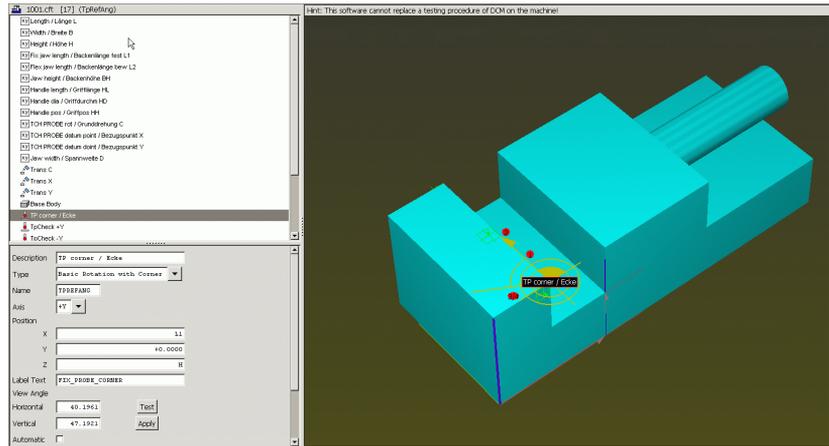
Proceed as follows to insert a probing cycle into the fixture kinematics:

- ▶ Insert the appropriate probing cycle at the corresponding location in the fixture kinematics. Specify the necessary values.
- ▶ At the beginning of the fixture kinematics, define variables of the type **manual probing cycle**. In the **Function** entry you specify which result from the probing cycle will be loaded in to the variable, see page 1469. The variable is addressed under the name given in the **Name** entry (e.g. TouchedX).
- ▶ After the variables, but before the actual fixture kinematics, insert variable transformations or rotations. These values are determined through the probing cycle, and are then available as variables (e.g. TouchedX).

As of KinematicsDesign version 1.3, a touch-probe cycle is no longer necessary for defining a fixture. This, together with the already existing possibility of specifying more than one insertion point in a fixture definition, will make it possible to realize so-called integrated fixture systems. An integrated fixture system with touch-probe cycles and more than one insertion point can be defined for these systems. The touch-probe cycle serves to define the position of the integrated fixture system. The insertion points serve to define the fixture positions if more than one fixture is used. The individual fixtures that can be installed in the integrated fixture system do not require individual separate touch-probe cycles. Their position and orientation must therefore be unambiguously defined by the integrated fixture system.

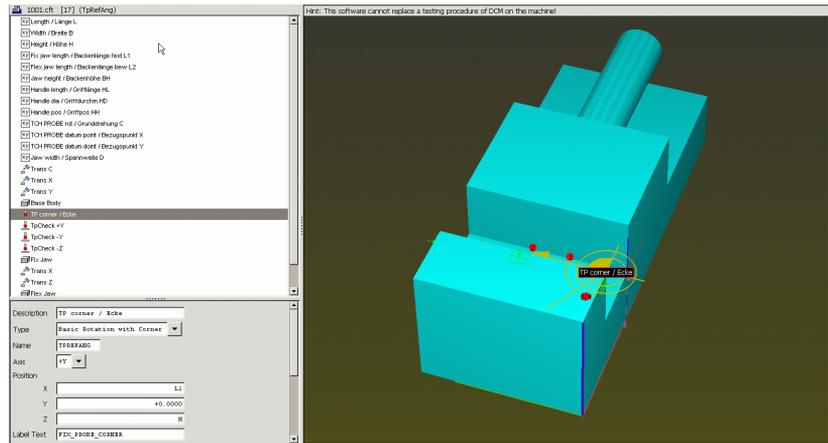
A view angle for the fixture can be entered manually for each touch-probe cycle. As before, you can also have KinematicsDesign select the view angle automatically.

Press the **Apply** button to save the current view of the fixture as view angle. Press the **Test** button to rotate the fixture to the position that is saved as view angle.

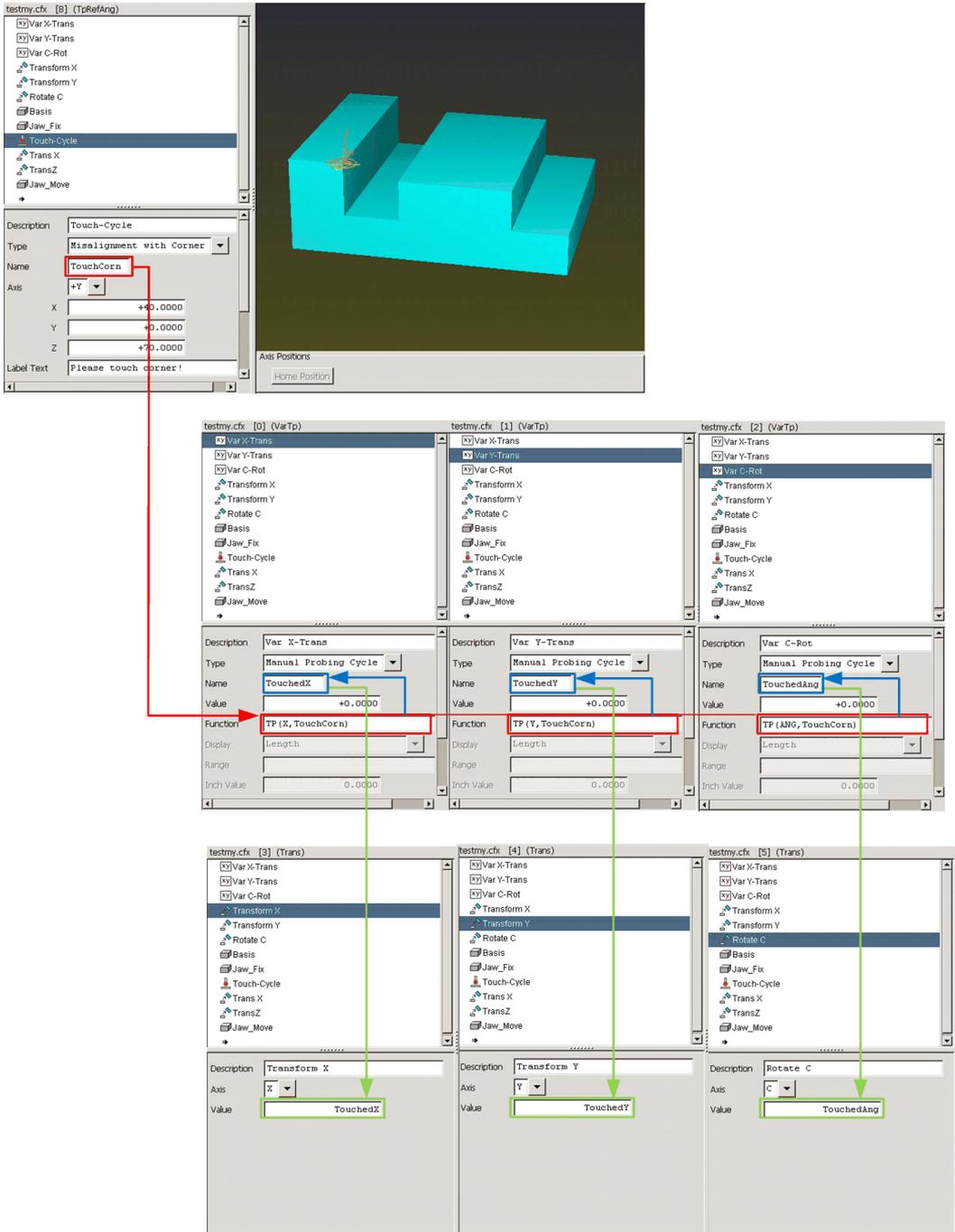


Entry	Input	Description
View angle (VIEWH, VIEWV)	View angle of the fixture	Manual definition of a horizontal and vertical view angle for the fixture. During this touch-probe cycle, the fixture is displayed to the operator at this angle.
View angle – Automatic (AUTOVIEW)	Automatic selection of the view angle	If this option is set, the view angle is calculated by KinematicsDesign and is used to display the fixture. Any manually entered view angles will be ignored.

The touch points required for a touch-probe cycle are automatically numbered to make the measurement of fixtures easier. The numbers inform the operator about the sequence in which the points have to be probed to define the position of the fixture. In the FixtureWizard, the touch points can be shown or hidden as desired.



The following overview illustrates the associations when data is exchanged between the probing cycle and variable transformations. In the example, a probing cycle with the name TouchCorn was inserted. The values from the probing cycle are loaded into the variables TouchedX, TouchedY and TouchedAng via TP functions. These variables, in turn, serve as values for the two transformations and the rotation. This way the exact position and orientation of the fixture is ascertained.



TP functions

You can use the TP functions to read out measured values from the manual probing cycles used in the insertion of the fixture and the ascertainment of its position and orientation.

TP functions have the following syntax:
TP(<value>,<cycle name>, {cycle index})

Entry	Input	Description
<Value>	X, Y, Z, ANG	<ul style="list-style-type: none">■ X: Principal axis value of a probing cycle■ Y: Minor axis value of a probing cycle■ Z: Value in direction of probing■ ANG: Angular value for rotation in the probing plane
<cycle name>	Text	Name of the probing cycle whose value is to be read.
<cycle index>	Whole number (1 to n)	<p>It can happen that the same part is used more than once in the fixture kinematics. This part can include a probing cycle that has the same name even though it is used more than once. Use the <cycle index> parameter to choose between the two cycles.</p> <p>Example: The CMO "Jaw chuck" contains 3 CMOs with the name "Jaw". Each "Jaw" includes a probing cycle with the name "Rotation". If you want to read the value of the rotation on the third "Jaw", the function must be entered as:</p> <p>TP (ANG,Rotation,3)</p>

Variables

For the flexible definition of fixtures, KinematicsDesign offers the possibility of creating variables for fixture kinematics. These are used for the insertion of the fixture (ascertaining its position and orientation), or for variable length information, for example.

Input possibilities for variables:

Entry	Input	Description
Description (DOC)	Any text	Object designation used in the fixture description.
Type (KEY)	■ VarEntry	Variable of the type: Manual input field Defines a number variable, whose value must be entered when inserting the workpiece via the control's user interface. The value can be changed by the operator without FixtureWizard and without ascertaining the position and orientation of the workpiece again (e.g., the distance between jaws changes, but the vise remains at the same position on the table).
	■ VarTp	Variable of the type: Manual probing cycle Defines a number variable, whose value is determined by a probing cycle when ascertaining the position and orientation of the fixture.
	■ Var	Variable of the type: Miscellaneous variable Defines a miscellaneous variable that can be used with the fixture kinematics, e.g. for the buffering of values. Value assignment by the operator not possible.
	■ VarParam	Variable of the type: Wizard parameter Defines a number variable in which a value can be entered in the FixtureWizard (e.g. variable distance between jaws).
Name (NAME)	Any text	Name of the variable with which it can be addressed.
Value (VALUE)	Numerical value	Value of the variable as a number or a formula for calculation.
Function (FUNC)	Function	Function for determining the value of a variable from a manual probing cycle (only for VarTp).
Format (SUBTYPE)	Position value	Format of a variable with manual input field (only for VarEntry, VarParam): Length: Linear value in [mm] or [inches] Angle: Angular value in degrees [°] Integer: Numeration type, only permits whole numbers Bool: Logical value, active/inactive through check box

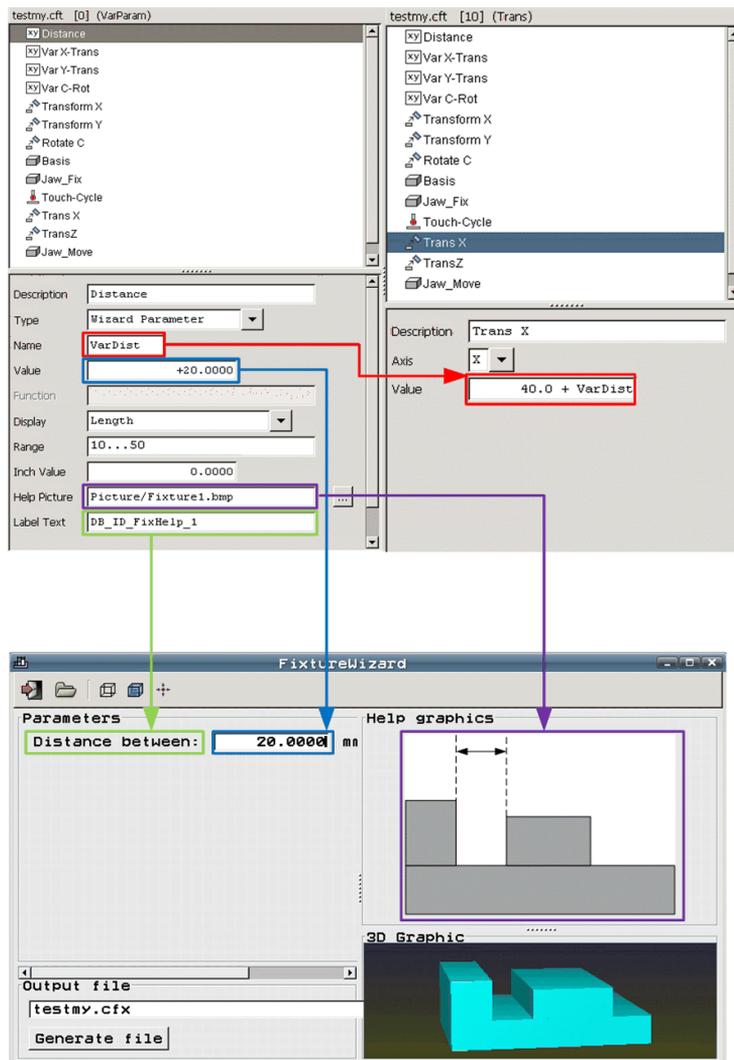


Entry	Input	Description
Value range (RANGE)	Numerical range	Permissible value range of a variable with manual input field. e.g. "1...10"; "100..."; "...5, 10...15, -20"
Value in inches (DEFINCH)	Numerical value	This is the default value for the variable when Inch is set. Only for linear input.
Help graphic (PICTURE)	Numerical value	Path to the help-graphic file for a variable or a variable subfixture.
Label text (LABELID)	Any text	Entry of a plain-language name or text ID. Is displayed for the operator as an additional aid when inserting the fixture. You can use the PLCtext software tool to create the texts as BMX texts. The texts must be located in a Text subdirectory of the respective fixture.

Variable of the type **Wizard parameter** as example:

In this example fixture the distance between the jaws is to be variable, and the operator must be able to enter this information in the FixtureWizard. Proceed as follows:

- ▶ Insert a variable of the type **Wizard parameter**, and enter a designation for the variable under Name. Under Value range, enter the permissible input range for the variable.
- ▶ Enter a default value for the variable in the **Value** field.
- ▶ You also have the option of entering a help graphic and a label text, which aids the operator when entering the parameters in the FixtureWizard.
- ▶ In this example the distance between jaws is to be variable. Therefore, the transformation that determines the position of the rear jaw must be linked to the variable. In this example, the result of $40.0 + \text{VarDist}$ indicates the position of the rear jaw. The fixed value 40 determines the position of the rear jaw when $\text{VarDist} = 0$ in the example.



Mathematical functions

The final value does not need to be entered directly for any input fields for numbers (e.g. position, transformation, function). These fields can also have formulas entered in them, in order to calculate the value. These formulas may again contain variables.

The C syntax with the usual precedence rules (multiplication/division before addition/subtraction) apply to the formulas. Only standard parentheses () are permitted for parenthetical expressions. The upper/lower case of the keywords listed below is unimportant.

Operation	Keyword / operator
Basic operations	+ - * /
Exponent	^
Constants	Fixed decimal point and floating point numbers with a period as decimal sign and exponent "e", e.g.: 1.234e5
Variables	<Name of the variable>
Trigonometric functions	sin, cos, tan, asin, acos, atan sinh, cosh, tanh, arsinh, arcosh, artanh (For trigonometric functions such as sin, the angle must be given in radians. Use the rad function for this.)
Conversion between degrees and radians	deg, rad
Logarithmic functions	log, ln, exp
Square root	sqrt
Absolute value	abs
Value as a whole number (integer)	int
Algebraic sign function -1 for $x < 0$ 0 for $x = 0$ 1 for $x > 0$	sign
Circle constant pi	pi

CMO

In a fixture, you can use additional CMOs. These collision monitored objects must exist as files.

Input possibilities for CMOs:

Entry	Input	Description
Description (DOC)	Any text	Object designation used in the fixture description.
File (FILE)	Path and file name	Path and file name of the CMO that is to be used at this spot in the fixture kinematics.
Wizard CMO (FCMO)	Active/Not active via check box	When the Wizard CMO check box is activated, a variable CMO is defined at this position. The selection of the CMO actually used occurs later in the FixtureWizard. The FixtureWizard then offers the operator a browser in which the path and file name of the actual CMO can be entered (such as for selection between step jaws and square jaws).
Help graphic (PICTURE)	Numerical value	Path to the help-graphic file for a variable or a variable subfixture.
Label text (LABELID)	Any text	Entry of a plain-language name or text ID. Is displayed for the operator as an additional aid when inserting the fixture. You can use the PLCtext software tool to create the texts as BMX texts. The texts must be located in a Text subdirectory of the respective fixture.



Test point

You use probing cycles of the type **Test point** to define probing points in the fixture kinematics, which are used after the fixture has been inserted in order to verify the position and orientation of the fixture. If the fixture kinematics contains test points, then once the fixture has been inserted and its position and orientation ascertained, the CREATE TEST PROGRAM soft key will be offered to the operator. It is used to create a test program that automatically probes all test points in order to check the position and orientation of the fixture.



Attention

HEIDENHAIN recommends assigning test points to all fixture kinematics. In this way, the operator can check the position and orientation of the fixture.

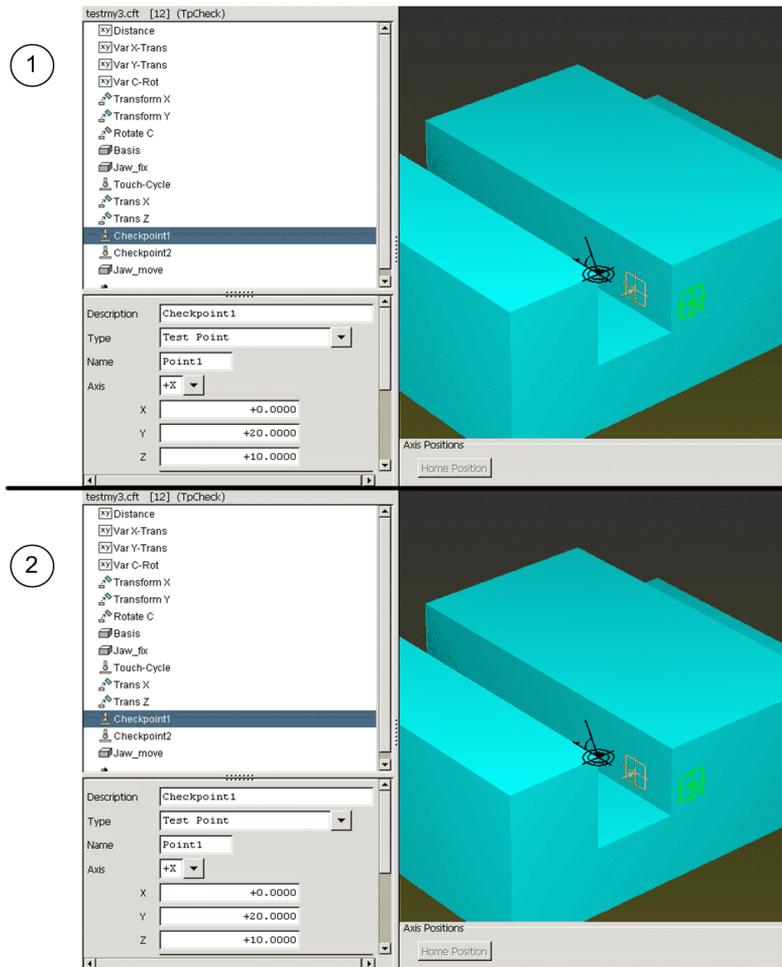
Input possibilities for variables:

Entry	Input	Description
Description (DOC)	Any text	Designation or name of the object, used in the fixture description.
Type (KEY)	■ Test point	Defines a number variable, whose value must be entered when inserting the workpiece via the control's user interface. The value can be changed by the operator without FixtureWizard and without ascertaining the position and orientation of the workpiece again (e.g., the distance between jaws changes, but the vise remains at the same position on the table).
Name (NAME)	Any text	Name for the probing cycle. The entry of a name is mandatory for the creation of a test program. It is used internally by the test program.
Axis (AXIS)	Axis identifier	Indication of the main axis for the probing cycle.
X, Y, Z	Position value	Entry of the position at which the fixture is to be probed.
Label text (LABELID)	Any text	For test points without function.

Probing cycle of the type **Test point** as example:

Two test points are to be inserted for the sample fixture, so that the operator can verify both the position and orientation of the fixture, as well as the entry of the distance between jaws. Proceed as follows:

- ▶ Insert a probing cycle of the type **Test point**, and enter a designation for the test point under Name.
- ▶ Enter the probing direction for the respective test point in the **Axis** field.
- ▶ Enter the position of the test point under **X, Y, Z**.



For the procedure for verifying the information of the fixture whose position and orientation were previously ascertained with the TNC, refer to Chapter 11.3 – "Fixture Monitoring" of the User's Manual for the iTNC 530.

The result of the test program's verification is displayed on the screen, and is available as a log file (*.txt). The log file is automatically given the same name as the test program, and is also saved to the same directory.

Automatic activation/deactivation of fixture situations

With the NC functions SEL FIXTURE and FIXTURE SELECTION RESET you can activate or deactivate previously saved fixture situations in automatic operation.

With SEL FIXTURE you can add a measured fixture to collision monitoring. The fixture must be archived in a .zip file for this purpose. The file name can be entered relative to **TNC:\system\Fixture\archiv** or as an absolute path. The file extension ".zip" does not have to be entered.

FIXTURE SELECTION RESET removes all fixtures from all insertion points of the collision monitoring.

A new pallet prototype **ToPa1PrFix.P** was introduced for pallet management. This prototype contains an additional **FIXTURE** column. In this column, the name of a fixture can be entered in **PAL**-type lines. Using the **FN18 SYSREAD ID510 NR7** function, the pallet macro can check whether a fixture has been entered in the pallet line and whether it can be activated. An existing fixture can be activated with the **FN17 SYSWRITE ID510 NR8** function.

Please note that the activation of a fixture is effective globally for the machine beyond the end of program execution.

8.6.3 Tool carrier kinematics and DCM

DCM collision monitoring (software option 40) of the iTNC 530 already protected tool carriers from collisions. The tool carriers to be monitored can be selected from a list in the same way as machine kinematics. If necessary, the operator can select the desired tool-carrier kinematics from the KINEMATIC column in the tool table TOOL.T.

The ASSIGN KINEMATICS soft key is available to the operator in the tool table for this. The list shows the operator all tool-carrier kinematics in *.TAB format that are in the **PLC:\Toolkinematics** directory, as well as all tool-carrier kinematics that are in *.CFX format from the **TNC:\system\Toolkinematics** directory

If the operator selects a *.CFX file from the TNC directory as tool-carrier kinematics for a tool, then when the TOOL CALL for this tool is performed, the file is converted to a *.TAB file in the background. This is then saved to the SYS partition and used for DCM collision monitoring.



Attention

If you change the carrier kinematics by editing the cfx file, you must re-assign the carrier kinematics description to a tool in the tool table, and execute a TOOL CALL. Not until the TOOL CALL does the TNC convert the cfx file into *.TAB format and then activate the corrected carrier kinematics description.

That chapter on collision monitoring describes the fundamentals of collision monitoring and the creation of collision-monitored objects (CMOs), see "DCM – monitoring the working space for collisions" on page 1424.



Attention

Finally, the tool-carrier model for collision monitoring and the behavior of the complete collision model must be tested on the machine as usual.

HEIDENHAIN makes tool-carrier kinematics for HEIDENHAIN touch probes available for downloading from its homepage. You will find the carrier kinematics descriptions under www.heidenhain.de > Services and Documentation > Software > Tool-carrier kinematics.



Listed below are the basic methods for using DCM to protect tool carriers from collisions:

■ **1. Modeling tool carriers with KinematicsDesign**

HEIDENHAIN provides its own carrier kinematics descriptions, which were created with the PC program KinematicsDesign, for downloading. You can create additional tool-carrier kinematics descriptions, and provide them to your end users. Templates for carrier kinematics descriptions have the file name extension *.cft

■ **2. Defining tool-carrier kinematics descriptions with the FixtureWizard**

With the FixtureWizard, the machine operator defines the exact dimensions of the tool carrier by entering parameter values in the template. The FixtureWizard is available as a stand-alone PC tool and as a component of the iTNC. It generates a tool carrier with concrete dimensions defined by you. Usable tool-carrier kinematics descriptions have the file name extension *.cfx.

For information on operating the FixtureWizard, refer to Chapter 11.3 – "Fixture Monitoring" of the User's Manual for the iTNC 530.

The most important functions are described below by using a simple example carrier. Use the tool-carrier templates from HEIDENHAIN as examples if you want to create more complex templates. However, be sure not to change the original HEIDENHAIN templates.

Prerequisites

- The software option 40 (DCM - Dynamic Collision Monitoring) must be enabled.
- Permissible insertion points for the tool carriers must be defined in the kinematics description. Do so in the KEY column of the description table via the entry TOOLFILE.

Files

The following files are relevant for tool-carrier kinematics.

■ **Template for a tool-carrier kinematics description – *.cft**

Variable template for a tool-carrier kinematics description. This template is created with KinematicsDesign. Tool-carrier templates from HEIDENHAIN are in the directory **PLC:\Toolkinematics**. Please save your own tool-carrier templates with names differing from those of the HEIDENHAIN templates. Use the FixtureWizard to edit the created *.cft file and to generate a *.cfx file from it.

■ **Individual tool-carrier kinematics – *.cfx**

Tool carrier that can be inserted directly in the machine kinematics. The *.cfx files created with FixtureWizard are saved in the same default directory, from which the *.cft file was opened.

■ **Kinematics tables for collision monitoring – *.tab**

If a TOOL CALL calls a tool for which a tool-carrier kinematics description is indicated in the tool table, then *.TAB files are automatically created during the TOOL CALL. These *.TAB files are then taken into account by DCM. Each *.CFX file is converted to a corresponding *.TAB file on the SYS partition. Therefore, all *.CFX files that are necessary for the description of a tool carrier must be on the control.

Tools

The following programs are involved in the creation of tool-carrier kinematics:

■ KinematicsDesign

KinematicsDesign is a program for interactive creation of control kinematics, fixture templates, completed fixtures and tool carrier kinematics. The program is available for download from the HEIDENHAIN homepage, and is available on the iTNC 530 control.

■ Fixture Wizard

With the FixtureWizard, you use the fixture templates and the templates for the tool carrier kinematics to generate executable kinematics files for the control. These kinematics files can be used for DCM collision monitoring of the control. The program is available for PCs as a part of KinematicsDesign. It is also available on the control.

Modeling tool carriers

KinematicsDesign helps you to model the tool carriers. You can define readily usable tool carriers (file extension *.cfx), or you can create variable tool-carrier templates (file extension *.cft).

For tool-carrier templates (*.cft) you define the shape of the tool carrier by using variables for different dimensions of the tool-carrier objects (CMOs). The tool carriers are designed in the same way as the machine kinematics: CMOs, translations and transformations are used. Input parameters permit variable design of the tool-carrier templates, and enable you to help the end user in the final definition of the tool carrier.

The *.cft files for tool-carrier templates are in a new table format in ASCII. The tables are an extension of the familiar columns of the kinematics tables *.tab of the machine kinematics, which were specially adapted for the modeling of fixtures and tool carriers. The tables can be opened with Microsoft Excel if the semicolon is selected as separator.

When saving tool-carrier kinematics to the **TNC:\system\Toolkinematics** directory, only give them names differing from the HEIDENHAIN templates, since the HEIDENHAIN templates will be overwritten during a software update. Tool carriers or tool-carrier templates that were created or changed by you should be saved in a separate directory or under their own names, in order to avoid losing data during a software update.



The following parameters or columns with the appropriate input options are available for modeling tool carriers:

Format of the CFT/CFX table:

Column	Input	Description
NR	0, 1, 2 ...	Automatic line numbering
KEY	■ CMO	Defines a monitorable collision body in the current kinematics sequence. This machine element is described by the file (table) referenced in the FILE column.
	■ Trans	Here a transformation of the indicated axis (AXIS column) in the current kinematics sequence by the value entered in the COORD column is performed. This can be a linear translation as well as a rotation about an axis.
	■ Cylinder	The following parameters for the X, Y, Z, AXIS, RADIUS , and HEIGHT columns define a cylindrical object, starting from the position of the previous translation, rotation or defined machine axis. Starting from this point, the cylinder is offset from X, Y and Z, and defined in the direction of the axis entered in AXIS , with radius (RADIUS) and height (HEIGHT).
	■ Cuboid	The following parameters for the X, Y and Z columns define a rectangular object, starting from the position of the previous translation, rotation or defined machine axis. Starting from this point, the cuboid is defined from a minimum point in X, Y and Z, to a maximum point in the next line in X, Y and Z, (as with the BLK definition in conversational programming).
Only for tool carriers (*.cft/*.cfx table):		
KEY	■ Var	Page 8 – 1483
	■ VarParam	Page 8 – 1483
AXIS	X, Y, Z, A, B, C ...	Entry of the axis designation for function given in the KEY column (valid for the MachAxis , Trans and TPXxx parameters in the KEY column).
RADIUS	E.g. 40 [mm]	Entry of the radius for "Cylinder"-type collision-monitored objects in the KEY column.
HEIGHT	E.g. 150 [mm]	Entry of the height for "Cylinder"-type collision-monitored objects in the KEY column.

Column	Input	Description
COORD	E.g. 47.092 [mm] or 45.05 [°]	Entry of the transformation value for linear axes (X, Y, Z, ...) in mm or for rotary axes (A, B, C) in degrees. The units are not entered. The iTNC infers the units from the axis designation entered (valid for the Trans parameter in the KEY column).
FILE	Path and file name ^a	Entry of the path and file name of a definition table for a collision-monitored object (valid for the CMO parameter in the KEY column).
DOC	Designator	Entry of a designator
X, Y, Z	Position value	Entry of a position shift in X, Y, Z for "Cylinder" or "Cuboid"-type objects under "Position" in KinematicsDesign.
XSIZE, YSIZE, ZSIZE	Dimensions	Dimensions of a cube within a tool-carrier kinematics description
NAME	Name	Name of a variable
VALUE	Value	Value of a variable, see page 1483
RANGE	Value range	Permissible value range of a variable with manual input field. e.g. "1...10"; "100..."; "...5, 10...15, -20"
SUBTYPE	Format	Format of a variable with manual input field, see page 1483.
DEFINCH	Value in inches	This is the default value for the variable when "Inch" is set. Only for linear input, see page 1483.
PICTURE	Help graphic	File with a help graphic for a variable, see page 1483
LABELID	Label text	Plain language or text ID for a variable, see page 1483.

- a. File paths in *.cft and *.cfx files must be entered relative to the active file.

Variables

For the flexible definition of tool carriers, KinematicsDesign also offers the possibility of creating variables for tool-carrier kinematics. These are used for variable length information, for example.

Input possibilities for variables:

Entry	Input	Description
Description (DOC)	Any text	Object designation that is used in the tool-carrier description.
Type (KEY)	■ Var	Variable of the type: Miscellaneous variable Defines a miscellaneous variable that can be used with the kinematics, e.g. for the buffering of values. Value assignment by the operator not possible.
	■ VarParam	Variable of the type: Wizard parameter Defines a number variable in which a value can be entered in the FixtureWizard (e.g. variable carrier height).
Name (NAME)	Any text	Name of the variable with which it can be addressed.
Value (VALUE)	Numerical value	Value of the variable as a number or a formula for calculation.
Format (SUBTYPE)	Position value	Format of a variable with manual input field (only for VarParam): Length: Linear value in [mm] or [inches] Angle: Angular value in degrees [°] Integer: Numeration type, only permits whole numbers Bool: Logical value, active/inactive through check box
Value range (RANGE)	Numerical range	Permissible value range of a variable with manual input field. e.g. "1...10"; "100..."; "...5, 10...15, -20"
Value in inches (DEFINCH)	Numerical value	This is the default value for the variable when Inch is set. Only for linear input.
Help graphic (PICTURE)	Numerical value	Path to a help graphic for a variable.
Label text (LABELID)	Any text	Entry of a plain-language name or text ID. You can use the PLCText software tool to create the texts as BMX texts. The texts must be located in a Text subdirectory of the respective fixture.

Mathematical functions

The final value does not need to be entered directly for any input fields for numbers (e.g. position, transformation). These fields can also have formulas entered in them, in order to calculate the value. These formulas may also contain variables.

The C syntax with the usual precedence rules (multiplication/division before addition/subtraction) apply to the formulas. Only standard parentheses () are permitted for parenthetical expressions. The upper/lower case of the keywords listed below is unimportant.

Operation	Keyword / operator
Basic operations	+ - * /
Exponent	^
Constants	Fixed decimal point and floating point numbers with a period as decimal sign and exponent "e", e.g.: 1.234e5
Variables	<Name of the variable>
Trigonometric functions	sin, cos, tan, asin, acos, atan sinh, cosh, tanh, arsinh, arcosh, artanh (For trigonometric functions such as sin, the angle must be given in radians. Use the rad function for this.)
Conversion between degrees and radians	deg, rad
Logarithmic functions	log, ln, exp
Square root	sqrt
Absolute value	abs
Value as a whole number (integer)	int
Algebraic sign function -1 for $x < 0$ 0 for $x = 0$ 1 for $x > 0$	sign
Circle constant pi	pi



CMO

In the tool-carrier kinematics, you need to define additional CMOs to describe the collision-monitored objects. These collision monitored objects must exist as files. The files for CMOs of the tool-carrier kinematics may also be in *.CFT format. Upon a TOOL CALL, the control automatically converts these files of the respective tool to the necessary *.TAB format.

In the definition of a CMO, shifts and rotations of subsequent collision objects can be entered via the keyword TRANS. This functions like entering the description of a kinematics configuration. However, the coordinate transformations in a CMO definition file are only effective within the respective object-description file.

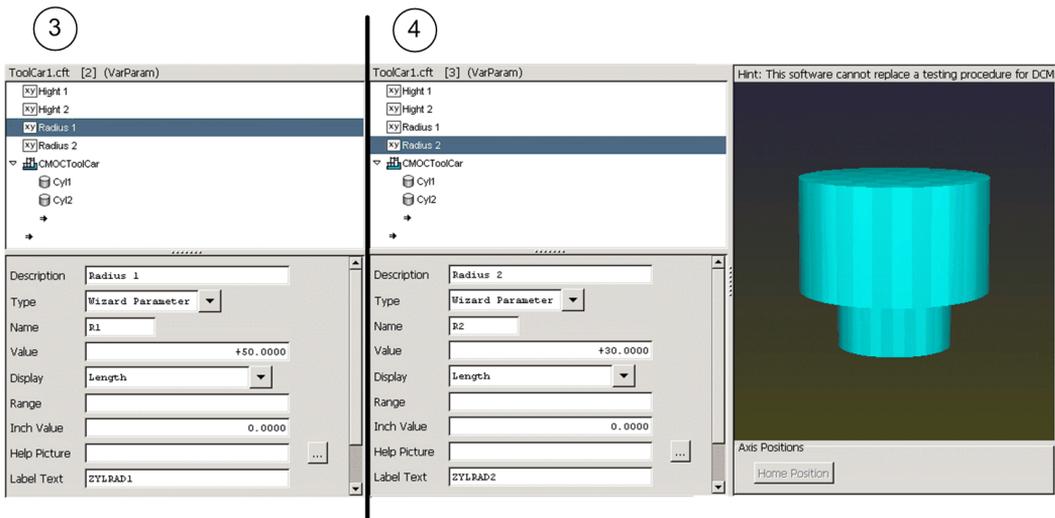
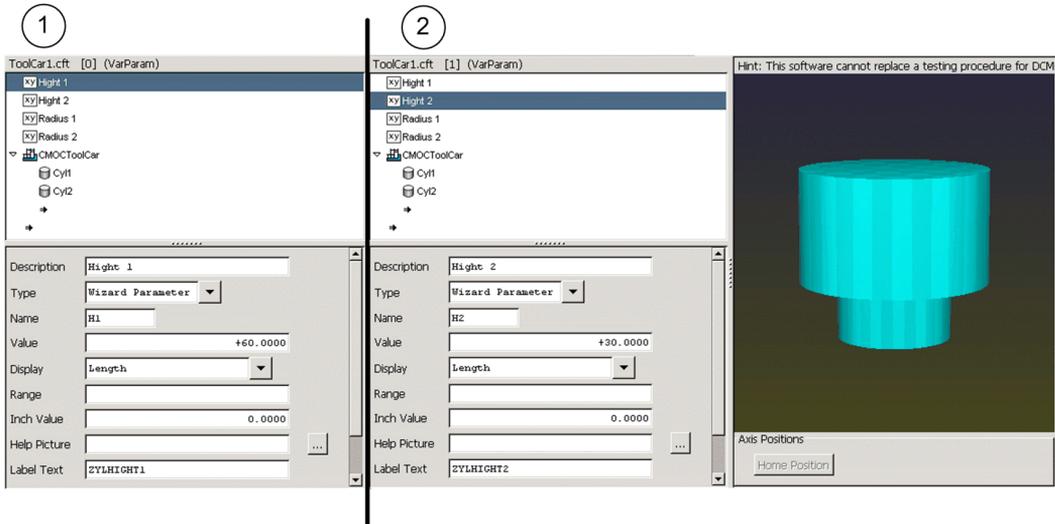
Input possibilities for CMOs:

Entry	Input	Description
Description (DOC)	Any text	Object designation that is used in the tool-carrier description.
File (FILE)	Path and file name	Path and file name of the CMO that is to be used at this spot in the tool-carrier kinematics.
Help graphic (PICTURE)	Numerical value	Path to the help-graphic file for a variable or a variable tool-carrier part.
Label text (LABELID)	Any text	Entry of a plain-language name or text ID. You can use the PLCtext software tool to create the texts as BMX texts. The texts must be located in a Text subdirectory of the respective fixture.

Example

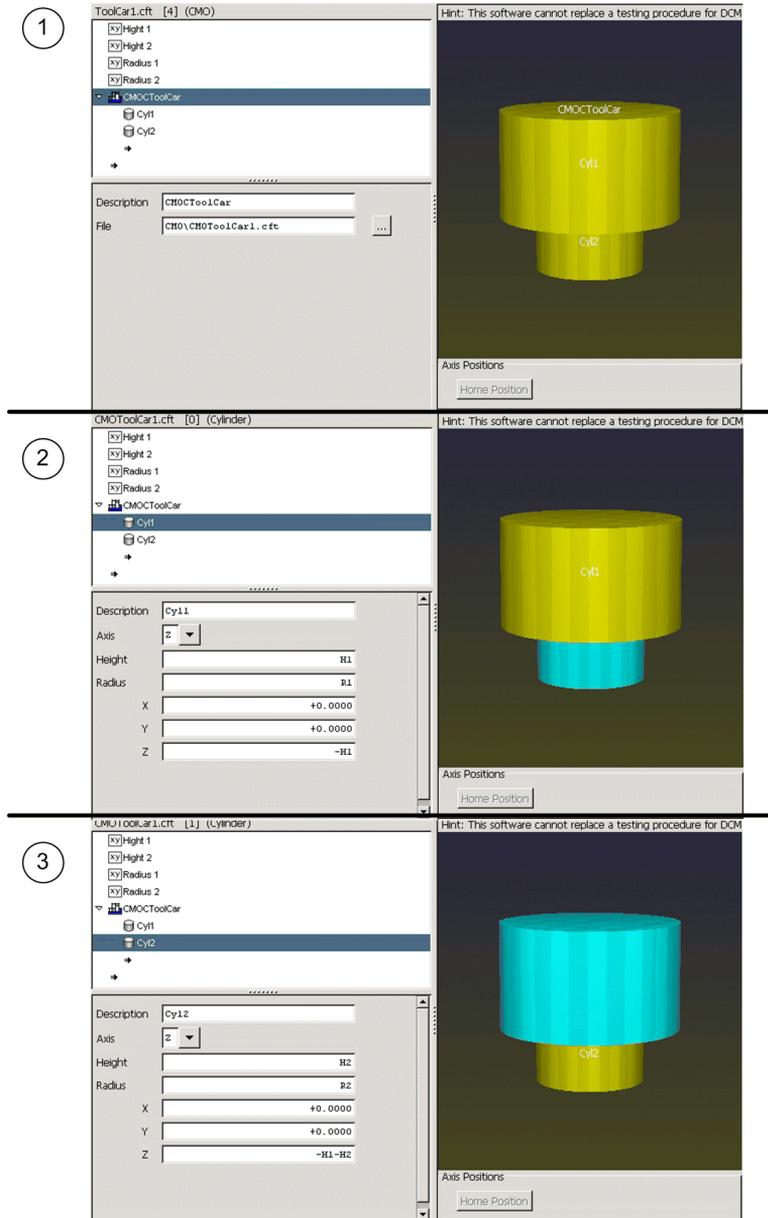
Here is an example of how a simple tool-carrier kinematics description (*.cft template) is created with KinematicsDesign:

- ▶ Start the KinematicsDesign PC tool.
- ▶ Under **File / New...**, select **Tool-Carrier Template**.
- ▶ Assign any file name (the extension .cft is added automatically).
- ▶ You can then create your variable **tool-carrier template** or a **tool-carrier kinematics description**. In the example, first four variables are defined for the height and radius of the two cylinders. This means that the appearance of the tool carriers is variable. The height and radius of the two cylinders will later be specified with the FixtureWizard.



- In the example, a CMO is inserted after the variables have been defined. This CMO describes the appearance of the tool carrier, and is specified in a separate file (here CMOToolCar.cft). The cylinders contain the variables for the height and radius, which will later be specified with the FixtureWizard.





- ▶ However, the variable tool carrier *.cft cannot yet be used for collision monitoring. You must open the *.cft file with the Fixture Wizard, specify values for all variables, and then save it as a *.cfx file. You can do this on the PC or the control.

This *.cfx file can then be used as a tool-carrier kinematics description.

The advantage of such a variable template is that you specify the basic shape. The operator can then use the FixtureWizard to define a multitude of tool carriers whose basic shape is the same. Only dimensions that were created as variables can be changed.

8.6.4 KinematicsDesign

The KinematicsDesign PC tool is also available on the control. The KinematicsDesign version on the control supports you in optimizing, adjusting and changing the kinematics of your machine. The working space and the collision objects of the active kinematics can be displayed and edited. The version in the control contains almost all the functions of the PC version.

The command "File – Save as" was removed, because the active kinematics for editing is automatically saved when the KinematicsDesign version on the control is used.

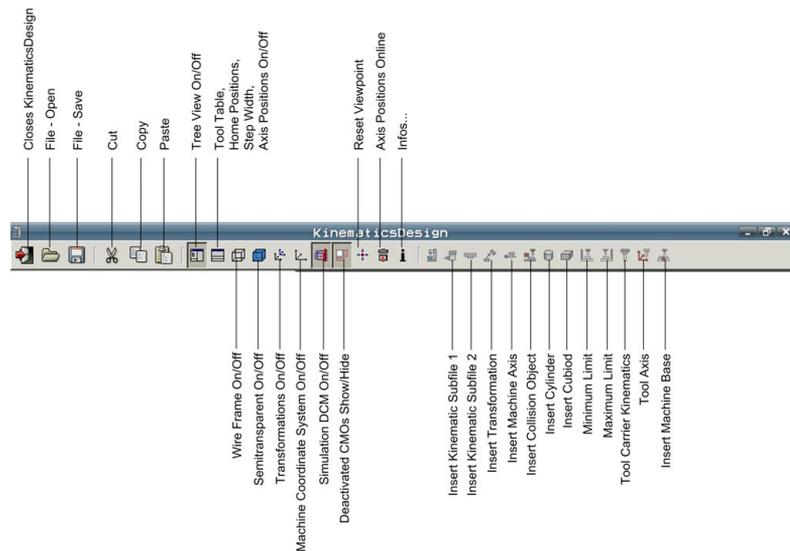
All other functions of the PC version that you find under "Edit," "View," "Paste," etc. have been implemented in the control version by using appropriate icons.

HEIDENHAIN recommends using a mouse for operating KinematicsDesign.

The documentation of the KinematicsDesign PC tool is stored in the **PLC:\tncguide** directory on the control. The operation of the KinematicsDesign PC tool described in the documentation substantially corresponds to that of the KinematicsDesign version stored in the control. To start the tool, switch to the PLC mode of operation. Press the **ADVANCED SETUP** soft key, and then the **KINEMATICS** soft key.

To switch between KinematicsDesign and the screen of the control, press the screen switchover key.

The corresponding icons provide you with the following functions:



Danger

- When you use the KinematicsDesign version on your control, you edit and change the active kinematics of your machine. The changes made immediately become effective in the active kinematics.
- The KinematicsDesign software is not a replacement for testing collision monitoring on the machine!



Requirements on the control:

- If no valid kinematics is active on the control, or if the entry **KINEMATIC =** is missing from the OEM.SYS, the error message **Kinematics table missing** appears, and KinematicsDesign is not started.
- If you have created the kinematics description only with the KinematicsDesign version on the control and without the KinematicsDesign PC tool, then no .ini file (KineDesign.ini) of KinematicsDesign is available in the control.

In this case, it may be necessary to create the file manually. If the machine configuration does not differ from the standard configuration (positive Z axis points up, and the positive X axis to the right) assumed by KinematicsDesign, you do not need to create the KineDesign.ini file. Also, if the kinematics description was transferred to the control by the KinematicsDesign PC tool, the .ini file is created automatically in the control and does not need to be created manually.

However, if the configuration of your machine differs from the assumed standard configuration, you must manually create the KineDesign.ini file on the control. The file must be stored in the same directory as the KINELIST.tab file. The KineDesign.ini file must be an ASCII file with the two following entries:

UpwardAxis =
RightwardAxis =

Example of KineDesign.ini:

UpwardAxis = -Z
RightwardAxis = -X

8.7 Pallet management

Configuring a pallet table

The pallet table is a "freely definable table":

- ▶ Define the prototype in the directory PLC:\PROTO with the file name extension .P.
- ▶ Create the prototype.
(See "Freely definable tables" on page 8 – 1409)

If you have more than one prototype with the file name extension .P, a menu for format selection will appear when you create a pallet table. Your PLC program must be adapted to the various formats.

The COPY SAMPLE FILES soft key copies prototypes for the pallet tables into the directory PLC:\PROTO.

The appropriate prototypes are offered when you create a new pallet table. If you do not want this to happen, delete unnecessary prototypes from the PLC:\PROTO directory. Then only existing prototypes are displayed.

- **PROTOTYP.P** = Standard prototype (**PAL/PGM, NAME, DATUM, X, Y, Z**)
- **PROTO_TO.P** = Prototype for tool-oriented machining
- **PROTOPR.P** = Standard prototype for preset tables
- **PRO_TOPR.P** = Prototype for tool-oriented machining with preset tables



Field names

The following types of fields are used in the pallet table:

- Mandatory fields: Values must be entered.
- Optional fields: Values can be entered. They have a fixed, predefined meaning for the NC.
- Freely definable fields: You can display additional fields. The names and meaning are defined as desired. The entries are for information, or you can interrogate and change them through the PLC.

Name	Type of machining	Meaning
PAL/PGM	Workpiece-oriented/tool-oriented	Mandatory field: Definition of the entry <ul style="list-style-type: none"> ■ PAL = Pallet ■ PGM = NC program ■ FIX = Fixture (only tool-oriented)
W-STATUS	Tool-oriented	Optional field: Machining status <ul style="list-style-type: none"> ■ BLANK = Workpiece blank ■ ENDED = Machining complete ■ INCOMPLETE = Machining not complete
METHOD	Tool-oriented	Mandatory field: Type of machining <ul style="list-style-type: none"> ■ TO = Tool-oriented ■ WPO = Workpiece-oriented ■ CTO = Tool-oriented for several entries
NAME	Workpiece-oriented/tool-oriented	Mandatory field: Name of the pallet or the NC program. NC program names without paths are searched for in the directory with the pallet file. Permit only decimal numbers, so that you can interrogate the pallet's name in the change macro with FN18: SYSREAD .
DATUM	Workpiece-oriented/tool-oriented	Optional field: Name of the datum table. Datum tables without paths are searched for in the directory with the pallet tables.
X, Y, Z, U, V, W, A, B, C	Workpiece-oriented/tool-oriented	Optional fields: Definition of the datum ^a In standard format only the columns X, Y and Z are used.
PRESET	Workpiece-oriented/tool-oriented	Optional field: Definition of the preset by entering a number from the preset table ^a
SP-X, SP-Y, SP-Z	Tool-oriented	Optional fields: Safe positions; with FN18: SYSREAD , these positions can be read in NC macros.

Name	Type of machining	Meaning
CTID	Tool-oriented	If—due to a tool change—an NC program must be stopped during tool-oriented machining, the iTNC enters a code. This code enables the iTNC to resume the machining process at the position where it has been stopped.
LOCATION	Workpiece-oriented/tool-oriented	Optional field (not used in standard format): Location of pallet. If the LOCATION column is used, an NC program can be run only if this column contains the entry MA (= pallet for the machine) for this pallet.
LOCK	Workpiece-oriented/tool-oriented	Optional field: Lines containing any entry in this column will not be run. If more than one program or pallet is to be run, the next permitted line is used. Unlocked lines in a locked pallet are also skipped.
PALPRES	Workpiece-oriented/tool-oriented	Optional field: Line number to be activated in the active pallet preset table.
Any names	Workpiece-oriented/tool-oriented	Freely definable

- a. For pallet entries (PAL), the values refer to the end of the kinematics chain (usually the machine datum MP960.x). In NC programs the values refer to the pallet datum if a PALPRES is active for the respective pallet.

Basically it does not make a difference whether a PRESET is called from a PAL line or a PGM line. In both cases, the PRESET refers to the end of the kinematics (usually table center point = machine datum). If an active pallet preset table exists in addition, however, the two are added (preset from pallet preset table for PAL + preset from preset table for PGM/PAL).

If a pallet preset is activated in a PAL line, any subsequent PRESET (from the PGM line or not) is based on it.

Executing a pallet table

The pallet table is selected and started with PGM MGT like a normal part program in the **Program Run, Single Block** and **Program Run, Full Sequence** operating modes.

Pallet entries (**PAL**) and fixture entries (**FIX**) result in a call for an NC macro. Program entries (**PGM**) are run like a PGM CALL.

- ▶ With MP7683 bit 6, specify if the NC program and the pallet table should appear simultaneously in the split screen, or if the active NC program or active pallet table should be shown individually.
- ▶ With MP7683 bits 0 to 2 and bit 8, specify the operating sequence following an NC start.
- ▶ With MP7683 bit 3, specify the operating sequence upon reaching the end of the pallet table.
- ▶ Determine with MP7683 bit 4 whether the current pallet table should be editable with the EDIT PALLET soft key in the **Program Run, Single Block** and **Program Run, Full Sequence** operating modes.
- ▶ Bit 9 of MP7683 allows you to select whether the EDIT PALLET soft key for editing the pallet table *.P is displayed in the Program Run modes of operation.

As soon as a pallet table is selected, M4160 is set.

Through the PLC you can graphically display the tool changer status in the PLC window and enable the user to control the tool changer through PLC soft keys.

You can provide the user with excerpts from the pallet table for editing (see "Freely definable tables" on page 1409).

With Module 9035 you can interrogate the active line of the pallet file, and with Modules 9090 or 9281 you select a certain line in the pallet table. Unlike Module 9090, a datum shift or datum setting can be executed immediately with Module 9281.

Example:

NR	PAL/PGM	W-STATUS	METHOD	NAME	DATUM	X	Y	Z	SP-X	SP-Y	SP-Z	CTID
0	PAL			120		0	0	0			150	
1	FIX										150	
2	PGM	BLANK	WPO	PART1.H								
3	PGM	BLANK	TO	PART2.H		120	120	0				
4	PAL			130	NULL1.D	0	10	15			150	
5	PGM	BLANK	TO	PART3.H		100	100	100				
6	PGM	BLANK	CTO	PART3B.H								
[END]												

Line 0:

The pallet with the name **120** is defined. The NC macro for changing the pallet is activated. The active datum equals the machine datum. A clearance height was programmed.

Line 1:

A fixture is defined and a clearance height is specified. The NC macro for changing the fixture is active.

Line 2:

The fixture holds an unmachined part to be machined with NC program **PART1.H** (workpiece-oriented machining).

Line 3:

The fixture holds a second unmachined part to be machined with NC program **PART2.H** (tool-oriented machining).

The active datum is offset from the pallet datum by the given values.

Line 4:

The pallet with the name **130** is defined. The NC macro for changing the pallet is activated. The active datum is offset from the machine datum by the given values. The datum table **NULL1.D** is active.

Lines 5 and 6:

The pallet holds two unmachined parts which are to be machined together in one setup with NC programs **PART3.H** and **PART3B.H** (tool-oriented machining). The active datum of the first part is offset from the pallet datum by the given values.

PLC Marker M4196 informs the PLC program if pallet machining was stopped because there are no more valid jobs according to the identifiers in the pallet table ("W status" column set on ENDED). For infinite machining (bits 2 and 3 of MP7683 set) the marker means that no more jobs are available. It is possible that the marker is only set several cycles after the deletion of M4176. The marker is not reset by the NC but must be reset by the PLC program when appropriate. Depending on the contents of the bits 0, 1, 2, 3 and 8 in MP7683, there may still be jobs that have not yet been processed, even if the marker is set.



Pallet preset table

A preset table for managing pallet datums can be activated in addition to the preset table for managing workpiece datums. This makes it possible now to manage the pallet datums independently of the workpiece datums.

Pallet datums are an easy way to compensate mechanical differences between individual pallets.

To activate the pallet preset table, the token **PALLETPRESET = <path/file name>** must be entered in OEM.SYS. Any name can be assigned to the pallet preset file *.PR, e.g. PALLETPRESET = TNC:\Palletpreset.pr. Only one workpiece datum and one pallet datum can be active at the same time. Both datums are effective in sum.

The NC function FN17: SYSWRITE ID540 NR1 is also introduced for activating a pallet datum. FN 18:SYSREAD ID 540 NR1 is used to read the active line number. In addition, a pallet datum can be activated by entry of a line number of the active pallet preset table in the **PALPRES** column of the pallet table.

The iTNC displays the number of the active pallet preset in the additional status display.

For determining the pallet datums, there is an additional ENTRY IN PALLET PRES. TAB soft key in the manual probing functions with which you can also store the probing results in the pallet preset table.

Markers and machine parameters

Settings possibilities for markers and machine parameters

		Set	Reset
M4160	Pallet table selected	NC	NC
M4196	Pallet machining stopped for lack of a valid job	NC	PLC

MP7683 Executing pallet tables and NC programs

Format: %xxxxxxxxx

Input:

Bit 0 – PROGRAM RUN, SINGLE BLOCK operating mode:
 0: During the start, a line of the NC program is run. The pallet change macro is executed completely.
 1: During the start, a complete NC program is run.

Bit 1 – PROGRAM RUN, FULL SEQUENCE operating mode:
 0: NC start executes a complete NC program.
 1: At the start all NC programs are executed up to next pallet.

Bit 2 – PROGRAM RUN, FULL SEQUENCE operating mode:
 0: As defined in bit 1
 1: All NC programs and pallets up to the end of the table are executed.

Bit 3 – When the end of the table is reached, the process begins again with the first line.
 0: Function is not in effect
 1: Function is effective (bit 2 = 1)

Bit 4 – Editing the active pallet table
 0: Active pallet table cannot be edited.
 1: The active pallet can be edited in the PROGRAM RUN, FULL SEQUENCE and PROGRAM RUN, SINGLE BLOCK modes.

Bit 6 – Display of pallet table and NC program
 0: Both simultaneously in a split screen
 1: Pallet table or NC program individually

Bit 7 – AUTOSTART function
 0: AUTOSTART by the NC
 1: AUTOSTART by the PLC

Bit 8 – Procedure for tool-oriented machining in the **Program Run** operating modes
 0: NC start machines all workpieces on the pallet until the next tool change
 1: NC start executes all NC programs until the end of the pallet

Bit 9 – Editing of pallet tables
 0: EDIT PALLET soft key is not displayed
 1: EDIT PALLET soft key is displayed



Module 9090 Select a line in the pallet table

With this module, you set the cursor on a particular line of the pallet table that you selected in the program run mode. If the iTNC is in another mode, the selection will be made when the control returns to the **Program Run, Single Block** or **Program Run, Full Sequence** operating mode.

The selection is possible only as long as no pallet file has been started.

Call only in a submit job or spawn job.

Call:

PS B/W/D/K <Line number in the pallet table>

CM 9090

PL B/W/D <Error code>

0: No error, line was selected

1: Module was not called in a spawn or submit job

2: Call during a running NC program

3: No pallet table selected in full sequence

4: Line does not exist

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Line does not exist
	20	Module was not called in a spawn or submit job
	21	Call during a running NC program
	56	No pallet table selected in Full Sequence

Module 9281 Select a line in the pallet table

With this module you set the cursor on a particular line of the pallet table that you selected in the **Program Run, Single Block** or **Program Run, Full Sequence** operating mode. Datum shift and datum setting can be run immediately. If the iTNC is in another mode, the selection will be made when the control returns to the **Program Run, Single Block** or **Program Run, Full Sequence** operating mode.

Call:

PS B/W/D/K <Line number in the pallet table>

PS B/W/D/K <Mode>

Bit 0 –

0: Do not run datum shift or set the datum

1: Run the datum shift/set the datum immediately

Bit 1 – Shift the datum/set the datum

0: Do not run if the line is disabled by an entry in the LOCK column

1: Run even if the line is disabled by an entry in the LOCK column

CM 9281

PL B/W/D <Error code>

0: No error, line was selected

1: Module was not called in a spawn or submit job

2: Call during running program

3: No pallet table selected in full sequence

4: Line does not exist

5: Called in smarT.NC or error during datum setting, in the datum table or pallet table

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Line does not exist
	20	Module was not called in a spawn or submit job
	21	Call during a running NC program
	36	Called in smarT.NC or error during datum setting, in the datum table or pallet table
	56	No pallet table selected in Full Sequence

NC macro for changing the tool during tool-oriented machining

A special tool-change macro is required for tool-oriented pallet machining. This is defined through the keyword **TCTOOLMODE=** in NCMACRO.SYS.

This specific NC macro is called for tool oriented machining instead of the standard tool-change macro.

The macro must perform the following functions:

- Positioning to clearance height
- Execution of M146
- Tool change through **TOOL CALL**. The standard tool-change macro is called.

In the NC macro you can use **FN18: SYSREAD Qxxx = ID510 NR5** or **NR6 IDX<Axis>** to find whether a clearance height has been programmed for an axis, and if so, the value specified for the clearance height.

With the M function M146 the current geometry information is saved in a temporary file. This information is required for continuing NC program run after an interruption due to a **TOOL CALL** during tool-oriented machining. In addition, a code is entered in the **CTID** column and the entry in **W-STATE** is changed to **INCOMPLETE**, if required.

A simple example of an NC macro for tool changing with tool-oriented machining:

```
0   BEGIN PGM T0change MM
1   L Z-32 R0 FMAX M91
2   FN 18: SYSREAD Q1 = ID60 NR1
3   FN 18: SYSREAD Q2 = ID60 NR2
4   FN 18: SYSREAD Q3 = ID60 NR3
5   FN 18: SYSREAD Q4 = ID60 NR4
6   FN 18: SYSREAD Q5 = ID60 NR5
7   FN 18: SYSREAD Q7 = ID60 NR7
7   FN 18: SYSREAD Q8 = ID60 NR8
8   M146
9   TOOL CALL Q1 Z SQ3 DL+Q4 DR+Q5 DR2:+Q7
10  END PGM T0change MM
```

NC macro for changing pallets and fixtures

- ▶ In NCMACRO.SYS, use the entry **PALETT=** to define the complete path and name of the NC macro that is to be called when a pallet entry (**PAL**) is run.
- ▶ In NCMACRO.SYS, use the entry **CLAMP=** to define the complete path and name of the NC macro that is to be called when a fixture entry (**FIX**) is run.

In these macros you can request the current line or pallet name with **FN18: SYSREAD Qxxx = ID510 NR1** or **NR2**, respectively.

This NC macro also can be started from the PLC with Module 9280. To synchronize the current machine status and the look-ahead calculation with an NC macro call, see "NCMACRO.SYS" on page 1709.

A simple example of an NC macro for changing fixtures:

```
0      BEGIN PGM TOclamp MM
1      FN 18: SYSREAD Q1 = ID510 NR5 IDX1
2      FN 18: SYSREAD Q2 = ID510 NR5 IDX2
3      FN 18: SYSREAD Q3 = ID510 NR5 IDX3
4      FN 18: SYSREAD Q4 = ID510 NR5 IDX4
5      FN 18: SYSREAD Q5 = ID510 NR5 IDX5
6      FN 9: IF +Q3 EQU +0 GOTO LBL 1
7      FN 18: SYSREAD Q13 = ID510 NR6 IDX3
8      L Z+Q13 RO FMAX
9      LBL 1
10     FN 9: IF +Q1 EQU +0 GOTO LBL 2
11     FN 18: SYSREAD Q11 = ID510 NR6 IDX1
12     L X+Q11 RO FMAX
13     LBL 2
14     FN 9: IF +Q2 EQU +0 GOTO LBL 3
15     FN 18: SYSREAD Q12 = ID510 NR6 IDX2
16     L Y+Q12 RO FMAX
17     LBL 3
18     FN 9: IF +Q4 EQU +0 GOTO LBL 4
19     FN 18: SYSREAD Q14 = ID510 NR6 IDX4
20     L C+Q14 RO FMAX
21     LBL 4
22     FN 9: IF +Q5 EQU +0 GOTO LBL 5
23     FN 18: SYSREAD Q15 = ID510 NR6 IDX5
24     L B+Q15 RO FMAX
25     LBL 5
26     END PGM TOclamp MM
```

If such an NC macro is canceled (e.g. **PALLET =** with an internal stop), Marker M4185 is set and Module 9320 provides the same status information as in the event of an NC program cancelation. Pallet and fixture macros that are canceled with an external/internal stop behave just like NC programs regarding the setting of M4185 and the status data of Module 9320. In connection with MACROCANCEL they are regarded as NC macros, however, see page 1709.



Module 9280 Start the NC macro (Run pallet entry)

The NC macro must be defined in NCMACRO.SYS with the entry **PALETT =.** It can only be activated if the control is in the **Program Run, Single Block** or **Program Run, Full Sequence** operating mode, a pallet table is selected, and no macro or NC program is running.

Call:

PS B/W/D/K <Pallet number>

PS B/W/D/K <Line number>

CM 9280

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	7	The file entered in the entry PALETT= does not exist.
	8	Control is not in the SINGLE BLOCK or FULL SEQUENCE mode
	20	Module was not called in a spawn or submit job
	28	An NC program or NC macro is running.
	29	Selected file is invalid or does not exist
	30	There is no PALETT= entry in the NCMACRO.SYS file
	36	NCMACRO.SYS does not exist

NC macro at the end of an NC program

- ▶ In NCMACRO.SYS, use the entry **PALEPILOG=** to define the complete path and name of the NC macro that is to be called at the end of an NC program that was started from the pallet table.

To synchronize the current machine status and the look-ahead calculation with an NC macro call, see "NCMACRO.SYS" on page 1709.

8.8 Electronic handwheel

The following handwheels can be connected with HEIDENHAIN controls:

- One panel-mounted HR 130 handwheel, or
- Three HR 150 panel-mounted handwheels via the HRA 110 handwheel adapter, or
- Up to two HR 410 portable handwheels, or
- Up to two HR 520/HR 420 portable handwheels with display, or
- One HR 550 portable wireless handwheel with display

For information on the operation of the electronic handwheel, see the User's Manual.

- ▶ In MP7640, enter the type of handwheel connected to the control. If you enter a value greater than zero and no handwheel is connected, the error message **HANDWHEEL?** appears.
- ▶ If you use more than one handwheel together with the HRA 110 handwheel adapter, enter in MP7650 for each axis the counting direction of the individual handwheels. If you use only one handwheel, enter the counting direction in bit 0.

Shock or vibrations can cause a slight motion at the handwheel and produce an unintentional axis movement. In this case:

- ▶ Enter a threshold sensitivity in MP7660.

With W1062/W1176 you can disable the handwheel pulses for specific axes, if more than one handwheel is used in connection with the HRA 110 handwheel adapter. If the marker M4576 is set, all axes are disabled. If it is reset, W1062/W1176 applies. If only one handwheel is used, a selectable axis can be disabled with W1062/W1176.

The subdivision factor specifies the traverse per handwheel revolution:

- ▶ For the HR 130 or HRA 110, choose the **HANDWHEEL** operating mode and enter a subdivision factor for each handwheel. To ensure that the rapid traverse rates specified in MP1010.x are not exceeded, the smallest possible input step is preset by the control.
- ▶ For the HR 130 or HR 110, with MP7641 you specify whether the subdivision factor is entered directly through the TNC keyboard or via PLC Module 9036.

Subdivision factor	Traverse distance per revolution [mm]	Effective beginning from rapid traverse: MP1010.x [mm/min]
0	20	12 000
1	10	6000
2	5	3000
3	2.5	1500
4	1.25	750
5	0.625	80



Subdivision factor	Traverse distance per revolution [mm]	Effective beginning from rapid traverse: MP1010.x [mm/min]
6	0.312	80
7	0.156	80
8	0.078	80
9	0.039	80
10	0.019	80

MP7674.x can always be used to set a minimum axis-specific subdivision factor for the HR 130, HR 330, and HR 332. It can be used for the HR 410 if MP7641 bit 1 is not set (without detent encoder).

You can choose a larger input step for the traverse distance per rotation than that calculated by the NC:

- ▶ In MP7670.x, enter a subdivision factor.
- ▶ In MP7645.x, enter an initialization parameter for the handwheel.

The two machine parameters are evaluated by the HRA 110 or HR 410, but not by the HR 420 or HR 5xx.

MP7675.x has no meaning for the HR 130, HR 330, or HR 332. If MP7641 bit 1 is set (with detent encoder) for the HR 4xx or HR 5xx, then a maximum distance in [mm] per detent can be entered for each axis. If MP7641 bit 1 is not set (without detent encoder) for the HR 420 or HR 5xx, then a maximum distance in [mm] per handwheel revolution can be entered for each axis. These limitations serve to prevent the movement monitoring from becoming active because of abrupt motions when a large gear ratio is in effect on handwheels for axes with poor dynamics.

The VT axis (Virtual Tool Axis) can now be activated for handwheel superimposition in the active tool-axis direction by setting bit 4 of MP7641.

If bit 5 of MP7641 is set, the keys of the HR 420 or HR 5xx (PLC markers 4661 to 4668) are reported to the PLC even if the HR is not active. This enables the PLC to make the keys of the HR 420 or HR 5xx (e.g. NC stop, spindle stop) become effective even if the handwheel is not selected.

Bit 6 of MP7641 is used to enable traversing of auxiliary axes with the HR 420 or HR 5xx. This makes it possible to select axes that are indicated with lowercase letters in MP100 in the operator menu of the HR 420 or HR 5xx handwheel, and to move them with the handwheel.

With the standard settings, the vertical soft keys (PLC soft keys) cannot be used on the machine side while the HR 420 or HR 5xx handwheel is active. These soft keys can only be used if bit 9 is set in MP7641.



Danger

MP7641 bit 9 may only be set if the vertical soft keys (PLC soft keys) do not have any functions that could pose danger to the operator (e.g. starting an axis movement or a spindle movement).

Two handwheels on the TNC

As of software version 60642x-03, two handwheels can be connected and used alternately in the HSCI system.

One handwheel is connected to X23 of the machine operating panel. It has the index .0 in the handwheel MPs. The second handwheel must be connected to X29 of the MC main computer unit. It has the index .4 in the handwheel MPs. Machine parameters MP7640 (type of handwheel) and MP7650 (counting direction) were therefore expanded to 5 indices. MP7642 (handwheel name) was added, as well as MP7646 to MP7649, which—just like MP7645—contain handwheel-specific parameters for the other handwheels. In future software versions the number of possible handwheels is planned to be increased further. The corresponding indices of the mentioned machine parameters have already been reserved for this purpose. MP7641 for the handwheel settings is globally effective for all handwheels in the system.

Please note that at present only the handwheel connected to X23 of the last machine operating panel in the HSCI chain is supported if there is more than one machine operating panel in the HSCI system. A machine operating panel at X501 of the MC main computer unit is automatically always the last machine operating panel in the HSCI system.

As before, the permissive buttons of all handwheels must be evaluated by suitable external devices. No more than one permissive device may be active on the machine at any one time. If you are using a control system with integrated functional safety (FS), FS inputs can be used for the evaluation of the permissive buttons. As before, the emergency stop buttons of all handwheels must be evaluated by suitable external devices. All emergency-stop devices of the machine must function reliably at all times. Under no circumstances are these devices to be deactivated. If you are using a control system with integrated functional safety (FS), the emergency-stop inputs (–ES.A, –ES.B) should be used for the evaluation of the emergency-stop buttons. Several emergency-stop buttons can be connected in series for this purpose.

If only handwheels with display (HR 420, HR 520, HR 550) are involved, the active handwheel is defined solely through the selection key on the handwheel. If there is exactly one HR 110 panel-mounted handwheel in the system, it becomes active automatically if control is not assigned to the display handwheel.

In all other cases, the handwheel must be activated via the PLC program if more than one handwheel is involved. The active handwheel can be defined with PLC Module 9036 and the new selection parameters 20 (active handwheel) and 21 (standard handwheel). Number 70 (active handwheel) and number 71 (standard handwheel) of the PLC Module 9035 can be used to read back the currently active handwheel.

The PLC interface for the handwheels will not be expanded. The currently active handwheel always accesses the markers and double words of the PLC that have already been used before (e.g. M466x for the HR keys).



Further constraints:

- If there is only one handwheel in the HSCI system, this is the standard handwheel.
- If one of the display handwheels is active, the other handwheels are disabled, and this is indicated by a corresponding dialog in the display.
- If there is more than one display handwheel in the system, control can be handed over only if override control is also withdrawn from this handwheel.
- If there are several handwheels in the system and one of them is an HR 110, the panel-mounted handwheel is automatically the standard handwheel.
- If there is more than one handwheel without display (HR 410 and earlier versions), this type of handwheel must be activated via the PLC program (e.g. through evaluation of the permissive buttons). PLC Module 9036 was expanded accordingly.
- If the PLC program wants to prevent an HR 420 from withdrawing control from an HR 410, the Marker 4680 (Disable HR 420 activation) must be set.
- Software version 60642x-03 supports only one HR 550 wireless handwheel in the system.



Note

The PLC program must ensure that only one of the operating units (handwheels, machine operating panels) is active at any one time so as to avoid danger to the operator.

Please note that it may be necessary for the operator to confirm which operating unit is active before it can actually be used. The risk assessment of your machine must show whether this is necessary. Such a requirement could be realized, for example, by pressing the permissive button of the respective operating unit. Then the operating unit that should be active according to the PLC program cannot be used for any machine functions until the permissive button of this operating unit has been pressed.

The emergency stop function of the individual operating stations must always be active, however!

MP7640.x Handwheel

MP7640.0 for X23 of the last machine operating panel in the HSCI chain, and MP7640.4 for X29 of the MC main computer

Input: 0: No handwheel
1: Reserved
2: HR 130
3: Reserved
4: Reserved
5: Up to three HR 150 via HRA 110
6: HR 410
7, 8: Reserved
9: HR 410FS
10: Reserved
11: HR420/HR520 without LED activation
12: HR 550FS handwheel with wireless transmission
13: HR520 with LED activation

MP7641 Handwheel settings

Input: Bit 0 – HR 130, HR 330, HRA 110: Input of the subdivision factor
0: Through iTNC keyboard
1: Through PLC Module 9036
Bit 4 – Handwheel superimposition in the active tool-axis direction
0: Behavior as before
1: VT axis can be selected
Bit 5 – Inactive behavior of HR 420, HR 5xx
0: Report the keys of the HR to the PLC only when the HR is active
1: Report the keys of the HR to the PLC even if the HR is not active
Bit 6 – Selecting and traversing auxiliary axes with HR 420, HR 5xx
0: Traversing auxiliary axes not possible
1: Traversing auxiliary axes is possible
Bit 7 – Teach-In button on HR 550
0: By the NC
1: By the PLC
Bit 8 – CTRL button on HR 550
0: By the NC
1: By the PLC
Bit 9 – Vertical soft keys when the handwheel is active
0: PLC soft keys are not active when HR is active
1: PLC soft keys are active when HR is active

MP7642.x Handwheel name

MP7642.0 for X23 of the last machine operating panel in the HSCI chain, and MP7642.4 for X29 of the MC main computer

Format: String
Input: 31 arbitrary characters



MP7649 Initialization parameter for handwheel at X29 of the MC main computer:

MP7649.0 Assignment of the handwheel keypad for HR 410
 Input: 0: Evaluation of the keys by NC, including LEDs
 1: Evaluation of the keys by PLC

MP7649.1 Fixed assignment of a third handwheel if MP7649.2 = 1
 Input: 1: X axis
 2: Axis Y
 4: Axis Z
 8: Axis IV (MP410.3)
 16: Axis V (MP410.4)

MP7649.2 Assignment of a third handwheel by axis selection switch or MP7649.1
 Input: 0: Assignment by axis selection switch according to MP7645.0
 1: Assignment by MP7645.1

MP7649.3 HRA 55xFS: Behavior of the permissive buttons
 Input: 0: Permissive buttons not pressed → Relay contacts A and B are open, but not cross-circuit proof
 1: Handwheel permissive buttons are cross-circuit proof

MP7649.4–7 Without function

MP7650.x Handwheel counting direction (for HRA 110: for each axis)
 MP7650.0 for X23 of the last machine operating panel in the HSCI chain, and MP7650.4 for X29 of the MC main computer

Input: Bit 0:
 0: Negative counting direction
 1: Positive counting direction

Only for HRA 110: Bits 0 to 17 represent axes 1 to 18
 0: Negative counting direction
 1: Positive counting direction

MP7660 Sensitivity for electronic handwheel
 Input: 0 to 65 535 [increments]

MP7670 Subdivision factor for handwheel
 Input: 0 to 10

MP7670.0 Subdivision factor for slow speed
 MP7670.1 Subdivision factor for medium speed (only HR 410)
 MP7670.2 Subdivision factor for fast speed (only HR 410)

MP7674.x Axis-specific subdivision factor for handwheel
 Input: 0: No limitation
 1 to 10

MP7675.x Handwheel, axis-specific maximum path
 Input: 0: No limitation
 0.0001 to 10.0000 [mm]

		Set	Reset
M4576	Disable the handwheel	PLC	PLC
W1062/1176	Lock the handwheel for specific axes	PLC	PLC



Module 9035 Read status information

Call:

PS B/W/D/K <70 or 71>

CM 9035

PL B/W/D <Handwheel information>
70: Active handwheel
71: Standard handwheel

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Status information invalid
	20	Call was not in a submit or spawn job

Module 9036 Write status information, activate/deactivate functions

Condition: MP7641 bit 0 = 1

In addition to writing status information, this module can now also be used to activate or deactivate certain functions. Numbers starting from 1000 are used for this purpose. The first function being introduced (number 1000) is host computer operation. Starting with number range 1000, calls must be made from a submit/spawn job.

The information to be overwritten is designated with a transferred number.

Handwheel subdivision factors are limited to a smallest possible value, depending on the rapid traverse rate of the respective axis.

CAUTION: No error message!

Number	Function	Value
0	Handwheel subdivision key X (MP410.0)	0 to 10
1	Handwheel subdivision key Y (MP410.1)	0 to 10
2	Handwheel subdivision key Z (MP410.2)	0 to 10
3	Handwheel subdivision key IV (MP410.3)	0 to 10
4	Handwheel subdivision key V (MP410.4)	0 to 10
5	Handwheel subdivision of all axes	0 to 10
6	Select the handwheel axis (not for HRA 110)	0 to 17 for axes 1 to 18; -1: Deselect all axes (also VT) 999: Virtual tool axis (VT)
7	Handwheel speed range ■ 0 = Slow speed ■ 1 = Medium speed ■ 2 = Rapid speed	0 to 2



Number	Function	Value
10	See "Incremental Jog Positioning"	
11	Handwheel subdivision of axis 1	0 to 10
12	Handwheel subdivision of axis 2	0 to 10
13	Handwheel subdivision of axis 3	0 to 10
14	Handwheel subdivision of axis 4	0 to 10
15	Handwheel subdivision of axis 5	0 to 10
16	Handwheel subdivision of axis 6	0 to 10
17	Handwheel subdivision of axis 7	0 to 10
18	Handwheel subdivision of axis 8	0 to 10
19	Handwheel subdivision of axis 9	0 to 10
20	Select active handwheel (only SW 60642x)	
21	Select standard handwheel (only SW 60642x)	
1000	Activate/deactivate host computer operation	0 or 1 0: Deactivate 1: Activate

Call:

PS B/W/D/K <Number of the status information>

PS B/W/D/K <Value to be written>

CM 9036

PL B/W/D <Error code>

0: Status written

1: Incorrect status code

2: Transferred value out of range

3: Input disabled

4: Remote operation already activated/deactivated or PLC module called in impermissible operating mode (e.g. impermissible editing mode, PLC diagnosis, Test Run started , ...)

Error code:

Marker	Value	Meaning
M4203	0	Status information was written
	1	Error code in W1022
W1022	3	Transferred value is outside the valid range
	2	Incorrect number of the status information
	6	Input disabled
	20	Only function 1000: Call not from submit/spawn
	61	Only in function 1000: Function cannot be executed

8.8.1 HR 130 panel-mounted handwheel

- ▶ Enter MP7640 = 2 (HR 130).

When the axis keys are pressed, the associated cursor and the handwheel symbol are displayed simultaneously.

8.8.2 HR 410 portable handwheel

- ▶ Enter MP7640 = 6 (HR 410).
- ▶ In MP7645, specify whether the keys on the handwheel are to be evaluated by the NC or PLC.

Evaluation of the keys by NC: MP7645.0 = 0

X		IV
Y		V
Z		
Slow feed rate	Medium feed rate	Fast feed rate
-		+
O109 I173	O110 I174	O111 I175



With the exception of the function keys A, B and C, all keys are evaluated by the NC.

- ▶ With MP7670.x, select the subdivision factors for slow, medium and fast speed.
- ▶ With MP7671.x, define the values for slow, medium and fast speed. The speed is entered as a percentage of the manual feed rate (MP1020.x).
- ▶ If MP7641 bit 1 is set, then with MP7672.x you can set the distance per handwheel step for the HR 410. Different distances per step can be set for three speed stages.

Evaluation of the keys via the PLC: MP7645.0 = 1

O96 I160		O97 I161
O98 I162		O99 I163
O100 I164		O103 I167
O104 I168	O105 I169	O106 I170
I171		I172
O109 I173	O110 I174	O111 I175

All keys are evaluated by the PLC.

Module 9036 sets the handwheel axis and handwheel subdivision.

With W766 you can influence the feed rate of the axis direction keys.

MP7645 Initializing parameter for handwheel. If an HR 410 is installed, MP7645.0 has the following meaning:

MP7645.0 Assignment of the handwheel keypad for HR 410
Input: 0: Evaluation of the keys by NC, including LEDs
 1: Evaluation of the keys by PLC
MP7645.1–7 Have no function

MP7670 Subdivision factor for handwheel

Input: 0 to 10
MP7670.0 Subdivision factor for slow speed
MP7670.1 Subdivision factor for medium speed (only HR 410)
MP7670.2 Subdivision factor for fast speed (only HR 410)

MP7671 Handwheel feed rate in the Handwheel operating mode with HR 410

Input: 0 to 1000 [% of MP1020]
MP7671.0 Slow speed
MP7671.1 Medium speed (only HR 410)
MP7671.2 Fast speed (only HR 410)

MP7672 HR 410, distance per handwheel step

Input: 0.0000 to 1.0000 [mm]
MP7672.0 Slow speed
MP7672.1 Medium speed
MP7672.2 Fast speed



8.8.3 HR 420 portable handwheel

Settings

- ▶ Enter MP7640 = 11 (HR 420)
- ▶ In MP7641, specify whether you are using an HR 420 with or without detent, and whether the keys on the handwheel are to be evaluated by the NC or PLC.

Information about MP7641:

■ Bit 2

- Bit 2 = 0: If the HR 420 assumes control, then W1046/W1144 (Manual traverse in positive direction), W1048/W1148 (Manual traverse in negative direction), W1050/W1152 (Incremental jog positioning in positive direction), W1052/W1156 (Incremental jog positioning in negative direction) and M4561 (Rapid traverse) have no effect. Only the keys on the HR 420 are valid.
- Bit 2 = 1: The direction keys and the rapid-traverse key of the HR 420 must be evaluated by the PLC. For safety reasons, once the HR 420 has assumed control, only the keys of the HR 420 (M4660) should be active.

■ Bit 3

- Bit 3 = 0: M4564 (NC start) has no effect once the HR 420 assumes control. Only the NC start from the HR 420 is valid. An NC stop is performed via the corresponding key on the HR420 or via M4560 (NC stop).
- Bit 3 = 1: The NC start and NC stop keys of the HR 420 must be evaluated by the PLC. For safety reasons, NC start should only be activatable from the HR 420 once it has assumed control (M4660).

- Spindle start and spindle stop must always be conducted by the PLC.



All keys are evaluated by the NC. Certain keys are mapped to markers.

F1	F2	F3	F4	F5
X	Y	Z	IV	V
	↑	Handwheel active/ inactive	↓	
	- (M4667)	Rapid traverse (M4663)	+ (M4666)	
	Spindle start (M4664)	Actual position capture	NC start (M4661)	
	Spindle stop (M4665)	Ctrl (M4668)	NC stop (M4662)	

Activation and override potentiometers

If the HR 420 is activated (via the handwheel key on the HR), a small pop-up window appears on the screen of the iTNC and entries via the keys (keyboard and horizontal soft keys) are disabled. However, the override potentiometers of the keyboard remain active.

The activation of the HR 420 can be disabled by PLC. If you set the marker M4680, the message **HR not allowed** appears in the handwheel's display. If the marker is set and the user presses the handwheel activation key, the error message **Wrong operating mode for handwheel** appears.

If the marker M4680 is set when the handwheel is active, the marker has no effect.

If the override potentiometers of the handwheel are to become active, then this can only be done with the <Ctrl> + <handwheel symbol> key combination on the HR 420.

If this key combination is pressed, then a selection menu appears on the handwheel, in which the potentiometers to be activated must be selected.

■ Display: **Handwheel1/ Operating panel**

■ Soft keys on the handwheel:

- Soft key F1: **HW** (potentiometers on the handwheel become active)
- Soft key F2: **KBD** (potentiometers on the operating panel become active)

If the override potentiometers on the handwheel are switched active, then the message **Handwheel override active** also appears in the pop-up window on the screen of the iTNC, and the handwheel can only be deactivated if the control of the override potentiometers is returned to the operating panel. If you try to deactivate the HR 420 while it has control over the override, then a window appears requesting that you first deactivate the handwheel's control over the override.



Danger

Please note that the feed rate or spindle speed set becomes higher or lower depending on the potentiometer setting.

PLC Marker M4670 provides information on whether the override potentiometers of the HR 420 or of the machine operating panel are active.

M4670 – Potentiometer of HR 420 is active

M4670 = 1: Potentiometer of machine operating panel is active

The override factor of the active override potentiometer can also be influenced with M764 or W766.

When marker M4626 is set, the PLC locks all key inputs of the TE keyboard unit, including the soft keys, except for the CE and the screen switch-over key. The lock is only effective for the machine screen (iTNC - machine), the editing screen (iTNC - editing) is not affected by the lock. Furthermore, this disabling does not affect the keys on the MB machine operating panel.

This has the same effect as pressing the handwheel activation key on the HR 420. Setting the new marker M4626 automatically results in setting M4660 (HR 420 assumes control). If a corresponding message window informing you about the current status on the screen is desired, it must be displayed via the PLC program. Control must also be returned to the operating panel via the PLC program.

Window upon activation of the HR 420

When the HR 420 is activated, a pop-up window appears on the iTNC, indicating that the HR 420 has assumed control. This window is a special pop-up window. The text color is defined in MP7366.1, and the window background color in MP7350.

MP7641	Handwheel settings
Input:	Bit 1 – HR 420, HR 5xx: With detent positions 0: Without detent positions 1: With detent positions Bit 2 – HR 420, HR 5xx: Axis direction keys and rapid traverse 0: By the NC 1: By the PLC Bit 3 – HR 420, HR 5xx: NC start / NC stop 0: By the NC 1: By the PLC

		Set	Reset
M4626	Disabling of keys on the TE	PLC	PLC
M4660	HR 420, HR 5xx assumes control	NC	NC
M4661	NC start on HR 420, HR 5xx	NC	NC
M4662	NC stop on HR 420, HR 5xx	NC	NC
M4663	Rapid traverse key on HR 420, HR 5xx	NC	NC
M4664	Spindle start on HR 420, HR 5xx	NC	NC
M4665	Spindle stop on HR 420, HR 5xx	NC	NC
M4666	Plus (+) key on HR 420, HR 5xx	NC	NC
M4667	Minus (-) key on HR 420, HR 5xx	NC	NC
M4668	CTRL key on HR 420, HR 5xx	NC	NC
M4670	Potentiometer of HR 420, HR 5xx is active	NC	NC
M4680	Disable activation of HR 420, HR 5xx	PLC	PLC

Free soft-key definition for HR 420, HR 5xx

You will find information about the freely definable soft-key menu for the HR 420, HR 5xx under "Soft-key project file for screen" on page 1372.

8.8.4 HR 520 portable handwheel

With the HR 520, HEIDENHAIN is introducing the first handwheel of the new generation HR 5xx. Unlike its predecessor model HR 420, the new HR 520 features a 6th axis key, which can be configured, for example, for selecting the "virtual tool axis" (option 44). The HR 520 also offers 6 LEDs, which can be controlled by PLC markers. For example, these LEDs can be used for optical confirmation of menu-driven switching processes on the machine. In addition, it is possible to scan all handwheel keys from the PLC and use them accordingly in the PLC program.

The handwheel can be fastened to the machine using the magnetic holding pads on the rear of the handwheel or an optionally available mount (ID 591 065-xx).

The HR 520 handwheel is available in versions with or without detent.

The HR 520 is connected with the same cables and adapters as the HR 420.

Power consumption of HR 520: 1 W

Settings

- ▶ Set MP7640 to 11 or 13, depending on whether you want to use the LEDs of the HR 520.
- ▶ In MP7641, specify whether you are using an HR 5x0 with or without detent, and whether the keys on the handwheel are to be evaluated by the NC or PLC.

All settings that can be defined in the machine parameters of the HR 420 can also be used for the HR 520. The PLC markers of the HR 420 also match those of the HR 520.

The six LEDs of the HR 5x0 can be controlled by the specified PLC markers.

F1	F2	F3	F4	F5
	X	Y	Z	
	IV	V	VI	
	↑	Handwheel I active/ inactive	↓	
	- (M4667)	Rapid traverse (M4663)	+ (M4666)	
	Spindle start (M4664) LED (M4684)	Actual position capture LED (M4689)	NC start (M4661) LED on (M4681)	
	Spindle stop (M4665) LED (M4685)	Ctrl (M4668) LED (M4688)	NC stop (M4662) LED (M4682)	

8.8.5 HR 550FS wireless handwheel

Another product of the new HR 5xx handwheel family, the HR 550FS wireless handwheel for the iTNC 530 (HSCI), is now available in addition to the HR 520. The standard iTNC 530 (HSCI) supports the use of electronic handwheels. The handwheels are connected to X23 of the MC, or to X23 of the machine operating panel in the case of HSCI.

A wireless handwheel system consists of the HR 550FS mobile part and the HRA 551FS handwheel adapter. The HR 550FS has a rechargeable battery for mobile operation on the machine. During operation it continuously communicates via radio with the HRA 551FS handwheel adapter, which communicates directly with the control via a serial cable connection.

The HRA 551FS handwheel adapter, which functions both as receiving and charging station for the mobile part, features an M23 connector system for connection to the control via a standard handwheel adapter. If the wireless handwheel system is damaged, the dummy plug for the emergency stop circuit (ID 271 958-03) can be used so that the operational readiness of the machine can be restored quickly.

To ensure the necessary and unambiguous assignment of the HR 550FS to the HRA 551FS, a pairing is created during commissioning. Once they have been paired, a wireless handwheel and a handwheel adapter can only be operated with the assigned device. As a precautionary measure, paired devices must be marked by the stickers (in five different colors) included in the items supplied so that they can be identified visually (the soft-key label strip for the mobile part must have the same color as the round sticker for the handwheel adapter). After successful pairing, they must be affixed to the wireless handwheel system. The available colors are gray, blue, red, green and yellow.

The wireless handwheel system can be used in many different places thanks to the 2.4 GHz ISM radio band, which is freely accessible almost all over the world. The HR 550FS complies with the internationally accepted standards. For radio communication, the HEIDENHAIN HR 550FS uses 2.4 GHz ZigBee transmission technology with a special HEIDENHAIN communications protocol.



HR 550 wireless handwheel

The HR 550FS wireless handwheel must always be used together with the HRA 551FS handwheel adapter. The handwheel and the handwheel adapter communicate with each other via radio transmission. The handwheel adapter is connected to X23 of the control system via the connecting cable.

When the handwheel is located in the HRA, contacts on the HRA and on the rear of the HR are used to communicate via serial data transmission instead of via radio transmission. The function of the HR 550FS remains the same, regardless of whether it communicates via radio transmission or via the handwheel adapter.

The HRA handwheel adapter features an integrated charger for the handwheel batteries. In addition, the safety-related signals (emergency stop, permissive buttons) are translated into relay contacts through the HRA. They must be used to connect the emergency stop button and the permissive buttons of the handwheel to the control and the associated safety circuits.



Danger

If more than one control with a portable HR 550FS handwheel is located in a factory hall, a room or a working area, the handwheels must be identified by unique color markings. The color marking must be applied to the handwheel **and** the associated machine tool at a clearly visible location. The marking on the handwheel and the machine tool must enable the machine operator to identify unambiguously which handwheel belongs to which machine tool. The color marking must ensure that no confusion occurs. A set of different markers is included with the wireless handwheel.

In addition, the machine operator must be informed that he must ensure/verify every time before using the handwheel that he uses the correct handwheel for the respective machine tool.

If the machine operator moves with the HR too far away from the HRA, a vibration alarm announces that transmission is weak before the machine operator has actually left the radio range. If the machine operator leaves the radio range anyway, or if radio communication is interrupted for other reasons, the HRA relay contacts for emergency stop and for the permissive buttons will open. You must ensure through appropriate wiring that this triggers an emergency stop on the control.

HEIDENHAIN also recommends that you always place the wireless handwheel into the HRA 551FS adapter when you are not using it. This prevents you from confusing the different handwheels. It also ensures that the handwheel battery is recharged and prevents an unexpected emergency stop reaction triggered by an empty battery.

When using a wireless handwheel and the HRA handwheel adapter at X23, you must ensure that the assignment between the handwheel and the HRA is unambiguous. The wireless handwheel serial number and the HRA serial number are used for mutual assignment. When the control to which the HR and the HRA are connected is switched on for the first time, both units exchange their serial numbers and store them. This process is called "pairing." To do this, the wireless handwheel must be located in the HRA. If this is the case, the HR and the HRA communicate only over the serial interface. Both serial numbers (HRA and HR) will then also be saved in the control. If you remove the HR from the HRA during the initialization process, the process will be canceled and a corresponding error message will be issued.

During the emergency stop test and when the connection to the wireless handwheel is set up, the system ensures that only the HR and the HRA are addressed, whose serial numbers have been assigned unambiguously to each other. If a difference is found in one of the comparisons of serial numbers, the connection will be terminated. During this entire process the relays in the AP for the emergency stop and the permissive buttons remain open.



Switching the HR 550FS on/off

To switch the HR 550FS off, press the CTRL key and the medium soft key simultaneously. To switch the handwheel on, it is sufficient to press one of the two permissive buttons. Then the HR 550FS runs a power-up test and can be used after having completed the test. During the power-up test, the background illumination of the HR 550FS is active, but the display remains blank. The permissive buttons are tested at the end of the power-up test. None of the permissive buttons may be pressed at this time. In the event of error, an error message appears on the display, informing you that a permissive button is pressed. As a result, the power-up test cannot be completed. The display is filled with characters as soon as the power-up test has been completed successfully, and the "OFF-LINE" message appears on the display. If an error is found during the power-up test, an error message appears on the display and the service department must be informed.

The handwheel is configured in a menu that you open by pressing the SET UP WIRELESS HANDWHEEL soft key. After you have pressed the MOD key, this soft key is displayed in the second soft-key row in the Programming and Editing mode of operation if machine parameter MP7640 has been set to the value of 12. The following functions are available for configuration:

- Assigning the handwheel to a specific handwheel holder
- Setting the transmission channel
- Analyzing the frequency spectrum for determining the optimum transmission channel
- Selecting the transmitter power
- Statistical information on the transmission quality

For configuring the HR 550FS handwheel, refer to the chapter "Configuring the HR 550FS Wireless Handwheel" in the User's Manual for the control.

The functions that can be defined for the soft keys on the HR 550FS are the same as for the HR 520(FS).

After the wireless handwheel has been configured and switched on, it can be activated via the handwheel activation key. Machine operation then switches from the machine operating panel to the handwheel. The machine tool builder must ensure through the PLC program that the machine can be operated by only one operating unit at any one time. Machine operating panel and handwheel must never be active at the same time. The display of the wireless handwheel and the display on the screen inform the operator when the wireless handwheel is active.

Settings

- ▶ Enter MP7640 = 12 (HR 550FS).
- ▶ In MP7641, specify whether you are using an HR 5x0 with or without detent, and whether the keys on the handwheel are to be evaluated by the NC or PLC.

All settings that can be defined in the machine parameters of the HR 520 can also be used for the HR 550FS. The PLC markers of the HR 520 also match those of the HR 550FS.

All keys of the HR 5x0 are evaluated by the NC. Certain keys are mapped to PLC markers. The six LEDs of the HR 5x0 can be controlled by the specified PLC markers.

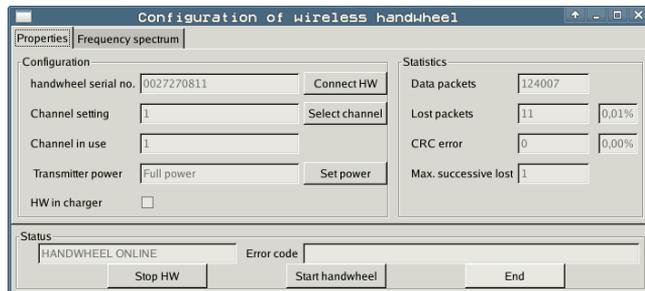
F1	F2	F3	F4	F5
	X	Y	Z	
	IV	V	VI	
	↑	Handwheel active/ inactive	↓	
	- (M4667)	Rapid traverse (M4663)	+ (M4666)	
	Spindle Start (M4664) LED on (M4684)	Actual position capture LED on (M4689)	NC start (M4661) LED on (M4681)	
	Spindle Stop (M4665) LED on (M4685)	Ctrl (M4668) LED on (M4688)	NC stop (M4662) LED on (M4682)	

Information in the handwheel configuration dialog

The general procedure for the configuration of the HR 550FS handwheel is described in the iTNC 530 User's Manual.

To access the handwheel configuration dialog, proceed as follows:

- ▶ Press the MOD key
- ▶ Scroll through the soft-key row
- ▶ Press the SET UP WIRELESS HANDWHEEL soft key



HEIDENHAIN generally recommends that in **Channel setting** you manually define the radio channel to be used. To select a vacant radio channel, please follow the information provided in this document. We advise against using the **Best channel** setting because it does not allow planning for the radio environment, and therefore unexpected disturbances during the operation of the wireless handwheel system may occur.

8.8.6 HR 150 panel-mounted handwheels with HRA 110 handwheel adapter

- ▶ Enter MP7640 = 5 (HR 150 via HRA 110)

You can use the step switch S1 to choose the subdivision factor. For this purpose you must evaluate the inputs I160 to I167 of the switch in the PLC and activate the corresponding interpolation factor with Module 9036.

Axes X and Y are permanently assigned to the handwheel inputs X1 and X2. You can assign the third handwheel (input X3) to any other axis. All handwheel axes are indicated by the handwheel symbol.

- ▶ Take the designation for axes IV and V from MP410.x.
- ▶ In MP7645.2, specify how the axis for the third handwheel is selected.
 - Selection by axis selection switch (switch S2, see MP7645.0)
 - Selection permanently defined in MP7645.1

MP7645 Initializing parameter for handwheel

MP7645.0 Assignment of a third handwheel via axis selector switch S2, when MP7645.2 = 0

Input: 0: Position 1 (at the left stop)3rd handwheel axis Z
Position 23rd handwheel axis IV
Position 33rd handwheel axis V
1: Position 1 (at the left stop)3rd handwheel axis X
Position 23rd handwheel axis Y
Position 33rd handwheel axis Z
Position 43rd handwheel axis IV
Position 53rd handwheel axis V
2: Position 33rd handwheel axis Z
Position 43rd handwheel axis IV
Position 53rd handwheel axis V

MP7645.1 Fixed assignment of a third handwheel if MP7645.2 = 1

Input: 1: X axis
2: Y axis
4: Z axis
8: Axis IV (MP410.3)
16: Axis V (MP410.4)

MP7645.2 Assignment of a third handwheel by axis selection switch or MP7645.1

Input: 0: Assignment by axis selection switch according to MP7645.0
1: Assignment by MP7645.1

MP7645.3–7 Have no function

Assignment of switch positions to PLC inputs

The tables below list the assignments of switch positions of S1 and S2 to the PLC inputs I160 to I175.

The two switches work with a 0 V logic circuit.

Example: If switch S1 is in position 3, input I162 is logically 0, and the inputs I160, I161, I163 to I167 are logically 1.

Step switch 1: Step switch for choosing the subdivision factor

Switch position	PLC input
1 (at the left stop)	I160
2	I161
3	I162
4	I163
5	I164
6	I165
7	I166
8 (at the right stop)	I167

Step switch 2: Axis selection switch

Switch position	PLC input
1 (at the left stop)	I168
2	I169
3	I170
4	I171
5	I172
6	I173
7	I174
8 (at the right stop)	I175

8.9 PLC inputs/outputs

The following PLC modules are no longer supported in HSCI systems. The modules return error code 2 in W1022 when they are called in HSCI systems:

- PLC Module 9002
- PLC Module 9003 (returns error code 3 in W1022)
- PLC Module 9005
- PLC Module 9007
- PLC Module 9008
- PLC Module 9009
- PLC Module 9030
- PLC Module 9137
- PLC Module 9138
- PLC Module 9139

8.9.1 24 V– switching input/outputs

In PLC addresses you can find the current states of the switching inputs and outputs. In a system with HSCI, the PLC inputs and outputs are configured in the IOconfig software tool for PCs.

With Module 9004 you can evaluate the rising or falling edge of the PLC inputs.



Note

Before the PLC program is converted, the PLC outputs are reset. In addition, the memory of the PLC outputs is reset. During a loss of power (power fail), the control tries to reset the PLC outputs.

If all PLC outputs are switched off (e.g., during PLC program compilation or due to a PLC run-time error), the outputs that can not be switched off by an emergency stop can be switched off with a delay.

If all PLC outputs are switched off (e.g., during PLC program compilation or due to a PLC run-time error), correspondingly configured outputs can be switched off with a delay.

In a system with HSCI, the settings for the delay times are configured in the IOconfig software tool for PCs.

The delay only affects outputs that cannot be switched off by an emergency stop, since for the outputs that can be shut off by an emergency stop, the 24 V supply is shut off immediately. The outputs that are to be switched off via an emergency stop are defined via an appropriate wiring of the 24 V supply of these outputs (see the basic circuit diagram). Please refer to the documentation for IOconfig.



After the SHUTDOWN soft key has been pressed, the PLC program can delay or prevent the display of the pop-up window (shutdown: yes/no). The message "The TNC will be shut down and machine functions will be performed" in the operating modes window indicates this condition. An appropriate user guidance must have been programmed by using the available tools, such as dialog texts, pop-up windows, vertical soft key, etc., in order to inform the user about concluding operating steps that may be necessary.

The possibilities of configuring a shutdown (MP4040 and the subsequent functions) specified below can continue to be used, because these settings do not become effective until the prompt "Do you really want to switch off?" has been acknowledged.

■ **M4195**

The marker M4195 is set when the SHUTDOWN soft key is pressed (= 1). The PLC program can prevent the control from shutting down by resetting the marker in the next PLC cycle. In this phase (M4195 = 0), the text "The TNC will be shut down and machine functions will be performed" is displayed in the operating modes window and the PLC program can execute specific actions before shutting down the control. The TNC remains in this condition until the marker M4195 is set again.

■ **M4197**

If the PLC actions cannot be concluded, the PLC can cancel shutdown of the control by setting marker M4197 = 1. Then the TNC returns to its original mode of operation and the two markers also return to their initial state.

With MP4043 you can delay the shutdown of the control (after pressing the SHUTDOWN soft key) to enable the PLC to execute final actions.

When you press the SHUTDOWN soft key, the control starts shutting down, PLC marker M4179 is set and the delay time entered in MP4043 starts to elapse. The PLC program can execute final actions during the delay time. If the PLC marker M4179 is reset by the PLC program, the SHUTDOWN process will be continued before the time set in MP4043 has elapsed.

The SHUTDOWN process is automatically continued at the latest after the expiration of the time in MP4043.

If the following conditions apply, however, it cannot be guaranteed that final PLC actions (e.g. data backup of non-volatile memory ranges of the PLC) are executed:

- If no PLC program is active at the time the control is shut down, (e.g. PLC run time error has occurred, power interruption status), the shutdown time is not delayed.
- If a PLC run time error occurs during shutdown, the delay is canceled.

MP4043 Delay during shutdown for the PLC to execute final actions

Input: 1 to 1000 [s]
 0 = No delay

		Set	Reset
M4195	Press the SHUTDOWN soft key	NC	NC
M4197	Cancel the shutdown of the control	PLC	NC



Module 9004 Edges of PLC inputs

With this module you set, upon falling or rising edges of the PLC inputs, specified end markers or bits in the specified byte range. Changes in the inputs are recognized only if a change occurs also in the PLC addresses.

Ensure that the specified edge markers or edge bytes are in an unoccupied area. The edge bytes are written beginning with the least significant bit. Superfluous bits are erased.

Call:

PS B/W/D/K <Number of the first PLC input>

PS B/W/D/K <Number of the first edge marker or edge byte>

PS B/W/D/K <Number of PLC inputs>

PS B/W/D/K <Edge evaluation>

0: Rising edge. Entry in edge marker

1: Falling edge. Entry in edge marker

2: Rising edge. Entry in edge byte

3: Falling edge. Entry in edge byte

CM 9004

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Invalid transfer parameter



8.9.2 Analog inputs

± 10 V DC analog inputs and inputs for connecting PT 100 thermistors are available on the external PL via a PL module (see Chapter 3, Mounting and Electrical Installation).

In HSCI systems, no PLC module is necessary to read the current temperatures of the PT 100 thermistors or the values of analog inputs. The respective analog value is copied by the PLC run-time system into the corresponding PLC operand that was defined for this input in IOconfig. This operand is a 16-bit input word IWx. The assignment entered in IOconfig is saved in a *.DEF file of the PLC program.

Only voltages of up to ± 10 V– are permissible at the analog inputs.

The change rate of the PT 100 inputs can be set in discrete steps of [0.666 K/s] per attached module. The function can be activated or deactivated for each PT 100 input. The value 12000 (120 °C) is set if a PT 100 gets disconnected or is not connected.

Along with the change rate, there is also a filter that forms a mean value for the PT 100 inputs. Similar filters are available for the ± 10 V voltage inputs.

These and other settings are made in the IOconfig software tool for PCs, which includes detailed help documentation about the available possibilities.

8.9.3 Analog outputs

± 10 V DC analog outputs are available on the external PL via a PL module (see Chapter 3, Mounting and Electrical Installation).

In HSCI systems, no PLC module is necessary to output analog voltages. The respective analog value is assumed by the PLC run-time system from the corresponding PLC operand that was defined for this output in IOconfig. This operand is a 16-bit output word OWx. The assignment entered in IOconfig is saved in a *.DEF file of the PLC program.



Note

Each analog axis or spindle needs an analog output on a CMA-H. These outputs are then not available to the PLC.

8.10 Incremental jog positioning

- ▶ The "incremental jog positioning" function is switched on and off with the **INCREMENT OFF/ON** soft key.
- ▶ To position with incremental jog, press the direction keys (W1046/W1144/W1048/W1148).

You can interrogate the current state with Marker M4579.

You can limit the jog increment with Module 9036.

You can ascertain the current jog increment with Module 9035.

You can switch the incremental jog function on and off through the PLC with Module 9186.

		Set	Reset
M4579	INCREMENT OFF/ON soft key	NC	NC

Module 9036 Write status information

The information to be overwritten is designated with a transferred number.

Number of the status information	Function	Value
0 to 7	See "Handwheel"	
10	Jog increment limiting	0.0001 to 10 mm: Jog increment limiting -1; < -2; > 50: Cancellation of jog increment limitation and activation of the jog increment entered last -2: Cancellation of the jog increment limitation and activation of the minimum from the jog increment entered last and the last limitation
11 to 19	See "Handwheel"	

Call:

PS B/W/D/K <Number of the status information>

PS B/W7D/K <Value to be written>

CM 9036

PL B/W/D <Error code>

0: Status written

1: Incorrect status code

2: Transferred value is out of input range

3: Input disabled

Error code:

Marker	Value	Meaning
M4203	0	Status information was written
	1	Error code in W1022
W1022	3	Transferred value outside permissible range
	2	Incorrect number of the status information
	6	Input disabled

Module 9035 Read status information

Call:

PS B/W/D/K <26>

CM 9035

PL B/W/D <Jog increment>

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Status information invalid
	20	Call was not in a submit or spawn job

8.11 Hirth coupling

Hirth coupling describes a type of clamping of rotary axes and swivel heads. Finely splined disks mesh together in order to create a rigid connection.

During datum setting, the NC rounds off according to the grid spacing from MP430.x:

- ▶ Configure the exact positioning in the Hirth grid as PLC positioning.

After positioning an axis with Hirth coupling, Module 9148 can use the nominal position value of the axis as its actual position value. This affects the actual-value display and other internal calculations, such as for the transformation chain for tilting axes.

MANUAL operating mode

As soon as an axis direction key is pressed, the NC resets the corresponding bit in W1026/W1104 (axis in position).

- ▶ As soon as the axis-in-position bit is set again, you check the nominal position with the Hirth grid and derive from it a PLC positioning command to the next grid point.

ELECTRONIC HANDWHEEL operating mode

For the current handwheel axis, the corresponding bit is reset in W1026/W1104 (axis in position).

As soon as you select another handwheel axis, "axis in position" is set for the previous axis.

The Hirth axis can be positioned with the handwheel:

- ▶ Check the actual position with the Hirth grid and derive from it a PLC positioning to the next grid point.

Controlled positioning

The positions of the Hirth axis must be programmed in the grid:

- ▶ Check the positions in the PLC during program run.
- ▶ As soon as "axis in position" is reset, check the target position with the Hirth grid.
 - If the target position is not in the Hirth grid, output a PLC error message.

MP420.x Hirth coupling
Input: 0: No Hirth coupling
1: Hirth coupling

MP430.x Prescribed increment for Hirth coupling
Input: 0.0000 to 180.0000 [°]



Module 9148 Use nominal value as actual value

With Module 9148 you can use the nominal value as actual value for selected axes when the position loop is open. This makes it possible to use the nominal value for certain internal functions such as the actual value display and calculations such as transformation chains of tilting axes.

If you switch from using the actual value to the nominal value with Module 9148, then the actual value is also no longer available in the PLC via Module 9041.

Call:

PS B/W/D/K <Axis>
Bits 0 to 17 represent axes 1 to 18

PS B/W/D/K <Mode>
0: Use nominal value as actual value

CM 9148

Error code:

Marker	Value	Meaning
M4203	0	Nominal value used as actual value
	1	Error code in W1022
W1022	1	Invalid mode
	2	Invalid axes
	24	Module was called in a spawn job or submit job

8.12 Datum shift

With the datum shift function you can offset the defined datum point.

The same initial position must apply for the description of the machine's geometry (see "Tilting and swivel axes" on page 701) and for the datum shift.

You can activate the datum shift during an M/S/T/Q strobe.

Datum shift with D528 to D544

- ▶ In D528 to D544, enter for each axis the distance by which the datum is to be shifted, or use Module 9230. For axes 6 to 18, use only Module 9230.
- ▶ Activate the datum shift with M4132. After the datum shift the NC resets M4132.

The offset is calculated into the position display — the display now shows the position values according to the shifted coordinate system.

Example:

Actual value display for X axis without datum shift = 50

Shift value in D528 = +20

M4132 is set, i.e. the offset is active

New actual value display X = +70 (the old datum receives the value 20).

	Set	Reset
D528 - 544 Datum shift for axes 1 to 5	PLC	PLC
M4132 Activate datum shift from D528 to D544, or call Module 9230	PLC	NC

Module 9230 Datum shift

With this module you transfer the axis and the amount by which the datum is to be shifted. If the module is called for axes 1 to 5, the value is copied to D528 to D544.

M4132 is set when Module 9230 is called. After execution of the datum shift, the NC resets M4132.

Call:

PS B/W/D/K <Axes bit-encoded>

PS B/W/D/K <Shift [0.1 µm]>

CM 9230

Error code:

Marker	Value	Meaning
M4203	0	No errors
	1	Error code in W1022
W1022	2	Invalid axis number
	21	Missing strobe or control is active
	24	Module was called in a spawn job or submit job



8.13 Touch probe

The following touch probes can be connected:

- TS 440 with compact dimensions
 - TS 444 with alternative battery-free power supply via compressed air through the spindle head
 - TS 640 with wide-range infrared transmission and long operating time
 - TS 642 with mechanical activation switch in the clamping shank
 - TS 740 with high probing accuracy and repeatability, and low probing forces
 - TS 220 with signal transmission over cable connection for machines with manual tool change
 - TT 140 with rated break point
 - TT 449 for infrared communication with the SE 642
 - SE 640 for integration in the machine workspace
 - SE 642 for mounting in the working space, for operation of a TS 44x, TS 64x, TS 74x infrared workpiece touch probe and the TT 449 infrared tool touch probe
For this purpose, the SE 642 must be connected to X112 of the PLB via the cable with the ID 663 631-xx. With machine parameter MP6508 = 1, this SE 642 is activated at X112. The battery warning of the TT 449 is also evaluated if it is connected via the SE 642 to X112.
The machine parameter MP6508 for configuring the SE 642 can be changed via the PLC or an FN function. Hence, up to three touch probes can be connected using a macro or a cycle, for example.
 - The battery warning of the TT 449 is not evaluated. If the battery of the TT 449 is empty, the TT is no longer ready for operation and gives a corresponding message.
 - SE 540 for integration in the spindle head
- Specify in MP6010 which touch probe is connected.
- Make sure that the spindle is locked during the measuring process.

With **FN18: SYSREAD** you can read the current touch probe data.

MP6010 Selection of the touch probe

Input: 0: Touch probe with cable transmission (TS 120, TS 220)
1: Touch probe with infrared transmission (TS 632)
2: Touch probe with infrared transmission (TS 440, TS 640)
3: Battery-free touch probe with infrared transmission (TS 444)

MP6508 SE 642 transmitter-receiver unit

Input: 0: Not active (TS at X112, TT at X113)
1: SE 642 with TT and TS at X112

8.13.1 Using the touch probes

TS 440, TS 444, TS 640

The TS 440, TS 444 and TS 640 touch probes are activated on the rising edge of the starting signal and deactivated on the falling edge. For safety reasons, the touch probes are automatically switched off after 30 minutes of no activity (no stylus deflection).

HEIDENHAIN recommends:

- ▶ Set M4056 as soon as the touch probe is in the spindle. This way the touch probe is switched on via the rising edge of the start signal.
- ▶ Reset M4056 in order to switch the touch probe off. The touch probe is switched off via the falling edge of the start signal.

As long as M4056 is set, the NC checks the ready signal of the touch probe, and automatically triggers it anew as soon as it switches itself off. This ensures that the touch probe is switched on as long as it is within the working space.

If it is not possible to keep the touch probe ready, positioning motions in the **Manual** and **Handwheel** modes of operation are interrupted with the **Probe system not ready** error message. After acknowledging the error message with the CE key, the touch probe can be freely traversed for 60 seconds in the **Manual** mode of operation. The error message then appears again.



Attention

Touch-probe monitoring is not active as long as the touch probe does not report that it is ready.

If, in an exceptional case, the touch probe must be triggered anew during a probing cycle, the **Probe system not ready** error message appears, and the cycle must be restarted from the beginning.



Note

The iTNC 530 always emits a start signal when beginning a touch probe cycle, meaning Modules 9135 and 9136 do not need to be used for HEIDENHAIN touch probes.

If M4056 is set and the stylus is deflected, the NC stops the machine in all operating modes. The maximum feed rate is limited to the value specified in MP6150. If M4056 is set and the touch probe does not provide a ready signal, the feed-rate enabling is reset.

If you do not set M4056, the control detects a deflection of the stylus only if a probing function has been started.

Before the probing process is started, the NC sets M4055. Before executing the function, the NC waits until you reset M4055. This allows you to take a break, for example, to clean the measured object with compressed air before starting the probing process.

M4051 is set if the stylus is deflected before the probe block has been started. If it is, the probing block start is delayed by 1 second.



Marker M4054 is set for the battery-free TS 444 touch probe (MP6010 = 3) when the touch probe is ready for operation (M4050) and the supply voltage for the battery-free touch probe is too low. If no touch probe is connected, the value of marker M4054 is undefined. Power is supplied to the battery-free touch probe by compressed air driving an integrated air turbine generator. The capacitor in the touch probe stores energy for approx. two minutes of continuous operation. Evaluation of marker M4054 makes it possible to purposefully control the supply of compressed air.

The NC takes over control of the probing process. Certain conditions are indicated in M4050 to M4054.

		Set	Reset
M4050	Touch probe not ready, ready signal is missing	NC	NC
M4051	Stylus deflected before start of probing cycle	NC	NC
M4052	Stylus is deflected, probing process is completed	NC	PLC
M4053	Probing process has been completed or canceled	NC	NC
M4054	Battery voltage too low (battery warning at touch probe connection); evaluated only during the probing process^a	NC	NC
M4055	Enable the probing process	NC	PLC
M4056	NC stop in all operating modes if stylus is deflected	PLC	PLC
M4057	Touch probe cycles active (FN17: ID990 NR2)	NC	NC

- a. is only supported for the battery-free TS 444 touch probe (MP6010 = 3).

Non-HEIDENHAIN touch probes

Module 9135 Switch on the 3-D touch probe

With Module 9135 you can switch on or retrigger certain 3-D touch probes. If the touch probe is already switched on, the module call has no effect. If M4056 is set and the touch probe does not provide a ready signal, the feed-rate enabling (M4563) is reset.

Call:
CM 9135

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error in module run

Module 9136 Switch the touch probe on/off

Module 9136 switches a touch probe at X12 on or off once. If the touch probe does not supply a ready signal, and if M4056 is set (NC stop for deflected touch probe in all operating modes), the feed-rate enable is reset. After Module 9136 is called the first time, the NC does not output any more on/off signals, meaning that the touch probe is now solely controlled by the PLC program.

Call:
PS B/W/D/K <Touch probe state>
0: Switch off touch probe
1: Switch on touch probe

CM 9136

Error code:

Marker	Value	Meaning
M4203	0	Touch probe on or off
	1	Error code in W1022
W1022	1	Invalid touch probe state

8.13.2 Touch probe cycles

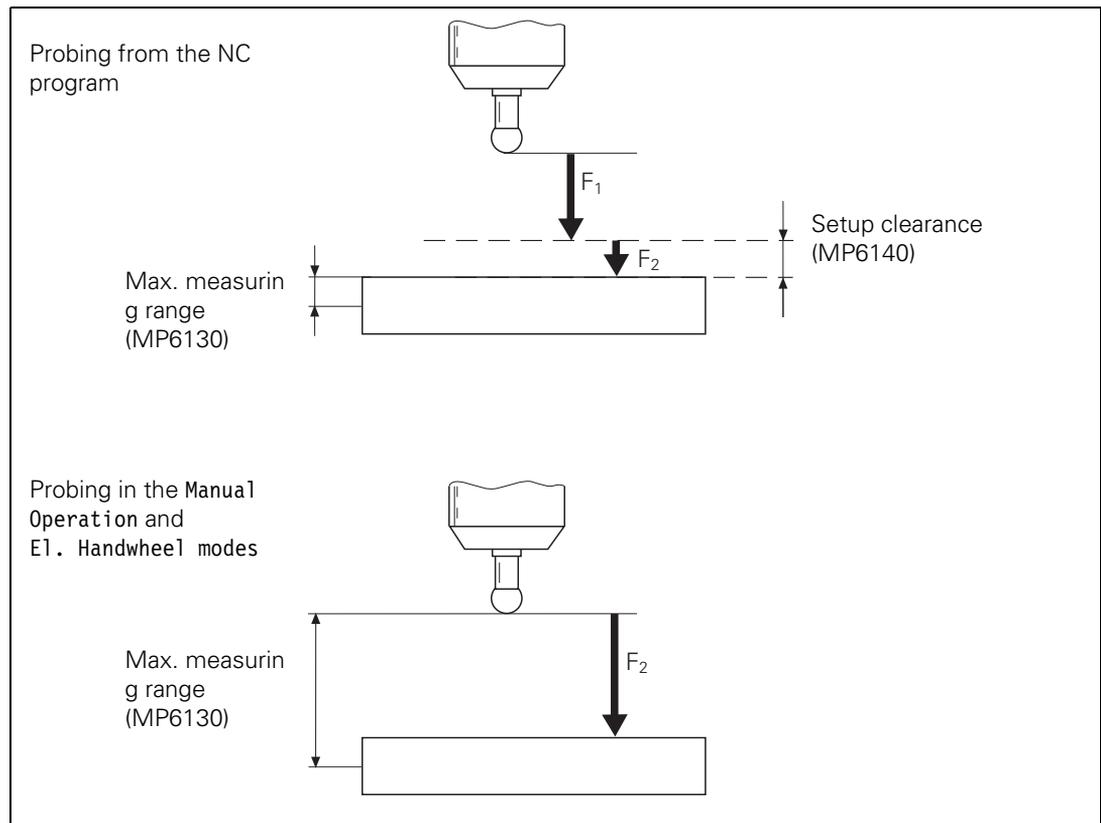
The probing cycles are available in the **Manual Operation** and **Electronic Handwheel** modes and in the NC program (see the Touch Probe Cycles User's Manual).

- ▶ With the machine parameters, adjust the touch probe to the measuring conditions.
- ▶ With MP6165, you can specify if during manual measurement and in the touch probe cycles 0 and 1 the touch probe with infrared transmission is oriented so that it is always deflected in the same direction.
- ▶ With MP6166 you can set whether the probing directions is transformed in the rotated plane in the manual measuring cycles and when basic rotation is active.



Note

Please note that MP6166 is not in effect for the calibration cycles and the cycles for determining the basic rotation.



F_1 = rapid traverse during probing from the NC program:
MP6150 for triggering touch probe

F_2 = probing feed rate:
MP6120 for triggering touch probe

If the maximum measuring range (MP6130) is exceeded, the error message **Touch point inaccessible** appears.

MP6140 and MP6150 have no meaning in the **Manual** and **Electronic Handwheel** operating modes.

- MP6120 Probing feed rate (triggering touch probe)**
Input: 1 to 10 000 [mm/min]
- MP6130 Maximum measuring range**
Input: 0.001 to 99 999.9999 [mm]
- MP6140 Setup clearance above measuring point**
Input: 0.001 to 99 999.9999 [mm]
- MP6150 Rapid traverse in probing cycle**
Input: 10 to 20 000 [mm/min]
- MP6151 Pre-positioning in probing cycle with rapid traverse**
Input: 0: Pre-position with speed from MP6150
 1: Pre-positioning at rapid traverse
- MP6165 Orient the probe before approaching with Cycle 0 or 1, or with manual probing**
Input: 0: Probe is not oriented before each probing
 1: Probe is oriented and always deflected in the same direction
- MP6166 Probing direction of the touch probe with consideration of an active basic rotation**
Input: 0: Inactive (default)
 1: Active

**Special case:
Horizontal / vertical
swivel head**

If you are using a **horizontal/vertical swivel head**, the compensation values of the touch probe must be accounted for in different axes:

- ▶ With Module 9153, switch the touch probe axis in order to correctly account for the compensation values.

Module 9153 Switch the touch probe axis

Specify a new touch probe axis (axis 0, 1 or 2) for manual measurement. A new touch probe axis can be specified only if MP7490 bit 2 = 1.

Call:
PS B/W/D/K <Axis number 0 to 2>
CM 9153

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	2	Invalid axis number
	20	Module was not called in a spawn or submit job



Calibration data

The iTNC can save the calibration data for up to three touch probes at once:

- ▶ Set MP7490 bit 2 = 1.
- ▶ Use the traverse range switching function to activate the current data with M4574/M4575.
- ▶ Specify with MP7411 bit 0 whether a probing block is to use the tool data (length, radius, axis) from the last **TOOL CALL** block or from the calibrated data of the touch probe. If MP7411 bit 0 = 1, you can use soft keys to take the effective length and effective radius over into the tool table. Also, the tool name and tool number of the touch probe are then displayed in manual touch probe cycles.
- ▶ In the **Manual** and **Electronic Handwheel** operating modes, enter the tool number in the menu for touch probe calibration.

		Set	Reset
M4574	Select the traverse range (with M4575)	PLC	PLC
M4575	Select the traverse range (with M4574)	PLC	PLC
MP7411	Tool data in the touch probe block		
Input:	Bit 0 – 0: Use the calibrated data of the touch probe 1: Use the current tool data from the last TOOL CALL		
MP7490	Functions for traverse ranges		
Format:	%xxxx		
Input:	Bit 2 – Calibration data: touch probe for workpiece measurement: 0: One set of calibration data for all traverse ranges 1: Every traverse range has its own set of calibration data		

Multiple probe calibration data blocks

You can use the tool table to manage several blocks of touch probe calibration data. Use the tool table columns **CAL-OF1** (touch probe center offset in the reference axis), **CAL-OF2** (touch probe center offset in the minor axis) and **CAL-ANG** (spindle angle when calibrating). In the standard setting, these columns are hidden. They can be shown, however, with MP7266.28, MP7266.29 and MP7266.30. The current touch probe calibration data can be viewed and edited in the calibration menu for manual measurement:

- ▶ With MP7411 bit 1, activate the probe calibration management function in the tool table. If bit 1 = 1, bit 0 has no function.

MP7411	Tool data in the touch probe block		
Input:	Bit 1 – 0: Only one set of touch probe calibration data 1: Use the tool table to manage more than one set of touch probe calibration data		

Probing from OEM cycles

- ▶ With **FN17:SYSWRITE ID990 NR1** adjust the approach behavior. If the input value = 0, the setup clearance from MP6140 and the effective radius are accounted for. If the input value > 0, the workpiece is approached as if the effective radius and set-up clearance were zero. This function can be used, for example, for measuring small holes.

If you are using a TS 440 or TS 640 infrared touch probe, then before starting the probing cycle, you should orient the touch probe to the position at which it was calibrated. Define an M function for automatically orienting the probe to this position before probing. If the ready signal of the touch probe is already available, the touch probe is not oriented.

- ▶ In MP6161, enter the number of the M function.
- ▶ Enter the orientation angle in MP6162.
- ▶ In MP6163, enter a minimum angle difference for orienting with the defined M function..

Orientieren, wenn $MP6163 < (\text{Aktueller Spindelwinkel} - MP6162)$

MP6161 M function for orienting the touch probe before every measuring process

Input: -1: Spindle orientation directly by the NC
0: Function inactive
1 to 999: Number of the M function

MP6162 Orientation angle

Input: 0 to 359.9999 [°]

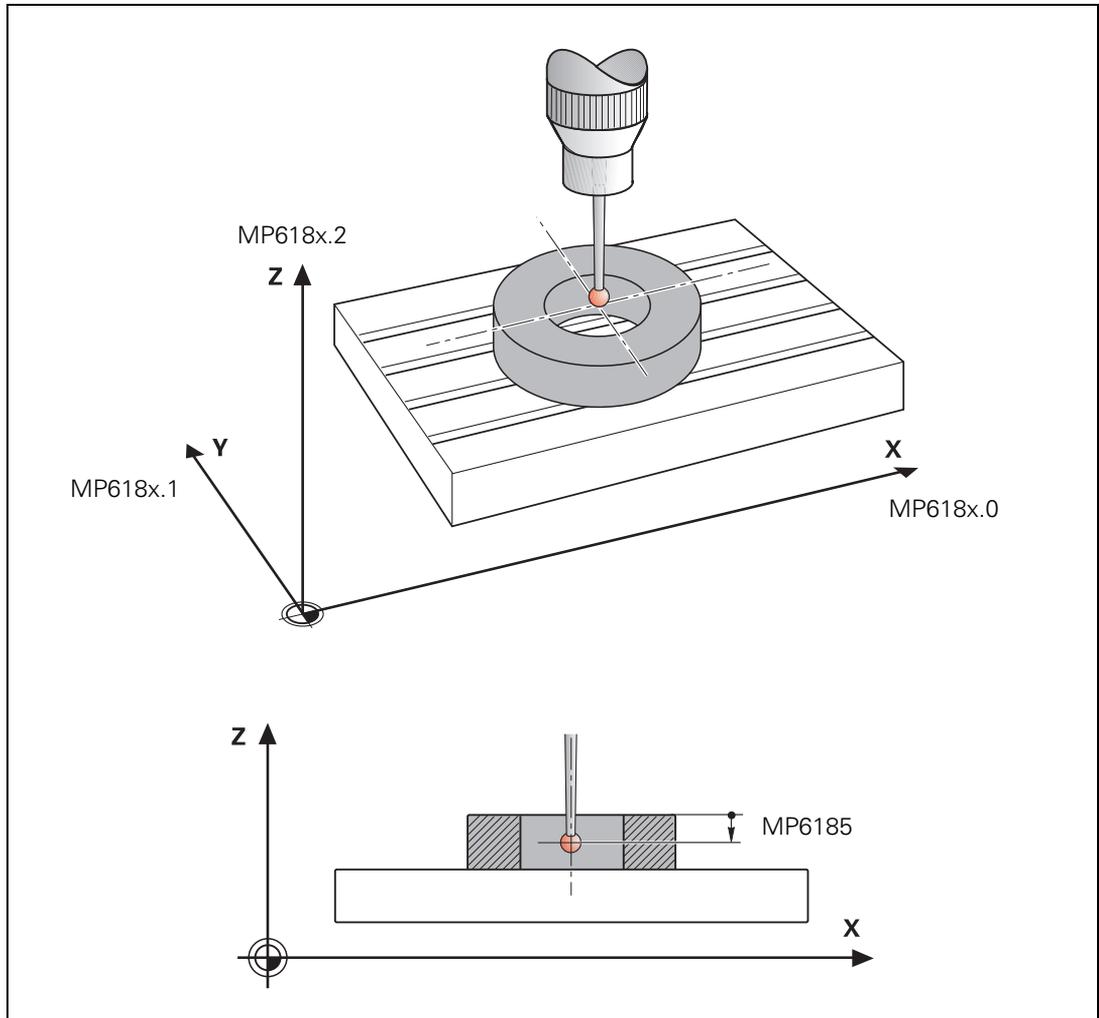
MP6163 Minimum difference between the current spindle angle and MP6162 before executing an oriented spindle stop

Input: 0 to 3.0000 [°]

Calibration

To calibrate the touch probe from within the NC program:

- ▶ In MP618x.0 and MP618x.1, enter the approximate position of the ring gauge center.
- ▶ In MP618x.2, enter the surface of the ring gauge with respect to the spindle nose. Be sure to consider the length of the touch probe or of the tool.
- ▶ In MP6185, enter the distance of the probing point below the ring's top surface.



If you probe from opposite orientations during calibration, the control stores the spindle orientation position during calibration (calculation of center offset for X and Y). You can probe at any spindle angle at a later date and the control will consider the current spindle angle and compensate for the center offset accordingly. Therefore, you do not need to orient the spindle to a specific position for probing.

The center offset is then automatically compensated during all probing processes (see the User's Manual):

- ▶ With MP6160, select whether the spindle should be oriented in a 180° rotation directly through the NC or through the PLC.
For spindle orientation by the NC you must reset M4012.
For spindle orientation by the PLC you must enter the number of the M function in MP6160. The respective position is transferred as in the "oriented spindle stop" cycle.

In the **Manual Operation** and **Electronic Handwheel** operating modes, the rotation is activated after a soft key is pressed.

Special case: tilting axes

The actual position of the spindle position encoder can vary with tilted axes. It depends on the machine's mechanical design. Since the iTNC uses the actual position of the spindle as its reference when compensating the eccentricity, it would be necessary to recalibrate the touch probe for each new tilt in position:

- ▶ In D760, enter the current offset with respect to the initial position.
- ▶ Calibrate the touch probe in the initial position.

The iTNC compensates the entered offset when compensating the eccentricity. In the initial position, D760 must equal 0.

		Set	Reset
D760	Offset in tilting axes touch probe center offset [1/10 000°]	PLC	PLC
M4012	Open the spindle control loop	PLC	PLC
MP6160	M function for probing from opposite orientations		
Input:	-1: Spindle orientation directly by NC 0: Function inactive 1 to 999: Number of the M function for spindle orientation by the PLC		
MP6180	Coordinates of the ring gauge center for Probing Cycle 2 with respect to the machine datum (traverse range 1)		
Input:	0 to +99 999.9999 [mm]		
MP6180.0	X coordinate		
MP6180.1	Y coordinate		
MP6180.2	Z coordinate		
MP6181	Coordinates of the ring gauge center for Probing Cycle 2 with respect to the machine datum (traverse range 2)		
Input:	0 to +99 999.9999 [mm]		
MP6181.0	X coordinate		
MP6181.1	Y coordinate		
MP6181.2	Z coordinate		



MP6182 **Coordinate of the ring gauge center for Probing Cycle 2 with respect to the machine datum (traverse range 3)**

Input: 0 to +99 999.9999 [mm]
MP6182.0 X coordinate
MP6182.1 Y coordinate
MP6182.2 Z coordinate

MP6185 **Distance of probing point below ring top surface during calibration**

Input: +0.001 to +99 999.9999 [mm]

Measuring tolerance

In the touch probe cycles for NC programs for automatic workpiece measurement you can enter limit values and use them for tolerance monitoring.

The following markers are set by the NC. You can evaluate them through the PLC:

- M4065: All workpiece dimensions are OK
- M4066: Workpiece must be remachined
- M4067: Workpiece to be scrapped

When probing from the NC program you can repeat measurements as desired in order to increase measurement precision:

- ▶ In MP6170 enter the number of measurements to be performed per probing process.
- ▶ Enter in MP6171 a value by which the measurement result may differ.

The mean value is formed from the measurement results. If the individual results of measurement differ by more than the tolerance defined in MP6171, an error message is output. This function can be used to detect whether a measurement has been influenced, for example, by chips.

		Set	Reset
M4065	Workpiece dimensions are OK	NC	PLC
M4066	Workpiece must be reworked	NC	PLC
M4067	Workpiece is scrap	NC	PLC

MP6170 **Number of measurements in a programmed measurement (touch probe block)**

Input: 1 to 3

MP6171 **Confidence range for programmed measurement (MP6170 > 1)**

Input: 0.002 to 0.999 [mm]

8.13.3 Measurement log in manual touch probe cycles

For every manual touch probe cycle there is one print mask per language. Standard print masks are saved on the control's hard disk before it is shipped from the factory.

With the print masks, the output format of the measurement results is defined in the %TCHPRNT.A file:

- ▶ In the MOD menu "RS232/RS422 Setup" in the **PRINT** line, define the path for the %TCHPRNT.A file:

If the path name begins with **RS232:** or **RS422:** the measurement results are transferred over the data interface.

If no path is entered, the file is saved in the root directory TNC:\.

- ▶ Start the output of the measurement data with the **PRINT** soft key in the manual probe cycle.

If you do not wish to use the standard print masks, you can create your own print masks:

- ▶ Save these masks in the language-specific paths on the PLC partition (see "Conversational language" on page 1283).

File names of the individual print masks

- Calibration for length, triggering touch probe: TSLCAL.A
- Calibration for radius, triggering touch probe: TSRCAL.A
- Basic rotation: ROT_2PTS.A
- Point measuring: DAT_SURF.A
- Corner as datum: DAT_CORN.A
- Circle center as datum: DAT_CC.A
- Basic rotation over 2 holes: ROT_2HLS.A
- Datum over 4 holes: DAT_IS4H.A
- Circle center over 3 holes as datum: DAT_CC3H.A
- Calibration for length, measuring touch probe: TBLCAL.A
- Calibration of measuring touch probe: TMCAL.A

Format of the print masks

For the text lines of the print masks:

- Lines of text must be put into quotation marks.
- Each line must be concluded with a semicolon.
- Format instructions can be given in the C programming language.
- Variables of the format instructions must be separated by commas and placed after the text string.
- Special control commands:
 - **MM** and **INCH**: Switch the display to mm or inches. The commands affect only number types that allow an inch representation.
 - **mm_display**: The following values are displayed only if under MOD **Change M/INCH** is set to **MM**.
 - **inch_display**: The following values are displayed only if under MOD **Change MM/INCH** is set to **INCH**.
 - **all_display**: The following values are displayed regardless of the setting in **Change MM/INCH** under MOD.



Variable names:

Time management		
Name	Format type	Description
HOUR	int	No. of hours from real-time clock
MIN	int	No. of minutes from real-time clock
SEC	int	No. of seconds from real-time clock
DAY	int	Day from real-time clock
MONTH	int	Month as no. from real-time clock
STR_MONTH	string	Month as string abbr. from real-time clock
YEAR2	int	Two-digit year no. from real-time clock
YEAR4	int	Four-digit year from the real-time clock

Settings of the manual measuring cycles		
Name	Format type	Description
TCH.AXIS	string	Selected probe axis
TCH.PLANEROT	double	Basic rotation angle
TS.RAD	double	Calibrated probe radius
TS.LEN	double	Calibrated stylus length
TS.OFF1	double	Calibrated center offset in reference axis
TS.OFF2	double	Calibrated center offset in minor axis
TS.RINGRAD	double	Radius of calibration ring

Results or input from the manual measuring cycles		
Name	Format type	Description
BZ	double	Reference point
BEZA	string	String reference point axis

Reference point at corner, circle, 4 holes, 3 holes on a circle		
Name	Format type	Description
BZ_HA	double	Reference point in reference axis
BZ_NA	double	Reference point in minor axis
LKALBEZ	double	Reference point entered with calibrated probe length
HA	string	Reference-axis character
NA	string	Minor-axis character
TA	string	Probe axis character

Calculated straight lines from straight-line probing		
Name	Format type	Description
GE_HA[2]	double	Straight-line axis section of reference axis
GE_NA[2]	double	Straight-line axis section of minor axis
GE_WI[2]	double	Straight-line angle

Calculated radii from circle probing		
Name	Format type	Description
RAD[8]	double	8 radii

Calculated center points from circle probing		
Name	Format type	Description
MP_HA[8]	double	Reference axis of center points
MP_NA[8]	double	Minor axis of center points

Accumulated touch points from probes		
Name	Format type	Description
AP_HA[32]	double	Touch points in reference axis
AP_NA[32]	double	Touch points in minor axis
AP_TA[32]	double	Touch points in probe axis

Example

```

"Touch probe calibration";
"_____";
%02.2d-%02.2d-%4d:%02.2d:%02.2d "Time", DAY, MONTH, YEAR4, HOUR, MIN, SEC;
Probe axis: "%s", TA;
Probe radius: "%4.31f" TS.RAD;
Probe length: "%4.31f" TS.LEN;
Ring radius: "%4.31f", TS.RINGRAD;
Center offset in reference axis: "%4.31f" TS.OFF1;
Center offset in minor axis: "%4.31f" TS.OFF2;

```



8.13.4 Measurement log in the touch probe cycles for probing from the NC program

For every touch probe cycle for probing from the NC program there is a print mask for all languages. For the HEIDENHAIN touch probe cycles a print mask is saved for every cycle on the hard disk. This print mask cannot be changed. However, you can provide an OEM touch probe cycle with a print mask of your own.

Unlike the print masks for the manual touch probe cycles, for the touch probe cycles for probing from the NC program you only need one print mask. The individual text blocks are distinguished through language code words. The text block that is defined in MP7230.0 is always output. Otherwise the syntax of the print masks is identical.

Conversational language	Language code word
English	L_ENGLISH
German	L_GERMAN
Czech	L_CZECH
French	L_FRENCH
Italian	L_ITALIAN
Spanish	L_SPANISH
Portuguese	L_PORTUGUE
Swedish	L_SWEDISH
Danish	L_DANISH
Finnish	L_FINNISH
Dutch	L_DUTCH
Polish	L_POLISH
Hungarian	L_HUNGARIA
Russian	L_RUSSIAN
Language neutral texts	L_ALL

Example

Here you see the print mask of Cycle 421 for English and German.

```
L_ENGLISH;
-----
-----"; "***** Measuring Log for Probing Cycle 421: Hole
Measuring *****";
"Date: %02.2d-%02.2d-%4d",DAY,MONTH,YEAR4;
"Time: %2d:%02.2d:%02.2d",HOUR,MIN,SEC;
"Measuring program: %S",CALL_PATH;
-----
-----";
" ";
"Nominal values:   Center in 1st axis: %6.4LF", Q273;
"                  Center in 2nd axis: %6.4LF", Q274;
"                  Diameter: %6.4LF", Q262;
"";
-----";
"";
"Given limit values:Maximum dimension for center in 1st axis: %6.4LF",
Q31;
"                  Minimum dimension for center in 1st axis: %6.4LF",
Q32;
"";
"                  Maximum dimension for center in 2nd axis: %6.4LF",
Q33;
"                  Minimum dimension for center in 2nd axis: %6.4LF",
Q34;
"";
"                  Maximum dimension for hole: %6.4LF", Q275;
"                  Maximum dimension for hole: %6.4LF", Q276;
"";
*****";
"";
"Actual values:   Center in 1st axis: %6.4LF", Q151;
"                  Center in 2nd axis: %6.4LF", Q152;
"                  Diameter: %6.4LF", Q153;
"";
-----";
"";
"Deviations: Center in 1st axis: %6.4LF", Q161;
"              Center in 2nd axis: %6.4LF", Q162;
"              Diameter: %6.4LF", Q163;
"";
*****";
"";
"Further measuring results: measuring height: %6.4LF", Q261;
"";
***** End of Measuring Log
*****";
L_GERMAN;
-----
-----"; "***** Messprotokoll Antastzyklus 421 Bohrung
```



```

messen *****";
"Datum: %02.2d-%02.2d-%4d", DAY, MONTH, YEAR4;
"Uhrzeit: %2d:%02.2d:%02.2d", HOUR, MIN, SEC;
"Messprogramm: %S", CALL_PATH;
"-----";
"";
"Sollwerte: Mitte Hauptachse: %6.4LF", Q273;
"           Mitte Nebenachse: %6.4LF", Q274;
"           Durchmesser : %6.4LF", Q262;
"";
"-----";
"";
"Vorgegebene Grenzwerte: Größtmaß Mitte Hauptachse: %6.4LF", Q31;
"                        Kleinstmaß Mitte Hauptachse: %6.4LF", Q32;
"";
"                        Größtmaß Mitte Nebenachse: %6.4LF", Q33;
"                        Kleinstmaß Mitte Nebenachse: %6.4LF", Q34;
"";
"                        Größtmaß Bohrung: %6.4LF", Q275;
"                        Kleinstmaß: %6.4LF", Q276;
"";
"*****";
"";
"Istwerte: Mitte Hauptachse: %6.4LF", Q151;
"          Mitte Nebenachse: %6.4LF", Q152;
"          Durchmesser      : %6.4LF", Q153;
"";
"-----";
"";
"Abweichungen:Mitte Hauptachse: %6.4LF", Q161;
"             Mitte Nebenachse: %6.4LF", Q162;
"             Durchmesser      : %6.4LF", Q163;
"";
"*****";
"";
"Weitere Messergebnisse: Messhöhe : %6.4LF", Q261;
"";
"***** Messprotokoll-Ende
*****";

```



8.13.5 Tool measurement

With the HEIDENHAIN tool touch probes (TT) you can measure and inspect tools. HEIDENHAIN provides standard cycles for automatic tool measurement and calibration of the TT (see the Touch Probe Cycles User's Manual).

Technical prerequisites

You need:

- TT (TT 449/TT 140)
- Central tool file TOOL.T must be active (via machine parameter)

The iTNC can save the calibration data for up to three touch probes at once:

- ▶ Use the traverse range switching function to activate the current data with M4574/M4575.
- ▶ Set MP7490 bit 3 to save three separate sets of calibration data.

MP7490 Functions for traverse ranges

Format: %xxxx

Input: Bit 3 – Calibration data: touch probe for tool measurement:
0: One set of calibration data for all traverse ranges
1: Every traverse range has its own set of calibration data

Standard measuring cycles

The TT must be mounted and interfaced:

- ▶ With MP6500 bit 0, enable the cycles for tool measurement.

MP6500 Tool measurement with TT table touch probe

Format: %xxxxxxxxxxxx

Input: Bit 0 –
0: Cycles for tool measurement disabled
1: Cycles for tool measurement not disabled



Note

In the standard measuring cycles for tool measurement, the PLC program may command a gear shift during output of the spindle speed without interrupting the cycle.



Tool radius and tool length measurement

- ▶ With MP6500 bits 1 and 2, specify whether tool radius and tool length measurements are allowed and whether individual teeth are to be measured.
- ▶ Specify in MP6500 bit 14 if tool measurement with stationary spindle is to be carried out for tools with the value 0 in the "number of teeth" column (**CUT.**) in the tool table. This can be necessary for tools with diamond teeth, for example.

MP6500

Tool measurement with TT table touch probe

Format: %xxxxxxxxxxxxxxxx

Input: Bit 1 –

0: Tool radius measurement allowed

Tool length measurement with rotating spindle

1: Tool radius measurement and individual tooth measurement disabled

Bit 2 –

0: Tool length measurement with rotating spindle (bit 1=1)

1: Tool length measurement with rotating spindle, only if a tool radius offset (**TT: R-OFFS**) has been entered in the tool table

Bit 14 – Tool measurement with number of teeth = 0

0: Tool measurement with rotating spindle

1: Tool measurement with stationary spindle

Oriented spindle stop

Spindle orientation must be active for individual tooth measurement, otherwise the tool radius measurement is subject to error:

- ▶ Define with MP6500 bit 3 whether the tool is measured with or without spindle orientation. If MP6500 bit 3 = 1 (without spindle orientation), each entry in MP6560 that does not equal 0 is not considered.
- ▶ With MP6560, specify whether the spindle is to be oriented directly via NC or through the PLC.
 - For spindle orientation directly by NC:
Reset M4012.
 - For spindle orientation by PLC:
Enter the number of the M function in MP6560.

The respective positions are transferred as in the "oriented spindle stop" cycle. M4017 is set during every spindle orientation.

MP6500

Tool measurement with TT table touch probe

Input: Bit 3 –

0: Tool measurement with spindle orientation

1: Tool measurement without spindle orientation.

Individual tooth measurement not possible. Tool radius measurement possibly faulty.

MP6560

M function for spindle orientation during individual tooth measurement

Input: –1: Spindle orientation directly by NC

0: Function inactive

1 to 999: Number of the M function for spindle orientation by PLC

Probing direction

- ▶ In MP6505.x, define the probing direction for tool radius measurement.

MP6505 Probing direction for tool radius measurement for 3 traverse ranges

Input: 0: Positive probing direction in the angle reference axis (0° axis)
1: Positive probing direction in the +90° axis
2: Negative probing direction in the angle reference axis (0° axis)
3: Negative probing direction in the +90° axis

MP6505.0 Traverse range 1
MP6505.1 Traverse range 2
MP6505.2 Traverse range 3

Offset of probe contact to the tool

- ▶ In MP6530.x enter the distance from the tool end to the top of the probe contact during tool radius measurement.
- ▶ In the **L-OFFS** field of the tool table, enter an additional tool-specific offset.

MP6530 Distance from the tool end to the top of the probe contact during tool radius measurement for 3 traverse ranges

Input: 0.001 to 99.9999 [mm]

MP6530.0 Traverse range 1
MP6530.1 Traverse range 2
MP6530.2 Traverse range 3

Safety zone

After a cycle for tool measurement starts, the tool automatically moves at the feed rate defined in MP6550 from the clearance height defined in the cycle to the limit of the safety zone.

- ▶ In MP6540.x, define a safety zone around the probe contact of the TT table touch probe.
- ▶ In MP6550, define the feed rate at which the border of the safety zone is approached.

MP6540 Safety zone around the probe contact of the TT table touch probe for pre-positioning

Input: 0.001 to 99 999.9999 [mm]

MP6540.0 Safety clearance in tool axis direction
MP6540.1 Safety clearance in the plane perpendicular to the tool axis

MP6550 Rapid traverse in probing cycle for TT table touch probe

Input: 10 to 1 000 000 [mm/min]



Probe contact

- ▶ In MP6531.x, enter the diameter (disk) or the edge length (cube) for the probe contact.
- ▶ In MP6580.x, MP6581.x and MP6582.x, enter the coordinates of the probe contact center with respect to the machine datum. After calibration, the NC internally stores the exact center of the probe contact.
- ▶ If a PLC datum shift should be included in the tool measurement, set MP6500 bit 12 = 1.

For a cube it suffices to probe from one direction:

- ▶ Set MP6500 bit 8 = 1.
- ▶ With MP6500 bit 9, specify whether the basic rotation of the cube is measured automatically or whether it should be aligned to the axes mechanically. During automatic measurement, the edge of the touch probe is probed twice and the basic rotation is calculated. All subsequent probing is done automatically at a right angle to the touch probe edge.
- ▶ With MP6500 bit 10, select how to pre-position to the starting point. If bit 10 = 1, bit 9 must equal 0.

MP6500 Tool measurement with TT table touch probe

Format: %xxxxxxxxxxxxxx

Input: Bit 7 – Reserved

Bit 8 – Probing routine

0: Probe contact is probed from several directions

1: Probe contact is probed from one direction

Bit 9 – Automatic measurement of the direction of the probe contact basic rotation (bit 8 = 1)

0: Basic rotation is not measured

1: Basic rotation of the probe element is automatically measured

Bit 10 – Probing routine (bit 8 = 1)

0: Pre-positioning to starting point in all three principal axes

1: Pre-positioning to starting point in the tool axis and in the axis of the probing direction (MP6505) (bit 9 = 0)

Bit 12 – Inclusion of the PLC datum shift

0: Do not include the PLC datum shift

1: Include the PLC datum shift

MP6531 Diameter or edge length of the TT table touch probe stylus contact for 3 traverse ranges

Input: 0.001 to 99.9999 [mm]

MP6531.0 Traverse range 1

MP6531.1 Traverse range 2

MP6531.2 Traverse range 3

MP6580.0-2 Coordinates of the TT table touch probe stylus contact center with respect to the machine datum (traverse range 1)

Input: -99 999.9999 to +99 999.9999 [mm]

MP6581.0-2 Coordinates of the TT table touch probe stylus contact center with respect to the machine datum (traverse range 2)

Input: -99 999.9999 to +99 999.9999 [mm]

MP6582.0-2 Coordinates of the TT table touch probe stylus contact center with respect to the machine datum (traverse range 3)

Input: -99 999.9999 to +99 999.9999 [mm]

Probing feed rate and spindle speed

The probing feed rate from MP6520 is used for tool measurement with a nonrotating tool.

The iTNC automatically calculates the probing feed rate and the spindle speed for tool measurement with rotating tool. The speed is calculated from the maximum permissible surface cutting speed (MP6570) and the tool radius in the tool table:

- ▶ Enter MP6500 bit 4 = 0.
- ▶ In MP6572, enter the maximum permissible speed.
- ▶ In MP6570, enter the maximum permissible surface speed of the tool edge.

The control calculates the speed from the following formula:

$$n = \frac{\text{MP6570}}{2 \cdot \pi \cdot r \cdot 10^{-3}}$$

n: Speed [min^{-1}]

MP6570 = Maximum permissible surface speed of the tool edge [m/min]

r: Tool radius [mm]

High frequency spindles often cannot function at speeds under 1000 min^{-1} :

- ▶ In this case enter MP6500 bit 4 = 1, in order to always use the lowest possible speed for that spindle. This is automatically calculated by the TNC. MP6570 and MP6572 then are without function.

The probing feed rate is calculated from the revolutions per minute and the measuring tolerance defined in MP6510.0.

- ▶ In MP6510.0, enter the maximum permissible measuring error, the "measuring tolerance."

$$v = \text{measuring tolerance} \cdot n$$

v: Probing feed rate [m/min]

Measuring tolerance: Measuring tolerance [mm] from MP6510.0 depending on MP6507

n: Speed [min^{-1}]

- ▶ With MP6507, specify the type of calculation of the probing feed rate.

MP6507=0: Calculation of the probing feed rate with constant tolerance

The measuring tolerance remains constant, regardless of the tool radius. For large tools, however, the probing feed rate becomes so small that it falls below the smallest programmable increment and becomes zero. The smaller the maximum surface cutting speed and the measuring tolerance, the sooner this effect begins.

MP6507=1: Calculation of the probing feed rate with variable tolerance

The permissible measuring tolerance changes depending on the tool radius. A probing feed rate results even for large tool radii.



The measuring tolerance is changed according to the following table:

Tool radius	Measuring tolerance
Up to 30 mm	MP6510.0
30 mm to 60 mm	2 · MP6510.0
60 mm to 90 mm	3 · MP6510.0
90 mm to 120 mm	4 · MP6510.0

MP6507=2: Constant probing feed rate

The probing feed rate remains the same, regardless of the tool radius. The absolute measuring error grows proportionally with the size of the tool radius.

$$\text{Messtoleranz} = \frac{r}{5 \text{ [mm]}} \cdot \text{MP6510.0}$$

r: Tool radius [mm]

MP6510.0: Max. permissible measuring error [mm]

$$v = \frac{\text{MP6570} \cdot \text{MP6510}}{2 \cdot \pi \cdot 10^{-3}}$$

v: Probing feed rate [m/min]

MP6570: Maximum permissible surface speed of the tool edge [m/min]

MP6500 Tool measurement with TT table touch probe

Format: %xxxxxxxxxxxxxx

Input: Bit 4 –

0: Automatically determine speed

1: Always use minimum spindle speed

MP6507 Calculation of the probing feed rate

Input: 0: Calculation of the probing feed rate with constant tolerance

1: Calculation of the probing feed rate with variable tolerance

2: Constant probing feed rate

MP6520 Probing feed rate for tool measurement with non-rotating tool

Input: 1 to 10 000 [mm/min]

MP6570 Max. permissible surface cutting speed at the tooth edge

Input: 1.0000 to 129.0000 [m/min]

MP6572 Maximum permissible speed during tool measurement

Input: 1 to 1000 [min⁻¹]

0: 1000 [min⁻¹]

Individual tooth measurement

The TNC attempts to maintain the tolerance from MP6510.0 during the tooth search for individual tooth measurement. At the same time MP6510.0 is used to calculate the probing feed rate.

If the tolerance from MP6510.0 cannot be maintained during the tooth search, e.g. due to the missing spindle accuracy, the TNC attempts to maintain the tolerance from MP6510.1. If this also cannot be maintained, the error message **Tolerance in MP6510 too small** appears.

▶ Enter the first maximum measuring error in MP6510.0.

▶ Enter the second maximum measuring error in MP6510.1.

MP6510 **Maximum permissible measuring error for tool measurement with rotating tool**

Input: 0.002 to 0.999 [mm]
MP6510.0 First measurement error
MP6510.1 Second measurement error

Monitoring of the rotary axes and secondary linear axes

To ensure that the rotary axes and the secondary linear axes are always in a defined position during the tool measuring cycles:

▶ In MP6585, enter the axes to be monitored.

▶ In MP6586.x, enter the reference coordinate at which the axis should be located during the tool measuring cycles.

If, during activated monitoring, the nominal position does not match the position from MP6586.x, an error message is displayed.

MP6585 **Monitoring the position of the rotary and additional linear axes during the tool measurement cycles**

Format: %xxxxxx
Input: 0: Axis is not monitored
 1: Axis is monitored
 Bit 0 – A axis
 Bit 1 – B axis
 Bit 2 – C axis
 Bit 3 – U axis
 Bit 4 – V axis
 Bit 5 – W axis

MP6586 **Ref. coordinate for monitoring the position of the rotary and additional linear axes during the tool measurement cycles**

Input: –99 999.9999 to +99 999.9999 [mm] or [°]
MP6586.0 A axis
MP6586.1 B axis
MP6586.2 C axis
MP6586.3 U axis
MP6586.4 V axis
MP6586.5 W axis



Tool measurement in a tilted coordinate system

- ▶ If the tool is to be measured in a tilted position other than that in which the tool touch probe was calibrated, set MP6500 bit 13 = 1.

MP6500 Tool measurement with TT table touch probe

Format: %xxxxxxxxxxxxx

Input: Bit 13

0: Tool is measured in the tilt position in which the tool touch probe was also calibrated

1: Tool is measured in another tilt position



Attention

If the tool is not measured in the same tilt position as that in which the tool touch probe was calibrated, ensure that the tool is perpendicular to the contact plate!

Tool breakage

- ▶ With MP6500 bits 5 and 6, specify whether the NC program should stop when the breakage tolerance is exceeded. M4063 is always set when the breakage tolerance is exceeded.
- ▶ With bit 11, specify whether the result of "tool checking" measurement is to be entered in the tool table.

MP6500 Tool measurement with TT table touch probe

Format: %xxxxxxxxxxxxx

Input: Bit 5 – NC stop during "tool checking"

0: The NC program is not stopped when the breakage tolerance is exceeded

1: If the breakage tolerance is exceeded, the NC program is stopped and the error message "Tool broken" is displayed.

Bit 6 – NC stop during "tool measurement"

0: The NC program is not stopped when the breakage tolerance is exceeded.

1: If the breakage tolerance is exceeded, the NC program is stopped and the error message "Touch point inaccessible" is displayed.

Bit 11 – "Tool checking" and changing in the tool table

0: After "tool checking" the tool table is changed

1: After "tool checking" the tool table is not changed

M functions before and after tool measurement

Machine parameters **MP6562.0** and **MP6562.1** can be used to run M functions before and after a tool measurement cycle (TT cycle). **MP6562.0** activates an M function before the beginning of a cycle, **MP6562.1** after the end of a cycle. You need to program M-function macros in an NC block to be able to run them.

If the M-function macros are used to make changes that are not automatically undone at an internal stop, you must ensure that the machines are restored to a consistent state (e.g. through a suitable cancel macro).

MP6562 M function before and after tool measurement

Input: 0 to 999 [M function macro]
-1: Function inactive

Markers in the PLC

M4060 is set if a cycle for tool measurement is started.

M4061 displays whether a cycle was activated for tool measurement or for tool checking.

M4062 and M4063 are set if during tool checking one of the entered tolerances was exceeded. The tool is locked.

The markers M4050, M4051, M4052, M4053, M4055 and M4056 function as in the standard cycles. You must enable the cycles for tool measurement with M4055. For spindle orientation directly by the NC (MP6560 = -1), you must reset M4012.

		Set	Reset
M4060	Cycle for tool measurement started	NC	NC
M4061	0: Measure the tool 1: Check the tool	NC	NC
M4062	0: Wear tolerance not exceeded 1: Wear tolerance exceeded	NC	NC/PLC
M4063	0: Breakage tolerance not exceeded 1: Breakage tolerance exceeded	NC	NC/PLC



8.14 Special functions for laser cutting machines

You can activate special functions to interface the iTNC to laser cutting machines and water jet machines.

8.14.1 Analog voltage output

If you do not need the analog output S of the CMA-H for the spindle, you can define other functions for this output:

- ▶ With MP3011, select the function of analog output S. If MP3010 > 3, MP3011 has no effect.

MP3011 Function of analog output S, if MP3010 < 3

Input: 0: No special function
 1: Voltage is proportional to the current contouring feed rate, depending on MP3012
 2: Voltage is defined as through Module 9130
 3: Voltage is defined through M functions (M200 to M204)

Voltage proportional to the contouring feed rate, MP3011 = 1

A voltage proportional to the current contouring feed rate is output:

- ▶ In MP3012, enter the feed rate achieved when a 10 V analog voltage is output.

MP3012 Feed rate from output of an analog voltage of 10 V, MP3011 = 1

Input: 0 to 300 000 [mm/min]

Voltage from the PLC, MP3011 = 2

The voltage that you have defined with Module 9130 is output.

Definition of the voltage through M functions, MP3011 = 3

The voltage to be output is defined through M functions M200 to M204:

- ▶ Set MP3011 = 3, otherwise the M functions described above are not available.

The M functions are executed synchronously to the positioning blocks and are effective at the beginning of the positioning blocks.

Direct output of the programmed voltage: M200 V...

The iTNC outputs the value after M200 V... as a voltage.

Input: 0 to 9.999 [V]

Duration of effect: M200 V... is effective until a new voltage is output with M200 to M204.

Voltage output varies with the distance: M201 V...

The iTNC outputs the voltage as a function of the traversed distance. Starting from the active voltage, the iTNC increases or decreases the voltage linearly to the value programmed behind M201 V.

Input: 0 to 9.999 [V]

Duration of effect: M200 V... is effective until a new voltage is output with M200 to M204.

Voltage output varies with the velocity: M202 FNR

The iTNC outputs the voltage as a function of the velocity:

- ▶ In MP3013.x and MP3014.x, define up to three characteristic curves in a table.

In the table, certain analog voltages are assigned to certain feed rates:

- ▶ With M202 FNR select the curve in which the iTNC finds the voltage to be output.

Input: 1 to 3

Duration of effect: M202 FNR is effective until a new voltage is output with M200 to M204.

You can enter up to four kink points per curve in the table. The values to be distributed are interpolated linearly between the kink points. The first kink point must start with the input value zero. For the following kink points of the curve the input values must rise steadily. The iTNC detects the beginning of a new curve from the input value zero.

Example:

Velocity		Voltage		Curve
MP3013.0	0	MP3014.0	0	1
MP3013.1	25	MP3014.1	0	
MP3013.2	500	MP3014.2	4.5	
MP3013.3	1000	MP3014.3	9.999	
MP3013.4	0	MP3014.4	0	2
MP3013.5	10 000	MP3014.5	9.999	
MP3013.6	0	MP3014.6	0	3
MP3013.7	50	MP3014.7	0.5	
MP3013.8	300	MP3014.8	1.5	
MP3013.9	5000	MP3014.9	9.999	
MP3013.10	0	MP3014.10	0	Not used
MP3013.11	0	MP3014.11	0	

MP3013.x Characteristic curve kink points (velocity) for output of the analog voltage with M202

Input: 10 to 300 000 [mm/min]

MP3014.x Characteristic curve kink points (voltage) for output of the analog voltage with M202

Input: 0.000 to 9.999 [V]



Voltage output varies with the time (time-dependent ramp):
M203 V... TIME...

The iTNC outputs the voltage as a function of the time. Starting from the active voltage, the iTNC increases or decreases the voltage linearly in the time programmed behind TIME to the value programmed behind V.

Input: Voltage V: 0 to 9.999 [V]

TIME: 0 to 1.999 [sec]

Duration of effect: M203 V... TIME... is effective until a new voltage is output with M200 to M204.

Voltage output varies with the time (time-dependent pulse):
M204 V... TIME...

The iTNC outputs the value programmed after V... as a pulse. The duration of the pulse is specified with "TIME..."

Input: Voltage V: 0 to 9.999 [V]

TIME: 0 to 1.999 [sec]

Duration of effect: M204 V... TIME... is effective until a new voltage is output with M200 to M204.

8.14.2 Graphic simulation (without TOOL CALL)

Graphic simulation is also available on machines that operate without tool definition (e.g., water jet and laser cutting machines):

- ▶ In MP7315, specify the tool radius for the graphic simulation.
- ▶ In MP7316, define the penetration depth of the simulated tool.
- ▶ Use M functions to mark the program sections to be simulated and define the functions in MP7317.x.

Furthermore, by entering in MP7312 as small a value as possible between 1 and 8, you should be able to detect an acceleration of the graphic simulation (the 3-D graphics) in the Test Run operating mode. However, HEIDENHAIN instead recommends entering the real values of the corresponding tooth lengths in the LCUTS column of the tool table.

MP7312 Limitation of the tooth length LCUTS

Input: 0 = No limitation, infinitely long tooth length
 > 0: Tooth length = 2 * tool radius * MP7312

MP7315 Tool radius for graphic simulation without TOOL CALL

Input: 0.0000 to 99 999.9999 [mm]

MP7316 Penetration depth of the tool

Input: 0.0000 to 99 999.9999 [mm]

MP7317 M function for graphic simulation

MP7317.0 Beginning of graphic simulation

Input: 0 to 88

MP7317.1 Interruption of graphic simulation

Input: 0 to 88



8.14.3 Program stop for M functions and TOOL CALL S

TOOL CALL S means a TOOL CALL in which only one spindle speed was programmed.

For TOOL CALL S and also in the **PROGRAM RUN, FULL SEQUENCE** and **PROGRAM RUN, SINGLE BLOCK** modes, the output of an M function interrupts the program run until you confirm execution with M4092.

However, on applications such as laser cutting machines, the program should not be interrupted:

► With MP7440 bit 2 and MP3030 bit 0, specify whether the program run should be interrupted.

If you deselect the program stop, you must not perform the following functions during output:

- PLC positioning
- Datum shift
- Oriented spindle stop
- Limit switch range switchover



Attention

Do not use this function on milling machines and boring mills!

MP3030

Behavior of the spindle

Input:

Bit 0 –

0: Axis stop for TOOL CALL S...

1: No axis stop for TOOL CALL S...

MP7440

Output of M functions

Format:

%xxxxx

Input:

Bit 2 – Program stop with M functions:

0: Program stop until acknowledgment of the M function

1: No program stop, no waiting for confirmation



8.15 Tool changer

You control the tool changer through PLC outputs.

If the tool changer is to be driven by controlled axes, use PLC axes. See page 630. You can also control the tool changer through proximity switches:

- ▶ Save the information about the tool in the tool table and the information about the tool changer in the pocket table.

Tool management (replacement tool, tool life, etc.) is handled by the NC. Markers and words provide you with the information necessary for driving the tool changer.

8.15.1 Tool and pocket table

You can edit the tool table in the machining modes of operation:

- ▶ Ensure that the tool table and pocket table are neither locked nor protected via MP7224.x. See page 1406.
- ▶ Press the TOOL TABLE soft key.

From the tool table you can call the pocket table (see the User's Manual):

- ▶ Ensure that the POCKET TABLE soft key is not hidden by MP7263 bit 0.
- ▶ Press the POCKET TABLE soft key.

The current tool table is TOOL.T, and the pocket table is TOOL_P.TCH. Both files are saved in the root directory TNC:\.

Definition of the tool and pocket table:

- ▶ In MP7266.x, specify the fields of the tool table that are to be displayed and the sequence in which they appear.
- ▶ In MP7267.x, specify the fields of the pocket table that are to be displayed and the sequence in which they appear.
- ▶ Ensure that the tool table and pocket table are neither locked nor protected via MP7224.x. See page 1406.
- ▶ In MP7260, specify the number of tools in the tool table.
 - If $MP7260 = 0$, no tool table is used (TOOL.T does not exist). In this case, you must program the tool length and radius in the NC program with a **TOOL DEF** block (see the User's Manual). There is no automatic tool management.
- ▶ If you are only using one tool magazine, set the number of pockets in MP7261.0 and enter $MP7261.1-15 = 0$. If you use multiple tool magazines, see "Managing multiple tool magazines" on page 1590.
 - If $MP7261.0-15 = 0$, no pocket table is generated.

With Modules 9092, 9350 (9093), 9351 (9094) and 9096 you can read the tool and pocket tables and overwrite them.

The operating system cyclically saves tables to the hard disk approximately every five seconds. In case of a power failure, the control automatically blocks access to the hard disk, in order to protect it from damage. If tables or table contents are changed, and the rare case of a power failure occurs within the next five seconds before the next automatic saving, the last changes can be lost. Should this cause problems on your machines, then more frequent saving of tables can be forced by setting MP4020 bit 15. If MP4020 bit 15 = 1, then tables are saved each time they are closed. However, if tables are changed very frequently (e.g. via PLC program or submit job), then this can have a negative effect on the computing performance, since the number of accesses to the hard disk would increase significantly.

If an input field is open in the editor at the time the modules are called, this field is closed automatically.

The status display shows the current tool data.

With MP7263 bit 1 you configure the output of the column in the pocket table during backup and during conversion from binary format to ASCII.

By setting MP7263 bit 2, the "Edit ON/OFF" soft key can be hidden when the pocket table is displayed. This makes it possible to prevent manual editing of the pocket table.

MP7263 bit 3 can be used to hide the RESET POCKET TABLE and RESET COLUMN T soft keys.

MP7260 to MP7267 can also be overwritten by the PLC or the LSV2 protocol.

The bits 4, 5 and 6 of machine parameter MP7263 are used to define settings for deleting tools and their index entries. These settings apply to tools in the pocket table.

During control start-up the prototype pocket table (**PLC:\PROTO\PROTOTYP.TCH**) is used when a pocket table is created. The standard pocket table is only created if no prototype pocket table exists.

If bit 12 is set in MP7682, the error message "Tool radius too large" is suppressed if $R2 > R$ for a tool in the tool table. This might be necessary if barrel cutters are used. Please keep in mind that the cutter form compensation with LN blocks does not work with this cutter shape.



MP7260 Number of tools in the tool table

Input: 0 to 30 000

MP7261.0-15 Number of pockets in the tool magazine 1 to 16

Input: 0 to 9999

MP7263 Pocket table

Format: %xxxxxxx

Input: Bit 0 –

0: POCKET TABLE soft key is shown

1: POCKET TABLE soft key is hidden

Bit 1 – Output of the pocket table for file functions

0: Output only the displayed columns

1: Output all columns

Bit 2 – Show the "Edit ON/OFF" soft key in the pocket table

0: Display soft key

1: Do not display soft key

Bit 3 – Display the RESET POCKET TABLE and RESET

COLUMN T soft keys in the pocket table

0: Display soft key

1: Do not display soft key

Bit 4 - Deletion possible for a tool that is in the pocket table.

Deletion must be confirmed.

0: Deletion impossible

1: Deletion possible (with confirmation)

Bit 5 – Deletion of a tool possible even without confirmation (if bit 4 = 1)

0: Deletion not possible without confirmation

1: Deletion possible without confirmation

Bit 6 – Deletion of index entries of a tool behaves like deletion of a tool. The settings of bits 4 and 5 also apply to the index entries if bit 6 is set.

0: Deletion always impossible

1: Deletion possible depending on settings in bits 4 and 5

MP7266 Elements of the tool table

Input: 0 = no display

1 to 99 = position in the tool table

MP7682 MP with multiple function

Input: Bit 12 – Error message "Tool radius too large" if R2 > R

0 = Error message displayed

1 = Error message is not displayed

MP4020 PLC functions

Input: Bit 15 – Saving the changes to tables to memory medium (HDR, SSDR)

0: Cyclic saving of tables (behavior same as up to and including 60642x-01 SP 04)

1: Save tables each time they are closed

MP	Meaning	Column name	Value range	Column width	Data type
MP7266.0	Alphanumeric tool name	NAME	32 ^a characters	32	String
MP7266.1	Tool length	L	±99999.9999 [mm/inches]	11	FLOAT
MP7266.2	Tool radius	R	±99999.9999 [mm/inches]	11	FLOAT
MP7266.3	Tool radius 2 for toroidal cutter	R2	±99999.9999 [mm/inches]	11	FLOAT
MP7266.4	Oversize in tool length	DL	±999.9999 ^b [mm/inch]	9	FLOAT
MP7266.5	Oversize in tool radius	DR	±999.9999 ^b [mm/inch]	9	FLOAT
MP7266.6	Oversize in tool radius 2	DR2	±999.9999 ^b [mm/inch]	9	FLOAT
MP7266.7	Locked tool?	TL	'L' or ''	2	UINT
MP7266.8	Replacement tool	RT	0 – 32767	5	FLOAT
MP7266.9	Maximum tool age (M4543)	TIME1	0 – 9999 [min]	5	FLOAT
MP7266.10	Maximum tool life TOOL CALL	TIME2	0 – 9999 [min]	5	FLOAT
MP7266.11	Current tool age	CUR.TIME	0 – 99999 [min]	8	FLOAT
MP7266.12	Comment on the tool	DOC	32 ^a characters	32	STRING
MP7266.13	Number of tool teeth	CUT	0 – 99	4	UINT
MP7266.14	Wear tolerance for tool length	LTOL	0 – 0.9999	7	FLOAT
MP7266.15	Wear tolerance for tool radius	RTOL	0 – 0.9999	7	FLOAT
MP7266.16	Cutting direction of the tool	DIRECT	'+' or '-', 0 or 255	7	?
MP7266.17	Additional information for PLC (Module 9093)	PLC	%00000000 – %11111111, 0 – 255	9	BIN
MP7266.18	Tool offset: length	TT: LOFFS	±99999.9999 [mm/inches]	11	FLOAT
MP7266.19	Tool offset: radius	TT: ROFFS	±99999.9999 [mm/inches]	11	FLOAT
MP7266.20	Breakage tolerance for tool length	LBREAK	0 – 3.2767	7	FLOAT
MP7266.21	Breakage tolerance for tool radius	RBREAK	0 – 0.9999	7	FLOAT
MP7266.22	Tooth length	LCUTS	0 – +99999.9999	10	FLOAT
MP7266.23	Plunge angle	ANGLE	0 – 90.0000	7	FLOAT

MP	Meaning	Column name	Value range	Column width	Data type
MP7266.24	Tool type <ul style="list-style-type: none"> ■ BOR: Boring tool ■ BCKBOR: Back-boring tool ■ CENT: NC spot drill / center drill ■ CSINK: Countersinking tool ■ DRILL: Drilling tool ■ MILL: Milling cutter ■ MILL_R: Rough cutter ■ MILL_F: Finishing cutter ■ MILL_RF: Rough and finishing cutter ■ MILL_FD: Floor finishing cutter ■ MILL_FS: Side finishing cutter ■ MILL_FACE: Face-milling cutter ■ REAM: Reamer ■ TAP: Tapping tool ■ GF: Thread miller ■ GSF: Thread miller with chamfer ■ EP: Thread miller for single threads ■ WSP: Thread miller with indexable insert ■ BGF: Thriller ■ ZBGF: Circular thread miller ■ TSINK: Piloted counterbore 	TYPE	10 characters	10	STRING



MP	Meaning	Column name	Value range	Column width	Data type
MP7266.25	Tool material	TMAT	16 characters	16	STRING
MP7266.26	Cutting data table	CDT	16 characters	16	STRING
MP7266.27	PLC value	PLC-VAL	±99999.9999	11	FLOAT
MP7266.28	Probe center misalignment in ref. axis	CAL-OF1	±99999.9999	11	FLOAT
MP7266.29	Probe center misalignment in minor axis	CAL-OF2	±99999.9999	11	FLOAT
MP7266.30	Spindle angle during calibration	CAL-ANG	0.0 – 360.0000	8	FLOAT
MP7266.31	Tool type for pocket table	PTYP	0 – 99	2	UINT
MP7266.32	Maximum shaft speed [rpm]	NMAX	- or 0 – 999999	6	FLOAT
MP7266.33	Retract tool	LIFTOFF	Y or N	2	String
MP7266.34	PLC value	P1	±99999.9999	11	FLOAT
MP7266.35	Input range: -99999.9999 to +99999.9999	P2	±99999.9999		FLOAT
MP7266.36		P3	±99999.9999		FLOAT
MP7266.37	Additional kinematics description for tool and tool carrier	KINEMATICS	16 characters	16	STRING
MP7266.38	Point angle for DRILL and CSINK	T-ANGLE	±180.0000	9	FLOAT
MP7266.39	Thread pitch for TAP	PITCH	0 – 99999.99	10	FLOAT
MP7266.40	Control strategy name for AFC (Adaptive Feed Control)	AFC	10 characters	10	String
MP7266.41	Tool value or tool radius R2	R2TOL	0 – 0.9999	5	FLOAT
MP7266.42	Compensation value table for 3DToolComp	DR2TABLE	16 characters	16	STRING
MP7266.43	Time stamp during tool changing	LAST_USE	16 characters	16	STRING
MP7266.44	Active Chatter Control on/off	ACC	'0' or '1'	2	UINT

- a. In SW 606 42x-02 increased from 16 to 32 characters
- b. In SW 606 42x-02 increased from ±99.9999 [mm/inch] to ±999.9999 [mm/inch]



MP7267	Elements of the pocket table
Input:	0: No display 1 to 99: Position in the pocket table
MP7267.0	Tool number (T)
MP7267.1	Special tool (ST)
MP7267.2	Fixed pocket (F)
MP7267.3	Locked pocket (L)
MP7267.4	PLC status (PLC)
MP7267.5	Tool name (TNAME)
MP7267.6	Comment on the tool (DOC)
MP7267.7	Tool type for pocket table (PTYP)
MP7267.8	Value 1 (P1)
MP7267.9	Value 2 (P2)
MP7267.10	Value 3 (P3)
MP7267.11	Value 4 (P4)
MP7267.12	Value 5 (P5)
MP7267.13	Reserve pocket (RSV)
MP7267.14	Pocket above locked (LOCKED_ABOVE)
MP7267.15	Pocket below locked (LOCKED_BELOW)
MP7267.16	Pocket at left locked (LOCKED_LEFT)
MP7267.17	Pocket at right locked (LOCKED_RIGHT)
MP7267.18	S1 value (P6)
MP7267.19	S2 value (P7)



Note

The **TNAME** (tool name) column contains the name of the tool from the tool table and therefore cannot be edited. For indexed tools, the name of the tool is entered with the index 0.

Module 9092 Search for an entry in the tables selected for execution (.T/.D/.TCH)

Prerequisite for table: M status must be set.

The entry or value sought is given as a natural number, shifted by the number of decimal places that can be entered.

As return code the function replies with the number of the line in which the value was found.

It is possible, for example, to look for the vacant pocket (corresponds to T0) in the pocket table.

If you wish to look for more occurrences of the same value, you must enter the line number of the last occurrence plus one as the starting line.

Call:

```
PS    B/W/D/K <File type>
        0: *.T file (tool table)
        1: *.D file (datum table)
        2: *.TCH file (pocket table)
PS    B/W/D/K <Element value>
PS    B/W/D/K <Element number>
```

***.T file**

- 0: Tool length (L)
- 1: Tool radius (R)
- 2: Reserved
- 3: Replacement tool (RT); (-1= not defined)
- 4: Reserved
- 5: TIME 1
- 6: TIME 2
- 7: CURRENT TIME
- 8: Tool radius 2 (R2)
- 9: Oversize for tool length (DL)
- 10: Oversize for tool radius (DR)
- 11: Oversize for tool radius 2 (DR2)
- 12: Tool locked (TL); (0: No, 1: Yes)
- 13: Number of the tool teeth (CUT)
- 14: Wear tolerance for tool length (LTOL)
- 15: Wear tolerance for tool radius (RTOL)
- 16: Cutting direction of the tool (DIRECT); (0:+; 1: -)
- 17: PLC status (PLC)
- 18: Tool offset for tool length (TT:LOFFS)
- 19: Tool offset for radius (TT:ROFFS); (\$7FFF FFFF = R)
- 20: Breakage tolerance for tool length (LBREAK)
- 21: Breakage tolerance for tool radius (RBREAK)
- 22: Tooth length (LCUTS)
- 23: Plunge angle (ANGLE)
- 24: Tool number
- 25: Tool index
- 26: PLC value (PLC-VAL)
- 27: Probe center offset in reference axis (CAL-OF1)
- 28: Probe center offset in minor axis (CAL-OF1)
- 29: Spindle angle during calibration (CAL-ANG)
- 30: Tool type for pocket table (PTYP)
- 31: Maximum shaft speed [rpm] (NMAX)
- 32: Retract tool (LIFTOFF)
- 33: Value for PLC (P1)



- 34: Value for PLC (P2)
- 35: Value for PLC (P3)
- 36: Point angle for tool types DRILL and CSINK (T-ANGLE)
- 37: Thread pitch for tool type TAP (PITCH)
- 38: Wear tolerance for tool radius 2 (R2TOL)
- 39: Time stamp (LAST_USE)
- 40: Numerical value for type of tool (TYP)
- 41: ACC active/inactive

***.D file:**

- 0: Shift in axis 1 (\$7FFF FFFF = -)
- 1: Shift in axis 2 (\$7FFF FFFF = -)
- 2: Shift in axis 3 (\$7FFF FFFF = -)
- 3: Shift in axis 4 (\$7FFF FFFF = -)
- 4: Shift in axis 5 (\$7FFF FFFF = -)
- 5: Shift in axis 6 (\$7FFF FFFF = -)
- 6: Shift in axis 7 (\$7FFF FFFF = -)
- 7: Shift in axis 8 (\$7FFF FFFF = -)
- 8: Shift in axis 9 (\$7FFF FFFF = -)

***.TCH file:**

- 0: Tool number (T);
(-1, if no tool is entered)
- 1: Special tool (ST);
(0: No, 1: Yes)
- 2: Fixed pocket (F);
(0: No, 1: Yes)
- 3: Locked pocket (L);
(0: No, 1: Yes)
- 4: PLC status (PLC)
- 5: Tool type for pocket table (PTYP)
- 6: Reserve pocket (RSV)
- 7: Value 1 (P1)
- 8: Value 2 (P2)
- 9: Value 3 (P3)
- 10: Value 4 (P4)
- 11: Value 5 (P5)

PS B/W/D/K <Line number for beginning of search>

CM 9092

PL B/W/D <Line number (in case of error -1)>

PL B/W/D <Error number>

- 0: No error. Element was found.
- 1: Call was not in a submit or spawn job
- 2: File type does not exist
- 3: No file of the entered type was found with M status
- 4: Line number not in file
- 5: Incorrect element number
- 6: Element value not found

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022



Marker	Value	Meaning
W1022	2	Invalid file type or element number
	7	Error while reading from the file
	20	Call was not in a submit or spawn job
	36	Error while opening the file



Module 9093 Read data from tables selected for program run (.T/.D/.TCH)

Prerequisite for table: M status must be set.

You transfer the line number (i.e. tool number for *.T, vector number for *.D or pocket number for *.TCH) and the number of the element to be read.

The value is given as a natural number, shifted by the number of decimal places that can be entered.

The module must be called in a submit job or spawn job.

Call:

PS B/W/D/K <File type (see Module 9092)>
PS B/W/D/K <Line number>
PS B/W/D/K <Element number (see Module 9092)>
CM 9093
PS B/W/D <Element value>
PL B/W/D <Error number>

0: No error

1: Call was not in a submit job

2: File type does not exist

3: No file of the entered type was found with M status

4: Line number not in file

5: Incorrect element number

Error code:

Marker	Value	Meaning
M4203	0	No errors
	1	See above for errors

Module 9094 Write data into a tool and datum table

Prerequisite for table: M status must be set.

You transfer the line number and the element number of the element to be overwritten.

The value is given as a natural number, shifted by the number of decimal places that can be entered.

The execution of Module 9094 reinitializes the geometry.

The module must be called in a submit job or spawn job.

Call:

PS B/W/D/K <File type (see Module 9092)>

PS B/W/D/K <Line number>

PS B/W/D/K <Element number (see Module 9092)>

PS B/W/D/K <Element value>

CM 9094

PL B/W/D <Error number>

0: No error. Element was written.

1: Call was not in a submit or spawn job

2: File type does not exist

3: No file of the entered type was found with M status

4: Line number not in file

5: Incorrect element number

6: Element value is outside the permissible range

Error code:

Marker	Value	Meaning
M4203	0	No errors
	1	See above for errors



Module 9096 Delete a line from the tool table

You remove a line from the tool table and cancel any link with a replacement tool.

The module must be called in a submit job or spawn job.

Call:

PS B/W/D/K <Tool number / pocket number>

PS B/W/D/K <Mode>

Bit 0: Delete entries in pocket table

0: Pocket table remains unchanged

1: Tool number in pocket table is deleted

Bit 1: Tool or pocket number

0: Transferred value = tool number

1: Transferred value = pocket number

CM 9096

Error code:

Marker	Value	Meaning
M4203	0	Line was deleted
	1	Error code in W1022
W1022	2	Invalid pocket or tool number
	21	Module was not called in a spawn or submit job
	24	File error

Module 9350 Read data from the tool table

Module 9350 reads the contents of a cell in the tool table with the status **M**.
The value is read as an integer value.

Call:

PS B/W/D/K <Tool number>

PS B/W/D/K <Tool index>

≤ 0: Main entry

PS B/W/D/K <Element number (see Module 9092)>

CM 9350

PL B/W/D <Element value>

PL B/W/D <Error number>

0: No error, element value was read

1: Module was not called in a spawn or submit job

2: File type does not exist

3: No tool table with status **M**

4: Line number does not exist

5: Incorrect element number

Error code:

Marker	Value	Meaning
M4203	0	Element value read
	1	Error code in W1022
W1022	2	Incorrect element number
	7	Line number does not exist
	20	Module was not called in a spawn or submit job
	36	No tool table with status M
	41	Data type for the element is not supported



Module 9351 Write data to tool table

Module 9351 writes the contents of a cell in the tool table with the status **M**. The value must be given as an integer value.

Call:

PS B/W/D/K <Tool number>

PS B/W/D/K <Tool index>

-1: Write all indexes of a tool

PS B/W/D/K <Element number>

See Module 9350

PS B/W/D/K <Element value>

CM 9351

PL B/W/D <Error number>

0: No error, element value was written

1: Module was not called in a spawn or submit job

2: File type does not exist

3: No tool table with status **M**

4: Line number does not exist

5: Incorrect element number

6: Element value is out of range

7: Error while writing to the file

Error code:

Marker	Value	Meaning
M4203	0	Element value written
	1	Error code in W1022
W1022	2	Incorrect element number
	7	Line number does not exist
	20	Module was not called in a spawn or submit job
	36	No tool table with status M

Definition of the tool magazine using magazine rules

You can usually place more than one tool type in a tool magazine. Depending on the tool, however, surrounding pockets may have to be locked. The ASCII file *.TCR contains magazine rules for such definitions.

- ▶ In OEM.SYS, use the keyword **TCHRULES** = to enter the name and path of the ASCII file *.TCR.

or

- ▶ Use Module 9343 to compile the ASCII file with the magazine rules.
- ▶ Create the file *.TCR with the following keywords.

Keyword	Meaning
[magazine]a	All the following rules apply to the tool magazine a . Example: [magazine]4
[search]a = a b ...	Definition of the search sequence for the tool type a . The tool type named first (here a) is searched for first, then the next tool type (separated by a space) (here b). You may only enter tool types in whose pockets tools of the type a may be placed! Example: [search]1 = 1 2
[tooltype]a	All the following rules apply to the tool type a . 99 tool types can be specified. Example: [tooltype]2
[place]a = bx cx ...	Description of the pocket a . Define the pockets (here b and c) that are affected by pocket a when the current tool type (keyword [tooltype]a) is placed there. Immediately after the pocket number the identifier x follows, indicating which area of the pocket is affected (r = right area, l = left area, b = bottom area, a = top area) Example: [place]21 = 20r 22l



Note

You must define the search sequence for each magazine separately, not just globally at the beginning of the *.TCR file. If the search sequence is to be defined the same for each magazine, then it must be repeated after each magazine definition.

The following columns in the pocket table are used for magazine rules:

- **PTYP:** Tool type
- **RSV:** Reserved pocket
- **P1 to P5:** Values 1 to 5 for evaluation in the PLC (e.g., axis positions of a pocket in the box magazine). Module 9304 copies the values to a word memory.
- **LOCKED_ABOVE:** Pocket above is locked
- **LOCKED_BELOW:** Pocket below is locked
- **LOCKED_LEFT:** Pocket to the left is locked
- **LOCKED_RIGHT:** Pocket to the right is locked

With **FN18: SYSREAD ID51** the cells of the pocket table can be read.

Module 9340 searches a magazine for vacant, reserved or unavailable pockets.

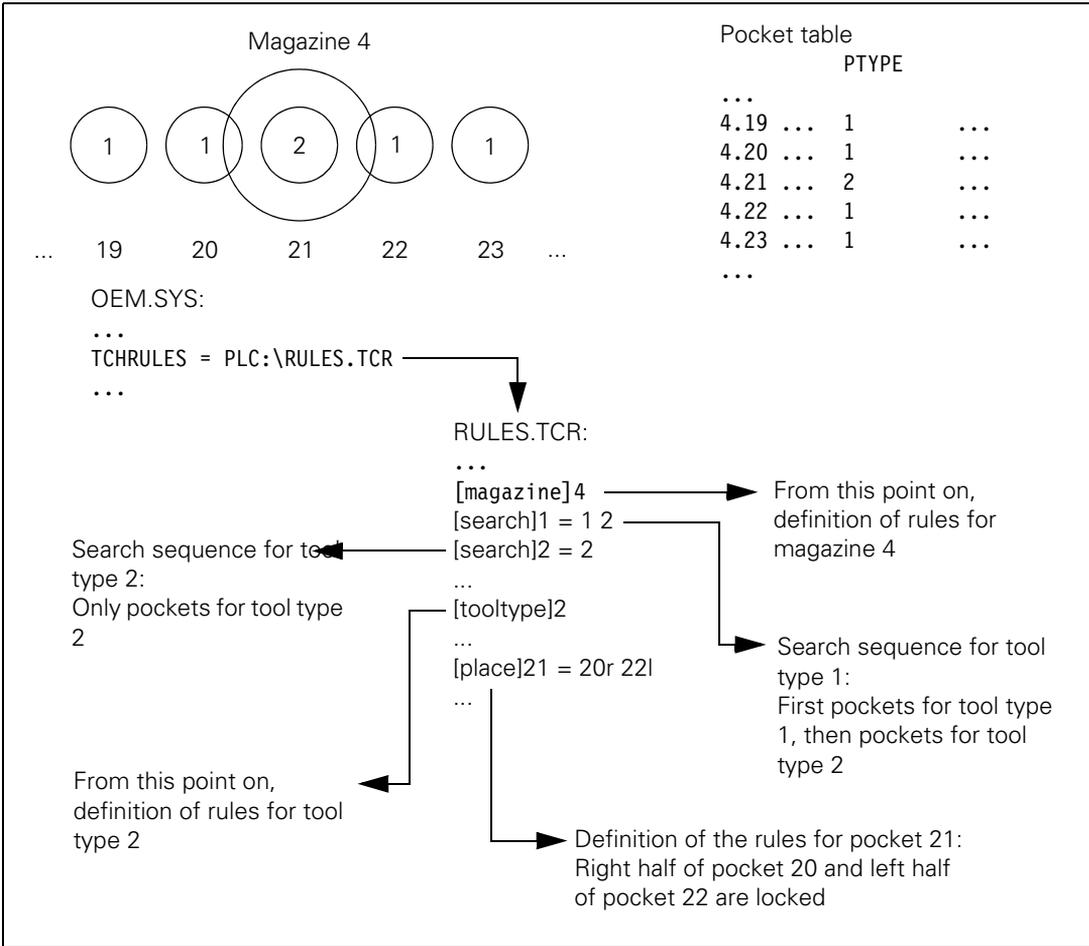
Module 9342 uses a tool number to determine the magazine number and pocket number.

Module 9341 processes the pockets depending on the magazine rules.

Pockets can be reserved, released and made unavailable.

Module 9216 is used to display a selection list in a pop-up window for placing tools into magazines and for removing them. The selection list is created by the NC at run time, and contains tools with and without pocket assignment as well as empty pockets. The user selects an entry from the selection list with the arrow keys, and Module 9216 reports the selection to the PLC for further processing.

Example for a description of the tool magazine using magazine rules:



Module 9216 Pop-up window with tool selection list

Module 9216 opens a pop-up window in which the arrow keys are used to make a selection for the tool management. The selection list is created by the NC at the run time for the module. The module responds with the tool or pocket number for further processing.

- Entries using the iTNC keyboard are registered by the pop-up window.
- The pop-up window is only shown in the machining modes.
- If another pop-up window is active, this window is placed in the background. After the pop-up window with the selection list is closed, this other pop-up window is returned to the foreground.
- If the pop-up window with the selection list is active, and another pop-up window is opened, any keystrokes on the iTNC keyboard will be registered by the second pop-up window, not by the selection list.
- The module should be called in its own spawn process, since the module does not return until the pop-up window is closed, and would therefore block all subsequent submit jobs.
- The window title is displayed in the language set in MP7230.0.

Call:

```
PS    B/W/D/K  <Mode>
        0: Tools in tool table not in the magazine
        1: Tools in tool table in the magazine
        2: Empty pockets in the magazine
        3: Tools with reserved pockets in the magazine
PS    B/W/D/K  <Magazine number>
        Only for mode 2
PS    B/W/D/K  <Tool number>
        Only for mode 2 (determining the tool type)
CM    9216
PL    B/W/D    <Tool or pocket number>
        -1: Error code in W1022
        -2: General error
        -3: Selection list closed without selection
        -4: Menu file not available
        -5: Pop-up window cannot be opened
        -6: Selection window already active
        -7: Selection window not available
        -8: Menu file without selection list
```

Error code:

Marker	Value	Meaning
M4203	0	Selection complete
	1	Error code in W1022
W1022	1	Line in the pocket table could not be found
	2	Invalid magazine number
	3	Invalid mode
	4	Invalid tool number or type
	6	Tool number is already contained in the pocket table
	20	Module was not called in a submit or spawn job
	36	File error in the tool or pocket table
45	Module execution canceled, see return value for error	

Module 9304 Copy columns P1 to P5 to the pocket table

Module 9304 transfers the contents of columns P1 to P5 from the pocket table to the defined double-word address.

Call:

PS B/W/D/K <Magazine number>
 PS B/W/D/K <Pocket number>
 PS B/W/D/K <Double-word address>
 CM 9304

Error code:

Marker	Value	Meaning
M4203	0	Columns copied
	1	Error code in W1022
W1022	1	Invalid pocket number
	2	Invalid magazine number
	4	Invalid double-word address
	20	Module was not called in a submit or spawn job
	36	File error in pocket table



Module 9340 Search for a pocket depending on magazine rules

Module 9340 searches a tool magazine for vacant, locked or unavailable pockets. The search for free pockets is according to the magazine rules.

Call:

PS B/W/D/K <Magazine number>

PS B/W/D/K <Pocket number for starting the search>

PS B/W/D/K <Tool number or type>

PS B/W/D/K <Mode>

Bit 0 = 0: Programmed number is tool type

Bit 0 = 1: Programmed number is tool number

Bit 1=1: Search for a vacant pocket

(depending on magazine rules)

Bit 2=1: Search for a reserved pocket

Bit 3=1: Search for an unavailable pocket

CM 9340

PL B/W/D/K <Pocket number>

-1: Error code in W1022

-2: No free pocket or tool not found

Error code:

Marker	Value	Meaning
M4203	0	Search for pocket completed
	1	Error code in W1022
W1022	1	Invalid pocket number
	2	Invalid magazine number
	3	Invalid mode
	4	Invalid tool number or type
	20	Module was not called in a submit or spawn job
	36	File error in the tool or pocket table
	45	Module execution canceled, see return value for error
	55	Pocket table could not be locked

Module 9341 Edit a pocket table depending on magazine rules

Module 9341 reserves, releases, or makes pockets unavailable in the pocket table, in accordance with the magazine rules.

The module affects the columns **RSV**, **LOCKED_ABOVE**, **LOCKED_BELOW**, **LOCKED_LEFT**, and **LOCKED_RIGHT**. Therefore these columns may not be changed manually nor by the PLC program.

Call:

PS B/W/D/K <Magazine number>

PS B/W/D/K <Pocket number>

PS B/W/D/K <Tool number>

PS B/W/D/K <Mode>

0: Release pocket (depending on magazine and tool number)

1: Release pocket (depending on magazine and pocket number)

2: Reserve pocket (depending on magazine, pocket and tool number)

3: Make pocket unavailable (depending on magazine and pocket number)

4: Reserve pocket if previously unavailable (depending on magazine and pocket number)

CM 9341

Error code:

Marker	Value	Meaning
M4203	0	Pocket table edited
	1	Error code in W1022
W1022	1	Invalid pocket number
	2	Invalid magazine number
	3	Invalid mode
	4	Invalid tool number
	6	Reservation not possible
	7	Magazine rules not compiled or not present
	20	Module was not called in a submit or spawn job
	36	File error in pocket table
	45	Module cancellation, error evaluation using return value
	55	Pocket table could not be locked



Module 9342 Find magazine and pocket number

Module 9342 determines the magazine and pocket number from the tool number. The module takes the **RSV** column of the pocket table into account if magazine rules are in effect. If the module is used to find reserved pockets, it returns the first reserved pocket with ascending magazine number. However, further pockets can be reserved. In this case the search must be repeated with another "start magazine for the search" command.

Call:

PS B/W/D/K <Tool number>

PS B/W/D/K <Mode>

0: Look for occupied pocket

1: Look for reserved pocket

PS B/W/D/K <Start magazine for the search>

CM 9342

PL B/W/D/K <Magazine number>

-1: Magazine could not be found

PL B/W/D/K <Pocket number>

-1: Pocket could not be found

Error code:

Marker	Value	Meaning
M4203	0	Magazine and pocket number found
	1	Error code in W1022
W1022	1	Invalid mode
	2	Invalid start magazine for the search
	20	Module was not called in a spawn or submit job
	30	Tool not found
	36	File error in pocket table

Module 9343 Compile and activate magazine rules

Module 9343 is used to compile and activate magazine rules (*.TCR), independent of the entry **TCHRULES** = in OEM.SYS. If the entry exists in OEM.SYS, the magazine rules are overwritten when Module 9343 is called. If an error occurs during compilation, the PLC program is stopped. The magazine rules must be activated during the first run of the PLC program or before the first call of Modules 934x.

Call:

PS B/W/D/K/S<Path and file name of the magazine rules>

CM 9343

Error code:

Marker	Value	Meaning
M4203	0	Magazine rules have been compiled and activated
	1	Error code in W1022
W1022	11	Invalid string programmed
	20	Module was not called in a spawn or submit job
	38	Error during compilation



Pocket exchange in the pocket table

To switch the pockets of two tools in the pocket table:

- ▶ Lock the pocket table with Module 9300.
- ▶ Switch the pockets with Module 9305.
- ▶ Release the pocket table with Module 9300.

Module 9300 Lock/release the pocket table

Module 9300 locks the pocket table for pocket switching with Modules 9305, 9306 and also 9301, 9302 or 934x, and then releases it again. The module can also be called while an NC program is running.

Call:

PS B/W/D/K <Lock/release pocket table>
0: Release the pocket table
1: Lock the pocket table

CM 9300

PL B/W/D <Error>
0: Pocket table locked/released
1: Pocket table could not be locked
2: Pocket table could not be released
3: Transfer parameter invalid
4: Module was not called in a spawn or submit job
5: Not used
6:
Code 0: Pocket table already released
Code 1: Pocket table already locked

Error code:

Marker	Value	Meaning
M4203	0	Pocket table locked/released
	1	Error code in W1022
W1022	2	Invalid parameter for locking/releasing the pocket table
	6	Pocket table was already locked/released
	20	Module was not called in a submit or spawn job
	21	Module was called during an NC program run

Module 9305 Tool exchange in the pocket table

Module 9305 is used to exchange the tools in the pocket table. The module can be called while an NC program is running. Only column T (tool number) is changed. All other columns remain unchanged. The pocket table must be locked with Module 9300 before switching the pockets, and then it must be released again.

Call:

PS B/W/D/K <Original pocket>

PS B/W/D/K <New pocket>

CM 9305

Error code:

Marker	Value	Meaning
M4203	0	Pocket has been exchanged
	1	Error code in W1022
W1022	2	Invalid parameter
	6	Magazine management using magazine rules is active
	20	Module was not called in a submit or spawn job
	21	Module was called during an NC program run without any locking in place
	30	No valid tool in the original pocket

Managing multiple tool magazines

Up to eight different tool magazines can be managed in the pocket table. In the pocket table the tool magazines are listed from 1 to 8, i.e., tool magazine 1 with tool 1 to <MP7261.0> is in first position. Immediately thereafter, tool magazine 2 appears with tool 1 to <MP7261.1>, then tool magazine 3, etc.

- ▶ Enter the number of pockets in tool magazines 1 to 8 in MP7261.0 to MP7261.7.

The current tool magazine number is saved in W268.

Module 9302 searches for an open pocket in a tool magazine, and Module 9306 switches tools between the tool magazines.

Module 9301 determines the number of the entry in the pocket table. The number of the entry depends on the tool magazine and pocket numbers.

- ▶ Enter this number in the modules which cannot accept tool magazine numbers (e.g. Modules 9092, 9093, 9094).

	Set	Reset
W268 Tool magazine number	NC	NC
-1: External tool		
0: Tool in the spindle		
1 to 8: Number of the tool magazine		



Search sequence of the tool magazines

Sequence for searching in tool magazines for the tool to be called. To call a tool through tool names, it is possible in machine parameter 7484.0 to .15 to specify a sequence (index from MP7261, or -1 cancel) for searching in tool magazines. If several tools are available in the magazine and a tool usage list is available, the tool with the shortest sufficient service life is chosen. If the tool is not found in any given magazine, the first tool with sufficient service life from the TOOL.T file is used.

MP7484.x Search sequence in tool magazines

Input: 0 to 15 [index from MP7261]
-1 = Cancel

Module 9301 Find the number of an entry in the pocket table

Module 9301 determines the number of an entry in the pocket table. This number is necessary for the modules in which no tool magazine numbers can be entered.

Call:

PS B/W/D/K <Tool magazine number>

PS B/W/D/K <Pocket number>

CM 9301

PL B/W/D <Number of the entry in the pocket table>
-1: M4203 = 1

Error code:

Marker	Value	Meaning
M4203	0	Number of the entry was found
	1	Error code in W1022
W1022	1	Invalid tool magazine number
	2	Invalid pocket number
	20	Module was not called in a submit or spawn job

Module 9302 Search for a vacant pocket in the tool magazine

Module 9302 searches for a vacant pocket in a tool magazine.

Call:

PS B/W/D/K <Tool magazine number>

PS B/W/D/K <Pocket at which the search is to be started>

CM 9302

PL B/W/D <Number of the vacant pocket>
-1: No vacant pocket available

Error code:

Marker	Value	Meaning
M4203	0	Search completed
	1	Error code in W1022
W1022	1	Invalid pocket number
	2	Invalid tool magazine number
	20	Module was not called in a spawn job or submit job
	36	Error in file handling

Module 9306 Exchange tools between tool magazines

With Module 9306, tools are exchanged between tool magazines. The pocket table must be locked with Module 9300 before calling this module, and then it must be released again. In the original and new entry only the tool number is changed. Pocket-specific data remains unchanged. The module must be called at standstill or during a strobe output. However, it can be called while an NC program is running.

Call:

PS B/W/D/K <Original tool magazine>
PS B/W/D/K <Original pocket>
PS B/W/D/K <New tool magazine>
PS B/W/D/K <New pocket>
CM 9306

Error code:

Marker	Value	Meaning
M4203	0	Pockets exchanged successfully
	1	Error code in W1022
W1022	1	Invalid pocket number
	2	Invalid tool magazine number
	6	Magazine management using magazine rules is active
	20	Module was not called in a submit or spawn job
	21	Module was called during an NC program run without any locking in place
	30	No valid tool in the original pocket
	36	Error in file handling



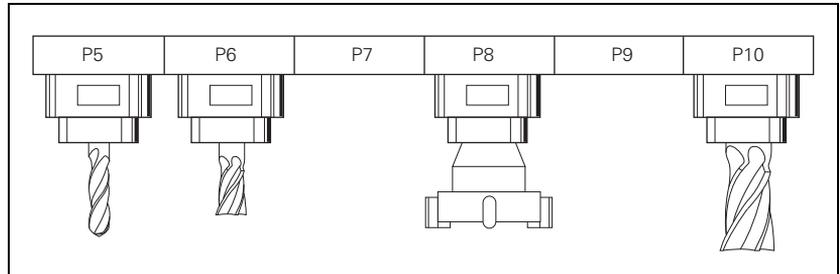
Special tools

In the pocket table:

- ▶ In the column **ST** you define tools as special tools.

For oversized special tools:

- ▶ Leave a pocket free in the tool magazine on both sides of the pocket (see illustration).
- ▶ In the column **L** you lock pockets that are to remain empty.
- ▶ With M4541, block the variable tool-pocket coding for special tools.



As soon as M4541 is set, all special tools are returned to their original pocket in spite of the "variable tool-pocket coding" function.

With the column **F** (fixed pocket) you can define this function selectively for individual tools.

		Set	Reset
M4541	Special tool in original pocket in spite of variable pocket coding	PLC	PLC

Tool life, replacement tool

You can enter two tool life values (**TIME1** and **TIME2**) and one replacement tool (**RT**) for each tool in the tool table.

For the **TOOL CALL**:

- **CUR.TIME** (current tool age) > **TIME2**: Pocket or tool number (MP7480) of the replacement tool and a T strobe M4073 are output and M4525 is set.
- **CUR.TIME** (current tool age) > **TIME2** > 0 and no replacement tool is defined: After expiration of the time, the error message **Max. tool age expired** is displayed for this tool, and M4546 and M4525 are set.
- **CUR.TIME** (current tool age) > **TIME1**: The NC sets M4543 and M4525.

You decide in the PLC what should happen when M4543 or M4546 is set (e.g. display a PLC error message).

With M101, you activate the automatic insertion of the replacement tool after expiration of the tool life (**TIME1** or **TIME2**). With M102, you deactivate the insertion. The tool is not changed immediately after expiration of the tool life, but rather it varies depending on the processor load. The tool change is transmitted delayed by at least one block and by no more than one minute. In order to also be able to activate the automatic insertion of the replacement tool with TCPM, you must program a retraction with M140 in the tool change macro. After the tool change, the tool moves with an approach logic to the pre-compensated position and then returns to the contour.

To synchronize the current machine status and the look-ahead calculation with an NC macro call, see "NCMACRO.SYS" on page 1709.



Note

In standard NC programs (NC block with **RR**, **RL** or **RO**), the same radius must be defined for the replacement tool as for the original tool.

No radius compensation is given in NC blocks with surface-normal vectors. One delta value for tool length and radius (**DR** and **DL**) can be entered for each tool in the tool table. These delta values are taken into account by the iTNC.

Setting bit 14 in MP7682 prevents a **TOOL CALL** from being output if no tool with sufficient **TIME2** for the precalculated processing time is found when searching for a tool in the sequence given by MP7484 (search sequence in tool magazines). If MP7682 bit 14 is set, the error message "Tool life expired" is displayed and the machining program is interrupted with an NC Stop. The machine parameter is also effective for the representation in the tool-usage list.

If the radius of the replacement tool differs from the original tool, you must define this in the **DR** column. The delta value must always be negative. If you enter a positive delta value, the error message **Tool radius too large** appears.

You can suppress this error message with the M function M107, and reactivate it with M108.

You can select whether the tool length is given with respect to the south pole or the ball center of a spherical cutter:

- With MP7680, select whether the tool radius (**R2**) should be taken into account for the calculation of the tool length.



The current tool age is calculated in the **Program Run, Single Block** and **Program Run, Full Sequence** operating modes if the following conditions are fulfilled.

- Spindle ON
- No F MAX
- F enable
- Control-in-operation symbol is on

After program interruption with "internal stop," M02, M30 or **END PGM**, the tool age counter is stopped.

The tool age counter does not run in the **Manual Operation, Electronic Handwheel** and **Positioning with MDI** operating modes.

The user can reset the current tool age by entering zero.

		Set	Reset
M4543	Tool life 1 expired (TIME1 in the tool table)	NC	NC/PLC
M4546	Tool life 2 expired (TIME2 in the tool table)	NC	NC/PLC
MP7680 Input:	Machine parameter with multiple function Bit 6 – Tool length in blocks with normal vectors: 0: Without R2 from tool table (south pole) 1: With R2 from tool table (center of sphere)		
MP7682 Input:	Machine parameter with multiple function Bit 14 – No insertion of the tool if TIME2 has expired 0: Tool is inserted even if TIME2 has expired 1: TOOL CALL for a tool where TIME2 has expired leads to an error message and NC stop		



Indexed tools

You can also work with indexed tools in the tool table, e.g., when you use a stepped drill with more than one length compensation value. For indexed tools, the tool number is given an index (e.g. 1.1).

► In MP7262, enter the maximum tool index number.

The index number of the programmed tool is saved in W266.

If you are working with indexed tools and wish to use Modules 9092, 9093 or 9094, you must first find the line number of the tool, since these modules will need it. As an alternative you can use Modules 9350 and 9351; the tool number and tool index can be transferred in these modules.

► Use Module 9091 to determine the line number of a tool in the tool table.

MP7262 Maximum tool index number for indexed tools

Input: 0 to 9

MP7262 can also be overwritten by the PLC and the LSV2 protocol.

		Set	Reset
W266	Index number of a programmed indexed tool	NC	NC

Module 9091 Find the line number of a tool in the tool table

Call:

PS B/W/D/K <Tool number>

PS B/W/D/K <Tool index>

CM 9091

PL B/W/D <Line number>

Error code:

Marker	Value	Meaning
M4203	0	Line number was found
	1	Error code in W1022
W1022	2	Invalid value for tool number or tool index number
	20	Module was not called in a spawn or submit job
	29	Tool table (TOOL.T) not found
	30	Tool number not found
	32	Tool index number not found



8.15.2 Tool-usage test

When testing an NC program in the **Test Run** operating mode (calculate machining time: active) or via an LSV2 command, a tool-usage file (*.T.DEP) can be created automatically. It contains all required tools (number, index, name, radius), the machining times (at 100% override) and their program calls.

► Enable the function for generating the tool-usage file with MP7246 bit 2=1.

In the **Program Run, Single Block** and **Program Run, Full Sequence** operating modes, press the **TOOL USAGE TEST** soft key to compare the data in the tool usage file with the data in the tool table. If the tool-usage file is not current or does not exist, the error message **Generate tool usage file!** appears. Otherwise a pop-up window with the results of the comparison appears. The value entered for **TIME2** in the tool table must be at least 10% greater than the time required.

In MP7485 a percentage value (0 to 100, default 10) can be entered for application to the usage time found in the tool usage list. This makes it possible to influence the selection if the selection of a tool depends on its usage time. This machine parameter is also effective with the **TOOL USAGE TEST** soft key in the **Program Run, Full Sequence** operating mode. Up to now, a constant value of 10% was applied.

If MP7246 bit 4 is set, the calculation of the tool usage times is written to an ASCII file. This file contains the run time and the absolute end time of each NC block listed by NC block numbers. The name of the ASCII file is derived from the name of the NC program (e.g. NCPROG.H) and has the extension *.POS.DEP (e.g. NCPROG.H.POS.DEP).

MP7246 Machine parameter with multiple function

Input: Bit 2 – Tool usage file
 0: Do not generate
 1: Generate
 Bit 4 – ASCII file for machining time per NC block
 0: Do not create ASCII file for machining time per NC block
 1: Create ASCII file for machining time per NC block

MP7485 Add usage time

Input: 0 to 100 [%]
 Default setting: 10

Tool usage test for pallet tables

The **TOOL USAGE TEST** soft key is also available for pallet tables.

- If a line with an NC program is active, the test is performed only for the NC program in question. The tool-usage file must first have been created in the **Test Run** operating mode.
- If a line with a pallet entry is active, the test is performed for the complete pallet table. The tool-usage files of the NC programs called must first have been created in the **Test Run** operating mode. One tool-usage file is created for the entire pallet table.
- With Module 9282, the tool-usage test for a pallet table can be performed by the PLC. One tool-usage file is created for the entire pallet table.

Module 9282 Tool usage test for pallet table

Module 9282 allows you to check the tools used in a pallet table. The pallet file must be selected in the **Program Run, Single Block** or **Program Run, Full Sequence** operating mode. The tool usage file for the pallet table and the test result file in ASCII format are created. The test result file contains the results from the comparison of the tool usage file with the tool table.

If a pallet call is given as the line number, all subordinate machining operations are checked. If a program call is given, only the tool usage file of the NC program is checked.

Call:

```
PS    B/W/D/K  <Line number of the pallet table>
PS    B/W/D/K/S<Name of the test result file (*.P.T.DEP)>
PS    B/W/D/K/S<Name of the tool usage file (*.T.DEP)>
CM    9282
PL    B/W/D    <Result>
                0: No error
                1: Tool usage file of an NC program of the pallet table not
                   available or no longer up-to-date
                2: Tool life not sufficient
                3: Required tool not available
                4: Radius of required tool is incorrect
                5: Tool is not in magazine
                6: Tool usage file of an NC program of pallet table cannot be
                   opened
                7: Test result file could not be created
                8: Test result file cannot be written to
                9: Tool usage file of an NC program of pallet table cannot be
                   read
                10: No memory for creating the tool usage file
```

Error code:

Marker	Value	Meaning
M4203	0	Tool usage test has been performed
	1	Error code in W1022
W1022	11	Invalid string programmed
	20	Module was not called in a spawn or submit job



Structure of the tool-usage file

The columns of the tool-usage file have the following meanings:

Column	Description
NR	Line number
T	Tool number and index combined. (e.g. '14.4' for tool 14 with index 4).
TOKEN	Designation for line type: TOOL : Usage time per TOOL CALL ; the entries are listed in chronological order TTOTAL : Total usage time of a tool STOTAL : Call of a subprogram (including cycles). The entries are listed in chronological order. TIMETOTAL : Sum of all entries in the TIME and WTIME columns. In the other columns this line contains the value 0. TOOLFILE : The PATH column contains the tool table that was active when the tool-usage file was created.
TNR	Tool number (-1: no tool inserted yet)
IDX	Tool index
NAME	Tool name
TIME	Tool-usage time in seconds Time duration that the tool was used to machine a workpiece (spindle on and not at rapid traverse!)
WTIME	Tool-selection time in seconds Time duration that a tool was selected during the program. It also includes the times where the tool was moved without M3/M4, as well as times where it was moved at FMAX .
RAD	Tool radius R + Oversize for tool radius DR ; Data from the tool table in 1/10 µm
BLOCK	Line number with TOOL CALL
PATH	Token = TOOL : Active program or sub program Token = STOTAL : File name of the subprogram or cycle

Example of a tool-usage file:

NR	TOKEN	TNR	IDX	NAME	TIME	RAD	BLOCK	PATH
0	TOOL	+2	+0	MILLER2	+144	+60000	+4	TNC:\Stefan\Test\TOOLING.H
1	TOOL	+3	+0	DRILLER1	+7	+30000	+9	TNC:\Stefan\Test\TOOLING.H
2	TOOL	+4	+0	DRILLER2	+100	+40000	+11	TNC:\Stefan\Test\TOOLING.H
3	TOOL	+2	+0	MILLER2	+1588	+60000	+14	TNC:\Stefan\Test\TOOLING.H
4	TTOTAL	+3	+0	DRILLER1	+7	+30000	+0	
5	TTOTAL	+4	+0	DRILLER2	+100	+40000	+0	
6	TTOTAL	+2	+0	MILLER2	+1732	+60000	+0	
7	STOTAL	+0	+0		+0	+0	+0	SYS:\JHCYC\NC\252.CYC
8	STOTAL	+0	+0		+0	+0	+0	SYS:\JHCYC\NC\INIT25X.CYC
9	STOTAL	+0	+0		+0	+0	+0	SYS:\JHCYC\NC\PSRD25X.CYC
10	STOTAL	+0	+0		+0	+0	+0	SYS:\JHCYC\NC\MILL25X.CYC
11	STOTAL	+0	+0		+0	+0	+0	TNC:\Stefan\Test\DRILL.H
12	STOTAL	+0	+0		+0	+0	+0	SYS:\JHCYC\NC\200.CYC
13	STOTAL	+0	+0		+0	+0	+0	TNC:\Stefan\Test\DRILL2.H
14	STOTAL	+0	+0		+0	+0	+0	SYS:\JHCYC\NC\205.CYC
15	STOTAL	+0	+0		+0	+0	+0	SYS:\JHCYC\NC\251.CYC
	[END]							

Module 9283 Generate tool-usage file

This module is used to create a tool-usage file for a specified part program, including its subprograms.

Constraints:

- In order to create the file, the Test Run mode must be the active editing mode on the control.
- The NC program is selected automatically and started in Test Run.
- Generation of the tool-usage file must be permitted with bit 2 of MP7246.
- The module must be programmed in a spawn job, since program run can take a long time and submit subprograms would be blocked for a long time.

Call:

```
PS   B/W/D/K <Mode>
      0: NC program including subprograms
PS   B/W/D/K <Additional parameter>
      Content is currently not evaluated, but must be
      programmed
PS   B/W/D/K/S<Path>
      0: Path of the NC program
CM   9283
PL   B/W/D <Error code>
      0: No error
      1: Call was not in a submit/spawn job
      2: Invalid value in the <Mode> or <Additional parameter>
      parameter or
      3: Invalid value in the <Path> parameter
      4: Internal error
      5: File in <Path> parameter does not exist, or invalid file type
      6: File generation not permitted or test run already started
      7: Generation of .DEP not enabled (MP7246, bit 2)
      8: Error during program run
```

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Faulty execution of module
W1022	1	Mode parameter is invalid
	2	NC program does not exist or wrong file type
	6	Generation of .T.DEP files not enabled in the configuration
	8	Control is in an operating mode from which the call is not possible, or test run is already started
	11	An invalid PLC string was programmed
	20	Call was not in a submit/spawn job
	45, 58, 61	Internal error



8.15.3 Automatic calculation of cutting data

The optimum spindle speed and the corresponding contouring feed rate is calculated from the values entered in cutting tables for the tool and workpiece material.

For cutters, the cutting data table lists the cutting speed and the feed rate per tooth. For drills it lists the feed rate per revolution.

$$S = \frac{v_c \cdot 1000}{d \cdot \pi}$$

S: Spindle speed [min^{-1}]

v_c : Cutting speed [m/min]

d: Tool diameter [mm]

Milling cutter: $F = f_z \cdot S$

F: Feed rate [mm/min]

f_z : Feed rate per tooth [mm]

z: Number of teeth

Drill: $F = f_u \cdot S$

f_u : Feed rate per revolution [mm]

Tool table

- ▶ In the **CDT** column of the tool table, enter the name of the cutting data table that is to be used for that tool.
- ▶ In the TYP column define the type of tool:
 - **DRILL** = Drilling tool
 - **TAP** = Tapping tool
 - **MILL** = Milling cutter
- ▶ Enter the following values in the table:
 - Tool radius **R**
 - Tool material **TMAT**
 - For cutter: Number of teeth **CUT**.

The tool types are defined in the file PLC:\TTYP.TAB.

If you edit this file, you must use the command **TTYP =** to enter the new name and path in the system file OEM.SYS.

Cutting data table

The cutting data for specific tools are available from the tool manufacturer.

Cutting data tables have the file name extension .CDT.

Each line in the cutting data table contains the data for a specific combination of workpiece and tool material. For milling cutters you can enter up to four cutting speeds with the corresponding feed rates per teeth. In the tables of the manufacturers these data are specified for different infeeds and for climb and up-cut milling. For drills you enter a cutting speed with the corresponding feed rate per revolution.

A standard cutting data table is saved in the root directory of the iTNC (TNC:\). You can add as many cutting data tables as desired.

If you change the standard cutting data table, you must copy the changed table to another path. Otherwise your changes will be overwritten with HEIDENHAIN standard data during the next software update:

- ▶ In the system file TNC.SYS, use the code word **PCDT** = to enter the path in which your cutting data tables are saved.

Material tables

The workpiece materials used are defined in the table WMAT.TAB, the tool materials in the table TMAT.TAB.

Standard tables are in the root directory of the TNC (TNC:\).

You can arbitrarily expand and change all tables.

If you change the tables, you must copy them to another path. Otherwise your changes will be overwritten with HEIDENHAIN standard data during the next software update:

- ▶ In the system file TNC.SYS, use the code words **TMAT** = and **WMAT** = to enter the path and file names of your tables.

In the material data tables:

- ▶ In the **Name** column, enter a brief name for the material (e.g. **HSS**).
- ▶ Enter additional information on the material in **DOC** column.

Calculation of cutting data

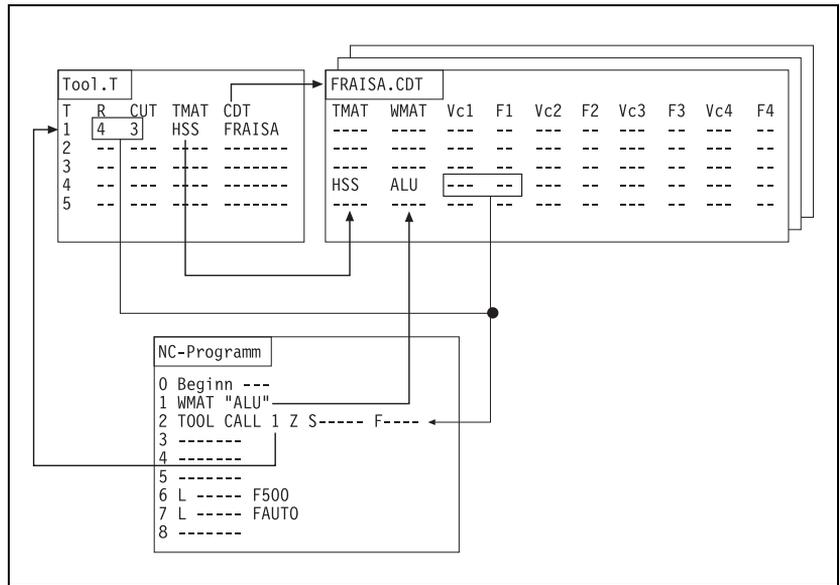
- ▶ Define the workpiece material in the NC program with the **WMAT** soft key.

The **TOOL CALL** block provides soft keys for automatic acceptance of various speeds (S1 to S4) and for the selection of the feed rate (F1 to F4).

If you enter the spindle speed manually, this value is taken into account in the calculation of the feed rate. You cannot, however, enter F for calculation of S. If you enter the feed rate manually, the entered value applies until you program another feed rate. With the **F AUTO** soft key you can again activate the feed rate from the **TOOL CALL** block.



Principle



8.15.4 Automatic tool recognition

Automatic tool identification is possible with the Balluff tool identification system (BIS).

Please contact HEIDENHAIN for further information.

8.15.5 Controlling the tool changer

You program the control of the tool changer in the PLC. This includes:

- Positioning of the changing arm and carousel
- Tool change sequence

The NC handles the tool management. This includes:

- Tool life
- Pocket assignment
- Evaluation of the **TOOL DEF** blocks
- Evaluation of the **TOOL CALL** blocks

The NC and PLC communicate through markers and words.

For execution of the **TOOL CALL** block, the NC takes the tool geometry data from the tool table:

- ▶ Activate with M4538 the geometry of the tool defined in W264. With this marker you make sure that the current tool geometry is always active even if the tool change sequence is canceled.

CAUTION: Activate only together with an M/S/T/Q strobe or if M4176 = 0 (control-in-operation display is lit or blinking)!

With the **TOOL DEF** block you can pre-position the tool changer:

- ▶ After a tool has been changed, program the next tool with **TOOL DEF**.
- ▶ Evaluate the tool and pocket number and pre-position the tool changer to the follow-up tool via a PLC positioning.
- ▶ In MP7682 bit 6 you specify the behavior of the control when a **TOOL DEF** strobe is current.

Bit 6 = 1: The control always waits for an acknowledgment of the **TOOL DEF** strobe by the PLC before continuing with the NC program.

Bit 6 = 0: Continuation of the NC program depends on the NC program. If the **TOOL DEF** is within a contiguous contour, then an acknowledgment of the **TOOL DEF** strobe is **not** waited for. Depending on the type of tool changer, this behavior can lead to problems:

Example:

NC program	Description
<pre>... L IX+50 F500 ... TOOL DEF 1 ... L IX+50 F500</pre>	The TOOL DEF block is within a contiguous contour. The acknowledgment of the strobe is not waited for.
<pre>... ... TOOL DEF 1 ... L IX+50 F500</pre>	The TOOL DEF block is not within a contiguous contour. The acknowledgment of the strobe is waited for.

TOOL DEF commands are used by machines with tool magazines to prepare tools. Until now, the machining program was interrupted upon a TOOL DEF command for a tool that was not available. If MP7682 bit 13 is set, then when a preparatory TOOL CALL (TOOL DEF block) is output, machining is not even interrupted if the programmed tool is invalid. The following error message are then shown only as warnings, but the machining program continues running (no T2 strobe is sent to the PLC upon TOOL DEF):

- Tool definition is missing
- Tool life expired
- Tool locked
- Tool not defined
- Tool radius too large

MP7682 Machine parameter with multiple function

Input:

Bit 6 – Behavior with TOOL DEF strobe

0: Depending on the NC program, the TOOL DEF strobe must be acknowledged by the PLC (TOOLDEF within a contiguous contour)

1: TOOL DEF strobe must always be acknowledged by the PLC

Bit 13 – No program interruption upon invalid TOOL DEF

0: Error message and NC stop upon invalid TOOL DEF

1: Only warning upon invalid TOOL DEF



Calling an NC program with TOOL CALL

With the NC block TOOL CALL you can call an NC program of your own definition:

- ▶ With the keyword **TC = <path name>\<file name>** in the PLC:NCMACRO.SYS file, define the name of the NC program to be called.

To synchronize the current machine status and the look-ahead calculation with an NC macro call, see "NCMACRO.SYS" on page 1709.

The tool geometry is not taken over then. You must program a TOOL CALL at another place to update the tool data.

NC functions that must be reset at the beginning of a tool-change macro:

The tool-change macro requires non-radius-compensated movements with M91 (coordinates refer to the machine datum) or M92 (coordinates refer to a position defined by the machine manufacturer):

- M103 (Reduce feed rate during plunging to factor F)
- M112 (Insert rounding radius between nontangential straight lines)
- M114 (Automatic correction of machine geometry when machining with tilting axes)
- M118 (Superimpose handwheel positioning during program run)
- M124 (Ignore points when machining non-radius compensated straight line blocks)
- M128 (Retain position of tool tip when positioning tilting axes) or **FUNCTION TCPM**
- Cycle19 (**WORKING PLANE**) or **PLANE**
- Possibly M126 (Permit zero crossover on 360° rotary axes), if rotary axes are moved and their traverse ranges permit multiple paths.
- Possibly M136 (Feed rate F in millimeters per spindle revolution), if feed rates other than **FMAX** are used.
- Possibly M144 (Compensate the machine's kinematics configuration for ACTUAL/NOMINAL positions at end of block), if this function is activated via MP7502.
- Possibly Cycle 32 (**TOLERANCE**), if a certain tolerance (MP1096.x) is required in the tool-change macro.

The functions **FN17: SYSWRITE ID 230 ...** are available to enable you to influence the software limit switches in a tool change macro. This function is therefore effective only for a limited period of time. The original software limit switches are restored under the following conditions:

- End of macro or cancelation of macro
- Call of an NC or PLC function that results in machine parameters being read in (e.g. switching the traverse range, activating an MP subfile)
- Selection of an NC program or call of a GOTO block during NC program run

Program example: Positioning to the tool change position:

- ▶ Preferably file the program in the PLC partition so that it cannot be changed by the end user.

The tool data in the current program are not active. They must be requested with **FN18: SYSREAD** (see "Data transfer NC > NC program (FN18: SYSREAD)" on page 1725).

- ▶ In the called program, enter a **TOOL CALL** so that the tool data becomes active and a T strobe is transferred to the PLC.

With **FN17: SYSWRITE** you can overwrite the software limit switch for the tool-change position. If you use **FN18: SYSREAD** to call the programmed position after the **TOOL CALL**, you can program a continuous positioning movement of the spindle from the tool magazine to the next position.

With **FN20: WAIT FOR** you can delay execution of the NC program until the entered condition is fulfilled. These conditions can be comparisons of a PLC variable with a constant (see "Interrogate PLC operands in the NC program (FN20: WAIT FOR)" on page 1740).

With **FN17: SYSWRITE ID420 NRO IDX = 0**, all coordinate transformations (e.g. cycles 7, 8, 10, 11, 19) performed in the tool-change program become globally effective. Without this block, they remain locally effective (only in the tool-change program). If a coordinate transformation status is to be globalized, FN17: ID420 must be programmed after the nominal condition has been established.

With **FN18: SYSREAD ID61 NRO IDX<tool number>** you find the corresponding tool-change sequence. This information is reported to the PLC at the same time with the pocket and magazine number. You can also find this information with Module 9035.

To ensure that during a block scan the tool-change program is not run until the end of the scan, you must enter the instruction **NCMACRO = TC** in the MGROUPS.SYS file (also see "Returning to the contour" on page 1244). If no NC program is specified in the NCMACRO.SYS file, the **TOOL CALL** is executed without calling the tool-changing program.

For test purposes, the tool-change program can be called from the TNC partition. In this case, the program call is handled as **PGM CALL**, i.e. defined values such as Q parameters and feed rate remain globally effective. If the tool-change program is called from the PLC partition, the tool-change program is handled as a cycle call, i.e. defined values remain only locally effective.



Note

In contrast to the TNC 426/TNC 430, the iTNC 530 handles the tool number and tool index as two separate parameters. Therefore, with the iTNC 530, the tool number must be read with **FN18: SYSREAD Q1 = ID60 NR1** and the tool index with **FN18: SYSREAD Q7 = ID60 NR8**. Both values are then transferred in **TOOL CALL Q1 .Q7**.



```

0 BEGIN PGM TOOLCALL MM
1 * - ? DATUM SHIFT PLC OFF/ON
2 * - 3D ROTATION OFF/ON
3 * - READ TOOL CALL DATA
4 FN18: SYSREAD Q1 = ID60 NR1 IDX0 ; Tool number
5 FN18: SYSREAD Q2 = ID60 NR2 IDX0 ; Tool axis
6 FN18: SYSREAD Q3 = ID60 NR3 IDX0 ; Spindle speed
7 FN18: SYSREAD Q4 = ID60 NR4 IDX0 ; Oversize in tool length DL
8 FN18: SYSREAD Q5 = ID60 NR5 IDX0 ; Oversize in tool radius DR
9 FN18: SYSREAD Q6 = ID60 NR7 IDX0 ; Oversize in tool radius DR2
10 FN18: SYSREAD Q7 = ID60 NR8 IDX0 ; Tool index
11 * - T-NEW POSITION AXIS 7 (FKA)
12 FN 18: SYSREAD Q6 = ID2000 NR70 IDX98; Read W98
13 * - POSITION AXIS 8 (FK)
14 FN 18: SYSREAD Q7 = ID2000 NR80 IDX98; Read D98
15 * - TC Z AXIS SAFETY CLEARANCE
16 L Z+0 RO F MAX M91
17 * - ALIGN SPINDLE M21
18 M21
19 FN 18: SYSREAD Q8 = ID1000 NR4210 IDX11 ; Read MP4210.11
20 FN 18: SYSREAD Q9 = ID1000 NR4210 IDX12 ; Read MP4210.12
21 FN 18: SYSREAD Q10 = ID230 NR3 IDX2 ; Pos. software limit switch Y
22 * - OPEN TRAVERSE RANGE Y
23 Q9 = Q6 + 0.5
24 FN 17: SYSWRITE ID230 NR3 IDX2 = +Q9
25 * - Z AXIS APPROACH TC AREA
26 L Y+Q6 RO F MAX M91
27 L Z+Q7 RO F MAX M91
28 * - TOOL MACRO ACTIVE ?
29 FN 20: WAIT FOR SYNC M1999==1
30 * - WRITE TOOL CALL DATA ->PLC
31 FN 9: IF +Q2 EQU +0 GOTO LBL 11
32 FN 9: IF +Q2 EQU +1 GOTO LBL 12
33 FN 9: IF +Q2 EQU +2 GOTO LBL 13
34 FN 9: IF +Q2 EQU +3 GOTO LBL 14
35 FN 9: IF +Q2 EQU +4 GOTO LBL 15
36 FN 9: IF +Q2 EQU +5 GOTO LBL 16
37 TOOL CALL Q1 SQ3 DL+Q4 DR+Q5
38 FN 9: If +0 EQU +0 GOTO LBL 17
39 LBL 11
40 TOOL CALL Q1 .Q7 X SQ3 DL+Q4 DR+Q5 DR2:+Q6
41 FN 9: If +0 EQU +0 GOTO LBL 17
42 LBL 12
43 TOOL CALL Q1 .Q7 Y SQ3 DL+Q4 DR+Q5 DR2:+Q6
44 FN 9: If +0 EQU +0 GOTO LBL 17
45 LBL 13
46 TOOL CALL Q1 .Q7 Z SQ3 DL+Q4 DR+Q5 DR2:+Q6
47 FN 9: If +0 EQU +0 GOTO LBL 17
48 LBL 14
49 TOOL CALL Q1 .Q7 U SQ3 DL+Q4 DR+Q5 DR2:+Q6
50 FN 9: If +0 EQU +0 GOTO LBL 17
51 LBL 15
52 TOOL CALL Q1 .Q7 V SQ3 DL+Q4 DR+Q5 DR2:+Q6
53 FN 9: If +0 EQU +0 GOTO LBL 17
54 LBL 16
55 TOOL CALL Q1 .Q7 W SQ3 DL+Q4 DR+Q5 DR2:+Q6

```

```

56 LBL 17
57 * - Z AXIS LEAVE TC AREA
58 L Z+0 RO F MAX M91
59 L Y+0 RO F MAX M91
60 * - CLOSE TRAVERSE RANGE Y
61 FN 17: SYSWRITE ID230 NR3 IDX2 = +Q8
62 END PGM TOOLCALL MM

```

Variable and fixed pocket coding

If you work with **one magazine**, you must specify the type of pocket coding for this magazine:

- ▶ Set MP7482 = %0000.
- ▶ Specify with MP7480.x whether the tool or pocket number is to be transferred to the PLC:
 - Variable pocket coding: Pocket number must be transferred. Set MP7480.x = 3 or 4.
 - Fixed pocket coding: Working with the tool number is preferred. Set MP7480.x = 5 or 6

If you work with **more than one magazine**, you must specify the type of pocket coding for each magazine individually:

- ▶ Set MP7480.x to 3 or 4 for variable pocket coding.
- ▶ Define in MP7482 the type of pocket coding for each magazine.

Depending on the setting of MP7480.x, the NC transfers either only the number of the programmed tool to word W264 or the tool and pocket number to W262 and W264.

The NC sets M4073 (**TOOL CALL**) or M4074 (**TOOL DEF**). The strobe markers are not reset until you have set M4093 (**TOOL CALL**) or M4094 (**TOOL DEF**) after the tool or pocket number, respectively, have been processed. After you have reset the strobe marker, the NC program is resumed (only with **TOOL CALL**). If a **TOOL CALL** block is followed by the output of a T strobe and G strobe, then M4547 is set by the output of the T strobe and reset by output of the G strobe. If there is no output of either the T or G strobe, M4547 is not set.

After the acknowledgment marker for an NC strobe is set (e.g. Marker M4093 for T strobe), the corresponding strobe marker (e.g. Marker M4073) is not reset (= 0) by the NC software until the NC has actually concluded the actions immediately triggered by acknowledgment of the strobe. This makes it possible for the PLC program to know when this processing by the NC is actually concluded. For example: renewed access to the pocket table only once pocket exchanging by the NC has concluded. The behavior of the strobe markers can be configured with bit 16 of MP4020.

If the tool number zero is processed, the NC sets marker M4521. The marker is not reset until there is a **TOOL CALL** for another tool.

With MP7483 you specify if the tool name or tool number or both can be used for **TOOL CALL** or **TOOL DEF**. For example, if a tool number is used, an NC error message will be output if only tool names are allowed (MP7483 = 1).



Example of how the FN18 functions for tool name and number can be used in combination with MP7483. In this example the Q-parameter number Q330 could have been transferred as a string or a number from a cycle. The following example can be used for further processing:

```

...
FN 9:   IF +Q330 EQU +0 GOTO LBL 131
FN 18:  SYSREAD Q35 = ID1000 NR7483      Interrogate MP7483
                                           whether tool name and
                                           number are permitted

FN 18:  SYSREAD Q30 = ID30 NR52 IDX330   Interrogate if the tool
                                           name is programmed

FN 9:   IF +Q30 EQU +0 GOTO LBL 45       No tool name
                                           programmed

FN 10:  IF +Q35 NE +2 GOTO LBL 46       Tool name programmed

FN 14:  ERROR = 1094                    Tool name is not
                                           permitted

LBL 46
FN 18:  SYSREAD Q330 = ID990 NR10 IDX330 Determine the tool
                                           number for the tool name

FN 10:  IF +Q330 NE -1 GOTO LBL 34

FN 14:  ERROR = 1092                    No tool number could be
                                           found

LBL 45
FN 10:  IF +Q35 NE +1 GOTO LBL 34

FN 9:   IF +Q330 EQU +0 GOTO LBL 34     Tool number = 0 always
                                           permitted

FN 14:  ERROR = 1093                    Tool number not
                                           permitted

LBL 34
FN 18:  SYSREAD Q0 = ID50 NR22 IDX330
...

```

MP7480 Output of the tool or pocket number

MP7480.0

With a TOOL CALL block

Input:

0: No output

1: Tool number output only when tool number changes

2: Tool number output for every TOOL CALL block

3: Pocket number and tool number output only when tool number changes

4: Pocket number and tool number output for every TOOL CALL block

5: Pocket number and tool number output only when tool number changes. Pocket table is not changed.

6: Pocket number and tool number output for every TOOL CALL block. Pocket table is not changed.

MP7480.1

With a TOOL DEF block

Input:

0: No output

1: Tool number output only when tool number changes

2: Tool number output for every TOOL DEF block

3: Pocket number and tool number output only when tool number changes

4: Pocket number and tool number output for every TOOL DEF block

MP7482 Pocket coding of the tool magazines

Format:

%xxxxxxx

0: Variable pocket coding

1: Fixed pocket coding

Input:

Bit 0: Magazine 1

Bit 1: Magazine 2

Bit 2: Magazine 3

Bit 3: Magazine 4

Bit 4: Magazine 5

Bit 5: Magazine 6

Bit 6: Magazine 7

Bit 7: Magazine 8

MP7483 Tool name/number for TOOL CALL / TOOL DEF

Input: 0: Names and numbers are permitted (as before)
 1: Only names are permitted
 2: Only numbers are permitted

MP4020 Bit 16 – Time at which strobe markers are reset

Input: 0: Do not reset strobe markers until the action triggered by acknowledgment of the strobe is concluded by the NC
 1: Reset strobe markers immediately upon acknowledgment (behavior same as up to and including 606 42x-01 SP 04)

		Set	Reset
W262	Tool pocket number	NC	NC
W264	Tool number	NC	NC
M4073	Strobe signal T code (P code) with TOOL CALL	NC	NC
M4074	Strobe signal T code (P code) with TOOL DEF	NC	NC
M4093	Acknowledgment of T code (P code) with TOOL CALL	PLC	PLC
M4094	Acknowledgment of T code (P code) with TOOL DEF	PLC	PLC
M4521	Tool number zero programmed	NC	NC
M4538	Geometry of the tool from W264	PLC	NC
M4547	T and G strobes with TOOL CALL	NC	NC

Output of the tool number with fixed pocket coding

For fixed pocket coding of tools you must evaluate the tool number:

- ▶ With MP7480.x, specify when the tool number is to be transferred.
 - For every **TOOL CALL** or **TOOL DEF** block: MP7480.x = 2
 - When the tool number changes: MP7480.x = 1
 During execution of a **TOOL CALL** or **TOOL DEF** block the tool number is saved in W264. W262 is not used.
 - For MP7480.x = 5 or 6: The pocket number is saved in W262. The assignment of tool and pocket number in the pocket table does not change.

Output of the pocket number with variable pocket coding

With variable pocket coding (MP7480.x = 3 or 4) the pocket number of the called tool is transferred to the PLC and the assignment of tool and pocket number is changed in the pocket table. The current tool number is additionally saved in W264. The NC takes over responsibility for variable pocket management.

If you have set M4542, the assignment of tool and pocket numbers in the pocket table does not change, although variable pocket coding was selected. You set this marker, for example, during a block scan (except if MP7681 bit 1=1).

► In MP7261, enter the number of tools with pocket number. The maximum input value is the number of the pockets in the tool magazine.

In the tool table you can define more tools than can be held by the tool magazine (MP7260 > MP7261). If a tool number is programmed for which no pocket was defined, during a **TOOL CALL** the pocket number -1 (W262) is transferred and M4523 is set.

During programming of **TOOL DEF** the tool and pocket numbers are transferred. A **TOOL DEF** for a manual tool has no relevance for the PLC.

Define in the column **F** a fixed pocket. If a fixed pocket has been defined for a tool, it will be returned to its original pocket in spite of the variable pocket coding.

		Set	Reset
M4520	Another T code (P code) follows with TOOL CALL	NC	NC
	0: Normal tool follows a normal tool (N → N)		
	Manual tool follows a manual tool (M → M)		
	Special tool follows a special tool (S → S), if M4541 = 0		
	1: Special tool follows a manual tool (M → S), if M4541 = 1		
	Special tool follows a special tool (S → S), if M4541 = 1		
	Manual tool follows a special tool (S → M)		
	Manual tool follows a normal tool (N → M)		
	Normal tool follows a manual tool (M → N)		
	Normal tool follows a special tool (S → N) see M4540		



		Set	Reset
M4522	Tool with pocket number programmed is in effect with MP7480.0 = 3 or 4 and TOOL CALL	NC	NC
M4523	Tool without pocket number programmed is in effect with MP7480.0 = 3 or 4 and TOOL CALL	NC	NC
M4524	Special tool called, TOOL CALL	NC	NC
M4525	TOOL CALL after expiration of tool life 1: TOOL CALL after expiration of tool life	NC	NC
M4540	Sequence of tool number or pocket number transfer (M4520 = 1) ■ 0: First the number for the old tool, then the number for the new tool (single changing arm) ■ 1: First the number for the new tool, then the number for the old tool (double changing arm)	PLC	PLC
M4541	Special tool in original pocket in spite of variable pocket coding	PLC	PLC
M4542	Do not update pocket number in pocket table	PLC	PLC

A variety of tool types can be called from the machining program. The abbreviations below are defined for the following examples:

- **N:** Tool for which one pocket is defined in the tool table (**N**ormal)
- **M:** Tool for which no pocket number is defined in the tool table. You must change the tool manually (**M**anual).
- **S:** Special tool, definition in the tool table

There are nine possible combinations in the tool-change sequence. For some sequences it is necessary during **TOOL CALL** to output two pocket or tool numbers in sequence. You can see in M4520 if another tool or pocket number is transferred. The sequence of transfers for tool and pocket numbers can be defined in two manners:

- Define the sequence in MP7481.x. In this case set M4540 = 0.
- Define the sequence in M4540. In this case set MP7481.x = %00000000.

You must evaluate and acknowledge both pocket or tool numbers. As an alternative, all of this information can be found with Module 9035.

MP7481 Sequence for new and returned tool when changing tools

Format: %xxxxxxxxxxxxxxxxxx
0: First, output the pocket of the tool to be returned
1: First, output the pocket of the new tool

Input: Bit 0: New tool from magazine 1
Bit 1: New tool from magazine 2
Bit 2: New tool from magazine 3
Bit 3: New tool from magazine 4
Bit 4: New tool from magazine 5
Bit 5: New tool from magazine 6
Bit 6: New tool from magazine 7
Bit 7: New tool from magazine 8
Bit 8: New tool from magazine 9
Bit 9: New tool from magazine 10
Bit 10: New tool from magazine 11
Bit 11: New tool from magazine 12
Bit 12: New tool from magazine 13
Bit 13: New tool from magazine 14
Bit 14: New tool from magazine 15
Bit 15: New tool from magazine 16

MP7481.0 Tool to be returned to magazine 1
MP7481.1 Tool to be returned to magazine 2
MP7481.2 Tool to be returned to magazine 3
MP7481.3 Tool to be returned to magazine 4
MP7481.4 Tool to be returned to magazine 5
MP7481.5 Tool to be returned to magazine 6
MP7481.6 Tool to be returned to magazine 7
MP7481.7 Tool to be returned to magazine 8
MP7481.8 Tool to be returned to magazine 9
MP7481.9 Tool to be returned to magazine 10
MP7481.10 Tool to be returned to magazine 11
MP7481.11 Tool to be returned to magazine 12
MP7481.12 Tool to be returned to magazine 13
MP7481.13 Tool to be returned to magazine 14
MP7481.14 Tool to be returned to magazine 15
MP7481.15 Tool to be returned to magazine 16



Module 9035 Read status information

Call:

PS B/W/D/K <Number of the status information>
50: Tool change sequence
(see **FN18: SYSREAD ID61 NR0**)
51: Pocket number for reserve
52: Magazine number for reserve
53: Pocket number for insertion
54: Magazine number for insertion

CM 9035

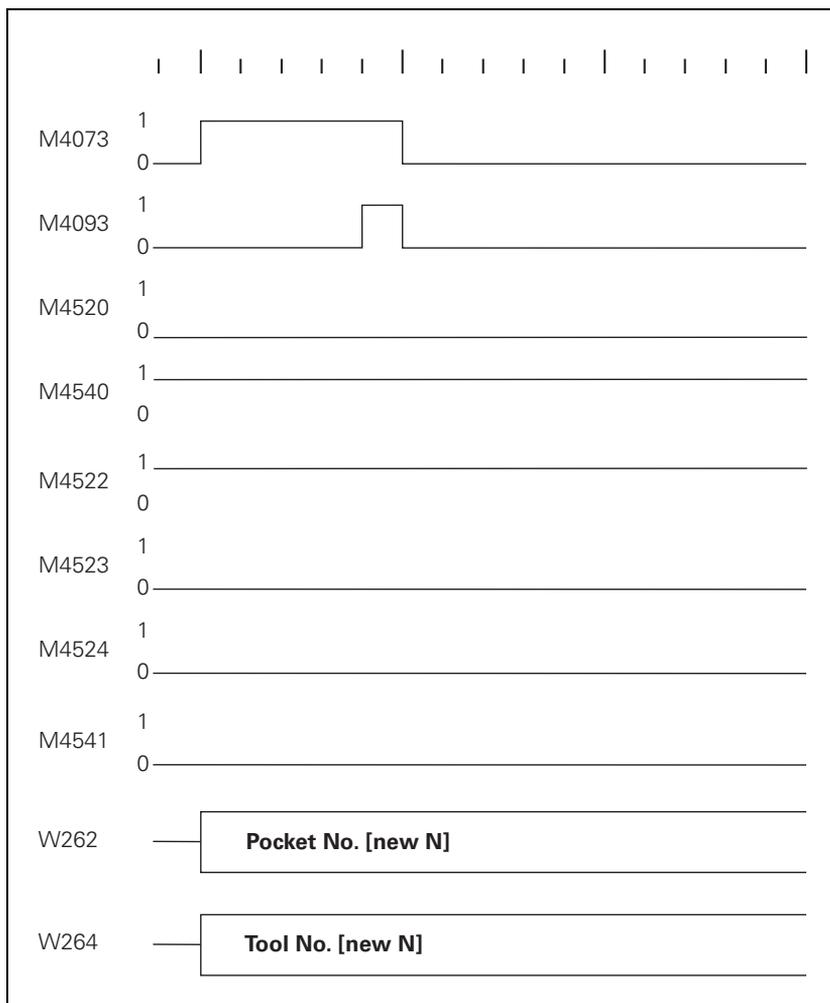
PL B/W/D <Status information>

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022
W1022	1	Status information invalid
	20	Call was not in a submit or spawn job

**N → N:
Normal tool follows
a normal tool**

The pocket number and the tool number of the called tool are transferred.



**S → N:
Normal tool follows
a special tool**

With this change sequence, two pocket numbers or two tool numbers must be transferred in succession. M4520 indicates that another **TOOL CALL** strobe (M4073) will follow:

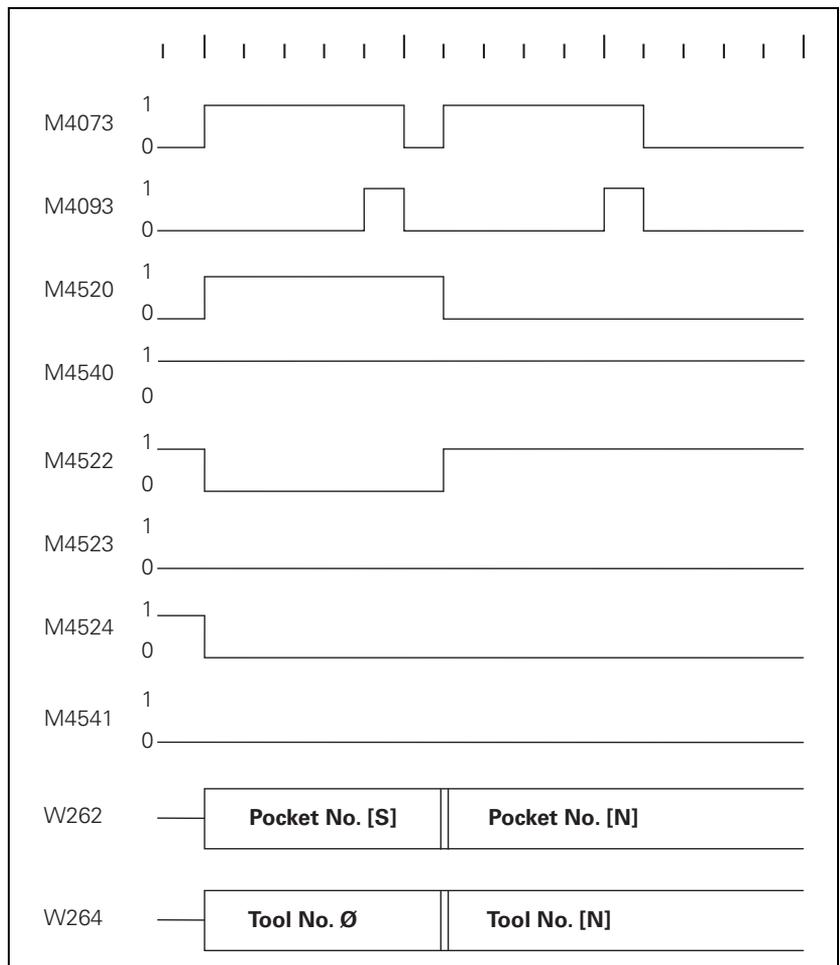
- ▶ With M4540 or MP7481.x specify the sequence in which the pocket numbers are transferred, depending on whether single or double changing arm.

**S → N,
Single changing
arm, M4540 = 0 or
MP7481.x, bit x = 0**

First the pocket number of the old tool and the tool number zero are transferred.

Zero means clear the spindle!

- ▶ Clear the spindle and acknowledge with M4093. Then the pocket and tool numbers of the new tool are transferred.



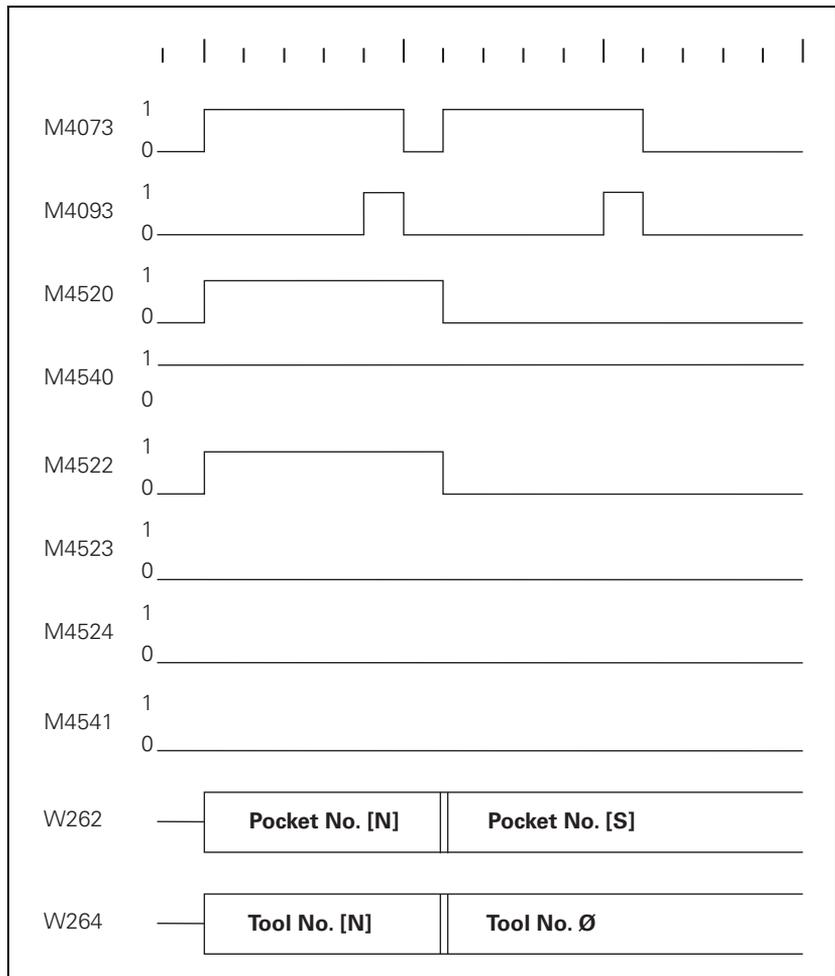
**S → N,
Double changing
arm, M4540 = 1 or
MP7481.x, bit x = 1**

First the pocket and tool numbers of the new tool are transferred.

► Acknowledge with M4093.

Then the pocket tool number of the old tool and the tool number zero are transferred.

Zero means clear the spindle!



**M → N:
Normal tool follows
a manual tool**

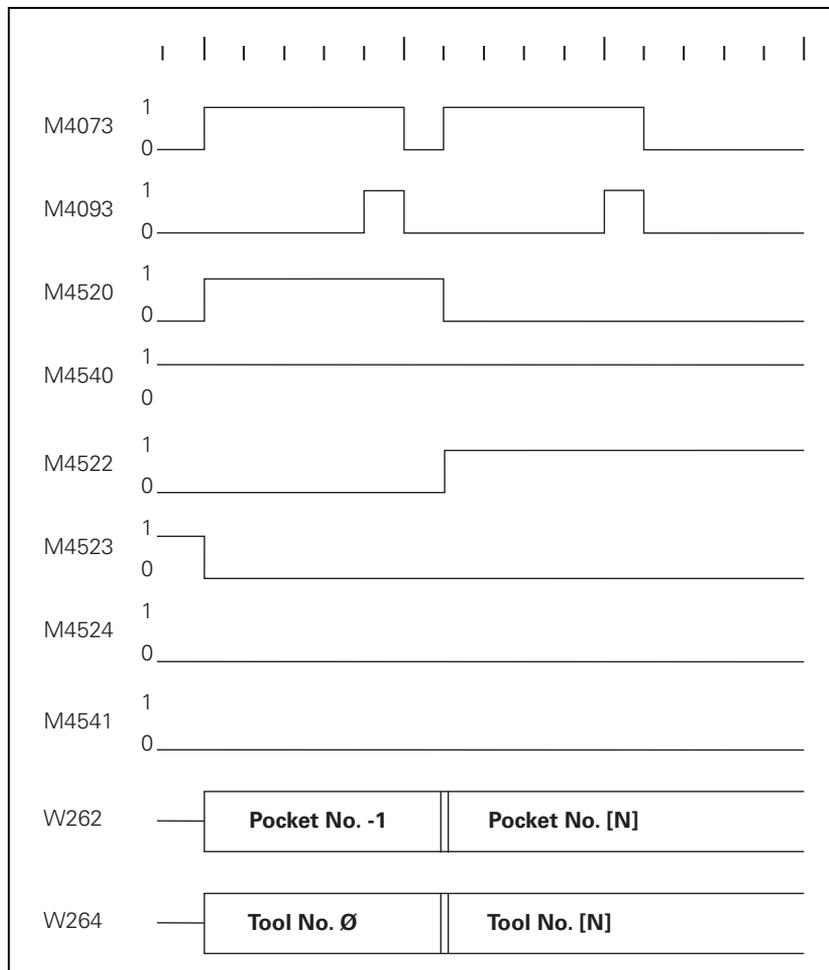
With this change sequence, two pocket numbers or two tool numbers must be transferred in succession. M4520 indicates that another **TOOL CALL** strobe (M4073) will follow. Regardless of M4540 or MP7481.x, the pocket number – 1 and tool number zero are transferred first.

Zero means clear the spindle!

Pocket number –1 means: no pocket in the tool magazine!

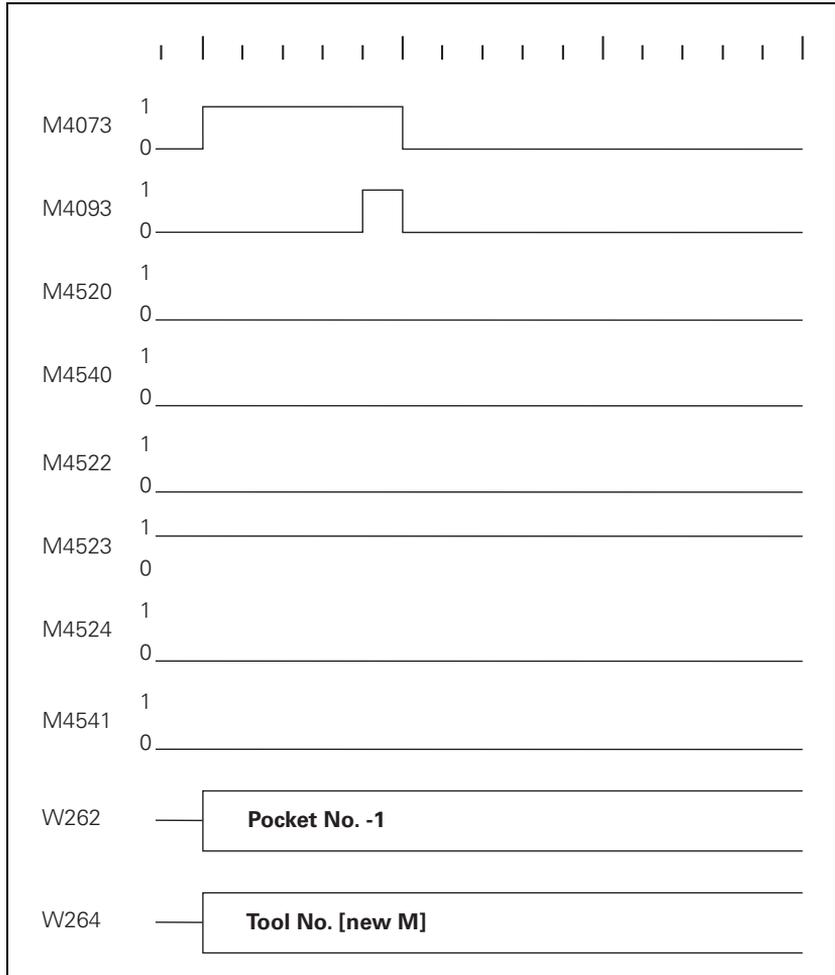
► Acknowledge with M4093.

Then the pocket number and tool number of the new, called tool are transferred.



M → M: Manual tool follows a manual tool

Pocket number -1 means: no pocket in the tool magazine!



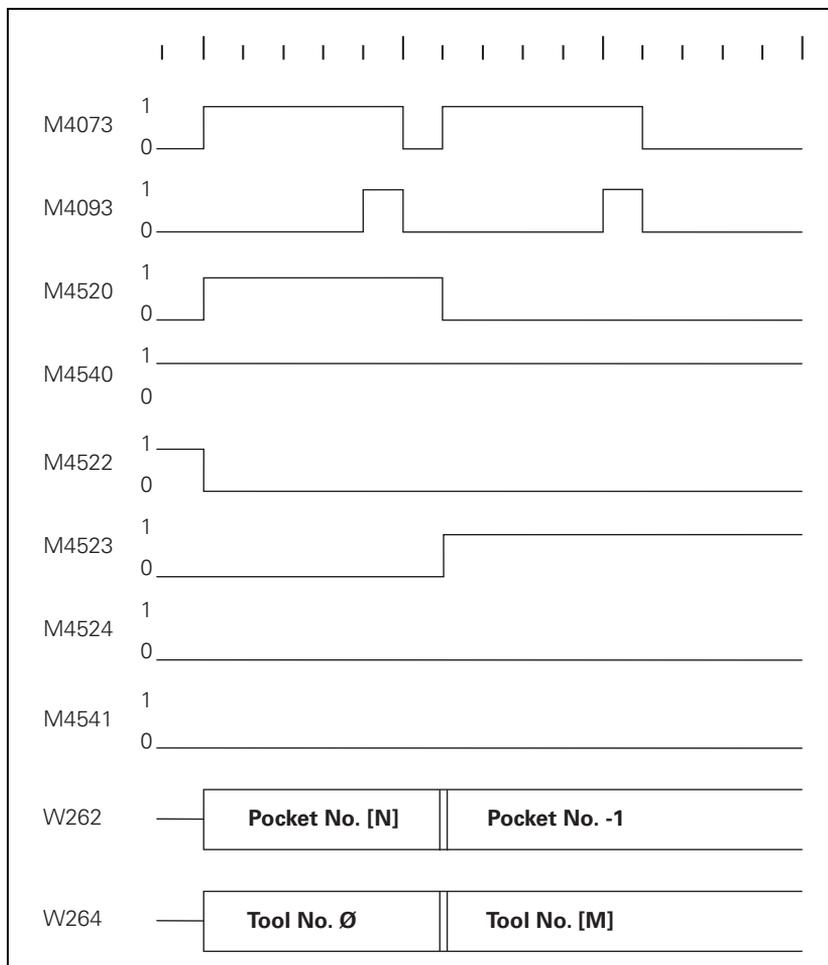
N → M: Manual tool follows a normal tool

With this change sequence, two pocket numbers or two tool numbers must be transferred in succession. M4520 indicates that another **TOOL CALL** strobe (M4073) will follow. Regardless of M4540 or MP7481.x, the pocket number of the old tool and tool number zero are transferred first.

Zero means clear the spindle!

- Acknowledge with M4093. Then the pocket number -1 and tool number of the new, called tool are transferred.

Pocket number -1 means: no pocket in the tool magazine!



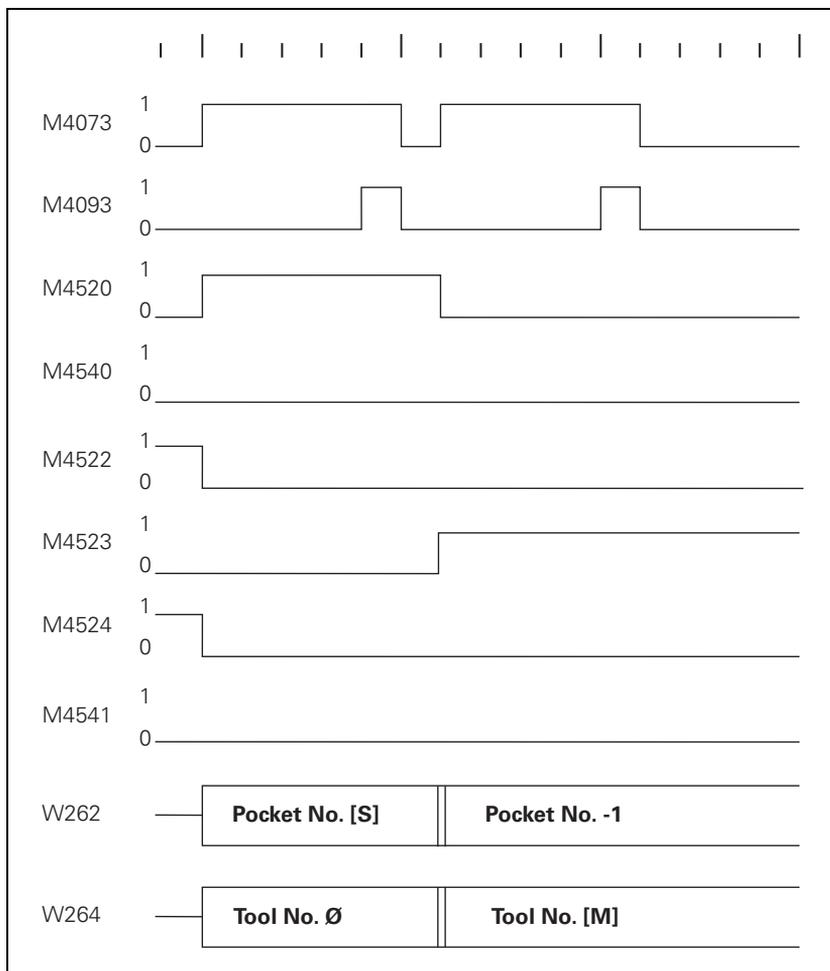
S → M: Manual tool follows a special tool

With this change sequence, two pocket numbers or two tool numbers must be transferred in succession. M4520 indicates that another **TOOL CALL** strobe (M4073) will follow. Regardless of M4540 or MP7481.x, the pocket number of the old tool and tool number zero are transferred first.

Zero means clear the spindle!

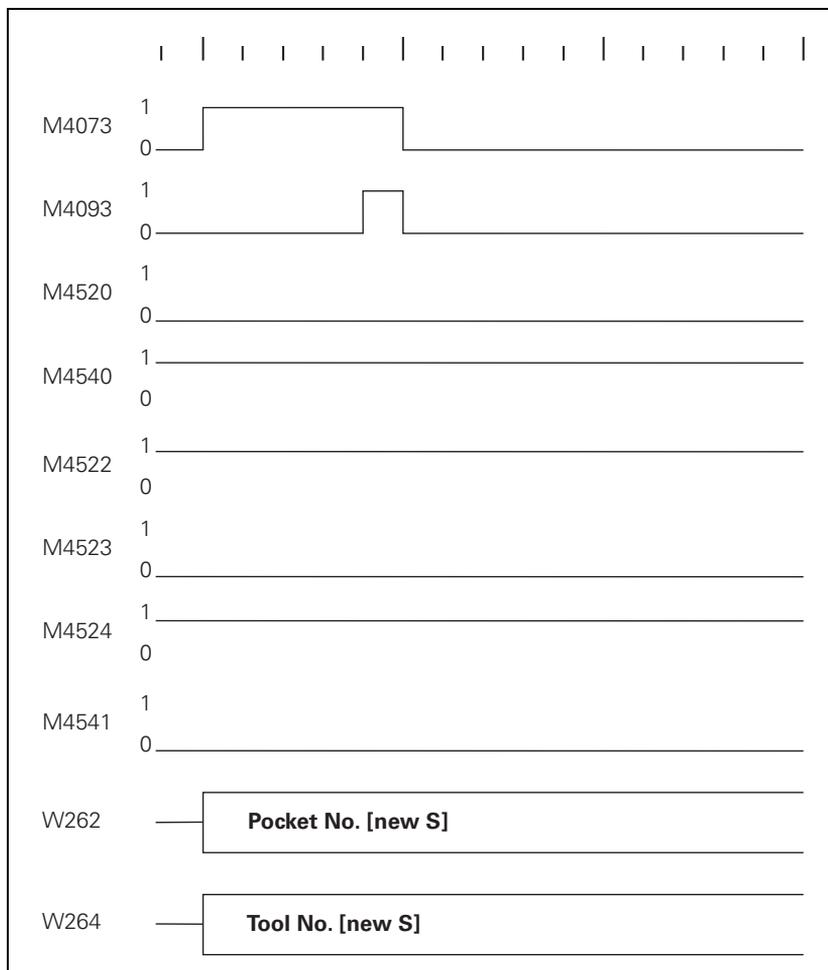
- Acknowledge with M4093. Then the pocket number -1 and tool number of the new, called tool are transferred.

Pocket number -1 means: no pocket in the tool magazine!



S → S: Special tool follows a special tool

- ▶ With M4541 or the column **F** in the pocket table, specify whether the special tool should be returned to the original pocket in spite of variable pocket coding.
 - No, M4541 = 0
The same logic program applies for single and double changer arms.
 - Yes, M4541 = 1
Single and double changer arms have different sequences of pocket number transfer.

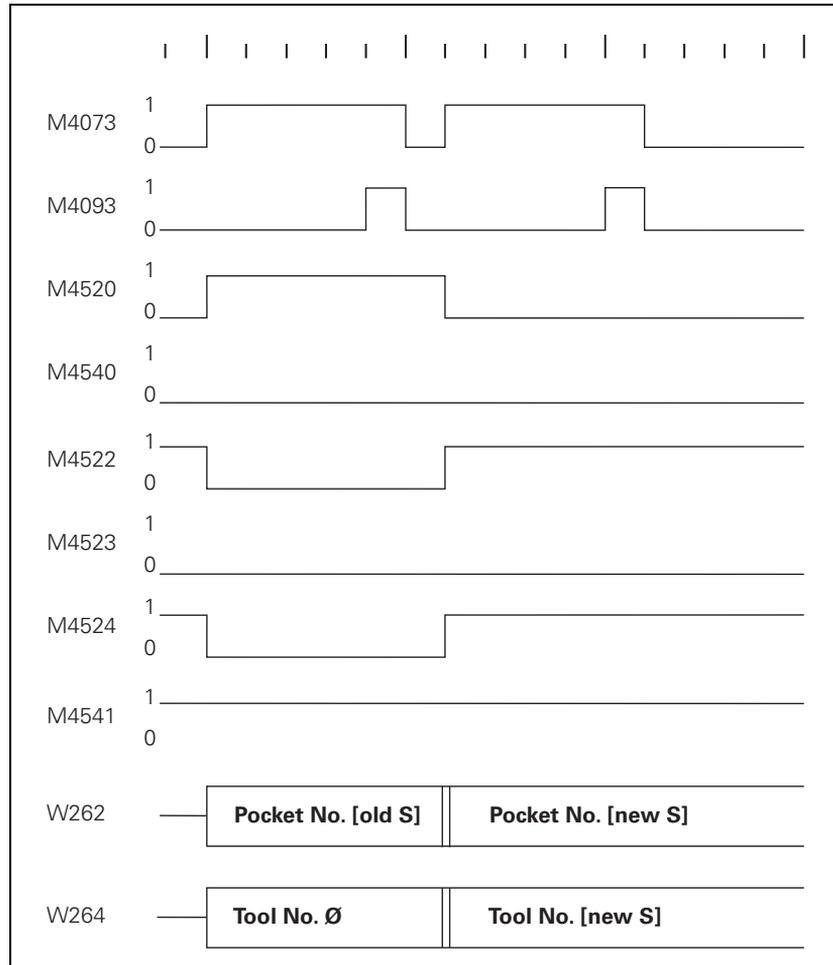


**S → S,
Single changing
arm, M4540 = 0 or
MP7481.x, bit x = 0**

First the pocket number of the old tool and the tool number zero are transferred.

Zero means clear the spindle!

- ▶ Acknowledge with M4093.
Then the pocket number and tool number of the new tool are transferred.



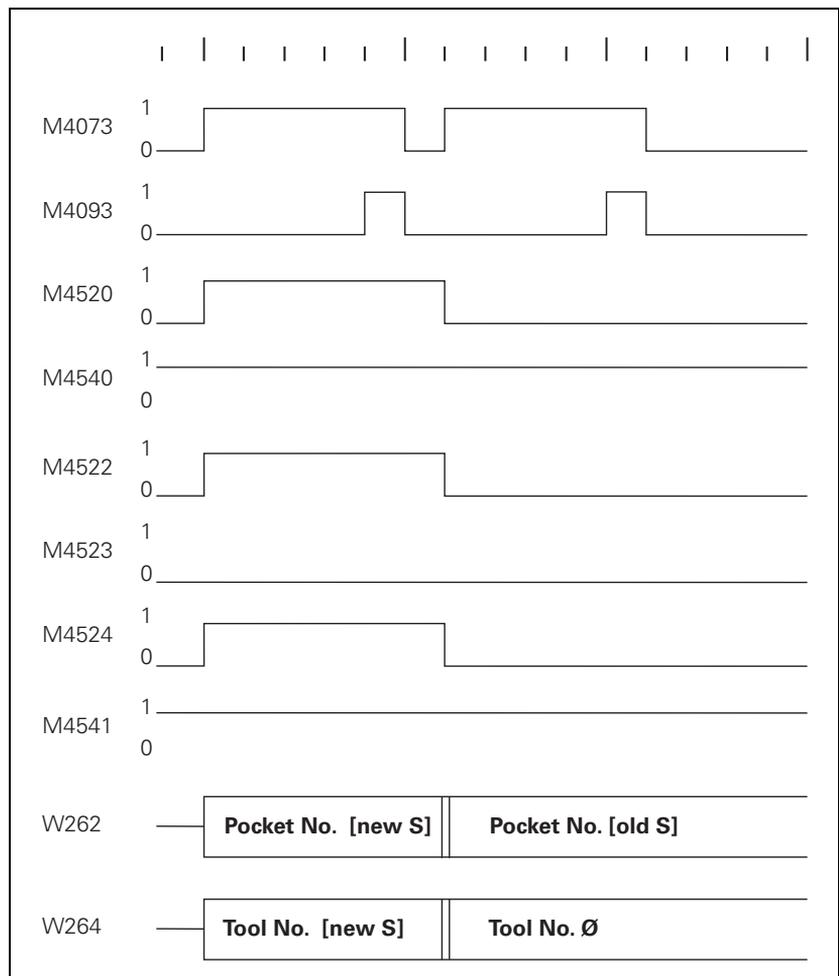
**S → S,
Double changing
arm, M4540 = 1 or
MP7481.x, bit x = 1**

First the pocket number and tool number of the new tool are transferred.

► Acknowledge with M4093.

Then the pocket tool number of the old tool and the tool number zero are transferred.

Zero means clear the spindle!



**N → S:
Special tool follows
a normal tool**

With this change sequence, two pocket numbers or two tool numbers must be transferred in succession. M4520 indicates that another **TOOL CALL** strobe (M4073) will follow. Regardless of M4541, there is a different sequence for the pocket number transfer for single and double-arm changers (M4540 or MP7481.x).

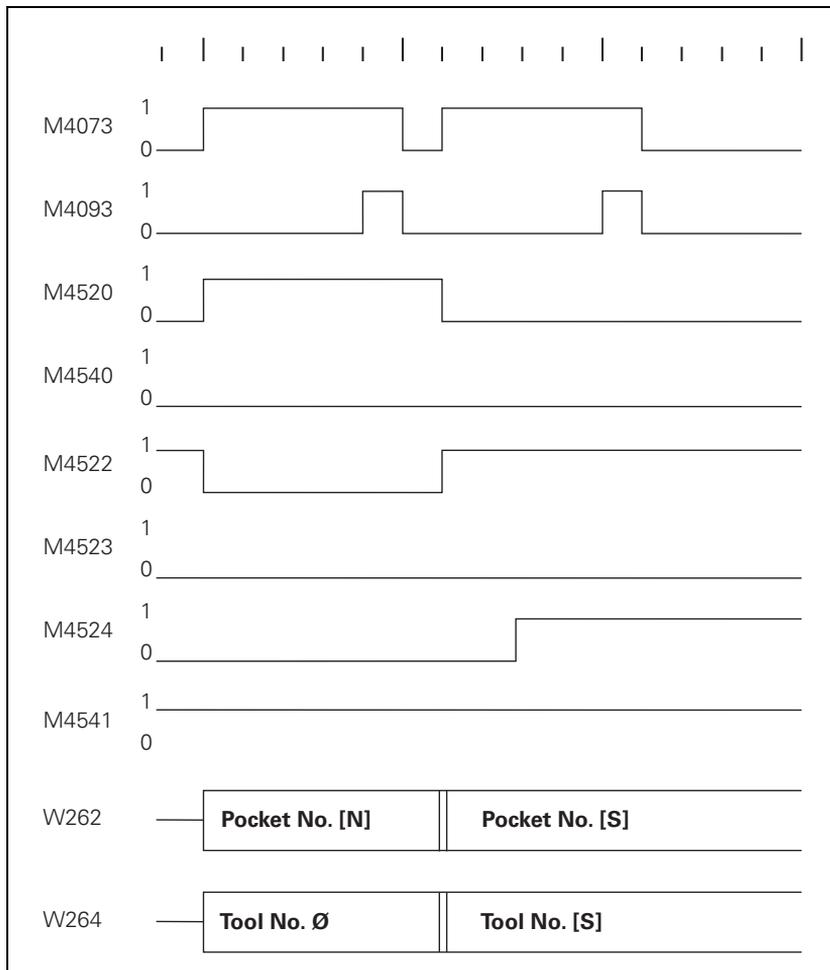
**N → S,
Single changing
arm, M4540 = 0 or
MP7481.x, bit x = 0**

First the pocket number of the old tool and the tool number zero are transferred.

Zero means clear the spindle!

► Acknowledge with M4093.

Then the pocket number and tool number of the new tool are transferred.



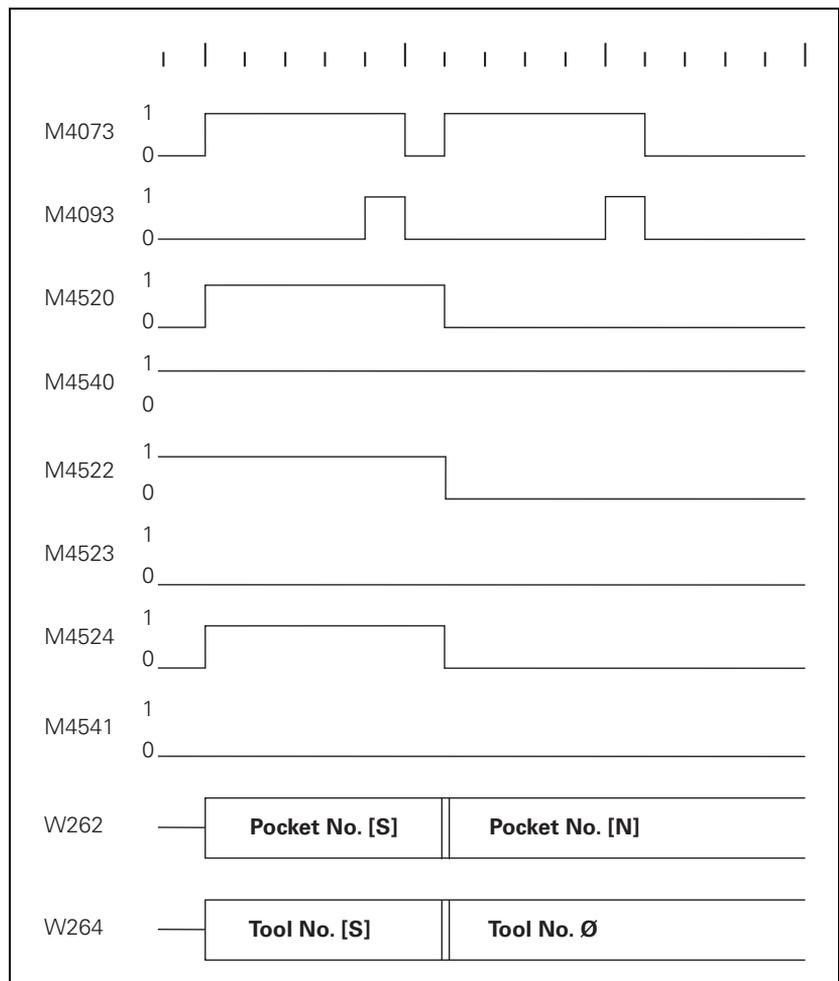
**N → S,
Double changing
arm, M4540 = 1 or
MP7481.x, bit x = 1**

First the pocket and tool numbers of the new tool are transferred.

► Acknowledge with M4093.

Then the pocket tool number of the old tool and the tool number zero are transferred.

Zero means clear the spindle!



**M → S:
Special tool follows
a manual tool**

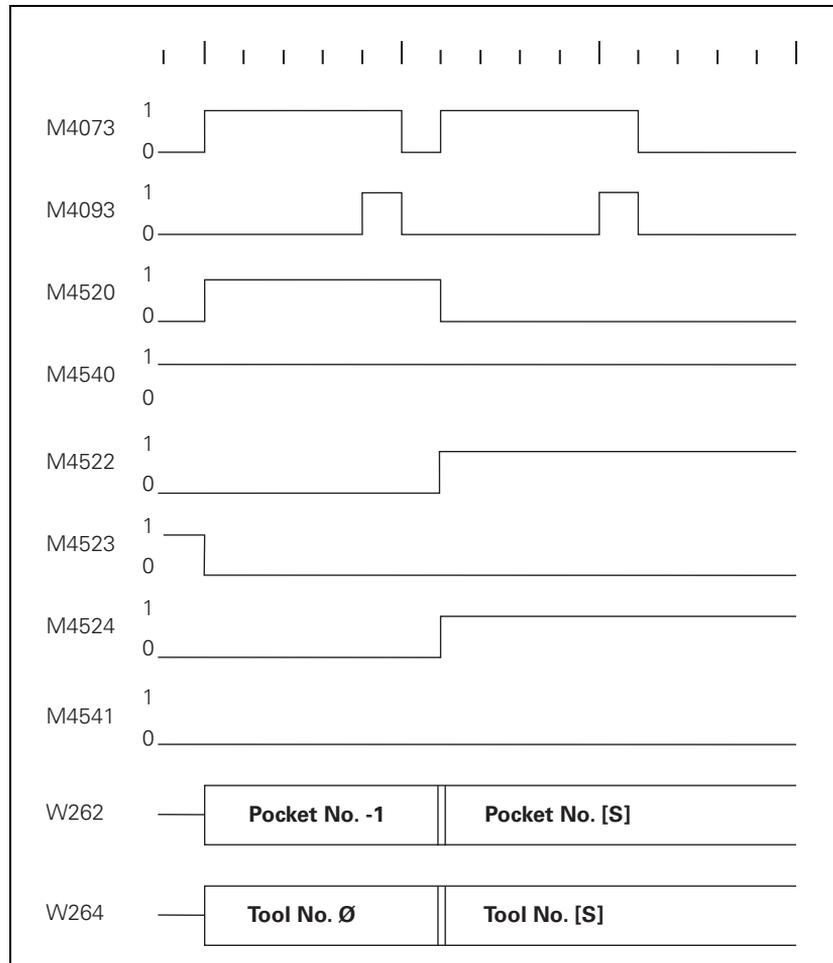
With this change sequence, two pocket numbers or two tool numbers must be transferred in succession. M4520 indicates that another **TOOL CALL** strobe (M4073) will follow. Regardless of M4541 and M4540 or MP7481.x, the pocket number -1 and tool number zero are transferred first.

Tool number zero means clear the spindle!

Pocket number -1 means: no pocket in the tool magazine!

► Acknowledge with M4093.

Then the pocket number and tool number of the new, called tool are transferred.



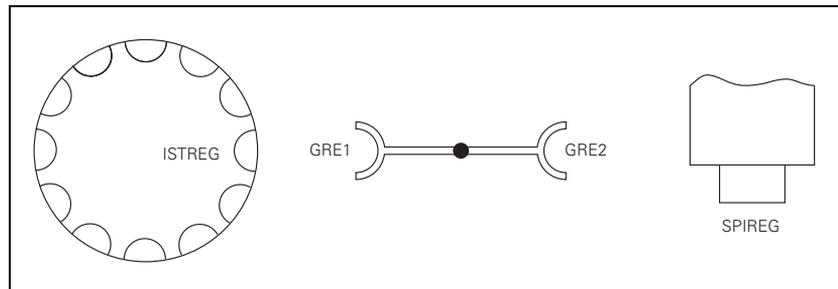
8.15.6 PLC programming example

Tool changer and basic flowcharts of the associated PLC program

► Create the PLC program with the PLC development software PLCdesign.

The tool changer treated here has the following features:

- Up to 254 tools
- Variable pocket coding, MP7480.x = 4
- Special tools allowed
- Providing the next tool with **TOOL DEF**
- Tool change with **TOOL CALL**
- Tools can be manually changed without pocket number definition in the tool table
- Double changing arm
- Special tools variable, MP4541 = 0



The following variables are used in the basic flowchart:

- ISTREG = The pocket number at the tool change position of the tool magazine
- GRE1 = Pocket number of tool in changing arm facing tool magazine
- GRE2 = Pocket number of the tool in the arm facing the spindle
- SPIREG = Pocket number of the tool in the spindle

		Set	Reset
W262	Tool pocket number	NC	NC
W264	Tool number	NC	NC
M4073	Strobe signal T code (P code) with TOOL CALL	NC	NC
M4074	Strobe signal T code (P code) with TOOL DEF	NC	NC
M4093	Acknowledgment of T code (P code) with TOOL CALL	NC	NC
M4094	Acknowledgment of T code (P code) with TOOL DEF	NC	NC
M4520	Additional T code (P code) follows with TOOL CALL	NC	NC
M4524	Special tool called (TOOL CALL)	NC	NC
M4540	Sequence of the tool numbers or pocket number transfer, M4520 = 1	PLC	PLC
M4541	Special tool to original pocket in spite of variable pocket coding	PLC	PLC

Machines parameter that are used:

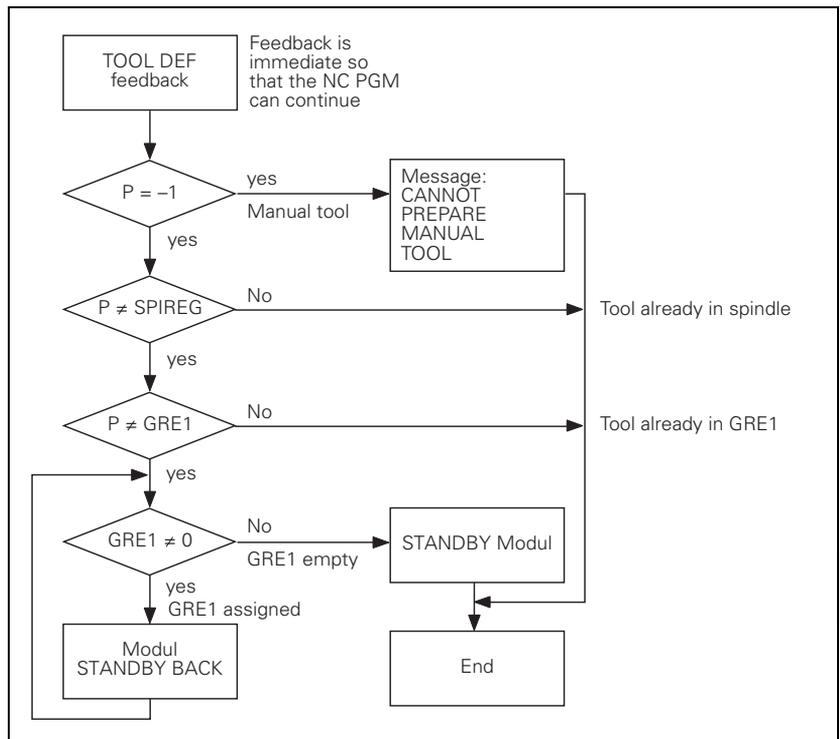
Machine parameter	Meaning
MP7260 = 90	Number of tools in the tool table
MP7261 = 12	Number of the pockets in the tool magazine
MP7480.0 = 4	Pocket number and tool number output for every TOOL CALL block.
MP7480.1 = 4	Pocket number and tool number output for every TOOL DEF block

The flowchart for this tool changer is divided into the following modules or subprograms:

Module	Meaning
TOOL DEF	Search for tool and load in GRE1
TOOL CALL	Automatic tool change
STANDBY	Search for tool and load in GRE1
STANDBY BACK	Return tool from GRE1 to the magazine
MANUAL TOOL IN	Manual tool follows a normal or special tool
MANUAL TOOL OUT	Normal or special tool follows a manual tool
MANUAL IN/OUT	Manual tool follows a manual tool
INSERT	Replace old tool with new tool
COMPUTE SHORTEST DIRECTION	
COMPARE P CODE WITH ISTREG	
COMPARE GRE1 WITH ISTREG	

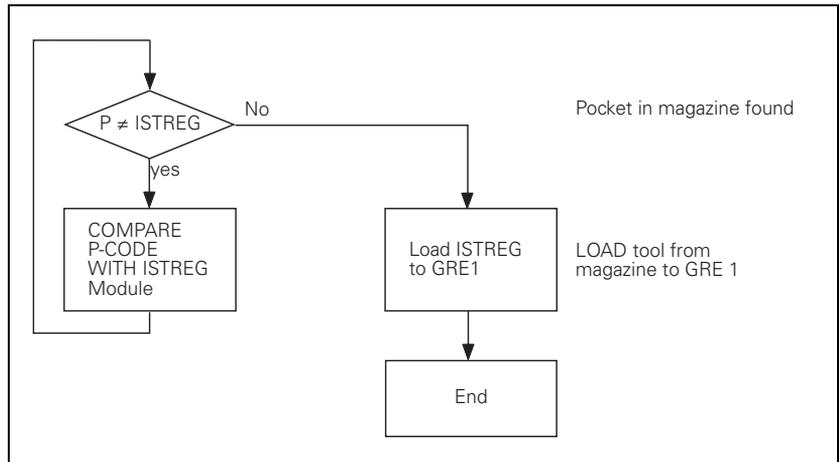
TOOL DEF program module

Search for tool and load in GRE1



**STANDBY BACK
program module**

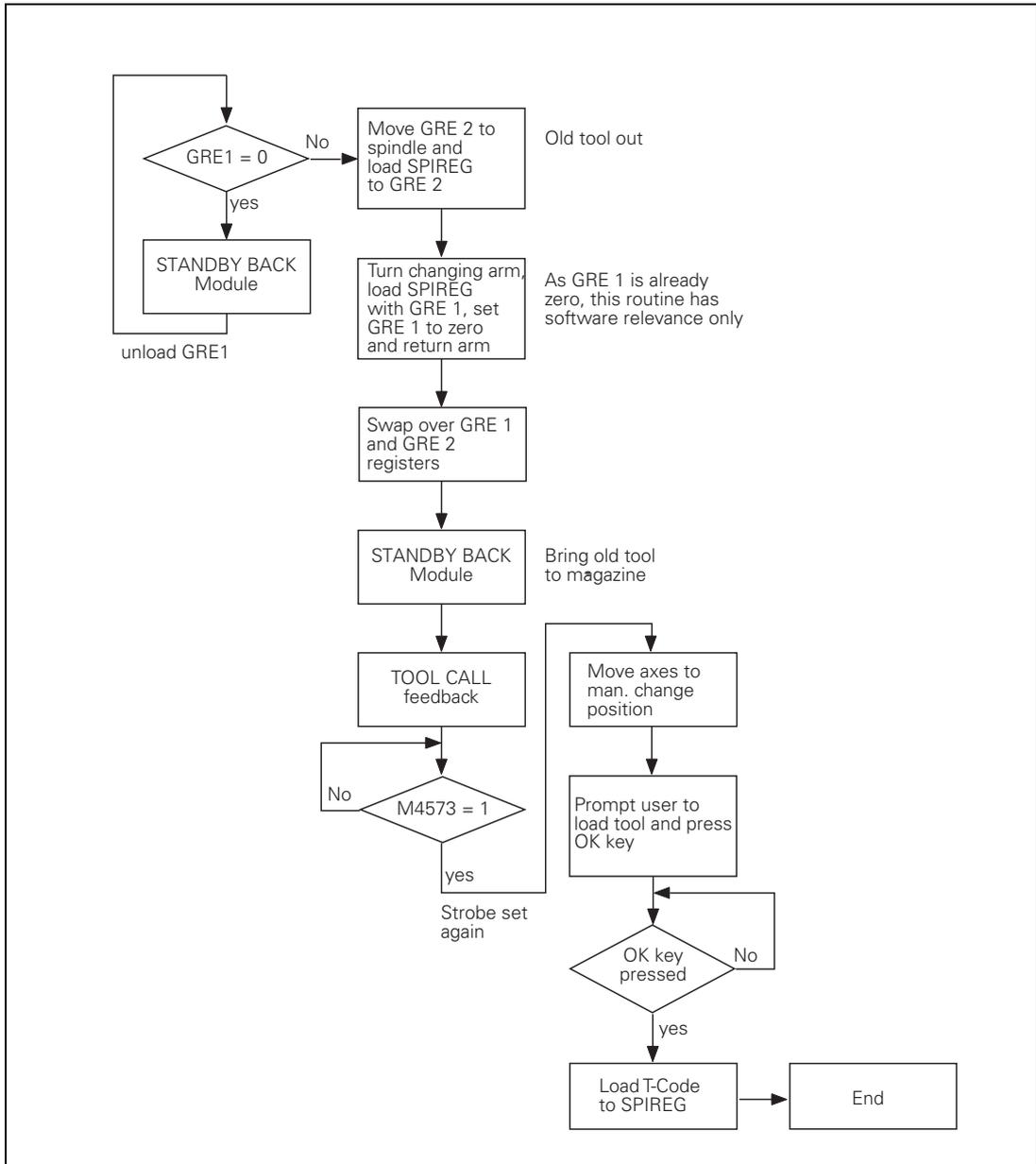
Return tool from GRE1 to the tool magazine



**MANUAL TOOL IN
program module**

N → M or S → M:

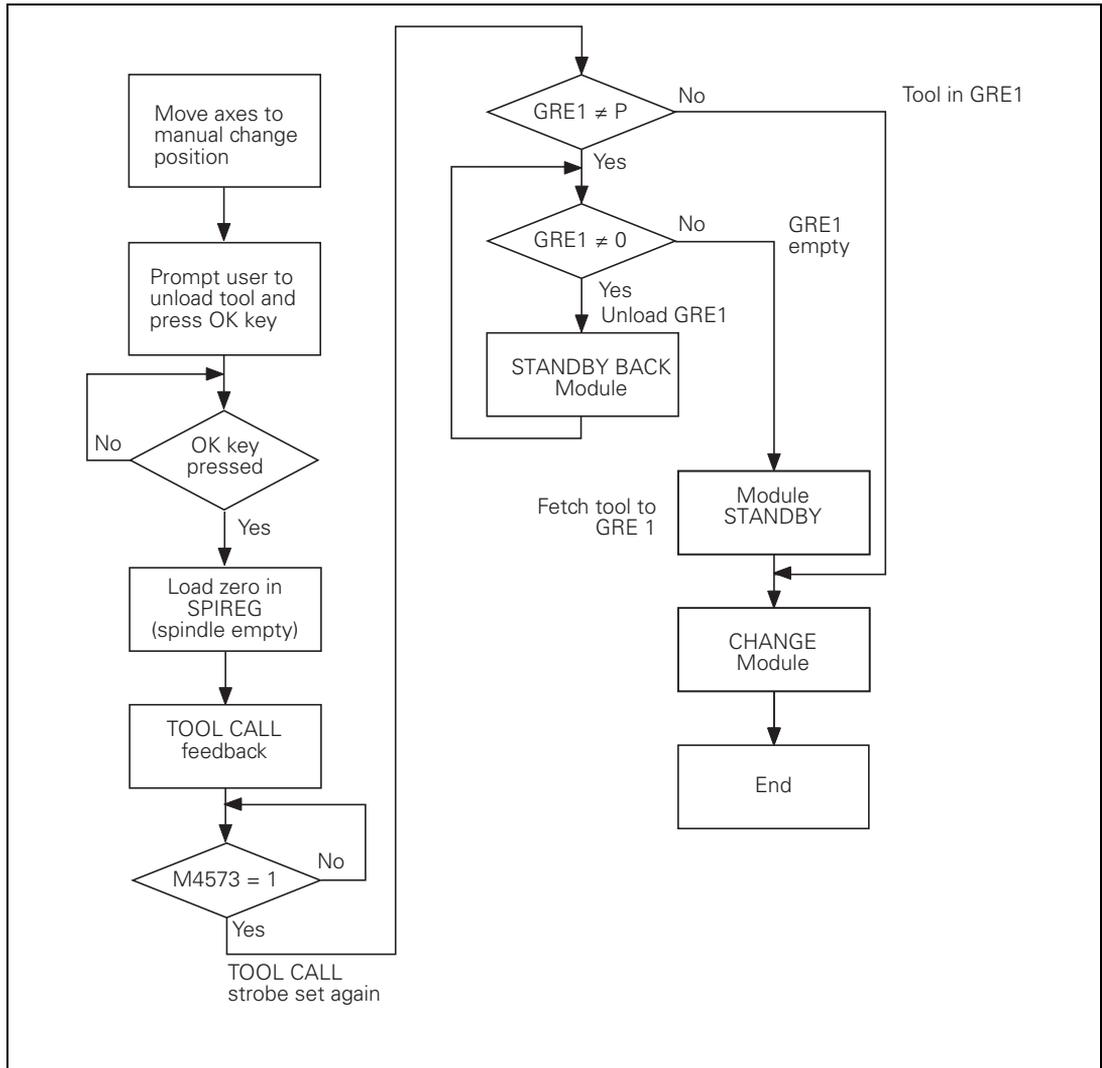
Manual tool follows a normal or special tool The old tool is placed in the tool magazine and the user is prompted to insert a manual tool (which is not in the tool magazine).



**MANUAL TOOL
OUT program
module**

M → N or M → S:

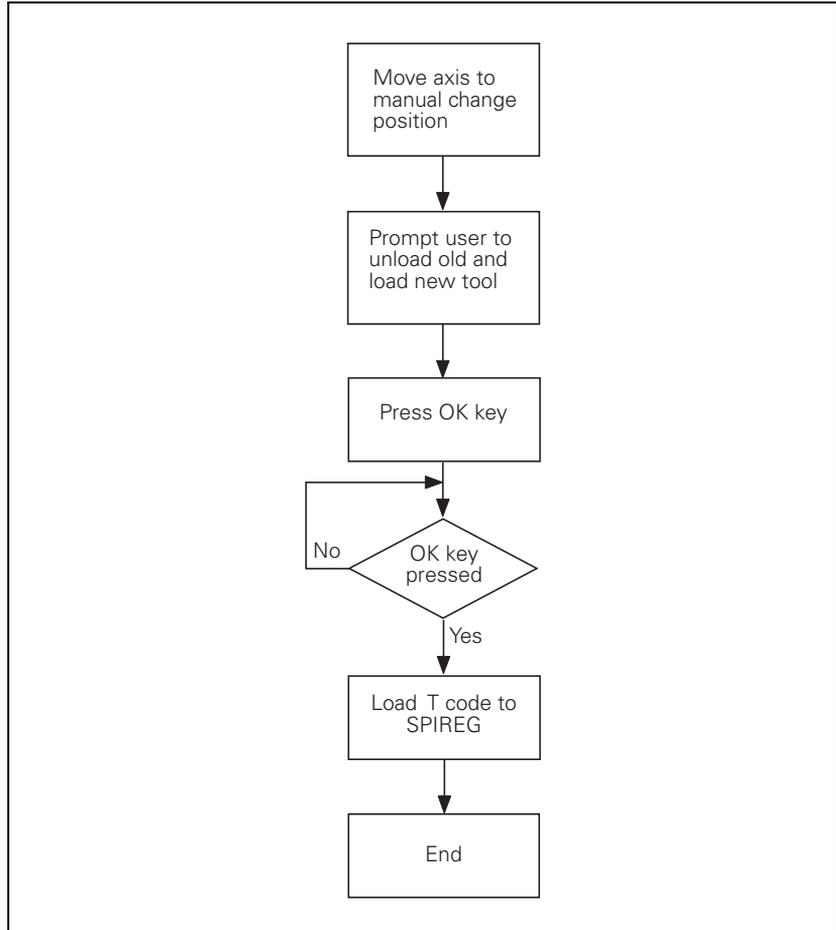
Normal or special tool follows a manual tool. The operator is prompted to empty the spindle manually, since there is no room in the tool magazine for the current tool. The called tool is automatically inserted.



**MANUAL TOOL
OUT/IN program
module**

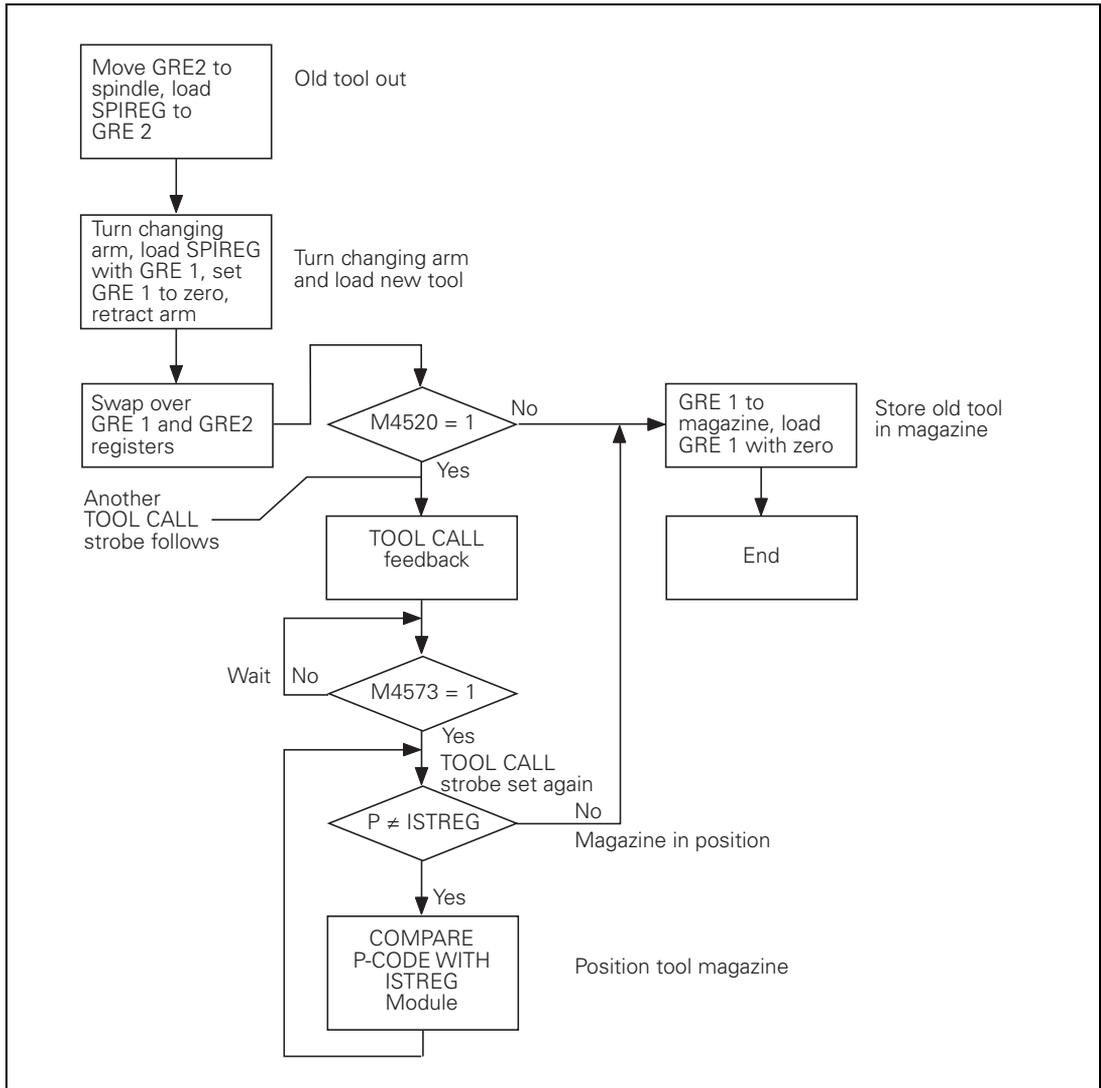
M → M:

Manual tool follows a manual tool. The user is prompted to remove the tool from the spindle manually and insert the new tool, since there is not room for the tools in the tool magazine.



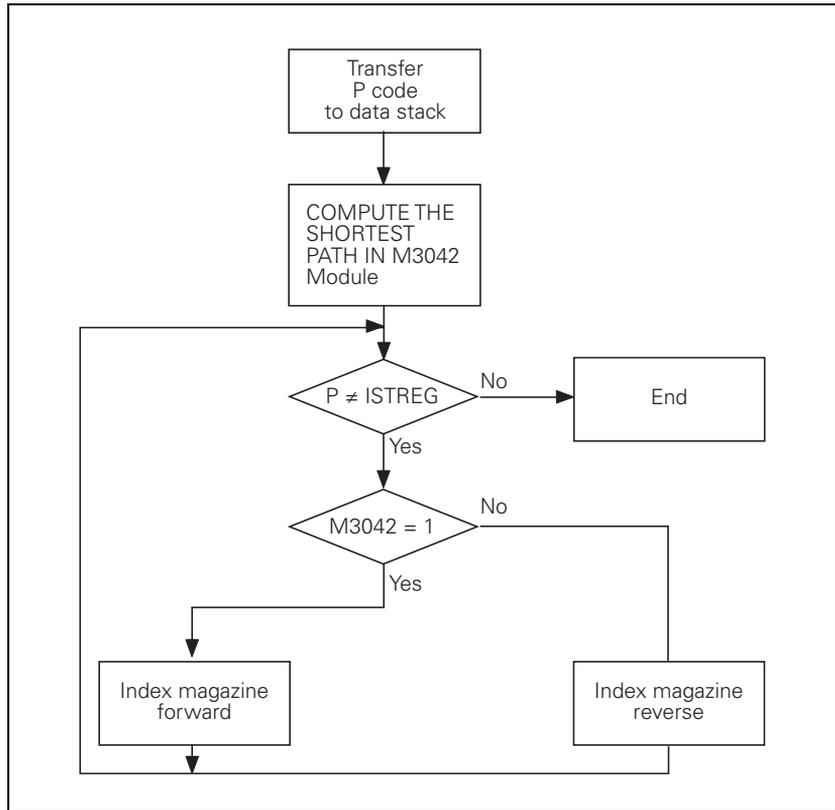
INSERT program module

The spindle is emptied and the new tool is automatically inserted. The program takes into account whether the tool should be returned to its original pocket in the tool magazine (e.g. special tool).



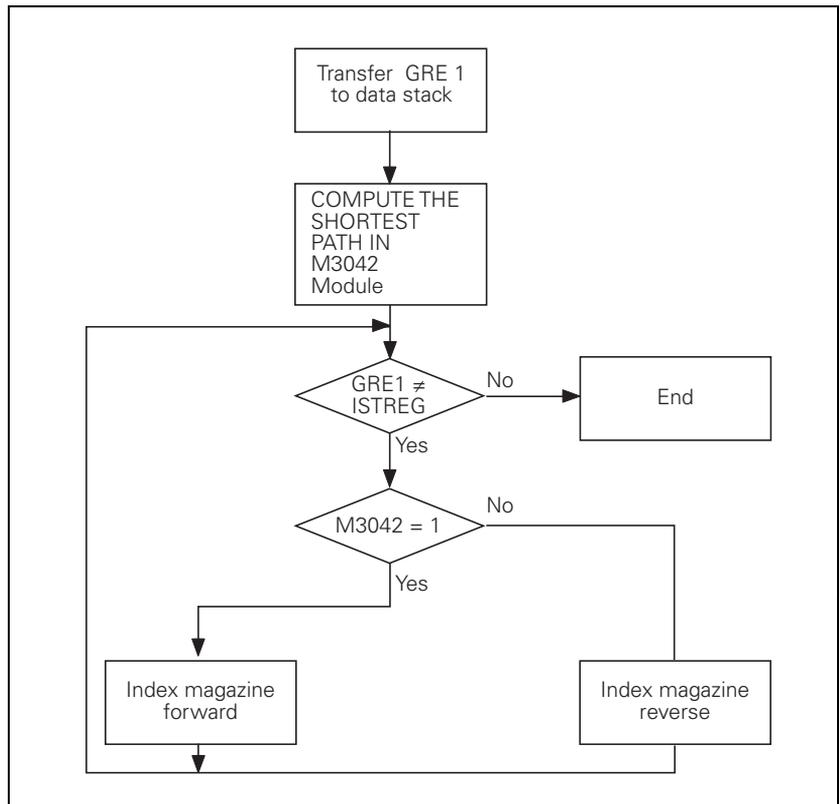
**COMPARE P CODE
WITH ISTREG
program module**

The tool magazine is positioned in the shortest direction to the desired pocket number.



**COMPARE GRE1
WITH ISTREG
program module**

The tool magazine is positioned in the shortest direction to the pocket number that is located in GRE1.



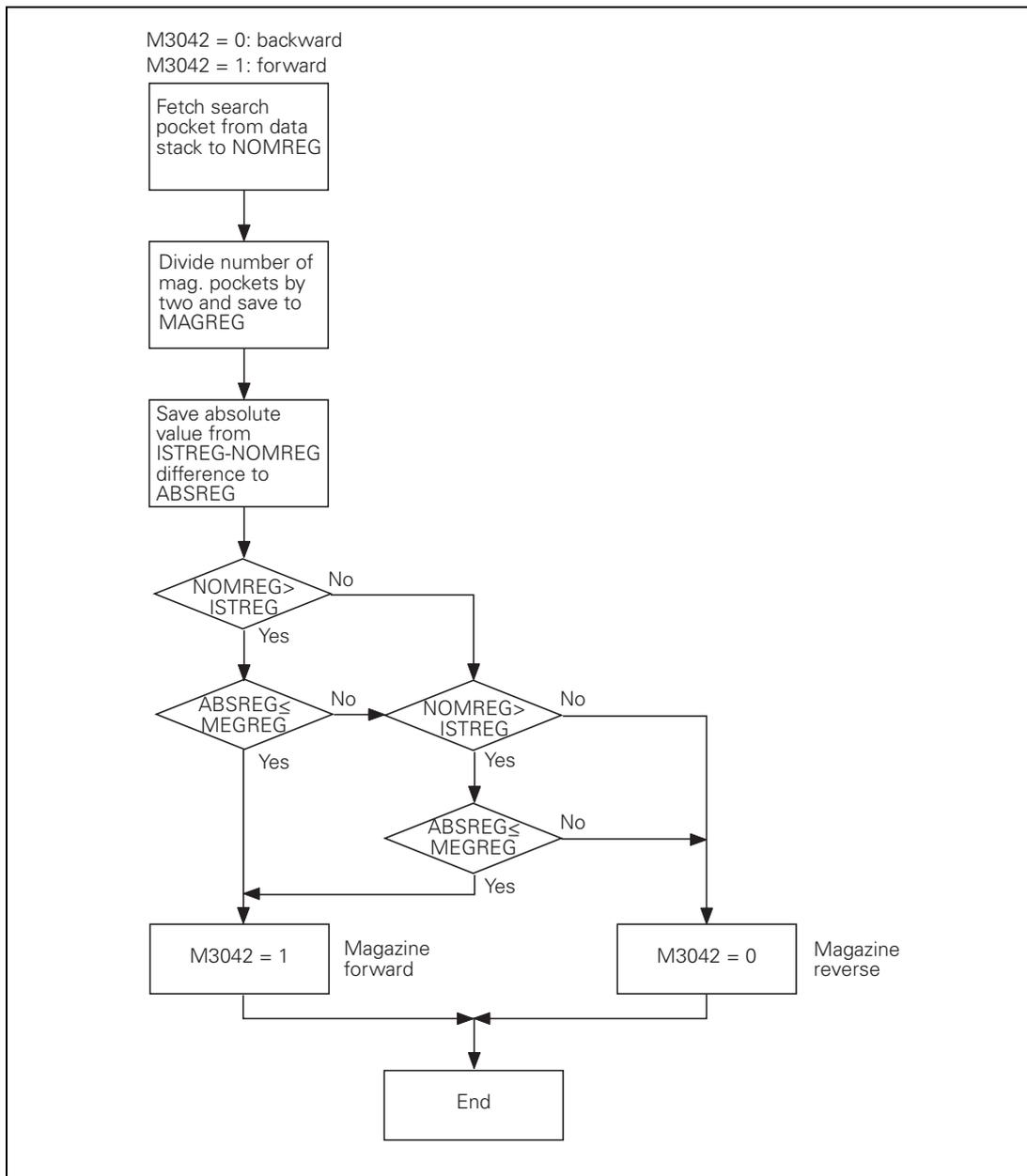
**COMPUTE
SHORTEST
DIRECTION
program module**

The PLC ascertains the direction of tool-magazine movement for the shortest traverse distance to the desired pocket number.

The direction is saved in M3042:

M3042= 0: Backward

M3042= 1: Forward



8.15.7 Extended tool management

Expanded tool management							Programming and editing
Tools							
T	NAME	PTYP	TL	POCKE	MAGAZINE	Tool life	REMAI
0	NULLWERKZEUG	0			Spindle	Not monitored	
1	D2	0		1	Main magazine	Not monitored	
2	D4	0		2	Main magazine	Not monitored	
3	D6	0		3	Main magazine	Not monitored	
4	D8	0		4	Main magazine	Not monitored	
5	D10	0		5	Main magazine	Not monitored	
6	D12	0		6	Main magazine	Not monitored	
7	D14	0		7	Main magazine	Not monitored	
8	D16	0		8	Main magazine	Not monitored	
9	D18	0		9	Main magazine	Not monitored	
10	D20	0		10	Main magazine	Not monitored	
11	D22	0		11	Main magazine	Not monitored	
12	D24	0		12	Main magazine	Not monitored	
13	D26	0		13	Main magazine	Not monitored	
14	D28	0		14	Main magazine	Not monitored	
15	D30	0		15	Main magazine	Not monitored	
16	D32	0		1	Add-on magazine	Not monitored	
17	D34	0		2	Add-on magazine	Not monitored	
18	D36	0		3	Add-on magazine	Not monitored	
19	D38	0		4	Add-on magazine	Not monitored	
20	D40	0		5	Add-on magazine	Not monitored	
21	D42	0		6	Add-on magazine	Not monitored	
22	D44	0		7	Add-on magazine	Not monitored	
23	D46	0		8	Add-on magazine	Not monitored	
24	D48	0		9	Add-on magazine	Not monitored	
25	D50	0		10	Add-on magazine	Not monitored	
26	D52	0		11	Add-on magazine	Not monitored	

BEGIN
END
PAGE
PAGE

FORM TOOL
END

With the extended tool management, you can provide many functions with regard to tool handling. Examples:

- Easily readable and, if you desire, adaptable representation of the tool data in tables and forms
- Any description of the individual tool data in the new table view
- Mixed representation of data from the tool table and the pocket table
- Fast sorting of all tool data by mouse
- Use of graphic aids, e.g. color coding of fields for different tool or magazine status
- List of all available tools of the current NC program
- Program-specific usage sequence of all tools
- Tool import and export

The new tool management function is included in the HEIDENHAIN standard features and can be activated any time. However, the familiar tables are active after the software has been installed.

The extended tool management function additionally includes the "tooling list" table and the "T usage order" table as software option 93 Extended Tool Management.

- Tooling list
List of all tools in the NC program that is selected in the Program Run mode (only if you have already created a tool usage file)
- T usage order
Sequence of all tools that are inserted in the program selected in the Program Run mode (only if you have already created a tool usage file)

You activate the extended tool management function through the PYTHON.SYS system file. The PLC:\PYTHON.SYS file links soft-key functions with a Python script, among other things. You will find a more detailed description of the PYTHON.SYS file in the "Python in the iTNC" supplement to the Technical Manual on HEIDENHAIN HESIS-Web including Filebase.

The standard PYTHON.SYS provides three options (A, B, C). Each of them activates a different variant of the tool management function:

■ Variant A

The new tool management function is started/activated directly through the TOOL TABLE soft key

This always activates the new tool management. The previous tables (tool table, pocket table) are no longer available to the user.

■ Variant B

The new tool management function is started directly through the TOOL TABLE soft key after a reset/activation

Same as variant A, but the tables of the new tool management function are already loaded in the background when the **Power interrupted** message is acknowledged. The time required for opening the new tables after pressing the TOOL TABLE soft key is reduced, especially for large tool tables and pocket tables.

■ Variant C

The new tool management function is started through the TOOL MANAGEMENT soft key from the tool table, pocket table and the Program Run, Full Sequence mode of operation after a reset/activation

The familiar soft keys TOOL TABLE and POCKET TABLE can continue to be used to activate the previous tables. The new tool management function can be started by pressing the TOOL MANAGEMENT soft key in the last soft-key row that appears after pressing the TOOL TABLE, POCKET TABLE soft key in the **Program Run, Full Sequence** mode of operation.

You do not need to activate software option 46 (Python OEM Process) if you want to use one of the above-mentioned standard features.



Note

If the extended tool management function is active, a TNC backup can only be restored before the **Power interrupted** message has been acknowledged.

Later the tool table and the pocket table are constantly exchanging data with the Python scripts and are therefore locked for the restore process.

Also, the tool management function can be configured to meet your requirements. The following rule files and configuration files are used for configuration:

- OEMSettings.py
- OEMRuleList.py
- OEMRules.py
- OEMEntryList.py
- OEMEntries.py

The new tool management function is configured using the columns from the previous tool table and pocket table. These remain available in the background. The tool management function mainly changes the user interface. In addition, columns with new contents can be created for the new tool management function by linking the data of the previous table columns (e.g. by calculating new values).

OEMSettings.py

Configuration file for OEM-specific extended tool management

In this file, you specify the OEM-specific files and settings. This way, you can also configure individual parts of the extended tool management function on your own.

You need to save the OEMSettings.py in the directory **PLC:\Python\ETMC\gui_jh**. If the file exists, the TNC evaluates the entries of this file. In this file, all OEM-specific entries are marked as comments by default, and are therefore not active. If you activate individual entries, the TNC will use this OEM-specific configuration.

The paths indicated in OEMSettings.py are relative paths to the file itself. You should not change any paths or file names to ensure that the complete project is easily understandable if servicing becomes necessary. Therefore, you must save the OEM-specific files with the paths and files names indicated in OEMSettings.py.

OEM-specific files offer you the possibility of defining the following default settings:

- Rule-based configuration of table columns and tables of the tool table and pocket table. If software option 93 is active, this also applies to the tooling list and T usage order.
- Language-sensitive files for the texts used
- Rule-based configuration of fillable forms
- Soft-key row of magazine management
- Context menu for magazine management

Definition of the following views in the rule-based configuration file OEMRuleList.py:

- Tool table
- Pocket table
- Tooling list (software option 93)
- T usage order (software option 93)

The respective views (tables) are defined in the OEMRuleList.py file by a sequence of column configurations (saved in OEMRules.py). The individual tables are configured by specifying the column names. The column names used must match those used in the OEMRules.py file for the definition of the columns.

The individual tables have a permanent identifier (key) that is used to assign the column list to the table:

- **rule_t**: Identifier for tool table
- **rule_p**: Identifier for pocket table
- **rule_u**: Identifier for tooling list
- **rule_s**: Identifier for T usage order

The columns are displayed in the configured view in exactly the sequence in which they were defined.

The structure of the list of desired columns must match that of a Python list. Example of configuring a tool table:

```
rule_t = [
    rule_tt
    ,rule_pocket
    ,rule_time
    ,rule_rtime
    ,rule_mag
    ,rule_tplcbit0
    ,
]
range_s = '19'
```

This example results in a tool table that consists of the six specified columns. The structure of the columns is defined in the OEMRules.py file.

HEIDENHAIN recommends creating tables with a maximum of 1500 lines. Basically, more entries are possible, but this affects computing performance and may lead to unnecessary waiting times when accessing tables.

The entry **range_s** = '<number>' specifies the number of precalculated tool calls for the "T usage order" table. Remember: The larger the number of values to be calculated, the higher the load on the control. Look-ahead calculation is possible for up to 50 tools. If you enter a fixed value in '<number>', the number of lines displayed in the table will be exactly the number of lines specified. If you use a fixed value in NC programs with many tool calls, not all of the tool calls of the program might be displayed. This can be avoided by entering a '+' sign in addition to the number, e.g. '30+'. In this way, you can ensure that at least all of the tools (tool calls) programmed in the NC program will be displayed in the "T usage order" table. Otherwise, the entered number of precalculated tool calls will be displayed (here: 30).

OEMRules.py

Rule-based configuration file for columns

Rules are used to define the individual columns in the configuration file OEMRules.py. Using rules, you can use or process the columns of the known tool or pocket table and the two tool usage lists, and display them in the new tool management.

Every column defined by the configuration can be used in multiple views if the possible tables links allow it.

The data of the existing views (tables) are accessed in the same way as table data are usually accessed from Python. The following tokens are currently possible for tool management:

- **TOOL** – active tool table
- **TOOL_P** – active pocket table
- **GE0\channel\0\toolPrep\toolingList** – list of all available tools of the currently selected NC program (software option 93)
- **GE0\channel\0\toolPrep\toolSequence** – tool usage order in the currently selected NC program (software option 93)

Configuration

In general, the JH Python functions make it possible to read the tables in the control line by line. A string called the ID string must be entered for identifying each individual line. An ID string must identify the path to the table and the desired data contents (also see the supplement to the Technical Manual: Python in HEIDENHAIN controls).

Examples:

- The line identified by "T=5" is read from the "TOOL" tool table:
`\\TABLE\\TOOL\\T\\5*`
- The length "L" entered in the line identified by "T=5" is read from the "TOOL" tool table:
`\\TABLE\\TOOL\\T\\5\\L`
- The complete line identified by "P=1.12" is read from the "TOOL_P" pocket table:
`\\TABLE\\TOOL_P\\P\\1.12*`

An asterisk ("*") at the end of the ID string means that the complete line is read. But you can also specify a certain column, e.g. "L". Then only the cell value of the desired line and column will be returned.

Tool management views

The extended tool management function makes it possible to display the previous tables of the control in a different view. The described functions and rules enable you to "build" the new enhanced tool-management tables. The respective views of the new tool-management tables (tools, pockets, tooling list, T usage order, etc.) are each based on an existing table.

- The "Tools" view is based on the table `\\TABLE\\TOOL\\`
- The "Pockets" view is based on the table `\\TABLE\\TOOL_P\\`
- The "Tooling list" view is based on the table `\\GEO\\channel\\0\\toolPrep\\toolingList`
- The "T usage list" view is based on the table `\\GEO\\channel\\0\\toolPrep\\toolSequence`

This means that the previously known tool table provides the basic information for displaying the "Tool" view, for example. Therefore, the number of lines in the "Tools" view is identical to the number of lines in the tool table. All lines of the tool table are read in order to display the "Tools" view. In the "Tools" view, one line is created for every line read, and the table cells in the view are displayed according to the column rules.

Column rules

A view (table) of the extended tool management function is defined column by column by specifying column rules. The column rules are summarized in the `OEMRuleList.py` file and define the configuration of a complete view (table). In the `OEMRules.py` file, however, individual columns are defined, which can be used (more than once) in different views. A column rule ultimately results in the value for a table cell of the view. The value of a table cell is defined by entering a data path in the form of an ID string in the respective column definition. Because a column rule does not contain any information about a certain line, a placeholder must be used as a line identifier. A question mark "?" is normally used as a placeholder.

Example:

```
rule_tt = {'value': (('\\TABLE\\TOOL\\T\\'?\\T', '=') ,)}
```

Meaning:

This rule results in the value of column 'T' from the tool table. If the "Tools" view is active, the 'T' value from the respective line of the tool table is used in the corresponding table cell of the view, for example. If a view other than "Tools" is active, the value to be entered in the table cell must first be determined by linking the tables.

For more examples of column rules, see page 1652.



Linking tables

You can specify different source data paths in the configuration of a view (e.g. "Tools") in order to display data from different source tables in a view.

Example:

```
rule_tt = t{'value': (('\\TABLE\\TOOL\\T\\?\\T', '=') ,)}  
rule_tp = t{'value': (('\\TABLE\\TOOL_P\\P\\?\\P', '=') ,)}
```

The tables are linked in the OEMRuleList.py file:

```
rule_t = [rule_tt, rule_tp]
```

Meaning:

Two columns are displayed in the "Tools" view:

1st column 'T' from the tool table and

2nd pocket number 'P' of the tool from the pocket table (if present)

In this case, both source tables must contain the column 'T' for the tables to be linked. Also, the value in column 'T' must have the same meaning in both source tables (e.g. tool number). Tables that are not related to each other cannot be linked to each other.

The placeholder ? in the rule "rule_tp" has a completely different meaning from the "Tools" point of view than it has from the "Pockets" point of view:

- rule_tp when used from a "Tools" point of view:
Show pocket number from the pocket table for the tool currently being read from the tool table
- rule_tp when used from a "Pockets" point of view:
Show pocket number of the line currently being read from the pocket table

Possible table links

Every view only allows certain links, because the related column values are not always unique.

- Valid data paths for the "Tools" view
 - \\TABLE\\TOOL\\ [basic information]
 - \\TABLE\\TOOL_P\\
- Valid data paths for the "Pockets" view
 - \\TABLE\\TOOL_P\\ [basic information]
 - \\TABLE\\TOOL\\
- Valid data paths for the "Tooling list" view
 - \\GEO\\channel\\0\\toolPrep\\toolingList [basic information]
 - \\TABLE\\TOOL\\
 - \\TABLE\\TOOL_P\\
- Valid data paths for the "T usage list " view
 - \\GEO\\channel\\0\\toolPrep\\toolSequence [basic information]
 - \\TABLE\\TOOL\\
 - \\TABLE\\TOOL_P\\

If you are in the Tools" or "Pockets" view, you cannot create a link to the "Tooling list" and "T usage order" tables, because there is no unique value in a common column (here 'T'). The same applies to linking values of the "Tooling list" to values of the "T usage order."

The prerequisite for linking, i.e. the relationship between the "Tools" view and the "Pockets" view, is the unique column 'T' (tool number). However, a tool can occur more than once in the "Tooling list" and "T usage order" views (e.g. because of multiple tool calls in an NC program).

Column definition: The description of a column of the view (table) consists of multiple configuration settings. The structure of a column description must correspond to that of a Python dictionary:

Entry	Description
'name':	Column identifier that is displayed as the column title (e.g. 'T')
'type':	Data type of values in the column ('float' for numbers, 'str' for string, 'check' for check box). The 'float' and 'str' column types differ only in the type of sorting used (sorting by string or numerical value). The 'check' column type generates a column with check boxes.
'sort':	Conditions for sorting <ul style="list-style-type: none"> ■ 'blanks_on_top', blank cells at the beginning of the table ■ 'blanks_on_bottom', blank cells at the end of the table ■ 'number_before_letter', numerical values before strings ■ 'letter_before_number', strings before numerical values <p>The numerical values and strings are each sorted separately. The settings for blank cells and number/string sorting can be combined.</p>
'align':	Text alignment in the column ('left'/'right'/'center')
'gettext':	Translate or do not translate the cell content ('value') with the gettext function ('true'/'false')
'value':	Value to be used as cell content. Further rules can be used to adapt the value. The rules are executed sequentially one after the other, thereby enabling you to nest the rules. The last applicable rule determines the displayed value.
optional 'background':	Color to be used for the background of the cell Rules can be used to define a background color for the cell.
optional 'background-pix':	A semi-transparent window is shown in the currently selected line to make the line better visible. The background color of the cell shines through the cursor ('true'/'false'). For this purpose, an area to the left of the text is reserved in a cell. This area then contains a transparent, colored square.
optional 'condition':	The content of the cell ('value') will be displayed or the cell remains blank, depending on a condition. If the result of the condition is 'false', the value will not be displayed and the cell remains blank.



A column definition is evaluated for every read line of the table that provides the basic information. Linking the line to be evaluated to the column definition to be used therefore results in exactly one cell value of the view. This also means that the computing time required for displaying a view increases significantly with the number of columns defined.

The system automatically tries to translate the **'name'** entries by means of the **'gettext'** function. For the cell contents (**'value'**), however, the rule **'gettext'='true'** or **'false'** must be used to activate translation.

The language-sensitive *.po files for the gettext function can be created through PLCtext and must be located in the directory **PLC:\Python\LANGUAGE**. For more information about this, refer to the Python texts help topic in the PLCtext help.

If the **'gettext'** function does not find a translation for a bookmark into the respective language, the bookmark will automatically be displayed. HEIDENHAIN therefore recommends using the English text for the bookmark. This results in the English text being displayed if no translation is found. You can use the PLC-Text tool for PCs to create the texts for the **'gettext'** function.

When defining a column, use the **'type'** entry to define the type of column. The **'check'** column type generates a column with check boxes.

To show a check box, you must define a rule that returns the result **'true'** for "activated" and **'false'** for "not activated." Alternatively, you can also define a bitmask function using the **'&'** operator. This function will return **'true'** for "bit is set" and **'false'** for "bit is not set." Another possibility is the definition of a **'true-false'** rule. For this type of rule, the value that activates or does not activate a cell is entered directly.

Example:

```
('\\{cell}', 'true-false', '-1', '0')
```

'-1' activates the cell, '0' means "not activated"

To include values of different columns in calculations or to link them with each other, the following logical operations are available:

Operand	Description
<	Less than
<=	Less than or equal to
==	Comparison for "equal to"
!=	Comparison for "not equal to"
>=	Greater than or equal to
>	Greater than
&	Logical AND gating (masking)
true-false	Value assignment for "true" and "false"

Operand	Description
+	Addition
-	Subtraction
=	Accept the value
*	Multiplication
/	Division
*.	Only integer places
.*	Only decimal places
M	The magazine the tool is located in
.P	The pocket in the magazine the tool is located in
TNR	Returns the T number if the tool number is programmed; returns an empty string " " if the tool name is programmed
NOT	Returns the string 'true' if the value is not set; otherwise, 'false' is returned

All of these operations return a character string that is displayed as cell value in the view. More than one rule operation can be defined for the '**value**' configuration setting of a column. All conditions defined are evaluated sequentially, and the last valid value determined is loaded as cell value into the view.

Special columns

The check-box column plays a special role in tabular views. A check box is shown active/inactive if:

- The value calculated from the rule operation is **'true'** or **'false'** (comparison rules except **'true-false'** and **'&'**).
Example: ('\\TABLE\\TOOL\\T\\?\\TL', '<>', '0', 'true', 'false')
-> 'TL' <> '0' means "activated"
 'TL' = '0' means "not activated"
- A value that is permanently defined as **'true'** or **'false'** can be determined (comparison rule **'true-false'**).
Example: ('\\TABLE\\TOOL\\T\\?\\TL', 'true-false', '-1', '0')
-> 'TL' = '-1' means "activated"
 'TL' = '0' means "not activated"
- Masking bits to a numerical value of "bit is set" or "bit is not set" results in (comparison rule **'&'**)
Example: ('\\TABLE\\TOOL\\T\\?\\PLC', '&', '1')
-> Masking bits to the integer value 1 and comparing it to the 'PLC' byte results in:
 Logical '1'. This means "activated."
 Logical '0'. This means "not activated."

Examples of configuring a table column:

The rules are executed sequentially one after the other, thereby enabling you to nest the rules. The last applicable rule determines the displayed value.

First example

The column **rule_tt** is filled with the values from column T of the tool table ('='):

```
rule_tt = {
    'name'      : 'T'
    , 'type'    : 'float'
    , 'sort'    : 'letter_before_number'
    , 'align'   : 'center'
    , 'gettext' : 'false'
    , 'value'   : (
        ('\TABLE\TOOL\T\?\T', '=')
    )
}
```

Second example

The column **rule_pocket** is filled with the values from column P of the pocket table ('='). However, only the pocket number is entered. If the pocket number is equal to 0 ('=', '0', 'DB_Spindle'), then the text from 'DB-SPINDLE' will be displayed.

```
rule_pocket = {
    'name'      : 'DB_POCKET'
    , 'type'    : 'str'
    , 'sort'    : 'number_before_letter,blanks_on_bottom'
    , 'align'   : 'left'
    , 'gettext' : 'true'
    , 'value'   : (
        ('\TABLE\TOOL_P\P\?\P', '.P'), '='
        , ('\TABLE\TOOL_P\P\?\P', '.P'), '==', '0', 'DB_SPINDLE')
    )
}
```



Third example

The column **rule_time** is filled with the values resulting from the subtraction of the values from the CUR.TIME column from the values of the TIME2 column. If the resulting value is less than or equal to 5, then **'DB_TIME_1'** will be displayed. Otherwise, **'DB_TIME_2'** will be displayed. If the calculated value is greater than 10, then **'DB_TIME_3'** will be displayed.

The following rule applies to **'background'**: If the calculated value is less than 5, then the background of the cell will be red. Otherwise, it will be yellow. If the calculated value is greater than 10, then the background of the cell will be green.

```
rule_time = {
  'name'      : 'DB_TIME_AV'
  , 'type'    : 'str'
  , 'gettext' : 'true'
  , 'value'   : (
    (
      ('\TABLE\TOOL\T\?\TIME2', '-', '\TABLE\TOOL\T\?\CUR.TIME')
    , '<=', '5', 'DB_TIME_1', 'DB_TIME_2'),
    (
      ('\TABLE\TOOL\T\?\TIME2', '-', '\TABLE\TOOL\T\?\CUR.TIME')
    , '>', '10', 'DB_TIME_3'),
    )
  , 'background':(
    (
      ('\TABLE\TOOL\T\?\TIME2', '-', '\TABLE\TOOL\T\?\CUR.TIME')
    , '<=', '5', 'red', 'yellow'),
    (
      ('\TABLE\TOOL\T\?\TIME2', '-', '\TABLE\TOOL\T\?\CUR.TIME')
    , '>', '10', 'green')
    )
  , 'background-pix' : 'true'
}
```

Fourth example

The column **rule_tplcbit** is composed of check boxes. A check box is shown as being set if the AND operation performed on the value from the PLC column and the decimal number "1" results in logical "1". The decimal number is converted to the corresponding binary number for comparison.

```
rule_tplcbit0 = {
  'name'      : 'PLC_Bit'
  , 'type'    : 'check'
  , 'gettext' : 'false'
  , 'value'   : (
    ('\TABLE\TOOL\T\?\PLC', '&', '1')
  )
}
```

Configuration file for data transfer to the Tool Data form

You can define the value to be shown in every user interface element in the input form. The display of values (TNC data) is only supported for certain user interface elements:

- **Entry**: Input field
- **Checkbutton**: Check button
- **ComboBoxEntry**: Input field with list box

All table values contained in the TNC table, which provides the basic information, can be used as calculation values. Table values from related tables can also be displayed, provided that this is compatible with the table links (see Possible table links of OEMRules.py).

Definition of a data element

A data element shows the contents of a user interface element and is defined by several configuration settings:

Entry	Description
'widget':	Type of widget. Supported widget types are 'Entry' , 'CheckButton' or 'ComboBoxEntry'
'value':	Value to be used as cell content. Values can only be assigned using the "equal to" operator ('=')
optional 'edit':	As an option, you can define in the OEMEntries.py whether the user can: edit the entry ('edit':('false','=')), or cannot edit the entry ('edit':('true','=')). If no information is defined for 'edit' , editing is possible. Further rules can be used to create a conditional edit function. The rules are executed sequentially one after the other, thereby enabling you to nest the rules.
optional 'range':	As an option, you can enter permissible input values for the cell contents. The following entries are possible: Numerical input range 'range':('1 to 99') or permissible ASCII characters 'range':('ABC123') . This requires the user to enter a value in the field, which is defined by 'range' as permissible value. All other entries are not accepted.
optional 'visible':	Here you can enter a condition for the display of a surface element.
optional 'option':	Only for ComboBoxEntry: Column of any desired table, all of whose values are made available in the list box of the input field. Example of displaying values in a list box: option : ('\\TABLE\\TTYP\\NR\\?\\NAME') -> All of the table values contained in the 'NAME' column of the 'TTYP' table are now available in the list box.



Entry	Description
optional 'optpath':	Only for ComboBoxEntry: Any directory containing files that match the file filter and are offered for selection. Cutting data (CDT) for example displayed as: optpath : ('TNC:', '.cdt') -> all *.cdt files in the TNC:\ directory are offered in the selection box.

Values and operators

Values from the available table cells can be used as calculation values for element rules. Constraints for **'value'**:

- Values can be edited only if the displayed value is from the table that provides the basic information.
- Only the operators '=', '&' and 'true-false' are allowed, because only these operators can ensure that a value is unique. For example, if a sum of two table values is shown in a cell, after an editing process the rule algorithm cannot determine which of the two original table values should be edited.

A valid element rule for **'edit'** must return the result **'true'** or **'false'** (for a more detailed description of rules, see "Logical operations" in OEMRules.py)



Example of a OEMEntries.py file:

The rules are executed sequentially one after the other, thereby enabling you to nest the rules. The last applicable rule determines the displayed value.

Tool table

```
entry_tool_name = {
    'widget' : 'Entry'
    , 'value' : ('\\TABLE\\TOOL\\T\\?\\NAME', '=')
    #, 'edit' : ('\\TABLE\\TOOL\\T\\?\\NAME', '!=', '', 'true')
    ,
}
entry_tool_t = {
    'widget' : 'Entry'
    , 'value' : ('\\TABLE\\TOOL\\T\\?\\T', '=')
    , 'edit' : ('false', '=')
    ,
}
entry_tool_doc = {
    'widget' : 'Entry'
    , 'value' : ('\\TABLE\\TOOL\\T\\?\\DOC', '=')
    ,
}
entry_tool_p = {
    'widget' : 'Entry'
    , 'value' : ('\\TABLE\\TOOL_P\\P\\?\\P', '=')
    , 'edit' : ('false', '=')
    ,
}
checkboxbutton_tool_tl = {
    'widget' : 'CheckBox'
    , 'value' : ('\\TABLE\\TOOL_P\\P\\?\\P', '=')
    ,
}
comboboxentry_tool_typ = {
    'widget' : 'ComboBoxEntry'
    , 'value' : ('\\TABLE\\TOOL\\T\\?\\TYP', '=')
    ,
}
comboboxentry_tool_cdt = {
    'widget' : 'ComboBoxEntry'
    , 'value' : ('\\TABLE\\TOOL\\T\\?\\CDT', '=')
    , 'optpath' : ('\\TABLE\\TNCYS\\NR\\?\\PCDT', '.cdt')
    ,
}
```



In the OEMEntryList.py file, you create the connection between the user interface elements of a data form, such as the tool form, and the data to be displayed (defined in OEMEntries.py). A data form displays a data block or part of a data block from the current view. Additionally, you can define the user interface element that is to have the input focus when the form opens.

Creating a data form

The Glade User Interface Designer (Glade is a free visual programming environment [Rapid Application Development] for creating GTK+ user interfaces graphically and interactively) is used to create a data form for a certain view. Using Glade you can create your own dialog boxes with various operating and display elements. Each of those user interface elements has specific properties, which can be changed as required. The display of TNC data is only supported for certain user interface elements, however:

- **gtk.Entry()**: Input field
- **gtk.CheckButton()**: Check button
- **gtk.ComboBoxEntry()**: Input field with list box

Assigning data to a user interface element

In order to define the data to be displayed in a certain user interface element, you must create a "connection" between the user interface element and the data element to be used. You can define this connection using the name of the user interface element and by gating the element with a saved rule. When editing the data form of a user interface element, you can use the "Properties dialog" in "Glade" to define a name for a certain user interface element (also called 'widget'); see examples below. This name will be assigned to a data element in the OEMEntryList.py file.

Each form definition must be saved under a certain key. The following forms are currently supported:

- **entry_t**: "Tools" data form
- **entry_p**: "Pockets" data form

Special configuration settings:

- **entry_t_focus_element**:
User interface element from '**entry_t**', which has the input focus when the "Tools" data form opens.
- **entry_p_focus_element**:
User interface element from '**entry_p**', which has the input focus when the "Pockets" data form opens.

Example of a OEMEntryList.py:

The structure of the keys of the form definitions must be correspond to that of a Python dictionary, i.e. a list or sequence of keys<-->value pairs. The key to be specified is the name of the respective user interface element (widget). The value to be specified is the name of the rule from OEMEntries.py that defines the data content of the user interface element. The value in the rule designator **entry_tool_name**, for example, is transferred to the user interface element with name **entry_TOOL_NAME**.

For the form concerned, the two entries **entry_x_focus_element** define the user interface element that has the focus when the form opens.

```
entry_t_focus_element = 'entry_TOOL_NAME'
entry_t = {
    'entry_TOOL_NAME'      : entry_tool_name
    , 'entry_TOOL_T'       : entry_tool_t
    , 'entry_TOOL_DOC'     : entry_tool_doc
    , 'entry_TOOLP_P'      : entry_toolp_p
    , 'checkboxbutton_TOOL_TL' : checkboxbutton_tool_tl
    , 'comboboxentry_TOOL_TYP' : comboboxentry_tool_typ
    ...
}
```

```
entry_p_focus_element = 'entry_TOOLP_DOC'
entry_p = {
    'entry_TOOL_NAME'      : entry_tool2_name
    , 'entry_TOOL_T'       : entry_toolp_t
    , 'entry_TOOLP_DOC'    : entry_toolp_doc
    , 'entry_TOOLP_P'      : entry_toolp_p
    , 'checkboxbutton_TOOLP_ST' : checkboxbutton_toolp_st
    ...
}
```



Host computer operation

The "Host Computer Operation" soft key is displayed in the MOD dialog in the programming modes screen. If this operation is selected, then no files (NC programs) are open anymore in the Editing and Test Run operating modes. The PLC can interrogate Marker M4630 to determine whether host computer operation is active. Deactivation of host computer operation via soft key can be disabled via PLC Marker M4631. This enables the PLC program to ensure that host computer operation can only be activated if it is useful and will not cause problems.

The PLC can control host computer operation using Module 9036 (identifier 1000).

A program list (NC programs) can be transferred to the TNC. The list can then be used to generate entire tool usage files and tool usage times. This makes it possible to specify the NC programs and their sequence, as well as the tools, through an external host computer. If external tool data are already available, these data can also be transferred (as a *.CSV file) to the control, if desired, and be used.

- The host computer transfers a table (with only one column **PATH**) together with the NC program sequence. This .TAB file can be generated as a freely definable table on the control and be transferred to the host computer by means of TNCremo. There, the NC program list can be edited anytime.
- In addition, a CSV file containing information for incorporation into the tool usage file can be transferred, if desired. The file must contain the columns **NAME** and/or tool number **T**, as well as the columns **TIME** and **PATH**. The **TIME** column shows the time (in seconds) during which the tool is engaged. If the tool name and the tool number are contained, both will be transferred, but the tool number has priority. Each column, even if it is empty, is concluded with a semicolon in CSV format. Example of a CSV file:

NAME	TIME	PATH	T
CUTTER5MM	235	TNVC:\AUFTR1\NcPrgm1.h	12
DEEPHOLEDRILL 123	33	TNVC:\AUFTR1\NcPrgm1.h	
CUTTER12MM	60	TNVC:\AUFTR1\NcPrgm1.h	
DRILL5MM	60	TNVC:\AUFTR1\NcPrgm2.h	
DRILL2MM	60	TNVC:\AUFTR1\NcPrgm2.h	

- The tool usage file is generated by the NC software if this is initiated via HEIDENHAIN DNC (software option). HEIDENHAIN DNC supports this as of version 1.4, which will be available as of mid-2012. The generation can only be initiated if host computer operation is already enabled. The parameter being transferred is the name of the table containing the NC programs and, optionally, also the name of the CSV file. One specific list per NC program is generated every time. Tables and CSV files that are no longer needed after generation may have to be deleted from the host computer.
Together with the tool usage file, an .H.T.DEP file is generated and stored in the directory of the NC program. This file can be deleted automatically together with the NC program if, in the settings MOD key -> soft key: 'RS232/RS422 Setup', the 'dependent files' entry is set to 'automatic'.



9 PLC Programming

9.1 PLC Functions

The integrated PLC of the iTNC contains its own text editor for creating the list of statements for the PLC program. You enter PLC commands and comments using the keyboard. It's easier, however, to create your PLC programs on a PC with the PLC compiler software **PLCdesignNT**. If you want to use **PLCdesignNT**, please contact HEIDENHAIN.

The iTNC supports you with the COMPILE function, which checks the PLC program for logical errors, and the TRACE, TABLE and WATCH LIST functions, with which you can check the status of the operands.

Due to symbol information used in common by the main program and the soft-key file, a specific sequence must be followed in order to avoid conflicts when compiling the PLC program.

- Soft-key project file
- Main program of the PLC
- Soft-key project file (2nd compilation procedure for checking the symbol information)

The process memory works with a compiled PLC program up to a size of 512 KB. At least every 12 ms—this is the PLC cycle time—the iTNC starts a new PLC scan. The PLC cycle time can be set with MP7602 and ascertained with Module 9196.

Cycle times PLC I/Os

Cycle times during reading and writing of PLC inputs/outputs

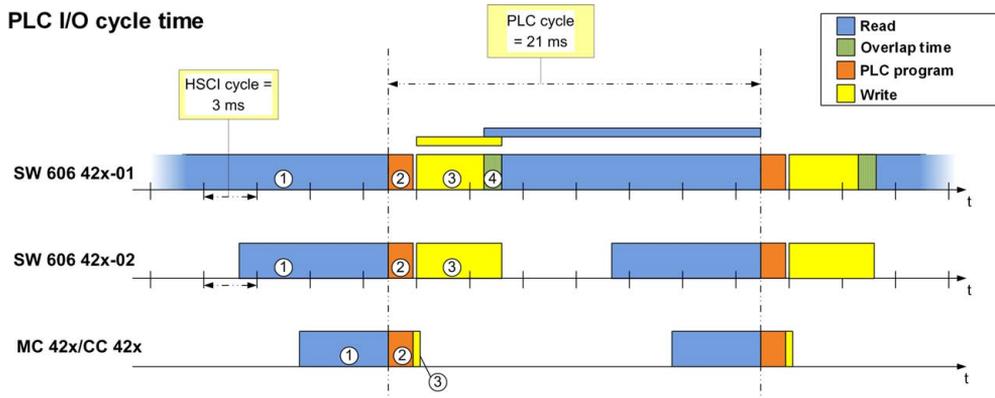
Compared with the hardware components in systems with MC 42x/CC 42x, there are different cycle times during reading and writing of PLC inputs and outputs with the new control system with HSCI. The differences result from the changed communication of the individual components via HSCI and the synchronization required during data transfer.

The procedures for these software versions are described in the following:

- NC software iTNC 530: 606 42x-01
- Or as of NC software iTNC 530: 606 42x-02

Overview of cycle times

The flow chart shows a comparison of the HSCI system with the respective software versions and a system with MC 42x/CC 42x. The decisive worst-case cycle times for reading and writing as well as the typical run times for the PLC cycle and the PLC program were used in the diagram:



Times in the flow chart:

- 1) Reading of PLC inputs
Time needed to read in the physical input states and transfer them to the PLC memory (e.g. filter run times, hardware transmission paths). The specified PLC cycle time passing until the input information is actually processed by the PLC program must also be considered here.
 - Worst case for HSCI SW 606 42x-01: 15.7 ms
 - Best case for HSCI SW 606 42x-01: 10.5 ms
 - Worst case for HSCI as of SW 606 42x-02: 9.7 ms
 - Best case for HSCI as of SW 606 42x-02: 4.5 ms
 - Worst case for MC 42x/CC42x: 5.0 ms
- 2) Run time of the PLC program
Along with the cycle times for reading and writing of PLC inputs and outputs, the PLC run time must be considered. The typical run time of the PLC program is up to 1.5 ms.

■ 3) Writing of PLC outputs

Time until the output states generated by the PLC program are physically present at the terminals.

- Worst-case scenario for HSCI SW 606 42x-01: 4.8 ms
- Worst-case scenario for HSCI SW 606 42x-02: 4.8 ms
- Best case for HSCI SW 606 42x-xx: 3.0 ms
- Worst case scenario for MC 42x/CC42x: 0.25 ms

■ 4) Overlap time

In a worst-case scenario, an overlap can occur in the HSCI system SW 606 42x-01 during reading and writing of the PLC inputs/outputs. In the scenario described here, writing of the output states has not been completed when reading of the next PLC cycle starts. In this case, the output states of the preceding cycle are not yet available to the PLC program of the subsequent cycle.

The table shows a comparison of the worst-case cycle times for reading and writing of PLC inputs and outputs:

Cycle times	MC 42x/ CC 42x	HSCI SW 606 42x-01	HSCI SW 606 42x-02
Reading of PLC inputs	5.0 ms	15.7 ms	9.7 ms
Writing of PLC outputs	0.25 ms	4.8 ms	4.8 ms

With a PLC program run time of 1.5 ms, the following, typical total times result for the systems described here:

- Worst-case scenario for HSCI SW 606 42x-01: 22.0 ms
- Worst-case scenario for HSCI SW 606 42x-02: 16.0 ms
- Worst-case scenario for MC 42x/CC42x: 8.25 ms
(including any interruptions of the PLC program by the interpolator)

With a typical PLC cycle time of 21 ms and using the MC 42x/CC 42x, it is possible to write PLC outputs within one PLC cycle and to evaluate this information again in the next PLC cycle via a PLC input. In the HSCI system with SW 606 42x-01, however, an overlap time can occur during reading and writing of PLC inputs/outputs in the worst-case scenario described here. This means that an additional PLC cycle is required until updated output states are available again to the PLC program.

HEIDENHAIN has optimized the reading of PLC inputs in HSCI systems by 6 ms as of SW 606 42x-02. Even in an HSCI system with a typical PLC cycle time of 21 ms, PLC outputs can thus be written in one PLC cycle and the information can then be evaluated via a PLC input in the next PLC cycle.

Please note the following information regarding control systems with HSCI:



Note

- Please note the worst-case time behavior described above for your PLC program.
- Test the behavior of the PLC program especially with regard to time-critical applications.
- If required, conduct an update of the PLC program on the affected machines in the field.



Detection of pulses at PLC-inputs

The following data apply for single-channel standard PLC inputs and dual-channel FS inputs of the safe PLC (SPLC).

Minimum required pulse widths in the HSCI system for signal detection by the control at the standard inputs:

Input	Required pulse width
(S)PLC inputs on: X9 of a PLB, PLD in slots 1 to 4, machine operating panel or UxC 1xx	> 8 ms
PLC inputs on PLD in slots 5 to 8	> 11 ms

The (S)PLC cycle time must also be observed for actual processing in the (S)PLC program. The pulse width must be longer than the above mentioned time + the (S)PLC cycle time in order to ensure that the pulse is processed in the (S)PLC program.

Minimum required pulse widths in the HSCI system for signal detection by the control at the special inputs:

Input	Required pulse width
Fast PLC inputs on PLD in slots 1 to 4	> 8 ms
Emergency stop inputs –ES.A, –ES.B	> 5.5 ms

Pulse widths at (S)PLC inputs that are not detected due to internal hardware filter run times:

Input	Maximum pulse width	Effect
All (S)PLC inputs Edge low → high	< 0.4 ms	High pulse is not detected due to internal filter run times
All (S)PLC inputs Edge high → low	< 2.0 ms	Low pulse is not detected due to internal filter run times

Some devices that operate with self-generated test pulses at their input (e.g. test pulses for wire breakage or cross-circuit detection) should be connected to machines at the (S)PLC inputs. For example, a low test pulse (high → low flank) of such devices must be smaller than 2.0 ms so that they are not detected due to internal filter run times. If pulses are greater than 2.0 ms it can cause the control to detect them, forward them to the (S)PLC run-time environment and trigger the corresponding reactions.

In most cases a "double" loop in the (S)PLC program can be used to prevent the triggering of such reactions, for example with low test pulses with a pulse width greater than 2.0 ms. This filters the test pulses with the goal of further processing only static signals that are available at the (S)PLC input for more than two (S)PLC cycles.

Example of such a loop for filtering during use of a guard-door switch with test pulses:

```

;Filter I_GuardDoorSwitch_closed by one PLC cycle
;
L      I_Workspace_to_sensor      ;Sensor signal is pulsed
A      ML_Workspace_was_closed
S      I_Workspace_closed        ;modified signal, used in the PLC
program

LN     I_Workspace_to_sensor
AN     ML_Workspace_was_closed
R      I_Workspace_closed

L      I_Workspace_to_sensor
=      ML_Workspace_was_closed

```



Note

Remember the following when using such a loop:

- The information on the status of such filtered signals is processed by the loop with a delay of one (S)PLC cycle. This has to be considered in the control's reaction times.
- It must be possible to rule out a synchronization between the (S)PLC program and the test pulse. Otherwise a test pulse beyond several (S)PLC cycles might be detected and therefore not be filtered out. The repetition rate or the cycle of the test pulse must be greater than twice the (S)PLC cycle time for this purpose. (e.g. a repetition rate of $> 2 \times 21$ ms)
- Test the (S)PLC program on the machine.

Module 9196 Find the PLC cycle time

The PLC cycle time is determined in μs .

Call:

CM 9196

PL D <>PLC cycle time in μs >

MP7602 PLC cycle time

Input: 0 to 60 ms

0 to 12 = 12 ms



Note

An exact PLC cycle time is only available after the emergency stop test.



9.1.1 Selecting the PLC mode

To select the PLC mode:

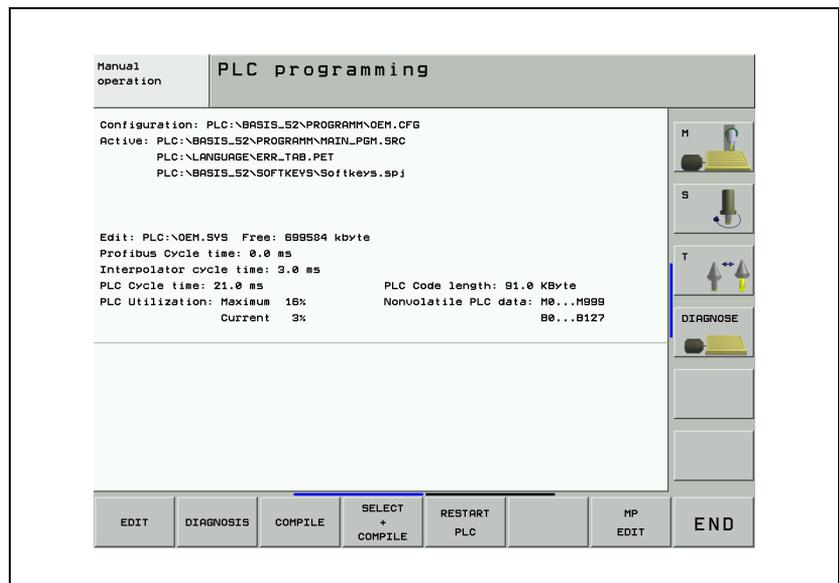
- ▶ Select the **Programming and Editing** operating mode.
- ▶ Press the MOD key.
- ▶ Enter the code number 807667 and confirm your entry with the ENT key, or if you already entered the code number, press the PLC EDIT soft key.

Exit PLC mode:

- ▶ Press the END soft key or the END key.

9.1.2 PLC main menu

After you have entered the code number (or pressed the PLC EDIT soft key) the iTNC displays the PLC main menu:



Configuration:

Active PLC configuration file. The machine-specific configurations for the PLC are set in this file.

Active:

Files of the integrated PLC in the process memory. These include the PLC program, PLC error table, soft-key project file and possibly the Profibus configuration file or configuration file for source-code programming.

At start-up, the iTNC automatically compiles the files defined in OEM.SYS. The files only become active after they have been compiled.

Edit:

Name of the file located in RAM memory.

Free:

The available memory in the PLC partition (PLC:\) is shown in kilobytes. Additionally, the PLC partition is checked to see if there are at least 10 MB of memory available. If the available memory is less than 10 MB, the **PLC partition: Not enough memory** error message is output.

Profibus cycle time:

Currently needed cycle time for requests and write-processes of the Profibus system.

Interpolator cycle time:

Current interpolator cycle time.

PLC cycle time:

Current PLC cycle time, set via MP7602.

PLC utilization Maximum:

Maximum run time of the PLC program

The PLC processing time (time for a PLC cycle) is given as a percentage of the maximum time: 100% is the equivalent of a run time of 1 ms at a cycle time of 21 ms. Use the following formula to calculate the run time t_{run} [ms] as a function of the PLC cycle time t_{PLC} [ms] and the processing time t_{calc} [%]:

$$t_{\text{run}} = \frac{t_{\text{PLC}} \cdot t_{\text{calc}}}{21}$$

If the maximum run time of the sequential program is exceeded, the iTNC displays the blinking error message **PLC: time out**.

PLC utilization Currently:

The time taken for the latest PLC scan in %.

PLC code length:

Length of the compiled sequential program in KB.



PLC functions of the main menu

From the PLC main menu you can use soft keys to access the following PLC functions:

Soft key	Function
	Edit the file located in RAM memory
	Call diagnostic functions, see page 1298
	Compile files registered in OEM. SYS, see page 1682
	Select and compile files to be compiled, see page 1682
	Stop and restart the PLC program (M4173 is supported)
	Additional settings available to the machine tool builder
	Display a list of machine parameters
	Display the statuses of selected operands in tabular format, see page 1671
	Set inputs and outputs. The PLC program is ignored, see page 1674
	Display the logical states of the PLC operands, see page 1676
	Show the logic diagram, see page 1680
	Display the TRACE function, see page 1678
	Display the process monitor, see page 1837
	Activate the integrated oscilloscope, see page 992
	End PLC programming.

9.1.3 File management

File management in the PLC mode is largely the same as in the **Programming and Editing** mode of operation (see the iTNC 530 User's Manual). If you press the PGM MGT key while in the PLC mode, the iTNC displays the PLC partition as well, at the upper left next to the TNC partition.



Note

Space characters are not permissible in path and file names.

Differences from file management of NC part programs

File types displayed by the iTNC when you press the SELECT TYPE soft key:

Soft key	Function
	Show all files
	Show only PLC programs (*.PLC)
	Show only ASCII files (*.A)
	Show only help files (*.HLP)
	Show only system files (*.SYS)
	Show only compensation value tables (*.COM)
	Show only tables with compensation value assignments (*.CMA)
	Show only PLC error tables (*.PET)
	Show only PLC source files (*.SRC)
	Show only soft-key project files (*.SPJ)
	Return to previous menu

9.1.4 The WATCH LIST function

With the WATCH LIST function you can create a table with dynamic display of the states of the selected operands.

Meaning of the columns in the WATCH LIST:

- **MODULE:** <Global> for global symbolic operands or path with the name of the *.SRC file in which the operand is defined
- **SYMBOL:** Symbolic address of the operand
- **ADDR:** Absolute address of the operand
- **VALUE:** Content of the operand
- **COMMENT:** Comment for the operand

Soft keys within the WATCH LIST function:

Soft key	Function
	Jump to the beginning of the WATCH LIST
	Jump to the end of the WATCH LIST
	Scroll back one page in the WATCH LIST
	Scroll forward one page in the WATCH LIST
	Jump to the beginning of the current line
	Jump to the end of the current line
	Show contents of operands as decimals or hexadecimal
	Insert a new line above the current line
	Delete current line

Soft key	Function
	Display a selection list with all symbolic operands used in the active PLC program
	Jump to the beginning of the selection list
	Jump to the end of the selection list
	Scroll back one page in the selection list
	Scroll forward one page in the selection list
	Search the selection list for a specific text
	Load selected operand into the WATCH LIST
	Close the pop-up window
	If the cursor is in the SYMBOL or ADDR column, then this soft key becomes visible, and the column can be sorted alphabetically ascending or descending.
	Load selected operands into the logic diagram, see page 1680
	Search text in the WATCH LIST (first the SYMBOL column and then the ADDR column is searched)
	Add selected input or output into the I/O Force List, see "The I/O-FORCE LIST" on page 1674
	Display the logical states of the PLC operands, see page 1676
	Select columns to display in the WATCH LIST and their order
	Assume column settings of the WATCH LIST
	Close selection window without applying new settings
	Return to previous menu

Display of symbolic operands in the WATCH LIST

- ▶ Press the WATCH LIST soft key to call the menu of the WATCH LIST function.
- ▶ Press the SYMBOL LIST soft key to open a list box with all local and global operands used in the PLC program.
- ▶ Select the desired operand with the arrow keys and load it with the SELECT soft key or with the ENT key.
- ▶ Press the END soft key to close the list box.



Note

Operands can only be selected with the SYMBOL LIST soft key if you are working with the *.SRC source files of the PLC program on the control. Otherwise the error message **Selection list is empty** appears.

Display of operands in the WATCH LIST

- ▶ Press the WATCH LIST soft key to call the menu of the WATCH LIST function.
- ▶ Press the ZEILE EINFÜGEN soft key.
- ▶ In the **ADDR** column, enter the absolute address of the operand, i.e. W1022.
- ▶ Press the ENT key.

Add operands to I/O-FORCE LIST

- ▶ Press the WATCH LIST soft key to call the menu of the WATCH LIST function.
- ▶ Press soft key ADD TO I/O-FORCE LIST.

The operand is written to the I/O-FORCE LIST, and a message confirming the transfer of the operand to the I/O-FORCE LIST appears.

Internal process of the WATCH LIST function

DEBUGPATH = PLC:\DEBUG is automatically entered in OEM.SYS. This is the working directory for the WATCH LIST function.

If you are working with the source files on the control, a *.WLC file is generated from the *.MAP file when compiling the PLC program. This *.WLC file has the same name as the PLC main program, and contains all local and global symbolic operands. The *.WLC file is saved in the working directory mentioned above for the WATCH LIST function.

The selection window is used to select the symbolic operands from the *.WLC file for the WATCH LIST. The first time operands are selected and a WATCH LIST is created, the file TEMP.WLT is created and saved in the working directory. The entry in OEM.SYS is expanded to **DEBUGPATH = PLC:\DEBUG\TEMP.WLT**. This ensures that when the WATCH LIST function is next called, the most recent WATCH LIST will be active.

If there is more than one *.WLT file in the working directory, the desired file can be chosen with PGM MGT. Selecting a new *.WLT file also changes the **DEBUGPATH =** entry in OEM.SYS.

9.1.5 The I/O-FORCE LIST

With the I/O-Force List you create a table in which the status of selected **inputs and outputs** is written based on the process image.



Danger

The I/O-FORCE LIST can overrule safety-relevant monitoring operations as well as any inputs and outputs in the PLC program! Activation of the I/O-FORCE LIST can therefore only be done with extreme care. Also make sure that hanging axes are supported.

Meaning of the columns in I/O-FORCE LIST:

- **SYMBOL:** Symbolic address of the operand
- **ADDR:** Absolute address of the operand
- **VALUE:** Content of the operand
- **COMMENT:** Comment for the operand

Add operands to I/O -FORCE LIST

- ▶ Press the WATCH LIST or TABLE soft key in the **PLC Programming** operating mode.
- ▶ **With WATCH LIST:** In the WATCH LIST, select the line with the desired operand (input or output). You may also select the operand via the SYMBOL LIST first.
- ▶ **With TABLE:** In the TABLE, select the desired operand via soft key and the keyboard arrow keys.
- ▶ Press the ADD TO I/O-FORCE LIST soft key.

The operand is written to the I/O-FORCE LIST, and a message confirming the transfer of the operand to the I/O-FORCE LIST appears.



Note

When using the I/O-FORCE LIST, remember that:

- If an operand is entered and the I/O-FORCE LIST is activated via soft key, then this operand is marked in color in the TABLE, and the **SET** and **RESET** soft keys have no meaning.
- The green error message **I/O-Force** is output in the **PLC Programming** mode of operation if the I/O-FORCE LIST is active.

Soft keys in the I/O-Force List:

Soft key	Function
	Jump to the beginning of the I/O-Force List
	Jump to the end of the I/O-Force List
	Scroll one page backward in the I/O-Force List
	Scroll one page forward in the I/O-Force List
	Insert a new line above the current line
	Delete current line
	Switch on or off the I/O-Force List. When this soft key is ON, the values entered in VALUE are written via the corresponding inputs or outputs. The states of each input are not evaluated. In addition, the outputs set by the PLC program are not sent to the hardware.
	If the cursor is in the SYMBOL or ADDR column, then this soft key becomes visible, and the column can be sorted alphabetically ascending or descending.
	Return to previous menu

Multiple I/O-Force Lists possible

It is possible to use different lists for the I/O Force function in the PLC operating mode. If an I/O Force List is open, press the PGM MGT key to switch to an existing list or create a new list. Create a new I/O Force List by entering a new file name with the extension .FLT. The selected list is identified by the automatically generated **FORCELISTPATH = PLC:*.FLT** entry in OEM.SYS.

9.1.6 The TABLE function

From the main menu, choose the TABLE soft key to select the table of the PLC memories in order to show their states dynamically on the screen. To select a certain operand, use the cursor keys or the GOTO key. Press the END key to return to the main menu.

Within the TABLE function, you can navigate with the appropriate soft keys, or with the m, i, o, c, t, b, w, d and s keyboard keys for each operand type. The b, c, and d keys are an exception in the B, W, D and HEX views, since they are needed for the entry of hexadecimal values.

In the TABLE function the first two symbols (in alphabetical order) are displayed in the table if in the symbol definition more than one symbol is assigned to the PLC operand through the /c switch.

Soft key	Function
	Set the selected operand
	Reset the selected operand
	Show a list of the markers
	Show a list of the inputs
	Show a list of the outputs
	Show a list of the counters
	Show a list of the timers
	Show a list of the bytes
	Show a list of the words
	Show a list of the double words



Soft key	Function
	List of strings (only the first 70 characters). Overwriting is not possible.
	Show contents of operands as decimals or hexadecimals
	Save states of selectable operand areas in an ASCII file. Areas of more than one operand can be saved, e.g. M0 to M100, W100 to W118.
	Display saved ASCII file with states of operands
	Load selected operands into the WATCH LIST, see page 1671
	Return to previous menu

9.1.7 The TRACE function

The TRACE function enables you to:

- Control the logical states of markers, inputs, outputs, timers, and counters
- Check the content of bytes, words and double words

From the PLC main menu, press the TRACE IN-CODE soft key to select the TRACE function. The iTNC then shows:

- The statement list (STL) of the selected PLC program
- For every program line, the content of the operand and the accumulator in HEX or decimal code (selectable by soft key)

The iTNC identifies every cyclically executed command with a **C**. With the arrow keys or the GOTO function you can select the program section that the iTNC should display on the screen.

The PLC program to be displayed is chosen with PGM MGT, and must be the currently active main program or a file integrated with USES.

Operand	Accu	Index	CZ#	Command
			22	
			23	;-----
			24	LBL Spindel_Getriebe
			25	;-----
			26	
0	0		C 27	L NP_M4172_1_PLC_Durchlauf_na
0	0		C 28	O NP_M4173_1_PLC_Durchlauf_na
0	0		C 29	S MG_Getriebe_initialisieren
			30	
1	0		C 31	LN MG_Spindel_aktiv
			32	EMT
			33	
0	0		C 34	L MG_Getriebe_initialisieren
			35	IFT
-	-		36	L MG_Steuer_Spannung_Verzoeg
-	-		37	R MG_Getriebe_initialisiere
-	-		38	S PW_M4194_Strobe_G_Stufe_D



Soft keys within the TRACE function:

Soft key	Function
	Show the logic diagram, see page 1680
	Show operand or accumulator contents in hexadecimal or decimal notation
	Stop dynamic display of the operand content, the accumulator content, and the logic diagram with STOP; continuously update again with START.
	
	Start/End the trace
	
	Load selected operands into the WATCH LIST, see page 1671
	Jump to the beginning of the STL
	Jump to the end of the STL
	Scroll back one page in the STL
	Scroll forward one page in the STL
	Find text in the STL
	Return to previous menu

9.1.8 The logic diagram

Soft keys within the LOGIC DIAGRAM function:

Soft key	Submenu	Function
SELECT M/I/O/T/C		Select Markers/Inputs/Outputs/Timers/Counters for a logic diagram, trigger logic and recording time.
	LOGIC DIAGRAM	Return to the recording interface
	SELECT TRIGGER LOGIC	Selecting the trigger logic: Here you specify if recording begins when a change in the signal state (0 or 1) first occurs (OR), or only after all trigger conditions have occurred once (AND).
	SELECT SCAN TIME	Selecting the recording time. Here you specify how long the signal states are recorded from the defined trigger time point. 4 different times are available, depending on the PLC cycle time (a total of 2048 PLC cycles are recorded).
TRACE IN-CODE		Display the TRACE function (, see page 1678)
SAVE TRACE BUFFER		Save current logic diagram in an ASCII file (*.A)
RESTORE TRACE BUFFER		Show the saved logic diagram
START TRACE		Stop dynamic display of the operand content, the accumulator content, and the logic diagram with STOP; continuously update again with START.
FREEZE TRACE		
START LOGIC TRACE		Start/End the trace
STOP LOGIC TRACE		
END		Return to previous menu

With the LOGIC DIAGRAM function you can graphically display the logical states of up to 16 operands (M/I/O/T/C) at once. 2048 PLC cycles are recorded.

The operands to be shown must be saved in a table that you create with the SELECT M/I/O/T/C soft key. The iTNC asks per dialog for the individual positions in the table. To delete the lines, simply press DEL.

For each operand you can enter one trigger condition. The iTNC records from the trigger time for as long as it was instructed to. A total of 2048 PLC cycles is always recorded, and depending on the setting, 128, 256, 512 or 1024 PLC cycles from the trigger time. The appropriate cycles before the trigger time are also shown.

The following trigger conditions are possible:

- 1: Record if operand is logically 1 (trigger on positive edge).
- 0: Record if operand is logically 0 (trigger on negative edge).
- No trigger condition: Only record in combination with other triggers.

If you do not need a trigger condition, you can enter this in the **Trigger** column with the **CE** key. The iTNC records the states of this operand in parallel with the triggered operands.

To start recording:

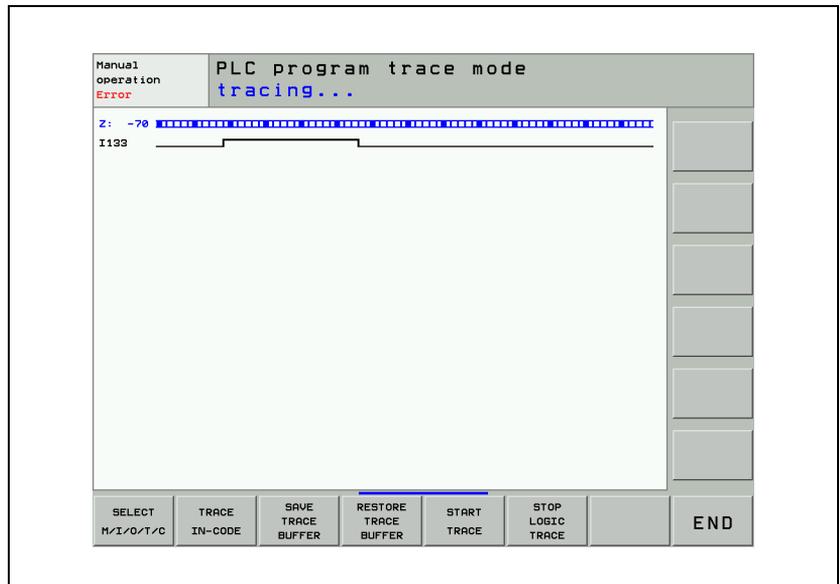
- ▶ Press the START LOGIC TRACE soft key.

To stop recording:

- ▶ Press the STOP LOGIC TRACE soft key, or the iTNC terminates recording automatically as soon as the trigger event occurs.

The "PCTR" indicator blinks in the status window as long as the iTNC is recording logical states. As soon as recording ends, you can use the arrow keys to select the desired area in the TRACE buffer.

Example of logic diagram:



9.1.9 The COMPILE function

Compiling a completed PLC program transfers it to the process memory where it can then become active. The name of the compiled program then appears in the main menu next to **PGM IN EXEC.MEM.**

The compiled PLC program is saved in the control, i.e. after confirmation of **power interruption**, the PLC program need not be compiled. It is compiled only if one of the source files has changed. A binary PLC program can be created with PLCdesignNT for test purposes and transferred to the control. This does not change the entry in OEM.SYS.

Soft keys within the COMPILE function:

Soft key	Function
<div style="border: 1px solid black; padding: 2px; text-align: center;"> COMPILE ALL </div>	Compile current PLC program, current PLC error table, and current soft-key project file (PLCMAIN= , PLCERRTAB= , and SOFTKEYPROJECT= entries in OEM.SYS)
<div style="border: 1px solid black; padding: 2px; text-align: center;"> COMPILE PLC-MAIN PROGRAM </div>	Compile only the current PLC program (PLCMAIN= entry in OEM.SYS)
<div style="border: 1px solid black; padding: 2px; text-align: center;"> COMPILE PLC- ERRORTAB. </div>	Compile only the current PLC error table (PLCERRTAB= entry in OEM.SYS) (, see page 1176)
<div style="border: 1px solid black; padding: 2px; text-align: center;"> COMPILE SOFTKEY- PROJECT </div>	Compile only the current soft-key project file (entry SOFTKEYPROJECT= in OEM.SYS), see page 1372
<div style="border: 1px solid black; padding: 2px; text-align: center;"> COMPILE IOC </div>	Compile only the current IOC file (IOCCFG= entry in OEM.SYS)
<div style="border: 1px solid black; padding: 2px; text-align: center;"> END </div>	Return to PLC main menu

Soft keys within the SELECT + COMPILE function:

Soft key	Function
<div style="border: 1px solid black; padding: 2px; text-align: center;"> SELECT COMPILER CONFIG. </div>	Select and compile the configuration file for the compilation of the source code, see page 1684
<div style="border: 1px solid black; padding: 2px; text-align: center;"> SELECT PLC-MAIN PROGRAM </div>	Select and compile a PLC program
<div style="border: 1px solid black; padding: 2px; text-align: center;"> SELECT PLC- ERRORTAB. </div>	Select and compile a PLC error table, see page 1176
<div style="border: 1px solid black; padding: 2px; text-align: center;"> SELECT SOFTKEY- PROJECT </div>	Select and compile a soft-key project file, see page 1372
<div style="border: 1px solid black; padding: 2px; text-align: center;"> SELECT MAGAZINE RULES </div>	Select and compile a magazine rule file, see page 1580
<div style="border: 1px solid black; padding: 2px; text-align: center;"> SELECT IOC </div>	Select and compile an IOC file
<div style="border: 1px solid black; padding: 2px; text-align: center;"> END </div>	Return to PLC main menu

For example, to compile any PLC program:

- ▶ Press the SELECT + COMPILE soft key: the iTNC displays the soft keys for the SELECT + COMPILE function.
- ▶ Press the SELECT PLC-MAIN PROGRAM soft key: the iTNC opens the program manager.
- ▶ Use the arrow keys to select the PLC program to be compiled.
- ▶ Press the ENT key.

The name and path of the compiled PLC program are entered with **PLCMAIN=** in OEM.SYS.

Press the COMPILE PLC-MAIN PROGRAM soft key if you only want to compile the PLC program from this entry.

PLC Marker M4188 is set at the beginning of each compilation process in the PLC project. Marker M4188 is reset as soon as the initialization of all components involved in the compilation process has been completed and the PLC has been started.

		Set	Reset
M4188	Compilation process of the PLC project active	NC	NC

9.1.10 ADVANCED SETUP

Press the ADVANCED SETUP soft key to get additional settings:

Soft key	Function
	Settings for the XFCE Window Manager, see page 1332
	Managing software options via the SIK, see page 140
	Creation and settings of the encoded PLC partition (PLCE), see page 1690
	Optimization, adjustment and changing of the kinematics with KinematicsDesign on the control, see page 1488
	Return to PLC main menu

9.2 Conditional Compilation

Depending on the machine parameters, a PLC program can be conditionally compiled on the iTNC. This allows you to select and deselect machine options by entering the options in machine parameters. Therefore, only one PLC program is necessary for all variants of machine options.

You use the PLC configuration file *.cfg also to influence the memory allocation of the control and change the configuration data of the PLC. Enter certain keywords to perform changes. After compilation by the PLC, the NC detects the changed settings, and the control must be restarted.

The values you enter may only be greater than the default values.

Available keywords:

Keyword	Description	Example
DEFINE	Configuration definitions for controlling the conditional compilation, soft-key menu generation and cycle-project configuration	
REMBYTEMIN	Start address of the bytes, words or double words whose data remains stored after a power interruption (remanence). Default value: 0	REMBYTEMIN = 0
REMBYTEMAX	End address of the bytes, words or double words whose data remains stored after a power interruption (remanence). The range defined by REMBYTEMIN and REMBYTEMAX may not be larger than 1024 bytes. Default value: -1 = Deactivated	REMBYTEMAX = 200



Keyword	Description	Example	
REMARKERMIN	Start address of the markers whose data remains stored after a power interruption (remanence). Default value: 0	REMARKERMIN = 0	
REMARKERMAX	End address of the bytes, words or double words whose data remains stored after a power interruption (remanence). The range defined by REMMARKERMIN and REMMARKERMAX may not consist of more than 2048 markers. Default value: -1 = Deactivated	REMARKERMAX = 150	
MARKERS	A total of 100 000 bytes is available for all keywords, timers, counters and strings combined	Number of markers available. Default value: 10000	MARKERS = 15000
BYTES		Size in bytes for the byte/word/double word memory. Default value: 10000	BYTES = 20000
INPUTS		Number of input markers available. Default value: 384	INPUTS = 450
OUTPUTS		Number of output markers available. Default value: 192	OUTPUTS = 250
INPUTBYTES		Size in bytes for the byte/word/double word memory range used by the Profibus inputs. Default value: 1000	
OUTPUTBYTES		Size in bytes for the byte/word/double word memory range used by the Profibus outputs. Default value: 1000	

Along with the keywords there are also system parameters. System parameters are identified by a preceding and following \$ character. Depending on which options have been set, they are defined via the compiler and can be used for conditional compilation. For more information about this, please refer to the PLCdesignNT help.

Available system parameters:

System parameter	Meaning
For interrogation with #ifdef:	
\$MULTIFILE\$	Multifile option is set
\$SYM_LABELS\$	Symbolic labels option is set
\$<nc model>\$	Current control model, e.g. #ifdef \$TNC407\$
\$<nc variante>\$	Current NC ID number, e.g. #ifdef \$340490_05\$
\$<nc basisvar.>\$	Current NC basic ID number, e.g. #ifdef \$340490\$
For interrogation with #if:	
\$<VARIANT>\$	Current NC ID number as numerical value, e.g. #if \$VARIANT\$ >= 34049005
\$<VARIANTBASE>\$	Current NC basic ID number as numerical value, e.g. #if \$VARIANTBASE\$ = 340490
\$<VARIANTINDEX>\$	Current NC ID number index as numerical value, e.g. #if \$VARIANT\$ >= 05
\$<VERSIONBASE>\$	Current compiler basic version as numerical value
\$<VERSIONINDEX>\$	Current compiler index as numerical value

Strings in #if conditions for conditional compilation in the PLC program, in soft key and cycle resources can now also contain blank spaces. The PLC tools can also process #if conditions with strings containing blank spaces and integer values with algebraic sign (e.g **#if TEST "1 2 3"**).

A PLC program can identify whether the current system is the conventional hardware (MC 42x, CC 42x) or HSCI hardware, which is addressed through the IOC configuration. For this purpose, the environmental variable **%HsciHardware%** is provided for the configuration file of the PLC program (PLCCOMPCFG entry in OEM.SYS file). The variable contains the value 1 if software 606 42x, and thus the HSCI hardware, is detected. Marker M4641 can be used for interrogation at run time.



The variable **#targetpath** is available so that a PLC program, in which individual modules are on the "PLCE:" partition, can be compiled correctly on the programming station and the control.

Example:

```
#targetpath PLCE:\PLC\  
uses plce\geheim.src  
#targetpath PLC:\basis\program
```

On the PC: **plce\geheim.src** is searched for the module.

On the control: **PLCE:\PLC\plce\geheim.src** is searched for the module.

The environmental variable **%GraphicsResolution%** in the PLC configuration file is also available for conditional compilation of the PLC program. The variable is set to the value 1024x768 or 1280x1024, depending on the screen resolution. The content of the environmental variable can be evaluated and used for conditional compilation.

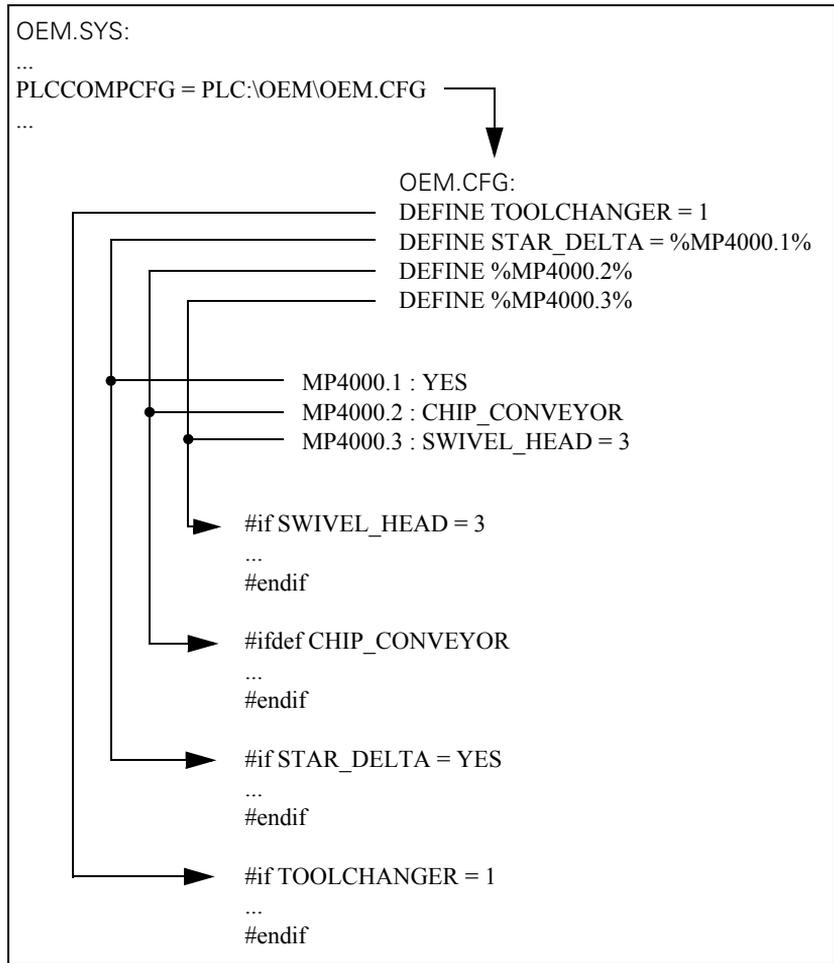
Example:

```
define resolution=%GraphicsResolution%  
...  
# if resolution=1024x768  
...  
#endif  
...  
# if resolution=1280x1024  
...  
#endif
```

Example of a config file:

- ▶ Enter the commands for the conditional compilation in the PLC program.
- ▶ Create the config file.
- ▶ In OEM.SYS, enter **PLCCOMPCFG** = followed by the path for the config file.
- ▶ Enter the machine options in the machine parameters MP4000.x.
- ▶ Reset the iTNC or recompile the PLC program.

Example:



MP4000.0-63 Options for the conditional compilation of the PLC program



9.3 Hard-Disk Organization

The hard disk of the iTNC is divided into three partitions:

TNC partition

User-specific data such as NC programs, tool tables, datum tables, and pallet tables.

PLC partition

Your OEM-specific data such as system files, PLC programs, machine parameters, help files, PLC dialogs, PLC error tables, compensation value tables and OEM cycles. The PLC partition is visible only after you have entered the code number 807667.

As a machine tool builder, you are concerned primarily with the PLC partition.

SYS partition

System-specific files such as system files, NC dialogs, HEIDENHAIN cycles, etc. The SYS partition is not visible and cannot be selected.



Attention

Alterations to the system partition can impair proper function of the iTNC!

Size of the partitions

HDR

The following sizes are valid for the partitions on HDR hard disks for the iTNC 530 that were delivered with the NC software 606 42x-01:

Partition	Contents	Size
SYS	System files	2 GB (for multiple software versions in packed format)
PLC	OEM files	1 GB
TNC	User files	> 138 GB

SSDR

The following sizes are valid for the partitions on the SSDR solid state disk for the iTNC 530 that were delivered with the NC software 606 42x-01:

Partition	Contents	Size
SYS	System files	2 GB (for multiple software versions in packed format)
PLC	OEM files	1 GB
TNC	User files	> 21 GB



Note

A single NC program can be up to 2 GB large.

Checking the file system

(Only for single-processor version)

The TEST FILE SYSTEM soft key was introduced in the second soft-key row in the Programming and Editing mode under MOD. It can be used to start the program **dosfsck** in order to check and automatically repair the file system of the PLC and TNC partition of the control's hard disk.

Please ensure that no NC program is being run, and save open files if necessary. When the **dosfsck** program is started, the NC software will be terminated and an EMERGENCY STOP will be triggered as soon as you confirm the associated prompt with the YES soft key. The outputs of the corresponding program **dosfsck** are shown in a window and must be acknowledged with the <Enter> key or OK button. The control then boots again. The outputs are also written to the log. The SYS: partition cannot be checked, but it is checked each time the control is switched on.

9.3.1 Encrypted PLC partition (PLCE:)

The machine manufacturer has the possibility to create an additional PLC partition, which is automatically encrypted. The **PLCE:** partition enables the machine manufacturer to protect his files and data from unauthorized access.

The encrypted PLC partition **PLCE:** can only be accessed with a password that the OEM chooses. The password is stored in encrypted form either in the OEM.SYS file or in the SIK.



Note

HEIDENHAIN recommends:

- Storing the password in the SIK, because it provides a higher degree of security than the OEM.SYS.
- Using the PLCE partition generally only for specific files that are to be protected from unauthorized access.
- Against using the PLCE partition for system files (such as MP files, NCMACRO.SYS). Please keep in mind that HEIDENHAIN cannot access the data in the encrypted partition, even if servicing becomes necessary.

When creating the PLCE: partition, you must specify the desired memory size (max. 100 MB) of the encrypted partition. Because the PLCE partition saved in the **PLC:\PLCE.BIN** file is a part of the PLC partition, the memory available on the PLC partition is reduced accordingly.



Danger

- When you lose the password for the **PLCE:** partition, the data is permanently lost.
- Deleting the **PLC:\PLCE.BIN** binary file leads to loss of the data on **PLCE:**
- HEIDENHAIN does not have any possibility to access the data on the encrypted **PLCE:** partition. This also applies when servicing becomes necessary.
- The data of the encrypted **PLCE:** partition cannot be written to the service file.
- HEIDENHAIN provides the encrypted PLC partition (PLCE) to its customers. The encrypted PLCE partition enables the machine manufacturer to protect data and files better because the encryption makes it more difficult for third parties to access this information. HEIDENHAIN points out that no data encryption method offers 100% protection of the data, especially against access, damage or destruction by unauthorized persons, etc. Therefore, HEIDENHAIN of course cannot guarantee that data stored there will not be tampered with, and as a consequence cannot assume liability for any resulting damage. This applies to the PLCE partition of the control, as well as to the programming station version.

If the password is stored in the control, the NC software can access the data and files in the **PLCE:** partition at any time. The control can therefore run on the PLCE partition in the same way as on the standard PLC partition.

The NC software automatically searches the PLC partition for files that it needs during run time. If the search is not successful and a PLCE partition exists, the search is continued in the PLCE partition.

This does not apply to the OEM.SYS file itself, or to data and files that are called by using absolute paths (e.g. in PLC modules, OEM.SYS). The OEM.SYS file must always be stored in the PLC partition of the control. If data or files are called by using an absolute path, the NC software only searches the given path for the data or files. The search is not automatically continued in the other partition.



Note

Do not store the same files both in the PLC partition and the PLCE partition.

All data or files on the PLCE partition are automatically encrypted and are therefore protected from unauthorized access. The only exception is the active machine parameter file. If the active MP file is stored in the PLCE partition, it can nevertheless be edited when you switch to the **Machine Parameter Programming** mode of operation.

Backup/Restore and PLCE

The control operator does not need to know the password to back up or restore the PLCE data. The encrypted content remains protected. When restoring a backup on a control, you must make sure that the password on the control (in the SIK or OEM.SYS) is identical to the password that was used for creating the **PLCE.BIN** file or the PLCE partition.

To enable automatic updating of the PLCE partition through a restore process or software update, you must manually separate the PLCE partition before the restore process by using **Unmount**.

You must keep the following in mind to ensure that the PLCE partition is successfully overwritten with the **PLC:\PLCE.BIN** file during a restore process or a software update:

- The password for the PLCE partition is stored in the SIK or OEM.SYS in the control.
- The password in the control matches the password in PLCE.BIN.
- The PLCE partition has been separated (**Unmount**).
- The PLCE partition offers sufficient storage space for the new data.

Software update and PLCE

A software update can contain either the **PLC:\PLCE.BIN** file or the **PLC:_mpupdate\plce.zip** file.

If the software update contains a **PLC:\PLCE.BIN** file, the requirements described above under "Backup/Restore and PLCE" apply as well.

If the software update contains a **PLC:_mpupdate\plce.zip** file, the following requirements apply:

The .zip file must be encrypted with the same password as the **PLCE:** drive. When the control is started up the next time, this file is unpacked and the corresponding files are copied to **PLCE:**. This .zip file can be created using PLCdesignNT with version 2.5 or later.



Danger

Please note that the encryption algorithm of this .zip file is less secure than encrypting the PLCE partition on the hard disk of the control.



You must keep the following in mind to make sure that the data is transferred to the PLCE partition (with the **PLC:_mpupdate\plce.zip** file) during a software update:

- The password for the PLCE partition is stored in the SIK or OEM.SYS in the control.
- The password in the control matches the password in plce.zip.
- The PLCE partition exists and is mounted (**Mount**).
- The PLCE partition offers sufficient storage space for the new data.

Automatic binary-to-ASCII conversion is not possible for the encrypted **PLCE** partition during a software update. This means that you must manually convert binary-coded files on the PLCE partition to ASCII format before a software update. After the software update, the files must be reconverted back to the binary format. To convert the files manually, proceed as follows:

- ▶ Switch to the Machine Parameter Programming mode of operation.
- ▶ Press the MOD key and then the UPDATE DATA soft key.
- ▶ Use the CONVERT BIN -> ASC soft key to convert the files to ASCII format before a software update. Use the CONVERT ASC -> BIN soft key to convert the files back to binary format after the software update.

In order to avoid converting the files manually, you could also load the binary-coded files onto the PLCE partition after a software update.

PLC project

A plce.sys file can be stored in the PLC:_mpupdate directory. In this file, you can save the information for creating an encrypted PLCE: drive in the following form:

CREATE -s=<size> -p=<Password> [-oemsys] [-force]

This enables the PLC project to create the PLCE partition automatically. You no longer need to create the PLCE partition manually before loading the PLC project. Where:

- <size>
Size of the drive to be created (in megabytes between 1 MB to 100 MB)
- <Password>
Encryption password, this password can also be in encrypted form: For example, if a password for an encrypted drive has been created manually, saved in OEM.SYS and then copied from OEM.SYS. The password must be between 12 and 21 characters long.
- [-oemsys]
If this option is set, the password will be saved in OEM.SYS. If the option is not set, it will be saved in the SIK.
- [-force]
If this option is set, an existing encrypted drive will be overwritten. If the option is not set, it will not be overwritten.
- Examples:
CREATE -s=1 -p=VeryLongPassword -oemsys
CREATE -s=80 -p=ShortPassword -force

**ENCCYC.
ZIPNAME =**

The possibility of machining with or without preset tables in OEM cycles can also be used with the encrypted **PLCE** partition. A separate directory is created on the PLCE partition for each cycle project (PLCE:\OEMCYC_ZIP\, PLCE:\OEMCY2_ZIP\, etc.), just as is done on the PLC partition. In the OEM.SYS, enter after the code word the name of the *.ZIP file to be unpacked. For example, this is for the PLCE partition: **ENCCYC.ZIPNAME = ABC.ZIP, ENCCY2.ZIPNAME = DEF.ZIP**, etc. The *.ZIP files contain all information for the cycles, including the directory structure. When the control is started up, the appropriate *.ZIP files are unpacked in the folders. The documentation for CycleDesign contains more detailed information.



HEIDENHAIN offers the following PC tools for working with the data of the encrypted partition: PLCdesignNT, PLCtext, CycleDesign and TNCremoNT.

The PLCE partition cannot be accessed until the correct password has been entered.



Note

An encrypted **PLCE:** partition can be created on the iTNC 530 programming station as of NC software version 606 42x-02.

To activate the PLCE partition on the iTNC 530 programming station, you must additionally install a special Windows driver. One driver each for 32-bit Windows systems and 64-bit Windows systems is provided on the iTNC 530 programming station CD with the ID 535 373 starting with variant 15 (ID 535 373-15). For 32-bit operating systems, the drivers are located in the `\jh\i386` directory of the CD and for 64-bit operating systems in the `\jh\amd64` directory. The basic information about your computer (Control panel/System) tells you which Windows operating system you are using. Install your respective driver as follows:

- ▶ Copy all files of a CD directory for the respective driver into any directory on your computer, e.g. `C:\Programs\iTNC530\606424\Drivers`.
- ▶ Go to Accessories to open a command window or prompt and switch to the directory into which you have copied the files.
- ▶ Execute the **fdcnt1 install** command
- ▶ Execute the **fdcnt1 start** command
- ▶ Then you can create, mount and use the PLCE partition as usual.

In the programming station version, the password for the PLCE partition is saved on the programming station dongle, and not on the SIK. This function is currently only supported for the blue Aladdin dongle.

To uninstall the driver, please take the following steps:

- ▶ Go to Accessories to open a command window or prompt and switch to the directory into which you have copied the files during the installation.
- ▶ Execute the **fdcnt1 uninstall** command.



Note

- Please note that after uninstalling the driver, the computer must be restarted.
- Please note that by uninstalling the driver, the data of the PLCE partition are not deleted! The PLCE.BIN file is retained, only the driver is uninstalled. To delete the data, the PLCE partition must be formatted prior to the uninstallation.

Possible settings

In the PLC mode of operation, press the ADVANCED SETUP soft key and then the ENCRYPTED DRIVE soft key to call the settings of the encrypted partition.

The following information and options are available on the **Password** tab:

PLC encrypted volume

Password **Drive**

State

no password

password stored in OEM.SYS

password stored in SIK

Actions

Set Password Delete Password MODE OEM.SYS ->SIK

End

Password – State:

Display	Meaning
No password	Active if you have not yet assigned a password to the PLCE: partition.
Password stored in OEM.SYS	Active if the password for the PLCE: partition is saved in the OEM.SYS file.
Password stored in SIK	Active if the password for the PLCE: partition is saved in the SIK.

Password – Actions:

Press	Meaning
Set password	Assign a password for access to the PLCE: partition.
Delete password	Delete the password for access to the PLCE: partition.
Move OEM.SYS -> SIK	Move the password for access to the PLCE: partition from the OEM.SYS file to the SIK.

The following information and options are available on the **Drive** tab:

PLC encrypted volume

Password | Drive |

State

no encrypted drive drive is mounted
 encrypted drive, size = MB drive is not mounted, but ok
 drive is not formatted or wrong password

Actions



Drive – State:

Display	Meaning
No encrypted drive	Active if no PLCE: partition has been created yet.
Encrypted drive, size = xxx MB	Active if PLCE: partition has been created. The size of the partition is indicated in [MB].
Drive is mounted	Active if PLCE: partition is mounted.
Drive is not mounted, but OK	Active if PLCE: partition is available, but not mounted.
Drive is not formatted or wrong password	Active if the PLCE: partition cannot be used because it is formatted incorrectly, or because the password is incorrect.

Drive – Actions:

Button	Meaning
Create Drive	The encrypted PLCE: partition is created and the size of the partition is indicated in [MB]. Only available if no PLCE: partition has been created yet. Maximum size: 100 MB Minimum size: 1 MB
Delete Drive	Delete the encrypted PLCE: partition. Only available if the PLCE: partition is not mounted.
Change Size	Change the size in [MB] of the PLCE: partition. Only available if the PLCE: partition is not mounted.
Mount	Mount the PLCE: partition. Only available if the PLCE: partition is not mounted.
Format	Format the PLCE: partition. Only available if the PLCE: partition is not mounted.
Unmount	Unmount the PLCE: partition. Only available if the PLCE: partition is mounted.



Create and mount the PLCE partition

1. To assign a password to the encrypted partition, proceed as follows:
 - ▶ While in the **Programming and Editing** operating mode, press the MOD key.
 - ▶ Enter the code number 807 667 to switch to the PLC operating mode.
 - ▶ Press the ADVANCED SETUP soft key.
 - ▶ Press the ENCRYPTED DRIVE soft key.
 - ▶ Press the **Set Password** button under **Actions**.
 - In **New Password**: you enter a valid password for access to the encrypted partition.
The password must be between 12 and 21 characters long. You can use numbers and special characters, but do not use any umlauts. The password is case sensitive.
 - Repeat the selected password in **Verification**:
 - In **Store Password in** you select whether the password is saved in the SIK or in the OEM.SYS.
 - Confirm your entry with the **OK** button.



Note

- When entering the password, keep in mind that the password is case sensitive.
- HEIDENHAIN recommends storing the password in the SIK.

The control confirms the password by displaying the message **Password has been set**, and under **State** it shows where the password is stored.

2. To create an encrypted partition, proceed as follows:

- ▶ Select the **Drive** tab.
- ▶ Press the **Create Drive** button under **Actions**.
 - In **Size**:, you enter the desired size for the encrypted partition [MB].
 - Confirm your entry with the **OK** button.

The control confirms the creation of the **PLCE**: partition by displaying the message **Encrypted drive has been created** and changes the status under **State** to indicate that the partition has been created.

The right column beneath **State** shows that the partition exists, but that it is not mounted.

3. To mount the encrypted partition, proceed as follows:

- ▶ The newly created **PLCE**: partition must first be formatted:
To perform the formatting process, press the **Format** button and then press the **Apply** button.



Danger

All data stored on the PLCE: partition are lost during the formatting process.

- ▶ To mount the encrypted partition, press the **Mount** button and then press the **Apply** button.

The control confirms that the **PLCE:** partition has been mounted by displaying the message **Encrypted drive has been mounted** and changes the status under **State** to indicate that the partition has been mounted.

Access to PLCE: by the user

The control operator cannot access the mounted **PLCE:** partition until he has entered the correct password for the PLCE partition.

- ▶ While in the **Programming and Editing** operating mode, press the MOD key.
- ▶ Enter the password for the encrypted **PLCE:** partition.

If you then press the PGM MGT key, the iTNC displays the **PLCE:** partition in the directory structure on the left side of the screen.

If the PLCE partition is displayed in the **File Management**, files and data can be handled in the same way as in all other partitions (e.g. move, delete, rename). Encryption is automatic.



9.4 System Files

9.4.1 OEM.SYS

In the OEM.SYS file you must enter code words to call certain functions. After the code word, and separated by an equal sign = you enter the directory in which the files for these functions are to be found, as well as the file names themselves.

You must make your entries in the OEM.SYS file either manually or with Module 9271 (MPFILE and PLCMAIN can only be entered manually). Module 9271 overwrites the content of existing code words and inserts non-existent code words at the end of the OEM.SYS file.

With Module 9270 you can read all entries in the OEM.SYS file.

OEM.SYS is reevaluated during activation of the **machine-parameter programming** operating mode and before downloading a machine parameter file. In connection with TNCremoNT 2.2, during the restoring of a backup, first OEM.SYS is transferred and also evaluated before transferring the machine parameter file.

The following code words are defined (in alphabetical sequence):

- AXISNUMBER =** Number of the indexes of the machine parameters (except MP2xxx.y) in the machine-parameter file.
Input example: **AXISNUMBER = 6**
- DEBUGPATH =** Path for the most recently active *.wlt file for the WATCH LIST function. Other *.wlt files can also be saved in this folder. The folder is also used as the working directory for the WATCH LIST function. When you select a new *.wlt file, the iTNC automatically enters that name in OEM.SYS.
Input example: **DEBUGPATH = PLC:\DEBUG\TEMP.WLT**
- FNERRFIX =** **FN14: ERROR** = gives additional information about fixing an error. The text and information about the cause of the error (**FNERRREASON =**) are shown after pressing the HELP key. The setup of the file must correspond to that of the file for PLC error messages (**PLCERRFIX =**).
Input example: **FNERRFIX = FN14-FIX.A**
- FNERROR =** ASCII file (*.A) that contains the error messages for **FN14: ERROR** = (0 to 299) in lines 1 to 300. The file must be located under PLC:\LANGUAGE<language>, where <language> depends on MP7230.3.
Input example: **FNERROR = FN14-ERR.A**
- FNERRREASON =** **FN14: ERROR** = gives additional information about the cause of an error. The text and information about fixing the error (**FNERRFIX =**) are shown after pressing the HELP key. The setup of the file must correspond to that of the file for PLC error messages (**PLCERRREASON =**).
Input example: **FNERRREASON = FN14-REASON.A**

FIXTURE. FUNCTION =	Activating fixture monitoring in combination with collision monitoring DCM. Input example: FIXTURE.FUNCTION = YES
IOCCFG =	Complete configuration of the HSCI system as an IOC file. This is done using the IOconfig tool. Input example: IOCCFG = PLC:\IOC\HSCI_SYS.IOC
KINEMATIC =	Path for the assignment table of the tilting-axis geometry description. Input example: KINEMATIC = PLC:\KINELIST.TAB
LOCKEDOPTIONS =	You can use the LOCKEDOPTIONS = keyword to select software options that will not be enabled with the General Key. <option> is the number of the software option to be disabled. This way you can disable those software options for which your machine is not prepared. Input example: LOCKEDOPTIONS = 44,45
LOGO =	Path for customer-specific company logo during control power-up. Input example: LOGO = PLC:\LOGO\OEM-LOGO.BMP
LOGOSP =	Path for a customer-specific company logo shown when the control is started, instead of the note regarding an installed service pack. Input example: LOGOSP = PLC:\LOGO\SP-LOGO.BMP
LSV2TIME0 =	Timeout for block reception (STX to ETX).
LSV2TIME1 =	Timeout for acknowledging ENQ or check sum.
LSV2TIME2 =	Timeout during transmission of DLE 0, DLE 1 or NAK until reception of a valid character.
MODEHELP =	Path for help texts and machine commands. Input example: MODEHELP = PLC:\LANGUAGE\GERMAN\OPTIMIER.HLP
MPFILE =	(Mandatory entry): Path for the active MP file. If you have loaded an MP file editor and you exit the editor, the iTNC automatically enters this MP file in the OEM.SYS file! Input example: MPFILE = PLC:\MP\NC530V02.MP



MPFRAGMENT
<Value> =

Up to 10 (<value> from 0 to 9) machine parameter subfiles can be defined. They can be activated with **FN17: SYSWRITE ID1020 NR1 = <value>**. Changes to the spindle machine parameters (MP3xxx or MP13xxx) are only active after an S output. All other machine parameters are active immediately. The changes also remain in effect if a new NC program is selected, but not if the control is restarted.

Input example: **MPFRAGMENT0 = PLC:\MP\FINISH.MP**



Note

Please note that **MPFRAGMENTx** and **MPFRAGMENTFILE...** of course cannot be used as keywords at the same time.

MPFRAGMENTFILE
=<path/file name>

By entering **MPFRAGMENTFILE = <path\file name>**, you specify a file containing the paths to machine-parameter subfiles and the file names, which can be activated via FN17.

The value entered in the **FN17 SYSWRITE ID 1020 NR1 = <line number>** function then corresponds to the MP subfile reference in this file.

This makes it possible to enter any number of MP subfiles (until now only 10 files **MPFRAGMENT0-9 = <path>**).

Changes to spindle machine parameters (MP3xxx or MP13xxx) are only active after an S output. All other machine parameters are active immediately.

The changes also remain in effect if a new NC program is selected, but not if the control is restarted.

Input example: **MPFRAGMENTFILE = PLC:\MP\MPFILELIST.MP**



Note

Please note that **MPFRAGMENTx** and **MPFRAGMENTFILE...** of course cannot be used as keywords at the same time.

MPLOCKFILE =

Path of a machine parameter subfile. If there are differences between this file and the current machine parameter file, an error message appears and the value from this subfile is offered for acceptance.

Input example: **MPLOCKFILE = PLC:\MP\340420.MPL**

MPPASSWORD =

Code number for calling the machine parameter file (instead of 95148).

Input example: **MPPASSWORD = MP**



Note

Do not enter a code number that has already been defined by HEIDENHAIN!

- NUMBERMP4111 =** Number of required timers > 96. The corresponding number of machine parameters MP4111.96 to MP4111.x is created.
Input example: **NUMBERMP4111 = 10**
(Machine parameters MP4111.96 to MP4111.105 are created.)
- NUMBERMP4230 =** Sets a number in the PLC for Module 9032. The corresponding number of machine parameters MP4230.x is created. The maximum input value is 99; no entry or an invalid entry defines 32 indexes.
Input example: **NUMBERMP4230 = 40**
(Machine parameters MP4230.0 to MP4230.40 are created.)
- OEMCYC.
ZIPNAME =** The possibility of machining with or without preset tables can also be used in OEM cycles. A separate directory is created on the PLC partition for each cycle project (PLC:\OEMCYC_ZIP\, PLC:\OEMCY2_ZIP\, etc.). In OEM.SYS, enter after the keyword the name of the *.ZIP file to be unpacked, e.g.
OEMCYC.ZIPNAME = ABC.ZIP, OEMCY2.ZIPNAME = DEF.ZIP, etc. The *.ZIP files contain all information for the cycles, including the directory structure. When the control is started up, the appropriate *.ZIP files are unpacked in the folders. The documentation for CycleDesign contains more detailed information.
- PALETPRESET =** Activate pallet preset table in order to manage reference points for pallets, see page 1495
Input example: **PALETPRESET = TNC:\Palletpreset.pr**
- PLCCOMPCFG =** Configuration file for conditional compiling.
Input example: **PLCCOMPCFG = PLC:\OEM\OEM.CFG**
- PLCDIALOG =** Name for text file with PLC dialogs; the path for the text file is permanently defined.
Input example: **PLCDIALOG = DIALOG.A**
- PLCERRFIX =** Path for "Corrective action" help text.
Input example: **PLCERRFIX = FIX.A**
- PLCERROR =** Name for text file with PLC error messages; the path for the text file is permanently defined.
Input example: **PLCERROR = PLC_ERR.A**
- PLCERRREASON =** Path for "Cause of error" help text.
Input example: **PLCERRREASON = REASON.A**



- PLCERRTAB =** (Mandatory entry for PLC error messages): Path for PLC error message table. If you compile a PLC program, the iTNC automatically enters it in the OEM.SYS file.
Input example: **PLCERRTAB = PLC:\ PLC_PGM \ERR_TAB.PET**
- PLCEVENTS =** Path for event list (SPAWN command).
Input example: **PLCEVENTS = PLC:\EVENTS.PEV**
- PLCMAIN =** (Mandatory entry): Path for the active PLC program. If you compile a PLC program, the iTNC automatically enters it in the OEM.SYS file.
Input example: **PLCMAIN = PLC:\PLC_PGM\MAIN_530.PLC**
- PLCPASSWORD =** Code number for calling the PLC mode (instead of 807667).
Input example: **PLCPASSWORD = 123456789**
- 
- Note
- Do not enter a code number that has already been defined by HEIDENHAIN!
- PLCPWM =** Path for PLC program for commissioning of digital axes
Input example: **PLCPWM = PLC:\IB_PGM\IB530.PLC**
- PLCSOFTVERS =** (Mandatory entry): iTNC displays PLC software version when the MOD key is pressed.
Input example: **PLCSOFTVERS = BASIS--33-03**
- PR.LINESLOCKED =** Lines in the preset table that are to be write-protected, see page 1208.
Input example: **PR.LINESLOCKED = 1,4-8,22**
- PWM PARAMETER =** Number of the indexes of machine parameters MP2xxx.y (for the current and speed controller) in the machine-parameter file (up to 30).
Input example: **PWMPARAMETER = 6**
- REMOTE.LOCKSOFTKEY VISIBLE =** Display **External access ON/OFF** soft key.
Input example: **REMOTE.LOCKSOFTKEYVISIBLE = YES**
- REMOTE.PLCPASSWORD FORCED =** Setup, machine backup and full backup only with the password from **PLCPASSWORD =**
Input example: **REMOTE.PLCPASSWORDFORCED = YES**
- REMOTE.PLCPASSWORD NEEDED =** Access to the PLC partition using the LSV2 protocol only with the password from **PLCPASSWORD =**
Input example: **REMOTE.PLCPASSWORDNEEDED = YES**

REMOTE. TIMEKEEPIDLE =	<p>Network connections for LSV2 applications are cyclically checked. If the remote station does no longer respond (e.g. because the connecting cable was disconnected), the LSV2 connection will automatically be terminated. In the default setting, the connection is checked four times within a ten-second cycle. If the remote station does not respond after the fourth attempt, the connection is automatically terminated.</p> <p>If you do not want a cycle of ten seconds, you can increase the cycle time to a maximum of 6000 seconds with the REMOTE.TIMEKEEPIDLE token.</p> <p>Input example: REMOTE.TIMEKEEPIDLE = 250</p>
RUNTCH =	<p>Indicate a specific pocket table. If this token exists in OEM.SYS, files of the type *.tch are not displayed in the file manager and in the machining modes.</p>
SERVICE. REQUEST. CONTENT =	<p>Only for remote diagnosis with the TeleService PC software: Path of a text file containing the content of the UDP service request package, with which the machine can be identified uniquely. The iTNC reports this content to the TSAgent when the SERVICE soft key is pressed. With short texts it can also be entered directly in quotation marks. If this entry is missing, then the default value PLC:\REQUEST.SYS is used automatically.</p> <p>Input example: SERVICE.REQUEST.CONTENT = PLC:\REQUEST.SYS</p>
SERVICE. REQUEST.HOST =	<p>Only for remote diagnosis with the TeleService PC software: IP address of the PC that receives the service request.</p> <p>Input example: SERVICE.REQUEST.HOST = 160.1.180.99</p>
SERVICE. REQUEST.PERIOD =	<p>Only for remote diagnosis with the TeleService PC software: Repeat time in seconds for the UDP packets. If this entry is missing, then the default value 10 is used automatically.</p> <p>Input example: SERVICE.REQUEST.PERIOD = 10</p>
SERVICE. REQUEST.PORT =	<p>Only for remote diagnosis with the TeleService PC software: Port for the UDP packets. If this entry is missing, then the default value 19001 is used automatically.</p> <p>Input example: SERVICE.REQUEST.PORT = 19001</p>
SERVICE. REQUEST. TIMEOUT =	<p>Only for remote diagnosis with the TeleService PC software: Timeout checking interval in minutes—if there has been no data transfer over the Ethernet since the last timeout check, transmission of the service requests is terminated. If this entry is missing, then the default value 15 is used automatically.</p> <p>Input example: SERVICE.REQUEST.TIMEOUT = 15</p>
SOFTKEY PROJECT =	<p>Path for PLC soft-key project file *.spj with the structure of the vertical PLC soft key.</p> <p>Input example: SOFTKEYPROJECT = PLC:\SOFTKEY.SPJ</p>

- SPLCSOFTVERS =** Version identifier of the SPLC program. The entry can be displayed by pressing the MOD key and can be requested via LSV2.
- SOFTKEYPRESS =** Display and behavior of the CHECK and RADIO types of soft keys. If **SOFTKEYPRESS = ON** is entered, the soft-key graphics and the states of the PLC markers already change when the soft key is pressed, see page 1376.
Input example: **SOFTKEYPRESS = ON**
- TABCMA =** Path for compensation value tables for axis error compensation (see "Nonlinear axis error compensation" on page 652).
Input example: **TABCMA = PLC:\AXIS_COR\CORRECT.CMA**
- TCHRULES =** Path for the *.TCR definition file containing magazine rules for tool magazines.
Input example: **TCHRULES = PLC:\RULES.TCR**
- TNCOPT.LOCK-SOFTKEYVISIBLE =** After pressing the MOD key, the **TNCOPT ON/OFF** soft key is displayed in order to enable commissioning on the control via TNCopt, see page 1023.
Input example: **TNCOPT.LOCKSOFTKEYVISIBLE = YES**
- TTYP =** Path and file name for list of the tool types.

Module 9270: Read a code word

With Module 9270 you can read an entry from the OEM.SYS file.

Call:

PS B/W/D/K/S<>String with code word>

PS B/W/D/K <>String number for result>

CM 9270

Error code:

Marker	Value	Meaning
M4203	0	Interface enabled
	1	Error. See W1022.
W1022	3	Not a valid string for code word or result
	12	String for code word is too long
	20	Module was not called in a spawn job or submit job
	30	Code word was not found

Module 9271: Write a code word

With Module 9271 you can write an entry into the OEM.SYS file.

Call:

PS B/W/D/K/S<>String with code word>

PS B/W/D/K <>String number for result>

CM 9271

Error code:

Marker	Value	Meaning
M4203	0	Interface enabled
	1	Error. See W1022.
W1022	3	Entry was written
	6	PLCMAIN or MPFILE was transferred
	12	String for code word is too long
	30	Module was not called in a submit or spawn job

9.4.2 NCMACRO.SYS

The NC macros are defined in this file. Certain NC macros are predefined. You can also define new NC macros, see "Module 9291 Call an NC macro" on page 1711.

The following NC macros are predefined:

- AFC = <Name of the macro that is called as a reaction to a detected overload from AFC>
-
- TC = <Name of the tool change macro>
- PALETT = <Name of the pallet change macro>
- CLAMP = <Name of the macro for changing the fixture (FIX)>
- RUNCANCEL = <Name of the macro called when an NC program is canceled>
- MACROCANCEL = <Name of the macro called when an NC program is canceled>
- RESETINIT = <Name of the macro called when **traverse reference point** is left>
If the macro is not performed completely,
 - you cannot switch to the program run operating modes,
 - the error message **Machine not initialized** appears, and
 - the soft key **INIT** appears. The soft key can be used to restart the macro.
- STARTUPCANCEL = <Name of the macro called when mid-program startup is not completed with **Restore machine status**>
- TCSIMU = <Name of the tool-change macro that is called in the Test Run mode for a tool change. As with the TOOL CALL macro for the machine operating modes, a TOOL CALL block must be executed within a macro in order to activate the new tool>
- RUNENDPGM = <Name of the macro that is run with M02, M30 or END PGM when a program is concluded. In the Manual operating mode, this macro must also be saved in the M-function table under M02 and M30.>
- PALEPILOG = <Name of the macro called at the end of an NC program that was started from the pallet table>

Example entry: TC=PLC:\NC_MACRO\TOOLCALL.H

With RUNCANCEL you can define a macro that is run when an NC program is terminated due to an error, emergency stop or external/internal stop. Use MACROCANCEL to do the same for an NC macro. If an NC program calls a macro (tool call or M function) and the macro is canceled, this is considered a program cancelation, not a macro cancelation.

In order to increase the speed with which NC macros are executed, limit-switch monitoring can be turned off with **FN17: SYSWRITE ID230 NR5**. At the end of an NC macro the limit-switch monitoring is always switched on. For transferring Q parameters between NC programs and NC macros, see page 1714.

PGM CALL, including NC macros, and **CYCL CALL** (for cycles greater than 68) are calculated automatically with the look-ahead function and run without exact stop. At the beginning and end of the called program or cycle, it can happen that a missing synchronization between machine status and look-ahead calculation may lead to problems.

Example:

A **TOOL CALL** is run in look-ahead calculation. In this **TOOL CALL** a PLC function is needed (e.g. opening the tool changer gate). The tool is automatically changed on the machine. During this time the look-ahead calculation reaches another **TOOL CALL**. Since the PLC function has been fulfilled (the tool changer gate is open), the look-ahead calculation is continued. After the first tool change has been completed, the PLC function is no longer fulfilled (the tool changer gate is closed). The second **TOOL CALL** would be executed if the PLC function were not fulfilled (the tool changer gate is closed).

The function **FN20: WAIT FOR SYNC** provides a remedy for this problem. If this function is programmed at the beginning of an NC program (NC macro) or cycle, in the look-ahead calculation the **PGM CALL** (NC-macro call) or **CYCLE CALL** (NC-macro call) or **CYCL CALL**.

Look-ahead is halted for **FN20: WAIT FOR SYNC**, no matter at what point this block is programmed.



Module 9291 Call an NC macro

With Module 9291, you can call an NC macro in any operating mode. They are executed like cycles, without block display. The control-in-operation symbol is displayed while the macro is being executed. No macros can be activated if there is currently an **External emergency stop** error message.

The predefined code words of the NCMACRO.SYS file and the code words defined by the user can be transferred. They only need to be entered in NCMACRO.SYS to be defined. To prevent name conflicts with future HEIDENHAIN code words, your code words should begin with the character "P\$" or with the name of the company.

PLC Module 9291 starts an NC macro with the same mechanisms as program selection and starting in the user interface, i.e. with the selection of the NC macro the same modal settings are reset as with PGM MGT and Enter. This means that the continuation of an already started NC program must be prevented after Module 9291 has been called.

Call:

PS B/W/D/K/S<>Keyword>

CM 9291

Error code:

Marker	Value	Meaning
M4203	0	NC macro was executed
	1	Error code in W1022
W1022	2	NCMACRO.SYS does not exist, code word does not exist, or invalid string
	7	Macro cannot be executed.
	8	External emergency stop active
	20	Module was not called in a submit or spawn job
	28	NC program or other macro is already running
	29	The file given under the keyword is not an NC program (*.H or *.I)
	36	The file given under the keyword does not exist

9.4.3 MGROUPTS.SYS

In the system files PLC:\MGROUPTS.SYS and PLC:\MSPLIT.SYS, you define the M functions to be output after a block scan (see "Returning to the contour" on page 1244).

9.4.4 MSPLIT.SYS

M functions that are effective in several groups are divided in the MSPLIT.SYS file into function components (see "Returning to the contour" on page 1244).

9.4.5 PLCSOFTK.SYS

Path for the file names of the PLC soft-key pictures. (See "PLC soft keys" on page 1372.)

9.4.6 CYCLE.SYS

Definition of the soft-key structure, if you have integrated OEM cycles. This file is created automatically by the PC software CycleDesign (see OEMCYC directory).

9.4.7 TNC.SYS

The end user can define certain paths and functions in the TNC:\TNC.SYS file:

- **TMAT** = <Path for list of tool materials>
- **WMAT** = <Path for list of workpiece materials>
- **PCDT** = <Path for cutting data tables>
- **REMOTE.TNCPASSWORD** = <Password for LSV2 access>
- **REMOTE.TNCPRIVATEPATH** = <Path to be protected by the password>
- **REMOTE.PERMISSION** = <computer>

External access to the control via LSV2 can be limited to specified computers.

If the entry **REMOTE.PERMISSION = <computer>;<computer>** exists in the TNC.SYS file, then only the computers listed will be granted access to the control. The entry can be either the names or IP addresses of computers and must be separated by semicolons. Access to the control is refused if the REMOTE.PERMISSION entry exists in the TNC.SYS file but the computer is not listed. If the REMOTE.PERMISSION entry does not exist, then every computer can access the control.

- **PR.LINESLOCKED** = <line number>

You can write-protect line of the preset table. A write protection for individual lines or entire areas in the preset table can now be set by an entry of **PR.LINESLOCKED** = (e.g. **PR.LINESLOCKED = 1,4-8,22**). Write-protection can be activated only for the first 255 lines of the preset table. Protected lines are indicated in the editor by a different color.



9.5 Data Transfer NC > PLC, PLC > NC

Information is exchanged between PLC and NC by markers, bytes, words and double words. The function of the individual markers, bytes, words and double words is fixed.

The transfer of certain data to the PLC is controlled by strobes:

- M codes
- S codes
- T codes
- G codes
- Q codes

Example:

If an M function is output, the NC sets the strobe signal M4072. After evaluating the M function, you must first set the acknowledgement marker M4092 and then reset it. The PLC must then reset the acknowledgment marker, otherwise no further strobes can be sent by the NC.



Note

- PLC messages that would terminate an NC strobe are only released once all PLC functions that are only permitted during the strobe have been completed.
- If there is an emergency stop, current NC strobes are automatically acknowledged by the NC, and in this case should not be acknowledged additionally by the PLC.

9.5.1 Data transfer NC program > PLC (FN19: PLC =)

With the Q-parameter function **FN19: PLC =** you can transfer two values from an NC program to the PLC. The iTNC stores the transferred values as integer values of the form 1/10 000 in the double words D280 and D284. M4570 defines the unit of measure of both numerical values. During transfer, the marker M4075 is set by the NC. The PLC must acknowledge the transfer by setting marker M4095.

		Set	Reset
M4075	Transfer active with FN19	NC	NC
M4095	Acknowledgment of transfer with FN19	PLC	PLC
M4570	Unit of measure for transfer with FN19 0: mm 1: inches	NC	NC
D280	First numerical value from FN19	NC	NC
D284	Second numerical value from FN19	NC	NC

9.5.2 Data transfer PLC > NC program (Q-parameters)

Data transfer from the PLC to the NC program goes through Q parameters Q100 to Q107, i.e., from the PLC you can overwrite Q parameters Q100 to Q107:

- ▶ In double word D528, enter the numerical value to be transferred.
- ▶ In word W516, define the target parameter (0=Q100, 7=Q107).
- ▶ Activate transfer with strobe marker M4131.
- ▶ The iTNC transfers the values with the next strobe.

		Set	Reset
M4131	Activation of Q-parameter transfer to the NC; data from D258, Q number from W516	PLC	NC
D528	Double word with multiple function; here data for transfer from PLC to NC	PLC	PLC
W516	Q no. 0-7 for numerical data transfer PLC → NC	PLC	PLC

Q parameters in OEM cycles are only effective locally if the NC program of the OEM cycle is saved in the *.CYC format, otherwise (*.H) the Q parameters are effective globally.

Cycle 12 (PGM CALL) acts like **CALL PGM**, meaning the Q parameters are effective globally.

Q parameters	Effectiveness/Meaning
Q0 to Q99	Freely applicable parameters, as long as no overlapping with SL cycles can occur, globally effective for all programs stored in the TNC memory
Q100 to Q199	Parameters for special TNC functions
Q200 to Q1199	Parameters that are primarily used for cycles, globally effective for all programs stored in the TNC memory
Q1199 to Q1399	Parameters that should primarily be used for OEM cycles, and are globally effective for all programs stored in the TNC memory. This may require coordination with third-party suppliers.
Q1400 to Q1499	Reserved for OEM cycles (CALL-active) Parameters that are primarily used for call-active OEM cycles, and are globally effective for all programs that are stored in the TNC memory.

Q parameters	Effectiveness/Meaning
Q1500 to Q1599	Reserved for OEM cycles (DEF-active) Parameters that are primarily used for DEF-active OEM cycles, globally effective for all programs that are stored in the TNC memory
Q1600 to Q1999	Freely applicable parameters, globally effective for all programs stored in the TNC memory
QL0 to QL499	Freely usable QL parameters, only effective locally (within a program)
QR0 to QR499	Freely usable QR parameters that are nonvolatile, i.e. they remain in effect even after a power interruption

QS parameters (the S stands for string) are also available on the TNC and enable you to process texts. In principle, the same ranges are available for QS parameters as for Q parameters (see table above).



Note

- Some Q and QS parameters are always assigned the same data by the TNC. For example, Q108 is always assigned the current tool radius (please also note the list of preassigned Q parameters).
- If you are using the parameters Q60 to Q99 in encoded OEM cycles, define via MP7251 whether the parameters are only to be used locally in the OEM cycles (.CYC file), or may be used globally.
- With MP7300 you specify whether the TNC should reset Q parameters at the end of the program, or if the values should be saved. Make sure that this setting does not have any effect on Q-parameter programs!
- Note that for the QS parameters the QS100 to QS199 range is reserved for internal texts.

MP7251 **Number of global Q parameters starting from Q99 (up to Q60) that are transferred from the OEM cycle to the calling program.**

Input: 0 to 40

9.5.3 Data transfer NC program > NC (FN17: SYSWRITE)

You can use the **FN17: SYSWRITE** function particularly for OEM cycles if you wish to overwrite certain NC data, e.g., an active basic rotation, from the NC program. A group number, a system data number, and an index specify the particular item of system data that you write: **FN17: SYSWRITE IDxxxx NRxxxx IDXxxxx = Qxxx** or numerical value; comment. In the NC program you cannot define function FN17 (soft keys: Q-parameter programming, special functions) until you have entered the code number 555 343. After a control reset the code number must be entered again if you want to program **FN17**. The iTNC provides the following functions:



Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
Switch the spindle				
	20	13	–	0 = spindle 1 1 = spindle 2
		18	–	Second axis-specific rapid traverse
Data from the tool table				
	50	1	Tool no.	Tool length L
		2	Tool no.	Tool radius R
		3	Tool no.	Tool radius R2
		4	Tool no.	Oversize for tool length DL
		5	Tool no.	Oversize for tool radius DR
		6	Tool no.	Oversize for tool radius DR2
		7	Tool no.	Tool locked TL 0 = not locked, 1 = locked
		8	Tool no.	Number of the replacement tool RT
		9	Tool no.	Maximum tool age TIME1
		10	Tool no.	Maximum tool age TIME2
		11	Tool no.	Current tool age CUR. TIME
		12	Tool no.	PLC status
		13	Tool no.	Maximum tooth length LCUTS
		14	Tool no.	Maximum plunge angle ANGLE
		15	Tool no.	TT: Number of tool teeth CUT
		16	Tool no.	TT: Wear tolerance for length LTOL
		17	Tool no.	TT: Wear tolerance for radius RTOL
		18	Tool no.	TT: Direction of rotation DIRECT 0 = positive, -1 = negative
		19	Tool no.	TT: Offset in plane R-OFFS R = 99999.9999
		20	Tool no.	TT: Offset for length L-OFFS
		21	Tool no.	TT: Break tolerance for length LBREAK
		22	Tool no.	TT: Break tolerance for RBREAK radius
		23	Tool no.	PLC value
		24	Tool no.	Probe-center offset in reference axis CAL-OF1
		25	Tool no.	Probe-center offset in minor axis CAL-OF2
		26	Tool no.	Spindle angle for calibration CAL-ANG

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
		27	Tool no.	Tool type for pocket table
		28	Tool no.	Maximum rpm NMAX
		29	Tool no.	PLC value P1
		30	Tool no.	PLC value P2
		31	Tool no.	PLC value P3
		32	Tool no.	Point angle for DRILL and CSINK T-ANGLE
		33	Tool no.	Thread pitch for TAP PITCH
Edit multiple-task tool				
	53	See FN 17-ID50	Tool no.	Completely edit the data of a multiple-task tool for all indexes. Behaves like FN17-ID50, only that here the same value is written to all indexes.
		7	–	Completely disable all indexes of the current multiple-task tool: FN 17: SYSWRITE ID 53 NR7 = 1.0
Edit multiple-task tool				
	59	10	–	Recalculation of the spindle speed by means of the present tool radius from TOOL.T if a cutting velocity is programmed in the TOOL CALL block. The spindle speed is entered in the system datum that can be read with FN18: SYSREAD ID60 NR3.
Coordinate transformations				
	210	1	–	Basic rotation (manual)
		3	–	Active mirror axis Bits 0 to 2 and 6 to 8: Axes X, Y, Z and U, V, W
		6	–	Tilt working plane in Program Run mode (0 = inactive, -1 = active)
		7	–	Tilt working plane in Manual mode (0 = inactive, -1 = active)
Exchange tool axis				
	212	–	–	0: Tool axis Z 1: Tool axis X 2: Tool axis Y 3: Tool axis from TOOL CALL

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
Traverse range				
	230	2	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Negative software limit switches
		3	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Positive software limit switches
		4	Number of axes whose software limit switches are to be overwritten	Number of the first of several consecutive Q parameters 1st Q: Neg. limit switch in 1st axis 2nd Q: Pos. limit switch in 1st axis 3rd Q: Neg. limit switch in 2nd axis etc.
		5	–	Limit-switch monitoring (1 = off, 0 = on)
Swivel axes				
	290	1	–	Description of tilting-axis geometry
		5	–	0: Temperature compensation not active in the kinematics table 1: Temperature compensation active in the kinematics table

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
TS touch trigger probe				
	350	10	–	Tool axis
		11	–	Effective radius
		12	–	Effective length
		13	–	Calibration ring radius
		14	1	Center offset (reference axis)
			2	Center offset (minor axis)
		15	–	Direction of center offset
TT touch probe for tool measurement				
	350	20	1	Center in 1st axis
			2	Center in 2nd axis
			3	Center in 3rd axis
		21	–	Effective radius
		22	1	Probing position 1 in axis X
			2	Probing position 1 in axis Y
			3	Probing position 1 in axis Z
		23	1	Probing position 2 in axis X
			2	Probing position 2 in axis Y
			3	Probing position 2 in axis Z
		24	1	Probing position 3 in axis X
			2	Probing position 3 in axis Y
			3	Probing position 3 in axis Z
		25	1	Probing position 4 in axis X
			2	Probing position 4 in axis Y
			3	Probing position 4 in axis Z
Coordinate transformations				
	420	0	0	0 = globally effective
Write values into active datum table				
	500	Line	Column	Depends on MP7475
	501	Line	Column	

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
Write values into the active preset table				
	502	Line	Column	Write value with conversion from the currently active coordinate system into the preset table
	503	Line	Column	Write value into the preset table without conversion
	504	Line	Column	Write basic rotation into the preset table
	530	1	–	Activate preset
Pallet preset table				
	540	1	–	Active line in the pallet preset table
Save machine statuses				
	590	1	1 to 9	"Variables" 1 to 9 can only be read with FN18: SYSREAD ID590 NR1 . Deletion depends on MP7300.
		2	1 to 9 (X, Y, Z, A, B, C, U, V, W)	The "variables" for ID621. Independent of MP7300.
		3	1 to 9 (X, Y, Z, A, B, C, U, V, W)	The "variables" for ID621. Independent of a power failure.
	621	0	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Stop measurement of the dynamic load and save the result.
Velocity semifeedforward control				
	600	1	Axis	Factor for velocity semifeedforward
		2	0 or NO ENT	Use factor from MP1396.x
Adaptive Feed Control (AFC)				
	622	0	0.0	0: AFC not active 1: AFC active

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
		0	1.0	Time in seconds after which the teach-in cuts of the AFC are automatically ended. 0: Function not active
		0	3.0	Definition of reference power in [%] of the rated power output of the spindle
		0	4.0	Monitoring of tool wear: Definition of a monitoring limit in [%] of the reference power. PLC Marker 4510 is set if the monitoring value is exceeded.
		0	5.0	Spindle load monitoring (cut-based): Definition of a monitoring limit in [%] of the reference power An NC stop is triggered if the monitoring value is exceeded.
Spindle load monitoring independent of cutting				
	622	Filter constant: 1 to 60 ms	6.0	Spindle load monitoring (independent of cutting): Definition of monitoring limit in [%] of the rated power output of the spindle. An NC stop is triggered if the monitoring value is exceeded.
Touch probe cycles				
	990	1	–	Approach behavior: 0 = standard behavior 1 = effective radius, set-up clearance is zero
		2	–	0.0 = touch probe monitoring off, M4057 not used 1.0 = touch probe monitoring on, M4057 not used 2.0 = touch probe monitoring off, M4057 used 3.0 = touch probe monitoring on, M4057 used
		3	–	Place probe data of the manual probing cycles into the tool table
		6	–	Touch probe cycle 3 0.0 = TS inputs at X112 1.0 = TT inputs at X112

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
Coordinate transformation				
	990	4	1	Transformation from the coordinate system in Manual mode to the active coordinate system (e.g. rotated, shifted).
			2	Transformation of the active coordinate system (e.g. rotated, shifted) into the coordinate system in Manual mode.
		5	5	Interrogation whether axis is transformed to another axis in the untilted coordinate system through a swivel motion. The number of the first of two successive Q parameters must be transferred. The parameter contains the axis to be enquired (0 = X, 1 = Y, 2 = Z). The second Q parameter supplies the respective image (0 = X, 1 = Y, 2 = Z, -1 = axis is not imaged).
		8	-	Spindle orientation including the angle
Changing machine parameters in the run-time memory				
	1000	MP number	MP index	Value of the machine parameter. The change remains in effect until the control is rebooted or the MP file or MP subfile is selected again. The unit must be the same as the unit of the MP in the MP file. For linear entries the values are always written in [mm].
Activate machine parameter subfile				
	1020	1	<Value depends on type of reference in OEM.SYS>	The machine-parameter subfiles defined in OEM.SYS via MPFRAGMENT<value> = (0 to 9) can be activated, or if MPFRAGMENTFILE = <path/file name> given in OEM.SYS is used, you can enter the line with the corresponding machine-parameter subfile (0 to n).

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
PLC data				
PLC data				
Block transfer of up to 8 variables possible				
Example, "Writing Q parameters to PLC markers":				
FN 17: SYSWRITE ID2000 NR10 IDX880 = BLOCK Q1620 - Q1627				
	2000	10	Marker no.	PLC marker
		60	Byte no.	PLC byte
		70	Word no.	PLC word
		80	Double-word no.	PLC double word
The function FN 17: SYSWRITE ID 2001 is identical to the function FN 17: SYSWRITE ID 2000, but it makes it possible to exchange data between the NC program and the PLC without synchronization. However, there are many constraints on using the function FN 17: SYSWRITE ID 2001, which are difficult to estimate.				
This new FN function (FN 17: SYSWRITE ID 2001) may only be used after consultation with HEIDENHAIN.				
	2001	10	Marker no.	PLC marker
		60	Byte no.	PLC byte
		70	Word no.	PLC word
		80	Double-word no.	PLC double word
		90	Output marker no.	Profibus operand output marker
		100	Output byte no.	Profibus operand output byte
		110	Output word no.	Profibus operand output word
		120	Output double word no.	Profibus operand output double word
Set PLC markers for tapping				
	2020	1	–	Set PLC markers 4030 and 4031 for tapping.

9.5.4 Data transfer NC > NC program (FN18: SYSREAD)

The **FN18: SYSREAD** function is particularly useful for OEM cycles if you wish to have access from the part program to certain NC data, such as active tool compensation. A group number, a system data number and an index specify the particular item of system data that you read:

FN18: SYSREAD Qxxx = IDxxxx NRxxxx IDXxxxx (xxxx: Q parameter or numerical value); comment

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
Program information				
	10	1	–	mm = 0, inch = 1
		2	–	Overlap factor for pocket milling
		3	–	Number of the active fixed cycle
		4	–	Number of the last DEF-active OEM cycle
Machine status				
	20	1	–	Active tool number
		2	–	Prepared tool number
		3	–	Active tool axis 0 = X 6 = U 1 = Y 7 = V 2 = Z 8 = W
		4	–	Programmed spindle speed
		5	–	Active spindle status –1 = Spindle status undefined 0 = M3 active 1 = M4 active 2 = M5 active after M3 3 = M5 active after M4
		8	–	Active coolant status 0 = Off, 1 = On
		9	–	Active feed rate
		11	–	Index of active tool
		13	–	Number of the active spindle (0 or 1)
		15	Number of logical axis	Assignment of the logical axes and geometrical axes (0 = X, 1 = Y, 2 = Z, 3 = A, 4 = B, 5 = C, 6 = U, 7 = V, 8 = W)
		17	–	Current range of traverse

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
Cycle parameters				
	30	1	–	Set-up clearance
		2	–	Hole depth / milling depth
		3	–	Infeed depth
		4	–	Feed rate for plunging
		5	–	First side length of pocket
		6	–	Second side length of pocket
		7	–	First side length of slot
		8	–	Second side length of slot
		9	–	Radius of circular pocket
		10	–	Feed rate for milling
		11	–	Rotational direction of the milling path
		12	–	Dwell time
		13	–	Thread pitch
		14	–	Finishing allowance
		15	–	Roughing angle
		52	<no.>	Interrogation whether Q-parameter number <no.> contains a string or numerical value: 0 = tool number 1 = tool name
Data from the tool table				
	50	1	Tool no.	Tool length L
		2	Tool no.	Tool radius R
		3	Tool no.	Tool radius R2
		4	Tool no.	Oversize for tool length DL
		5	Tool no.	Tool radius oversize DR
		6	Tool no.	Tool radius oversize DR2
		7	Tool no.	Tool locked TL 0 = Not locked, 1 = Locked
		8	Tool no.	Number of the replacement tool RT
		9	Tool no.	Maximum tool age TIME1
		10	Tool no.	Maximum tool age TIME2
		11	Tool no.	Current tool age CUR. TIME
		12	Tool no.	PLC status
		13	Tool no.	Maximum tooth length LCUTS
		14	Tool no.	Maximum plunge angle ANGLE
		15	Tool no.	TT: Number of tool teeth CUT

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
		16	Tool no.	TT: Wear tolerance for length LTOL
		17	Tool no.	TT: Wear tolerance for radius RTOL
		18	Tool no.	TT: Direction of rotation DIRECT 0 = Positive, -1 = Negative
		19	Tool no.	TT: Offset in plane R-OFFS R = 99999.9999
		20	Tool no.	TT: Offset in length L-OFFS
		21	Tool no.	TT: Break tolerance in length LBREAK
		22	Tool no.	TT: Breakage tolerance in radius RBREAK
		23	Tool no.	PLC value
		24	Tool no.	Probe-center offset in reference axis CAL-OF1
		25	Tool no.	Probe-center offset in minor axis CAL-OF2
		26	Tool no.	Spindle angle for calibration CAL-ANG
		27	Tool no.	Tool type for pocket table
		28	Tool no.	Maximum rpm NMAX
		29	Tool no.	PLC value P1
		30	Tool no.	PLC value P2
		31	Tool no.	PLC value P3
		32	Tool no.	Point angle for DRILL and CSINK T-ANGLE
		33	Tool no.	Thread pitch for TAP PITCH
Data from the pocket table				
	51	1	Pocket number	Tool number
		2	Pocket number	0 = no special tool 1 = special tool
		3	Pocket number	0 = no fixed pocket 1 = fixed pocket
		4	Pocket number	0 = pocket not locked 1 = pocket locked
		5	Pocket number	PLC status
		6	Pocket number	Tool type
		7 to 11	Pocket number	P1 to P5
		12	Pocket number	0 = not a reserved pocket 1 = reserved pocket
		13	Pocket number	Pocket above is locked
		14	Pocket number	Pocket below is locked
		15	Pocket number	Pocket to the left is locked
		16	Pocket number	Pocket to the right is locked

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
Tool pocket				
	52	1	Tool number	Pocket number P
		2	Tool number	Tool magazine number
File information				
	56	1	–	Number of lines of the selected tool table
		2		Number of lines of the selected datum table
		3	No. of the 1st of 9 consecutive Q parameters for axes X, Y, Z, A, B, C, U, V, W	Number of axes programmed in the selected datum table (the function is identical to FN18: SYSREAD ID990 NR3)
		4	–	Number of rows in a freely definable table that has been opened with FN26: TABOPEN . The function returns the value –1.0 if no table is open at the time of reading.
Values programmed in TOOL CALL				
	60	1	–	Tool number T (the TNC 426/430 reads a tool index as a decimal character)
		2	–	Active tool axis 0 = X 6 = U 1 = Y 7 = V 2 = Z 8 = W
		3	–	Spindle speed S
		4	–	Oversize for tool length DL
		5	–	Tool radius oversize DR
		6	–	Automatic TOOL CALL 1 = Yes, 0 = No
		7	–	Tool radius oversize DR2
		8	–	Tool index

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
Tool-change sequence				
	61	0	Tool no.	-1: Sequence cannot be ascertained 0: Tool already in the spindle 1: Manual tool → manual tool 2: Normal tool → manual tool 3: Special tool → manual tool 4: T0 → manual tool 5: Manual tool → normal tool 6: Normal tool → normal tool 7: Special tool → normal tool 8: T0 → normal tool 9: Manual tool → special tool 10: Normal tool → special tool 11: Special tool → special tool 12: T0 → special tool 13: Manual tool → T0 14: Normal tool → T0 15: Special tool → T0
Position programmed after TOOL CALL				
	70	1	–	1 = valid position
		2	1	Position of X axis
			2	Position of Y axis
			3	Position of Z axis
		3	–	Feed rate (–1 = no feed rate programmed)
Tool compensation				
	200	1	–	Active radius (including oversizes) with algebraic sign
		2	–	Active length (including oversizes)

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum	
Coordinate transformations					
210	1	1	–	Basic rotation (manual)	
		2	–	Programmed rotation	
		3	–	Active mirror axis Bits 0 to 2 and 6 to 8: Axes X, Y, Z and U, V, W	
	4	1	1	1	Active scaling factor in X
			2	2	Active scaling factor in Y
			3	3	Active scaling factor in Z
			7	7	Active scaling factor in U
			8	8	Active scaling factor in V
			9	9	Active scaling factor in W
	5	1	1	1	3-D ROT A
			2	2	3-D ROT B
			3	3	3-D ROT C
	6	–	1	–	Tilt working plane in Program Run mode (0 = inactive, –1 = active)
			7	–	Tilt working plane in Manual mode (0 = inactive, –1 = active)
			8	–	Angle of misalignment between spindle and tilted coordinate system
			8	–	Angle of misalignment between spindle and tilted coordinate system
	214	8	8	–	Tolerance programmed with cycle 32 or MP1096
220	2	1 to 9 (X, Y, Z, A, B, C, U, V, W)	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Current datum shift	
		3	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Difference between reference point and datum point	
		4	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Current PLC datum shift	
Traverse range					
230	2	1 to 9 (X, Y, Z, A, B, C, U, V, W)	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Negative software limit switches	
		3	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Positive software limit switches	

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
Nominal position in the REF system				
	240	1	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Current NOMINAL position of the axis in the REF system
		8	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Current ACTUAL position of the axis in the REF system
Current position in the active coordinate system				
	270	1	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Note: Does not apply to axes that are only displayed. Solution only possible via PLC or FN 20: WAIT FOR SYNC FN 18: SYSREAD Q<xxx> = ID2000 ...
M128 active				
	280	1	-	-1 = M128 (TCPM) active, 0 = M128 (TCPM) not active
		2	-	Feed rate programmed with M128
Tilting and swivel axes				
	290	1	-	Rows of the active kinematics table
		2	Number of the bit	Values of the individual bits of the active MP7500 (kinematics table or machine parameters)
M144 active				
	310	144	-	-1 = M144 active 0 = M144 not active
		116	-	<> 0 = M116 active 0 = M116 not active
		128	-	<> 0 = M128 active 0 = M128 inactive
System time on the control				
	320	1	0	Current system time on the control
TS touch trigger probe				
	350	10	-	Tool axis
		11	-	Effective radius
		12	-	Effective length
		13	-	Calibration ring radius
		14	1	Center offset (reference axis)
			2	Center offset (minor axis)
		15	-	Direction of center offset with respect to spindle 0°

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum	
TT touch probe for tool measurement					
	350	20	1	Center in 1st axis	
			2	Center in 2nd axis	
			3	Center in 3rd axis	
	21		–	Effective radius	
	22		1	Probing position 1 in axis X	
			2	Probing position 1 in axis Y	
			3	Probing position 1 in axis Z	
	23		1	Probing position 2 in axis X	
			2	Probing position 2 in axis Y	
			3	Probing position 2 in axis Z	
	24		1	Probing position 3 in axis X	
			2	Probing position 3 in axis Y	
			3	Probing position 3 in axis Z	
	25		1	Probing position 4 in axis X	
			2	Probing position 4 in axis Y	
			3	Probing position 4 in axis Z	
Datum from touch probe cycle					
	360	1	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Last datum of a manual touch probe cycle, or last touch point from Cycle 0 without compensation of stylus length, but with compensation of stylus radius (workpiece coordinate system)	
			2	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Last reference point of a manual touch probe cycle, or last touch point from Cycle 0 without stylus length or stylus radius compensation (machine coordinate system)
			3	–	Measurement result of touch probe cycles 0 and 1 without probe radius or length compensation
	370	6	–	Measurement result on the screen	
Read values from active datum table					
	500	Line	Column	Read ACTUAL values (MP7475 = 0)	
	501	Line	Column	Read REF values (MP7475 = 1)	
	505	1	–	0 = no datum table selected 1 = datum table selected	

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
Read values from the active preset table				
	502	Line	Column	Read value with conversion in the currently active coordinate system from the preset table
	503	Line	Column	Read value from the preset table without conversion to the currently active coordinate system
	504	Line	–	Read basic rotation from the preset table
Values from the active pallet table				
	510	1	–	Active row (-1 = no pallet table active)
		2	–	Pallet number from column Name
		3	–	Current row of the pallet table
		4	–	Last line of the NC program of the current pallet
		5	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Tool-oriented machining 0 = safety height not programmed 1 = safety height programmed
		6	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Programmed clearance height in pallet table for tool-oriented machining
Preset table				
	530	1	–	Active row
		2	Row in the preset table	0 = row is not write-protected ≠ 0 = row is write-protected
Pallet preset table				
	540	1	–	Active row

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
Read machine statuses				
	590	1	1 to 9	"Variables" 1 to 9 can only be written with FN17: SYSWRITE ID590 NR1 . Deletion depends on MP7300.
	590	2	1 to 9 (X, Y, Z, A, B, C, U, V, W)	The "variables" for ID621. Independent of MP7300.
	590	3	1 to 9 (X, Y, Z, A, B, C, U, V, W)	The "variables" for ID621. Independent of a power failure.
	621	0	1 to 9 (X, Y, Z, A, B, C, U, V, W)	Start measurement of the dynamic load.
	630	0	1 to 175	You can explicitly determine whether the SIK option given in IDX is set or not. 1 = option is enabled 0 = option is not enabled
	630	1	–	It can be determined whether a Feature Content Level (for upgrade functions) is set, and which one. –1 = no FCL is set <No.> = FCL that is set
	630	2	–	Serial number of the SIK –1 = no valid SIK in the system

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
Touch probe cycles				
	990	1	–	Approach behavior 0 = standard behavior 1 = effective radius, set-up clearance is zero
		2	10	0.0 = execution not in block scan 1.0 = execution in block scan –1.0 = invalid index
			16	0.0 = execution not in Automatic operating mode 1.0 = execution in Automatic operating mode –1.0 = invalid index
		4	–	0.0 = spindle or 1.0 = touch probe deflected
		10	<no.>	Tool number that belongs to the tool name of Q parameter IDX <no.>
Coordinate transformation				
		3	No. of the 1st of 9 consecutive Q parameters for axes X, Y, Z, A, B, C, U, V, W	Number of axes that are programmed in the selected datum table
		8	–	Current spindle angle
Machine parameters				
	1000	MP number	MP index	Value of the machine parameter (even with MP1054.x and MP7530.x, if they don't have a formula). The entered values of non-active MPs are also supplied.
		410	3 or 4	ASCII value of the axis designation from MP410.3 or MP410.4
	1010	MP number	MP index	0 = MP does not exist 1 = MP exists

Group name	Group number ID....	System data number NR....	System data index IDX....	System datum
PLC data (block transfer of up to 8 variables possible)				
Example "Reading PLC markers to Q parameters":				
FN 18: SYSREAD BLOCK Q1620 - Q1627 = ID2000 NR10 IDX880				
	2000	10	Marker no.	PLC marker
		20	Input no.	PLC input
		30	Output no.	PLC output
		40	Counter no.	PLC counter
		50	Timer no.	PLC timer
		60	Byte no.	PLC byte
		70	Word no.	PLC word
		80	Double-word no.	PLC double word
		90	Input byte no.	Profibus operand input byte
		100	Input word no.	Profibus operand input word
		110	Input double word no.	Profibus operand input double word
		120	Output byte no.	Profibus operand output byte
		130	Output word no.	Profibus operand output word
		140	Output double word no.	Profibus operand output double word

9.5.5 Data transfer of machine parameters > PLC

Up to 192 machine parameters are reserved for data transfer to the PLC. The iTNC saves the contents of MP4210.x, MP4220.x and MP4310.x in PLC words. You must call the contents of MP4230.x and MP4231.x by using Module 9032. In these machine parameters you can save, for example, values for PLC positioning or datum shifts, feed rates for PLC positioning or coding for the release of certain PLC functions. The number of indexes for MP4230.x can be increased to a maximum of 99 with the **NUMBERMP4230 =** entry in OEM.SYS. No entry or an invalid entry defines 32 indexes for MP4230.x. You must evaluate the transferred numerical values in your PLC program. The iTNC internally rounds input values less than 0.001 mm (or °) to 0.001 mm (or °).

		Set	Reset
D768	Value from MP4210.0	NC	NC
D772	Value from MP4210.1	NC	NC
D776	Value from MP4210.2	NC	NC
D780	Value from MP4210.3	NC	NC
D784	Value from MP4210.4	NC	NC
D788	Value from MP4210.5	NC	NC
D792	Value from MP4210.6	NC	NC
D796	Value from MP4210.7	NC	NC
D800	Value from MP4210.8	NC	NC
D804	Value from MP4210.9	NC	NC
D808	Value from MP4210.10	NC	NC
D812	Value from MP4210.11	NC	NC
D816	Value from MP4210.12	NC	NC
D820	Value from MP4210.13	NC	NC
D824	Value from MP4210.14	NC	NC
D828	Value from MP4210.15	NC	NC
D832	Value from MP4210.16	NC	NC
D836	Value from MP4210.17	NC	NC
D840	Value from MP4210.18	NC	NC
D844	Value from MP4210.19	NC	NC
D848	Value from MP4210.20	NC	NC
D852	Value from MP4210.21	NC	NC
D856	Value from MP4210.22	NC	NC
D860	Value from MP4210.23	NC	NC
D864	Value from MP4210.24	NC	NC
D868	Value from MP4210.25	NC	NC
D872	Value from MP4210.26	NC	NC
D876	Value from MP4210.27	NC	NC

		Set	Reset
D880	Value from MP4210.28	NC	NC
D884	Value from MP4210.29	NC	NC
D888	Value from MP4210.30	NC	NC
D892	Value from MP4210.31	NC	NC
D896	Value from MP4210.32	NC	NC
D900	Value from MP4210.33	NC	NC
D904	Value from MP4210.34	NC	NC
D908	Value from MP4210.35	NC	NC
D912	Value from MP4210.36	NC	NC
D916	Value from MP4210.37	NC	NC
D920	Value from MP4210.38	NC	NC
D924	Value from MP4210.39	NC	NC
D928	Value from MP4210.40	NC	NC
D932	Value from MP4210.41	NC	NC
D936	Value from MP4210.42	NC	NC
D940	Value from MP4210.43	NC	NC
D944	Value from MP4210.44	NC	NC
D948	Value from MP4210.45	NC	NC
D952	Value from MP4210.46	NC	NC
D956	Value from MP4210.47	NC	NC
W960	Value from MP4220.0	NC	NC
W962	Value from MP4220.1	NC	NC
W964	Value from MP4220.2	NC	NC
W966	Value from MP4220.3	NC	NC
W968	Value from MP4220.4	NC	NC
W976	Value from MP4310.0	NC	NC
W978	Value from MP4310.1	NC	NC
W980	Value from MP4310.2	NC	NC
W982	Value from MP4310.3	NC	NC
W984	Value from MP4310.4	NC	NC
W986	Value from MP4310.5	NC	NC
W988	Value from MP4310.6	NC	NC
W990	Value from MP4310.7	NC	NC
W992	Value from MP4310.8	NC	NC
W994	Value from MP4310.9	NC	NC
M4300 - M4315	Value from MP4310.0	NC	NC
M4316 - M4331	Value from MP4310.1	NC	NC
M4332 - M4347	Value from MP4310.2	NC	NC
M4348 - M4363	Value from MP4310.3	NC	NC
M4364 - M4379	Value from MP4310.4	NC	NC
M4380 - M4395	Value from MP4310.5	NC	NC
M4396 - M4411	Value from MP4310.6	NC	NC
M4412 - M4427	Value from MP4310.7	NC	NC
M4428 - M4443	Value from MP4310.8	NC	NC
M4444 - M4459	Value from MP4310.9	NC	NC



MP4210.0-47 Setting a number in the PLC (D768 to D956)

Input: -99 999.9999 to +99 999.9999

MP4220.0-4 Setting a number in the PLC (W960 to W968)

Input: 10 to 30 000

MP4230.0-31 Setting a number in the PLC (Module 9032)**The number of indexes can be increased via an entry in OEM.SYS.**

Input: -99 999.9999 to +99 999.9999

MP4231.0-31 Setting a number in the PLC (Module 9032)

Input: -99 999.9999 to +99 999.9999

MP4310.0-9 Setting a number in the PLC (W976 to W994, M4300 to M4459)

Format: Number, \$xxxx [Hex], %xxxxxxxxxxxxxxxxxxxx [Bin]

Input: 0 to 65 535

Module 9032 Read machine parameters

With this module you can read the value of the given machine parameter from the active machine parameter file. The input value is transferred as a natural number with the decimal point shifted by the number of possible decimal places.

Only the value from the editable machine parameter file is read, not any value modified in the run-time memory by PLC Module 9031.

For non-indexed machine parameters, zero must be transferred as the index.

Call only in a submit job.

Call:

PS B/W/D/K <>MP number>

PS B/W/D/K <>MP index>

CM 9032

PL B/W/D <>MP value / Error code>

1: MP number does not exist

2: No separator (:)

3: MP value out of range

4: MP not found in file

5: No MP file found

6: Call was not in a submit job.

7: MP is of the "string" type

8: No system memory

9.5.6 Interrogate PLC operands in the NC program (FN20: WAIT FOR)

With **FN20: WAIT FOR** you can interrupt the NC program until the condition programmed in the FN20 block is fulfilled. These conditions can be comparisons of a PLC operand with a constant. Permitted PLC operands: M, B, W, D, T, C, I, O and their symbolic operands

Operator	Function
==	Equal to
!= or <>	Not equal to
<	Less than
>	Greater than
<=	Less than or equal to
>=	Greater than or equal to

If you enter no condition, the interruption will continue until the operand = 0.

Examples:

FN20: WAIT FOR I10==1

Continue the NC program if PLC input I10 is set.

FN20: WAIT FOR I10

Continue the NC program if PLC input I10 equals zero.

FN20: WAIT FOR B3000>250

Continue the NC program, if the content of B3000 is greater than 250.

FN20: WAIT FOR is processed in look-ahead, i.e. if a synchronization with real time is necessary, then **FN20: WAIT FOR SYNC** must be programmed in the preceding NC block. Look-ahead is then stopped, and **FN20: WAIT FOR** is not performed until this block is actually reached in the NC program.

Examples with symbolic operands:

FN 20: WAIT FOR TASTENCODE == 3 instead of

FN 20: WAIT FOR W274 == 3

Relational operands can also be symbolic. Example:

FN 20: WAIT FOR MARKER_A == MARKER_B



9.5.7 Data transfer PLC operand <=> NC program

READ FROM PLC KEY

With the **READ FROM PLC KEY** plain-language command, the value of a symbolic PLC operand or an absolute address of the currently active PLC program can be read from within the NC program. The read value is stored in the transferred Q parameter.

Proceed as follows to enter the **READ FROM PLC KEY** function on the iTNC 530 HSCI:

- ▶ Select the **Programming** mode of operation
- ▶ Select an existing NC program in the file manager or create a new NC program.
- ▶ Enter the MOD code number 555343.
- ▶ Press the SPEC FCT key.
- ▶ Press the PROGRAM FUNCTIONS soft key.
- ▶ Press the PLC FUNCTIONS soft key (fourth level).
- ▶ Press the READ FROM PLC soft key.

Command structure:

READ FROM PLC KEY [source] TO [target]

Permissible entries for **[source]**:

- Symbolic PLC operand or absolute address directly as string, e.g. "KG_kinematics_active" or "W20"
- Q string as the source of the operand's name, e.g. QS10

Permissible entries for **[target]**:

- Q parameter, e.g.:**Q10**
- Q parameter reference, e.g.:**Q REF Q10**
- Q string, e.g.:**QS10**
- Q string reference, e.g.:**QS REF Q10**
- QL and QR parameters, e.g.:**QL499**

The type of the Q parameter specified in **[target]** must match the interrogated PLC operand. The control will otherwise issue the error message **Access to PLC variable has failed.**



Note

The **READ FROM PLC KEY** command is run in look-ahead, meaning that the control executes the command immediately instead of waiting until the NC program has been executed up to the corresponding point. If a synchronization with real time is necessary, then **FN20: WAIT FOR SYNC** must be programmed in the preceding NC block. Look-ahead is then stopped, and **READ FROM PLC KEY** is not performed until this block is actually reached in the NC program.



Note

With the **FN17: SYSWRITE ID13 NR3** function, you can define a label to which the control will jump if an error occurs executing the command. The **Access to PLC variable has failed** error message will be suppressed in this case.

Examples:

READ FROM PLC KEY "KG_kinematics_active" TO Q102

Reads the value of the **KG_kinematics_active** constant and stores the read value in Q102.

READ FROM PLC KEY "WG_ParSet_active" TO Q100

Reads the value of the **WG_ParSet_active** symbolic PLC operand and stores the read value in Q100.

READ FROM PLC KEY "ACHSEN::axis" TO Q105

Reads the value of the **axis** static PLC operand (= locally defined variable) from the **ACHSEN.SRC** module and stores the value in Q105.

READ FROM PLC KEY "Test_Base\Test8001::ML_Test_RUNNING" TO Q106

Reads the value of the **ML_Test_RUNNING** static PLC operand (= locally defined variable) from the **Test_Base\Test8001.SRC** module and stores the read value in Q106.

WRITE TO PLC KEY

With the **WRITE TO PLC KEY** plain-language command, you can write values to symbolic PLC operands or an absolute address from within the NC program.

Proceed as follows to enter the **WRITE TO PLC KEY** function on the iTNC 530 HSCI:

- ▶ Select the **Programming** mode of operation
- ▶ Select an existing NC program in the file manager or create a new NC program.
- ▶ Enter the MOD code number 555343.
- ▶ Press the SPEC FCT key.
- ▶ Press the PROGRAM FUNCTIONS soft key.
- ▶ Press the PLC FUNCTIONS soft key (fourth level).
- ▶ Press the WRITE TO PLC soft key.

Command structure:

WRITE TO PLC KEY [target] [value]

Permissible entries for **[target]**:

- Symbolic PLC operand or absolute address directly as string, e.g. "MG_kinematics_activate" or "W20"
- Q string as the source of the operand's name, e.g. QS10



Permissible entries for **[value]**:

- Q parameter, e.g.:=Q10
- Undefined:SET UNDEFINED
- Q string, e.g.:=QS10
- Numerical value directly, e.g.:=10
- String directly, e.g.:="new value"

If you specify a Q parameter in **[value]**, it must match the interrogated PLC operand. The control will otherwise issue the error message **Access to PLC variable has failed**.



Note

The **WRITE TO PLC KEY** command is run in look-ahead, meaning that the control executes the command immediately instead of waiting until the NC program has been executed up to the corresponding point. If a synchronization with real time is necessary, then **FN20: WAIT FOR SYNC** must be programmed in the preceding NC block. Look-ahead is then stopped, and **WRITE TO PLC KEY** is not performed until this block is actually reached in the NC program.



Note

With the **FN17: SYSWRITE ID13 NR3** function, you can define a label to which the control will jump if an error occurs executing the command. The **Access to PLC variable has failed** error message will be suppressed in this case.

Examples:

WRITE TO PLC KEY "WG_ParSet_number" =+2

Writes the value 2 to WG_ParSet_number, e.g. for selecting an axis parameter set

WRITE TO PLC KEY "MG_ParSet_activation" =+Q100

Writes the value from Q100 to the MG_ParSet_activation PLC marker, e.g. for activating an axis parameter set

9.6 Operands

9.6.1 Overview of operands

Operand	Abbreviation	Address range
Marker	M (marker)	<p>M0 to M9999</p> <p>M0 to M999 are free; they are deleted only after entering the code number 531210, not during a reset (nonvolatile range); the range can be reduced in the *.CFG file of the PLC compiler.</p> <p>M1000 to M3999 free, are deleted upon reset M4000 to M5999 reserved for NC/PLC interface (M4800 to M4999 are deleted before the first run of the PLC program, e.g. after compilation or restarting).</p> <p>M6000 to M9999 are free; they are deleted during reset.</p>
Input	I (input)	<p>I0 to I31 (MC 42x(B,C))</p> <p>I128 to I152 (machine operating panel)</p> <p>I64 to I127 (first PL)</p> <p>I192 to I255 (second PL)</p> <p>I256 to I319 (third PL)</p> <p>I320 to I383 (fourth PL)</p>
Output	O (output)	<p>O0 to O30 (MC 42x(B,C))</p> <p>O0 to O7 (via machine operating panel)</p> <p>O32 to O62 (first PL)</p> <p>O64 to O94 (second PL)</p> <p>O128 to O158 (third PL)</p> <p>O160 to O190 (fourth PL)</p>
Counters	C (counter)	<p>Set counter: C0 to C47</p> <p>Counter contents: C48 to C95</p> <p>Counter pulse release: C96 to C143</p>
Timer	T (timer)	<p>Timer start: T0 to T47</p> <p>Timer is running: T48 to T95 and T96 to T999</p>
Byte	B (byte)	B0 to B9999 (8 bits)
Word	W (word)	<p>B0 to B255 are free; depending on the definition in the *.CFG file of the PLC compiler, the defined range is deleted only after entering the code number 531210, not during reset (nonvolatile range). If no range is defined in the *.CFG file, B0 to B127 is the nonvolatile range.</p> <p>B256 to B2047 are reserved for NC/PLC interface.</p> <p>B2048 to B9999 are free. They are deleted by a reset.</p>
Double word	D (double word)	
Constant	K	-2 147 483 647 to +2 147 483 647
String	S	S0 to S99

Module 9075 Write to symbolic PLC operand

PLC Module 9075 writes to a symbolic PLC operand whose name is contained in a PLC string.

Transfer:

- String with symbolic operand name
- Data word or, if it is a string memory, address of the target string.

Returns:

- Identifier of operand type, 0 in case of error

Constraints:

- The module can only be called from a SUBMIT/SPAWN job.

Call:

PS B/W/D/K/S<>Address of string buffer>
or constant string, e.g. PS S"DG_MP_DATA[68]"

PS B/W/D/K <>Address of the source string>
If the string was addressed, this is the address of the source string, otherwise of the source data

CM 9075

PL B/W/D <>Type of operand>
0: PLCNONE in case of error
1: PLCMARKER
2: PLCTIMER
3: PLCCOUNTER
4: PLCINPUT
5: PLCOUTPUT
6: PLCBYTE
7: PLCWORD
8: PLCDOUBLE
9: PLCIBYTE
10: PLCIWORD
11: PLCIDDOUBLE
12: PLCOBYTE
13: PLCOWORD
14: PLCODOUBLE
15: PLCSTRING

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022 (also see return values of the module)
W1022	2	Operand type is invalid or does not exist
	11	Invalid address was programmed
	20	The module was not called in a SUBMIT/SPAWN job.

Module 9076 Read symbolic PLC operand

PLC Module 9076 reads a symbolic PLC operand whose name is contained in a PLC string.

Transfer:

- String with symbolic operand name
- If it is a string memory, the second parameter is the address of the target string.

Returns:

- Identifier of operand type, 0 in case of error
- Data from PLC operand, if B/W/D/M/I/O/C/T type

Constraints:

- The module can only be called from a SUBMIT/SPAWN job.

Call:

PS B/W/D/K/S<>Address of string buffer>
or constant string, e.g. PS S"DG_MP_DATA[68]"

PS B/W/D/K <>Address of the source string>
If the string was addressed, this is the address of the target string.

CM 9076

PL B/W/D <>Type of operand>
0: PLCNONE in case of error
1: PLCMARKER
2: PLCTIMER
3: PLCCOUNTER
4: PLCINPUT
5: PLCOUTPUT
6: PLCBYTE
7: PLCWORD
8: PLCDOUBLE
9: PLCIBYTE
10: PLCIWORD
11: PLCIDDOUBLE
12: PLCOBYTE
13: PLCOWORD
14: PLCODOUBLE
15: PLCSTRING

PL B/W/D <>Data>
Data from PLC operand, if B/W/D/M/I/O/C/T type

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Error code in W1022 (also see return values of the module)
W1022	2	Operand type is invalid or does not exist
	11	Invalid address was programmed
	20	The module was not called in a SUBMIT/SPAWN job



9.6.2 Operand addressing (byte, word and double word)

The memory for operands B (8 bits), W (16 bits), and D (32 bits) is only 8 bits wide. Since the operands can be 8, 16 or 32 bits wide, an overlap of the memory areas will occur, which you must take into account when addressing the memory.

Double word	Word	Byte	Memory	Word address	Double-word address	
D0	W2	B3	8 bits	High byte	Highest byte	
		B2	8 bits	Low byte		
	W0	B1	8 bits	High byte		Lowest byte
		B0	8 bits	Low byte		
D4	W6	B7	8 bits	High byte	Highest byte	
		B6	8 bits	Low byte		
	W4	B5	8 bits			
		B4	8 bits			
•	•	•	•	•	•	
•	•	•	•	•	•	
•	•	•	•	•	•	
D9996	W9998	B9999	8 bits	High byte	Highest byte	
		B9998	8 bits	Low byte		
	W9996	B9997	8 bits	High byte		Lowest byte
		B9996	8 bits	Low byte		

For byte addressing, every address is accessible; for word addressing, every second address; and for double word addressing, every fourth from 0 to 9996. The address parameter indicates the low byte of the word address (W) and the lowest byte of the double-word address (D).

Markers, timers and counters are addressed with the corresponding code letters M, T or C followed by the operand number (e.g. M500, T7, C18).

9.6.3 Timers

The PLC has 952 timers, which you control through special markers with the symbol T. You define the run time of the timers T0 to T47 in MP4110.x, and the run time of timers T96 to T999 in MP4111.x. MP4111.x is defined by entering the keyword **NUMBERMP4111** = followed by the required number of timers in the OEM.SYS file. The unit of time (input value 1 in MP4110.x and MP4111.x) is seconds.

You can start the first 48 timers by setting one of the timers T0 to T47 for at most one PLC scan (otherwise the iTNC restarts the timer with the negative edge for each additional scan). The iTNC reserves the timer with the duration defined in MP4110.x, and sets the markers T48 to T95 (timer is running) until the defined duration has expired.

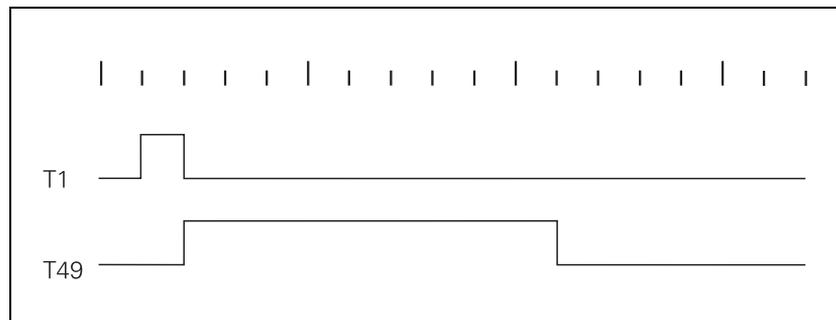
You can also set and start timers T0 to T47 with Module 9006. Timers T96 to T999 can only be started through Module 9006.

Cyclic timers (> T96) can be defined and started with Module 9197. They are reset for one PLC cycle, and are then restarted automatically.

Example:

Start of timer 1

Run time in MP4110.1 = 9 (PLC cycles)



Start timer	Timer is running	Machine parameters
T0	T48	MP4110.0
T1	T49	MP4110.1
T2	T50	MP4110.2
T3	T51	MP4110.3
T4	T52	MP4110.4
T5	T53	MP4110.5
T6	T54	MP4110.6
T7	T55	MP4110.7
T8	T56	MP4110.8
T9	T57	MP4110.9
T10	T58	MP4110.10
T11	T59	MP4110.11

Start timer	Timer is running	Machine parameters
T12	T60	MP4110.12
T13	T61	MP4110.13
T14	T62	MP4110.14
T15	T63	MP4110.15
T16	T64	MP4110.16
T17	T65	MP4110.17
T18	T66	MP4110.18
T19	T67	MP4110.19
T20	T68	MP4110.20
T21	T69	MP4110.21
T22	T70	MP4110.22
T23	T71	MP4110.23
T24	T72	MP4110.24
T25	T73	MP4110.25
T26	T74	MP4110.26
T27	T75	MP4110.27
T28	T76	MP4110.28
T29	T77	MP4110.29
T30	T78	MP4110.30
T31	T79	MP4110.31
T32	T80	MP4110.32
T33	T81	MP4110.33
T34	T82	MP4110.34
T35	T83	MP4110.35
T36	T84	MP4110.36
T37	T85	MP4110.37
T38	T86	MP4110.38
T39	T87	MP4110.39
T40	T88	MP4110.40
T41	T89	MP4110.41
T42	T90	MP4110.42
T43	T91	MP4110.43
T44	T92	MP4110.44
T45	T93	MP4110.45
T46	T94	MP4110.46
T47	T95	MP4110.47

MP4110.0-47 Run time PLC timer T0 to T47

Input: 0 to 1 000 000.000 [s]

MP4111.96-x Run time PLC timer T96 to x (defined in OEM.SYS)

Input: 0 to 1 000 000.000 [s]

Module 9006: Set and start PLC timer

Use Module 9006 to define the run time for a PLC timer and start the timer.
Constraints:

- If during a PLC scan a timer from T0 to T47 is set in the PLC program, and the same timer is activated through Module 9006, then the direct activation through T0 to T47 has priority regardless of whether the module is called before or after setting T0 to T47.
- Immediately after the module call, one of the markers T48 to T96 is set. T0 to T47 are not set.
- The iTNC rounds the actual run time to integral PLC cycle times.
- Cancel run time: Reset timers T48 to T999.

Call:

```
PS   B/W/D/K <>Timer number>
      Input value: 0 to 999
PS   B/W/D/K <>Run time>
      0 to 1 000 000 000 [ms]
      -1: Run time from MP4110.x or MP4111.x

CM   9006
```

Error code:

Marker	Value	Meaning
M4203	0	Timer started
	1	Error. See W1022.
W1022	1	Invalid timer number or excessive run time
	2	Timer already assigned for cyclic timer
	3	Timer is started as cyclic timer (Module 9197)

Module 9197 Start cyclic timer

Module 9197 can define and start a timer > T96 as cyclic timer. After expiration of the defined time, the timer is reset for a PLC cycle and afterwards is automatically restarted.

- Stop timer: Transfer run time 0
- The iTNC rounds the actual run time to integral PLC cycle times.

Call:

```
PS   B/W/D/K <>Timer number>
      96 to 999
PS   B/W/D/K <>Run time>
      0 to 1 000 000 000 [ms]
      -1: Run time from MP4111.x

CM   9197
```

Error code:

Marker	Value	Meaning
M4203	0	Timer started
	1	Error. See W1022.
W1022	1	Excessive run time
	3	Invalid timer number



9.6.4 Counters

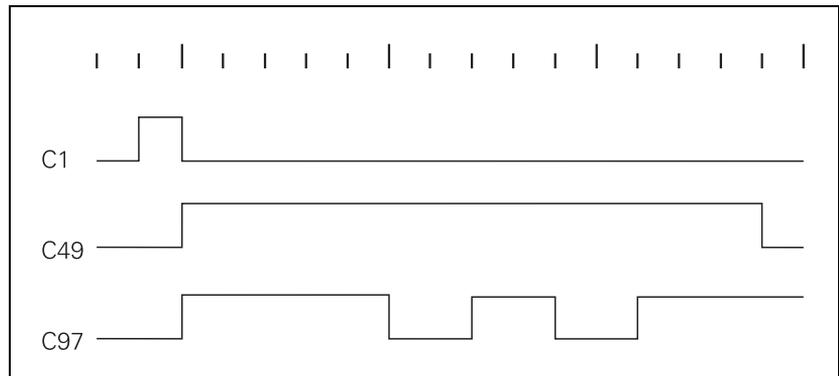
The PLC has 48 counters, which you control through special markers with the symbol C. After you have set a marker from the C0 to C47 range, the iTNC loads the counter with the value that is saved in machine parameter MP4120.x. The marker range C48 to C95 indicates whether the counter has expired. With markers C96 to C144 you can start and stop the counter.

MP4020 bit 11 defines whether the counter is defined in PLC cycles or seconds. In this way, the counters can also be used as timers. With this definition of counters in PLC cycles, the decimal places are not evaluated by MP4120.x.

Example:

Logic diagram for counter C1

Preset value in MP4120.1 = 10 (PLC cycles or seconds)



Set counter	Counter is running	Start counter	Machine parameters
C0	C48	C96	MP4120.0
C1	C49	C97	MP4120.1
C2	C50	C98	MP4120.2
C3	C51	C99	MP4120.3
C4	C52	C100	MP4120.4
C5	C53	C101	MP4120.5
C6	C54	C102	MP4120.6
C7	C55	C103	MP4120.7
C8	C56	C104	MP4120.8
C9	C57	C105	MP4120.9
C10	C58	C106	MP4120.10
C11	C59	C107	MP4120.11
C12	C60	C108	MP4120.12
C13	C61	C109	MP4120.13
C14	C62	C110	MP4120.14

Set counter	Counter is running	Start counter	Machine parameters
C15	C63	C111	MP4120.15
C16	C64	C112	MP4120.16
C17	C65	C113	MP4120.17
C18	C66	C114	MP4120.18
C19	C67	C115	MP4120.19
C20	C68	C116	MP4120.20
C21	C69	C117	MP4120.21
C22	C70	C118	MP4120.22
C23	C71	C119	MP4120.23
C24	C72	C120	MP4120.24
C25	C73	C121	MP4120.25
C26	C74	C122	MP4120.26
C27	C75	C123	MP4120.27
C28	C76	C124	MP4120.28
C29	C77	C125	MP4120.29
C30	C78	C126	MP4120.30
C31	C79	C127	MP4120.31
C32	C80	C128	MP4120.32
C33	C81	C129	MP4120.33
C34	C82	C130	MP4120.34
C35	C83	C131	MP4120.35
C36	C84	C132	MP4120.36
C37	C85	C133	MP4120.37
C38	C86	C134	MP4120.38
C39	C87	C135	MP4120.39
C40	C88	C136	MP4120.40
C41	C89	C137	MP4120.41
C42	C90	C138	MP4120.42
C43	C91	C139	MP4120.43
C44	C92	C140	MP4120.44
C45	C93	C141	MP4120.45
C46	C94	C142	MP4120.46
C47	C95	C143	MP4120.47

MP4120.0-47 Preset value for PLC counters

Input: 0 to 1 000 000.000 [s or PLC cycles, depending on MP4020, bit 11]



9.6.5 Fast PLC inputs

Fast PLC inputs are possible on the UEC 1xx, UMC 1xx and slots of a PL 6xxx. Only the first four slots of a PL 6xx can be used, however. The fifth slot and the successive slots (on PL 6x06, PL 6x08) must not be defined as fast PLC inputs.

With MP4130 you can define PLC inputs that are not interrogated within the PLC cycle, but rather in the control loop cycle. Markers M4590 to M4593 show the current state of the fast PLC inputs.

You must activate the **fast PLC inputs** in the PLC program with W522 bit 2 to bit 5.

For the iTNC to identify with certainty a signal change, the signal duration at the fast PLC input must last a minimum of 4 ms.

MP4130 Numbers of fast PLC inputs

Input: 0 to 20 000 [no. of the PLC input]
-1: Function inactive

MP4130.2 Fast PLC input sets marker M4590
MP4130.3 Fast PLC input sets marker M4591
MP4130.4 Fast PLC input sets marker M4592
MP4130.5 Fast PLC input sets marker M4593

MP4131.2-5 Activation criterion for fast PLC inputs

Input: 0: Activate at LOW level
1: Activate at HIGH level

		Set	Reset
W522	Activate the high-speed PLC inputs Bit 2: Fast PLC input defined in MP4130.2 Bit 3: Fast PLC input defined in MP4130.3 Bit 4: Fast PLC input defined in MP4130.4 Bit 5: Fast PLC input defined in MP4130.5	PLC	PLC
M4590	Status of fast PLC input from MP4130.2	NC	PLC
M4591	Status of fast PLC input from MP4130.3	NC	PLC
M4592	Status of fast PLC input from MP4130.4	NC	PLC
M4593	Status fast PLC input from MP4130.5	NC	PLC

9.7 Program Creation



Attention

HEIDENHAIN would like to point out the following information for the processing of signals in the PLC:

- For all PLC markers, the status 0 (not set) must be used as the safe status. Only this way can the safety of a machine be ensured if the power fails.
- The following must be kept in mind for non-volatile markers:
Non-volatile markers, bytes, words and double words can also be deleted (e.g. if the buffer battery is empty). In order to detect this state, we recommend setting a flag marker in the non-volatile memory. Set this marker once, at a time when it is ensured that all signals in the PLC are valid.
This marker should be used as a reference in the PLC program. As long as this marker is set, the non-volatile markers have not been deleted (e.g. by an empty buffer battery or an error in the PLC). If this flag marker is deleted, all non-volatile markers are invalid. An appropriate safety reaction with an error message must occur if this happens.

In order to attain an optimum level of safety for your machine and the operator, HEIDENHAIN recommends the combined use of these safety measures.

In connection with this, a possibility for outputting the cause of the last PLC run-time error was introduced. PLC word W1002 can be used to determine the last PLC run-time error that led to the stop of PLC program execution.

If a PLC run-time error occurs, the generated error code is saved in W1002 and simultaneously in the non-volatile memory of the control. When a new PLC run-time error occurs, W1002 and the value in the non-volatile memory are overwritten.

If the control is restarted or a PLC program start is triggered in some other manner (by Compile, Restart PLC or acknowledgment of PLC run-time errors), the value in W1002 regenerates itself via the information in the non-volatile memory. Once the value has been entered in W1002 again, this information is deleted from the non-volatile memory. This way the last PLC run-time error is always available in W1002, even after the PLC is restarted. In addition, the error code can already be seen in the PLC table as soon as the **Power interrupted** status after the control has been restarted.



The error code in W1002 can be output and evaluated in the PLC program. The following applies to the evaluation of the error code, which should be performed during the first PLC run-through:

- W1002 = 0: No PLC run-time error occurred before the last PLC program start
- W1002 = 1: Error information in W1002 is invalid (e.g. because of an error in the hardware). If necessary: Implement safety reaction.
- W1002 ≥ 50: PLC run-time error with number code (see the PLCdesignNT help for the error name). If necessary: Implement safety reaction.

When creating service files or shutting down the control, the nonvolatile memory of the PLC program is automatically saved to the file PLC:\JHPLCLOG\PLCREMADATA.LOG. Restoring the nonvolatile area from the file is possible only by using the RESTORE M/B/W/D soft key in the PLC mode of operation under the TABLE soft key.

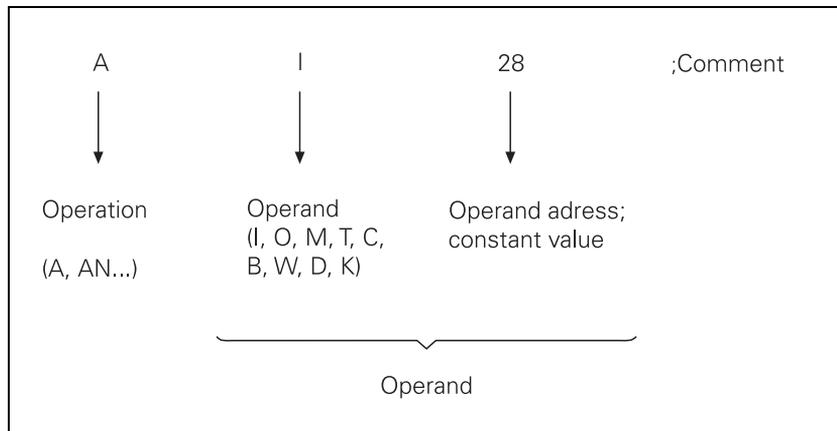
9.7.1 ASCII editor

With the integrated editor you can create the PLC program and all other necessary files right at the control through the ASCII keyboard. You will find a comprehensive description of the editor including its soft keys in the User's Manual of the control.

9.7.2 Program format

Command

A command is the smallest unit of a PLC program. It consists of the operation part and the operand part.



The operation describes the function to be executed. It defines how the operand is to be processed by the iTNC. The operand shows what is to be operated with. It consists of the operand abbreviation and a parameter (address). With the PLC commands you can combine (gate), delete and load register and memory contents, both with bit and word processing. For word processing, you can address memory contents with a length of 8 bits (byte), 16 bits (word) or 32 bits (double word).



9.7.3 Program structure

To make it easier to maintain and expand your PLC program, you should give it a modular structure. Modular means that you write a separate program module for each function. You can then call the individual modules from the main program. You should interrogate improper functioning of the machine in the PLC program and indicate such malfunctions on the screen with plain-language error messages.

Module 9019 Size of the processing stack

To debug functions, you can use Module 9019 to interrogate the contents of the processing stack. The function answers with the number of the bytes that lie on the processing stack of the PLC at the moment. If the processing stack is empty, the iTNC returns the value zero. A byte, word, double word or string occupies four bytes on the stack; a marker, input, output, timer or counter occupies two bytes.

Call:

CM 9019

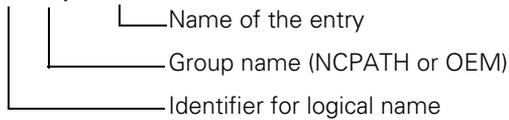
PL B/W/D <>Number of bytes on processing stack>



9.7.4 Logical names for files

Instead of a permanent file name, you can also enter logical names in PLC modules, such as **PS S">OEM.PLCMAIN"**.

Syntax: **>Group.name**



Examples

>NCPATH.NCEDIT:

The iTNC transfers the complete name and path of the file that is currently selected in the editing mode.

>OEM.PLCMAIN:

The iTNC transfers the complete name and path of the PLC program that was entered in the OEM.SYS file with the command **PLCMAIN**.

List of the logical names:

Group	Entry	Meaning
NCPATH		
	PLCEDIT	Selected file in the PLC Programming mode
	NCEDIT	Selected file in the Programming and Editing mode
	RUNPGM	Selected file in the Program Run mode
	RUNDATUM	Selected datum table in the Program Run mode
	SIMPGM	Selected file in the Test Run mode
	SIMDATUM	Selected datum table in the Program Test mode
	SIMTOOL	Selected tool table in the Program Test mode
	RUNBRKPGM	Target file in the block scan in the Program Run mode
	SIMBRKPGM	Target file in the block scan in the Program Test mode
	MDIPGM	Selected file in the Positioning with Manual Data Input operating mode
	TCHPATH	Selected datum table for manual probing
OEM		
	TABCMA	Active compensation table
	MODEHELP	Active help file
	PLCMAIN	Active PLC main program
	PLCPWM	Active PLC commissioning program for digital axes
	PLCEVENTS	Active event list for spawn command
	PLCERRTAB	Active PLC error message list (PET)
	WMAT	Active tool material file
	TMAT	Active workpiece material file
	MPPFILE	Active machine parameter list
	Your own entry	In the OEM.SYS file you can indicate the desired file names with path behind your own entry. For example: HUGO=TNC:\HUGO\TEST.H

9.8 Command Set

9.8.1 Overview

The following table provides an overview of all commands explained in this chapter:

Group of functions	Syntax	Function
Loading and saving commands		
	L	Load
	LN	Load NOT
	L-	Load two's complement
	LB	Load BYTE
	LW	Load WORD
	LD	Load DOUBLE WORD
	=	Assignment
	B=	Assign BYTE
	W=	Assign WORD
	D=	Assign DOUBLE WORD
	=N	Assign NOT
	=-	Assign two's complement
Setting commands		
	S	Set
	R	Reset
	SN	Set NOT
	RN	Reset NOT
Logical operations		
	A	And
	AN	And NOT
	O	or
	ON	Or NOT
	XO	Exclusive OR
	XON	Exclusive OR NOT
Arithmetic commands		
	+	Addition
	-	Subtraction
	x	Multiplication
	/	Division
	MOD	Remainder



Group of functions	Syntax	Function
Increment		
	INC	Increment operand
	INCW	Increment word accumulator
	INCX	Increment index register
Decrement		
	DEC	Decrement operand
	DECW	Decrement word accumulator
	DECX	Decrement index register
Comparisons		
	==	Equal to
	<	Less than
	>	Greater than
	<=	Less than or equal to
	>=	Greater than or equal to
	<>	Not equal to
Parenthetical expressions in logical operations		
	A[]	And []
	AN[]	And NOT []
	O[]	Or []
	ON[]	Or NOT []
	XO[]	Exclusive OR []
	XON[]	Exclusive OR NOT []
Parenthetical expressions with arithmetical instructions		
	+ []	Addition []
	- []	Subtraction []
	x []	Multiplication []
	/ []	Division []
	MOD []	Remainder []
Parenthetical expressions in comparisons		
	== []	Equal to []
	< []	Less than []
	> []	Greater than []
	<= []	Less than or equal to []
	>= []	Greater than or equal to []
	<> []	Not equal to []



Group of functions	Syntax	Function
Shifting commands		
	<<	Shift left
	>>	Shift right
Bit commands		
	BS	Bit set
	BC	Bit clear
	BT	Bit test
Stack operations		
	PS	Push data onto the data stack
	PL	Pull data from the data stack
	PSL	Push logic accumulator onto the data stack
	PSW	Push word accumulator onto the data stack
	PLL	Pull logic accumulator from the data stack
	PLW	Pull word accumulator from the data stack
Jump commands		
	JP	Unconditional jump
	JPT	Jump if logic accumulator = 1
	JPF	Jump if logic accumulator = 0
	CM	Call module
	CMT	Call module if logic accumulator = 1
	CMF	Call module if logic accumulator = 0
	EM	End of module, program end
	EMT	End of module if logic accumulator = 1
	EMF	End of module if logic accumulator = 0
	LBL	Label

9.8.2 LOAD (L)

Logic processing with the LOAD command

Syntax: L (LOAD)

Operands: M, I, O, T, C

Action:

Load the value of the addressed operand into the logic accumulator. Always use the L command at the beginning of a logic chain in order to be able to gate the operand in the following program sequence.

Example:

Gate the inputs I4 and I5 with AND, and assign the result to output O2.

Initial state:

Input I4 = 1

Input I5 = 0

Output O2 = ?

Function	STL	Logic accumulator	Operand content
Load the operand content into the logic accumulator.	L I4	Logic accumulator = 1	
Gate the content of the logic accumulator and input I5 with AND.	A I5		0
Assign the gating result to output O2.	= O2		0

**Word processing
with the LOAD
command**

Syntax: L (LOAD)

Operands: B, W, D, K

Action:

Load the value of the addressed operand, or of a constant, into the word accumulator. If necessary, the accumulator is supplemented with the correct algebraic sign. In contrast to logical operations, you must always begin a sequence of word gating operations with an L command. You cannot replace the L command with a logical gating instruction.

Example:

Gate a constant and byte B5 with AND, and assign the result to byte B8.

Initial state:

Constant 54 = 36 (hex)

Byte B5 = 2A (hex)

Output B8 = ?

Function	STL	Accumulator content	Operand content
Load the constant into the word accumulator.	L K+54	36	
Gate the contents of the word accumulator and byte B5 with AND.	A B5		2A
Assign the gating result to byte B8.	= B8		22



9.8.3 LOAD NOT (LN)

Logic processing with the LOAD NOT command

Syntax: LN (LOAD NOT)

Operands: M, I, O, T, C

Action:

Load the one's complement of the addressed operand into the logic accumulator. Always use the L command at the beginning of a logic chain in order to be able to gate the operand in the following program sequence.

Example:

Gate the inverted logical state of inputs I4 and I5 with AND, and assign the result to output O2.

Initial state:

Input I4 = 0

Input I5 = 1

Output O2 = ?

Function	STL	Accumulator content	Operand content
Load the inverted operand content into the logic accumulator.	LN I4	0	
Gate the content of the logic accumulator and input I5 with AND.	A I5		1
Assign the gating result to output O2.	= O2		1



**Word processing
with the LOAD NOT
command**

Syntax: LN (LOAD NOT)

Operands: B, W, D, K

Action:

Load the complement of the addressed operand, or of a constant, into the word accumulator. If necessary, the accumulator is supplemented with the correct algebraic sign. In contrast to logical operations, you must always begin a sequence of word gating operations with an L command. You cannot replace the L command with a logical gating instruction.

Example:

Gate the complement of byte B6 and byte B5 with AND, and assign the result to byte B8.

Initial state:

Byte B5 = 2A (hex)

Byte B6 = B6 (hex)

Byte B8 = ?

Function	STL	Accumulator content	Operand content
Invert byte B6, and load into the word accumulator	LN B6	2A	
Gate the contents of the word accumulator and byte B5 with AND.	A B5		B6
Assign the gating result to byte B8.	= B8		22



9.8.4 LOAD TWO'S COMPLEMENT (L-)

Syntax: L- (LOAD MINUS)

Operands: B, W, D, K

Action:

Load the two's complement of the addressed operand, or of a constant, into the word accumulator. If necessary, the iTNC fills the accumulator with the correct algebraic sign. The two's complement allows negative numbers to be stored, i.e., a number loaded with the L command appears in the accumulator with an inverted sign. This command can be used only with word processing.

Example:

Negate the content of byte B5 and then add it to the content of byte B6. Assign the result to byte B8.

Initial state:

Byte B5 = 15 (dec)

Byte B6 = 20 (dec)

Byte B8 = ?

Function	STL	Accumulator content	Operand content
Load byte B5 into the word accumulator, invert the algebraic sign.	L- B5	-15	+15
Add the contents of the word accumulator and byte B6.	+ B6	+5	+20
Assign the gating result to byte B8.	= B8	+5	+5



9.8.5 LOAD BYTE (LB)

Syntax: LB (LOAD BYTE)

Operands: M, I, O, T, C

Action:

Copy 8 markers, inputs, outputs, timers or counters with ascending numbering into the word accumulator. Each operand occupies one bit in the accumulator. The iTNC saves the entered operand address in the accumulator as LSB, the entered address +1 as LSB +1 etc. The last (8th) operand is now the MSB! If necessary, the iTNC fills the accumulator with the correct algebraic sign.

Example:

A pure-binary coded value is read through inputs I3 to I10 and saved in byte B8 in order to process it later.

Initial state:

Input	I3	= 1	Input	I7	= 0
Input	I4	= 1	Input	I8	= 1
Input	I5	= 1	Input	I9	= 1
Input	I6	= 0	Input	I10	= 0

Function	STL	Accumulator content	Operand content
		7 6 5 4 3 2 1 0	I10 I9 I8 I7 I6 I5 I4 I3
Load inputs I3 to I10 into the accumulator (bit 0 to bit 7).	LB I3	0 1 1 0 0 1 1 1	0 1 1 0 0 1 1 1
			7 6 5 4 3 2 1 0
Assign accumulator contents to byte 8.	= B8	0 1 1 0 0 1 1 1	0 1 1 0 0 1 1 1

9.8.6 LOAD WORD (LW)

Syntax: LW (LOAD WORD)

Operands: M, I, O, T, C

Action:

Copy 16 markers, inputs, outputs, timer or counters with ascending numbering into the word accumulator. Each operand occupies one bit in the accumulator. The iTNC saves the entered operand address in the accumulator as LSB, the entered address +1 as LSB +1 etc. The last (16th) operand is now the MSB! If necessary, the iTNC fills the accumulator with the correct algebraic sign.

Example:

See example command LB. Use command LW in the same way as LB. However, the iTNC processes 16 operands.



9.8.7 LOAD DOUBLE WORD (LD)

Syntax: LD (LOAD DOUBLE WORD)

Operands: M, I, O, T, C

Action:

Copy 32 markers, inputs, outputs, timers or counters with ascending numbering into the word accumulator. Each operand occupies one bit in the accumulator. The iTNC saves the entered operand address in the accumulator as LSB, the entered address +1 as LSB +1 etc. The last (32nd) operand is now the MSB! If necessary, the iTNC fills the accumulator with the correct algebraic sign.

Example:

See example command LB. Use command LD in the same way as LB. However, the iTNC processes 32 operands.

9.8.8 ASSIGN (=)

Logic processing with the ASSIGN command

Syntax: = (STORE)

Operands: M, I, O, T, C

Action:

Assign the content of the logic accumulator to the addressed operand. Use the = command only at the end of a sequence of logical gating operations in order to transfer a gating result to a logic operand. This command can be used several times in succession (see example).

Example:

Gate the inputs I4 and I5 with AND, and assign the result to outputs O2 and O5.

Initial state:

Input	I4	=	1
Input	I5	=	0
Output	O2	=	?
Output	O5	=	?

Function	STL	Accumulator content	Operand content
Load the operand content into the logic accumulator.	L I4	1	1
Gate the content of the logic accumulator and input I5 with AND.	A I5	0	0
Assign the gating result to output O2.	= O2	0	0
Assign the gating result to output O5.	= O5	0	0



Word processing with the ASSIGN command

Syntax: = (STORE)

Operands: B, W, D

Action:

Assign the content of the word accumulator to the addressed operand. Unlike bit processing, in word processing you can also use the = command within a sequence of word-gating operations. This command can be used several times in succession.

Example:

Gate a constant and byte B5 with AND, and assign the result to byte B8 and byte B10.

Initial state:

Constant 54 = 36 (hex)
Byte B5 = 2A (hex)
Byte B8 = ?
Byte B10 = ?

Function	STL	Accumulator content	Operand content
Load the constant into the word accumulator.	L K+54	36	
Assign the contents of the word accumulator to byte B8.	= B8	36	36
Gate the contents of the word accumulator and byte B5 with AND.	A B5	22	2A
Assign the gating result to byte B8.	= B8	22	22
Assign the gating result to byte B10.	= B10	22	22

9.8.9 ASSIGN BYTE (B=)

Syntax: B= (STORE BYTE)

Operands: M, I, O, T, C

Action:

Assign 8 bits from the word accumulator to markers, inputs, outputs, timers or counters with ascending numbering. Every bit occupies an operand. The iTNC assigns the LSB in the accumulator to the operand address specified in the command, the specified address +1 as LSB +1 etc. The last (8th) operand is assigned the MSB.

Example:

See example of command W=. Use command B= in the same way as W=. However, the iTNC processes 8 operands.



9.8.10 ASSIGN WORD (W=)

Syntax: W= (STORE WORD)

Operands: M, I, O, T, C

Action:

Assign 16 bits from the word accumulator to markers, inputs, outputs, timers or counters with ascending numbering. Every bit occupies an operand. The iTNC assigns the LSB in the accumulator to the operand address specified in the command, the specified address +1 as LSB +1 etc. The last (16th) operand is assigned the MSB.

Example:

Transfer a certain bit pattern, located in word W8, to the output addresses O1 to O16.

Initial state:

Word W8 = 36FF (hex)

Function	STL	Accumulator content	Operand content
Load content of word W8 into the word accumulator.	L W8	36FF	
			O16 .. O1
Assign accumulator content to outputs O1 to O16.	W= O1	36FF	0 0 1 1 0 1 1 0 1 1 1 1 1 1 1 1

9.8.11 ASSIGN DOUBLE WORD (D=)

Syntax: D= (STORE DOUBLE WORD)

Operands: M, I, O, T, C

Action:

Assign 32 bits from the word accumulator to markers, inputs, outputs, timers or counters with ascending numbering. Every bit occupies an operand. The iTNC assigns the LSB in the accumulator to the operand address specified in the command, the specified address +1 as LSB +1 etc. The last (32nd) operand is assigned the MSB.

Example:

See example of command W=. Use command D= in the same way as W=. However, the iTNC processes 32 operands.



9.8.12 ASSIGN NOT (=N)

Logic processing **Syntax:** =N (STORE NOT)

Operands: M, I, O, T, C

Action:

Assign the complement of the logic accumulator to the addressed operand.
For an example, see the ASSIGN (=) command.

Word processing **Syntax:** =N (STORE NOT)

Operands: B, W, D

Action:

Assign the complement of the word accumulator to the addressed operand.
For an example, see the ASSIGN (=) command.

9.8.13 ASSIGN TWO'S COMPLEMENT (=–)

Syntax: =– (STORE MINUS)

Operands: B, W, D

Action:

Assign the TWO'S COMPLEMENT of the word accumulator to the addressed operand. For an example, see the ASSIGN (=) command.



9.8.14 SET (S)

Syntax: S (SET)

Operands: M, I, O, T, C

Action:

If the logic accumulator = 1, then set the addressed operand to 1, otherwise do not change it. Use the S command at the end of a sequence of logical gating operations in order to influence an operand, depending on the result of gating. This command can be used several times in succession (see example).

Example:

Gate input I4 and I5 with OR. If the gating result is 1, then set output O2 and marker M500.

Initial state:

Input I4 = 1

Input I5 = 0

Output O2 = ?

Marker M500 = ?

Function	STL	Accumulator content	Operand content
Load the operand content into the logic accumulator.	L I4	1	1
Gate the content of the logic accumulator and input I5 with OR.	O I5	1	0
Since the result of the operation is 1, set output O2.	S O2	1	1
Since the result of the operation is 1, set marker M500.	S M500	1	1

9.8.15 RESET (R)

Syntax: R (RESET)

Operands: M, I, O, T, C

Action:

If the logic accumulator = 1, then set the addressed operand to 0, otherwise do not change it. Use the R command at the end of a sequence of logical gating operations in order to influence an operand, depending on the result of gating. This command can be used several times in succession (see example).

Example:

Gate input I4 and I5 with OR. If the gating result is 1, then reset output O2 and marker M500.

Initial state:

Input I4 = 1
Input I5 = 0
Output O2 = ?
Marker M500 = ?

Function	STL	Accumulator content	Operand content
Load the operand content into the logic accumulator.	L I4	1	1
Gate the content of the logic accumulator and input I5 with OR.	O I5	1	0
Since the result of the operation is 1, reset output O2.	R O2	1	0
Since the result of the operation is 1, reset marker M500.	R M500	1	0



9.8.16 SET NOT (SN)

Syntax: SN (SET NOT)

Operands: M, I, O, T, C

Action:

If the logic accumulator = 0, then set the addressed operand to 1, otherwise do not change it. Use the SN command at the end of a sequence of logical gating operations in order to influence an operand depending on the result of gating. This command can be used several times in succession (see example).

Example:

Gate input I4 and I5 with OR. If the gating result is 0, then set output O2 and marker M500.

Initial state:

Input I4 = 0

Input I5 = 0

Output O2 = ?

Marker M500 = ?

Function	STL	Accumulator content	Operand content
Load the operand content into the logic accumulator.	L I4	0	0
Gate the content of the logic accumulator and input I5 with OR.	O I5	0	0
Since the result of the operation is 0, set output O2.	SN O2	0	1
Since the result of the operation is 0, set marker M500.	SN M500	0	1



9.8.17 RESET NOT (RN)

Syntax: RN (RESET NOT)

Operands: M, I, O, T, C

Action:

If the logic accumulator = 0, then set the addressed operand to 0, otherwise do not change it. Use the RN command at the end of a sequence of logical gating operations in order to influence an operand depending on the result of gating. This command can be used several times in succession (see example).

Example:

Gate input I4 and I5 with OR. If the gating result is 0, then reset output O2 and marker M500.

Initial state:

Input I4 = 0

Input I5 = 0

Output O2 = ?

Marker M500 = ?

Function	STL	Accumulator content	Operand content
Load the operand content into the logic accumulator.	L I4	0	0
Gate the content of the logic accumulator and input I5 with OR.	O I5	0	0
Since the result of the operation is 0, reset output O2.	RN O2	0	0
Since the result of the operation is 0, reset marker M500.	RN M500	0	0



9.8.18 AND (A)

Logic processing with the AND command

Syntax: A (AND)

Operands: M, I, O, T, C

Action:

- At the beginning of a logic sequence, this command functions like an L command, i.e., the logical state of the operand is loaded into the logic accumulator. This is to ensure compatibility with the TNC 355, which does not have the special L command. In PLC programs, a sequence of logical gating operations should always be started with a load command (see L, LN, L-).
- Within a logic sequence, gate the content of the logic accumulator and the logical state of the operand with AND. The iTNC saves the result of the operation in the logic accumulator.

Example:

Gate the inputs I4 and I5 with AND, and assign the result to output O2.

Initial state:

Input I4 = 1
Input I5 = 0
Output O2 = ?

Function	STL	Accumulator content	Operand content
Load the operand content into the logic accumulator.	L I4	1	1
Gate the content of the logic accumulator and input I5 with AND.	A I5	0	1
Assign the gating result to output O2.	= O2	0	0

**Word processing
with the AND
command**

Syntax: A (AND)

Operands: B, W, D, K

Action:

Gate the contents of the word accumulator and the operand with AND. In accordance with the different data widths of the operands (B = 8 bits; W = 16 bits; D = K = 32 bits), 8, 16 or 32 bits, respectively, are influenced in the accumulator. Thus, bit 0 of the accumulator is gated with bit 0 of the operand, bit 1 of the accumulator with bit 1 of the operand, etc. The iTNC saves the result of the operation in the word accumulator.

Example:

Gate the contents of byte B5 and byte B6 with AND, and assign the result to byte B8.

Initial state:

Byte B5 = 2A (hex)

Byte B6 = 36 (hex)

Byte B8 = ?

Function	STL	Accumulator content	Operand content
Load byte B6 into the word accumulator.	L B6	2A	2A
Gate the contents of the word accumulator and byte B5 with AND.	A B5	22	36
Assign the gating result to byte B8.	= B8	22	22



9.8.19 AND NOT (AN)

Logic processing with the AND NOT command

Syntax: AN (AND NOT)

Operands: M, I, O, T, C

Action:

- At the beginning of a logic sequence, this command functions like an LN command, i.e., the logical state of the operand is loaded into the logic accumulator. You should always begin a sequence of logical gating operations with a load command (see L, LN, L-).
- Within a logic sequence, gate the content of the logic accumulator and the logical state of the operand with AND NOT. The iTNC saves the result of the operation in the logic accumulator.

Example:

Gate the inputs I4 and I5 with AND NOT, and assign the result to output O2.

Initial state:

Input I4 = 1
Input I5 = 1
Output O2 = ?

Function	STL	Accumulator content	Operand content
Load the operand content into the logic accumulator.	L I4	1	1
Gate the content of logic accumulator and input I5 with AND NOT.	AN I5	0	1
Assign the gating result to output O2.	= O2	0	0

**Word processing
with the AND NOT
command**

Syntax: AN (AND NOT)

Operands: B, W, D, K

Action:

Gate the contents of the word accumulator and the operand with AND NOT. In accordance with the different data widths of the operands (B = 8 bits; W = 16 bits; D = K = 32 bits), 8, 16 or 32 bits, respectively, are influenced in the accumulator. Thus, bit 0 of the accumulator is gated with bit 0 of the operand, bit 1 of the accumulator with bit 1 of the operand, etc. The iTNC saves the result of the operation in the word accumulator.

Example:

Gate the content of words W4 and W6 with AND NOT, and assign the result to word W8.

Initial state:

Word W4 = 36 AA (hex)

Word W6 = 3C 36 (hex)

Word W8 = ?

Function	STL	Accumulator content	Operand content
Load W6 into the word accumulator.	L W6	3C36	3C36
Gate the contents of word accumulator and word W4 with AND NOT.	AN W4	814	36AA
Assign the gating result to word W8.	= W8	814	814



9.8.20 OR (O)

Logic processing with the OR command

Syntax: O (OR)

Operands: M, I, O, T, C

Action:

- At the beginning of a logic sequence, this command functions like an L command, i.e., the logical state of the operand is loaded into the logic accumulator. You should always begin a sequence of logical gating operations with a load command (see L, LN, L-).
- Within a logic sequence, gate the content of the logic accumulator and the logical state of the operand with OR. The iTNC saves the result of the operation in the logic accumulator.

Example:

Gate the inputs I4 and I5 with OR, and assign the result to output O2.

Initial state:

Input I4 = 0

Input I5 = 1

Output O2 = ?

Function	STL	Accumulator content	Operand content
Load the operand content into the logic accumulator.	L I4	0	0
Gate the content of the logic accumulator and input I5 with OR.	O I5	1	1
Assign the gating result to output O2.	= O2	1	1

**Word processing
with the OR
command**

Syntax: O (OR)

Operands: B, W, D, K

Action:

Gate the contents of the word accumulator and the operand with OR. In accordance with the different data widths of the operands (B = 8 bits; W = 16 bits; D = K = 32 bits), 8, 16 or 32 bits, respectively, are influenced in the accumulator. Thus, bit 0 of the accumulator is gated with bit 0 of the operand, bit 1 of the accumulator with bit 1 of the operand, etc. The iTNC saves the result of the operation in the word accumulator.

Example:

Gate the content of byte B5 and byte B6 with OR, and assign the result to word W8.

Initial state:

Byte B5 = 2A (hex)

Byte B6 = 36 (hex)

Word W8 = ?

Function	STL	Accumulator content	Operand content
Load byte B6 into the word accumulator.	L B6	36	36
Gate the contents of the word accumulator and byte B5 with OR.	O B5	3E	2A
Assign the gating result to word W8.	= W8	3E	3E



9.8.21 OR NOT (ON)

Logic processing with the OR NOT command

Syntax: ON (OR NOT)

Operands: M, I, O, T, C

Action:

- At the beginning of a logic sequence, this command functions like an LN command, i.e., the complement of the operand is loaded into the logic accumulator. You should always begin a sequence of logical gating operations with a load command (see L, LN, L-).
- Within a logic sequence, gate the content of the logic accumulator and the logical state of the operand with OR NOT. The iTNC saves the result of the operation in the logic accumulator.

Example:

Gate the inputs I4 and I5 with OR NOT, and assign the result to output O2.

Initial state:

Input I4 = 0

Input I5 = 0

Output O2 = ?

Function	STL	Accumulator content	Operand content
Load the operand content into the logic accumulator.	L I4	0	0
Gate the content of logic accumulator and input I5 with OR NOT.	ON I5	1	0
Assign the gating result to output O2.	= O2	1	1

**Word processing
with the OR NOT
command**

Syntax: ON (OR NOT)

Operands: B, W, D, K

Action:

Gate the contents of the word accumulator and the operand with OR NOT. In accordance with the different data widths of the operands (B = 8 bits; W = 16 bits; D = K = 32 bits), 8, 16 or 32 bits, respectively, are influenced in the accumulator. Thus, bit 0 of the accumulator is gated with bit 0 of the operand, bit 1 of the accumulator with bit 1 of the operand, etc. The iTNC saves the result of the operation in the word accumulator.

Example:

Gate the content of words W4 and W6 with OR NOT, and assign the result to word W8.

Initial state:

Word W4 = 36 AA (hex)

Word W6 = 3C 36 (hex)

Word W8 = ?

Function	STL	Accumulator content	Operand content
Load W6 into the word accumulator.	L W6	3C36	3C36
Gate the contents of word accumulator and word W4 with OR NOT.	ON W4	FD77	36AA
Assign the gating result to word W8.	= W8	FD77	FD77



9.8.22 EXCLUSIVE OR (XO)

Logic processing with the EXCLUSIVE OR command

Syntax: XO (EXCLUSIVE OR)

Operands: M, I, O, T, C

Action:

- At the beginning of a logic sequence, this command functions like an L command, i.e., the logical state of the operand is loaded into the logic accumulator. You should always begin a sequence of logical gating operations with a load command (see L, LN, L-).
- Within a logic sequence, gate the content of the logic accumulator and the logical state of the operand with EXCLUSIVE OR. The iTNC saves the result of the operation in the logic accumulator.

Example:

Gate the inputs I4 and I5 with EXCLUSIVE OR, and assign the result to output O2.

Initial state:

Input I4 = 1

Input I5 = 1

Output O2 = ?

Function	STL	Accumulator content	Operand content
Load the operand content into the logic accumulator.	L I4	1	1
Gate the content of logic accumulator and input I5 with EXCLUSIVE OR.	XO I5	0	1
Assign the gating result to output O2.	= O2	0	0

**Word processing
with the
EXCLUSIVE OR
command**

Syntax: XO (EXCLUSIVE OR)

Operands: B, W, D, K

Action:

Gate the contents of the word accumulator and the operand with EXCLUSIVE OR. In accordance with the different data widths of the operands (B = 8 bits; W = 16 bits; D = K = 32 bits), 8, 16 or 32 bits, respectively, are influenced in the accumulator. Thus, bit 0 of the accumulator is gated with bit 0 of the operand, bit 1 of the accumulator with bit 1 of the operand, etc. The iTNC saves the result of the operation in the word accumulator.

Example:

Gate the contents of byte B5 and byte B6 with EXCLUSIVE OR, and assign the result to word W8.

Initial state:

Byte B5 = 2A (hex)

Byte B6 = 36 (hex)

Word W8 = ?

Function	STL	Accumulator content	Operand content
Load byte B6 into the word accumulator.	L B6	36	36
Gate the contents of the word accumulator and byte B5 with EXCLUSIVE OR.	XO B5	1C	2A
Assign the gating result to word W8.	= W8	1C	1C



9.8.23 EXCLUSIVE OR NOT (XON)

Logic processing with the EXCLUSIVE OR NOT command

Syntax: XON (EXCLUSIVE OR NOT)

Operands: M, I, O, T, C

Action:

- At the beginning of a logic sequence, this command functions like an LN command, i.e., the logical state of the operand is loaded into the logic accumulator. You should always begin a sequence of logical gating operations with a load command (see L, LN, L-).
- Within a logic sequence, gate the content of the logic accumulator and the logical state of the operand with EXCLUSIVE OR NOT. The iTNC saves the result of the operation in the logic accumulator.

Example:

Gate the inputs I4 and marker M500 with EXCLUSIVE OR NOT, and assign the result to output O2.

Initial state:

Input I4 = 0

Marker M500 = 0

Output O2 = ?

Function	STL	Accumulator content	Operand content
Load the operand content into the logic accumulator.	L M500	0	0
Gate the content of logic accumulator and input I4 with EXCLUSIVE OR NOT.	XON I4	1	0
Assign the gating result to output O2.	= O2	1	1



**Word processing
with the
EXCLUSIVE OR
NOT command**

Syntax: XON (EXCLUSIVE OR NOT)

Operands: B, W, D, K

Action:

Gate the contents of the word accumulator and the operand with EXCLUSIVE OR NOT. In accordance with the different data widths of the operands (B = 8 bits; W = 16 bits; D = K = 32 bits), 8, 16 or 32 bits, respectively, are influenced in the accumulator. Thus, bit 0 of the accumulator is gated with bit 0 of the operand, bit 1 of the accumulator with bit 1 of the operand, etc. The iTNC saves the result of the operation in the word accumulator.

Example:

Gate the content of words W4 and W6 with EXCLUSIVE OR NOT, and assign the result to word W8.

Initial state:

Word W4 = 36 AA (hex)

Word W6 = 3C 36 (hex)

Word W8 = ?

Function	STL	Accumulator content	Operand content
Load W6 into the word accumulator.	L W6	3C36	3C36
Gate the contents of word accumulator and word W4 with EXCLUSIVE OR NOT.	XON W4	F563	36AA
Assign the gating result to word W8.	= W8	F563	F563



9.8.24 ADDITION (+)

Syntax: + (PLUS)

Operands: B, W, D, K

Action:

The iTNC extends the operand to the width of the accumulator (32 bits) and then adds the content of the operand to the content of the word accumulator. The result of the operation is stored in the word accumulator where you can process it further.

Example:

Add the constant and the number saved in word W6, then assign the result to double word D8.

Initial state:

Constant = 10 000 (dec)

Word W6 = 200 (dec)

Double word D8 = ?

Function	STL	Accumulator content	Operand content
Load the constant into the word accumulator.	L K10000	10000	
Add the content of the word accumulator and word W6.	+ W6	10200	200
Assign the result to double word D8.	= D8	10200	10200

9.8.25 SUBTRACTION (-)

Syntax: - (MINUS)

Operands: B, W, D, K

Action:

The iTNC extends the operand to the width of the accumulator (32 bits) and then subtracts the content of the operand from the content of the word accumulator. The result of the operation is stored in the word accumulator where you can process it further.

Example:

Subtract the number saved in word W6 from the constant, and then assign the result to double word D8.

Initial state:

Constant = 10 000 (dec)

Word W6 = 200 (dec)

Double word D8 = ?

Function	STL	Accumulator content	Operand content
Load the constant into the word accumulator.	L K10000	10000	
Subtract word W6 from the content of the word accumulator.	- W6	9800	9800
Assign the result to double word D8.	= D8	9800	9800



9.8.26 MULTIPLICATION (X)

Syntax: x (MULTIPLY)

Operands: B, W, D, K

Action:

The iTNC extends the operand to the width of the accumulator (32 bits) and then multiplies the content of the operand with the content of the word accumulator. The result of the operation is stored in the word accumulator where you can process it further. If the iTNC cannot execute the multiplication correctly, it then sets marker M4200, otherwise it resets it.

Example:

Multiply the constant by the number saved in word W6, then assign the result to double word D8.

Initial state:

Constant = 100 (dec)

Word W6 = 20 (dec)

Double word D8 = ?

Function	STL	Accumulator content	Operand content
Load the constant into the word accumulator.	L K100	100	
Multiply the content of the word accumulator by word W6.	x W6	2000	20
Assign the result to double word D8.	= D8	2000	2000

		Set	Reset
M4200	Overflow during multiplication	NC	PLC

9.8.27 DIVISION (/)

Syntax: / (DIVIDE)

Operands: B, W, D, K

Action:

The iTNC extends the operand to the width of the accumulator (32 bits) and then divides the content of the word accumulator by the content of the operand. The result of the operation is stored in the word accumulator where you can process it further. If the iTNC cannot execute the division correctly, it then sets marker M4201, otherwise it resets it.

Example:

Divide the constant by the number saved in word W6, then assign the result to double word D8.

Initial state:

Constant = 100 (dec)

Word W6 = 20 (dec)

Double word D8 = ?

Function	STL	Accumulator content	Operand content
Load the constant into the word accumulator.	L K100	100	
Divide the content of the word accumulator by word W6	/ W6	5	20
Assign the result to double word D8.	= D8	5	5

		Set	Reset
M4201	Division by 0	NC	PLC



9.8.28 REMAINDER (MOD)

Syntax: MOD (MODULO)

Operands: B, W, D, K

Action:

The iTNC extends the operand to the width of the accumulator (32 bits) and then calculates the REMAINDER resulting from the division of the content of the word accumulator by the content of the operand. The REMAINDER is stored in the word accumulator where you can process it. If the iTNC cannot execute the MOD command correctly, it sets marker M4202, otherwise it resets it.

Example:

Divide the number saved in word W6 by the constant, then calculate the REMAINDER and assign the result to double word D8.

Initial state:

Word W6 = 50 (dec)
 Constant = 15 (dec)
 Double word D8 = ?

Function	STL	Accumulator content	Operand content
Load W6 into the word accumulator.	L W6	50	50
Divide the content of the word accumulator by a constant, then save the integral REMAINDER in the word accumulator.	MOD K15	11	15
Assign the REMAINDER to double word D8.	= D8	11	11

		Set	Reset
M4202	Incorrectly executed modulo	NC	PLC

9.8.29 INCREMENT (INC)

INCREMENT operand **Syntax:** INC (INCREMENT)
Operands: B, W, D
Action:
Increase the content of the addressed operand by one.

INCREMENT word accumulator **Syntax:** INCW (INCREMENT WORD)
Operands: None
Action:
Increase the content of the word accumulator by one.

INCREMENT index register **Syntax:** INCX (INCREMENT INDEX)
Operands: None
Action:
Increase the content of the index register by one.

9.8.30 DECREMENT (DEC)

DECREMENT operand **Syntax:** DEC (DECREMENT)
Operands: B, W, D
Action:
Decrease the content of the addressed operand by one.

DECREMENT word accumulator **Syntax:** DECW (DECREMENT WORD)
Operands: None
Action:
Decrease the content of the word accumulator by one.

DECREMENT index register **Syntax:** DECX (DECREMENT INDEX)
Operands: None
Action:
Decrease the content of the index register by one.



9.8.31 EQUAL TO (==)

Syntax: == (EQUAL)

Operands: B, W, D, K

Action:

This command sets off a direct transition from word to logical processing. Compare the content of the word accumulator with the content of the addressed operand. If the word accumulator and the operand are equal, the condition is true and the iTNC sets the logic accumulator to 1. If they are not equal, the logic accumulator is set to 0. The comparison takes place over the number of bits corresponding to the operand:
where B = 8 bits, W = 16 bits, and D = K = 32 bits.

Example:

Compare a constant with the content of double word D8, and assign the result to marker M500.

Initial state:

Constant = 16 000 (dec)

Double word D8 = 15 000 (dec)

Marker M300 = ?

Function	STL	Accumulator content	Operand content
Load the constant into the word accumulator.	L K16000	16000	
Compare the content of the word accumulator with the operand content D8; if not equal, set the logic accumulator to 0.	== D8	0	15000
Assign the result to marker M500.	= M500	0	0

9.8.32 LESS THAN (<)

Syntax: < (LESS THAN)

Operands: B, W, D, K

Action:

This command sets off a direct transition from word to logical processing. Compare the content of the word accumulator with the content of the addressed operand. If the word accumulator is less than the operand, the condition is true and the iTNC sets the logic accumulator to 1. If the word accumulator is greater than or equal to the operand, it sets the logic accumulator to 0. The comparison takes place over the number of bits in the operand:

where B = 8 bits, W = 16 bits, and D = K = 32 bits.

Example:

Compare a constant with the content of double word D8, and assign the result to marker M500.

Initial state:

Constant = 16 000 (dec)

Double word D8 = 15 000 (dec)

Marker M500 = ?

Function	STL	Accumulator content	Operand content
Load the constant into the word accumulator.	L K16000	16000	
Check whether word accumulator < operand; if not, set logic accumulator to 0.	< D8	0	15000
Assign the result to marker M500.	= M500	0	0



9.8.33 GREATER THAN (>)

Syntax: > (GREATER THAN)

Operands: B, W, D, K

Action:

This command sets off a direct transition from word to logical processing. Compare the content of the word accumulator with the content of the addressed operand. If the word accumulator is greater than the operand, the condition is true and the iTNC sets the logic accumulator to 1. If the word accumulator is less than or equal to the operand, it sets the logic accumulator to 0. The comparison takes place over the number of bits in the operand: where B = 8 bits, W = 16 bits, and D = K = 32 bits.

Example:

Compare a constant with the content of double word D8, and assign the result to marker M500.

Initial state:

Constant = 16 000 (dec)

Double word D8 = 15 000 (dec)

Marker M500 = ?

Function	STL	Accumulator content	Operand content
Load the constant into the word accumulator.	L K16000	16000	
Check whether word accumulator > operand; if so, set logic accumulator to 1.	> D8	1	15000
Assign the result to marker M500.	= M500	1	1

9.8.34 LESS THAN OR EQUAL TO (<=)

Syntax: <= (LESS EQUAL)

Operands: B, W, D, K

Action:

This command sets off a direct transition from word to logical processing. Compare the content of the word accumulator with the content of the addressed operand. If the word accumulator is less than or equal to the operand, the condition is true and the iTNC sets the logic accumulator to 1. If the word accumulator is greater than the operand, it sets the logic accumulator to 0. The comparison takes place over the number of bits in the operand:

where B = 8 bits, W = 16 bits, and D = K = 32 bits.

Example:

Compare a constant with the content of double word D8, and assign the result to marker M500.

Initial state:

Constant = 16 000 (dec)

Double word D8 = 15 000 (dec)

Marker M500 = ?

Function	STL	Accumulator content	Operand content
Load the constant into the word accumulator.	L K16000	16000	
Check whether word accumulator <= operand; if not, set logic accumulator to 0.	<= D8	0	15000
Assign the result to marker M500.	= M500	0	0



9.8.35 GREATER THAN OR EQUAL TO (>=)

Syntax: >= (GREATER EQUAL)

Operands: B, W, D, K

Action:

This command sets off a direct transition from word to logical processing. Compare the content of the word accumulator with the content of the addressed operand. If the word accumulator is less than or equal to the operand, the condition is true and the iTNC sets the logic accumulator to 1. If the word accumulator is greater than the operand, it sets the logic accumulator to 0. The comparison takes place over the number of bits in the operand:

where B = 8 bits, W = 16 bits, and D = K = 32 bits.

Example:

Compare a constant with the content of double word D8, and assign the result to marker M500.

Initial state:

Constant = 16 000 (dec)

Double word D8 = 15 000 (dec)

Marker M500 = ?

Function	STL	Accumulator content	Operand content
Load the constant into the word accumulator.	L K16000	16000	
Check whether word accumulator >= operand; if so, set logic accumulator to 1.	>= D8	1	15000
Assign the result to marker M500.	= M500	1	1

9.8.36 NOT EQUAL (<>)

Syntax: <> (NOT EQUAL)

Operands: B, W, D, K

Action:

This command sets off a direct transition from word to logical processing. Compare the content of the word accumulator with the content of the addressed operand. If the word accumulator and the operand are not equal, the condition is true and the iTNC sets the logic accumulator to 1. If they are equal, the logic accumulator is set to 0. The comparison takes place over the number of bits corresponding to the operand:
where B = 8 bits, W = 16 bits, and D = K = 32 bits.

Example:

Compare a constant with the content of double word D8, and assign the result to marker M500.

Initial state:

Constant = 16 000 (dec)

Double word D8 = 15 000 (dec)

Marker M500 = ?

Function	STL	Accumulator content	Operand content
Load the constant into the word accumulator.	L K16000	16000	
Check whether word accumulator <> operand; if so, set logic accumulator to 1.	<> D8	1	15000
Assign the result to marker M500.	= M500	1	1



9.8.37 AND [] (A[])

Syntax: A[] (AND [])

Operands: None

Action:

The use of parentheses enables you to alter the sequence of processing logical commands in a statement list. The opening-parenthesis command puts the content of the accumulator onto the program stack. If you address the logic accumulator in the last command before an opening-parenthesis statement, the iTNC puts the content of the logic accumulator onto the program stack. If you address the word accumulator, the iTNC saves the content of the word accumulator. The "close-parenthesis" statement gates the buffered value from the program stack with the content of the logic accumulator or the word accumulator, depending on which accumulator was addressed prior to the "open-parenthesis" instruction. The iTNC assigns the result of the gating operation to the corresponding accumulator. Maximum nesting depth: 16 parentheses.

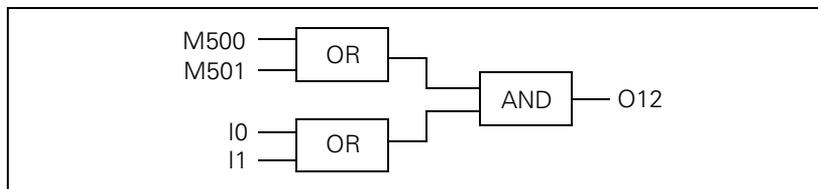
Please note:

The sequence of function is the same for word processing, however, the iTNC writes the complete word accumulator onto the program stack.

Example:

Example for the commands AND [], AND NOT [], OR [], OR NOT [], EXCLUSIVE OR [], EXCLUSIVE OR NOT []:

Use parentheses to develop a statement list in accordance with the following logic circuit diagram:



Initial state:

Marker M500 = 0
 Input I0 = 0
 Marker M501 = 1
 Input I1 = 1
 Output O12 = ?

Function	STL	Accumulator content	Operand content
Load marker M500 into the logic accumulator.	L M500	0	0
Gate logic accumulator with marker M501.	O M501	1	1
Opening parenthesis: Buffer the accumulator content onto the program stack.	A[
Load the state of input I0 into the logic accumulator.	L I0	0	0
Gate the logic accumulator with the state of input I1.	O I1	1	1
Closing parenthesis: Gate the accumulator content with the program stack (A[, O[...]).]		
Assign the result of the total operation to output O12.	= O12	1	1



9.8.38 AND NOT [] (AN[])

Syntax: AN[] (AND NOT [])

Operands: None

Action:

See example of command A[] (AND [])

9.8.39 OR [] (O[])

Syntax: O[] (OR [])

Operands: None

Action:

See example of command A[] (AND [])

9.8.40 OR NOT [] (ON[])

Syntax: ON[] (OR NOT [])

Operands: None

Action:

See example of command A[] (AND [])

9.8.41 EXCLUSIVE OR [] (XO[])

Syntax: XO[] (EXCL: OR [])

Operands: None

Action:

See example of command A[] (AND [])

9.8.42 EXCLUSIVE OR NOT [] (XON[])

Syntax: XON[] (EXCL: OR NOT [])

Operands: None

Action:

See example of command A[] (AND [])

9.8.43 ADDITION [] (+ [])

Syntax: +[] (PLUS[])

Operands: None

Action:

Use parentheses together with arithmetical commands **only** for word processing. By using parentheses you can change the sequence of processing in a statement list. The opening-parenthesis command puts the content of the word accumulator onto the program stack. This clears the accumulator for calculation of intermediate results. The closing-parenthesis command gates the buffered value from the program stack with the content of the word accumulator. The iTNC saves the result in the accumulator again. Maximum nesting depth: 16 parentheses. If an error occurs during calculation, the iTNC sets the marker M4201.

Example:

Example for the commands ADD [], SUBTRACT [], MULTIPLY [], DIVIDE [], REMAINDER [].

Divide a constant by double word D36, add the result to double word D12, and assign the result to double word D100.

Initial state:

Constant = 1000 (dec)
 Double word D12 = 15000 (dec)
 Double word D36 = 100 (dec)
 Double word D100 = ?

Function	STL	Accumulat or content	Operand content
Load the double word D12 into the word accumulator.	L D12	15000	15000
Opening parenthesis: Buffer the accumulator content onto the program stack.	+[
Load the constant K1000 into the word accumulator	L K1000	1000	
Divide the word accumulator by the content of the double word D36.	/ D36	10	100
Closing parenthesis: Gate the accumulator content with the program stack (+[, -[.....).]		
Assign the result of the total operation to double word D100.	= D100	15010	15010

		Set	Reset
M4200	Overflow during multiplication	NC	PLC
M4201	Division by 0	NC	PLC
M4202	Incorrectly executed modulo	NC	PLC
M4203	Error status for PLC module	NC	NC/PLC



9.8.44 SUBTRACT [] (-[])

Syntax: -[] (MINUS -[])

Operands: None

Action:

See example of command ADD []

9.8.45 MULTIPLY [] (x[])

Syntax: x[] (MULTIPLY [])

Operands: None

Action:

See example of command ADD []

9.8.46 DIVIDE [] (/[])

Syntax: /[] (DIVIDE [])

Operands: None

Action:

See example of command ADD []

9.8.47 REMAINDER [] (MOD[])

Syntax: MOD[] (MODULO [])

Operands: None

Action:

See example of command ADD []



9.8.48 EQUAL TO [] (==[])

Syntax: ==[] (EQUAL[])

Operands: None

Action:

By using parentheses you can change the sequence of processing comparative commands in a statement list. The opening-parenthesis command puts the content of the word accumulator onto the program stack. This clears the accumulator for calculation of intermediate results.

The closing-parenthesis command gates the buffered value from the program stack with the content of the word accumulator. The iTNC saves the result in the accumulator again. Maximum nesting depth: 16 parentheses.

Comparative commands cause a direct transition from word to logical processing. If the specified comparative condition is true, the iTNC sets the logic accumulator to 1; if the condition is not fulfilled, it sets it to 0.

Example:

Multiply a constant with double word D36, compare the result with double word D12, and assign the result to output O15.

Initial state:

Constant = 1000 (dec)
 Double word D12 = 15000 (dec)
 Double word D36 = 10 (dec)
 Output O15 = ?

Function	STL	Accumulator content	Operand content
Load the double word D12 into the word accumulator.	L D12	15000	15000
Opening parenthesis: Buffer the accumulator content onto the program stack.	== [
Load the constant into the word accumulator.	L K1000	1000	
Multiply the content of the word accumulator with double word D36.	x D36	10000	10
Closing parenthesis: Gate the accumulator content with the program stack (==[, >=[...]); if condition not fulfilled, set logic accumulator to 0.]		
Assign the result to output O15.	= O15	0	0



9.8.49 LESS THAN [] (<[])

Syntax: <[] (LESS THAN [])

Operands: None

Action:

See example of command EQUAL TO []

9.8.50 GREATER THAN [] (>[])

Syntax: >[] (GREATER THAN [])

Operands: None

Action:

See example of command EQUAL TO []

9.8.51 LESS THAN OR EQUAL TO [] (<=[])

Syntax: <=[] (LESS EQUAL [])

Operands: None

Action:

See example of command EQUAL TO []

9.8.52 GREATER THAN OR EQUAL TO [] (>=[])

Syntax: >=[] (GREATER EQUAL [])

Operands: None

Action:

See example of command EQUAL TO []

9.8.53 NOT EQUAL [] (<>[])

Syntax: <>[] (NOT EQUAL [])

Operands: None

Action:

See example of command EQUAL TO []



9.8.54 SHIFT LEFT (<<)

Syntax: << (SHIFT LEFT)

Operands: B, W, D, K

Action:

A SHIFT LEFT statement multiplies the content of the word accumulator by two. This is done by simply shifting the bits in the accumulator by one place to the left. The result must lie in the range of -2 147 483 648 to +2 147 483 647, otherwise the accumulator contains an undefined value. You define the number of shifts through the operand. The iTNC fills the right end of the accumulator with zeros.

This statement is one of the arithmetic commands because it includes the sign bit. For this reason, and to save time, you should not use this command to isolate bits.

Example:

Shift the content of double word D8 four times to the left, then assign it to double word D12.

Initial state:

Double word D8 = 3E 80 (hex)

Double word D12 = ?

Function	STL	Accumulator content	Operand content
Load the double word D8 into the word accumulator	L D8	3E80	3E80
Shift the content of the word accumulator to the left by the number of bits that are specified in the operand.	<< K+1	7D00	
	<< K+1	FA00	
	<< K+1	1F400	
	<< K+1	3E800	
Assign the result to double word D12.	= D12	3E800	3E800

Instead of using the << K+1 command four times, simply use the << K+4 command.



9.8.55 SHIFT RIGHT (>>)

Syntax: >> (SHIFT RIGHT)

Operands: B, W, D, K

Action:

A SHIFT RIGHT statement divides the content of the word accumulator by two. This is done by simply shifting the bits by one place to the right. You define the number of shifts through the operand. The bits that the iTNC shifts to the right out of the accumulator are then lost. The iTNC extends the left side of the accumulator with the correct sign.

This statement is one of the arithmetic commands because it includes the sign bit. For this reason, and to save time, you should not use this command to isolate bits.

Example:

Shift the content of double word D8 four times to the right, then assign it to double word D12.

Initial state:

Double word D8 = 3E 80 (hex)

Double word D12 = ?

Function	STL	Accumulator content	Operand content
Load the double word D8 into the word accumulator	L D8	3E80	3E80
Shift the content of the word accumulator to the right by the number of bits that are specified in the operand.	>> K+1	1F40	
	>> K+1	FA0	
	>> K+1	7D0	
	>> K+1	3E8	
Assign the result to double word D12.	= D12	3E8	3E8

Instead of using the >> K+1 command four times, simply use the >> K+4 command.

9.8.56 BIT SET (BS)

Syntax: BS (BIT SET)

Operands: B, W, D, K, X

Action:

With the BIT SET command you can set each bit in the accumulator to 1. The corresponding bits are selected (addressed) by the content of the specified operand or by a constant. As to the bit numbering, bit 0 = LSB and bit 31 = MSB. For operand contents greater than 32, the iTNC uses the operand value modulo 32, i.e. the integral remainder of the result of the operand value divided by 32.

Example:

Load double word D8 into the accumulator, set bit 0 of the accumulator to 1, and save the result in double word D12.

Initial state:

Double word D8 = 3E 80 (hex)

Double word D12 = ?

Function	STL	Accumulator content	Operand content
Load the double word D8 into the word accumulator	L D8	3E80	3E80
Set the bit specified in the operand to 1.	BS K+0	3E81	
Assign the result to double word D12.	= D12	3E81	3E81



9.8.57 BIT CLEAR (BC)

Syntax: BC (BIT CLEAR)

Operands: B, W, D, K, X

Action:

The BIT CLEAR command is used to set each bit in the accumulator to 0. The corresponding bits are selected (addressed) by the content of the specified operand or by a constant. As to the bit numbering, bit 0 = LSB and bit 31 = MSB. For operand contents greater than 32, the iTNC uses the operand value modulo 32, i.e. the integral remainder of the result of the operand value divided by 32.

Example:

Load double word D8 into the accumulator, set bit 0 of the accumulator to 0, and save the result in double word D12.

Initial state:

Double word D8 = 3E 81 (hex)

Double word D12 = ?

Function	STL	Accumulator content	Operand content
Load the double word D8 into the word accumulator	L D8	3E81	3E81
Set the bit specified in the operand to 0.	BC K+0	3E80	
Assign the result to double word D12.	= D12	3E80	3E80

9.8.58 BIT TEST (BT)

Syntax: BT (BIT TEST)

Operands: B, W, D, K, X

Action:

With the BIT TEST command, you can interrogate the status of each bit in the accumulator. With the BT command there is a direct transition from word to logic processing, i.e. the iTNC checks the state of a bit in the word accumulator and then sets the logic accumulator. If the interrogated bit = 1, the iTNC sets the logic accumulator to 1; otherwise it sets it to 0. The corresponding bits are selected (addressed) by the content of the specified operand or by a constant. As to the bit numbering, bit 0 = LSB and bit 31 = MSB. For operand contents greater than 32, the iTNC uses the operand value modulo 32, i.e. the integral remainder of the result of the operand value divided by 32.

Example:

Load the double word D8 into the accumulator and assign the logical state of bit 0 to output O12.

Initial state:

Double word D8 = 3E 81 (hex)

Output O12 = ?

Function	STL	Accumulator content	Operand content
Load the double word D8 into the word accumulator	L D8	3E81	3E81
Check the state of the bit specified in the operand.	BT K+0	1	
Assign the result to output O12.	= O12	1	1



9.8.59 PUSH DATA ONTO THE DATA STACK (PS)

Logic processing with the PS command

Syntax: PS (PUSH)

Operands: M, I, O, T, C

Action:

The PS command enables you to buffer data. The iTNC loads the addressed operand onto the data stack. Because the data stack has a width of 32 bits, you must write to it with a minimum width of one word. The iTNC copies the operand value into bit 7 of the data stack's current address. The vacant bits of the occupied memory remain undefined or unused. In the event of a stack overflow, the iTNC outputs an error message.

Data stack [bit]																
31	...	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	...		x	x	x	x	x	x	L	x	x	x	x	x	x	x

Example:

See PSW command.

Word processing with the PS command

Syntax: PS (PUSH)

Operands: B, W, D, K

Action:

The PS command enables you to buffer data. The iTNC copies the addressed operand value into the current address of the data stack. During the word processing, the iTNC copies two words per PS command onto the data stack and extends the operand—in accordance with the MSB—with the correct algebraic sign. In the event of a stack overflow, the iTNC displays an error message.

Data stack for byte, word, double word and constant [bit]																
31	24	23	16	15	8	7	0					
x	x	x	x	x	x	x	x	x	x	x	x	x	B	B	B	B
x	x	x	x	x	x	x	x	x	x	x	x	x	W	W	W	W
D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
K	K	K	K	K	K	K	K	K	K	K	K	K	K	K	K	K

Example:

See PSW command.



9.8.60 PULL DATA FROM THE DATA STACK (PL)

Logic processing with the PL command

Syntax: PL (PULL)
Operands: M, I, O, T, C
Action:

The PL command is the counterpart to the PS command. Data that has been buffered with the PUSH command can be taken from the data stack by using the PULL command. The iTNC copies bit 7 of the data stack's current address into the addressed operand. If the stack is empty, the iTNC displays an error message.

Example:

See PSW command.

Word processing with the PL command

Syntax: PL (PULL)
Operands: B, W, D, K
Action:

The PL command is the counterpart to the PS command. Data that has been buffered with the PUSH command can be taken from the data stack by using the PULL command. During word processing, the iTNC copies with the PL command two words of the current data stack address into the addressed memory area. If the stack is empty, the iTNC displays an error message.

Example:

See PSW command.

9.8.61 PUSH LOGIC ACCUMULATOR ONTO THE DATA STACK (PSL)

Syntax: PSL (PUSH LOGICACCU)
Operands: None
Action:

The PSL command enables you to buffer the logic accumulator. With the PSL command, the iTNC copies the logic accumulator onto the data stack. Because the data stack has a width of 32 bits, you must write to it with a minimum width of one word. The iTNC copies the operand value into bit 7 of the data stack's current address. The vacant bits of the occupied memory remain undefined or unused. In the event of a stack overflow, the iTNC outputs an error message.

Memory allocation data stack [bit]																
31	...	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	...		x	x	x	x	x	x	L	x	x	x	x	x	x	x

Example:

See PSW command.



9.8.62 PUSH WORD ACCUMULATOR ONTO THE DATA STACK (PSW)

Syntax: PSW (PUSH WORDACCU)

Operands: None

Action:

The PSW command enables you to buffer the word accumulator. With the PSW command, the iTNC copies the word accumulator onto the data stack. The content of the word accumulator (32 bits) occupies two words on the data stack. In the event of a stack overflow, the iTNC displays an error message.

Example:

Since the sequence is the same for all stack operations, this example also applies to the commands PS, PL, PSW, PLL, PLW. The difference between the individual operations lies merely in the transferred data width.

Call Module 15 at a certain place in the program. After returning to the main program, restore the original accumulator content. Accumulator content before the module call: 1A 44 3E 18

Function	STL	Accumulator content	Data stack
Buffer the word accumulator in the data stack	PSW	1A443E18	1A443E18
Call subroutine 15	CM 15		
Restore data stack into word accumulator.	PLW	1A443E18	1A443E18

9.8.63 PULL LOGIC ACCUMULATOR FROM THE DATA STACK (PLL)

Syntax: PLL (PULL LOGICACCU)

Operands: None

Action:

The PLL command is the counterpart to the PSL command. Data that has been buffered with the PUSH command can be restored from the data stack by using the PULL command. The iTNC copies bit 7 of the data stack's current address into the logic accumulator. If the stack is empty, the iTNC displays an error message.

Example:

See PSW command.



9.8.64 PULL WORD ACCUMULATOR FROM THE DATA STACK (PLW)

Syntax: PLW (PULL WORDACCU)

Operands: None

Action:

The PLW command is the counterpart to the PSW command. Data that has been buffered with the PUSH command can be restored from the data stack by using the PULL command. During word processing, the iTNC copies with the PLW command two words of the current data stack address into the word accumulator. If the stack is empty, the iTNC displays an error message.

Example:

See PSW command.

9.8.65 UNCONDITIONAL JUMP (JP)

Syntax: JP (JUMP)

Operands: Label (LBL)

Action:

After a JP command, the iTNC jumps to the label that you have entered and resumes the program from there. The JP command interrupts a logic sequence.

Example:

See JPT command.



9.8.66 JUMP IF LOGIC ACCUMULATOR = 1 (JPT)

Syntax: JPT (JUMP IF TRUE)

Operands: Label (LBL)

Action:

The JPT command is a conditional jump command. If the logic accumulator = 1, the iTNC resumes the program at the label that you have entered. If the logic accumulator = 0, the iTNC does not jump. The JPT command interrupts a logic sequence.

Example:

This example also applies to the JP and JPF commands.

Depending on the state of the input I5, skip a certain program section.

Initial state:

Input I5 = 1

Function	STL	Accumulator content	Operand content
Load the operand content into the logic accumulator.	L I5	1	1
If logic accumulator =1, jump to LBL 10.	JPT 10	1	
Skip the function.	L I3		
Skip the function.	O M500		
Skip the function.	= 020		
Label	LBL 10		
Resume the program run.	L M100	0	0

9.8.67 JUMP IF LOGIC ACCUMULATOR = 0 (JPF)

Syntax: JPT (JUMP IF FALSE)

Operands: Label (LBL)

Action:

The JPF command is a conditional jump command. If the logic accumulator = 0, the iTNC resumes the program at the label that you have entered. If the logic accumulator = 1, the iTNC does not jump. The JPF command interrupts a logic sequence.

Example:

See JPT command.



9.8.68 CALL MODULE (CM)

Syntax: CM (CALL MODULE)

Operands: Label (LBL)

Action:

After a CM command, the iTNC calls the module that begins at the label that you have entered. Modules are independent subroutines that must be ended with the command EM. You can call modules as often as you wish from different places in your program. The CM command interrupts a logic sequence.

Example:

See CMF command.

9.8.69 CALL MODULE IF LOGIC ACCUMULATOR = 1 (CMT)

Syntax: CMT (CALL MODULE IF TRUE)

Operands: Label (LBL)

Action:

The CMT command is a conditional module call. If the logic accumulator = 1, the iTNC calls the module that begins at the label that you have entered. If the logic accumulator = 0, the iTNC does not call the module. The CMT command interrupts a logic sequence.

Example:

See CMF command.



9.8.70 CALL MODULE IF LOGIC ACCUMULATOR = 0 (CMF)

Syntax: CMF (CALL MODULE IF FALSE)

Operands: Label (LBL)

Action:

The CMF command is a conditional module call. If the logic accumulator = 0, the iTNC calls the module that begins at the label that you have entered. If the logic accumulator = 1, the iTNC does not call the module. The CMF command interrupts a logic sequence.

Example:

This example also applies to the CM and CMT commands.

Depending on the state of the input I5, call the Module 10. Initial state:
Input I5 = 0

Function	STL	Accumulator content	Operand content
Load the operand content into the logic accumulator.	L I5	0	0
If logic accumulator =0, jump to LBL 10.	CMF 10	0	
Resume main program after module execution.	L M100	1	1
	⋮		
End of main program	EM		
Label: Beginning of module	LBL 10		
Statement in the module	L I3	0	0
Statement in the module	O M500	1	1
Statement in the module	= O20	1	1
End of module, resume the main program with the command L M100	EM		



9.8.71 END OF MODULE, END OF PROGRAM (EM)

Syntax: EM (END OF MODULE)

Operands: None

Action:

You must end each program or subroutine (module) with the EM command. An EM command at the end or within a module causes a return jump to the module call (CM, CMT, CMF). The iTNC then resumes the program with the instruction that follows the module call. The iTNC interprets the EM command as program end. The iTNC can reach the subsequent program instructions only through a jump instruction.

9.8.72 END OF MODULE IF LOGIC ACCUMULATOR = 1 (EMT)

Syntax: EMT (END OF MODULE IF TRUE)

Operands: None

Action:

An EMT command causes a return jump to the module call (CM, CMT, CMF) only if the logic accumulator = 1.

9.8.73 END OF MODULE IF LOGIC ACCUMULATOR = 0 (EMF)

Syntax: EMF (END OF MODULE IF FALSE)

Operands: None

Action:

An EMF command causes a return jump to the module call (CM, CMT, CMF) only if the logic accumulator = 0.

9.8.74 LABEL (LBL)

Syntax: LBL (LABEL)

Operands: ASCII name; maximum length: 32 characters

Action:

The label defines a program location as an entry point for the JP and CM commands. You can define up to 1000 jump labels per file. The ASCII name of the label may be up to 32 characters long. However, the iTNC evaluates only the first 16 characters.

For importing global labels, see EXTERN statement.



9.9 INDEX register (X register)

You can use the index register for:

- Data transfer
- Buffering results
- Indexed addressing of operands

The index register is 32 bits wide.

You can use the X register anywhere in the program. The iTNC does not check whether the current content is valid. Exception: During indexed write accessing the iTNC checks whether the amount of available address space is exceeded.

Example: = B100[X]

If the permitted addressable storage is exceeded, the iTNC issues a blinking error message: **PLC: index range incorrect**. Acknowledge the error message by pressing the END key. After restarting the iTNC you must not acknowledge the **POWER INTERRUPTED** message. Switch to the PLC editor, where you will be shown the error line.



Note

At the beginning of the PLC cycle the iTNC sets the index register to 0. Assign the index register a defined value before using it in your program.

The following addresses are valid:

- Mn[X]
- In[X]
- On[X]
- Cn[X]
- Tn[X] Operand number = n+X
- Bn[X] Operand number = n+X
- Wn[X] Operand number = n+2*X
- Dn[X] Operand number = n+4*X
- BTX Content of index register = operand
- BCX Content of index register = operand
- BSX Content of index register = operand
- Sn[X] String number = n+X
- S#Dn[X] Dialog text number = n+X
- S#En[X] Error text number = n+X
- S#An[X] ASCII code +X
- Sn^X Substring from X-th character of the n-th string

The types "S", K, and K\$ cannot be indexed.



Note

If you address S#Dn[X] or S#En[X], the iTNC loads the sequence <SUB>Dnnn or <SUB>Ennn in the string accumulator, where nnn is the modified string number.

Commands for operating the index register

The following commands are available for exchanging data between the word accumulator and index register, or between the stack and index register:

- LX (Load index to accumulator) Index register – word accumulator
- =X (Store accumulator to index) Word accumulator – index register
- PSX (Push index register) Index register – stack
- PLX (Pull index register) Stack – index register
- INCX (Increment index register)
- DECX (Decrement index register)

Error and dialog files

Starting with software version 05, indexed access to error and dialog files is possible via the operands B/W/D.

In the example, a copy of S1 ("TEST" is loaded into S2, and S3 then contains the string "TEST", but only starting from the 3rd character ("ST"). Also, indexed access to error and dialog files is shown:

```
#define ofs          D100
#define Start       D104
#define offset_dialog D108
#define offset_error D112

L   S#"TEST"
=   S1

L   K1
=   ofs

L   K2
=   Start
=X

L   S0[ofs]
=   S2

L   S2^Start
=   S3

L   K5
=   offset_dialog
L   S#D10[offset_dialog]
=   S0[offset_dialog]

L   K6
=   offset_error
L   S#E10[offset_error]
=   S0[offset_error]

EM
```



9.10 Commands for String Processing

String processing enables you to use the PLC program to generate and manipulate any texts. Use Module 9082 to display these texts in the PLC window of the screen, and delete them with Module 9080. The iTNC features one string accumulator and 100 string memories (S0 to S99), in each of which you can save up to 128 characters:

String accumulator (characters)	
1	128
x x x x x x x x x x x x x x x x x x x x x x x x x x	

String memory (characters)	
	128
S0	x x x x x x x x x x x x x x x x x x x x x x x x x x
...	x x x x x x x x x x x x x x x x x x x x x x x x x x

In the PLC program, you can also use operands for indexed access to string operands. Until now, the index register had to be used. For example, see page 1820.

Example

String accumulator (characters)	
1	128
C O O L A N T O N	

String accumulator and string memory are volatile, which means that they are erased by the iTNC when power is switched off. The operand "S" is available for string processing. You can use the operand "S" with different arguments.

Operand declaration

The "S" operands are to be used only for string processing. You can target the following addresses with the various arguments:

- Addressing string memory: After the operand designation, specify the number of the desired memory (S0 to S99).
- Address part of a string: Use the address S_n^X (see INDEX Register). The iTNC addresses the substring beginning with the X-th character of the specified string.
- Immediate string: You can also enter a string directly in the PLC program. The text string, which may contain a maximum of 128 characters, must be indicated by quotation marks.
Example: **"Coolant 1 on"**
- Texts from the PLC error message file or from the PLC dialog file: By specifying the line number you can read texts from the active error message file or dialog file: **PLC-ERROR: S#Exx xx**: Line number from the PLC error message file (0 to 2047)
PLC-DIALOG: S#Dxx xx: Line number from the PLC dialog file (0 to 2047). Enter the string #Exx or #Dxx in the argument <arg> of the string command. The iTNC then saves a 5-byte-long string <SUB> E0xx or <SUB> D0xx (<SUB> = ASCII <SUB>) in the accumulator. Instead of this string, the iTNC reads the line xx of the active error message or dialog file on the screen.
- Enter an ASCII character in the string. Define the ASCII character through its code: S#Axxx

Logical comparisons during string processing

Use the following procedure to compare two strings, depending on the argument:

- Compare the string memory or immediate string, then the iTNC checks both strings character by character. After the first character that does not fulfill the condition of comparison, the control resets the logic accumulator. Then the iTNC does not check the remaining characters. During a comparison, the iTNC always uses the significance of the characters from the ASCII table. This results, for example, in:
A < B
AA > A
- If you have entered PLC error messages or PLC dialog texts in the argument, the iTNC compares the position in the error-message file or dialog file (0 to 2047), but not the actual text as in an immediate string.

The processing times depend on the length of the strings.

9.10.1 LOAD String (L)

Syntax: L (LOAD)

Operands: S <arg>

Action:

Load the string accumulator. The string that the iTNC is to load is selected through the argument <arg> after the operand designation. See also "Operand declaration."

Example:

See OVWR command.

9.10.2 ADD String (+)

Syntax: + (PLUS)

Operands: S <arg>

Action:

Append another string to a string in the string accumulator. The string that the iTNC is to load is selected through the argument <arg> after the operand designation. See also "Operand declaration." The resulting string must not be longer than 128 characters.

Example:

See OVWR command.

9.10.3 STORE String (=)

Syntax: = (STORE)

Operands: S <arg>

Action:

Assign the content of the string accumulator to the string memory. The memory into which the iTNC is to copy the string is selected through the argument <arg> after the operand designation. Permissible arguments: 0 to 99 (String memory S0 to S99). See also "Operand declaration."

Example:

See OVWR command.

9.10.4 OVERWRITE String (OVWR)

Syntax: OVWR (OVERWRITE)

Operands: S <arg>

Action:

Save the string from the string accumulator in a string memory. This command differs from the = command in that the iTNC does not transfer the "string end" character along with it. In this way you can overwrite the beginning of a string that is already in the string memory. The memory into which the iTNC is to copy the string is selected through the argument <arg> after the operand designation. Permissible arguments: 0 to 99 (string memory S0 to S99). See also "Operand declaration."

Example:

This example also applies to the string commands L, + and =.

Add a string from the string memory S0 to an immediate string. The result is to overwrite the contents of string memory S1. Initial state:

Immediate string = **HYDRAULICS**
 String memory S0 = **OIL**
 String memory S1 = **COOLANT MISSING**

String memory (characters)	
	1 128
S0	O I L
S1	C O O L A N T M I S S I N G
...	...

Function	STL	String accumulator (characters)
		1 128
Load the immediate string into the string accumulator	L S "HYDRAUL."	O I L
Add content of string memory S0 to string accumulator.	+ S0	H Y D R A U L . O I L
Overwrite content of string memory S1 with content of string accumulator.	OVWR S1	H Y D R A U L . O I L

Final status:

String memory (characters)	
	1 128
S0	O I L
S1	H Y D R A U L . O I L M I S S I N G
...	...



9.10.5 EQUAL TO Command for String Processing (==)

Syntax: == (EQUAL)

Operands: S <arg>

Action:

This command sets off a direct transition from string to logical processing. Compare the content of the string accumulator with the string in the argument. If the string accumulator and the operand are equal, the condition is true and the iTNC sets the logic accumulator to 1. If they are not equal, the iTNC sets the logic accumulator is set to 0.

Example:

See command <>.

9.10.6 LESS THAN Command for String Processing (<)

Syntax: < (LESS THAN)

Operands: S <arg>

Action:

This command sets off a direct transition from string to logical processing. Compare the content of the string accumulator with the string in the argument. If the string accumulator is less than the operand, the condition is true and the iTNC sets the logic accumulator to 1. If the string accumulator is greater than or equal to the operand, it sets the logic accumulator to 0.

Example:

See command <>.

9.10.7 GREATER THAN Command for String Processing (>)

Syntax: > (GREATER THAN)

Operands: S <arg>

Action:

This command sets off a direct transition from string to logical processing. Compare the content of the string accumulator with the string in the argument. If the string accumulator is greater than the operand, the condition is true and the iTNC sets the logic accumulator to 1. If the string accumulator is less than or equal to the operand, it sets the logic accumulator to 0.

Example:

See command <>.

9.10.8 LESS THAN OR EQUAL TO Command for String Processing (<=)

Syntax: <= (LESS EQUAL)

Operands: S <arg>

Action:

This command sets off a direct transition from string to logical processing. Compare the content of the string accumulator with the string in the argument. If the string accumulator is less than or equal to the operand, the condition is true and the iTNC sets the logic accumulator to 1. If the string accumulator is greater than the operand, it sets the logic accumulator to 0.

Example:

See command <>.

9.10.9 GREATER THAN OR EQUAL TO Command for String Processing (>=)

Syntax: >= (GREATER EQUAL)

Operands: S <arg>

Action:

This command sets off a direct transition from string to logical processing. Compare the content of the string accumulator with the string in the argument. If the string accumulator is greater than or equal to the operand, the condition is true and the iTNC sets the logic accumulator to 1. If the string accumulator is less than the operand, it sets the logic accumulator to 0.

Example:

See command <>.



9.10.10 NOT EQUAL Command for String Processing (<>)

Syntax: <> (NOT EQUAL)

Operands: S <arg>

Action:

This command sets off a direct transition from string to logical processing. Compare the content of the string accumulator with the string in the argument. If the string accumulator is not equal to the operand, the condition is true and the iTNC sets the logic accumulator to 1. If the string accumulator is equal to the operand, it sets the logic accumulator to 0.

Example:

This example of string processing also applies to the commands =, <, >, <=, >=, <>.

Compare the immediate string with the content of the string memory S0. Depending on the result, call Module 50.

Initial state:

String memory S0 = SPINDLE 2

Immediate string = SPINDLE 1

String memory (characters)	
	1 128
S0	S P I N D L E 2
...	...

Function	STL	String accumulator (characters), or logic accumulator
		1 128
Load the immediate string into the string accumulator	LS "SPINDLE 1"	S P I N D E L 1
Gate the content of string memory S0 with content of string accumulator (=, <, >, >=, ...)	<> S0	S P I N D E L 2
If the condition is fulfilled, set logic accumulator to 1 and call the module.	CMT 50	Logic accumulator = 1



Module 9071 Find the string length

The iTNC ascertains the length of the string in a selectable string memory (S0 to S99).

Call:

PS K/B/W/D/S<>String no. or string>

CM 9071

PL B/W/D <>Length of the string>

Error code:

Marker	Value	Meaning
M4203	0	String length found
	1	Error. See W1022.
W1022	2	Invalid immediate strings, address of the source or target string is out of range (S0 to S99), string memory was searched but no string end was found

Module 9072 Copy a byte block into a string

Module 9072 copies a byte block into a string. The module does not check whether the byte block consists of valid ASCII characters. The content of the string may not be correctly displayed (e.g. due to special characters or end-of-string identification). An end-of-string identifier is set automatically after the last copied byte. The entire programmed byte block is always copied, regardless of any end-of-string identifiers within the byte block.

Call:

PS B/W/D/K <>start byte>

PS B/W/D/K <>Length of the byte block>

PS B/W/D/K <>String number>

CM 9072

Error code:

Marker	Value	Meaning
M4203	0	Byte block copied into string
	1	Error code in W1022
W1022	1	Invalid length of the programmed byte block (max. 127 characters)
	3	Invalid address of the start byte
	4	Invalid sum of the start byte and the length of the byte block
	11	Invalid string

Module 9073 Copy a string into a byte block

Module 9073 copies a string into a byte block. The module does not check whether the string consists of valid ASCII characters. The module always copies over the entire programmed length of the byte block, regardless of any end-of-string identifier in the byte block.

Call:

PS B/W/D/K <>Target byte>
PS B/W/D/K <>Length of the byte block>
PS B/W/D/K <>String number>
CM 9073

Error code:

Marker	Value	Meaning
M4203	0	String was copied into byte block
	1	Error code in W1022
W1022	1	Invalid target byte
	2	Invalid length of the programmed byte block (max. 127 characters)
	4	Invalid sum of target byte and length of the byte block
	11	Invalid string



9.11 Submit Programs

Submit programs are subroutines that the PLC submits to the NC for processing. This allows you to solve problems that are very processor-intensive, require program loops, or must wait for external results. It is a prerequisite that these programs are not restricted to a definite time frame. Depending on the processor utilization, the iTNC provides a certain computing power for a submit program. You start submit programs from the PLC program. They can access all data memory areas (M/B/W/D) as the main program can. To prevent problems, ensure that data processed by the PLC program is clearly separated from data processed by the submit program. You can place up to eight submit programs in a queue (submit queue). Each submit program receives an "identifier" (a number between 1 and 255, assigned by the NC), which the iTNC enters in the word accumulator. With this identifier and the REPLY function you can then interrogate whether the program is in the queue, is being processed, or has already been processed. The iTNC processes the submit programs in the sequence in which they were entered in the queue. If errors occur during execution of the submit program, the NC sets the following markers:

		Set	Reset
M4200	Overflow during multiplication	NC	PLC
M4201	Division by 0	NC	PLC
M4202	Incorrectly executed modulo	NC	PLC
M4203	Error status for PLC module	NC	NC/PLC
M4204	Reserved for errors that the PLC programmer would like to catch	NC	NC

The iTNC lists these markers separately in the submit job. This means that the same markers can be edited simultaneously in the PLC run program without changing the original markers. No exact times can be stated for the commands for managing the submit queue.

9.11.1 Calling the submit program (SUBM)

Syntax: SUBM (SUBMIT)

Operands: Label (LBL)

Action:

Assign an identifier (1 to 255) to a labeled subroutine and put it in the queue. At the same time, the iTNC writes the assigned number in the word accumulator. If programs are already entered in the submit queue, the iTNC does not run the addressed program until the programs before it are finished. A submission to the queue may only take place from a PLC program. A SUBM command in a submit program is not possible.

If there is no room in the queue, or if you program the SUBM command in a submit program (nesting), the iTNC assigns the value "0" to the word accumulator.

Example:

See CAN command.

9.11.2 Interrogating the status of a submit program (RPLY)

Syntax: RPLY (REPLY)

Operands: B/W

Action:

Interrogate the status of the submit program with the specified identifier. You must have already stored the identifier in a byte or word when you call the submit program. With the RPLY command and the defined memory address (byte or word containing the identifier) the iTNC transfers one of the following processing states to the word accumulator:

- Word accumulator = 0: Program complete/not in the queue
- Word accumulator = 1: Program running
- Word accumulator = 2: Program in the queue

Example:

See CAN command.



9.11.3 Canceling a submit program (CAN)

Syntax: CAN (CANCEL)

Operands: B/W

Action:

Cancel a submit program with the specified identifier during processing, or remove it from the queue. You must have already stored the identifier in a byte or word when you call the submit program. After you have canceled the program, the iTNC immediately starts the next submit program from the queue. The following PLC modules cannot be canceled at just any location with CANCEL:

- PLC module for access to screen (908X)
- PLC module for reading NC files (909X)

For these modules, you must check with the RPLY command whether the CAN command may be executed.

Example:

This example also applies to the SUBM and RPLY commands.

Depending on input I10, submit the subroutine with the label LBL 300 to the NC for processing. In addition, check the processing status of the subroutine in the main program with the RPLY command, and cancel it with the CAN command, depending on input I11.

Function	STL
Load the content of input I10 into the logic accumulator	L I10
If logic accumulator = 0, jump to LBL 100.	JPF 100
Interrogate the status of the submit program and load it into the word accumulator.	RPLY B128
If the word accumulator is not equal to 0, i.e., the submit program has already been transferred to the NC for processing, set the logic accumulator to 1.	<> K+0
If logic accumulator = 1, jump to LBL 100.	JPT 100
Call submit program 300.	SUBM 300
Save the identifier of the submit program in byte 128.	= B128
Label	LBL 100
Load the state of input I11 into the logic accumulator	L I11
If logic accumulator = 0, jump to LBL 110 (skip the program cancellation).	JPF 110
Cancel the submit program.	CAN B128
Label	LBL 110
	⋮
End of main program	EM
Label: Beginning of the submit program	LBL 300
	⋮
End of the submit program	EM

Always insert submit programs, like any module, at the end of the main program. In this case, the content of the submit program could be a display in the PLC window that is realizable through permanently assigned PLC modules.



9.12 Cooperative multitasking

You can run several processes in the PLC with cooperative multitasking. Unlike genuine multitasking, with cooperative multitasking information and tasks are exchanged only at places that you define. Cooperative multitasking permits up to eight parallel PLC processes and the submit queue. In a program that you have started with SUBM, you can use commands for changing tasks and controlling events (Module 926x). You should additionally insert a task change between the individual jobs in the submit queue, so that the iTNC can execute parallel processes by the end of a job at the latest. The cyclic PLC main program does not participate in cooperative multitasking, but interrupts a submit job and the parallel processes at whatever their current stage is.

9.12.1 Starting a parallel process (SPAWN)

Syntax: SPAWN <label>

Operands: D

Action:

In the specified double word, the iTNC returns the identifier (, see page 1831). If no process could be started, the iTNC returns the value -1. You can call the spawn command only in a submit job or in another spawn process (maximum of 16 parallel processes are permitted). If a process ends with EM, the iTNC removes it from the memory to provide space.

9.12.2 Control of events

The parallel processes can make events available to one another. This saves processing time otherwise spent in the constant interrogating of operating states by the individual processes. A special feature of event control is the waiting period, during which the process can "sleep" for a programmed time. With this function you can repeat program sections in a slow time grid, for example for display or monitoring functions.

List of events

In the OEM.SYS file, enter the command **PLCEVENTS=** to enter the complete name of an ASCII file in which you define the events. The entries in the event file are listed line by line with the following syntax:

<Job name> ; <condition> ; <event mask>; [comment]

Event	Function
<Job name>	This name is identical with the label specified with the spawn command. The iTNC evaluates only the first 16 characters.
<Condition>	Logical expression in accordance with the C language convention, identical with the syntax used in function FN20 (see "Data transfer PLC > NC program (Q-parameters)" on page 1714). Operand: M/I/O/T/C/B/W/D with a number that is permissible for this type Condition: == Equal to != or <> Not equal to < Less than > Greater than <= Less than or equal to >= Greater than or equal to If you enter no condition, the interruption will continue until the operand = 0.
<Event mask>	Hexadecimally coded mask of events that are triggered if the condition is fulfilled. The constraints defined in Module 9260 apply for bits 16 to 31.

Example

Entry in the OEM.SYS file:

```
PLCEVENTS=PLC:\EXAMPLE.PEV
```

Content of the file PLC:\EXAMPLE.PEV:

```
JOB_1;I5==1;$0010; Event $0010 to process JOB_1, if I5==1  
JOB_1;B20==5;$0004; Event $0004 to process JOB_1, if B20==5  
AUXJOB;W6 <10;$0100; Event $0100 to process AUXJOB, if W6 <10
```

The iTNC triggers an event if a particular condition is met after one run of the cyclic PLC program and if this condition was not met after the previous run of the cyclic PLC program (edge formation). The number of events of this type that can be activated simultaneously is limited to 15.



If you produce a PLC process with the spawn command, the iTNC searches the event file for entries for this process. It places all relevant entries in a list that is run after every cycle of the cyclic PLC program. If a PLC process terminates itself, or if you terminate the process by recompiling the PLC program, the iTNC then deletes all entries in the list.

The iTNC does not monitor the entries in the event file. This means that syntactically incorrect entries of incorrect job names do not result in an error message.

The iTNC issues a blinking error message if

- A non-existent event file is listed in the OEM.SYS file (when the first spawn command is executed).
- Due to the number of entries in the event file, more events need to be monitored than the run-time list permits. Maximum number of entries in the run-time list: 15

Process monitor

In the PLC programming mode you can use the PROCESS MONITOR soft key to open a status screen in which the iTNC displays all parallel processes, including the process for the submit queue. In a time interval, which can be set with the "+" and "-" soft keys, the iTNC displays:

- The name of the process (**TASKNAME**)
- The current status of the process (**STATE**)
 - Executable (**SCHED**)
 - Running (**RUN+**)
 - Waiting for event (**EVWAIT**)
 - Waiting for time period (**TMWAIT**)
 - AND-gating of the bits in the event mask (**AND**)
 - OR-gating of the bits in the event mask (**OR**)
- The event mask (**EVMASK**)
- The PLC module letting the process wait (**MOD**)
- How often the process has changed contexts in the last time interval (**SCHED**).
- How much CPU time the processor has used from the defined time interval (**CPU(ms)**). The iTNC also shows the distribution of CPU time in a bar chart (**RATIO**).

Module 9260 Receive events and wait for events

Call the module only in a submit job or spawn job. The module enables a spawn job or submit job to interrogate or wait for the occurrence of one or more events. At the same time, the module triggers a change in context.

Markers 4200 to 4202 and 4204 have undefined changes after the module call. If you transfer the value zero for the event mask, the iTNC returns all set events without deleting them. Otherwise, in a call with a waiting period, the iTNC returns all the requested events and deletes them. For a call without a waiting period, the iTNC returns and deletes the events only if the condition is met.

Symbolic operands of the PLC program can be used for PLC Module 9260 if events were defined in a file and the token PLCEVENTS = in the OEM.SYS.

Example:

Token in OEM.SYS: **PLCEVENTS = PLC:\EVENTS.PEV**

Entry in the file EVENTS.PEV: **SPAWN_PROC1; KEYCODE==5; \$0008**

Condition: PLC variable **KEYCODE** corresponds to W274

The event 8 will be sent to the spawn process SPWAWN_PROC1 after pressing the Ctrl-E (= 5) key.

If the events are OR-gated, the iTNC returns and deletes only the set events. You can specify the events to be deleted by calling without a waiting period and with an OR gate.

Event bits 16 to 31 are reserved for the operating system:

- Bit 16: BREAK, cancels a function. Setting and reading is permitted. If you transmit this event, the iTNC cancels access to interfaces and the network!
- Bit 17: Reserved, do not use
- Bit 18: Reserved, do not use
- Bit 19: QUIT, acknowledgment of a request. Use this bit only in the immediate context of a request.
- Bit 20 to bit 31: Reserved, do not use

Call:

```
PS   B/W/D/K  <>Wait>
           0 = Do not wait
           -1 = Wait
PS   B/W/D/K  <>AND/OR>
           0 = OR-gated, otherwise AND-gated
PS   B/W/D/K  <>Event mask>
           0 = Available events
CM   9260
PS   B/W/D/K  <>Events>
           Read events
```

Error code:

Marker	Value	Meaning
M4203	0	Event has been read
	1	Error code in W1022
W1022	2	Incorrect transfer value for <Wait> parameter
	20	Module was not called in a spawn or submit job



Module 9261 Send events

With this module you can send events to a spawn or submit job and then interrogate them with Module 9260. You can call the module in the cyclic program section, in submit jobs and in spawn processes. The iTNC addresses the receiver through the identifier that the spawn command has returned. The submit queue is addressed through the identifier \$80000000 (not through the identifier returned by the SUBM command!). The iTNC always assigns the events that you send to the submit queue to the job that is running at the time of arrival. If they are not read by this job, they remain for the next one. If you wish the receiver process to start immediately, after Module 9261 you must also call Module 9262 to enable a change of context.

Event bits 16 to 31 are reserved for the operating system (see Module 9260).

Call:

```
PS    D/K    <>Identifier>
        Identifier from the spawn command of the receiver
        K$80000000 = submit queue
PS    B/W/D/K <>Events>
        Events to be triggered, bit encoded
CM    9261
```

Error code:

Marker	Value	Meaning
M4203	0	Event has been sent
	1	Error code in W1022
W1022	30	Incorrect identifier

Module 9262 Context change between spawn processes

You can call Module 9262 only in a submit job or spawn job. The module switches the context to another PLC process or submit queue if such a process exists and is not waiting for an event or for the expiration of a dwell time. Markers 4200 to 4202 and 4204 have undefined changes after the module call.

Call:
CM 9262

Error code:

Marker	Value	Meaning
M4203	0	Context was changed
	1	Error code in W1022
W1022	30	Module was not called in a spawn or submit job

Module 9263 Interrupt a spawn process for a defined time

You can call Module 9263 only in a submit job or spawn job. The module interrupts the calling process for at least the specified time. If other processes or the submit queue are ready to run, the iTNC changes the context to one of these processes. Markers 4200 to 4202 and 4204 have undefined changes after the module call. The waiting period is interpreted as an unsigned number, so that negative values result in very long waiting periods.

Call:
PS B/W/D/K <>Waiting period in ms>
CM 9263

Error code:

Marker	Value	Meaning
M4203	0	Delay is active
	1	Error code in W1022
W1022	30	Module was not called in a spawn or submit job



9.13 Constants field (KF)

You can use the constants-field data type to access one of several constants, defined in tabular form, depending on the value of the index register X. You address it with KF <Name>[X], where <Name> is a label indicating the beginning of the constants field. Constants fields must be introduced with the label KFIELD <Name>. This is followed by any quantity (other than zero) of constants and the end label ENDK. Constants fields can only be programmed where the program has previously been concluded with an EM or JP statement. The name of constants fields corresponds to the rules for naming labels.

Addressing

Types of addresses:

- L KF <Name> [X], with $X \geq 0$:
The iTNC transfers the value of the constant defined by X in the constants field <Name>.
- L KF <Name> [X], with $X = -1$:
The iTNC transfers the length of the constants field <Name>.
- L KF <Name>:
The iTNC transfers the absolute address of the constants field <Name>. This is only worthwhile in conjunction with modules (e.g. Module 9200). You can also use this addressing in a constants field.

Example:

Function	STL
Access value field with $X = [0 \text{ to } 3]$.	L KF VAL_FIELD [X]
Assign one of the constants to word W0.	= W0
End of main program	EM
Define the constants field. Constant to be loaded with $X = 0$	KFIELD VAL_FIELD K+10 K+1 K\$ABC
Constant to be loaded with $X = 3$ End of the constants field.	K-100000 ENDK

The iTNC checks the access to constants fields in the same way as the writing access for indexed operands. X can assume only positive values from 0 to <Length of constants field -1>.

9.14 Program Structures

To design an easily understandable program, divide it into program sequences. Use labels (LBL) as well as conditional and unconditional jumps. If you use structured statements, the compiler creates the labels and jump commands. Remember that using these labels and jump commands reduces the number of available labels accordingly. You can nest structured statements in up to 16 levels. It is not possible to share levels.

Example:

Correct program structure	Incorrect program structure
IFT	IFT
⋮	⋮
WHILEF	WHILEF
⋮	⋮
ENDW	ENDI
⋮	⋮
ENDI	ENDW

The statements IFT, IFF, WHILET, WHILEF, ENDW, UNTILT and UNTILF require a valid gating result in the logic accumulator. They conclude the sequence of gating operations. The statements ELSE, ENDI and REPEAT require that all previous operations sequences have been concluded.

9.14.1 IF ... ELSE ... ENDI structure

The IF ... ELSE ... ENDI structure permits the alternative processing of two program branches depending on the value in the logic accumulator. The ELSE branch is not mandatory. The following commands are available:

- IFT (IF LOGIC ACCU TRUE):
Following code only if logic accumulator = 1
- IFF (IF LOGIC ACCU FALSE):
Following code only if logic accumulator = 0
- ELSE (ELSE):
Following code only if IF is not fulfilled
- ENDI (END OF IF STRUCTURE):
End of the IF structure

Function	STL
Load input I0 into the logic accumulator	L I0
Run the following code if logic accumulator = 1	IFT
Program code for I0 = 1	⋮
Run the following code if logic accumulator = 0; command can be omitted	ELSE
Program code for I0 = 0, can be omitted	⋮
End of the conditional processing	ENDI

9.14.2 REPEAT ... UNTIL structure

The REPEAT ... UNTIL structure repeats a program sequence until a condition is fulfilled. Under no circumstances can you wait with this structure in the cyclic PLC program for the occurrence of an external event! The following commands are available:

- REPEAT (REPEAT):
Repeat the program sequence from here.
- UNTILT (UNTIL TRUE):
Repeat the sequence until the logic accumulator = 1.
- UNTILF (UNTIL FALSE):
Repeat the sequence until the logic accumulator = 0.

The iTNC runs a REPEAT ... UNTIL loop at least once!

Function	STL
Assign the content of the logic accumulator to marker 100; conclusion of the previous commands	= M100
Repeat the following program code	REPEAT
Program code to be run	⋮
Load the index register	L X
Check the index register	>= K100
Repeat until X >= 100	UNTILT

9.14.3 WHILE ... ENDW structure

The WHILE ... ENDW structure repeats a program sequence if a condition is fulfilled. Under no circumstances can you wait with this structure in the cyclic PLC program for the occurrence of an external event! The following commands are available:

- **WHILET (WHILE TRUE):**
Run the sequence if logic accumulator = 1.
- **WHILEF (WHILE FALSE):**
Run the sequence if logic accumulator = 0.
- **ENDW (END WHILE):**
End of the program sequence, return to the beginning

The iTNC runs a WHILE ... ENDW loop only if at the beginning the WHILE condition is fulfilled. Before the ENDW statement you must reproduce the condition for execution. For the WHILE ... ENDW structure the iTNC generates two internal labels. The condition can also be produced in a way different from before the WHILE statement!

Function	STL
	⋮
Load marker 100 into the logic accumulator; create condition for 1st WHILE scan.	L M100
Run the following code if logic accumulator = 1	WHILET
Program code for logic accumulator = 1	⋮
Produce the condition of repeated execution: Load marker 101 in the logic accumulator and gate the content of marker M102 with AND.	L M101 A M102
Jump back to the WHILE request	ENDW

9.14.4 CASE branch

Indexed module call (CASE)

Syntax: CASE (CASE OF)

Operands: B/W

Action:

Selects a certain subroutine from a list of module calls (CM). These CM commands must follow the CASE statement immediately and are numbered internally in ascending order from 0 to a maximum of 127. The content of the operand (B, W) addresses the desired module. Subsequent entries in the jump table (CM) must have addresses at least four bytes higher than the previous entry.

Example:

See ENDC command.



**End of indexed
module call (ENDC)**

Syntax: ENDC (ENDCASE)

Operands: None

Action:

Use the ENDC command in connection with the CASE command. It must come immediately after the list of CM commands.

Example:

Function	STL
Case command and operand; the internal address of the desired module must be saved in the operand	CASE B150
Call module if operand content = 0 Internal addressing from 0 to max. 127	CM 100
Call module if operand content = 1	CM 200
Call module if operand content = 2	CM 201
Call module if operand content = 3	CM 202
Call module if operand content = 4	CM 203
Call module if operand content = 5	CM 204
Call module if operand content = 6	CM 300
End of the CASE statement	ENDC



9.15 Linking Files

You can store the source code of the PLC program in several files. To manage these files, use the following commands:

- USES
- GLOBAL
- EXTERN

These statements must be located at the beginning of your PLC program—i.e., before the first PLC command. With the USES statement you link another file into the program. The GLOBAL statement supplies a label from its own file for an entry that can be used by all other files. The EXTERN statement provides a label that is defined in another file and is identified there with GLOBAL. You can then call this label from the active file. You can dramatically improve the transparency of your program by dividing your source code by function into individual groups and then save these groups in individual files. The number of labels is not limited. You can link up to 256 files to one program. The total size is only limited by the available memory. If the memory is exceeded the error message **System memory overflow** appears. With multiple files, the main program in the directory must have the "M" status flag. This can be done in RAM by using the PLC program function "COMPILE" once and choosing the main program in the file window.



9.15.1 USES Statement (USES)

Syntax: USES <file name>

Operands: None

Action:

You can use the USES statement in the main program to link other files. Files that are linked with USES can themselves also use the statement to link further files. It is also permissible to use the USES statement to link a single file to several other files. The code for this file is generated only once. The USES statement requires a file name as an argument. The USES statement only links a file; it does not run the file's program code. The USES statement cannot be compared with a CM statement. The linked files must therefore contain individual modules that you can then call with the CM statement.

Example:

```
USES PLCMOD1
USES EPRUPG
USES RAMPLC
```

Linking of files:

Function	STL
Main program	PLCMAIN.PLC
Link the file for spindle control.	USES SPINDLE.PLC
Link the file for tool change.	USES TCHANGE.PLC
Program code	⋮

Function	STL
File for spindle control	SPINDLE.PLC
Integrate file with general subroutines.	USES PLCUPG.PLC
Program code	⋮

Function	STL
File for tool change	TCHANGE.PLC
Integrate file with general subroutines.	USES PLCUPG.PLC
Program code	⋮

Function	STL
File with general subroutines	PLCUPG.PLC
Program code	⋮

9.15.2 GLOBAL statement (GLOBAL)

Syntax: GLOBAL <Label, declaration beyond the file boundary>

Operands: None

Action:

There is no limit to the number of labels in each file linked with USES. To enable a module that was defined in a file to be called from another file, you must declare the module to be global. This is done by entering the GLOBAL statement at the beginning of the file. You can set labels globally only if they are defined with LBL (and not with KFIELD!) later on in the program.

The main program must not contain any GLOBAL definitions. A single label cannot be declared global by more than one module. However, a name that is declared global in file A can be used again locally in file B. The number of labels is not limited.

9.15.3 EXTERN Statement (EXTERN)

Syntax: EXTERN <Label, a module from another file can now be called with a CM command>

Operands: None

Action:

To enable a label in one file to access modules that other files have declared as GLOBAL, you must declare the label with EXTERN. You must write the EXTERN statement at the beginning of the file. In the program code you can then jump to this label with the commands CM, CMT and CMF.

The following functions are not permitted with external labels:

- JP, JPF, JPT
- Access to a constants field
- Linking a CM statement in a CASE branch

The name of the external label cannot be used again as a local label in the same file.



9.16 PLC Modules

A number of PLC modules are available for PLC functions that are very difficult or even impossible to perform with PLC commands alone. You will find descriptions of these modules under the corresponding functions. (See "Overview of modules" on page 583.)

If the iTNC does not execute a module correctly, it sets M4203 and transmits an error code in W1022. You can then evaluate M4203 and W1022 in order to display an error message. If desired, the errors of PLC modules can be logged in a special log book (see "Logs" on page 1286).

		Set	Reset
M4203	Error status for PLC module	NC	NC/PLC
W1016	PLC module that was last processed incorrectly	NC	NC
W1022	Error status of the last called PLC module	NC	NC

9.16.1 Markers, bytes, words, and double words

Module 9000/9001 Copy in the marker or word range

Modules 9000 (markers) and 9001 (byte/word/double) copy a block with a certain number of markers or bytes, beginning with the start address, to the specified target address. For Module 9001 the length should always be defined in bytes.

The iTNC copies sequentially, beginning with the first memory cell. Therefore the function is not ensured if the source block and the target block overlap and the source block begins at a lower address than the target block. In this case the iTNC overwrites the overlapping part of the source block before the copying process.

Call:

PS B/W/D/K <>Number of the 1st marker in source block>

PS B/W/D/K <>Number of the 1st marker in target block>

PS B/W/D/K <>Length of block in markers>

CM 9000

PS B/W/D/K <>Number of the 1st word in source block>

PS B/W/D/K <>Number of the 1st word in target block>

PS B/W/D/K <>Length of block in markers>

CM 9001

Error code:

Marker	Value	Meaning
M4203	0	Markers, bytes, words, or double words were copied
	1	Error code in W1022
W1022	1	Operand address invalid
	2	Address too high or block too long
	4	Programmed source or target block too long

Module 9010/9011/9012 Read in the word range

From the specified location in the word memory the iTNC reads a byte, word or double word and returns it as an output quantity to the stack. Indexed reading is possible by specifying a variable as designation of the memory location.

Call:

PS B/W/D/K <>Address of the byte to be read>
 CM 9010 ; READ BYTE
 PL B <>Target address for byte that was read>

PS B/W/D/K <>Address of the word to be read>
 CM 9011 ; READ WORD
 PL B <>Target address for word that was read>

PS B/W/D/K <>Address of the double word to be read>
 CM 9012 ; READ DOUBLE WORD
 PL B <>Target address for double word that was read>

Error code:

Marker	Value	Meaning
M4203	0	Byte was read
	1	Error code in W1022
W1022	3	Invalid address was programmed
	5	Module 9011: Specified address is not a word address Module 9012: Specified address is not a double word address

Example of Module 9010

Initial state:
 Byte B10 = 35 (address)
 Byte B35 = 80 (byte to be read)
 Byte B100 = ?

Function	STL	Accumulator content (dec)	Data stack (dec)
Save the address (B10) of the byte to be read from the word accumulator in the data stack	PS B10	35	35
Read byte B35 and save in the data stack	CM 9010		80
Save data stack in byte B100.	PL B100	80	80



Module 9020/9021/9022 Write in the word range

The iTNC writes the given byte, word or double word to the defined location in the word memory. Indexed writing is possible by specifying a variable as designation of the memory location.

Call:

PS B/W/D/K <>Address of the byte to be written>

PS B/W/D/K <>Byte to be written>

CM 9020 ; WRITE BYTE TO ADDRESS

PS B/W/D/K <>Address of the word to be written>

PS B/W/D/K <>Word to be written>

CM 9021 ; WRITE WORD TO ADDRESS

PS B/W/D/K <>Address of the double word to be written>

PS B/W/D/K <>Double word to be written>

CM 9022 ; WRITE DOUBLE WORD TO ADDRESS

Error code:

Marker	Value	Meaning
M4203	0	Byte was written
	1	Error code in W1022
W1022	3	Invalid address was programmed
	5	Module 9021: Specified address is not a word address Module 9022: Specified address is not a double word address

Example of Module 9020

Initial state:

Byte B10 = 35 (address)

Byte B100 = 120 (byte to be written)

Byte B35 = ?

Function	STL	Accumulator content (dec)	Data stack (dec)
Save the address (B10) of the byte to be written from the word accumulator in the data stack	PS B10	35	35
Save byte B100 from the word accumulator in the data stack	PS B100	120	120
Write data stack to byte B35	CM 9020	120	

9.16.2 Number conversion

Module 9050 Conversion from binary → ASCII

Module 9050 converts a binary numerical value consisting of a mantissa and exponent to base 10 into an ASCII-coded decimal number and saves it as a string in the specified address. The exponent refers to the least significant place of the number. The iTNC detects a negative number when the mantissa corresponds to a negative number in the notation as a two's complement. The iTNC sets an algebraic sign only before negative numbers. The control does not convert trailing zeros after the decimal point or leading zeros before the decimal point. The iTNC writes the string left-aligned in the string address that you specify.

Constraints:

The decimal character is defined by Machine Parameter MP7280 as a comma (MP7280 = 0) or a period (MP7280 = 1).

Call:

PS B/W/D/K <>Mantissa of the number to be converted>

PS B/W/D/K <>Exponent to base 10 of the value>

PS B/W/D/K <>String address in which the iTNC saves the ASCII-coded decimal number>

CM 9050

Error code:

Marker	Value	Meaning
M4203	0	Number was converted
	1	See W1022 for error code
W1022	2	Invalid string address or invalid exponent



Module 9051 Conversion from binary → ASCII

Module 9051 converts a binary numerical value into an ASCII-coded decimal number in the specified format and saves it as a string in the specified address. The number is interpreted as a two's complement. For algebraically unsigned notation, the iTNC converts the absolute amount of the number without putting a sign before the string. For algebraically signed notation, the iTNC sets an algebraic sign (+ or -) in front of the string in any event. For notation in inches, the number is divided by 25.4 before conversion. If the number has more decimal places than the total that you have specified for the number of places before and after the decimal point, then the iTNC omits the most highly significant decimal places. In right-aligned notation leading zeros before the decimal point are replaced by blanks; in left-aligned notation they are suppressed. Trailing zeroes after the decimal point are always converted.

Constraints:

The decimal character is defined by Machine Parameter MP7280 as a comma (MP7280 = 0) or a period (MP7280 = 1).

Call:

PS B/W/D/K <>Numerical value to be converted>

PS B/W/D/K <>Display modes, bit-encoded>

Bit 1/0: Format

00: Sign and number left-aligned

1: Sign left-aligned, number right-aligned

10: Sign and number right-aligned

11: Not permitted

Bit 2: Display converted to INCH

Bit 3: Display with sign

PS B/W/D/K <>Number of places after the decimal point>

PS B/W/D/K <>Number of places before the decimal point>

PS B/W/D/K <>String address in which the iTNC saves the ASCII-coded decimal number>

CM 9051

Error code:

Marker	Value	Meaning
M4203	0	Number was converted
	1	Error code in W1022
W1022	2	Invalid string address, invalid display mode or invalid number of places before or after the decimal point

Module 9052 Conversion of ASCII numbers → binary

Module 9052 converts an ASCII-coded decimal number (possibly with decimal places) into a signed number and an exponent to the base of 10. You must assign the ASCII-coded decimal number to one of the string memories. If the number has no algebraic sign, the iTNC interprets it as a positive number and accepts both a point and a comma as decimal character. If the full extent of the mantissa cannot be represented in a double word, then the last places are omitted and the exponent is corrected accordingly. If possible, the iTNC adjusts the exponent so that it corresponds to the ASCII notation.

Call:

PS B/W/D/K <>String address in which the ASCII-coded decimal number is saved>

CM 9052

PL B/W/D <>Numerical value>

PL B/W/D <>Exponent to the base of 10 of the value>

Error code:

Marker	Value	Meaning
M4203	0	Number was converted
	1	See W1022 for error code
W1022	2	Invalid string address or string contains none or too many characters

Module 9053 Conversion from binary → ASCII/hexadecimal

Module 9053 converts blocks of binary values from the word-marker range into a string of ASCII-coded hexadecimal numbers. The iTNC reads the specified number of bytes from the word address that you have specified and converts it to a hexadecimally coded ASCII string. Each byte produces two characters in the string memory.

Call:

PS B/W/D/K <>Word address from which the binary values are saved>

PS B/W/D/K <>String address in which the iTNC saves the hexadecimal numbers>

PS B/W/D/K <>Number of data bytes>

CM 9053

Error code:

Marker	Value	Meaning
M4203	0	Number was converted
	1	See W1022 for error code
W1022	1	Too many data bytes
	2	Invalid string address
	4	Invalid word address



Module 9054 Conversion from ASCII/hexadecimal → binary

Module 9054 converts strings of ASCII-coded hexadecimal values into a block of binary values in the word-marker range. The string in the specified string memory is interpreted as a sequence of ASCII-coded hexadecimal numbers and converted into a block of corresponding binary bytes. Two ASCII characters produce one binary byte. The iTNC saves the binary block beginning at the specified address in the word-marker range.

Call:

```
PS   B/W/D/K <>String address in which the hexadecimal value is saved>
PS   B/W/D/K <>Word address from which the iTNC saves the binary
      values>
CM   9054
```

Error code:

Marker	Value	Meaning
M4203	0	Number was converted
	1	See W1022 for error code
W1022	2	Invalid string address
	11	Invalid word address
		Odd number of characters in the string or a character that cannot be interpreted as hexadecimal

Example

Initial state:

SO = „63“

BO = 99

Function	STL	String accumulator (characters), data stack [bits]
Push string address S0 onto the data stack.	PS K+0	63
Push word address B0 onto the data stack.	PS B0	99
Conversion of the two ASCII characters 6 and 3 into the binary number 99.	CM 9054	01100011

9.17 Python

Software option #46 "Python OEM Process" is an effective tool for you to use an object-oriented high-level programming language in the control (PLC).

"Python OEM Process" can be used for machine functions and complex calculations, as well as to display special information. It is especially useful for the implementation of user-specific or machine-specific solutions, regardless of whether special algorithms or interfaces for special functions have to be created.

The applications created can be displayed by the PLC in the familiar PLC windows or in separate freely definable windows that can be expanded to the TNC's full screen size.

The system requires at least 512 MB of RAM. The RAM of the MC can be checked in the **herosdiagnose.txt** file. The RAM of the MC is listed in [kB] under the **Total Memory heading**.

Proceed as follows to create the **herosdiagnose.txt file**:

- ▶ While in the Programming and Editing operating mode, press the MOD key.
- ▶ Press the DIAGNOSIS soft key, and then the HEROS DIAGNOSIS soft key.

A separate documentation exists for Python OEM Process, which you can download free of charge from the HEIDENHAIN HESIS-Web including Filebase under "PC software > Python" together with simple examples and additional PC tools for debugging and for the development of user interfaces.

Please note the Python documentation for the respective NC software. It is also available on HEIDENHAIN HESIS-Web including Filebase, under "PC Software > Python".



10 Data Interfaces

10.1 Introduction

In addition to their Central Processing Unit (CPU), computer systems usually include various peripheral devices.

A CPU is, for example:

- PC
- Control

Peripheral devices include:

- Printers
- Scanners
- External storage devices, such as hard disks, floppy-disk drives or USB memory sticks
- Other computer systems

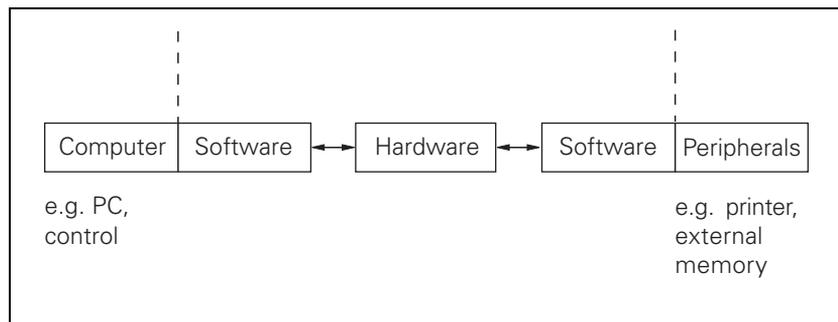
A data interface makes it possible for the CPU and its peripheral devices to communicate.

The interfaces, which consist of physical links between the computer system and the peripherals, need a transmission line and appropriate software in order to transfer data between the individual units.

Standard interfaces include:

- Ethernet
- USB 1.1
- RS-232-C/V.24 or
- RS-422/V.11

The relationship between hardware and software, which fully defines an interface, is illustrated by the following diagram:



The hardware in the diagram covers all the physical components, such as

- Circuit design
- Pin layout
- Electrical characteristics

The software is the operating software, which includes, for example, the drivers for the output modules.

10.1.1 Principles of data transfer

The term "data" is used to describe all of the information that the computer is capable of collecting and processing.

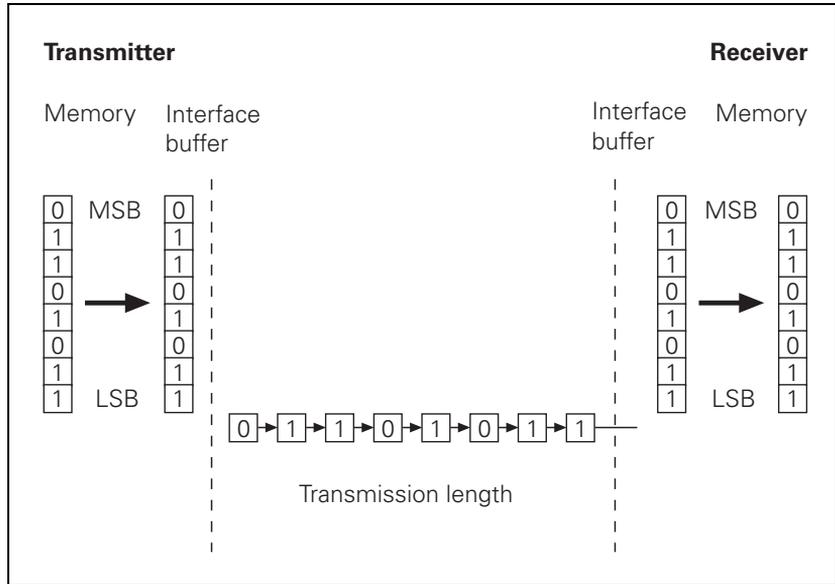
Serial/Parallel

Data can be transmitted in either serial or parallel format.

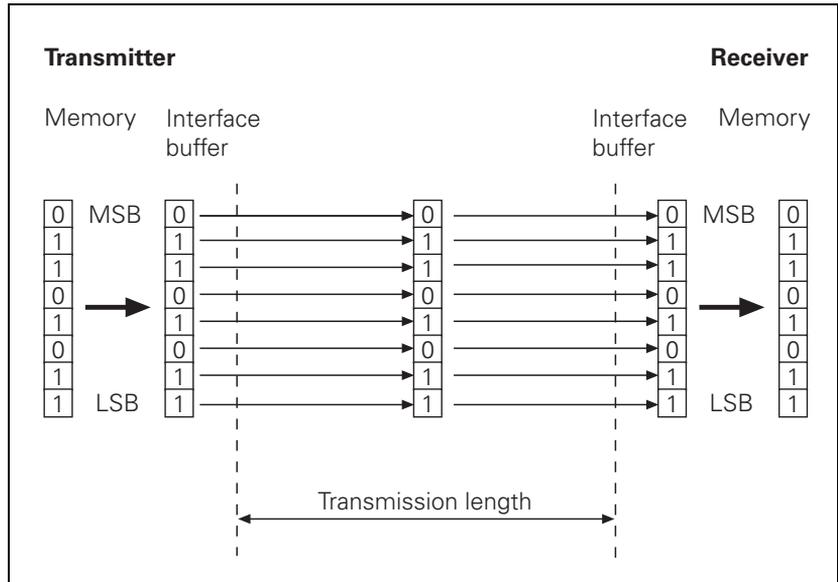
Basically, data is coded in the computer system, e.g. as bytes (8 bits), and supplied to the interface in parallel.

In the case of serial data transmission, the parallel information from the computer system has to be converted into a serial data flow by a USART (Universal Synchronous/Asynchronous Receiver/Transmitter).

The receiver accepts the serial data flow and converts it back again into parallel information:



For parallel data transmission, the interface needs line drivers, e.g. a 36-pin ribbon cable, instead of the USART illustrated above.



Advantages of serial data transmission:

- Economical
- Ideal for covering long distances

Disadvantages of serial data transmission:

- Slow

Advantages of parallel data transmission:

- Fast

Disadvantages of parallel data transmission:

- Somewhat more expensive

Data coding

A common code for data transfer is the ASCII code (American Standard Code for Information Interchange), which codes all characters with seven bits. This means that, in all, $2^7 = 128$ characters are coded.

Example:

The letter "z" is represented by the following combination of bits:

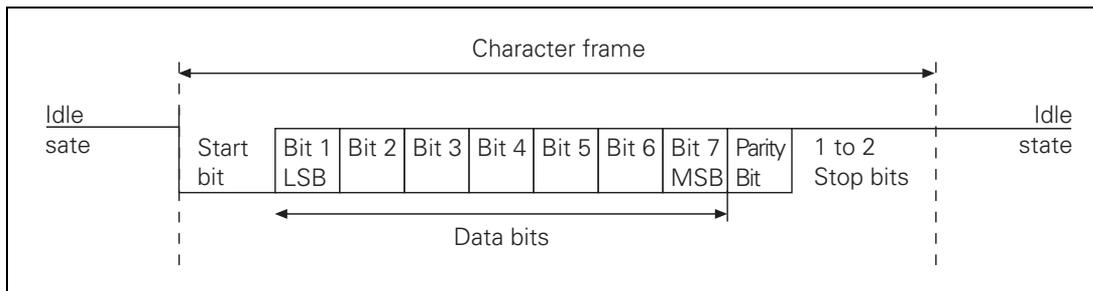
1 1 1 1 0 1 0 = 122 dec = 7A hex

When the letter "z" is transmitted via a serial interface, the appropriate characters are sent one after the other.

Synchronization

A synchronization process ensures that the receiver correctly detects the first character of a transmission.

With an asynchronous character frame, the transmission of a data word can begin at any time, starting from an idle state.



A start bit is transmitted before each data word. If the first bit of the data word had the same value as the idle state, the receiver would not notice any difference from the idle state.

After the start bit has been sent, the data word is transmitted, bit by bit, starting with the LSB (Least Significant Bit). The MSB (Most Significant Bit) of the data word is followed by the so-called parity bit in order to detect transmission errors.

The character frame is concluded by one or two stop bits. The stop bits enable the receiver to recognize the transmitter again before the start of the next character.

Synchronization is repeated before each character and applies for one character frame.

Transmission reliability: Parity bit

With an asynchronous character frame, transmission errors can be detected by using the parity bit.

The parity bit can take three different forms.

- No parity check (no parity): Error detection is dispensed with.
- Even parity:
The transmitter counts bits with a value of one. If the number is odd, the parity bit is set to one, otherwise it is cleared to zero. The sum of set data bits and the parity bit is therefore always even. Upon receiving a word, the receiver counts all of the set bits, including the parity bit. If the count is odd, there is a transmission error and the data word must be repeated, or an error message will be displayed.
- Odd parity:
The parity bit is so chosen by the transmitter that the total number of all the set bits is odd. An error will thus be detected if the receiver observes an even number of set bits in its evaluation.

Example

The letter "z" corresponds to the bit sequence: 1 1 1 1 0 1 0

Parity bit

- With even parity = 1
- With odd parity = 0

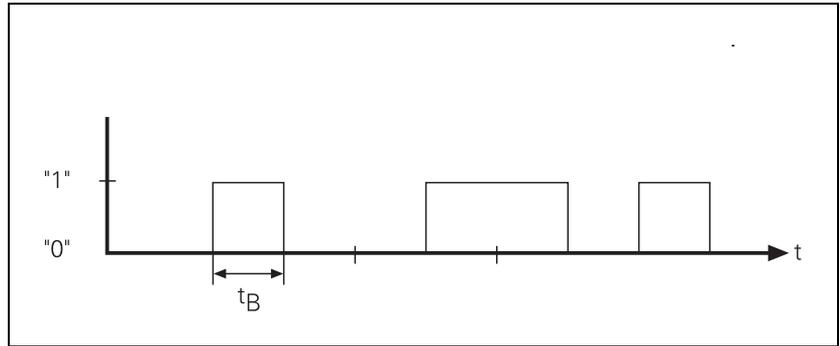
Data transfer rate

The data transfer rate is given in bits per second.

Common transfer rates are:

110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 bps

The time taken to transmit one bit (t_B) can be calculated from the transfer rate:



$$t_B = \frac{1}{\text{transfer rate (bit/s)}}$$

For example, a transfer rate of 19 200 bps will have a bit duration of $t_B = 52.083 \mu\text{s}$.

$$t_B = \frac{1}{19200 \text{ (Bit/s)}} = 52.083 \mu\text{s}$$

The number of characters transmitted per second can be calculated from the transfer rate and the transmission format:

$$\text{characters per second} = \frac{\text{transfer rate (bit/s)}}{\text{numbers of bit per characters}}$$

Example:

With a transmission format of one start bit, 7 data bits, two stop bits and a data transfer rate of 300 bps, exactly 30 characters per second will be transmitted.

$$\text{characters per second} = \frac{300 \text{ (Bit/s)}}{1 + 7 + 2} = 30$$

10.1.2 Data transfer check: Handshaking

By handshaking, two devices control data transfer between them. A distinction is made between software handshaking and hardware handshaking.

You can choose either of the two procedures:

Hardware handshaking

Data transfer is controlled by electrical signals. Information, such as Clear to Send (CTS), Request to Send (RTS), "Start transmission" and "Stop transmission" is passed on by the hardware.

Example:

When a computer is to transmit a character, it checks the CTS signal line to see whether it is active (ON). If it is, the character is transmitted.

Hardware handshaking requires:

- The data lines TxD and RxD (transmitted and received data)
- The RTS control line (switch transmitting unit on)
- The CTS signal line (Clear to Send)
- A ground connection

Software handshaking

Control of data transfer is achieved by control characters transmitted via the data line.

Example: XON/XOFF method with the RS-232-C/V.24 interface

The meaning XON is assigned to control character DC1 and the meaning XOFF to control character DC3. Before transmitting a character, the computer checks whether the receiving unit is transmitting the XOFF character. If it is, the computer delays transmission until it receives the character XON. XON indicates that the connected unit is ready to receive further characters.

Software handshaking requires:

- The data lines TxD and RxD (transmitted and received data)
- A ground connection

10.2 The Ethernet Interface of the iTNC

General information

You can integrate the iTNC into your plant's intranet or use a transposed cable to connect directly with a PC. The data transfer rate is dependent on the amount of traffic at the time on the net. For information on the pin layout: See "X26, X116: Ethernet interface RJ45-port" on page 325.

The iTNC requires an NFS server (Network File System) or a Windows PC (SMB = Server Message Block) as the remote station. It must operate according to the TCP/IP protocol principle.

OSI 7-layer model		iTNC
7	Application layer	NFS, SMB
6	Presentation layer	
5	Communications layer	
4	Transport layer	TCP protocol
3	Network layer	IP protocol
2	Data link layer	Ethernet card
1	Physical layer	

Before networking, the TNC must be properly configured. Please discuss the required settings with your network supervisor.

SMB signing

HEIDENHAIN does not support SMB signing, which is a further development from Microsoft as part of the CIFS (Common Internet File System). It is a method for authentication, and is intended to thwart a "Man in the middle" attack. In such an attack, computer C intrudes into a connection between computers A and B. It then forces computer A to send all packages intended for B to C instead, where the packages are modified before being passed on to B.

The CIFS is not included with the HEIDENHAIN SMB driver. HEIDENHAIN supports SMB, in the version familiar from Windows 2000, but not SMB signing.

X26, X116: Ethernet interface RJ45-port (100BaseT)

- Maximum cable length:
 - Unshielded: 100 m
 - Shielded: 400 m
- Network topology: Star configuration

This means a hub serves as a central node that establishes the connection to the other participants.



Danger

The Ethernet interfaces of the MC 6xxx comply with the requirements of PELV ("low voltage electrical separation") according to EN 61800-5-1 and are powered internally by 24 V NC. All devices connected to these Ethernet interfaces must comply with the requirements of SELV or PELV according to EN 61800-5-1.

10.3 Connecting the iTNC to the Network

10.3.1 Settings at the iTNC

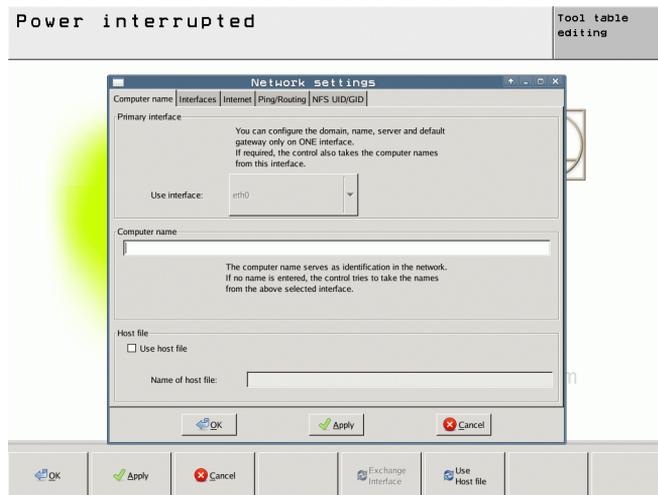
- ▶ Press the MOD key in the **Programming and Editing** operating mode and enter the code number NET123. Use the soft keys to select the required network options:

Soft key	Meaning
DEFINE NET	Settings on the iTNC 530 for networking. The network settings for connecting the iTNC 530 to a network were completely revised and are now divided into several pages of a window, which can be selected by tabs.

The settings of the non-primary Ethernet interface(s), the selection of the primary interface and the DHCP server settings are password protected. This protects the settings from unauthorized changes. The code number 807667 or the PLCPASSWORD defined in the OEM.SYS are required for these settings. The code number is entered using the **OEM authorization** button or soft key.

Or you can open the network settings as follows:

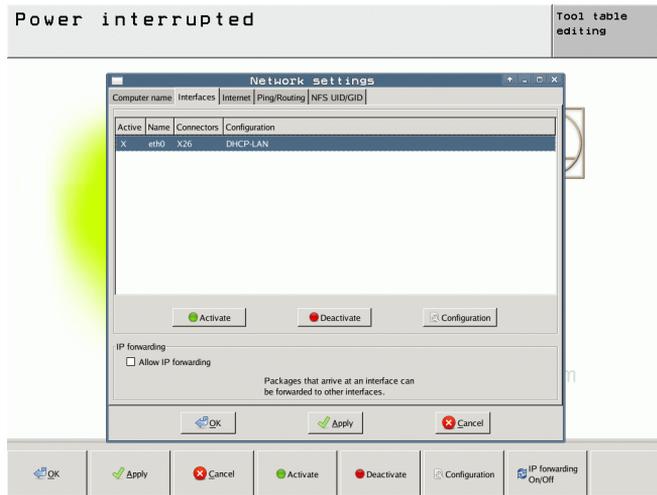
- ▶ Make the taskbar at the bottom of the screen visible by moving the mouse pointer over it.
- ▶ Press the green HEIDENHAIN button to open the JH menu.
- ▶ Select the Settings menu item.
- ▶ Select the Networks menu item.



Tab	Option	Meaning
Computer name		Settings on the iTNC 530 for networking. The name of the control as well as other data can be defined.
	Primary interface	Name of the Ethernet interface to be integrated in your company network. Only active if a second, optional Ethernet interface (MC 62xx) is available on the control hardware.
	Computer name	The iTNC registers itself in the network with this name, which you can also use to reach it (with TNCremoNT, for example). If you use a host-name server, you must enter the "Fully Qualified Host Name" here. If you do not enter a name here, the iTNC uses the so-called null authentication. If null authentication is used, the entries under UID, GID, DCM and FCM are ignored.
	Host file	<p>Only required for special applications: Name of a file in which the assignments of IP addresses to computer names are defined. Then you can enter the names instead of the IP addresses in the network settings. Example: DOMAIN = NET.A File NET.A: PC1 160.1.180.20 PC2 160.1.180.21 ...</p>

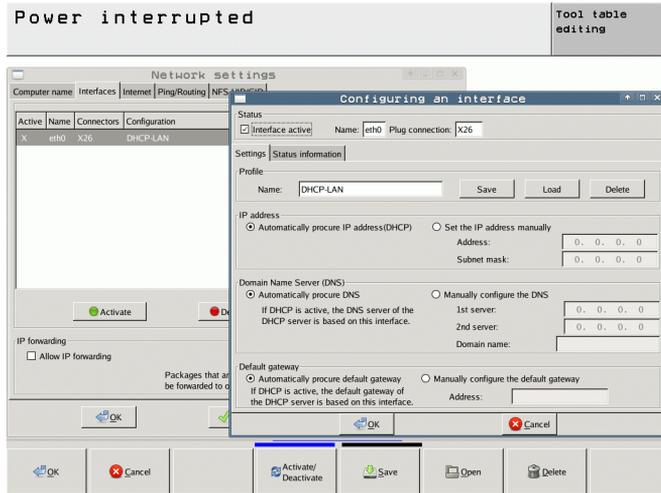


- ▶ Select the **Interfaces** tab in the configuration menu of the interface:



Tab	Option	Meaning
Interfaces		List of the active Ethernet interfaces with information as to whether the interface is active, the name, the connector and the IP configuration <ul style="list-style-type: none"> ■ Activate button: Activate the selected interface (the letter X appears in the Active column) ■ Deactivate button: Deactivate the selected interface (a dash (–) appears in the Active column) ■ Configuration button: Open the Configuration menu
	IP forwarding	This function must be inactive by default. Only activate this function if external access to the second, optional Ethernet interface of the control (available on the MC 62xx) via the TNC is necessary for diagnostic purposes.

- Press the **Configuration** button below the interface list to open the Configuration menu:



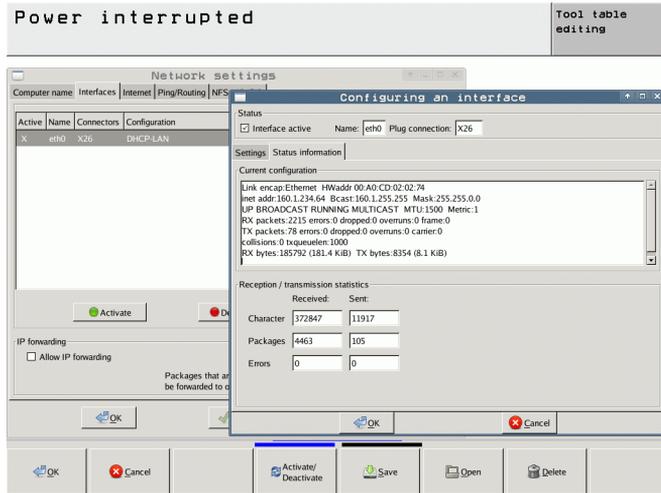
Option	Meaning
Status	<ul style="list-style-type: none"> ■ Interface active: Status of the selected Ethernet interface. ■ Name: Name of the interface you are currently configuring. ■ Plug connection: Number of the plug connection of this interface on the logic unit of the control.
Profile	<p>Here you can create or select a profile in which all settings shown in this window are stored. HEIDENHAIN provides two standard profiles:</p> <ul style="list-style-type: none"> ■ LAN-DHCP: Settings for the standard TNC Ethernet interface, should work in a standard company network. ■ MachineNet: Settings for the second, optional Ethernet interface; for configuration of the machine network (available on the MC 62xx). <p>Press the corresponding buttons to save, load and delete profiles.</p>
IP address	<ul style="list-style-type: none"> ■ Automatically procure IP address: The control is to procure the IP address automatically from the DHCP server. ■ Set the IP address manually: Manually define the IP address and subnet mask. Enter as four decimal numbers separated by points (dotted-decimal notation, for example: 160.1.180.20 and 255.255.0.0). Your network specialist determines both addresses of the iTNC.

Option	Meaning
Domain Name Server (DNS)	<ul style="list-style-type: none"> ■ Automatically procure DNS: The control is to automatically procure the IP address of the DNS server. ■ Manually configure the DNS: Manually enter the IP addresses of the servers and the domain name. Your network specialist determines both entries.
Default gateway	<ul style="list-style-type: none"> ■ Automatically procure default gateway: The control is to automatically procure the default gateway. ■ Manually configure the default gateway: Manually enter the IP address of the default gateway. Your network specialist determines the address.

▶ Press the **OK** button to confirm the settings, or press the **Cancel** button to reject the settings.



- ▶ Select the **Status Information** tab in the configuration menu of the interface:

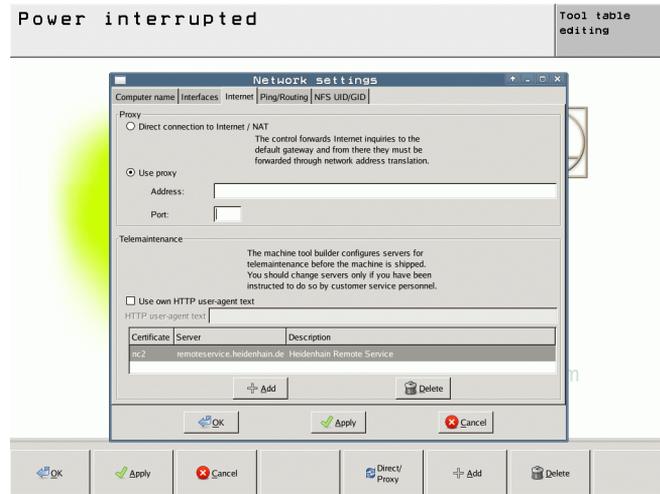


Option	Meaning
Current configuration	Information for your network specialist about the current configuration of the Ethernet interface.
Reception/ Transmission statistics	Information for your network specialist about the received and transmitted characters, packets and errors.

Other possible settings

■ Network settings for Internet access

With HeROS 5, the control can access the Internet. After entering the code number NET123, you will find the settings for Internet access on a tab in the Network settings window. These settings are used for remote maintenance and as default settings by the control's web browser (Firefox). However, you can also use your own program-specific settings to configure the web browser.

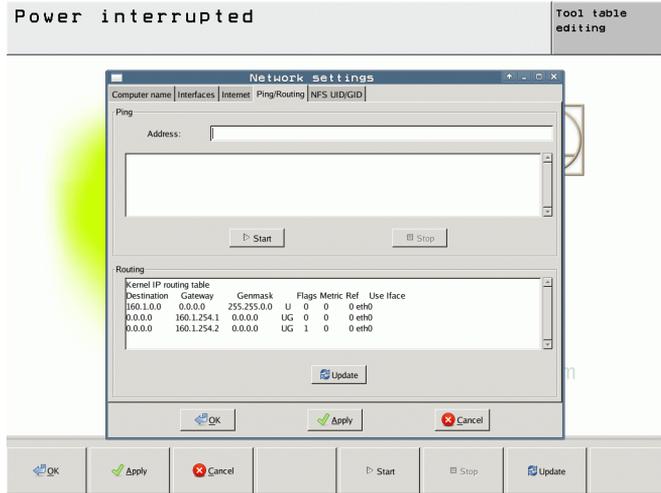


Tab	Option	Meaning
Internet		Settings for Internet access (proxy) and remote maintenance:
	Proxy	<ul style="list-style-type: none"> ■ Direct connection to Internet / NAT The control forwards Internet inquiries to the default gateway and from there they must be forwarded through network address translation (e.g. if a direct connection to a modem is available). ■ Use proxy These settings are made by your network specialist. <ul style="list-style-type: none"> • Address: Enter the address of the Internet router in the network • Port: Enter the port number
	Telemaintenance	Function reserved for future expansion

The network in which the control is connected must be protected by the integrated browser Firefox against the downloading of viruses and malware. The same applies to access to contents of the Internet or other networks. The protective measures for this network are the responsibility of the machine tool builder and the respective network administrator, for example through a firewall.

■ **PING function**

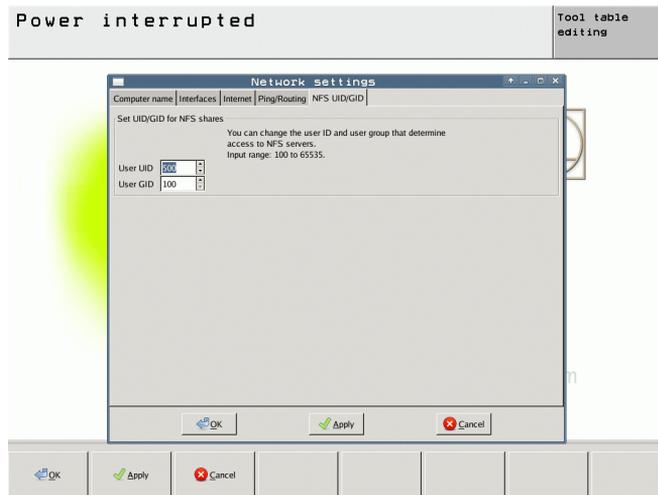
Press the PING soft key to call this function.



Tab	Option	Meaning
Ping/ Routing		Entry of ping and routing settings:
	Ping	<p>In the Address: field, enter the IP number for which you want to check the network connection. Input: Four numerical values separated by points, e.g. 160.1.180.20. As an alternative, you can enter the name of the computer whose connection you want to check.</p> <ul style="list-style-type: none"> ■ Start button: Start the test. The control shows status information in the Ping field. ■ Stop button: Conclude the test.
	Routing	<p>For network specialists: Status information of the operating system for the current routing process.</p> <ul style="list-style-type: none"> ■ Update button: Update routing.

■ Network settings UID/GID

With HeROS 5, too, the configuration possibilities for UID and GID will be available on a tab in the Network settings window after you have entered the code number NET123:



Tab	Option	Meaning
NFS UID/ GID		Entry of user ID and user group for access to the NFS server
	User ID:	Definition of which user identification the control uses to access files in the network. Your network specialist determines this value.
	Group ID:	Definition of which group identification the control uses to access files in the network. Your network specialist determines this value.

Configuring an interface

To log your control onto a standard network, proceed as follows:

- ▶ Enter the code number NET123 in the PROGRAMMING AND EDITING operating mode.
- ▶ Press the DEFINE NET soft key
- ▶ In the **Computer name** field, enter the name under which the control is to identify itself in the network.
- ▶ Select the **Interfaces** tab. Select the name of the interface (**eth0**), connector **X26** and press the **Configuration** button.
- ▶ In the **Settings** of the interface under **Profile**, press the **Load** button.
- ▶ Select the profile **LAN-DHCP**, and press **OK** to confirm your selection.
- ▶ All other settings can be left at "Automatically procure", unless required otherwise for your network.
- ▶ To leave the **Settings**, press the **OK** button.
- ▶ Then press the **OK** button again to confirm your changes in the **network settings**.
- ▶ Then restart the control with the **Restart** button.

Mount settings

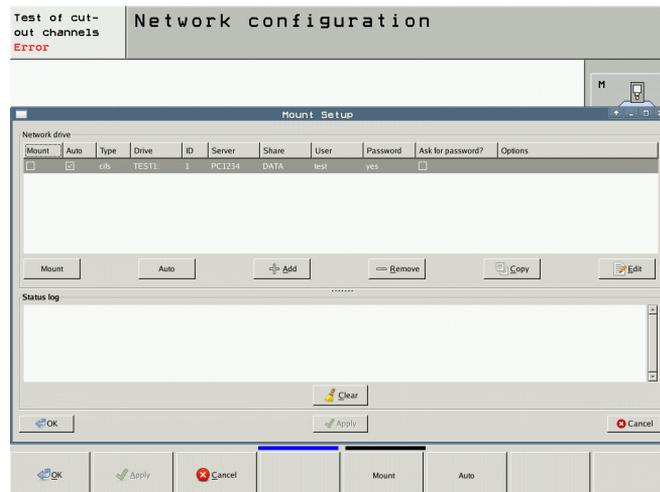
The configuration options for connecting a network drive after entering the code number NET123 have been adapted to the display of the network settings.

- ▶ Press the MOD key in the **Programming and Editing** operating mode and enter the code number NET123. Use the soft keys to select the required network options:

Soft key	Meaning
DEFINE MOUNT	Definition of the devices in the network that can be addressed from the iTNC. The network settings for connecting the iTNC 530 to a network are divided into several pages of a window, which can be selected by tabs.

Or you can open the Mount settings as follows:

- ▶ Make the taskbar at the bottom of the screen visible by moving the mouse pointer over it.
- ▶ Press the green HEIDENHAIN button to open the JH menu.
- ▶ Select the Settings menu item.
- ▶ Select the Share menu item.



Screen button / soft key	Screen button / soft key	Meaning
The following tabs are available in the Networks window:		
	Mount	Shows whether a drive is connected (indicated by a check mark).
	Auto	Shows whether a drive is connected automatically if a network connection exists (indicated by a check mark). For networks that are to be connected automatically, the iTNC attempts to do so for up to a minute after booting. In this way, the automount succeeds even when a device in the network (e.g. a switch in the machine) does not start up until the machine switches on.
	Type	Type of network share: <ul style="list-style-type: none"> ■ CIFS/SMB: Windows share or Samba server share ■ NFS: Unix share
	Drive	The device name entered here is displayed at the iTNC in the program management for the mounted network, e.g. WORLD :
	ID	Information for distinguishing between several identical network drives that are to be accessed with different user names, for example.
	Server	Network name or IP address of the computer or server to which the iTNC logs on.
	Share	Share name of the directory to which the iTNC logs on.
	User	User name with which the iTNC logs on.
	Password	Password with which the iTNC logs on.
	Ask for password?	Shows whether the password is requested when the drive is connected (indicated by check mark).

Screen button / soft key	Screen button / soft key	Meaning
	Options	<p>Shows special network settings, for example:</p> <p>Options that concern the file system type nfs: Options are entered without spaces, separated only by commas. Pay attention to capitalization.</p> <p>Options:</p> <p>rsize: Packet size in bytes for data reception. Input range 512 to 8192</p> <p>wsiz: Packet size in bytes for data transmission. Input range 512 to 8192</p> <p>timeo: Time in tenths of a second after which the iTNC repeats a Remote Procedure Call not answered by the server. Input range 0 to 100000. If there is no entry, the standard value 7 is used. Use higher values only if the iTNC must communicate with the server over more than one router. Your network specialist determines this value.</p> <p>soft: The Remote Procedure Call is repeated until the NFS server answers. If soft is entered, it is not repeated.</p> <p>Options that concern the file system type smb: Options are entered without spaces, separated only by commas. Pay attention to capitalization.</p> <p>Options:</p> <p>ip=: IP address of the Windows PC to which the iTNC is to be connected</p> <p>username: User name with which the iTNC should log in</p> <p>workgroup: Workgroup under which the iTNC should log in</p> <p>password: Password with which the iTNC should log in (up to 80 characters)</p>



Screen button / soft key	Screen button / soft key	Meaning
The following screen buttons are available in the Network window:		
Connect		The selected network drive is connected.
Auto		Activate/Deactivate automatic connection of the selected network drive.
Add		Configure and add a new network drive:
	Define Name	<ul style="list-style-type: none"> ■ Drive name: Enter a volume name for the network connection. The device name entered here is displayed at the iTNC in the program management for the mounted network. Only capital letters are allowed, and the name must end with a colon (":"). ■ Volume ID: Information for distinguishing between several identical network drives that are to be accessed with different user names, for example.
	Type	<p>Choose the type of network share.</p> <ul style="list-style-type: none"> ■ CIFS/SMB: Windows share or Samba server share ■ NFS: Unix share
	Define Share	<ul style="list-style-type: none"> ■ Server name: Network name or IP address of the computer or server to be connected. Slash ("/") or colon (":") are not allowed. ■ Share name: Colon (":") is not allowed. <ul style="list-style-type: none"> • Connecting via NFS: UNIX directory that is to be mounted. Pay attention to capitalization when entering the path name. • Connecting via CIFS: Enter the share name of the directory.
	Windows users	<ul style="list-style-type: none"> ■ Windows User: The user name and password are required for connecting to a Windows network. Enter your personal Windows user name or a generally valid access to the network. ■ Ask for password? This setting defines whether the password is requested when the drive is connected (indicated by check mark). If a password is requested during connection setup, automatic connection (automount) of the network drive is not possible. ■ Password: The user name and password are required for connecting to a Windows network. Enter your personal Windows password or a generally valid access to the network (max. 80 characters). ■ Password verification: Repeat the password you have entered.



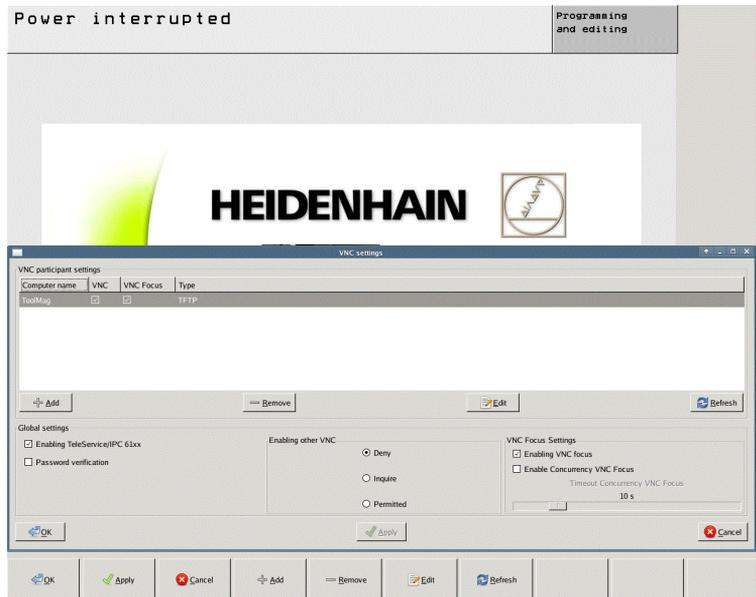
Screen button / soft key	Screen button / soft key	Meaning
	Automount	Select whether the drive is to be connected automatically during control start-up if a network connection exists.
	Expert Options	These settings are only intended for your network specialists.
	Inspection	Summary of the settings made. Please check the settings before saving them with the Apply button. You can later change the settings using the Edit button.
Remove	Remove the selected network drive from the list.	
Copying	Copy the data of the selected network drive and insert it in the list as a new entry.	
Editing	Edit the settings of the selected network drive (see Add).	
Status Log:		
	The Status Log window provides various information during the connection of a network drive. This information assists you in troubleshooting if a network drive cannot be connected, for example.	
The following screen buttons are available in the Status Log window:		
Delete	Delete the contents of the Status Log window.	
The following screen buttons are available in the Mount Setup window:		
OK	Accept the changes and exit the network settings.	
Apply	Load the changes.	
Cancel	Exits the network settings. The changes made are discarded.	

VNC settings

The behavior of the different VNC clients (e.g. IPC) can be configured using the VNC settings. This includes, for example, operation via soft keys, mouse and the ASCII keyboard.

Open the VNC settings as follows:

- ▶ Make the taskbar at the bottom of the screen visible by moving the mouse pointer over it.
- ▶ Press the green HEIDENHAIN button to open the JH menu.
- ▶ Select the **Settings** menu item
- ▶ Select the **VNC settings** menu item



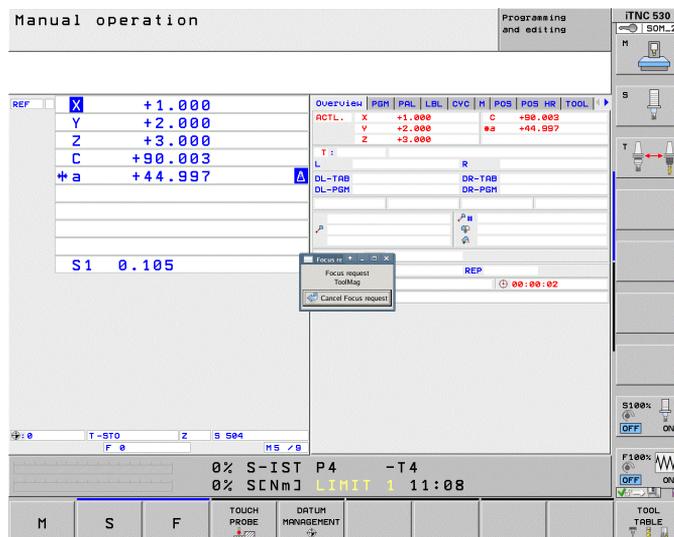
Dialog	Option	Meaning
VNC settings		
Overview of the VNC settings		
	Computer name:	IP address or computer name
	VNC:	Connection of the client to the VNC viewer
	VNC Focus	The VNC client participates in the focus assignment

Dialog	Option	Meaning
	Type	<ul style="list-style-type: none"> ■ Manual Manually entered VNC client ■ Denied This VNC client is not permitted to connect ■ TeleService/IPC 61xx VNC client via TeleService connection ■ TFTP Currently not yet supported. Preparation for future expansions. ■ DHCP Other computer that retrieves an IP address from this computer ■ Add Add new VNC viewer/client ■ Remove Deletes the selected client. Possible only for clients that were entered manually. ■ Edit Edit the configuration of the selected client ■ Refresh Updates the display. Necessary during connection attempts while the dialog is open.
Global settings		
	Enabling TeleService/IPC 61xx	Connection via TeleService/IPC 61xx is always permitted
	Password verification	The client must enter a password for verification. If this option is active, the password must be entered when the connection is established.
Enabling other VNC		
	Deny	Access generally denied to all other VNC clients.
	Inquire	During connection attempts a corresponding dialog is opened.
	Permitted	Access is generally granted to all other VNC clients.
VNC Focus Settings		
	Enabling VNC focus	Enable focus assignment for this system. Otherwise there is no central focus assignment. In the default setting, the focus is actively reassigned by the owner of the focus by clicking the focus symbol. This means that the owner of the focus must first release the focus by clicking the focus symbol before any other client can retrieve the focus.
	Enable Concurrency VNC Focus	In the default setting, the focus is actively reassigned by the owner of the focus by clicking the focus symbol. This means that the owner of the focus must first release the focus by clicking the focus symbol before any other client can retrieve the focus. If concurrency focus is selected, any client can retrieve the focus at any time without having to wait for the current owner of the focus to release it.



Dialog	Option	Meaning
	Timeout Concurrency VNC Focus	Time period within which the current owner of the focus can object to the focus being withdrawn or can prevent the reassignment of the focus. If a client requests the focus, a dialog in which the reassignment of focus can be refused appears on all clients' screens. If a VNC client requests this focus, a corresponding dialog window appears on all VNC clients' screens. The current owner of the focus can refuse the reassignment of focus in this dialog.
Focus icon		
		Current status of VNC focus on the respective client: Focus is owned by other client. Mouse and keyboard are disabled. Focus assignment is controlled either by clicking the focus icon with the mouse or by using the Connect key.
		Current status of VNC focus on respective client: Focus is owned by current client. Entries can be made.
		Current status of VNC focus on respective client: Owner of focus is requested to give the focus to another client. Mouse and keyboard are disabled until the focus is assigned unambiguously.

If **Enable Concurrency VNC Focus** is selected, the dialog window shown below appears. This dialog makes it possible to refuse that the focus be given to the requesting client. If this is not done, the focus will be given to the requesting client after the specified time period has elapsed.



The PLC Modules 9286, 9287 and 9288 enable you to use the PLC program to control the focus assignment.

For multiple clients or operating units, the focus assignment procedure depends on the design and the operating situation of the machine. The task of realization of and the responsibility for the focus assignment therefore lies with the machine tool builder.

All external VNC clients (non-HEIDENHAIN devices) have read-only rights to display the screen contents, and not write rights.

Module 9286 Select the VNC client and control of the focus assignment

PLC Module 9286 is used to program the following status requests regarding the control and assignment of multiple VNC clients:

- Select the active VNC client and simultaneously the assignment of the focus
- Enable or disable focus assignment (mouse operation, touchpad) for the user

Constraints:

- Call only from a submit/spawn job
- The names of the VNC clients (host name or MAC address) can be ascertained with Module 9287

Call:

PS	B/W/D/K	<>Mode>	0: Define the active VNC client and simultaneously assign the focus 1: Enable or disable focus assignment
PS	B/W/D/K	<>Name>	Mode 0: PLC string with host name or IP address Mode 1: - - -
PS	B/W/D/K	<>Timeout [ms]>	Module rounds up to the next 100 ms limit 0: Default timeout 3000 ms
PS	B/W/D/K	<>Additional parameter>	Mode 0: 0: Enable focus assignment 1: Disable focus assignment 2: Focus assignment unchanged Mode 1: 0: Enable focus assignment 1: Disable focus assignment
CM	9286		
PL	B/W/D	<>Error code>	0: No error 1: PLC error, error code in PLC error word (see below) >= 100: Operating-system error



Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Faulty execution of module
W1022	1	Mode parameter is invalid
	2	Timeout parameter or additional parameter parameter is invalid
	3	PLC string address parameter invalid
	20	Call was not in a submit/spawn job
	45	PLC module cancelled (operating-system error)
	51	NC software does not support module
	61	Function could not be executed
	100	Invalid parameter
	110	Focus assignment unclear, currently no master
	111	Client could not be set as master
	120, 121, 122	Disabling of focus assignment not supported
	130	Unknown status of focus assignment
	131, 132, 133, 134	Interference in communication for focus assignment
200, 201	Internal error	

Module 9287 General status requests regarding VNC clients

PLC Module 9287 is used to program the following status requests regarding control and assignment of multiple operator terminals (IPC, thin client):

- Number of VNC clients
- Status of focus assignment
- Name of the active VNC client (master)
- List of VNC clients
- Individual name of a VNC client from the client list

Constraints:

- Call only from a submit/spawn job
- If no name is configured for a client, its MAC address is returned in the PLC strings.
- Mode 3/4: The sequence and numbering of the client list can vary from one boot instance to the next, since this information depends on the point in time at which the clients register themselves in the system.

Call:

PS B/W/D/K <>Mode>
 0: Number of active clients suitable for focus assignment
 1: Status of focus assignment
 2: Name of the active VNC client
 3: List of all clients
 4: Name of client n from the client list

PS B/W/D/K <>Timeout [ms]>
 Module rounds up to the next 100 ms limit
 0: Default timeout 3000 ms

PS B/W/D/K <>Additional parameter 1>
 Mode 0: - - -
 Mode 1: - - -
 Mode 2: Address of PLC string
 Mode 3: Address of PLC string
 Mode 4: Address of PLC string

PS B/W/D/K <>Additional parameter 2>
 Mode 0: - - -
 Mode 1: - - -
 Mode 2: - - -
 Mode 3: - - -
 Mode 4: Number of clients n

CM 9287

PL B/W/D <>Error code>
 0: No error
 1: PLC error, error code in PLC error word (see below)
 >= 100: Operating-system error

PL B/W/D <>Status>
 Mode 0: Number of clients
 Mode 1: Focus assignment 0/1: Enabled/disabled
 Mode 2: - - -
 Mode 3: - - -
 Mode 4: - - -



Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Faulty execution of module
W1022	1	Mode parameter is invalid
	2	Timeout parameter or additional parameter parameter is invalid
	3	PLC string address parameter invalid
	11	List of clients is empty
	12	Error while evaluating the client list
	20	Call was not in a submit/spawn job
	45	PLC module cancelled (operating-system error)
	51	NC software does not support module
	61	Function could not be executed
	100	Invalid parameter
	110	Focus assignment unclear, currently no master
	111	Client could not be set as master
	120, 121, 122	Disabling of focus assignment not supported
	130	Unknown status of focus assignment
131, 132, 133, 134	Interference in communication for focus assignment	
200, 201	Internal error	

Module 9288 Status requests regarding a VNC client

PLC Module 9288 (call only from within a submit/spawn job) is used to program the following status requests regarding control and assignment of multiple operator terminals (IPC):

- Properties of a client (master / can be master / can be displayed)

Constraints:

- Call only from a submit/spawn job.
- The names of the clients (host name or MAC address) can be ascertained with Module 9287.

Call:
 PS B/W/D/K <>Mode>
 0: Properties of the thin client
 PS B/W/D/K <>Name>
 PLC string with name of the client
 PS B/W/D/K <>Timeout [ms]>
 Module rounds up to the next 100 ms limit
 0: Default timeout 3000 ms
 PS B/W/D/K <>Additional parameter 1>
 Mode 0: - - -
 CM 9288
 PL B/W/D <>Error code>
 0: No error
 1: PLC error, error code in PLC error word (see below)
 >= 100: Operating-system error
 PL B/W/D <>Status>
 Mode 0:
 Bit 0: 1 = Client is master
 Bit 1: 1 = Client can be master
 Bit 2: 1 = Client can be displayed

Error code:

Marker	Value	Meaning
M4203	0	No error
	1	Faulty execution of module
W1022	1	Mode parameter is invalid
	2	Timeout parameter or additional parameter parameter is invalid
	3	Parameter name is invalid
	20	Call was not in a submit/spawn job
	45	PLC module cancelled (operating-system error)
	51	NC software does not support module
	61	Function could not be executed
	100	Invalid parameter
	110	Focus assignment unclear, currently no master
	111	Client could not be set as master
	120, 121, 122	Disabling of focus assignment not supported
	130	Unknown status of focus assignment
	131, 132, 133, 134	Interference in communication for focus assignment
200, 201	Internal error	



DHCP server

As of software version 60642x-03, a DHCP/DNS server can be configured and started. This server can be used to supply the devices that are installed in the machine network with dynamic IP addresses and to provide a name resolution ("DNS service").

The server cannot be started on the primary network interface of the MC. For this function to be available, the MC must have at least two network interfaces so that the server can be activated.

The server provides the following services for this interface:

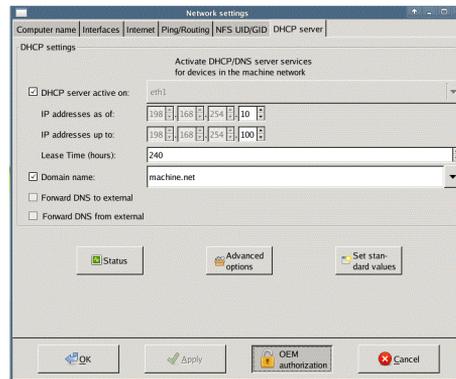
- DHCP service (dynamic IP address assignment)
- DNS service (name resolution)
- NTP service (time synchronization)

Configuring the DHCP/DNC server

■ DHCP/DNS server configuration

Open the network settings of your control. The configuration dialog is protected from unauthorized changes by the PLCPASSWORD.

- ▶ Select the **DHCP** tab
- ▶ Press the **OEM authorization** button or soft key
- ▶ Enter the defined PLCPASSWORD or the code number 807667



Tab	Option	Meaning	
DHCP	<p>Configuration of the DHCP server</p> <p>DHCP server active on:</p>	<p>DHCP/DNS server:</p> <p>Activation of the DHCP server and configuration of the Ethernet interface. Also activates other services, such as DNS or NTP. The option appears dimmed if the MC has only one Ethernet interface.</p> <ul style="list-style-type: none"> ■ Selection of the interface on which the DHCP server is to start. The primary interface (eth0) cannot be selected here. The primary interface can be selected on the first tab Computer name. If the MC has two interfaces, the non-primary interface appears dimmed here. The interface used here must have a static IP address and network mask. This is necessary because the address range of the dynamic IP addresses of the DHCP server is derived from this IP address and network mask. ■ IP addresses as of: Defines which IP addresses are output by the DHCP server. The values that appear dimmed are transferred from the static IP address of this network interface of the MC. The number of segments that appear dimmed in the IP-range setting is defined in the network mask of the static IP address of the network interface (Interfaces tab -> Configure). The start (lowest IP address) of the dynamically used IP address range is defined here. ■ IP addresses up to: Definition of the highest IP address of the dynamically used IP address range. The IP addresses are not retrieved in consecutive order. ■ Lease Time (seconds): Lease Time is the time during which an IP address is reserved for a client. If the client logs on again within this time, he will definitely be assigned the same dynamic address again. During operation, a client automatically extends the validity of the assigned dynamic IP address. This means that if a client is switched off longer than the Lease Time, it can happen that the dynamic IP address is used for another client. 	
		<p>Domain name:</p>	<p>Here you can enter the name of the internal network. The name is required, for example, to distinguish devices in the internal network from devices with the same name in an external network, or if you want to access the internal network from devices in an external network. The domain name can be chosen freely or you can select one of the three suggestions (machine.net, tnc.net or <hostname>-net.com) using the arrow.</p>
		<p>Forward DNS to external:</p>	<p>This option can only be selected if the IP-Forwarding option has also been selected on the Interfaces tab. If this option is activated, the name resolution for the devices in the internal network can also be used by devices in the external network. The option is required, for example, if you want to reach a device in the internal network with dynamic IP address from a PC in the external network by using the host name of the device.</p>

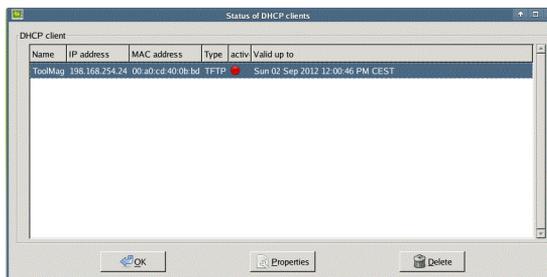


Tab	Option	Meaning
	Forward DNS from external:	This option can only be selected if the IP-Forwarding option has been selected on the Interfaces tab. If this option is set, DNS requests from devices in the internal network that cannot be answered by the MC's DNS server are forwarded to the name servers of the external network. In the configuration of the primary interface, you can define the name servers to be dynamic (via DHCP) or static.
	Status	Using the Status button or soft key, you can call an overview of the internal devices with dynamic IP address and define settings for these devices, see page 1890.
	Advanced options	Here, experts can enter additional parameters for the DNS/DHCP server, see page 1892.
	Set standard values	Using the Set standard values button or soft key, you activate the DHCP/DNS server (if there are several network interfaces and a static IP address on the non-primary network interface). The factory default settings are set.



Status of DHCP clients

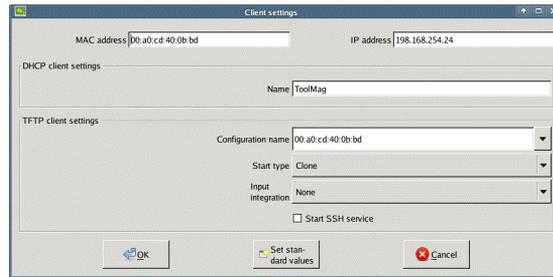
- Requesting the status
All valid IP addresses output by the DHCP server are displayed in the **Status of DHCP clients** window. Valid means that the **Lease Time** of the IP addresses has not expired yet. If you select a line in the list, you can define settings for the device using the **Properties** button, see page 1891. The IP address and all settings for the device can be deleted by pressing the **Delete** button. The displayed list is updated approx. every 5 seconds.



Column	Meaning
Name	Host name of the device. If the device logs on to the DHCP server using a host name, this name will be used. If the device does not report a name, the standard name will be generated from the MAC address (all colons will be replaced by hyphens). For DHCP devices, the device name can be overwritten using the Properties button.
IP address	The dynamically assigned IP address of the device in the internal network
MAC address	The MAC address of the device in the internal network
Type	For external devices, the type DHCP must be entered if they request a dynamic IP address. Devices with static IP in the internal network are not shown because these devices do not log on to the DHCP server.
Active	If the device replies to an ICMP echo request (ping), a green dot is entered here. If the device is not available or does not react, a red dot is entered.
Valid up to	Time at which the validity of the IP address will expire without being renewed. If you subtract the Lease Time from this time you can ascertain at what time the device last booted or extended the validity of the IP address.

DHCP client settings

- Client settings
In this window you can enter the names of all devices (overwrite).

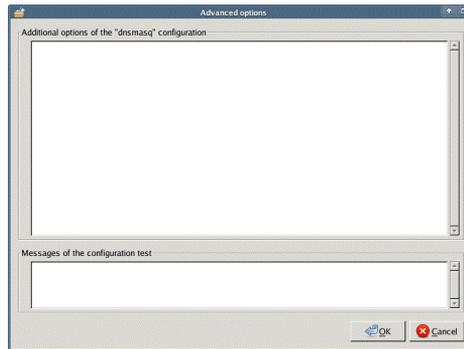


Dialog	Option	Meaning
Client settings	MAC address:	MAC address of the client
	IP address:	IP address of the client
	DHCP client settings	<ul style="list-style-type: none"> Name: Here you can overwrite the host name of the device. The DHCP server tries to set the host name of this device as indicated here. It depends on the device, however, whether it will really change its name as specified by the DHCP server. On most devices the name must be specified in the device itself, or at least the configuration in the device must be modified such that the device retrieves its host name from the DHCP. Regardless of whether the device itself changes its name as specified, this IP address name is entered in the name server on the MC, and therefore this name must be used to access the device in the network.
	TFTP client settings	Currently not yet supported. Preparation for future expansions.
	Start SSH service	Currently not yet supported. Preparation for future expansions.

Advanced options for DHCP

■ Advanced options

The advanced options are intended for experts who want to exploit the full range of possibilities offered by the "dnsmasq" program that provides the DHCP and DNS services.



Column	Meaning
Additional options	Here you can enter the additional options.
Messages	If you confirm the window with OK , the options entered are checked. If an error is found, the error is described here and the window will not be closed. If no error is found, the window will be closed. The window can always be closed with Cancel . All changes are rejected in this case.

10.4 Protection Against Data Tampering

Due to the possibility of networking the iTNC 530 and accessing it remotely, protection from data tampering became necessary.

The following protection mechanisms are integrated:

- General disabling of data access by soft key
- Restricting access to the PLC partition
- Restricting access to parts of the TNC partition
This function was added for the end user. Please see the notes in the User's Manual.

General disabling of data access

Soft key	Meaning
	This soft key disables access to the control using the LSV2 protocol, via both the serial and the Ethernet interfaces. It can be displayed in any operating mode with the MOD key.

The soft key is normally not shown. To display the soft key:

- ▶ Enter the keyword **REMOTE.LOCKSOFTKEYVISIBLE = YES** in OEM.SYS.

Restricting access to the PLC partition

In the standard setting, the PLC partition can be accessed via the LSV2 protocol using the code number 807667. To permit this access only with the code number defined in OEM.SYS under **PLCPASSWORD** = (no longer with 807667):

- ▶ Enter the keyword **REMOTE.PLCPASSWORDNEEDED = YES** in OEM.SYS.

Code number 807667 is used during machine backup and full backup with the LSV2 protocol to access the PLC partition. To permit this access only with the code number defined in OEM.SYS under **PLCPASSWORD** = (no longer with 807667):

- ▶ Enter the keyword **REMOTE.PLCPASSWORDFORCED = YES** in OEM.SYS.

10.5 The USB Interface of the iTNC

The USB interface is a standard serial interface.
(USB = Universal Serial Bus)

Various USB devices, such as mouse, external hard disks, and USB memory sticks, can be connected to the iTNC via the USB interface of the MC 6xxx. The USB interfaces of the MC 6xxx support USB 2.0.



Note

If USB components require more than 0.5 A, a separate power supply becomes necessary for these components. One possibility is the USB hub from HEIDENHAIN (368 735-01).

The USB interface features "hot-plug capability." This means that you can connect USB devices to the USB interface and remove them, without having to shut down and then restart the control.

Transmission distance without hub: Up to 6 meters



Note

For greater transmission distances, you must use a USB hub after every six meters in order to amplify the signal. You can use more than one hub for one transmission distance. USB cables with a length of up to 36 meters (with 6 integrated USB hubs) are available from HEIDENHAIN:

- USB cable, ID 354 770-xx
These USB cables support USB 1.1 and USB 2.0
- USB cable, ID 624 775-xx
The hubs integrated in the USB cable only support USB 1.1

X141, X142, X143, X144

Pin layout for USB connection (Type A):

USB connection (female) 4-pin	Pin layout
1	+5 V
2	USBP-
3	USBP+
4	GND



Note

If USB components that are connected to one of the USB ports require more than 0.5 A, a separate power supply becomes necessary for these components. One possibility is the USB hub (USB 2.0) from HEIDENHAIN (582 884-02).

If a USB hub is connected to one of the USB ports, the maximum permissible length of the USB cable (ID 624 775-xx) is reduced to 20 m.

USB hub

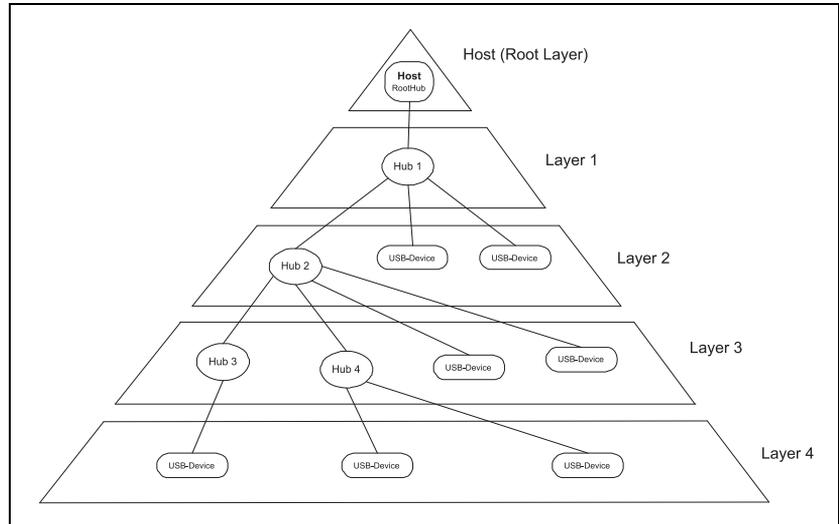
The power supply for the USB hub must comply with EN 50 178, 5.88 requirements for "low voltage electrical separation."

For more information on the USB interface, see the iTNC 530 Technical Manual.

Setup

The USB interface connects the USB peripheral devices with the USB host. The topology of a USB connection may consist of several levels arranged in a star configuration. Every level consists of a USB hub to which other USB devices or hubs are connected in a star configuration. A maximum of 127 USB devices can be connected to a USB host in this way.

Topology of the USB interface:



Functionality and signal designations

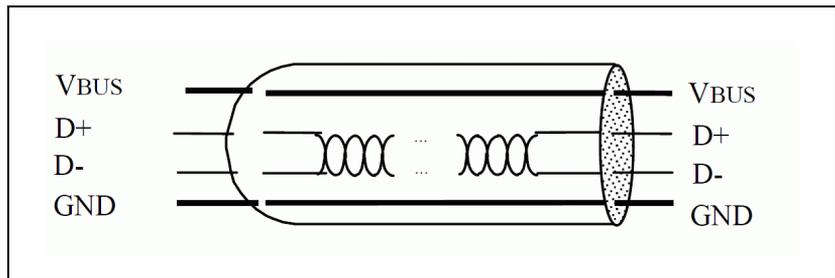
USB uses packet-based communication over two differential data lines. This reduces radiation and increases transmission reliability. USB provides significantly higher data transfer rates than the parallel / Centronics, and serial / RS-232, RS-422 external interfaces:

- USB full speed of up to 12 Mbps
- USB low speed of up to 1.5 Mbps

Conventional interfaces, such as the RS-232, are more suitable for time-critical applications because they are not based on packets which reduce the transfer rate (in case of packets with only a few bytes) or delay transmission (when collecting bytes for filling a packet).

Only four wires are required in a USB cable. Two for a power supply of 5 V (with max. 500 mA / 2.5 W) and two for data transmission.

Structure of the USB cable:



USB data carriers on the control (Upgrade function – FCL)

The USB interface of the control allows for convenient and fast exchange of data. You can connect USB block devices, such as memory sticks, hard disks, CD-ROM drives, to your control via the USB interface without having to reboot the system. The data media can be accessed immediately after connection.

The control supports the following USB block devices:

- Floppy disk drives with FAT/VFAT file system or ISO9660
- Memory sticks with FAT/VFAT file system or ISO9660
- Hard disks with FAT/VFAT file system or ISO9660
- CD-ROM drives with FAT/VFAT file system or ISO9660

The control does not support USB devices with other file systems (e.g. NTFS). If you try to connect such devices, the control will issue an error message.



Note

- It should basically be possible to connect all USB block devices with the above-mentioned file systems to the control. If you nevertheless encounter problems, please contact HEIDENHAIN.
- Please note that the "Automatic Update" function on the control with USB data carriers is independent of the Feature Content Level.

Connecting and disconnecting USB data carriers (upgrade function)

There are two soft keys in the iTNC's file manager for connecting and removing USB data carriers (hard disks, memory sticks, etc.):

Soft key	Meaning
	Connect a USB data carrier to the control
	Disconnect a USB data carrier from the control

These appear after the MORE FUNCTIONS soft key is pressed, if the USB device was selected in the file manager. If the USB device is not selected, then the iTNC only shows the soft key for connection.



Attention

In order to remove a USB data carrier, you must always press the MORE FUNCTIONS and soft keys.



Data can be lost if the USB data carrier is removed from the control before the appropriate soft key is pressed.

USB data carriers tested by HEIDENHAIN (upgrade function)

A variety of USB storage media from different manufacturers is available on the market. Numerous USB devices are supported by the control, but you should test them for proper functioning on the control before using them. It may happen in isolated cases that a USB device is not supported by the control.



Drive name and user group for USB device

You can define the drive name and the authorized user group for USB devices. This requires that **USBMOUNTS = <path\file>** is entered in OEM.SYS. This entry refers to a file containing one or more lines with the following syntax:
<Volume>,<Mountpoint:>,<User>

Example of entry in the file that is referred to, e.g. Stickconfig.a:

KINGSTON,PLCSTICK:,OEM

If the control detects a USB device during startup (only if it was switched on with the main switch), which has a VFAT partition with one of the specified volume names, the device is mounted and displayed with the name specified in <Mountpoint:>, and not with the name **USBn:**.

If the device also has other partitions, the other partitions are ignored except if they are also mounted with a volume name.

If a USB device with one of the indicated volume names is connected after the control has started up, the device is mounted with the name **USBn:** as usual. The new functionality does not become effective until the control is switched off and then back on again.

The following settings can be defined in any ASCII file (e.g. Stickconfig.a):

- <Volume>
Volume name of a USB device formatted with VFAT. If the name contains blank spaces, the complete name must be enclosed between quotation marks (" "). Also, the name must always be given in capital letters.
- <Mountpoint>
Drive name under which the USB device is to be mounted (e.g. PLCSTICK:). The last character of the name must be a colon. The name can have up to 8 characters (without colon).
- <User>
User group for which the USB device is mounted and displayed. The following user groups are possible:
 - TNC:
The USB device can be used by the machine operator.
 - OEM:
The USB device can only be used by the OEM (code number 807667, PLC program, NC cycles).
 - SYS:
The USB device can only be used by the HeROS system.

Connecting more than one USB device with the same volume name may cause problems. When the control is started up, the first detected USB device with the new drive name is mounted. As a result, it may occur that each time the control is started up, another device with the same volume name is mounted.

HEIDENHAIN therefore recommends using unique volume names that cannot be confused (e.g. "JHK247G4"). On a PC with Windows operating system, the volume name can be defined in the properties of the USB device.



10.6 iTNC Serial Data Interfaces

10.6.1 General information

The iTNC features an interface of the following type:

- RS-232-C/V.24

Three transmission protocols are available:

- Standard communications protocol
- Transmission protocol with Block Check Character (BCC)
- LSV2 transmission protocol



Note

No connection to the serial interface of the iTNC can be established if the file manager on the iTNC is open.

10.6.2 RS-232-C/V.24 interface

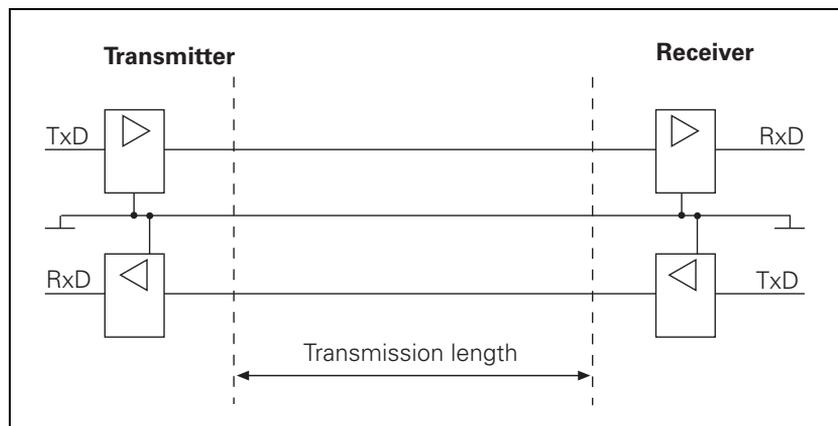
RS-232-C/V.24 is the designation for a serial interface. Data transfer is executed asynchronously, with a start bit before each character and one or two stop bits after each character.

Transmission distance: up to 20 m

Hardware

The physical connection between two RS-232-C/V.24 interfaces is an asymmetrical line, i.e. the common ground connection between transmitter and receiver is used as a return wire.

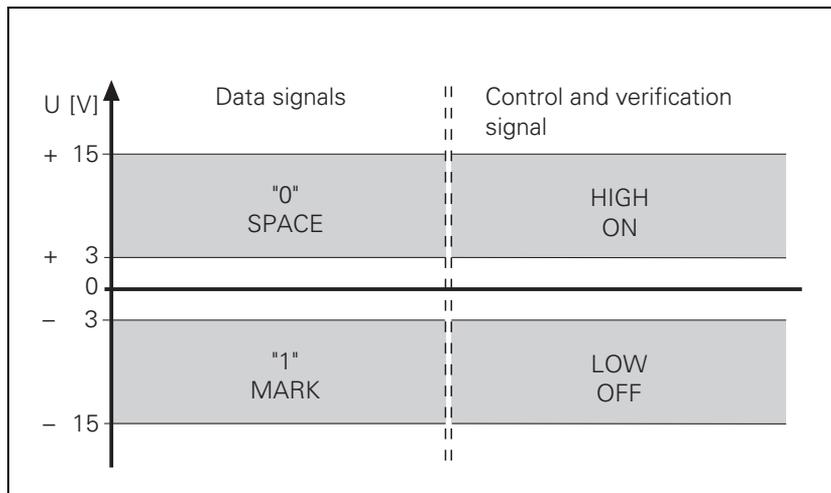
Physical connections:



Signal levels

The levels of the individual signal lines differ:

- Data lines: The data signals are defined as being logical zero (SPACE) over the range +3 V to +15 V and logical one (MARK) over the range -3 V to -15 V.
- Control and verification lines: These signals are defined as being ON (High) over the range +3 V to +15 V and as OFF (Low) over the range -3 V to -15 V.



Note

For all signals: The voltage range from -3 V to +3 V cannot be evaluated.

Signal designation

One must differentiate between the following types of lines and their signals:

■ Data lines:

- TxD Transmitted data
- RxD Received data

■ Control and signal lines:

- DCD (Data Carrier Detect):
Received signal level. The receiver signals that the information it has received lies within the defined level. The DCD signal is not used by the iTNC. The iTNC delivers no signal from this pin.
- DTR (Data Terminal Ready):
iTNC ready / not ready for service (e.g. the receiving buffer is full, the signal DTR indicates "LOW").
- DSR (Data Set Ready):
Peripheral device ready / not ready for service.
- RTS (Request to Send):
Switch transmission unit on. iTNC wishes to transmit data.
- CTS (Clear to Send):
Readiness for transmission. The peripheral wishes to transmit data.

■ Ground conductors (lines for power supply):

- Chassis GND:
Housing connection
- Signal GND:
0 V lines for all signals

Pin layout

Keep in mind that there might be a difference between the pin layout of the MC 42x(B) and the adapter block.

10.7 Configuration of Interfaces

10.7.1 Control characters

Overview of control characters specific to HEIDENHAIN

Character	Description	Description
SOH	Start of Header	Identifies the beginning of the data transfer header. The character string contains the program number and information about the type of program and the transfer mode.
STX	Start of Text	Identifies the beginning of a program block.
ETB	End of Text Block	Terminates a data transfer block. The character that follows (BCC) is used for data checking.
DC1	XON	Starts the transfer of data.
DC3	XOFF	Stops the transfer of data.
ETX	End of Text	Transmitted at the end of a program.
EOT	End of Transmission	Terminates the data transfer and establishes the idle state. This character is transmitted by the iTNC at the end of a program input and to the external device in the event of an error.
ACK	Acknowledgment	Transmitted by the receiver when a data block has been transferred without error.
NAK	Negative Acknowledgment	Transmitted by the receiver when a data block has been transferred with an error. The transmitter must re-transmit the data block.



10.7.2 Selection of interfaces and operating modes

To disable or enable the RS-232-C/V.24 interface:

- ▶ Select MP5000.

If the interface is enabled, you can select the following settings:

- ▶ Call the MOD functions.
- ▶ Choose the desired operating mode from the table.

For the following external devices	Choose the operating mode
PC with HEIDENHAIN software TNCremo	LSV2
Non-HEIDENHAIN devices such as printer, punch or PC with other data transfer software	EXT1 and EXT2

MP5000 Disable data interfaces

Input: 0: No interface disabled
 1: RS-232-C/V.24 interface disabled
 2: Reserved
 3: Reserved

Communication between iTNCs

- ▶ Set both iTNCs to LSV2 protocol. The control from which you start the data transmission is the master.

The PLC provides you with access to the data interfaces (EXT3).

10.7.3 Configuration of interfaces

To configure data format and the type of handshake in the EXT1/EXT2/EXT3/EXT4 operating modes (EXT3/EXT4 only for the PLC):

- ▶ Select machine parameter MP5020.x.

Data bits

With **bit 0** you determine whether transmission is to be with seven or eight data bits. Transmission with seven bits is normally used; for printer interfacing eight bits are required.

Block Check Character (BCC)

With **bit 1** you can ensure that the BCC is not interpreted as a control character.

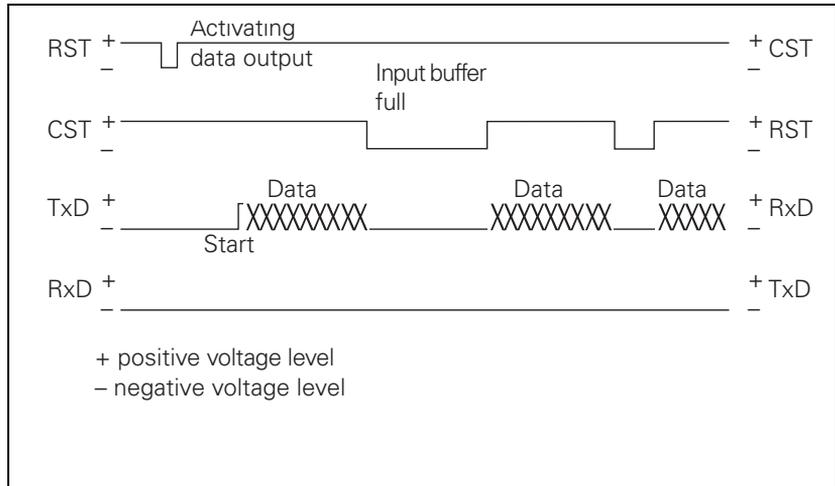
On the iTNC, numbers less than \$20 are defined as control characters. If calculation of the BCC produces a number less than \$20, then a blank space is sent in addition immediately before <ETB>. The BCC will consequently always be greater than \$20 and cannot therefore be interpreted as a control character.

Hardware handshaking

Bit 2 can be set to determine whether the iTNC stops transfer from an external device by sending an RTS signal.

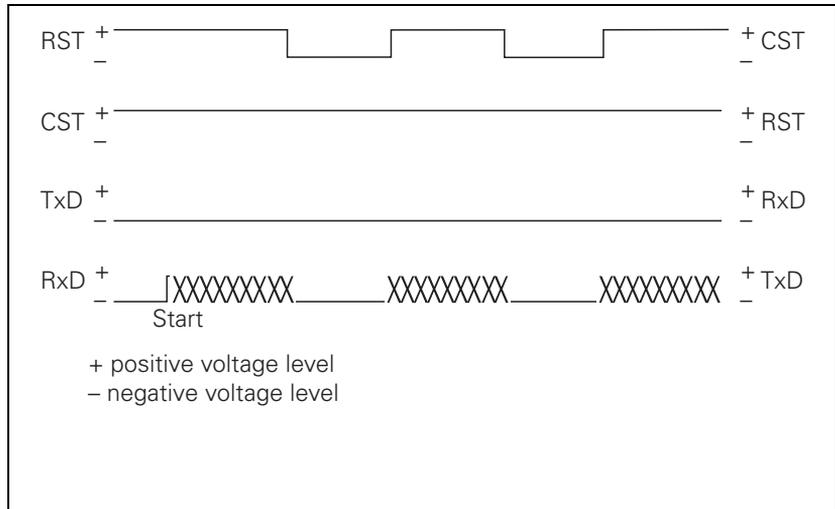
■ Data output from iTNC to EXT

When the receiving buffer is full, the external device resets the RTS signal. The iTNC detects that the peripheral unit receiving buffer is full at its CTS input:



■ Data input from EXT to iTNC

When the receiving buffer is full, the iTNC removes the RTS signal. This is detected by the peripheral device at its CTS input:



The DTR and DSR signals indicate the operational status of the iTNC and peripheral device:

- DTR: Interrogated by peripheral; it is logical one if iTNC is ready for service.
- DSR: Interrogated by iTNC.
 - HIGH level means: external data input/output ready.
 - LOW level means: external data input/output not ready.

Software handshaking

With **bit 3** you determine whether the iTNC stops transfer from an external device with control character <DC3>. Transfer is then resumed with character <DC1>. (XON/XOFF method)

If transfer is stopped with the control character <DC3>, up to three more characters can be stored; any further incoming characters are lost. Software handshaking is normally recommended when interfaces are connected to an external device.



Note

The iTNC reacts both to hardware and software handshakes, regardless of the setting in MP5020.x.

If no transmission stop is set in MP5020.x, the iTNC stops the peripheral unit with the software handshake.

If a transmission stop by RTS and by DC3 is active simultaneously, the iTNC stops transfer with the hardware handshake.

Character parity

Bits 4 and **5** determine the type of parity check.

Stop bits

Bits 6 and **7** determine the number of stop bits sent at the end of a character.

Testing the connection

If PLC Modules 910x transmit characters via the RS 232-/RS422 interface, the control no longer checks if a connection is available if **bit 10** is set to 1 in MP5020.x.

Bit 8 and **bit 9** in MP5020.x are reserved and currently have no function.

MP5020

Configuration of the data interface

Format:

%xxxxxxx

Input:

Bit 0: 0 = 7 data bits, 1 = 8 data bits

Bit 1: 0 = any BCC, 1 = BCC not control character

Bit 2: 0 = transmission stop by RTS not active, 1 = active

Bit 3: 0 = transmission stop by DC3 not active, 1 = active

Bit 4: 0 = character parity even, 1 = odd

Bit 5: 0 = character parity not desired, 1 = desired

Bit 6 = 0, Bit 7 = 0: Length of the stop: 1.5 bits

Bit 6 = 1, Bit 7 = 0: Length of the stop: 2 bits

Bit 6 = 0, Bit 7 = 1: Length of the stop: 1 bit

Bit 6 = 1, Bit 7 = 1: Length of the stop: 1 bit

Bit 8 = reserved

Bit 9 = reserved

Bit 10: 0 = check for missing connection active

Bit 10: 1 = check for missing connection not active

MP5020.0 Operating mode EXT1

MP5020.1 Operating mode EXT2

MP5020.2 Operating mode EXT3 (PLC)

MP5020.3 Operating mode EXT4 (PLC)

Communication protocol

The communication protocol for operating modes EXT1/EXT2/EXT3/EXT4 is defined with MP5030:

MP5030	Communications protocol
Input:	0 = standard data transfer protocol 1 = blockwise transfer 2 = without protocol (only for MP5030.2)
MP5030.0	Operating mode EXT1
MP5030.1	Operating mode EXT2
MP5030.2	Operating mode EXT3 (PLC)
MP5030.3	Operating mode EXT4 (PLC)

Example

The printer NEC P7 PLUS is to be configured with operating mode EXT1.

The parameters listed are preset by the printer (see the operating manual of the printer concerned):

- Serial interface
- 8 data bits
- Even character parity
- XON/XOFF protocol (software handshake)
- 9600 bps

The following settings are made at the iTNC:

- ▶ Select MP5000 = 0.
- ▶ Select MP5020.0 = %10101001.
- ▶ Select MP5030.0 = 0.

Machine parameters	Effect
MP5000 = 0	No interface inhibited
MP5020.0 = %10101001	Bit 0: 8 data bits Bit 1: any BCC character Bit 2: transmission stop by RTS not active Bit 3: transmission stop by DC3 active Bit 4: character parity even Bit 5: character parity desired Bit 6/7: 1 stop bit
MP5030.0 = 0	Standard data transfer

- ▶ Call the MOD functions.
- ▶ Choose the EXT1 operating mode.
- ▶ Set the baud rate for EXT1 to 9600 bps.



10.8 Data Transmission Protocols

10.8.1 Selection of transmission protocols

The operating modes are assigned the following transmission protocols:

Operating modes	Communications protocol
FE1	Select a protocol with BCC and with fixed control characters 1 start bit, 7 data bits, 1 stop bit
EXT1, EXT2, EXT3, EXT4	Select data format and transmission protocol using machine parameters
LSV2	Start this protocol from a PC or from the iTNC. The protocol runs in the background of the iTNC.

The following applies to all data transmission protocols except LSV2:

If an incoming file is already stored in the iTNC, the TNC will ask you whether you really wish to overwrite this file:

- ▶ Press a soft key to continue the transmission.

If you attempt to overwrite a write-protected file, the iTNC displays the **Protected file!** error message:

- ▶ Press the MORE FUNCTIONS soft key and then the UNPROTECT soft key to cancel write-protection and continue transmission.

If a file has been read out and the data transfer menu has been terminated with the END key, the iTNC outputs the characters <EXT> and <EOT>.

If a transmission is terminated with the END key, the error message **Program incomplete** is issued.

10.8.2 Standard communications protocol

General information

To set the standard data communication protocol in the operating modes EXT1/EXT2/EXT3/EXT4:

► Select MP5030.x = 0.

When outputting a file, the <NUL> character is sent exactly 50 times at the start of file. When reading in, however, the control unit ignores this character, regardless of how often the peripheral sends the <NUL> character before the file.

The program blocks are not checked for correctness but are transmitted one after the other.

If you wish to signal an error to the iTNC in the standard data transmission protocol, you must send the following sequence of instructions:
<ESC><1><Error number>

If the receiver's data buffer is full, the transmission can be stopped and resumed in one of two ways:

- Software handshaking
 - Stop transfer by sending the character <DC3> (XOFF)
 - Continue by transmitting the character <DC1> (XON)
- Hardware handshaking
 - By suitable levels on the control and signal lines RTS and CTS of interfaces RS-232-C/V.24 or RS-422/V.11

Twelve characters before the receiving buffer is full, the iTNC transmits the character <DC3> to the transmitter in order to terminate transmission.

Example: Protocol for conversational NC program

<NUL><NUL><NUL><NUL> 50 times

0 BEGIN PGM 1 MM<CR><LF> 1st program block

1 TOOL DEF 1 L+0 R+3<CR><LF> 2nd program block

26 END PGM 1 MM <CR><LF> End of program

... ..

<ETX><EOT> Close the data transmission menu

Example of software handshake

iTNC to peripheral	Peripheral to iTNC
12 Z + 2 FMAX<CR><LF>	Receiving buffer full: <DC3> Receiving buffer ready again: <DC1>
13 Z -10 F100 M03<CR><LF>	...



Output selected file

The EXT1 operating mode is set with software handshake.

The iTNC outputs all of the program lines in order.

The peripheral device can:

- Stop transmission with <DC3>
- Resume transmission with <DC1>

iTNC to peripheral	Peripheral to iTNC
<NUL> <NUL> <NUL>... 1st line of file <CR> <LF>... 5th line of file <CR> <LF>	Transmission stop: <DC3> Resume transmission: <DC1>
6th line of file <CR> <LF>... Last line of file <CR> <LF>	...

Load selected file

The EXT1 operating mode is set with software handshake.

To transfer a file from a peripheral device:

- ▶ Enter the file name in the iTNC.

The iTNC can:

- Stop transmission with <DC3>
- Resume transmission with <DC1>

iTNC to peripheral	Peripheral to iTNC
100.H "START" <DC1>	<NUL><NUL> 1st line PGM100<CR><LF> ... Last line PGM 100 <CR><LF><ETX>
<EOT>	

If the file name in the first line and the name indicated in the iTNC are not identical, the iTNC reads each block in and searches for the correct file name. If the END PGM block has been read in, and the selected name is not known, the iTNC remains static without an error message:

- ▶ In this case, terminate transfer with the END key.

10.8.3 Communications protocol with block check character

This protocol is specific to HEIDENHAIN and operates with its own control characters and an additional data check feature when transmitting.

The protocol is set with the following operating modes:

- FE1 mode
- EXT1/EXT2/EXT3/EXT4 mode if MP5030.x = 1

The data transfer protocol is identical for all these modes except for the FE1 mode. In FE1 mode, a command sequence is output at the beginning to request the contents directory from the peripheral device. For the FE1 mode, bit 1 of MP5020.x must be set to 1.

Header

When a file is transferred, the first block—called the header—consists of the following characters:

<SOH><K><Name><M><ETB><BCC><DC1>

Character	Meaning
<SOH>	Identifies the beginning of the header
<K>	File code
<Name>	File name
<M>	Data transfer mode (E = input, A = output)
<ETB>	Identifies the end of the header
<BCC>	Block Check Character
<DC1>	XON

Block Check Character (BCC)

In addition to checking the parity of the individual characters, the parity of the complete transferred block is also checked. The BCC always rounds the individual bits of the transferred characters in a data transfer block to even parity.

Example of BCC generation:

In this example, program 15, which has been written in HEIDENHAIN plain-language text (H), is input through the data interface (E).

Character	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SOH	0	0	0	0	0	0	1
H	1	0	0	1	0	0	0
1	0	1	1	0	0	0	1
5	0	1	1	0	1	0	1
E	1	0	0	0	1	0	1
ETB	0	0	1	0	1	1	1
BCC	0	0	1	1	1	1	1

A parity bit is also generated for the BCC. With even parity, the parity bit in this example is assigned the value 1.



At the end of every block, the receiver checks whether it has been transferred correctly.

To do this, the receiver computes a BCC from the received block and compares it with the received BCC. If the received BCC and the computed BCC are identical, the receiver transmits the character <ACK> for positive acknowledgment. If the two BCCs are not identical, the data block was not transmitted correctly. The receiver transmits the character <NAK> for negative acknowledgment. The block must be re-transmitted. This process is repeated up to 15 times, then the error message **Transferred data incorrect E** is output. Transmission is aborted.

If the header is acknowledged with <ACK>, the first file block can be transmitted:

The beginning of a file block is identified by the control character <STX>. The remaining control characters in this block are identical with the control characters in the header. If this block is acknowledged by <ACK>, then the next program block is transmitted. If <NAK> is transmitted, the same block has to be re-transmitted, etc. Once the last program block has been acknowledged by <ACK>, the transmission is terminated by the characters <ETX> (end of text) and <EOT> (end of transmission).

Handshaking

The character <DC1> (XON) follows the BCC. This character is required by many devices to explicitly request the transmission once again from the transmitter.

The <DC1> character is not required for reading in a file in the BCC format.

The transmitter waits and only resumes data transmission when the receiver has transmitted a positive (ACK) or negative (NAK) acknowledgment to indicate that the receiving buffer is ready.

To disable transmission of the <DC1> character in the EXT1, EXT2, EXT3 and EXT4 modes:

- ▶ Set MP5020.x bit 3 = 0.

Example:

To read out a pallet file with the name PPP to a peripheral device (e.g. FE 401).

iTNC to peripheral	Peripheral to iTNC
<SOH><L>PPP<A><ETB>BCC	<ACK>
<STX> "1st line"<ETB>BCC	<ACK>
...	...
<STX> "10th line"<ETB>BCC	<NAK>
<STX> "10th line"<ETB>BCC	<ACK>
<STX> "11th line"<ETB>BCC	<ACK>
...	...
<STX> "last line"<ETB>BCC	<ACK>
<ETX><EOT>	

Report error to the iTNC

FE1 mode is set.

If an error occurs at a peripheral device, the following block must be sent to the iTNC:

<SOH><Error text><ETB>BCC

Peripheral to iTNC	iTNC to peripheral
<SOH> "Error"<ETB>BCC	<ACK><EOT>

The received error message is displayed on the iTNC. To continue

► Press the CE key.

Request external directory

FE1 mode is set.

This protocol is not available in the EXT mode. In FE1 mode the following Escape sequence is sent to request the external directory:

<DC3><ESC><DC1><0><SP><D><CR><LF>

The iTNC expects the following input to this request:

xxxxxx<Code letters><Sectors><Name><P¹⁾><CR><LF>

¹⁾ P = Protected (optional)

The first four lines, each ending in <CR><LF>, are ignored. In subsequent lines ending with <CR><LF>, the program name and, after any number of blank characters, the number of sectors are stored.

If the character combination <FREE:> is detected, only a number—the number of free sectors—will be read in.

The iTNC requests the complete directory. The directory is saved and the files of the selected type are displayed.

The peripheral device ends transmission with <EXT>. The iTNC sends an <EOT>.

Output selected file

iTNC to peripheral	Peripheral to iTNC
<SOH><K>Name<A><ETB>BCC	<SOH><K>Name<A><ETB>BCC
<STX> "1st line"<ETB>BCC	<ACK>
...	...
<STX> "last line"<ETB>BCC <DC1>	<ACK>
<ETX><EOT>	

Output marked files

Marked files are output in the same protocol as for outputting the selected files. After each file, the control characters <EXT><EOT> are sent to the peripheral device.



Load selected file

To read in a file from an external memory, the iTNC sends a header with the file name.

iTNC to peripheral	Peripheral to iTNC
<SOH><K>Name<E><ETB>BCC <DC1>	<ACK> <STX>"1st line"<ETB>BCC<DC1>
<ACK>	...
...	<STX>"last line"<ETB>BCC <DC1>
<ACK>	<ETX>
<EOT>	

10.8.4 LSV2 transmission protocol

The LSV2 protocol is a data transfer protocol for the two-way transfer of commands and data.

The data is transferred in blocks—so-called telegrams—into which the data is split up.

The following functions are possible:

- Data transfer
- File management, such as deleting, copying and renaming files
- Changing, creating and deleting paths
- Remote operation of the control functions. The control screen appears on the computer monitor. All functions can be executed from the computer.
- Real DNC operation. Starting and stopping the machine from the PC
- Diagnosis of iTNC error messages and keystrokes for service purposes. The last 1000 events are stored in the iTNC.
- An LSV2 command can be used to disable/re-enable write-access on tables exclusively for HEIDENHAIN-DNC.
- The selected NC programs can be stopped and unselected in the **Program Run Single Block** and **Full Sequence** operating modes.

HEIDENHAIN offers the respective software packages. Please contact HEIDENHAIN for further information.

Timeouts

You can define your own times for timeouts in the system file OEM.SYS:

- LSV2TIME0 = Timeout for receiving block STX to ETX (default 3 s)
- LSV2TIME1 = Timeout for acknowledging ENQ or check sum (default 3 s)
- LSV2TIME2 = Timeout when sending DLE 0, DLE 1 or NAK until a valid character is received (default 1 s)

Input range: 0.001 to 3.6 s



Note

If the code words are not defined or if the input range is exceeded, the default values are used.

10.9 Saving and Loading Files

The table lists all the files that can be saved to external memory devices and loaded from them.

File	File extension	Identifier
NC program in HEIDENHAIN plain-language	.H	H
NC program, DIN/ISO	.I	D
Tool table	.T	T
Pallet table	.P	L
Datum table	.D	N
Machine parameters	.MP	M
Compensation-value table	.COM	V
Compensation-value assignment	.CMA	S
PLC program	.PLC	P
Text file	.A	A
Pocket table	.TCH	R
Help files	.HLP	J
Point table	.PNT	U
PLC error table	.PET	F
System file	.SYS	O
Cutting-data table	.CDT	–
Freely definable tables	.TAB	–
Motor table (asynchronous motors)	.ASN	–
Motor table (synchronous motors)	.SN	–
Motor table (servo amplifiers)	.AMP	–
Error file	.JOU	–
OEM cycles	.CYC .DES .PIC .ELE	–
Oscilloscope recordings	.DTA	–
Network settings	.N00 .M00 .P00	–

To write to or read from machine parameter files, compensation tables or PLC files, you must enter the correct code numbers with the MOD function:

You can output the current values of Q parameters, PLC error messages and dialogs over the RS-422/V.11 and RS-232-C/V.24 interfaces. (NC program: **FN 15: PRINT**).

During transmission with a Block Check Character (BCC), each device outputs and receives the correct file code.

Example:

If a pallet table is stored, it is given the file extension *.LNC.

For data transmission with the **TNCremoNT** PC software from HEIDENHAIN the identifier has no significance. The files are saved on the PC with the same extension as on the iTNC.



Note

Files that have no code (-) can only be transmitted with the LSV2 protocol of TNCremoNT.



10.10 Data Transfer by PLC

10.10.1 Settings

PLC modules make it possible for the PLC to transfer data via the RS-232-C/V.24 data interface. These modules, for example, permit communication between two MC 6xxx at PLC level via the interface.

During data transfer, use of the interface is inhibited for the input/output program of the user interface.

- ▶ Select a standard operating mode, FE1 or
- ▶ Configure the data interface with MP5020.x to MP5040.x in EXT3/EXT4 mode.

MP5040 Data transfer rate in operating mode EXT3 or EXT4 (data transfer through PLC)

Input:	0: 110 bps
	1: 150 bps
	2: 300 bps
	3: 600 bps
	4: 1200 bps
	5: 2400 bps
	6: 4800 bps
	7: 9600 bps
	8: 19200 bps
	9: 38400 bps
	10: 57600 bps
	11: 115200 bps
MP5040.0	Operating mode EXT3 (PLC)
MP5040.1	Operating mode EXT4 (PLC)



10.10.2 PLC modules

With the following PLC modules you can control the data interfaces from the PLC:

- Modules 9100 and 9101: Assign/release the data interfaces
- Module 9102: Interrogate the status of the interface
- Modules 9103 and 9104: Transmit and receive a string from the string memory. The transmit and receive buffers for the PLC are 128 characters long. Since every STRING ends with an END character, a STRING can only be up to 127 characters long.
- Modules 9105 and 9106: Transfer a block of binary values (bytes) from the word memory
- Module 9107: Read bytes from the receiving buffer without erasing the buffer

STRINGS and binary data are transferred using ASCII characters.

Example: Transferring a binary block

Address	Value	ASCII character
.	.	.
B126	11111010	\$FA
.	10000001	\$81
.	.	
.	.	
.	.	

When transferring binary data starting from the address B126, the ASCII characters <F> <A> <8> <1> etc. are transmitted in sequence from the word memory through the interface. Each byte contains two ASCII characters. The transmitting and receiving buffers each hold 63 bytes.

Module 9100 Assign data interface

With Module 9100 you assign an interface to the PLC and specify the transfer parameters. They initialize the interface, thereby erasing any errors that may have occurred. The interface is then ready to receive.

Once assigned to the PLC, the interface is disabled for use by the input/output program of the user interface. The assignment is canceled when the PLC program is recompiled.

Can only be called in a submit job or spawn job!

Call:

```
PS    B/W/D/K <>Interface>
        0: RS232
        1: RS422
PS    B/W/D/K <>Transfer parameters>
        0: From MP50x0.2
        1: From MOD function
        2: From MP50x0.3
CM    9100
```

Error code:

Marker	Value	Meaning
M4203	0	Interface was assigned
	1	Error code in W1022
W1022	1	Incorrect interface or incorrect transfer parameter
	13	No connection
	14	Interface busy or input/output not ready
	17	Incorrect data transfer rate
	20	Module was not called in a spawn job or submit job



Module 9101 Release data interface

Module 9101 cancels the assignment of an interface to the PLC. The receive mode of the interface is canceled.

Can only be called in a submit job or spawn job!

Call:

```
PS    B/D/W/K  <>Interface>
        0: RS232
        1: RS422
```

```
CM    9101
```

Error code:

Marker	Value	Meaning
M4203	0	Interface enabled
	1	Error code in W1022
W1022	1	Incorrect interface
	14	Interface not assigned
	20	Module was not called in a spawn or submit job

Module 9102 Status of data interface

Module 9102 reads the status information about an interface in bit-coded form.

The information "interface ready" is updated when the interface is assigned to the PLC or NC. If the interface is not assigned, the module reads the last valid status.

Call:

```
PS    B/W/D/K  <>Interface>
        0: RS232
        1: RS422
```

```
CM    9102
```

```
PL    B/W/D    <>Interface status>
        -1: Error code in W1022
        Bit 0: Interface is assigned
        Bit 1: Interface is assigned to the PLC
        Bit 2: Interface is ready
        Bit 3: Transmit buffer is empty
        Bit 4: Error during transmission
        Bit 5: Receive buffer is full
        Bit 6: Error in reception
        Bit 7: ETX was received (not ready to receive)
        Bit 8: Internal buffer from Module 9113 contains characters
```

Error code:

Marker	Value	Meaning
M4203	0	Status read
	1	Error code in W1022
W1022	1	Incorrect interface

Module 9103 Transmit string through data interface

You must first assign the interface to the PLC and initialize it with Module 9100.

Module 9103 transmits a string from a string memory through one of the two interfaces. Links to the PLC error file and PLC dialog file are deleted.

Can only be called in a submit job or spawn job!

Call:

PS B/W/D/K <>Interface>
 0: RS232
 1: RS422

PS K/B/W/D <>Number of source string in the string buffer>
CM 9103

Error code:

Marker	Value	Meaning
M4203	0	String sent
	1	Error code in W1022
W1022	1	Incorrect interface or incorrect string number
	12	No string end found
	13	Interface not ready
	14	Interface not assigned
	15	Transmit buffer not empty
	20	Module was not called in a submit or spawn job



Module 9104 Receive string through data interface

You must first assign the interface to the PLC and initialize it with Module 9100.

Module 9104 reads a string from the receive buffer of a serial interface in a string memory and resets the receive buffer.

Can only be called in a submit job or spawn job!

Call:

```
PS    B/W/D/K <>Interface>
      0: RS232
      1: RS422
```

```
PS    K/B/W/D <>Number of the string in the string buffer>
CM    9104
```

Error code:

Marker	Value	Meaning
M4203	0	String was received
	1	Error code in W1022
W1022	1	Incorrect interface or incorrect string number
	12	String too long
	14	Interface not assigned
	16	Receiving buffer empty
	18	Transmission error or input/output not ready
	20	Module was not called in a submit or spawn job

Module 9105 Transmit binary data through data interface

You must first assign the interface to the PLC and initialize it with Module 9100.

Module 9105 transmits a block of binary values from the word memory of the PLC to one of the two interfaces. The transfer is in the form of ASCII-coded hexadecimal values. Every byte in the source block makes two ASCII characters at the interface.

Can only be called in a submit job or spawn job!

Call:

PS B/W/D/K <>Interface>
0: RS232
1: RS422

PS K/B/W/D <>Number of the first byte in the binary block>

PS K/B/W/D <>Length of the binary block (0 to 63)>

CM 9105

Error code:

Marker	Value	Meaning
M4203	0	Data was transmitted
	1	Error code in W1022
W1022	1	Incorrect interface or incorrect byte number or block too long
	4	Block outside value range
	13	Interface not ready or no connection
	14	Interface not assigned
	15	Transmit buffer not empty
	20	Module was not called in a spawn or submit job



Module 9106 Receive binary data through data interface

You must first assign the interface to the PLC and initialize it with Module 9100.

Module 9106 reads a block of binary values from one of the two interfaces into the word memory of the PLC. The transfer is in the form of ASCII-coded hexadecimal values. Every two ASCII characters from the serial interface make one byte in the binary block.

The length of the read binary block is returned as the initial variable.

Can only be called in a submit job or spawn job!

Call:

PS B/W/D/K <>Interface>
 0: RS232
 1: RS422

PS K/B/W/D <>Number of the first byte in the binary block>

CM 9106

PL B/W/D <>Length of binary block in bytes>
 -1: Incorrect module call

Error code:

Marker	Value	Meaning
M4203	0	Data was received
	1	Error code in W1022
W1022	1	Incorrect interface or incorrect byte number or block too long
	4	Block outside value range
	11	Odd number of characters or illegal character
	12	String too long
	14	Interface not assigned
	16	Receiving buffer empty
	18	Transmission error or input/output not ready
20	Module was not called in a submit or spawn job	

Module 9107 Read from receiving buffer

You must first assign the interface to the PLC and initialize it with Module 9100.

Module 9107 reads two ASCII characters from the receive buffer to one of the two interfaces and codes them to a binary value.

You can specify an offset that corresponds to the position of the byte to be read in a binary block read by Module 9106. The contents of the receiving buffer are retained and can be read by Modules 9104 and 9106.

Can only be called in a submit job or spawn job!

Call:

PS B/W/D/K <>Interface>
0: RS232
1: RS422

PS B/W/D/K <>Offset of byte to be read in binary block>

CM 9107

PL B/W/D <>Binary value read>

Error code:

Marker	Value	Meaning
M4203	0	Receiving buffer was read
	1	Error code in W1022
W1022	1	Incorrect interface or incorrect byte number
	11	Illegal character
	12	String too long or offset too large
	14	Interface not assigned
	16	Receiving buffer empty
	18	Transmission error or input/output not ready
	20	Module was not called in a submit or spawn job



Module 9110 Transmit a message via LSV2

Module 9110 transmits a message (binary data or string) to a host computer connected by LSV2 protocol.

The message is transmitted to the host by the LSV2 command "M_PC<msg.l>".

Call:

PS B/W/D/K <>Data type>
0: Binary data double word
1: String

PS B/W/D/K <>Source address>
With binary: Number of the double word
With string: Number of the string

CM 9110

PL B/W/D <>Error code>
0: Message is being transmitted
1: No connection to host
2: Transmit buffer full
3: Incorrect data type (not 0 or 1)
4: Incorrect source address
5: String to be sent is too long (>= 128 characters)

Error code:

Marker	Value	Meaning
M4203	0	Message was transmitted
	1	Error code in W1022
W1022	2	Incorrect data type
	4	No double word address, or incorrect string number
	11	String too long
	13	No connection
	15	Transmit buffer not empty
	16	Receiving buffer empty

Module 9111 Receive a message via LSV2

Module 9111 reads a message (double word or string) that has been received from a host computer connected by LSV2 protocol.

The message must be transmitted from the host by the LSV2 command "M_PC<msg.l>".

Call:

PS B/W/D/K <>Data type>
0: Binary data double word
1: String

PS B/W/D/K <>Target address>
With binary: Number of the double word
With string: Number of the string

CM 9111

PL B/W/D <>Error code>
0: Message was read
1: No connection to host
2: No message of this type in receiving buffer
3: Incorrect data type (not 0 or 1)
4: Incorrect target address
5: String to be received is too long (>= 128 characters)

Error code:

Marker	Value	Meaning
M4203	0	Message received
	1	Error code in W1022
W1022	2	Incorrect data type
	4	No double word address, or incorrect string number
	11	String too long
	13	No connection
	15	Transmit buffer not empty
	16	Receiving buffer empty



Module 9112 Transmit ASCII characters via data interface

You must first assign the interface to the PLC and initialize it with Module 9100. Module 9112 transmits a single ASCII character.



Note

Set MP5030.x = 2 so that the transmitted characters do not disturb the set protocol procedure.

Define the characters in at least one word so that the values to 255 can be recognized.

Can only be called in a submit job or spawn job!

Call:

PS B/W/D/K <>Interface>

0: RS232

1: RS422

PS W/D/K <>ASCII code [0 to 255]>

CM 9112

Error code:

Marker	Value	Meaning
M4203	0	Character was transmitted
	1	Error code in W1022
W1022	1	Incorrect interface
	13	Interface not ready or no connection
	14	Interface not assigned
	15	Transmit buffer not empty
	20	Module was not called in a submit or spawn job

Module 9113 Receive ASCII characters via data interface

You must first assign the interface to the PLC and initialize it with Module 9100.

Module 9113 reads a single ASCII character from the receiving buffer of a serial interface and resets the receiving buffer.

If there is more than one character in the receiving buffer, the first is returned and the others are stored in a special buffer.

You can interrogate the current state with Module 9102, bit 8.

As long as data remains in the buffer, no further characters are collected from the interface.

If $MP5030.x < 2$, the characters cannot be read from the interface until the line with the character requested in the protocol has been executed.



Note

Store the result in at least one word so that the values up to 255 will be recognized.

Can only be called in a submit job or spawn job!

Call:

PS B/W/D/K <>Interface>
 0: RS232
 1: RS422

CM 9113

PL W/D <>ASCII character read>
 [0 to 255] = ASCII characters; [-1] = error

Error code:

Marker	Value	Meaning
M4203	0	Character was received
	1	Error code in W1022
W1022	1	Incorrect interface
	12	String too long
	13	Interface not ready or no connection
	14	Interface not assigned
	16	Receiving buffer empty
	18	Transmission error or input/output not ready
	20	Module was not called in a submit or spawn job
37	Receiving queue is full	



10.11 External Programming

Please remember the following when programming externally for subsequent transmission:

- At the program beginning and after every program block, <CR><LF> or <LF> must be programmed.
- After the End of Program block, <CR><LF> and also <EXT> must be programmed.
- For NC programs, the spaces can be omitted between the individual words.
- When reading-in DIN blocks, the asterisk character (*) is not required at the end of the block.
- Comments are separated from the NC block with a semicolon (;).
- Comments located before the program are not saved.
- With conversational programming, the block numbers are generated by the iTNC. They need not be programmed.





11 iTNC 530 and Windows

11.1 General information and important notes



Note

Microsoft Windows, Windows 2000, Windows XP, Windows Vista, Windows 7 and Internet Explorer are registered trade marks of Microsoft Corporation.



Note

Please pay attention to the Microsoft End User License Agreement (EULA), which is included with your product and add it to your machine documentation.

11.2 IPC 6341 – Industrial PC with Windows 7

With the IPC 6341 Windows computer you can start and remotely operate Windows-based applications via the iTNC's user interface. The user interface is displayed on the control screen. Software option 133 must be enabled for this purpose.

Since Windows runs on a separate computer, it does not influence the NC machining process. The Windows computer is connected to the NC main computer via Ethernet. A second monitor (VGA) is only necessary for installation and maybe also for maintenance purposes because the Windows applications are displayed on the monitor of the iTNC via remote accesses.



Note

Please pay attention to the Microsoft End User License Agreement (EULA), which is included with your product and add it to your machine documentation.

When using the IPC 6341, please note the following:

- HEIDENHAIN offers no support for the installation of non-HEIDENHAIN software and cannot guarantee the function of Windows applications.
- HEIDENHAIN is not liable for faulty hard disk contents caused by installing updates to non-HEIDENHAIN software or additional application software.
- If such changes in programs or data make service visits from HEIDENHAIN necessary, the service costs will be invoiced.
- The machine manufacturer or end user is responsible for any changes to the preinstalled Windows system (this especially regards changes to system settings and the installation of updates or additional software).

IPC 6341: Industrial PC for installation in electrical cabinet

Pin layout	Connector	Function	
	X29	Reserved	
	X26, X116	Ethernet data interface	Like MC 6xxx
	X27	RS-232-C/V.24 data interface	Like MC 6xxx
	X141, X142, X143, X144	USB 2.0 interface (Type A)	1933
	X101	+24 V NC power supply	Like MC 6xxx
	X500	Reserved	
	X501	Reserved	
	X600	Reserved	
	X249	VGA interface for monitor	1933
	X121	Reserved	
	X125	Reserved	
	X601	Reserved	
	⊕	Protective ground	Like MC 6xxx



Attention

Do not engage or disengage any connecting elements while the unit is under power!



Note

Apart from that, the same operating conditions apply as for the MC 6xxx main computer units.



X249: For commissioning and maintenance purposes, a VGA monitor with a resolution of 1280 x 1024 pixels can be connected to X249.

X141 to X144: For commissioning and maintenance purposes, USB devices (for example mouse and keyboards) can be attached to connectors X141 to X144.

Pin layout for USB 2.0 connection (Type A):

USB connection (female) 4-pin	Assignment
1	+5 V
2	USBP-
3	USBP+
4	GND

VNC access and screen resolution

The VNC server "Ultra VNC" is already installed on the IPC 6341. The password for connection to the VNC server is "123". Please change this password on the IPC and TNC during commissioning. The settings are described in "Programs/ Ultra VNC".

The programs listed with the following connection types are already installed on the IPC 6341 by the control:

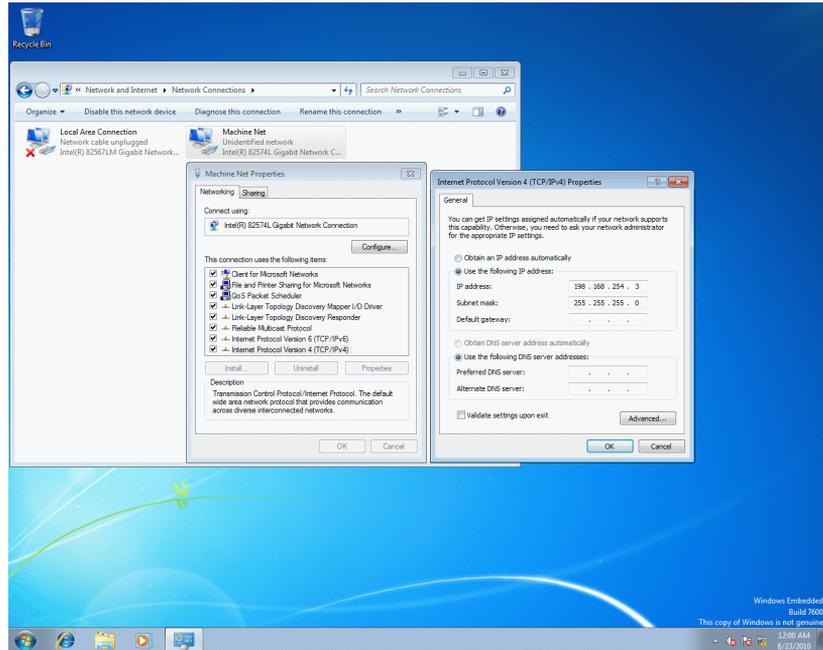
- VNC connection: UltraVNC Win32 Server 1.0.9.6.1
- RDP connection: 32-bit Windows 7 Embedded Standard
- NetShutdown connection: 32-bit Windows 7 Embedded Standard

Programs for the individual connection types are also preinstalled on the control. The versions of the above-mentioned programs, and of those installed on the control, may change with future software releases, when newer versions of the respective program packages are installed. The versions listed here are therefore meant as "at least version xxx".

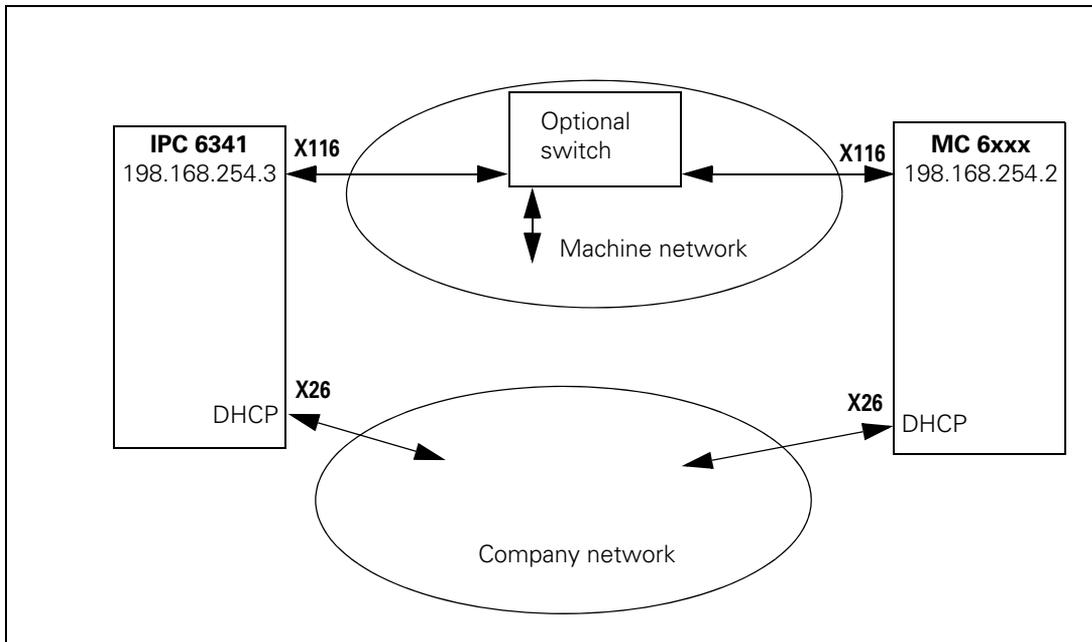
The default screen resolution of the IPC 6341 is 1280 x 1024 pixels. If required (BF 250), however, the resolution can be changed to 1024 x 748 pixels in the Windows settings. To do so, right-click in the desktop of Windows 7 and select the **Screen Resolution** menu item.

Network settings

In the standard version, the Ethernet interface X26 of the IPC 6341 is configured for DHCP. The second Ethernet interface is configured with a fixed IP address (198.168.254.3).



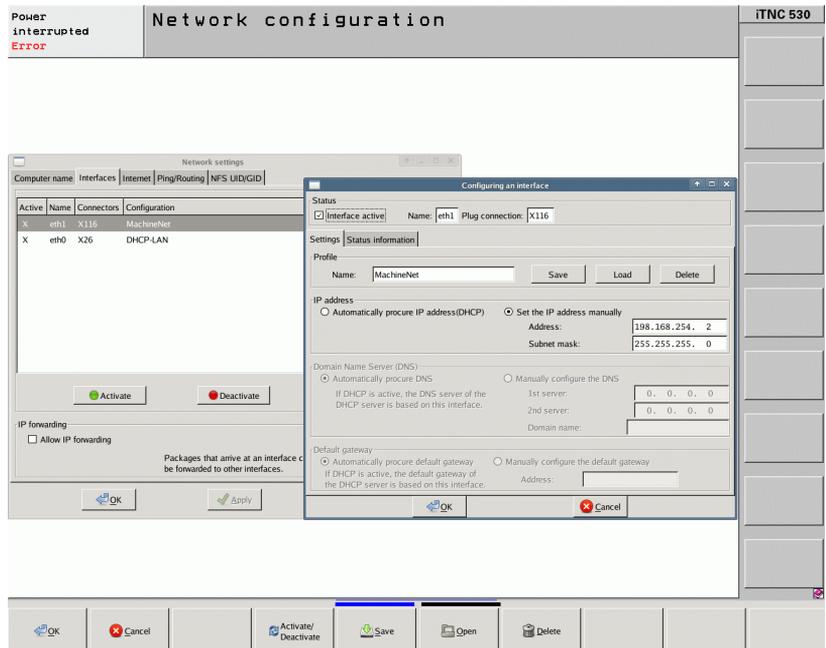
HEIDENHAIN recommends the following configuration for the connection of the IPC 6341:



To use the recommended configuration with the IPC 6341, configure the Ethernet interface "eth1/X116" of the NC main computer with the "MachineNet" profile that is already saved, and activate it. This gives the NC main computer the IP address 198.168.254.2.

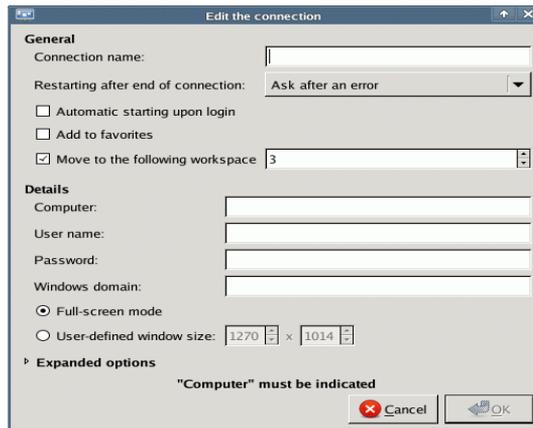
Proceed as follows to connect the IPC 6341 with an MC:

- ▶ Connect the two Ethernet connections X116 of the IPC and the MC with each other.
- ▶ Start up both main computers, IPC and MC.
- ▶ Switch to the network settings on the control via the MOD key and the code number NET123.
- ▶ Select the "**eth1/X116**" interface on the MC.
- ▶ Press the **Configuration** button to switch to the settings options.
- ▶ Press **Load** to select the existing configuration **MachineNet**.

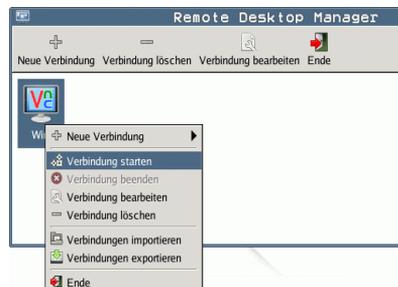


- ▶ Close the Settings with **OK**.
- ▶ Activate the **eth1/X11** interface.
- ▶ Close the Network Settings with **Apply**. When you have done this, you must restart the control to activate the changes.

- ▶ When the control has restarted, open the Remote Desktop Manager (software option 133) via the taskbar on the bottom of the screen.
- ▶ Press the **Add Connection** button and select the connection type **RDP**.



- ▶ In **Connection Name**, enter a name for the connection (e.g. Win7).
- ▶ In **Computer**, enter the IP address of the IPC 6341 (198.168.254.3) and in **Password**, enter the password for the VNC server (123). HEIDENHAIN recommends entering a new password when commissioning has been completed successfully.
- ▶ Close the dialog with **OK**. The new connection is now shown in the Remote Desktop Manager.



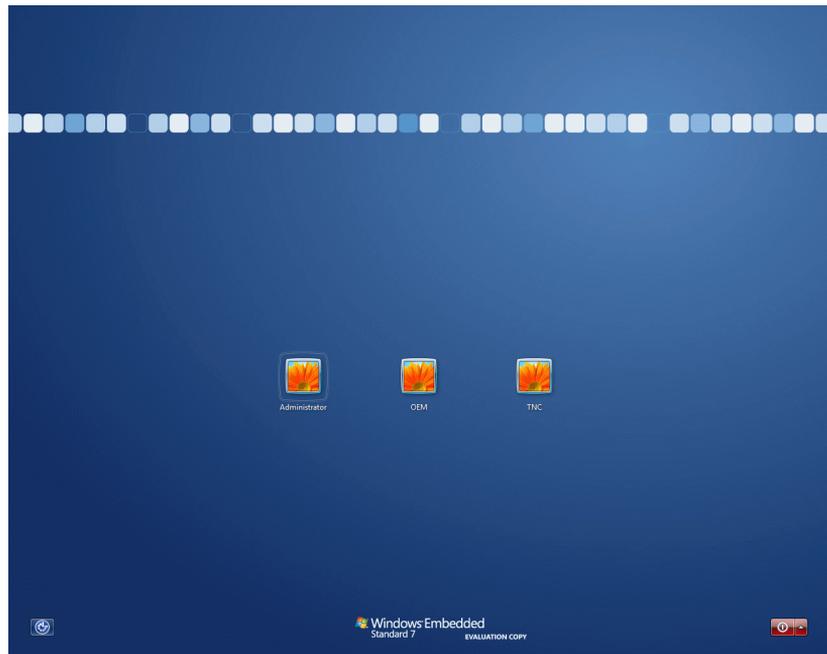
- ▶ Select the connection.
- ▶ Press the right mouse key to open a menu in which you can establish the connection to the IPC: **Start Connection**.
- ▶ When the connection has been established, a short message will be displayed on the screen.

- ▶ With the default settings, Windows 7 is now displayed on the third desktop and can be operated from there. You can switch to the third desktop via the taskbar or the DIADUR key on the keyboard unit.
The left DIADUR key opens the JH menu (operating system settings) of HEROS 5. Within a remote-desktop application, this key functions as a Windows key.
The right key switches to a remote-desktop application if the "Remote Desktop Manager" software option is enabled. If you happen to be in a remote desktop, this key switches to the next remote desktop or back to the NC application.
In a remote-desktop application, the screen switchover key has locally the same effect as an F12 key, but you have to use the right DIADUR key to switch back.

Remember to close Windows 7 on the IPC before shutting down the control. Failure to do this can lead to loss of data or damage to the Windows system during switch-off with the main switch of the machine.

Registered users

HEIDENHAIN has already installed the following users in Windows 7:



User name	Password	Description
TNC	123	Example user name for the end user ^a
OEM	807667	Administrator for the machine tool builder
Administrator	(not disclosed here)	Only for HEIDENHAIN Service ^b

- a. The end user must not be assigned to the Administrators group because otherwise it would have access through Windows to the PLC and SYS partition.
- b. The administrator must not be changed or deleted because otherwise HEIDENHAIN cannot offer any service.



Note

Pay attention to capitalization when entering the password.

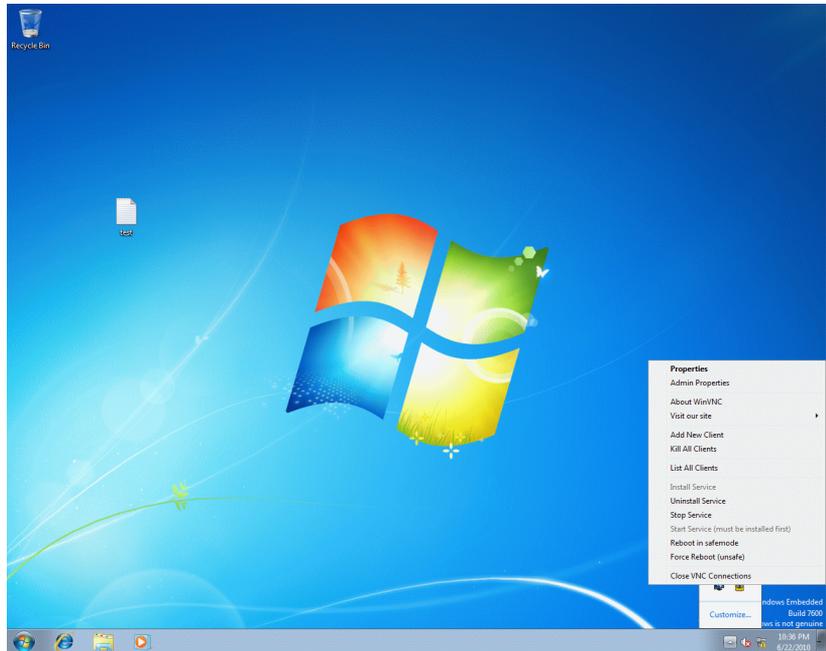
The Windows Help contains general information about user groups.

You need the administrator rights of the OEM or Administrator to be able to execute commands (e.g. PING function) via the Windows 7 prompt.

VNC password of the IPC

Proceed as follows to change the password of the VNC connection on the IPC:

- ▶ Log onto the IPC as a user with administrator rights, e.g. as **OEM**.
- ▶ Click the arrow in the taskbar (bottom right) with the left mouse key to display the background processes.
- ▶ Right-click the application **WinVNC** (symbol: eye on yellow background).
- ▶ Select **Admin Properties**.



- ▶ You can set the VNC password under **Authentication** on the **UltraVNC Server Property Page**.
- ▶ Take over the changes with **Apply** and close the window with **OK**.
- ▶ Restart the IPC to activate the changes.
- ▶ Switch back to the control screen.
- ▶ Open the Remote Desktop Manager and select the corresponding VNC connection.
- ▶ Right-click the VNC connection.
- ▶ Select **Edit Connection** and change the VNC **password** here, as well.
- ▶ After the IPC has rebooted, re-establish the connection with **Start Connection**.

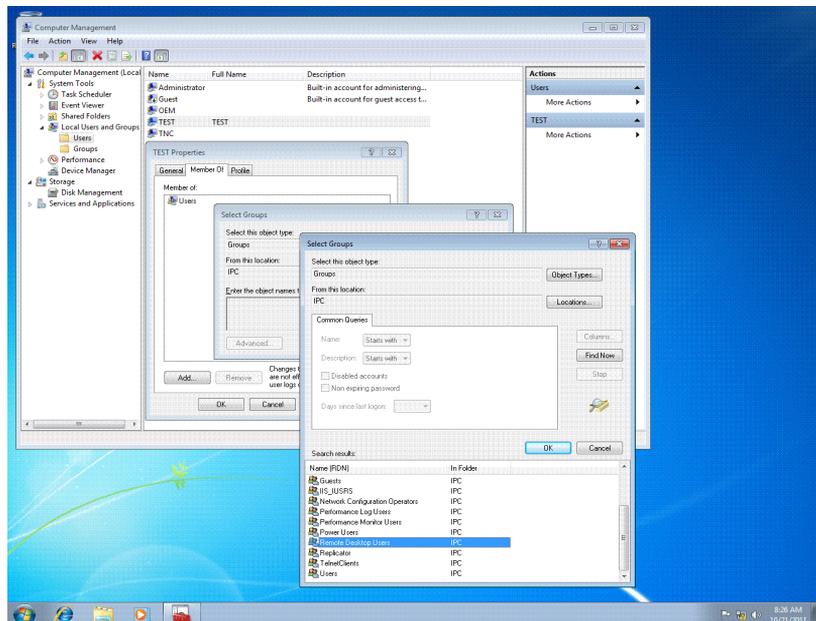
RDP settings of the IPC

Proceed as follows to configure an RDP connection on the IPC or to change the settings of the RDP connection:

- ▶ Log onto the IPC as a user with administrator rights, e.g. as **OEM**.

First of all, a Windows user must be configured or changed. To do so, open the settings of the User Accounts:

- ▶ Open the **Start** menu.
- ▶ Open the **Control Panel**.
- ▶ Select the settings of the **User Accounts**.
- ▶ Configure a new user or change an existing user. Please remember to enter a password for the user because remote login is only possible with a password.
- ▶ Then switch to **Start|Control Panel|System and Security|Administrative Tools|Computer Management**, to add the new user to the **Remote Desktop Users** group.
- ▶ Open the **Properties** with the right mouse key.
- ▶ Click the **Add** button.
- ▶ Click the **Advanced** button.
- ▶ Click the **Find now** button.
- ▶ Select **Remote Desktop Users** from the list and confirm with **OK**.
- ▶ Then close the window with **OK** to save the selection.



Language settings on the IPC

Proceed as follows to change the system language of the IPC 6341:

- ▶ Log onto the IPC as a user with administrator rights, e.g. as **OEM**.
- ▶ Open the command window (console)
- ▶ Go to the "C:\LanguagePacks" directory.
- ▶ The "IpcLanguageSetup.bat show" command gives you a list of the installed languages. Additional help is available with the command "IpcLanguageSetup.bat" without additional parameters.
- ▶ The "IpcLanguageSetup.bat de-DE" command replaces the currently installed language by German, for example. Please note that every new installation takes approx. 10 minutes.
- ▶ The system restart to be performed afterwards takes approx. 2 minutes. After the restart, the selected language is automatically active.

The language setting of the IPC can only be changed as described above. The familiar Windows menu for the language settings is not supported by the IPC. The following languages can be activated:

- de-DE: German
- en-US: English
- es-ES: Spanish
- fr-FR: French
- it-IT: Italian
- ja-JP: Japanese
- ko-KR: Korean
- ru-RU: Russian
- zh-CN: Chinese (simplified)
- zh-TW: Chinese (traditional)



12 Error messages

12.1 DSP/NC error messages

You can find a list of all NC and DSP error messages on the HEIDENHAIN homepage (www.heidenhain.de) under:

- ▶ Services and Documentation
- ▶ Documentation / Information
- ▶ User documentation
- ▶ TNCguide
- ▶ Select the desired language, such as English.
- ▶ TNC Controls
- ▶ NC error messages

Here you will find the current, control-specific help files (*.chm) with all error messages. These files are also part of the TNCguide context-sensitive help system for the respective control.

12.2 iTNC Error Messages during Data Transfer

The following error messages can occur during data transfer through an interface:

General error messages	
Interface already assigned	The interface is already transmitting, or the transmission was not concluded.
Program incomplete	A transmission was broken off or the file was not correctly concluded (no END character or END block).
Ext. in-/output not ready	The interface is not connected, or the peripheral unit is switched off or defective.
Data transfer erroneous x	x = error code (see table).

Error codes:

Error code	Meaning
E	Bei Datenübertragung mit BCC wurde 15mal hintereinander <NAK> empfangen
A to H except E	Error code of the receiver module with one of the following causes: <ul style="list-style-type: none"> ■ The transfer-rate setting of the iTNC and peripheral device do not match. ■ The parity bit is erroneous. ■ Erroneous data frame (e.g.: no stop bit). ■ The receiver module of the interface is defective.
K	During transmission of an error to the iTNC, the <1> character was not transmitted after the <ESC> character.
L	After the error sequence <ESC><1> an incorrect error number was received (error numbers 0 to 7 are permitted).
N	An expected acknowledgment <ACK> or <NAK> was not transmitted by a certain time.
M	During data transfer with BCC, the <NAK> signal was transmitted 15 times in succession.

Codes K and L are shown only during transmission with the standard data transmission protocol.

12.3 Error messages of the file system

The following error messages can be displayed on the iTNC:

Error message	Corrective action
File system error 1	Inform your service agency.
File system error 2	
File system error 3	
File system error 4	
File system error 5	
File system error 6	
File system error 7	
File system error 8	
File system error 9	
File system error 10	
File system error A	
File system error B	
File system error C	
File system error D	
File system error E	
File system error F	
File system error G	
File system error H	
File system error I	
File system error J	
File system error K	
File system error L	





13 Subject Index

#

123	1265
531210	1265
555343	1265
688379	1265
75368	890, 1265
807667	1265
857282	1265
95148	1265

A

Acceleration.....	812
Acceleration feedforward control	853
Accessories	123, 134
ACTION	1374, 1386
Active Chatter Control (ACC).....	699
Active damping.....	852
Actual-to-nominal value transfer.....	847
Adapters for encoder signals.....	123, 304
ADD STRING (+).....	1823
ADDITION (+).....	1787
Additional control loops	83
Addressing of Operands.....	1747
ADVANCED SETUP.....	1683
Advanced spindle interpolation	991
AFC adaptive feed control.....	1218
Analog axes	842
Analog inputs.....	1529
Analog nominal value output.....	308
Analog outputs	1529
AND (A)	1775
AND NOT (AN)	1777
Arc end-point tolerance	1263
ASSIGN (=).....	1767
ASSIGN BYTE (B=).....	1768
ASSIGN NOT (=N).....	1770
ASSIGN TWO'S COMPLEMENT (=–)	1770
ASSIGN WORD (W=)	1769
AUTOSTART.....	1215
Auxiliary cycles	1258
Axes, algebraic sign.....	605
Axes, clamping.....	845
Axis coordinates, ascertaining.....	1150
Axis designation	604
Axis display.....	603
Axis information, reading.....	621
Axis keys IV and V.....	604
Axis selection	603
Axis-error compensation	647
AXISNUMBER =.....	1701

B

BACK	1373, 1386
Backlash compensation	647
Balluff	1603
Band-rejection filter	851, 1088
Band-rejection filter, CC 424(B)	1088
Bar diagram	1357
Behavior during program interruption	1216
Bend radius	418
BF 250	377
BF 260	379
BF 2xx	101
BIT CLEAR (BC)	1809
BIT SET (BS)	1808
BIT TEST (BT)	1810
BLANK	1373, 1386
Block Check Character	1910
Block elements, sorted machining	1216
Block number increment	1210
Block numbers	1210
Block scan	1244
Block scan, interrupting	1248
Brake test	926
Braking power	871
Braking ramp	871
Brief description	127
Buffer battery	294
Bus diagnosis	1320
Bus diagnostics screen content	1323

C

Cable diameter	418
Cables, instructions for wiring	418
Cables, specifications	418
Calibrating	1543
Calibration data	1541
Calibration data blocks	1541
Call Module (CM)	1816
CALL MODULE IF FALSE (CMF)	1817
CALL MODULE IF TRUE (CMT)	1816
Call submit (SUBM)	1832
Canceling a submit program (CAN)	1833
CASE branch	1844
C-axis operation	987
CC 424 (B)	1067
CC 61xx	84
Character set	1278
CHECK	1374
Circular interpolation test	1003
CLOSEPLCWINDOW	1377
CLOSEPOPUPMENU	1376

CMA-H 04-00-04.....	308
CMA-H modules.....	93
CML 110.....	123
CML 110, specifications.....	416
CMO, collision-monitored object.....	1442
Code numbers.....	1265
Collision monitoring.....	1424
Collision monitoring, activating/deactivating.....	1439
Collision monitoring, definition table.....	1442
Collision monitoring, exclusions.....	1436
Color settings.....	1267
Commissioning IPC and kV with TNCopt.....	1025
Commissioning linear and torque motors.....	1128
Commissioning the feedforward functions with TNCopt.....	1024
Commissioning the reversal-spike compensation with TNCopt.....	1025
Commissioning with TNCopt.....	1023
Commissioning, preparation.....	1020
Compensation of offset for tilting axes.....	770
COMPILE.....	1682
Components, overview of.....	76, 197
Conditional compilation.....	1684
Constants field.....	1841
Constants field (KF).....	1841
Contour velocity at corners.....	892
Control characters.....	1902
Control in operation (*).....	1159
Control loop.....	810
Control loops, enabling.....	140
Control of events.....	1836
Conversational language.....	1283
Conversational languages, additional.....	1285
Converter TTL to 1 Vpp.....	304
Cooling.....	187
Cooperative multitasking.....	1835
Coordinate system.....	1275
Counters.....	1751
Cross Talk Compensation (CTC).....	688
Crossover Position Filter (CPF).....	1104
Current controller.....	866
Current controller, commissioning.....	1026
Current controller, commissioning with TNCopt.....	1023
Cutting data calculation.....	1601
Cutting-data table.....	1602
Cycle 17.....	977
Cycle 18.....	979
CYCLE.SYS.....	1712
Cycles.....	1241, 1539
Cylindrical surface.....	777, 1243



D

Data access.....	1893
Data backup.....	179
Data interfaces.....	325, 1857
Data protection.....	1893
Data tampering.....	1893
Datum management.....	1208
Datum setting.....	1206
Datum shift.....	1534
Datum table.....	1409
Datums.....	762
DC-link voltage.....	1020
DCM.....	1424
DCM, activating and deactivating.....	1439
DCM, collision exclusions.....	1436
DCM, definition table.....	1442
DCM, fixture monitoring.....	1453
DCM, Test Run mode.....	1441
DCM, tool carriers.....	1477
DEBUGPATH =.....	1701
Decimal character.....	1285
DECREMENT (DEC).....	1792
Deleting.....	1212
Demo operation for axes.....	604
DHCP server.....	1887
Diagnosis.....	1298
Diagnostics log.....	1287
Differential factor.....	850
Dimensions BF 760.....	439
Dimensions for touch-probe adapter cable.....	458
Dimensions of adapter block for the data interface.....	449
Dimensions of adapter cable for TT 120/TS 220.....	459
Dimensions of BF 250.....	435
Dimensions of BF 260.....	437
Dimensions of CC 6106.....	440
Dimensions of CC 6108, CC 6110.....	441
Dimensions of CML 110.....	464
Dimensions of line-drop compensator.....	451
Dimensions of MB 720.....	425
Dimensions of MC 6222.....	423, 424
Dimensions of MC 6x41.....	421
Dimensions of PL 6001.....	445
Dimensions of PL 6xxx.....	444
Dimensions of PSL 130.....	446
Dimensions of SE 540.....	461
Dimensions of SE 640.....	460
Dimensions of TE 620.....	432, 433, 434
Dimensions of TE 630.....	431



Dimensions of TE 635Q	426
Dimensions of TE 645Q	427
Dimensions of TE 730	429
Dimensions of TE 735	430
Dimensions of TS 220	459
Dimensions of TS 440	462
Dimensions of TS 640	463
Dimensions of TT 130	457
Dimensions of UEC 11x	443
Dimensions of USB hub	450, 464
Direction of rotation	957
Display step	1148
Distance	812
DIVISION (/)	1790
Documentation, User's Manuals	126
Double reference run	802
Double-speed control loop	1072
Drive test	1318
DriveDiag	1299
Drives, braking	870
DSP error messages	1943
Dual-spindle operation	985
Dynamic determination of load	1093

E

Electromagnetic compatibility	182
Electronic ID label	1319
Emergency stop monitoring	930
EMODE	382, 1376
ENABLE	1376
Encoder monitoring	610
Encoder signals	304, 608, 1069
Encoders	296, 606, 1109
END	1374
END OF MODULE (EM)	1818
END OF MODULE IF FALSE (EMF)	1818
END OF MODULE IF TRUE (EMT)	1818
EnDat	801
ENDSKMENU	382, 1373
EQUAL TO (==)	1793
EQUAL TO STRING (==)	1825
Error list	1177
Error messages	1176, 1943
Error messages during data transmission	1944
Error messages of the file system	1945
ESD precautions	183
Ethernet	1863
EXCLUSIVE OR (XO)	1783
EXCLUSIVE OR NOT (XON)	1785
Expanded compensations	688
Expansion PL	90
EXTERN statement	1848
External reference pulse	799

F

FAILTEST.....	1265
Fast PLC input.....	1753
Feature content level (FCL)	150
Feed rate limitation.....	839
Feedback control with following error (servo lag)	833
Feedback control with velocity feedforward	835
Feedback control with velocity semifeedforward	837
Feed-rate display	1155
Feed-rate enable.....	846
Feed-rate smoothing	819
Field orientation.....	1019, 1108
Field-angle definition	1121
File management.....	1670
Files	1406
Firmware	169
FirstInGroup.....	1377
Fixed cycles.....	1241
Fixture monitoring	1453
FIXTURE.FUNCTION =.....	1702
FN14	
ERROR =.....	1701
FN17	
SYSWRITE	1716
FN18	
SYSREAD	1725
FN19	
PLC =	1713
FN19GROUP =	1250
FN20	
WAIT FOR	1740
FN25	
PRESET	1206
FN31	
RANGE SELECT	1247
FN32	
PLC PRESET	1247
FNERRFIX =	1701
FNERROR =	1701
FNERRREASON =	1701
Format instructions	1359
Formula in machine parameters.....	479
Free rotation	1152
Freely definable table	1409
Frequent flexing	418
Friction, sliding	662
Friction, stick-slip.....	661, 1106
Functional safety (FS)	174



G

Gantry axes	778
Gear ranges	958
Gear shifting	959
GLOBAL statement	1848
Graphic display	1275
GREATER THAN (>)	1795
GREATER THAN OR EQUAL TO (<=)	1797
GREATER THAN OR EQUAL TO STRING (>=)	1826
GREATER THAN STRING (>)	1825
Grounding diagrams	465
GROUP=	1249

H

Handshaking	1862
Handwheel	383, 1502
Handwheels	106
Hard disk, checking	1690
Hard-disk organization	1689
HDL interface	329
HDR	195, 196
Heat generation	187
Heat sink temperature	906
Help	1189
Help system, context-sensitive	1195
HeROS 5	172
Hirth coupling	1532
Holding torque	857
Housing fan, MC/CC	907
HR 130	111, 413, 1510
HR 150	112, 1524
HR 410	106, 108, 383, 411, 1510
HR 410FS	106
HR 420	411, 1513
HR 520	383, 1516
HR 520FS	106
HR 550FS	393, 1518
HR 550FS handwheel with wireless transmission	393
HR 550FS wireless handwheel	1518
HR 5xx	106
HRA 110	112, 414, 1524
HROOT	382, 1376
HSCI configuration	169, 291
HSCI diagnosis	1322

I

I/O-FORCE LIST.....	1674
I2t monitoring.....	908
ID number for HR 130.....	111
ID number of HR 150.....	112
ID number of HR 410.....	108, 109, 110, 111
ID number of HR 520.....	109
ID number of HR 550FS.....	110
ID number of HR 551FS.....	111
IF...ELSE...ENDI.....	1843
include.....	1373, 1375
INCREMENT (INC).....	1792
INDEX register (X register).....	1819
Indexed tools.....	1596
Individual tooth measurement.....	1558
Input fields.....	1367
Input format for machine parameters.....	478
Installation elevation.....	188
Interpolation.....	607
Interpolator.....	814
Inverter status.....	922
IOCCFG =.....	1702
IOconfig.....	291
IPC.....	854
IPC 6110, hardware overview.....	124
IPC 6120, hardware overview.....	125
IPC 6341.....	113
IPC 6xxx.....	123
iTNC 530 with Windows 2000/XP.....	1931

J

Jerk.....	812
Jog increment.....	1530
JUMP (JP).....	1814
JUMP IF FALSE (JPF).....	1815
JUMP IF TRUE (JPT).....	1815

K

Key symbols.....	114
Keyboard unit, TE 6xx.....	96
Keystroke simulation.....	1392
KINEMATIC.....	1265
KINEMATIC =.....	1702
Kinematics.....	714
Kinematics, assignment table.....	718
Kinematics, definition table.....	1442
Kinematics, description table.....	720
Kinematics, PLC.....	731
Kinematics, SUBFILE.....	727
Kinematics, table overview.....	715
Kinematics, TOOLFILE.....	725
KinematicsComp.....	664
KinematicsDesign.....	1488
KinematicsOpt.....	746



L

LABEL (LBL)	1818
Large PLC window	1358
LARGEWINDOW.....	1376
Laser cutting machines	1561
LESS THAN (<)	1794
LESS THAN OR EQUAL TO (<=)	1796
LESS THAN OR EQUAL TO STRING (<=).....	1826
LESS THAN STRING (<)	1825
LIFTOFF.....	1098
LIFTOFF function.....	1217
LIFTOFF in case of power failure	1098
Line drop compensator	1124
Linear axis error.....	651
Linear motor	1122
Lines, deleting.....	1212
LOAD (L).....	1761
Load Adaptive Control (LAC)	41, 694
LOAD BYTE (LB)	1766
LOAD DOUBLE WORD (LD)	1767
LOAD NOT (LN).....	1763
LOAD STRING (L).....	1823
LOCKEDOPTIONS =	1702
LOG	1265
Log file.....	163
Log, writing to	1295
Logic diagram	1680
Logical file names.....	1757
LOGO =	1702
LOGOSP =	1702
Logs.....	1286
Look-ahead	1210
Low-frequency oscillations	852
Low-pass filter.....	850, 1088
Low-pass filter, CC 424(B).....	1088
LSV2	1913
LSV2TIME0 =	1702
LSV2TIME1 =	1702
LSV2TIME2 =	1702
Lubrication pulse	629

M

M functions	1157, 1251
M06	1256
M103	1256
M112	893
M114	770
M124	893
M128	771, 831
M134	1257
M144/M145.....	775
M150	629
M200	1561
M201	1561
M202	1562

M203	1563
M204	1563
M89	1256
M90	893
Machine datum	1205
Machine operating panel, MB 6xx.....	103
Machine Parameter Programming	477
Machine parameter subfile.....	493
Machine parameters, automated updating	155
Machine parameters, overwriting	483
Magazine rules	1580
Main operating mode	1165
Master-slave torque control	782
Material table.....	1602
Mating dimensions of the BF 750	436
Mating dimensions of the TE 720	428
Mating dimensions of the UMC 111	442
MB 6xx	103
MC 6xxx	76
Measurement log	1546, 1549
Measuring tolerance.....	1545
MFIRST=	1249
MFUNCT.TAB.....	1258
MGROUPS.SYS.....	1249, 1712
Milling heads, changing.....	745
MLAST =	1249
MMODE	382, 1376
MODEHELP =	1702
Modules for string processing.....	1828
Monitoring functions	894, 1087
Monitoring the 5 V supply	232
Motion Adaptive Control (MAC)	697
Motor brake.....	925
Motor brake test.....	926
Motor brake, testing.....	926
Motor table.....	1009, 1140
Motor temperature.....	905
Motor test	1318
Mounting attitude.....	189, 190, 191, 192
Movement monitoring.....	899
MP, overwriting.....	483
MPFILE =	1702
MPFRAGMENT =	1703
MPLOCKFILE =	1703
MPPASSWORD =	1703
MS 110.....	122
MS 111.....	122
MS 11x.....	122
MSPLIT.SYS	1250, 1712
Multifunction filter	1088
MULTIPLICATION (X).....	1789
Multiturn.....	897
Multiturn overflow.....	897



N

NC macro	1258, 1499, 1500, 1501, 1605
NC program cancelation	1214
NC program display	1212
NC program end	1213, 1244
NC program, checking	1212
NC software, activating	161
NC software, deleting	162
NCMACRO = TC, M	1250
NCMACRO.SYS	1709
NET123	1265
Network settings	1864
NODE	1373, 1386
Nominal position value filter	821
Nominal position value filters, selection criteria	823
Nominal position value filters, settings	825
Nominal position value filters, tips	826
Nominal speed value	953
Nominal speed value monitoring	898
Nonlinear axis error	652
NOT EQUAL (<>)	1798
NOT EQUAL TO STRING (<>)	1827
Number conversion	1852
NUMBERMP4111 =	1704
NUMBERMP4230 =	1704

O

OEM log	1288
OEM logo	1260
OEM.SYS	1701
OEMCYC.ZIPNAME =	1704
Offset adjustment	890
Offset of tilting axes	701
Open-loop axes, inclined milling	776
Open-loop axes, tilting	776
Operands	1744
Operating modes	980, 1165
Operating system	172
Operating times	1169
Options	83
Options for the OEM	149
Options, enabling	140
Options, overview	136
Options, temporary enabling	147
OR (O)	1779
OR NOT (ON)	1781
ORDER = PRIO	1250
Oriented spindle stop	968, 1553
Oscilloscope	992
Oscilloscope recording, saving/loading	1000
OVERWRITE STRING (OVWR)	1824

P

PALETTPRESET =	1704
Pallet management	1490
Parity bit	1860
Partitions	1689
PET table	1177
PGM (global program settings).....	1236
Phase increase, CC 424(B).....	1088
PL 61xx.....	90
PL 62xx.....	89
PLA-H modules	91
PLB 6001	105, 341
PLC axes	630, 1149
PLC axes, coupling function.....	635
PLC commands.....	1758
PLC cycle time	1087, 1661, 1668
PLC error messages.....	1176
PLC files	1419
PLC functions	1669
PLC inputs	1526
PLC log.....	1286
PLC main menu.....	1667
PLC module interface for 18 axes	1329
PLC outputs.....	265, 1526
PLC partition.....	1689
PLC partition, encrypting.....	1690
PLC pop-up window	1192
PLC positioning	640
PLC program format.....	1755
PLC program structure	1756
PLC soft keys	1372
PLC window	1354
PLCCOMPCFG =.....	1704
PLCDIALOG =	1704
PLCE partition.....	1690
PLCERRFIX =	1704
PLCERROR =	1704
PLCERRREASON =	1704
PLCERRTAB =.....	1705
PLCEVENTS =	1705
PLCMAIN =	1705
PLCPASSWORD =	1705
PLCPWM =	1705
PLCSOFTK.SYS.....	1712
PLCSOFTVERS =	1705
PLD-H modules	91



Pocket coding.....	1608
Pocket milling.....	1241
Pocket table.....	1565
Pop-up window	1192
POPUPMENU.....	1376
Position Adaptive Control (PAC)	693
Position control loop, opening	844
Position controller.....	832
Position controller cycle time	832
Position controller, commissioning	1038
Position display.....	1147
Position encoders.....	614, 616, 1069
Position loop resolution	842
Position monitoring	896
Positioning window	900
Power interruption.....	1263
Power limit	873
Power module table	1008
Power supply unit, monitoring	902
Power supply, internal.....	907
Powering up the control	1260
PR.LINESLOCKED =	1208, 1705
Preset table	1208
Principal axes.....	604
Print masks.....	1546
Process monitor	1837
Products, overview of	76, 197
Profibus	288, 289
Program interruption	1216
Programming station	1267
Project file	1372
Protection, degrees of.....	182
Prototype.....	1410
PSL 130	237
PSL 135	237
PSL 13x	94
PSL 13x power supply unit.....	94
PSL13x low-voltage power supply unit	237
PT 100	285
PT2 element	851
PULL (PL)	1812
PULL LOGICACCU (PLL).....	1813
PULL WORDACCU (PLW).....	1814
PULSE	1374, 1386
PUSH (PS)	1811
PUSH LOGICACCU (PSL).....	1812
PUSH WORDACCU (PSW).....	1813
PWMPARAMETER =	1705
Python	1856



Q

Q parameters	1714
--------------------	------

R

Radial acceleration.....	891
RADIO	1374
Radio environment	405
Radio transmission	396
Radius compensation	1264
Rapid traverse	839
Reactor	1012
READ FROM PLC KEY	1741
Reference end position	800
Reference mark.....	796
Reference pulse	799
Reference-point setting with PLANE and Hirth axes	763
REMAIN = OUTPUT	1249
REMAINDER (MOD)	1791
Remote Desktop Manager	1338
REMOTE.LOCKSOFTKEYVISIBLE =	1705
REMOTE.PLCPASSWORDFORCED =	1705
REMOTE.PLCPASSWORDNEEDED =	1705
REMOTE.TIMEKEEPIDLE =	1706
REPEAT...UNTIL	1843
Replacement tool	1594
RESET (R)	1772
RESET NOT (RN)	1774
Returning to the contour	1244
Reversal spikes	661
Rigid tapping.....	977
Rotary axes.....	605, 1149
RUNTCH =	1706

S

S code	966
Safety precautions.....	181
Safety zone.....	1554
Scaling factor	1243
Screen logo	1260
Screen mask.....	1369
Screen saver.....	1337
SE 640	120
Secondary linear axes.....	606
SELinux	175
Series reactor	1012
Service file.....	1185
Service pack	167
SERVICE.REQUEST.CONTENT =	1706
SERVICE.REQUEST.HOST =	1706
SERVICE.REQUEST.PERIOD =	1706
SERVICE.REQUEST.PORT =	1706
SERVICE.REQUEST.TIMEOUT =	1706



SET (S).....	1771
SET NOT (SN).....	1773
SHIFT LEFT (<<).....	1806
SHIFT RIGHT (>>).....	1807
Shipping brace.....	195
Shutting down the control.....	1261
Signal period.....	606
SIK.....	80, 140, 195, 196, 1265
SIK MC 62xx.....	82
SIK, read out.....	1734
Single-speed control loop.....	1072
SKMENU.....	382, 1373
SKPATH.....	1373
Small PLC window.....	1354
SMALLWINDOW.....	1377
smarT.NC.....	1165
SMB signing.....	1863
SOFT KEY.....	1373
Soft keys.....	1372
Soft keys, HR 420.....	1385
Soft keys, HR 5xx.....	1385
Soft-key project file.....	1372
SOFTKEYPRESS =.....	1707
SOFTKEYPROJECT =.....	1372, 1706
Software designation.....	139
Software exchange, automating.....	153
Software exchange, general tips and notes.....	152
Software limit switches.....	628
Software options.....	136
Software releases.....	180
Special characters.....	1277, 1363
Special commands.....	1363
Special functions.....	1088
Special tools.....	1593
SPECIAL=.....	1249
Specifications.....	127
Speed command outputs.....	614, 616
Speed controller.....	848
Speed controller, commissioning.....	1031
Speed controller, commissioning with TNCopt.....	1024
Speed controller, low-pass filter order.....	1093
Speed encoder.....	1070
Speed encoders.....	614, 616
SPI analog module.....	308
SPI expansion modules.....	93
Spindle.....	944
Spindle control.....	952
Spindle override.....	961
Spindle speed.....	965
SPINDLEGROUP =.....	1250
Standard color settings.....	1272
Standard log.....	1289
Standstill monitoring.....	900



STARTUPCANCEL =	1248
STATUS	1376, 1386
Status display	1147, 1161
Status information	1161
Status submit (RPLY)	1832
Storage temperature	133
STORE STRING (=).....	1823
Subdivision factor	1502
SUBFILE	720
Submit programs.....	1831
SUBTRACTION (-).....	1788
Supply module table.....	1011
SW option: Control loop groups	83
SW options.....	136
Switch positions	1525
Switching inputs.....	1526
Switching outputs	1526
Symbols used in this manual.....	71
Synchronized axes.....	778
SYS partition.....	1689
System files.....	1701
System PL.....	89
System time	1174

T

TABCMA =	1707
Table.....	1676
Tapping with floating tap holder.....	974
TCHRULES =	1707
TCPM	771
TDEFGROUP =	1250
TE 635Q	359
TE 645Q	360
TE 6xx.....	96
TE 740	364, 367, 370
TE 745	373
Temperature monitoring.....	904
Temperature sensor.....	1125
Temporary input values	1020, 1022
Thermal expansion	659
Thermal expansion of tilting axes.....	741
Tilting and swivel axes	701
Tilting axis combination	765
Timer	1748
TNC partition	1689
TNC.SYS.....	1712
TNCguide.....	1195
TNCopt	1023
TNCOPT.LOCKSOFTKEYVISIBLE =	1707
Tool axis	622, 1148
Tool breakage	1559
TOOL CALL.....	1603



Tool change	1565
Tool changer.....	1603
TOOL DEF	1603
Tool life.....	1594
Tool magazines.....	1590
Tool management	1641
Tool measurement	1552
Tool recognition.....	1603
Tool table.....	1565
Tool, retracting	1217
Tool-carrier kinematics	1477
Tool-change macro.....	1605
TOOLFILE.....	720
TOOLGROUP =	1250
Tool-usage file	1599
Tool-usage test.....	1597
Topology and cables, HSCI.....	207
Torque limit	873
Torque motor.....	1123
Torsion compensation	663
Touch probe	313, 1535
Touch probe calibration	1543
Touch probes.....	119
Touchpad, locking and enabling	1405
Trace.....	1678
Transfer of variables, blockwise NC/PLC	1724, 1736
Transfer speed	1861
Traverse direction.....	609
Traverse ranges.....	623
TRC, Torque Ripple Compensation	1101
TS 220	119
TS 440	120
TS 444	120
TS 640	120
TS 740	120
TS touch probe	119
TT 130	121
TT tool touch probe	119
TTYP =.....	1707



U

UE 212B, specifications	257
UE 230B, specifications	254, 257
UEC 11x	86
UMC 11x	86
Upgrade functions	150
USB	1894
USB data carriers, connecting and removing	1897
USB hub	328, 1895
USB interface	328, 1894
User functions	136
User parameters	1264
USES statement	1847
Using the TS 440, TS 640	1536
Utilization	918
UV 102, specifications	240, 395

V

Variable download	1734
Variable names	1360
Velocity	812
Virtual tool axis	776
Virtual tool axis (VT)	772
Visual display units, BF 2xx	101
VNC settings	1879
Voltage protection module	880
Volts-per-hertz control mode	967
VROOT	382, 1376

W

WAIT FOR SYNC	1710, 1740
WATCH LIST	1671
Weakened field operation	880
Weakened-field operation, CC424 settings	1103
WHILE...ENDW	1844
Window manager (XFCE)	1332
Windows 2000/XP	1931
Working plane, tilting	761
WRITE TO KINEMATIC	733
WRITE TO PLC KEY	1742
Wye-delta connection switchover	980



X

X1 to X3.....	414
X1 to X6.....	297, 298, 299
X101.....	231
X121.....	288, 289
X127.....	327
X141, X142.....	208, 329, 1894
X141, X142, X143, X144.....	328
X15 to X20.....	300, 301
X169.....	233
X201 to X210.....	297, 298, 299
X26.....	325
X27.....	327
X31.....	415
X34.....	268
X35 to X38.....	297, 298, 299
X51 to X62.....	307
X69.....	233
X80 to X85.....	300, 301

HEIDENHAIN

DR. JOHANNES HEIDENHAIN GmbH

Dr.-Johannes-Heidenhain-Straße 5

83301 Traunreut, Germany

☎ +49 8669 31-0

FAX +49 8669 5061

E-mail: info@heidenhain.de

Technical support FAX +49 8669 32-1000

Measuring systems ☎ +49 8669 31-3104

E-mail: service.ms-support@heidenhain.de

TNC support ☎ +49 8669 31-3101

E-mail: service.nc-support@heidenhain.de

NC programming ☎ +49 8669 31-3103

E-mail: service.nc-pgm@heidenhain.de

PLC programming ☎ +49 8669 31-3102

E-mail: service.plc@heidenhain.de

Lathe controls ☎ +49 8669 31-3105

E-mail: service.lathe-support@heidenhain.de

www.heidenhain.de