Commissioning Guide Edition 2004-02

motion control

information system

SINUMERIK 840D/840Di/810D

Machine Data Management

MDA Machine

SIEMENS

SIEMENS

Standard Software for
Production Automation

MDA Machine

Commissioning Guide

Edition 2004-02

Trademark

SIMATIC® is a registered trademark of Siemens

OPC™ (OLE for Process Control) is a trademark of the OPC Foundation®

Table of Contents

1 General

1.1 Prerequisites

General

➢ MDA collects its data with the Microsoft SQL Server Desktop Engine (MSDE) 2000. If MSDE is not installed, the set up can be started during the MDA set up.

➢ The MDA installation consists installing DCOM95 (under Windows 95) and MDA itself (including database interfaces) and requires approximately 80 MB of hard disk space plus the space required for the database (Single machine: 20 MB, Multiple machines: max. 400 MB).

➢ Any previous version must be removed before installing the new version (see chapter 2.15 "Uninstalling").

➢ Any version of **WinBDE** must be removed before installing MDA. Uninstall WinBDE in the Control Panel using **Add/Remove Programs** or **Software**. Afterwards remove the installation directory of WinBDE.

➢ Administrator rights are required when installing on Windows NT and Windows 2000.

Demo

➢ Standard PC (Pentium III), SIMATIC Panel PC 670 with keyboard, SIMATIC PC FI45

➢ 128 Mbytes of RAM

➢ Windows 95, Windows NT 4.0 / SP 6, Windows 2000

Workstation MMC 103 (Possible use on request)

➢ SINUMERIK 810D / 840D / FM-NC

➢ MMC 103 and OP031 / OP032

➢ 64 Mbytes of RAM

➢ MMC software version 4.3, 4.4, 5.1, 5.2, 5.3 (under Windows 95)

Workstation PCU 50

➢ SINUMERIK 810D / 840D / FM-NC

➢ PCU 50 and OP010 / OP012 / OP015

➢ Min. 128 Mbytes of RAM

➢ HMI software version 6.0, 6.1, 6.2 (under Windows NT) , 6.3 (under Windows NT and Windows XP)

Workstation PC with ProTool/Pro

➢ Standard-PC (Pentium III), SIMATIC Panel PC 670 with keyboard, SIMATIC PC FI45

➢ Min. 128 Mbytes of RAM

➢ Profibus DP/MPI for Standard PC; alternative Ethernet

➢ Windows NT 4.0 / SP 6, Windows 2000

➢ ProTool/Pro V5.2 / SP 3, (V6.0 for Ethernet)

➢ SIMATIC S7-300, SIMATIC S7-400, C7-6xx via Profibus DP or MPI

<u>Workstation PC with WinCC</u>

➢ Standard PC (Pentium III)

➢ Min. 256 Mbytes of RAM

➢ Profibus DP/MPI for Standard-PC; alternative Ethernet

➢ Windows NT 4.0 / SP 6, Windows 2000 / SP 2

➢ WinCC V5.1

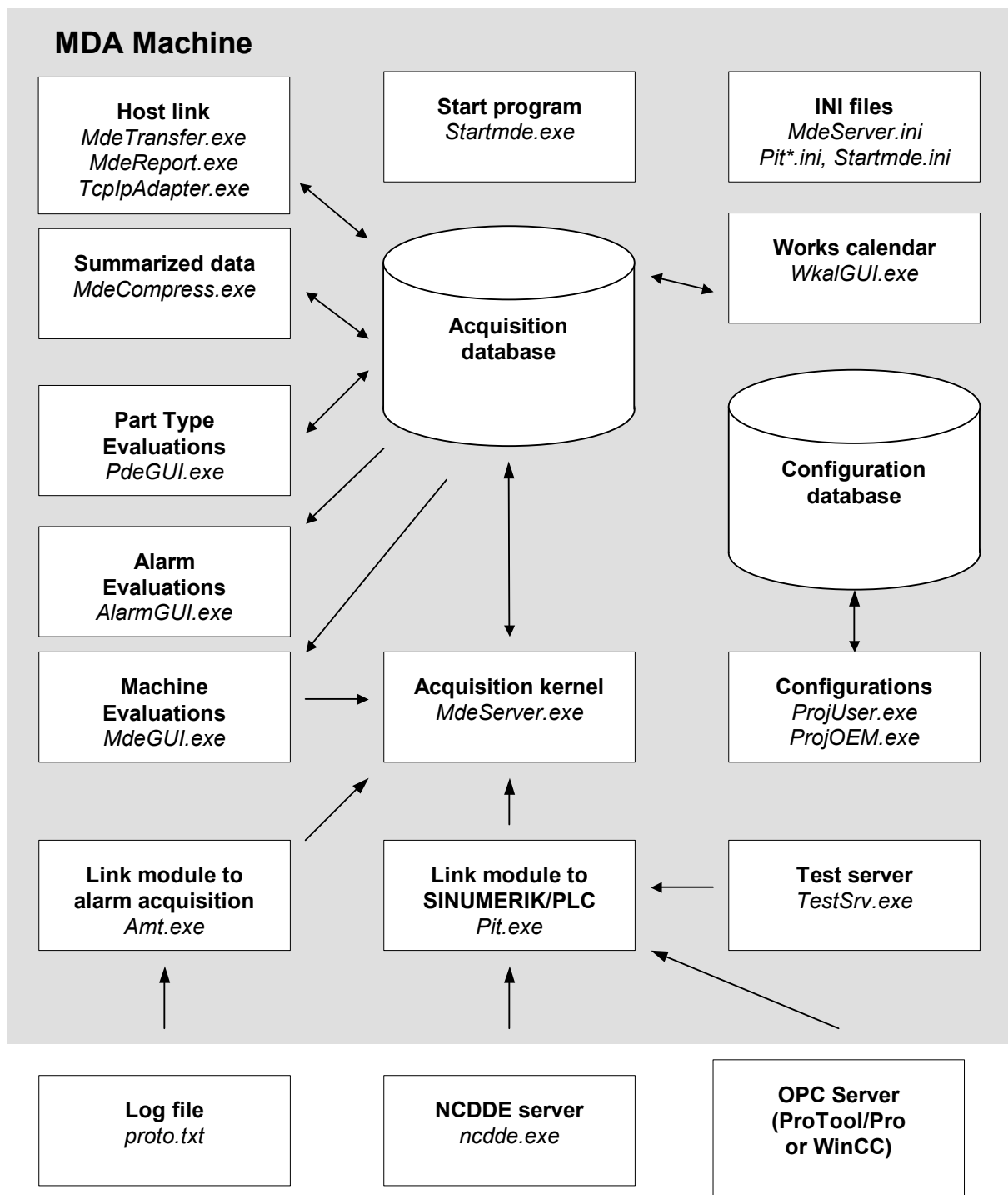➢ SIMATIC S7-300, SIMATIC S7-400, C7-6xx via Profibus DP or MPI

<u>Workstation PC with Simatic Computing</u>

➢ Standard-PC (Pentium III)

➢ Min. 256 MByte of RAM

➢ Profibus DP/MPI for Standard-PC; alternative Ethernet

➢ Windows NT 4.0 / SP 6, Windows 2000 / SP 2

➢ SIMATIC Computing, Version 3.0 / SP1

➢ SIMATIC S7-300, SIMATIC S7-400, C7-6xx via Profibus DP or MPI, WinLC of WinAC Basis, CPU 416-2 DP ISA of WinAC Pro

1.2   MDA Components

```
┌─────────────────────────────────────────────────────────────────────────────┐
│  MDA Machine                                                                  │
│                                                                               │
│  ┌────────────────────┐   ┌────────────────────┐   ┌────────────────────┐    │
│  │    Host link       │   │   Start program    │   │     INI files      │    │
│  │   MdeTransfer.exe  │   │    Startmde.exe    │   │   MdeServer.ini    │    │
│  │    MdeReport.exe   │   │                    │   │ Pit*.ini, Startmde.ini │ │
│  │   TcpIpAdapter.exe │   └────────────────────┘   └────────────────────┘    │
│  └────────────────────┘                                                      │
│  ┌────────────────────┐           ╭─────────╮      ┌────────────────────┐    │
│  │  Summarized data   │           │ Acquisition│    │   Works calendar   │    │
│  │  MdeCompress.exe   │           │  database  │    │     WkalGUI.exe    │    │
│  └────────────────────┘           ╰─────────╯      └────────────────────┘    │
│                                                          ╭─────────╮          │
│  ┌────────────────────┐                                 │Configuration│       │
│  │     Part Type      │                                 │  database   │       │
│  │    Evaluations     │                                 ╰─────────╯           │
│  │     PdeGUI.exe     │                                                       │
│  └────────────────────┘                                                       │
│  ┌────────────────────┐                            ┌────────────────────┐    │
│  │      Alarm         │                            │  Configurations    │    │
│  │   Evaluations      │                            │   ProjUser.exe     │    │
│  │   AlarmGUI.exe     │                            │   ProjOEM.exe      │    │
│  └────────────────────┘                            └────────────────────┘    │
│  ┌────────────────────┐   ┌────────────────────┐                              │
│  │     Machine        │   │ Acquisition kernel │                              │
│  │   Evaluations      │   │    MdeServer.exe   │                              │
│  │    MdeGUI.exe      │   └────────────────────┘                              │
│  └────────────────────┘                                                       │
│  ┌────────────────────┐   ┌────────────────────┐   ┌────────────────────┐    │
│  │  Link module to    │   │   Link module to   │   │    Test server     │    │
│  │  alarm acquisition │   │   SINUMERIK/PLC    │   │    TestSrv.exe     │    │
│  │      Amt.exe       │   │      Pit.exe       │   └────────────────────┘    │
│  └────────────────────┘   └────────────────────┘                              │
└─────────────────────────────────────────────────────────────────────────────┘

  ┌────────────────────┐   ┌────────────────────┐   ┌────────────────────┐
  │     Log file       │   │    NCDDE server    │   │    OPC Server      │
  │     proto.txt      │   │     ncdde.exe      │   │   (ProTool/Pro     │
  └────────────────────┘   └────────────────────┘   │    or WinCC)       │
                                                     └────────────────────┘
```

The **MDA** components have the following functionality:

The **Acquisition Database** contains all the dynamic data, such as the logbook, protocols, statistics and system messages. The database is backed up at regular intervals.
Configuration data is stored in the **Configuration Database**. Among other things, this includes global configuration parameters, screen form and message texts, the structure and hierarchy of the user

interface screen forms, colour definitions and user authorizations. The database is backed up at regular intervals.

All customizing can be carried out using the **Configuration User Interfaces** *ProjUser.exe* and *ProjOEM.exe*. The *ProjUser.exe* planning user interface can be used to generate work centers (machines and units), MDE statuses, user groups, colour definitions etc. and modify general settings. screen forms, table layouts and texts etc. can be modified using the *ProjOEM.exe* planning interface.

Shift and day types, working time models and the calendar can be defined using the **Works Calendar** *WkalGUI.exe*.

The **Start Program** *Startmde.exe* is used to automatically start up MDA and monitor the individual tasks. The start program uses the *Startmde.ini* to do this, in which all programs to be started are stored. The start program is stored on the MMC as a start service in the *[StartupConfiguration]* section of file *\add_on\regie.ini*.

The **Linking Module** *Pit.exe* handles the automatic acquisition of MDE data (statuses, counters) and BDE data (part types). The acquisition can take place on MMC (SINUMERIK) via PLC data blocks (PLC interface) or the status variables of the NCDDE server (Plug&Play interface, only MDE statuses). PC acquisition using ProTool/Pro or WinCC can take place using PLC data blocks (PLC interface) via the OPC server. For test purposes, acquisition can also take place manually using the *TestSrv.exe* test programs, whereby the PLC interface is simulated.

The **Linking Module** *Amt.exe* handles the automatic acquisition of alarms from the MMC file *proto.txt*.

The **Acquisition Kernel** *MdeServer.exe* accepts the acquired MDE data, BDE data and alarms and stores the messages in the form of data records in the logbook, protocol and priority report. The acquisition database is backed up and compressed by the acquisition kernel at planned time intervals.

The **Compression** program *MdeCompress.exe* generates the daily, weekly, monthly and annual statistics from the report with the acquired MDE data, BDE data and alarms. The compression program also deletes old data from the logbook and the protocols in order to limit the quantity of data in the database.

The **Host Coupling** *MdeTransfer.exe* handles the transfer of data between MDA Cell and the workstation. The planning and works calendar data can be transferred to the workstation from MDA Cell. The acquired data is transferred in the opposite direction at regular intervals. The logbook data are notified immediately to MDA Cell by *MdeReport.exe* and *TcpIpAdapter.exe*.

Fault reason analyses, status analyses, availability analyses and counter evaluations can be carried out in the **MDE User Interface** *MdeGUI.exe*. Data from the log book, the protocols and the system logbook can also be displayed and manual MDE statuses acquired.

OEE indicator calculations and counter analyses for part types can be carried out in the **BDE User Interface** *PdeGUI.exe*, and data from the logbook, protocols and statistics for part types displayed. The system logbook can also be displayed. The part type definitions can be displayed and edited. Part type filters can be defined.

The **Alarm User Interface** *AlarmGUI.exe* is used to display current alarms, alarm protocols and alarm statistics. Alarm filters can be defined.

The **Planning Files** *Pit_*ini* and *MdeServer.ini* contain basic **MDA** settings that are only modified for test purposes (e.g. database providers).

2 Installation

2.1   Overview

The installation of MDA on MMC 103 / PCU 50 of the SINUMERIK 810D / 840D / FM-NC must be carried out in the MMC service mode (see chapter 7.2 „Operating the MMC").

If several operator panels and NCUs are used (M:N concept of SINUMERIK) MDA Machine has to be installed at that panel which is the alarm and data management server. Alarms and machine data can only be collected there.

You must have **administrator** authorization to install MDA on a PC running Windows NT 4.0 (SP 5 or 6) or Windows 2000.

Installation is completed in several separate steps, which are handled by the **Setup_Machine.exe** installation routine:

> ➢ Installation of Microsoft Internet Explorer (only if installed version is older than version 5)

> ➢ Installation of Microsoft SQL Server Desktop Engine 2000 (if not installed yet)

> ➢ Installation of DCOM95 version 1.3 (Windows 95 only)

> ➢ Installing MDA with database interfaces (MDAC Version 2.8)

MDA handles its data with Microsoft SQL Server Desktop Engine (MSDE). Therefore a Microsoft Internet Explorer with version 5 or newer is needed. If the installed version is too old an actual version for German or English systems can be installed automatically. For other languages this installation has to done outside this set up. In any case after set up the security settings for the internet access should be checked and changed if necessary.

If the installation routine detects that DCOM95 V1.3 is already installed on the computer, it automatically skips this part of the installation and immediately goes over to installing MDA. If DCOM95 also needs to be installed, the installation routine prompts you to reboot the computer on completing installation.

During installation of MDA, amongst other things, you must enter a product key that the installation routine uses to detect the variant that is to be installed (e.g. MMC or PC variant, production or demo version).

There is little difference between installing on the MMC 103 and installing on the PCU 50. The main difference is that the MMC software for MMC 103 is installed on drive **c:** and for PCU 50 on drive **f:**. In other words, MDA is installed on **c:\add_on** for MMC 103 and on **f:\add_on** for PCU 50. The MMC directory is **c:\mmc2** for MMC 103 and **f:\mmc2** for PCU 50. The term MMC will be used in the following.

During installation of MDA on standard PCs, the MDA setup will be started automatically after reboot of the PC until the MDA setup is finished. During installation of MDA on MMC, the MDA setup has to be restarted manually after installing MSDE and following reboot of the system. This is necessary because the installation has to be done in service mode of the MMC. If the MMC restarts directly into the SINUMERIK HMI mode, an automatically restarted MDA setup would not be usable and could cause a failed installation.

Considering paths it is assumed in this documentation that MDA on standard PCs is installed into the directory C:\Siemens\MCIS or C:\Siemens\OSF respectively, on MMC103 into directory c:\add_on and on PCU50 into directory f:\add_on. In the sections below this MDA installation path is referred to as **<MDADIR>**.

User inputs of the dialogs and some internal function calls are written to the log file <MDADIR>\Install\aSetup.log. If there are any errors during installation this file may help you.

2.2   Backing up the Databases and Settings of Older Versions

If Version 6.x or 7.x of WinBDE or Version 1.0 of MDA Machine is already installed on the computer on which you want to install MDA Machine, and you want to import the data, you must proceed as described below **before** installing the new version.

2.2.1 WinBDE Upgrade

☞ In the new database, the system can only use the currently active configuration. This means that the **Configure OEM Data** (Start > Programs > SIMATIC IT WinBDE > Customizing (OEM)) or **Configure User Data** (Start > Programs > SIMATIC IT WinBDE > Customizing (User)) program must be used to define the variant that you want to make the active configuration in your old database.

☞ Save your old WinBDE databases **winbde.mdb** and **winbde_proj.mdb** so that the system can apply this data later in the new version.

☞ If you have changed the interfaces to the process and you wish to use them again, you have to save the files Pit*.ini und Amt.ini.

☞ After this, uninstall the older WinBDE version.

2.2.2 MDA Machine Upgrade

☞ Save your old MDA database **MDA.mdb** so that the system can apply this data later in the new version. For doing that start the script CopyMDADB.bat in directory <MDADIR>\tools.

☞ If you have changed the interfaces to the process and you wish to use them again, you have to save the files Pit*.ini und Amt.ini.

☞ After this, uninstall the older MDA Machine version.

2.3   Preparatory Step

Provision of installation files on a network drive

The Installation comprises various installation dialogues that are automatically started. For that purpose it is necessary to copy the contents of the installation CD onto the local hard disk or to provide it on a network drive.

Setting the system time

Before installing MDA, you **must** set the correct system time (date and clock time) on the computer. If you turn back the system time at a later date, MDA does not acquire any more data until the clock reaches the old time. If you reset the system time into the future at a later date, at the next start-up MDA generates data for the entire period of time by extending the last status conditions accordingly.

Softkey assignments (MMC only)

If you want to install MDA on an MMC, the following preparation is required:
Define keys for MDA (Machine, Part Type, Alarm Evaluations)
Determine which keys are still available for the MDA User Interfaces on the MMC's horizontal softkey bar. A different available key can also be used. No more than three keys are required.

Range applications on level 1

Key 0	Key 1	Key 2	Key 3	Key 4	Key 5	Key 6	Key 7
Machine	Parameter	Program	Services	Diagnosis	Start-up		

Range applications on level 2

Key 8	Key 9	Key 10	Key 11	Key 12	Key 13	Key 14	Key 15
MMC Help						Cycles	

Range applications on level 3

Key 16	Key 17	Key 18	Key 19	Key 20	Key 21	Key 22	Key 23
MDA Machines	MDA Orders	MDA Alarms					

Backing up the "regie.ini" file:
Create a backup of the \add_on\regie.ini file.
Determine the MMC version:
The **\mmc2\mmcvers.txt** file contains the exact version number of the MMC. Depending on the MMC version, you may have to reconfigure the key codes in MDA at a later date.

2.4   Starting Installation

At MMC you have to start the **MMC in service mode** (see chapter 7.1 „Operating the MMC"). This is **necessary for each start up during installation!**

At PCs with Windows 95, especially MMC103, you start the installation with:
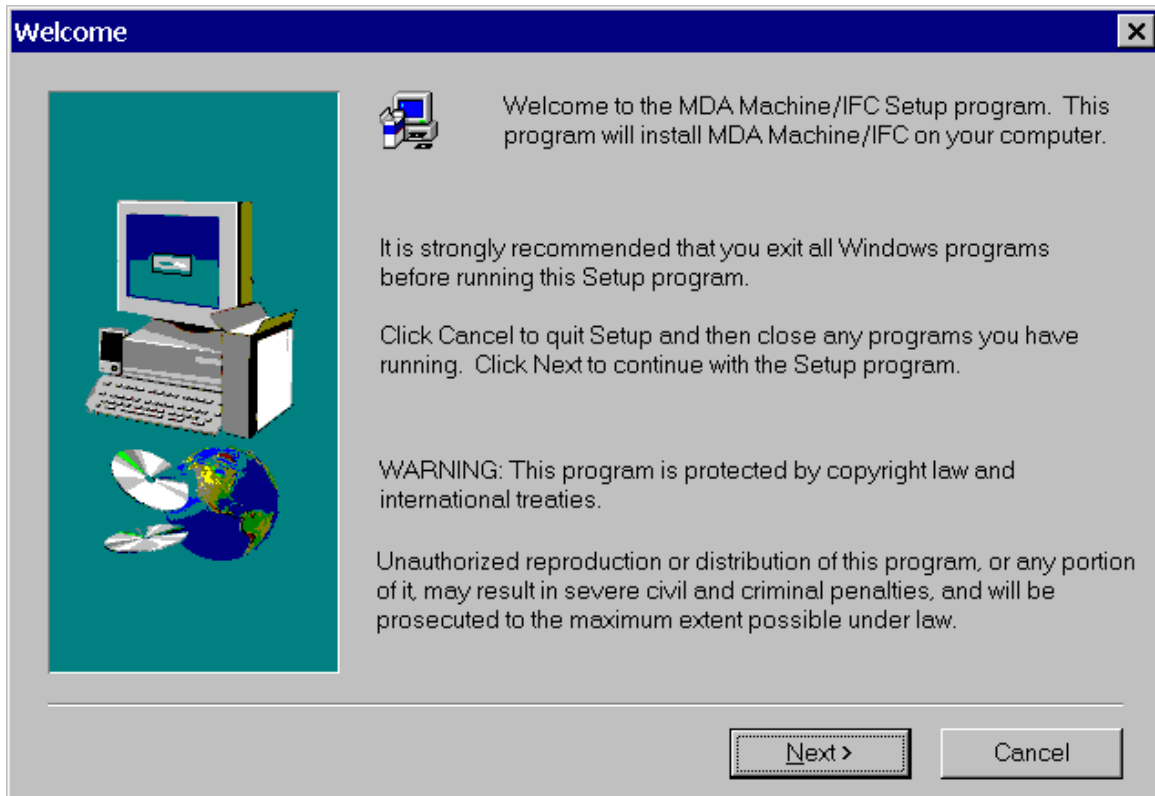
☞   Starting the script **mmc103.bat**

In all other cases:

☞   Start the **Setup_Machine.exe** program

from the MDA drive (local, CD-drive or network).

2.5   Installation Dialogs (Step 1)

During the first step all necessary dialogs are called:

☞   Continue installation by clicking the **Next** button.

☞   Accept the license conditions with clicking the **Next** button.

☞   Check that all other programs are finished and continue with clicking the **Next** button.

☞   Enter name and company of the registrated user and continue with clicking the **Next** button.

☞   Now enter the product key, the information whether a single machine or multiple machine configuration shall be installed, and the hardware platform in the following screen forms. This

information determines the MDA variant that is installed (Demo version product key: 01-01-01-059940).

☞   Now select the interface type to your process (see chapter 4 Interfaces).

☞   You can specify the installation directory in the next dialog window. For MMC you have to accept the default directory (*c:\add_on\MDA* or *f:\add_on\MDA*)! Enter the directory or accept the default directory and continue installation by clicking the **Next** button.

☞   In the next screen form, you can change the computer name of MDA Cell or MDA Plant. If an interface to these products is planned you have to enter the correct computer name. The name should have <u>maximal 8 characters</u>. If it is longer in the following mask you have to enter a maximal 8 characters long alias name for TCP/IP communication. This alias has to be administrated by network administration! Continue the installation by clicking the **Next** button.

☞   In the next screen form, you can choose the program folder in which you want to enter MDA. Accept the default settings and continue installation by clicking the **Next** button.

☞   All input data is stored. Then it is verified whether all necessary packets are already installed.

2.6   Installation of Microsoft Internet Explorer (Step 2)

(if necessary – not for demo version and MMC 103)
The MDA data administration is performed by the Microsoft SQL Server Desktop Engine (MSDE). This requires Microsoft Internet Explorer version 5 or newer. At first the installation program checks if a sufficiently new version already exists on your computer. In case of an older version being installed so far a current German or English version can be installed automatically. This must take place outside of the installation program for other languages. In any case the security settings with respect to the internet should be checked after the installation and modified if necessary.
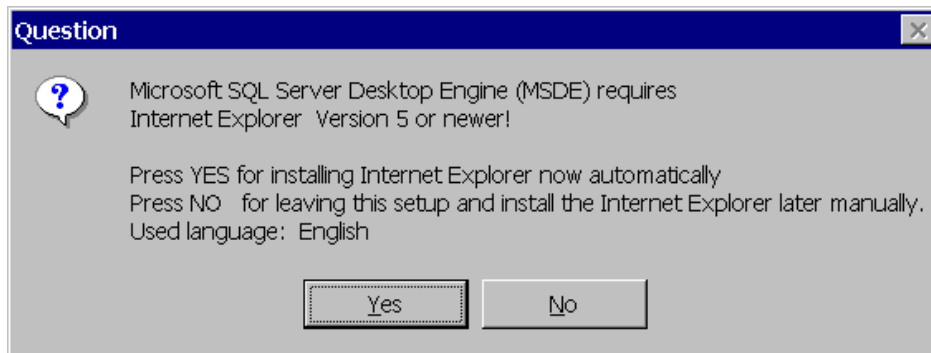
Notice: It is **not possible** to uninstall the internet explorer.
                     It is **retained** even if you **uninstall** MDA on the target system!

☞ If an appropriate Internet Explorer has already been installed you can immediately continue with chapter 2.7 "Installation of Microsoft SQL Server Desktop Engine (Step 3)".

☞ If an appropriate Internet Explorer has not yet been installed a respective hint occurs. Finish the installation by clicking the **No** button and install a newer version of the Internet Explorer. If you have set the language of the operating system to German or English and do not wish any special settings during the installation you can also let it install automatically: Continue the installation by clicking the **Yes** button.

☞ Now the installation program installs the Internet Explorer. Finish the installation after that.

☞ The **restart** of the PC is carried out automatically.

☞ After restart of the PC the MDA setup will be started automatically. If the setup does not come up automatically please start **Setup_Machine.exe** manually again.

Installation on the MMC:

☞ Start the MMC again in service mode.

☞ Start **Setup_Machine.exe** manually again

2.7   Installation of Microsoft SQL Server Desktop Engine (Step 3)

(only if necessary – not for demo version and MMC 103)
The MDA data administration is performed by the Microsoft SQL Server Desktop Engine (MSDE).
In this step the installation program checks if a sufficiently new version already exists on your computer. If not a current version can be installed automatically.

Notice: It is **not possible** to **automatically** uninstall MSDE. It is possible to uninstall MSDE within the control panel.
                     It is **retained** even if you **uninstall** MDA on the target system!

)   Continue the installation by clicking the **OK** button.

)   Now the installation program installs MSDE.
    In some cases the following dialog appears:
    „The following applications should be closed before continuing the install: MDA x.x".
    Continue with „Ignore".

)   Finish the installation of MDA after installation of MSDE.

)   The **restart** of the PC is carried out automatically.

)   After restart of the PC the MDA setup will be started automatically. If the setup does not come up automatically please start **Setup_Machine.exe** manually again.

Installation on the MMC:

)   Start the MMC again in service mode.

)   Start **Setup_Machine.exe** manually again

Notice: Installation at computer with **Windows XP**:
                     After executing the Setup of MSDE-Database the following message appears:

" Windows File Protection:
  Files that are required for Windows to run properly have been replaced by unrecognized versions. To maintain system stability, Windows must restore the original versions of these files. "
Insert your Windows XP Professional CD-ROM now.

           Retry   More Information   Cancel "

Please press "Cancel" and confirm the following message:

" You choose not to restore the original versions of the files.
This may affect Windows stability
Are you sure you want to keep these unrecognized file Versions ? "

with "YES".

After the necessary Reboot the MSDE-Database is working without problems.

2.8   Installing DCOM95 (Step 4)

(Windows 95 only)

In the first step the installation program checks whether DCOM95 V1.3 is already installed on your computer. If DCOM95 is not present, as is the case in the standard MMC installation (MMC P4/P5), DCOM95 is installed.
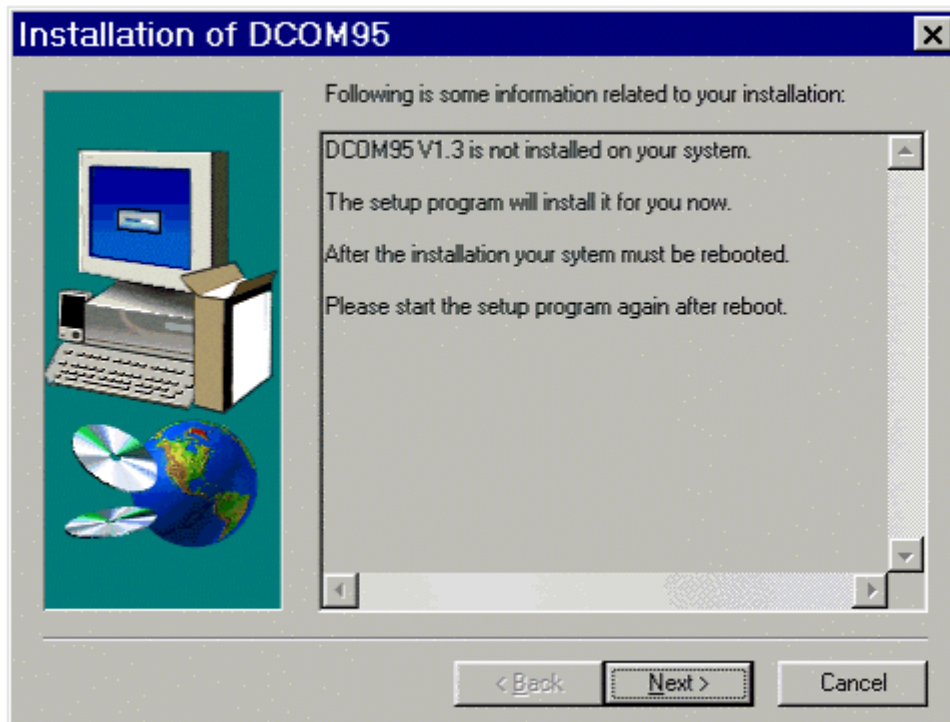
Notice: It is **not possible** to uninstall DCOM95.
                       It is **retained** even if you **uninstall** MDA on the target system!

Installation on the PC

- ☞ If version 1.3 of DCOM95 is already installed, then you can continue with chapter 2.9 "Installation of Microsoft Data Access Components (Step 5)".

- ☞ If version 1.3 of DCOM95 is not yet installed, the system displays the message box below. Continue installation by clicking the **Next** button.

- ☞ The install shield now installs DCOM95. Complete installation by clicking the **Finish** button.

- ☞ Restart the PC.

- ☞ After restart of the PC the MDA setup will be started automatically. If the setup does not come up automatically please start **Setup_Machine.exe** manually again.

Installation on the MMC

- ☞ After this, **shut down** Windows either by clicking the Windows Start button or by pressing Ctrl+Esc and **back up the MMC environment** (see chapter 7.1 "Operating the MMC"). Restart the MMC in service mode.

- ☞ Start Setup_Machine.exe manually again

2.9   Installation of Microsoft Data Access Components (Step 5)

(only if necessary)
If necessary the installation program installs the database interfaces of Microsoft Data Access Components (MDAC) V2.8 in this step.

☞ With the appearance of the message box "installing Microsoft Data Access Components (MDAC)..." the database interfaces (MDAC V2.8) are installed.

Notice: It is **not possible** to **automatically** uninstall MDAC.
It is **retained** even if you **uninstall** MDA on the target system!

2.10   Installing MDA (Step 6)

In this step the installation program installs MDA.

☞ Now all the MDA components are installed.

☞ In the case of the MMC: The following dialogue boxes ask which softkeys are to activate the MDA user interfaces and which text is to appear on the softkeys. Accept the default number of the softkey or enter a different number if for example *Key 16* is already occupied. Enter the number *0* if a user interface is not to be activated. Continue the installation by clicking the **Next** button.

☞ Now the message "Creating MDA Database ..." is shown and the MDA database is established. Afterwards the data of the delivery is copied – this process takes some time.

☞ Finish the installation in the next form by clicking the **Finish** button.

☞ Check date and time and adjust them if necessary.

Installation on the PC

☞ **Restart** the PC.

Installation on the MMC

☞ Disconnect the network drive which you used for the installation before starting the SINUMERIK in run-time mode.

☞ **Save the MMC environment** (cf. chapter 7.1 „Operating the MMC"). Start the MMC again in service mode.

2.11   Restoring Earlier Databases and Settings (Step 7, Optional)

2.11.1 WinBDE upgrade

If a WinBDE upgrade has been installed on the computer, the previous WinBDE databases (*winbde.mdb* and *winbde_proj.mdb*) and configuration files for process interfaces (Pit*.ini and Amt.ini) backed up before the installation can now be taken over.

Notice: Among other things, the work centre definitions (machines/units) are also taken over. If a MDA server (MDA Cell or Plant) is going to be used (see chapter 5„Interface to MDA Cell and MDA Plant"), it must be ensured when the machines/units are being taken over that the work centre numbers are unique over all machines/aggregates and workstations! On the MDA server itself, the same work centre numbers must be assigned for the machines/units as they were on the workstations! (see also chapter 3.2 "Customizing Machines/Units")

Procedure:

☞   Rename the previous WinBDE databases by prefixing the previous database names with **OLD_** (**OLD_winbde.mdb and OLD_winbde_proj.mdb**).

☞   Copy the databases to the MDA installation directory.

☞   Start the **MdaUpgradeDbs.exe** program.

☞   Check whether the correct work centre numbers (**Number**) have been assigned to the machines/units (**Symbolic Key**) in *Set* **Machine Number**. If this is not the case, the correct number must be entered for each machine/aggregate and confirmed with the **Set** button. If erroneous entries are made, previous corrections can be rolled back using the **Init all** button.

☞   Operate the **Execute** button.

The *MdaUpgradeDbs.exe* program copies all acquired data, statistics, status definitions, evaluation definitions, the works calendar and the work centers into the new database. Only the active configuration data is taken over. All screen form texts, error texts, user rights and the screen form layout are **not** taken over. General settings are partly updated so that you need to check them with the user interface "Customizing (User)" afterwards.

2.11.2 MDA Machine upgrade

If a MDA Machine upgrade has been installed on the computer, the previous MDA database (*MDA.mdb*) and configuration files for process interfaces (Pit*.ini and Amt.ini) backed up before the installation can now be taken over.

Notice: Among other things, the work centre definitions (machines/units) are also taken over. If a MDA server (MDA Cell or Plant) is going to be used (see chapter 5„Interface to MDA Cell and MDA Plant"), it must be ensured when the machines/units are being taken over that the work centre numbers are unique over all machines/aggregates and workstations! On the MDA server itself, the same work centre numbers must be assigned for the machines/units as they were on the workstations! (see also chapter 3.2 "Customizing Machines/Units")

You can either copy the complete database without changes or upgrade it in case of different MDA versions.

Database upgrade: (in case of different MDA versions necessary)

Procedure:

☞ Rename the previous MDA database by prefixing the previous database names with **OLD_** (**OLD_MDA.mdb**).

☞ Copy the databases to the MDA installation directory.

☞ Start the **MdaUpgradeDbs.exe** program*.*

☞ Check whether the correct work centre numbers (**Number**) have been assigned to the machines/units (**Symbolic Key**) in *Set* **Machine Number**. If this is not the case, the correct number must be entered for each machine/aggregate and confirmed with the **Set** button. If erroneous entries are made, previous corrections can be rolled back using the **Init all** button.

☞ Operate the **Execute** button.

The *MdaUpgradeDbs.exe* program copies all acquired data, statistics, status definitions, evaluation definitions, the works calendar and the work centers into the new database. Only the active configuration data is taken over. All screen form texts, error texts, user rights and the screen form layout are **not** taken over. General settings are partly updated so that you need to check them with the user interface "Customizing (User)" afterwards.

Copy complete Database without changes: (only in case of identical MDA versions)

Procedure:

☞ Rename the previous MDA database by prefixing the previous database names with **Old_** (**OLD_MDA.mdb**).

☞ Copy the database to the directory <MDADIR>\Tools.

☞ Start program <MDADIR>\Tools**CopyDatabase.exe**.

☞ Select as Source database :
Provider: MSDASQL
Connection: DRIVER={Microsoft Access Driver (*.mdb)};UID=;PWD=;DBQ=.\OLD_MDA.mdb;

and as Destination database:
Provider: SQLOLEDB
Connection:

driver={SQL Server};Server=(local);Database=PDA_DATA;User Id=PDA;Password=PDA;

☞ Operate the **OK** button.

☞ Operate the buttons **(De)Select All**, **Delete existing records at start copying** and finally **OK**.

The *CopyDatabase.exe* program copies all acquired data and all configuration data into the new database.

2.12   Customizing the Scan codes (Step 8, only on MMC P4)

In version 5.x of MMC the scan codes for the vertical softkeys have been changed compared to version 4.x. MDA will now be supplied with the scan codes for version 5.x as standard. These scan codes can also be used for MMC software version 6.x. In order to use MDA with MMC software version 4.x the scan codes must therefore be adapted using **Configure User Data** (see chapter 3 „Customizing") Parameters **KEYCODE_R1** to **KEYCODE_R8** must be adapted accordingly as shown in the following table to do this, using menu item **General Parameters > GLOB**:

Parameters	MMC Version 4.x	MMC Version 5.x/6.x
KEYCODE_R1	224	Shift+112
KEYCODE_R2	225	Shift+113
KEYCODE_R3	226	Shift+114
KEYCODE_R4	227	Shift+115
KEYCODE_R5	228	Shift+116
KEYCODE_R6	229	Shift+117
KEYCODE_R7	230	Shift+118
KEYCODE_R8	231	Shift+119

2.13   Synchronization (Step 9, Optional)

If MDA is installed on a **PC or MMC** with network installation, time synchronization with a Windows PC can take place (transfer of time from a Windows PC).

☞ The semicolon (comment character) in ";SyncWindows=1" in the **[TIME_SYNC]** section of file **Startmde.ini** in the MDA installation directory must be removed to do this.

☞ The **hostname** string in the "**SyncWindowsExe=...**" line must also be replaced with the name of the PC to be synchronized with. The "**SyncWindowsExe=...**" line is independent of the operating system – attention must be paid to the comment!

If MDA is installed on an **MMC**, time synchronisation to the PLC can take place (transfer of time from MMC).

☞ The semicolon (comment character) must be removed from "**;SyncSinumerik=1**" in the **[TIME_SYNC]** section of file **Startmde.ini** in the MDA installation directory to do this.

☞ The NCU name must also be entered in the "**SyncDDETopic=...**" line if the standard NCU name **ncu840d** is not being used (see chapter 7.2 „Installation on the MMC").

☞ For synchronisation of more than one PLC (multiple maschines, several NCUs) this can be configured with additional lines:
SyncDDETopic2=net:NCU_2
SyncDDETopic3=net:NCU_3
where NCU_2 and NCU_3 are the NCU names from netnames.ini

For the MMC variant, file **Startmde.ini** in directory **\mmc2 (MMC P4/P5)** or **\add_on (MMC P6)** must also be modified for time synchronization.

2.14   Starting MDA (Step 10)

☞ If MDA has been installed on an **MMC**, the MDA user interfaces can be started using one of the buttons specified in the Setup after the MMC has started up.

☞ If MDA has been installed on a **PC**, it can be started by clicking on the "Start" button in the taskbar and then using the "Programs > MDA > Start" menu.

If the Demo version of MDA has been installed, the demo data from 15th December 2001 to 5th February 2002 can be viewed and analyzed in MDA.
Immediately after installation a customizing facility is available. This uses the process interface selected during installation. Changes for that interface has to be configured explicitly (see chapter 4 "Interfaces").

2.15   Uninstalling

☞ Back up the old MDA database, if you wish to transfer the data to a new version at a later date. For that start the script CopyMDADB.bat in the tools-directory. The data are stored in the file MDA.mdb and can be copied into the new database with the program CopyDatabese.exe later.

☞ Uninstall MDA in the Control Panel using **Add/Remove Programs** or **Software**.

☞ Start the script Uninstall_MDA.bat.

☞ Delete the MDA installation directory.

☞ Reboot the computer.

3   Customizing

Customizing, i.e. making customer-specific adaptations to MDA to suit the on-site situation, is carried out using **Configure User Data** (Start > Programs > MDA > Customizing (User)) and **Configure OEM Data** (Start > Programs > MDA > Customizing (OEM)). Among other things, the following data can be generated using **Configure User Data**:

- ➢ General Parameters
- ➢ Work centers (Machines/Units)
- ➢ MDE Statuses
- ➢ Alarms
- ➢ OEE Code Definitions

Shift and day types, working time models and the calendar itself can be defined using **Works Calendar** (Start > Program->MDA > Works Calendar).
When **Configure User Data** is called up for the first time after installation, the product key that was entered during installation must be re-entered. This is not necessary for the Demo version.

Customizing is only possible on the MMC in Service mode. If **MDA Cell** or **MDA Plant** is being used (see chapter 6 "Interface to MDA Cell and MDA Plant") customizing can take place centralized.
This chapter only contains a description of the most important customizing items. More information on customizing can be found in the program documentation and the help.
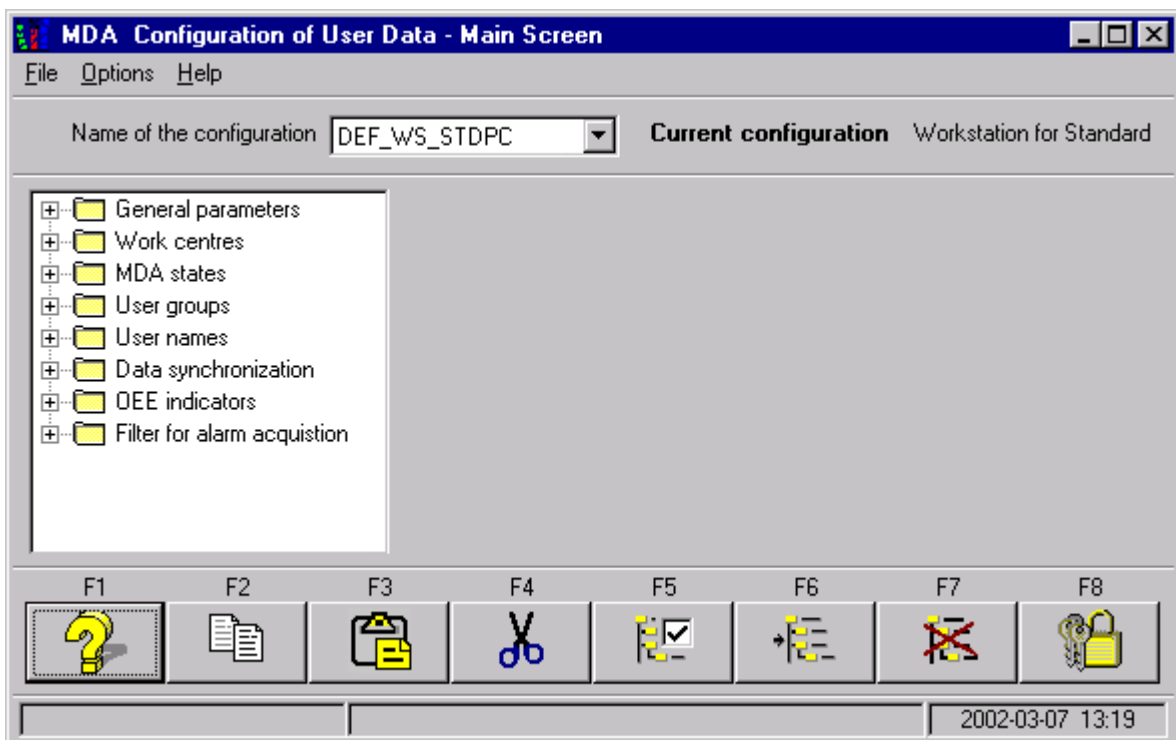
The user interface can be moved and the size changed using the mouse and the keyboard. The following entries in the window menu can be called up using the "ALT + Space bar" key combination:

Menu item	Function
Undo	Undoes previous action
Move	Allows the user interface to be moved using the cursor keys
Resize	Allows the size of the user interface to be adapted using the cursor keys
Minimize	Closes the application window without terminating the program
Maximize	Full screen display

3.1 General Parameters

Global and application-specific parameters can be modified using **General Parameters** (e.g. GUI_MDE, MDE_COMPRESS, MDE_SERVER, PIT, MDE_TRANSFER). A new configuration can be created from an existing configuration in the main screen form by copying and used as the current configuration for test purposes, for example.
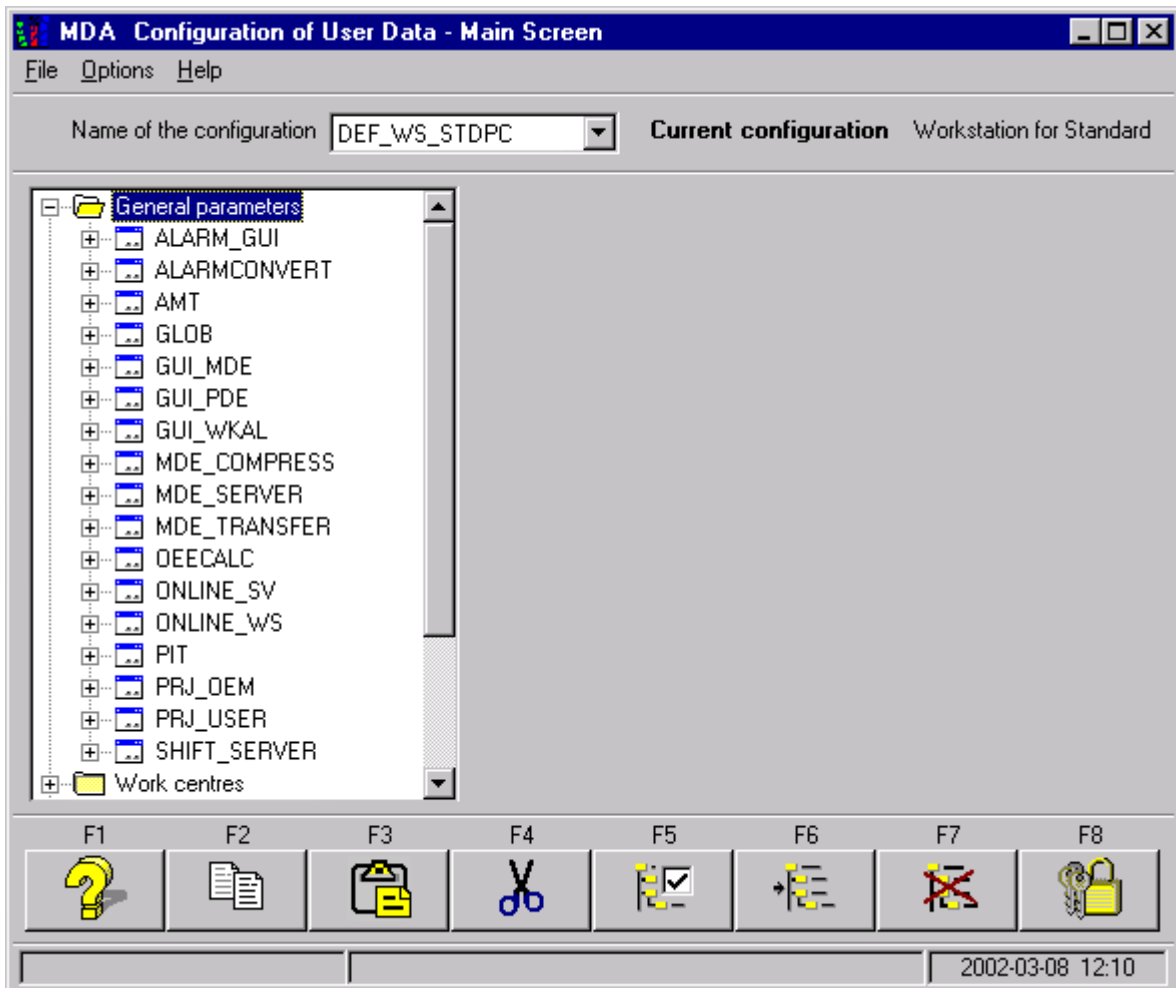
Among other things, the **General Parameters** can be used to configure which data is acquired and whether the data is acquired with/without the works calendar.

3.2   Customizing Machines/Units

The unique key for a machine / unit consists of the plant, BDE group and machine part keys. This three-part key is assigned to a unique work centre number within MDA. MDA uses the work centre number internally, and the three-part key is for display purposes only.

☞ The first entry is produced by selecting the **Work centers** folder in the tree display.

☞ Other entries can be produced by first displaying the data for an existing entry. Operating button **F3** ("Create New Machine") creates a new entry using the data from the previously selected entry.

☞ After an individual machine / unit has been selected in the tree display, the work centre data can be entered or modified using the tabs on the right-hand side of the screen.

"**Identification**" Tab

The work centre number (APLZ No.), the three-part key for display purposes and the designation are entered using the "**Identification**" tab. It must then be specified whether it is a machine ("**Main Unit**") or **Unit**. In the latter case the unit must be assigned to a machine. If there are more than one machines/units at a SINUMERIK, the NCU name has to be entered. If only one NCU exists the field may be empty. The NCU name will not be checked during acquisition then. The work centre number can only be entered when the work centre is created. The **Plant** and **BDE Group** key parts can also only be entered during creation at present.
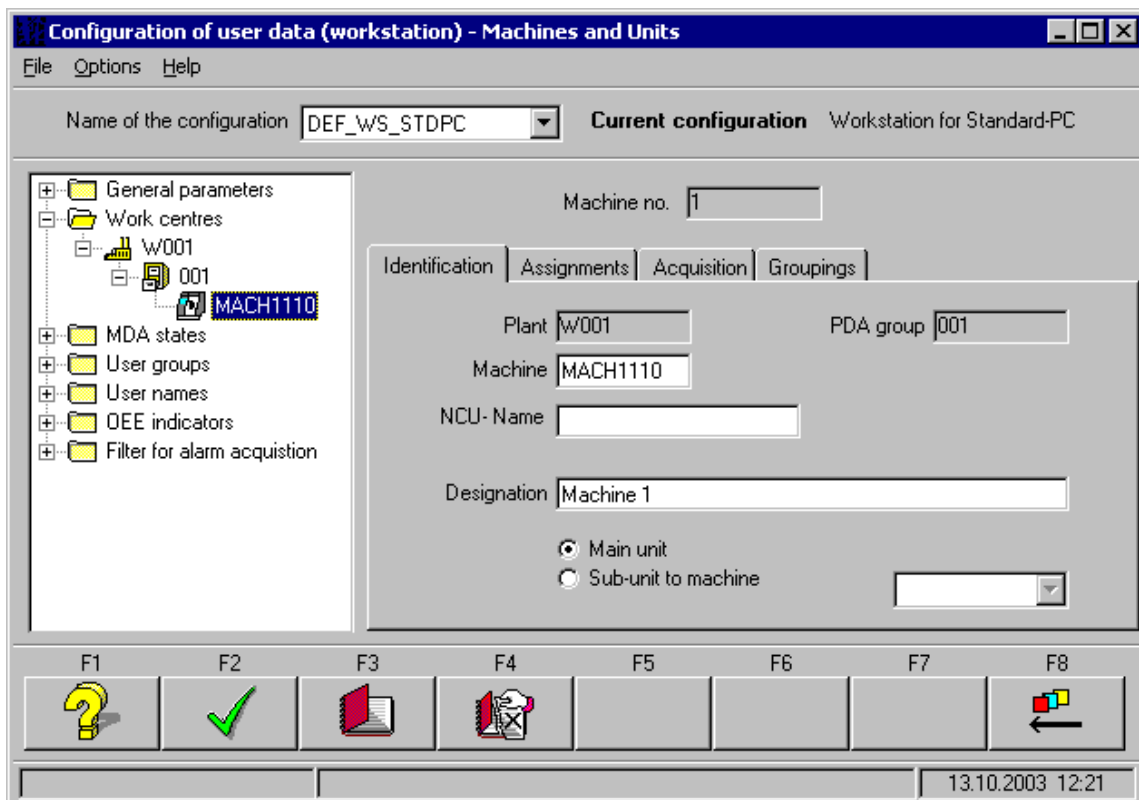
"**Assignments**" Tab

The statuses that a machine/unit can have are specified in the "**Assignments**" tab using the status group. Similar machines/units can belong to the same status group. The other information in the "**Assignments**" tab relates to the layout, alarm group and working time model. Whereas the layout parameters can always be entered (if no layout has been configured, this information is irrelevant), the alarm group (only with alarm acquisition) and the working time model (only with shifts/work calendar) can only be entered with the relevant configuration.
The EV Partner, PC Partner and Work Centre Matrix parameters are not currently used in the "**Assignments**" tab.
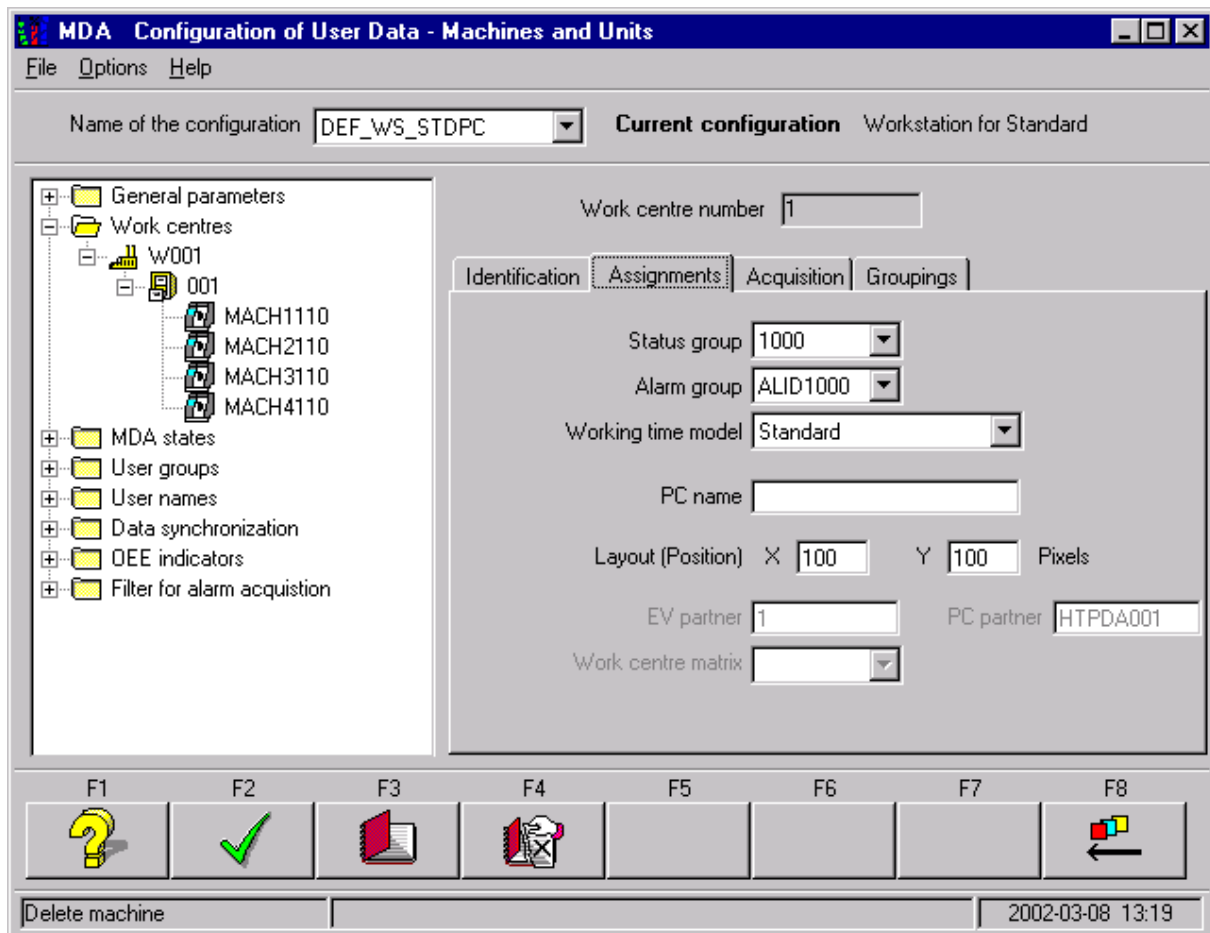
"**Acquisition** "Tab

The "**Acquisition**" tab is used to determine whether MDE and/or BDE data is to be acquired by selecting the type of automatic acquisition. For the PLC interface, the relevant data areas are addressed using the Cluster and Position parameters (see chapters entitled PLC Interface Machine (SINUMERIK), PLC Interface (ProTool/Pro) and PLC Interface (WinCC)). The machine/unit target cycle time (in seconds) can also be selected using the Cycle Time parameter.
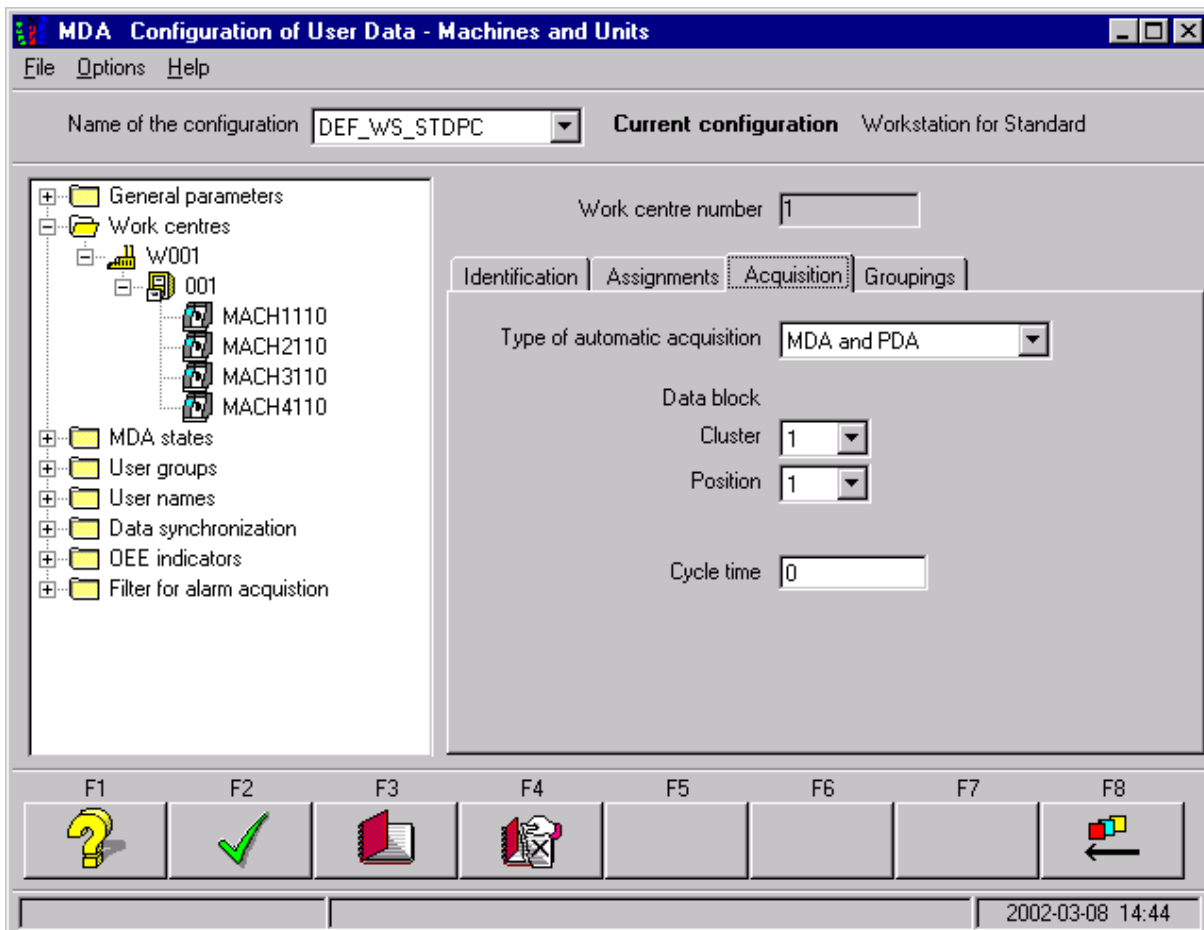
"**Groupings**" Tab

The parameters in the "Groupings" tab are not used at present.

3.2.1 Multiple Machines

On MMC multiple machines mean several NCUs. Details for the configuration you get in the SINUMERIK documentation
In "**Identification**" tab you have to configure the NCU name.

Attention: If you use several operator panels and NCUs (M:N concept of SINUMERIK) MDA Machine must be installed at that panel which is the alarm and data management server. Only there data blocks can be collected correctly ! At that panel a connection to all NCUs from which data should be collected must be configured.

3.3 Customizing MDA Status Conditions

Every machine and every unit is assigned to a status group. In a status group, you must define all the MDA status conditions that a machine or a unit can have. In principle, you can define a separate status group for each machine/unit. However, it is more practical to assign similar machines/units to the same status group reduce the number of configurations necessary.

☞ Status groups and MDE statuses are produced by selecting the **MDE states** folder in the tree display.
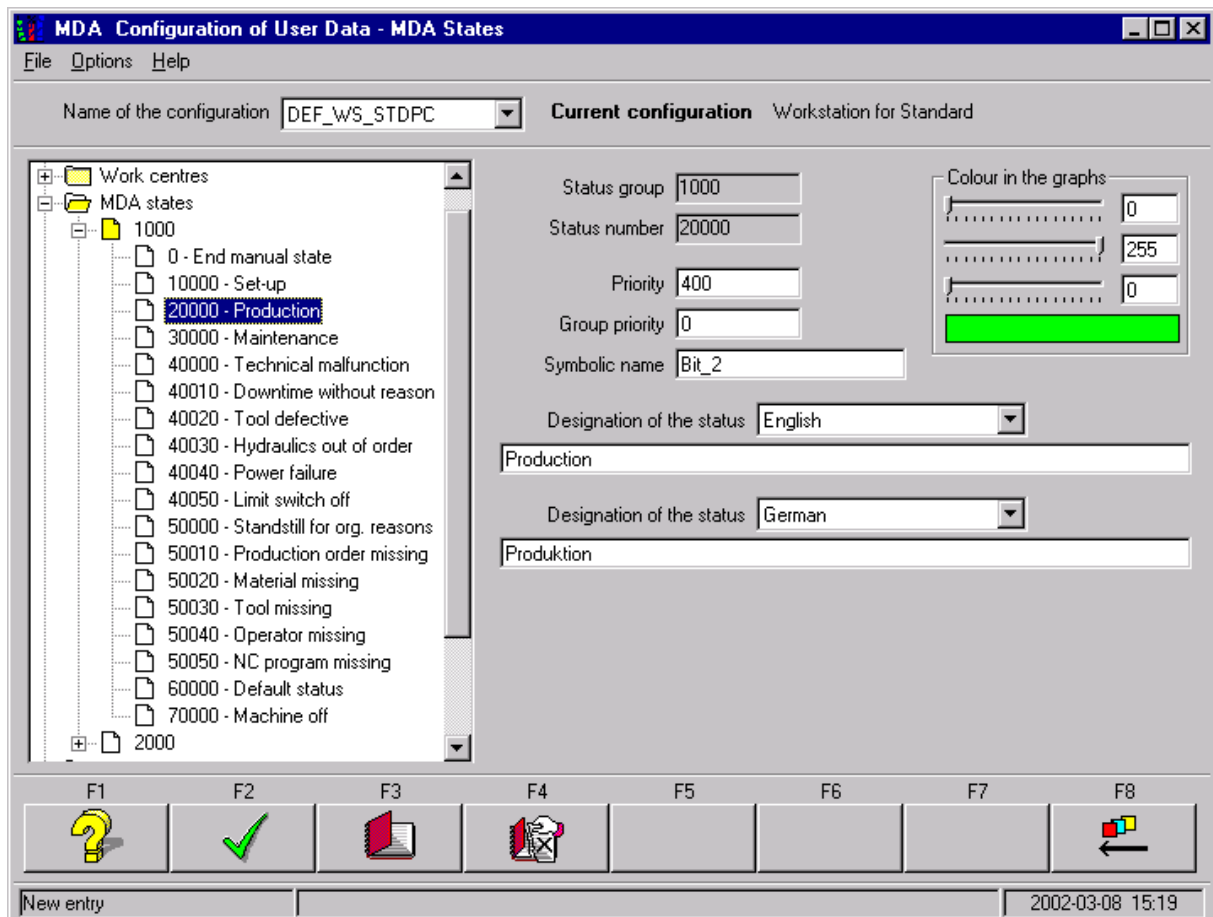
Every MDA status in a status group has a unique status number, a priority, a unique symbolic name, a designation and a colour for visualization.

In the **status number** a main and a sub-status condition are coded. The first digits of the number are for the main status condition and the last four represent the sub-status condition. You can define up to 80 main status conditions. Higher identification areas are spared for extensions. When defining a new status group you should take into account the fact that MDA's availability analysis only shows bars for the **main status conditions.** One of the bars in the availability chart contains the times of all the status conditions that belong to a main status condition (e.g. all the disturbances). Status analysis and disturbance cause analysis can be based on main or sub-status conditions.

In the **symbolic name** of an MDA status, the functionality is also implicitly coded. All the MDA status conditions that are to be manually acquired via the user interface must be prefixed by $MAN.

When using MDA's PLC interface (sample data in status group 1000), up to 128 status bits are possible per machine/unit. For each bit that is used, an MDA status must be defined with the prefix **Bit_** in the assigned status group. To be able to configure the MDA status for the second bit in the bit strip, for example, you must generate an entry containing the symbolic name Bit_2 in the status group.

If you use MDA's plug & play interface (**MMC**, sample data in status group 2000), you must define six MDA status conditions that all have symbolic names prefixed NCDDE_:

Symbolic Name	Mode
NCDDE_JOG	JOG mode
NCDDE_MDA	MDA mode
NCDDE_NCPLC	NC disturbance or PLC disturbance
NCDDE_ORG	Organizational disturbance
NCDDE_PLC	PLC disturbance
NCDDE_PROD	Production

Determining the highest priority MDA status

The **priority** feature allows you to regulate which MDA status has the highest priority in the event that there is more than one parallel MDA status conditions current at one time. Only the highest priority status at any one time is entered in the statistics. The lower the priority value of an MDA status, the higher its priority is. If two MDA status conditions with the same priority are current, the one that has been current the longest has the highest priority.
Internal status conditions $OFFLINE and $DEFAULT are used for status conditions machine switched off and machine status unknown. MDA assigns the lowest possible priority for these status conditions.

Manual MDA status conditions

Automatically acquired and manually entered MDA status conditions can overlap. In addition, with status conditions that are acquired manually via the MDA user interface, only one manual MDA status can be current at any one time. By implication, when choosing a new manual status, any manual status currently in the system is ended. You end the current manual MDA status without selecting a new manual MDA status by choosing $MAN_AUTO ("End manual status").

Notice: MDA status conditions $MAN_AUTO, $OFFLINE and $DEFAULT must be defined in every status group.

3.4   Customizing Alarms (MMC)

Alarm acquisition and evaluation is currently only allowed for the **MMC**.

Parameters must be set in the **General parameters** in order to read in alarm texts and acquire alarms (see chapter 3.7 "Overview of Important General Parameters")!

3.4.1 Read in Alarm Texts

In order to process alarms, the MMC alarm texts must first be read in. The alarm texts are required for correct generation and display of alarm statistics.

☞ The **Read in alarm texts** menu item only appears in the **File** menu in **Customizing(User)** if the **WITH_ALARM** parameter has been set to "1". The value of the **WITH_ALARM** parameter can be modified using the configuration (Customizing(User) > General Parameters > Glob > WITH_ALARM).

☞ The alarm texts are read in using **File > Read in alarm texts** menu item.

The standard MMC texts are usually stored in the **\dh\mb.dir** directory in files **al*??.com**, whereby "??" represents the language code (gr = German, uk = English etc., see chapter 7.2 "Installation on the MMC"). However, other alarm text files can exist. The names of these files and the directories in which they exist is stored in the [**TextFiles**] section of file **mbdde.ini** (priority is from left to right):

"\user\mbdde.ini, \oem\mbdde.ini, \add_on\mbdde.ini, \mmc2\mbdde.ini, \hmi_adv\mbdde.ini"
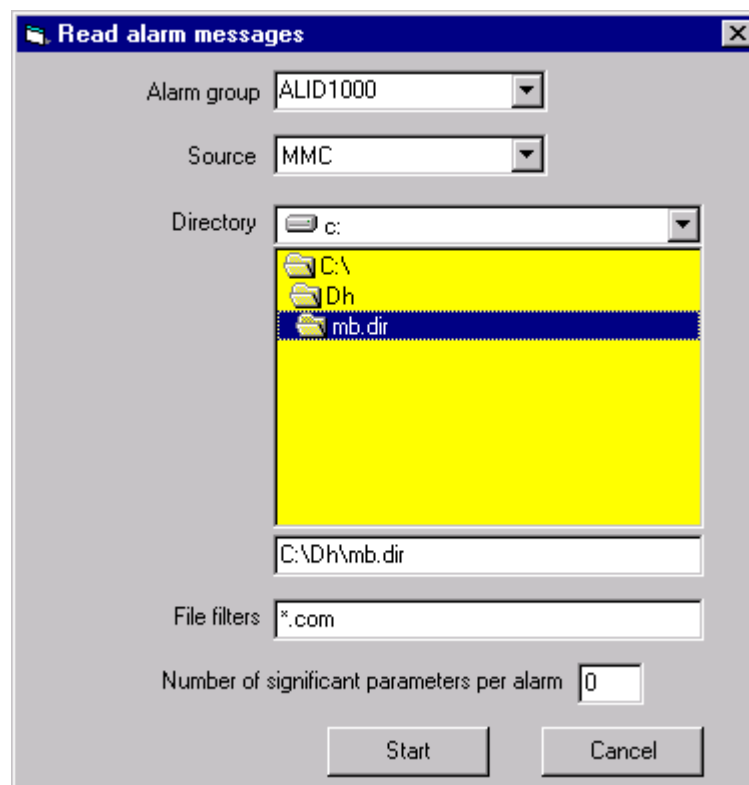
The following entries are required:

 ➢ **Alarm Group**: The alarm group that was specified when the work centers were defined must be specified.

 ➢ Source: Always enter MMC here.

 ➢ Directory: The directory containing the alarm text files must be entered here (default: "\dh\mb.dir").

 ➢ File Filters: The alarm text files with wildcards must be entered here, e.g. "al*.com"for the alarm texts for all languages, and "al*gr.com" or "al*uk.com" for all German and English alarm texts.

 ➢ Number of significant parameters per alarm: "0" should be entered here.
   Alarm texts can contain parameters, e.g. the channel number. The greater the number of relevant parameters, the greater the quantity of data in the alarm statistics.

3.4.2 DiskCare Parameter

The DiskCare Ini parameter in the **[protocol]** section must be set to "0" in file **mbdde.ini** (check the files corresponding to their priority from left to right):

"\user\mbdde.ini, \oem\mbdde.ini, \add_on\mbdde.ini, \mmc2\mbdde.ini, \hmi_adv\mbdde.ini"

3.4.3 Multiple Machines

On MMC multiple machines mean several NCUs. The different NCUs have to support their NCU name correctly while sending an alarm to the HMI-Base system (Details for the configuration you get in the SINUMERIK documentation).
With the user interface **Customizing(User)** you have to set the parameter **MMCALARM.NCU_NAME** to „1" in **General Settings** for program **AMT**.

Attention: If you use several operator panels and NCUs (M:N concept of SINUMERIK) MDA Machine must be installed at that panel which is the alarm and data management server. Only there alarms can be collected correctly! At that panel a connection to all NCUs from which alarms should be collected must be configured.

3.5   Customizing OEE Indicators

Calculation customizing and the display of availability and utilisation (OEE codes) is taken care of in the **OEE Indicators** folder in the **MDA Configure User Data** tree display.

☞   In order to modify the calculation and display of the OEE indicators, expand the **OEE indicators** folder in the tree display and select the **Calculate OEE indicators** or **Display OEE indicators** folder.
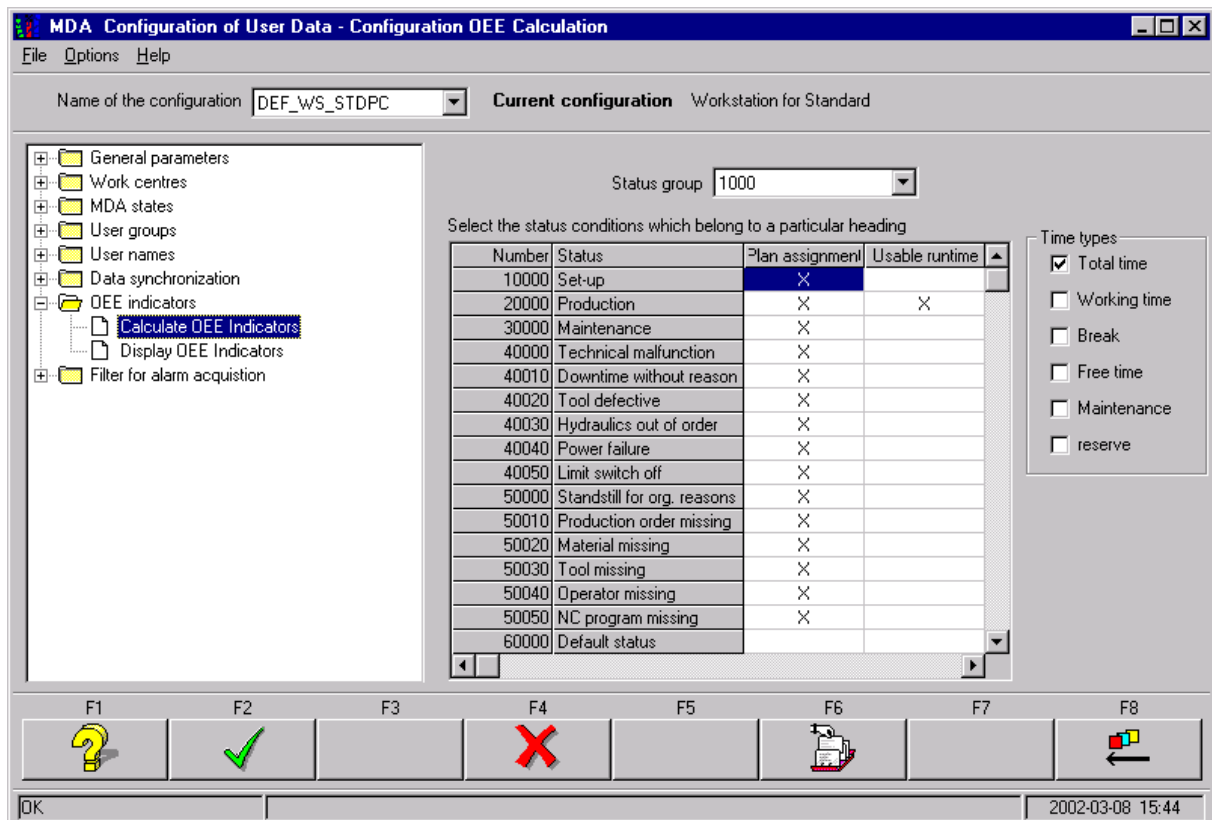
More information on customizing OEE Indicators can be found in the **MDA Configuration** user guide and help.

3.6   Customizing the Works Calendar

Shift and day types, working time models and the calendar can be defined using the **MDA Works Calendar** program (Start > Programs > **MDA** > Works Calendar). More information on customizing the works calendar can be found in the **MDA Works Calendar** user guide and help.

Notice: Parameters have to be set in **Customizing(User) > General parameters** in order to evaluate the works calendar during data acquisition (see chapter 3.7 "Overview of Important General Parameters")!

3.7   Overview of Important General Parameters

With/without acquisition of MDE data (statuses, counters)

Default installation values: **With** MDE data acquisition (without works calendar)

Parameters	Value with	Value without	Meaning
MDE_SERVER > ACQUISITION_ENABLED	1	0	General data acquisition (**always 0** on Supervisor)
GLOB > WITH_MDE_ACQUISITION	1	0	MDE processing
MDE_COMPRESS > WITH_STATE_ACQUISITION	1	0	MDE processing
MDE_COMPRESS > MAX_STATE_LOGBOOK	3000	0	3000 records, default value
MDE_COMPRESS > MAX_STATE_PRIOPROTO	1000	0	1000 records, default value
MDE_COMPRESS > MAX_STATE_PROTOCOL	1000	0	1000 records, default value
MDE_COMPRESS > MAX_STATE_SHIFTS	0	0	0 shifts, default value (**always 0** without works calendar/shifts)
MDE_COMPRESS > MAX_STATE_DAYS	60	0	60 days, default value
MDE_COMPRESS > MAX_STATE_WEEKS	10	0	10 weeks, default value
MDE_COMPRESS > MAX_STATE_MONTHS	12	0	12 months, default value
MDE_COMPRESS > MAX_STATE_YEARS	3	0	3 years, default value

<u>With/without BDE data acquisition (part types)</u>

Default installation values: **No** BDE data acquisition (without works calendar)

Parameter	Value with	Value without	Meaning
MDE_SERVER > ACQUISITION_ENABLED	1	0	General data acquisition (**always 0** on Supervisor)
GLOB > WITH_BDE_ACQUISITION	1	0	BDE processing
MDE_COMPRESS > WITH_ORDER_ACQUISITION	1	0	BDE processing
MDE_COMPRESS > MAX_ORDER_PROTOCOL	1000	0	1000 records, default value
MDE_COMPRESS > MAX_ORDER_SHIFTS	0	0	0 shifts, default value (**always 0** without works calendar/shifts)
MDE_COMPRESS > MAX_ORDER_DAYS	60	0	60 days, default value
MDE_COMPRESS > MAX_ ORDER_WEEKS	10	0	10 weeks, default value
MDE_COMPRESS > MAX_ ORDER _MONTHS	12	0	12 months, default value
MDE_COMPRESS > MAX_ORDER_YEARS	3	0	3 years, default value

<u>With/without alarm acquisition</u>

Default installation values: **No** alarm acquisition

Parameter	Value with	Value without	Meaning
MDE_SERVER > ACQUISITION_ENABLED	1	0	General data acquisition (**always 0** on Supervisor)
GLOB > WITH_ALARM	1	0	Alarm processing
AMT > WITH_MMCALARM	1	0	MMC alarm processing (**always 0** on Supervisor)
MDE_COMPRESS > WITH_ALARM_ACQUISITION	1	0	Alarm processing
MDE_COMPRESS > MAX_ALARM_PROTOCOL	1000	0	1000 records, default value
MDE_COMPRESS > MAX_ALARM_SHIFTS	0	0	0 shifts, default value (**always 0** without works calendar/shifts)
MDE_COMPRESS > MAX_ALARM_DAYS	10	0	10 days, default value
MDE_COMPRESS > MAX_ALARM_WEEKS	6	0	6 weeks, default value
MDE_COMPRESS > MAX_ALARM_MONTHS	6	0	6 months, default value
MDE_COMPRESS > MAX_ALARM_YEARS	2	0	2 years, default value

<u>With/without works calendar/shifts</u>

Default installation values: **No** works calendar/shifts

Parameter	Value with	Value without	Meaning
GLOB > WITH_SHIFT	1	0	Works calendar/shifts
MDE_COMPRESS > MAX_STATE_SHIFTS	90	0	90 shifts (e.g.)
MDE_COMPRESS > MAX_ORDER_SHIFTS	90	0	90 shifts (e.g.)
MDE_COMPRESS > MAX_ALARM_SHIFTS	10	0	10 shifts (e.g.)

4   Interfaces

There are 5 types of automatic MDE data acquisition (machine statuses/counters) and BDE data (part type counter/part type cycle times):

➢ PnP interface (SINUMERIK) (machine statuses only)

➢ PLC interface Machine (SINUMERIK)

➢ PLC interface Multiple Machines (SINUMERIK)

➢ PLC interface Units (SINUMERIK)

➢ OPC interface (ProTool/Pro)

➢ OPC interface (WinCC)

➢ OPC interface (SIMATIC Computing)

➢ Test interface

In the SINUMERIK **PnP Interface** (Plug&Play) variant, data that is available on the MMC as standard is used to determine the current machine status. In the PnP variant only the status of an NC channel can be acquired – there are no units.

In the other variants, defined data blocks that are in one or more data blocks are read from the PLC by the MDA coupling module and evaluated. The first data block (MDC Cluster) contains the MDE data (machine statuses, machine counters). The other data block (PDC Cluster) contains the BDE data (part type counters and cycle times). These data blocks must be provided with appropriate data by the PLC program.

In the SINUMERIK *PLC Interface Machine* variant, the machine statuses for a machine can be acquired via the MDC cluster. Up to 128 statuses and the machine counters can be acquired in an MDC cluster. The BDE data can be acquired via the PDC cluster.

In the SINUMERIK *PLC Interface Multiple Machines* variant, the machine statuses for up to 8 machines can be acquired via 8 MDC clusters. Up to 128 statuses and the machine counters can be acquired in an MDC cluster. The BDE data can be acquired via the PDC cluster. MDC clusters and PDC are in one data block each and both data blocks are in the same PLC. This PLC has to be on the standard NCU.

In the SINUMERIK *PLC Interface Units* variant, the machine statuses of up to 8 units can be acquired via the MDC cluster, whereby the machine itself is one of the 8 units. Up to 128 statuses per unit can be acquired. The machine counter for a machine can also be acquired in the MDC cluster. The BDE data can be acquired via the PDC cluster.

In the *OPC ProTool/Pro Interface* and *OPC WinCC Interface* variants, the machine statuses for a machine or unit can be acquired via the MDC cluster. Up to 128 statuses and the machine counters can be acquired in an MDC cluster. The BDE data can be acquired via the PDC cluster.
The test interface makes it possible to enter MDE and BDE data via a simulated PLC interface (see chapter 7.3 "Test Program TestSrv.exe").

4.1   Cluster Data Structures

The cluster data structures are described below. The <Unsigned Int32> and <Float> data types mentioned in the cluster description are referred to as <DWORD> and <REAL> in the S7 documentation. DW means data word (two bytes).

4.1.1 Structure of MDC Cluster

In previous descriptions this cluster was referred to *Structure DB13 compact*.
This cluster can be used for the following variants:

   ➢   PLC interface Machine (SINUMERIK)

   ➢   PLC interface Multiple Machines (SINUMERIK)

   ➢   OPC interface (ProTool/Pro)

   ➢   OPC interface (WinCC)

   ➢   Test interface

The AWL file for setting up the cluster on the PLC is on the MDA medium (\doc\MDC.awl). That applies to the multiple machines interface too (\doc\MDC8.awl).
The MDC cluster contains the MDE data for a machine/unit. Up to 8 MDC clusters for MDA can be defined for Multiple Machines, ProTool/Pro and WinCC, i.e. data for a maximum of 8 machines/units can be acquired.
At present, MDA evaluates the first three counters (good, waste and rework) in the MDC cluster and the 128-bit machine status array. Acquisition of the other part/cycle counters and the Analogue values is planned for future versions.

DB	Name	Description	Data Type	#Byte
00	Coordination value	Spare	Unsigned Int32	4
04	Unit status 1 = Machine status	Setup, processing, fault, maintenance, ...	128-bit array	16
20	Counter 1	Good (absolute)	Unsigned Int32	4
24	Counter 2	Waste (absolute)	Unsigned Int32	4
28	Counter 3	Rework (absolute)	Unsigned Int32	4
32	Counter 4	Spare	Unsigned Int32	4
36	Counter 5	Spare	Unsigned Int32	4
40	Counter 6	Spare	Unsigned Int32	4
44	Cycle counter 1	Spare	Unsigned Int32	4
48	Cycle counter 2	Spare	Unsigned Int32	4
52	Cycle counter 3	Spare	Unsigned Int32	4
56	Cycle counter 4	Spare	Unsigned Int32	4
60	Shift start	Spare	Unsigned Int32	4
64	Shift end	Spare	Unsigned Int32	4
68	Analogue value 1	Spare	Unsigned Int32	4
72	Analogue value 2	Spare	Unsigned Int32	4
76	Analogue value 3	Spare	Unsigned Int32	4
80	Analogue value 4	Spare	Unsigned Int32	4

Only the highlighted lines are currently evaluated by MDA. All the others are spare for future versions.

In MDA the individual bits in the 128-bit bit pattern for a unit are sequentially numbered. The first bit is given the symbolic name of Bit_1 and the last bit has the symbolic name of Bit_128. Addressing in the PLC program normally takes place using bytes and bits, whereby the first bit of a byte is bit 7 and the last byte is bit 0. This must be taken into consideration when customizing. The following table shows the assignment between the symbolic name in MDA and the PLC addresses for the bits:

Symbolic Name	PLC address Unit 1
Bit_1	DBX 4.7
Bit_2	DBX 4.6
Bit_3	DBX 4.5
Bit_4	DBX 4.4
Bit_5	DBX 4.3
Bit_6	DBX 4.2
Bit_7	DBX 4.1
Bit_8	DBX 4.0
Bit_9	DBX 5.7
Bit_10	DBX 5.6
Bit_11	DBX 5.5
Bit_12	DBX 5.4
Bit_13	DBX 5.3
Bit_14	DBX 5.2
Bit_15	DBX 5.1
...	...
Bit_127	DBX 19.1
Bit_128	DBX 19.0

4.1.2 MDC_MMC Cluster structure for MMC/SINUMERIK

This cluster was referred to as *Structure DB13* in previous descriptions.

This cluster can be used for the following variants:
- ➢ PLC interface Units (SINUMERIK)
- ➢ Test interface

This variant is used if the main machine consists of more than one unit and their statuses have to be evaluated.

The AWL file for setting up the cluster on the PLC is on the MDA medium (\doc\MDC_MMC.awl).
Only one MDC_MMC cluster can be defined for the MMC/SINUMERIK.
The 128-bit arrays for up to 20 machines/units are currently evaluated by MDA in the MDC_MMC-Cluster for the first three counters (good, waste, rework) for the main machine/main unit and the machine statuses. Acquisition of the other counters, the Analogue values, writing of the output bits and acquisition of up to 20 units is planned for projects.

DB	Name	Description	Data Type	#Byte
00	Coordination value	Spare	Unsigned Int32	4
04	Output	Spare	Unsigned Int32	4
08	Coordination value	Spare	Unsigned Int32	4
12	Analogue value 1	Spare	Unsigned Int32	4
16	Analogue value 2	Spare	Unsigned Int32	4
20	Analogue value 3	Spare	Unsigned Int32	4
24	Analogue value 4	Spare	Unsigned Int32	4
28	Analogue value 5	Spare	Unsigned Int32	4
32	Analogue value 6	Spare	Unsigned Int32	4
36	Analogue value 7	Spare	Unsigned Int32	4
40	Analogue value 8	Spare	Unsigned Int32	4
44	Analogue value 9	Spare	Unsigned Int32	4
48	Analogue value 10	Spare	Unsigned Int32	4
52	Analogue value 11	Spare	Unsigned Int32	4
56	Analogue value 12	Spare	Unsigned Int32	4
60	Analogue value 13	Spare	Unsigned Int32	4
64	Analogue value 14	Spare	Unsigned Int32	4
68	Analogue value 15	Spare	Unsigned Int32	4
72	ue value 16	Spare	Unsigned Int32	4
76	Coordination value	Spare	Unsigned Int32	4
80	Counter 1	Good (absolute)	Unsigned Int32	4
84	Counter 2	Waste (absolute)	Unsigned Int32	4
88	Counter 3	Rework (absolute)	Unsigned Int32	4

92	Counter 4	Spare	Unsigned Int32	4
96	Counter 5	Spare	Unsigned Int32	4
100	Counter 6	Spare	Unsigned Int32	4
104	Unit status 1 = Machine status	Setup, processing, fault, maintenance, ...	128-bit array	16
120	Unit status 2	Setup, processing, fault, maintenance, ...	128-bit array	16
136	Unit status 3	Setup, processing, fault, maintenance, ...	128-bit array	16
152	Unit status 4	Setup, processing, fault, maintenance, ...	128-bit array	16
168	Unit status 5	Setup, processing, fault, maintenance, ...	128-bit array	16
184	Unit status 6	Setup, processing, fault, maintenance, ...	128-bit array	16
200	Unit status 7	Setup, processing, fault, maintenance, ...	128-bit array	16
216	Unit status 8	Setup, processing, fault, maintenance, ...	128-bit array	16
232	Unit status 9	Setup, processing, fault, maintenance, ...	128-bit array	16
248	Unit status 10	Setup, processing, fault, maintenance, ...	128-bit array	16
264	Unit status 11	Setup, processing, fault, maintenance, ...	128-bit array	16
280	Unit status 12	Setup, processing, fault, maintenance, ...	128-bit array	16
296	Unit status 13	Setup, processing, fault, maintenance, ...	128-bit array	16
312	Unit status 14	Setup, processing, fault, maintenance, ...	128-bit array	16
328	Unit status 15	Setup, processing, fault, maintenance, ...	128-bit array	16
344	Unit status 16	Setup, processing, fault, maintenance, ...	128-bit array	16
360	Unit status 17	Setup, processing, fault, maintenance, ...	128-bit array	16
376	Unit status 18	Setup, processing, fault, maintenance, ...	128-bit array	16
392	Unit status 19	Setup, processing, fault, maintenance, ...	128-bit array	16
408	Unit status 20	Setup, processing, fault, maintenance, ...	128-bit array	16

Only the highlighted lines are currently evaluated by MDA. All the others are spare.

In MDA the individual bits in the 128-bit bit pattern for a unit are sequentially numbered. The first bit is given the symbolic name of Bit_1 and the last bit has the symbolic name of Bit_128. Addressing in the PLC program normally takes place using bytes and bits, whereby the first bit of a byte is bit 7 and the last byte is bit 0. This must be taken into consideration when customizing. The following table shows the assignment between the symbolic name in MDA and the PLC addresses for the bits:

Symbolic Name	PLC address Unit 1	PLC address Unit 2	PLC address Unit 3
Bit_1	DBX 104.7	DBX 120.7	DBX 136.7
Bit_2	DBX 104.6	DBX 120.6	DBX 136.6
Bit_3	DBX 104.5	DBX 120.5	DBX 136.5
Bit_4	DBX 104.4	DBX 120.4	DBX 136.4
Bit_5	DBX 104.3	DBX 120.3	DBX 136.3
Bit_6	DBX 104.2	DBX 120.2	DBX 136.2
Bit_7	DBX 104.1	DBX 120.1	DBX 136.1
Bit_8	DBX 104.0	DBX 120.0	DBX 136.0
Bit_9	DBX 105.7	DBX 121.7	DBX 137.7
Bit_10	DBX 105.6	DBX 121.6	DBX 137.6
Bit_11	DBX 105.5	DBX 121.5	DBX 137.5
Bit_12	DBX 105.4	DBX 121.4	DBX 137.4
Bit_13	DBX 105.3	DBX 121.3	DBX 137.3
Bit_14	DBX 105.2	DBX 121.2	DBX 137.2
Bit_15	DBX 105.1	DBX 121.1	DBX 137.1
...	...	...	...
Bit_127	DBX 119.1	DBX 135.1	DBX 151.1
Bit_128	DBX 119.0	DBX 135.0	DBX 151.0

4.1.3 Structure of PDC Cluster

This cluster can be used for the following variants:
- PLC interface Machine (SINUMERK)
- PLC interface Multiple Machines (SINUMERIK)
- PLC interface Units (SINUMERK)
- OPC interface (ProTool/Pro)
- OPC interface (WinCC)
- Test interface

The AWL file for setting up the cluster on the PLC is on the MDA medium (\doc\PDC.awl). That applies to the multiple machines interface too (\doc\PDC8.awl).

The PDC cluster contains the BDE data for a machine/unit. Up to 8 PDC clusters for MDA can be defined for Multiple Machines, ProTool/Pro and WinCC. This means that data for a maximum of 8 machines/units can be acquired.

At present, MDA evaluates the coordination word, the order ID, the counters (good, waste and rework) and the cycle counters (target and actual processing time) in the PDC cluster. The coordination word can only be set to 1 when all useful data has been updated.

DB	Name	Description	Data Type	#Byte
0	Coordination value	Access coordination on DB 1=OK, 0=DB updated	Unsigned Int32	4
4	Status	Spare	Unsigned Int32	4
8	Order ID	Order/Part Number/NC Prog.	CHAR	32
40	Spare 1	Spare	CHAR	32
72	Spare 2	Spare	CHAR	32
104	Counter 1	Good counter (Delta)	Unsigned Int32	4
108	Counter 2	Waste counter (Delta)	Unsigned Int32	4
112	Counter 3	Rework counter (Delta)	Unsigned Int32	4
116	Cycle counter 1	Target processing time [1]	Unsigned Int32	4
120	Cycle counter 2	Actual processing time [1]	Unsigned Int32	4
124	Shift data 1	Spare	Unsigned Int32	4
128	Shift data 2	Spare	Unsigned Int32	4

[1] Sum total of target/actual processing times for the registered parts (represents sum total of cycle times for registered parts). The target/actual processing time can be optionally taken from the order.

Only the highlighted lines are currently evaluated by MDA. All the others are spare for future versions.

Note: The PLC program is only allowed to write accumulated values into the PDC block (counters and clock times), because the data can only be acquired and processed by MDA with a cycle time of greater than 30 seconds. To avoid superfluous communication and processing, the PDC block must not be updated by the PLC program if the counters have not changed.

The OrderID field must be filled with blanks or binary 0 if less than 32 characters are used.

4.2   PnP Interface (SINUMERIK)

☞ The **General Parameters > PIT > PIT.PROFILE** parameter in **Configure User Data** (Start > Programs >MDA > Customizing (User)) must be set to the value "Pit_NCDDE_PNP.ini" to activate the interface.

The NCU name must be correctly configured for WinDBE in order to allow MDA to communicate with the MMC (Ncdde server). It must also be defined for which channel the data is to be acquired. By default, the coupling module is configured for NCU name **machineswitch** and channel **1**. If acquisition is to take place for another NCU name (see chapter 7.2 "Installation on the MMC") or another channel, the interface must be adapted.

The following parameters must be adapted using **Configure User Data** in folder **General Parameters -> PIT**:

☞ The **Machine1.Property1.Address** parameter must be set to "ncdde|**machineswitch**|/Bag/State/opMode[**u1**]".

☞ The **Machine1.Property2.Address** parameter must be set to "ncdde|**machineswitch**|/Channel/State/chanAlarm[**u1**]".

☞ The **Machine1.Property3.Address** parameter must be set to "ncdde|**machineswitch**|/Channel/State/progStatus[**u1**]".

☞ The **Machine1.Property4.Address** parameter must be set to "mbdde|alarme|NrOfAlarm".

In the above cases, **machineswitch** is the NCU name, **u1** is channel 1, **u2** is channel 2 etc.
The machine configured in the database (e.g. W001, MACH1110, 001) is to be assigned to status group 2000, for example. The required MDE statuses for this status group have already been configured. Assignment to the status group takes place when the machine/units are customized.
The following diagram the status data of the channel and operating type group (BAG) is accessed using the configured channel number:

Customizing
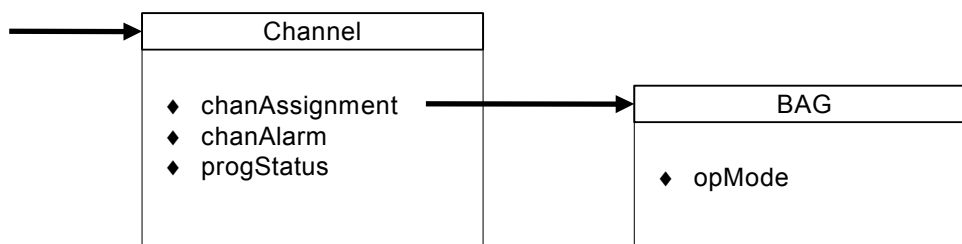
The following DDE variables in the NCDDE server and the alarm server are accessed on the MMC, whereby the default value of "ncu840d" is used as the DDE topic.

Assignment of channel to mode group (BAG):

➢ Application/Topic=/ncdde/machineswitch

➢ Item=/Nck/Configuration/chanAssignment[<Channel number>]

    0 = Channel not present
    n = Channel assigned to BAG n

Determine mode of a BAG:

> ➢ Application/Topic=/ncdde/machineswitch

> ➢ Item=/Bag/State/opMode[u< BAG-Nr.>]
>> 0 = JOG
>> 1 = MDA
>> 2 = AUTO

Determine program status:

> ➢ Application/Topic=/ncdde/machineswitch

> ➢ Item=/Channel/State/progStatus[u<Channel number>]
>> 1 = interrupted
>> 2 = stopped
>> 3 = running
>> 4 = waiting
>> 5 = aborted

Presence of NCK alarm:

> ➢ Application/Topic=/ncdde/machineswitch

> ➢ Item=/Channel/State/chanAlarm[u< Channel number >]
>> 0 = no alarm on this channel
>> 1 = Alarm without stop
>> 2 = Alarm with stop

Number of existing alarms:

> ➢ Application/Topic=/mbdde/alarme

> ➢ Item=/NrOfAlarm
>> 0=no alarms
>> 1=one alarm

The overall machine status can be determined from the following matrix:

Mode (opMode)	NCK Alarm (chanAlarm)	Progr.status (progStatus)	#Alarms (NrOfAlarm)	Overall status
0	*	*	*	JOG mode
1	*	*	*	MDA mode
2	0,1	3	*	Production
2	0,1	1,2,4,5	0	Organizational problem
2	0,1	1,2,4,5	> 0	Fault (PLC)
2	2	*	> 0	Fault (NC+ poss. PLC)

4.3   PLC Interface Machine (SINUMERIK)

☞ In order to activate the interface the **General Parameters > PIT > PIT.PROFILE** parameter has to be set to "Pit_NCDDE_PLC_MACH.ini" in **Configure User Data** (Start > Programs >MDA > Customizing (User)).

The automatic acquisition of MDE data via the PLC interface relies on having a prefabricated interface (MDC cluster) to the PLC. The interface has Plug&Play capability, but requires programming effort in the PLC to supply the MDC cluster with MDE data.
The automatic acquisition of BDE data via the PLC interface also relies on having a prefabricated interface (PDC cluster) to the PLC. The interface has Plug&Play capability, but requires programming effort in the PLC to supply the PDC cluster with BDE data.
The MDC and PDC clusters are in two different PLC data blocks. For special projects it is possible to put them into the same PLC data block (see description of PDC-Cluster). The structure of the MDC and PDC cluster is described in the chapter 4.1 "Cluster Data Structures". The counters (good, waste and rework) and the machine statuses are evaluated in the MDC cluster. The BDE data are evaluated in the PDC cluster.

MDC-Cluster

Customizing

The position of the MDC cluster in the data block has to be defined using **Configure User Data**. The following parameters have to be adapted using **Configure User Data** in the **General Parameters > PIT** folder:

☞ The **NCDDE_PLC.CLUSTER1.ITEM1** must be set to:
ncdde|machineswitch|/Plc/Datablock/Byte[**c13**,>**0**,#**84**]("!xl%02lx")

In this case **c13** stands for data block 13, and **0** is the start byte address of the cluster in the data block. The length of the MDC structure is **84** Bytes.
The NCU name must be correctly configured for MDA so that MDA can communicate with the MMC (Ncdde server). The coupling module is configured for NCU name **machineswitch** by default. The name has to be adapted in the following parameters for another NCU name (see chapter 7.2 "Installation on the MMC"):

☞ In parameter **NCDDE_PLC.CLUSTER1.ITEM1** the bold part of the value:
ncdde|**machineswitch**|/Plc/Datablock/Byte[c13,>0,#84]("!xl%02lx")

Notice: The assignment of a status field (128-bit array) to a machine or unit takes place when customizing the machines/units (see chapter 3.2 "Customizing Machines/Units ") using the Cluster and Position (always "1") parameters.

Examples:
➣ Cluster: 1, Position: 1 represents first machine

Notice: The **Type of automatic acquisition** parameter must also be used to define whether MDE or BDE data is being acquired when customizing the machines/units.

PDC Cluster

Customizing

The position of the PDC cluster in the data block has to be defined using **Configure User Data**. The following parameters have to be adapted using **Configure User Data** in the **General Parameters > PIT** folder:

> ➢ The **NCDDE_PLC.CLUSTER2.ITEM1** parameter must be set to:
>
>   ncdde|machineswitch|/Plc/Datablock/Byte[**c14**,>**0**,#132]("!xl%02lx")

> ➢ The **NCDDE_PLC.CLUSTER2.FLAG** parameter must be set to:
>
>   ncdde|machineswitch|/Plc/Datablock/Byte[**c14**,>**0**,#4]("!xl%02lx")

In this case **c14** stands for data block 14, and **0** is the start byte address of the cluster in the data block. The coordination flag (NCDDE_PLC.CLUSTER2.FLAG) is in the first four bytes of the PDC cluster. Both start byte addresses must therefore **always be identical**!

If the PDC cluster should be in the same data block 13 like the MDC cluster the configuration has to be like this:

> ➢ The **NCDDE_PLC.CLUSTER2.ITEM1** parameter must be set to:
>
>   ncdde|machineswitch|/Plc/Datablock/Byte[**c13**,>**84**,#132]("!xl%02lx")

> ➢ The **NCDDE_PLC.CLUSTER2.FLAG** parameter must be set to:
>
>   ncdde|machineswitch|/Plc/Datablock/Byte[**c13**,>**84**,#4]("!xl%02lx")

The NCU name must be correctly configured for MDA so that MDA can communicate with the MMC (Ncdde server). The coupling module is configured for NCU name **machineswitch** by default. The name has to be adapted in the following parameters for another NCU name (see chapter 7.2 "Installation on the MMC"):

> ☞ In parameter **NCDDE_PLC.CLUSTER2.ITEM1** the bold part of:
>   ncdde|**machineswitch**|/Plc/Datablock/Byte[c14,>0,#132]("!xl%02lx")

> ☞ In parameter **NCDDE_PLC.CLUSTER2.FLAG** the bold part of
>   ncdde|**machineswitch**|/Plc/Datablock/Byte[c14,>0,#4]("!xl%02lx")

Notice: The assignment to a machine or unit takes place when customizing the machines/units (see chapter 3.2 "Customizing Machines/Units") using the Cluster (always "1") and Position parameters (always "1").

The **Type of automatic acquisition** parameter must also be used to define whether MDE or BDE data is being acquired when customizing the machines/units.

4.4   PLC Interface Multiple Machines (SINUMERIK)

☞ In order to activate the interface the **General Parameters > PIT > PIT.PROFILE** parameter has to be set to "Pit_NCDDE_PLC_MACH8.ini" in **Configure User Data** (Start > Programs > MDA > Customizing (User)).

The automatic acquisition of MDE data via the PLC interface relies on having a prefabricated interface (MDC cluster) to the PLC. The interface has Plug&Play capability, but requires programming effort in the PLC to supply the MDC cluster with MDE data.

The automatic acquisition of BDE data via the PLC interface also relies on having a prefabricated interface (PDC cluster) to the PLC. The interface has Plug&Play capability, but requires programming effort in the PLC to supply the PDC cluster with BDE data.

All MDC are in one PLC data block and all PDC clusters in an other one. The structure of the MDC and PDC cluster is described in the chapter 4.1 "Cluster Data Structures". The counters (good, waste and rework) and the machine statuses are evaluated in the MDC cluster. The BDE data are evaluated in the PDC cluster.

By default data block DB13 for MDC cluster, datablock DB14 for PDC cluster and for communication with the MMC the NCU name **machineswitch** is used. Changes of this customizing have to be done in the configuration file Pit_NCDDE_PLC_MACH8.ini. In this case *c13* stands for data block 13.

MDC-Cluster

Customizing

The position of the MDC cluster in the data block has to be defined using **Configure User Data**. The following parameters have to be adapted using **Configure User Data** in the **General Parameters > PIT** folder:

☞ The **NCDDE_PLC.CLUSTER1.ITEM1** must be set to:
   ncdde|machineswitch|/Plc/Datablock/Byte[**c13**,>**0**,#**672**]("!xl%02lx")

☞ The **NCDDE_PLC.CLUSTER1.SIZE1** must be set to:
   672

In this case *c13* stands for data block 13, and *0* is the start byte address of the first cluster in the data block. The length of the MDC structure is 8 * 84 = **672** Bytes.
If there are less than 8 machines, the read length can be optimized. For that change the length of 672 to the necessary length n * 84 with n = number of machines.

The NCU name must be correctly configured for MDA so that MDA can communicate with the MMC (Ncdde server). The coupling module is configured for NCU name **machineswitch** by default. The name has to be adapted in the following parameters for another NCU name (see chapter 7.2 "Installation on the MMC"):

☞ In parameter **NCDDE_PLC.CLUSTER1.ITEM1** the bold part of the value:
   ncdde|**machineswitch**|/Plc/Datablock/Byte[c13,>0,#672]("!xl%02lx")

Notice: The assignment of a status field (128-bit array) to a machine or unit takes place when customizing the machines/units (see chapter 3.2 "Customizing Machines/Units ") using the Cluster and Position (always "1") parameters.

Examples:

➢ Cluster: 1, Position: 1 represents first machine

➢ Cluster: 2, Position: 1 represents second machine

Notice: The **Type of automatic acquisition** parameter must be used to define whether MDE or BDE data is being acquired when customizing the machines/units (**Configure User Data**).

PDC Cluster

Customizing

The position of the PDC cluster in the data block has to be defined using **Configure User Data**. The following parameters have to be adapted using **Configure User Data** in the **General Parameters > PIT** folder:

> ➢ The **NCDDE_PLC.CLUSTER2.ITEM1** parameter must be set to:
>
>   ncdde|machineswitch|/Plc/Datablock/Byte[**c14**,>**0**,#132]("!xl%02lx")

> ➢ The **NCDDE_PLC.CLUSTER2.FLAG** parameter must be set to:
>
>   ncdde|machineswitch|/Plc/Datablock/Byte[**c14**,>**0**,#4]("!xl%02lx")

In this case *c14* stands for data block 14, and *0* is the start byte address of the cluster in the data block. The coordination flag (NCDDE_PLC.CLUSTER2.FLAG) is in the first four bytes of the PDC cluster. Both start byte addresses must therefore **always be identical**!

The NCU name must be correctly configured for MDA so that MDA can communicate with the MMC (Ncdde server). The coupling module is configured for NCU name **machineswitch** by default. The name has to be adapted in the following parameters for another NCU name (see chapter 7.2 "Installation on the MMC"):

> ☞ In parameter **NCDDE_PLC.CLUSTER2.ITEM1** the bold part of:
>   ncdde|**machineswitch**|/Plc/Datablock/Byte[c14,>0,#132]("!xl%02lx")

> ☞ In parameter **NCDDE_PLC.CLUSTER2.FLAG** the bold part of
>   ncdde|**machineswitch**|/Plc/Datablock/Byte[c14,>0,#4]("!xl%02lx")

For further machines you have to adapt the following parameters:
        NCDDE_PLC.CLUSTER3.ITEM1 and NCDDE_PLC.CLUSTER3.FLAG for machine 2,
        NCDDE_PLC.CLUSTER4.ITEM1 and NCDDE_PLC.CLUSTER4.FLAG for machine 3,
        …
        NCDDE_PLC.CLUSTER8.ITEM1 and NCDDE_PLC.CLUSTER8.FLAG for machine 7,
        NCDDE_PLC.CLUSTER9.ITEM1 and NCDDE_PLC.CLUSTER9.FLAG for machine 8.

4.5   PLC Interface Units (SINUMERIK)

☞ In order to activate the interface the **General Parameters > PIT > PIT.PROFILE** parameter has to be set to "Pit_NCDDE_PLC_UNITS.ini" in **Configure User Data** (Start > Programs >MDA > Customizing (User)).

The automatic acquisition of MDE data via the PLC interface relies on having a prefabricated interface (MDC_MMC cluster) to the PLC. The interface has Plug&Play capability, but requires programming effort in the PLC to supply the MDC cluster with MDE data.
The automatic acquisition of BDE data via the PLC interface also relies on having a prefabricated interface (PDC cluster) to the PLC. The interface has Plug&Play capability, but requires programming effort in the PLC to supply the PDC cluster with BDE data.
The MDC_MMC and PDC clusters are in two different PLC data blocks. The structure of the MDC and PDC cluster is described in the chapter 3.2 "Cluster Data Structures". The counters (good, waste and rework) and the machine statuses are evaluated in the MDC_MMC cluster. The BDE data are evaluated in the PDC cluster.

MDC_MMC Cluster

Customizing

The position of the MDC_MMC cluster in the data block has to be defined using **Configure User Data**. The following parameters have to be adapted using **Configure User Data** in the **General Parameters > PIT** folder:

☞ The **NCDDE_PLC.CLUSTER1.ITEM1** must be set to:
ncdde|machineswitch|/Plc/Datablock/Byte[**c13**,>**0**,**#424**]("!xl%02lx")

In this case *c13* stands for data block 13, and *0* is the start byte address of the cluster in the data block. The length of the MDC_MMC structure is **424** Bytes.
The NCU name must be correctly configured for MDA so that MDA can communicate with the MMC (Ncdde server). The coupling module is configured for NCU name **machineswitch** by default. The name has to be adapted in the following parameters for another NCU name (see chapter 7.2 "Installation on the MMC"):

☞ In parameter **NCDDE_PLC.CLUSTER1.ITEM1** the bold part of the value:
ncdde|**machineswitch**|/Plc/Datablock/Byte[c13,>0,#424]("!xl%02lx")

Notice: The assignment of a status field (128-bit array) to a machine or unit takes place when customizing the machines/units (see chapter 3.2 "Customizing Machines/Units") using the Cluster (always "1") and Position parameters.

Examples:

➤ Cluster: 1, Position: 1 represents first 128-bit array

➤ Cluster: 1, Position: 3 represents third 128 bit array

Notice: The **Type of automatic acquisition** parameter must also be used to define whether MDE or BDE data is being acquired when customizing the machines/units.

PDC Cluster

Customizing

The position of the PDC cluster in the data block has to be defined using **Configure User Data**. The following parameters have to be adapted using **Configure User Data** in the **General Parameters > PIT** folder:

> ➢ The **NCDDE_PLC.CLUSTER2.ITEM1** parameter must be set to:
>> ncdde|machineswitch|/Plc/Datablock/Byte[**c14**,>**0**,#132]("!xl%02lx")

> ➢ The **NCDDE_PLC.CLUSTER2.FLAG** parameter must be set to:
>> ncdde|machineswitch|/Plc/Datablock/Byte[**c14**,>**0**,#4]("!xl%02lx")

In this case *c14* stands for data block 14, and *0* is the start byte address of the cluster in the data block. The coordination flag (NCDDE_PLC.CLUSTER2.FLAG) is in the first four bytes of the PDC cluster. Both start byte addresses must therefore **always be identical**!
The NCU name must be correctly configured for MDA so that MDA can communicate with the MMC (Ncdde server). The coupling module is configured for NCU name **machineswitch** by default. The name has to be adapted in the following parameters for another NCU name (see chapter 7.2 "Installation on the MMC"):

> ☞ In parameter **NCDDE_PLC.CLUSTER2.ITEM1** the bold part of:
>> ncdde|**machineswitch**|/Plc/Datablock/Byte[c14,>0,#132]("!xl%02lx")

> ☞ In parameter **NCDDE_PLC.CLUSTER2.FLAG** the bold part of
>> ncdde|**machineswitch**|/Plc/Datablock/Byte[c14,>0,#4]("!xl%02lx")

Notice: The assignment to a machine or unit takes place when customizing the machines/units (see chapter 3.2 "Customizing Machines/Units") using the Cluster (always "1") and Position parameters (always "1").

                     The **Type of automatic acquisition** parameter must also be used to define whether MDE or BDE data is being acquired when customizing the machines/units.

4.6   PLC Interface (ProTool/Pro)

☞ In order to activate the interface the **General Parameters > PIT > PIT.PROFILE** parameter has to be set to "Pit_ProTool.ini " in **Configure User Data** (Start > Programs > MDA > Customizing (User)).

The automatic acquisition of MDE data via the PLC interface relies on having a prefabricated interface (MDC cluster) to the PLC. The interface has Plug&Play capability, but requires programming effort in the PLC to supply the MDC cluster with MDE data.

The automatic acquisition of BDE data via the PLC interface also relies on having a prefabricated interface (PDC cluster) to the PLC. The interface has Plug&Play capability, but requires programming effort in the PLC to supply the PDC cluster with BDE data.

The MDC and PDC clusters are always in PLC data blocks. The MCD and PCD cluster contains the MDE and BDE data for one machine/unit. Data for a maximum of 8 machines/units can be acquired using the clusters. The structure of the MMC and PDC clusters is described in the chapter 4.1 "Cluster Data Structures".

MDC Cluster

Customizing

An MDC cluster corresponds to a variable in ProTool/Pro and contains a machine/unit. MDC clusters can be distributed over several data blocks. The data blocks may be on different controllers. The length of a data block is therefore variable (n * 84 bytes).

A variable of type **Byte** must be created in ProTool/Pro CS for each MDC cluster:

➢ Name:
>   Value = "MDA_M1, MDA_M2, ..."
➢ Acquisition cycle (s):
>   Value = "2"
➢ Number of elements:
>   Value = "84"
➢ Range:
>   Value = "DB"
➢ DB:
>   Value = "13" (possibly 14, 15, ...)
➢ DBB:
>   Value = "(n-1) * 84"

Notice: "OPC Server" must also be selected in ProTool/Pro in menu item **Target System > Settings**.

☞ The cluster is assigned to a machine or unit when customizing the machines/aggregates (see chapter 3.2 "Customizing Machines/Units") using the **Cluster** and **Position** parameters (Work centers > Acquisition), whereby the **Position** parameter must always be set to "1".

Examples:

• Cluster 1, Position 1 represents ProTool variables MDA_M1

• Cluster 3, Position 1 represents ProTool variables MDA_M3

☞ The **Type of automatic acquisition** parameter must also be used when customizing the machines/unit in order to define whether MDE and/or BDE data is to be acquired.

PDC Cluster

Customizing

A PDC cluster corresponds to a variable in ProTool/Pro and contains a machine/unit. PDC clusters can be distributed over several data blocks. The data blocks may be on different controllers. The length of a data block is therefore variable (n * 132 Byte).
Two variables of type **Byte** must be created in ProTool/Pro CS for each PDC cluster.

Variable 1 (PDC cluster):
 ➢ Name:
   Value = "MDA_P1, MDA_P2, ..."
 ➢ Acquisition cycle (s):
   Value = "5"
 ➢ Number of elements:
   Value = "132"
 ➢ Range:
   Value = "DB"
 ➢ DB:
   Value = "14" (possibly 13, 15, ...)
 ➢ DBB:
   Value = „"n-1) * 132"

Variable 2 (coordination flag in PDC cluster):
 ➢ Name:
   Value = "MDA_PF1, MDA_PF2, ..."
 ➢ Acquisition cycle (s):
   Value = "2"
 ➢ Number of elements:
   Value = "4"
 ➢ Range:
   Value = "DB"
 ➢ DB:
   Value = "14" (possibly 13, 15, ...)
 ➢ DBB:
   Value = (n-1) * 132

Notice: Menu item "OPC Server" must also be selected in **Target System > Settings** in ProTool/Pro.

The coordination flag is contained in the first four bytes of the PDC cluster. The data block number (DB) and the start byte address (DBB) must therefore **always be identical** in variable pairs MDA_P1, MDA_P2, ... and MDA_PF1, MDA_PF2, ...!

If MDE data is also being acquired for a machine/unit, ProTool variables MDA_M1, MDA_M2, ... in the MDC cluster must correspond with the ProTool variables for the PDC cluster MDA_P1, MDA_P2, ... and MDA_PF1, MDA_PF2, ...

☞ The cluster is assigned to a machine or unit when customizing the machines/aggregates (see chapter 3.2 "Customizing Machines/Units") using the **Cluster** and **Position** parameters (Work centers > Acquisition), whereby the **Position** parameter must always be set to "1".

Examples:

- Cluster 1, Position 1 represents ProTool variables MDA_P1 and MDA_PF1

- Cluster 3, Position 1 represents ProTool variables MDA_P3 and MDA_PF3

☞ The **Type of automatic acquisition** parameter must also be used when customizing the machines/unit in order to define whether MDE and/or BDE data is to be acquired.

MDC_MMC Cluster

For compatibility reasons the MDC_MMC cluster is still supported for ProTool/Pro in MDA. In the MDC_MMC cluster the counters (good, waste, rework) for the main machine/main unit and the machine statuses for up to 8 machines/units are evaluated. Only one MDC_MMC cluster can be defined.

The **General Parameters > PIT > PIT.PROFILE** parameter in **Configure User Data** must be set to "Pit_OPCPT_MPI.ini"in order to activate the interface.

Customizing

A variable of type **Byte** must be created in ProTool/Pro CS for the MDC_MMC cluster:

➢ Name:
   Value = "MDA_M1"
➢ Acquisition Cycle (s):
   Value = "2"
➢ Number of Elements:
   Value = "424"
➢ Range:
   Value = "DB"
➢ DB:
   Value = "13"
➢ DBB:
   Value = "0"

Note: "OPC Server" must also be selected in ProTool/Pro in the **Target system > settings** menu item.

☞ When machines/units are being customized (see chapter 3.2 "Customizing Machines/Units") the status field (128-bit array) is assigned to a machine or unit using the **Cluster** and **Position** parameters (Work Centers > Acquisition), whereby the **Cluster** parameter must always be set to "1".

Examples:

- Cluster 1, Position 1 represents the first status field in ProTool variable MDA_M1

- Cluster 1, Position 3 represents the third status field in ProTool variable MDA_M1

☞ During customizing the **Type of automatic Acquisition** parameter must also be used to define whether MDE and/or BDE data is to be acquired.

4.7   PLC Interface (WinCC)

☞ In order to activate the interface the **General Parameters > PIT > PIT.PROFILE** parameter has to
be set to "Pit_WinCC.ini" in **Configure User Data** (Start > Programs > MDA > Customizing
(User)).

The automatic acquisition of MDE data via the PLC interface relies on having a prefabricated interface
(MDC_MMC cluster) to the PLC. The interface has Plug&Play capability, but requires programming effort
in the PLC to supply the MDC_MMC cluster with MDE data.
The automatic acquisition of BDE data via the PLC interface also relies on having a prefabricated
interface (PDC cluster) to the PLC. The interface has Plug&Play capability, but requires programming
effort in the PLC to supply the PDC cluster with BDE data.
The MDC and PDC clusters are always in PLC data blocks. The MCD and PCD cluster contains the MDE
and BDE data for one machine/unit. Data for a maximum of 8 machines/units can be acquired using the
clusters. The structure of the MMC and PDC clusters is described in the chapter 4.1 "Cluster Data
Structures".

MDC Cluster

An MDC cluster corresponds to a variable in WinCC and contains a machine/unit. The MDC clusters can
be distributed over several data blocks. The data blocks can also be on different controllers. The length of
a data block is therefore variable (n * 84 bytes).

Customizing

A variable of type **Raw Data Type** must be created for each MDC cluster using the WinCC Explorer:
 ➢ Name:
       Value = "MDA_M1, MDA_M2, ..."
 ➢ Data Type:
       Value = "Raw Data Type"
 ➢ Data Range:
       Value = "DB"
 ➢ DB No.:
       Value = "13" (possible 14, 15, ...)
 ➢ Addressing:
       Value = "Byte"
 ➢ DBB:
       Value = "(n-1) * 84"
 ➢ Length:
       Value = "84"
 ➢ Raw Data Type:
       Value = "Transmit/receive block"

☞ The cluster is assigned to a machine or unit when the machines/units are customized (see chapter
3.2 "Customizing Machines/Units") using the **Cluster** and **Position** parameters (Work Centers >
Acquisition), whereby the **Position** parameter must always be set to "1".

Examples:
 • Cluster 1, Position 1 represents WinCC variable MDA_M1

 • Cluster 3, Position 1 represents WinCC variable MDA_M3

☞ When customizing the machines/units it must also be defined whether MDE and/or BDE data is being acquired using the **Type of automatic acquisition** parameter.

PDC Cluster

A PDC cluster corresponds to a variable in WinCC and contains a machine/unit. The PDC clusters can be distributed over several data blocks. The data blocks can also be on different controllers. The length of a data block is therefore variable (n * 132 bytes).

Customizing

Two variables of type **Raw Data Type** must be created for each PDC cluster using the WinCC Explorer. Variable 1 (PDC cluster):

- ➢ Name:
  - Value = "MDA_P1, MDA_P2, ..."
- ➢ Data Type:
  - Value = "Raw Data Type"
- ➢ Data Range:
  - Value = "DB"
- ➢ DB No.:
  - Value = "14" (possible 13, 15, ...)
- ➢ Addressing:
  - Value = "Byte"
- ➢ DBB:
  - Value = "(n-1) * 132"
- ➢ Length:
  - Value = "132"
- ➢ Raw Data Type:
  - Value = "Transmit/receive block"

Variable 2 (coordination flag in PDC cluster):

- ➢ Name:
  - Value = "MDA_PF1, MDA_PF2, ..."
- ➢ Data Type:
  - Value = "Raw Data Type"
- ➢ Data Range:
  - Value = "DB"
- ➢ DB No.:
  - Value = 14 (possibly 13, 15, ...)
- ➢ Addressing:
  - Value = "Byte"
- ➢ DBB:
  - Value = "(n-1) * 132"
- ➢ Length:
  - Value = "4"
- ➢ Raw Data Type:
  - Value = "transmit/receive block"

Notice: The coordination flag is in the first four bytes of the PDC cluster. For this reason the data block number (DB No.) and start byte address (DBB) must **always** be the **same** in variable pairs MDA_P1, MDA_P2, ... and MDA_PF1, MDA_PF2, ...!

        If MDE data is also acquired for a machine/unit, the WinCC variables MDA_M1, MDA_M2, ... in the MDC_Cluster must be the same as the WinCC variables in the PDC cluster MDA_P1, MDA_P2, ... and MDA_PF1, MDA_PF2, ...

☞ Assignment of the cluster to a machine or unit takes place when customizing the machines/aggregates (see chapter 3.2 "Customizing Machines/Units ") using the **Cluster** and **Position** parameters (Work Centers > Acquisition), whereby the **Position** parameter must always be set to "1".

Examples:

- Cluster 1, Position 1 represents WinCC variables MDA_P1 and MDA_PF1

- Cluster 3, Position 1 represents WinCC variables MDA_P3 and MDA_PF3

☞ When customizing the machines/units it must also be defined whether MDE and/or BDE data is being acquired using the **Type of automatic acquisition** parameter.

4.8   PLC Interface (SIMATIC Computing)

☞   In order to activate the interface the **General Parameters > PIT > PIT.PROFILE** parameter has to
be set to "Pit_S7_COMP.ini" in **Configure User Data** (Start > Programs > MDA > Customizing
(User)).

The automatic acquisition of MDE data via the PLC interface relies on having a prefabricated interface
(MDC cluster) to the PLC. The interface has Plug&Play capability, but requires programming effort in the
PLC to supply the MDC cluster with MDE data.
The automatic acquisition of BDE data via the PLC interface also relies on having a prefabricated
interface (PDC cluster) to the PLC. The interface has Plug&Play capability, but requires programming
effort in the PLC to supply the PDC cluster with BDE data.
The MDC and PDC clusters are always in PLC data blocks. The MDC and PDC cluster contains the MDE
and BDE data for one machine/unit. Data for a maximum of 8 machines/units can be acquired using the
clusters. The structure of the MDC and PDC clusters is described in the chapter 4.1 "Cluster Data
Structures".

In the **Pit_S7_COMP.ini** initialization file, OPC-related sections are especially important. Separate
[OPC.Cluster] sections for MDE and BDE data are provided for each of the 32 machines.

Example: The [OPC.Cluster1] section contains the addressing mode settings for the MDC block,
and [OPC.Cluster2] contains those for the PDC block of machine 1.

The addressing mode can be either symbolic, e.g.
        Item1=MDA_DATA/MACHINE_DATA/MDA_M1

or absolute, e.g.
        Item1=DB13.DBB4:BYTE[80]

Note: In the supplied **Pit_S7_COMP.ini** initialization file, the absolute addressing variant is
commented out.

In SIMATIC Computing, the addressing variant is set using the TagFile Configurator. For more
information, see SIMATIC Computing User Manual.

4.8.1 Symbolic addressing

The advantage of symbolic addressing is that no changes to the **Pit_S7_COMP.ini** initialization file are needed.
Symbolic addresses in the [OPC.Cluster] sections of the **Pit_S7_COMP.ini** initialization file are made up of three parts.

  ➢ MDA_DATA is the symbolic name of the STEP7 program in the SIMATIC Computing TagFile Configurator. This name must be set explicitly, otherwise a cryptic name such as SIMATIC_300-Station_CPU314(1) is automatically generated.

  ➢ MACHINE_DATA is the symbolic name of the data block (e.g. DB13) in the STEP7 program. This name must also be set explicitly, as it is blank by default.

  ➢ MDA_M1 is a symbol within this data block (in this case, an array of bytes).

To set up symbolic addressing for the SIMATIC Computing interface, proceed as follows.

  ☞ Create a symbol file using the TagFile Configurator in SIMATIC Computing (see SIMATIC Computing User Manual, Chapter 9.3)
  The S7 project you will need to create must comply with the MDA conventions. You can use the supplied source files **MDC_S7_COMP.txt** and **PDC_S7_COMP.txt** (both under the **STEP7 Demo** folder in the supplied CD-ROM) as a template for the S7 project.

  ☞ From the Windows **Start** menu, select the command **SIMATIC > PC Based Control > SIMATIC Computing Configuration** to access the **SIMATIC Computing Configuration** dialog box. Click the **OPC** tab.

  ☞ To use symbols for accessing data in the control engine or to access multiple control engines, select the **Connection via Tag Source** option and enter the name of the tag file. (Click on the **Browse** button to browse to the tag file.)

  ☞ Click the **Apply** button to apply your settings and confirm with **OK** to close the dialog.

4.8.2 Absolute addressing

With absolute addressing, the S7 addresses of the MDC and PDC block are directly written to the **Pit_S7_COMP.ini** initialization file, in the corresponding [OPC.Cluster] sections.

Note   With absolute addressing, SIMATIC Computing **limits** access to **only one** machine (connection to one control engine).
Please **pay attention** to this **limitation** during **customizing** the **Pit_S7_COMP.ini** initialization file.

  ☞ From the Windows **Start** menu, select the command **SIMATIC > PC Based Control > SIMATIC Computing Configuration** to access the **SIMATIC Computing Configuration** dialog box. Click the **OPC** tab.

  ☞ To connect to a specific control engine without using symbols, select the **Direct Connection** option and enter the name of the target computer and the name of the control engine.

  `wcS7= xx,a,b` (for other SIMATIC PLCs on the MPI network).

  where xx is the MPI address, a is the rack number and b is the slot number.

  ☞ Click the **Apply** button to apply your settings and confirm with **OK** to close the dialog.

  ☞ In the **Pit_S7_COMP.ini** initialization file, remove the comment signs beside the absolute addresses in the [OPC.Cluster] sections of the involved machine and comment out the corresponding symbolic addresses.

Example for machine 1:

```
[OPC.Cluster1]
;Item1=MDA_DATA/MACHINE_DATA/MDA_M1
Item1=DB13.DBB4:BYTE[80]
Size1=80

[OPC.Cluster2]
;Flag=MDA_DATA/PRODUCTION_DATA/MDA_PF1
Flag=DB14.DBB0:BYTE[4]
FlagSize=4
;Item1=MDA_DATA/PRODUCTION_DATA/MDA_P1
Item1=DB14.DBB4:BYTE[128]
Size1=128
WatchDelay=30
LookForDifferences=0
```

Explanation of the S7 data types used:
DB13.DBB4:BYTE[80] = data block 13 with 80 Bytes starting with Byte 4.

4.9   Overview

Interface	PIT.PROFILE	AWL File
PnP Interface (SINUMERIK)	Pit_NCDDE_PNP.ini	
PLC Interface Machine (SINUMERIK)	Pit_NCDDE_PLC_MACH.ini	MDC.awl PDC.awl
PLC Interface Multiple Machines (SINUMERIK)	Pit_NCDDE_PLC_MACH8.ini	MDC8.awl PDC8.awl
PLC Interface Units (SINUMERIK)	Pit_NCDDE_PLC_UNITS.ini	MDC_MMC.awl PDC.awl
OPC Interface (ProTool/Pro)	Pit_ProTool.ini	MDC.awl PDC.awl MDC8.awl PDC8.awl
OPC Interface (WinCC)	Pit_WinCC.ini	MDC.awl PDC.awl MDC8.awl PDC8.awl
OPC Interface (Simatic Computing)	Pit_S7_COMP.ini	MDC.awl PDC.awl MDC8.awl PDC8.awl
Test Interface	Pit_NCDDE_TST.ini	

5 MDA Machine and ProTool/Pro

If MDA Machine runs on an Operator Panel with ProTool/Pro and the user interface MDA Machine Evaluations should be used, several settings have to be done.

5.1   Starting sequence of MDA Machine and ProTool/Pro

The background processes of MDA Machine are started automatically during Login at Windows. The ProTool/Pro-Runtime-Project has to be started **before** MDA Machine, so MDA has access to the resources and programs (OPC-Server) of ProTool/Pro. It's best to start the ProTool/Pro-Runtime-Project within Autostart group too.

5.2   Switching from MDA user interface to ProTool/Pro

On some Operator Panels the MDA user interface MDA Machine Evaluations is running with full screen mode and it must be possible to switch directly to another user interface (e.g. ProTool/Pro) without using the status bar.

The function „F_CHANGEPROGR" is made available by MDA Machine Evaluations for this purpose.

If necessary you have to add access rights to function „Change to another program" for all user groups with **Customizing User Data** (Start > Program Files > MDA > Customizing (User)).

In addition to this you will have to specify the user interface to switch to. You may do this also in general settings (for delivery ProTool/Pro is set by default).
To do this set one of the following parameter within **Customizing User Data**:

- General Parameter → GUI_MDE → CHANGEPROGR_CLASS
  Here you can enter the class name of the desired user interface.
  For ProTool/Pro the following must be given: „SIMATIC ProTool/Pro RT"

- General Parameter → GUI_MDE → CHANGEPROGR_TITLE
  Here you can enter the window title of the desired user interface.
  If you specify a class name above, do not enter anything here.

5.3   Switching from ProTool/Pro to MDA Machine Evaluations

Normally ProTool/Pro is running in full screen mode and especially at some Touch Panel you are not able to switch to another program with the key code ALT+Tab.
In order to switch to the user interface MDA Machine Evaluations from ProTool/Pro using a pushbutton, you will have to insert a script into the ProTool/Pro project:

- Start Script, e.g. with name "ChangeMDA" of type "Sub" without parameters.

- Insert the following script:
  ```
  Dim Bde
  Set Bde = CreateObject("EventHandler.EventObject")
  Bde.BroadcastEvent("PopupWinMDE")
  ```

- Insert a pushbutton in ProTool/Pro project on any „diagram".

- Call the above script "ChangeMDA" within functions of this pushbutton properties.

The delivery-CD contains a ProTool/Pro-Example-Project (\ProTool Demo).

Notice: The script changes the focus to MDA Machine Evaluations. The program MDA Machine Evaluations has already to be started at this moment ! To ensure that, the program should

---

be started automatically by MDA Machine starting process. Two entries have to be inserted in <MDADIR>\MDA\startmde.ini to do that:

[START]
File4=*<MDADIR>*\MDA\MdeGUI.exe

[WATCH]
File4=MdeGUI

6 Interface to MDA Cell and MDA Plant

6.1 Overview

MDA is designed for group solutions. One MDA server (MDA Cell or MDA Plant) and several Workstations (acquisition PC's/MMC's) can work together in a group. The data acquired on the Workstations can be transferred to the MDA server, meaning that central evaluations can be carried out there. Workstation customizing can also be performed on the MDA server.
The data transfer between the MDA server and the Workstations is handled by the host link (*MdeTransfer)*. MdeTransfer copies the data over network drives as files. The file can be created manually or cyclically (MdeTransfer).

  ➢ **Download** from MDA server: manual

  ➢ **Upload** to MDA server: automatic, cyclic
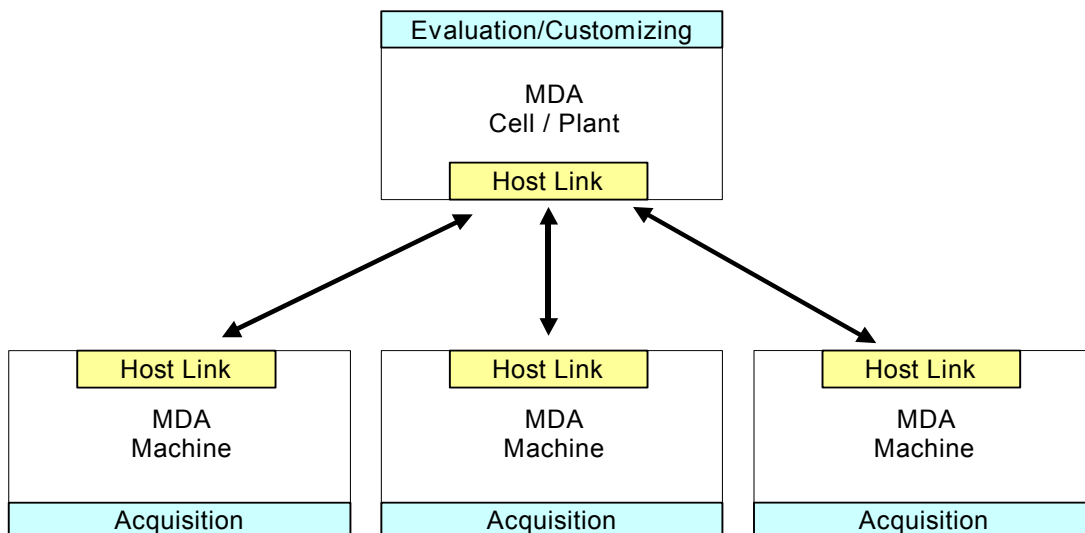
As standard, the following data **can** be transferred from the server to the Workstations by **manual download**:

  ➢ **CUSTOMIZE_DATA**:
    Contains all global parameters, user rights and user names that have been configured using **Configure User Data** in the **General Parameters**, **User Groups** and **User Names** folders.

  ➢ **CUSTOMIZE_STATES**:
    Contains all machine statuses and configuration of the OEE indicators that have been configured using **Configure User Data** in the **MDA states** and **OEE Indicators** folders. The evaluation definitions for the machine statuses are also transferred.

  ➢ **CUSTOMIZE_MACHINES**:
    Contains all work centre data (machines/units) that have been configured using **Configure User Data** in the **Work Centers** folder.

  ➢ **CUSTOMIZE_TEXTS**:
    Contains all texts that have been configured using **Configure OEM Data** in the **Screen Texts** and **Message Texts** folders.

  ➢ **CUSTOMIZE_OEM**:
    Contains all function key, menu line, toolbar, button, screen form and table definitions and all texts

---

that have been configured **Configure OEM Data** in the **Function Keys**, **Screen Forms**, **Screen Texts** and **Message Texts** folders.

➢ **CUSTOMIZE_TRANSFER**:
Contains all configuration data for the host link that has been configured using **Configure User Data** (see chapter 3 "Customizing") in the **Data Synchronization** folder.

➢ **CALENDAR_DATA**:
Contains all works calendar data (shift and day types, working time models, calendar).

➢ **ORDER_DATA**:
Contains the part types to be manufactured and that have been produced on the MDA server using the Part Type Evaluations user interface.

➢ **ALARM_TEXTS**:
Contains all alarm texts which are used at the Workstation.

As standard the following data is transferred from the Workstations to the MDA server by means of **automatic, cyclic uploading**:

➢ **SYSTEM_LOGBOOK**:
All system logbook messages that have been generated on the Workstations.

As standard, the following data is transferred immediately from the Workstations to the MDA server by MdeReport/TcpIpAdapter:

➢ MDE_LOGBUCH (actual MDE statuses, machine counters, part type counters)

➢ ALARM_LOGBUCH (Current alarms)

6.2   Configuration Variants

The MDA configuration is stored in a database. In principle, several configuration variants can be stored in such a database, whereby one variant is always the active configuration. Among other things the standard scope of delivery contains the following configuration variants:

DEF_WS_STDPC	Workstation Standard PC
DEF_WS_PC670	Workstation SIMATIC Panel PC 670
DEF_WS_FI45	Workstation SIMATIC PC FI45
DEF_WS_MMC	Workstation MMC with OP031/OP032
DEF_WS_OP10	Workstation MMC with OP010
DEF_WS_OP12	Workstation MMC with OP012
DEF_WS_OP15	Workstation MMC with OP015

The MDA server variant contains all the configuration variants, so that a download of the respective variant can be triggered via central customizing.

The following download orders are (partially) **configuration dependent:**
  ➢ CUSTOMIZE_DATA
  ➢ CUSTOMIZE_OEM
  ➢ CUSTOMIZE_STATES
  ➢ CUSTOMIZE_TRANSFER

7   Tips & Tricks

7.1   Operating the MMC

<u>Start the MMC in service mode:</u>

If you want to install new software or change system settings, you must start the MMC in service mode. Set service mode as follows:

> ☞ **Starting MMC in service for versions P4.x / P5.x**
> While Windows is booting, briefly press key 6 once when the *Starting Windows 95* screen is displayed. Then under item 4 in the menu, choose **Start Windows (Service Mode)**, and enter the password. Then, in menu item 2, choose **Windows (changing Environment for MMC2).**

> ☞ **Starting MMC in service mode for version P6.x**
> If the SINUMERIK menu appears while Windows is booting, press the Arrow Down key and Enter or Input. Then under item 4 in the menu, choose **Start Windows NT (Service Mode)** and then select **Standard Windows NT (without starting SINUMERIK HMI)**. The MMC boots after this. The password must be entered if this is requested.

<u>Back up the MMC environment:</u>

After installing additional software components on the MMC, you must back up the MMC environment (that contains, amongst other things, the Windows Registry). **It is absolutely crucial to back up the MMC environment,** since at activation or restarting of the MMC the system automatically starts up on the basis of the backed-up MMC environment.

> ☞ **Backing up the MMC environment with version P4.3:**
> You can save the MMC environment if you shutdown the MMC in service mode using the Windows Start button and dialog box choice **Shutdown the computer.** After shutting down the computer, when the system prompts you to **Save Windows Environment for next MMC start (Y/N),** press the **Y** key. At the next prompt, select item **2, "Overwrite your last saved Windows environment with the current one".** After this, you must start the MMC again in service mode and shut it down again. At the next prompt, choose item **1, "Save the last saved Windows environment with the current one".**

> ☞ **Backing up the MMC environment with version P4.4:**
> You can save the MMC environment if you shutdown the MMC in service mode using the Windows Start button and dialog box choice **Shutdown the computer.** The system backs up the environment if you enter **Y** at the **Save Windows Environment for next MMC start [Y/N]** prompt. It is also possible to overwrite the backup of the environment with the current one by entering **Y** when the system prompts you **Do you want to backup your old Environment [Y,N].** After this, you must start the MMC again in service mode, shut it down again and back up the MMC environment again.

> ☞ **Backing up the MMC environment with versions P5.x:**
> You can save the MMC environment if you shutdown the MMC in service mode using the Windows Start button and selecting **Shutdown the computer**. After this, you must start the MMC again in service mode, shut it down again and back up the MMC environment again.

> ☞ **Backing up the MMC environment with version P6.x:**
> An automatic back-up of the MMC environment is made if the MMC is shutdown in service mode.

7.2   Installation on the MMC

<u>Interlink Installation</u>

☞ If the MMC does not have a CD-ROM, MDA can be installed in Interlink Mode using a serial (slow!) or parallel null modem cable: In service mode, select item 8 **Start PC Link** in the menu and enter the password. Subsequently the CD can be copied to the MMC hard disk from a computer using Interlink and installed from there.

☞ The **Config.sys** file on the PC (MS-DOS, Windows 3.11, Windows 95) must contain the following line:
    "DEVICEHIGH=<Pfad>\interlnk.exe /drivers:1 /noprinter"

<u>Network Installation</u>

<u>If the MMC does not have a CD-ROM, MDA can be installed via network. You have to copy the content of all directories from CD to a shared network directory.</u>

<u>MDA Files for MMC Integration</u>

The MMC integration configuration files (\add_on\regie.ini and \add_on\language\re_??.ini) are created or modified by MDA during installation. "??" stands for the country code, e.g. gr for German and uk for English. The installation program stores the entries for MMC integration in **Regie.MDA** and **Re_xx.MDA** as an example.
The **\add_on\regie.MDA** file contains the following:

➢ [TaskConfiguration]

• Task16 = Name := Oemframe, CmdLine := "c:\\Add_on\\MDA\\MdeGUI.exe", Timeout := 60000, HeaderOnTop := True, PreLoad := False, WindowName := "MdeGUI"

   Task17 = Name := Oemframe, CmdLine := "c:\\Add_on\\MDA\\PdeGUI.exe", Timeout := 60000, HeaderOnTop := True, PreLoad := False, WindowName := "PdeGUI"

   Task18 = Name := Oemframe, CmdLine := "c:\\Add_on\\MDA\\AlarmGUI.exe", Timeout := 60000, HeaderOnTop := True, PreLoad := False, WindowName := "AlarmGUI

• ;Task16 = Name := Oemframe, CmdLine := "c:\\Add_on\\MDA\\MdeGUI.exe", Timeout := 60000, HeaderOnTop := True, PreLoad := False, WindowName := "MdeGUI", TerminateTasks := (Oemframe: 17, Oemframe: 18)

• ;Task17 = Name := Oemframe, CmdLine := "c:\\Add_on\\MDA\\PdeGUI.exe", Timeout := 60000, HeaderOnTop := True, PreLoad := False, WindowName := "PdeGUI", TerminateTasks := (Oemframe: 16, Oemframe: 18)

• ;Task18 = Name := Oemframe, CmdLine := "c:\\Add_on\\MDA\\AlarmGUI.exe", Timeout := 60000, HeaderOnTop := True, PreLoad := False, WindowName := "AlarmGUI", TerminateTasks := (Oemframe: 16, Oemframe: 17)

➢ [StartupConfiguration]
   Startup31 = Name := startmde, Timeout := 0

   In this case, Task16 is the machine evaluation task line for the MDA user interfaces, Task18 is the task line for the alarm and message evaluations and Task 17 is the task line for the part type evaluations. The following lines are an example of mutual termination of the user interfaces (recommended under MMC P4.x / P5.x, because of possible Windows resource problems). When making manual entry changes on MMC P6 the drive letter must be changed from **c:** to **f:**!

   The **\add_on\language\re_??.MDA** file contains the following:

➢ [HSoftkeyTexts]

- HSK16 = "MDA Machines"

- HSK17 = "MDA Orders"

- HSK18 = "MDA Alarms"

For the MDA user interfaces *MDA Machines* is the function key text for machine evaluations, *MDA Alarms* is the function key text for alarm and message evaluations and *MDA Orders* is the function key text for part type evaluations.

The *MDA* setup supports German, English, French, Italian and Spanish, whereby the function key text is the same for all the above-mentioned languages. The entries have to be configured manually if different texts or languages are required.

<u>MMC upgrade (P4/P5 only)</u>

Before performing an MMC upgrade (P4/P5 only) the *MDA* files **startmde.exe** and **startmde.ini**, which can be found in the **\mmc2** directory, must be backed up and copied back into the directory after the MMC upgrade.

<u>Determining NCU names</u>

In order to allow *MDA* to communicate with the MMC (Ncdde server) the NCU name for *MDA* must be configured correctly (see chapter 4 "Interfaces").
The NCU file can be found in the line starting with *NcddeMachineName* in the following files (priority from left to right)
        "\user\mmc.ini, \oem\mmc.ini, \add_on\mmc.ini, \mmc2\mmc.ini, \hmi_adv\mmc.ini"
(default value **ncu840d**) e.g.:
        NcddeMachineName = ncu840d

<u>Test of status data acquisition via DB13</u>

Bits in data block DB13 can be displayed and modified using the PLC status function in the MMC diagnosis application.
The first status byte of the first unit in DB13 is in data word DW52 and can be displayed by entering DB13.DBB104. In the displayed bits, e.g. 01000000, the left-hand bit corresponds to Bit_1 in the *MDA* configuration:
        DB13.DBB104   01000000.

7.3   Test Program TestSrv.exe

Test program **TestSrv.exe** makes it possible to enter MDE and BDE data via a simulated PLC interface. The coupling module obtains the data from the test program rather then the PLC. Data acquisition can be simulated via the MDC cluster, the MDC_MMC cluster or the PDC cluster.

☞ In order to test the simulated interface using the test program, the value "Pit_NCDDE_TST.ini" must be entered in the **General Parameters > PIT > PIT.PROFILE** folder of the **Configure User Data** program (post-installation presetting).

☞ The test server can be started by removing the semicolon in ";File4=..." in file **Startmde.ini** ion the **[START], [WAIT] and [WATCH]** sections.

☞ In the MMC variant, the test server can only be started in service mode. The **Machine Evaluations** (Start > Programs > MDA > Machine Data Evaluation) and **Part Type Evaluations** (Start > Programs > MDA > Production Data Evaluation) user interfaces have to be explicitly started in service mode.

☞ If it is not possible to display all the screen form controls in the test program because of lack of resolution, scroll bars are displayed at the bottom and right-hand edges of the window. The display area can also be scrolled using the **Ctrl** key and the arrow keys.

☞ The status field for each unit (currently 1 – 8) consists of 128 bits, which are in groups of 8. The significance of the bits runs from left to right and top to bottom in increasing order.
In order top set bit a you therefore have to enter a value of "1" in the top left input field, and to set bit 128 you have to enter a value of "1" in the bottom right input field.

Acquisition of MDE data via the MDC_MMC cluster

The following input fields are occupied in the screenshot shown below:

Name of Input Field	Input Field Value
Cluster	"**1**" (addressing of machine/unit with cluster 1 and position 3)
3rd Unit Status	MDE status in bit 2: "**1**"
1st Cycle Count	Number of good parts = "**5**"
2nd Cycle Count	Number of waste parts = "**3**"
3rd Cycle Count	Number of rework parts = "**0**"

The MDE status as status Bit_2 and the counter are transmitted to the acquisition server and booked there by pressing the **Update** button.

Warning: A prerequisite for acquisition is that a work centre was defined when customizing the machines/units (with cluster 1, position 3), for which MDE acquisition has been activated and to which an MDE status group has been assigned, for which the selected MDE status has also been configured with the name Bit_2 (e.g. MDE status group 1000 in the standard delivery).

The cluster must **always be 1**, i.e. the **Cluster** field must always contain a value of "1".

The status bits for up to eight machines/unit can be set using the test program (fields: **1**st unit status, ..., **8**th unit status). The counters relate to the machine(s)/unit(s) addressed via cluster 1 and position 3.

Please note that the counters are absolute.

If BDE data acquisition has also been activated for the work centre, BDE data is also transmitted to the acquisition server.
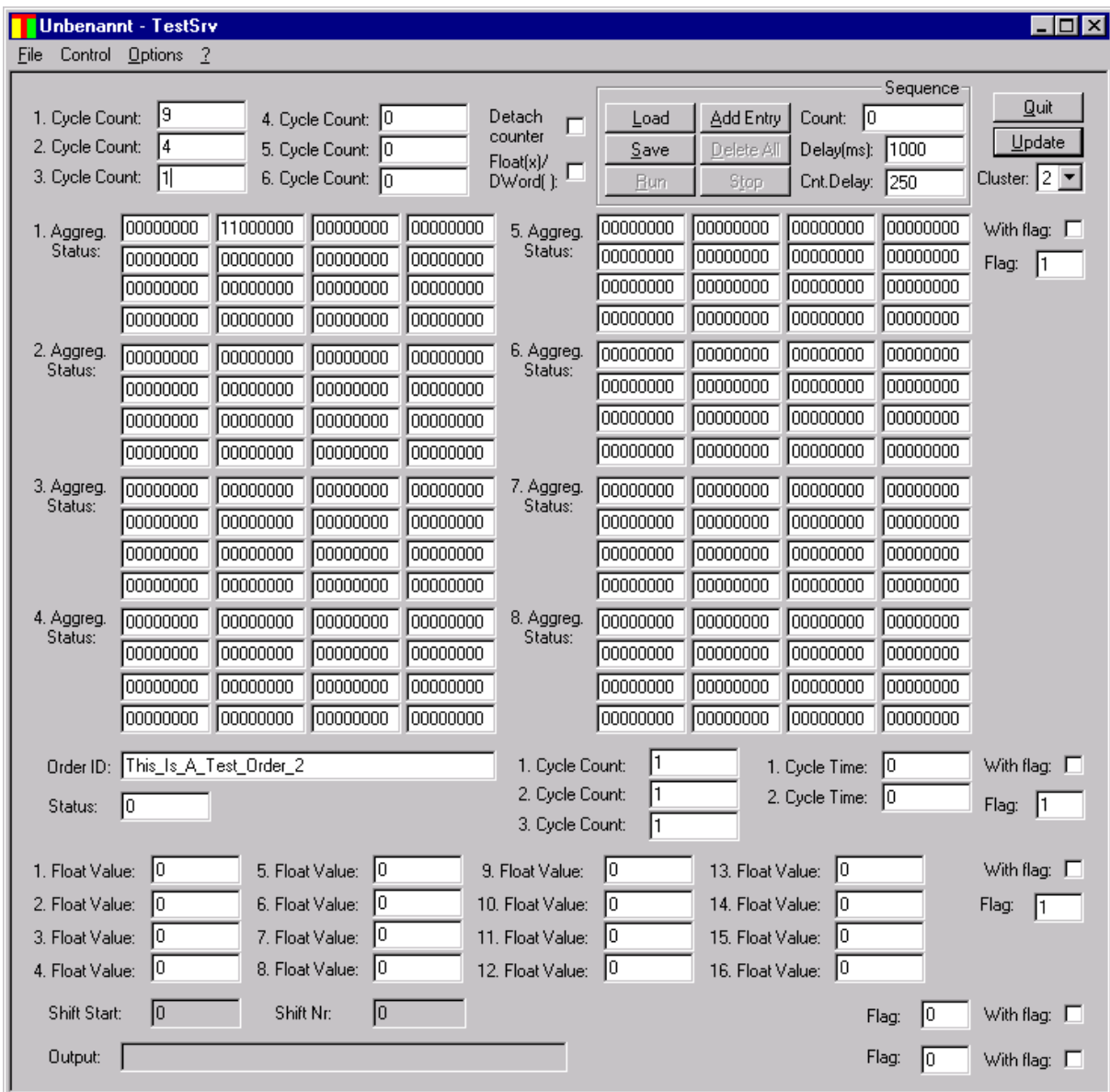
MDE Data Acquisition via MDC cluster

The following input fields are occupied in the following screenshot:

Name of Input Field	Input Field Value
Cluster	"**2**" (Machine/unit addressing with cluster 2 and position 1)
1. Unit Status	MDE status in bit 9 and bit 10: "**1**"
1. Cycle Count	Number of good parts = "**9**"
2. Cycle Count	Number of waste parts = "**4**"
3. Cycle Count	Number of rework parts = "**1**"

☞ Pressing the **Update** button will cause the MDE statuses (as status Bit_9 and Bit_10) and the counters to be transmitted to the acquisition server via the coupling module and booked there.

Warning: A prerequisite for acquisition is that a work centre was defined when customizing the machines/units (with cluster 2, position 1), for which MDE acquisition has been activated and to which an MDE status group has been assigned. The selected MDE statuses with the names Bit_9 and Bit_10 must also be configured for the work centre (e.g. MDE status group 1000 in the standard delivery).

The position must **always be 1**, i.e. only the status bits of the **1st Unit Status field are evaluated**.

It must be noted that the counters are absolute values.

If BDE acquisition has also been activated for the work centre, BDE data will also be transmitted to the acquisition server..

BDE data acquisition via PDC cluster

The following input fields are occupied in the screenshot below:

Name of Input Field	Input Field Value
Cluster	"**2**" (Addressing of machine/unit with cluster 2 and position 1)
1. Cycle Count	Number of good parts = "**4**"
2. Cycle Count	Number of waste parts = "**2**"
3. Cycle Count	Number of rework parts = "**1**"
1. Cycle Time	Total target cycle = "**60**" (seconds)
2. Cycle Time	Total actual cycle = "**70**" (seconds)
Order ID	"**NC-Prog4711**" from the **Identification** field in **MDA Part Type Evaluations > Process Part Type**

☞ Clicking on the **Update** button causes the part type identification, the counters and the clock times to be transmitted to the acquisition server via the coupling module and booked there.
The PDC cluster is also released for reading (by the coupling module) by setting the relevant flag (coordination value) to **1**. If the coupling module has read the PDC cluster, the counters, the clock times and the coordination value are automatically reset to **0**. Only then can new values be entered and transmitted. Since the read cycle for the PDC cluster is 30 seconds, a delay of up to 30 seconds is possible.

Warning: A prerequisite for acquisition is that a work centre for which BDE acquisition has been activated has been defined during machine/unit customizing (with cluster 2, position 1).

It must be noted that the counters are delta counters.

If MDE acquisition has been activated for the work centre, MDE data is also transmitted to the acquisition server.

7.4   Conversion Program MdeConvertDbs.exe

The **MdeConvertDbs.exe** conversion program makes it possible to convert a configuration variant to another configuration variant with regard to the screen layout (screen size, function keys etc.) This allows the configuration for a standard PC ("Workstation Standard PC") to be converted to a configuration for OP012 ("MMC Workstation with OP012").

Warning   A workstation configuration cannot be converted to a Supervisor configuration and vice-versa!

The conversion program is mainly required when different hardware is being used. In this case standard PC customizing can take place initially. This configuration variant can then be copied to the configuration user interface and converted to other variants using the conversion program (e.g. OP012, FI45).

7.5   Help Files

At installation, help files were copied to the *MDA* directory:
MdeGUI_deutsch.hlp: Help file for user interface **Machine Evaluations***. This help file is also called from the user interface if *MDA* is running on a normal PC.
PdeGUI_deutsch.hlp: Help file for user interface **Part Type Evaluations**. This help file is also called from the user interface if *MDA* is running on a normal PC.
AlarmGUI_deutsch.hlp: Help file for user interface **Alarm and Message Evaluations**. This help file is also called from the user interface if *MDA* is running on a normal PC.
WkalGUI_deutsch.hlp: Help file for user interface **Works Calendar**. This help file is also called from the user interface if *MDA* is running on a normal PC.
PrjGUI_deutsch.hlp: Help file for user interface **Configure User Data** and **Configure OEM Data**. This help file is also called from the user interface if *MDA* is running on a normal PC.
The help files are also available in English (**MdeGUI_english.hlp**, **PdeGUI_english.hlp,
AlarmGUI_english.hlp**, **WkalGUI_english.hlp**, **PrjGUI_english.hlp**).
The help functions are disabled on user interface if *MDA* was installed on an MMC. You only can start them using the Windows Explorer in service mode.

7.6   Hardcopies

On the MMC, you can generate hardcopies of *MDA* screen forms by clicking on the topmost vertical softkey. The system outputs hardcopies to the Windows default printer.

7.7   Missing Write Permissions as of Windows 2000

As of Windows 2000 it can happen that normal users don't have write permissions for the directory C:\ . This leads to the situation that the directory tree for MDA being created during the installation has also no write permissions for normal users. If now MDA is to be operated by such a normal user without the privileges of an administrator or power user errors occur.
After the installation you should check and assure in the Windows Explorer if also normal users have write permissions in the MDA directories.

7.8   No data transfer to / from MDA server by ‚MdeTransfer'

Check if the shared directory **Transfer** on the MDA cell/plant computer has full control permissions for the workstation. The ACCESS database for the download and upload must lie in the sub-directory **Group**. If there are no files created after the initiation of the download the error is due to the server side. If the files are created but not fetched **MdeTransfer** on the workstation has either no access to this shared directory or it does not run.

7.9   There is no logbook data sent to the MDA server

Check the Message Manager connection from the workstation to the MDA server. For this you enter the following command in a command prompt window on the workstation in the directory 'tools':
        mmping ISM1
As result the message *Open OK* must be shown. If the error 102 is reported the Message Manager license is not valid or has not be found. Does the file C:\Siemens\MDA\etc\Protect.key or \Add_on\etc\Protect.key exist? Is the environment variable MM_FCT defined and does it reference to the file Protect.key (can be checked with the command 'Set' in the command prompt)?

In the case of other errors check first if the access to the server is possible at all. For this you enter the following command in a command prompt window:

      ping pdaservr   'pdaservr' being the MDA server name
                                   (see the 3rd column in the file C:\Siemens\MDA\etc\function)

If the server name could not be resolved the server name must be made known by the network administrator or it must be entered in the file %Windows%/system32\host .

If mmping showed the error 4 the MDA start service or or the process ISM are probably not started on the server. Start the service again. If only the process ISM was not started look for ISM error messages in C:\Siemens\MDA\Errors .

Alternatively the configuration of the Message Manager can be faulty. Check the files C:\Siemens\MDA\etc\function and %Windows%/system32\Services on the server.

By executing the script MdeReport.bat in the directory 'tools' you can check if the entry LastReportedId_MDE_LOGBUCH in the table PDA_REGISTRY of the MDA database on the workstation matches the highest number LFD_NR in the table MDE_LOGBUCH . Is there any error message of the programs MdeReport or TcpIpAdapter on the workstation (error logbook of GUI-MDE or script 'listerrors.bat' in the directory 'tools')?

7.10   Backup of customer's database and restoring on a diagnostic computer

Often it would be desirable to analyze the errors occurring at a customer's site on a development computer. For this the real data of the customer are necessary. The export of data and the restore on a development computer into an existing database (MDA being installed) can be accomplished in different ways:

7.10.1 Database files from MSDE

Using MSDE as database server you can directly transfer the database files. But for this all MDA processes must be terminated.

Execute the script **DetachDB.bat** in the folder <MDADIR>\MDA\Install on the customer's computer. Thereby the database will be released.

Now you can copy the database files in the folder <MDADIR>\MDA\Data into the corresponding folder on the development computer.

Execute the script **AttachDB.bat** in the folder <MDADIR>\MDA\Install on the development computer. Thereby the database is attached on the database server.

7.10.2 Using the database Microsoft Access

You can copy the contents of the MDA database into the Microsoft Access database **MDA.mdb** . To this the script **CopyMDADB.bat** in the folder <MDADIR>\tools is executed. Now you bring the file <MDADIR>\tools\ MDA.mdb to the development computer. There you can directly work with the Access database or copy the data into the MDA database via the program CopyDatabase.exe.

7.11   Increase database size

The maximum database size can be increased by script <MDADIR>\Install**ExpandDBsize.bat**. As starting parameter you have to give the password of database user sa. In file ExpandDBsize.sql you can change the maximum database size before execute the script.

7.12   Computer names modified (Master CD with ghost image)

7.12.1   Computer name of the workstation self

During the installation the name of the computer self is prompted and stored as an environment variable for the remote scrpts. If the computer name is modified later on the environment variable must be adjusted. This is particularly the case if a machine tool builder creates a master CD and distributes it to the individual controls via ghost image.

MMC103 / Standard PC with Windows 95/98:
      Modify the following line in the file C:\autoexec.bat:
      set COMPUTERNAME=xxx

PCU50 / Standard PC with Windows NT, Windows 2000:
      Windows Control Panel/Environment Variable   COMPUTERNAME

7.12.2   Computer name of the MDA server

During the installation the name of the MDA server computer is promted and stored in various places. If the computer name is modified later on the following files must be adjusted:

MMC103 / Standard PC with Windows 95/98:
      File C:\autoexec.bat: set SERVERPCNAME=xxx

PCU50 / Standard PC with Windows NT, Windows 2000:
      Windows Control Panel/Environment Variable   SERVERPCNAME

Files
F: \ add_on \ MDA \ MdeServer.ini
F: \ add_on \ MDA \ StartMDE.ini
F: \ add_on \ StartMDE.ini
F: \ add_on \ MDA \ MdeUpd.bat
F: \ add_on \ MDA \ ReplySrv.bat
F: \ add_on \ etc \ function

7.13   Log files of Installation

User inputs of the dialogs and some internal function calls are written to the log file <MDADIR>\Install\aSetup.log. If there are any errors during installation this file may help you.

Logging informations of MSDE setup are in file msde.log.

Further log files contain informations of the database scripts.

8   Glossary

Button

A graphical element in a dialog box that triggers a certain function when it is activated. The button is activated by clicking on it with the mouse. If the button has the focus (the element that is currently pre-selected) the Return key (Enter key) can also be pressed.

Combo Box

A control that combines an edit control with a list box. This allows the user to type in an entry or choose one from the list.

Control Engine

As used by the Computing software, the term **control engine** applies to a processor or program that manages and manipulates data, which is used to control a process or machine.
The control engine can be either software or hardware.

Menu Item

Also known as "Selection item". This is an entry in a menu that can be selected using the keyboard or mouse. If a menu entry is not available for a certain situation (or unsuitable), the entry is greyed out in some applications, i.e. darker than the applicable menu items.

MMC

MMC stands for Man-Machine Communication. The MMC software module covers the operation, observation, programming, setup and diagnosis areas. However, the MMC102/103 with hard disk is also known as the MMC module, and provides OEM users with graduated performance by using different processor types and different memory configurations.

NCDDE Server

The NCDDE server is an auxiliary program that handles the variable service, domain service and program instance services data transfer tasks. DDE refers to the dynamic exchange of data between Windows applications.

Radio Button

Means of selecting options in a dialog box in graphical user interfaces. The option field is represented by a small circle that contains a smaller, solid circle when it is selected – otherwise it is empty. Option fields are similar to the station buttons on a radio: if you press a button, the last button that was selected jumps back to its original position. The same applies to the option fields – only one of the options in a group can be selected at any given time. If simultaneous selection of more than one option is required, check boxes are used.

SIMATIC Computing

The Computing software uses the ActiveX (also known as OLE) technology of Microsoft to provide you with access to the data provided by your control engine.
The Computing software consists of Simatic Controls for accessing control engines, an OPC server (OLE for Process Control), a TagFile configurator for creating tag files that allow symbolic addressing, a configuration tool for configuring remote access and an OLE container (SoftContainer) for creating process forms with the SIMATIC controls.

Tab Control

In graphical user interfaces you can use tab controls to present several pages of information as a single set. The control imitates printed index cards, which can be for example found in card index boxes.
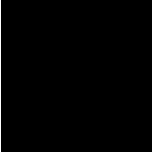
Tree View

A way of depicting hierarchical structures that usually contains linked folders. The top folder is called the root. With the exception of the root, each folder has exactly one parent folder, and each folder can have several child folders.

Windows Menu

A menu containing commands for adapting a window. Click on the program symbol or the document symbol on the left of the title bar to open the window menu.