Electric Drives and Controls | Hydraulics | Linear Motion and Assembly Technologies | Pneumatics | Service

Rexroth
Bosch Group

Rexroth IndraMotion MTX 12VRS Commissioning

R911334377
Edition 01

Commissioning Manual

Title Rexroth IndraMotion
MTX 12VRS
Commissioning

Type of Documentation Commissioning Manual

Document Typecode DOK-MTX***-STARTUP*V12-CO01-EN-P

Internal File Reference RS-ce037e1a0efb69d90a6846a001b5623d-1-en-US-5

Purpose of Documentation This documentation describes the commissioning of the IndraMotion MTX control. Apart from a complete overview, commissioning and configuration of the axes and the user interface as well as the PLC data are described.

Record of Revision

Edition	Release Date	Notes
Edition 01	05.2011	First edition for 12VRS

Published by Bosch Rexroth AG

Bgm.-Dr.-Nebel-Str. 2 ■ 97816 Lohr a. Main, Germany

Phone +49 (0)93 52/ 40-0 ■ Fax +49 (0)93 52/ 40-48 85

http://www.boschrexroth.com/

System Development Machine Tools MTX, ToMa (FrWe/MePe)

Note This document has been printed on chlorine-free bleached paper.

Table of Contents

Page

Table of Contents

Page

Table of Contents

Page

Table of Contents

Page

Table of Contents

Page

Table of Contents

Page

Table of Contents

Page

Table of Contents

Page

Page

Table of Contents

Page

Table of Contents

Page

Table of Contents

Page

Table of Contents

Page

Table of Contents

Page

Table of Contents

Page

Table of Contents

Page

Table of Contents

Page

Table of Contents

1 About this Documentation

1.1 Validity of the Documentation

Target group

This documentation is intended for users commissioning a control of the type IndraMotion MTX. Apart from a complete overview, configuration of the axes and the user interface as well as the PLC data are described.

This documentation supports the user in the following phases:

- Engineering and
- Commissioning

1.2 Required and Supplementing Documentations

1.2.1 Selecting

Documentation titles with type designation codes and parts numbers

Rexroth IndraMotion MTX xxVRS System Description
DOK-MTX***-SYS*DES*V12-PRxx-EN-P, R911334367
This documentation describes the Rexroth IndraMotion MTX control. It includes the designs, technical data, interfaces as well as the configuration of the control components.

xx Respective version or edition

Fig. 1-1: MTX documentation overview - Selecting

1.2.2 Configuring

Documentation titles with type designation codes and parts numbers

Rexroth IndraMotion MTX xxVRS Functional Description
DOK-MTX***-NC*FUNC*V12-RExx-EN-P, R911334355
This documentation describes the functions of the Rexroth IndraMotion MTX. The basic commissioning steps and the functions of the control are given as description and handling instruction.
Rexroth IndraMotion MTX xxVRS Machine Parameters
DOK-MTX***-MA*PAR**V12-RExx-EN-P, R911334363
This documentation describes handling, design and modification of the Rexroth IndraMotion MTX parameters available. It also includes the functions of the NC Configurator and its operation.
Rexroth IndraMotion MTX xxVRS PLC Interface
DOK-MTX***-PLC*INT*V12-PRxx-EN-P, R911334381
This documentation describes interface signals and program function blocks for the integrated PLC.

xx Respective version or edition

Fig. 1-2: MTX documentation overview - Configuring

1.2.3 Commissioning

Documentation titles with type designation codes and parts numbers

Rexroth IndraWorks xxVRS Software Installation DOK-IWORKS-SOFTINS*V12-COxx-EN-P, R911334396 This documentation describes the IndraWorks installation.
Rexroth IndraWorks xxVRS Engineering DOK-IWORKS-ENGINEE*V12-APxx-EN-P, R911334388 This documentation describes the application of IndraWorks in which the Rexroth Engineering tools are integrated. It includes instructions on how to work with IndraWorks and how to operate the oscilloscope function.
Rexroth IndraMotion MTX xxVRS Commissioning DOK-MTX***-STARTUP*V12-COxx-EN-P, R911334377 This documentation describes the commissioning of the IndraMotion MTX control. Apart from a complete overview, commissioning and configuration of the axes and the user interface as well as the PLC data are described.
PLC program development with Rexroth IndraLogic DOK-CONTRL-IL**PRO*V01-AWxx-EN-P, R911305036 This documentation describes the operating and programming interface IndraLogic.
Rexroth IndraWorks xxVRS Basic Libraries IndraLogic 1G DOK-IL*1G*-BASLIB**V11-LIxx-EN-P, R911332305 This documentation describes the system-comprehensive PLC libraries.
Rexroth IndraWorks xxVRS IndraLogic 2G Programming Instruction DOK-IWORKS-IL2GPRO*V12-APxx-EN-P, R911334390 This documentation describes the PLC programming tool IndraLogic 2G and its application. It includes the basic usage, first steps, visualization, menu items and editors.
Rexroth IndraWorks 12VRS Basic Libraries IndraLogic 2G DOK-IL*2G*-BASLIB**V12-LIxx-EN-P, R911333835 This documentation describes the system-comprehensive PLC libraries.
Rexroth IndraWorks xxVRS HMI DOK-IWORKS-HMI*****V12-APxx-EN-P, R911334392 This documentation describes the functions, configuration and operation of the user interfaces IndraWorks HMI Engineering and IndraWorks HMI Operation.
Rexroth IndraWorks xxVRS WinStudio DOK-IWORKS-WINSTUD*V12-APxx-EN-P, R911333844 This documentation describes the installation of the software, working with WinStudio and the creation and operation of applications.
Rexroth IndraWorks xxVRS FDT Container DOK-IWORKS-FDT*CON*V12-APxx-EN-P, R911334398 This documentation describes the IndraWorks FDT Container functionality. It includes the activation of the functionality in the project and working with DTMs.
Rexroth IndraWorks xxVRS Simulation DOK-IWORKS-SIMU****V10-AWxx-EN-P, R911327491 This documentation describes the functions of the simulation components View3D, virtual control panel, virtual control and its operation in IndraWorks.

xx Respective version or edition

Fig. 1-3: MTX documentation overview - Commissioning

1.2.4 Operating

Documentation titles with type designation codes and parts numbers

Rexroth IndraMotion MTX xxVRS Standard NC Operation DOK-MTX***-NC*OP***V12-APxx-EN-P, R911334371 This documentation describes the operation of the standard user interface of the NC control of the Rexroth IndraMotion MTX. It includes the operation of the interface, the NC program development as well as the tool management.
Rexroth IndraMotion MTX xxVRS Programming Manual DOK-MTX***-NC**PRO*V12-RExx-EN-P, R911334359 This documentation describes the standard programming of the Rexroth IndraMotion MTX control. Apart from the basics of the NC programming, the usage of NC functions according to DIN 66025 as well as the NC functions with high-level language syntax and CPL functions are described.
Rexroth IndraMotion MTX xxVRS Standard NC Cycles DOK-MTX***-NC*CYC**V12-PRxx-EN-P, R911334375 This documentation describes the application of the standard cycles of the different technologies for Rexroth IndraMotion MTX control.
Rexroth IndraMotion MTX xxVRS Shop Floor Programming Turning and Milling DOK-MTX***-SF*PROG*V09-AWxx-EN-P, R911324377 This documentation describes the operation and programming of the graphical NC interface.
Rexroth IndraMotion MTX xxVRS Block Pre-Run DOK-MTX***-BLK*RUN*V12-APxx-EN-P, R911334379 This documentation explains to the machine manufacturer how to setup the "Block pre-run" function at the machine for the end user.

xx Respective version or edition
Fig. 1-4: MTX documentation overview - Operating

1.2.5 Maintenance

Documentation titles with type designation codes and parts numbers

Rexroth IndraMotion MTX xxVRS Diagnostic Messages DOK-MTX***-DIAGMES*V11-RExx-EN-P, R911332311 This documentation provides an overview on errors, warnings and messages within the Rexroth IndraMotion MTX control.

xx Respective version or edition
Fig. 1-5: MTX documentation overview - Maintenance

1.2.6 OEM Engineering

Documentation titles with type designation codes and parts numbers

Rexroth IndraMotion MTX xxVRS OPC Communication
DOK-MTX***-OPC*COM*V12-PRxx-EN-P, R911334385
This documentation describes the syntax and the structure of the items for the communication with Bosch Rexroth devices.
Rexroth IndraMotion MTX xxVRS Integration of OEM Applications
DOK-MTX***-DEV*KIT*V09-AWxx-EN-P, R911324355
This documentation is intended to assist the integration of OEM applications in the IndraWorks MTX.
Rexroth IndraMotion MTX xxVRS Automation Interface
DOK-MTX***-AUT*INT*V12-APxx-EN-P, R911334842
This documentation describes the script-based access on the IndraWorks project data via the interface of the Automation Interface. Different objects including code examples are described. The Automation Builder is described afterwards.

xx Respective version or edition

Fig.1-6: MTX documentation overview - OEM engineering

1.2.7 Add-Ons

Documentation titles with type designation codes and parts numbers

Rexroth IndraMotion MTX xxVRS Efficiency Work Bench MTX cta, MTX ega
DOK-MTX***-EWB*****V12-APxx-EN-P, R911333909
This documentation describes the mode of operation and the use cases of the analysis tool IndraMotion MTX cta and IndraMotion MTX ega.
Rexroth IndraMotion MTX xxVRS Action Recorder
DOK-MTX***-ACR*****V11-APxx-EN-P, R911329943
This documentation describes the MTX action recorder. It includes the installation and commissioning as well as interface signals, application and operation.
Rexroth IndraMotion MTX xxVRS RCM
DOK-MTX***-RCM*****V01-APxx-EN-P, R911334383
This documentation describes the operation of the Remote Condition Monitoring System.

xx Respective version or edition

Fig.1-7: MTX documentation overview - AddOns

1.3 Information Representation

1.3.1 Names and Abbreviations

Term	Explanation
IWE	IndraWorks Engineering
IWO	IndraWorks Operation
OWG	Optical waveguide
NC	Numerical Control
OEM	Original Equipment Manufacturer
PROFIBUS	Communication connection
SERCOS	Communication connection

Fig.1-8: Names and abbreviations used

2 Introduction

2.1 About this Documentation

2.1.1 Purpose

This documentation describes the commissioning of the IndraMotion MTX control. Apart from a complete overview, the commissioning and the configuration of the axes and the user interface as well as the PLC data are described.

This documentation is a component of the full documentation for the Rexroth IndraMotion MTX.

2.1.2 Structure

The document is structured according to the procedures. The description of the individual functions corresponds to their operation. General operating instructions come first.

2.1.3 Notation, Fonts, Symbols

The following notations are used in this documentation to actuate operating elements (buttons, keys):

Keys on the keyboard and on the operator panel

Keys on the keyboard and on the operator panel are shown in angled brackets <>, e.g. pressing the enter key by <Enter>.

Keys that are pressed **in sequence** are separated by a comma, e.g. <1>, <2>, <Enter>.

Keys that are pressed **simultaneously** are combined using a plus symbol, e.g. <Ctrl>+<Alt>+<Del> or <Shift>+<Enter>.

Buttons on the user interface

Buttons in the dialogs of user interface are highlighted by quotation marks, e.g. "OK" or "Cancel".

Menu Sequences

To display menus, the individual sequential menu items are highlighted in bold and connected by triangles. For example, to go to the menu item "Save" in the menu "File": **File** ▸ **Save**.

3 Important Instructions for Use

3.1 Appropriate Use

3.1.1 Introduction

Bosch Rexroth products represent state-of-the-art developments and manufacturing. They are tested prior to delivery to ensure operating safety and reliability.

The products may only be used in the manner that is defined as appropriate. If they are used in an inappropriate manner, then situations can develop that may lead to property damage or injury of personnel.

☞ Bosch Rexroth, as manufacturer, is not liable for any damages resulting from inappropriate use. In such cases, the guarantee and the right to payment of damages resulting from inappropriate use are forefeited. The user alone carries all responsibility of the risks.

Before using Bosch Rexroth products, make sure that all the pre-requisites for appropriate use of the products are satisfied:

- Personnel that in a way, shape or form uses our products must first read and understand the relevant safety instructions and be familiar with appropriate use.
- If the product takes the form of hardware, then they must remain in the original state, in other words, no structural changes are permitted. It its not permitted to decompile software products or alter source codes.
- Do not mount damaged or faulty products or use them in operation.
- Make sure that the products have been installed in the manner described in the relevant documentation.

3.1.2 Areas of Use and Application

The Rexroth IndraMotion MTX control is used to

- Programming contour and machining technology (feedrate, spindle speed, tool change) or a workpiece.
- Guiding a machining tool along a programmed path.

Feed drives, spindles and auxiliary axes of a machine tool are activated via SERCOS interface.

☞ This additionally requires I/O components for the integrated PLC which, in combination with the actual CNC, controls the machining process as a whole and also monitors this process with regard to technical safety.

The unit may be operated only with the explicitly specified hardware component configurations and combinations and only with the software and firmware specified in the appropriate documentations and functional descriptions.

The Rexroth IndraMotion MTX has been developed for control tasks in multi-axis installations.

Typical applications are:

- lathes
- milling machines

- machining centers

3.2 Inappropriate Use

Using the Rexroth IndraMotion MTX outside of the above-referenced areas of application or under operating conditions other than described in the document and the technical data specified is defined as "inappropriate use".

The Rexroth IndraMotion MTX may not be used if ...

- they are subject to operating conditions that do not meet the above specified ambient conditions. This includes, for example, operation under water, in the case of extreme temperature fluctuations or extreme maximum temperatures or if
- Bosch Rexroth has not specifically released Rexroth IndraMotion MTX for that intended purpose. Please note the specifications outlined in the general safety instructions!

4 General Overview

4.1 System Overview

4.1.1 General

The "Rexroth IndraMotion MTX" control system is based on operator panels with integrated PC. PC-based operator panels consist of a display, M-keys and a 3-slot box. To operate the machine, a keypad-coupled (VAM 40.1/VAM 10.1) PROFIBUS is connected to the operator panel (IndraControl BTV 40.2 or BTV 16.2). The MTX control component is a PC plug-in card (IndraControl CMP 60).

Furthermore, various small control panels can be connected to the control panel via PROFIBUS. Small control panels can be classified into text-oriented displays (VCP 02/VCP 05) and graphics-oriented displays (VCP 08/VCP 20/VCP 25).

The IndraMotion MTX control system supports handwheels and simple manual operator panels. Such devices are directly connected to the rear panel of the machine control panel using internal wiring. Data exchange occurs via defined PLC function blocks.

The drive technology is based on the IndraDrive M drive controller family with the corresponding supply modules, power and control units and IndraDyn motors. A wide range of different motors is available, starting with servo motors (IndraDyn S) and proceeding to linear and torque motors (IndraDyn L/T) up to squirrel-cage induction housing motors (IndraDyn A).

4.1.2 BTV 40.2 / BTV 16.2 PC-based Operator Panels

General

Robust BTV HMI devices with key-based designs and different screen sizes are available for a wide variety of requirements in the industrial environment.

The BTV 40.2 operator panel is characterized by a large 15" display with a resolution of 1024 x 786 pixels.

General Overview

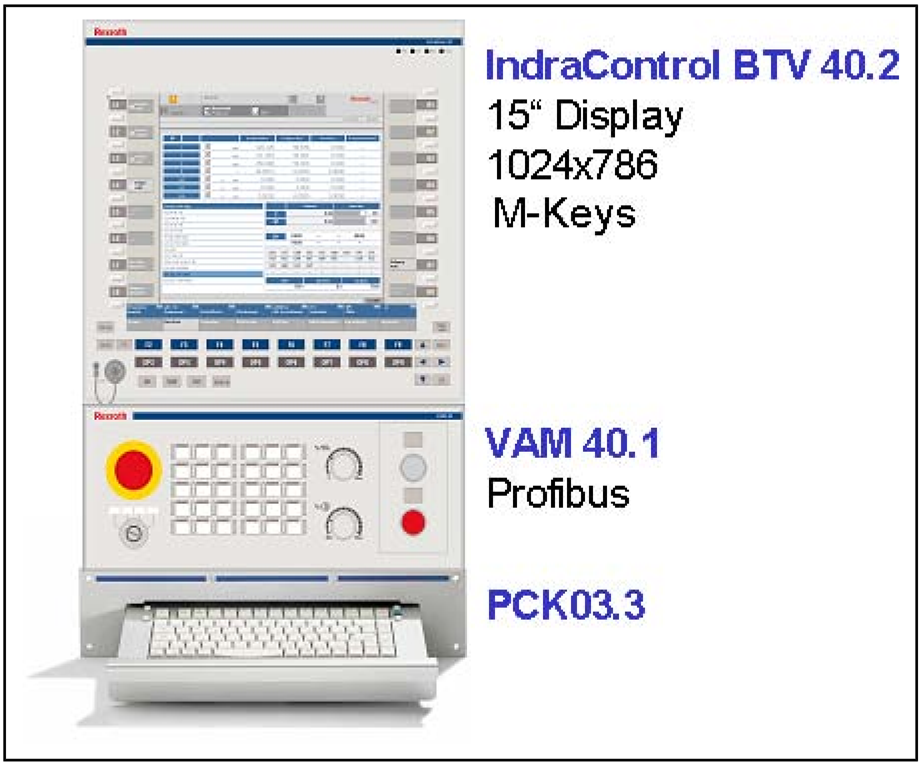


Fig.4-1: IndraControl BTV 40.2

The smaller version of the BTV 40.2 is the BTV 16.2 control panel.

The differences are:

- A smaller display (12")
- 800 x 600 pixel resolution

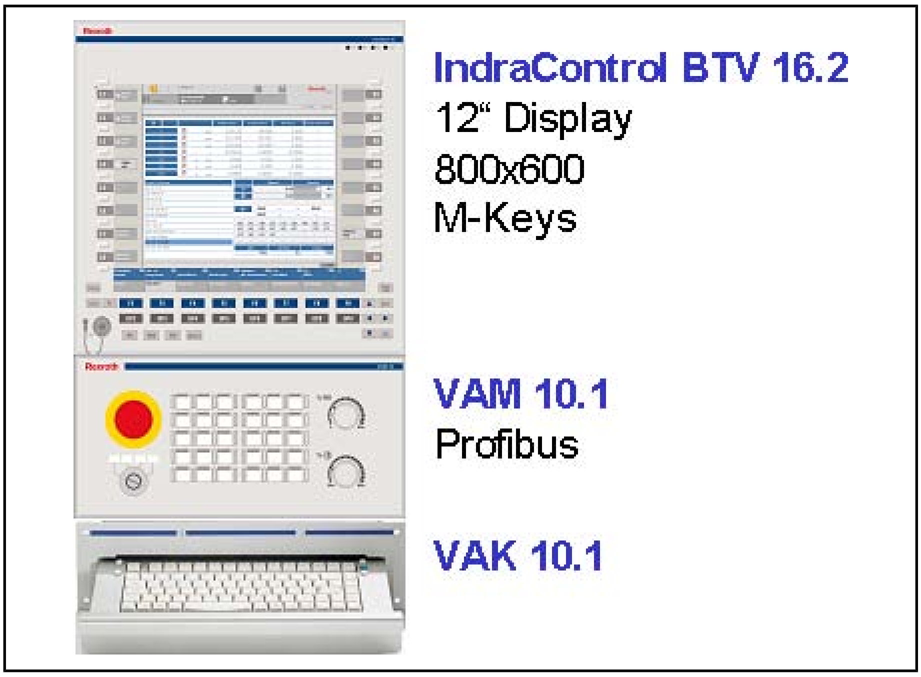


Fig.4-2: IndraControl BTV 16.2

IndraControl P60

The high-performance central module provides a CNC performance that allows activation of up to 64 axes in 12 independent CNC processing channels. The standard equipment includes interfaces allowing the activation of I/Os via PROFIBUS-DP, of intelligent drives via the SERCOS interface and of peripheral assemblies via Ethernet.

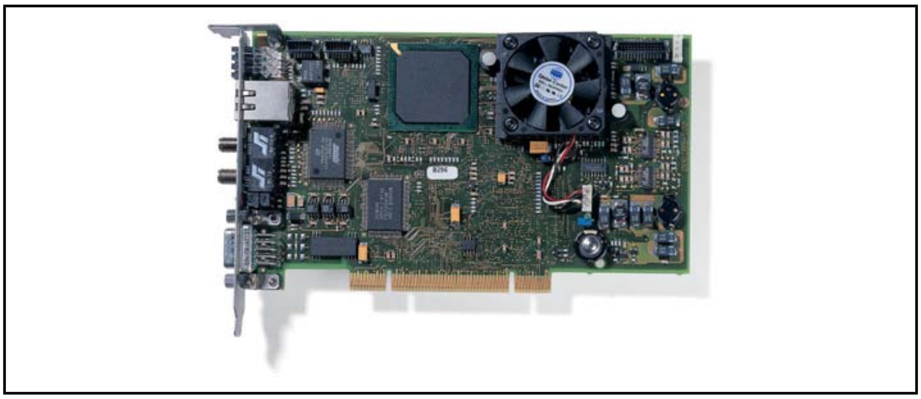


Fig.4-3: IndraControl P60

Function modules with a high-speed interface permit the module to be supplemented by additional fieldbuses, axes or interfaces.

IndraControl L40

This space-saving control module for the installation in a switch cabinet on a top-hat rail offers CNC functionality for the activation of up to 8 axes in 2 autonomous CNC processing channels. The standard equipment includes interfaces for the activation of intelligent drives via the SERCOS interface, the PROFIBUS DP master interface, the Ethernet connection (RJ45, 10/100 Base-T), the NC Ready contact, the RS232 interface and the Rexroth Inline interface.

Fig.4-4: CNC control module IndraControl L40

The control is equipped with 8 isolated digital inputs and 8 isolated digital outputs. The number of inputs and outputs can be further extended with Inline I/O modules.

The following table shows the performance data of the control module IndraControl L40.

Name	IndraControl L40
Number of axes	Max. 8
Thereof spindles	Max. 2
Number of interpolating axes	Max. 4
SERCOS cycle time	Min. 6 ms (for 8 configured axes, 4-axis interpolation)
Block cycle time	Min. 6 ms

Fig.4-5: Performance data, IndraControl L40

IndraControl VAM

The machine is operated with the keypad VAM 40.1 or VAM 10.1. These control panels are adapted to our operation and visualization devices and are connected to the control via PROFIBUS-DP.

Fig.4-6: Control panel for BTV40.2

As opposed to the VAM 10.1, the VAM 40.1 has a flexible module slot into which a machine on/off button can be used for example. Both machine control panels have the following technical data:

- PROFIBUS activation
- 30 illuminated pushbuttons (short-stroke keys) with foil cover and separating bars
- 24V DC power supply
- Emergency stop (1 make-contact / 1 break-contact)
- 4-step key switch
- 1x feed override
- 1x spindle override
- 16 24V DC digital inputs
- 8 24V DC/200 mA digital outputs
- Interface handwheel
- Digital interface for hand-held terminal

IndraControl VAK

These drawer keyboards with an integrated mouse have compact dimensions and are easy to install. When they are closed, they are protected from spraying water and soiling by protection class IP65.

Fig.4-7: PCK03.3 / VAK10.1 keyboard

4.1.3 Small Operator Panels

Small operator panels are connected to the control via PROFIBUS-DP interface. Communication is limited solely to accessing PLC variables via IndraLogic function blocks.

The operating screens are currently displayed using the Bosch Rexroth VI-Composer tool. No default functions are available.

There are two types of small operator panels:

- Text-oriented small operator panels
  Display only as text
- Graphics-oriented small operator panels
  Display of graphics also possible

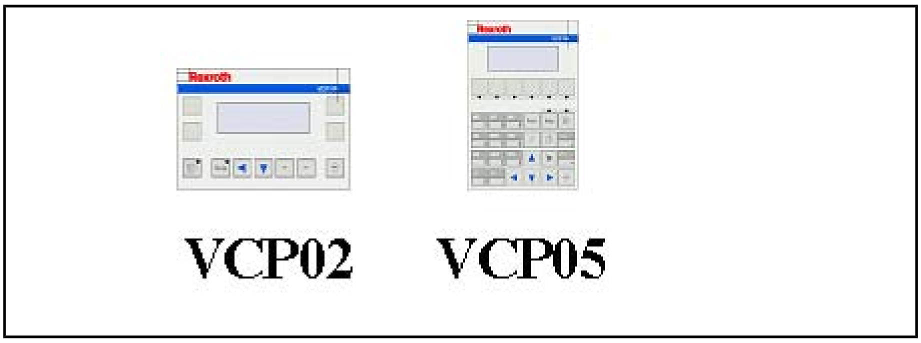


Fig.4-8: Small operator panels - text-oriented

Fig.4-9: Small operator panels - graphics-oriented

4.1.4 Hand-Held Terminals

A selection of hand-held terminals that can be used is shown below. The use is limited to Euchner hand-held terminals (maximum size: Euchner HBA-072910; no plain text display). Hand-held terminals are connected to the control using the rear panel connection on the VAM 10/40.

Fig.4-10: Hand-held terminals

4.1.5 Drive Technology "IndraDrive"

The "IndraDrive" drive modules consist of the following parts:

- IndraDrive - control unit (CSH)
- Inverter "IndraDrive M"
- MPH-03VRS / MPB-03VRS / MPD-03VRS firmware

Using the components listed above, every IndraDrive drive can be configured user-specifically and differs in performance and function. Together with various firmware combinations, it leaves nothing to be desired. Two drive modules are available for applications with the IndraMotion MTX.

- BASIC UNIVERSAL
  - Single-axis module
  - Double-axis module
- ADVANCED

The differences between the modules are shown in the following tables.

IndraDrive M	Current controller	Velocity controller	Position controller	Safety technology	No. of options
BASIC UNIVERSAL Single-axis module	125 µs	250 µs	500 µs	Possible	2 options
BASIC UNIVERSAL Double-axis module	125 µs	250 µs	500 µs	Possible	2 options
ADVANCED	62.5 µs	125 µs	250 µs	Possible	3 options

Fig.4-11: IndraDrive M overview

Various HMV01 power supply devices are available for the modular drive controller modules.

- HMV01.1E (E = without regeneration, 18 kW - 72 kW)
- HMV01.1R (R = with regeneration, 18 kW - 65 kW)

4.1.6 "IndraDyn" Motors

The following motor types are available for the IndraDrive M drive technology:

- IndraDyn S synchronous servo motors
  - MSK050 - MSK070
- IndraDyn L synchronous linear motors
  - MLF040 - MLF300
- IndraDyn T synchronous torque motors
  - MBT210 - MBT450
- IndraDyn A asynchronous housing motors
  - MAD100 - MAD180 (surface ventilation)
  - MAF100 - MAF180 (liquid-cooled)

4.2 IndraWorks - Engineering Desktop

4.2.1 General

The Engineering desktop comprises the tools for commissioning and troubleshooting. In addition, the system-comprehensive project management and the central data storage make it easy to clearly visualize user data.

The commissioning functions for parameterization and optimization of digital Bosch Rexroth drives are started from Engineering desktop. The project management is used to allocate and store the drive data according to the system used.

The Engineering desktop is the central desktop to

- create, restore and save projects
- edit configuration parameters
- edit the parameters of the connected Bosch Rexroth drives
- edit the PLC project
- edit the HMI project
- etc.

The Engineering desktop can be started using the "IndraWorks Engineering" icon or, with the Start menu, "Program Files/Rexroth/IndraWorks/Engineering".

☞ You can find more detailed descriptions of the Engineering desktop in the manual "Rexroth IndraWorks Engineering", in particular the description of the main menu.

4.2.2 IndraWorks Project

From the MTX point of view, an IndraWorks project is divided into two main areas in the Project Explorer:

- BTV40 / BTV16
- IndraMotion MTX

In turn, the two main areas are divided into several areas (nodes). This manual describes the special features of an IndraMotion MTX project.

☞ Creating a project is described in detail in the manual "Rexroth IndraWorks Engineering". Therefore, only special features in a project for the IndraMotion MTX are described.

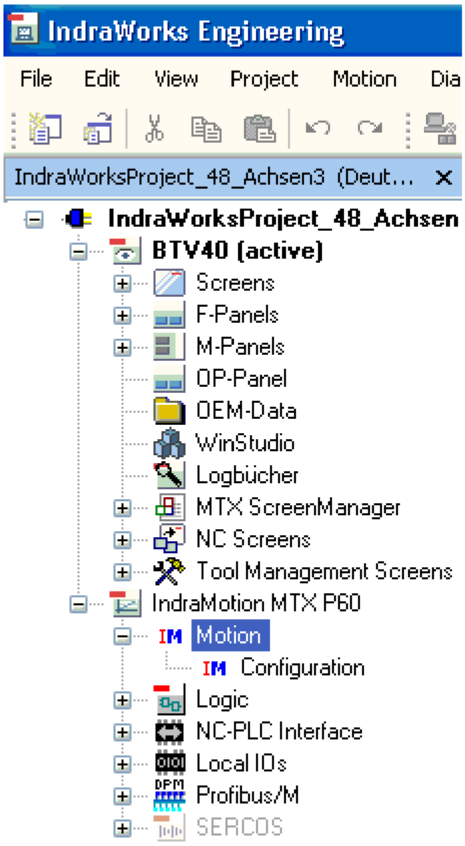


Fig.4-12: IndraWorks Engineering (Project Explorer of an IndraWorks project)

Before working with the IndraWorks project, it must first be created or an existing saved project must be imported (restored).

4.2.3 Creating a Project

General

A project can be created in the main menu under "File/New/Project". A "Visualization" device and a "Drive and Control" device must be configured in the Project Explorer.

Therefore, drag the desired visualization device from the library into the Project Explorer. The figure below shows the selection. Changing to another visualization hardware later on is only possible with some effort. A "BTV40" was created in the above-mentioned example (see fig. 4-12 "IndraWorks Engineering (Project Explorer of an IndraWorks project)" on page 34).

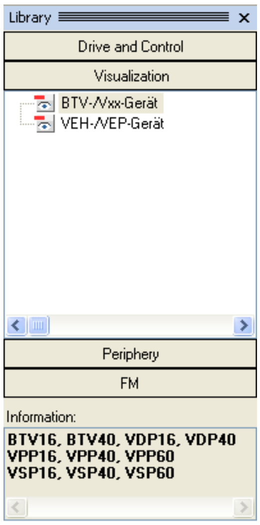


Fig.4-13: Library (visualization)

When the HMI device is configured, the corresponding nodes are created under the main "BTV40" node.

After the visualization device has been configured, the "IndraMotion MTX" device (e.g IndraMotion MTX P60) still has to be configured. As it is the case for the visualization device, this device is moved from the corresponding library to the project node via "drag&drop".

A project is completely described with these devices.

General Overview

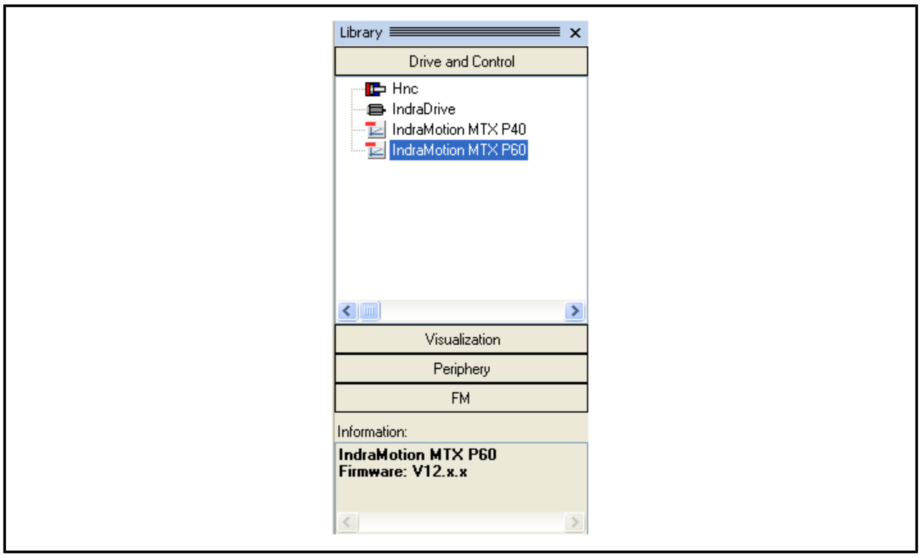


Fig.4-14: Libraries (visualization, Drive and Control)

Handling Instruction: Creating a Project in IW-Engineering

This handling instruction describes how a user creates a project in the IndraWorks Engineering desktop.

IW-Engineering / File: Create IndraWorks project

1. Start Engineering desktop.

   After the Engineering desktop has been started, the following screen appears. A new project is created by pressing the "Create new project" button.

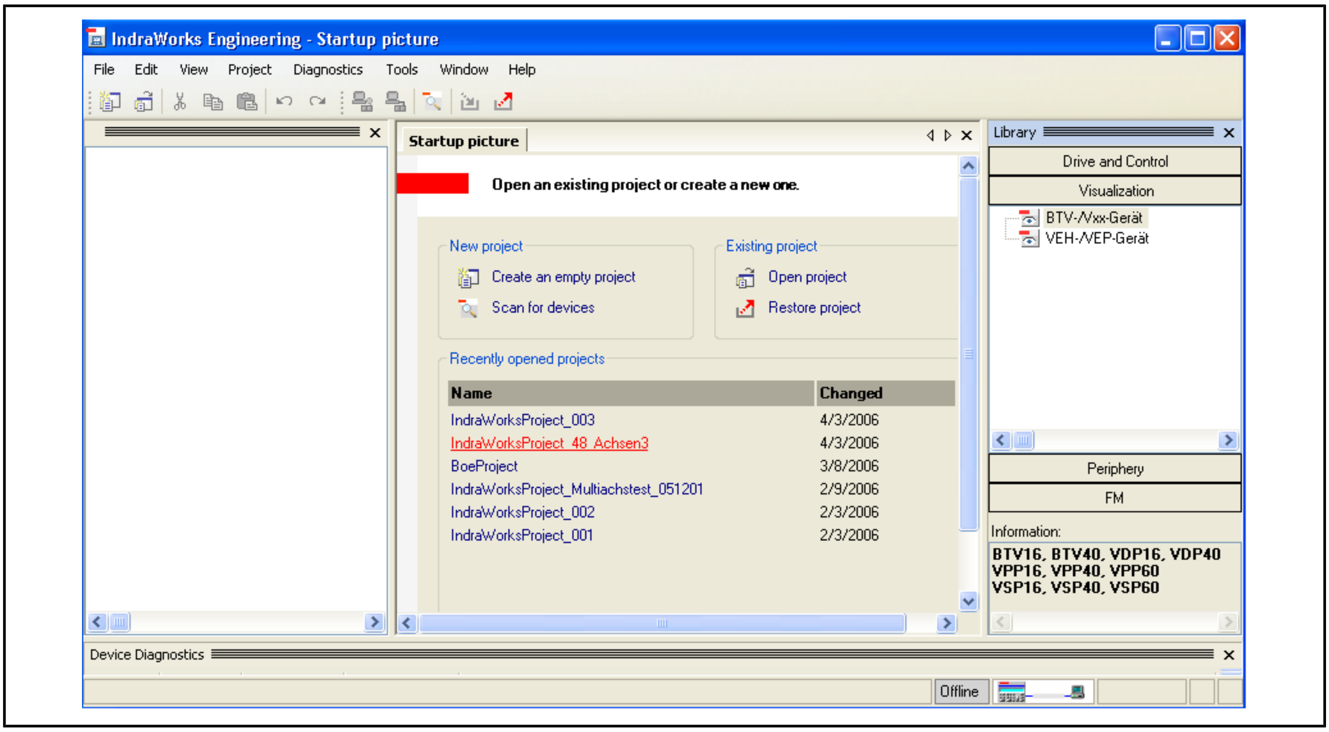


Fig.4-15: IndraWorks Engineering - Start screen (Create project)

2. Adapt the subsequent dialog according to the requirements of the project storage.

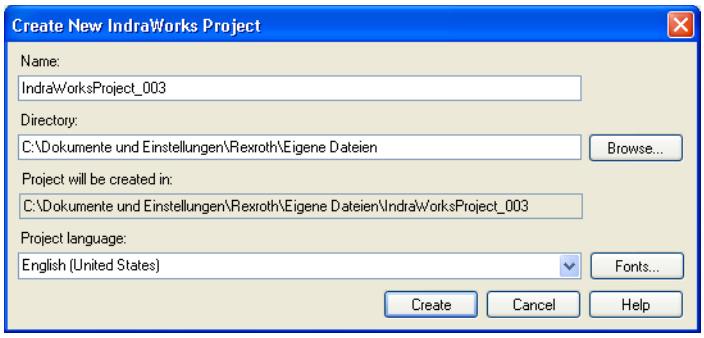


Fig.4-16: Dialog: Create New IndraWorks Project

☞ In certain cases (checkbox activated, etc.), the initial screen is not displayed; in such cases, a new project can be created in the main menu under **File** ▸ **Project** ▸ **New**.

		Documentation
Documentation:	IndraWorks Engineering	Working with IndraWorks

IW-Engineering / File: Creating and Configuring "IndraMotion MTX" Node

Drag the device "IndraMotion MTX xxx" into the project with the left mouse button.

General Overview

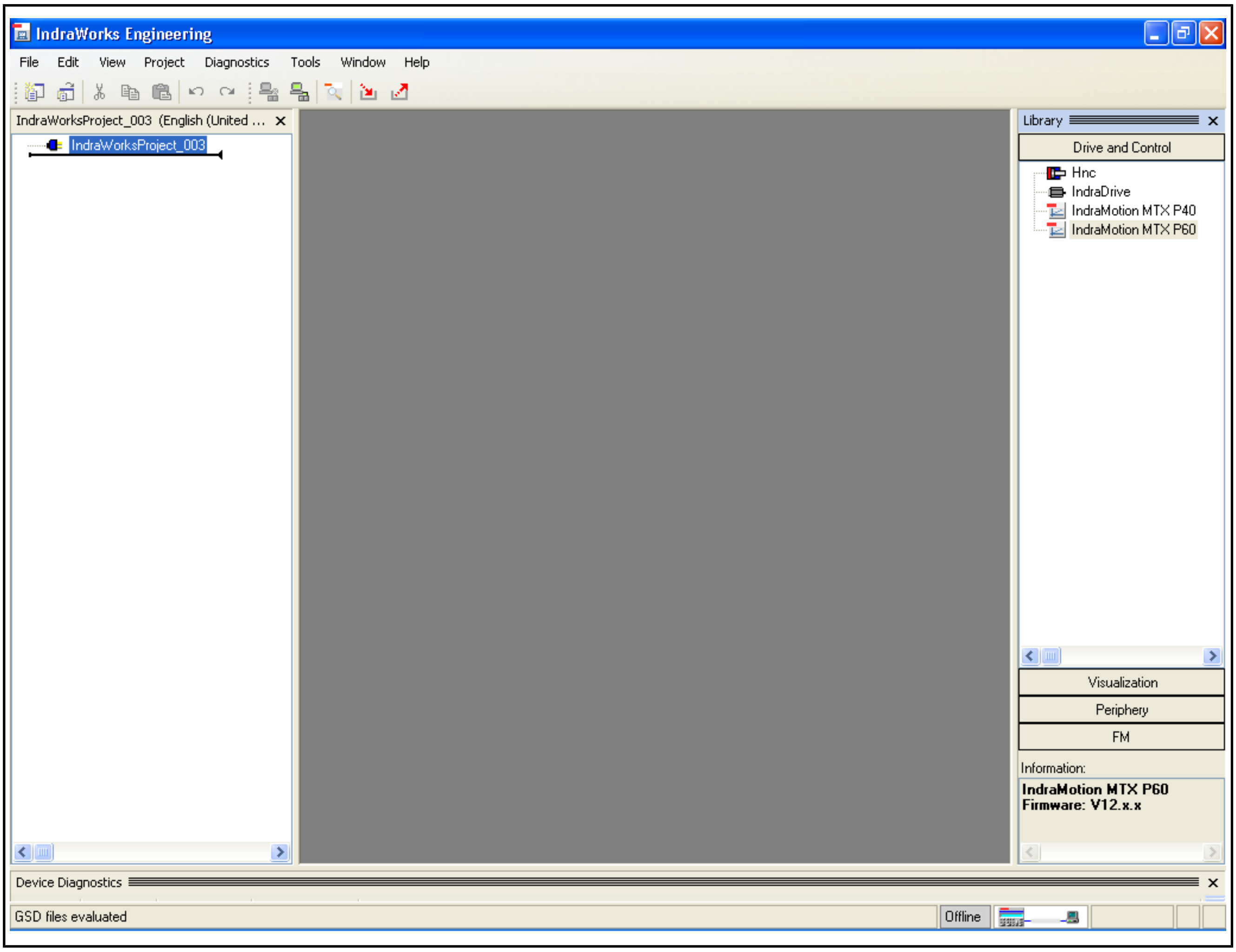


Fig.4-17: IndraWorks Engineering (Example: Configuring MTX P60 control hardware)

☞ The settings of the following wizard can remain. These settings have been made for a default setup.

		Documentation chapter 4.2.3 "Creating a Project" on page 35
Documentation:	IndraWorks Commissioning	General Overview

IW-Engineering / File: Creating and configuring a "BTV/Vxx device" node

1. Drag the "BTV / Vxx" device into the project with the left mouse button.

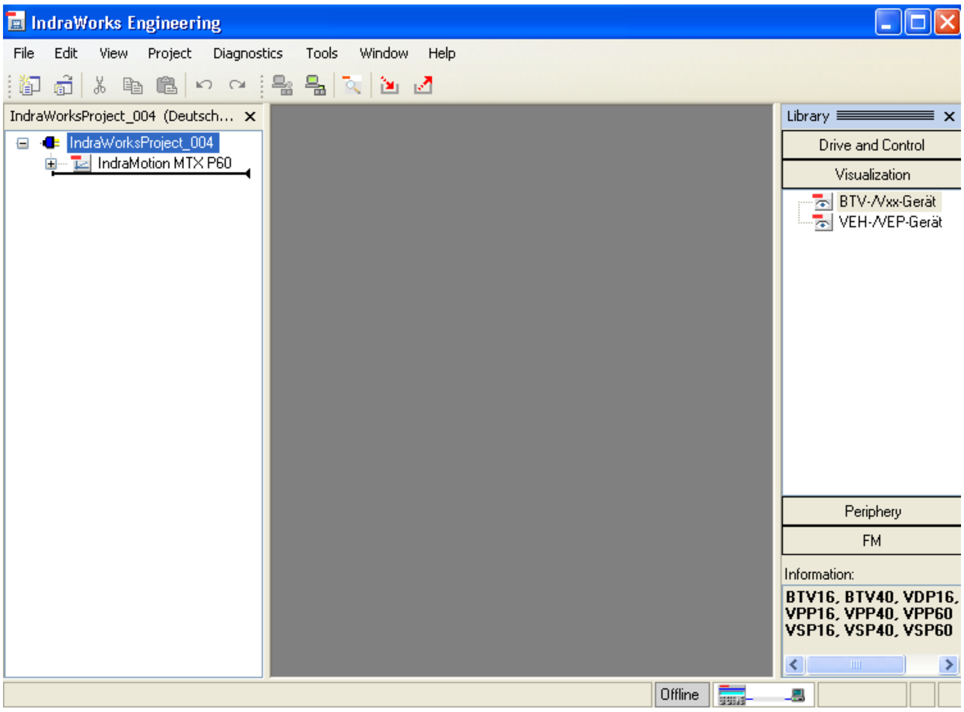


Fig.4-18: IndraWorks Engineering (configure visualization)

2. Change the following settings in the wizard or adapt them to the application.
   - Device type
   - Application type
   - Panel design

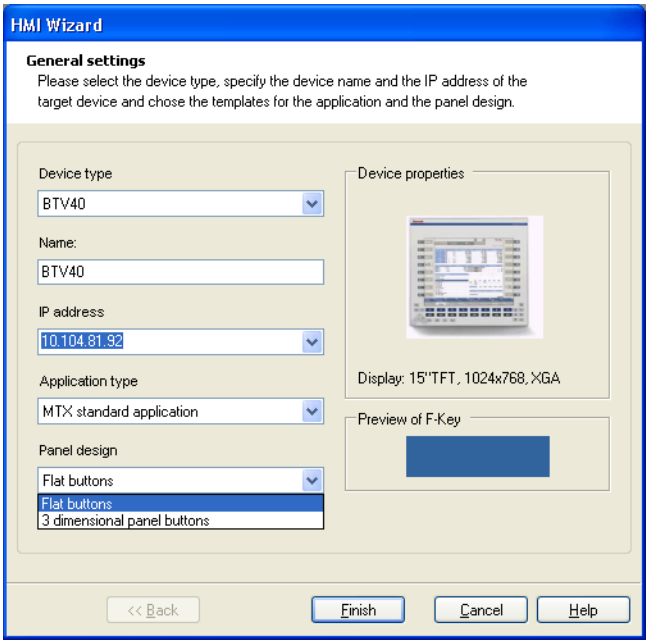


Fig.4-19: Dialog: HMI Wizard (configure visualization)

3. After the configuration has been completed, make the following setting. To open this dialog, click the "BTV / Vxx device" with the right mouse

button. The setting has to be changed to the previously created control ("IndraMotion MTX P60").

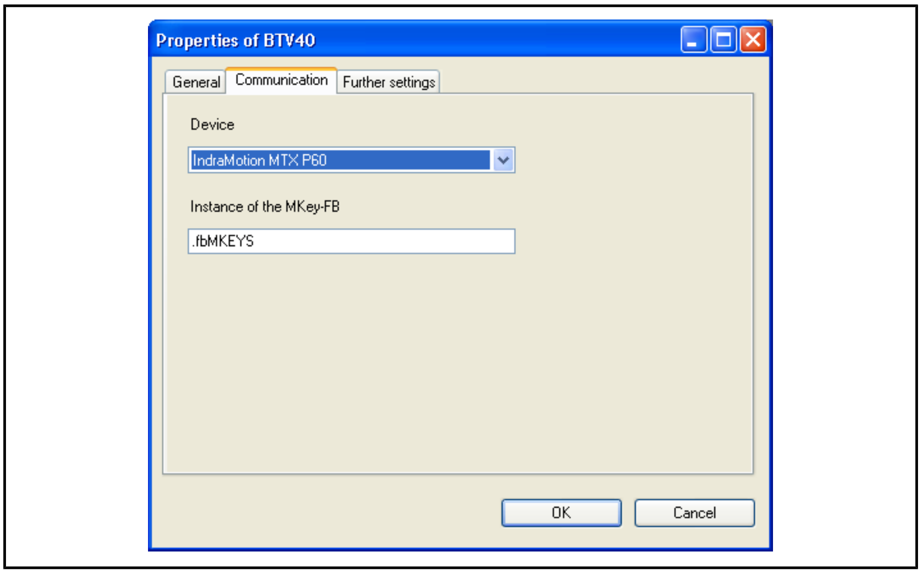


Fig.4-20: Dialog: Properties of BTV40

		Documentation
Documentation:	IndraWorks Engineering	Working with IndraWorks

4.2.4 The IndraMotion MTX Wizard

General

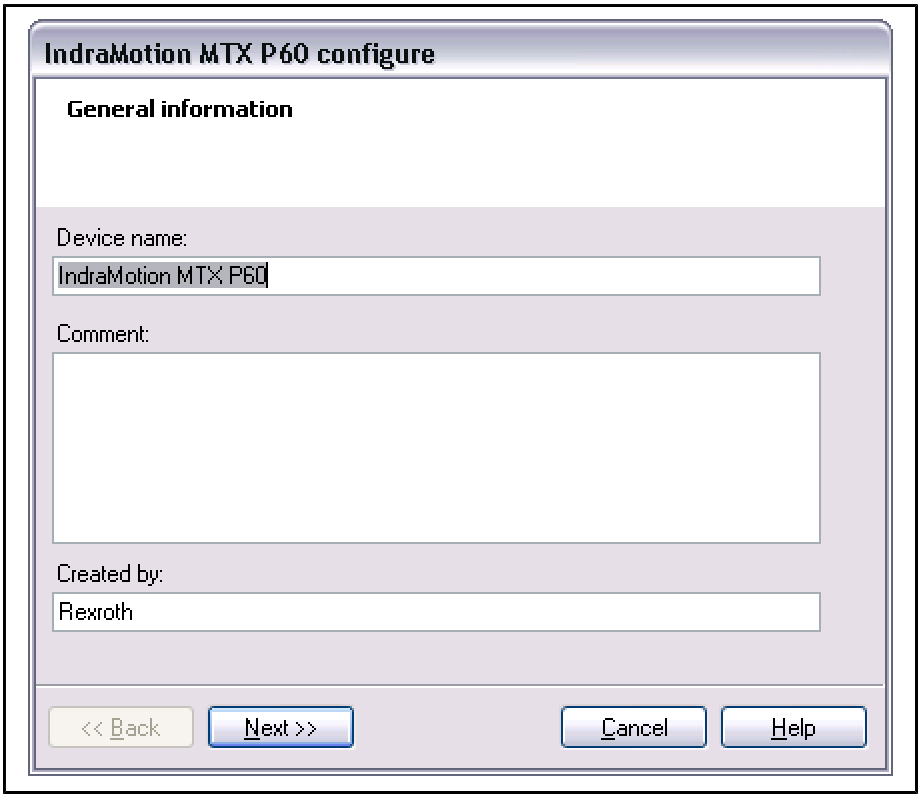


Fig.4-21: Dialog: Configure IndraMotion MTX P60 (General information)

The following general device data is entered here:

- Device name:

  Name of the device displayed in the Project Explorer. By default, the name is preassigned with the device type.

  The name must comply with the IEC 61131-3 standard.
- Comment:

  Comment on the device. This is shown as tool tip in the Project Explorer on the device.
- Created by:

  Creator of the device. The entry is set to the currently logged-on Windows user by default.

Communication Settings

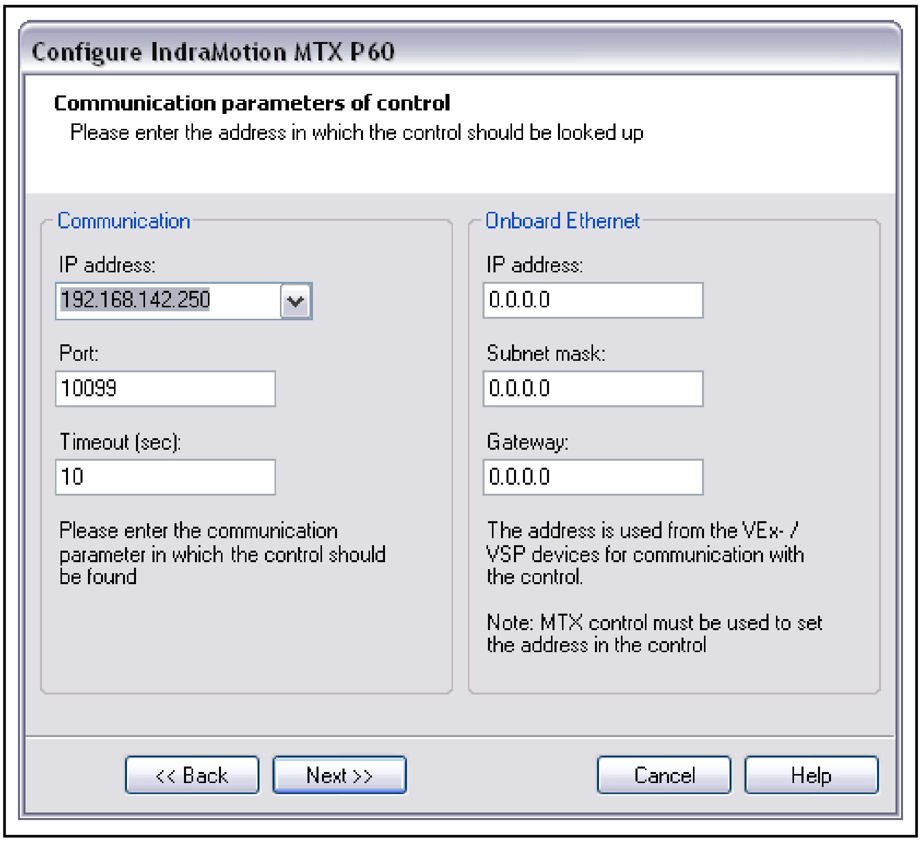


Fig.4-22: Dialog: Configure IndraMotion MTX P60 (Communication parameters of the control)

The communication settings for the control are set here. Normally, no changes need to be made.

- **Left side: Communication**
  - **IP address:**

    IP address under which the IndraMotion MTX can be reached. The correct address is preset for the PC-based variant (IndraControl P). For the other variants, the address has to be adjusted to the control.
  - **Port:**

    Port on which the control can be reached.
  - **Timeout:**

Time in which the communication requests must be answered to avoid an error being generated.

- **Right side: Onboard Ethernet (only for PC-based controls/IndraControl P, e.g. MTX standard, MTX performance)**

  The address assigned is used by embedded devices for communication purposes (e.g. VEx and VSP). The change is not applied to the control. In order to change the address in the control, the MTX-Control has to be used.

Configuration

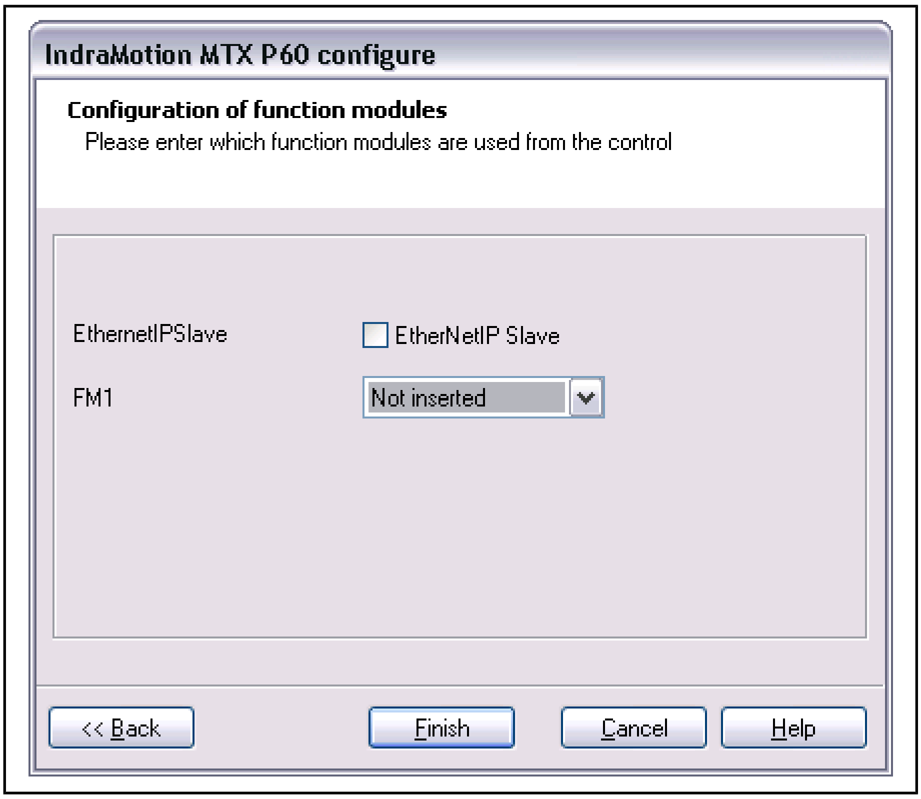


Fig.4-23: Dialog: Configure MTX P60 (configuration of function modules)

Optional device properties or function modules can be selected.

Example:

- EtherNetIP slave present
- Function module DeviceNetMaster plugged in

For the IndraMotion MTX L40, the SRAM memory extension has been preset on the first slot.

4.2.5 Project Node "BTV40"

The "BTV40" node is divided into several subnodes described in detail in the following section.

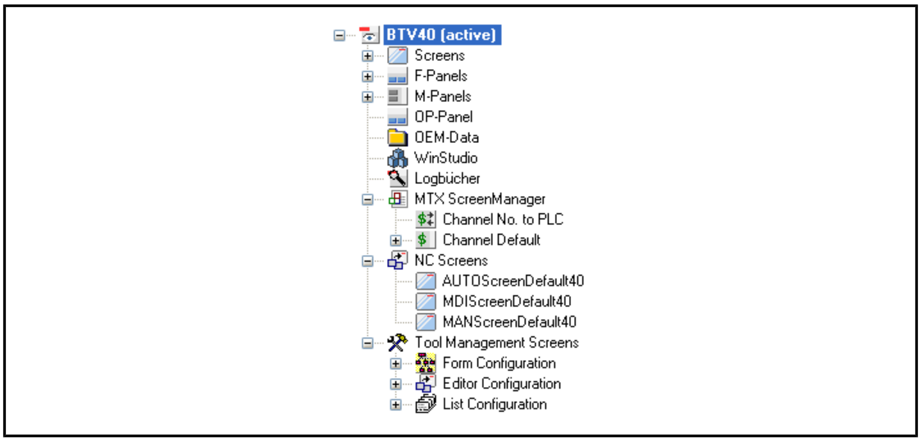


Fig.4-24: Project node - "BTV40"

- **Screens**

  The user-defined screens created with the configuration tool "WinStudio" are saved here. The user-defined screens are the special features of the machine.

- **F-panels/M-panels/OP-panels**

  The mentioned nodes describe the key panels created by the user. These key panels are used mainly in the user-defined screens.

- **OEM data**

  This directory is used to store data that is required by the OEM or by the end customer.

- **WinStudio**

  The program for creating user-defined screens can be opened by clicking with the right mouse button or by double-clicking this node.

- **Logbooks**

  The logbooks configured in the IndraMotion MTX system are listed here. At this point, only the name can be changed or the logbook can be deleted. The list serves only as an overview of the configured logbooks.

- **MTX ScreenManager**

  Configuration settings related to the data transfer from the interface and the PLC are specified in this node for the following areas:

  – Variables for transferring the channel number to the PLC and vice-versa.

  – Definition of variables for events, screens, F-keys and M-keys for the main operation modes "automatic", "MDI" and "manual".

- **NC screens**

  Definition of the NC screens for the main operation modes "automatic", "MDI" and "manual". The layout of these screens is specified by the IndraMotion MTX control system and can be adapted by the user.

- **ToolmanScreens (TlManScreen)**

  The node is used to configure the tool lists of the control. The subnode "ULC Configuration" is used to create the tool lists and editors. The node "Editor Screens" is used to configure the editor for online data (existing data is modified) or offline data (new data is created). The tool list can be modified by selecting the node "ToolList configuration". The tool list must have been configured before.

4.2.6 Project Node "IndraMotion MTX"

MTX Control with IndraLogic 1x

The "IndraMotion MTX" node in the Project Explorer describes functions regarding the control hardware (e.g. IndraControl P40/P60/P70).

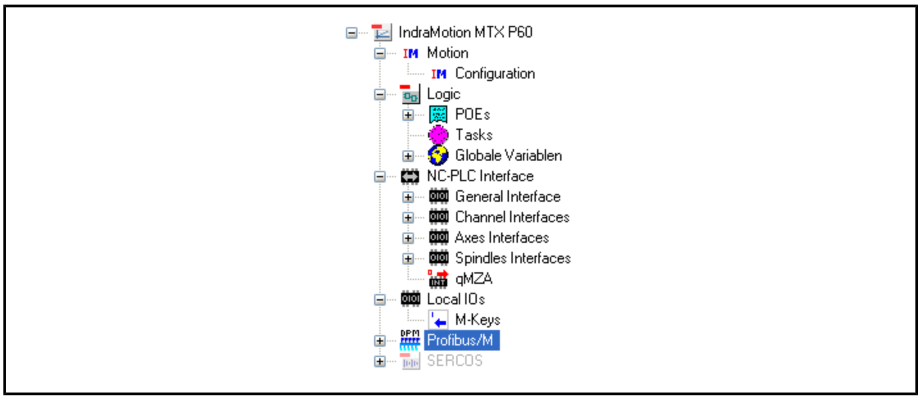


Fig.4-25: Project nodes - "IndraMotion MTX P60 / Profibus M"

- **IndraMotion MTX**

  There are different device functions at this node. These can be selected according to the communication state (online/offline).

  The following functions are available in the menu of the node:

  – Going online

  – Start offline parameterization

  The MTX is like in offline state, only the drives can be parameterized in offline state.

  Offline Parameterization page 98

  – Connection test

  – MTX System Status

  MTX System Status, page 59

  – NC restart with specification of the Startup mode (only online).

  – Set time (only online)

  – Firmware download (see note below)

  – Loading diagnostic texts to the control (only online)

  – Configuration of the Ethernet interface (only IndraControl L, only online)

  Configuration of the Ethernet Interface, page 50

  – Configuration of the mount directories (only online).

  Configuration of the Mount Directories, page 57

  – Archiving and restoring control data (only online).

  Data such as control parameters, NC programs and drive data can be saved.

  Archiving and Restoring Control Data, page 267

  – Export/Import

  Export/Import, page 265

  – Settings of the IndraLogic diagnostics.

  – Activating the I/O monitor (only online).

- Importing and exporting PLC project data.
- Restoring I/O data consistency.

- **Motion**

  This node represents the functions of the control hardware. Click the right mouse button to activate the following functions:

  - Create an icon file (only online):

    Creates an icon file for usage in WinStudio.

  - Saving configuration data Permanently (only online):

    The machine parameters are stored in the "userfep" directory of the control so that they can, for example, be automatically restored after a firmware download.

- **Motion - NC Configuration**

  This node represents the machine parameters of the control. If changes of the parameters are not yet active, this is visualized by a warning symbol on the node.

  The following functions can be called at this node:

  - Starting the configuration function for editing the machine parameters (only online).
  - Applying the changes depending on the parameters changed (NC restart, system reset).

- **Motion - NC file system**

  This node represents the file system of the control. The navigator opens with a double-click and allows a look at the files on the control and to edit them.

- **Logic**

  The node "Logic" represents the integrated PLC function "IndraLogic". Clicking with the right mouse button opens a menu used to execute the following functions:

  - Start of the programming system "IndraLogic".
  - Synchronization of the project data between IndraLogic and the Engineering desktop.
  - Printing the PLC project.
  - Setting the IndraLogic properties (e.g. communication settings, project settings, information about the target system, IndraLogic directories).
  - List of all POUs (**P**rogram **O**rganization **U**nit) of a project.
  - Task configuration under the node "Task".
  - Access to global variables.
  - Starting and stopping the PLC.

- **NC-PLC interface**

  Here is an overview on all processes in the project. The configured channels, axes and spindles are displayed so that address settings can be made in the corresponding views.

- **Local I/Os (only IndraControl P)**

  Configuration of the local inputs and outputs. The inputs and outputs are implemented on an add-on card for the IndraControl P40/60/70 control hardware.

- **Onboard I/O (IndraControl L only)**

  Configuration of inputs and outputs directly connected to the control.
- **Inline I/O (IndraControl L only)**

  Configuration of the participants in the inline bus of the control. The participants are configured and added.
- **PROFIBUS/M**

  The individual PROFIBUS participants are listed under this node. In addition, the individual participants can be modified and new ones can be added.
- **SERCOS**

  The node "SERCOS" opens the "gate" to the drive. "SERCOS parameters" drive parameters can be edited under this node. A data backup of the drive parameters can be executed. Furthermore, phase switching is possible. The commissioning tool "IndraWorks Drive" is activated to carry out these services.

  For more information, refer to: Axis Commissioning, page 69.

The documentation "Rexroth IndraWorks Engineering" contains a general description of the Engineering Desktop. This manual also describes:

- New project / Open project / Properties.
- Create / Add a device.
- Firmware download.

MTX Control with IndraLogic 2G

The "IndraMotion MTX" node in the Project Explorer describes functions regarding the control hardware (e.g. IndraControl L45/L65/L85).

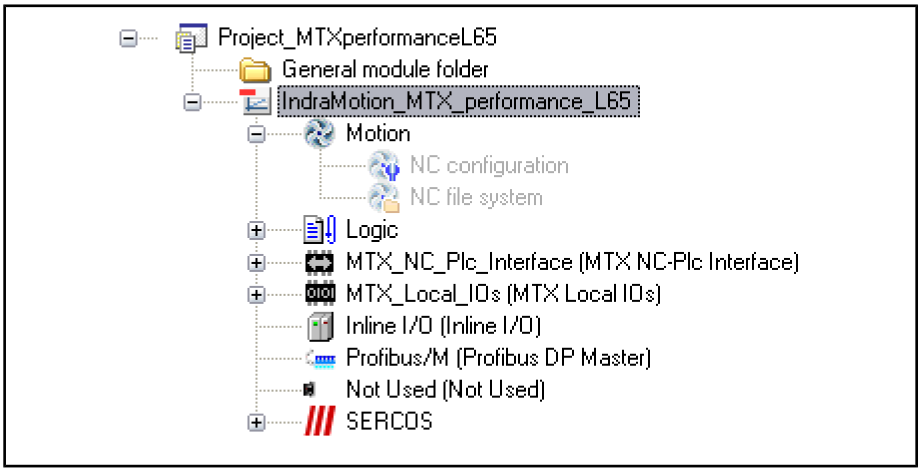


Fig.4-26: Project node of the IndraMotion MTX performance L65

- **IndraMotion MTX**

  There are different device functions at this node. These can be selected according to the communication state (online/offline).

  The following functions are available in the menu of the node:

  – Going online.

  – Starting offline parameterization.

The MTX is like in offline state, only the drives can be parameterized in offline state.

Parameterizing Drives, page 98

- Connection test
- MTX System Status

MTX System Status, page 59

- NC restart with specification of the Startup mode (only online).
- Set time (only online)
- Firmware download (see note below)
- Loading diagnostic texts to the control (only online).
- Configuration of the Ethernet interface (only online)

Configuration of the Ethernet Interface, page 50

- Configuration of the mount directories (only online).

Configuration of the Mount Directories, page 57

- Archiving and restoring control data (only online).

  Data such as control parameters, NC programs and drive data can be saved.

Archiving and Restoring Control Data, page 267

- Export/Import

Export/Import, page 265

- Settings of the IndraLogic diagnostics
- Activating the I/O monitor (only online)
- Importing and exporting PLC project data
- Restoring I/O data consistency

- **Motion**

  This node represents the functions of the control hardware. Click the right mouse button to activate the following functions:

  - Create an icon file (only online):

    Creates an icon file for usage in WinStudio.

  - Saving configuration data Permanently (only online):

    The machine parameters are stored in the "userfep" directory of the control so that they can, for example, be automatically restored after a firmware download.

- **Motion - NC Configuration**

  This node represents the machine parameters of the control. If changes of the parameters are not yet active, this is visualized by a warning symbol on the node.

  The following functions can be called at the node:

  - Starting the configuration function for editing the machine parameters (only online).
  - Applying the changes depending on the parameters changed (NC restart, system reset).

- **Motion - NC file system**

  This node represents the file system of the control. The navigator opens with a double-click and allows a look at the files on the control and to edit them.

- **Motion - NC Axes**

  This node represents the axes of the NC system. The axes can be created, deleted or edited.
- **Logic**

  The node "Logic" represents the integrated PLC function "IndraLogic". The programming system is reached from here.
- **NC-PLC interface**

  Here is an overview on all processes in the project. The configured channels, axes and spindles are displayed to make address settings in the corresponding views.
- **Local I/Os**

  Configuration of the inputs and outputs installed on the control hardware.
- **Inline I/O**

  Configuration of the participants in the inline bus of the control. The participants can be configured and added here.
- **PROFIBUS/M**

  The individual Profibus participants are listed under this node. In addition, the individual participants can be modified and new ones can be added.
- **SERCOS**

  The node "SERCOS" opens the "gate" to the drive. The drive parameters can edit the "SERCOS parameters" under this node. A data backup of the drive parameters can be executed. In addition, the phase can be switched. The commissioning tool "IndraWorks Drive" is activated to carry out these services.

For more information, refer to: Axis Commissioning, page 69.

☞ The documentation "Rexroth IndraWorks Engineering" contains a general description of the Engineering Desktop. This manual also describes:

- New project / Open project / Properties.
- Create / Add a device.
- Firmware download.

4.2.7 Configuring the MTX

Properties of the MTX

General

The general data for the device can be changed in the context menu of the control under "Properties":

- Device name:

  Name of the device displayed in the Project Explorer. The name must comply with the IEC 61131-3 standard.
- Comment:

  Comment on the device. This is shown as tool tip in the Project Explorer on the device.
- Created by:

  Author or editor of the device.

Communication Settings

Communication settings for the control can be carried out in the context menu of the control under "Properties".

- **Left side: Communication**
  - **IP address:**

    IP address under which the control can be reached.
  - **Port:**

    Port on which the control can be reached.

    Default value: 10099).
  - **Timeout (s):**

    Time in which the communication requests must be answered to avoid an error being generated.

    Default value: 10s for the variants "performance" and "advanced", 30s for the variants "compact" and "standard".
  - **Configuration of the control Ethernet interface (only IndraControl L, online):**

    Opening the Ethernet configuration under: Ethernet Configuration, page 50.

    After changing the IP address, it is immediately applied to the project.
- **Right side: Onboard Ethernet (only IndraControl P)**

  The address assigned is used by embedded devices for communication purposes (e.g. VEx and VSP). The change is not applied to the control. In order to change the address in the control, the MTX-Control has to be used.

  Data can be loaded from the control with the **Update** button.

Device Data

This view is only available for devices with "SERCOS III" support. Additionally, there must be an online connection.

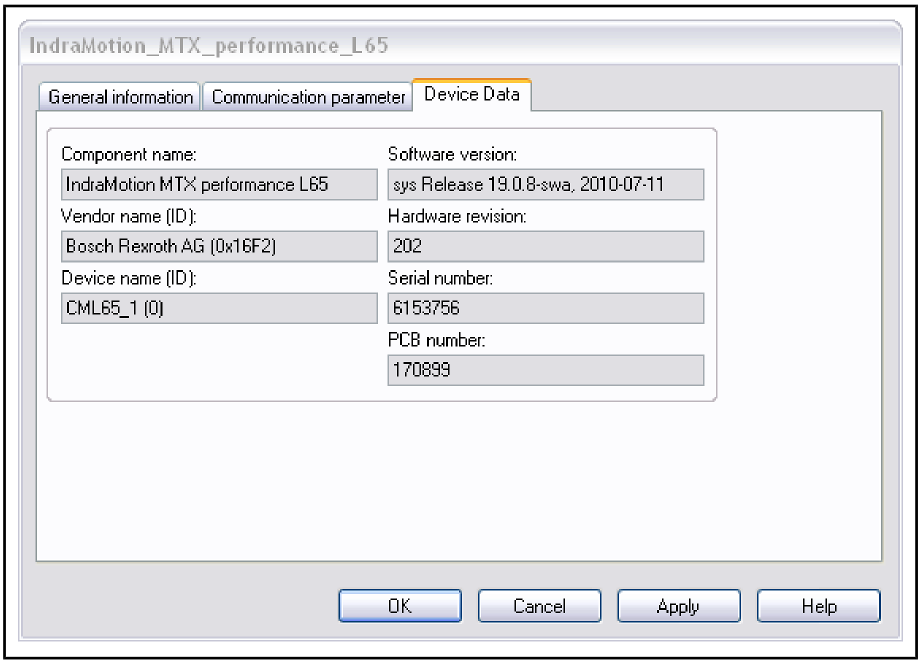


Fig.4-27: View of device data in the MTX properties

The information of the "Electronic Label" of the control can be seen. This information is used to uniquely identify the device.

Configuration of Ethernet Interface

This function is only available for MTX variants for control cabinet mounting (IndraControl L)

It is either opened via the context menu or in the "Properties" dialog of the MTX on the "Communication Settings" tab.

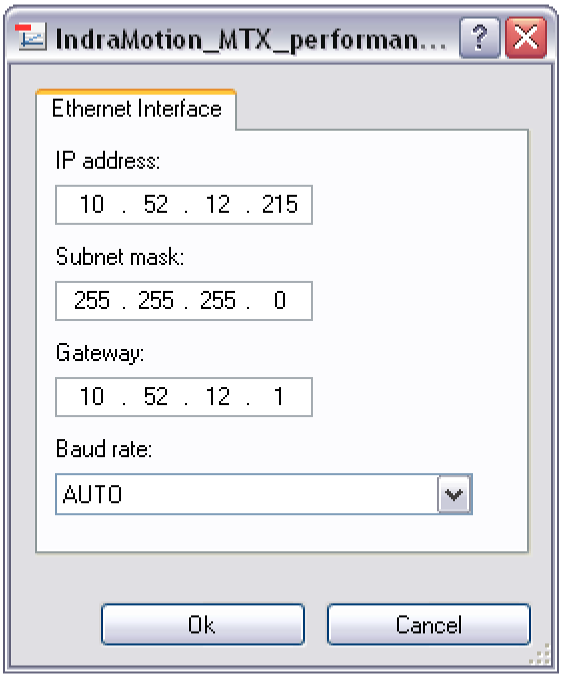


Fig.4-28: Configuration of the Ethernet interface of the MTX control

The parameters of the network interface of the control can be adjusted.

- IP address:

  IP address with which the control reports in the network or under which it can be reached.
- Gateway:

  The control establishes a connection to the network via the gateway. The gateway must be in the subnet of the control.
- Baud rate:

  The baud rate with which the network interface works. The default setting is "AUTO" and should only be changed if there are good reasons. In this setting, the best configuration is automatically determined.

☞ Changes of the Ethernet interface become effective only after a control restart. When the IP address changes, the setting in the project is additionally adjusted. After a successful control startup, the project can be immediately reconnected.

4.3 IndraWorks Operation Desktop

4.3.1 General

The Operation desktop is the main tool when working with the "IndraMotion MTX" control system. The status displays (position, velocity, override, etc.) of the drives are summarized in the NC screen.

The Operation desktop comprises the machine operation keys (M-keys), the function keys (F-keys) and the OP-keys to switch the individual operating functions (tool management, programming screen, NC screen, channel overview, etc.). The machine operation keys affect the PLC/interface so that PLC functions or switching functions of the interface can be triggered. "WinStudio" is provided to specify user-defined screens.

With machine parameters, the view of the NC screen can be configured user-specifically.

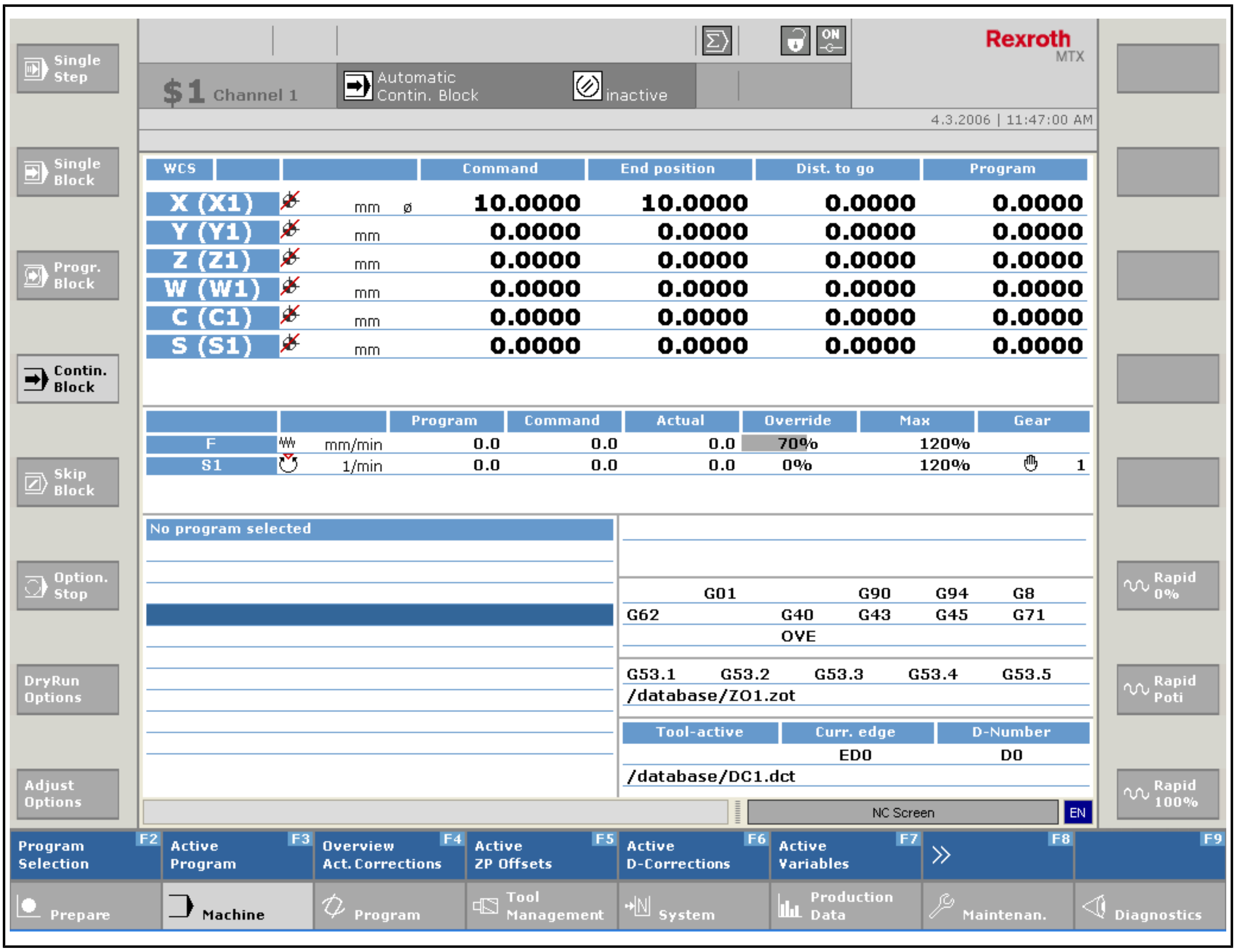


Fig.4-29: Example of the Operation desktop

4.3.2 Starting Operation Desktop

In the IndraWorks installation directory, IndraWorks Operation Desktop can be started by double-clicking "DDP.Panelservice.exe". It can also be started using the "IndraWorks Operation" icon .

If no active project exists, the following dialog appears:

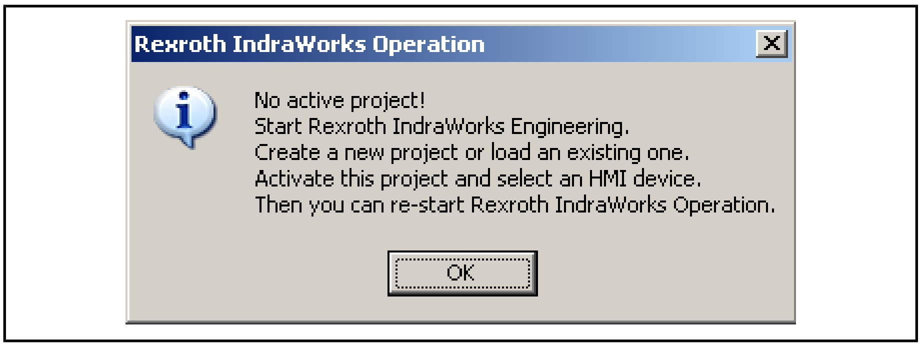


Fig.4-30: Dialog: Rexroth IndraWorks Operation (No active project)

First, start "Rexroth IndraWorks Engineering", create a new project or load an existing one and activate said project. Then restart Operation Desktop.

After Operation Desktop starts up without errors, the screen shown above is displayed.

4.3.3 Operation Keys (OP Keys)

General

Operation Desktop is equipped with 8 keys to control various functions. The OP key assignment is identical for all basic projects.

OP Key "OP1" - Help

Pressing the OP key **OP1** calls the Help function for the "IndraMotion - MTX" system.

OP Key "OP2 - Prepare"

The OP key **OP2 - Prepare** displays the user-defined user screens. These screens can be equipped with user-defined M-keys and F-keys. User-defined screens are created using "WinStudio".

OP Key "OP3 - Machine"

The OP key **OP3 - Machine** is the main operation screen for operating the machine. The axis data (position, velocity, override, etc.) are displayed. In addition, the main operation modes (automatic, MDI, manual) can be activated here.

OP Key "OP4 - Program"

Pressing OP key **OP4 - Program** opens various editors:

- NC programs
- Variable lists
- D-correction tables
- Zero point tables
- Placement tables

The editors can be used to create new programs/tables/lists or to edit existing ones.

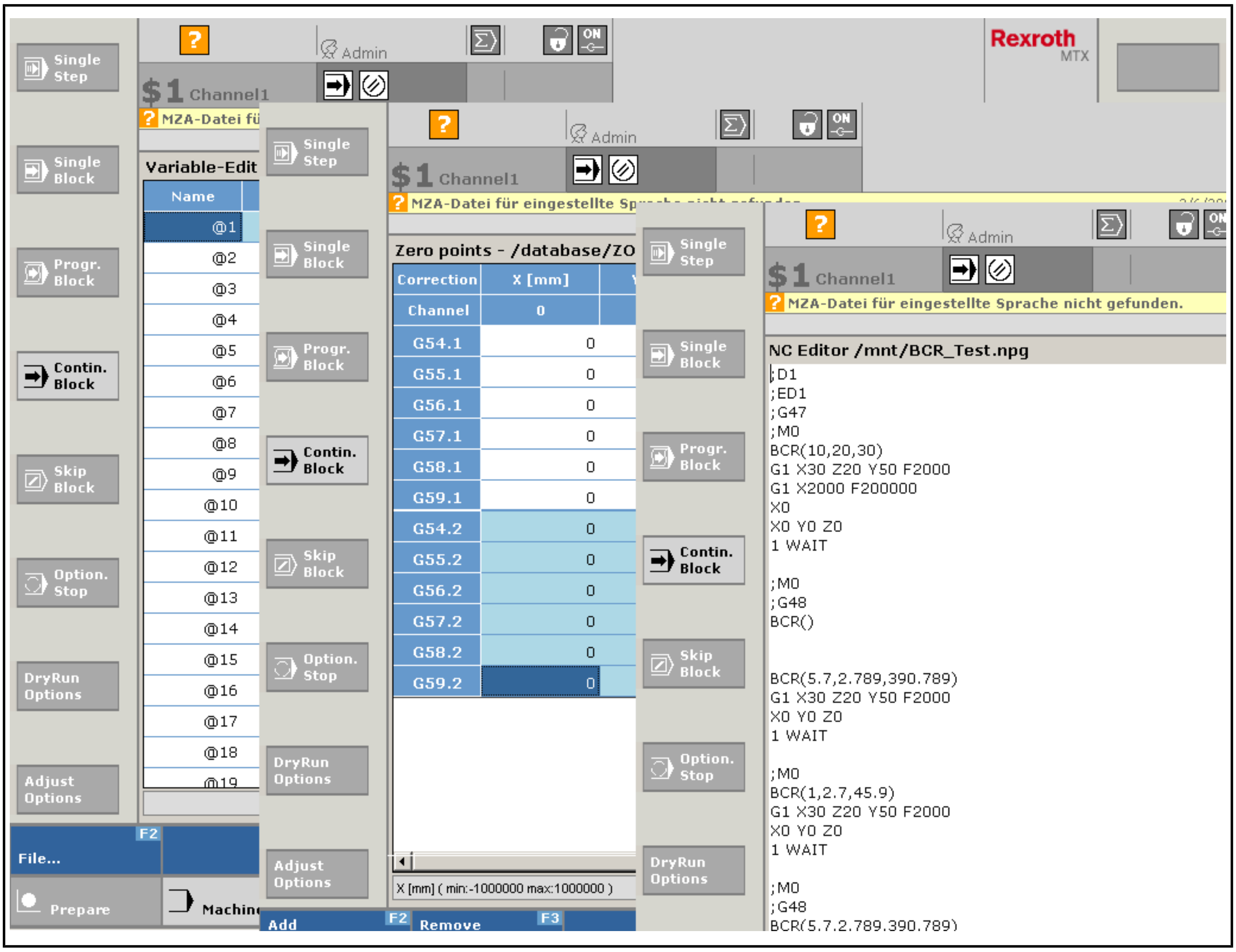


Fig.4-31: MTX editors

OP Key "OP5 - Tool Management"

Pressing the OP key **OP5 - Tool management** opens the tool management of the IndraMotion MTX. Tools can be inserted in the table, deleted and edited. New tool lists can be loaded and saved.

General Overview

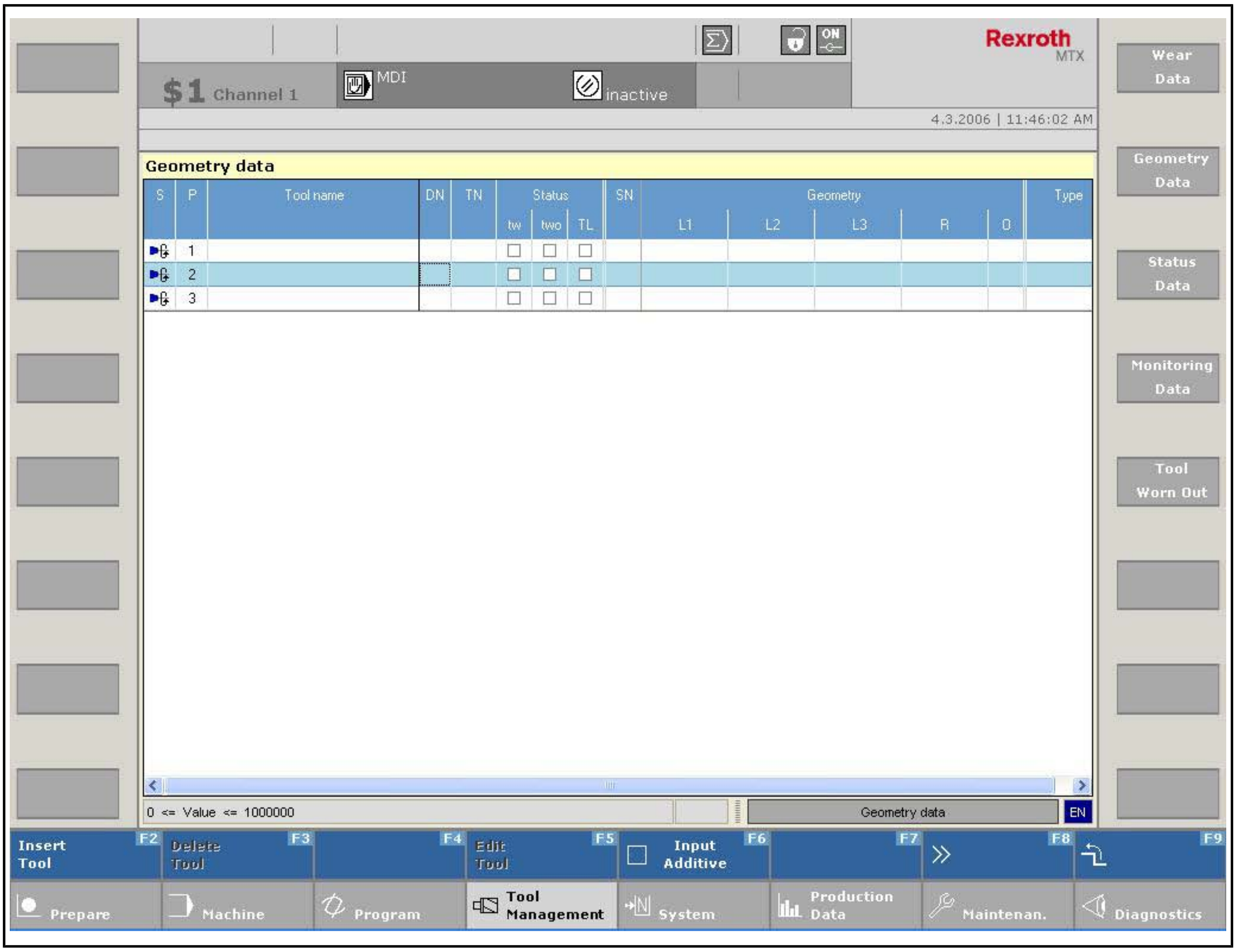


Fig.4-32: IW Operation desktop - Tool management

OP Key "OP6 - System"

OP6 - System (or Machine) can consist of up to 12 channels. Switching from one channel to the next is carried out here. In addition, status/diagnostic displays of the channels are shown. This screen is an overview for all channels.

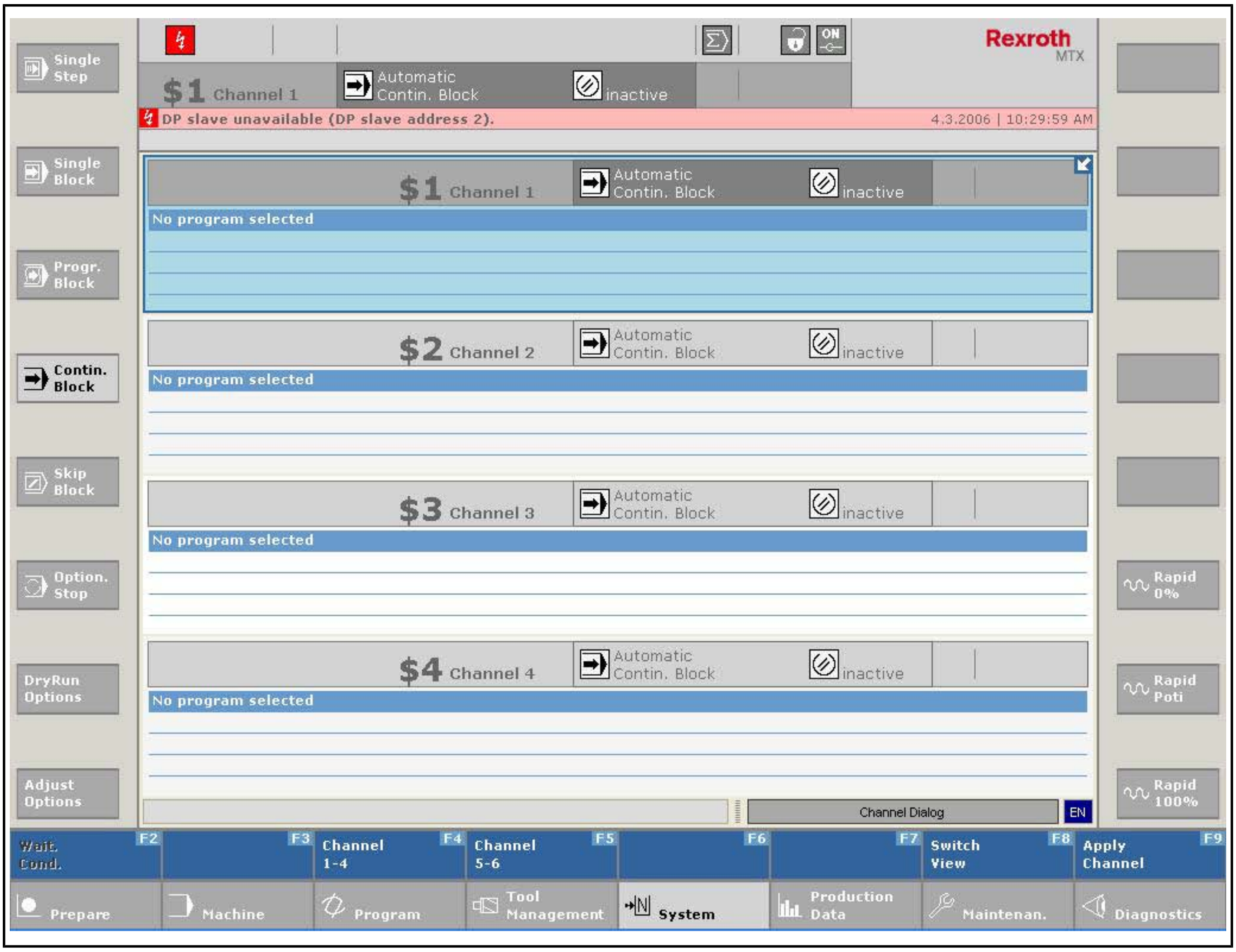


Fig.4-33: IW Operation desktop

OP Key "OP7 - Production Data"

Pressing the OP key **OP7 - Production data** displays the following data:

- Operating hours counter
- Items-produced meter

These data must be configured by the user himself. Bosch Rexroth provides only the possibility for displaying the data.

OP Key "OP8 - Maintenance"

The OP key **OP8 - Maintenance** switches to Engineering Desktop; furthermore, the interface can be exited using the F-key "F9 - Exit", provided that WinLock has not been activated.

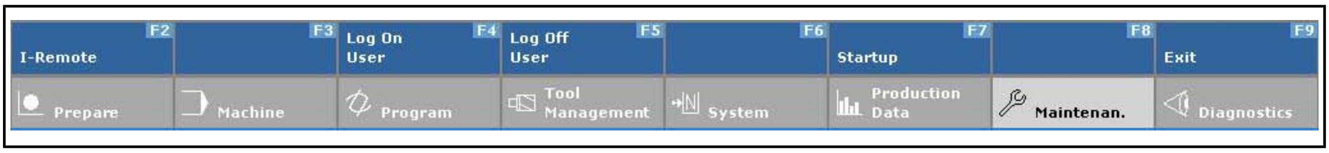


Fig.4-34: IW Operation desktop

OP Key "OP9 - Diagnostics"

The OP key **OP9 - Diagnostics** is used to centrally display all diagnostics and messages. If it exists, detailed information can be called up for each error. The logbook function set up in Engineering Desktop can be viewed here. The interface signals between the NC and the PLC can be visualized for troubleshooting and commissioning.

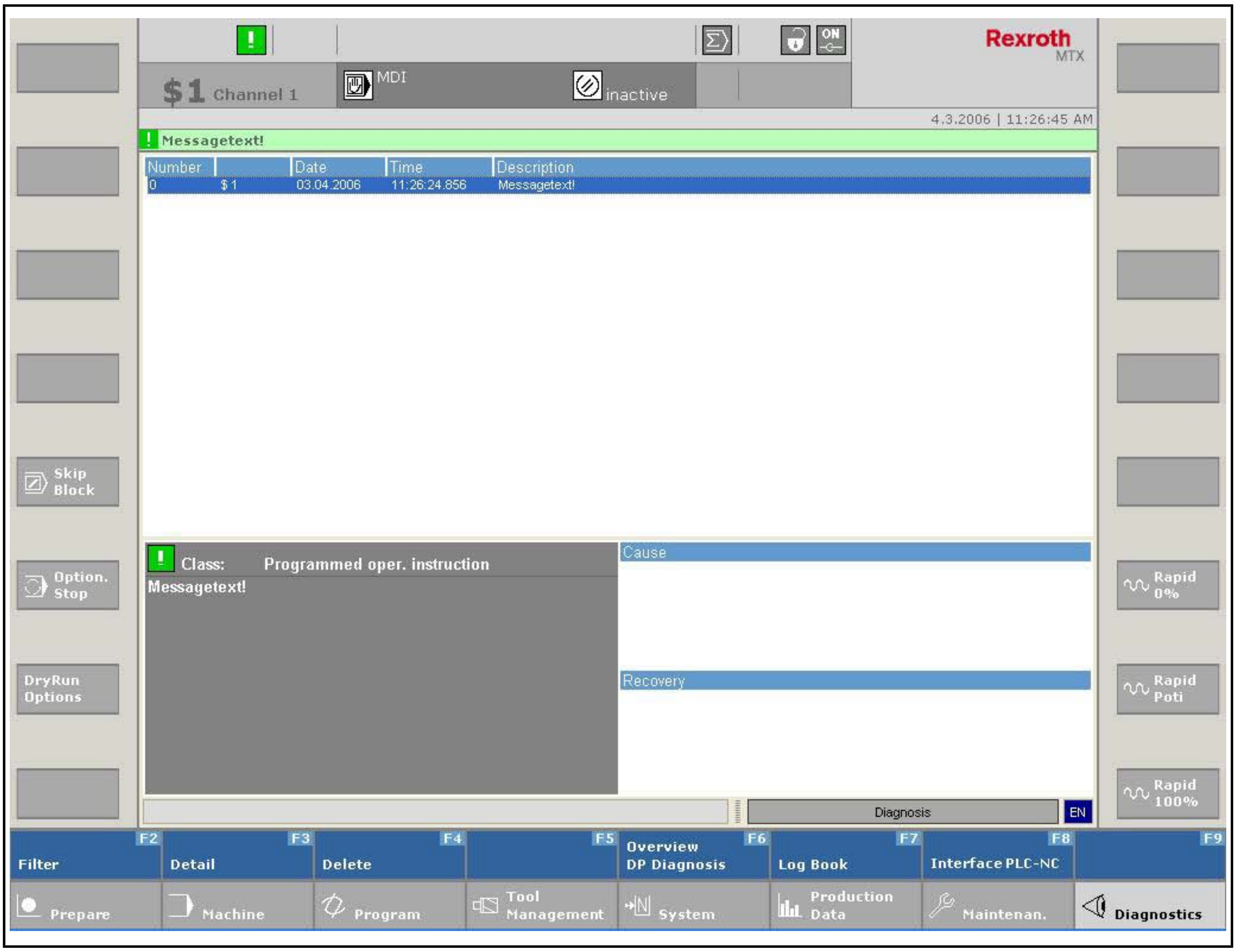


Fig.4-35: IW Operation desktop

4.4 Directory Structure of the IndraMotion MTX Control

4.4.1 General

The file system used by the IndraMotion MTX is located in the RAM of the control hardware. The control can also access directories which are not on the card, e.g. on the USB stick, the hard disk of the basic PC device or in the network. These so-called "mount directories" directories are used for backups and for general data storage and data exchange. The NC user interface can directly access up to 10 mount directories. Possible mount directories must exist and **enabled for mounting**.

4.4.2 Network Directories

General

The directory of a network computer can be connected as a mount directory only if the NFS server has also been installed and started on this computer so that the desired directory can be exported.

The NFS server has already been installed and started in the IndraMotion MTX control system.

The following handling instruction precisely describes how a network directory is created and mounted.

Handling Instruction: Mount Directories

Follow the steps described below to create a mount directory using the IndraWorks Engineering:

1. Go online with the control
2. Start "Configure Mount Directories..." in the context menu of the device node.
3. Create a new entry.

☞ The "new" mount directory becomes effective only after the system is restarted.

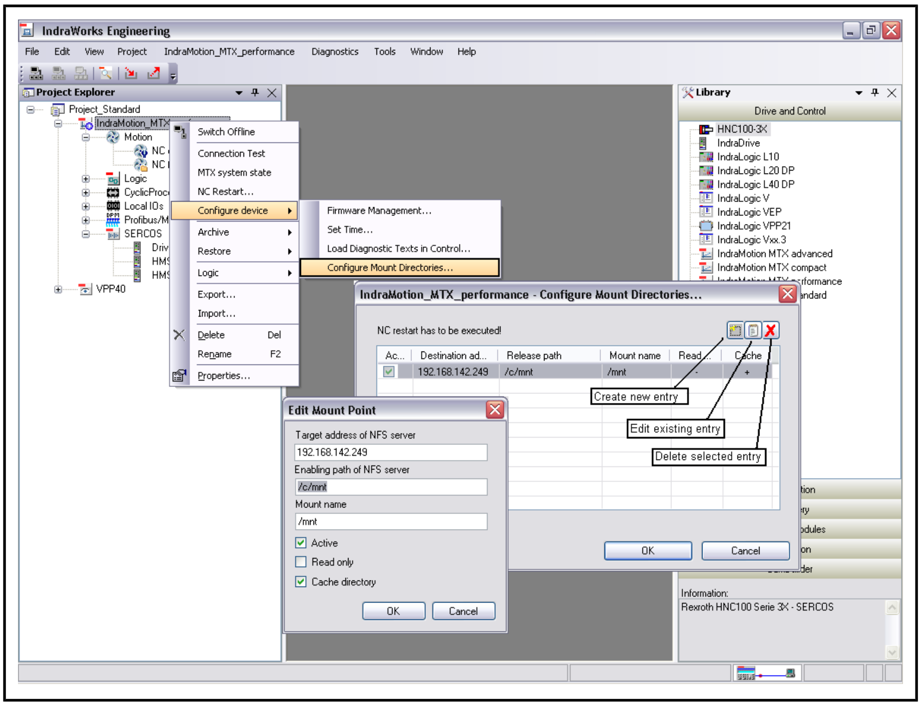


Fig.4-36: Context menu: IndraMotion_MTX_performance (Configure Mount Directories)

The following fields are to be entered in the "Edit Mount Point" dialog:

- Target address of the NFS server

  The default value is the IP address of the server. The value can be adapted if required.

- Shared path of the NFS server

  The path of the directory shared in Windows is to be entered (in the Windows notation).

- Mount name

  This name is shown in the control file system.

☞ Up to 10 mount directories can be created. If this number is exceeded, no further directories can be created. One of the existing mount directories must first be deleted.

Please **consider** the following for this function:

- Only one mount directory can be cached internally.
- Remote Engineering is not supported.

Handling Instruction : Importing/Exporting Data from the Windows Directory to a Directory of the Control System

This handling instruction describes briefly how data is imported from a directory in the Windows file system to the directory in the control file system.

The import dialog imports data from the Windows file system to the NC kernel file system:

1. Go online with the control
2. Start the MTX Navigator
3. Select a directory in the control file system
4. Open the context menu with the right mouse button
5. Start the "Import..." function
6. Highlight the files and/or directories to be imported (multi-selection)
7. Confirm the dialog

The data from the control file system can be exported in the Windows file system in the same manner, but with the "Export..." function.

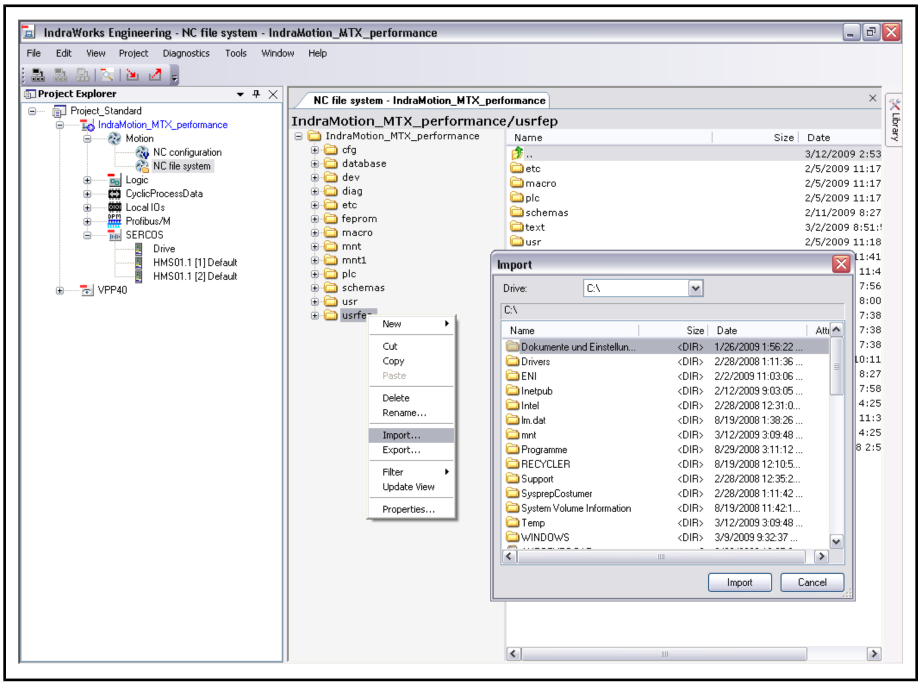


Fig.4-37: Dialog: IndraWorks Engineering (Import...)

☞ A detailed description of the import and export of directories can be found in the chapter 10.3.3 Directory Functions, page 258

A detailed description of the import and export of files can be found in the chapter 10.3.4 File Functions, page 260

4.5 MTX System Status

4.5.1 Introduction

The MTX system status is a commissioning and diagnostic application for PC-based MTX controls. This application controls, monitors and visualizes the state of the following component groups:

- MTX standard
- MTX performance
- MTX advanced

This application can be accessed via the device node context menu in IndraWorks Engineering.

☞ The control does not have to go online for this application!

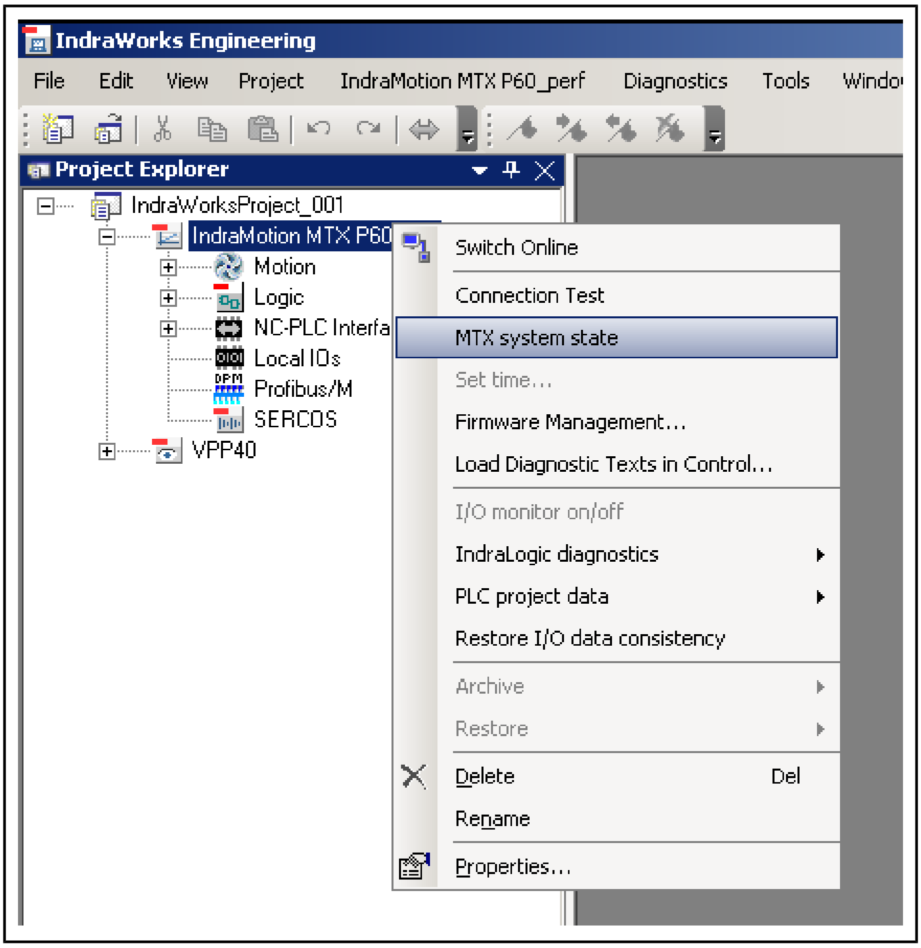


Fig.4-38: Starting of MTX system status

After opening the application, it is displayed in IndraWorks Engineering.

General Overview

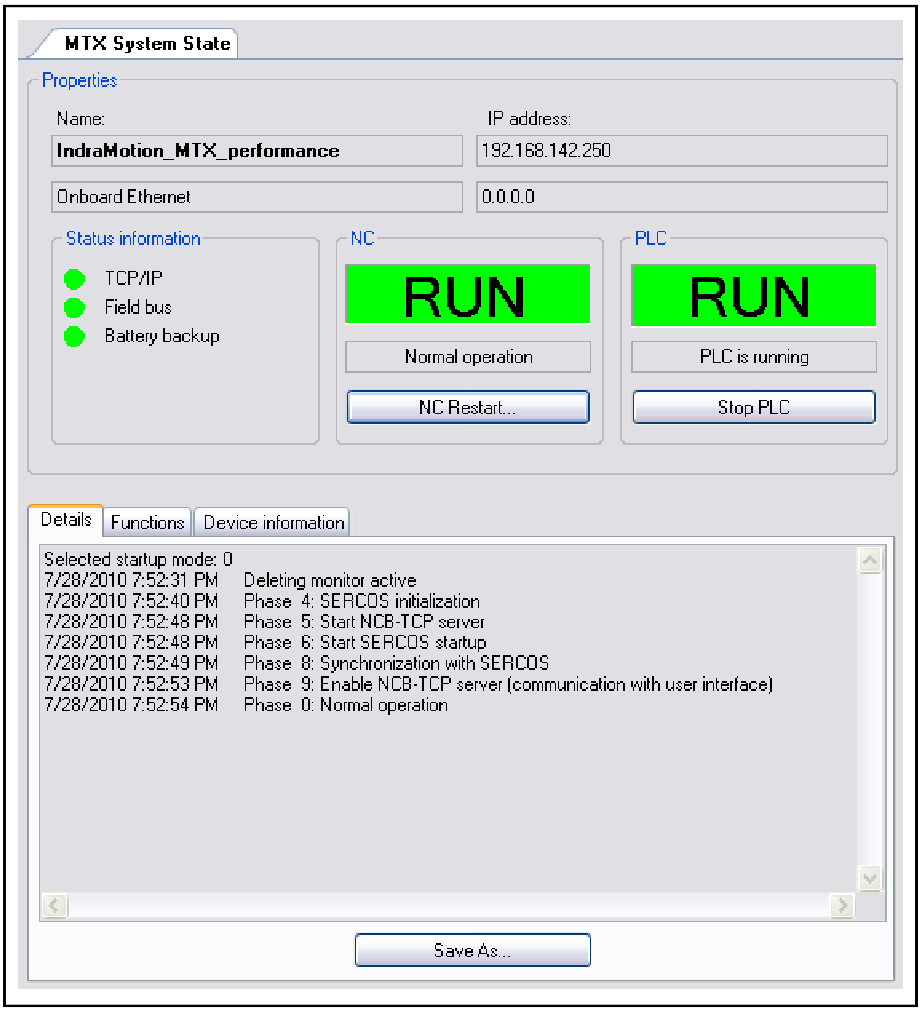


Fig.4-39: MTX System Status

The MTX system status is divided into two sections.

The general information which is divided into several sections and which is always visible to the user can be found in the upper section.

Several tabs which can be changed by the user contain various information and can be found in the lower section:

- "Details" tab
- "Functions" tab
- "Device Information" tab
- "SRAM" tab

4.5.2 General

Field "Name and IP Address"

This is an area of the upper application section in which the following information is displayed:

- **Name**

  The name to be configured for the respective control in the project tree is displayed. A name change in the project tree becomes immediately visible in the application. The name of the onboard Ethernet cannot be changed.

- **IP address**

The IP address and the onboard Ethernet address entered in the network configuration is visualized.

Field "Status Info"

The status information of the control is displayed using red and green LEDs:

- **TCP/IP**

  green LED = Data exchange is active

  green LED (not filled) = No data exchange between interface and control
- **Fieldbus**

  green LED = Fieldbus function ok

  red LED = Problem with fieldbus
- **Backup battery**

  green LED = Correct buffer battery state

  red LED = Buffer battery discharged or buffer battery not available

"NC" Field

In this area, the NC information is displayed.

- **Status field of the NC**

  This field visualized the current NC status which is displayed in color and short form. The used abbreviations are described in chapter "Startup" on page 65.

  This field is flashing during control startup.
- **Info field with long status name**

  The abbreviations in the status field are explained to the user in this field.
- **Button for NC restart**

  By means of this button, the user can execute an NC restart. After clicking this button, the following dialog is opened in which the user can specify the Startup mode (see chapter "Startup Mode" on page 66) for the restart:

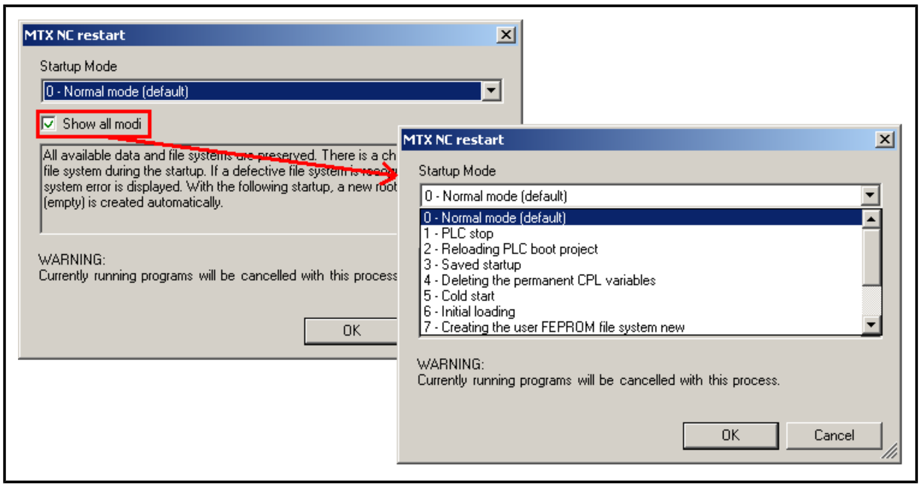


Fig.4-40: MTX NC restart

If the checkbox "Show all modes" is not selected, the user can only chose between 3 modes.

- 0 - Normal mode
- 6 - Bootstrapping

- 7 - Creating the user FEPROM file system.

"PLC" Field

In this area, the PLC information is displayed.

- **Status field of the PLC**

  This field visualizes the current PLC status. This status is displayed in color and as a short form.

  – **RUN** is displayed on green background during normal mode in the display.

  – If the PLC is at standstill, **STOP** is displayed on red background.

  – If no PLC program is loaded, a **"?"** is displayed on grey background.

- **Info field with long status name**

  The abbreviations in the status field are explained to the user in this field. The following detailed descriptions exist:

  – RUN = PLC is running

  – STOP = PLC is at rest

  – "?" = no program

- **Button to start/stop the PLC**

  If the PLC is running, it can be started or stopped using this button.

4.5.3 "Details" Tab

In this tab, the results of the last control restart **executed using this application** are visualized. Date, time, phase and their meanings are displayed. The display is refreshed after each restart.

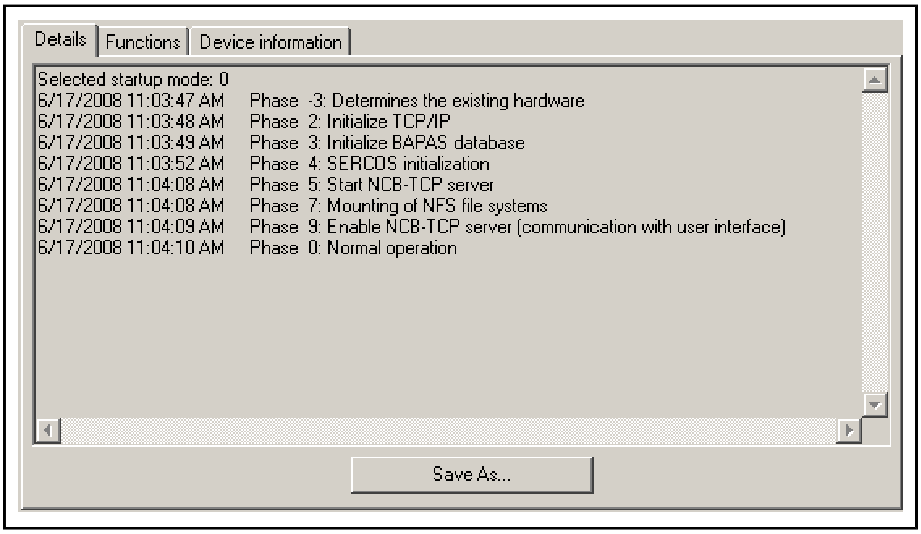


Fig.4-41: "Details" tab

If required, this information can be saved in a *.log file. The path under which the file is to be saved is arbitrary. The application saves this information under the default name "Details.log".

4.5.4 "Functions" Tab

The second tab is only available for authorized users. It is divided in three parts:

- Deleting memory
- TeleBugger
- Boot Parameter

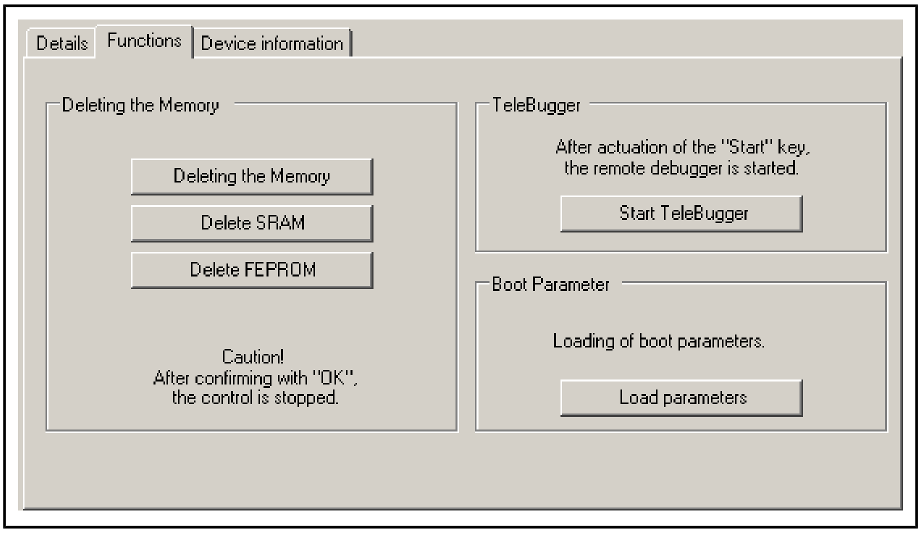


Fig.4-42: "Functions" tab

Delete Memory

There are three different memory sections in the control which can be deleted irrespective of each other.

Deleting memory	DRAM It deletes except the monitor and boot loader memory sections. This function is only required in exceptional cases in the software engineering field.
Delete SRAM	SRAM It deletes (root file system, permanent CPL variables, remanent PLC data, remanent system data). This function is only required in exceptional cases.
Delete FEPROM	FEPROM It deletes (firmware, FEPROM file system). This function is only required in exceptional cases.

Fig.4-43: "Deleting memory" functional description

Remote Debugger

Remote debugging is possible with the external application "TeleBugger" . This application allows searching an error on a remote control. The figure shows the main screen of the TeleBugger.

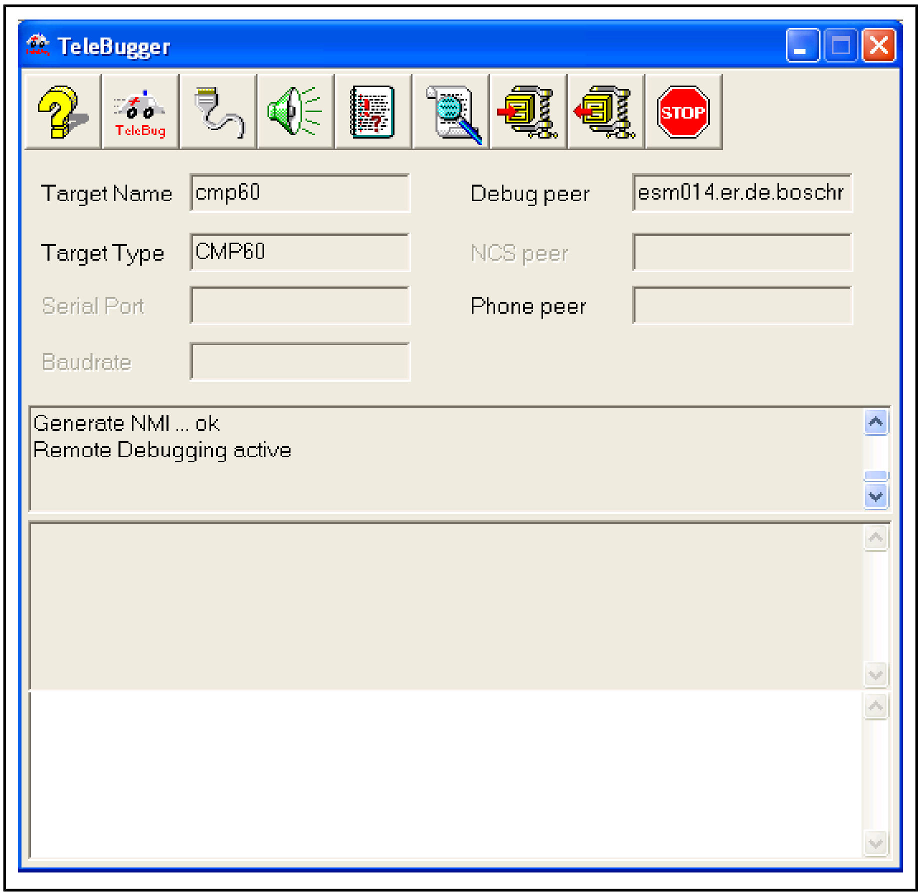


Fig.4-44: Main screen of TeleBugger

Detailed information on the functionality of this tool can be found in the online documentation of the TeleBugger.

Boot parameters

The MTX boot parameters are located in the file **mtxpboot<**PCB number **>.ini** in the home directory of the control. Upon each software or hardware reset, the boot parameters are read from this file and copied to the SRAM. Then, a startup is executed with the current boot parameters.

If the mount parameters or the Ethernet interface configuration is changed, the set values are saved to the boot parameter file and are also copied to the SRAM.

The boot parameter file contains information on the incorporated external file systems and the configuration parameters of the Ethernet interface.

☞ To avoid an incorrect parameterization, the boot parameter file should not be modified manually.

4.5.5 "Device Information" Tab

The following information on the control is displayed on this tab:

- Firmware version
- PCB number
- Hardware version
- Serial number

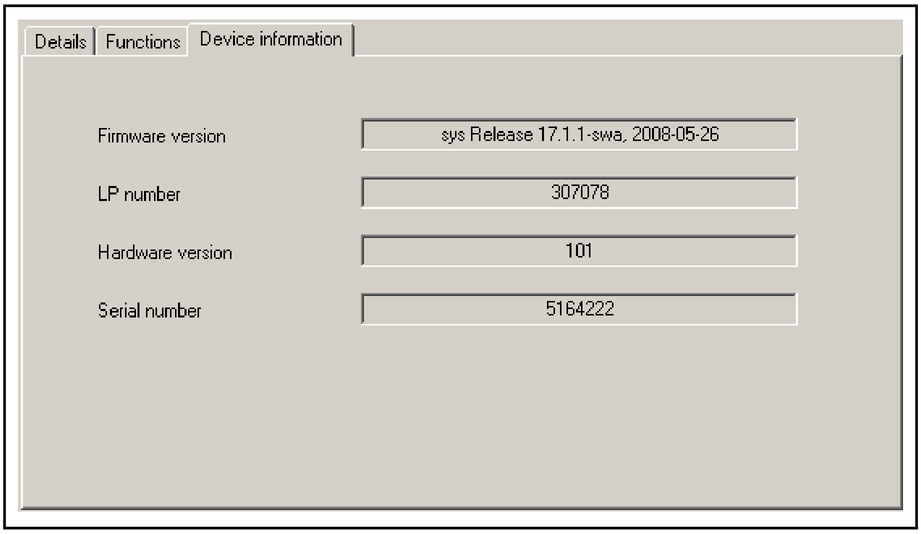


Fig.4-45: "Device Information" tab

If no connection is established to the control, the fields on this tab are blank.

4.5.6 Startup Configuration

Startup

The MTX starts when switching on the control or triggering an NC restart. The startup occurs synchronized in 12 phases displayed in the MTX system status:

P: -3	Determining the existing hardware
P: -2	RTOS startup, configure file systems
P: -1	Start RTOS monitor
P: 1	Initialize basic NCS communication
P: 2	Initialize TCP/IP
P: 3	Initialize BAPAS database
P: 4	SERCOS initialization
P: 5	Start NCB-TCP server
P: 6	Start SERCOS startup
P: 7	Mounting NFS file systems
P: 8	Synchronization with SERCOS
P: 9	Enabling NCB-TCP server (communication with user interface)
RUN	Normal operation

Fig.4-46: Display of startup phases

For critical system errors, boot panic errors or active MTX shutdown, the display changes to:

SF	A critical system fault is pending
BP	A boot panic error is pending
SD	Shutdown active

Fig.4-47: Error status display

If the monitor is active, the monitor status is displayed:

M: A	Ethernet active
M: 8	Ethernet inactive
M: L	Loading active
M: d	Deletion active
M: NMI	The monitor is in an NMI routine (error or power down)
M: E	Internal error in monitor
M: H	Hardware unknown

Fig.4-48: Display of active monitor

Startup Mode

The startup mode determines the behavior of the MTX during startup.

Startup mode changes become effective only during the next startup.

Startup mode	Meaning
0	**Normal operation** All existing data and file systems are retained. The root file system is checked during startup. If a defective file system is detected, a critical system error is displayed. A new (empty) root file system is automatically created during next startup.
1	**PLC stop** The behavior corresponds to Startup mode 0 with the difference that the PLC remains in the STOP state and the PLC user program is not processed.
2	**Reloading the PLC boot project** The PLC boot project is loaded from the user FEPROM. Any PLC boot project that exists in the root file system is discarded. Otherwise, the behavior corresponds to startup mode 0.
3	**Save startup** In extreme cases, due to faulty machine parameter specifications, it can be impossible to execute a control startup. Startup mode 3 carries out a startup in this error situation, regardless of the set machine parameters. A startup with the minimum configuration is carried out and the machine parameters set are ignored. After the startup, the invalid machine parameter settings can be corrected and a new startup in startup mode 0 can be carried out.
4	**Deleting the permanent CPL variables** The permanent CPL variables are deleted. Otherwise, the behavior would correspond to startup mode 0.
5	**Cold start** The power-up management logic is not run through. Otherwise, the behavior would correspond to startup mode 0.

Startup mode	Meaning
6	**Bootstrapping** A new root file system is created. As a result, all old file system data is lost. If an intact user FEPROM file system exists, the PLC boot project and configuration data are loaded from there.
7	**Creating the user FEPROM file system again** The user FEPROM is created again. As a result, all the old file system data is lost. This is required, for example, if a user FEPROM file system is defective. The root file system is retained. The permanent CPL variables are deleted.
9	**Debug mode** This is the usual for the debugging if the control does not automatically boot after a reset. After the basic monitor is initialized, the boot loader is activated and the subsystems are automatically loaded.
10	**Debug mode (without automatic loading)** After the basic monitor has been initialized, the boot loader is activated. Further loading can take place via TCP/IP.
11	**Debug mode (without activating the boot loader)** The basic monitor is initialized. Further loading can take place via TCP/IP.
15	**Debug mode (basic monitor start)** Only the basic monitor is activated.

Fig.4-49: Startup mode

5 Axis Commissioning

5.1 Commissioning Tools

5.1.1 NC Configurator - Machine Parameters

General

The Engineering desktop is the main tool for modifying and configuring data and settings of the MTX control system. After the Engineering desktop has been started using the "IndraWorks Engineering" icon, the following screen appears.

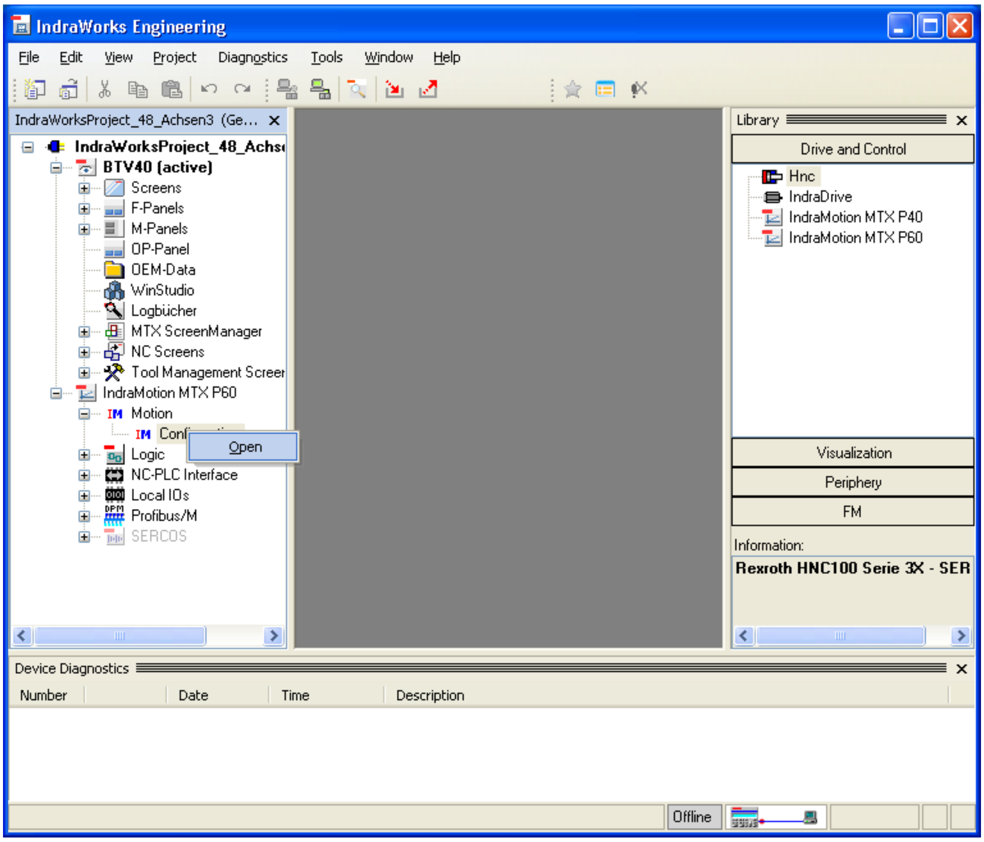


Fig.5-1: Engineering desktop

The Project Explorer (left column) is used to navigate through the individual topic areas.

The "Configuration" section under the node "IndraMotion MTX P60 \ Motion" is important to commission the NC axes. This section is described in more detail in the following.

Axis Commissioning

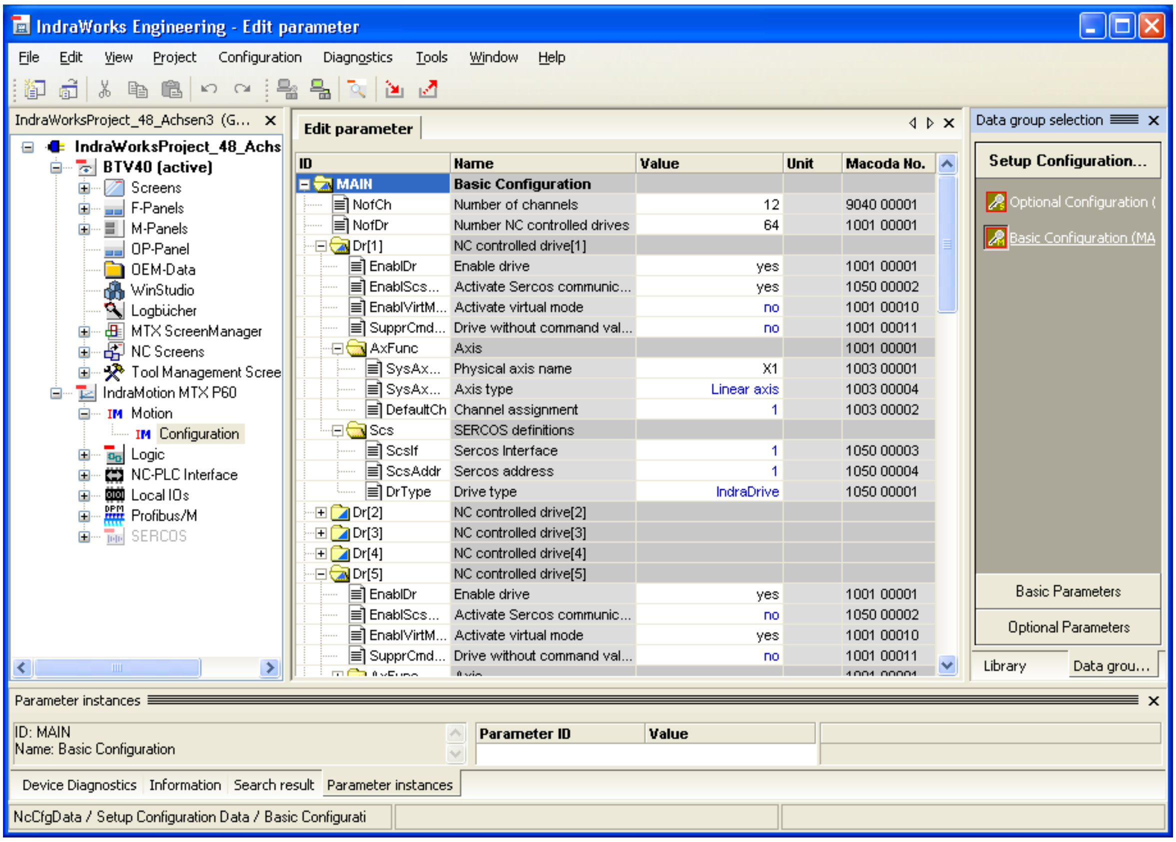


Fig.5-2: "Configuration" node

The desktop is divided into several sections:

- Project Explorer (left column)
- "Edit parameter" window (centered)
- Data group selection (right column)

Select Parameters

The section to be edited can be selected in the "Data group selection" column under "Basic Parameters".

Important for the commissioning of the drives are:

- Main configuration (MAIN),
- Axes (AX),
- Spindles (SP) and
- SERCOS communication (SCS)

The optional parameters can be selected in the section "Setup Configuration Data". These parameter ranges cover specific topics, such as "electronic couplings", "measuring", "traveling against fixed stop" etc. They are only displayed in "Optional Parameters" after they have been selected.

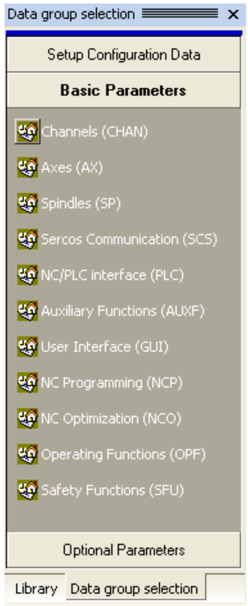


Fig.5-3: "Basic Parameters"

The "Configuration" tool is described in the manual "Bosch Rexroth IndraMotion MTX Machine Parameters". Only the important steps to commission the drives are described in the following section.

Finding Parameters

The configuration parameters relevant to commission the drives are described in another chapter later on.

In order to find the listed parameters more quickly, a search function is available in the main menu under **Edit\Search**. This find function can also be started using the shortcut <Ctrl> + <F3>.

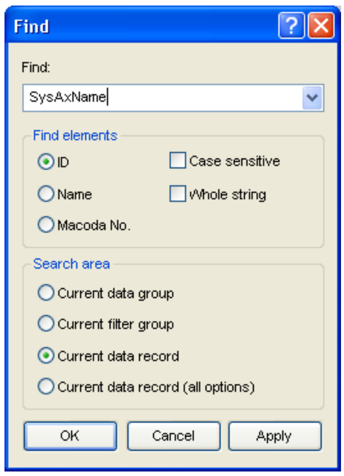


Fig.5-4: "Find" menu

The search for "SysAxName" is shown here as example. The search results are displayed in Engineering Desktop under the tab **Search Results** in the lower part of the screen. Double-clicking on a search result listed there results in a jump to the corresponding parameter.

5.1.2 NC Editor - SCS Files

The IndraWorks Operation desktop is the main interface for operating the system/machine. The OP key **Program** is important for configuring the drives on the control. This OP-key can be used to configure the required SCS files. The precise meaning of the SCS files is described in the following chapter.

A detailed description of the Operation desktop can be found in the manual "Rexroth IndraMotion MTX - Commissioning".

The SCS files are important for commissioning the drives. They are used for the SERCOS initialization of the connected drives while the control is starting up.

☞ The handling of the SCS files and the configuration syntax are described in the next chapter of this manual.

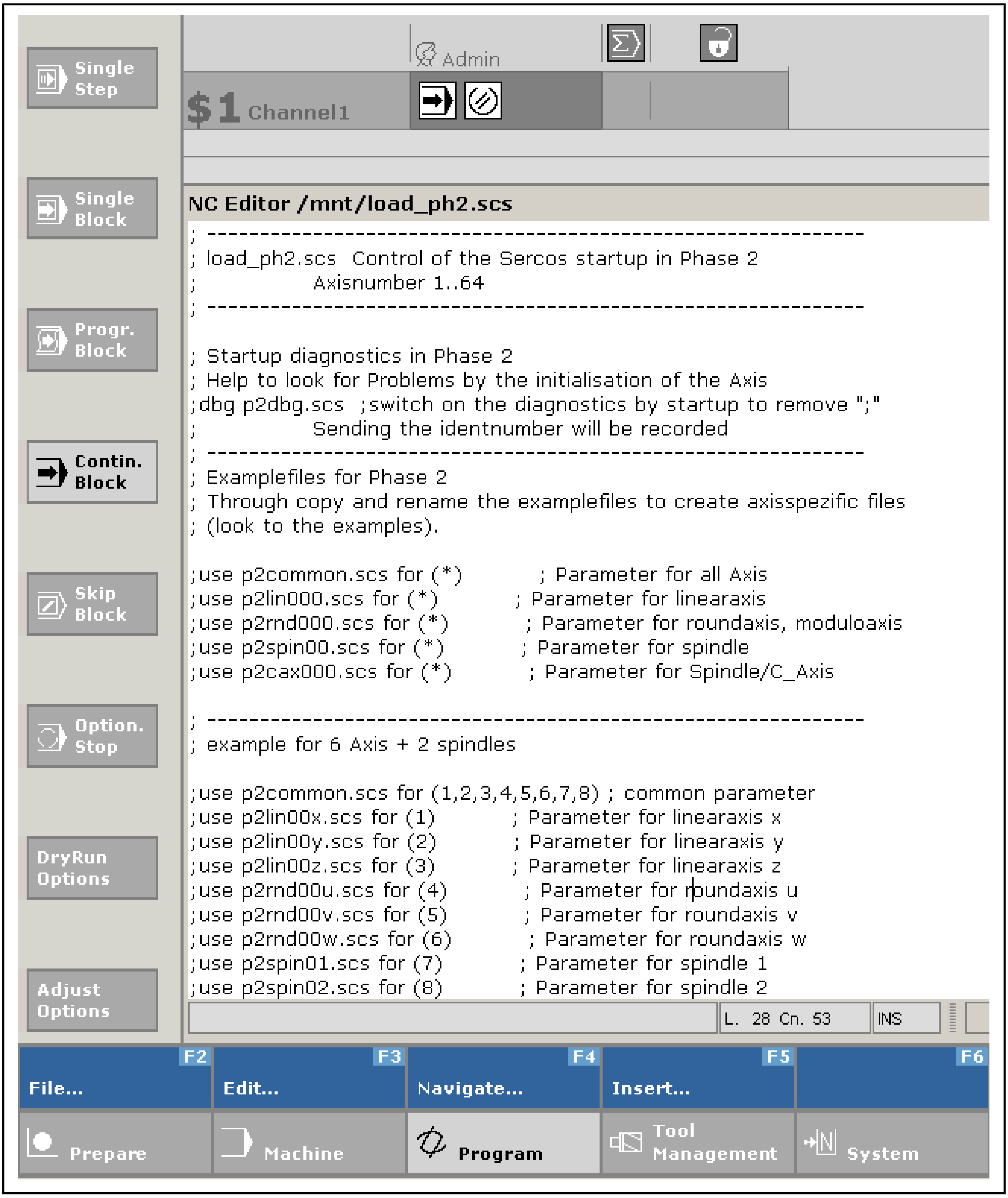


Fig.5-5: Editor for SERCOS files

The "IndraMotion MTX" control system can already provide SCS example files located in the control directory "//root/feprom/scsindra/" ("root" stands for the control name). Copy the examples to the directory "//root/" and adjust them to his application.

The SCS files can be opened and edited by double-clicking on them.

5.1.3 SERCOS Master

General

After the Engineering desktop has been opened, communication is not possible via the non-cyclic service channel of the SERCOS ring. The control must at first go "Online" in order to be able to edit drive data. This can be accomplished via the icon in the toolbar or via the main menu under "Project - Online". If the SERCOS ring is at least in Phase 2, all the associated drives are created in the project. However, communication is not yet active.

The communication with the drives is already activated when the drive is selected. Alternatively, it is possible to use the context menu of the SERCOS node to go online simultaneously with several drives.

The following functions are provided at the SERCOS node.

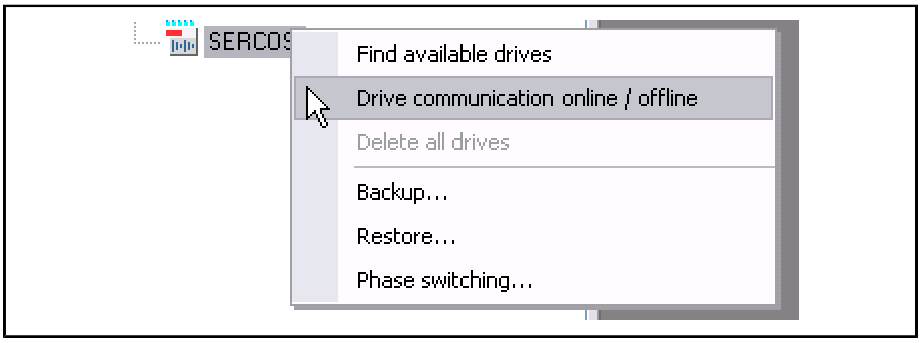


Fig.5-6: SERCOS context menu

- **Find available drives:**

  All drives configured in the NC data are created. If required, a switch to SERCOS phase 2 is triggered upon request. This function is already triggered when the control goes online.
- **Drive communication online / offline:**

  This function establishes and cancels the communication with the drives. In a dialog, the drives can be selected for going online. This function implicitly includes searching for drives if not all existing drives are in the project.
- **Delete all drives:**

  This function removes all the drives below the SERCOS node, including the offline data, from the project. They are created again during the next time "connection establishment/drive search".
- **Backup...:**

  A dialog opens in which the drives are selected for backup. Furthermore, it can be determined whether all parameters or only the changed parameters are to be backed up. The selected drives go online accordingly.
- **Restore...:**

  In a dialog, the drives, for which a previously created archive is to be written back again, are selected. The selected drives go online accordingly.
- **Phase switching...:**

  A dialog for switching the SERCOS phase opens.
- **SERCOS participant, address specification (only for SERCOS III):**

  It is possible to set the SERCOS address of the bus participant. The participants are identified via their position in the ring

chapter "SERCOS Participants Address Specification (only for SERCOS III)" on page 75

Behavior when Going Online

The behavior of the drives when the control goes online can be set in the "Properties" of the SERCOS node.

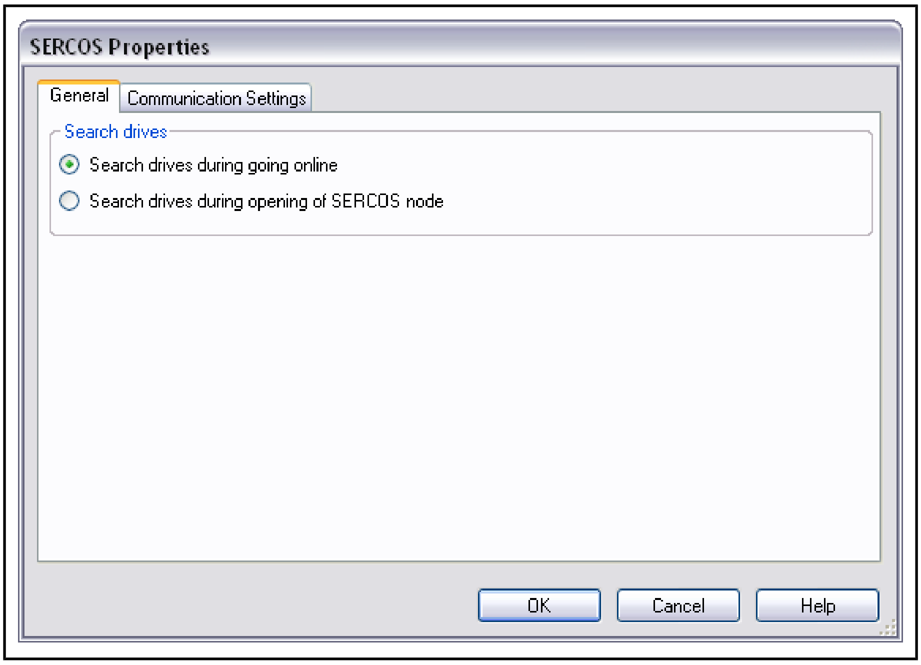


Fig.5-7: Find SERCOS settings for the drive

Before a communication can be established with the drives, the basic configuration of the drives in the project must be compared and, if necessary, updated with the drives available at the control. This might take several seconds depending on the number of participants at the SERCOS ring. Thus, it can be selected when the adjustment should be carried out.

- Finding drives when going online (default)

  When the control goes online, it is searched for drives. It can be immediately used for operation.

- Find drive when opening the SERCOS node

  When the control goes online, the SERCOS node collapses. The drive configuration is not checked. To access the drive parameterization, the SERCOS node is expanded. Only at this point in time, there is check.

  This setting can be very reasonable especially if there is a high number of drives in the ring and if the drive parameterization is rarely required.

SERCOS Participants Address Specification (only for SERCOS III)

This function is only available for SERCOS III. Therefore, the control must be in online.

This function compares the configuration of the participants in the project with the participants actually found in the control.

The data from the project is displayed on the left.

The data of the real bus participants is displayed on the right. It is shown in the sequence of the participants in the ring.

Axis Commissioning

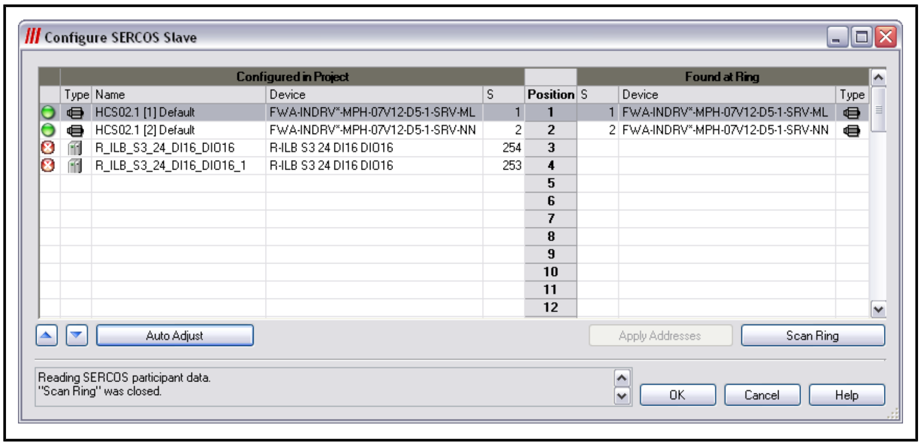


Fig.5-8: Dialog to configure SERCOS IIII participants

Column description

- **Project - Type:**

  Type of participant: Drive or I/O

  I/Os can be activated/deactivated by clicking on the icon. Clicking does not work for drives.
- **Project - Name:**

  Name of the participant in the project.
- **Project - Device:**

  Name of the participant: Drive firmware or I/O type
- **Project - S (SERCOS address):**

  Configured SERCOS address of the participant.

  This address can be changed for I/Os. The configured SERCOS address of a drive can only be changed in the NC configurator. Changed addresses are displayed in "**bold**".
- **Control - Type:**

  Type of participant: Drive or I/O
- **Control - Device:**

  Name of the participant: Drive firmware or I/O type
- **Control - S (SERCOS address):**

  Set SERCOS address of the participant.

  A new SERCOS address is set for the participant. Changed addresses are displayed in "**bold**".

Describing the operating elements:

- **Project - Arrow up;**

  It moves the selected participant one up in the project.
- **Project - Arrow down;**

  It moves the selected participant one down in the project.
- **Project - AUTO:**

  It sorts the configured participants in order to achieve a high number of matching cases.
- **Control - Scan Ring**

  The data of the participants at the ring are read in again.
- **Control - Apply Addresses**

The changed addresses are written to the participants and a new SERCOS startup is executed.

Status information

The status of the respective place is given as colored icon in the first column. More information on the respective status is given when keeping the mouse pointer on the icon for a moment (tooltip). The following status messages can occur:

- **Grey**

  The ring has not yet been scanned. The status is unknown.
- **Green**

  The configuration of the participant in the ring does not match the participant.
- **Yellow**
  - The device name does not match.

    There is another I/O participant at the ring as expected in the project.
  - Firmware does not match.

    The firmware release of the drive in the ring is different than expected in the project.
  - The SERCOS addresses do not match.

    The participant on the control matches the configured participant but not with the address. The SERCOS address can be adjusted in the project or at the participant.
- **Red**
  - The SERCOS address is used multiple times.

    A SERCOS address exists multiple times. The SERCOS addresses have to be corrected until they are unique.
  - No participant was found at this position.

    A configured participant was not found at the ring. An I/O can be deactivated in the project
  - No participant is configured at this position.

    A participant was found at the ring that is not configured. Either the participant has to be removed from the ring or added to the project.
  - The configured device type does not fit to the hardware found.

    A drive is configured in the project, but an I/O is available in the ring or vice versa.

Using the SERCOS III IP Channel

General

It is possible with SERCOS III to directly activate drives via their IP address. This allows to directly execute a firmware download or a PLC download for example via the SERCOS interface. Furthermore, tasks such as parameterization backups are normally faster this way.

Activating the IP channel

To use IP communication, certain settings have to be made. These can be found in the properties of the SERCOS node. The provided options differ whether being in online or offline mode.

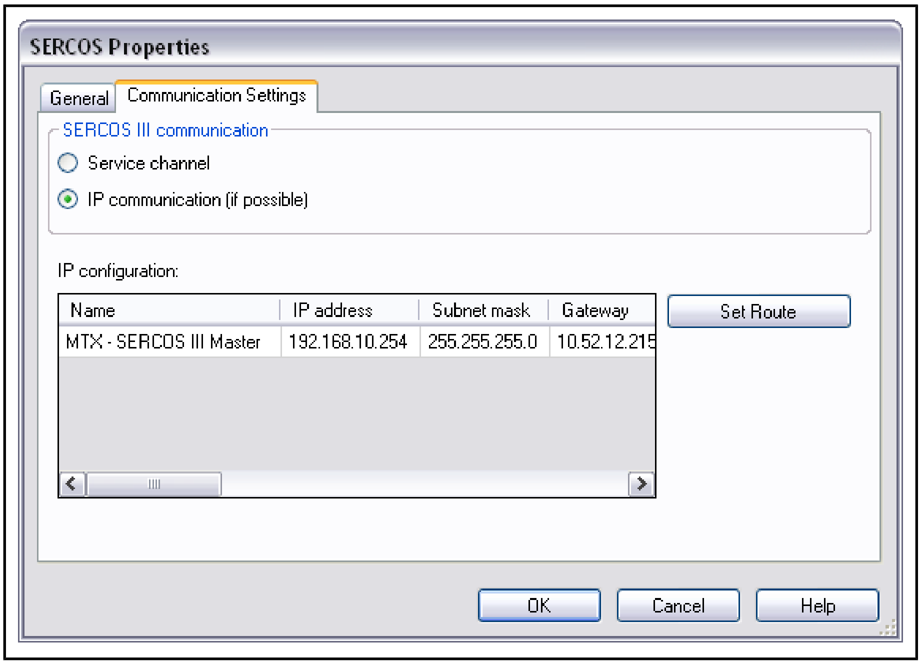


Fig.5-9: Activating the IP communication

It can be selected whether the IP channel should be used. If no IP connection is established to the drive, the service channel is automatically used. If problems occur with this setting, it can also be specified to always use the service channel. This information is saved in the project.

To use the IP channel, the Engineering PC must be familiar with the subnet of the drives. If this is not the case, there is the button **Set Route**. By selecting this function, the route to the drives is enabled.

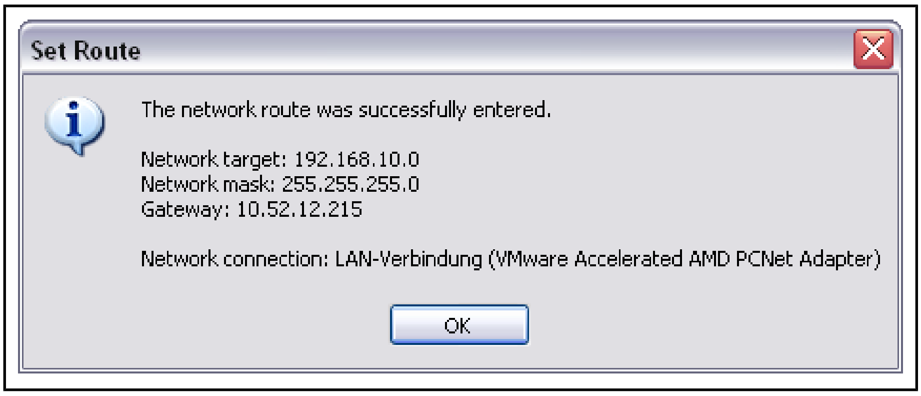


Fig.5-10: "Set Route" was carried out

Everything is done to use the IP channel. When going online next time with the drive, it will communicate via IP channel.

- Administrator rights are required to set the route.
- A route set this way is again deleted with the next PC restart and has to be created again.

- Another option to configure the route to the drives is provided by the command "route" in the Microsoft prompt. It can be also specified via parameters that the route remains after a PC re-start. For more information on this command, refer to the Windows help. The function used in the dialog corresponds to this call:

  ```
  route add 192.168.0.0 10.52.12.215 -mask 255.255.255.0
  ```

- For more options, contact your network administrator.

  The subnet to be routed in is determined by the address of the SERCOS III master. ("IP configuration of the master " on page 79). The gateway is the control itself.

IP configuration of the master

By opening "Advanced" at the SERCOS III master, different settings can be selected.

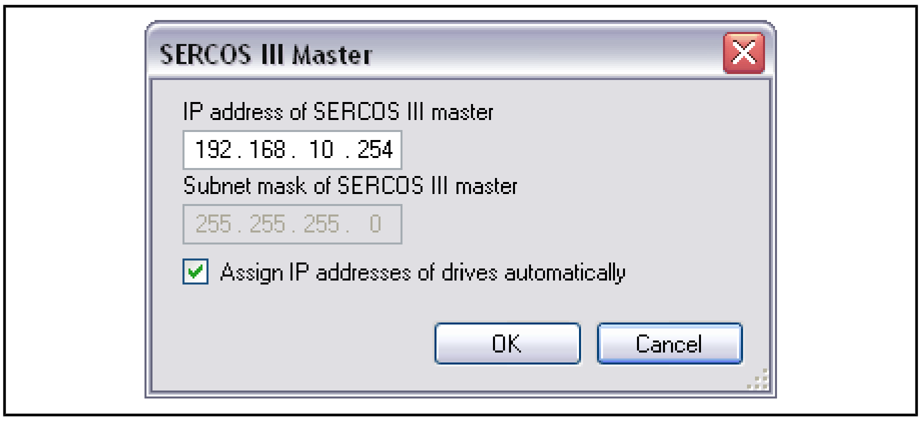


Fig.5-11: Advanced setting of the SERCOS III master

- **IP address:**

  IP address of the SERCOS III master interface. (delivery state: 192.168.143.254)

- **Subnet:**

  The subnet cannot be modified and is set to the value 255.255.255.0

- **IP addresses of the drives are automatically specified:**

  If this option is activated (delivery state), a permitted IP address is assigned to each drive during SERCOS startup. This consists of the subnet and the SERCOS address of the drive. (e.g.: IP address of the master: 192.168.143.254, SERCOS address of the drive: 4, IP address of the drive: 192.168.143.4)

  If this option is not enabled, the address has to be specified one time correctly for each drive.

5.2 Drive Commissioning

5.2.1 Overall Procedure

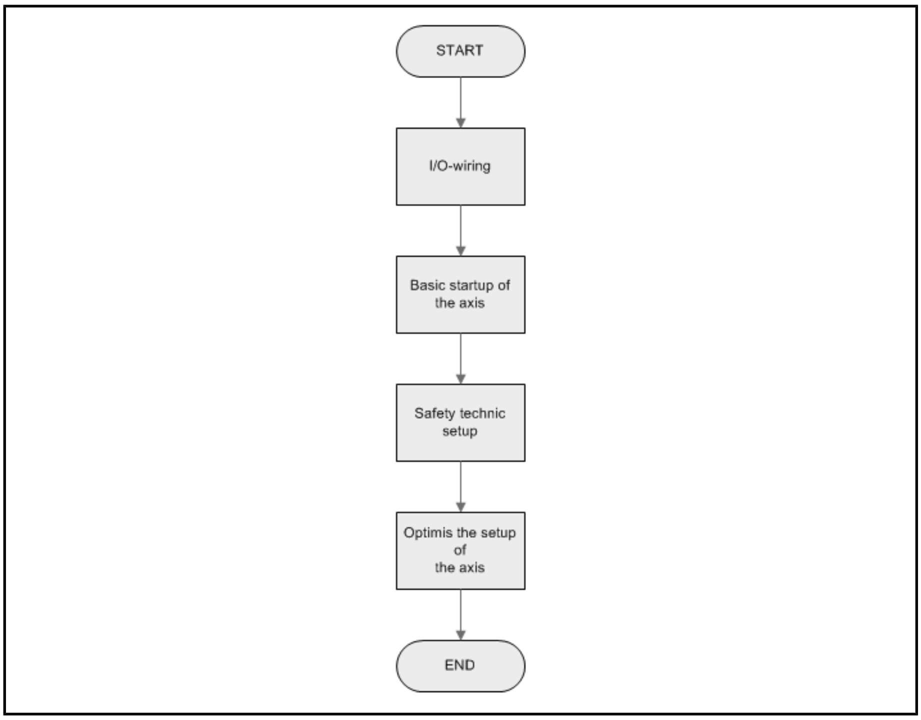


Fig.5-12: Drive commissioning procedure

5.2.2 I/O Wiring

General

⚠ CAUTION **Danger of personnel injuries and machine damage due to incorrect wiring!**

Carry out the wiring only according to the guidelines of Bosch Rexroth!

Handling Instruction: I/O Wiring

General wiring of the drives to prepare for commissioning.

Drives: Wiring the inputs and outputs, power wiring

The wiring of the drives is carried out according to the guidelines of Bosch Rexroth.

☞ Follow the instructions of the IndraDrive documentation to ensure proper use of the drives.

Handling Instruction: Activating the E-Stop Function

Before the axis can be operated with the "IndraMotion MTX" NC control, it must be ensured that the safety equipment is functioning properly.

System/Machine: Check the E-stop circuit

1. Check the E-stop circuit.
2. Check the safety technology of the drives and set the parameters accordingly if required.

☞ Observe the safety instructions on electrical drives.

		Documentation
Documentation:	Rexroth IndraDrive Drive System	Control circuits for power connection

Handling Instruction: General Drive Commissioning

Before commissioning, if possible, the drives of the system/machine should be operated together with the control but without the NC. This comprises the following steps:

IndraWorks Drive: Drive commissioning

- Mechanical installation of the drive amplifiers
- Electrical wiring of the motor, the drive amplifier and the control
- Commissioning of the overall electrical system and the switch cabinet
- Basic commissioning of the drives without NC
- Basic settings of the axis with IW-Drive

☞ The drive documentation should be referred to commission the drives.

The drives are optimized together with the control at a later point in time (after the control has been commissioned).

		Documentation
Documentation:	Rexroth IndraDrive Drive System	

5.2.3 Drive-side Parameterization

General

In order to parameterize the drives, the communication must be activated. This is carried out via the menu item "Online / offline drive communication". There, the necessary drives are selected and switched online. Depending on the number of the drives selected, this procedure will take some time.

During initial activation, the drives existing in the system are created in the project.

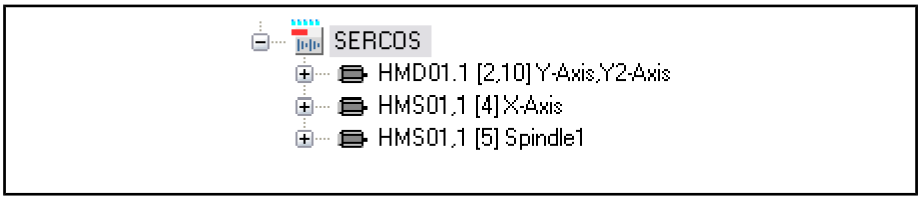


Fig.5-13: Project tree with drives

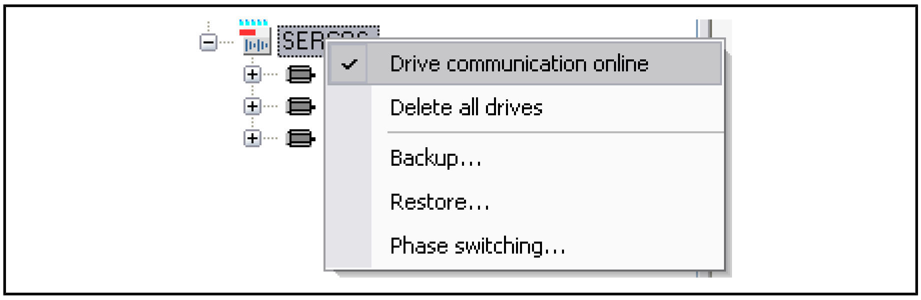


Fig.5-14: Context menu "Drive communication online"

The current phase is shown in green in the upper part of the dialog.

The appropriate phase is selected via the buttons "Parameterize" and "Operation". All phases can be selected explicitly in the advanced phase selection behind the right button. Ongoing phase switching is indicated with a progress bar.

Furthermore, the functions to be carried out during phase switching can be selected:

- Delete error of state class 1
- Transfer SCS files to SERCOS phase 2
- Transfer SCS files to SERCOS phase 3

A status message is displayed at the end of switching or if errors occur.

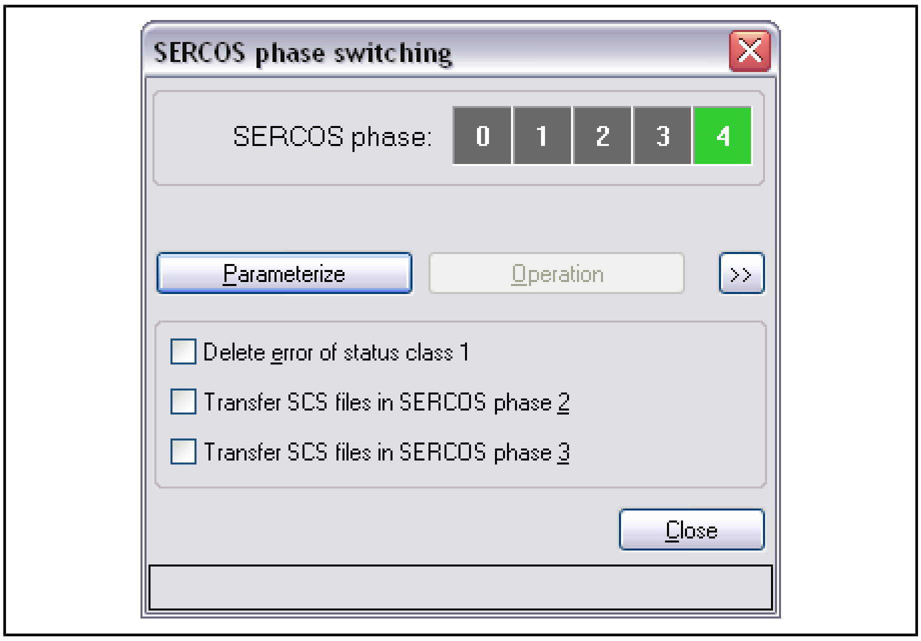


Fig.5-15: Phase switching dialog

☞ In case of IndraDrive drives, all relevant parameters can be transferred to phase P2. The parameter transfer in phase P3 is not applicable.

Handling Instruction: Setting Parameters for Drives in IW Drive

IW Engineering / Project tree: supplementing the IW data structure

1. Switch the control "Online" in order to be able to create drives

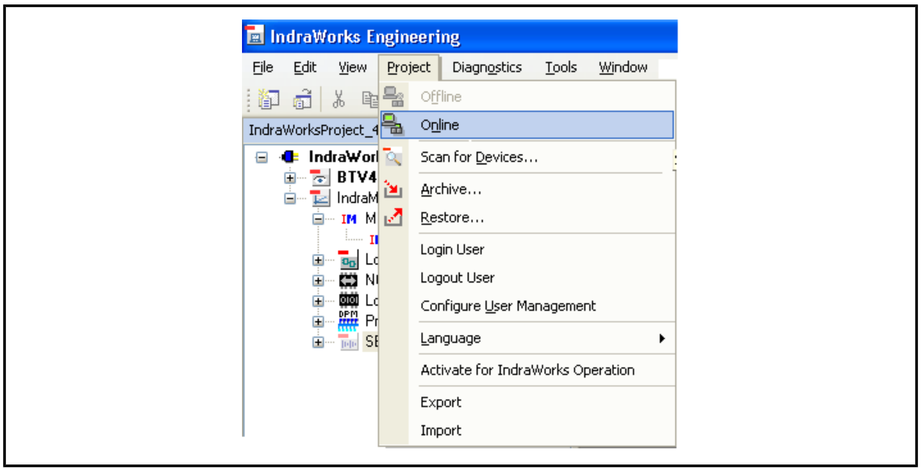


Fig.5-16: Switching the control "online"

2. Creating drives automatically
3. The function "Drive communication online / offline" can be started by clicking the right mouse button on the "SERCOS" node.

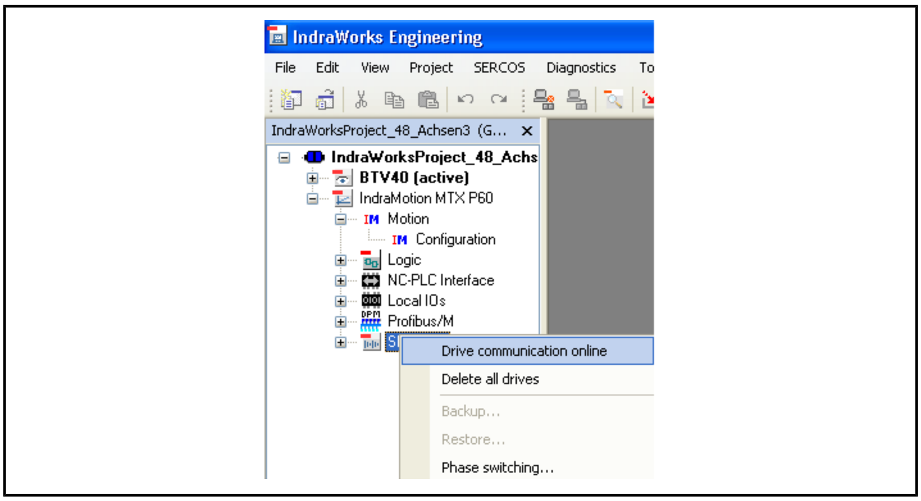


Fig.5-17: Creating drives

To completely create the tree structure of the drives, switch all drives "online" first. Only the drives that you want to edit need to be selected.

This function may take several minutes. The duration depends on the number of selected axes.

The successful creation of all the axes is then displayed as follows.

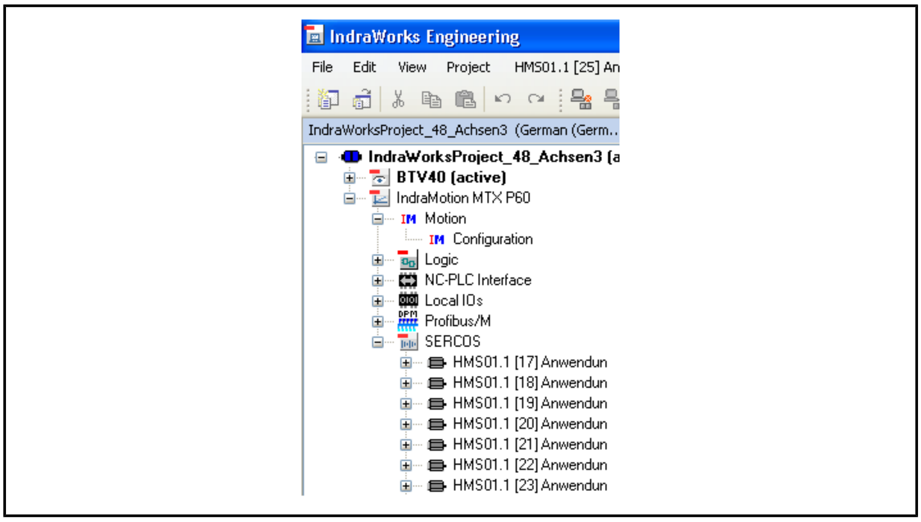


Fig.5-18: Displaying drives

☞ The data structure in in the Engineering desktop must be supplemented so that you can access the parameters of the drive. This is required when changing or archiving the drive parameters.

All the drives that have been created in the configurator should also be created in the IW data structure.

		Documentation
Documentation:	IndraWorks Commissioning	Commissioning the Axes

5.3 General Axis Commissioning

5.3.1 General

Handling Instruction: Traveling Axis with Control

After the axis has been configured, it must be traveled with the control. Before the axis is moved in Automatic mode, various basic settings should be checked.

IW Operation/operation mode "Jogging": Manual traveling of the axis

1. Check the traveling direction and the display
2. Homing
3. Set the zero point of the axis
4. Determine and set the zero point on the machine (in the case of absolute encoders)

☞ The override potentiometer should be set to a low value before starting the axis so that dangerous situations cannot occur.

IW Operation/operation mode "Automatic": Move axis with test program

1. Create a test program for the axis (if desired, adapt/use the delivered test program)
2. Start the test program and test the axis
3. Circularity test

☞ Before the drive is operated in "Automatic" mode, "MDI" mode can be used to selectively start an NC block.

IW Engineering / SERCOS: Optimizing the drives

Optimize the drive with IW-Drive

☞ A final optimization of the drives should be carried out by trained personnel.

5.4 Control Commissioning

5.4.1 Overall Procedure

In general, the steps shown in the following diagram are required to add NC axes to an existing control configuration.

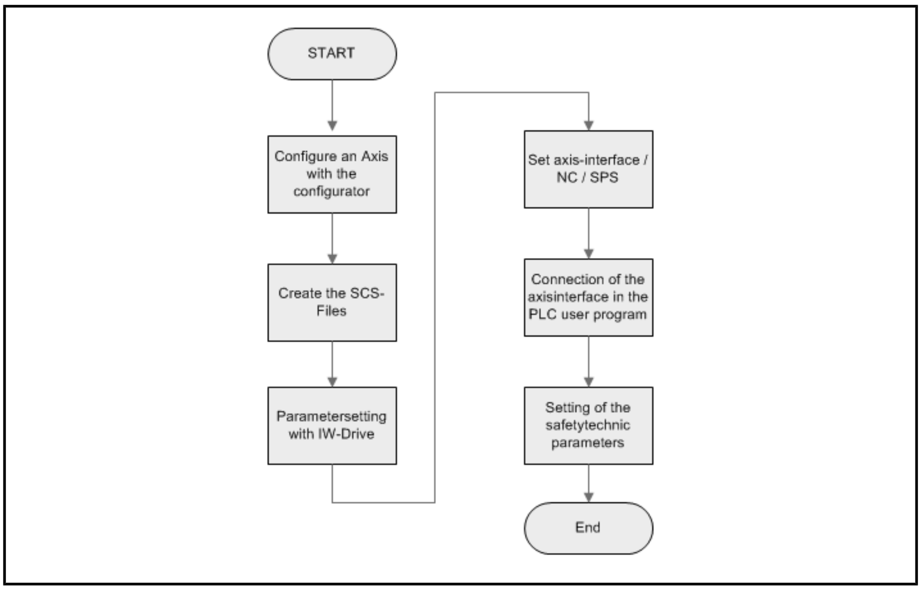


Fig.5-19: Axis commissioning procedure

5.4.2 Control-side Machine Parameters

General

Channels/axes must be created for a new project and a project extension. When the channel/axis structure is created, the parameters of the channels/axes are specified and default values are set. An axis or a channel can only be displayed or configured if a parameter set exists. Proceed as follows to configure channels/axes:

1. Open the editor for parameter configuration

Axis Commissioning

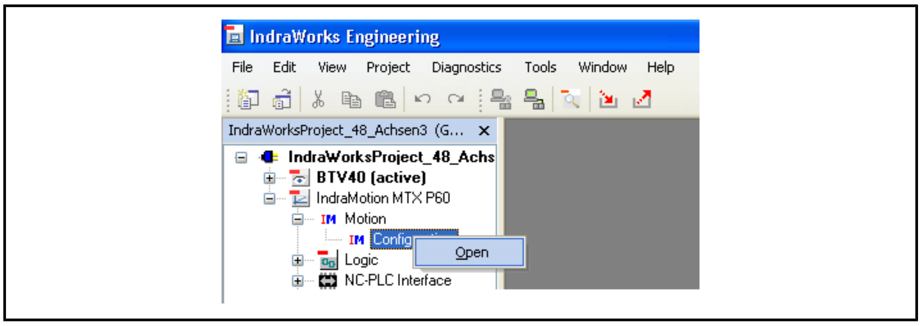


Fig.5-20: Opening the configurator

2. Create channel/axis structure

   No channels exist in a newly created IndraWorks project. Before axis parameters can be configured, the channel must be configured.

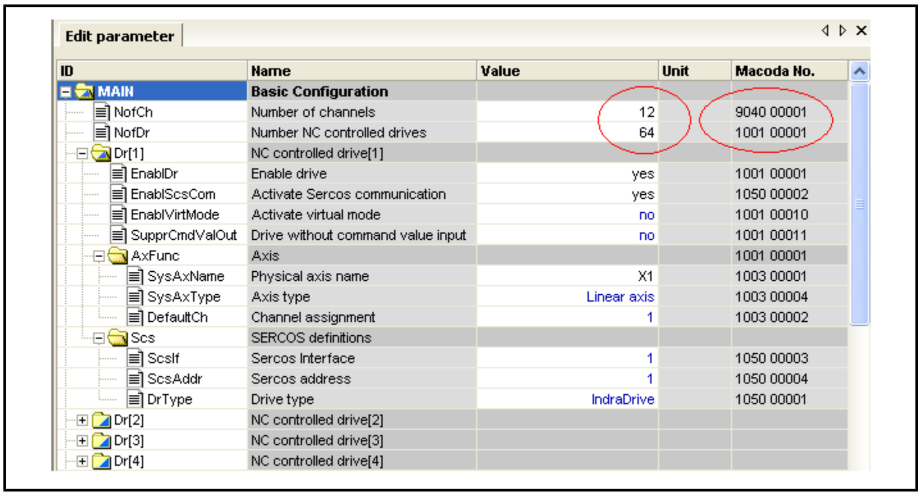


Fig.5-21: Setting the number of channels and axes

3. To set the number of channels/axes, modify the entry accordingly (see fig. 5-21 " Setting the number of channels and axes" on page 86). The channels and axes are automatically created by the system.
4. Add channel parameter structure.
5. Press the right mouse button on the "CHAN" node and a menu opens. In this menu, a channel can be created/added under "New".

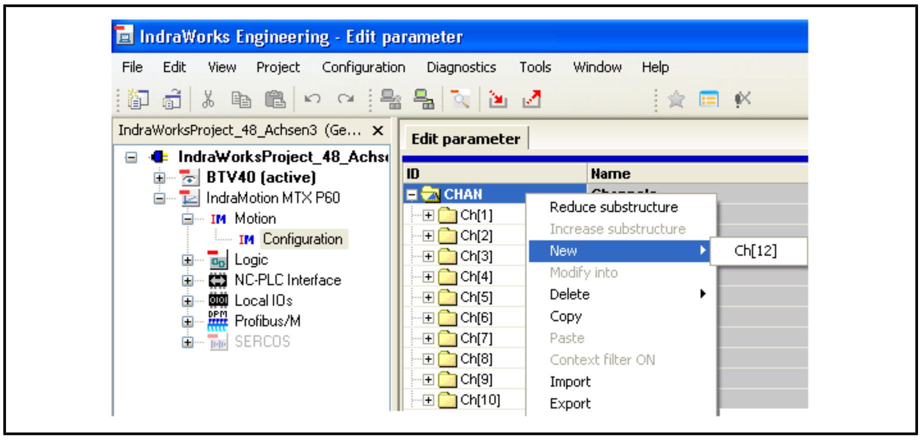


Fig.5-22: Creating a channel structure

6. Add axis parameter structure

If channels exist, axes can be created below the "MAIN" node. This is accomplished by clicking on the "MAIN" node with the right mouse button. During the axis selection, ensure that the correct axis type is configured. The following axis types are available in the menu:

- AxFunc[i] for linear and rotary axes (i = {0, 1, ..., 64})
- SpFunc[j] for spindles (j = {0, 1, ..., 32})
- AxFunc/SpFunc[i] for spindles/c-axes (i = {0, 1, ..., 64})

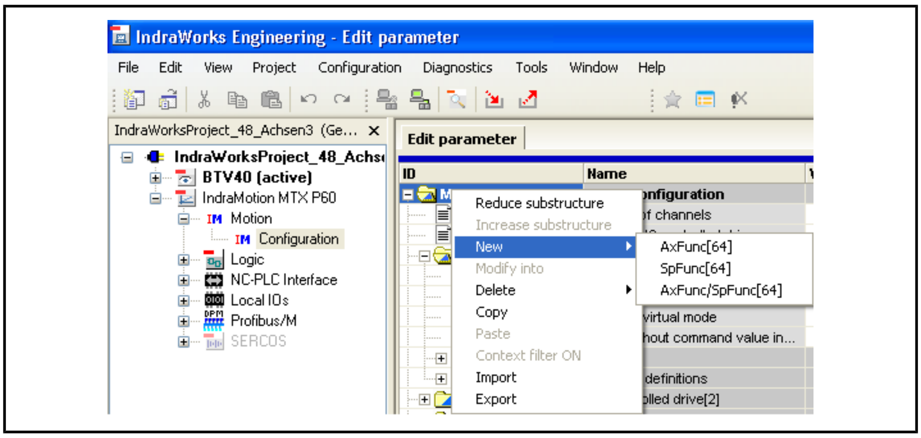


Fig.5-23: Creating an axis parameter structure

7. The NC must be restarted to transfer the parameters.

   The NC is restarted by clicking the right mouse button on the "IndraMotion MTX P60\Motion" node in the project tree of the Engineering desktop.

Handling Instruction: Creating a channel/axis

Add a drive to the created/restored project. The general operation of IndraWorks is described in the manual "Bosch Rexroth IndraMotion MTX IndraWorks".

IW-Engineering/configuration: Create a channel/axis

1. Create a channel
2. Create an axis/spindle

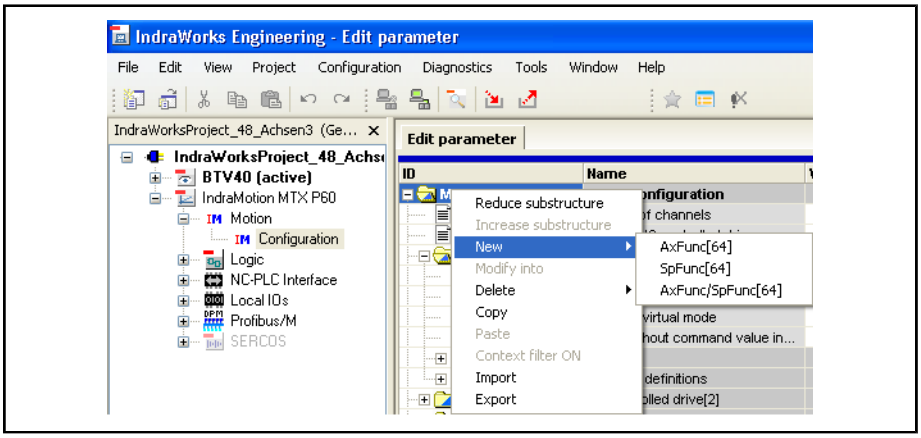


Fig.5-24: Creating a channel/axis

☞ An axis can only be created if a channel was created before.

		Documentation
Documentation:	IndraWorks Engineering	Working with Projects and Devices

Handling Instruction: Configuring Axis Parameters

The relevant axis parameters for the relevant machine data is checked or the recorded value is entered.

1. Set the NC parameters according to the machine data.
2. Check and, if necessary, modify the axis parameters and SERCOS parameters.

☞ No double addresses must be assigned to the parameter [Addr]. The addresses must be identical to the physical drive addresses.

Default settings can be retained.

3. Restart the control after the parameters have been checked.

IW Engineering/Configuration: Edit Parameters

1. Open the configuration tool to edit the parameters.
2. Check/edit the following parameters:
   - **SysAxType** "Axis type" (1003 00004)
   - **SysAxName** "Physical axis name" (1003 00001)
   - **TravLim(1) / TravLim(2)** "Travel distances"
   - **MaxVel** "Maximum velocity" (1005 00002)
   - **MaxAcc** "Maximum acceleration" (1010 00001)
3. Edit the parameters for SERCOS master communication
4. Edit axis/spindle parameters

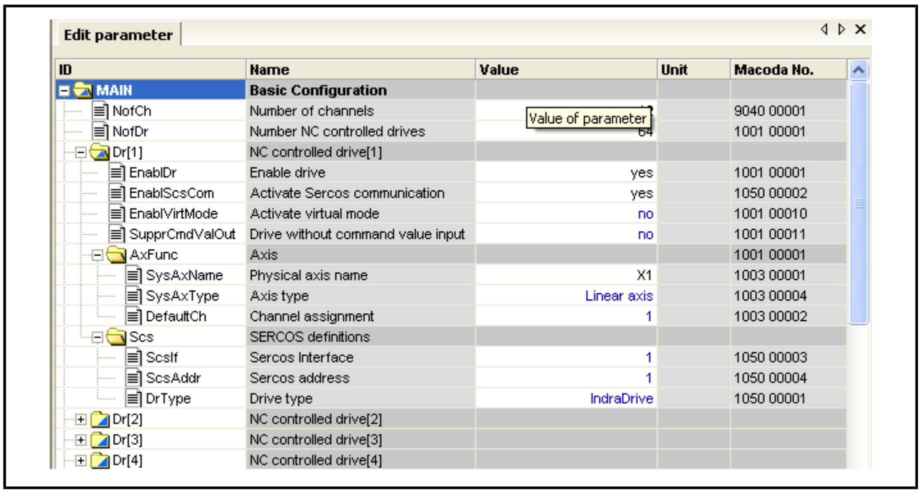


Fig.5-25: Configurator for axis parameters (AX)

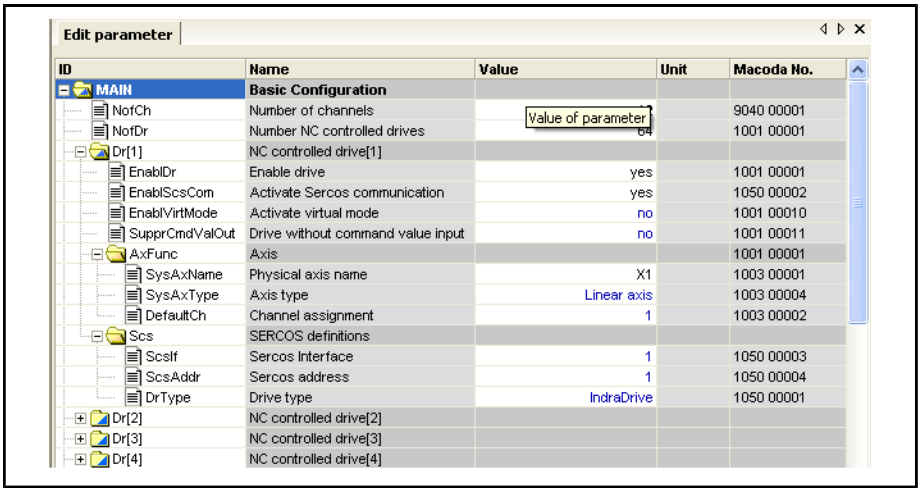


Fig.5-26: Configurator for spindle parameters (SP)

To avoid dangerous situations, check all parameters for correctness.

		Documentation
Documentation:	MTX Configuration	NC Configurator

IW Engineering/Configuration: Applying Parameters

Restart the NC to transfer the modified NC parameters.

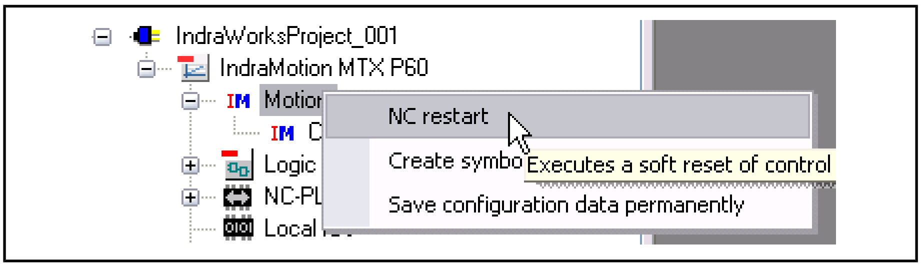


Fig.5-27: NC restart

When using the NC emulation:

Exit EMU with a value of 14 and restart.

		Documentation
Documentation:	MTX Configuration	Edit machine parameters

5.4.3 SCS Handling

General

The NC provides a number of functions which significantly simplify handling with the connected SERCOS drives.

- SERCOS initialization for all connected SERCOS drives during control startup.
- Automatic parameter download for all connected SERCOS drives during control startup.
- Read-only access to the SERCOS parameters of all connected drives using the CPL commands SCS (..) and SCSL (..) under program control.

- Write access to the SERCOS parameters of all connected drives by "WriteId" (WID) under program control.
- Selectable debug functions can be used to log the following functions when starting up the NC:
  - The SERCOS parameters transmitted in the phases 2 and 3.
  - The SERCOS timing in phase 3.

☞ Detailed information on these CPL commands can be found in the "IndraMotion MTX Programming Manual".

SERCOS Initialization

The tasks of the SERCOS initialization of the NC are as follows:

- Closing of all SERCOS rings. Thus, it is checked whether the OWG transfer lines are OK.
- Establishment of the SERCOS timing and the communication paths to the connected drives.
- Starting up the phases of all SERCOS drives up to phase 4 (cyclic operation is activated).

☞ SERCOS initialization can be executed either with or without automatic parameter downloading to the drive

Relevant configuration parameters:

SpInd		(1001 00001)
ENAVirtMode		(1001 00010)
SysAxName	"Physical axis name"	(1003 00001)
DefaultCh	"Channel assignment"	(1003 00002)
SysAxType	"Axis type"	(1003 00004)
DrType	"Drive type"	(1050 00001)
ENAScsCom	"Enable drive in the ring"	(1050 00002)
ScsIf	"Number of the SERCOS ring"	(1050 00003)
ScsAddr	"SERCOS address"	(1050 00004)
KindAutoTime[1)]	"Type of automatic transmission time calculation"	(1050 00010)
TransTime (see footer 1)	"Transmission point in time of drive telegrams"	(1050 00011)
TransTimeMdt (see footer 1)	"Transmission point in time of master data telegrams"	(1050 00012)
EnableTestMode	"Enable Test mode"	(1050 00020)
ScsTimeOut	"Time monitoring for SERCOS startup"	(1050 00021)
ModeScsFiles	"Download SERCOS data in phase 3"	(1050 00022)
OpticTransPow (see footer 1)	"Optical transmission power state"	(1050 00031)

1) *Parameter is only supported by SERCOS 2*

ScsBaudRate (see footer 1)	"SERCOS transmission rate"	(1050 00032)
ScsCycTime	"SERCOS cycle time"	(9030 00001)

Configuring SCS Files

Starting parameterization and initialization:

- Switch off the power supply to the drives!
- The listed parameters must be adapted to the application.

  The "Automatic Calculation of Transmission Time" (1050 00010) function is used to simplify the SERCOS commissioning. When "Automatic Calculation of Transmission Time" is active, the following applies:

  – During startup, the NC automatically calculates the SERCOS transmission points in time T1 to T4 from the telegram lengths (AT, MDT) and from the drive protection times (T1min, TATMT, TMTSG, T4min).

Abbreviation	Name
AT	Drive telegram
MDT	Master data telegram
TATMT	
TMTSG	

- The following are no longer applicable:

  TransTime

  "Transmission Point in Time of Drive Telegrams" (1050 00011)

  TransTimeMdt

  "Transmission Point in Time of Master Data Telegrams" (1050 00012)
- The configured SERCOS addresses must be identical to the SERCOS addresses set on the drives. To do this, refer to the documentation on the drives.
- Trigger control reset

☞ When parameter downloading is active, the current drive parameters are overwritten by parameter values in the SERCOS files of the NC. Failing to adapt the SERCOS files may result in unintended drive reactions or dangerous states of the machine.

For this reason, deactivate automatic parameter downloading for all drives for which there are presently no adapted SERCOS files in the NC.

- After the control startup, check the SERCOS phase of the drives on the drive itself. Ideally, the drives should display phase "4".

  If phase "4" is not displayed, the following has to be considered:

  – The drive displays phase "0":

  The NC cannot close the ring (the transmitted MST does not return to the NC). Check whether all SERCOS participants in the ring are activated, the OWG connections have been installed correctly, and the optical transmission power of the NC [**OpticTransPow** "Optical Transmission Power Stage" (1050 00031)] and of all modules is sufficient for the ring length used.

– The drive displays phase "1":

  The NC cannot find a specified drive. Check whether all SERCOS addresses parameterized in the NC correspond to those of the drives.

– The drive displays phase "2":

  The NC was not able to establish a communication path or a correct timing to the drive, or the drive could not be switched to phase "3" due to a parameterization problem. Check the drive parameterization (if automatic parameter downloading is activated, check the respective SERCOS file for phase 2 in the NC. If applicable, activate the logging function for transmitted SERCOS parameters - see chapter "Logging Transmitted SERCOS Parameters" on page 95).

– The drive displays phase "3":

  The drive could not be switched to phase 4 due to a parameterization or timing problem. The drive parameterization must be checked (if automatic parameter downloading is activated, check the respective SERCOS file for phase 3 in the NC. If applicable, activate the logging function for transmitted SERCOS parameters - see chapter "Logging Transmitted SERCOS Parameters" on page 95).

☞ If the parameters of drives are not set correctly, the power supply must remain switched off.

Creating SERCOS Files (SCS Files)

Automatic parameter downloading is always activated (by default) in the IndraMotion MTX NC control. When parameter downloading is active, the NC is able to parameterize connected drives completely while it starts up. As a precondition, correctly adapted SERCOS files (*.scs) must exist in the NC. SERCOS files are always saved in ASCII format.

The two files "load_ph2.scs" and "load_ph3.scs" have a central significance for the parameter download. In these files, specify which SERCOS files are to be transmitted to which drives by means of the "use..." command. Here,

- the file "load_ph2.scs" is responsible for transmission in phase 2 and
- the file "load_ph3.scs" is responsible for transmission in phase 3

Syntax of the "use" command:

use <FileName> **for** (<Number>[,<Number>])

- use

  Download command

  – If the command is in "load_ph2.scs", the content of <FileName> is sent to phase 2.

  – If the command is in "load_ph3.scs", the content of <FileName> is sent to phase 3.

- <FileName>

  SERCOS file whose content is to be transmitted to the drive <Number>.

- <Number>

  System drive index (If <FileName> is to be transmitted to several drives, separate the individual numbers by commas.)

Example:

Excerpt from the file "load_ph2.scs"

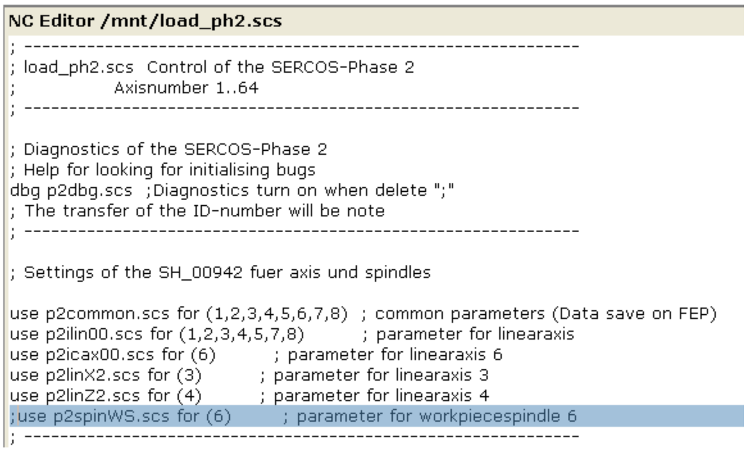
use p2ilin00.scs for (1,2,3) ; Parameter file for linear axis

use p2ilin01.scs for (4,9) ; Parameter file for a linear axis with an external encoder

☞ The parameter **ModeScsFiles** "Download SERCOS Data in Phase 3" (1050 00022) specifies the behavior of the NC for all SERCOS files entered in "load_ph3.scs". If hiding is permitted, the files are sent to the corresponding drive only when required.

All files which can be used for automatic downloading in "load_ph2.scs" and "load_ph3.scs" via the "use" command may contain

- comment lines and
- lines for the drive parameterization.



```
NC Editor /mnt/load_ph2.scs
; ----------------------------------------------------------------
; load_ph2.scs  Control of the SERCOS-Phase 2
;             Axisnumber 1..64
; ----------------------------------------------------------------

; Diagnostics of the SERCOS-Phase 2
; Help for looking for initialising bugs
dbg p2dbg.scs  ;Diagnostics turn on when delete ";"
; The transfer of the ID-number will be note
; ----------------------------------------------------------------

; Settings of the SH_00942 fuer axis und spindles

use p2common.scs for (1,2,3,4,5,6,7,8)  ; common parameters (Data save on FEP)
use p2ilin00.scs for (1,2,3,4,5,7,8)        ; parameter for linearaxis
use p2icax00.scs for (6)         ; parameter for linearaxis 6
use p2linX2.scs for (3)        ; parameter for linearaxis 3
use p2linZ2.scs for (4)        ; parameter for linearaxis 4
;use p2spinWS.scs for (6)       ; parameter for workpiecespindle 6
; ----------------------------------------------------------------
```

Fig.5-28: Example of a SERCOS file

Comment lines **always** start with a semicolon (;). The NC interprets all characters from the semicolon up to the end of the line as comment. Comments are not transferred to the drives. They are used to improve structure and clarity.

Lines for the drive parameterization are structured as follows:

<ID Number> = <Value> [;<CommentText>]

The following applies:

- <ID Number>:

  SERCOS parameter in the format S-x-xxxx or P-x-xxxx (S- and P-parameters). See the drive documentation for the available SERCOS parameters.
- <Value>:

  Parameter value in the following formats:

  Decimal: e.g. 500

  Binary: e.g. 0b ...

  String: e.g. "Text"

  Parameter list: e.g. (S-0-0047, S-0-0189)
- <CommentText>:

Comment. It must be separated from <Value> by at least one space and one semicolon.

Example:

;This is a comment line (comment)

S-0-0121 = 1 ;Input revolutions (parameter with com.)

S-0-0122 = 1 (decimal value)

S-0-0032 = 0b0011 (binary value)

S-0-0142 = "Application block" (string)

S-0-0016 = (S-0-0051, S-0-0189) (ID list)

If **all** adapted SERCOS files (*.scs) required for your application are already available, copy all relevant SERCOS files into the "root directory" ("/") of the NC (for the directory structure, see the "IndraWorks" documentation). To activate and deactivate, proceed as described in the following sections.

If **no** SERCOS files (*.scs) have been generated for your application, create new files in the "root directory" ("/") of the NC by means of the editor, or copy the example files enclosed in the delivery into the "root directory" ("/") and adapt them to your requirements (see Handling Instruction). The example files are located in the directory "//root/usrfep/".

Activate:

1. Remove the character ";" in the files "load_ph2.scs" and/or "load_ph3.scs" to the left of the relevant "use" commands.

   As a result, the NC will no longer interpret these lines as comment lines but as download commands the next time it is started.

2. Save the modified file(s).
3. Create a backup copy of the file(s)

Example:

Automatic download (in phase 2) of the files

- "p2linall.scs" in the drives with the system drive numbers 1, 2 and 4
- "p2lin3.scs" in the drive with the system drive number 3.

Excerpt from the file "load_ph2.scs"

;

use p2linall.scs for (1,2,4)

use p2lin3.scs for (3)

;

Deactivate:

1. Set the character ";" in the files "load_ph2.scs" and/or "load_ph3.scs" to the left of the relevant "use" commands.

   As a result, the NC will no longer interpret these lines as download commands, but as comment lines the next time that it is started.

2. Save the modified file(s).
3. Create a backup copy of the file(s)

Example:

Automatic downloading is to be deactivated for both files from the example shown above:

Excerpt from the file "load_ph2.scs"

;

;use p2linall.scs for (1,2,4)

;use p2lin3.scs for (3)

;

☞ If backup copies of all relevant SERCOS files exist in the "user FEPROM" ("/usrfep"), it is not sufficient to merely delete or rename the relevant SERCOS files in the "root directory" ("root") to deactivate automatic parameter downloading. The NC also searches for these files in the "user FEPROM (usrfep)" if it cannot find them in the "root" directory.

Logging Transmitted SERCOS Parameters

All SERCOS parameters transmitted in phase 2 or phase 3 can be logged for diagnostic purposes. To do so, program the "dgb" command in the "load_ph2.scs" file (for phase 2) or in the "load_ph3.scs" file (for phase 3).

Prerequisite:

The "dbg" command has to be programmed prior to the first "use" command.

Syntax of the "dbg" command:

dbg <FileName>

- dbg

  "Create log" command

  – If the command is in "load_ph2.scs", the logging function is activated in phase 2.

  – If the command is in "load_ph3.scs", the logging function is activated in phase 3.

- <FileName>

  Name of the file which is to be in the log.

Activate:

1. The function can be activated by removing the character ";" to the left of the "dbg" command in the file "load_ph2.scs" or "load_ph3.scs".

   As a result, the NC will no longer interpret the line as a comment line, but as a command to log the SERCOS parameters the next time that it is started.

2. The file must be saved when exiting the editor.

Deactivate:

1. The function can be deactivated by setting the character ";" to the left of the "dbg" command in the file "load_ph2.scs" or "load_ph3.scs".

   Thus, the NC will interpret the line as a comment line the next time that the control starts up.

2. The file must be saved when exiting the editor.

☞ The diagnostic function is already in the SERCOS files delivered by Bosch Rexroth.

Logging SERCOS Timing

For diagnostic purposes, the NC can log the SERCOS timing starting from phase 3. To do so, program the "opt" command in the "loadph3.scs" file.

Prerequisite:

The "opt" command has to be programmed after the last "use"command.

"opt" command syntax:

opt -m <FileName>

- opt -m

  "Logging of SERCOS Timing" command
- <FileName>

  Name of the file which is to be in the timing log.

☞ This function should only be used during commissioning!

Activate:

1. The function can be activated by removing the character ";" to the left of the "opt" command in the file "load_ph3.scs".

   As a result, the NC will no longer interpret the line as a comment line, but as a command to log the SERCOS timings the next time that it starts up.
2. The file must be saved when exiting the editor.

Deactivate:

1. The function can be deactivated by setting the character ";" to the left of the "opt" command in the file "load_ph3.scs".

   Thus, the NC will interpret the line as a comment line the next time that the control starts up.
2. The file must be saved when exiting the editor.

Handling Instruction Create SCS files

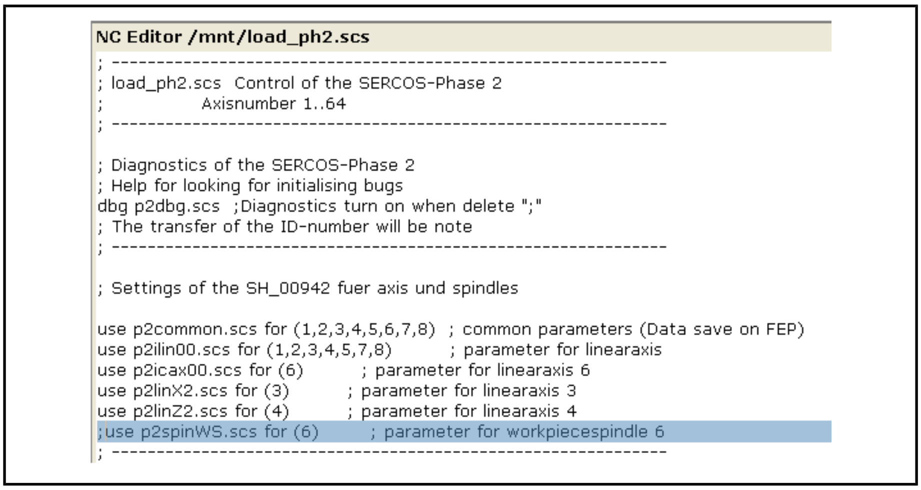
IW Operation / Program (OP4): Adapt SCS files

1. Select root directory
   - Required SCS files:
   - load_ph2.scs
   - load_ph3.scs
   - p2common.scs
   - p2xxxx000.scs – This file can have another name, depending on the drive.
2. Copy file "p2xxx000.scs" as many times as there are drives in the ring and name them according to the names of the drives.

   Identical drive types can be addressed using one SCS file.
   - lin ;for linear axis
   - rot ;for rotary axis
   - spin ;for spindle

- cax ;for the c-axis (spindle with rotary axis functions)

3. Open the SCS files by double-clicking on them and adapt them



```
NC Editor /mnt/load_ph2.scs
; ---------------------------------------------------------------
; load_ph2.scs  Control of the SERCOS-Phase 2
;           Axisnumber 1..64
; ---------------------------------------------------------------

; Diagnostics of the SERCOS-Phase 2
; Help for looking for initialising bugs
dbg p2dbg.scs  ;Diagnostics turn on when delete ";"
; The transfer of the ID-number will be note
; ---------------------------------------------------------------

; Settings of the SH_00942 fuer axis und spindles

use p2common.scs for (1,2,3,4,5,6,7,8)  ; common parameters (Data save on FEP)
use p2ilin00.scs for (1,2,3,4,5,7,8)        ; parameter for linearaxis
use p2icax00.scs for (6)        ; parameter for linearaxis 6
use p2linX2.scs for (3)        ; parameter for linearaxis 3
use p2linZ2.scs for (4)        ; parameter for linearaxis 4
;use p2spinWS.scs for (6)       ; parameter for workpiecespindle 6
; ---------------------------------------------------------------
```

Fig.5-29: Example of a SERCOS file

The SCS templates (files) are stored in the root directory of the control.

The file "load_ph3.scs" primarily consists of remarks; only one log file is activated with the entry "dbg p3dpg.scs". This diagnostic file logs the data transfer to the drive. This diagnostic file can also be activated in the SCS file "load_ph2.scs".

If it is stored elsewhere in the control's file structure, ensure that the data is not accidentally overwritten in another location.

Handling Instruction: Activating an NC restart

After the SCS files have been successfully configured, an NC restart must be activated so that the control can read the current SCS files and start up the drives.

IW Engineering/Configuration: Activate an NC restart

Activating an NC restart

Only after the NC control is restarted, the modified data (parameters, SCS files) is applied.

5.5 PLC Commissioning

5.5.1 Activating Axes/Spindles in the PLC Application Program

General

The required connection of the inputs and outputs on the axis and the channel interface and the creation of the axis and channel interface are described in this handling instruction.

A detailed description is given in chapter 8 "Configuration of PLC-Specific Data in IndraWorks" on page 221.

Handling Instruction: Creating a Channel/Axis/Spindle Interface

This handling instruction describes the creation of a channel/axis/spindle interface in IndraLogic.

IW-Engineering/IndraLogic: Create a channel interface

This handling instruction describes how the user can create and connect a channel interface.

		Instruction chapter 8.1.3 "Channel Interface " on page 222
Instruction:	PLC data configuration	Channel interface

IW-Engineering/IndraLogic: Create an AxisInterface

This handling instruction describes how the user can create and connect an AxisInterface.

		Instruction chapter 8.1.4 "Axis Interface" on page 223
Instruction:	PLC data configuration	AxisInterface

IW-Engineering/IndraLogic: Create a spindle interface

This handling instruction describes how the user can create and connect a spindle interface.

		Instruction chapter 8.1.5 "Spindle Interface" on page 225
Instruction:	PLC data configuration	Spindle interface

5.6 Offline Parameterization

Instead of parameterizing the drive parameters online, they can also be parameterized offline. Therefore, the control has to be switched to offline parameterization.

The offline parameterization is only relevant for drive data. Control data cannot be parameterized offline.

Prerequisites:

To use the offline parameterization, offline data has to be generated first.

That can be carried out as follows (MTX is online):

- Call the function **Update Offline Parameters** at the MTX node
- Call the function **Adjust Offline Data** at the drive node
- Going offline with the MTX

  When going offline with the MTX, the data of all drives online at this point in time are compared with each other. This function depends on the setting in the "Options" dialog (see fig. 5-30 ""Option" dialog: Offline parameterization of the MTX" on page 99).

Data adjustment

When going online with the drive, all possibly existing offline data is compared to the data in the drive. Differences are displayed and can be applied in the drive or in the offline parameterization. This behavior can be switched off in the "Options" dialog (see fig. 5-30 ""Option" dialog: Offline parameterization of the MTX" on page 99).

Options

To set up the offline parameterization, there are two settings in the "Options" dialog (**Tools ▸ Options**)

Checks can be switched off when going online/offline.

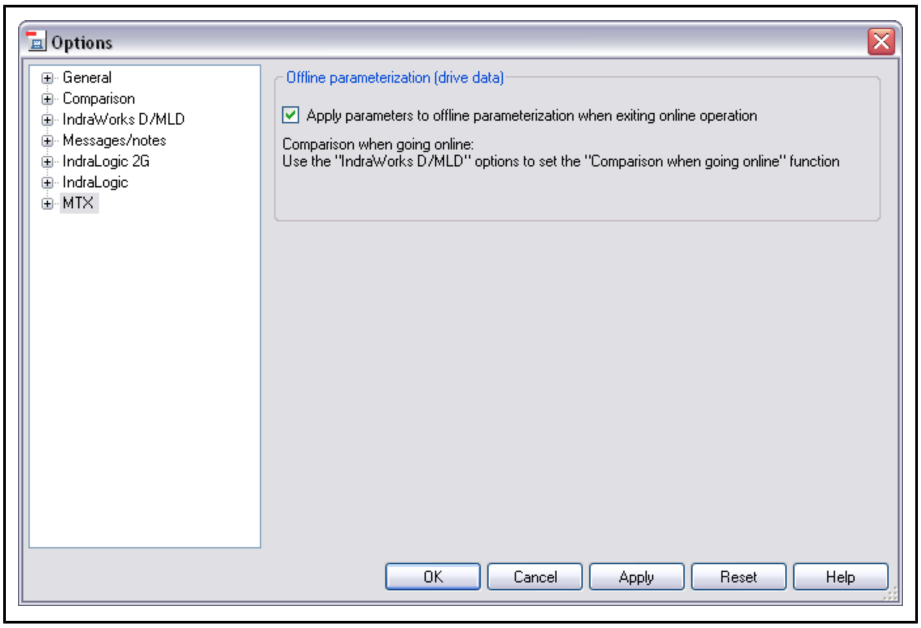


Fig.5-30: "Option" dialog: Offline parameterization of the MTX

Using the offline parameterization

First, the control has to be switched to offline parameterization.

The individual drives can also be switched to offline parameterization via mouse click or via the function **Drive Communication Online/Offline**. The drive data can also be archived.

☞ Other options of the the offline parameterization of a drive are described more in detail in the drive documentation.

6 HMI Setup

6.1 General

6.1.1 What is an F-Key?

F-key = function key

The F-keys are located on the lower edge of a screen, directly above the operation area (OP) keys. The individual key of an F-panel is called the F-key. An F-key is always part of an F-panel.

Fig.6-1: F-key

6.1.2 What is an F-Panel?

F panel = function keypad

A field of 8 F-keys is called an F-panel. Each F-panel can be assigned to one or more screens.

Fig.6-2: Example F-panel

6.1.3 What is an M-Key?

M-key = machine key

A single key to the left or to the right of the BTV is called an M-key. It is always part of an M-panel.

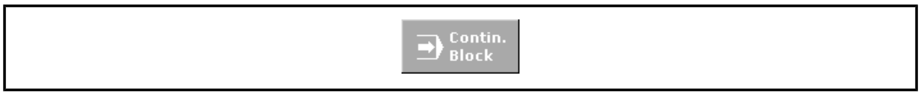


Fig.6-3: Example M-key

Often, all functions are referred to as M-keys (e.g. when creating the local I/O).

6.1.4 What is an M-Panel?

M-panel = machine keypad

A field of 8 M-keys is called an M-panel. Each M-panel can be assigned to one or more screens.

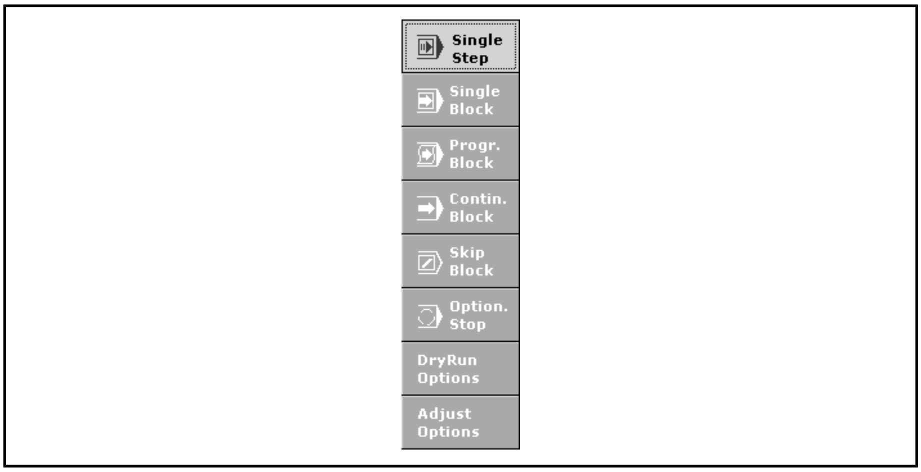


Fig.6-4: Example M-panel

6.2 Channels

6.2.1 Transferring an Active Channel Number to the PLC

General

Brief Description -

Description The MTX can control up to 12 channels. The states of a channel are displayed in the Operation Desktop, the MTX user interface. Using subsequent commands, the PLC is informed which channel is currently active in the Operation Desktop.

Boundary Conditions -

Example -

Handling Instruction: Transferring an Active Channel Number to the PLC

This handling instruction describes the procedure for transferring to the PLC the number of the channel that is displayed in the Operation Desktop as being active.

IW-Engineering/IndraLogic: Generate channel number variables

1. Call IndraLogic
2. Generate channel number variables

☞ A variable can be generated as a global or a local variable. Any instance name can be used.

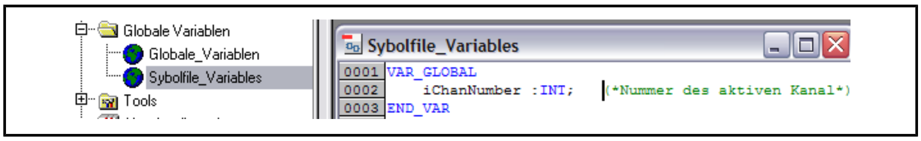


Fig.6-5: Example for generating a variable

		Documentation
Documentation:	IndraLogic Programming	Generating variables

IW-Engineering/IndraLogic: Export an instance path to a symbol file

Mark the program or the list of global variables with "Output Variables of the Object".

The declaration must not have the pragma ({flag noread, nowrite}).

fig. 6-13 "Procedure for configuring the symbol file" on page 113		Documentation
Figure:		Symbol configuration procedure
Documentation:	IndraLogic Programming	Options for icon configuration

IW-Engineering/IndraLogic: Transmit all

		Documentation chapter 24.7 "Transmitting and Activating a PLC Project" on page 571
Documentation:	IndraWorks Commissioning	Transmit and activate a PLC project

IW-Engineering/IndraLogic: Log in

		Documentation chapter 15.2 " User Log-on/Log-off" on page 291
Documentation:	IndraWorks Commissioning	Log in

IW-Engineering/IndraLogic: Start PLC program

		Documentation chapter 24.7 "Transmitting and Activating a PLC Project" on page 571
Documentation:	IndraWorks Commissioning	Start PLC program

IndraWorks Engineering / HMI: Announce the channel number variable of the HMI

1. Switch to IndraWorks.
2. Expand the node "MTX ScreenManager"
3. Open the "Properties" dialog of the node "Channel No. to the PLC" using its context menu.

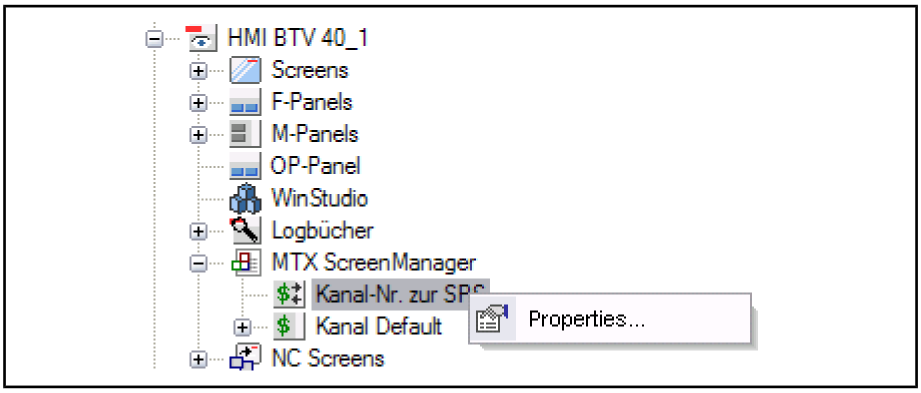


Fig.6-6: Open the "Properties" dialog

4. Enter the instance name of the variable under "Write channel No. to the PLC" in the "Properties" dialog.

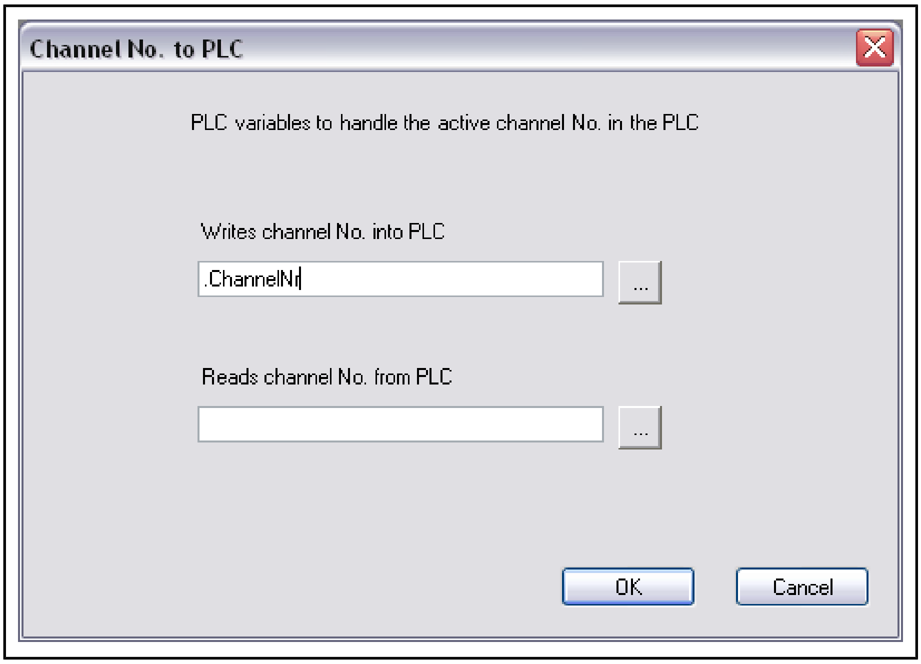


Fig.6-7: Entering the instance name of the variable

In the case of global variables, a period must be located before the variable name.

In the case of local variables, the complete instance path must be provided. No period may be located before the instance path here.

5. Confirm the dialog with "OK".

		Documentation
Documentation:	IndraWorks HMI	Announce the channel number variable of the HMI

6.2.2 Switching the Active Channel Using the PLC

General

Brief Description -

Description The MTX can control up to 12 channels. The states of a channel are always displayed in Operation Desktop, the MTX user interface. Using the subsequent instructions, the channel which is currently active in Operation Desktop can be changed via the PLC.

Boundary Conditions -

Example -

Handling Instruction: Switching the Active Channel Using the PLC

This handling instruction describes the procedure for changing the channel displayed in Operation Desktop using the PLC.

IW-Engineering/IndraLogic: Generate channel number variables

1. Start IndraLogic
2. Generate channel number variables

A variable can be generated as a global or a local variable. Any instance name can be used.

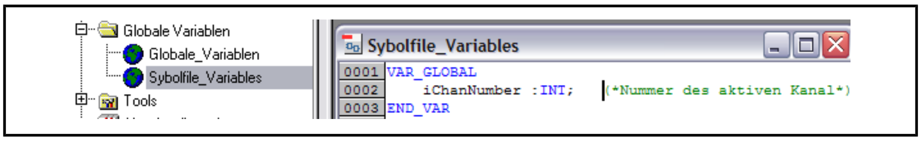


Fig.6-8: Example: Generating variables

		Documentation
Documentation:	IndraLogic Programming	Generating variables

IW-Engineering/IndraLogic: Export an instance path to a symbol file

Mark the program or the list of global variables with "Output Variables of the Object".

The declaration must not have the pragma ({flag noread, nowrite}).

fig. 6-13 "Procedure for configuring the symbol file" on page 113		Documentation
Figure:		Symbol configuration procedure
Documentation:	IndraLogic Programming	Options for icon configuration

IW-Engineering/IndraLogic: Transmit all

		Documentation chapter 24.7 "Transmitting and Activating a PLC Project" on page 571
Documentation:	IndraWorks Commissioning	Transmit and activate a PLC project

IW-Engineering/IndraLogic: Log in

		Documentation chapter 15.2 " User Log-on/Log-off" on page 291
Documentation:	IndraWorks Commissioning	Log in

IW-Engineering/IndraLogic: Start PLC program

		Documentation chapter 24.7 "Transmitting and Activating a PLC Project" on page 571
Documentation:	IndraWorks Commissioning	Start PLC program

IndraWorks Engineering / HMI: Announce the channel number variable of the HMI

1. Switch to IndraWorks
2. Expand the node "MTX ScreenManager"

HMI Setup

3. Open the "Properties" dialog of the node "Channel No. to the PLC" via its context menu.

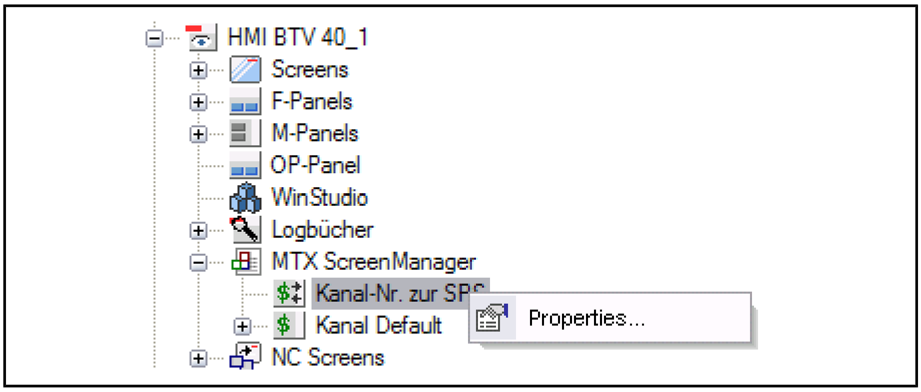


Fig.6-9: Open the "Properties" dialog

4. Enter the instance name of the variable under "Write channel No. to the PLC" and under "Read channel No. from the PLC" in the "Properties" dialog.

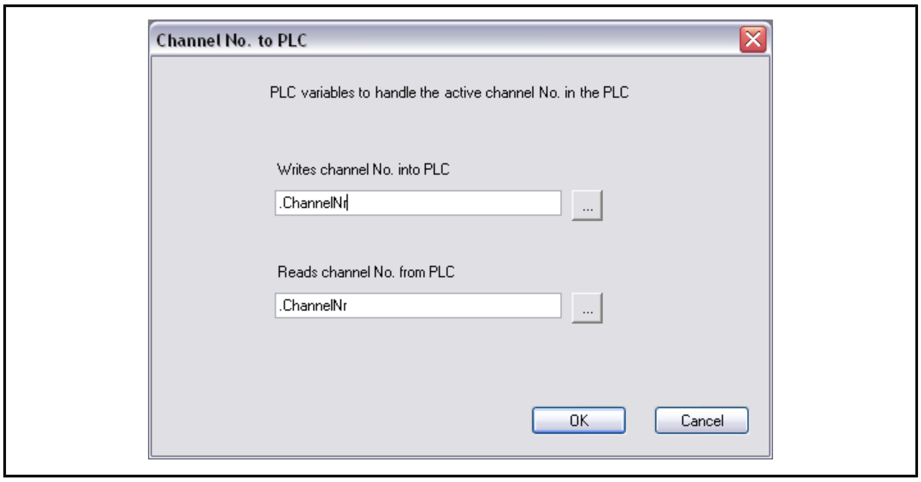


Fig.6-10: Entering the instance name of the variable

5. Confirm the dialog with "OK".

In the case of global variables, a period must be located before the variable name.

In the case of local variables, the complete instance path must be provided. No period may be located before the instance path here.

		Documentation
Documentation:	IndraWorks HMI	Announce the channel number variable of the HMI

6.3 Commissioning M-Keys

6.3.1 General

Description

Brief Description Eight machine keys (one M-panel always has eight M-keys) are located to the right and to the left of each BTV. These keys are for operating the machine; they can trigger actions using the PLC or the SCP interface.

Description In order for designed M-keys / M-panels to function, they must first be put into operation. The steps required for this are described in the following.

Boundary Conditions -

Example -

Handling Instruction: Commissioning M-Keys

This handling instruction describes which steps are required to put M-keys into operation.

IW Engineering: Define the PLC hardware inputs of the M-keys

First, the machine keys must be defined as a hardware input for the PLC.

		Documentation chapter 8.2 "Configuration of the Local Inputs" on page 229
Documentation:	IndraWorks Commissioning	Configuring the local inputs

IW-Engineering/IndraLogic: Implement FM "MKEYS"

1. Call IndraLogic
2. Implement the function module "MKEYS" in the program.
3. Connect the M-key hardware input addresses with the inputs of the function module.

☞ Ensure that the program is taken into account in the PLC program execution (e.g. calling from the main program or directly from the task).

---

fig. 6-14 "Creating a function block for M-keys" on page 114		Documentation chapter 8.2.2 "M-keys" on page 229
Figure:		Implement M-keys
Documentation:	IndraWorks Commissioning	Implement M-keys

IW-Engineering/IndraLogic: Export the "MKEYS" FM instance path to a symbol file

Mark the complete instance path of the declared function module "MKEYS" with "Output Variables of the Object".

☞ The declaration of the FM must not have the pragma ({flag noread, nowrite}).

---

fig. 6-13 "Procedure for configuring the symbol file" on page 113		Documentation
Figure:		Procedure for exporting a symbol file
Documentation:	IndraWorks Commissioning	Exporting a symbol file

IW-Engineering/IndraLogic: Transmit all

		Documentation chapter 24.7 "Transmitting and Activating a PLC Project" on page 571
Documentation:	IndraWorks Commissioning	Transmit all

IW-Engineering/IndraLogic: Log in

		Documentation chapter 15.2 " User Log-on/Log-off" on page 291
Documentation:	IndraWorks Commissioning	Log in

IW-Engineering/IndraLogic: Start PLC program

		Documentation chapter 24.7 "Transmitting and Activating a PLC Project" on page 571
Documentation:	IndraWorks Commissioning	Start PLC program

IW Engineering / M-panels: Announce the function module of the instance path

1. Open the "M-panel properties" dialog
2. Enter the instance path for the "MKEYS" FM declared in IndraLogic.
3. Close the dialog with "OK".

fig. 6-15 "M-key FM symbol file" on page 115		Documentation
Figure:		Procedure of announcing the instance path FM
Documentation:	IndraLogic Programming	FM instances

6.4 Creating an M-Key

6.4.1 General

Description

Brief Description

Eight machine keys (one M-panel always has eight M-keys) are located to the right and to the left of each BTV. These keys are for operating the machine; they can trigger actions using the PLC or the SCP interface.

Description

-

Boundary Conditions

Before the M-keys can be used, they must be put into operation.

You can find more information on M-keys in the "IndraWorks HMI" documentation.

Example -

Handling Instruction: M keys commissioning

This handling instruction describes which steps are required to create an M key.

IW Engineering / HMI: Start-up of M keys

1. Create an M panel or open an existing one.
2. Enter the control (enter * to use the first control that is found).
3. Label the M key in "Display".
4. If necessary, select a function in "Function" (see overview of M key functions).
5. If necessary, enter SPS variables "SPS flag".

If the "PLC flag" field is used, the variables entered there must be specified in the symbol file.

		Documentation
Documentation:	IndraWorks HMI	Configuring M keys (sect. 3.5)

6.4.2 Overview of M-Key Functions

When a single M-key is configured, various functions can be assigned to it.

The following table provides an overview of all available functions, what they mean and where they are described in more detail.

Function	Brief Description	Documentation
Screen change	Calls a screen	IndraWorks HMI
Level change	Calls an M-panel	IndraWorks HMI
Keyboard code trigger	Triggers a keyboard code	IndraWorks HMI
Complete lock on/off	Axis output is interrupted	
Next channel	Switches to the next channel	Functional Description IBN NewChannel
Zero position display on/off	Sets the position display to zero.	
Skip block on/off	NC blocks marked with "/" are not processed.	PLC Interface ("qCh_BlkSlash" and "iCh_BlkSlash") Functional Description
NC block search/NC restart	Calls the operating screen for NC block search and NC program restart.	
Set operation mode	Changes the operation mode	PLC Interface ("qCh_OpModeSel_00-03" and "iCh_OpModeSel_00-03")
Set channel	Change to a certain channel.	Functional Description
Test rapid traverse	Activate another velocity in rapidtraverse (G0).	PLC Interface ("qCh_TestRap" and "iCh_TestRap") Functional Description
Test feed	Activate a velocity. Programmed feeds are ignored.	PLC Interface (Ch. "qCh_TestFeed" and "iCh_TestFeed") Functional Description
TL_Delete	Deletes the selected tool data record.	

Function	Brief Description	Documentation
TL_Edit	Starts the online tool editor.	
TL_Edit_Additive	Switches between entry modes Absolute and Additive in the online tool editor and the tool list.	
TL_Export	Starts the dialog to export tool lists or individual tools.	
TL_Import	Starts the dialog to import tool lists or individual tools.	
TL_Insert	Starts the offline editor for inserting a new tool.	
TL_Move	Starts the dialog to move a tool data record.	
ToolCursor	The current cursor position (sector and location) is written to a PLC variable.	
ToolList1-16	Switches the display to the appropriate tool list.	
Switch coordinate sys.	Switches the display between the machine and the tool coordinate system.	Functional Description
Optional stop on/off	Optional stop on: NC program stops at M1 and waits for NC restart. Optional stop off: NC program ignores M1 and continues to run normally.	PLC Interface ("qCh_OptStop" and "qCh_OptStop")
Return to contour	Starts repositioning at the contour.	Functional Description
Program restart strategy	Switches to the selected strategy for restarting to contour.	Functional Description

Fig.6-11: Overview of M-key functions

6.5 Creating F-Keys

6.5.1 General

Description

Brief Description -

Description -

Boundary Conditions You can find more information on F-keys in the "IndraWorks HMI" documentation.

Example -

Handling Instruction: Commissioning F-Keys

This handling instruction describes which steps are required to create an F-key.

IndraWorks Engineering / HMI: Commissioning F-keys

1. Create an F-panel or open an existing one.
2. Enter a control
3. Label the F-key in "Display".
4. If applicable, select a function in "Function" (see chapter 6.5.2 "Overview of F-Key Functions" on page 111).

☞ If the "PLC flag" field is used, the variables entered there must be specified in the symbol file.

		Documentation
Documentation:	IndraWorks HMI	Configuring F-keys (sect. 3.3)

6.5.2 Overview of F-Key Functions

When a single F-key is configured, various functions can be assigned to it. The following table provides an overview of all available functions, what they mean and where they are described in more detail.

Function	Brief Description	Documentation
Active variables	Opens the variable editor with the currently active list of variables.	
Active D-corrections	Opens the display with the active list of D-corrections.	
Active zero point offsets	Opens the display with the active list of ZO corrections.	
Exit the interface	Closes Operation Desktop.	
Screen change	Calls a screen.	IndraWorks HMI
Level change	Calls an M-panel.	IndraWorks HMI
Editor of active program	Opens the NC program editor with the currently active program.	
I-Remote		
Commissioning	Starts the Engineering Desktop.	
Interface PLC-NC	Opens a display which shows the status of the individual interface signals between the PLC and the NC (channel, axis and spindle signals).	
Reset channel	Initiates a control reset for the active channel.	
Next channel	Switches to the next channel.	Functional Description IBN NewChannel
Zero position display on/off	Sets the position display to zero.	
Program selection	Opens the window for program selection.	
NC block search/NC restart	Calls the operating screen for NC block search and NC program restart.	
Set channel	Change to a certain channel.	Functional Description
System reset	Triggers control reset for all channels.	
Keyboard code trigger	Triggers a keyboard code.	IndraWorks HMI
TL_Delete	Deletes the selected tool data record.	
TL_Edit	Starts the online tool editor.	
TL_Edit_Additive	Switches between entry modes Absolute and Additive in the online tool editor and the tool list.	

HMI Setup

Function	Brief Description	Documentation
TL_Export	Starts the dialog to export tool lists or individual tools.	
TL_Import	Starts the dialog to import tool lists or individual tools.	
TL_Insert	Starts the offline editor for inserting a new tool.	
TL_Move	Starts the dialog to move a tool data record.	
ToolCursor	The current cursor position (sector and location) is written to a PLC variable.	
ToolList1-16	Switches the display to the appropriate tool list.	
Overview of curr. corrections	Opens a display with an overview of the active zero offsets, D-corrections and tools.	
Switch coordinate sys.	Switches the display between the machine and the tool coordinate system.	
Switch variant position display	Switches between the variants for the position display defined in Operation Desktop.	

Fig.6-12: Overview of F-key functions

6.6 Graphic Links for Chapter 06_HMI-Setup

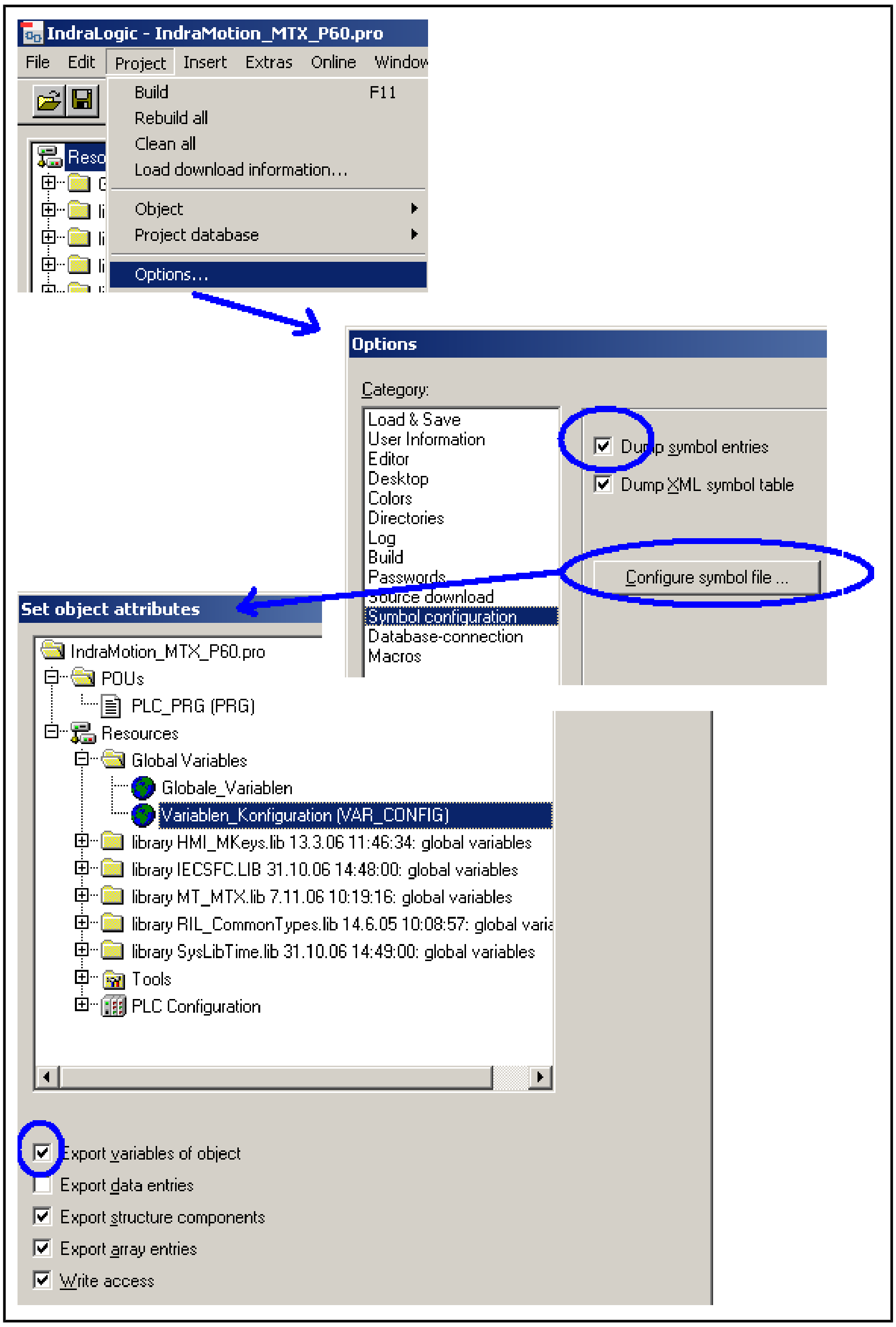


Fig.6-13: Procedure for configuring the symbol file

HMI Setup

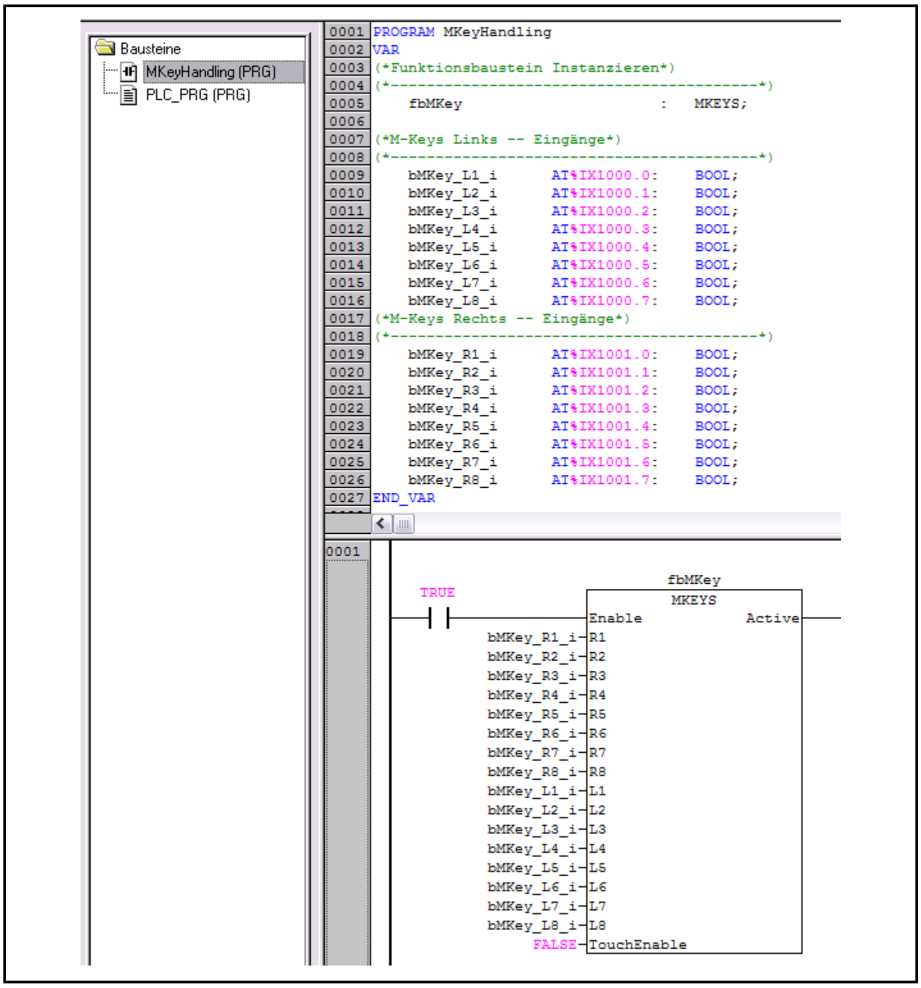


Fig.6-14: Creating a function block for M-keys

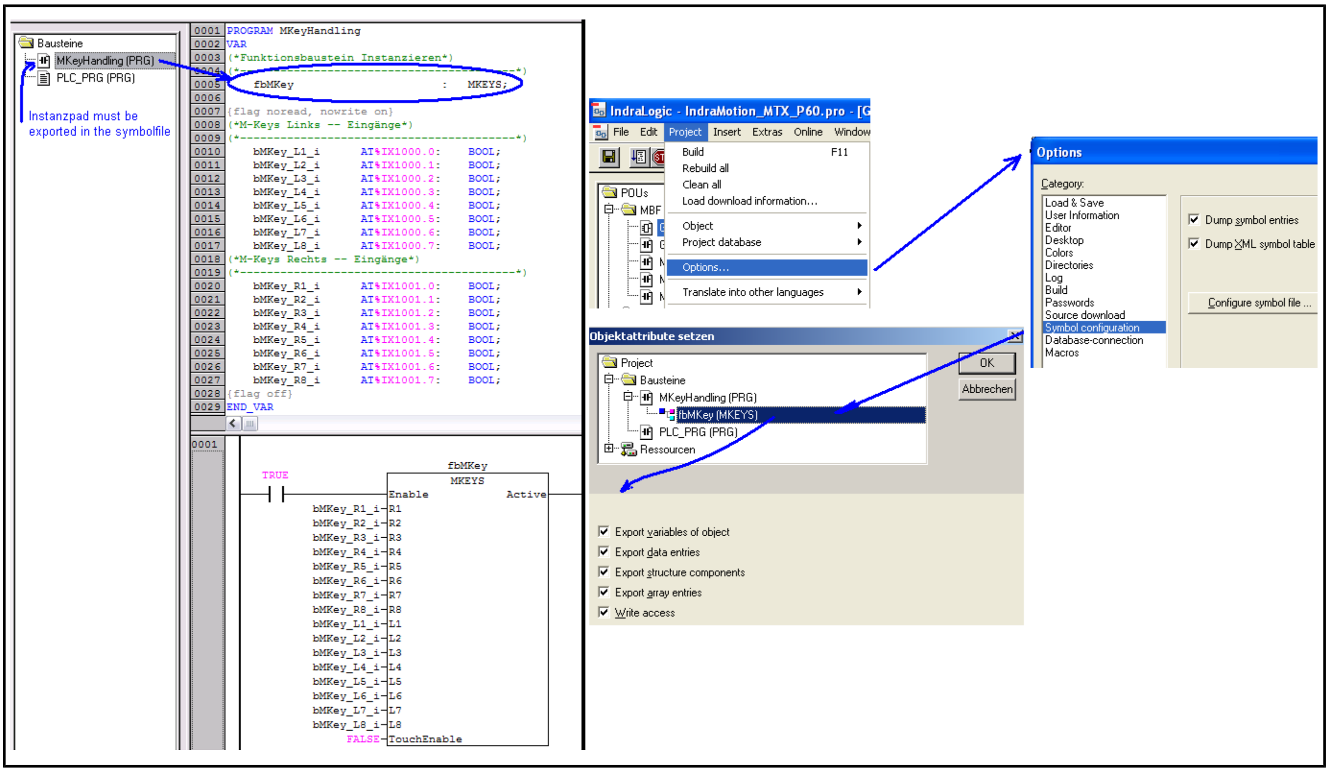


Fig.6-15: M-key FM symbol file

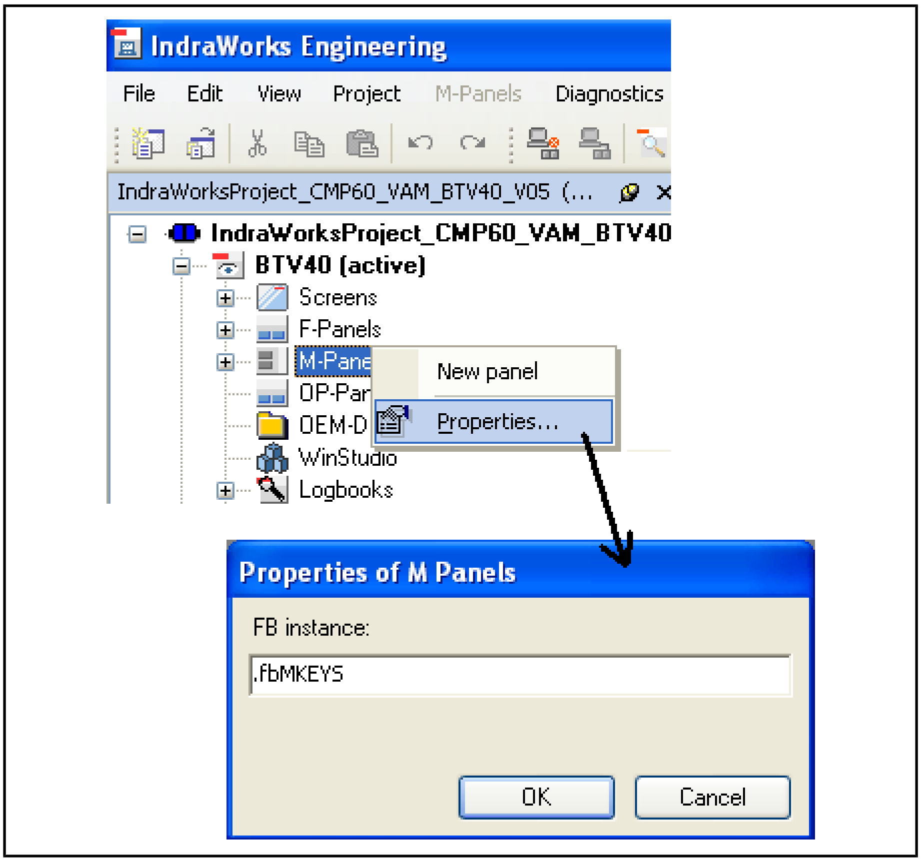


Fig.6-16: Characteristics of M-key

7 Configuring the User Interface

7.1 Configuring ACI Screens

7.1.1 General

Short description

In den ACI (Application Container Interface) screens, controls created by the customers (user controls) and delivered for the system installation, are embedded and allocated in screen segments. The ACI screens can be integrated as screens of type 'UserACIScreen' in the HMI screens.

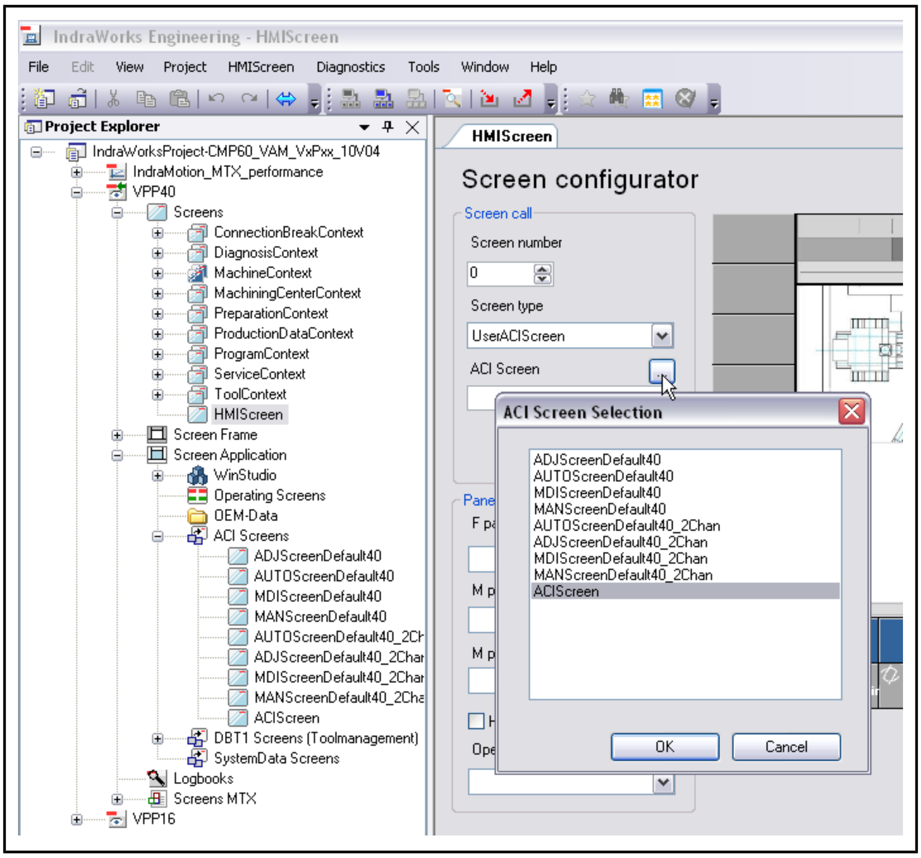


Fig. 7-1: Dialog to add a UserACIScreens to an HMI screen

Description

As it is the case for the user screens created in WinStudio, the displays of a wide variety of process data can be summarized in the ACI screens. This is carried out by distributing an ACI screen in screen segments and adding the respective control.

The various process data of the control is shown using separate displays (e.g. position display). These are called controls. These controls are implemented as software ActiveX or .NET controls. To provide the controls with an advanced functionality within the ACI screens, they must have implemented the type library "MTXACIInterface.dll" as interface. This is used to report additional events from the ACI screen and to activate commands. The advanced controls are called ACI Controls.

The distribution, selection and arrangement of the controls of the existing ACI screens can be changed using configuration dialogs. New ACI screens can be created and ACI screens that are no longer needed can be deleted.

7.1.2 Configuration in Engineering Desktop

General

Short description In the Engineering Desktop, existing ACI screens can be modified and deleted and new ACI screens can be created using appropriate configuration dialogs and functions that can be activated directly.

Description The ACI screens are managed in the Engineering Desktop under the project node "ACI Screens".

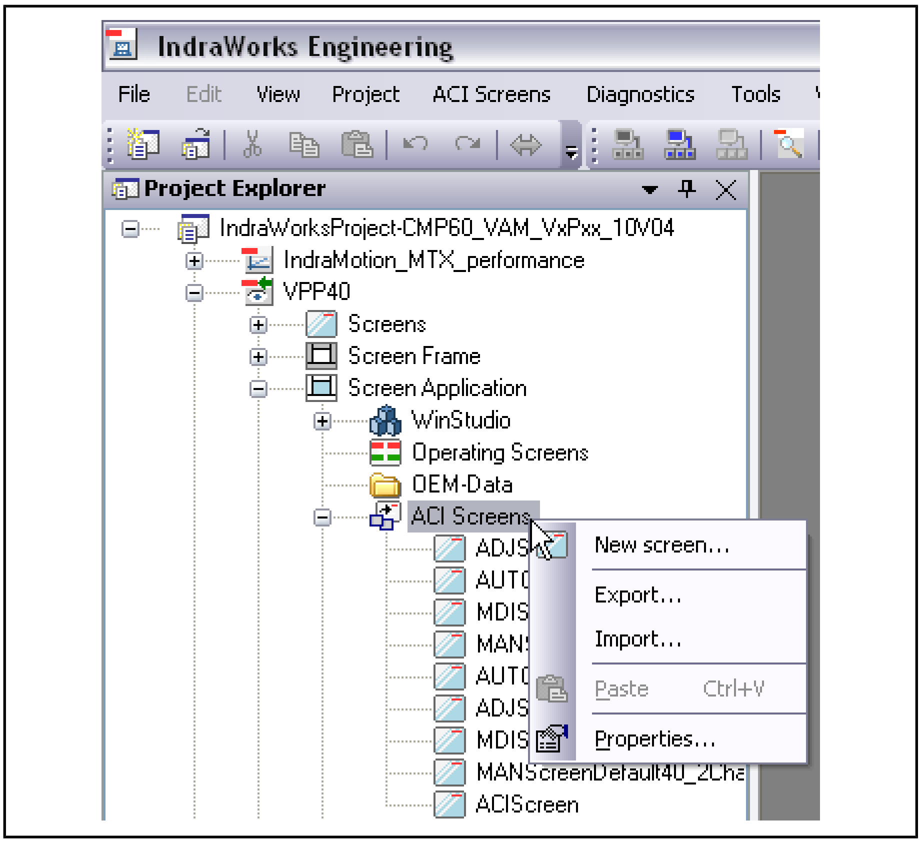


Fig.7-2: Context menu: ACI Screens

The dialogs and functions to configure properties shared by all the ACI screens as well as those to create a new ACI screen can be opened using the context menu of the project node "ACIScreens" (fig. 7-2 "Context menu: ACI Screens" on page 118).

The specific configuration of an individual ACI screen, such as type, number and position of the process data displays that are to be summarized in the ACI screen, can be opened using the context menu of the project node of the corresponding ACI screen.

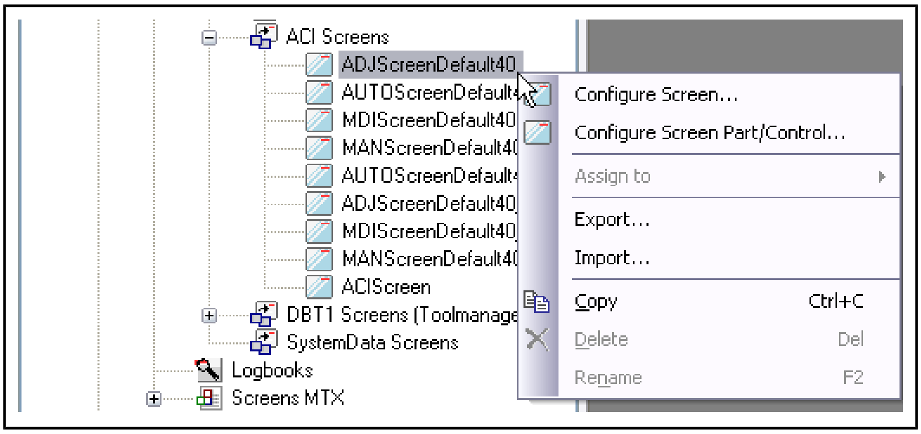


Fig.7-3: Context menu: ACI screens (AUTOScreenDefault40)

"New Screen" Dialog

A new ACI screen can be created using the "New Screen" dialog which is opened by selecting the entry "New Screen" in the context menu of project node "ACI Screens".

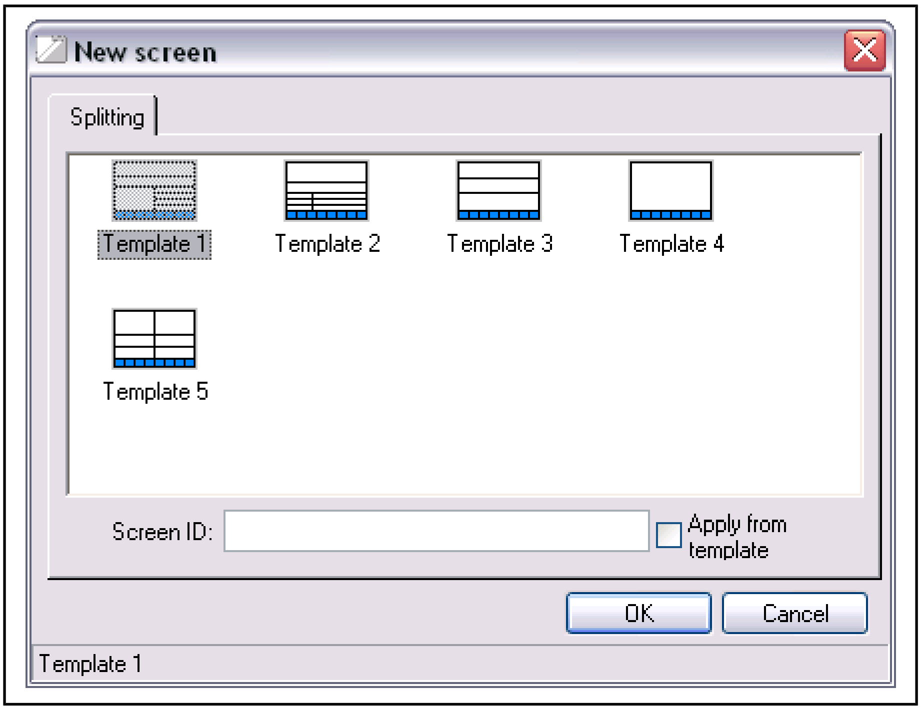


Fig.7-4: Dialog: New screen (layout templates)

To create a new screen, it is necessary to select a layout template and to enter a unique screen ID. The screen ID can be entered in the input field or transferred to the layout template by selecting the checkbox.

The layout and content of the screen segments of the screen created can be adapted to your needs in the dialogs "Configure Screen" (fig. 7-9 " Dialog: “Screen configuration” (with controls without process connection inserted into segments)" on page 124) and "Configuration screen segment/control..." (fig. 7-11 "Dialog: Configuration screen segment/control (with controls without process connection)" on page 126).

☞ It is not possible to create a new screen with a name (screen ID) that is already in use. If you attempt to do this, a corresponding error message appears.

"Properties of All Screens" Dialog

This dialog is opened using the context menu entry "Properties..." of the project node "ACI Screens".

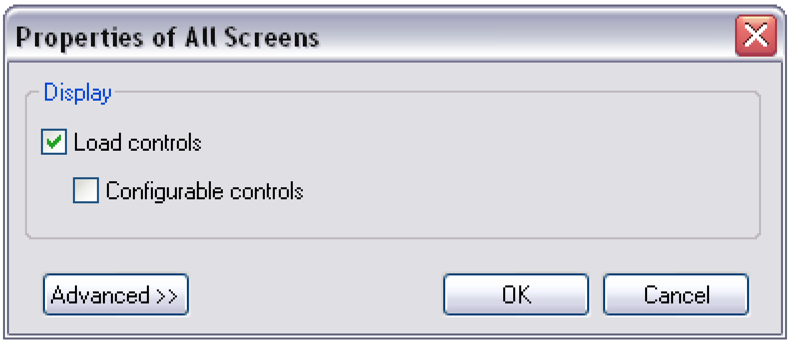


Fig.7-5: Dialog: Properties of all screens

"Display" indicates whether the controls have been initialized in dialogs "Configuration Screen" and "Configuration screen segment/control" or if, instead, only the name of the controls is shown as their placeholders. Furthermore, you can specify whether a "Configurations screen segment/control..." button is to be displayed in "Configure Control..." (fig. 7-11 "Dialog: Configuration screen segment/control (with controls without process connection)" on page 126). This button opens the configuration dialog of the focused control if the control has been initialized and if it has its individual configuration dialog.

- **Advanced >>**:

  Shows the advanced dialog.

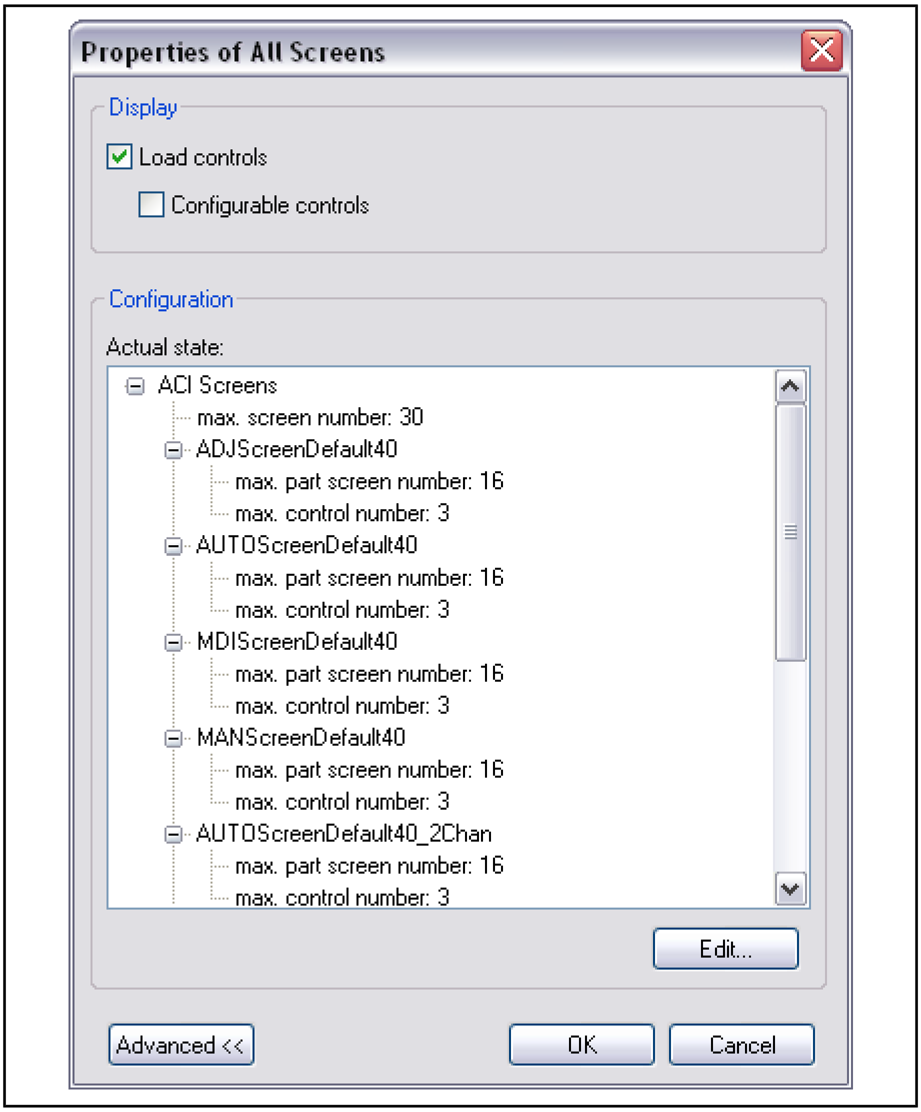


Fig.7-6: Dialog: "Properties of All Screens" (advanced dialog)

The "Advanced" dialog also shows the current configuration of the project node "ACI Screens" in terms of the maximum number of ACI screens that have to be managed, as well as the screen segments that can be created in an ACI screen and the controls that can be inserted into a screen segment.

☞ In the dialog "Configuration screen segment/control..." (fig. 7-11 "Dialog: Configuration screen segment/control (with controls without process connection)" on page 126) the specified maximum number of controls is taken into account in the function "Insert control...".

If the maximum number of controls that can be inserted in a screen segment is reduced, the controls that are already in the screen segment but which, as a result, exceed the maximum number are not removed. The same applies to the screens and the screen segments if their maximum number is reduced.

- **Modifying...**:

  In terms of their maximum number of screens, screen segments and controls, the configuration of the ACI screens can be adapted by the user in a separate "Modify - configuration" dialog which is called by pressing the "Change..." button.

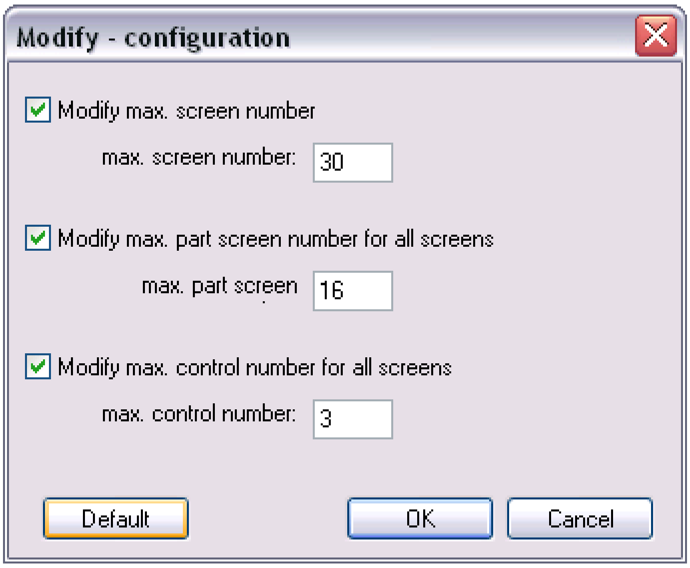


Fig.7-7: Dialog: Modify - configuration (default configuration)

- **Default**:

  The values of the default configuration are automatically entered in the input fields.

☞ Values between 1 and 9999 can be entered for the maximum number of screens, screen segments and controls.

Continue to: "Properties of all Screens" dialog

If the values in the dialog "Modify - configuration" have been confirmed by pressing the "OK" button, the dialog "Properties of all Screens" shows the command status instead of the actual status and the modified values are highlighted.

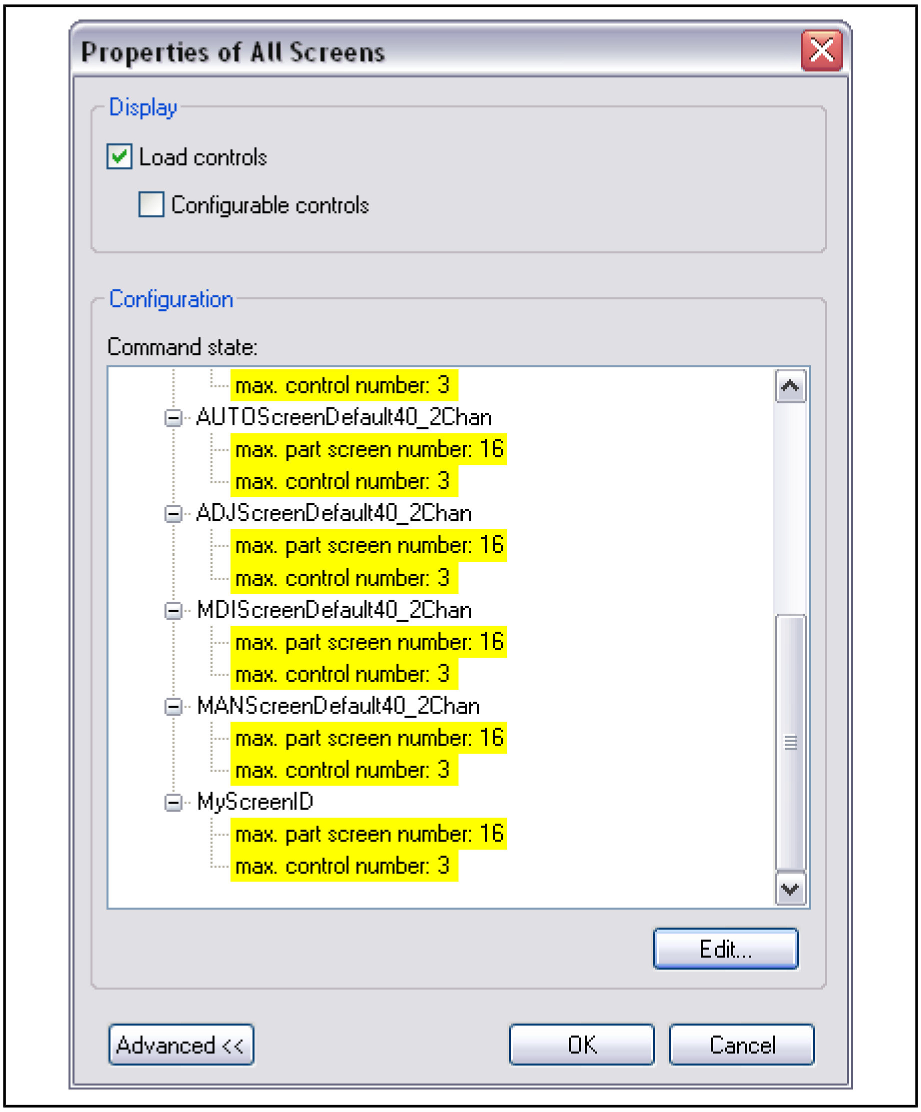


Fig.7-8: Dialog: Properties of All Screens (modifying configuration values)

The modified values for the ACI screens are saved only after they have been confirmed by pressing the "OK" button in the dialog "Properties of All Screens".

- **Advanced <<**:

  Hides the advanced dialog.

"Screen Configuration" Dialog

The dialog "Screen Configuration" is opened via the context menu entry of the same name in the project node of the ACI screen to be modified. The division of the screen into segments, including their number and position, can be changed in this dialog.

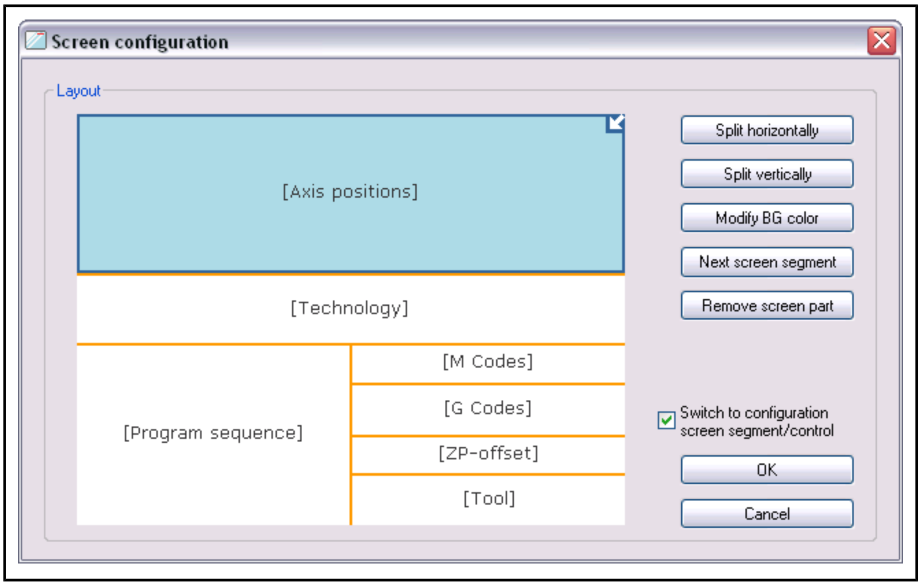


Fig.7-9: Dialog: "Screen configuration" (with controls without process connection inserted into segments)

The following functions are available to design the screen layout as the division of the screen into segments:

- **Divide horizontally / vertically**:

  The focused screen segment is divided horizontally or vertically through the middle.

☞ If this division results in the maximum number of screen segments for this screen being exceeded, a corresponding error message appears in the status line and the dividing process is cancelled.

The maximum number of screen segments into which the screen can be divided can be displayed and modified in the dialog "Properties of All Screens".

- **Change BG color**:

  To improve the visual separation of the screen segments, the background color of the focused screen segment can be chosen from two shades of gray.

- **Next screen segment**:

  In the screen, the screen segment that follows the currently focused screen segment in the list of screen segments is focused.

☞ The screen segments are arranged in a specific sequence in a list of screen segments. This order is specified due to the position of the screen segments in the screen - usually "from the top to the button, beginning at the left-most" - and cannot be changed.

Screen segments can also be focused directly regardless of the sequence in the list of screen segments by moving the cursor to the screen segment and clicking with the left mouse button. This method of focusing does not work if the screen segment contains a control that is not correctly programmed.

- **Delete screen segment**:

The focused screen segment is deleted and the dialog for filling the removed screen segment is adapted. In this adapted dialog, fill the vacated location of the deleted screen segment by increasing the size of neighboring screen segments. A corresponding handling instruction summary in the dialog explains the steps required.

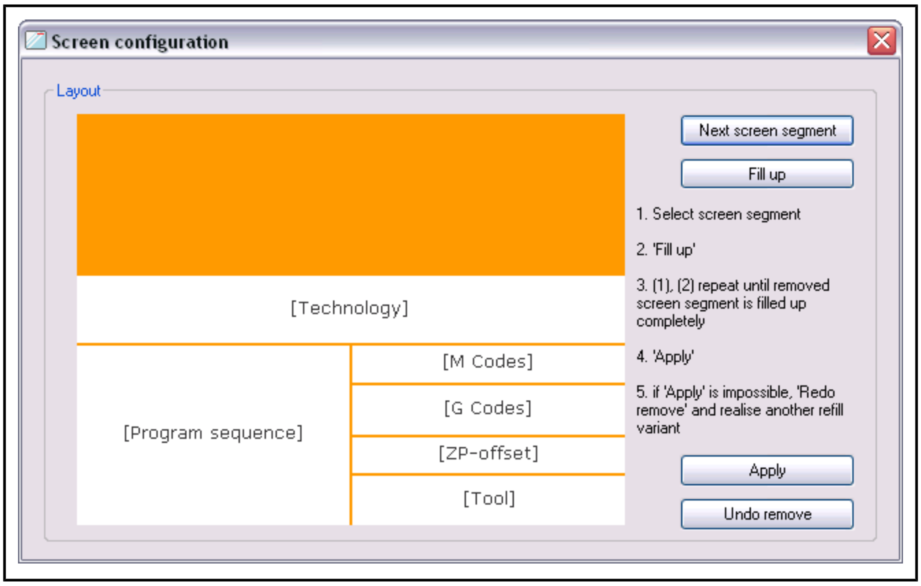


Fig.7-10: Dialog: "Screen configuration" (after selecting the function: Delete screen segment)

- **Next screen segment**:

  See above

- **Fill up**:

  The focused screen segment is expanded into the vacant area of the deleted screen segment.

- **Apply**:

  The filling variant that is used is accepted; the display returns to the main dialog.

- **Undo remove**:

  The filling variant that is used is undone and the deleted screen segment is regenerated. The display returns to the main dialog.

☞ To ensure that the screen is consistently divided into screen segments, the free space of the deleted screen segment must be filled in completely before another division can be accepted. If the division cannot be accepted after the filling procedure has been carried out, undo the deletion, repeat it and use another filling variant.

The steps required for a filling procedure are listed in the dialog.

The division of a screen segment is undone by deleting the screen segment and expanding the adjacent screen segments(s) into the free space.

Continue to: "Screen Configuration" dialog

- **Dragging the screen segment dividing line**:

  The size of the screen segments can be changed by moving their dividing lines. Move the cursor to the dividing line. The cursor icon changes. While holding the left mouse button down, drag the dividing line with the mouse.

- **Change to Configuration screen segment/control**:

  When you press "OK", the modified data is accepted and the dialog is closed. The checkbox can be used to specify whether the "Configuration screen segment/control" dialog for the same screen segments is opened automatically.

"Configuration Segment/Control" Dialog

The "Configuration screen segment/control" dialog, which can be opened via the corresponding context menu entry in the project node of the associated ACI screen, can be used to change type, number and sequence of the controls in the screen segments and their associated F-panel, as well as certain focusing properties of the screen segments.

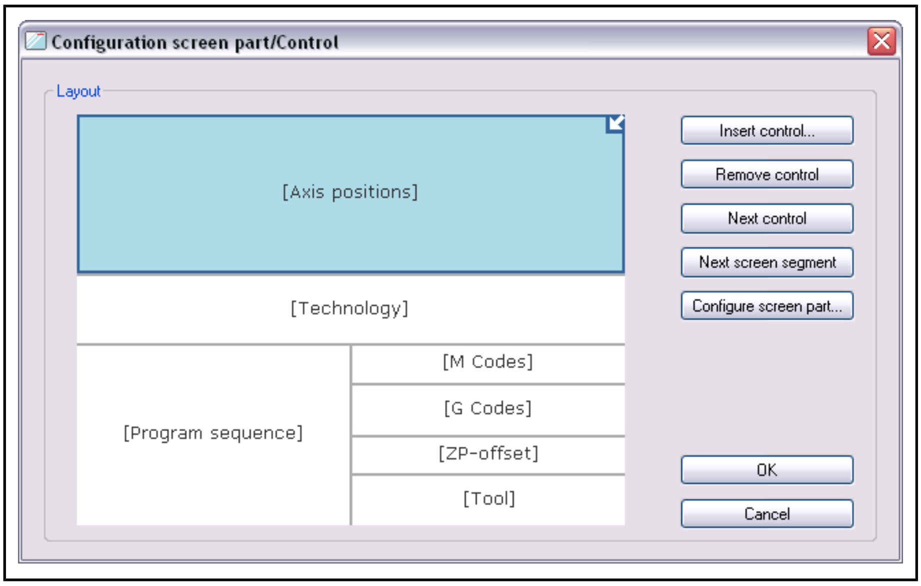


Fig.7-11: Dialog: Configuration screen segment/control (with controls without process connection)

Controls can be inserted as content for the individual screen segments. A specific F-panel can be assigned to each control. The F-panel is shown in the user interface when the control is focused.

It is possible to insert several controls into a screen segment. If this is done, they are positioned congruently one in front of the other in the screen segment so that only the control on the top is visible.

☞ If the control properties should be changed via their "Options" dialog in the Engineering Desktop, they have to be loaded before. This is also required to insert a control into an ACI screen to create a personal configuration profile. The condition to load (instantiate) the controls is set under "Properties..." in the context menu of the ACI screen node (chapter "Properties of All Screens Dialog " on page 120).

- **Insert control...**:

  The control to be inserted into the focused screen segment can be selected from the list of known controls using the called dialog "Insert control" (fig. 7-12 " Dialog: "Insert control"" on page 127). In its original state, this "Known controls" list contains default entries for controls that are frequently used in ACI screens. The "Add..." button can be used to supplement entries for any additional controls needed.

If in the maximum number of controls for this screen segment would be exceeded due to the insertion, a corresponding error message appears in the status line and the insertion is cancelled, i.e. the dialog is not opened.

The maximum number of controls that can be inserted into the screen segment can be displayed and modified in the dialog "Properties of all Screens" (fig. 7-6 "Dialog: "Properties of All Screens" (advanced dialog)" on page 121).

The default entries in the "Known Controls" list may vary from software version to software version in order to fulfill the current requirements for the displays in ACI screens. The control entries that have been subsequently added to the "Known Controls" list are retained if the software is updated.

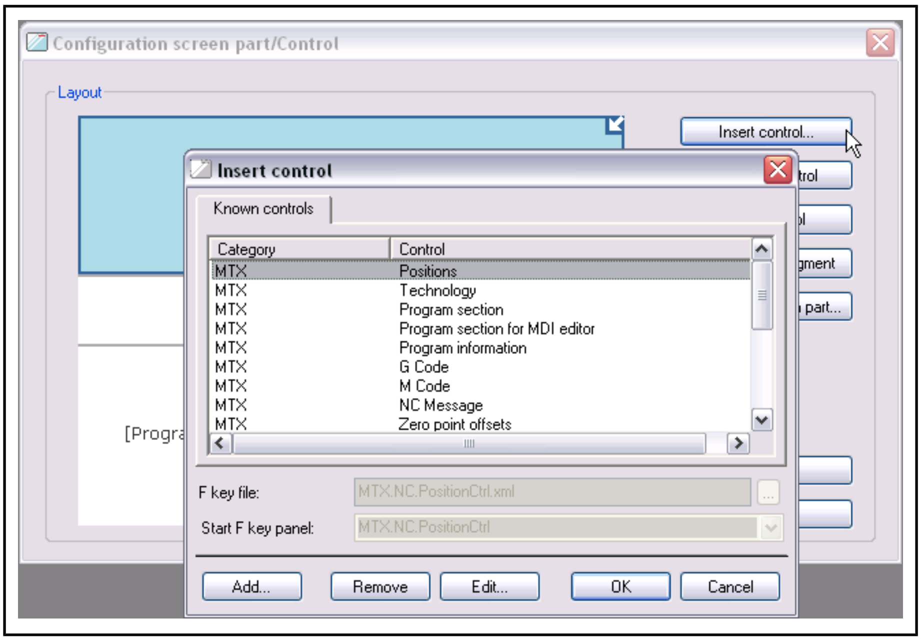


Fig.7-12: Dialog: "Insert control"

- **Add...**:

  Using the opened "Add control" dialog (fig. 7-13 "Dialog: "Add control" (after selecting "Add...")" on page 128), you can search the computer for "*.ocx" and "*.dll" files. If this file contains information about controls, these controls are incorporated into the "Known controls" list. The list entries consist of the category – which is used mainly as an arrangement criterion to assign the control to a certain group – and the control name, which is displayed in a screen segment if the control cannot be loaded into it. Both list entries can be freely assigned in the "Add control" dialog. However, entering a control name that differs from the one read from the "*.ocx"/"*.dll" file is only useful if the file contains information on only one control. Otherwise, all the controls in this file would appear with the same name in the list. But it is still possible to change the control names for each individual control at a later point in time.

☞ When ActiveX controls are added, they are automatically entered in the registry (Windows database) using the program "regsvr32.exe". For .NET controls, the program "regasm.exe" is searched on the computer. This may take a few minutes. Prior to the search, a corresponding message appears pointing out the possibility to cancel the adding to the control. If a "*.tlb" file with the same name exists in the directory of the "*.dll" file in the case of .NET controls, the control information is extracted from this file and no registration is carried out.

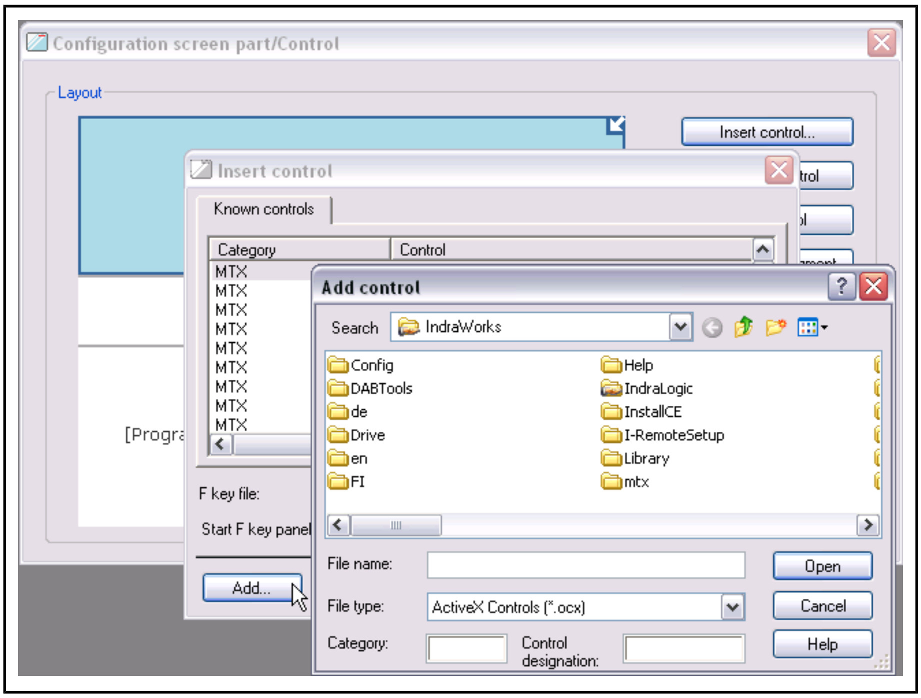


Fig.7-13: Dialog: "Add control" (after selecting "Add...")

Continue to: Dialog "Insert control"

- **Delete**:

  The selected entry is deleted from the list of "Known controls".

- **Edit**:

  The entries in the list of "Known controls" are specified in detail using the function "Edit" in the opened dialog "Change properties: Control entry" (fig. 7-14 "Dialog: "Change properties" (Control entry after opening the "Edit..." function)" on page 129) and can be changed there:

  – The control name, which has either been read from the "*.ocx"/"*.dll" file or defined in the dialog "Add control..." when a control is added, can be renamed later or, by entering the corresponding token number, it can be read from a text file. This way, it can even become language-dependent. The text file which is used for language dependency must be arranged according to a certain structure and must have the abbreviation of the regional scheme set in the system as a supplement to the file name. For example, see the file "ACIScreens_Controls_de.txt" in the installation directory used for the controls inserted by default in the regional scheme "German" (DE). To provide language dependency, it is also required that all the controls added to the "Known controls" list use the same language file. This can be entered in the input field only after selecting the checkbox "Change for all added control entries".

- The same applies to renaming and the language-dependent specification of the category to which the control was assigned.
- A separate F-panel can be assigned to every control added to the "Known controls" list. This is displayed when the control is focused in the ACI screen of the Operation Desktop. Select the appropriate F-key file and enter the name of the F-panel. The default controls in the "Known controls" list are fixedly assigned to a specific F-panel. The checkbox "Assignment can be changed" can be used to cancel this fixed assignment and to change the F-panel.
- If a control contains the implementation of the MTXACIInterface.dll interface, the configuration parameter that can be entered in "Instance" field can be transferred to it. If the control can display different process data or if it has various types of displays, this configuration parameter can be used during the initial instancing of the control to determine the instances to be used by the control.
- Controls that have been added later to the "Known controls" list can be easily deleted from the list via "Delete" in the "Add control" dialog. On the contrary, the default entries are protected against accidental deletion. However, they can be deleted by selecting the checkbox "Control entry can be deleted from the 'Known controls' list".

The language file can be changed only for all control entries added to the default entries in the "Known controls" list.

If an F-panel is not assigned to a control, a default F-panel is displayed when the control is focused in the ACI screen of the Operation Desktop. This applies to all ACI screens. This can currently not be reconfigured.

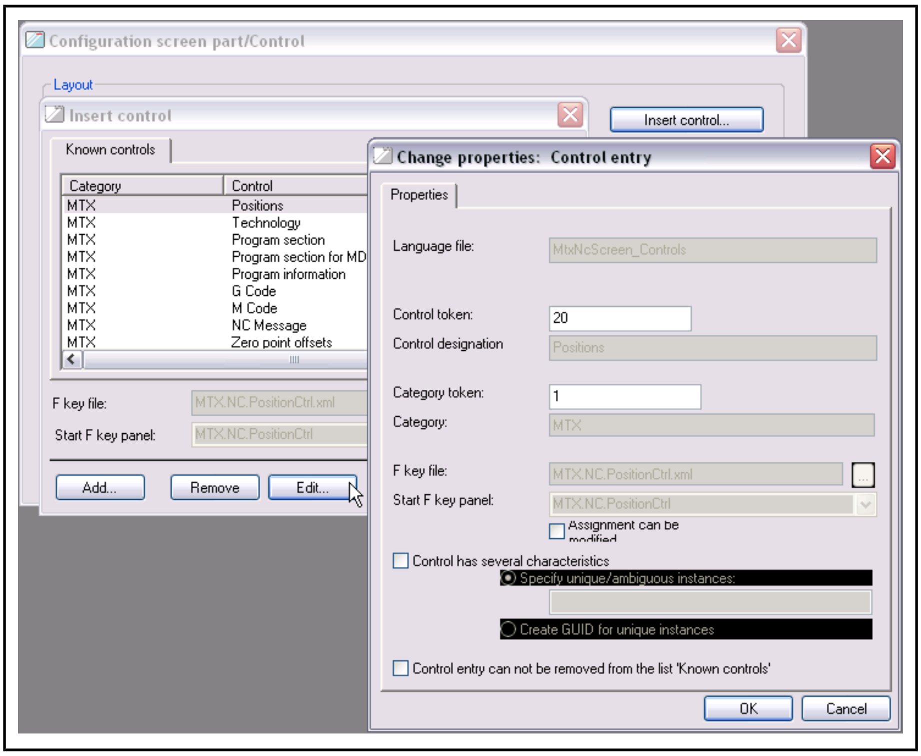


Fig.7-14: Dialog: "Change properties" (Control entry after opening the "Edit..." function)

Continue to: Dialog "Configuration screen segment/control"

- **Remove control**:

  The visible control of the focused screen segment is deleted after a security prompt is confirmed.
- **Next control**:

  The control that follows the currently visible control in the list of controls of the screen segment is shown in the focused screen segment.
- **Next screen segment**:

  see dialog "Screen Configuration" (fig. 7-9 " Dialog: “Screen configuration” (with controls without process connection inserted into segments)" on page 124)
- **Configure screen segment...**:

  The dialog "Configuration screen segment" (fig. 7-15 "Dialog: Configuration screen segment" on page 131) is called for the focused screen segment. This contains the following information and setting options:

  – The controls of the screen segment are listed in the display order in this dialog. The "up" and "down" buttons can be used to change the priority of a control, i.e. its position (index) in the display order.

  – In addition, the instance – transferred as configuration parameter – and the assigned F-key file and F-panel are displayed for a control that is selected in the list of controls. The F-key file and panel can be changed if the respective option "Allocation can be changed" is selected for the control in the dialog "Change properties: control entry" (fig. 7-14 "Dialog: "Change properties" (Control entry after opening the "Edit..." function)" on page 129; set by default for controls that added to the list "Known controls" in dialog "Add control" (fig. 7-12 " Dialog: "Insert control"" on page 127).

  – Furthermore, additional options can be set to focus the screen segment in the ACI screen of the Operation Desktop . The checkboxes "Screen segment focusable" and "Screen segment is focusing during first display of screen" can be used to determine whether the screen segment can be focused in the first place and whether this screen segment should be immediately and automatically focused when the screen is displayed for the first time after starting the Operation Desktop. In the Operation Desktop, press the F-key "Next window" to switch to the next screen segment in a specified sequence and focus it. The screen segment index listed in dialog "Configuration screen segment" indicates the position of the screen segment in this focused sequence.

☞ In the Operation Desktop (except the configuration mode), a screen segment can be focused only if this property has been selected in "Configure screen segment" in the dialog "Configuration screen segment/control" of the Engineering Desktop and if the screen segment contains at least one control.

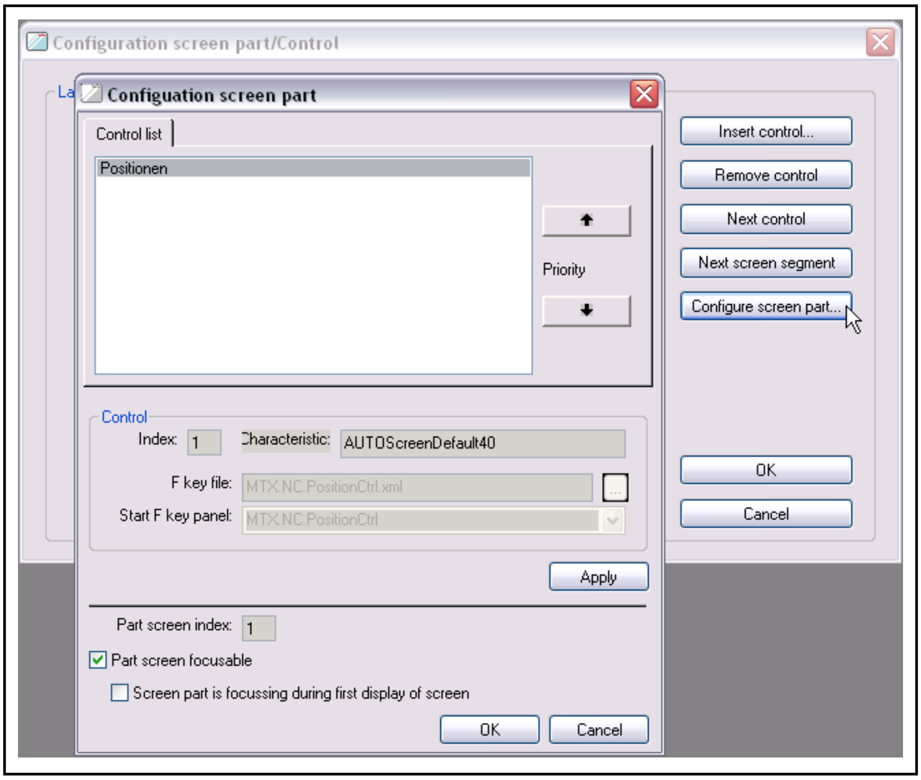


Fig. 7-15: Dialog: Configuration screen segment

Functions "Copy"/"Paste"

The functions "Copy" and "Paste" are selected via the respective context menu entry in the project node of the respective ACI screen or via the project node "ACI Screens".

A new ACI screen is created that is equal to the previously copied ACI screen. Its name is the copied ACI screen plus the incremental counter. In the event that the copied ACI screen is inserted into another visualization device of the project and does not yet exist there, the incremental counter is omitted.

The layout and content of the screen segments of the screen created can be adapted to your needs in the dialogs "Configure Screen" (fig. 7-9 " Dialog: “Screen configuration” (with controls without process connection inserted into segments)" on page 124) and "Configuration screen segment/control..." (fig. 7-11 "Dialog: Configuration screen segment/control (with controls without process connection)" on page 126).

☞ If an ACI screen with the identical name already exists in the same or in another visualization device when inserting the ACI screen, the configuration settings of the controls in the inserted screen are reset and it is initialized in the original version. Thus, in case of changing the options of the controls, the user cannot change the controls of another screen by default.

Functions "Export"/"Import"

The functions "Export" and "Import" are selected via the respective context menu entry in the project node of the respective ACI screen or via the project node "ACI Screens". This depends on what should be imported or exported.

During export, a project-independent file is written that contains the selected ACI screens. During import, new ACI screens are created that are equal to the previously exported ACI screens including there names if there are no ACI screens with the same name. Otherwise, the user can choose whether the already existing ACI screens should be overwritten or whether the exported ACI screens should be inserted as additional screens. In the latter case, their names get an incremental counter.

The layout and content of the screen segments of the screens created can be adapted to your needs in the dialog "Screen Configuration" (fig. 7-9 " Dialog: “Screen configuration” (with controls without process connection inserted into segments)" on page 124) and "Configuration screen segment/control" (fig. 7-11 "Dialog: Configuration screen segment/control (with controls without process connection)" on page 126).

If an ACI screen already exists with the identical name in the same or in another visualization device when importing the ACI screen and if the exported ACI screen should be inserted as additional screen, the configuration settings of the controls in the inserted screen are initialized in the original version. Thus, in case of changing the options of the controls, the user cannot change the controls of another screen by default.

Function "Delete"

The "Delete" function is called using the respective context menu in the project node of the associated ACI screen. The affected screen is deleted after a security prompt has been confirmed.

Function "Rename"

The name (screen ID) of an ACI screen can be changed directly in the respective project node.

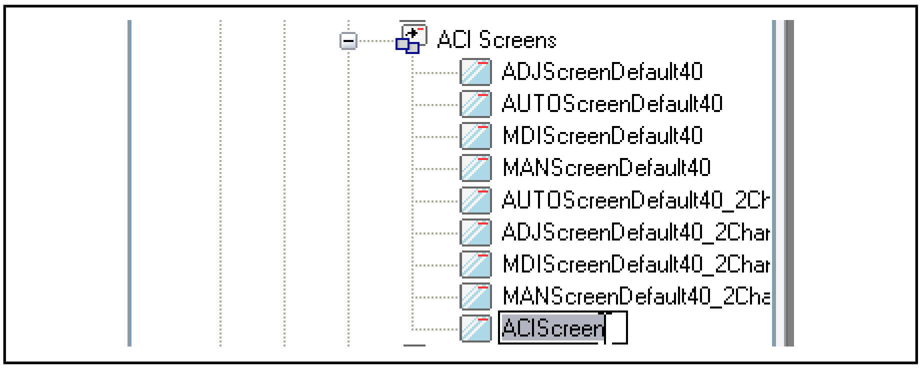


Fig.7-16: Renaming an ACI screen in the project node

Select the function "Rename" via the respective context menu entry in the project node of the ACI screen or click on the project node name again after selecting it with the mouse.

It is not possible to rename a screen using a name (screen ID) already in use. If you attempt to do this, a respective message appears.

7.1.3 Configuring in the Operation Desktop

General

Short description

In Operation Desktop, a special configuration mode can be used to adapt the screen segments of the ACI screens and of the control displays to the actual size of the ACI screens in the real user environment using the OP, M and F keys and the header.

☞ To enter the configuration mode of an ACI screen, a key combination (shortcut) must be defined its call. This is executed in the "Options" dialog that is located in the Operation desktop in the **Tools ▸ Options...** menu item. In this dialog box, under "General/ Shortcuts" of the Explorer, a shortcut has to be assigned to the command "Screen Layout Configuration" of the category "Configuration" (fig. 7-18 "Dialog box: Options (Shortcut definition for the configuration mode)" on page 133).

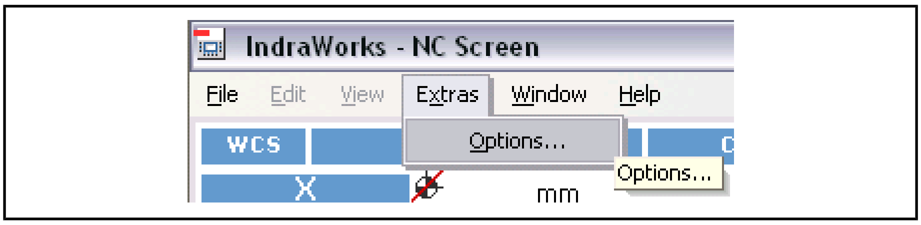


Fig.7-17: Menu: Options (in the Operation Desktop)

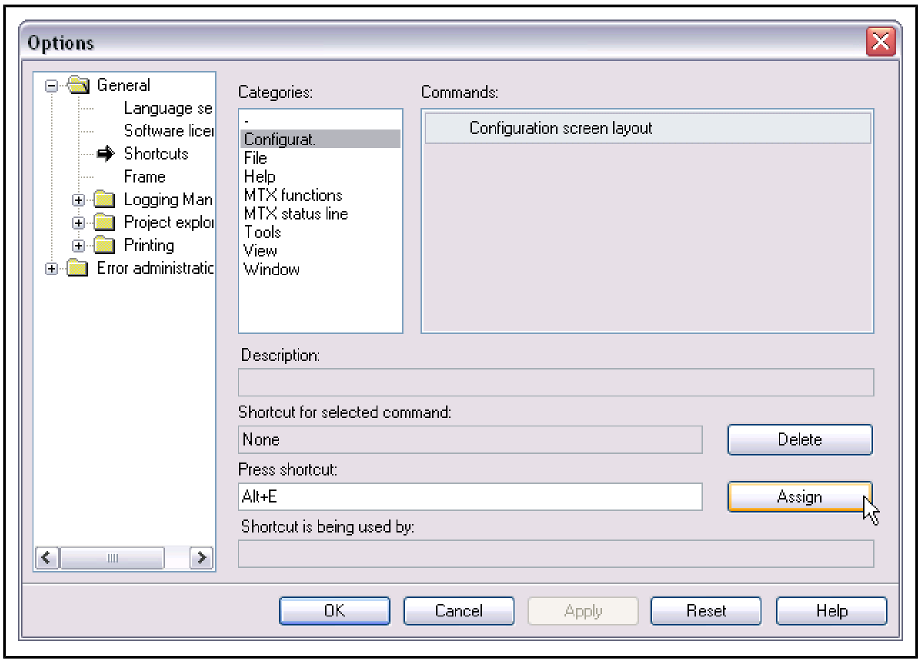


Fig.7-18: Dialog box: Options (Shortcut definition for the configuration mode)

Description

In the user interface, the controls in the ACI screens have process connections. As a result, they take on a special visual appearance. To provide any overall visual appearance of the ACI screens, the size of their screen segments can be set in the configuration mode. The possibilities for setting the visual appearance of the controls, as well as further options, depend on the control's own configuration dialog box.

Configuring the User Interface

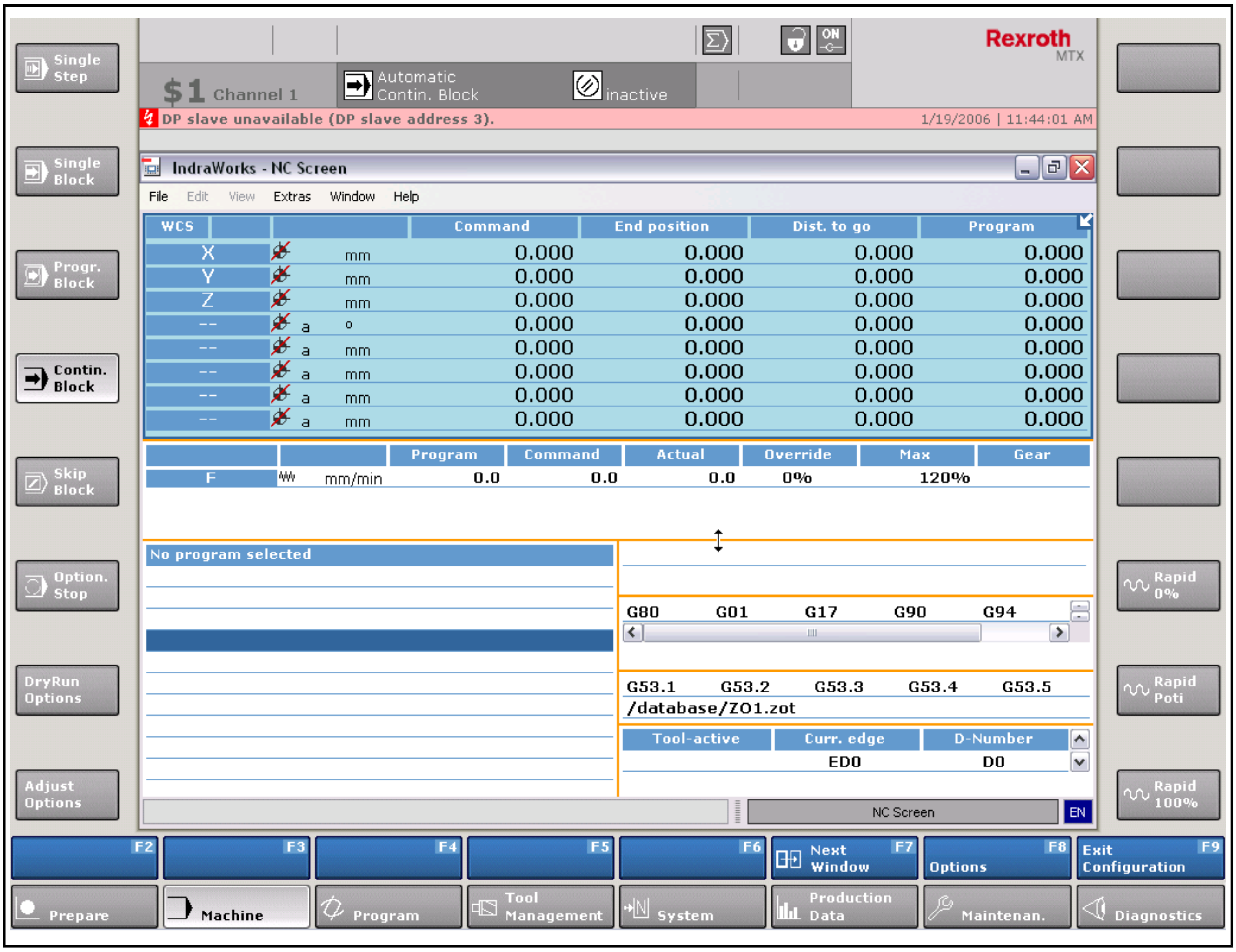


Fig.7-19: Operation Desktop with ACI screen in configuration mode

- **Dragging the screen segment dividing line**:

  The size of the screen segments can be changed by moving their dividing lines. To do this, move the cursor to the dividing line; the cursor symbol will change. While holding the left mouse button down, you can now drag the dividing line using the mouse.

The following dialog boxes and functions can be opened using the appropriate F key:

- **Options**:

  If the visible control of the highlighted screen segment has its own configuration dialog box, this is displayed.

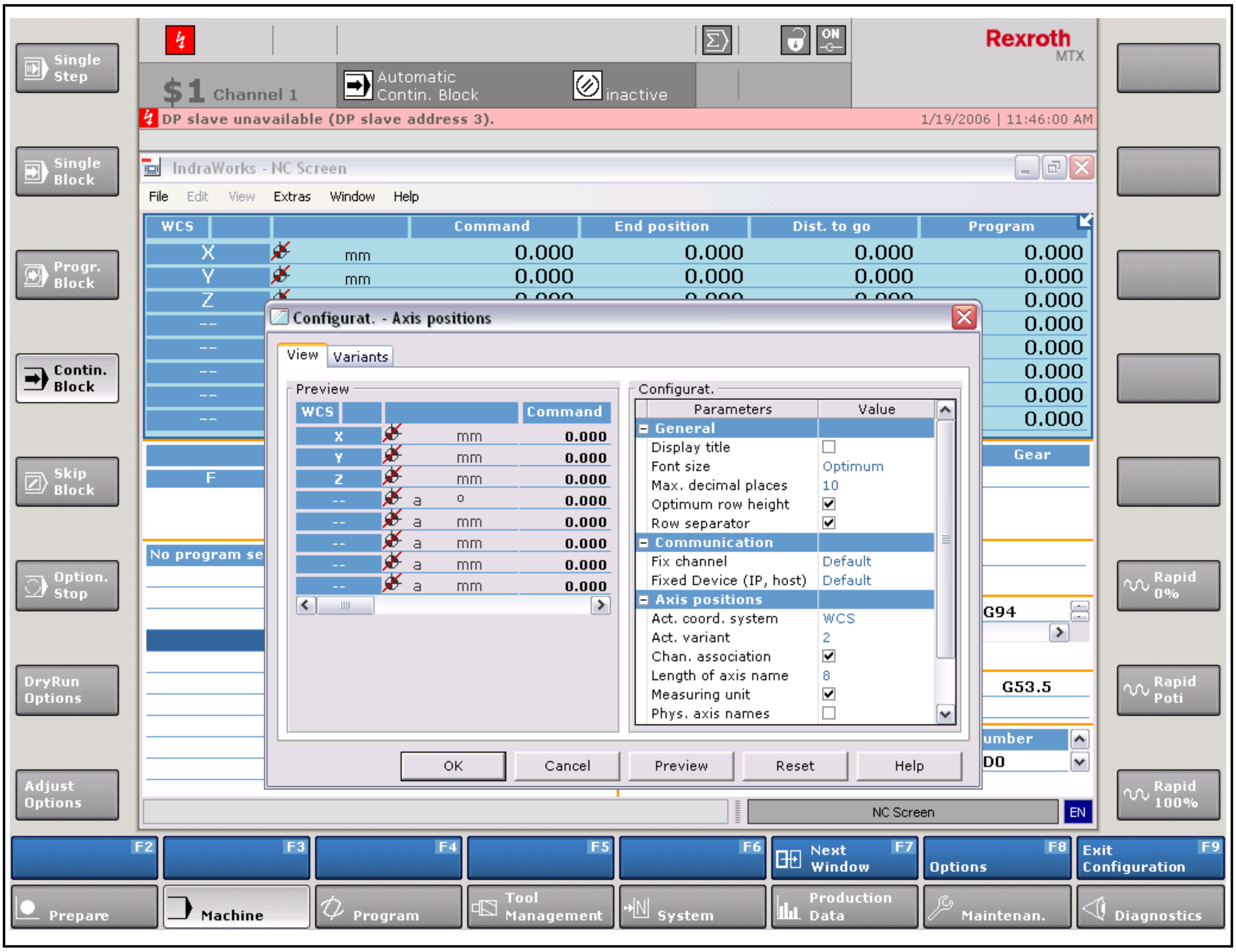


Fig.7-20: Dialog box: Example of a control's own configuration dialog box: "Configuration – Axis positions" of the control for the axis position display

- **Next window**:

  In the screen, the screen segment that follows the currently highlighted screen segment in the list of screen segments is highlighted.

☞ The screen segments are arranged in a specific sequence in a list of screen segments. This order is specified due to the position of the screen segments in the screen - usually "from the top to the button, beginning at the left-most" - and cannot be changed.

Screen segments can also be highlighted directly regardless of the sequence in the list of screen segments by moving the cursor to the screen segment and clicking with the left mouse button. This method of highlighting does not work if the screen segment contains a control that is not correctly programmed.

- **Exit configuration**:

  A security prompt appears asking whether the current layout of the screen is to be saved. Then the configuration mode is exited.

☞ The security prompt appears even if no changes were made.

Handling Instruction: Create and Configure a New ACI Screen

Using an example, a new ACI screen will be created and configured on a step-by-step basis.

IW Engineering / Configuration of ACI screens: Create a new ACI screen

1. In the Project Explorer, click on the node "ACI Screens" using the right mouse button.

   The context menu of "ACI Screens" opens.

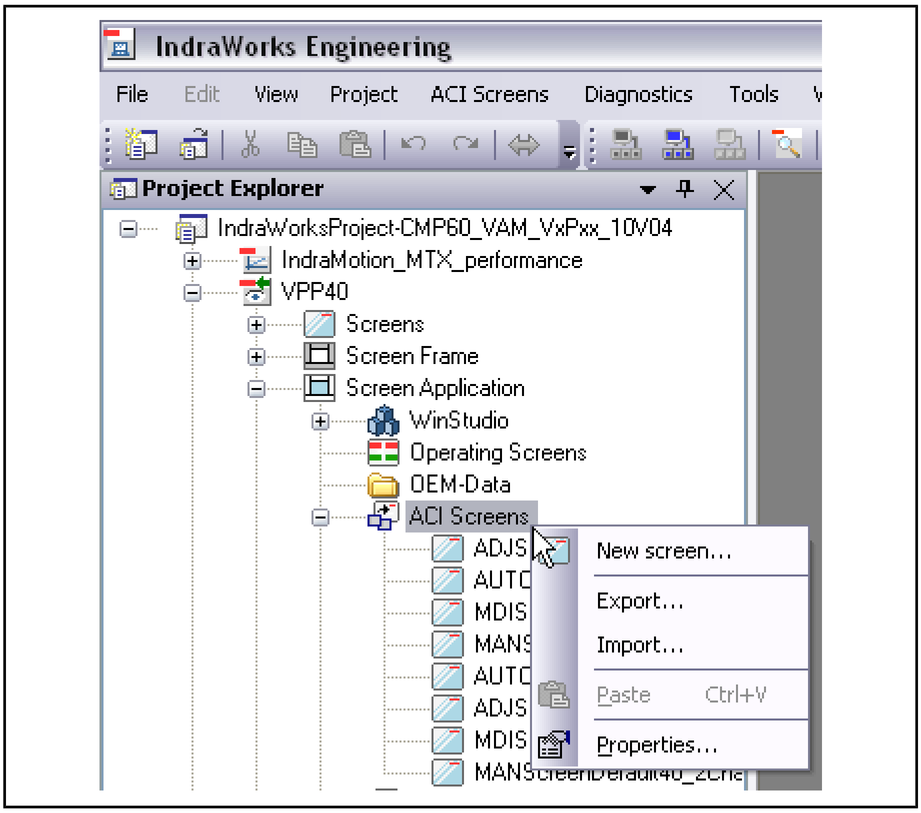


Fig.7-21: Context menu: ACI Screens

2. In the context menu, click on the entry "New Screen".

   The "New Screen" dialog is displayed.

3. In the "New Screen" dialog, click on template 2 and select the checkbox "Transfer from Template".

   The screen division and screen ID of template 2 are used for the new screen.

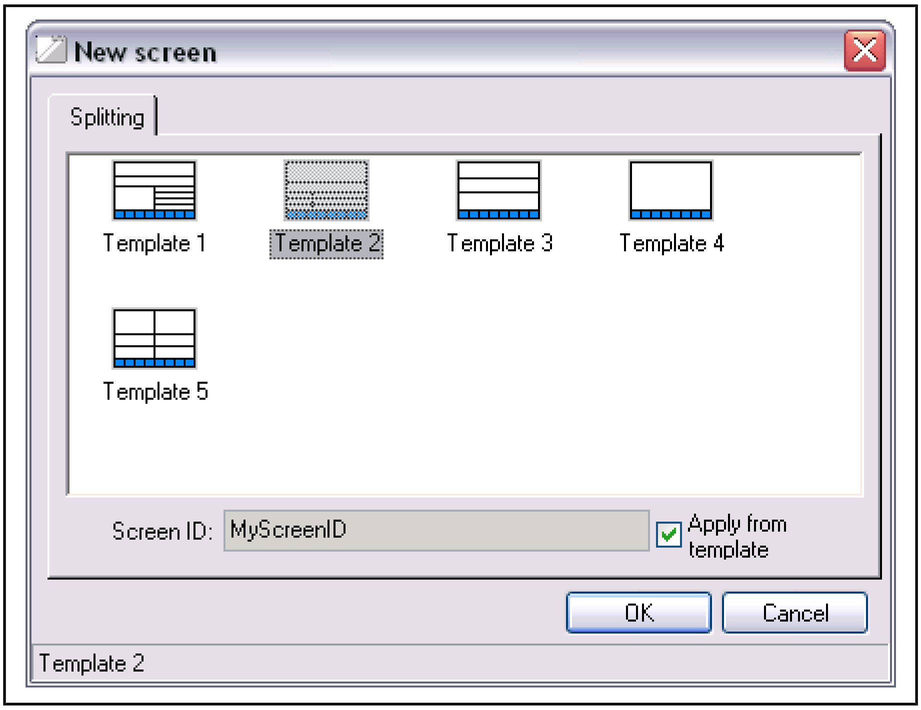


Fig.7-22: Dialog: "New screen" (with selected template)

4. Confirm the entries made in the dialog "New Screen" with "OK".

   The name of the new screen is displayed below the project node "ACI Screens".

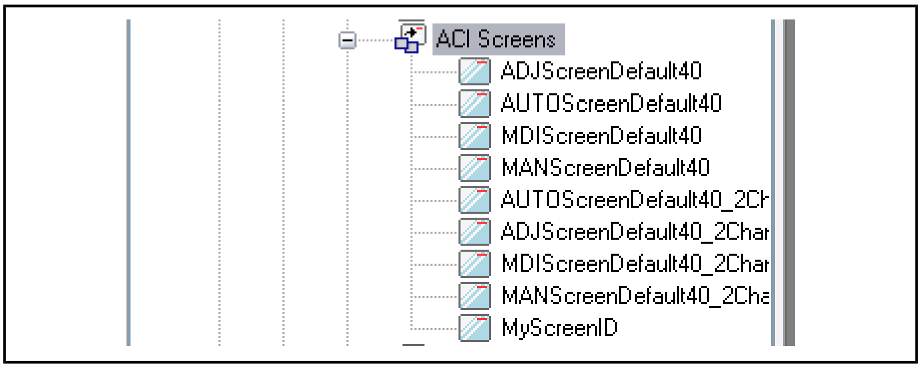


Fig.7-23: Project node "ACI Screens" with new ACI screen

IW Engineering / Configuration of ACI screens: Change the screen division of the new ACI screen

1. Click the node of the new ACI screen using the right mouse button.

   The context menu of the ACI screen opens.

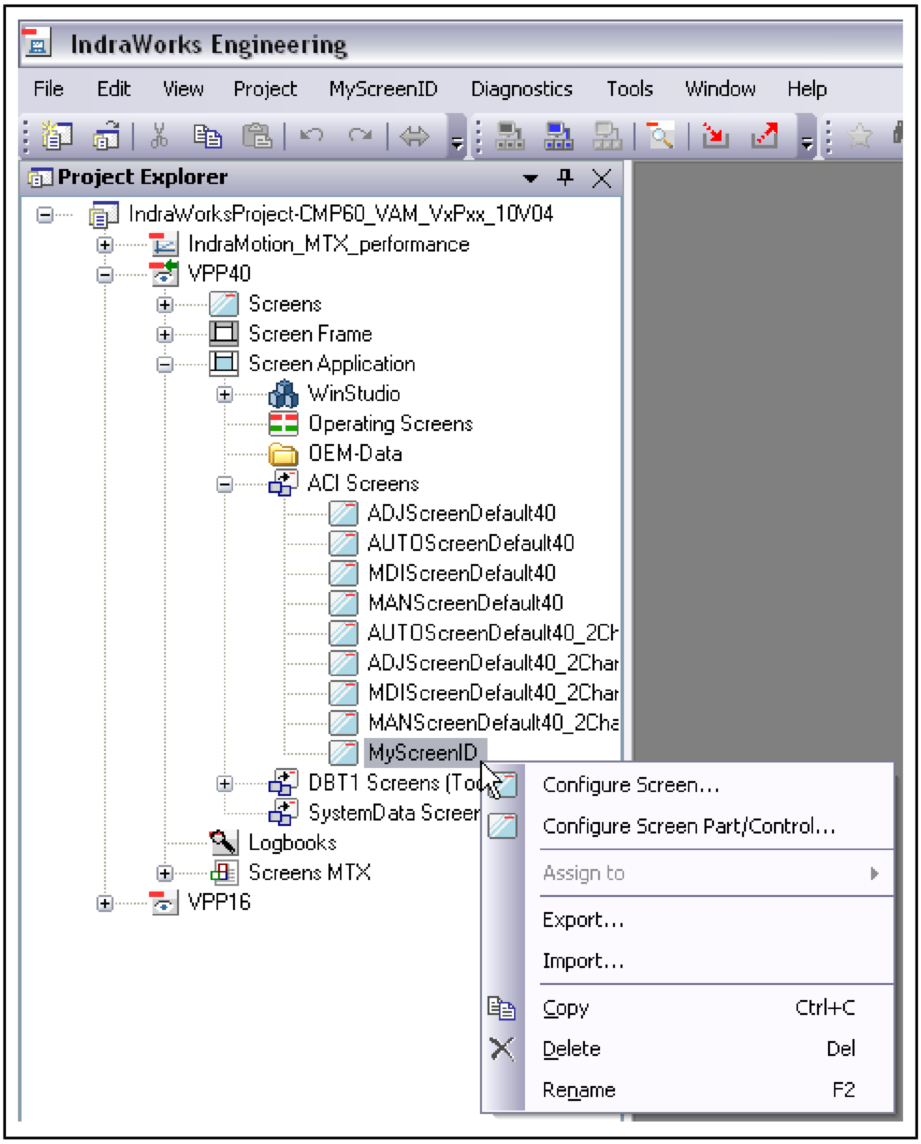


Fig.7-24: Context menu: MyScreenID

2. In the context menu, click on "Screen Configuration".

   The "Screen Configuration" dialog is displayed.

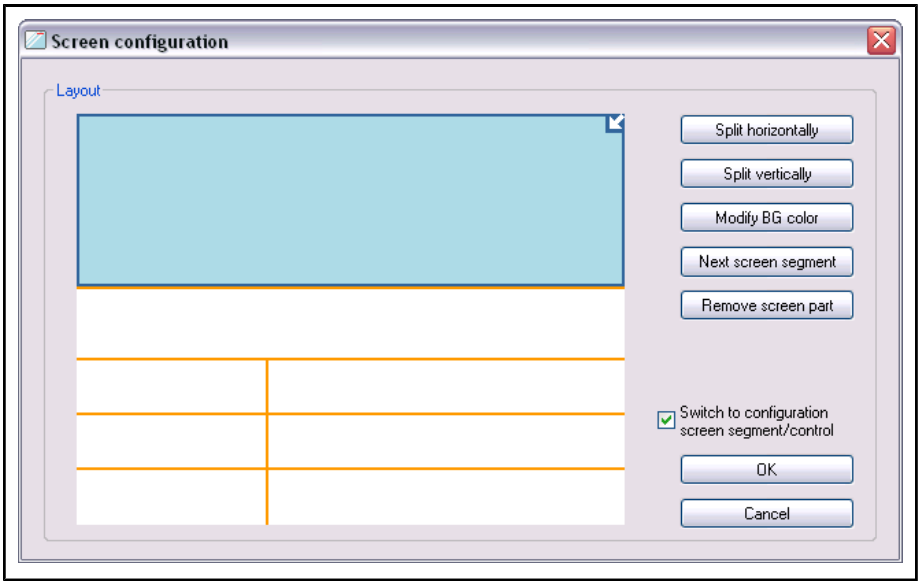


Fig.7-25: Dialog: "Screen configuration" (for the new ACI screen)

3. Press "Divide horizontally".
4. Press "Change BG color" twice.
5. Press "Next screen segment".
6. Press "Divide vertically".

   The focused screen segment is divided horizontally and the background color of the newly focused screen segment changes first to white, then to gray. Then the display switches to the next screen segment, which is divided vertically.

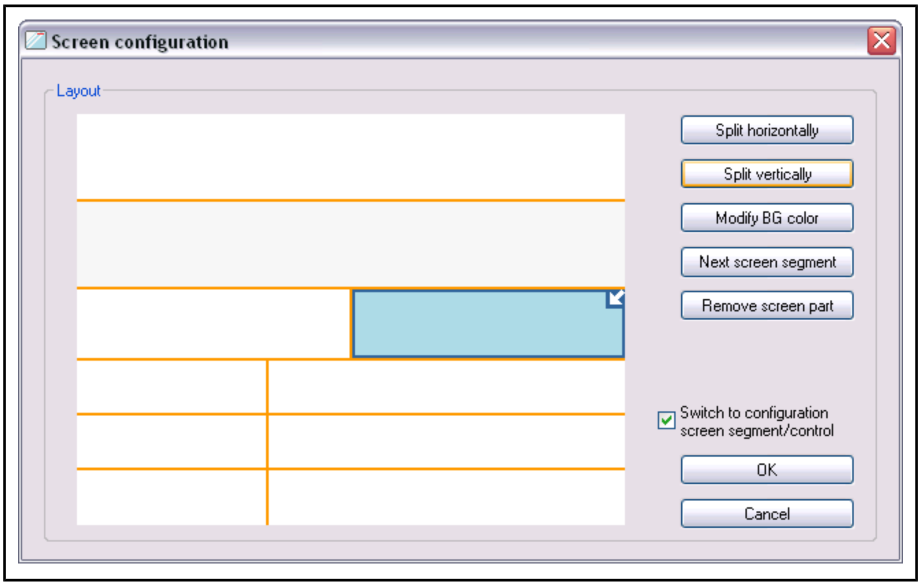


Fig.7-26: Dialog: Screen configuration (division of the new ACI screen after dividing)

7. Press "Delete screen segment".

   The focused screen segment is deleted and the dialog to fill the removed screen segment is adapted.

Configuring the User Interface

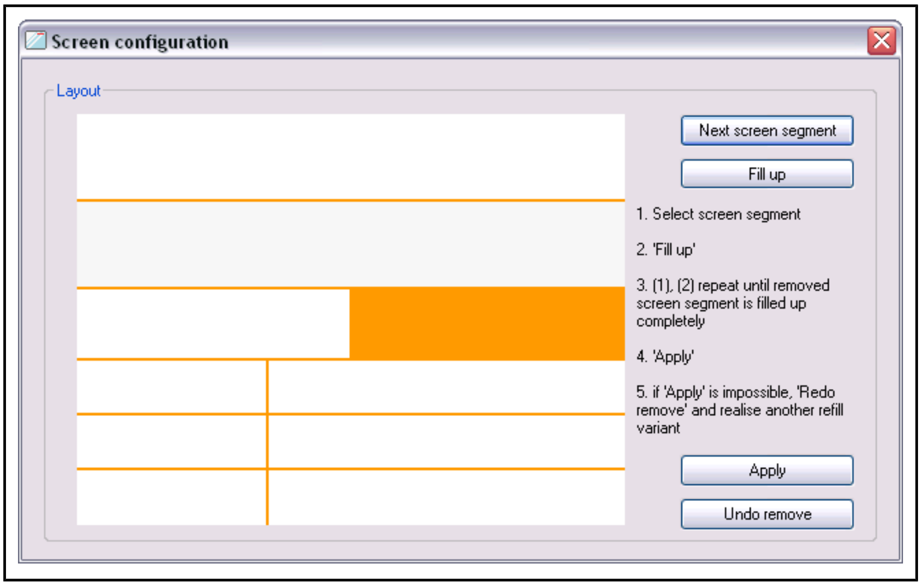


Fig.7-27: Dialog: "Screen configuration" (for filling the deleted screen segment)

8. Press "Next Subscreen" thrice.
9. Press "Fill".
10. Press "Accept".

    The focus moves to the screen segment to the left of the one that was deleted. This screen segment is expanded to the position of the deleted screen segment and filling is accepted.

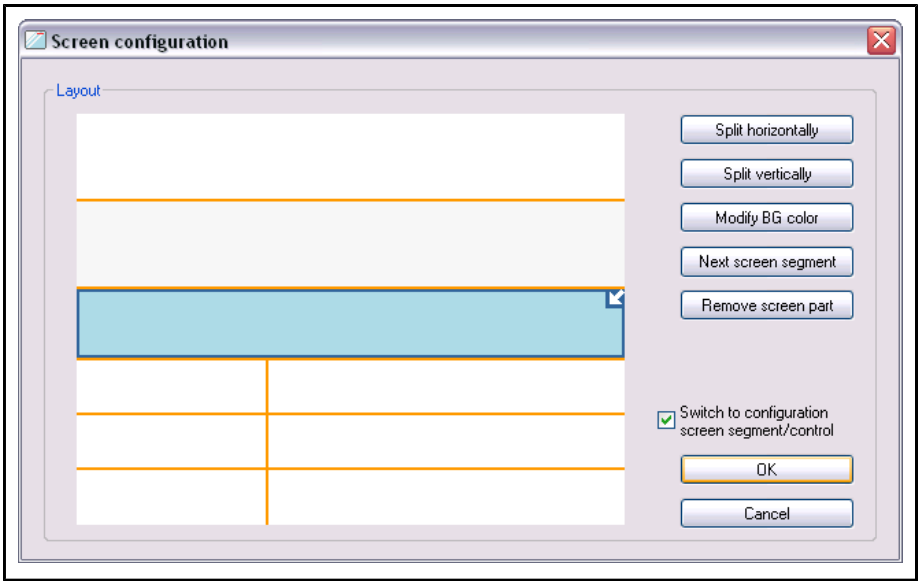


Fig.7-28: Dialog: "Screen configuration" (division of the new ACI screen after deleting and filling the screen segment)

☞ The deletion and filling of the screen segment undoes the vertical division.

11. Move the cursor to the uppermost horizontal screen segment divider.

    The cursor symbol becomes a double arrow.

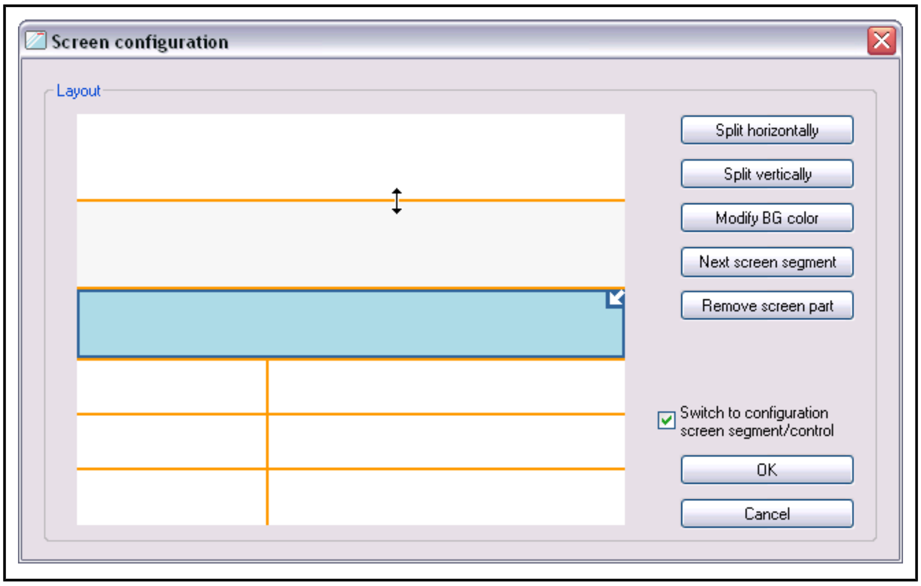


Fig.7-29: Dialog: "Screen configuration" (cursor symbol indicates the possibility for moving the screen segment divider)

12. Press and hold the left mouse button and move the divider somewhat using the mouse.

    The size of the adjacent screen segment changes accordingly.

13. Press "OK".

    The modified division of the new ACI screen is accepted and the dialog is closed.

☞ If the checkbox "Switch to Screen Segment/Control Configuration" is activated, the dialog "Screen Segment/Control Configuration" is opened automatically after the dialog has been closed.

IW Engineering / Configuration of ACI screens: Insert controls in the new ACI screen

1. In the dialog "Configuration screen", activate the checkbox "Switch to Configuration screen segment/control" and exit the dialog with "OK".

   – or –

   Click the node of the new ACI screen using the right mouse button.

   The context menu of the ACI screen opens.

2. In the context menu, click on "Screen Segment/Control Configuration".

   The "Screen Segment/Control Configuration" dialog is displayed.

Configuring the User Interface

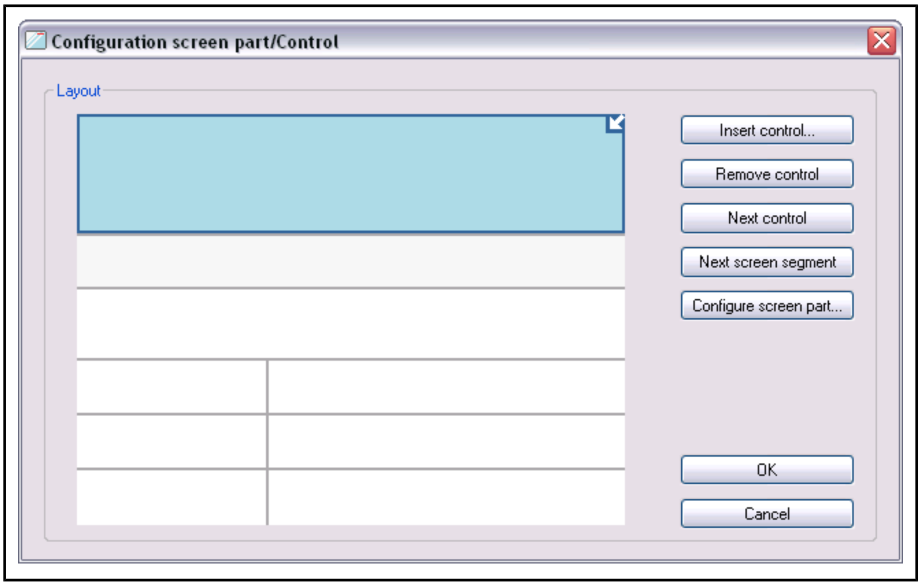


Fig.7-30: Dialog: "Screen segment/control configuration" (for the new ACI screen)

3. Press "Insert Control...".

   The "Insert Control" dialog is displayed.

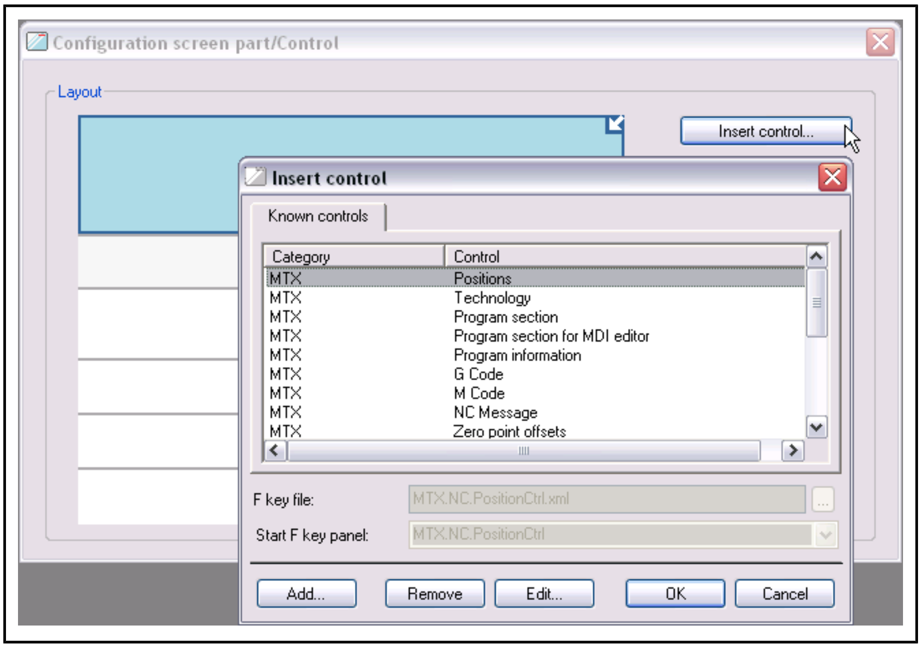


Fig.7-31: Dialog: "Insert control"

4. Select a control and click "OK" to confirm.

   The selected control is inserted into the focused screen segment.

5. Insert another control into the screen segment or proceed in the same manner in the other screen segments.

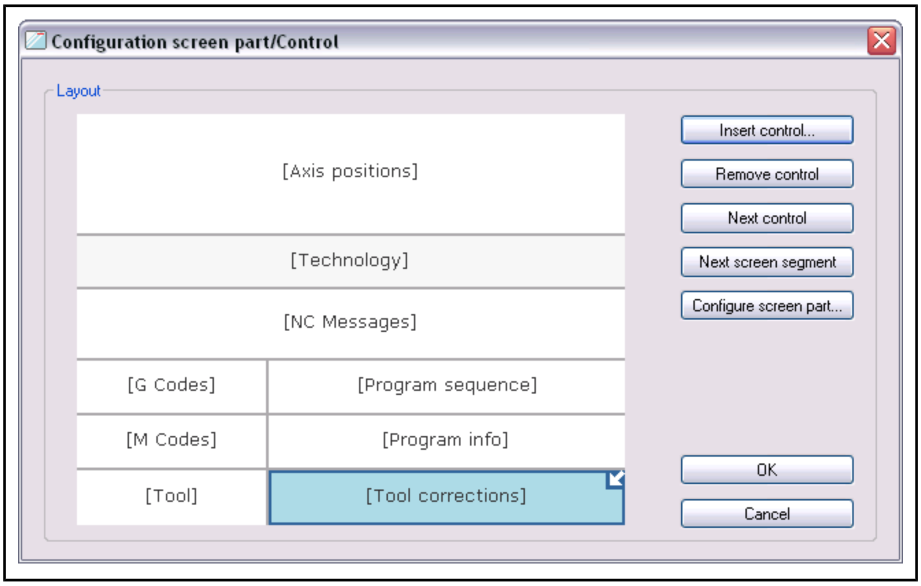


Fig.7-32: Dialog: "Screen segment/control configuration" (Example configuration of the new ACI screen with controls)

6. Press "OK".

   The modified configuration of the new ACI screen with controls is accepted and the dialog is closed.

IW Engineering / Configuration of ACI screens: Rename a new ACI screen

1. Click the node of the new ACI screen using the right mouse button.

   The context menu of the ACI screen opens.

2. In the context menu, click on "Rename".

   The name (screen ID) of the new ACI screen can be changed directly in the associated project node.

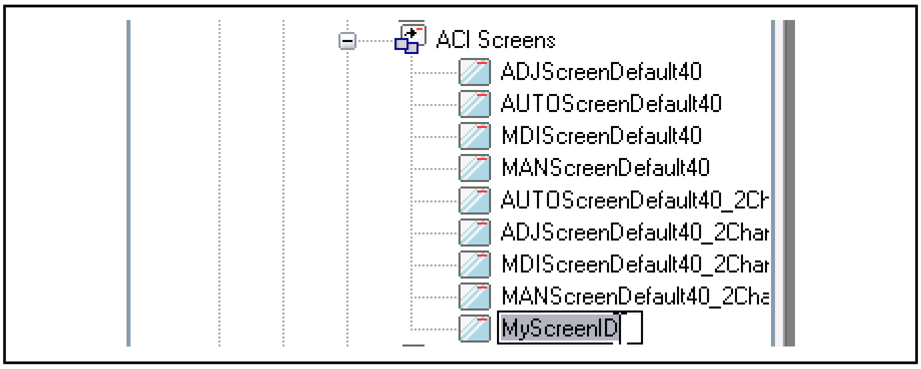


Fig.7-33: Renaming the new ACI screen in the project node

7.1.4 Integrating User Controls into ACI Screens

General

Short description

In the screen segments of an ACI screen, even user-created controls can be inserted. Thus, user-specific data visualization as well as its embedding in existing screens is possible. By using the ACI interface in user controls, it behaves like system-specific displays. Additionally, the user control can use the advanced ACI functionality to react on certain events and to communicate with its environment.

Integrating ActiveX Controls (Visual Basic 6)

Reporting the user control focusing to the ACI via mouse click event

One way of focusing a user control is using the F-key command "Next Window" to switch to the next screen segment. The other way is to click directly on the user control. In the second case, the user control must trigger the "Click" event so that the ACI is informed and thus displays the focus frame, the background color and the F-key bar belonging to the user control. The control should trigger the "MouseMove" event if the mouse curser is in the range of the user control.

Program:

```
Event Click()
```

☞ When focusing a user control, all user entries have to be directed towards this control.

When focusing an ACI control to the ACI ("Next Window") using a command, the ACI interface command "SetFocus()" of the control is additionally activated.

Setting the background color of a user control

The background color of a user control in the ACI depends on whether the user control has the focus and it is set by the ACI directly on the user control if the user control is provided with "BackColor" property.

To be able to set the focus-dependent background color for all user controls – whether this has been developed especially for the use in ACI – the user controls must be provided with the public property "BackColor". This is already implemented by default in the general controls, such as the Microsoft "Chart" Control. It is implemented for the ACI user controls in the user control using procedures Get and Let for the property "BackColor":

Program:

```
Public Property Get BackColor() As OLE_COLOR
    BackColor = UserControl.BackColor
End Property

Public Property Let BackColor(ByVal New_BackColor As OLE_COLOR)
    UserControl.BackColor = New_BackColor
End Property
```

It should be possible for the user to change the background color for the non-focused state during the configuration of the screens. There is one color for the focused state of the user controls and a choice of two colors (gray and white) for the non-focused state.

The default background color for the non-focused state is specified by the ACI when a user control is inserted in a screen. This background color can be changed by the user during the configuration of the screens. The subsequent change between the two background colors specified by the ACI is accomplished via a respective button. If a user control has a PropertyPage and/or an "Options" dialog in which the background color can also be changed, these forms can be opened and the background color can be changed via the respective button.

"Enabled" property of the user control must be "TRUE"

If the "Enabled" property of a user control is set to "FALSE", the ACI can be used to change the background color of the user control using "BackColor", but the user control cannot trigger the "Click" and "MouseMove" events and the ACI cannot call the control method "SetFocus". To provide a consistent function in the ACI, the "Enabled" property of the user control must be "TRUE".

☞ If the purpose of setting the "Enabled" property to "FALSE" is to prevent the user control from being focused, this can instead be accomplished by selecting "Configure screen segment" for the screen segment (see "Configure screen segment").

Integrating .NET User Controls

Ensuring visibility for COM

.NET user controls must be provided with the "COM-Visible" feature. The user control is to be registered on the target system as shown below:

```
regasm.exe ...\IndraWorks\MyACIControl.dll /tlb
```

Implementing and Using the ACI Interface

To implement the ACI interface in the user control, a reference on "MTXACIInterface.dll" is to be carried out first.

In Visual Basic 6 (VB6), the ACI interface is implemented using the command "Implements":

Program:

```
Implements MTXACIInterface.ACIInterfaceBase
```

In the following, the interface methods are listed and their usage is described.

CallCommand

Syntax: `(VB) Sub ACIInterfaceBase_CallCommand(ByVal strCommandName As String)`

`(C#) void ACIInterfaceBase.CallCommand(string strCommandName)`

Usage: This ACI directly sends individual (subsequently defined) and external (from F-keys or from other user controls) commands to the ACI control using this method. No parameters are transferred.

The transferred command has to be known to the ACI control in order to evaluate it and to act correspondingly. Can be mapped on CallCommandEx().

Parameters: strCommandName = Command name

Return value: -

CallCommandEx

Syntax: `(VB) Sub ACIInterfaceBase_CallCommandEx(ByVal strCommandName As String, strParameter() As String)`

`(C#) void ACIInterfaceBase.CallCommandEx(string strCommandName, ref System.Array strParameter)`

Usage: This ACI directly sends individual (subsequently defined) and external (from F-keys or from other user controls) commands to the ACI control using this method. Additional parameters are transferred.

The transferred command and the parameters have to be known to the ACI control in order to evaluate it and to act correspondingly. The command editing can be mapped on the CallCommand() if the parameter array is empty.

In the following, the subsequently defined commands by the ACI to report certain events to the ACI control are shown:

Language has been switched

`strCommandName = LanguageChanged`

`strParameter[0] = LCID (1031=DE / 1033=EN / ...)`

IndraWorks Operation is closed

`strCommandName = Terminate`

`strParameter[0] = -`

Parameters: strCommandName = Command name

strParameter = String array of parameters

Return value: -

DeleteConfig

Syntax: `(VB) Sub ACIInterfaceBase_DeleteConfig(ByVal strConfig As String)`

`(C#) void ACIInterfaceBase.DeleteConfig(string strConfig)`

Usage: This method is enabled by the ACI if the ACI control is removed again from the screen segment while configuring the ACI screen. The transferred configuration identifier is equal to the one requested with StoreConfig() from the ACI control by the ACI.

With this method, it is possible for the ACI control to delete the configuration data (e.g. from the profile section of a control-related storage file) created for the control instance to be removed since they are no longer required.

Parameters: strConfig = Configuration identifier specified by the ACI control

Return value: -

GetActionStatus This interface method is currently not active!

Syntax: `(VB) Function ACIInterfaceBase_GetActionStatus(ByVal strMethod As String) As ActionStatus`

`(C#) ActionStatus ACIInterfaceBase.GetActionStatus(string strMethod)`

Usage: If either the "CallCommand" method or the "CallCommandEx" method is assigned to the F-key of the focused control to be shown as callable function, the ACI asks the ACI control for support of the transferred command via the "GetActionStatus" method. Depending on the response of the ACI control - whether the command should currently be callable, the ACI enables/ disables the F-key or even hides it.

Due to the modified mechanism of the F-keys, the "GetActionStatus" query is currently not in use.

Parameters: strMethod = Command that should be called directly in the ACI control via Command()/CallCommandEx()

Return value: actEnabledact | Disabled | actHided

GetActivityStatus

Syntax: `(VB) Function ACIInterfaceBase_GetActivityStatus() As ActivityStatus`

`(C#) ActivityStatus ACIInterfaceBase.GetActivityStatus()`

Usage: This method is used to prompt the current state of activity of the ACI control.

The ACI can - but does not have - to wait before the ACI control is closed (when removing the control from the screen segment, deleting the complete ACI screen or closing the ACI screen) until its tasks are completed.

Parameters: -

Return value: aysReady = The control waits only for events triggered while modifying process data for example

aysWaiting = The control waits for data/response of another application for example

aysWorking = The control is currently processing/storing its data

SetActive

Syntax: `(VB) Sub ACIInterfaceBase_SetActive(ByVal aesActive As ActiveStatus)`

`(C#) void ACIInterfaceBase.SetActive(ActiveStatus aesActive)`

Usage: This method is used by the ACI to switch on or off the functions of the ACI control.

This message can be used by the ACI control to establish or cancel process connections (start or stop of the OPC communication). If the ACI control has already communicated, it is sufficient to switch the OPC groups only active or inactive.

Parameters: aesActive = aesON / aesOFF

Return value: -

SetActiveFocus

Syntax: `(VB) Sub ACIInterfaceBase_SetActiveFocus(ByVal blnActiveFocus As Boolean)`

`(C#) void ACIInterfaceBase.SetActiveFocus(bool blnActiveFocus)`

Usage: In IndraWorks Operation, this method is called in an ACI control if this ACI control is focused (shown with a blue frame and background) in the ACI screen displayed by triggering the ACI command "Set_ActiveFocus" (e.g. via F-key). Additionally, the Windows focus is set to the ACI control by the ACI if possible. Subsequently, the ACI control can transfer the Windows focus to the control element which expects the user to enter something via keyboard first.

In IndraWorks Operation, this method is also called if the focus does no longer remain with the ACI control since either another screen segment is focused or it is switched to a view of the ACI screen without focus.

Parameters: blnActiveFocus = TRUE: Control is focused | FALSE: Focus is removed from the control

Return value: -

SetChannel

Syntax: `(VB) Sub ACIInterfaceBase_SetChannel(ByVal strChannel As String)`

`(C#) void ACIInterfaceBase.SetChannel(string strChannel)`

Usage: This method reports the active channel of the control to the ACI control during instantiation and channel change.

This parameter is for example required to check in / out cannel-dependent items.

Parameters: strChannel = Active channel

Return value: -

SetConfig

Syntax: `(VB) Sub ACIInterfaceBase_SetConfig(ByVal strConfig As String)`

`(C#) void ACIInterfaceBase.SetConfig(string strConfig)`

Usage: By this method, the configuration identifier saved with StoreConfig() is reported to the ACI control during its instantiation. Thus, an individual and instance-specific view is possible for each ACI control.

Parameters: strConfig = Configuration identifier

Return value: -

SetDevice

Syntax: `(VB) Sub ACIInterfaceBase_SetDevice(ByVal strDevice As String)`

`(C#) void ACIInterfaceBase.SetDevice(string strDevice)`

Usage: This method is used to report the current device to the ACI control while it is instantiated. The transferred parameter is the active device GUID of the LDX that is required to establish an OPC process connection.

If the ACI control goes offline in IndraWorks Engineering - that means that no process connection is instantiated - the GUID "E61948E9-9913-47c0-ACDB-7C716902CD66" especially defined for this case is transferred.

Parameters: strDevice = GUID of the active device | "E61948E9-9913-47c0-ACDB-7C716902CD66"

Return value: -

SetFrameActivate

Syntax: `(VB) Sub ACIInterfaceBase_SetFrameActivate(ByVal blnFrameActivate As Boolean)`

`(C#) void ACIInterfaceBase.SetFrameActivate(bool blnFrameActivate)`

Usage: This method is used to report to all ACI controls of an ACI screen that the ACI screen looses or gets the Windows focus again. That means that it is hidden behind another program window or returns to the desktop. This can be the case if the screen or operating range changes in IndraWorks Operation.

Parameters: blnFrameActivate = TRUE: ACI screen contains Windows focus | FALSE: ACI screen is removed from Windows focus

Return value: -

SetSetupFocus

Syntax: `(VB) Sub ACIInterfaceBase_SetSetupFocus(ByVal blnSetupFocus As Boolean)`

`(C#) void ACIInterfaceBase.SetSetupFocus(bool blnSetupFocus)`

Usage: In IndraWorks Engineering, this method is called in an ACI control if this ACI control is focused (shown with a blue frame and background) in the ACI screen displayed by triggering the ACI command "Set_ActiveFocus" (e.g. via a respective button in the background). Additionally, the Windows focus is set to the ACI control by the ACI if possible. Subsequently, the ACI control can transfer the Windows focus to the control element which expects the user to enter something via keyboard first.

In IndraWorks Engineering, this method is also called if the focus does no longer remain with the ACI control since either another screen segment is focused or it is switched to a view of the ACI screen without focus.

Parameters: blnSetupFocus = TRUE: Control is focused | FALSE: Focus is removed from the control

Return value: -

SetVisible

Syntax: `(VB) Sub ACIInterfaceBase_SetVisible(ByVal blnVisible As Boolean)`

`(C#) void ACIInterfaceBase.SetVisible(bool blnVisible)`

Usage: This method reports to the ACI control that the top control is located in the screen segment and that it is switched visibly or invisibly together with the ACI screen.

During instantiation, the ACI control can use this method call to visibly switch all its visual elements at the same time. The switching depends on their size adaptations and initializations to ensure a smooth screen composition.

Parameters: blnVisible = TRUE: Control is visible | FALSE: Control is invisible

Return value: -

SetZoomMode

Syntax: `(VB) Sub ACIInterfaceBase_SetZoomMode(ByVal blnZoomMode As Boolean)`

`(C#) void ACIInterfaceBase.SetZoomMode(bool blnZoomMode)`

Usage: If the ACI gets the "Zoom_In" command (via F-key for example), it zooms in the focused screen visibly and thus the included controls up to the complete ACI screen width and height. The "Zoom_Out" command returns to the original size of the screen segment. After zooming, the ACI sends the "SetZoomMode" method with either "TRUE" or "FALSE" to the ACI control changed in size. Thus, the ACI controls are provided with information on their zoom state and can adapt visually to the new size (changing the number of lines or columns, the font size etc.).

Parameters: blnZoomMode =TRUE: ACI control is in zoom mode | FALSE: ACI control is in normal display mode again

Return value: -

ShowOptionDialog

Syntax: `(VB) Function ACIInterfaceBase_ShowOptionDialog() As Boolean`

`(C#) bool ACIInterfaceBase.ShowOptionDialog()`

Usage: If the ACI control can allow changes in its configuration settings by the user, it has to provide the respective "Options" dialog. The request to display this configuration dialog has to be sent to the ACI with the "Show_OptionDialog" via F-key. The ACI calls the "ShowOptionDialog" method in the focused ACI control.

The ACI control has to inform the ACI in the return value about how to proceed with possible changes.

Parameters: -

Return value: TRUE = Closes dialog with OK (changes saved)

FALSE = Dialog either not implemented or aborted with "Cancel"

StoreConfig

Syntax: `(VB) Function ACIInterfaceBase_StoreConfig() As String`

`(C#) string ACIInterfaceBase.StoreConfig()`

Usage: With this method, the ACI requests the ACI control to save its configuration settings instance-specifically (e.g. in an individual profile section of a control-related storage file). To identify the instance-related configuration data, the ACI control has to specify a unique identifier that is saved as return value by the ACI. During the next instantiation of the ACI control, the ACI returns this identifier to the ACI control (for the "SetConfig" method) so that it can parameterize respectively.

The "StoreConfig" method is called at the first instantiation of the ACI control (when inserting into a screen segment) in IndraWorks Engineering by the ACI. The ACI can use this call to display a configuration dialog to allow the user to set the initial settings.

In IndraWorks Operation, the ACI calls the method for a focused ACI control if the ACI control displayed the "Options" dialog previously (by calling the "ShowOptionDialog" method) and if the settings were saved (return value "TRUE"). The reason is that the ACI control might try to save the changed configuration settings under a different identifier that has to be queried and saved by the ACI.

If the ACI control was inserted in a screen segment in IndraWorks Engineering without instantiating it, the "StoreConfig" method is called in IndraWorks Operation but the "Options" dialog was not previously displayed. Thus, after starting IndraWorks Operation, a configuration dialog is displayed if the ACI control required it for the initial instantiation. The identifier queried via "StoreConfig" is only saved in the configuration mode of the ACI screen, after the display of the "Options" dialog, in IndraWorks Operation. Thus, it is required to first switch into configuration mode and then to open the configuration dialog ("Options" dialog) and to close the configuration mode after the changes have been completed and the prompt for saving has been confirmed. Even in case of a closed IndraWorks Operation, the project in IndraWorks Engineering can be opened [for the respective ACI screen, the dialog "Configuration screen segment/control" with the setting "Loading controls..." can be made (previously set in the dialog "Properties of All Screens", see chapter 7.1.2 "Configuration in Engineering Desktop" on page 118)]. The changes can also be made in the appearing configuration dialog of the ACI control and saved by closing the dialog "Configuration screen segment/control" with **OK** .

Parameters: -

Return value: Unique identifier of the instance-related saved configuration data

Command sequence for the ACI control instantiation

For the first / each following instantiation of an ACI control (in IndraWorks Engineering/Operation), the following commands are called in a specified sequence:

Program:

```
StoreConfig/SetConfig <Identifier>
SetDevice <E61948E9-9913-47c0-ACDB-7C716902CD66 or no. of the active device>
SetChannel <No. of the active channel>
SetActive aesON
SetVisible True
SetSetupFocus//SetActiveFocus True
```

Reaction on a Connection Abort

If the user interface detects that the connection to the control has been interrupted, the ACI informs all ACI controls using the command

`CallCommand("Terminate")`.

In this case, all ACI controls must execute a Disconnect() on the SCP server / OPC server. The connection can only be restored if all clients have logged off.

Query ACI configuration mode

If an ACI control is configured in an ACI screen in the IndraWorks Engineering, the GUID `"E61948E9-9913-47c0-ACDB-7C716902CD66"` is transferred with the

`SetDevice()`

interface method to identify a non-existent "pseudo" device. In this case, the ACI is in editing mode without connection to the control. The establishment of the control connection can be omitted for the ACI control.

Communication of User Controls via Notify Event

☞ The Notify event is only recommended for experienced user control engineers with an extended knowledge about ACI-internal communication options.

By using the "Notify" event, which was especially defined for the ACI, the user controls can communicate with the ACI and other controls within the ACI screen in IndraWorks Operation.

```
(VB6) Event Notify(sMethod As String, sParameters() As String)
```

```
(C#) public delegate void NotifyEventHandler(string sMethod, string[] sParameters); public event NotifyEventHandler Notify;
```

☞ The event declaration is strictly specified including the variable names and can thus be identified by the ACI.

Sending commands to the ACI

With "SendCommandTo_Application" as first parameter "sMethod" in the Notify event, it is possible to send the commands to the ACI. They can also be assigned to F-keys.

Program:

```
(C#)  if (Notify != null)
      {
        string[] strParams = new String[5];
        strParams[0] = "MTXACI"; //application name
        strParams[1] = "MTXACI"; //application name
        strParams[2] = "Show_FKeyPanel"; //ACI command
        strParams[3] = "MDIEditor_Panel"; //command parameter 1
        strParams[4] = "MTXMDIEditor.xml"; //command parameter 2
        Notify("SendCommandTo_Application", strParams); //command target
      }
```

Since the usage of this command direction is internal and thus not visible for the user, it is only suitable to a limited extent and has to be used carefully.

To trace changes better in the user interface, required command calls should be executed via F-keys by the user.

Sending commands to other controls

In IndraWorks Operation, a user control can send a command to all controls, all instances of a specified control or only to a specified instance of a specified control via the ACI interface methods "CallCommand" and "CallCommandEx" within the ACI screen.

The following transfer parameters of the "Notify" event can send a command:

sMethod	Explanation	sParameters[i]	Explanation
SendCommandTo_All	Sends commands to all controls	i = 0	Command to be sent
		i = 1	Parameter array of the command
SendCommandTo_Control	sends commands to all instances of a control	i = 0	ProgID of the control
		i = 1	Command to be sent
		i = 2	Parameter array of the command
SendCommandTo_Instance	sends command to a determined instance of a control	i = 0	ProgID of the control
		i = 1	TimeID of the control
		i = 2	Command to be sent
		i = 3	Parameter array of the command

In order to specify the ProgID and the TimeID for a control, these have to be read from the configuration file.

The TimeID - a type of instance identification - unambiguously identifies the instance of a control. It is created for the instance of a control if the control is inserted into a screen segment in IndraWorks Engineering using the "Configuration screen segment/control" dialog.

If no parameter array of the command is specified in the "Notify" event, the command is transferred by the ACI interface method "CallCommand" or by "CallCommandEx".

Describing the status line (MTX)

A user control can write its text messages in the status line of the ACI using a "Notify" event. Writing into the status line using the "Notify" event has the advantage that the way of the status line usage by the user controls is the same. Thus, the user controls do not have to reference an additional library.

The following transfer parameters of the "Notify" event can be used to access the status line:

sMethod	Explanation	sParame-ters[i]	Explanation
ClearStatus-barMessage	Clears the content of the status line	i=0	Corresponds to the message type: ClearStatusText | ClearInformation-Text | ClearErrorText
SetStatusbar-Message	Describes the con-tent of the status line	i = 0	Corresponds to the message type: SetStatusText | SetInformationText | SetErrorText
		i = 1	Text to be displayed

Program:

```
(VB6)  Dim ssmParam(1) As String
       ssmParam(0) = "SetInformationText"
       ssmParam(1) = "SetInformationText"
       Notify "SetStatusbarMessage", ssmParam
```

The text to be displayed may contain umlauts. But text containing an XML syntax is not allowed and not displayed.

User controls are only supposed to write messages in the status line of the messages focused. Otherwise, the origin of the message cannot be traced.

Configuring User Controls

ACI screen

User controls have to be inserted into the ACI screens like controls of the system.

The user controls have to save their individual configuration settings as well. Access to the options set by the user is to be executed via the PropertiesPages or "Options" dialogs (for ACI controls) by the user.

ACI control

Controls that were especially developed for their usage in ACI screens - the ones containing the ACI interface - should save their instance-related data (e.g. in an individual profile section of a control-related storage file) so that they access it via an identifier. This identifier is queried via the interface (StoreConfig()) by the ACI when the control is created the first time which is normally during insertion in a screen segment. Additionally, the identifier is also queried after each "Options" dialog if the ACI return the value "TRUE" for the ACI interface command "ShowOptionDialog". In each future instantiation of the control, the ACI transfers the noted identifier (SetConfig()) to the control so that it can be displayed instance-specifically.

7.1.5 F-key Functions for ACI Screens and Controls

Short description

To call certain actions of the ACI and specific methods of the ACI controls, the respective functions can be assigned to the F-keys.

Description

In the F-panel editor (fig. 7-34 "F-panel editor: Assign ACI functions to F-keys" on page 155), the functions "Command to ACI screen" and "Command to ACI control" can be used to assign ACI functionalities to F-keys.

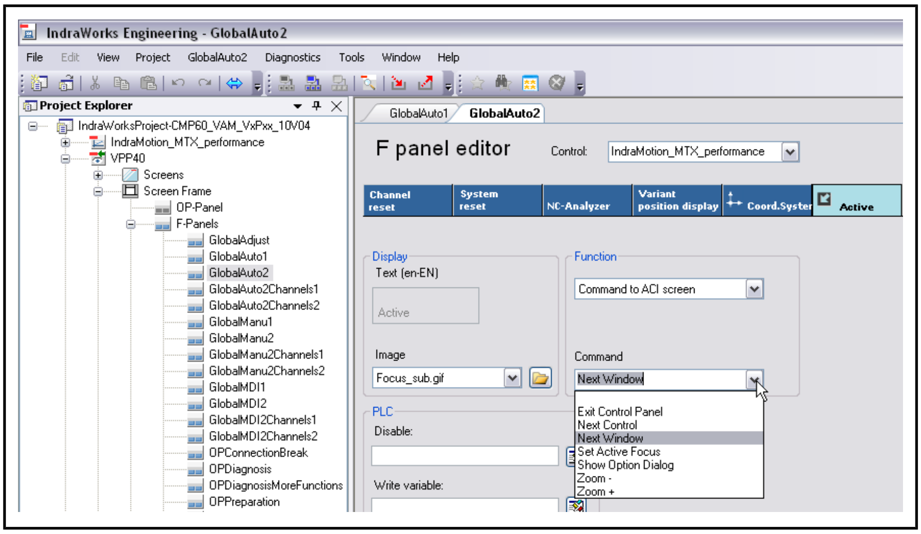


Fig.7-34: F-panel editor: Assign ACI functions to F-keys

The following commands can be sent to the ACI via "Command to ACI screen". The commands can be assigned to the F-panel for an ACI screen (A1), an ACI control (A2) or/and a control without ACI interface (A3).

- Exit Control Panel = ("Exit_Control_Panel") Exits the focusing of the active screen segment and shows the F-panel belonging to the ACI screen. (A2, A3)
- Next Control = ("Next_Control") Changes to the next control and shows its F-panel within the screen segment. (A2, A3)
- Next Window = ("Set_ActiveFocus") Changes the focus to the next screen segment "from left to right - from top to bottom" and shows the F-panel of the control displayed in the screen segment. (A2, A3)
- Set Active Focus = ("Set_ActiveFocus") Focuses the latest active screen segment and shows the F-key panel of the control displayed in the screen segment. (A1)
- Show Option Dialog = ("Show_OptionDialog") Shows the "Options" dialog or the PropertyPages of the control if it was implemented by the control. (A2, A3)
- Zoom + = ("Zoom_In") Zooms in the active screen segment to the complete screen width and height. (A2, A3)
- Zoom - = ("Zoom_Out") Zooms out the zoomed screen segment to its original size. (A2, A3)
- Additionally, the commands "Zoom_In", "Zoom_Out", "Show_Control", "Call_ICallCommand", "Call_ICallCommandEx" and "SendCommand-To_..." that require additional parameters (P) can be entered. The respective command and its individual parameters are separated by spaces.
  - Zoom_In [P1 [P2 [P3 [P4]]]] = It is like +, but it provides additional options. (A2, A3)

P1=""/"Name"	F-panel is not switched / panel to be displayed
P2=0/1,..,n	Focused screen segment (default) / indexed screen segment
P3=-1/0/1	Not focused (default) / only focused in zoom mode (only if it is possible to be focused) / keep focus even after zoom mode (not possible if it cannot be focused)
P4=0/1	not to be focused if not configured as focusable / focusable in zoom mode even if it is not configured as focusable (forcing focus)

Parameters in []-brackets are optional. They have to be specified to use the following parameter.

– Zoom_Out [P1] = It is like a zoom -, but provides additional options. (A2, A3)

P1=""/"Name"	Default F-panel of the control / F-panel to be displayed

– Show_Control P1 P2 [P3] = Switches to a specified control in a screen segment and sets/removes the screen segment focus. (A1, A2, A3)

P1=-1/0/1,..,n	Next control / only screen segment to be focused / indexed control
P2=0/1,..,n	In the focused screen segment / in the indexed screen segment
P3=""/0/1	Keep focus unchanged (default) / remove focus / set focus

– Call_ICallCommand <M> = Transfers the given method (M) to the focused ACI control in the ACI interface command "CallCommand". This method must be known to react respectively. (A2)

– Call_ICallCommandEx <M> [<P1> <P2> ...] = Transfers the given method with the parameters to the focused ACI control in the ACI interface command "CallCommandEx". (A2)

– SendCommandTo_All <M> [<P1> <P2> ...] = Transfers the specified method with or without parameters to all ACI controls in the ACI interface command "CallCommand" or "CallCommandEx". (A1, A2)

– SendCommandTo_Control <ProgID> <M> [<P1> <P2> ...] = Transfers the specified method with or without parameters to all ACI controls containing the specified ProgID in the ACI interface command "CallCommand" or "CallCommandEx". (A1, A2)

– SendCommandTo_Instance <ProgID> <TimeID> <M> [<P1> <P2> ...] = Transfers the specified method with or without parameters to the ACI control containing the specified ProgID and TimeID in the ACI interface command "CallCommand" or "CallCommandEx". (A1, A2)

☞ In order to specify the ProgID and the TimeID for a control, these have to be read from the configuration file.

The ACI interface commands "CallCommand" and "CallCommandEx" can be entered using the "Command to ACI control" function (see above). The ProgID of the ACI controls has to be in the front as additional obligatory parameter. Only the method with or without parameters has to be transferred.

- <ProgID> Call_ICallCommand <M> = Transfers the given method (M) to the ACI control with the specified ProgID in the ACI interface command "CallCommand". This method must be known to react respectively. (A1, A2)
- <ProgID> Call_ICallCommandEx <M> [<P1> <P2> ...] = Transfers the specified method with parameters to all ACI controls with specified ProgID in the ACI interface command "CallCommandEx". (A1, A2)

7.1.6 Using ACI Screens as NC Screens (MTX)

Brief Description

ACI screens used to display different processes an called by a respective the "Screens MTX" project node whenever a channel-dependent event occurs, are also named NC screens.

Description

In the Engineering Desktop, it is configured below the "Screens MTX" node that either an HMI screen or an ACI screen are called together with M-keys and F-keys in the Operation Desktop when a channel-dependent event occurs (changing an SCP item or a PLC variables) (fig. 7-35 "Project node "Screens MTX" (Mode Automatic)" on page 158). ACI screen used that way are also name NC screens.

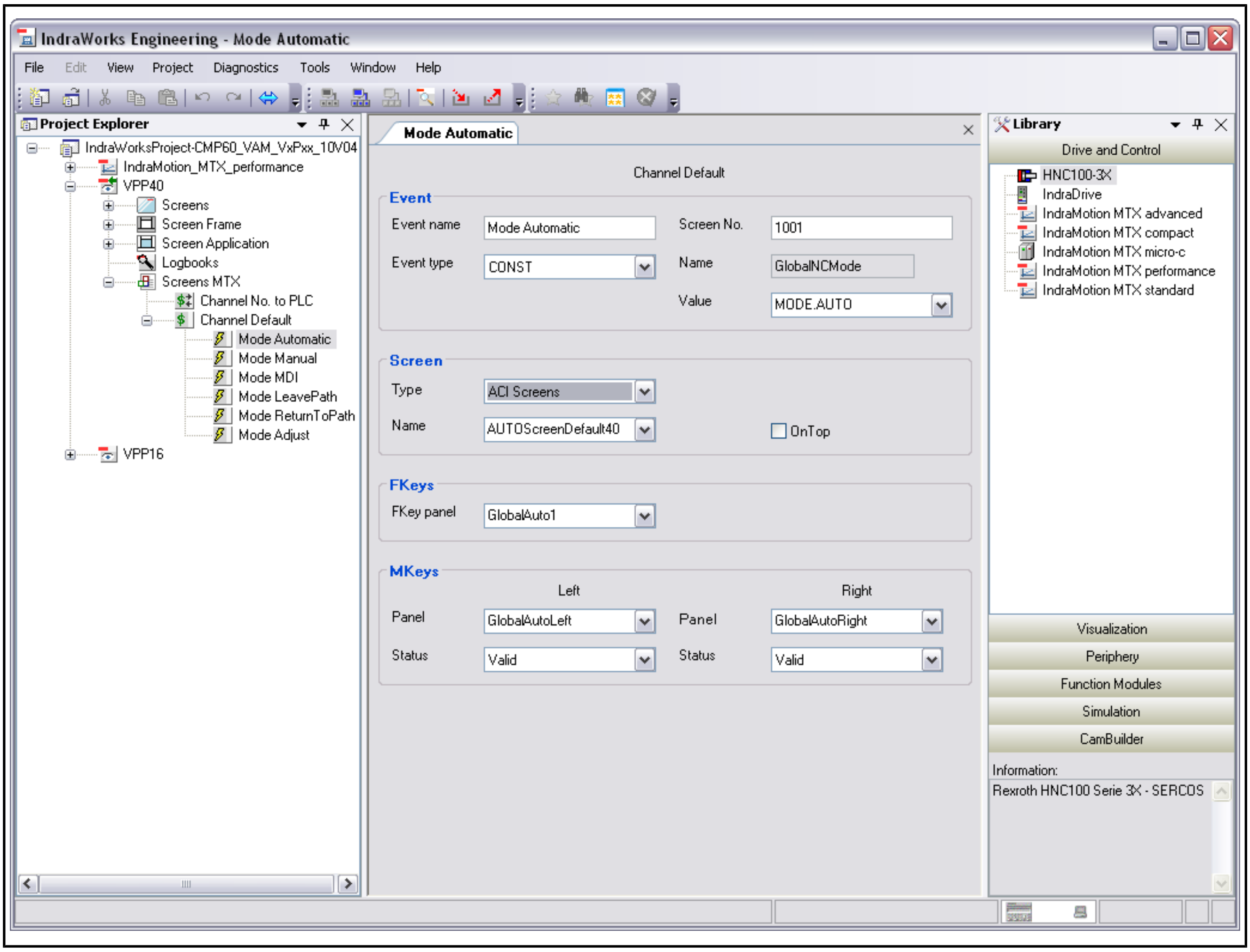


Fig.7-35: Project node "Screens MTX" (Mode Automatic)

For every operation mode, there is a separate predefined standard NC screen that contains the most common process data displays for that operation mode.

7.2 Standard NC Screens

7.2.1 Overview

Brief Description

In the example shown in fig. 7-36 "Default NC screens" on page 159, the machine operator panel contains the following display windows:

- Header (at the top; in the example, it contains the text "NC hardware: Battery 1 is discharged or connected with incorrect polarity")
- Axis position (directly below the header)
- Technology (directly below the display "axis positions")
- Program section (left, below the display "technology")
- M-codes (right, below the display "technology")
- G-codes (directly below the display "M-codes")
- ZO (directly below the display "G-codes")
- Tool (directly below the display "ZO")

Fig. 7-36: Default NC screens

Additional display windows become visible by selecting a display window and then press <F3> ("Next Display").

Variants of a display window become visible via <F2> ("Variant") when a display window has been selected.

A display window is selected by clicking with the mouse or by pressing <F7> ("Next Window").

The exact appearance of the individual displays can be configured as follows:

- Select a display window and press <F8> ("options"); double-click the header if the header is to be configured.

Additional setting options exist in the configurator:

- Open IndraWorks Engineering.
- Double-click configuration.

  ⇒ The configurator opens.
- Click the user interface data groups.

  ⇒ The setting options shown in the following figure appear.

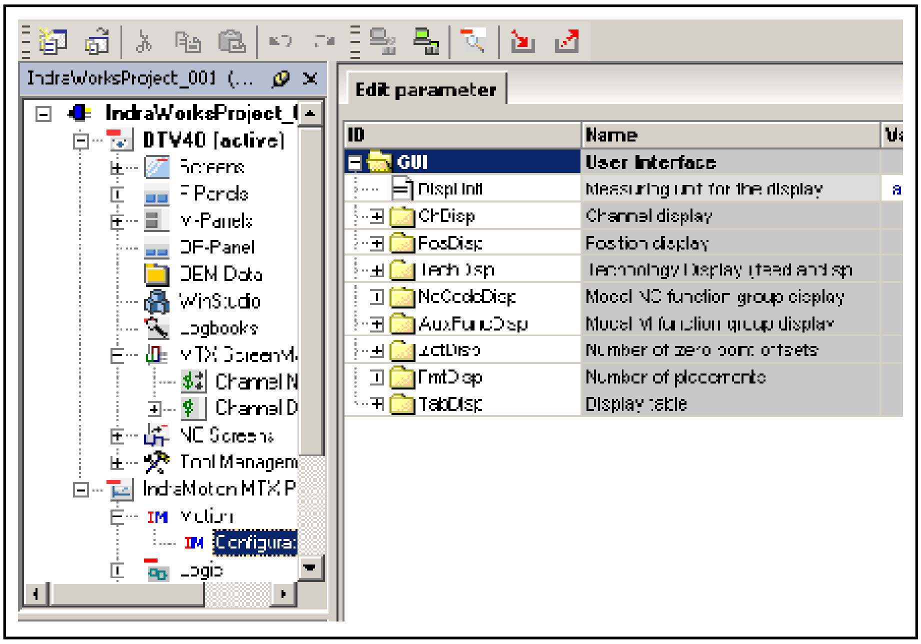


Fig.7-37: Configurator: User interface (GUI) data group

7.2.2 System Messages

General

Brief Description The system message window can be configured using the "Configuration - System Messages" dialog window.

Description The "Configuration - System Messages" dialog window can be opened and edited by double-clicking in the system data area.

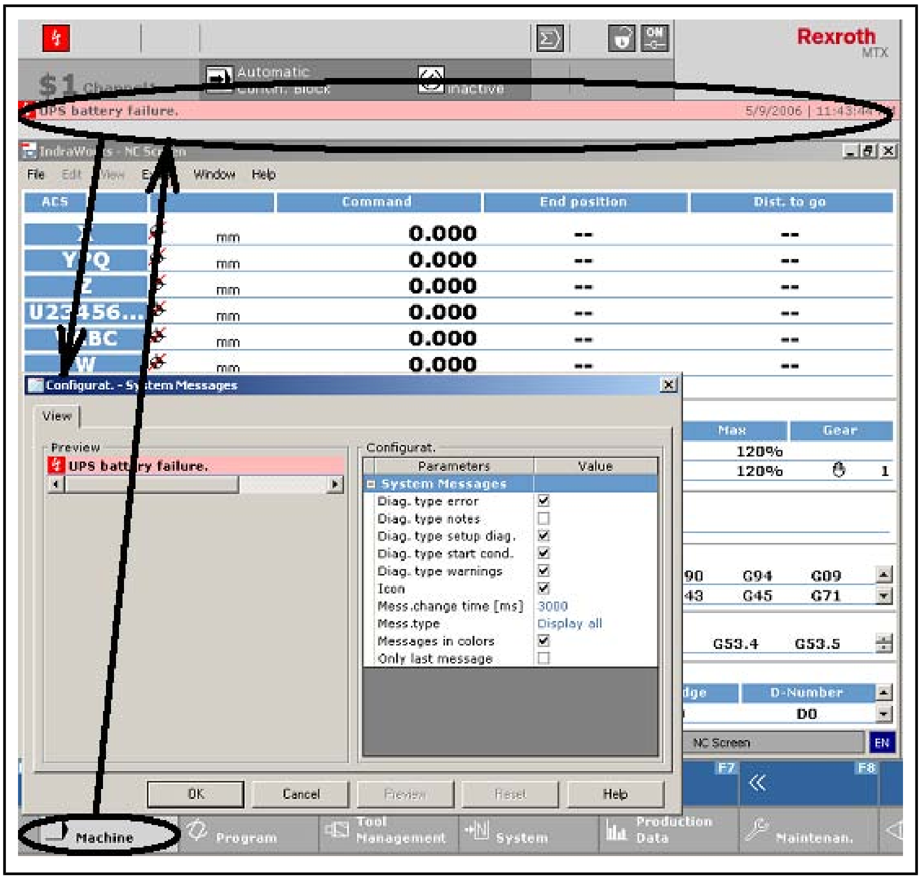


Fig.7-38: Configuration

Use this box to select the system messages to be displayed

- Setup diagnostic messages
- Error
- Notes
- Start conditions
- Warnings

.

If desired, an icon can be displayed in addition to the message text, e.g. a white lightning bolt with a red background in fig. 7-38 "Configuration" on page 161, or the message type "All", "ProVi" or "MTX" can also be issued.

Furthermore,

- The timing for which the messages are to be updated (e.g. every 3000 milliseconds) can be specified,
- It can be specified whether the messages are to be output in color or in black-and-white, and
- It can be specified whether all messages are to appear alternating with the system message window, or only the last.

Boundary Conditions The settings go into effect as soon as the dialog window is exited by pressing "OK".

Handling Instruction: Configure the System Message Window

This handling instruction describes how the system message window can be configured using the "System Message Configuration" dialog window.

IW Operation: Open IW Operation

1. On the desktop, double-click the <IW Engineering> icon

   – or –

2. **Windows Start button ▸ Programs ▸ Rexroth ▸ MTX ▸ MTX Control**

		Documentation chapter 7 "Configuring the User Interface" on page 117
Documentation:	IndraWorks Commissioning	HMI configuration

IW Operation: Open the dialog window "Configuration - System Messages"

1. Select the "Machine" operation mode.
2. Double-click the system message window in the header.

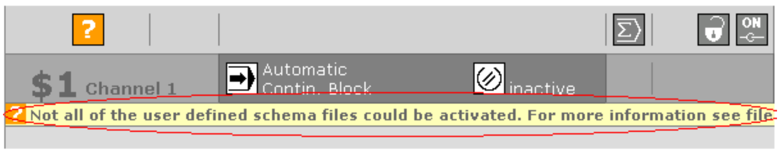


Fig.7-39: System message window within the header

3. Carry out the desired changes in the open "Configuration - System Messages" dialog window.

		Documentation chapter 7 "Configuring the User Interface" on page 117
Documentation:	IndraWorks Commissioning	HMI configuration

7.2.3 Position Display

Description

1. Select the "Machine" operating area.
2. Select the position display by clicking the mouse or by pressing (several times, if desired) <F7> (Next window).
3. Click the "Options" <F>key.

   The window "Configuration - Axis Positions" appears.

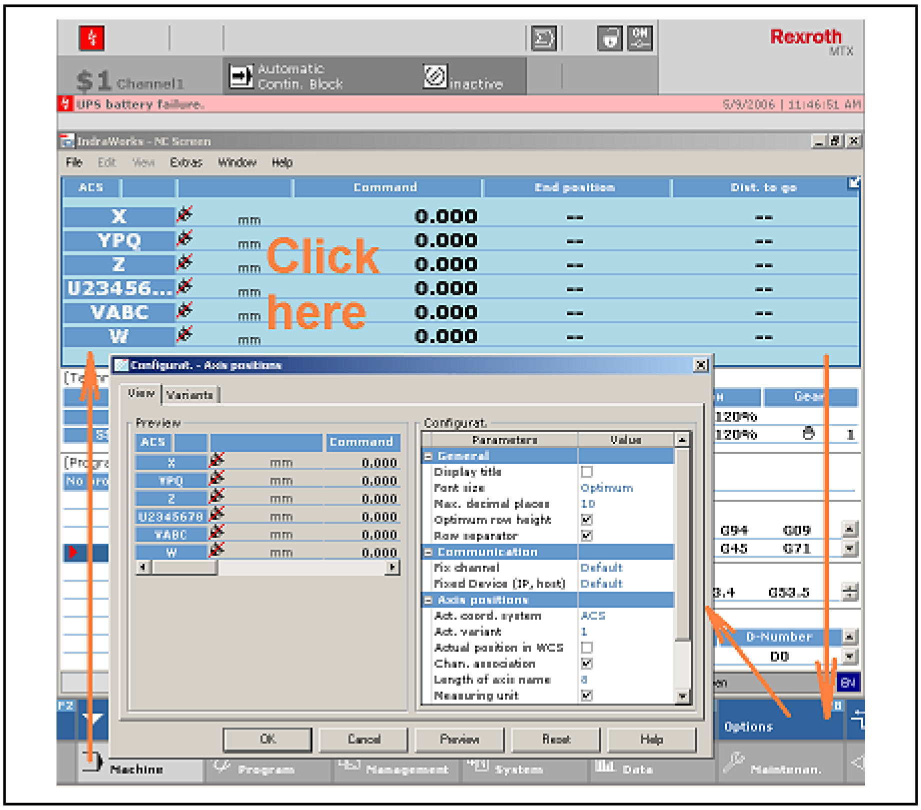


Fig.7-40: Dialog: Configuration axis positions (view)

4. Click the tab "Variants".

⇒ The following window appears:

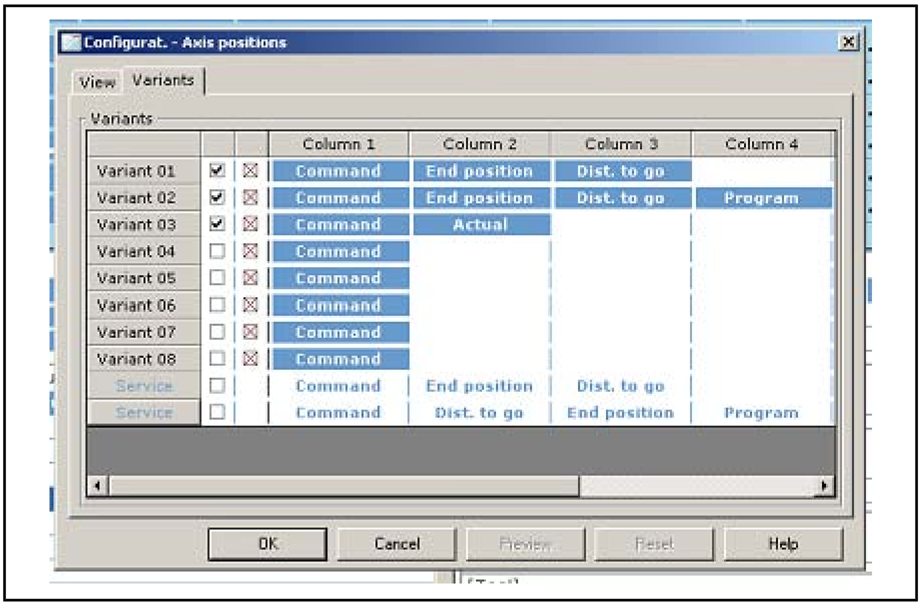


Fig.7-41: Dialog: Configuration axis positions (variants, columns 1-4)

5. Columns 5-6 can be viewed using the horizontal scroll bar.

Configuring the User Interface

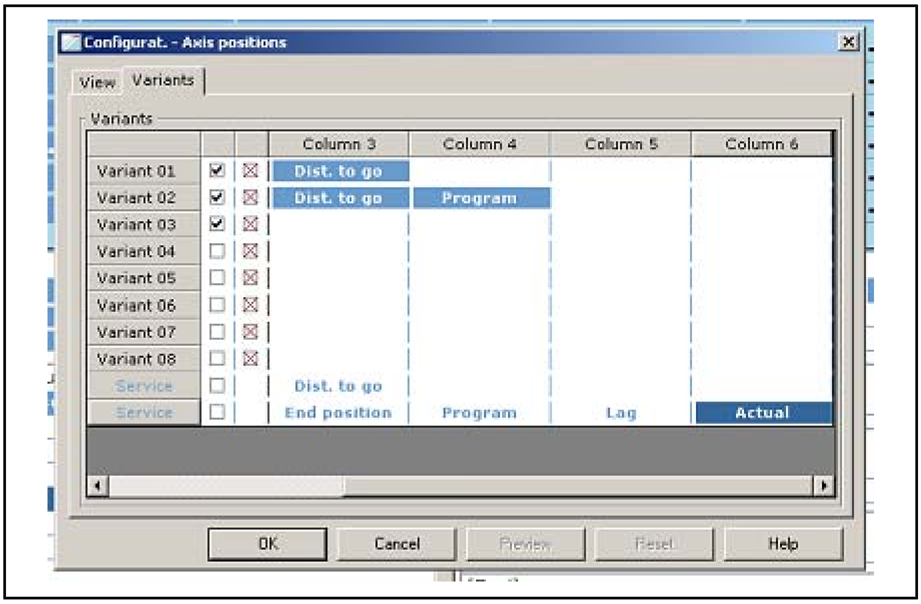


Fig.7-42: Dialog: Configuration axis positions (variants, columns 4-6)

6. The tab "Variants" can be used to specify what you want to see in the position display columns for the axes/coordinates shown.
7. Activate the checkmarks for variants 01, 02 and 03.
8. For example, click on the entry in the 1st column for variant 1. In the example, "Command Position" is in this column.

   A small arrow appears next to the word "Command Position".

9. Click this arrow.

   A selection box appears in which you can choose between Command position, Distance to go, End position, Program (programmed position), Coasting and Actual position by clicking the mouse.

10. Select "Actual Position".
11. In the same way, replace the 2nd entry (2nd column) of variant 1 by "Coasting" and the 3rd entry by "Program".
12. Click the empty field behind the 3rd entry of variant 1.

    A small arrow appears.

13. Click this arrow.

    A selection box appears in which you can choose between Command position, Distance to go, End position, Program (programmed position), Coasting and Actual position by clicking the mouse.

14. Select "Command Position".
15. In the same manner, you can make a selection for column 5 and **then** one for column 6.
16. Also make a selection for variants 02 and 03.
17. Exit "Configuration - Axis Positions" by pressing "OK".
18. If you press <F2>, scroll through the variants selected with a checkmark.

    However, note that the positions display must be selected, i.e. has a blue background, to do this.

19. Open "Configuration - Axis Positions" again and go to the tab "View".

    While you used tab "Variants" to specify which position data you want to display, you can specify the display method in this tab. These settings options are divided into the 3 suboptions "General", "Communication" and "Axis Positions".

    - Under "general", it is a good idea to leave the font size and line height on the "optimum" setting because only the size of the position display window is adapted. Of course, you can also select a fixed setting for the font size by clicking the "optimum" value; as a result, an arrow appears, which you must click to make the setting.
    - You can use "communication - fixed channel" to specify the channel whose axes/coordinates you want to have displayed. If you select "default", the currently active channel is always displayed.

      You can use "axis positions" to specify which information you want to see for every axis/coordinate, e.g.:

      – Channel assignment, i.e. the number of the channel to which the axis/coordinate belongs
      – Measurement unit, i.e. you can decide whether you want to display the unit of measurement in which the position display occurs or not

        Checking "phys. axis name" means that this is displayed in parentheses in addition to the logical axis name.
      – Symb. Diameter means that a corresponding symbol is displayed in the case of diameter programming for the affected axis.
      – Symb. Unit means that the selected unit is displayed. symb. coord. system means that the coordinate system is displayed (e.g. WCS, BCS, MCS, ACS).

20. After the selection has been made under "View", exit "Configuration - Axis Positions" by pressing "OK".

Pressing <F3> switches back and forth between the coordinate systems. However, only the coordinate systems whose parameters have been set as selectable in the configurator can be displayed.

Therefore, if the setting shown in the following figure is used, only the WCS and ACS coordinate systems can be selected.

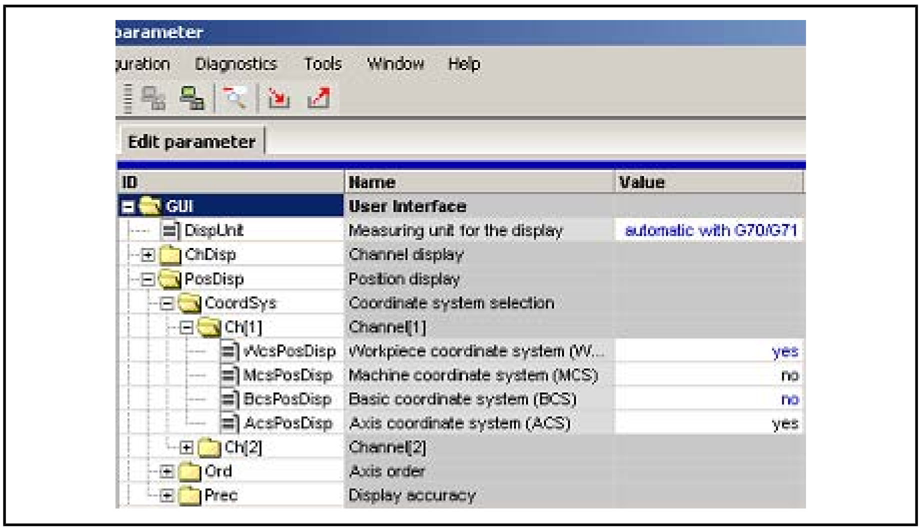


Fig.7-43: Specifying the selectable coordinate systems in the configurator

Configuring the User Interface

Only the channel axes for which a corresponding specification has been made in the configurator are displayed in the channel window (upper portion of the position display). As a result, the setting shown in fig. 7-43 "Specifying the selectable coordinate systems in the configurator" on page 165 specifies that channel axis 2 of channel 1 is not to be displayed in the channel window.

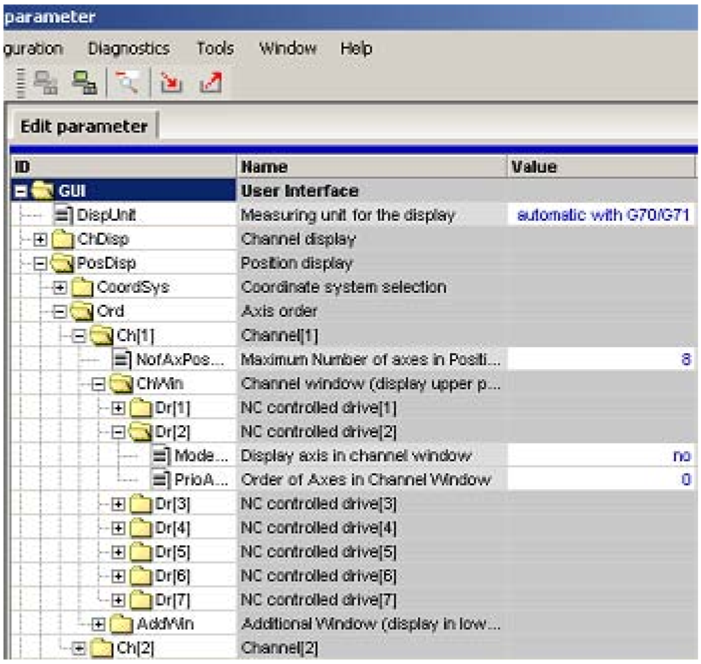


Fig.7-44: Specifying the channel axes to be displayed in the upper portion of the position display

Only the channel axes for which a corresponding specification has been made in the configurator are displayed in the auxiliary window (lower portion of the position display). As a result, the setting shown in fig. 7-45 " Specifying the channel axes to be displayed in the lower portion of the position display" on page 167 specifies that channel axis 2 of channel 1 is to be displayed in the auxiliary window only if it is in channel 1.

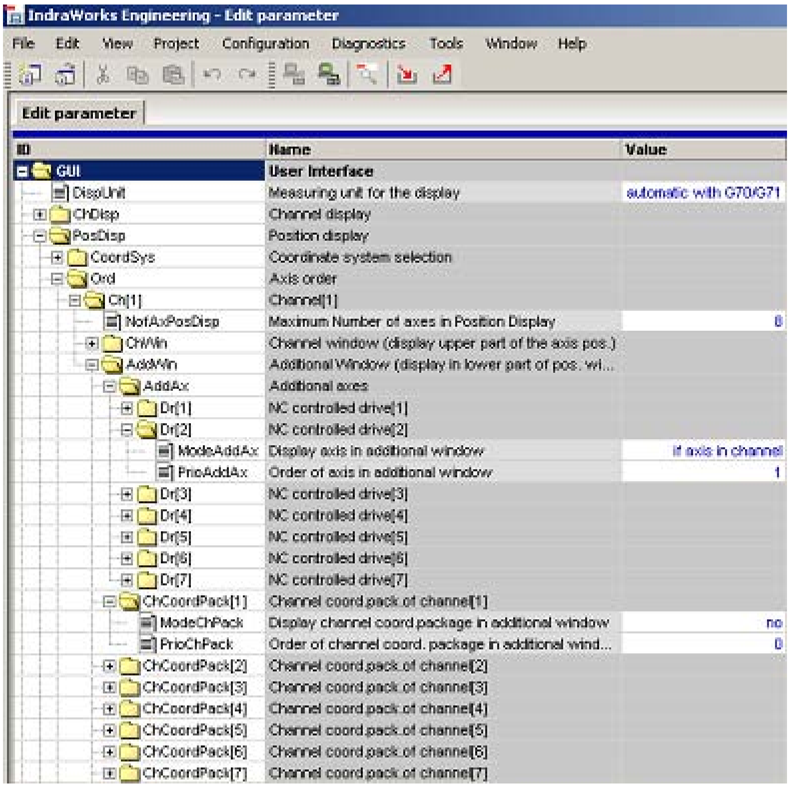


Fig.7-45: Specifying the channel axes to be displayed in the lower portion of the position display

The precision of the position display can also be specified in the configurator. See fig. 7-46 "Specifying the precision of the position display in the configurator" on page 167.

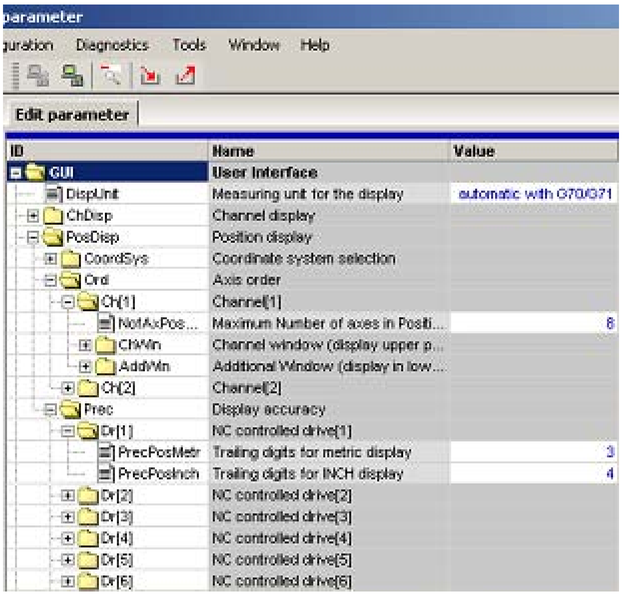


Fig.7-46: Specifying the precision of the position display in the configurator

7.2.4 Technology Display

Description

1. Select the "Machine" operating area.

Configuring the User Interface

2. Select the technology display by clicking the mouse or by pressing (several times, if desired) the <F7> key (Next window):

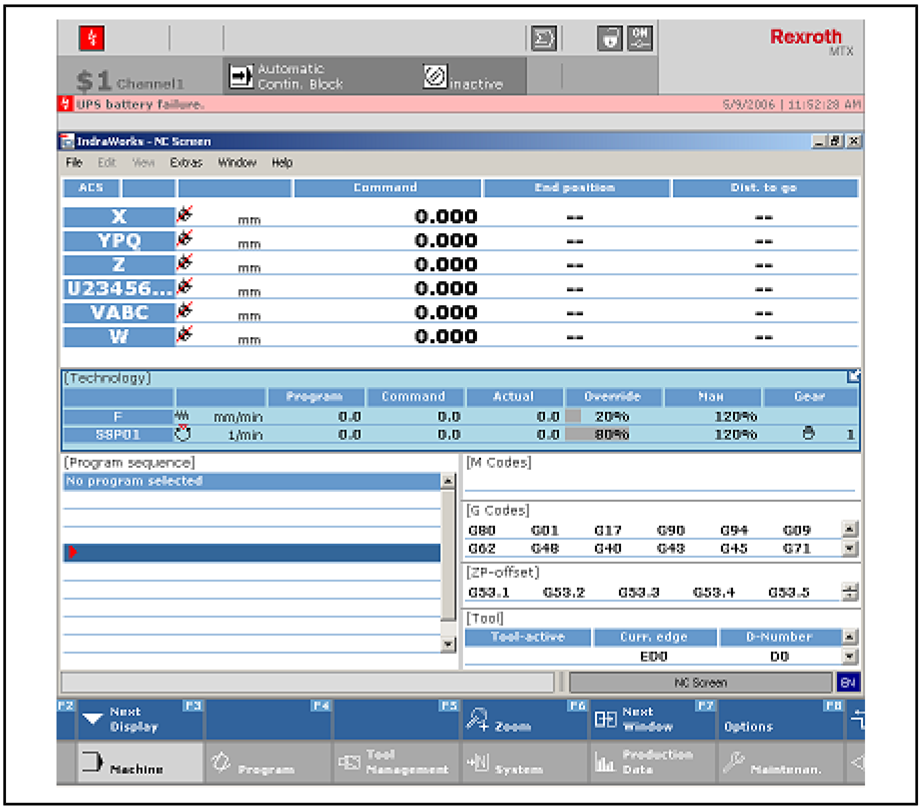


Fig.7-47: Technology display (blue background)

3. Press <F8> (Options).

The "Configuration - Technology" window opens:

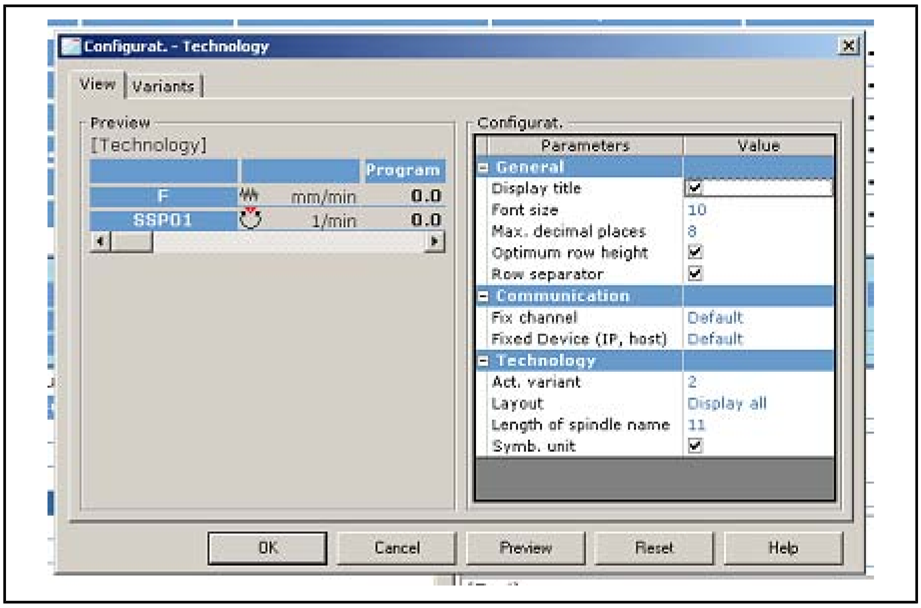


Fig.7-48: Dialog: Configuration - Technology

4. Click the tab "Variants".

⇒ The following window appears:

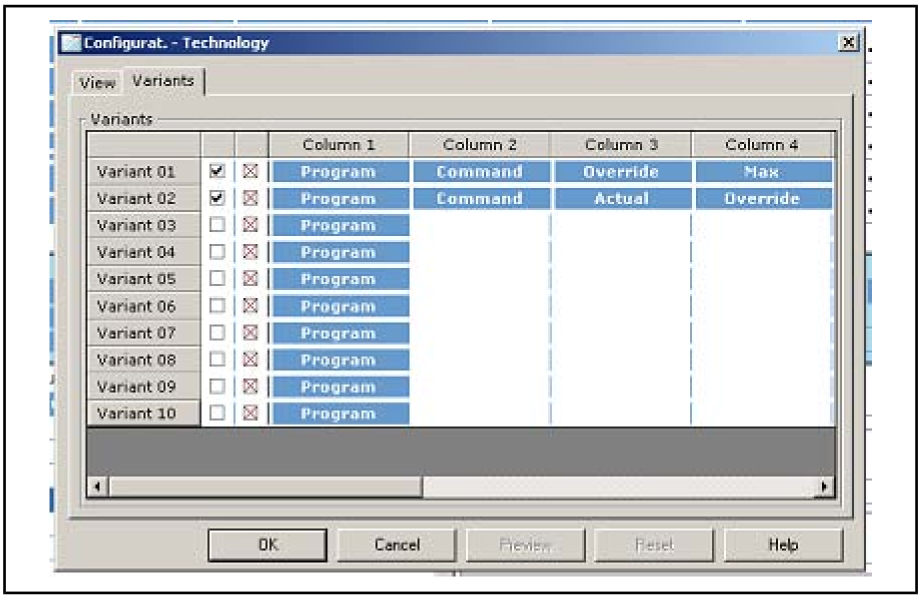


Fig.7-49: Dialog: Configuration technology (variants, columns 1-4)

5. Columns 5-8 can be viewed using the horizontal scroll bar:

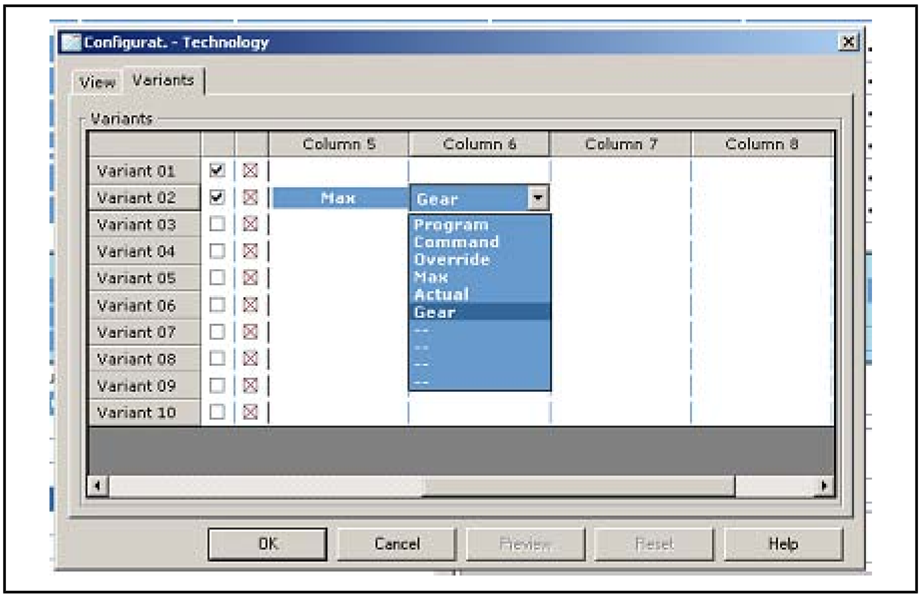


Fig.7-50: Dialog: Configuration technology (variants, columns 5-8)

6. The tab "Variants" can be used to specify what you want to see in the technology display columns for the technology values shown.
7. Activate the checkmarks for variants 01, 02 and 03.
8. For example, click on the entry in the 1st column for variant 1. In the example in fig. 7-52 "Displaying the NC program section that is currently being processed" on page 172, "Program" is shown in this column.

   An arrow appears next to the word "Program".
9. Click this arrow.

A selection box appears to choose between "Program", "Command value", "Override", "Max", "Actual value" and "Gear" via mouse click.

10. Select "Max".
11. In the same way, replace the 2nd entry (2nd column) of variant 1 by "Gear ", the 3rd entry by "Actual Value" and the 4th entry by "Override".
12. Click the empty field behind the 4th entry of variant 1 (5th column).

    An arrow appears.

13. Click this arrow.

    A selection box appears to choose between "Program", "Command value", "Override", "Max", "Actual value" and "Gear" via mouse click.

14. Select "Override".
15. In the same manner, you can make a selection for column 6 and **then** for the columns 7 and 8.
16. Also make a selection for variants 02 and 03.
17. Exit "Configuration - Axis Positions" by pressing "OK".
18. If you press <F2>, scroll through the variants selected with a checkmark.

    However, note that the technology display must be selected, i.e. must be highlighted in blue (as shown in fig. 7-47 " Technology display (blue background)" on page 168), to do this.
19. Open "Configuration - Technology" again and go to the tab "View".
20. While you used tab "Variants" to specify which technology data you want to display, you can specify the display method in this tab. These settings options are divided into the 3 suboptions "General", "Communication" and "Axis Positions".
    - Under "general", it is a good idea to leave the font size and line height on the "optimum" setting because only the size of the technology display window is adapted. Of course, you can also select a fixed setting for the font size by clicking the "optimum" value; as a result, an arrow appears, which you must click to make the setting.
    - You can use "communication - fixed channel" to specify the channel whose technology data you want to have displayed. If you select "default", the currently active channel is always displayed.
    - You can use "technology" to specify which information you want to see for each technology datum (e.g. the layout, where you can choose between "feed", "spindle" and "all".
21. After the selection has been made under "View", exit "Configuration - Axis Positions" by pressing "OK".

Additional settings for the technology display can be made in the configurator:

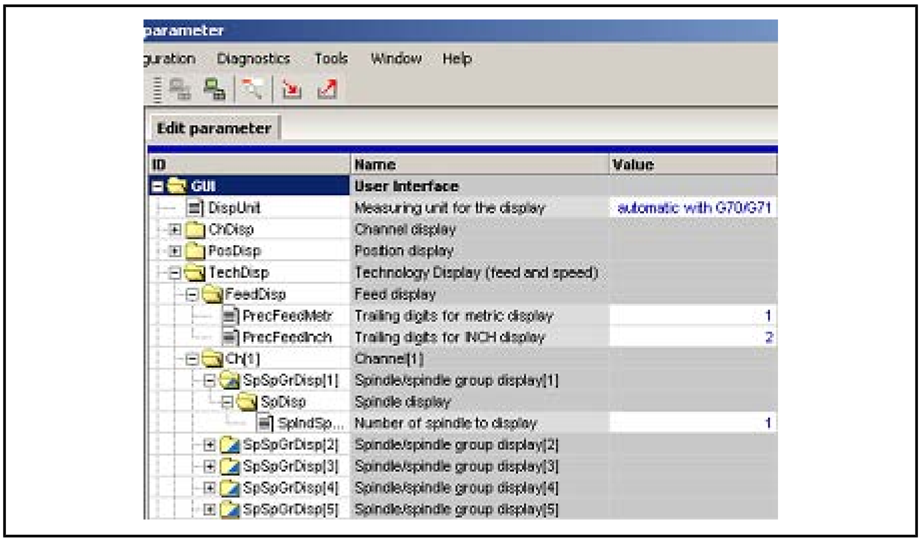


Fig.7-51: Settings options for the technology display in the configurator

7.2.5 Display of the Program Section

Description

1. Select the "Machine" operating area.
2. Display the NC program section that is currently being processed by clicking the mouse or by pressing (several times, if desired) <F7> (Next window) (fig. 7-52 "Displaying the NC program section that is currently being processed" on page 172).
3. Press <F8> (Options).

   The "Configuration - Program Section" window opens (fig. 7-52 "Displaying the NC program section that is currently being processed" on page 172).

Configuring the User Interface

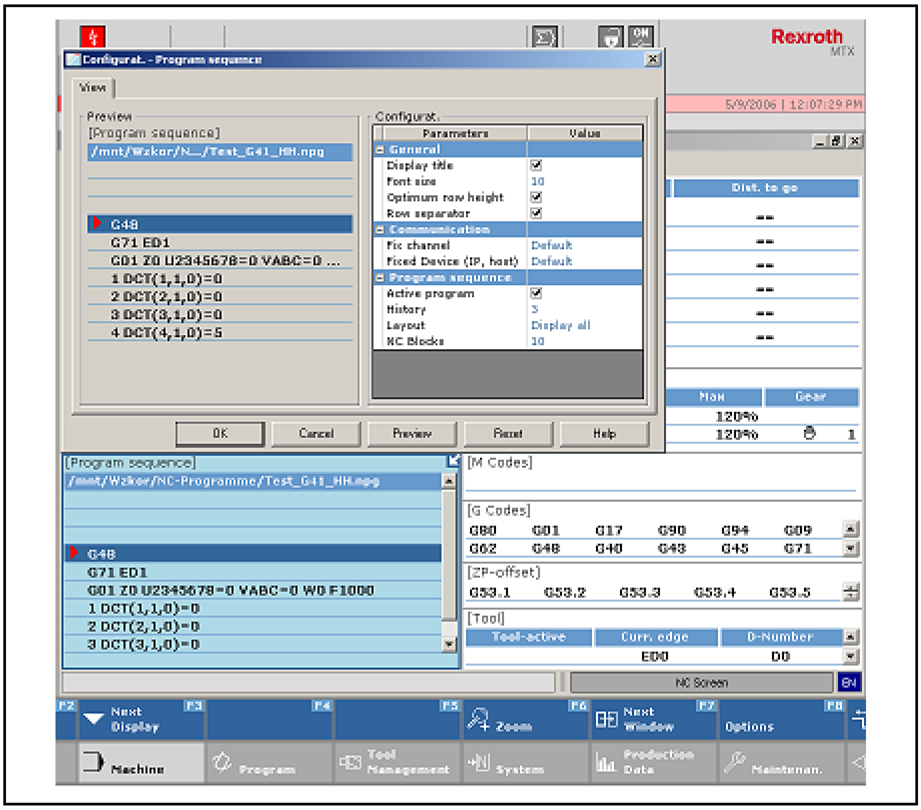


Fig.7-52: Displaying the NC program section that is currently being processed

You can make the following settings under "general":

- Select the font size for the displayed NC program by double-clicking the currently set value (the value 10 in the example in fig. 7-52 "Displaying the NC program section that is currently being processed" on page 172).
- Specify whether the name of the window (program section) is to be displayed.
- Specify the line height (optimum or exactly according to the font size)
- Specify whether you want to have separators between the lines
- You can use "communication - fixed channel" to specify the channel whose active NC program you want to have displayed. If you select "default", the currently active channel is always displayed.
- You can use "program section" to specify further specific characteristics of the NC program display, e.g.:
  - Under "active program", you can determine whether the name of the active NC program is to be displayed (checkmark) or not (no checkmark).
  - Under "layout", you can determine whether the active block is to be marked by highlighting ("bar"), by a arrow ("Icon") or by both ("all").
  - You can use "NC blocks" to specify how many lines of the NC program you want to have displayed. If the space is insufficient, a vertical scroll bar appears.

4. After the selection has been made, exit "Configuration - Program Section" by pressing "OK".

7.2.6 Subroutine Nesting Display

Description

1. Select the "Machine" operating area.
2. Display the NC program section that is currently being processed by clicking the mouse or by pressing (several times, if desired) <F7> (Next window) (fig. 7-52 "Displaying the NC program section that is currently being processed" on page 172).
3. Press <F3> (Next display).
4. Press <F8> (Options).

The "Configuration - Program Information" window opens (fig. 7-52 "Displaying the NC program section that is currently being processed" on page 172).
You are currently in subroutine UP5, which was called by subroutine UP4, which was in turn called by UP3, UP3 by UP2 and UP2 by UP1.

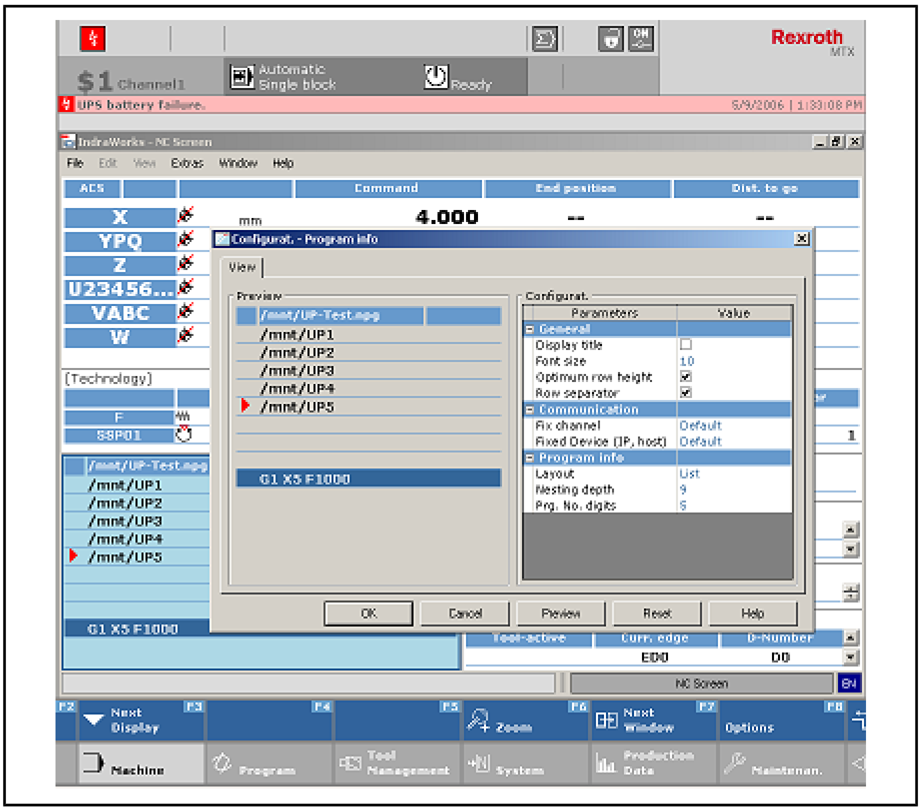


Fig.7-53: Subroutine nesting display

You can make the following settings under "general":

- Select the font size for the displayed NC program by double-clicking the currently set value (the value 10 in the example in fig. 7-53 "Subroutine nesting display" on page 173).
- Specify whether the name of the window (program info) is to be displayed.
- Specify the line height (optimum or exactly according to the font size)
- Specify whether you want to have separators between the lines
- You can use "communication - fixed channel" to specify the channel whose active NC program you want to have displayed. If you select "default", the currently active channel is always displayed.
- You can use "program information" to specify further specific characteristics of the subroutine nesting display, e.g.:

- Under "layout", you can specify whether the subroutines are to be indented or not (list).
- Under "nesting depth", you can specify how many nesting layers are to be displayed. A nesting depth of at least 5 would be needed for the display in fig. 7-53 "Subroutine nesting display" on page 173.

5. After the selection has been made, exit "Configuration - Program Section" by pressing "OK".

7.2.7 M-code Display

Description

1. Select the "Machine" operating area.
2. Select the M-code display by clicking the mouse or by pressing (several times, if desired) the <F7> key (Next window)(fig. 7-54 "Dialog: Configuration M-codes" on page 174).
3. Press <F8> (Options).

The "Configuration - M-codes" window opens:

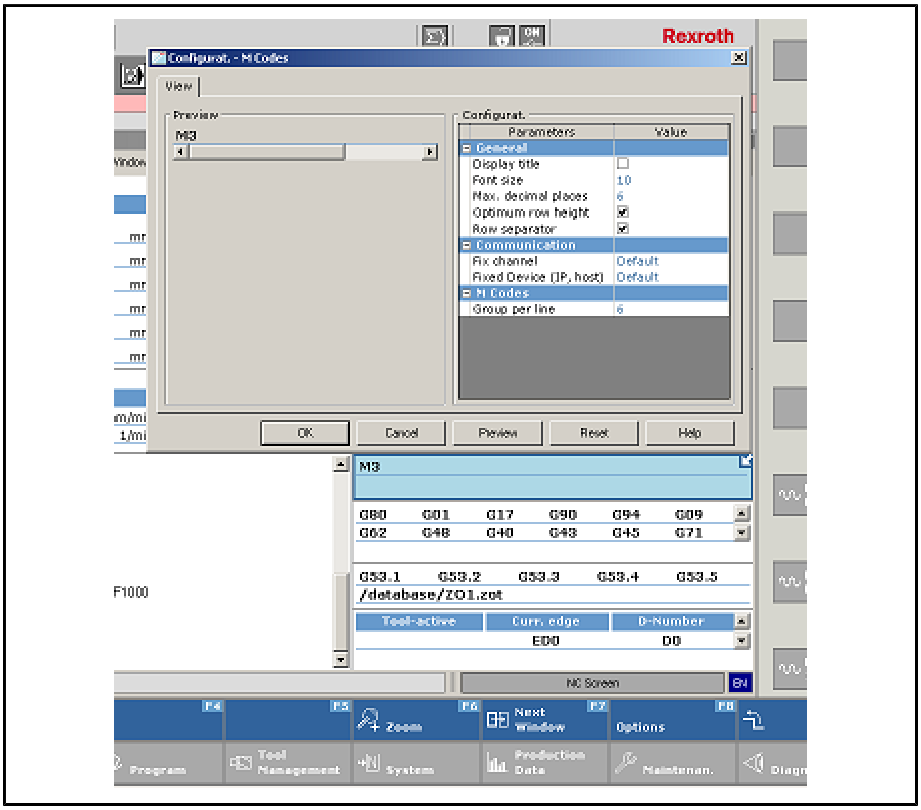


Fig.7-54: Dialog: Configuration M-codes

You can make the following settings under "general":

- Select the font size for the displayed M-codes by double-clicking the currently set value (the value 10 in the example in fig. 7-54 "Dialog: Configuration M-codes" on page 174).
- Specify whether the name of the window ("M-codes") is to be displayed.
- Specify the line height (optimum or exactly according to the font size).
- Specify whether you want to have separators between the lines

- You can use "communication - fixed channel" to specify the channel whose active M-codes you want to have displayed. If you select "default", the currently active channel is always displayed.
- You can use "M-codes" to specify how many M-code groups you want to have displayed per line.

4. After the selection has been made, exit "Configuration - M-codes" by pressing "OK".

You must specify which M-code groups are to be displayed in the configurator:

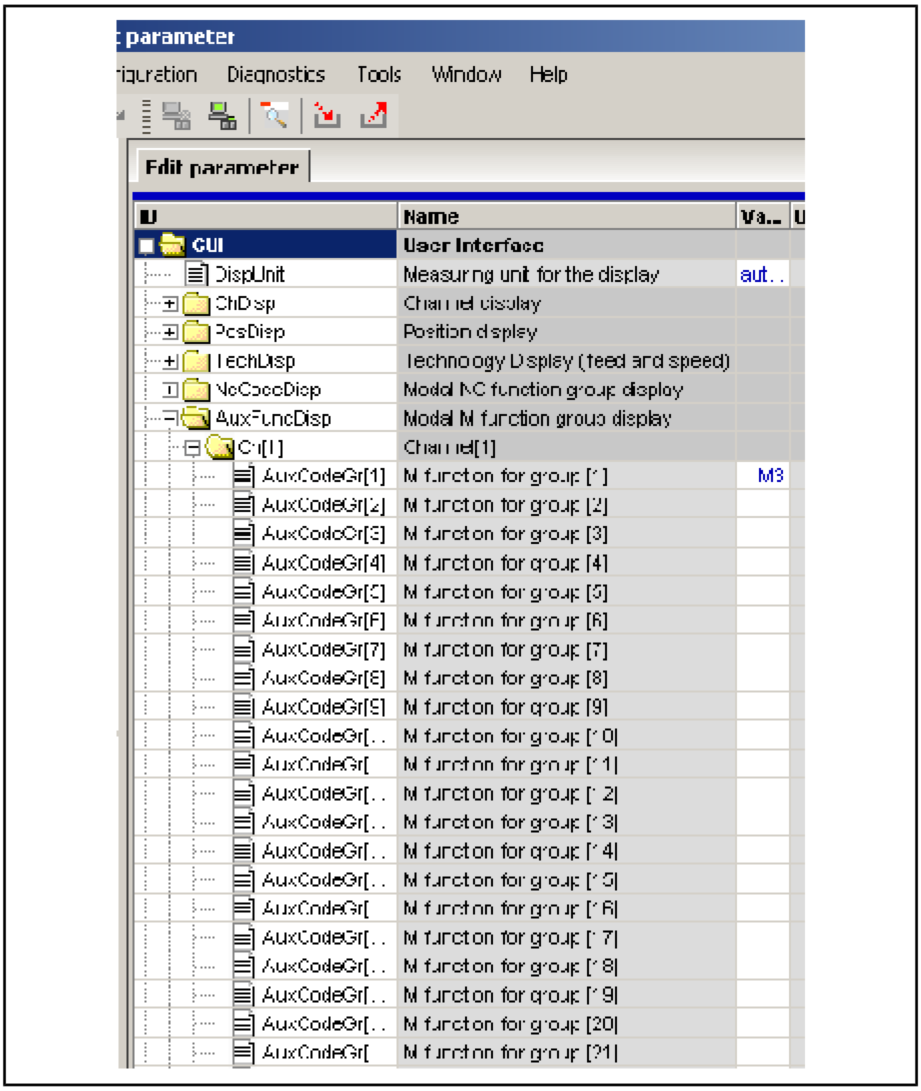


Fig.7-55: Configurator settings for the M-code display

7.2.8 G-code Display

Description

1. Select the "Machine" operating area.
2. Select the G-code display by clicking the mouse or by pressing (several times, if desired) <F7> (Next window)(fig. 7-56 "Dialog: Configuration G-codes" on page 176).
3. Press <F8> (Options).

   The "Configuration - G-codes" window opens:

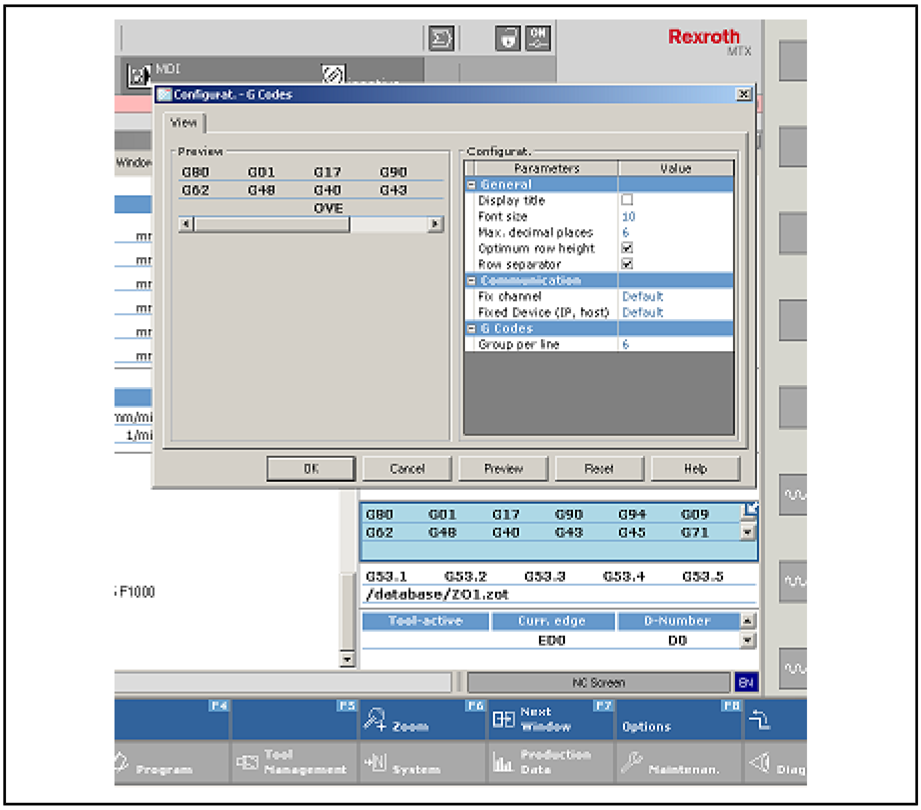


Fig.7-56: Dialog: Configuration G-codes

You can make the following settings under "general":

- Select the font size for the displayed G-codes by double-clicking the currently set value (the value 10 in the example in fig. 7-56 "Dialog: Configuration G-codes" on page 176).
- Specify whether the name of the window ("G-codes") is to be displayed.
- Specify the line height (optimum or exactly according to the font size).
- Specify whether you want to have separators between the lines
- You can use "communication - fixed channel" to specify the channel whose active G-codes you want to have displayed. If you select "default", the currently active channel is always displayed.
- You can use "G-codes" to specify the maximum number of G-code groups you want to have displayed per line.

4. After the selection has been made, exit "Configuration - G-codes" by pressing OK.

You must specify which G-code groups are to be displayed in the configurator (see fig. 7-57 "Configurator settings for the M-code display" on page 177). Here, you must select a proxy (an NC-function) for each desired G-code group, i.e. G17 as the proxy for the G-code group "Plane Selection", consisting of the NC functions G16, G17, G18, G19 and G20.

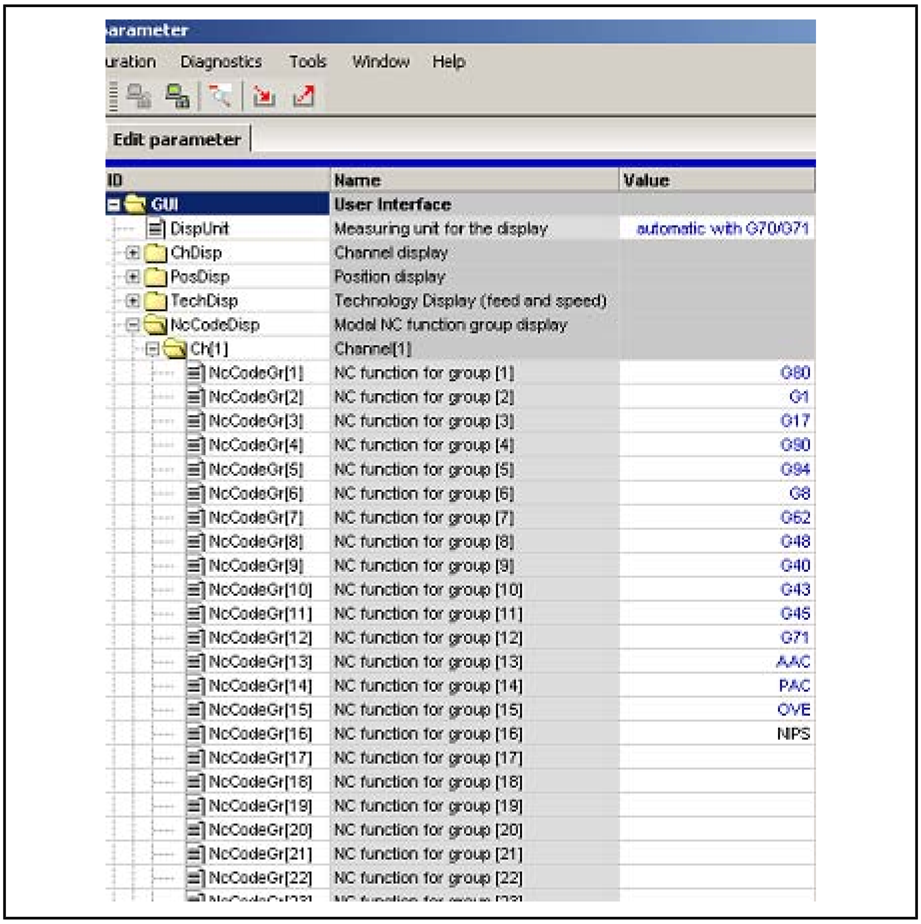


Fig.7-57: Configurator settings for the M-code display

7.2.9 Display of Zero Offsets

Description

1. Select the "Machine" operating area.
2. Select the ZO display by clicking with the mouse or by pressing <F7> (several times, if applicable) (Next window) (fig. 7-58 "Dialog: Configuration NPV" on page 178).
3. Press <F8> (Options).

   The "Configuration - ZO" window opens:

Configuring the User Interface

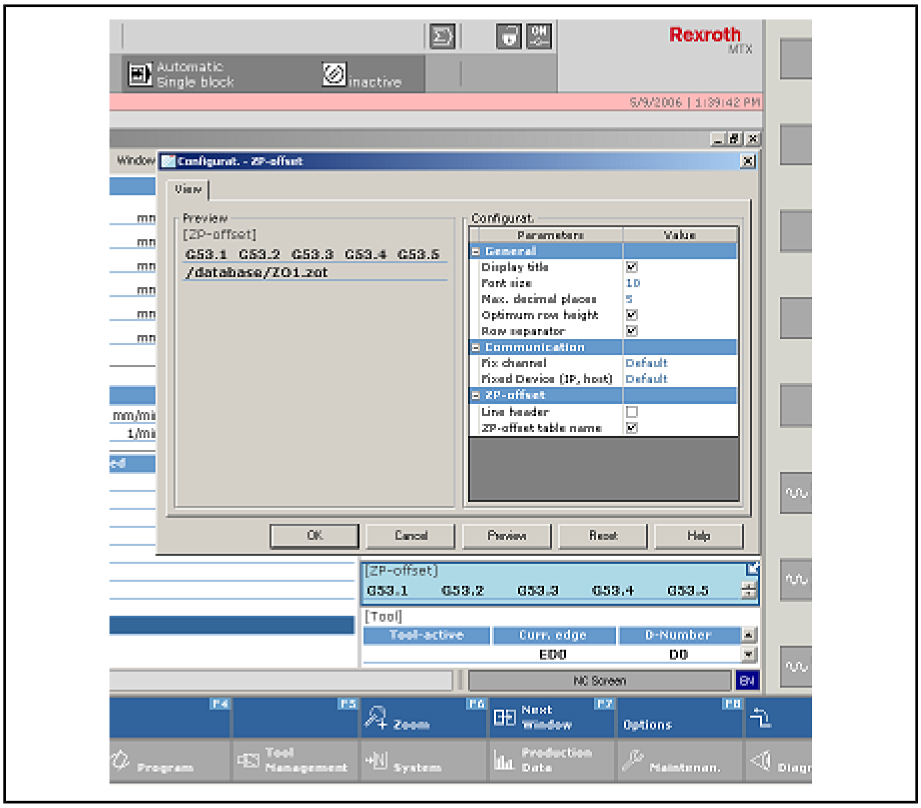


Fig.7-58: Dialog: Configuration NPV

You can make the following settings under "general":

- Select the font size for the displayed ZOs by double-clicking the currently set value (the value 10 in the example in fig. 7-58 "Dialog: Configuration NPV" on page 178).
- Specify whether the name of the window ("ZO") is to be displayed.
- Specify the line height (optimum or exactly according to the font size).
- Specify whether you want to have separators between the lines
- You can use "communication - fixed channel" to specify the channel whose active ZOs you want to have displayed. If you select "default", the currently active channel is always displayed.
- Under "ZOs", you can specify whether you want to display the ZO table name in addition to the active ZOs and/or whether a title ("ZO") is to be shown for every line of the ZO display window.

4. After the selection has been made, exit "Configuration - ZO" by pressing "OK".

You must specify which ZOs are to be displayed by default in the configurator (see fig. 7-59 "Configurator settings for the ZO display" on page 179). For example, the configuration in fig. 7-59 "Configurator settings for the ZO display" on page 179 means that if they are active, G54.1, G54.2, G54.3, G54.4 and G54.5 can be displayed.

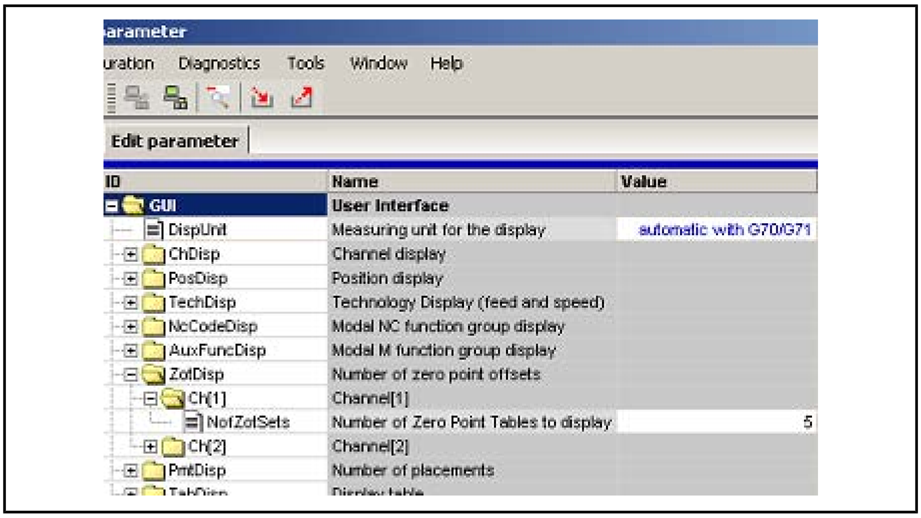


Fig. 7-59: Configurator settings for the ZO display

7.2.10 Placement Display

Description

1. Select the "Machine" operating area.
2. Select the placement display by clicking with the mouse or by pressing <F7> (several times, if applicable) (Next window) (fig. 7-60 " Dialog: Configuration - Placement" on page 180).
3. Press <F8> (Options).

   The "Configuration - Placements" window opens:

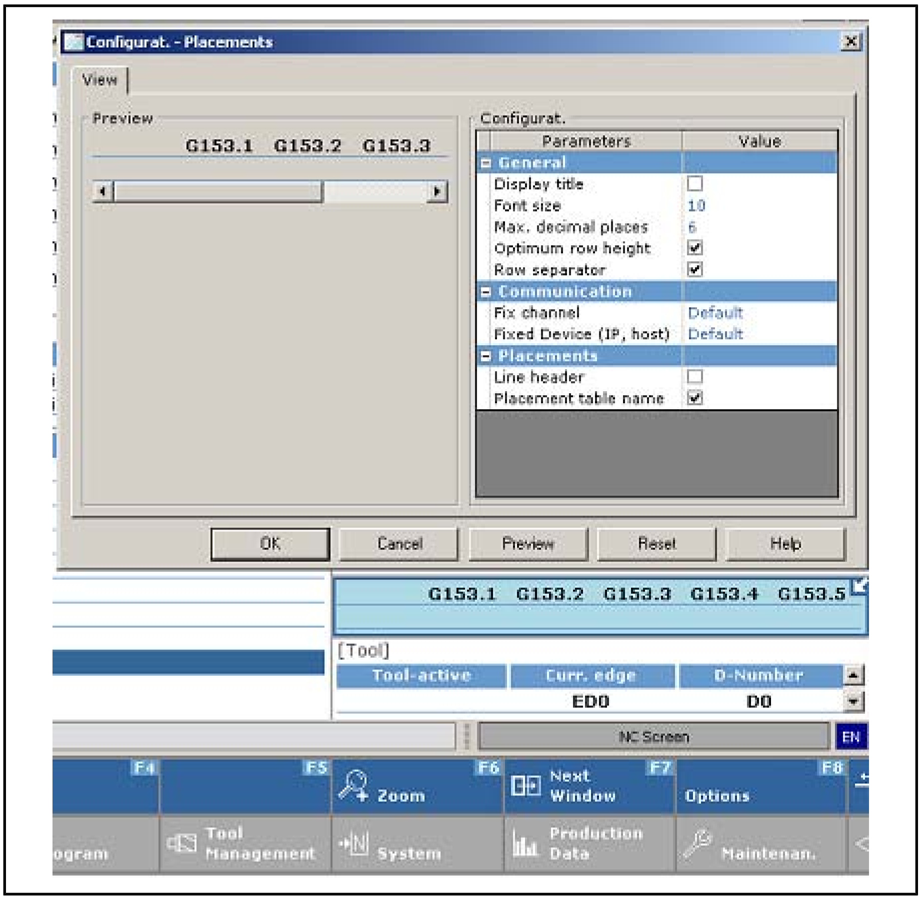


Fig.7-60: Dialog: Configuration - Placement

You can make the following settings under "general":

- Select the font size for the displayed placements by double-clicking the currently set value (the value 8 in the example in fig. 7-60 " Dialog: Configuration - Placement" on page 180).
- Specify whether the name of the window ("placements") is to be displayed.
- Specify the line height (optimum or exactly according to the font size).
- Specify whether you want to have separators between the lines
- You can use "communication - fixed channel" to specify the channel whose active placements you want to have displayed. If you select "default", the currently active channel is always displayed.
- Under "placements", you can specify whether you want to display the placement table name in addition to the active placements and/or whether a title ("placements") is to be shown for every line of the placement display window.

4. After the selection has been made, exit "Configuration - Placements" by pressing "OK".

You must specify which placements are to be displayed by default in the configurator (see fig. 7-61 " Configurator settings for the placements display" on page 181). For example, the configuration in fig. 7-61 " Configurator settings for the placements display" on page 181 means that if they are active, BCR, G154.1, G154.2, G154.3, G154.4 and G154.5 can be displayed.

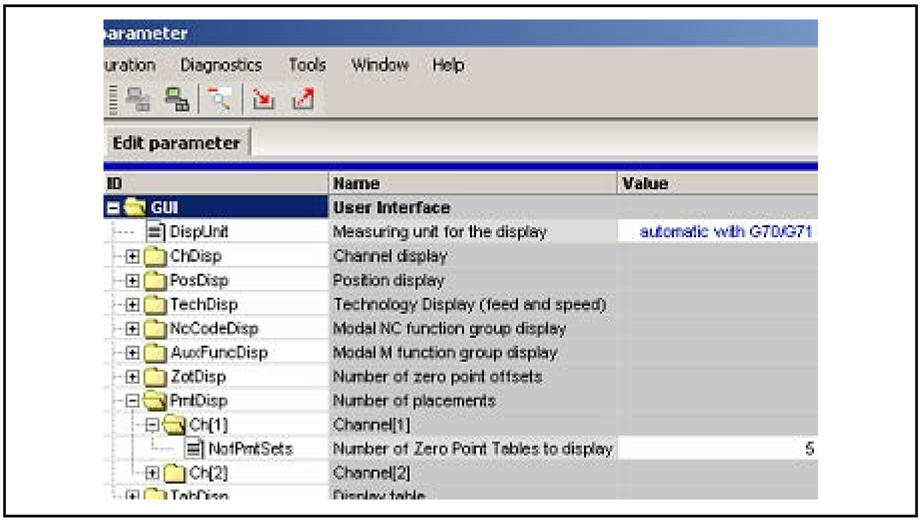


Fig.7-61: Configurator settings for the placements display

7.2.11 Display of the Active Tool

Description

1. Select the "Machine" operating area.
2. Select the display of the active tool by clicking with the mouse or by pressing <F7> (several times, if applicable) (see note!) (fig. 7-62 "Dialog: Configuration - Tool" on page 181).
3. Press <F8> (Options).

   The "Configuration - Tool" window opens:

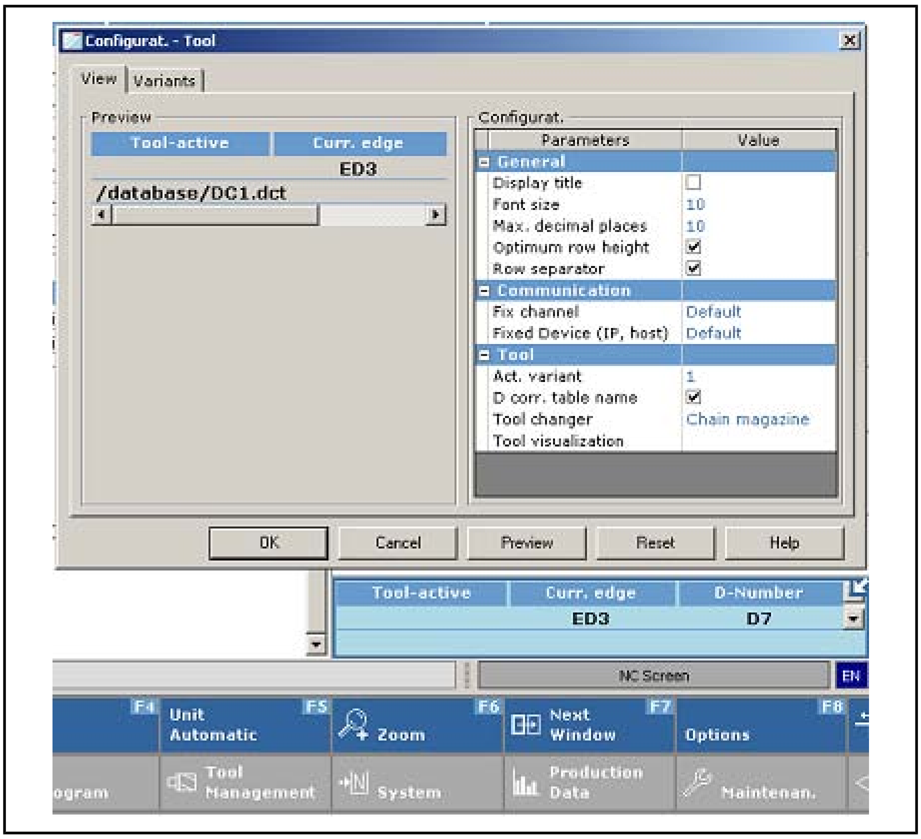


Fig.7-62: Dialog: Configuration - Tool

4. Click the tab "Variants".

   ⇒ The following window appears:

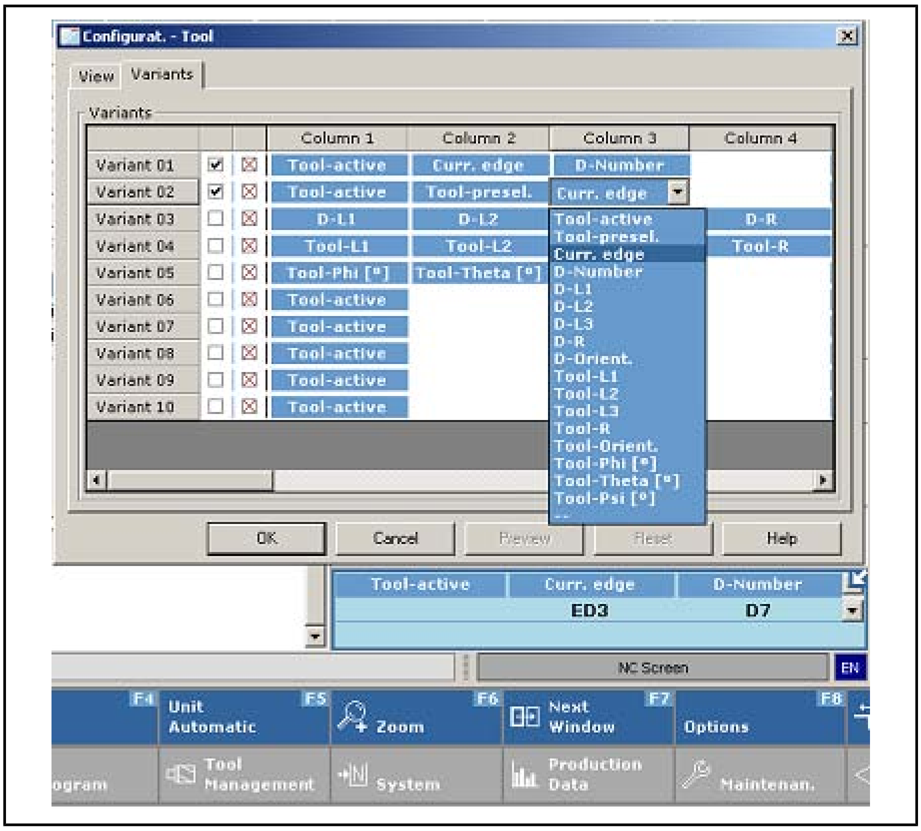


Fig.7-63: Dialog: Configuration - Tool (variants)

Columns 5-17 can be viewed using the horizontal scroll bar.
The tab "Variants" can be used to specify which tool data you want to see in the columns of the display of the active tool.

5. Activate the checkmarks for variants 01, 02 and 03.
6. For example, click on the entry in the 3rd column for variant 2. In the example in fig. 7-63 "Dialog: Configuration - Tool (variants)" on page 182, "act. TE" is shown in this column (Active tool edge).

   An arrow appears next to the words "Act. TE".

7. Click this arrow.

   A selection box appears in which you can choose between various data of the active tool by clicking the mouse.

8. Select "D number".
9. In the same way, replace the 1st entry (1st column) of variant 2 by "T-L3" and the 3rd entry by "D-L1".
10. Click the empty field behind the 3rd entry of variant 2.

    An arrow appears.

11. Click this arrow.

    A selection box appears in which you can choose between various data of the active tool by clicking with the mouse.

12. Select "Tool orient.".

    In the same manner, you can also make a selection for each of the columns 5-17 (in increasing order).
13. Also make a selection for variants 01 and 03.
14. Exit the "Configuration - Tool" by pressing OK.

If you press <F2>, scroll through the variants selected with a checkmark.

However, note that the display of the active tool must be selected, i.e. must be highlighted in blue (as shown in fig. 7-62 "Dialog: Configuration - Tool" on page 181), to do this.

1. Open "Configuration - Tool" again and go to the tab "View".

   While you used tab "Variants" to specify which tool data you want to display, you can specify the display method in this tab. These settings options are divided into the 3 suboptions "General", "Communication" and "Tool".

   - You can make the following settings under "general":
     – Select the font size for the displayed data of the active tool by double-clicking the currently set value (the value 10 in the example in fig. 7-62 "Dialog: Configuration - Tool" on page 181).
     – Specify whether the name of the window ("tool") is to be displayed.
     – Specify the line height (optimum or exactly according to the font size).
     – Specify whether you want to have separators between the lines
   - You can use "communication - fixed channel" to specify the channel whose axes/coordinates you want to have displayed. If you select "default", the currently active channel is always displayed.
   - Under "tool", you can specify which additional information you want to see for each tool, e.g. the name of the active D-correction table.
2. After the selection has been made under "View", exit "Configuration - Tool" by pressing "OK".

☞ The F-key becomes only "next window" if a windows has already been selected. As long as no window is selected, the labeling remains "active variables".

7.3 Process Displays

7.3.1 Overview

The process displays have machine- and interface-dependent parameters.

- Machine-dependent parameters are set in the NC configurator in IndraWorks Engineering. For details on the parameters refer to "Rexroth IndraMotion MTX Machine Parameters".
- Interface-dependent parameters are set in the "Properties" dialog of the relevant displays in IndraWorks Operation.

7.3.2 General Parameters

NC Configurator

NC configurator path	MACODA	Name
/GUI/DispUnit	6020 00030	Measurement unit for the display

Fig.7-64: General parameters for NC configurator

Properties

Name	Values	Description
Maximum decimal digits	{10,11,12}	Specifies the maximum number of the decimal digits that can be displayed in the columns. The parameter defines the width of the columns in the columns in which decimal digits are displayed.
Font size	{Default,7	Specifies the font size of the display. The value "optimal" determines the best font size possible.
Display title	{yes, no}	Specifies whether the title of the display is to be displayed.
Line height optimal	{yes, no}	Specifies whether the optimum line height is to be displayed.
Line separator	{yes, no}	Specifies whether a line separator is to be displayed.
Colored lines	{yes, no}	Specifies whether the lines of the display are to be colored. This improves readability.

Fig.7-65: General Parameters - Properties

7.3.3 Communication Parameters

NC Configurator

NC configurator path	MACODA	Name
-	-	-

Fig.7-66: Communication parameters for NC configurator

Properties

Name	Values	Description
Fixed channel	{Default, 1	Specifies whether the display is to be fixed on a channel. In this case, the display always shows the selected channel, regardless of the active channel of the NC main screen. The parameter enables configuration of NC screens displaying information across several channels. In the "default" setting, the display shows information of the active channel.
Fixed device (IP, host)	{default, [IP address] [host name]}	Specifies whether the display is to be set for a specified LDX device. In this case, the display always shows the selected device, regardless of the device active in the NC main screen. To identify a device, specify the IP address or the host name of the computer where the device is located. This computer must be registered as a DCOM server. For further information on this topic; see the documentation: "Rexroth IndraMotion MTX integration of OEM applications". The parameter enables configuration of NC screens displaying information across several devices. In the "default" setting, the display shows information of the active channel.

Fig.7-67: Communication parameters - Properties

7.3.4 Position Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/PosDisp/CoordSys/Ch[1-12]/Wcs	6005 00100	Coordinate system Wcs available in the channel
/GUI/PosDisp/CoordSys/Ch[1-12]/Mcs	6005 00100	Coordinate system Mcs available in the channel
/GUI/PosDisp/CoordSys/Ch[1-12]/Bcs	6005 00100	Coordinate system Bcs available in the channel
/GUI/PosDisp/CoordSys/Ch[1-12]/Acs	6005 00100	Coordinate system Acs available in the channel
/GUI/PosDisp/Ord/Ch/NofAxPosDisp	6005 00023	Maximum number of axes
/GUI/PosDisp/Ord/Ch[1-12]/Win/Dr[j]/ModeAxChWin	6005 00022	Display axis in the channel window
/GUI/PosDisp/Ord/Ch[1-12]/Win/Dr[j]/PrioAxChWin	6005 00022	Order of the Axis in the Channel Window
/GUI/PosDisp/Ord/Ch[1-12]/AddWin/Dr[j]/ModeAddAx	6005 00020	Display Axis in the Auxiliary Window
/GUI/PosDisp/Ord/Ch[1-12]/AddWin/Dr[j]/PrioAddAx	6005 00020	Order of the axis in the auxiliary window
/GUI/PosDisp/Ord/Ch[1-12]/AddWin/ChCoordPack[j]/ModeChPack	6005 00021	Display channel coordinate package in the auxiliary window

NC configurator path	MACODA	Name
/GUI/PosDisp/Ord/Ch[1-12]/AddWin/ChCoordPack[j]/PrioChPack	6005 00021	Order of the channel coordinate package in the auxiliary window
/GUI/PosDisp/Prec/Dr[i]/PrecPosMetr	6020 00011	Number of decimal places for axis display (metric)
/GUI/PosDisp/Prec/Dr[i]/PrecPosInch	6020 00012	Number of decimal places for axis display (inch)

Fig.7-68: Position display - NC Configurator

Properties

View .

Name	Values	Description
Axis name length	{4…16}	Specifies the maximum length of the axis names that can be displayed. The parameter determines the width of the column for axis names.
Active coordinate system	{ACS, MCS, BCS, WCS)	Specifies the active coordinate system. The coordinate system can also be switched via the key <F4 coordinate system> in the menu of this display or via the key <F6 coordinate system> in the submenu of the NC main screen. Only those coordinate systems are offered whose existence has been registered with the NC configurator.
Active variant	{1..10}	Specifies the active display variant. The variant can also be switched via the key <F2 variant> in the menu of this display or via the key <F5 variant position display> in the submenu of the NC main screen. Available variants are set in the variant editor in the "Properties" dialog.
Channel assignment	{yes, no}	Specifies whether an extra column will be displayed showing the channel assignment of the axes.
Measuring unit	{yes, no}	Specifies whether an extra column will be displayed showing the measurement unit of the axes.
System Axis names	{yes, no}	Specifies whether the system axis names are to be displayed in brackets behind the channel axis names. The parameter is effective only if the channel axis name differs from the system axis name.

Name	Values	Description
Radius in diameter programming	{yes, no}	Specifies whether the axis positions are to be displayed as the radius when programming the diameter. The parameter is in effect only in the coordinate system ACS and with axes which are programmed in diameter. "r" = Axis position is a radius value "Ø" = Axis position is a diameter value
Unit symbol	{yes, no}	Specifies whether the unit symbol [mm] or [inch] is to be displayed in the title line of the display. The parameter is in effect only if the display is not in the unit mode "automatic unit according to G70/G71". This mode is set in the NC configurator (measurement unit for the display)

Fig.7-69: Position display - Properties

Variants

The variant editor serves to create and edit display variants. Variants are visual instances with different column information that are established globally for all displays of this kind and that can be attributed and selected separately for each individual display. You can also attribute one variant to several instances. The variants are switched using the F-key <F2 variant> in the F-key menu in a focused display.

Display Filter

The axes to be displayed, set in the NC Configurator under "/GUI/PosDisp/…", apply globally to all position displays in IndraWorks Operation. Moreover, you can permanently show or hide individual axes for each position display by means of their system axis name addresses. Hiding thus overwrites the global settings in NC Configurator and the axis is never visible in this position display, although it is globally parameterized in this manner.

By means of this feature, it is possible to, for example, hide individual axes in the position display of the operation mode "Manual" as opposed to display in the operation mode "Automatic" of the same channel.

Activate the control box "Axis hidden" for those axes that should never be visible in the display.

Display of the actual position in WCS

The display of the actual position in the WCS is computing intense especially in case of active transformation on the NC-side (backward transformation required) and must thus be activated via the NC configurator if required.

If a value > 0 is entered for the parameter `TRA/GuiDisp/UpdateTimePosDisp`, this axis position is calculated in the kernel in this IPO cycle. The position display indicates the real actual position of the drives in the WCS instead of "--".

7.3.5 Technology Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/TechDisp/FeedDisp/PrecFeedMetr	6020 00001	Number of decimal places for feed display (metric)
/GUI/TechDisp/FeedDisp/PrecFeedInch	6020 00002	Number of decimal places for feed display (inch)
/GUI/TechDisp/Ch[1-12]/SpSpGrDisp[j]/SpDisp/SpIndSpDisp	6005 00030	Number of the spindle to be displayed

Fig.7-70: Technology display for NC configurator

Configuring the User Interface

Properties - View

View

Name	Values	Description
Active variant	{1..10}	Specifies the active display variant. The variant can also be switched via the key <F2 variant> in the menu of this display. Available variants are set in the variant editor in the "Properties" dialog.
Layout	{all, feed, spindle}	Specifies the layout of the display. In the display, only spindle, only feed or all technology data can be displayed as an option.
Spindle name length	{4…16}	Specifies the maximum length of the spindle names that can be displayed. The parameter determines the width of the column for axis names.
Unit symbol	{yes, no}	Specifies whether the unit symbol [mm] or [inch] is to be displayed in the title line of the display. The parameter is in effect only if the display is not in the unit mode "automatic unit according to G70/G71". This mode is set in the NC configurator (measurement unit for the display).

Fig.7-71: Technology display - Properties

Variants chapter "Properties" on page 186

Properties - Spindle Power/Torque

View Definition of the visual expression of the spindle power and torque display

Name	Values	Description
Display value	{yes, no}	Specifies whether the current value is displayed as a number.
Display maximum value	{yes, no}	Specifies whether the maximum value is displayed as a number behind the current value as well as a colored bar.
Reset maximum value		The maximum value can always be reset using F-key "Reset maximum value" in the F-keypad of the focused display or using the action of the same name. This can be allocated to an M-key, for example. You can also specify a PLC variable of type "BOOLEAN". If the edges change from "False" to "True" resetting is executed as well.
Color gradient	{various color gradient types}	Specifies the color gradient type. Various color gradient types are available. Specify the color and the change in color in relation to a certain percentage here.

Fig.7-72: Technology display - Properties

Variants chapter "Properties" on page 186

7.3.6 Display of Program Section

NC Configurator

NC configurator path	MACODA	Name
-	-	-

Fig.7-73: Display of program section - NC Configurator

Properties

View

Name	Values	Description
Active program	{yes, no}	Specifies whether the name of the active subroutine is to be displayed in the title line of the display.
History	{1…10}	Specifies the number of the NC blocks of the past (history). The parameter has to be lower than the parameter "NC blocks".
Layout	{all, bar, icon}	Specifies the layout of the display. As an option, the active NC block is highlighted in color, marked with an icon or by both visualization options.
NC blocks	{4…40}	Number of the total NC blocks to be displayed (including the history).

Fig.7-74: Program section display - Properties

7.3.7 Display of Program Nesting

NC Configurator

NC configurator path	MACODA	Name
-	-	-

Fig.7-75: Display of program nesting - NC Configurator

Properties

View

Name	Values	Description
Layout	{indented, list}	Specifies the layout of the display. The calling sequence of the NC programs can either be realized as a list or as a tree with indentations.
Program no. digits	{1…15}	Specifies the maximum number of decimal digits for the NC program numbers. The higher the parameter is set, the less space remains for displaying the subroutine paths.
Nesting depth	{0…24}	Specifies the maximum nesting depth of the calling sequence.

Fig.7-76: Display of program nesting - Properties

7.3.8 Offset Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/Zotdisp/Ch[1-12]/Nof-ZotSets	6005 00061	Number of pages to be displayed for zero offsets.

Fig.7-77: Offset display for NC configurator

Properties

View

Name	Values	Description
ZO table name	{yes, no}	Specifies whether the ZO table name is to be displayed.
Line title	{yes, no}	Specifies whether the line title is to be displayed.

Fig.7-78: Offset display - Properties

7.3.9 Offset Table Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/Zotdisp/Ch[1-12]/Nof-ZotSets	6005 00061	Number of pages to be displayed for zero offsets.
/GUI/TabDisp/PrecLinMetr	6020 00021	Number of decimal digits for linear axes (metric).
/GUI/TabDisp/PrecRotMetr	6020 00021	Number of decimal digits for rotary axes (metric).
/GUI/TabDisp/PrecLinInch	6020 00022	Number of decimal digits for linear axes (inch).
/GUI/TabDisp/PrecRotInch	6020 00022	Number of decimal digits for rotary axes (inch).

Fig.7-79: Offset table display for NC configurator

Properties

View

Name	Values	Description
Number of characters in the table	{8…32}	Specifies the maximum character length of the ZO table. This parameter is not active if parameter "extra information" is set to the value "no".
Display unit	{Default, [mm], [inch]	Measurement unit for the display
System Axis names	{yes, no}	Specifies whether the system axis names are to be displayed in brackets behind the channel axis names. The parameter is effective only if the channel axis name differs from the system axis name.
Additional information	{yes, no}	Specifies whether extra information is to be displayed. An additional column with extra information is displayed.

Fig.7-80: Offset table display - Properties

Variants chapter "Properties" on page 186

7.3.10 Placement Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/Pmtdisp/Ch[1-12]/ NofPmtSets	6005 00071	Number of pages to be displayed for placements.

Fig.7-81: Placement display for NC configurator

Properties

View

Name	Values	Description
Placement table name	{yes, no}	Specifies whether the placement table name is to be displayed.
Line title	{yes, no}	Specifies whether the line title is to be displayed.

Fig.7-82: Placement display - Properties

7.3.11 Placement Table Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/Zotdisp/Ch[1-12]/ NofPmtSets	6005 00071	Number of pages to be displayed for placements
/GUI/TabDisp/PrecLinMetr	6020 00021	Number of decimal digits for linear axes (metric)
/GUI/TabDisp/PrecRotMetr	6020 00021	Number of decimal digits for rotary axes (metric)
/GUI/TabDisp/PrecLinInch	6020 00022	Number of decimal digits for linear axes (inch)
/GUI/TabDisp/PrecRotInch	6020 00022	Number of decimal digits for rotary axes (inch)

Fig.7-83: Placement table display for NC configurator

Properties

View

Name	Values	Description
Number of characters in the table	{8…32}	Specifies the maximum character length of the placement table. This parameter is not active if parameter "extra information" is set to the value "no".
Display unit	{Default, [mm], [inch]}	Measurement unit for the display
Additional information	{yes, no}	Specifies whether extra information is to be displayed. An additional column with extra information is displayed.

Fig.7-84: Placement table display - Properties

Variants chapter "Properties" on page 186

7.3.12 Tool Display

NC Configurator

NC configurator path	MACODA	Name
-	-	-

Fig.7-85: Tool display for NC configurator

Properties

View

Name	Values	Description
Active variant	{1..10}	Specifies the active display variant. The variant can also be switched via the key <F2 variant> in the menu of this display. Available variants are set in the variant editor in the "Properties" dialog.
D-correction table name	{yes, no}	Specifies whether the NC configurator path of the D-correction table is displayed.

Name	Values	Description
Tool source	{CPL data, system data}	Determines the data source for the display of the activated and preselected tool. **System data (system data mode)** The system data structure "SD.SysTool[channel]" is to be described in the channel with the PLC or NC program with location and sector of the active and preselected tool. Examples: SD.SysTool[1].ActTool.K2 = 2 SD.SysTool[1].ActTool.K1 = 1 SD.SysTool[1].PreTool.K2 = 5 SD.SysTool[1].PreTool.K1 = 1 **CPL data (CPL mode) (not recommended)** **Displaying the tool number (TN):** For each channel, one CPL variable of type INT is to be created for the active and preselected tool (e.g. in the " wmhperm.dat" file). Example: `DEF INT @ACTTOOL01;` The variable is to be described program with the corresponding tool number (TN) from the PLC or NC. The "tool active" {ta} tool property is additionally to be set for the active tool. Important: If the CPL variable is written from an NC program, a WAIT instruction has to be programmed afterwards to update the display. **Displaying the tool name (SKQ):** For each channel, one CPL variable of type CHAR-ARRAY (string) is to be created for the active and preselected tool (e.g. in the " wmhperm.dat" file). Example: `DEF CHAR @ACTTOOL01(32);` The variable is to be described program with the corresponding tool name (SKQ) from the PLC or NC. The "tool active" {ta} tool property is additionally to be set for the active tool. Important: If the CPL variable is written from an NC program, a WAIT instruction has to be programmed afterwards to update the display.
Edit tool source	{[profiles file]}	Can only be edited if the "tool source" parameter is set to the value "CPL data". Opens the assignment file for defining the CPL variables for the active and the preselected tool. Changes in the file become effective only after the interface is restarted. If the CPL variable does not exist, the relevant information is not displayed.
Tool changer	{chain magazine, turret}	Specifies the column labeling of the tool visualization. Chain magazine: Tool active, tool preselected Turret: Tool active, place active
Display unit	{Default, [mm], [inch]	Measurement unit for the display

Fig.7-86: Tool display - Properties

Variants chapter "Properties" on page 186

7.3.13 Tool Correction Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/TabDisp/PrecLinMetr	6020 00021	Number of decimal digits for linear axes (metric)
/GUI/TabDisp/PrecRotMetr	6020 00021	Number of decimal digits for rotary axes (metric)
/GUI/TabDisp/PrecLinInch	6020 00022	Number of decimal digits for linear axes (inch)
/GUI/TabDisp/PrecRotInch	6020 00022	Number of decimal digits for rotary axes (inch)

Fig.7-87: Tool correction display for NC configurator

Properties

View

Name	Values	Description
Active variant	{1..10}	Specifies the active display variant. The variant can also be switched via the key <F2 variant> in the menu of this display. Available variants are set in the variant editor in the "Proper-ties" dialog.
Number of characters for the tool name	{8…32}	Specifies the maximum character length of the tool name. This parameter is not active if parameter "extra information" is set to the value "no".
Tool source	{CPL data, system data}	Refer to chapter "Properties" on page 192
Edit tool source	{[profiles file]}	Refer to chapter "Properties" on page 192
Additional information	{yes, no}	Specifies whether extra information is to be displayed. An additional column with extra information is displayed.
Tool changer	{chain magazine, turret}	Specifies the line labeling of the tool visualization. Chain magazine: Tool active, tool preselected Turret: Tool active, place active
Display unit	{Default, [mm], [inch]}	Measurement unit for the display

Fig.7-88: Tool correction display - Properties

Variants chapter "Properties" on page 186

7.3.14 Tool Correction Register Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/TabDisp/PrecLinMetr	6020 00021	Number of decimal digits for linear axes (metric)
/GUI/TabDisp/PrecRotMetr	6020 00021	Number of decimal digits for rotary axes (metric)
/GUI/TabDisp/PrecLinInch	6020 00022	Number of decimal digits for linear axes (inch)
/GUI/TabDisp/PrecRotInch	6020 00022	Number of decimal digits for rotary axes (inch)

Fig.7-89: Tool correction register display for NC configurator

Properties

View

Name	Values	Description
Active variant	{1..10}	Specifies the active display variant. The variant can also be switched via the key <F2 variant> in the menu of this display. Available variants are set in the variant editor in the "Properties" dialog.
Display unit	{Default, [mm], [inch]	Measurement unit for the display

Fig. 7-90: Tool correction register display - Properties

Variants chapter "Properties" on page 186

7.3.15 Input Tool Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/TabDisp/PrecLinMetr	6020 00021	Number of decimal digits for linear axes (metric)
/GUI/TabDisp/PrecRotMetr	6020 00021	Number of decimal digits for rotary axes (metric)
/GUI/TabDisp/PrecLinInch	6020 00022	Number of decimal digits for linear axes (inch)
/GUI/TabDisp/PrecRotInch	6020 00022	Number of decimal digits for rotary axes (inch)

Fig. 7-91: Input tool display for NC configurator

Properties

View

Name	Values	Description
System Axis names	{yes, no}	Specifies whether the system axis names are to be displayed in brackets behind the channel axis names. The parameter is effective only if the channel axis name differs from the system axis name.
Display unit	{Default, [mm], [inch]	Measurement unit for the display

Fig. 7-92: Input tool display - Properties

Variants chapter "Properties" on page 186

7.3.16 Program Coordinate Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/TabDisp/PrecLinMetr	6020 00021	Number of decimal digits for linear axes (metric)
/GUI/TabDisp/PrecRotMetr	6020 00021	Number of decimal digits for rotary axes (metric)
/GUI/TabDisp/PrecLinInch	6020 00022	Number of decimal digits for linear axes (inch)
/GUI/TabDisp/PrecRotInch	6020 00022	Number of decimal digits for rotary axes (inch)

Fig. 7-93: Program coordinates display for NC configurator

Properties

View

Name	Values	Description
System Axis names	{yes, no}	Specifies whether the system axis names are to be displayed in brackets behind the channel axis names. The parameter is effective only if the channel axis name differs from the system axis name.
Display unit	{Default, [mm], [inch]	Measurement unit for the display

Fig.7-94: Program coordinate display - Properties

Variants chapter "Properties" on page 186

7.3.17 G Code Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/NcCodeDisp/Ch[1-12]/NcCodeGr[j]	6005 00040	NC function for group

Fig.7-95: G-code display for NC configurator

Properties

View

Name	Values	Description
Groups per line	{2…15}	Specifies the maximum number of G-code groups per line. The parameter determines the number of lines.

Fig.7-96: G-code display - Properties

7.3.18 M Code Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/AuxFuncDisp/Ch[1-12]/AuxCodeGr[j]	6005 00040	NC function for group

Fig.7-97: M-code display for NC configurator

Properties

View

Name	Values	Description
Groups per line	{2…15}	Specifies the maximum number of M-code groups per line. The parameter determines the number of lines.

Fig.7-98: M-code display - Properties

7.3.19 Precision Correction Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/TabDisp/PrecLinMetr	6020 00021	Number of decimal digits for linear axes (metric)
/GUI/TabDisp/PrecRotMetr	6020 00021	Number of decimal digits for rotary axes (metric)
/GUI/TabDisp/PrecLinInch	6020 00022	Number of decimal digits for linear axes (inch)
/GUI/TabDisp/PrecRotInch	6020 00022	Number of decimal digits for rotary axes (inch).

Fig.7-99: Precision correction display - NC Configurator

Properties

View

Name	Values	Description
System Axis names	{yes, no}	Specifies whether the system axis names are to be displayed in brackets behind the channel axis names. The parameter is effective only if the channel axis name differs from the system axis name.
Display unit	{Default, [mm], [inch]	Measurement unit for the display

Fig.7-100: Precision correction display - Properties

7.3.20 Online Correction Display

NC Configurator

NC configurator path	MACODA	Name
/GUI/TabDisp/PrecLinMetr	6020 00021	Number of decimal digits for linear axes (metric)
/GUI/TabDisp/PrecRotMetr	6020 00021	Number of decimal digits for rotary axes (metric)
/GUI/TabDisp/PrecLinInch	6020 00022	Number of decimal digits for linear axes (inch)
/GUI/TabDisp/PrecRotInch	6020 00022	Number of decimal digits for rotary axes (inch)

Fig.7-101: Online correction display - NC Configurator

Properties

View

Name	Values	Description
System Axis names	{yes, no}	Specifies whether the system axis names are to be displayed in brackets behind the channel axis names. The parameter is effective only if the channel axis name differs from the system axis name.
Display unit	{Default, [mm], [inch]	Measurement unit for the display

Fig.7-102: Online correction display - Properties

7.3.21 System message display

NC Configurator

NC configurator path	MACODA	Name
-	-	-

Fig.7-103: System message display - NC configurator

Properties

View

Name	Values	Description
Diagnostic type "Setup diagnostics"	{yes, no}	Specifies whether appearing messages of the diagnostic type "Setup diagnostics" are to be displayed.
Diagnostic type "error"	{yes, no}	Specifies whether appearing messages of the diagnostic type "Errors" are to be displayed.
Diagnostic type "Notes"	{yes, no}	Specifies whether appearing messages of the diagnostic type "Notes" are to be displayed.
Diagnostic type "Start requirements"	{yes, no}	Specifies whether appearing messages of the diagnostic type "Start requirements" are to be displayed.
Diagnostic type "Warnings"	{yes, no}	Specifies whether appearing messages of the diagnostic type "Warnings" are to be displayed.
Icon	{yes, no}	Specifies whether the message is to be displayed with an icon. Refers to the diagnostic types "Errors", "Warnings" and "Notes".
Type of message (2)	{all, MTX, ProVi, step sequences}	Specifies the message source and/or the message type. **ProVi** Only ProVi messages are displayed. **MTX** Only MTX messages are displayed (NC kernel, general diagnostics). **Step sequences** Only step sequence messages are displayed. **All** messages are displayed.
Message exchange time	{500…5000}	Specifies the exchange time between the appearing messages. If more messages are to be shown than can be displayed, the messages are sorted according to their priority and then exchanged on a cyclic basis, as with a circular buffer. The parameter is not effective if the parameter "only latest message" is set to the value "yes".
Message lines	{1…20}	Specifies the maximum number of the messages displayed.

Name	Values	Description
Messages in color	{yes, no}	Specifies whether the messages are to be highlighted in color when displayed.
Only latest message	{yes, no}	Specifies whether only the latest message is to be displayed. If the value of the parameter is set to "yes", the parameter will have no effect on the "message exchange time".

Fig.7-104: System message display - Properties

7.4 Configuration of an External Application (Official HMI Documentation)

7.4.1 General

Handling Instruction: Configuration of an External Application

The following instructions show how to integrate and configure an external application.

IW Engineering: Establishing an HMI screen of the type "External application"

Refer to chapter 7.6 "Configuration of an HMI Screen (Official HMI Documentation)" on page 200

IW Engineering: Assign an external application

Refer to chapter 7.6 "Configuration of an HMI Screen (Official HMI Documentation)" on page 200

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

7.5 Configuration of the User NC Screen (Official HMI Documentation)

7.5.1 General

Handling Instruction: Creating a User NC Screen

The following instructions show how to create a user NC screen.

IW Engineering: Create an NC Screen

Refer to chapter 7.1 "Configuring ACI Screens" on page 117

IW Engineering: Create an HMI screen of the type "MtxUserNcScreen"

Refer to chapter 7.6 "Configuration of an HMI Screen (Official HMI Documentation)" on page 200

IW Engineering: Assign an NC Screen

Refer to chapter 7.6 "Configuration of an HMI Screen (Official HMI Documentation)" on page 200

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

7.6 Configuration of an HMI Screen (Official HMI Documentation)

7.6.1 General

Handling Instruction: Create an HMI Screen

The following instructions show how to create an HMI screen.

IW Engineering: Create an HMI screen

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

IW Engineering: Use the HMI screen editor

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

IW Engineering: Assign an HMI screen number (optionally)

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

IW Engineering: Select an HMI screen type

- **User-defined screen**

  Refer to chapter 7.6.2 "WinStudio General" on page 201
- **User NC screen**

  Refer to chapter 7.5 "Configuration of the User NC Screen (Official HMI Documentation)" on page 199
- **Operating screen**

  Refer to the official HMI documentation

  "DOK-IWORKS-HMI*Vxx**-APxx-EN-P"**.
- **Diagnostic screen**

  Refer to the official HMI documentation

  "DOK-IWORKS-HMI*Vxx**-APxx-EN-P"**.
- **DP diagnostic screen**

  Refer to the official HMI documentation

  "DOK-IWORKS-HMI*Vxx**-APxx-EN-P"**.
- **External application**

  Refer to the official HMI documentation

  "DOK-IWORKS-HMI*Vxx**-APxx-EN-P"**.
- **Logbook**

  Refer to the official HMI documentation

  "DOK-IWORKS-HMI*Vxx**-APxx-EN-P"**
- **Other axes**

  Refer to the official HMI documentation

  "DOK-IWORKS-HMI*Vxx**-APxx-EN-P"**.

IW Engineering: Create an F-Panel

Refer to chapter 7.6.3 "Configuration of F-Panels" on page 203

IW Engineering: Assign an F-panel

Refer to chapter "Handling Instruction: Use the F-Panel Editor" on page 203

IW Engineering: Create an M-panel on the left and M-panel on the right

Refer to chapter 7.6.5 "Configuration of M Panels" on page 204

IW Engineering: Assign an M-panel on the left and M-panel on the right

Refer to chapter "Handling Instruction: Use the M Panel Editor" on page 205

IW Engineering: Assign a screen to OP area

Refer to chapter 7.6.4 "Configuration of OP Panels and Operating Areas" on page 204

IW Engineering: Configure the screen attributes

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Define an HMI Start Screen

The following instruction shows how to define an HMI start screen.

IW Engineering: Define an HMI start screen

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Save an HMI Screen

The following instruction shows how to save an HMI screen.

IW Engineering: Save an HMI screen

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Rename an HMI Screen

The following instruction shows how to rename an HMI screen.

IW Engineering: Rename an HMI screen

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Delete an HMI Screen

The following instruction shows how to delete an HMI screen.

IW Engineering: Delete an HMI screen

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

7.6.2 WinStudio General

Handling Instruction: Start WinStudio

The following instruction shows how to start WinStudio

IW Engineering: Start WinStudio

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Define Screen Attributes

The following instruction shows how to define WinStudio screen attributes.

Define WinStudio screen attributes

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Edit a WinStudio Screen

The following instruction shows how to edit WinStudio screens.

IW Engineering: Edit a WinStudio screen

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Reaction to F-and M Keys in the WinStudio Screen

The following instruction shows the reaction to F-and M-keys in the WinStudio screen.

IW Engineering: Reaction to F-and M-keys in the WinStudio screen

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Multilingualism in the WinStudio Screen

The following instruction shows how to implement multilingualism in WinStudio screens.

IW Engineering: Multilingualism in the WinStudio screen

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Important Settings in WinStudio

The following instruction shows the important settings in WinStudio.

IW Engineering: Important settings in WinStudio

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Integration of WinStudio in User Management

The following instruction shows how to integrate WinStudio in the user management.

IW Engineering: Integration of WinStudio in user management

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

7.6.3 Configuration of F-Panels

Handling Instruction: F Panel - General Description

The following instruction gives the General Description of F-panels.

IW Engineering: F panel - General description

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Create an F-Panel

The following instruction shows how to create a new F-panel.

IW Engineering: Create an F-Panel

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Use the F-Panel Editor

The following instruction shows how to use the F-panel editor.

IW Engineering: Use the F-panel editor

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Rename an F-Panel

The following instruction shows how to rename an F-panel.

IW Engineering: Rename an F-panel

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Delete an F-Panel

The following instruction shows how to delete an F-panel.

IW Engineering: Rename an F-panel

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

7.6.4 Configuration of OP Panels and Operating Areas

Handling Instruction: OP Panel - General Description

The following instruction provides a general description of OP panels.

IW Engineering: OP panel - General description

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Use the OP Panel Editor

The following instruction shows how to use the OP panel editor.

IW Engineering: Use the OP panel editor

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Configure the Operating Area

The following instruction shows how to configure the operating areas.

IW Engineering: Configure the operating area

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

7.6.5 Configuration of M Panels

Handling Instruction: M Panel - General Description

The following instruction provides a general description of M-panels.

IW Engineering: M-panel - General description

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Create an M Panel

The following instruction shows how to create a new M-panel.

IW Engineering: Create an M-panel

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Use the M Panel Editor

The following instruction shows how to use the M-panel editor.

IW Engineering: Use the M-panel editor

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Rename an M Panel

The following instruction shows how to rename an M-panel.

IW Engineering: Rename an M-panel

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Delete an M Panel

The following instruction shows how to delete an M-panel.

IW Engineering: Rename an M-panel

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Integrate an M Panel into PLC

The following instruction shows how to integrate an M-panel into the PLC.

IW Engineering: Rename an M-panel

See HMI manual **"DOK-IWORKS-HMI*Vxx****-APxx-EN-P"**.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

7.7 Configuration of HMI Screens (MTX Documentation)

7.7.1 General

Description

Brief Description

In addition to the screens that exist by default in the interface, screens can be freely defined by the user. In this way, PLC variables can be displayed, for example.

The "WinStudio" tool is available to configure HMI screens. The various functions of this tool are described in detail in the "User`s Guide and Technical Reference Manual". There is also a quick start manual with the title "Rexroth WinStudio".

The following parts of this documentation describe the most important functions in detail using a handling instruction.

Handling Instruction: Create a User Screen

This handling instruction describes the procedure for generating a user screen using WinStudio.

☞ Depending on the application, certain steps may not be required or additional steps may be necessary.

IW Engineering: Create new IndraWorks project (if necessary)

Before WinStudio can be called, an HMI project must be created.

Handling Instruction: Start WinStudio

The following handling instruction describes how to start the "WinStudio" design software.

IW Engineering: Start WinStudio

1. If necessary, install WinStudio.
2. Double-click the WinStudio node within the Project Explorer

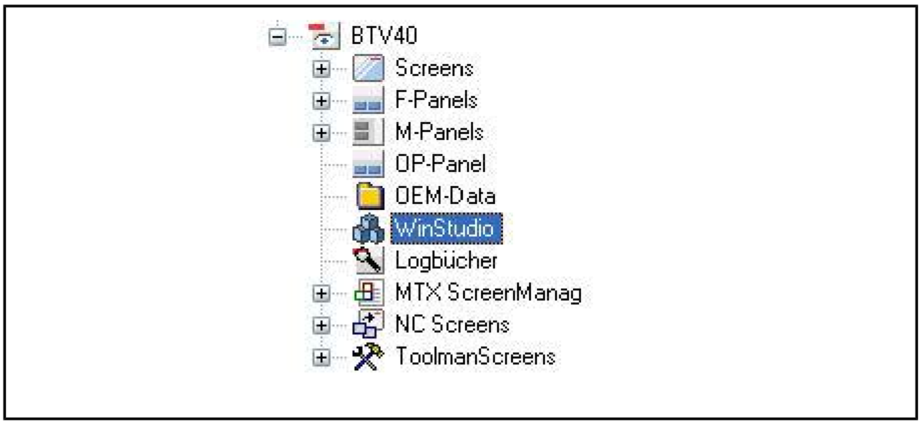


Fig.7-105: Starting WinStudio by double-clicking

– or –

Click the right mouse button on the WinStudio node within the project tree.

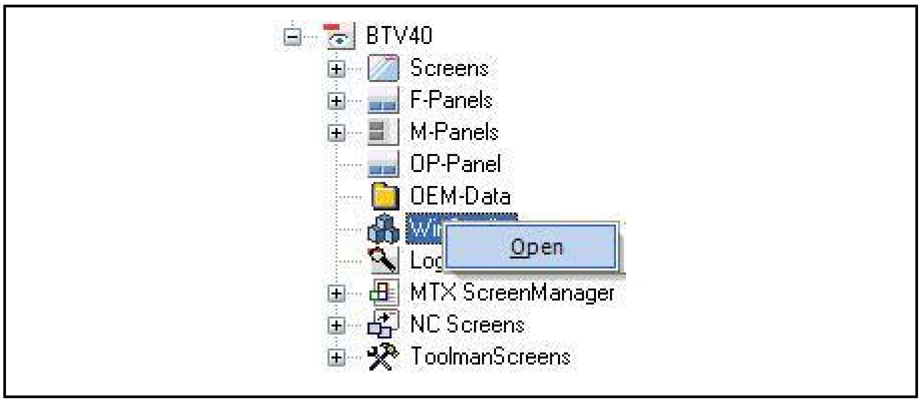


Fig.7-106: Starting WinStudio using the right mouse button

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Create a WinStudio Screen

The following handling instruction describes how to create a new user screen.

IW Engineering / WinStudio: Create a new screen

1. New user screen via the item "Insert", "Screen" of the toolbar

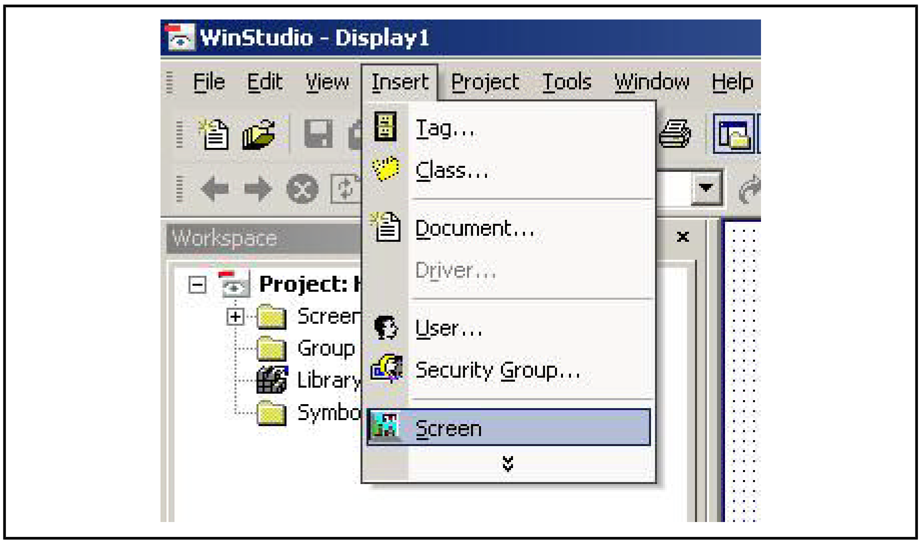


Fig.7-107: Menu: Insert / Screen (Creating a new screen)

– or –

by using the <right mouse button> to select "Insert" from the "Screens" node in the "Graphics" workspace.

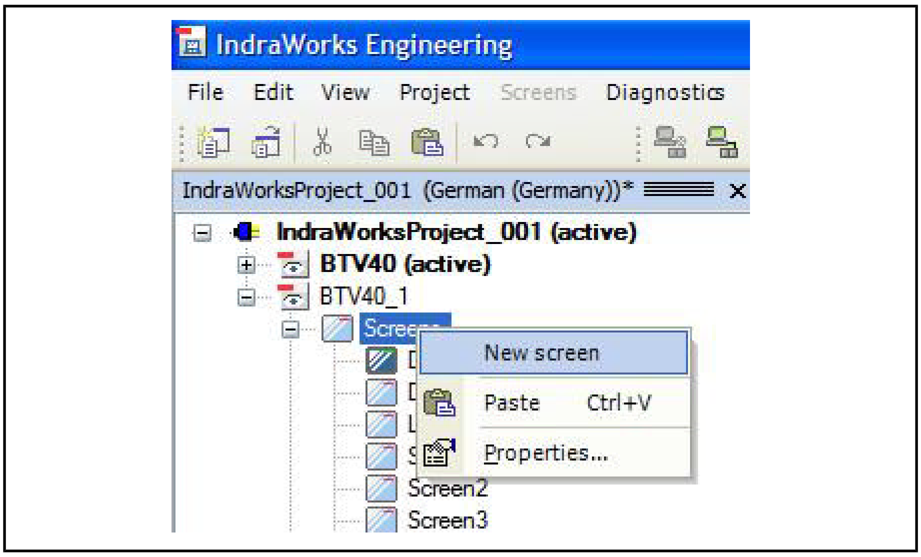


Fig.7-108: Context menu: Screens (Creating a new screen)

When a new screen is created, a dialog is generated, in which the screen properties can be defined.

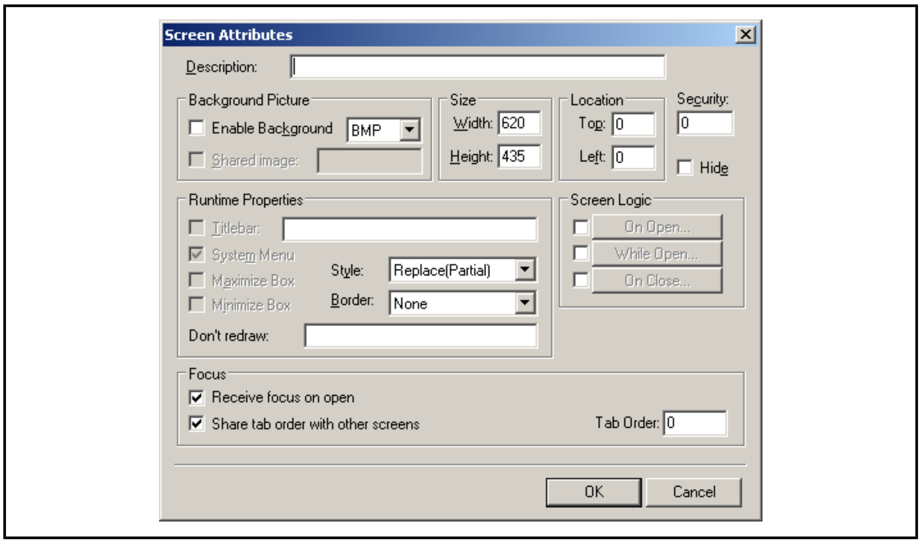


Fig.7-109: Dialog: Screen properties

- The "description" field is used exclusively as a comment field, for example, to define the purpose of the screen.
- "Enable background" allows a background image to be assigned to the screen at a later point in time.
- The "size" attribute defines the resolution of the screen. In general, the following resolutions are recommended:
  - BTV40: 765 x 537
  - BTV16: 620 x 412
- The screen attributes can also be adapted at any time in the future. To do this, open the screen and click it using the right mouse button. Then select "screen attributes".

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

IW Engineering / WinStudio: Adapt the attributes of the screen

The screen attributes can be adapted at any time in the future.

1. To do this, open the screen and click with the right mouse button.
2. Then select "Screen Attributes".

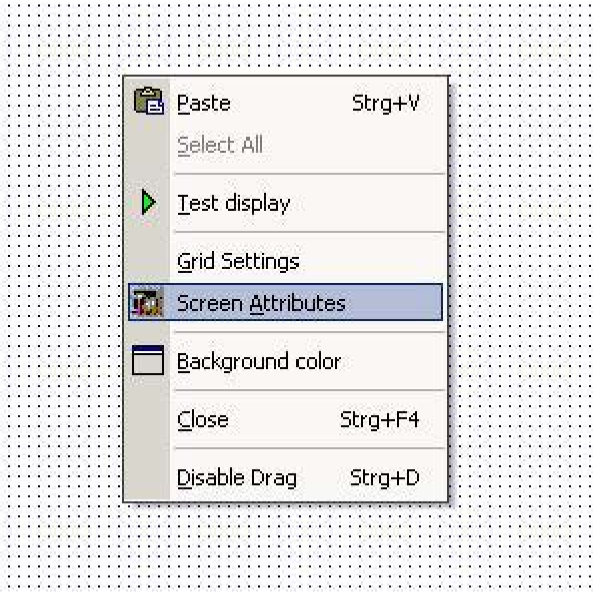


Fig.7-110: Context menu: Screen Attributes (Modifying the properties of a screen)

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

IW Engineering / WinStudio: Save screen

After a new screen has been created, it must be saved.

1. To do this, go to the "File" menu and select "Save as...".

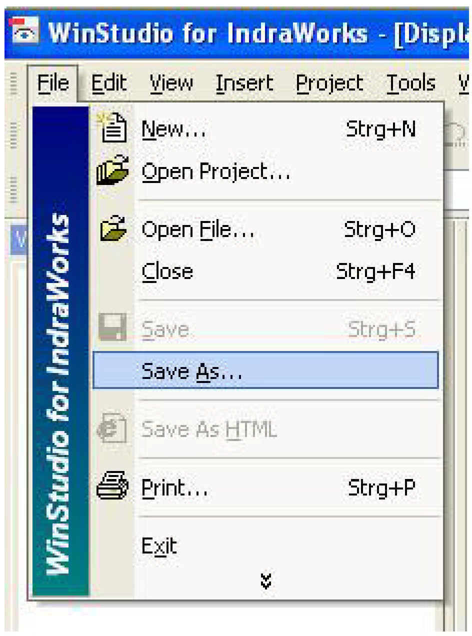


Fig.7-111: Menu. File (Save as - Save screen)

☞ The storage location and the name may no longer be changed after saving!

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Create a New Tag

This handling instruction describes how to create a new tag. A tag is a variable that is used for communication, computation or display.

IW Engineering / WinStudio: Create a new tag

There are several ways to create a tag.

1. E.g. via the toolbar "Insert", "Tag",

   – or –

   by means of the right mouse button on "Application Tags" under the "Database" workspace.

   Both methods lead to the following dialog.

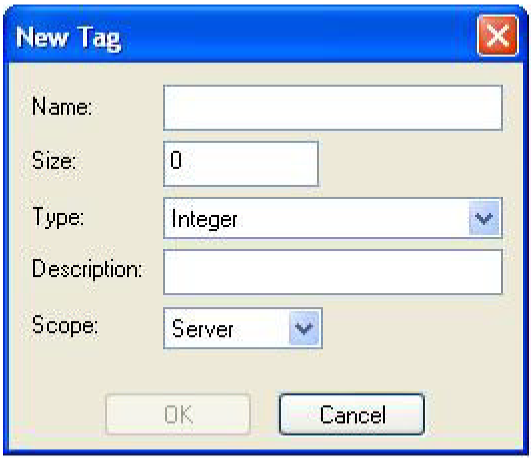


Fig.7-112: Dialog: New Tag

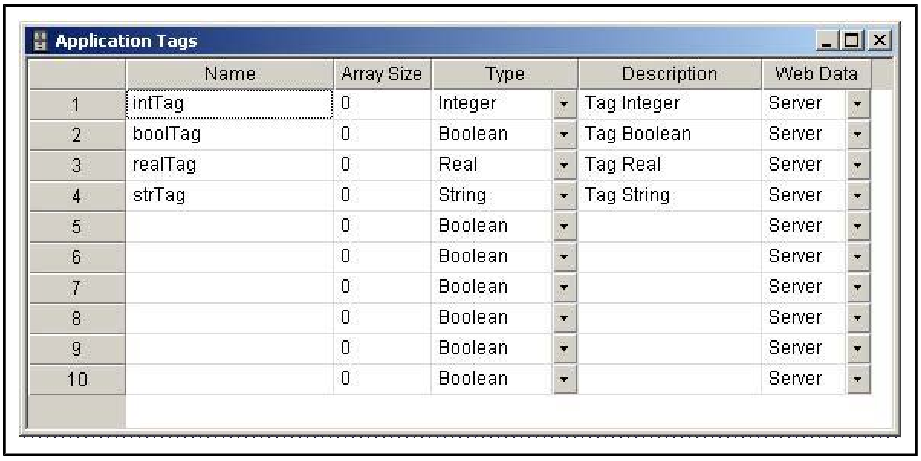
2. In the dialog "New Tag", define the tag name, the variable and the type.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

IW Engineering / WinStudio: Modify a tag

The declared tags can be edited at any time.

A list of the tags can be opened at any time using the "Datasheet View" node in the "Database" workspace.



	Name	Array Size	Type	Description	Web Data
1	intTag	0	Integer	Tag Integer	Server
2	boolTag	0	Boolean	Tag Boolean	Server
3	realTag	0	Real	Tag Real	Server
4	strTag	0	String	Tag String	Server
5		0	Boolean		Server
6		0	Boolean		Server
7		0	Boolean		Server
8		0	Boolean		Server
9		0	Boolean		Server
10		0	Boolean		Server

Fig.7-113: Dialog: Application Tags (tag list)

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

IW Engineering / WinStudio: Delete a tag

A tag can be deleted from the tag list by simply deleting the name.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Establishing Communication between WinStudio Variables and the Control

This handling instruction describes the generation of the communication connection between a WinStudio variable and the IndraMotion MTX.

IW Engineering / WinStudio: Creating a new OPC sheet

In order to make it possible to read and write PLC or NC items in WinStudio, the corresponding item must first be assigned to the tag.

For this purpose, a communication sheet has to be created via the tab "Comm" in the "Workspace".

1. By clicking with the right mouse button on "OPC" and
2. by then selecting "Insert".

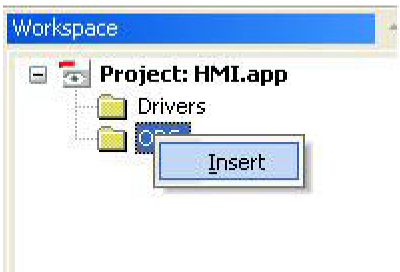


Fig.7-114: Context menu: OPC (Insert - Creating a new OPC sheet)

The following dialog appears:

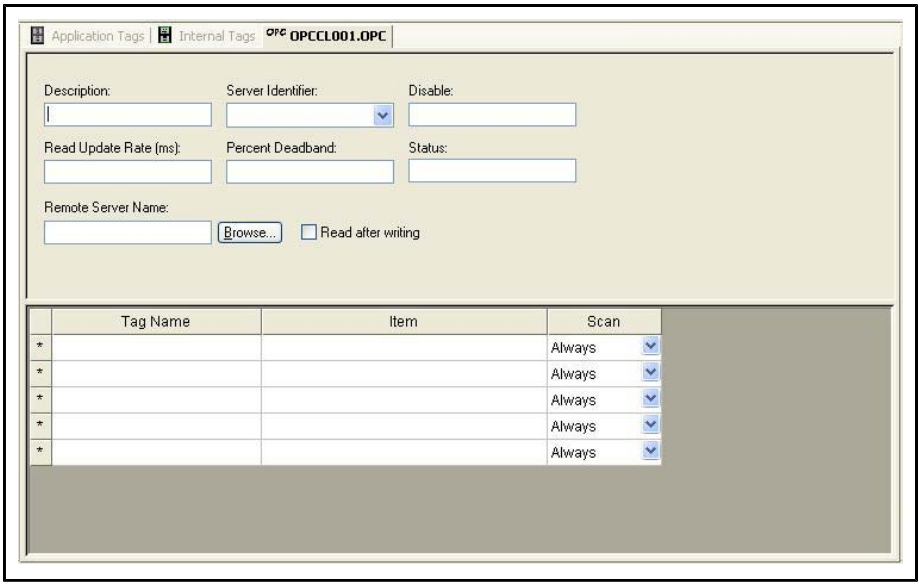


Fig.7-115: Dialog: Creating a new OPC sheet

Important input fields are "Description", "Server Identifier" and "Read Update Rate".

☞ For the "Server Identifier", "OPC.IwSCP.1" or higher is used. This makes it possible to access PLC and NC items.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

IW Engineering / WinStudio: Define the process connection

The connection between the tag and the item is generated in the OPC sheet.

1. Use the <right mouse button> and select "Insert Tag" within "Tag Name" to select a tag from the list.

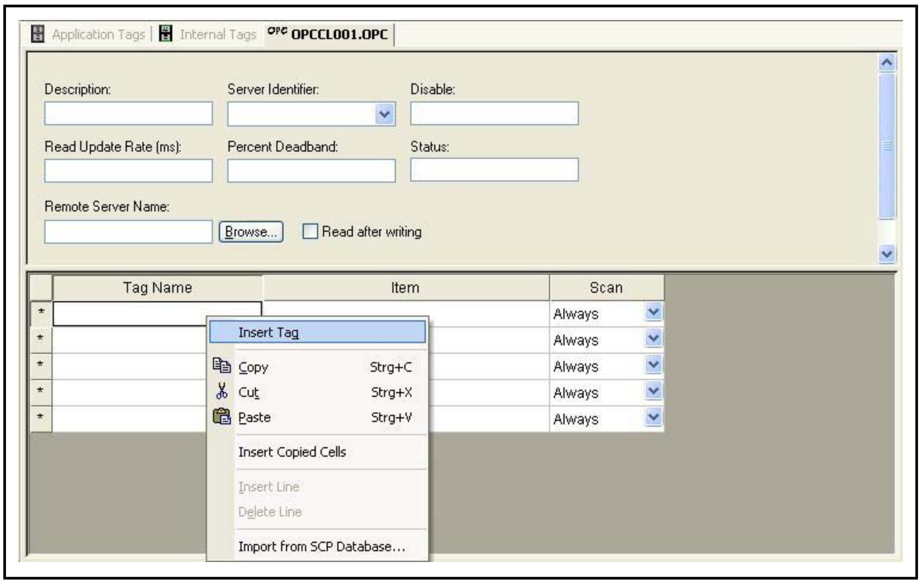


Fig.7-116: Context menu: Inserting a tag variable

2. Use the right mouse button and select "OPC Browser" within "Item" to select an item from the list.

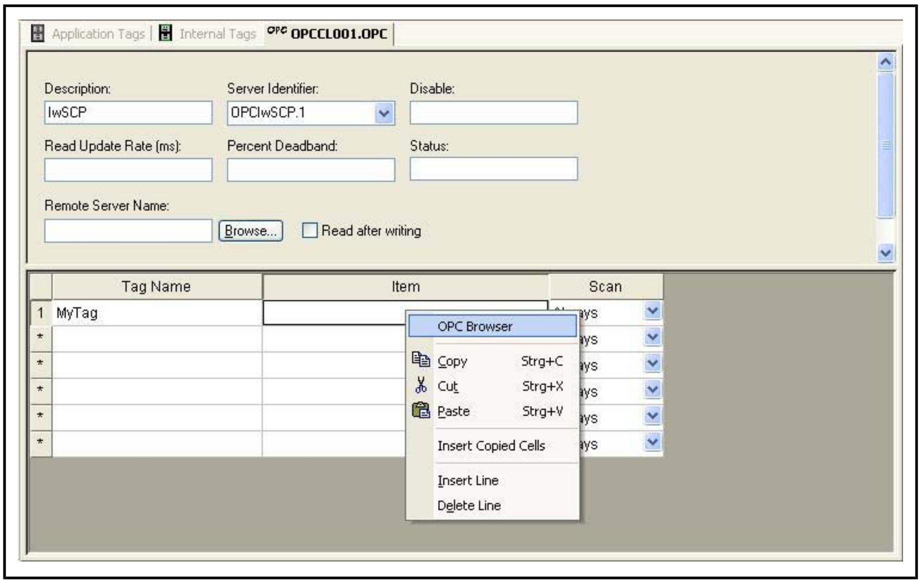


Fig.7-117: Context menu: Assigning an item

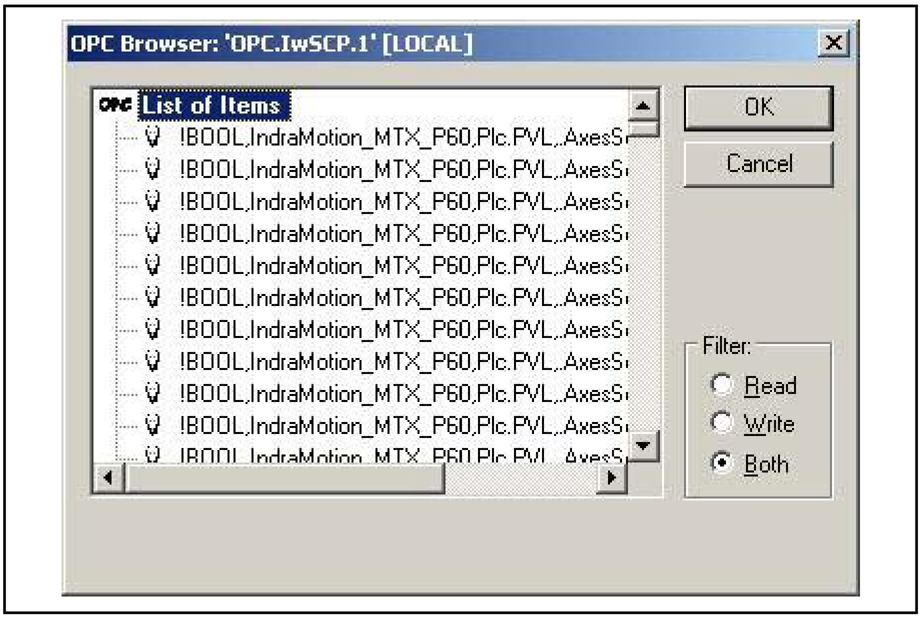


Fig.7-118: Dialog: OPC Browser (Select an item)

3. To specify "Scan",

   "Screen" = communication only if the screen is active

   "Always" = constant communication

   can be selected.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

Handling Instruction: Screen Logic

This handling instruction describes how logical operations can be executed for a user screen. To do this, so-called screen mathematics sheets must be created and filled in according to the desired logic.

IW Engineering / WinStudio: Activate the screen mathematics sheets

In order to be able to use the mathematics sheets of a screen, they must be activated.

To do this, the checkmarks before the buttons "On Opening", "While Open" und "On Closing" under "Screen Mathematics" in the dialog window "Screen Attributes" have to be set accordingly:

- "On opening" is carried out once every time that the screen is selected.
- "On closing" is carried out once every time that the screen is deselected.
- For "while open", the specified logic is carried out in a cycle or depending on events while the screen is selected.

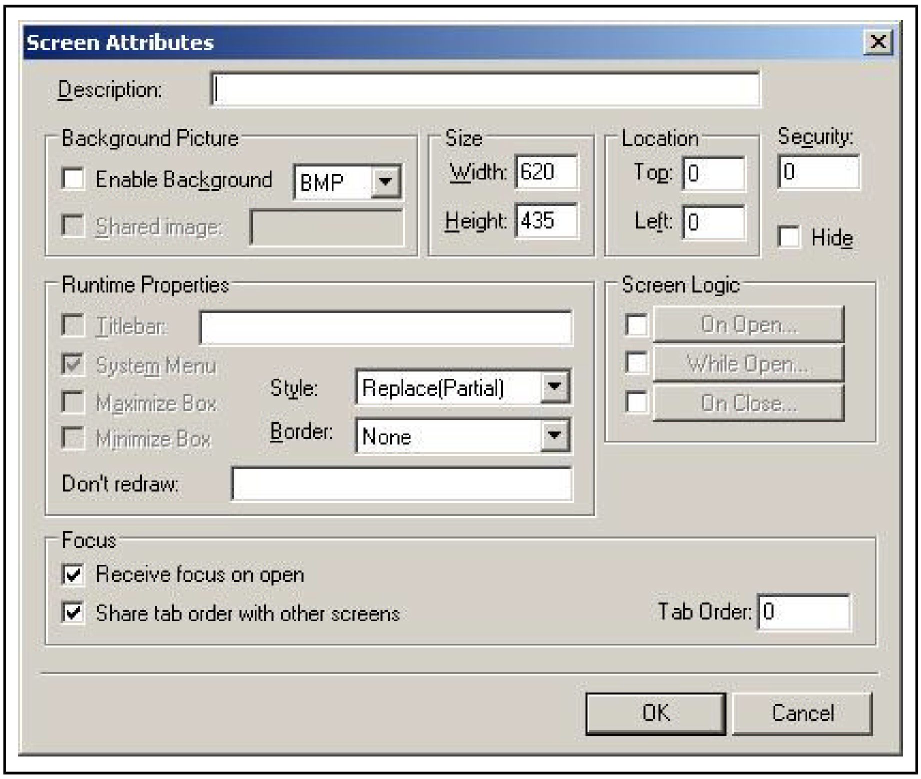


Fig.7-119: Dialog: Screen properties

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

IW Engineering / WinStudio: Execute logic when selecting a screen (On Opening) (if necessary)

Tick the button "On opening" with the mouse.

The following screen is shown:

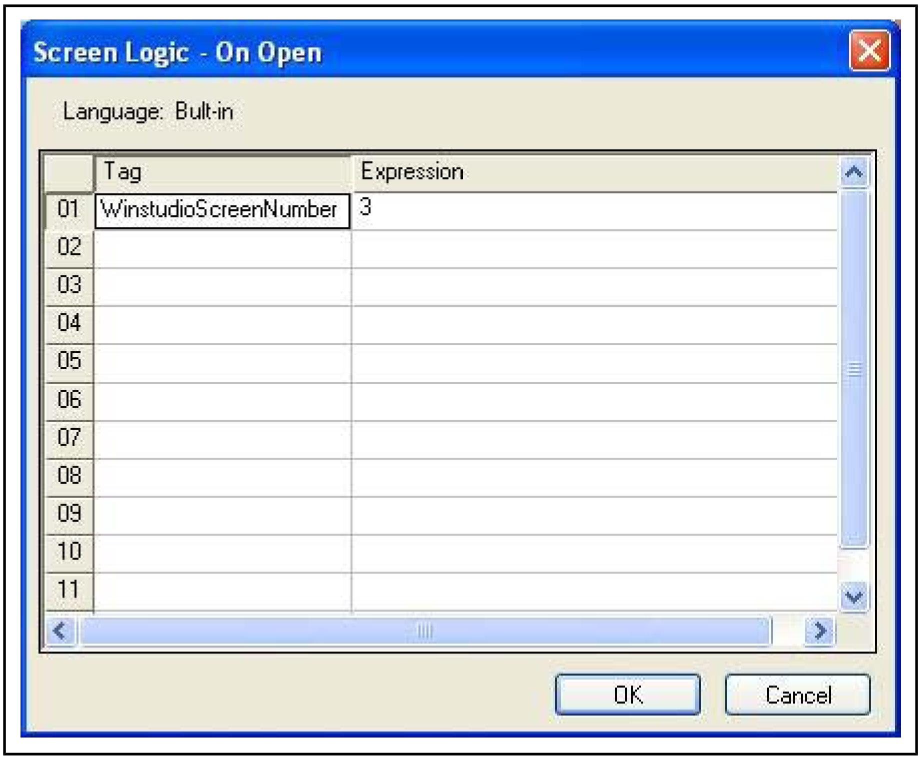


Fig.7-120: Dialog: Screen Logic (logic operations to be executed when calling a screen)

In the example above ("On Opening"), a tag that, for example, can be linked to a PLC variable is described once.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

IW Engineering / WinStudio: Execute logic while a screen is displayed (While Open) (if necessary)

Tick the button "While open" with the mouse.

The following screen is shown:

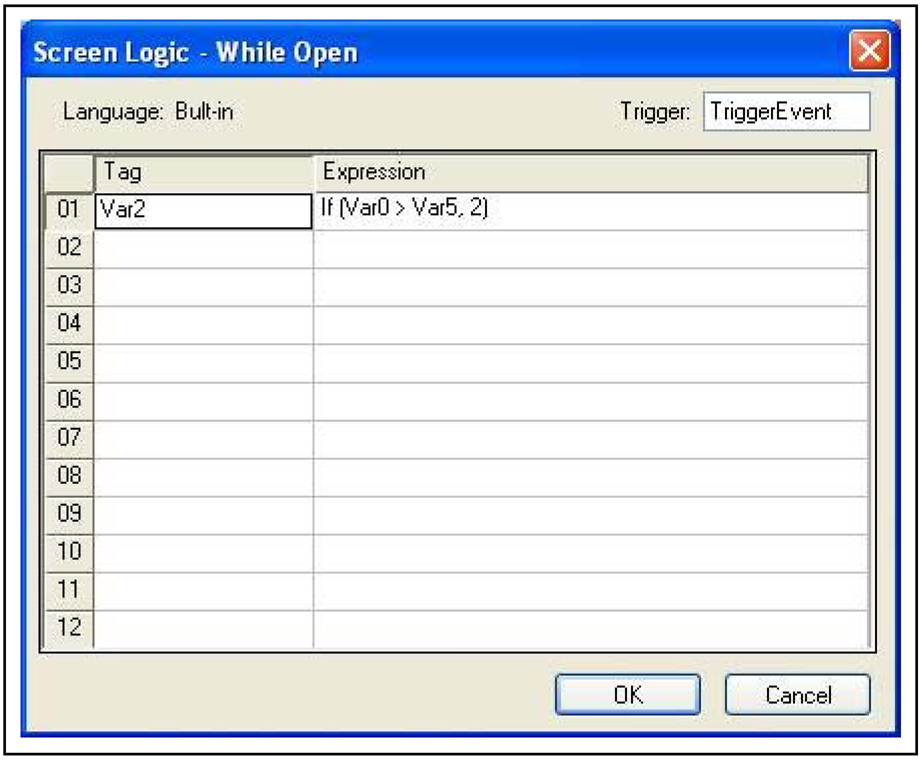


Fig.7-121: Dialog: Screen Logic (for entering logic operations that are to be executed cyclically while a screen is displayed)

If no entry is made in the field "Trigger", the mathematics sheet is called on a cycle basis. Otherwise, the mathematics sheet is executed once each time that the entered tag is modified.

		Documentation
Documentation:	Rexroth WinStudio	Working with WinStudio

IW Engineering / WinStudio: Execute logic when deselecting a screen (On Closing) (if necessary)

Tick the button "On closing" with the mouse.

The following screen is shown:

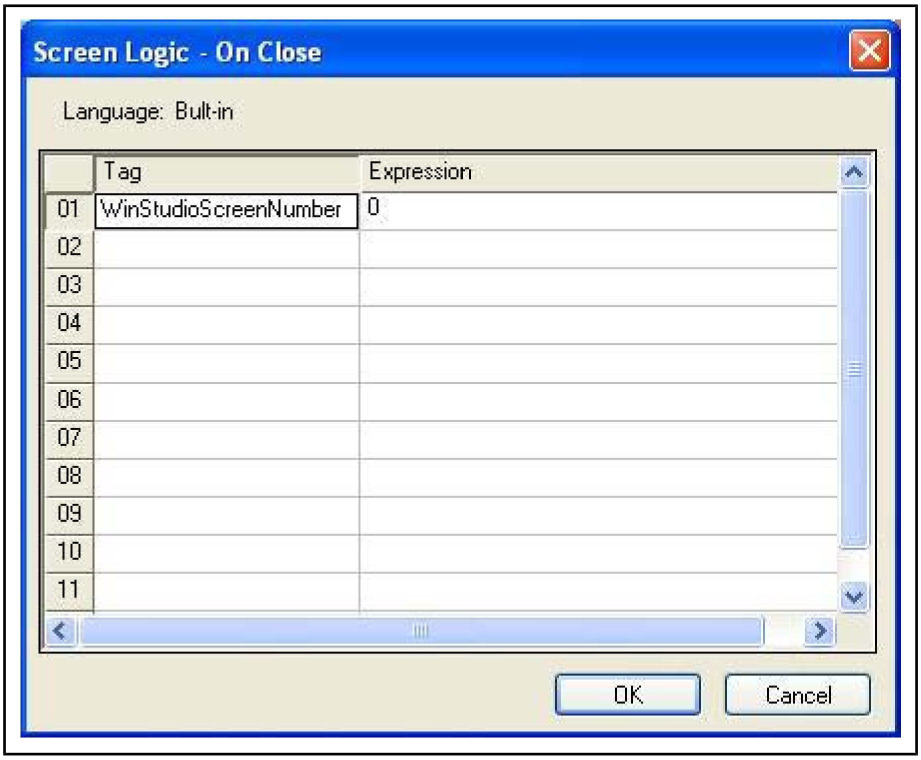


Fig.7-122: Dialog: Screen logic (for entering logic operations that are to be executed when a screen is closed)

In the example above, a tag that, for example, can be linked to a PLC variable is described once.

Handling Instruction: Generate a User Screen Using IndraWorks

This handling instruction describes the generation of a user screen in IndraWorks that was previously created in WinStudio.

IW Engineering: Incorporate a WinStudio screen

In order to be able to display the created WinStudio screen in the MTX interface, it must be incorporated into IndraWorks.

1. To do this, click the <right mouse button> on "Screens" -> <New Screen> in the Project Explorer under the HMI project.

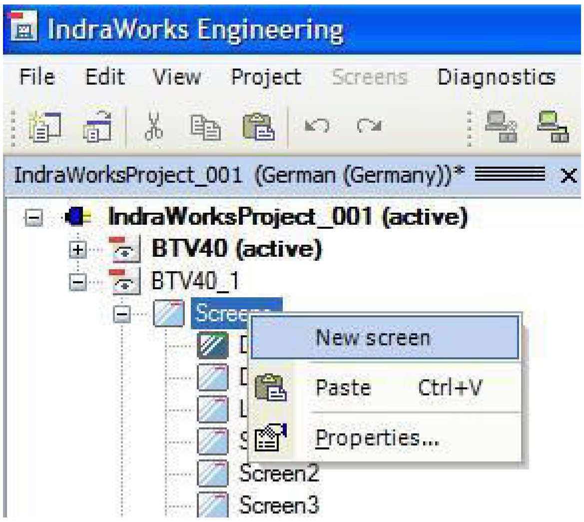


Fig.7-123: Context menu: Screens (Incorporating a WinStudio screen)

2. The screen name is entered in the following dialog.

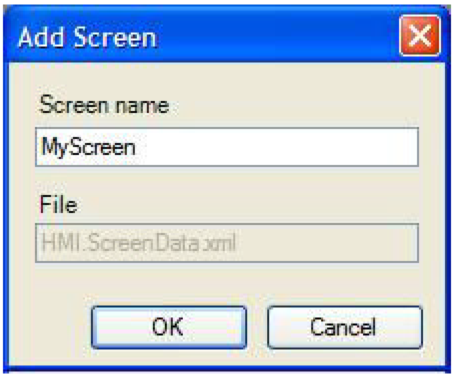


Fig.7-124: Dialog: Add screen (screen name specification)

After the dialog is closed with "OK", the newly generated screen is shown in the Project Explorer next to the screens already generated.

IW Engineering: Edit screen attributes

The screen attributes are called using the node "Screen" and the generated screen name.

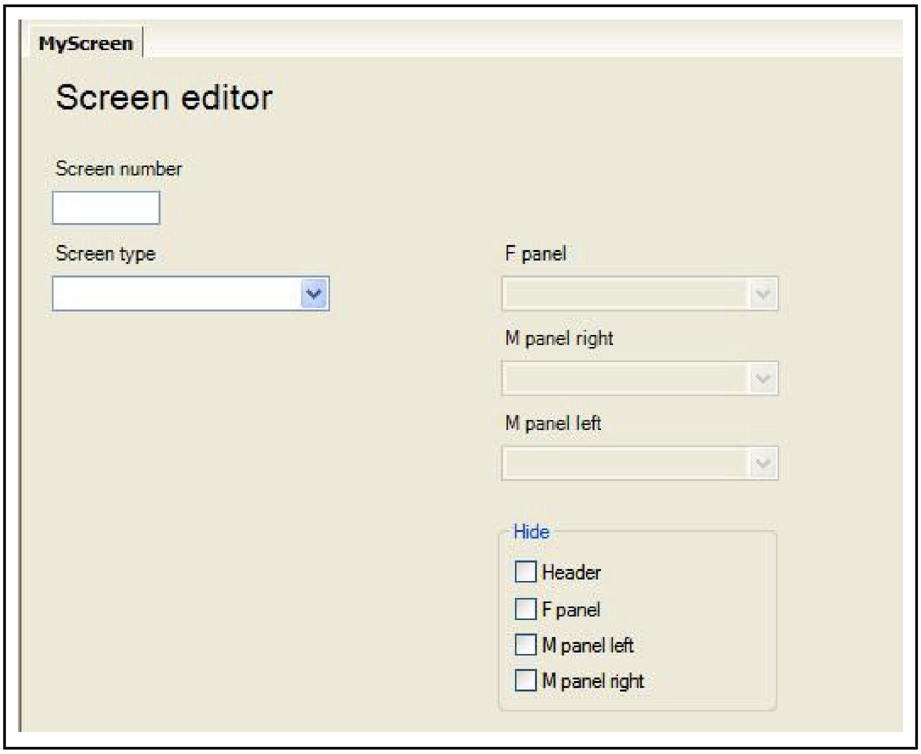


Fig.7-125: Dialog to edit the screen attributes

- The screen number is freely selectable.
- Select "user screen" for the screen type.
- For the screen name, enter the file name under which the screen was saved in "WinStudio".

IW Engineering: Save screen

A modification is saved by selecting "Save Project" on the project node or when the screen dialog is closed.

In order for the modification to be effective in the MTX interface, the HMI node must be used to carry out an update.

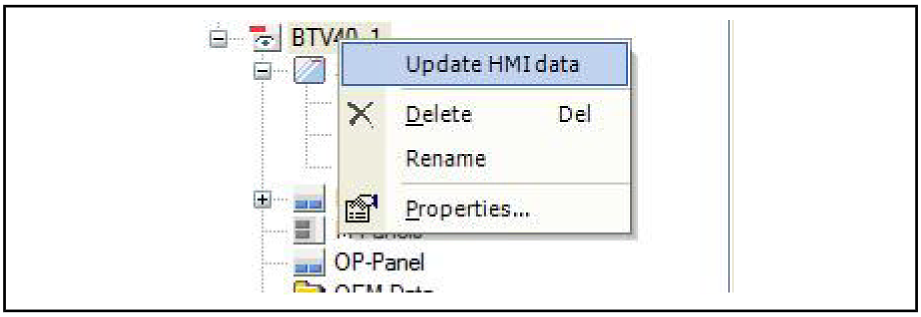


Fig.7-126: Context menu: HMI node (update HMI data)

8 Configuration of PLC-Specific Data in IndraWorks

8.1 Configuration of the PLC-NC Bit Interface

8.1.1 General

Brief Description

The configuration of the PLC-NC bit interfaces can be found as a subnode of the IndraMotion MTX control. The associated node name is "CyclicProcess-Data".

Description

The bit interface consists of the General Interface, the Channel Interface, the Axis Interface and the Spindles Interface. The MSD (Machine Status Display) can also be added to the bit interface.

The configuration of the MSD is described in chapter 9.1 "Machine Fault and Status Display (MSD)" on page 235.

By default, only the general data interface exists and is preassigned.

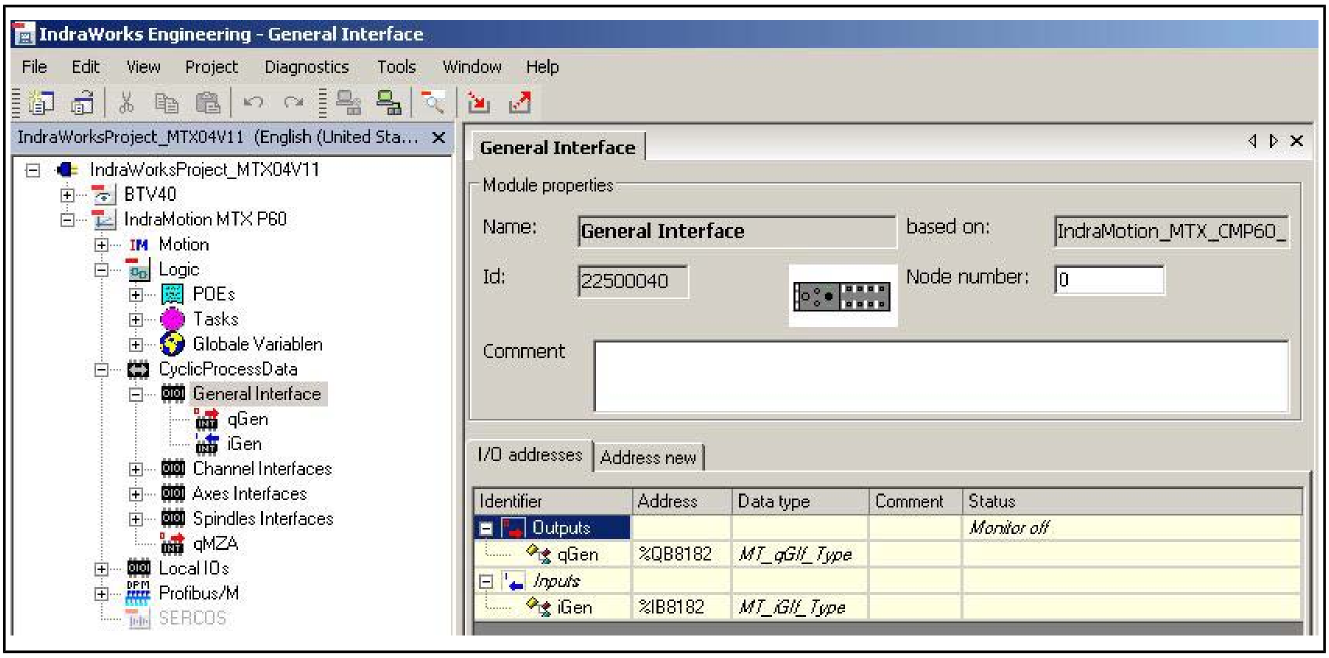


Fig.8-1: The preset bit interface with the "Properties" dialog for the general interface

The bit interface signals should always be seen from the PLC. For example, the signals from qGen are general interface signals from the PLC to the NC.

8.1.2 General Interface

Description

The General Interface always exists; it cannot be deleted. The symbolic name for output signals is "qGen" and the start address for the structure MT_qGif_Type is 5800. The symbolic name for input signals is "iGen" and the start address for the structure MT_iGif_Type is 5800. You can access the "Properties" dialog by double-clicking on the node "General Interface" (or in the context menu via "Open"). In this "Properties" dialog, preassigned names and addresses for the inputs and outputs can be changed and comments can also be assigned comments.

By default, only the general data interface exists and is preassigned.

8.1.3 Channel Interface

General

Brief Description

The MTX supports a switching function channel (channel 0) and up to twelve machining channels. By default, no channels are defined.

Description

New channels are created via the context menu on the "Channel Interfaces" node.

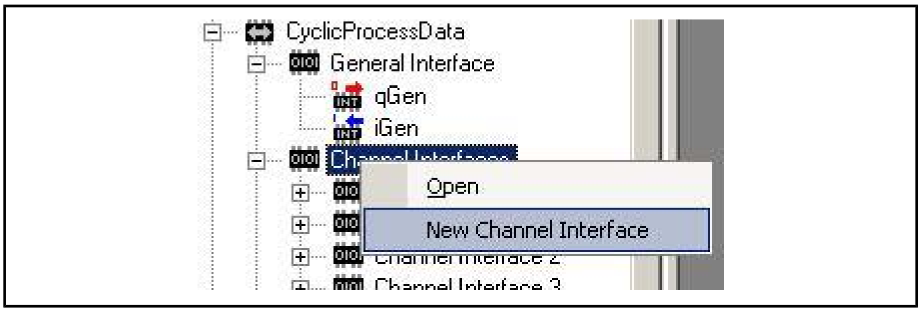


Fig.8-2: Example for creating channel 1

New channels are continuously created starting with channel 0 (switching function channel). The node number is the deciding property for the meaning of a channel. For example, a channel interface with a node number of 2 is the interface for the machining channel 2.

The symbolic name for output signals is "qChan_<node number with preceding zero>" and the start address for the structure MT_qCh_Type (length of 14 byte) is 5804 + (node number x 14). The symbolic name for input signals is "iChan_<node number with preceding zero>" and the start address for the structure MT_iCh_Type is 5804 + (node number x 14).

Open the "Properties" dialog by double-clicking the node "Channel interface <NodeNumber>" node (or via "Open" in the context menu).

The "Properties" dialog can be used to change symbolic names, addresses and comments as well as the corresponding node number.

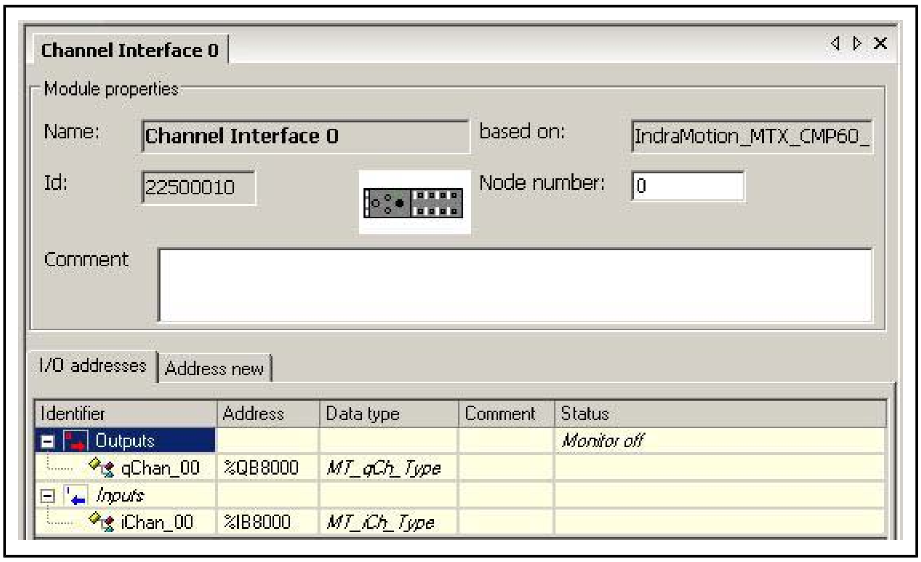


Fig.8-3: "Properties" dialog for the channel interface

☞ The node number for channel interfaces must always be unique. Therefore, each node number may occur only once within the channel interfaces. In this case, the node number is the physical channel number!

Handling Instruction: Channel interface

This chapter describes the use of the Channel Interface.

IW Engineering / Channel Interface: Create new channel interface

1. Click with the right mouse button on the "Channel Interfaces" node.
2. Select "New Channel Interface" by means of the left mouse button.

		Documentation chapter 8.1.3 "Channel Interface " on page 222
Documentation:	IndraWorks Commissioning	Channel interface

IW Engineering / Channel Interface: Delete channel interface

1. Click with the right mouse button on the "Channel Interfaces <Number>" node.
2. Select "Delete" by means of the left mouse button.

		Documentation chapter 8.1.3 "Channel Interface " on page 222
Documentation:	IndraWorks Commissioning	Channel interface

IW Engineering / Channel Interface: Edit

If you double-click the new channel node, additional information and configuration options for the corresponding channel are displayed. This "Properties" dialog can change symbolic names, addresses and comments as well as the corresponding node number. The modifications become effective after exiting the dialog.

☞ The node number for channel interfaces must always be unique. Therefore, each node number may occur only once within the channel interfaces. In this case, the node number is the physical channel number!

		Documentation chapter 8.1.3 "Channel Interface " on page 222
Documentation:	IndraWorks Commissioning	Channel interface

8.1.4 Axis Interface

General

Brief Description The MTX supports up to 64 axes. By default, no axes are yet defined.

Description New axes are created on the "Axes Interfaces" node via the context menu.

Configuration of PLC-Specific Data in IndraWorks

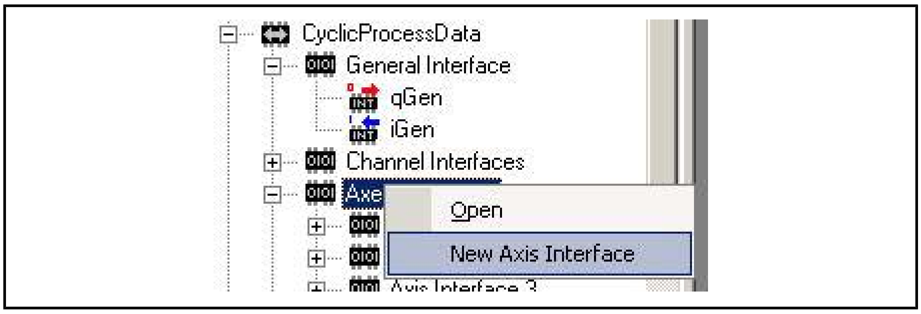


Fig.8-4: Example for creating axis 2

New axes are continuously created starting with axis 1. The node number is the deciding property for the meaning of an axis. For example, an AxisInterface with a node number of 4 is the interface for axis 4.

The symbolic name for output signals is "qAxis_<node number with preceding zero>" and the start address for the structure "MT_qAx_Type" (length of 12 byte) is 5986 + (node number x 12). The symbolic name for input signals is "iAxis_<node number with preceding zero>" and the start address for the structure "MT_iAx_Type" is 5986 + (node number x 12).

Open the "Properties" dialog by double-clicking on the "AxisInterface <Node-Number>" node (or via "Open" in the context menu).

The "Properties" dialog can change symbolic names, addresses and comments as well as the corresponding node number.

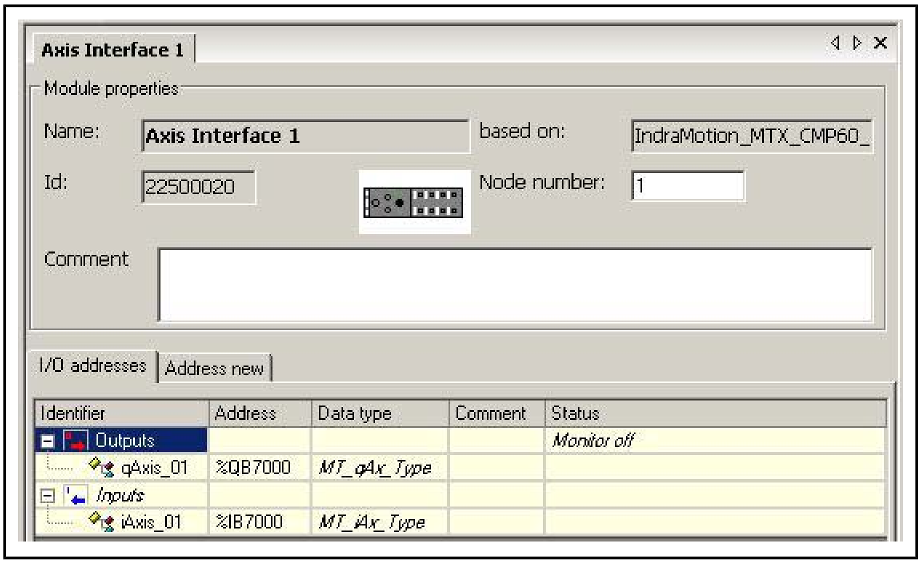


Fig.8-5: "Properties" dialog for the AxisInterface

☞ The node number for Axis Interfaces must always be unique; therefore, each node number may occur only once within the Axis Interfaces. In this case, the node number is the physical axis number!

Handling Instruction: Axis interface

This handling instruction explains how to handle the AxisInterface.

IW Engineering / Axis Interface: Create a new AxisInterface

1. Click with the right mouse button on the "Axis interfaces" node.
2. Select "New AxisInterface" by means of the left mouse button.

		Documentation chapter 8.1.4 "Axis Interface" on page 223
Documentation:	IndraWorks Commissioning	Axis interface

IW Engineering / Axis Interface: Delete an Axis Interface

1. Click with the right mouse button on the "Axis Interface <Number>" node.
2. Select "Delete" by means of the left mouse button.

		Documentation chapter 8.1.4 "Axis Interface" on page 223
Documentation:	IndraWorks Commissioning	Axis interface

IW Engineering / Axis Interface: Edit

If you double-click the new axis node, additional information and configuration options for the corresponding axis are displayed. This "Properties" dialog can change symbolic names, addresses and comments as well as the corresponding node number. The modifications become effective after exiting the dialog.

☞ The node number for Axis Interfaces must always be unique; therefore, each node number may occur only once within the Axis Interfaces. In this case, the node number is the physical axis number!

		Documentation chapter 8.1.4 "Axis Interface" on page 223
Documentation:	IndraWorks Commissioning	Axis interface

8.1.5 Spindle Interface

General

Brief Description The MTX supports up to 16 spindles. By default, no spindles are yet defined.

Description New spindles are created via the context menu on the "Spindle Interfaces" node.

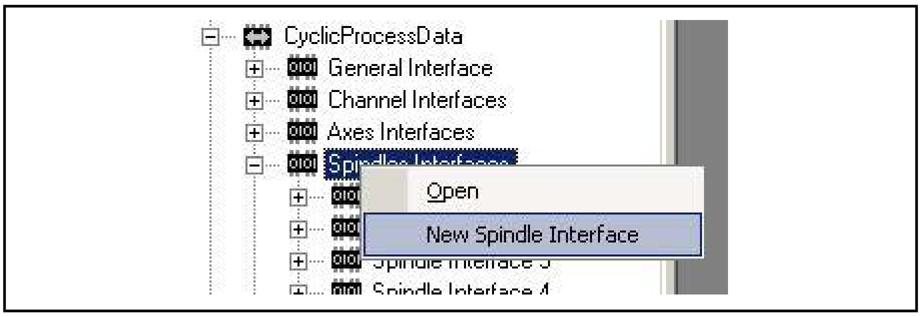


Fig.8-6: Example for creating spindle 2

New spindles are continuously created starting with spindle 1. The node number is the deciding property for the meaning of a spindle. For example, a spindle interface with a node number of 6 is the interface for spindle 6.

The symbolic name for output signals is "qSpindle_<node number with preceding zero>" and the start address for the structure "MT_qSp_Type" (length of 12 bytes) is 6754 + (node number x 12). The symbolic name for input signals is "iSpindle_<node number with preceding zero>" and the start address for the structure "MT_iSp_Type" is 6754 + (node number x 12).

You can open the "Properties" dialog by double-clicking the node "Spindle Interface <NodeNumber>" (or via "Open" in the context menu).

The "Properties" dialog can change symbolic names, addresses and comments as well as the corresponding node number.

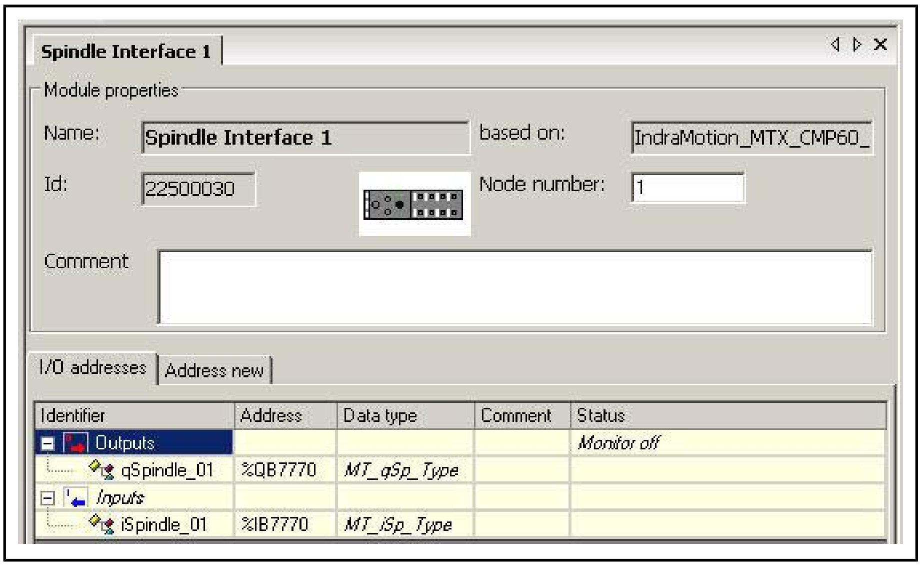


Fig.8-7: "Properties" dialog for the spindle interface

The node number for spindle interfaces must always be unique; therefore, each node number may occur only once within the spindle interfaces. In this case, the node number is the physical spindle number.

Handling Instruction: Spindle interface

This handling instruction explains how to use the Spindle Interface.

IW Engineering / Spindle Interface: Create new Spindle Interface

1. Click with the right mouse button on the "Spindle Interface" node.
2. Select "New Spindle Interface" by means of the left mouse button.

		Documentation chapter 8.1.5 "Spindle Interface" on page 225
Documentation:	IndraWorks Commissioning	Spindle interface

IW Engineering / Spindle Interface: Delete Spindle Interface

1. Click with the right mouse button on the "Spindle Interface <Number>" node.
2. Select "Delete" by means of the left mouse button.

		Documentation chapter 8.1.5 "Spindle Interface" on page 225
Documentation:	IndraWorks Commissioning	Spindle interface

IW Engineering / Spindle Interface: Edit

If you double-click the new spindle node, additional information and configuration options for the corresponding spindle are displayed. This "Properties" dialog can change symbolic names, addresses and comments as well as the corresponding node number. The modifications become effective after exiting the dialog.

> The node number for spindle interfaces must always be unique. Therefore, each node number may occur only once within the spindle interfaces. In this case, the node number is the physical spindle number.

		Documentation chapter 8.1.5 "Spindle Interface" on page 225
Documentation:	IndraWorks Commissioning	Spindle interface

8.1.6 Function "Readdress"

Brief Description

The "Readdress" function is present in the "Properties" dialog of almost every node of the PLC-NC bit interface. This function can be used to redefine the I/O addresses of all subordinate nodes.

Description

If you enter a new start address for input and output signals in the "Readdress" dialog, the I/O addresses of all subsidiary nodes or - if no subordinate nodes exist - of this node are redefined after the "Accept" button is pressed.

The following figure shows an example of the "Readdress" dialog for all channel interfaces.

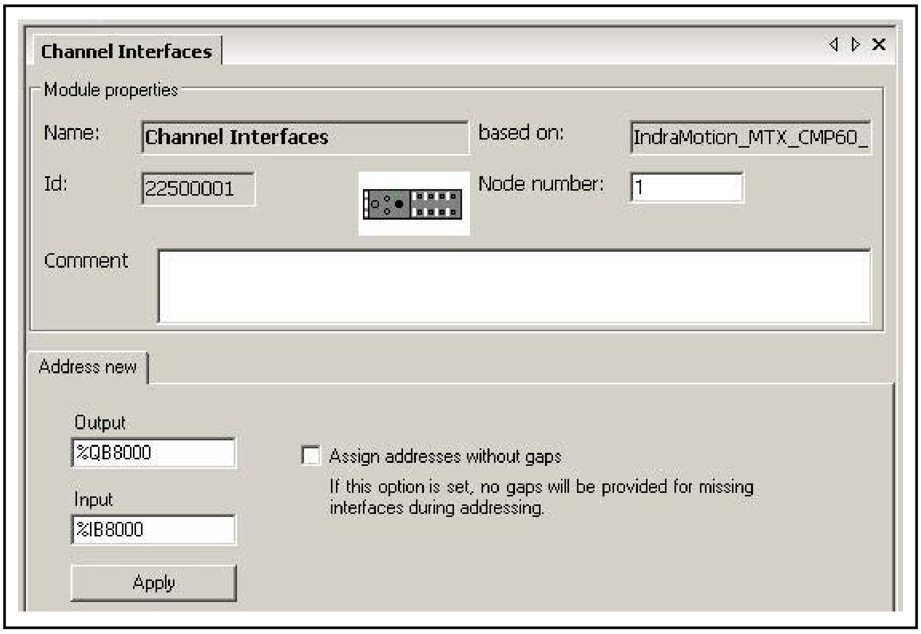


Fig.8-8: "Properties" dialog for all channel interfaces

The addressing of the I/O addresses for each node occurs according to the following pattern:

Address = (start address) + (node number) x (length of the interface type in bytes)

Length of the corresponding interface:

- General interface 4 byte

- Channel interface 14 bytes
- Axis interface 12 bytes
- Spindle interface 12 bytes

Using this calculation, gaps in addressing may occur due to missing interfaces.

☞ To be able to carry out addressing without gaps in the case of such interfaces, the option "Assign Addresses without Gaps" must be set.

8.1.7 Use of the Interface Signals in the PLC

Brief Description The defined interface signals are automatically announced as global variables in the PLC.

Description Therefore, the interface signals can be used immediately, without an additional declaration, in the program portion of the PLC.

The individual interface signals are based on structures. As a result, you can access interface signals according to the syntax <designation of the interface type>.<associated interface signal>.

The Intellisense function provides a great deal of help in making entries here. If a dot "." is entered after the designation for the interface type, a selection list of all associated variables opens. An element can be selected from this list and inserted after the dot by pressing the ENTER key. Insertion also functions by double-clicking the list element.

The following figure shows an example of using the channel output signal "qCh_NCStart" (processing start of an NC block or an NC program) for channel 1.

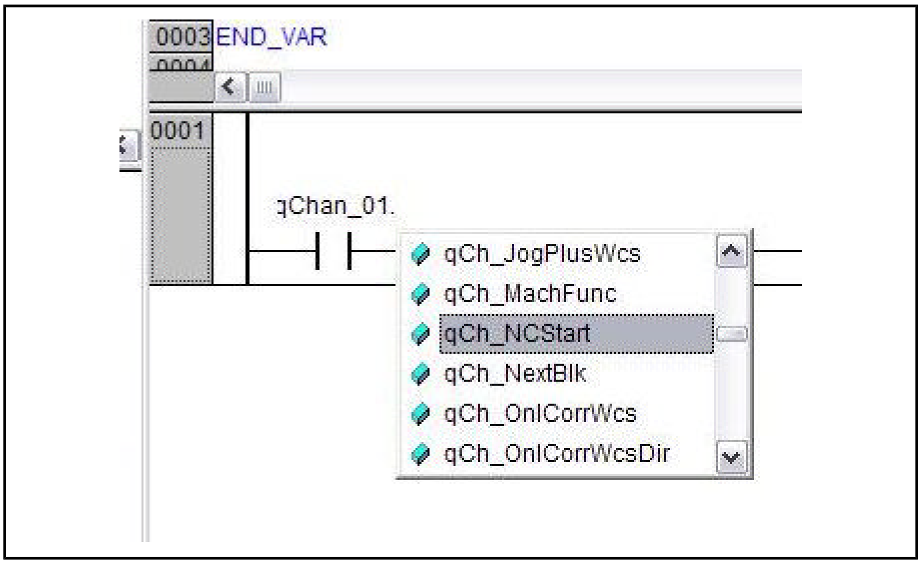


Fig.8-9: Example of using a channel output signal

☞ The Intellisense function works only if the option "List components" is activated in the IndraLogic project options in the category "Editor".

8.2 Configuration of the Local Inputs

8.2.1 General

Brief Description

Local I/Os in the MTX 04VRS are digital inputs/outputs connected with the IndraMotion MTX CMP60 control card using ribbon cables. These local I/Os include:

- M-keys (freely configurable machine function keys)
- High-speed I/O card (8 digital inputs)
- High-speed I/O card (8 digital outputs)

Description

Local I/Os are added by clicking on the "Local IOs" node in the context menu.

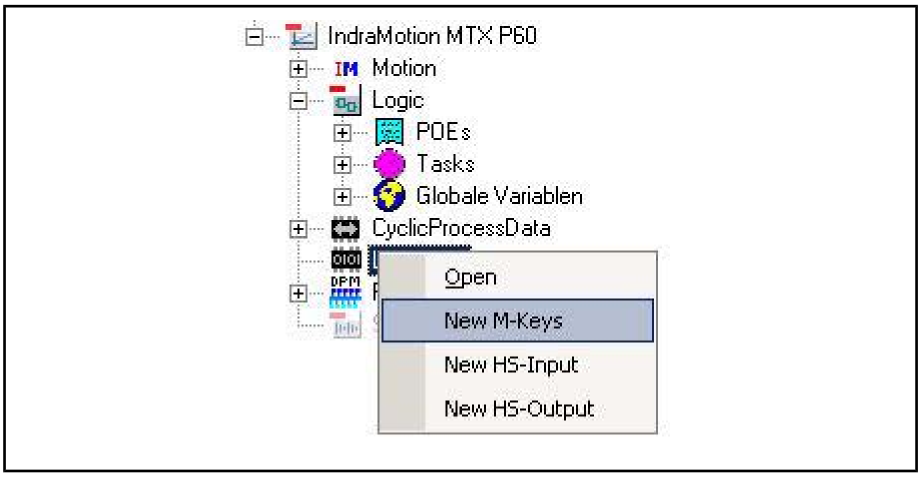


Fig.8-10: Context menu for adding local I/Os

8.2.2 M-keys

General

Description

After adding the M-keys, the corresponding "Properties" dialog is opened by double-clicking on the node "M-keys" in "Local I/Os". In this dialog, you can assign or change symbolic names for the 2 byte inputs and the corresponding addresses as well as remarks. The M-keys are 2-byte digital inputs that can be evaluated in the PLC.

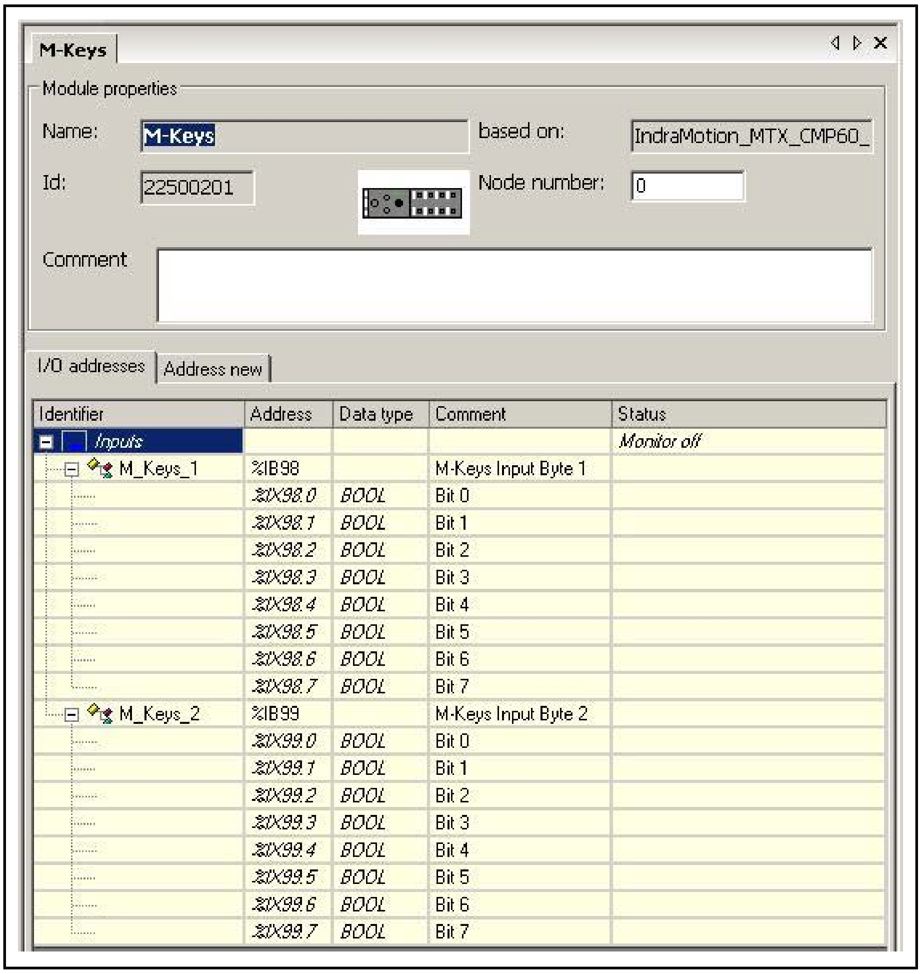


Fig.8-11: "Properties" dialog for M-keys

☞ The function of each machine function key is defined in the HMI configuration.

Handling Instruction: M-keys

This handling instruction describes the use of the machine function (M) keys.

IW Engineering / M-keys: Create new M-keys

1. Click with the right mouse button on the "Local IOs" node.
2. Select "New M-keys" by means of the left mouse button.

		Documentation chapter 8.2.2 "M-keys" on page 229
Documentation:	IndraWorks Commissioning	M-keys

IW Engineering / M-keys: Delete M-keys

1. Click with the right mouse button on the "M-keys" node.
2. Select "Delete" by means of the left mouse button.

		Documentation chapter 8.2.2 "M-keys" on page 229
Documentation:	IndraWorks Commissioning	M-keys

IW Engineering / M-keys: Edit

If you double-click the new M-keys node, additional information and configuration options for the corresponding M-keys are displayed. This "Properties" dialog can change symbolic names, addresses and comments. The modifications become effective after exiting the dialog.

		Documentation chapter 8.2.2 "M-keys" on page 229
Documentation:	IndraWorks Commissioning	M-keys

8.2.3 Digital Inputsof the I/O Card(HS Input)

General

Description

After adding the digital inputs of the I/O card, open the corresponding "Properties" dialog by double-clicking on the "HS input" node under "Local I/Os". In this dialog, you can assign or change symbolic names for the input byte and related addresses as well as comments. The 8 digital inputs of the I/O cards can be evaluated both in the PLC and in the NC.

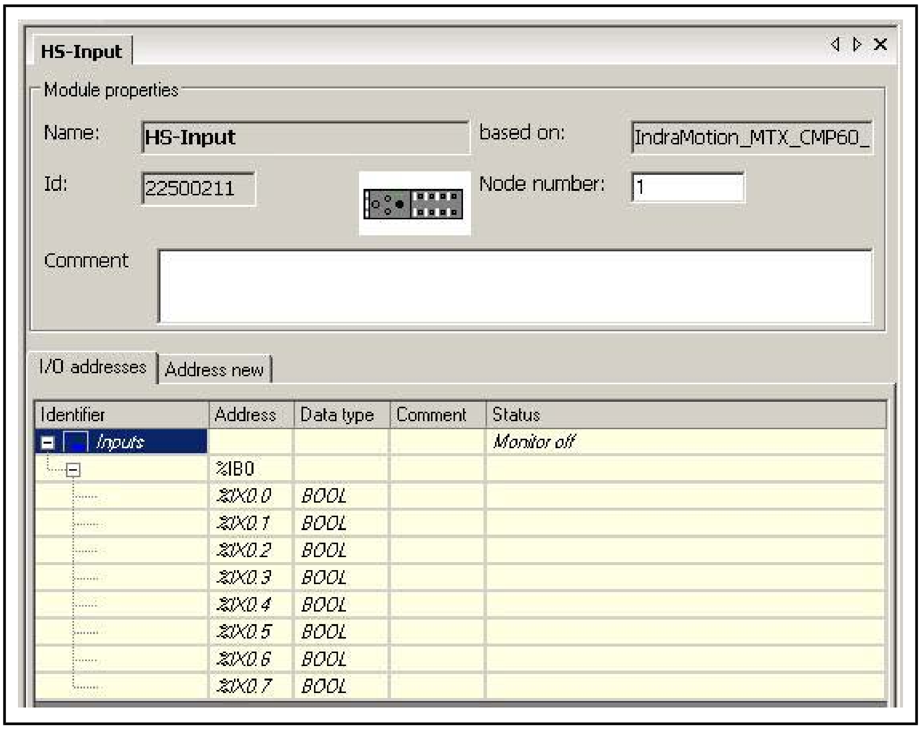


Fig.8-12: "Properties" dialog for the HS input

☞ The NC-side configuration is described in manual "Bosch Rexroth MTX Machine Parameters".

Handling Instruction: HS input

This handling instruction describes the use of the high-speed inputs.

IW Engineering / HS Input: Create new HS inputs

1. Click with the right mouse button on the "Local IOs" node.
2. Select "New HS input" by means of the left mouse button.

		Documentation chapter 8.2 "Configuration of the Local Inputs" on page 229
Documentation:	IndraWorks Commissioning	HS input

IW Engineering / HS Input: Delete HS inputs

1. Click with the right mouse button on the "HS input" node.
2. Select "Delete" by means of the left mouse button.

		Documentation chapter 8.2 "Configuration of the Local Inputs" on page 229
Documentation:	IndraWorks Commissioning	HS input

IW Engineering / HS Input: Edit

If you double-click the new HS input node, additional information and configuration options for the corresponding HS input are displayed. This "Properties" dialog can change symbolic names, addresses and comments. The modifications become effective after exiting the dialog.

		Documentation chapter 8.2 "Configuration of the Local Inputs" on page 229
Documentation:	IndraWorks Commissioning	HS input

8.2.4 Digital Outputs of the IO Card (HS Output)

General

Description

After adding the digital outputs of the I/O card, open the corresponding "Properties" dialog by double-clicking on the "HS Output" node under "Local I/Os". In this dialog, you can assign or change symbolic names for the output byte and related addresses as well as comments. The 8 digital inputs of the I/O cards can be evaluated both in the PLC and in the NC.

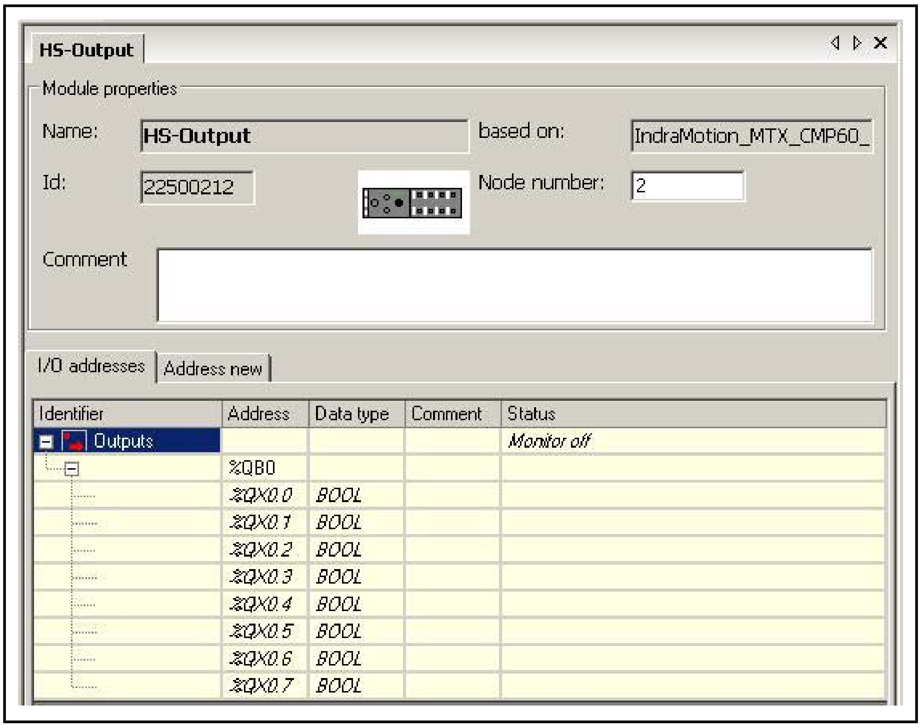


Fig.8-13: "Properties" dialog for HS output

The NC-side configuration is described in manual "Bosch Rexroth IndraMotion MTX Machine Parameters".

Handling Instruction: HS output

This handling instruction describes the use of the high-speed output.

IW Engineering / HS Output: Create new HS outputs

1. Click with the right mouse button on the "Local IOs" node.
2. Select "New HS output" by means of the left mouse button.

		Documentation chapter 8.2 "Configuration of the Local Inputs" on page 229
Documentation:	IndraWorks Commissioning	HS output

IW Engineering / HS Output: Delete HS outputs

1. Click with the right mouse button on the "HS output" node.
2. Select "Delete" by means of the left mouse button.

		Documentation chapter 8.2 "Configuration of the Local Inputs" on page 229
Documentation:	IndraWorks Commissioning	HS output

IW Engineering / HS Output: Edit

If you double-click the new HS output node, additional information and configuration options for the corresponding HS outputs are displayed. This "Properties" dialog can change symbolic names, addresses and comments. The modifications become effective after exiting the dialog.

		Documentation chapter 8.2 "Configuration of the Local Inputs" on page 229
Documentation:	IndraWorks Commissioning	HS output

9 Diagnostics

9.1 Machine Fault and Status Display (MSD)

9.1.1 What is the MSD?

Definition

In order to carry out trouble-shooting quickly in case of interruptions in the sequence of operations of machine tools, a reporting system that issues the corresponding messages to the operator in plain text is required.

The Machine fault and Status Display, abbreviated MSD in the following, permits a total of 8096 messages to be displayed in the NC user interface. These can be displayed as faults, warnings or notes.

The messages are defined in plain text in a file (MSD file) and activated when the respective marker is set by the PLC sequential program.

All MSD messages can be saved as plain text in a logbook with the corresponding timestamps.

9.1.2 Displaying MSD Messages

Description

MSD messages are displayed in message lines in the header of the user interface. By default, fault messages and warnings are shown alternatively in the upper line and notes are shown in the lower line. In addition, a corresponding bitmap is displayed if messages appear in the header. If fault messages appear, the bitmap for the fault display flashes.

Diagnostics

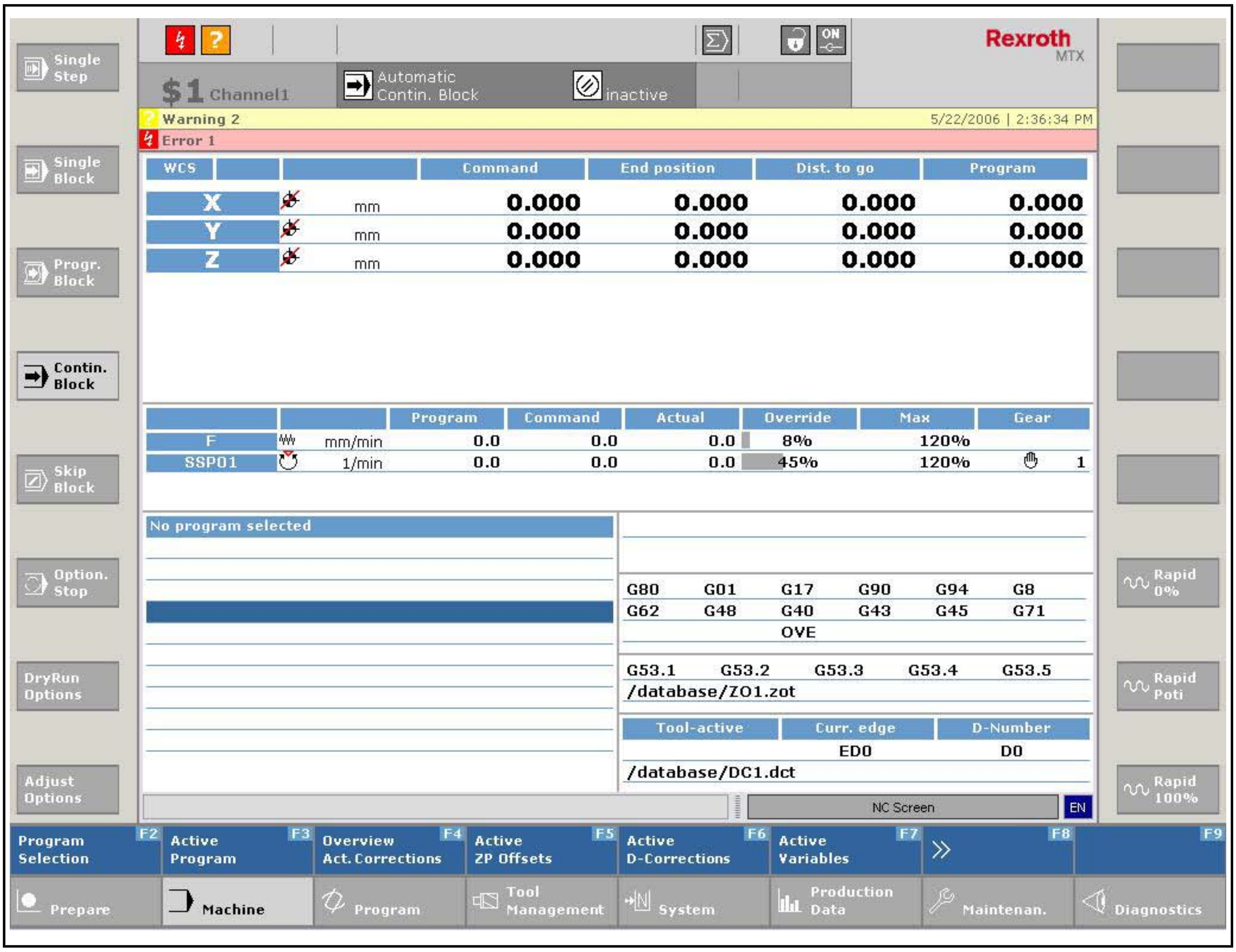
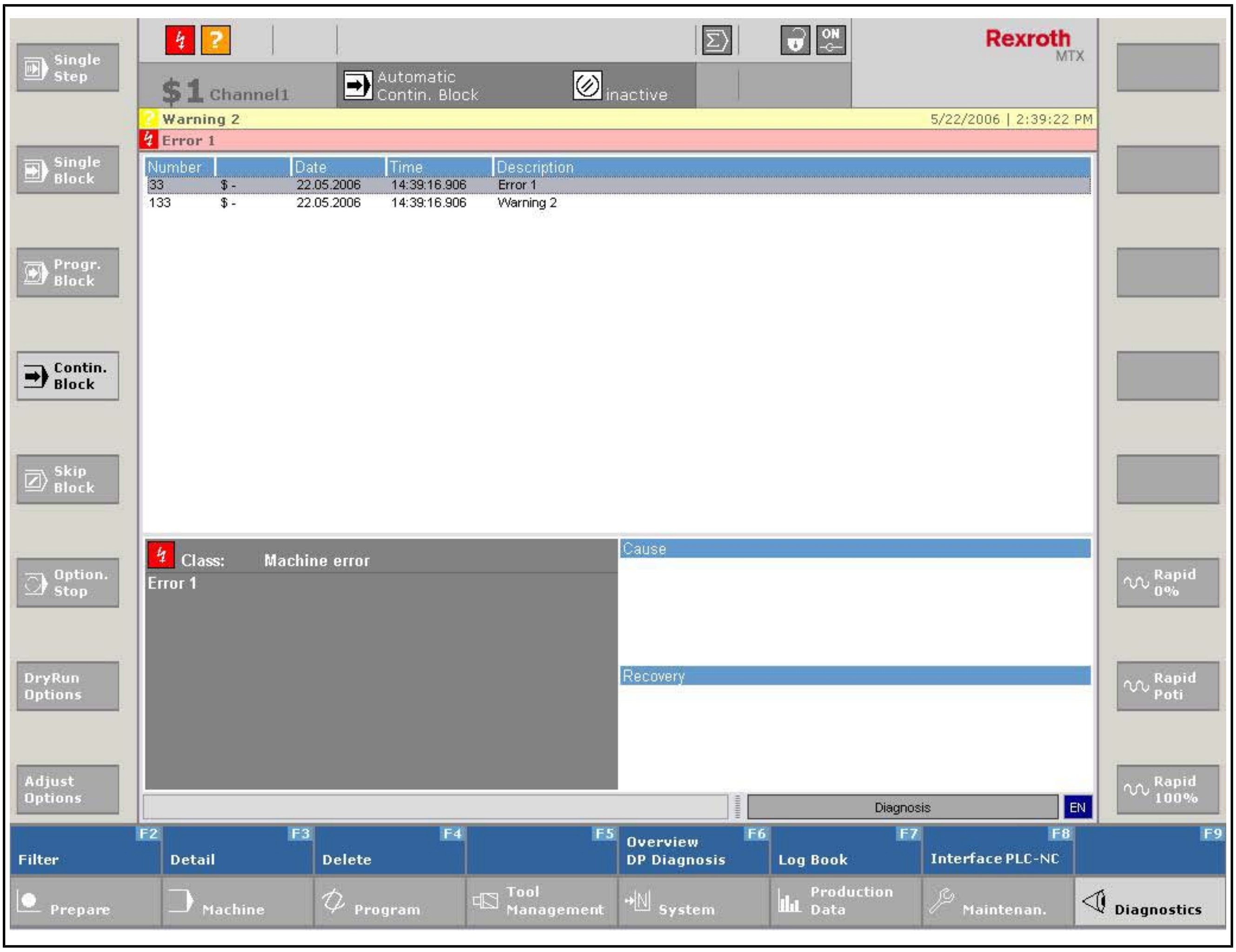


Fig.9-1: Display of MSD messages in the header

All MSD messages that occur are displayed in the "Diagnostics" (OP9). The message that was activated last is located at the top of the list. The MSD message number (1 - 8096), the date and time of occurrence and the message text (description) are displayed for each message. In addition, cause and remedy texts can be displayed for the message.



1 Bitmap for fault message
2 Bitmap for warning
3 Bitmap for note

Fig.9-2: Display of MSD messages in "Diagnostics"

If a message is selected using the cursor keys and <F7> Detail is then pressed, additional details regarding the selected message are displayed.

Further information about the "Diagnostics" screen of IndraWorks HMI can be found in the IndraWorks HMI documentation.

9.1.3 Commissioning Procedure

The following steps are required to commission the MSD message system:

- Setting the parameters of the NC
- Generating the MSD text file(s)
- Configuring the PLC interface
- Programming the PLC interface

9.1.4 Setting the Parameters of the NC

The MSD message system is configured using the parameter **"Cycle Time"**. This parameter specifies the updating time of the message system in milliseconds. In general, an updating time of 500 ms is sufficient.

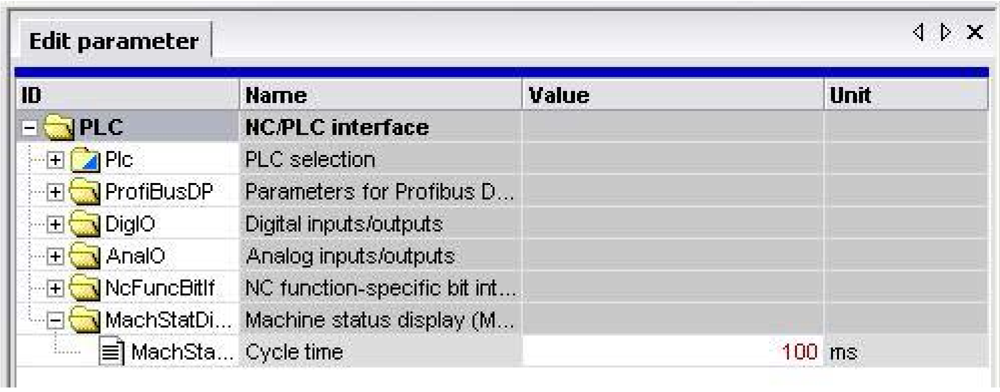


Fig.9-3: Setting MSD message system parameters

If 0 is entered under **"Cycle Time"**, no MSD messages are displayed or entered in the logbook!

9.1.5 Structure of MSD Files

The MSD texts are entered in so-called MSD files. These are stored in the MTX file system in subdirectory **/usrfep**. The files can be saved in ANSI or UTF-8 format. The UTF-8 format is required to display Asian characters, for example.

Currently, only MSD files in ANSI format can be edited using the NC Editor. Files in UTF-8 format must be edited externally

A separate file is generated for each language in which the MSD messages are displayed. The file name is always msdtexts.xxx; the file extension xxx has been specified as follows for each language:

Directory /usrfep can be accessed only using the MTX user interface. Files can (currently) not be edited directly in this directory. Therefore, the files should be generated/saved in the /mount directory (c:\mnt). Then they can be copied from /mnt to /usrfep within the MTX interface.

Language	File extension
German	.049
English	.044
French	.033
Italian	.039
Czech	.420

Fig.9-4: File extensions

The languages German and English are always included in the scope of delivery. A corresponding language extension must be purchased as an option for every additional language.

Definition

A total of 8096 messages can be defined. Each message is assigned a serial number from 1 to 8096. The number list can contain gaps. The messages are distinguished as follows:

- Machine errors have an ID of E
- Machine warnings have an ID of W
- Machine notes have an ID of M

Precisely one ID can be assigned to each message number.

There is a message text for every message; optionally, a cause-and-remedy text can also be defined. The length of the message text is limited to 128 characters. A line break can be forced within the texts by inserting "\n". The file can contain comments in the form (<comment>).

9.1.6 Structure of a Message Line

Program:

```
(<E,M,W><1..2048>,<message text>{##<cause text>##<remedy text>})
|       |         |                |               |
|       |         |                |               | remedy text max.
|       |         |                |               | 128 characters
|       |         |                |
|       |         |                | cause text max. 128 characters
|       |         |
|       |         | message text max. 128 characters
|       |
|       | message number
|
| message type

Example:
(E200,error message number 200##cause of error 200##remedy of error 200)
(W201,warning number 201\nsecond line of warning 201)
```

9.1.7 Example MSD File (msdtexts.049)

Program:

```
(Fault in function group 1)
(E4, (FG1-3) +S2-A40/P100.1; AS-i Master 1: Configuration is inactive)
(E5, (FG1-4) +S2-A40/P100.1; AS-i Master 1: AS-i  Power Fault)
(E0006, (FG1-5) +S2-A40/P100.1; AS-i Master 2: Configuration is inactive)
(E7, (FG1-6) +S2-A40/P100.1; AS-i Master 2: AS-i  Power fault)

(Warnings for Function Group 7)
(W289, (FG7-0) +S2-I32.7/P151.8; circuit breaker is not switched on)
(W290, (FG7-1) +S2-I3.1/P144.2; Machine is not switched on)
(W0291, (FG7-2) +M-S96.0/P251.6; safety door 1 of the workplace is not locked)
(W292, (FG7-3) +M-S97.0/P252.2; safety door 2 of the workplace is not locked)

(Fault of the function group 11)
(E482, (FG11-1) Drive Lock X,Y,Z1,Z2-Axis; Spindle 1 HDK pressure achieved missing)
(E483, (FG11-2) Drive Lock X,Y,Z1,Z2-Axis; Spindle 2 HDK pressure achieved missing)
(E484, (FG11-3) Drive Lock X,Y,Z1,Z2-Axis; Spindle 1 speed achieved missing)
(E485, (FG11-4) Drive Lock X,Y,Z1,Z2-Axis; Spindle 2 speed achieved missing)

(Notes on the function group 10)
(M1159, (FG10-6) control reset (M25 channel 1) missing)
(M1160, (FG10-7) control reset (M25 channel 2) missing)
(M1161, (FG10-8) control reset (M25 channel 3) missing)
(M1162, (FG10-9) control reset (M25 channel 4) missing)
(M1163, (FG10-10) control reset (M25 channel 5) missing)
```

☞ Modifications to MSD files go into effect by switching the active language or by restarting the system (soft reset).

9.1.8 PLC Interface

The PLC interface to the MSD consists of a data structure that is configured within the PLC/CNC interface. You can choose between 2048 or 8096 messages for the width of the MSD interface.

9.1.9 Configuration of the PLC Interface

The interface is executed in the hardware configuration of the PLC/CNC interface (CyclicProcessData). The MSD interface can be inserted by clicking the right mouse button on node "CyclicProcessData". Here, you must choose between "New qMZA" for 2048 messages and "New qMZA_Ext" for 8096 messages.

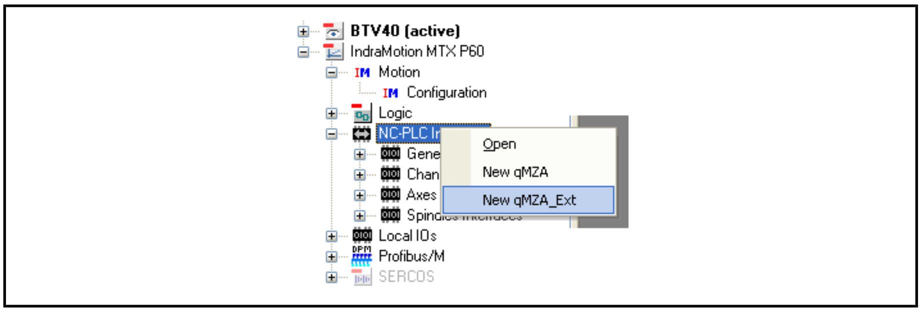


Fig.9-5: Inserting the MSD interface

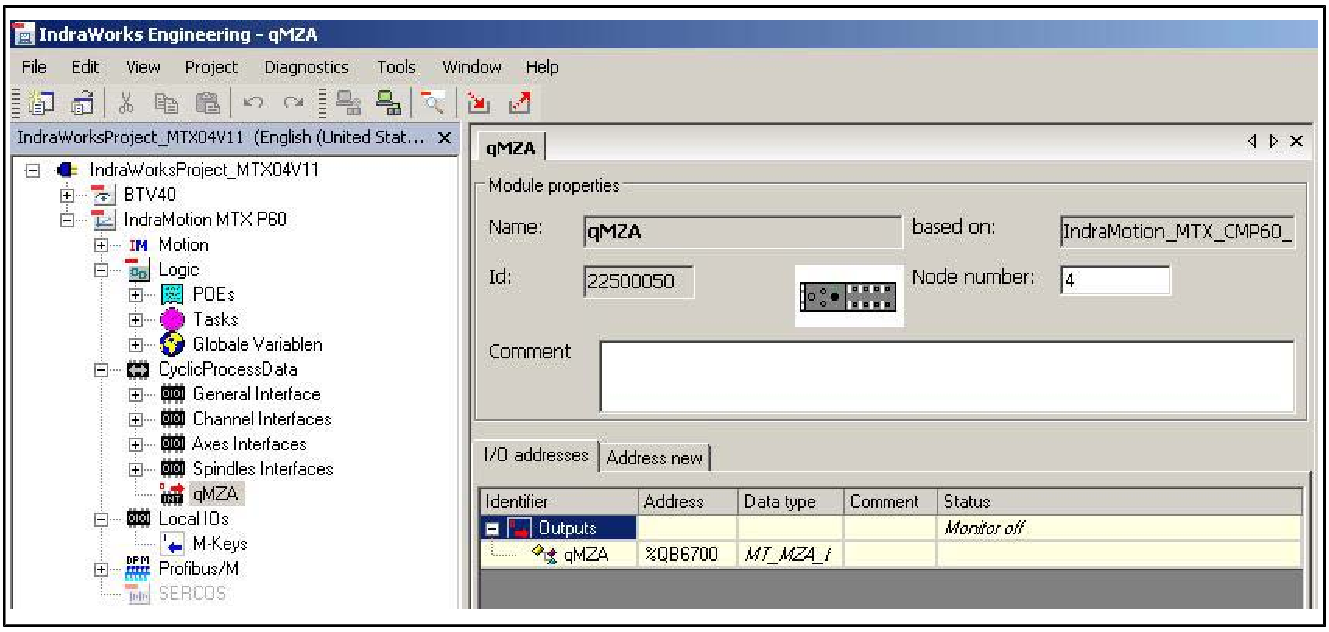


Fig.9-6: Configuring the MSD interface

The symbolic name of the structure should be qMZA. The default address is at %QB1000.

9.1.10 Programming the PLC Interface

General

The "qMZA" data structure provides a separate bit for each message of the MSD. The bits are addressed using their symbolic names. The message is shown in the diagnostics as long as the relevant bit for a message is TRUE. The time of the rising flank for the bit is entered in the logbook as "Message arrives" and the time of the falling flank is entered as "Message departs".

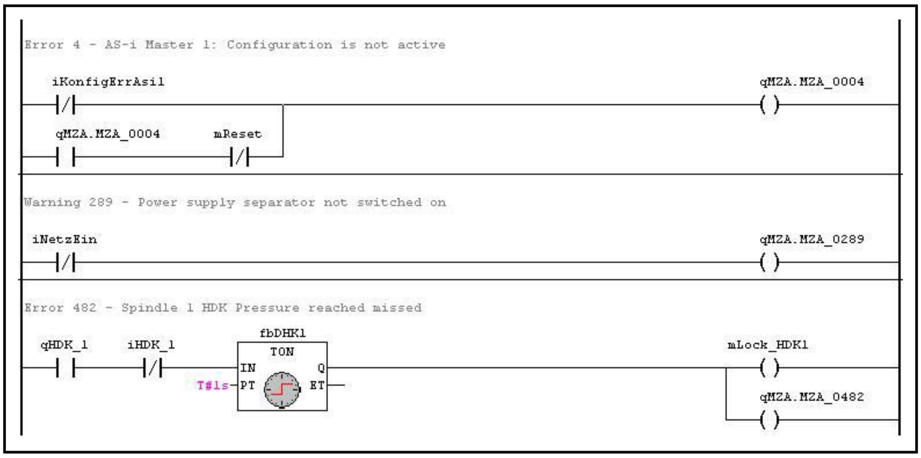


Fig.9-7: Example of MSD programming

Handling Instruction: MSD Interface

This chapter describes the use of the MSD interface.

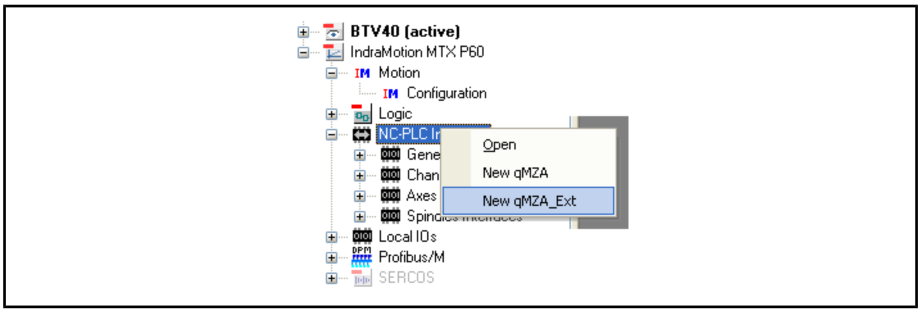


Fig.9-8: Creating an MSD interface

IW Engineering / CyclicProcessData: Create a new MSD interface

1. Click with the right mouse button on the "CyclicProcessData" node.
2. Select the MSD interface.

   Here, you must choose between "New qMZA" for 2048 messages and "New qMZA_Ext" for 8096 messages.

		Documentation chapter 9.1 "Machine Fault and Status Display (MSD)" on page 235
Documentation:	IndraWorks Commissioning	MSD diagnostics

IW Engineering / CyclicProcessData: Delete an MSD interface

1. Click with the right mouse button on the "qMSD" or on the "qMSD_Ext" node.
2. Select "Delete" by means of the left mouse button.

		Documentation chapter 9.1 "Machine Fault and Status Display (MSD)" on page 235
Documentation:	IndraWorks Commissioning	MSD diagnostics

IW Engineering / CyclicProcessData: Edit

1. Double-click the node "qMSD" or "qMSD_Ext"

Additional information and configuration possibilities for the corresponding MSD interface are shown. This "Properties" dialog can change symbolic names, addresses and comments.
The modifications become effective after exiting the dialog.

		Documentation chapter 9.1 "Machine Fault and Status Display (MSD)" on page 235
Documentation:	IndraWorks Commissioning	MSD diagnostics

Handling Instruction: MSD Configuration

This chapter describes how to configure the MSD.

IW Engineering / Configuration: Enable MSD

The MSD message system is configured using the parameter "Cycle Time". This parameter specifies the updating time of the message system in milliseconds. In general, an updating time of 500ms is sufficient.

		Documentation chapter 9.1 "Machine Fault and Status Display (MSD)" on page 235
Documentation:	IndraWorks Commissioning	MSD diagnostics

IW Engineering / Configuration: Disable MSD

If 0 is entered under **"Cycle Time"**, no MSD messages are displayed or entered in the logbook.

		Documentation chapter 9.1 "Machine Fault and Status Display (MSD)" on page 235
Documentation:	IndraWorks Commissioning	MSD diagnostics

9.2 ProVi

9.2.1 Commissioning and Programming ProVi Messages

General

ProVi messages are issued by the PLC. They can be displayed in the HMI interface. ProVi messages can also be logged in a logbook. They can be grouped into five message types:

- Error
- Note
- Warning
- Startup prerequisite
- Setup diagnostics

All message types can be grouped into different modules.

A fault category and a message group can also be assigned to each message. ProVi messages can be programmed to be set, i.e. the message is displayed until it is reset by calling an FM (function module). The message texts can be entered multilingually directly in IndraLogic.

Handling Instruction: Commission and Program ProVi Messages

There is a separate documentation for ProVi which answers all questions in detail.

The chapter "First Steps" helps the user to become familiar with ProVi.

IW Engineering: Work with ProVi

		Documentation chapter 9.2 "ProVi" on page 242
Documentation:	PLC program development with Rexroth IndraLogic	ProVi messages

9.3 NC Diagnostics for Small Operator Panels

9.3.1 General

To implement the NC diagnostics for small operator panels of the VEP, VCP and VCH series, the diagnostics texts have to be transferred from the source path **"...\Rexroth\IndraWorks\mtx\text"** to control to provide them in the PLC. A download mechanism in the IndraWorks Engineering is used for the transfer.

9.3.2 Downloading Language Files

Calling Download Dialog

The download dialog is started either via the context menu of a device or via the device-specific entries in the main menu. Therefore select **Load Diagnostic Text in Control...**

Diagnostics

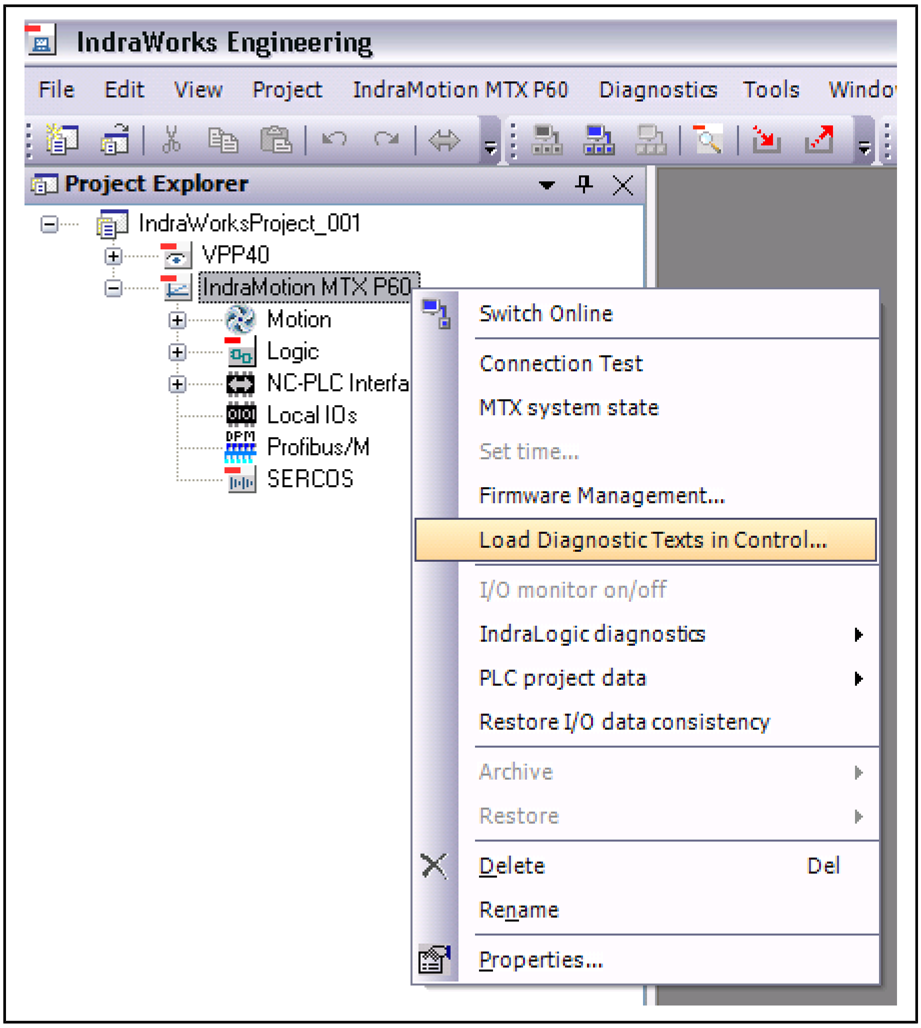


Fig.9-9: Calling Download Dialog

The download dialog provides the following functionalities:

- Loading diagnostic texts
- Displaying available diagnostics texts
- Displaying the diagnostics texts currently located in the control
- Deleting diagnostics texts in the control

Describing the dialog elements

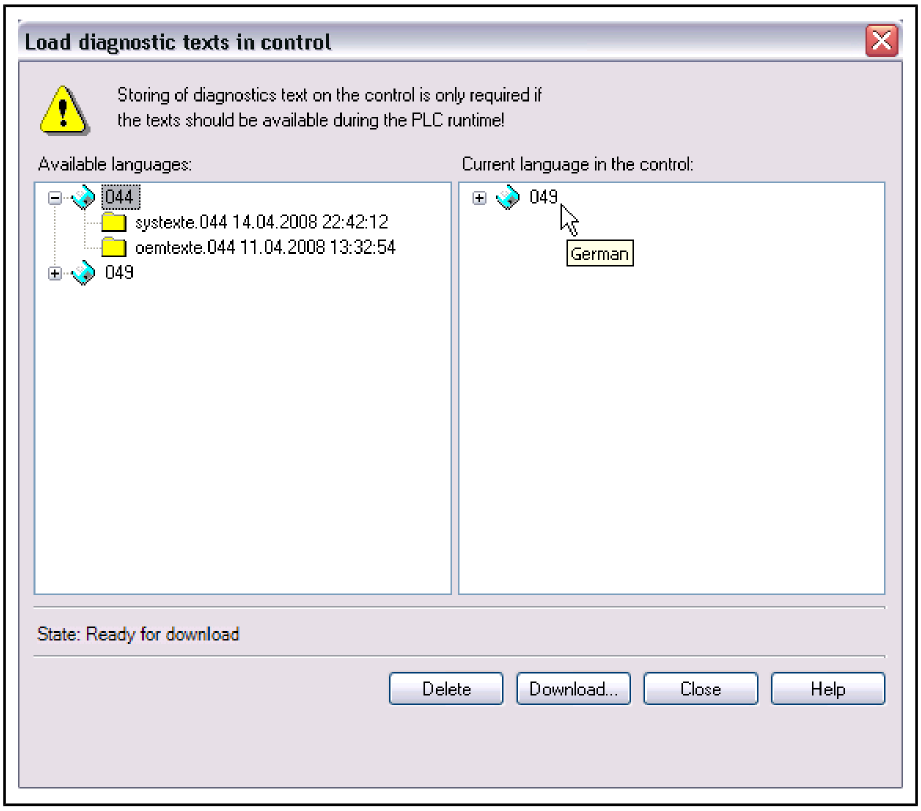


Fig.9-10: Dialog "Load Diagnostic Text in Control"

The left window of the dialog shows the languages of the diagnostics texts available as country code for a download (a tool tip shows the language as plain text). If you open the language version tree, the corresponding language files are show with time stamp.

The language versions in the control are shown in the right window. By comparing times stamps, as possible required update can be identified.

Downloading single files is not possible. The language selected is always completely loaded with all its files available in the installation directory. Additionally, index files providing a faster access on the language files are created in the control. The progress bars informs on the download status.

Storage location of the language and index files in the control is the **"root/usrfep/text"** directory. If any files exist, they are overwritten.

The "Delete" function removes all language and index files for the selected language version in the control.

> If the are desired texts in the control, the **MT_DiagText** function block can be used in the PLC later on to provide an image of the NC error and the message database. Further information, the description of the function block as well as a configuration example can be found in the documentation"PLC Interface".

9.4 SCP Analyzer

9.4.1 General

The SCP Analyzer determines a reduced OPC or SCP communication. All client processes requesting data of a local SCP server are displayed with their object hierarchy (processes, devices, clients, groups, items).

When the application is started, it synchronizes automatically with the local server and displays the current object hierarchy.

Save The current state of the SCP server can be exported in a file for further processing. A file named "SCPAnalyzer.log " is created which records and saves the current state of all SCP objects. Press "Save to file" in the toolbar or the menu item. **File ▸ Save**.

Clipboard The selected item of the group item list can be copied to the clipboard. Press the "Save to clipboard" button in the toolbar.

Always on Top The application can always be switched visibly via the menu item **Edit ▸ Always on Top** .

9.4.2 Display Elements

SCP process tree The left process tree represents the object hierarchy with regard to the logical devices via the local SCP server. By means of filter buttons in the toolbar, the individual layers can be shown or hidden.

Group item list Items running dynamically in one communication group (group items) are dynamically recorded across all clients and entered in the group item list. There, some measured parameters are visualized as values and some as bar chart.

- Column "Status":
  Group item or related group is active/inactive
- Column "Group Item":
  Name of the group item
- Column "Update Rate [ms]":
  Nominal update rate of the group items in [ms]
- Column "Cache [ms]":
  Actual update rate between the SCP server and the logical device in [ms]
- Column "Client Update [ms]":
  Actual update rate between the SCP server and the client in [ms]

9.4.3 SCP Object Hierarchy

Each SCP/OPC client is identified as best as possible and entered in an object tree.

Processes:

On the first level, there are the processes that communicate via the local SCP server.

Devices:

Each process is connected with diverse logical devices (MTX, logbook, drives, etc.).

Clients:

The clients connect to the logical devices.

Static items:

Below the clients are the static items.

Groups:

Below the clients, the created groups can be found.

Group items:

The group items are below the group.

Restriction for OPC Clients

It is not possible to uniquely identify all clients communicating with the SCP server via the OPC interface. Each OPC client anonymously logs in on the server and can only be identified as pair by a reference on the root and the SCP server. Therefore, unfortunately, the identification of the client's name or the process ID (PID) is not possible for OPC clients. However, OPC clients could be easily identified by means of the items used - for example, WinStudio uses unique tag definitions with a defined syntax and can thus be recognized.

9.4.4 Interpreting Values

Column "Update Rate [ms]":

This column shows the maximum nominal update rate achievable of the group item. Usually, this corresponds to the group update rate in which the item is running. For example, if the item runs in a group with 100 ms, this is the maximum update rate achievable of the item.

Column "Server Cache [ms]"

This column shows the access rate of the item, i.e. the rate with which the SCP server monitors or queries the changes of the item on the logical device. The closer the value comes to the nominal value, the larger the bar (100% = nominal).

If the SCP server cannot access the logical device for any reason, "--" is displayed.

Column "Client Update [ms]"

This column shows the update rate of the item in case of a change, i.e. the rate with which the SCP server transmits the changes of the item to the client.

If the item does not change, "--" is displayed.

9.5 NC Kernel: Diagnostic Monitor

9.5.1 General

The NC kernel diagnostic monitor displays the CPU load of the individual kernel task. Thus, it can be efficiently identified if there is a system overload due to an incorrect user configuration or programming.

9.5.2 NC Kernel: Load Distribution

Numerous tasks are processed in the NC kernel. The NC diagnostic monitor categorizes all tasks:

- Interrupt runtime (IPO + SERCOS): The user can change this load by setting the interpolator cycle (parameter SysCycTime, 9030 00001).
- PLC runtime: The load of all PLC tasks is summarized (user PLC tasks as well as internal PLC tasks). This load can be changed via the setting of the PLC cycle time.
- SAV & CPL runtime: Load due to NC and CPL block processing tasks.
- Idle runtime: Load for the idle task. This visualizes the free capacity of the control.
- Remaining runtime: The load of the tasks not included into the categories mentioned above are visualized. This includes communication tasks, file server tasks, database access tasks...

☞ The NC kernel diagnostic monitor is only available for the control family of the IndraMotion MTX performance (hardware CMP 60).

☞ The NC kernel diagnostic monitor can negatively influence the runtime behavior of the operating machine. Thus, the diagnostic tool should be switched off after the commissioning.

10 MTX Navigator

10.1 General

10.1.1 Characteristics of the MTX Navigator

The MTX Navigator is a directory navigator and file navigator. It visualizes the directory and file structure of the NC control with reference to the Windows Explorer.

This navigator is provided in IndraWorks Operation as well as in IndraWorks Engineering to navigate in the NC file system.

Tree + list

In the "Tree+List" characteristic, the directory structure is shown as tree on the left. The subdirectories and files are listed on the right.

This is used in IndraWorks Operation as well as in IndraWorks Engineering.

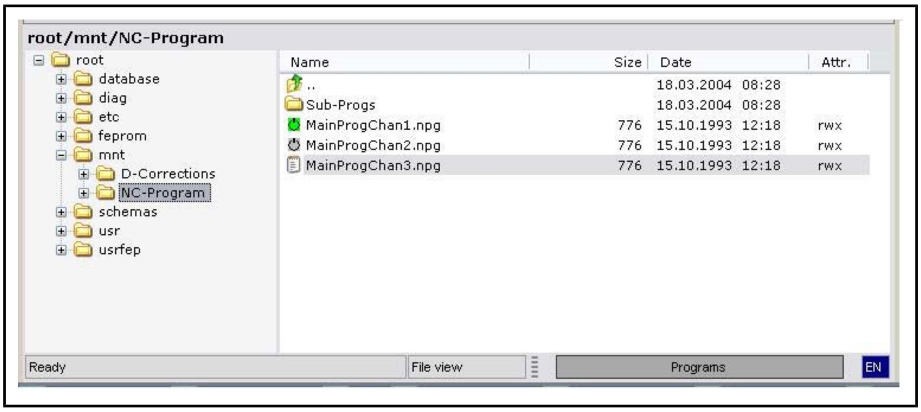


Fig. 10-1: MTX Navigator in the characteristic "Tree + List" of IndraWorks Operation

List

The subdirectories and the files are listed in the "List". This representation is only used in the context "Program selection" (chapter 10.1.2 "Context-Dependent Functions of the MTX Navigators" on page 250) in IndraWorks Operation.

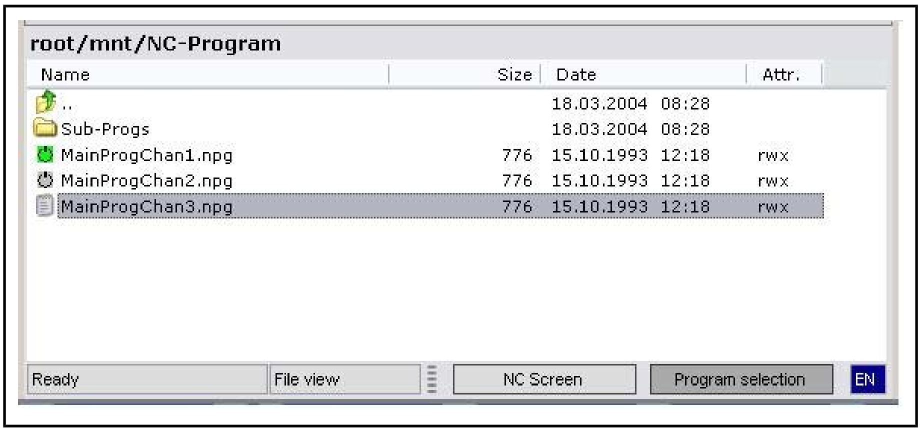


Fig. 10-2: MTX Navigator in the characteristic "List" from IndraWorks Operation

Object types

Basically every object type named in the following tree as well as in the list of the MTX Navigator can have its individual context menu and individual assignments of the function keys and the menu bar. This causes a change in the assignment of the function toolbar and the characteristics of the menu bar or the context menu, when changing between tree and list or selecting another object type.

Distinguish between the following object types:

- the folder "root" in the tree  (Root = the root directory)
- all other folders in the tree
- the return flag in the list 
- all other folders in the list
- all files of the list
- the list itself

Status bar

In the status bar, the currently selected object (directory or file) is visualized.

If the focus is on a directory in the tree,

- the number of the objects (directories and files) and
- the sum of the sizes of all files

which are directly below the selected directory are shown.

The status bar in IndraWorks Operation is - as shown in fig. 10-1 " MTX Navigator in the characteristic "Tree + List" of IndraWorks Operation" on page 249 - located directly below the navigator. In IndraWorks Engineering, the general status bar of IndraWorks Engineering is used.

☞ In IndraWorks Engineering, the status display is deleted automatically after 10 seconds.

10.1.2 Context-Dependent Functions of the MTX Navigators

The MTX Navigator in IndraWorks Operation is used in two different operating areas:

- In the operating area "Program"
- In the operating area "Machine"

Additionally, the navigator is also used in IndraWorks Engineering.

According to the context, the navigator provides different functions.

Context program

The instance of this context is always activated as soon as the operating area "Program" is called in IndraWorks Operation. In this case, the navigator is the basic application for this operating area.

In this context, the navigator provides all functions for the file editing.

The characteristic of the navigator is always "Tree+List" and cannot be switched.

Context program selection

The instance of this context is called and started with the operation mode "Automatic" via the "F2 Program Selection" function in the operating area "Machine" of IndraWorks Operation. This function is not possible in other operation modes.

In this context, the MTX Navigator provides apart from the general functions (see chapter 10.3.1 "General Functions" on page 252), the functionalities for the program selection.

By default, the MTX Navigator is started in this context with the characteristic "List". However, the user can switch between both characteristics "Tree+List" and "List".

Context Engineering

The instance of this context is called in IndraWorks Engineering via the "NC File System" node below the "Motion" node.

In this context, the navigator provides all functions for the file editing.

The characteristic of the navigator is always "Tree+List" and cannot be switched.

10.2 Operation

10.2.1 General

This section describes the different operating possibilities of the MTX Navigator.

The operation in different contexts proceeds on the assumption of different entering media. The interface is partially operated by mouse (operating areas "Program" and in IndraWorks Engineering) and partially by the keyboard (operating area "Machine"). However, an operation with the keyboard is always possible. However, in certain areas, the operation with the mouse is easier.

Context menu For entries with an own context menu, this can be shown by clicking on the right mouse button or by pressing <Shift>+<F10>.

Within the context menu, the user can highlight an entry with the mouse, the cursor buttons or by entering the underlined or the first letter and can execute it via <Enter> or with the mouse. Via <ESC> or by clicking on the mouse outside of the context menu, this can be closed again without executing any action.

Menu bar The menu bar can contain individual entries with regard to the currently focused object type of the MTX Navigator. The entries can be selected and activated by the mouse or after pressing <F10> or <Alt>+<Letter> just as for context menus.

Function bar The function bar is only available in IndraWorks Operation. It is located at the lower margin of the user interface and consists of buttons which represent the function keys <F2> to <F9>. These buttons can be triggered by clicking on the left mouse button or by pressing the corresponding function key. The assignment of the individual buttons in the MTX Navigator depends on the currently focused object type.

10.2.2 Switching Between Tree and List

Switching the entry focus between tree and list is either done by pressing <Tab> or by selecting the desired object with the mouse. It is important, since keyboard entries always refer to the currently focused entry. To navigate in the list and if the focus is currently on the tree, it is sufficient to press <Tab> to focus in the list. The tree entry is then focused again by pressing <Tab> again.

10.2.3 Navigation in the Tree

The navigation in the tree can be executed via the keyboard and by means of the mouse. Individual folders can be directly clicked on for highlighting with the mouse. A click on the plus or minus sign at the left side of a folder or a double-click on a folder itself opens or closes the latter.

The cursor keys are meant for navigation with the keyboard, whereas the folder currently highlighted can be opened with <CursorRight> and closed with <CursorLeft>.

Folders are placed one position upwards or downwards by the cursor keys <CursorUp> and <CursorDown>.

Additionally, it can be jumped directly to the folders by entering the first sign of the node name. Ambiguities result in a jump to the next appropriate folder.

With the keys <PgUp>, <PgDn>, <Pos1> and <End>, jump one screen page upwards or downwards or to the first or last folder in the tree.

When pressing <Backspace>, the highlighting moves to the respective superordinated folder in case the latter exists.

10.2.4 Navigation in the List

In the list, entries (folder, files or return label) are selected via the keys <CursorUp> and <CursorDown> or by the mouse. With the keys <CursorLeft> and <CursorRight>, the list section is shifted horizontally. Pressing the entry key or double-clicking on the mouse causes the predefined command of the currently highlighted entry:

- Entry = Return label ⇒ a change to the next higher level takes place.
- Entry = Folder⇒ it is placed and shown in the respective directory.
- Entry = File ⇒ the function assigned to the file type (e.g. open editor) is carried out.

Via <Backspace>, change to the next higher level as shown in the tree view.

By pressing <Ctrl>+<A>, all entries of the list can be highlighted (multiple selection). By means of the keys <Shift> and <Ctrl> in connection with the key <Space> areas of entries or several individual entries can be marked/unmarked.

With the keys <PgUp>, <PgDn>, <Pos1> and <End>, jump one screen page upwards or downwards or to the first or last entry in the list.

The columns of the list can be maximized or minimized by the mouse. In the extreme case, a column can be "hidden" by completely moving together the column with the mouse. In order to reopen these hidden columns later on, the mouse has to be moved from the right side to the respective column separator and the column has to be zoomed out to the right while pressing the left mouse button.

The changing of the sorting column and direction is only possible with the mouse by clicking on the respective column head for one or several times. A little triangle highlights the current sorting column and indicates the selected sorting direction whereas the apex of the triangle shows in the direction of the ascending values.

In spite of sorting, the order of the following groupings is always adhered to:

- The return label is always at the top.
- Afterwards, all directories are shown.
- The last group is the group of the files.

The MTX Navigator saves all column widths as well as the sorting column and the sorting direction. These data is always saved on the hard disk when the navigator is left and exited.

10.3 Functions

10.3.1 General Functions

General functions do not depend on the context!

Place file filter

With the filter dialog , certain data types can be excluded from the display in the list and thus the selection of the files can be limited.

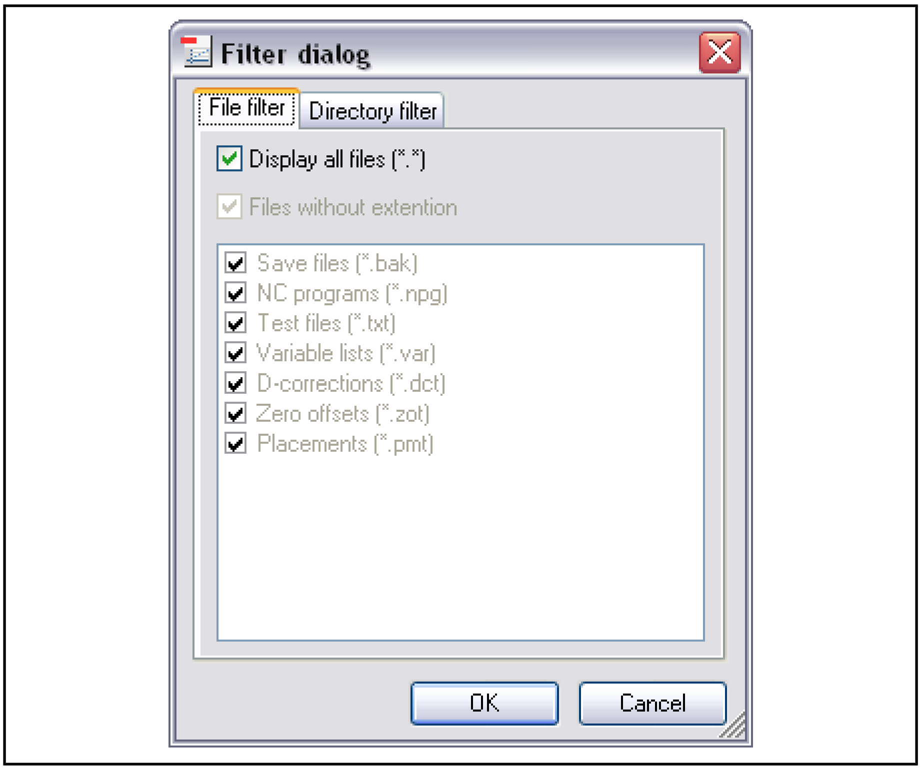


Fig. 10-3: Filter dialog for files

Only files which correspond to the file types selected in the dialog are shown. All other files are excluded from being shown. In the example above, only NC programs are displayed.

Add file filter

The list in the filter dialog for files can be extended by the function "Add filter". This function is called via the context menu (right mouse button) of the list.

Delete file filter

It is possible to delete entries from the list in the file filter dialog . Note that only the elements added by the user can be deleted. The extension specified by the installation cannot be deleted.

This function is called via the context menu (right mouse button) of the file filter list.

Set editor

This function enables to set a user-defined editor for a certain file extension which is to be used when opening a file with this extension. By default, an intern ASCII editor is used unless an editor has been assigned externally.

This function is called via the context menu (right mouse button) of the file filter list.

This function is only available in IndraWorks Engineering.

Set Directory Filter

With the filter dialog , certain default directories of the control can be excluded from the display in the tree and thus the selection of the directories can be limited.

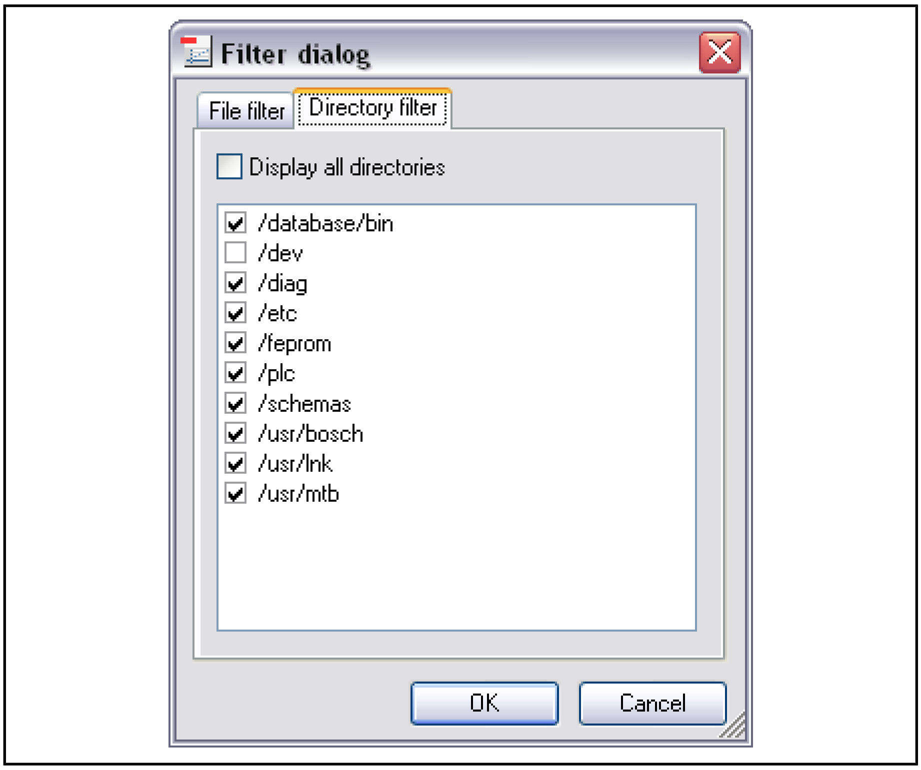


Fig.10-4: Filter dialog for directories

All connected directories and the directories selected in the directory dialog (including mount directories) are always shown.

Add directory filter

The list of the directories in the default status can be extended by the function "Add directory filter" In this case, it is possible to include subdirectories in the list as well. Furthermore, individual directories can be deleted from the list via the context menu.

Properties of the Directory

The "Properties" dialog shows the following directory information:

- Directory name → this can be changed
- Location of the directory
- File system of the directory
- Assigned memory of the file system
- Free memory of the file system
- Memory capacity of the file system
- Creation date of the directory
- The attributes of the directory

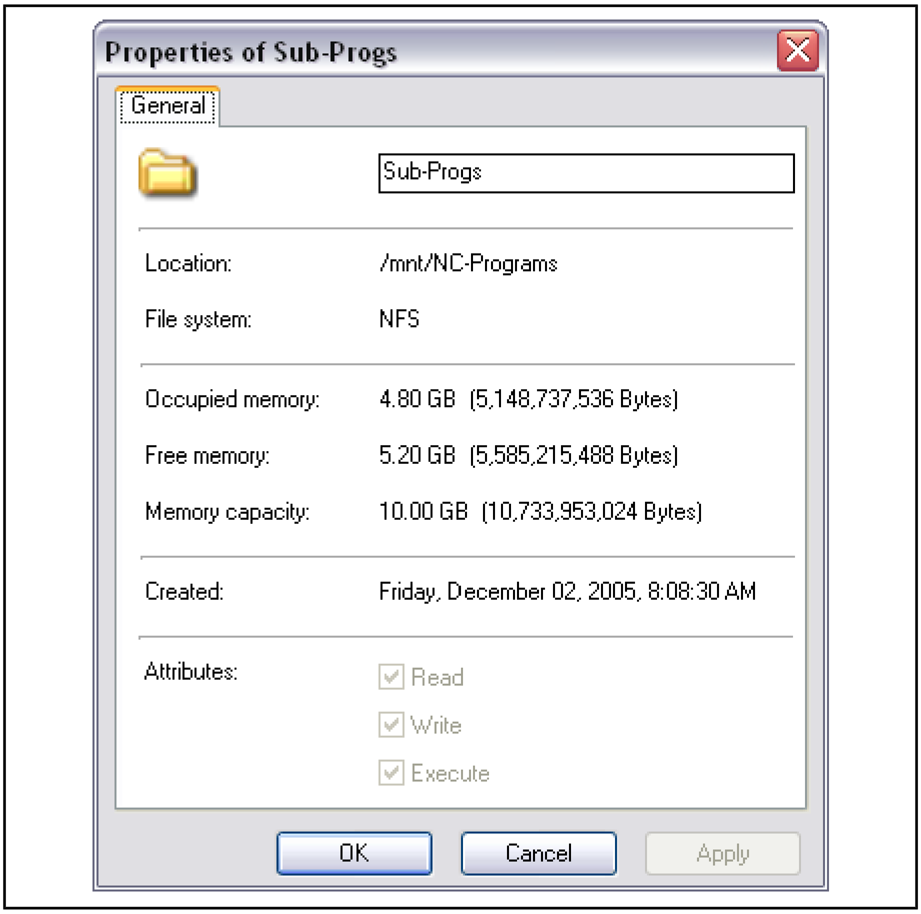


Fig. 10-5: "Properties" dialog of the directory

☞ On the mount directory, the attributes of the directory cannot be changed!

Properties of a File

The "Properties" dialog shows the following file information:

- File name → this can be changed
- Location of the file
- File system of the file
- Size of the file
- Creation date, change date and date of the last access
- File attributes

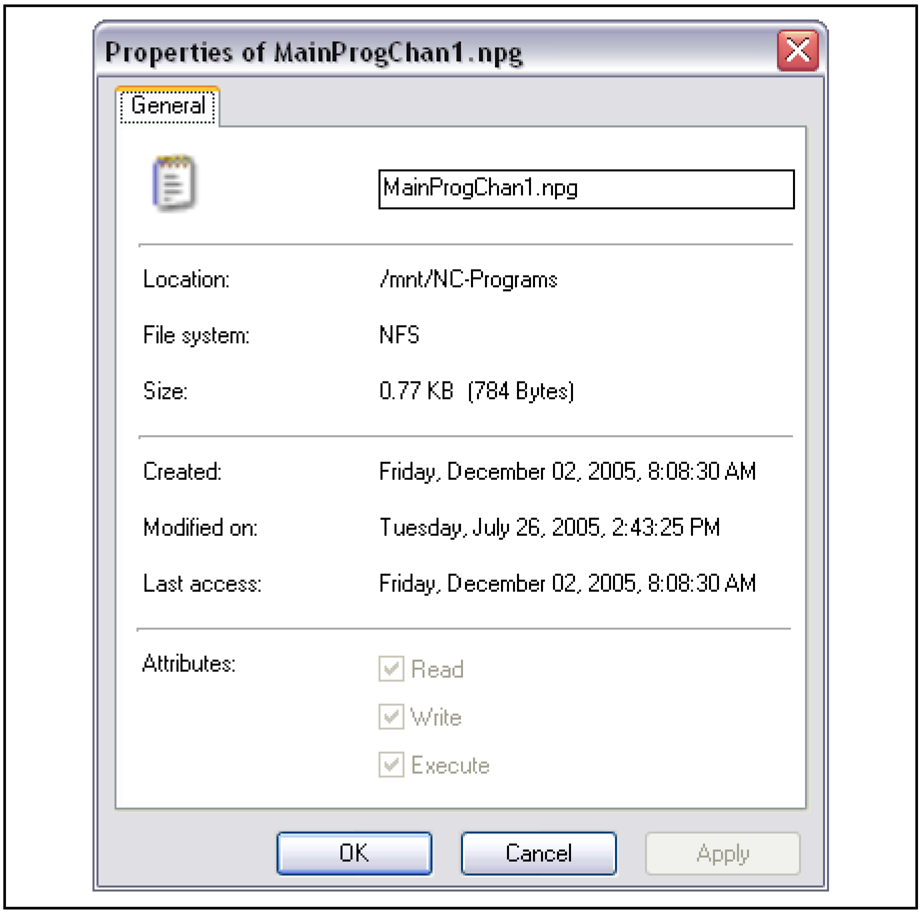


Fig.10-6: "Properties" dialog of the file

☞ On the mount directory, the attributes of a file cannot be changed!

Import Directories/Files

The user can use the "Import" dialog to import directories and/or files from the Windows file system into the file system of the NC kernel. The import function is only activated if the focus is on a directory (also the "root" directory), no matter if it is in the tree or in the list. If several objects have been selected in the list, the import function is deactivated. If the focus is on the "empty" list, the import function is possible.

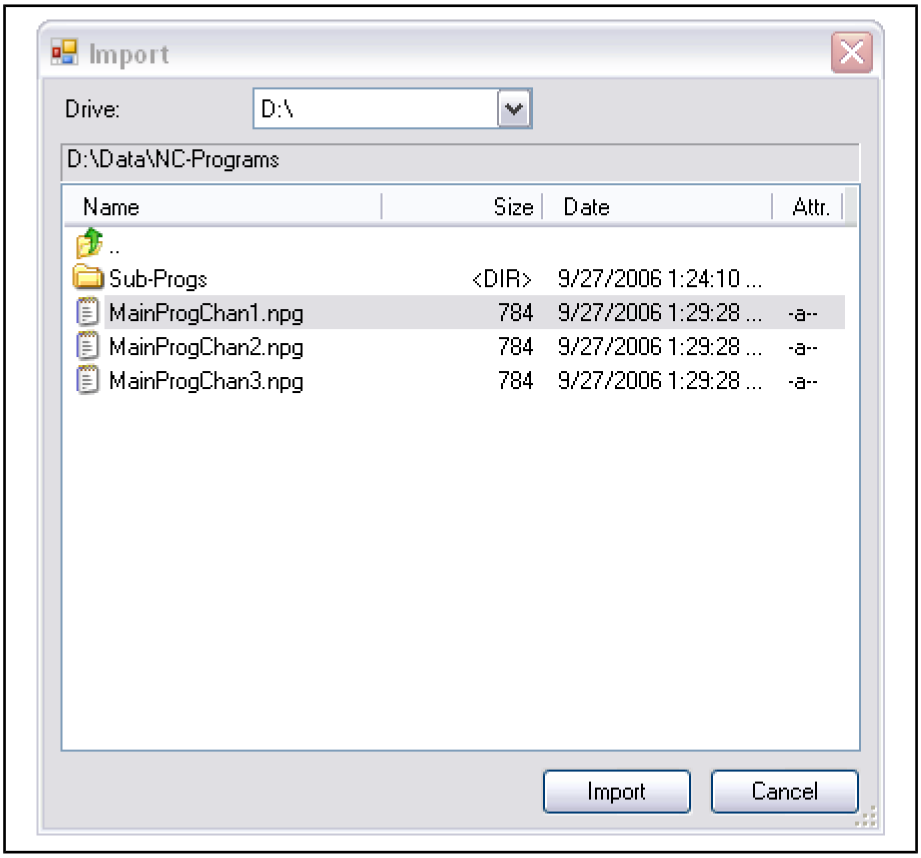


Fig. 10-7: "Import" dialog

- During the first call after the MTX Navigator start, "C:" is set as default drive. is placed. The last selected directory is placed as default directory for every following call (as long as the user interface has not been closed)
- Several files and/or directories can be highlighted and imported. After pressing the "Import" button, all selected objects (files/directories and subdirectories and files) are copied in the directory selected in the MTX Navigator.
- If a file with the same name already exists in the target directory, the user is asked whether he would like to overwrite this file or to cancel the copying process for this file.
- The dialog can be operated with the keyboard (i.e. also without a mouse)!

Export Directories/Files

With the export function, directories and/or files are copied from the file system of the NC kernel in a Windows directory selected by the user. The export function is active under the following conditions:

- Not the node "root" is selected in the tree, but another directory.
- In the list, one or several file(s) / directory/ies is/are selected.

Fig. 10-8: "Export" dialog

In this dialog, the user selects the target directory in which the selected files/directories are copied from the MTX Navigator.

If the element (file/directory) to be exported already exists in the target directory, it is overwritten without asking the user.

Furthermore, the dialog provides the possibility to create a new subfolder which can immediately be selected as target directory.

10.3.2 Program Selection Functions

Link NC Program If this function is selected in the function bar, the selected NC program is automatically linked in case of a program selection.

Show tree With this function, the view of the MTX Navigator is switched between Tree +List.

Cancel selection This function closes the program selection without a selected NC program.

Confirm selection This function executes a program selection with the selected NC program. The selected NC program is visualized in the NC block display (operating screen machine).

☞ All selected NC programs are marked with an individual symbol in the list of the MTX Navigator.

- Selected NC programs in the current channel contain a "green" symbol. 
- Selected NC programs in non-selected channels contain a "gray" symbol. 

10.3.3 Directory Functions

These are functions which can only be used for directories (folders). These functions can only be called if a folder object is focused in the tree or in the list.

☞ The directory functions are only active in the context "Program" or in IndraWorks Engineering.

New -> Directory

With this function, a new subdirectory is created in the currently selected directory. After this function has been activated, a dialog is opened in which the name of the new subdirectory has to be entered. The maximum length of a directory name within the directory system of the NC kernel is limited to 30 characters. On the mount directory, the maximum length for a directory corresponds to the Windows conventions.

Fig.10-9: Dialog: Create new directory

- With "Ok", the dialog is closed and the directory is created.
- With "Cancel", the dialog is closed and no new directory is created.
- If no name is entered and "Ok" is pressed, an error message is output.
- If the name already exists in the directory, an error message is shown.
- If the directory name has more than 30 characters and the file is in the directory system of the NC kernel, an error message is shown.

Delete

The current directory and all subdirectories and files contained therein are deleted. The deletion has to be confirmed by the user:

Yes The directory is deleted

No Deletion is canceled

It is possible to mark several directories and/or files in the list (multiple selection) and delete them afterwards.

During deletion, a status box is shown which visualizes the progress at deletion.

☞ The root directory cannot be deleted!

Rename

With this function, the currently selected directory can be renamed. After this function has been activated, a dialog is opened in which the user has to enter the name of the new directory. The default assignment in the input field is the current directory name. The maximum length of a name in the directory system of the NC kernel is limited to 30 characters. On the mount directory, the maximum length for a directory corresponds to the Windows conventions.

Fig.10-10: Dialog: Rename directory

- With "Ok", the dialog is closed and the directory is renamed.
- With "Cancel", the dialog is closed - the directory is not renamed.
- If no name is entered and "Ok" is pressed, an error message is output.

- If the name entered already exists in the directory, an error message is shown.
- If the directory name has more than 30 characters and this directory is in the directory system of the NC kernel, an error message is shown.

> ☞ The root directory cannot be renamed!
> In case of a multiple selection, directories cannot be renamed.

Copy With this function, the currently selected directory is saved in the clipboard. This function is required for the "Insert (Directory)" function. Only if an element of the "Directory" type is in the clipboard, the function "Insert (Directory)" is activated and can be selected by the user.

Every new copying causes the current content of the clipboard to be overwritten!

> ☞
> - The "root" directory cannot be copied!
> - The return label cannot be copied!
> - A multiple selection of directories in the list is possible.

Insert This function requires an element of the type "Directory" in the clipboard. Only in this case, this function is active and can be executed by the user.

As soon as the function has been selected, the program determines the source directory from the clipboard and copies the latter with all its subdirectories and files in the currently selected target directory.

If a subdirectory with the same name of the directory to be copied already exists in the target directory, the user has to decide by means of a dialog whether the current directory is to be overwritten or whether the insertion is to be canceled.

When inserting a directory with more than 30 characters from the Windows file system into the file system of the NC kernel, an error message occurs.

Cut Basic function like "Copy"!

In addition, the source directory is deleted after it has been inserted in another directory via the function "Insert (Directory)".

10.3.4 File Functions

These are functions which can only be used for files. This function is only possible if one or several files are selected in the list. In case of multiple selection, some functions are not possible!

> ☞ File functions are only possible in the context "Program" or in IndraWorks Engineering.

New This allows to create a new file of a certain type.

The following types are provided:

- NC programs
- NC variable list
- D-corrections
- Zero offsets
- Placements
- Text file

After selection of one of these types, the corresponding editor is started automatically. It is used to create and edit the file.

Delete This function deletes the currently selected file. The deletion has to be confirmed by the user:

Yes	The file is deleted.
No	Deletion is canceled.

☞ In case of deleting, a multiple selection is possible. In this case, all selected files are deleted!

Selected, i.e. active NC programs cannot be deleted.

Edit When operating this function, the data type of the selected file calls the corresponding editor and loads the selected file.

If the data type of the selected files is not known, the NC program editor is started in IndraWorks Operation and a simple ASCII editor is started in IndraWorks Engineering. This editor is part of the MTX Navigator.

☞ In case of a multiple selection, files cannot be edited.

Rename This function renames the currently selected file. After this function has been activated, a dialog is opened in which the user has to enter the new file name. The default assignment in the input field is the current file name. The maximum length of a name is limited to 30 characters. On the mount directory, the maximum length for a file corresponds to the Windows conventions.

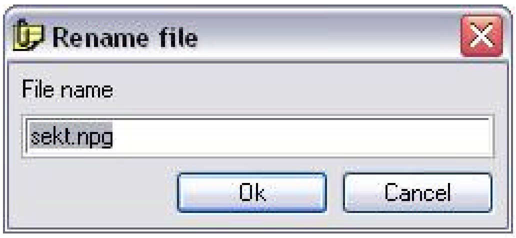


Fig.10-11: dialog "Rename file"

- With "Ok", the dialog is closed and the file is renamed.
- By pressing "Cancel" the dialog is closed but the file is not renamed.
- If no name is entered and "Ok" is pressed, an error message is output.
- If the name entered already exists in the directory, an error message is shown.
- If the file name has more than 30 characters and this file is in the directory system of the NC kernel, an error message is shown.

☞
- In case of a multiple file selection, the "Rename" function is disabled.
- Selected, i.e. active NC programs may not be renamed.

Copy With this function, the currently selected file is saved in the clipboard. This function is required for the function "Insert (File)". Only if an element of the type "File" is in the clipboard, the function "Insert (File)" is activated and can be selected by the user.

Every new copying causes the current content of the clipboard to be overwritten!

- A multiple selection of files in the list is possible.
- A mixed multiple selection of directories and files in the list is also possible.

Insert This function requires an element of the type "File" in the clipboard. Only in this case, this function is active and can be executed by the user.

As soon as the function has been selected, the program determines the source file from the clipboard and copies the latter into the currently selected target directory.

If a file with the same name of the file to be copied already exists in the target directory, the file name is prefixed by the text "Copy of", in case of the source directory and the target directory being identical. If both directories are different, the user is asked whether the file is to be overwritten in the target directory.

Cut Basic function like "Copy"!

In addition, the source file is deleted after it has been inserted in another directory via the function "Insert (File)"!

10.3.5 Search Functions

Find The "Search" function can be used to search for files in the file system of the NC kernel and in all mounted directories of the Windows file system or on a Compact Flash (CF) card of a controller-based control.

This function is only activated if the focus is on a directory (folder) of the tree (also the "root" directory) or the list. If several objects have been selected in the list, the search function is deactivated.

The search function can be used in IndraWorks Engineering and in both IndraWorks Operation contexts.

If the "Search" function is called in a directory (folder), the following dialog appears:

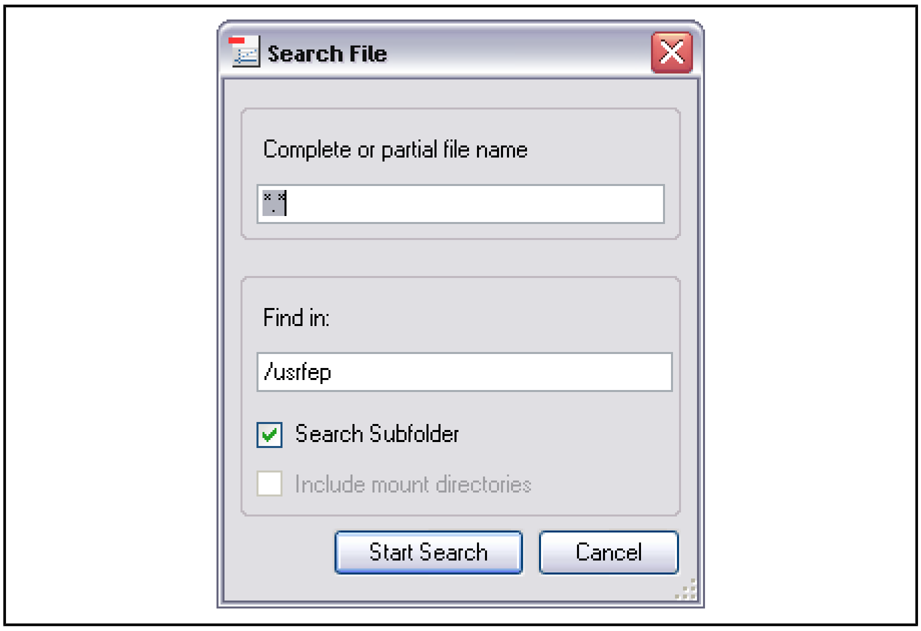


Fig. 10-12: "Search File" dialog

To search for certain files, the following settings can be made:

- Entry of a file name combined with a placeholder "*"

- ***.*** - Search for all files (default setting)
- **<FileName>.<ext>** - Search for a certain file
- ***.<ext>** - Search for all files with a certain extension
- **<FileName>.*** - Search for files with a certain name and any extension
- **<FileName>*.*** - Search for files starting with certain characters and any extension
- **<FileName>*.<ext>** - Search for files starting with certain characters and with certain extension

- Entering a path

  In the "Find in" field, the path of the focused directory is applied when opening the dialog. This search path can be manually adjusted by the user. It has to be observed that a valid (existing) path is entered. If the path is invalid, searching is aborted with an error.
- Search Subfolder

  This checkbox can be used to limit the file search to the specified directory or to include all subdirectories (default setting).
- Include mount directories

  Select this checkbox only if the selected directory is root ("/"), since only "root" can be provided with one or several mount directories and subdirectories.

If the search is started, all files with the set specifications are searched. While searching, a progress bar is displayed. At the end of the search, the node "Search Results" is newly created and automatically selected in the tree. Are files found in the list are displayed with their complete path. These are "links" on the original files.

Functions of the "Search Results" node

Generally, the search result remains until a new search is started and explicitly completed by the user. The advantage is that it can be switched as often as desired between the individual directories of the NC file system and the search result without loosing the search result as it is common for other browsers. The disadvantage is that updates such as modifications or deletion of original files made after the search cannot be seen in the search result list.

Repeat search This function starts the previous search again without displaying the search dialog. This function is especially useful if the original files have been modified after the search but the search result should be up-to-date.

Complete search This function deletes the search result list (list of links on the files found) and the "Search Results" node.

Functions of the elements (links) in the search result list

Go to This function switches the directory to the original file and focuses it. The "Search Results" node remains.

Available file functions The number of available file functions on the elements of the search result list is limited to the following functionalities:

- Delete - To delete the original file as well as the link from the search list.
- Export - To export the original file.
- Properties - To visualize the properties of the original file.

All other file functions are not possible for elements of the search result list.

11 Exporting and Importing Project Tree Elements

11.1 Exporting MTX Nodes

11.1.1 Starting the Process

In the context menu of the MTX node, select **Export...**

11.1.2 User Input

The storage location and the name of the export file are specified in the Export Wizard.

No further subelements can be selected for the MTX. Instead, there are several options to specify what the export should include.

- Archive of the drive data
- Archive of the control data
- Extended archives
- PLC project data

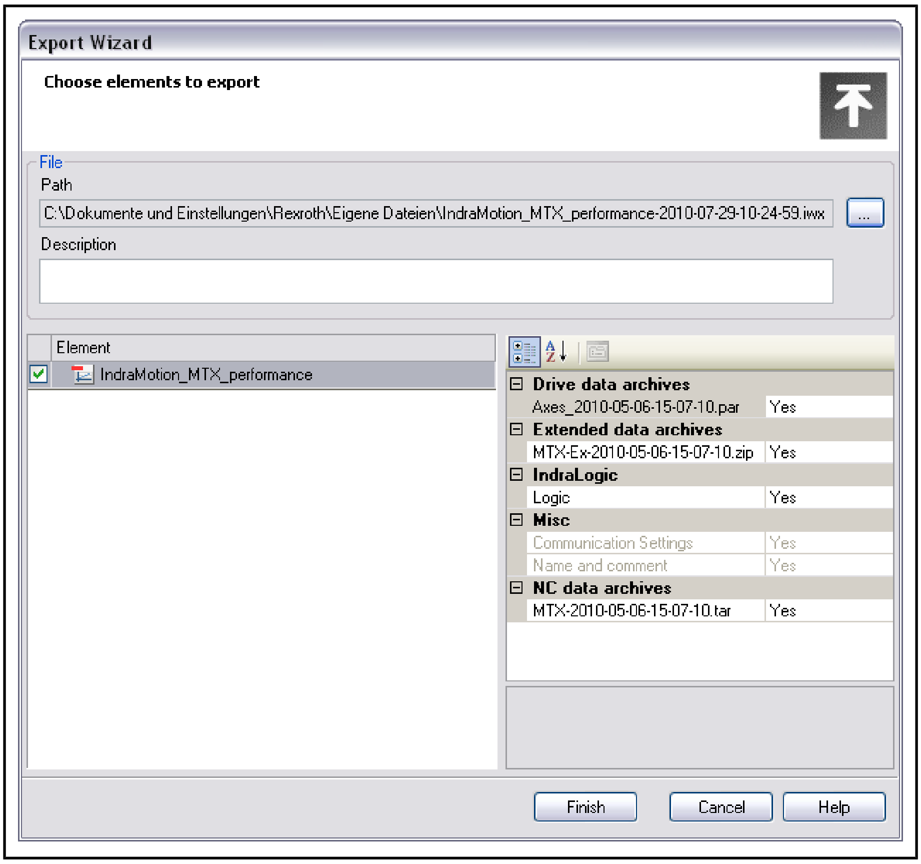


Fig. 11-1: Dialog: Export wizard (exporting an IndraMotion MTX)

Click on "Finish".

11.1.3 Summary

The summary of the process as well as the status of the element are shown here.

11.2 Importing MTX Nodes

11.2.1 Starting the Process

In the context menu of the MTX node, select **Import...** and select the file to be imported.

11.2.2 User Input

All elements in the files are shown. The element to be imported can be selected.

Via **More information...** , further settings can be made.

- The archives can be included or excluded for the MTX.
- Furthermore, the IndraLogic data can be selected with options according to the import of the PLC project data in the menu of the control.

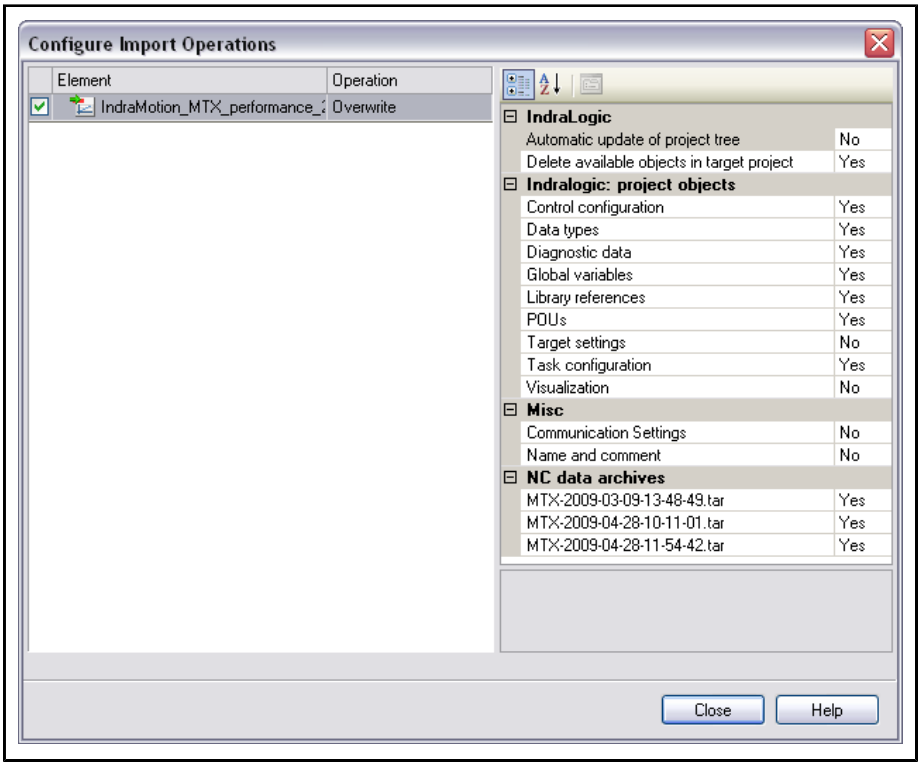


Fig.11-2: Dialog: Configuring import operations (import of an IndraMotion MTX)

Complete the dialog with "Close".

11.2.3 Summary

The summary of the process as well as the status of the element are shown here.

12 Archiving and Restoration

12.1 General

An IndraWorks project consists of several parts

- Data (HMI screen, F-keys, M-keys,...)
- MTX device data (control data, extended data, drive data)
- PLC project

Data backup includes all the data and settings generated by the machine tool manufacturer and the end user. The drive parameters (SERCOS parameters) are an exception. These have to be saved before a data backup in order to ensure that the parameters are recorded in the project tree. Then, the drive parameters are backed up with the project backup.

☞ The data backup and restoration of project data are described in detail in the documentation "IndraWorks" in the chapter "Archiving and Restoring Projects".

In contrast to that documentation, the backup of the control data (see chapter 12.4 "Control Data" on page 270) is carried out during the device data backup for the "Rexroth IndraMotion MTX"

12.2 IndraWorks Project (Complete Data Backup)

12.2.1 Handling Instructions

Handling Instruction: Data Backup of the IndraWorks Project (Complete Data Backup)

This handling instruction describes the data backup of the IndraMotion MTX control data range.

IndraWorks Engineering / Project: Archive

1. Start the archiving in Engineering Desktop under the directory in the main menu "Project".
2. Follow the instructions of the wizard.

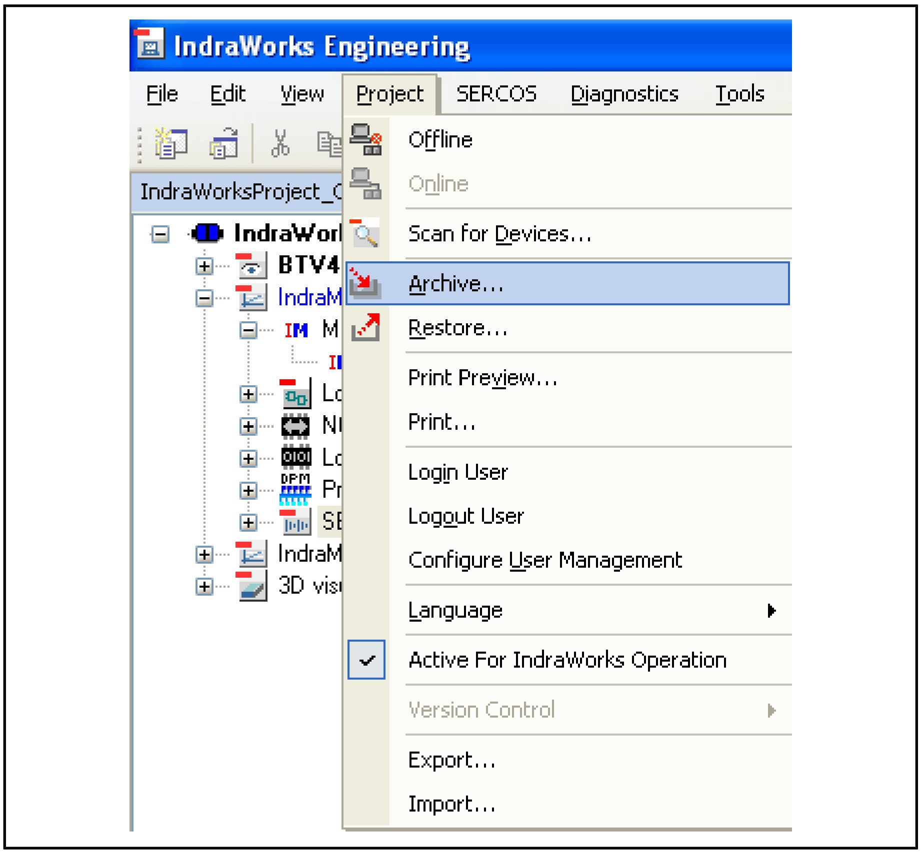


Fig.12-1: Menu: Archiving a Project

In this data backup variant, the following data is backed up:

- Device data (control data, extended data, drive data)
- PLC project
- HMI data

These areas cover the entire scope of the control data. All data is backed up here.

If an HMI project should be backed up, individual device data types can be deselected. This archive only contains the HMI components of the project.

		Documentation
Documentation:	IndraWorks Engineering	Archiving and restoring projects

Handling Instruction Restoration of IndraWorks project data

This handling instruction describes the restoration of IndraWorks project data.

IW Engineering / Project: Restore

1. Start the function "Restore" in Engineering Desktop under the directory in the main menu "Project".
2. Follow the instructions of the wizard.

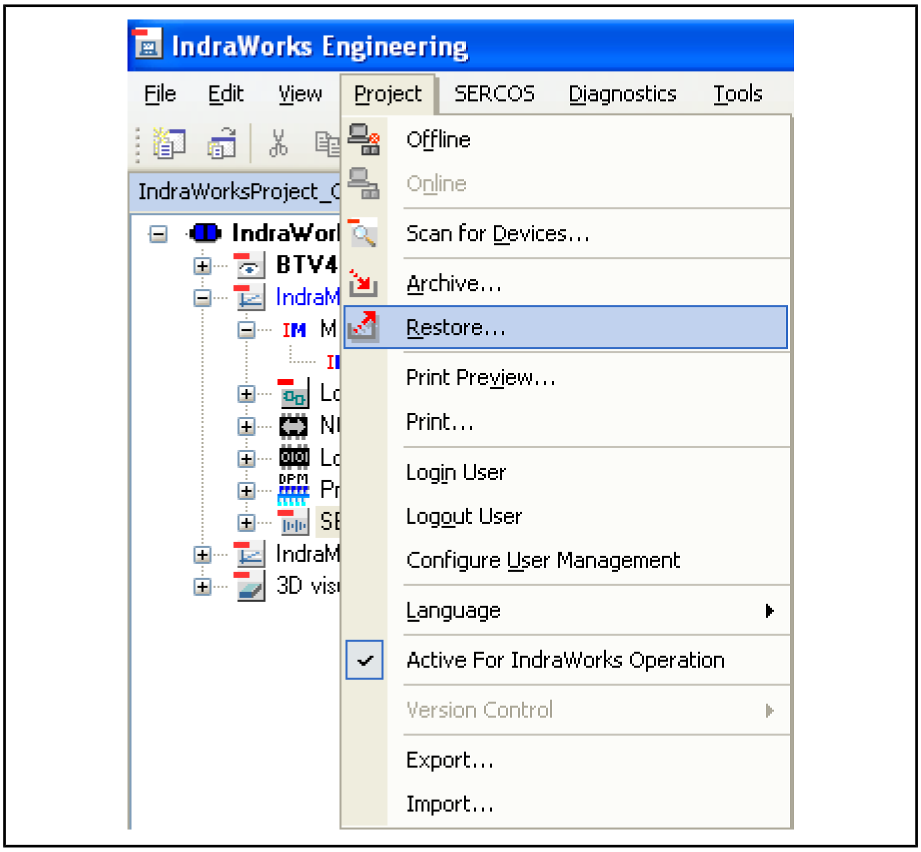


Fig. 12-2: Menu: Restoring a project

The device data must be separately restored following the instruction "Restoration Control data".

		Documentation
Documentation:	IndraWorks Engineering	Archiving and restoring projects

12.3 Device Data

The device data contains all data of a device (control) that can be archived.

- Control data
- Extended data (mount directories, user partition, user-defined directories)
- Drive data

The archiving and restoration can be started in several ways.

IndraWorks Engineering

Project Node ▸ Archive/Restore

MTX Device node ▸ Archive/Restore

A wizard opens to support in the following actions.

IndraWorks Operation

OP8 - Maintenance ▸ More functions ▸ Archive device data...

12.4 Control Data

12.4.1 General

Control data can be stored as archive with the file extension "*.tar".

12.4.2 Archiving Control Data

User Input

Archive Assign the name and storage location of the archive to be generated. Enter the name of a new archive file by clicking the button on the right of the archiving path.

☞ The name of the archive file must be in accordance with the Windows® convention and must not be longer than 29 characters, excluding the file path.

Last Archives Optionally, you can overwrite an existing archive file from the "last archives" list by selecting this archive file from the list. When an archiving path is offered, this is also to be found as the first entry in the list so that you can return to this default path at any time.

Contents Select the content to be stored in the archive. To save all parts, check all the boxes.

The following options can be selected:

- Machine Parameters
- System data
- Tool tables
- User FEPROM - file system
- RAM
- Permanent CPL variables
- Residual data of PLC

After completing the entries, press "Next".

Settings

- Switch PLC to STOP state before
- Switch PLC to START state afterwards

☞ If you select the "Non-volatile PLC data" option, you must switch the PLC to STOP to avoid inconsistent states in the archive.

Error Messages If an error has occurred for a file, this file is marked as faulty.

See the following table for error numbers and causes.

No.	Error message	Cause	Elimination
1	File not processed (yet)!	Internal	Contact service
2	Error while calling the stat() function!	Internal	See no. 1
3	File or directory not processed!	Internal	See no. 1
4	Directory could not be opened!	Internal	See no. 1
5	Error reading the file!	Internal	See no. 1
6	It is impossible to access the file!	Read-protected file	Change file protection
7	File already exists!	Internal	See no. 1
8	File not available!	Internal	See no. 1

No.	Error message	Cause	Elimination
9	Memory not available!	Memory in the control is full	Delete files which are not needed
10	Read-only file system!	Internal	See no. 1
11	Protection different!	Internal	See no. 1
12	Size different!	Internal	See no. 1
13	Date different!	Internal	See no. 1
14	File data different!	Internal	See no. 1
15	Other differences!	Internal	See no. 1
16	Date/protection not restored!	Internal	See no. 1
17	Error while creating a directory!	Internal	See no. 1
18	Unknown error!	Internal	See no. 1
19	Error applying the machine data!	Internal	See no. 1
20	Error while applying the tool table!	Internal	See no. 1
21	File destroyed!	A file to be archived was changed during the archiving process. This can be caused e.g. by a CPL program.	Do not change files during archiving; e.g. deselect all CPL programs etc.
22	Write protection suspended!	A file in the archive is write-protected. After the file has been restored in the control, this write protection does not exist any more; if required, the user must set it accordingly once more.	This is a message for the user to enable him to decide what to do about file protection.
23	Error while applying the residual PLC data!	Internal	See no. 1
24	Error while applying the residual PLC data!	Internal	See no. 1
25	Error while applying the CPL variables!	Internal	See no. 1
26	RAM file system is not available in archive!	The archive part is not in the archive.	Select another archive, or change the selection of the archive parts in the dialog.
27	User FEPROM is not available in archive!	See no. 26	See no. 26
28	Machine data is not available in archive!	See no. 26	See no. 26
29	Tool table is not available in archive!	See no. 26	See no. 26
30	Residual data of PLC are not available in archive!	See no. 26	See no. 26
31	Residual data of PLC are not available in archive!	See no. 26	See no. 26
32	Permanent CPL variable is not available in archive!	See no. 26	See no. 26
33	PLC is not in STOP condition!	Internal	See no. 1
34	Error applying the system data!	Internal	See no. 1
35	System data is not available in archive!	Internal	See no. 1

No.	Error message	Cause	Elimination
36	Data contains invalid data record K1/K2!	Backup file contains data record that is not in the tool table.	See no. 1
37	Data does not correspond to data record schema!	Data record structure from backup file does not correspond to the data record structure in the tool table.	See no. 1

Fig. 12-3: Error Messages

Summary

12.4.3 Restoring Control Data

User Input

Archive Enter the name and the storage location of the archive file that is to be restored by clicking the button to the right of the archiving path.

> **Loss of data!**
>
> During restoration, any existing files are overwritten. Save your files by archiving before executing the restoration process.

> The name of the file path may not contain more than 29 characters.

Last Archives Optionally, you can select an existing archive from the "Last archives" list.

Contents Select the content to be restored.

The following contents can be selected:

- Machine Parameters
- System data
- Tool tables
- User FEPROM - file system
- RAM
- Permanent CPL variables
- Residual data of PLC

After completing the entries, press <Next>.

Settings

- Switch PLC to STOP state before
- Switch PLC to START state afterwards

Execute NC restart The files of the control data archive become effective only after an NC restart. Tick the checkbox if an NC restart should be executed after the restoration of the archive.

Clear RAM file system The target directories can be cleared before the archive is being restored. Tick the check box if the RAM file system should be cleared. This option can only be selected if the archive contains the RAM file system (at least one 1) and if "RAM file system" was selected.

> **Files which are not in the archive will be lost!**

Clear user FEPROM - file system The target directories can be cleared before the archive is being restored. Tick the check box if the USRFEP file system should be cleared. This option can only be selected if the archive contains the FEPROM file system (at least one 1) and if "User FEPROM file system" was selected.

☞ **Files which are not in the archive will be lost!**

☞ If all checkboxes are ticked, **only** the data of the archive is stored in the control after the restoration.

If there is no PLC program in the archive that can be run, it is to be compiled again and to be loaded after the completion of the project.

After completing the entries, press **Next**.

12.5 Archiving Extended Data

12.5.1 General

Extended archiving allows any directory and file of the control computer to be created and restored as an archive with the file extension "*.zip". Thus, it is for example possible to save and restore the mount directories belonging to an MTX, the user partition of the CF medium (if available) as well as all non-control-based peripheral data.

Extended archiving can be accessed in IndraWorks Engineering via the context menu of the IndraMotion MTX device node

12.5.2 Archiving Extended Data

User Input

Archive Assign the name and storage location of the archive to be generated.

Enter the name of a new archive file by clicking the button on the right of the archiving path.

Contents All locally mounted directories of the MTX are provided.

The user partition of the CF card (if available) is also provided.

☞ The user partition can only be backed up and restored as a whole. Selecting single files is not possible.

In addition, it is possible to add and remove up to 99 user-defined folders. To do so, select the buttons "Add" or "Remove".

The following is admissible:

- Local folders

  (Example: "C:\MyDocuments\MyMTXDocuments")
- UNC network shares as \\HOST\\SHARE

  (Ex.: "\\MyServer\\MyShareDirectory")

The following is not admissible:

- Complete disk drives

  (Example: "C:\")
- Connected network disk drives

  (Example: "X:\")
- Windows® system folder

  (Example: "C:\WINDOWS\SYSTEM32")

No check for required memory space!

No check for the required memory space will be carried out! The user alone is responsible for the selection of the folders to be stored.

Restriction for folders of the same name!

If several folders of the same name and with different drive specifications are added, only the first folder will be saved.

If there are the following entries

"C:\MyDocuments\MyMTXDocuments" and

"D:\MyDocuments\MyMTXDocuments", in the list, only the folder

"C:\MyDocuments\MyMTXDocuments" is backed up.

Error Messages

If an error has occurred for a file, this file is marked as faulty.

See the following table for error numbers and causes.

No.	Error message	Cause	Elimination
1	File not processed (yet)!	The file has not been processed yet.	-
2	You do not have the required authorization!	You do not have the required authorization to carry out the requested file operation.	Grant authorization
3	Path is too long!	The file path is too long and/or does not comply with the Windows® convention.	Check path
4	Path is invalid!	The file path does not exist or does not comply with the Windows® convention.	Check path
5	File is write-protected!	The file is write-protected and cannot be restored and/or overwritten.	Cancel file protection
6	File not found!	The file does not exist or could not be restored.	Check the content of the archive

Fig. 12-4: Error Messages

12.5.3 Restoring Extended Data

User Input

Archive

Enter the name and the storage location of the archive file that is to be restored by clicking the button to the right of the archiving path.

Loss of data!

During restoration, any existing files are overwritten.

Save your files by archiving before executing the restoration process.

Before the restoration, a backup image off all target directories is created. It is tried to restore this backup if an error occurred during the process. If the user partition of the CF card is restored, the data is downloaded to the PC first.

Contents

The files in the archive can optionally be displayed as a list or as a tree. Here, select the files and folders to be restored by selecting and deselecting the checkboxes.

☞ The user partition can only be backed up and restored as a whole. Selecting single files is not possible.

Push the button **select none** for deselecting a file and **select all** for selecting all files.

If you chose the tree view, you can use the buttons **Enlarge** and **Reduce** for better navigation.

After having made the entries, press **Next** or **Finish**.

Clearing target directories

Target directories can be cleared before restoring the archive. Tick the check box if the target directories should be cleared.

☞ **Files which are not in the archive will be lost!**

12.6 Drive Data

12.6.1 General

If drive parameter sets are to be transferred from one machine to the other or if they are to be saved to a data carrier (e.g. a hard disk), several options are available for the user.

Parameter backup using:

- IndraWorks drive in the user's IndraWorks project
- Command instruction via the drive display on a multimedia card

The procedure for backing up data is described below, followed by the procedure for restoring the drive data (drive parameters) using IndraWorks Drive.

The backup/restoration of the drive parameters using a multimedia card and the commands available on the IndraDrive display are described in detail in the drive documentation "Functional Description - Loading and Saving Parameters" and will therefore not be discussed here.

In IW Drive, there are generally two ways to backup/restore the drive parameters.

Backing up and loading of a drive's drive parameters, of a selected number of drives and of all drives in one go in the current IndraWorks project. The storage directory and the file name are assigned automatically by IndraWorks Drive.

12.6.2 Handling Instructions

Handling Instruction: Backup the Drive Parameters of an Axis

The following handling instruction describes how to back up the drive parameters of an individual axis or any number of axes.

IW Engineering / SERCOS: Backup parameters

1. Switch the MTX online in Engineering Desktop in the main menu under "Project - Online".
2. Focus the project tree in the "SERCOS" node, activate the "Archiving..." dialog in the context menu (using the right mouse button!) in order to start the data backup of the drive parameters.

Archiving and Restoration

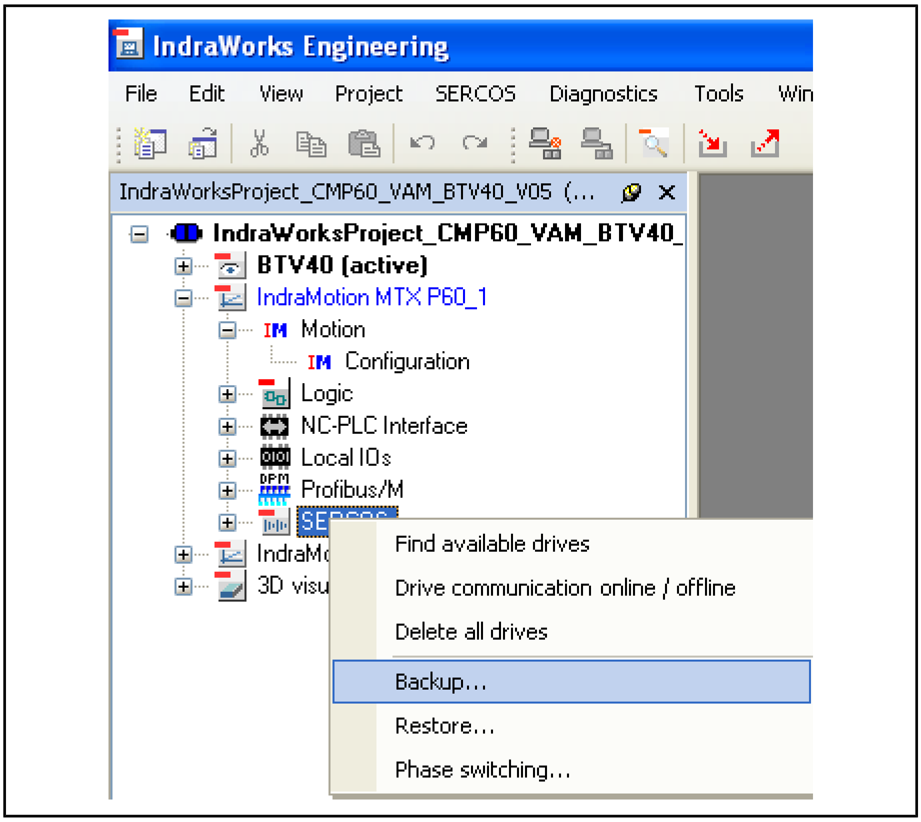


Fig. 12-5: Context menu: SERCOS (Archiving drive parameters)

3. Select the drive(s) the parameters of which are to be saved.

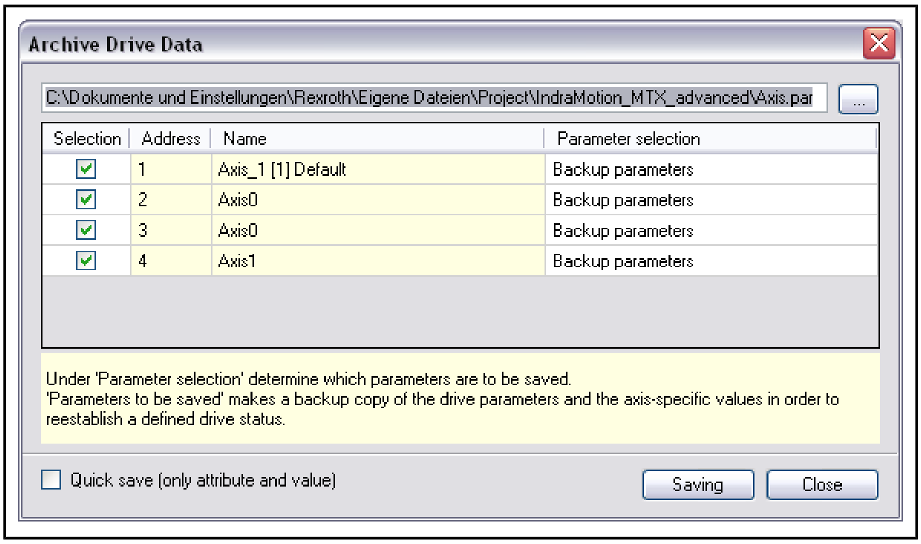


Fig. 12-6: Dialog: Archiving drive data (select drives)

After the confirmation, the selected drives are switched to online and the data is exported in a parameter file. The progress is displayed comprehensively for all axes.

Fig. 12-7: Dialog: Export parameters (progress bar)

☞ The data of the drive parameter backup files have the extension "*.par". This extension should be retained; otherwise restoring (import) of the parameter backup is no longer possible.

		Documentation chapter 5 "Axis Commissioning" on page 69
Documentation:	IndraWorks Commissioning	Commissioning axes

Handling Instruction: Restore the Drive Parameters of an Axis

The following handling instruction describes how to restore the drive parameters of an individual axis or any number of axes.

IW Engineering / SERCOS: Restore parameters

1. Switch the MTX control online in the main menu under "Project - Online" in the Engineering Desktop.
2. In the project tree under "SERCOS", enable the context menu "Restore ..." to start the restoration of the drive data.

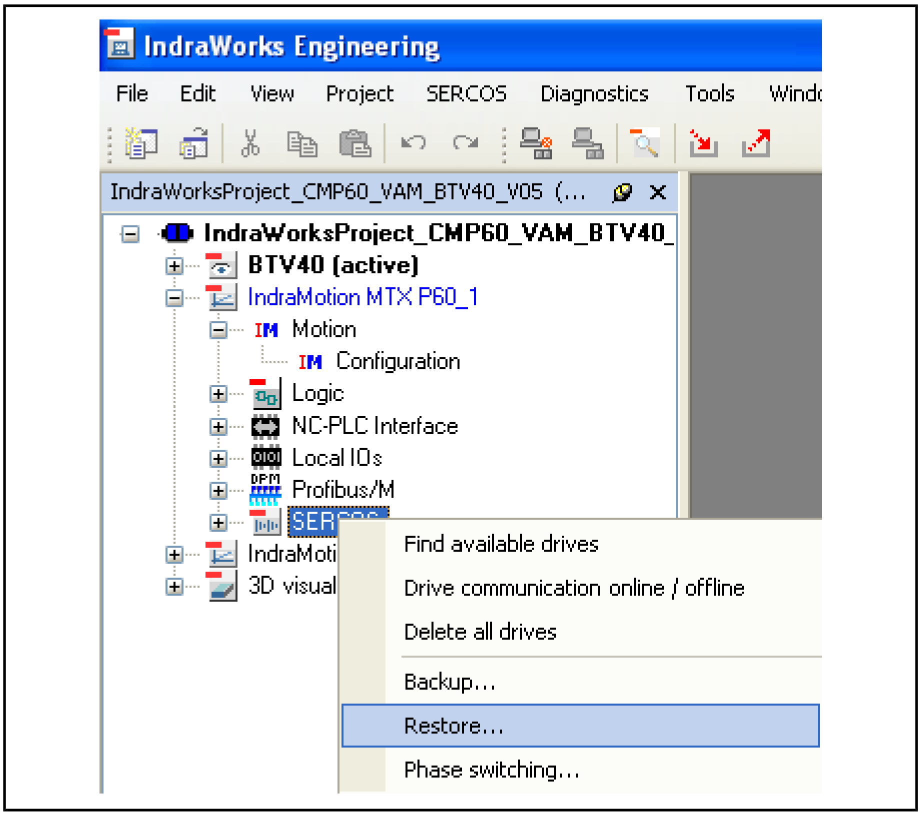


Fig.12-8: Context menu: SERCOS (restore drive data)

3. Select the drive(s) the drive parameters of which are to be restored. Additionally, the assignment of the exported parameter sets to axes can be changed.

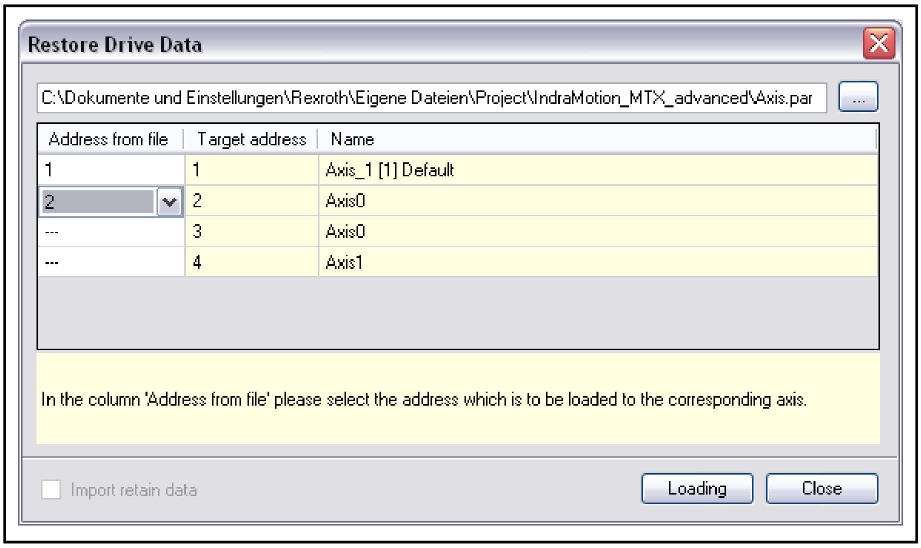


Fig. 12-9: Dialog: Restoring drive data (select drives)

4. The confirmation of the selection via **OK** starts the restoration of the drive data.

   The drives selected are switched to online and the data is imported in the drive. The progress is displayed comprehensively for all axes.

5. In order to be able to transfer the parameters to the drive, the drives must be switched to SERCOS phase 2.

   The following error message is displayed if the drives are not in SERCOS phase 2.

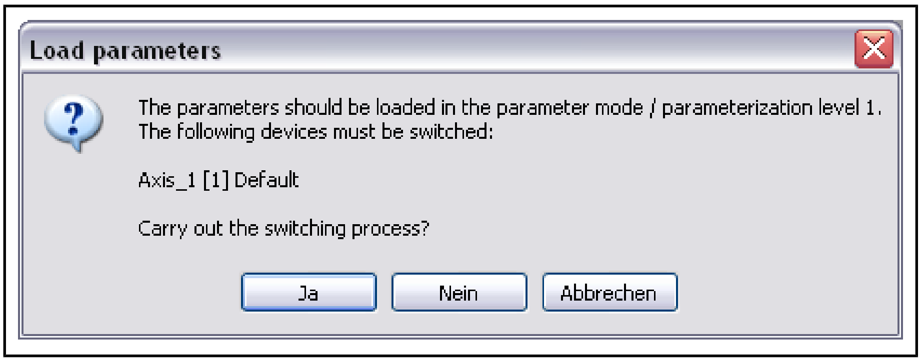


Fig. 12-10: Dialog: Load parameter (error message during the restoration of the drive parameters)

6. Phase switching can be carried out in the context menu under the "SERCOS" mode by means of the "Phase switching..." function

☞ The backup files of the drive parameters have the extension "*.par". The file type is preset to "*.par" in the Explorer window.

		Documentation chapter 5 "Axis Commissioning" on page 69
Documentation:	IndraWorks Commissioning	Commissioning axes

12.7 PLC Project

12.7.1 General

A PLC project (pro file) can be saved independently of the IndraWorks project.

The "*.pro" file contains the following data:

- Project file
- Blocks
- Data type definitions
- Resources
- ProVi diagnostics
- Link to GSD files used
- Link to libraries used

A project file can also be imported. An existing IndraWorks project can be linked to another PLC project. The following handling instructions describe the backing up and restoration of the project file.

The user-defined libraries and GSD files are backed up in the project directory. An export function for these data does not exist. The backup is carried out along with the overall data backup of the IndraWorks project.

☞ When changing the PLC programs, the PLC library "RIL_VExUtil.lib" has to be included.

12.7.2 Handling Instructions

Handling Instruction: Backing up the PLC Project File

The project file data can be backed up in Engineering Desktop under node "Logic".

IW-Engineering/IndraLogic: Save the PLC project

1. Focus node "Logic" and open the following menu using the right mouse button.

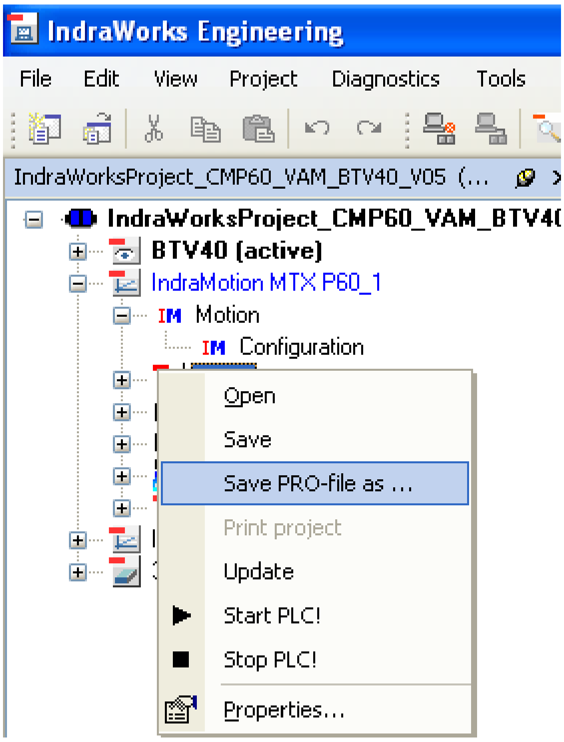


Fig. 12-11: Context menu: Logic (backing up PLC project)

2. To complete the data backup, follow the instructions in the wizard.

   Select the storage directory and start the backup.

   The preset directory (default directory) is taken into account for the general data backup. The user can retain this selection.

In order to be able to save a project file, IndraLogic has to be exited.

		Documentation
Documentation:	Rexroth IndraLogic	Saving the project file

Handling Instruction: Importing the PLC Project File

A PLC project file can be imported into an existing IndraMotion project.

IW-Engineering/IndraLogic: Importing a PLC project file

1. Focus the "IndraMotion MTX P60" node in Engineering Desktop and open the context menu (see figure) by means of the right mouse button.

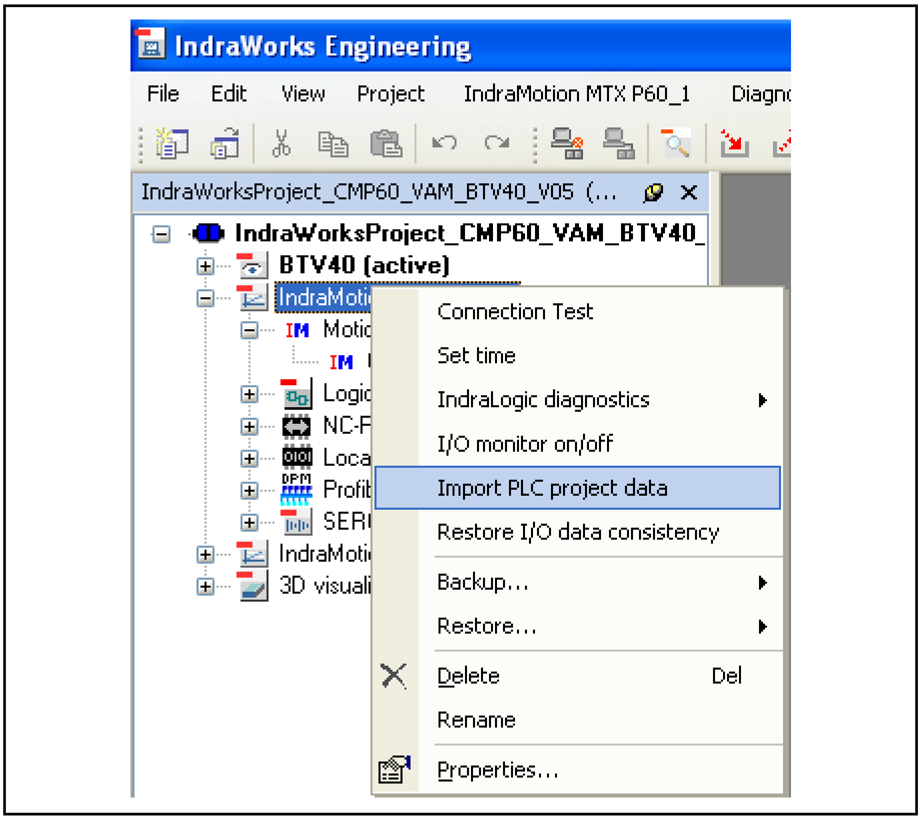


Fig. 12-12: Context menu: Project (importing PLC project)

2. Activate the function "Import PLC project data".
3. After the Import function has been started, the source file must be selected in the next step.
4. Configure the scope of the Import function.
5. Start the Import function with **Import**

   The termination of the Import function is displayed in the "Status" window.

☞ It may take several minutes to import the project file.

		Documentation
Documentation:	Rexroth IndraLogic	Import of the project file

Handling Instruction: Checking the Communication Parameters

After a PLC project file has been imported successfully, the communication parameters must be checked.

IW Engineering / IndraLogic / Properties: Communication parameters

Checking the communication parameters

☞ This handling instruction can be found under "Commissioning IndraWorks / Emulation / PLC Design".

		Documentation chapter 21.5 "Configuring the PLC" on page 502
Documentation:	IndraWorks Commissioning	Emulation / PLC design

13 Version Control of the MTX

13.1 General

IndraWorks Engineering allows working with versioned projects located on one team server. Several IndraWorks users can work simultaneously on such version-controlled projects.

The following MTX controls support the version control:

- IndraMotion MTX standard L45
- IndraMotion MTX performance L65
- IndraMotion MTX advanced L85

The documentation on the version control system in IndraWorks can be found in the manual "Rexroth IndraWorks 12VRS Engineering".

13.2 MTX Project Data

Use case The version control focuses on the PLC project planning and configuration.

These include POUs, variable lists, I/O configuration, etc.

Versioned project data The following project data is versioned

- PLC project data (POUs, variable definitions, I/O configuration, function modules, etc.)
- Drive data (offline parameterization MLD)
- FDT Container

Non-versioned project data The following data is currently not versioned.

- NC project planning and configuration (machine parameters, NC program, system data, etc.)
- HMI project planning and configuration as "MTX standard application"
- Virtual operating panel/simulation

13.3 Working with the Control

Going online If it is detected when going online that the drive configuration has been changed, the SERCOS node is changed to "Hijacked" state. Then, the drives are added or deleted according to the configuration in the project.

Communication settings The communication settings are not added to version control.

14 Remote Engineering

14.1 Introduction

An IndraWorks operating station is generally used to operate and monitor a running machine or system. Therefore, the IndraWorks operating interface is started. It gets its engineering data from an IndraWorks project stored in the operating station. This project is edited and modified using the IndraWorks Engineering interface.

"Remote Engineering" permits the engineering interface to be operated on a separate computer connected to the operating station via network.

The steps required to work in this mode are:

1. Enable "Remote Engineering" on the operating station
2. On the engineering station:
   - Connect to the operating station
   - Open and edit the project on the operating station
   - Save the project
   - Disconnect it from the operating station
3. "Remote Engineering" on the operating station can be disabled again if the engineering station has no longer access to the operating station.

14.2 Enabling and Disabling "Remote Engineering" on the Operating Station

14.2.1 Enabling

It must be possible to explicitly enable/disable the "Remote Engineering" on an IndraWorks operating station.

This carries out the application "Remote Engineering Configuration", which can be reached via the "Start" menu:

Fig. 14-1: Context menu: Remote Engineering Configuration (start)

The dialog of the "Remote Engineering Configuration" application shows the current enabling state after the start (here: not enabled).

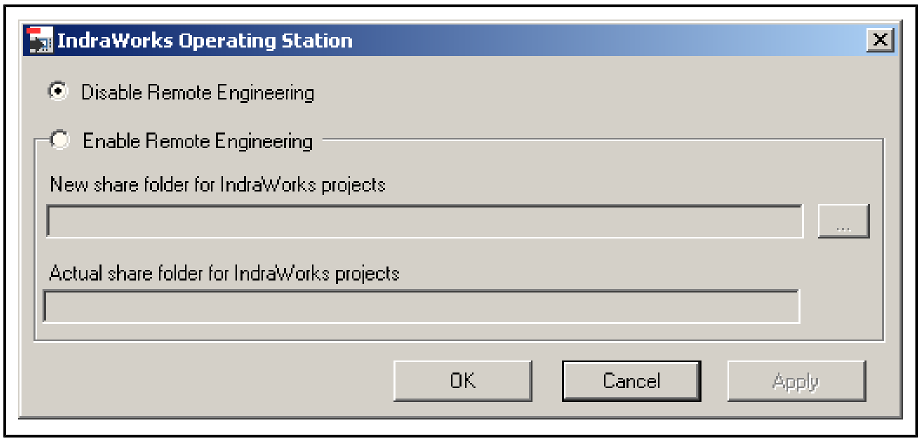


Fig. 14-2: Dialog: IndraWorks Operating Station (Remote Engineering configuration)

The "OK" and "Cancel" buttons close the application without saving the changes, i.e. the Remote Engineering remains disabled.

If you select the "Enable Remote Engineering" option button, you can specify the share directory for IndraWorks projects via an entry field or a folder selection dialog. Engineering stations can access IndraWorks projects in this directory.

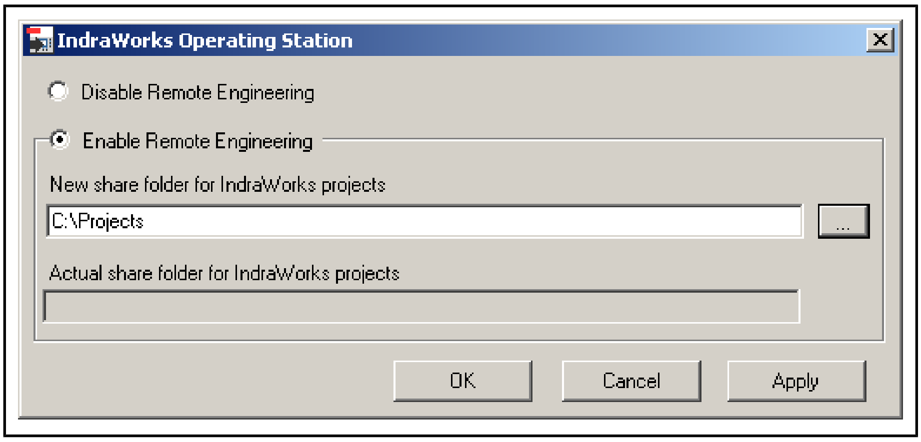


Fig. 14-3: Dialog: IndraWorks Operating Station (Specify the share directory)

"Cancel" closes the application without changing the share.

"OK" and "Apply" start the following sequence. With "OK", the application is subsequently closed:

- A user that is used internally by IndraWorks is created or his user account is activated.
- It is ascertained that the the engineering station of the IndraWorks Engineering interface has full access to the shared project directory and to the IndraWorks installation directory.
- Network shares for the project and installation directories are created.
- In Windows XP, the folder option "Use simplified file share" is reset. This is necessary so that the engineering station can use special login information to be able to use the network shares of the operating station.
- DCOM server mode is shared.

As long as the sequence is running, the dialog is in wait mode, i.e. it cannot be operated and the cursor is displayed as an hourglass.

14.2.2 Disabling

"Remote Engineering enabled" is displayed as follows:

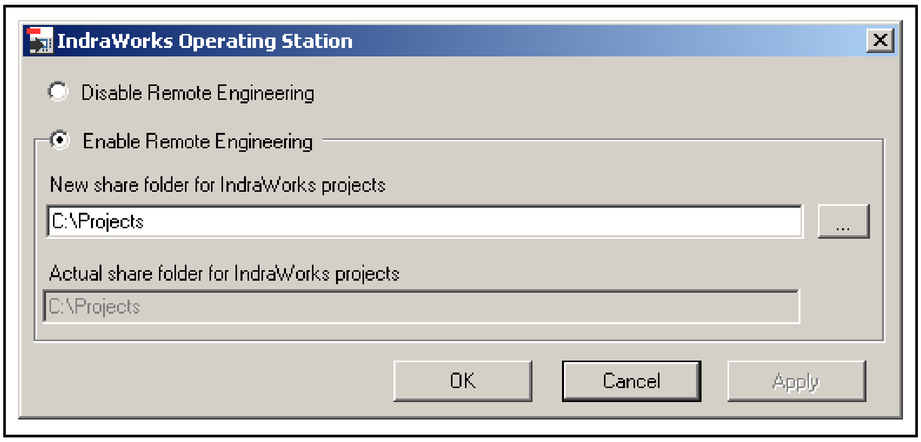


Fig. 14-4: Dialog: IndraWorks Operating Station (enabled configuration)

Remote Engineering is disabled by selecting the corresponding option switch and pressing "Apply" or "OK", i.e. the actions described above are reset in reverse order.

"OK" closes the application.

14.3 Operating the Engineering Station

14.3.1 Connecting to the Operating Station

The corresponding operating station for Remote Engineering must have been enabled before (see above).

To connect to the operating station, select the menu item of the same name in the IndraWorks Engineering interface:

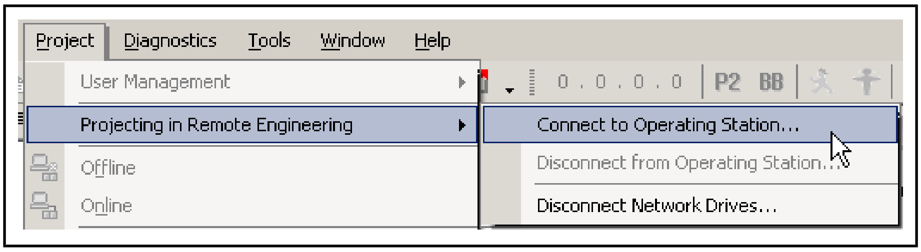


Fig. 14-5: Menu: Project (Connect to operating station)

If a project is open, close it before.

The user is requested to enter the computer name or the IP address of the desired operating station.

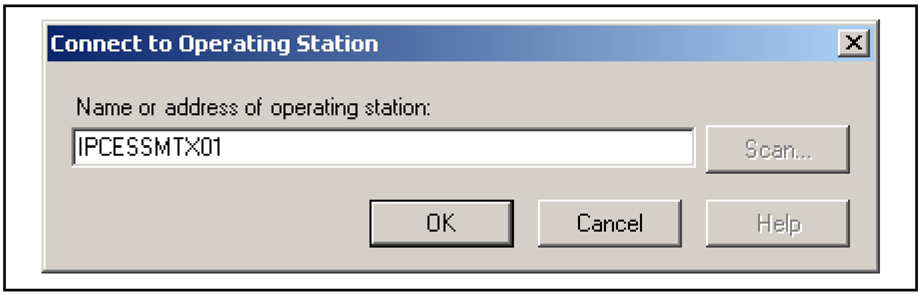


Fig. 14-6: Dialog: Connect to Operating Station (entering PC name)

"Cancel" closes the dialog without any further action.

"OK" starts the following procedure:

- DCOM settings for OPC client mode are made.
- The availability of the indicated computer is checked via network (using a ping command).
- Two network drive connections to the indicated computer are established.
  - \\<Computer name>\IW IndraWorks installation directory.
  - \\<Computer name>\IWP IndraWorks project directory.
- Remote Engineering is enabled.

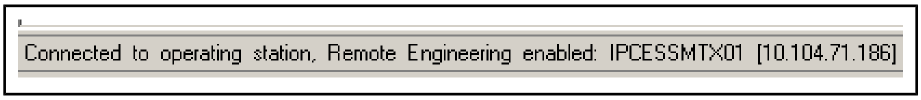


Fig.14-7: Status: Connection enabled

14.3.2 Remote Engineering

When Remote Engineering is enabled, the following changes when working with the IndraWorks Engineering interface occur:

- Two network drives are available to access directories of the connected operating station.
- The dialogs "Open Project", "Create new IndraWorks Project" and "Save Project As" indicate the project directory of the operating station.
- In the menu **Project ▸ Active for IndraWorks Operation** ("Activate project") the project to be loaded for the user interface of the connected operating panel is determined.
- The communication servers (e.g. IndraLogic gateway, SCP LogicalDevices, etc.) of the operating station are used, unless another computer name or IP address is explicitly assigned to the project.

14.3.3 Disconnecting from Operating Station

An existing connection to an operating station can be aborted via the menu **Project ▸ Remote Engineering ▸ Disconnect from Operating Station...**

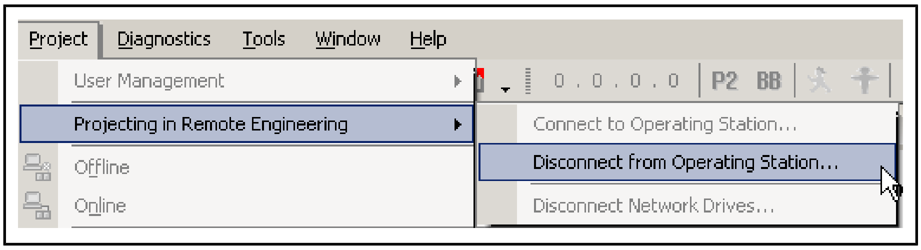


Fig.14-8: Menu: Project (Disconnect from Operating Station)

If the Engineering interface is editing a project, it is closed after a query before disconnection.

Disconnection from the operating station causes:

- Termination of network drive connections.
- Termination of the connections to the communication servers of the operating station.

The status shown in the status line:

Fig.14-9: Status: Disconnected from operating station

14.3.4 Disconnecting Existing Network Drives

To be able to successfully connect to an operating station, close all the existing network drive connections to this operating station because Windows does not permit network drive connections to be made to the same computer using different user names.

Start the standard Windows dialog **Disconnect Network Drives** and remove all existing network drive connections to the operating station that you want to connect to using the IndraWorks Engineering interface.

15 User Management

15.1 Overview

User management protects the MTX environment against unintended access to information, functions and operating actions.

The utilization and configuration of user management from the view of the user are described in this chapter.

User management works with users, groups and privileges.

The user is identified by his data and belongs to at least one group. His privileges are determined by the privileges of this group or these groups. A user that does not belong to any group has no privileges.

User data can be recreated, edited, copied, locked or deleted.

There are standard groups and freely configurable groups.

The standard groups will be determined by the manufacturer and the privileges will be assigned via a DLL.

Freely configurable groups can be recreated, edited, copied and equipped with privileges.

The dialog privileges of the freely configurable groups are assigned by an administrator.

Functions are assigned to the privileges. The user must be a member of the authorized group to be able to execute a function.

User management may be configured only by a user with the privileges of standard group "Administrators".

15.2 User Log-on/Log-off

In Engineering Desktop, logging the user on and off in menu **Project** and configuring of the user management is possible when the login procedure "Login with name and password" is selected.

User Management

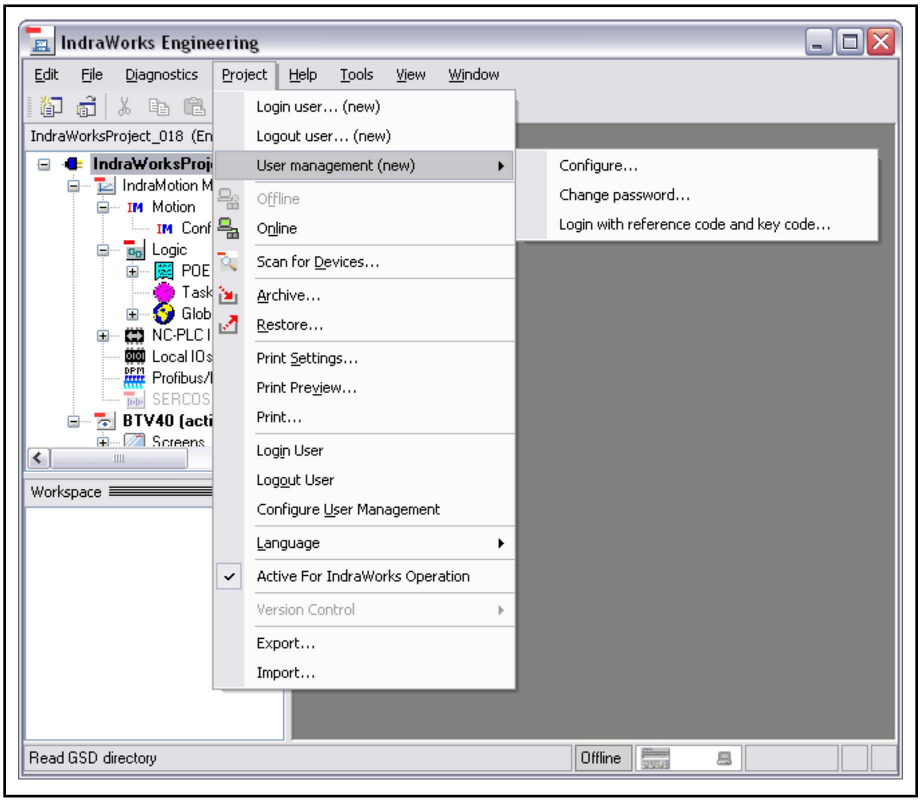


Fig. 15-1: Starting user management

Logging the user on and off is possible in Operation Desktop in **MenuFile** (not implemented yet).

During log-on, the user is requested to enter the user name and password in a dialog box.

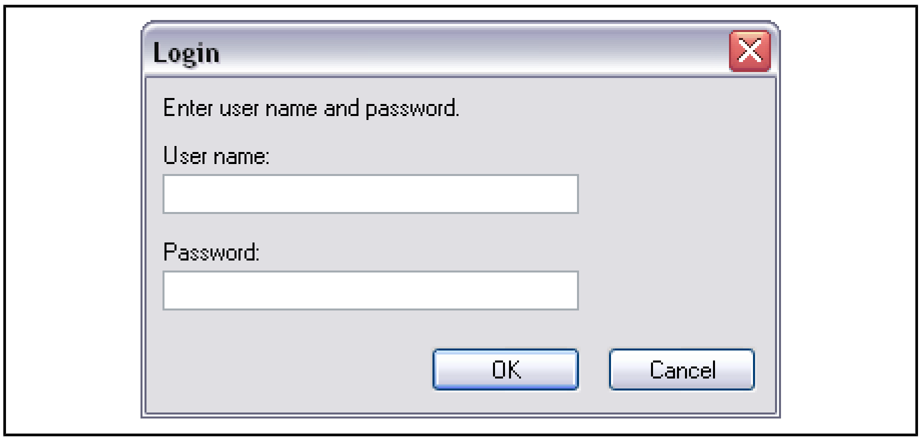


Fig. 15-2: Log on user

If the user is logged on for the first time or if the password has been reset, a password is determined in another input window.

15.3 Mode of Operation of User Management

If the saved functions are called in the MTX, the user privileges are checked before execution. If there are sufficient privileges available, the function is executed without informing the user about the examination. If the privileges are insufficient, a message appears.

Fig. 15-3: Log-on of unauthorized users

The input window "Log-on user" is displayed after pressing "Ok"; an authorized user can then log on immediately. If an unauthorized user logs on, the function is not executed.

15.4 Configuration of User Management

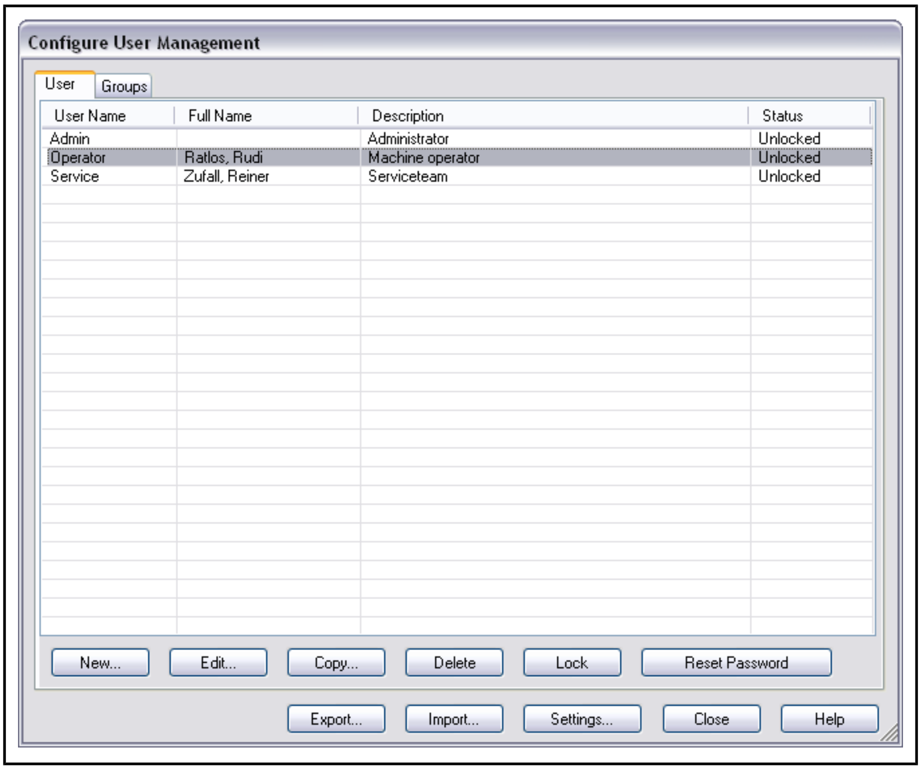


Fig. 15-4: Configuration of user management

Editing a User

User data can be modified via creating or editing using a dialog box. On the **User data** side, the personal data of user will be saved.

User Management

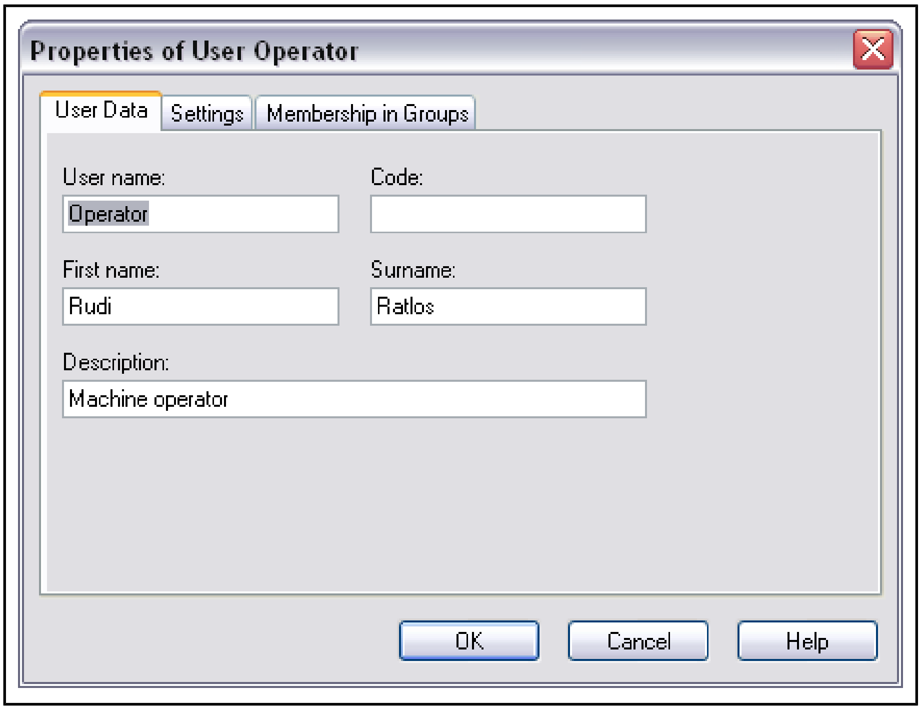


Fig. 15-5: Editing user data

Defining Group Membership

On the **Membership in groups** side, one or more group memberships of the user are determined. The user privileges are determined by the group membership.

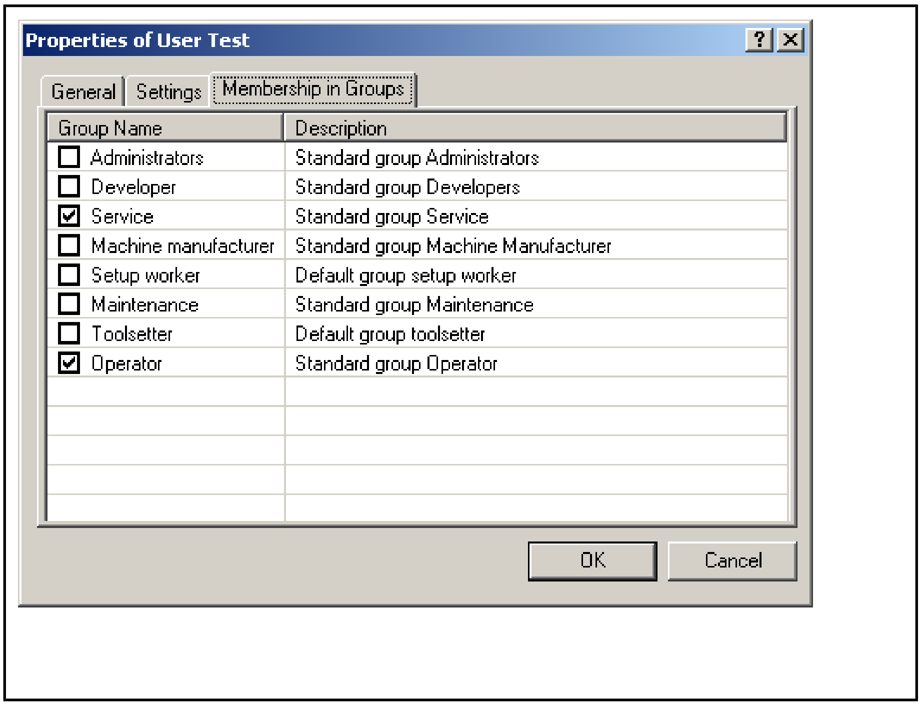


Fig. 15-6: Group membership

Settings

Specific settings for the user are possible in **Settings**. The user privileges are limited there temporally, or updating of the password is requested.

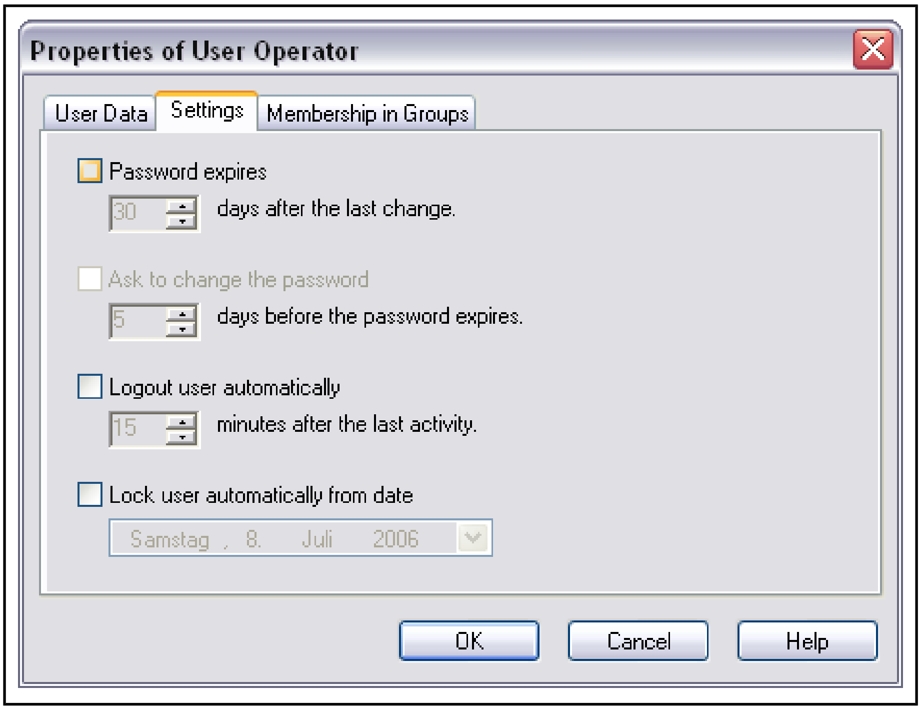


Fig. 15-7: User properties

Copying When "Copying" a user, a new user with the same personal data and group membership is created. A new user name must be determined for a unique assignment.

The settings which are assigned to a user must be applied during copying.

Delete All personal data, the group membership and all settings are removed permanently via "Delete".

Disable The user privileges are neutralized temporarily with "Disable". All data of the user remain and can be activated again with "Enable".

Password During the first log-on, the user is requested to enter the password. No privileges will be released without a password.

The password can be reset in the Configuration dialog box.

15.5 Configuring Groups

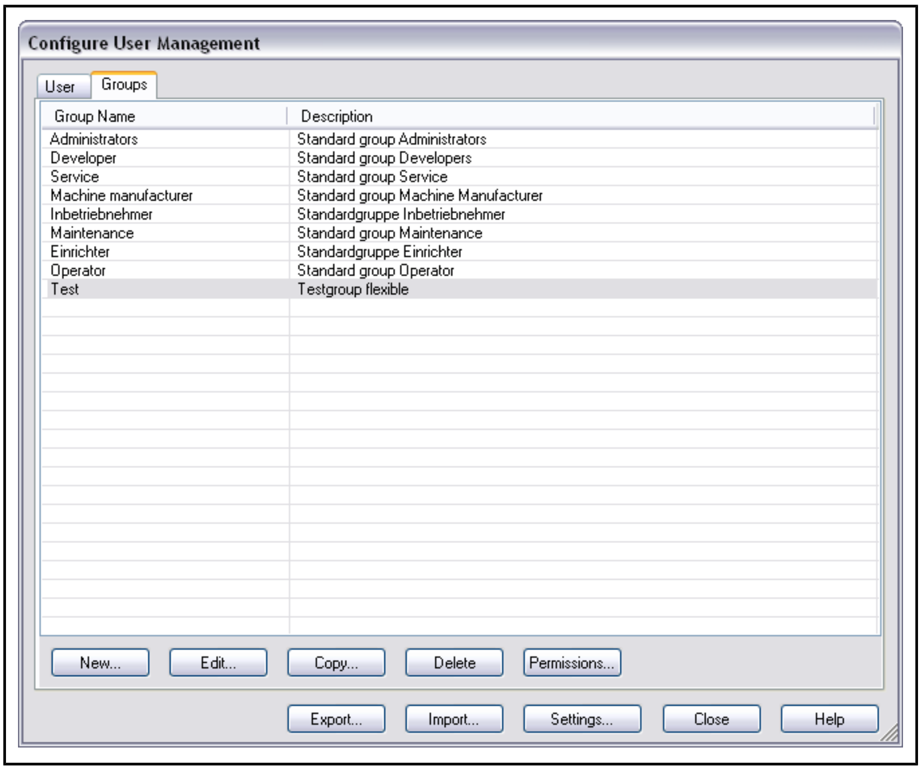


Fig. 15-8: Configuration of a group

Editing Groups

There are differences between standard groups and freely configurable groups during group configuration.

Unlimited creation and editing is possible only for freely configurable groups.

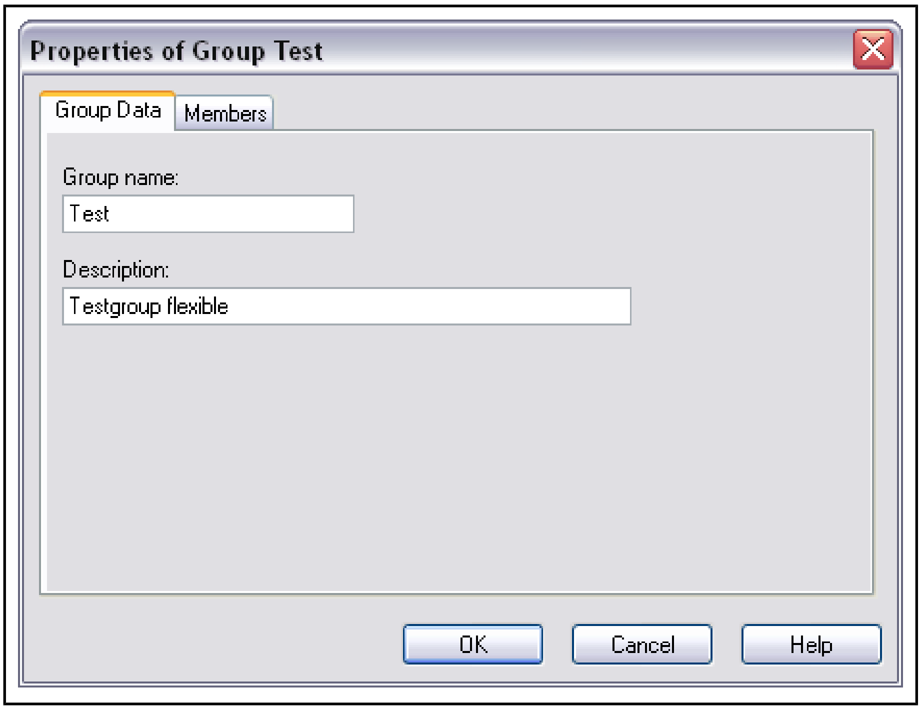


Fig. 15-9: Editing group data

The properties "Group name" and "Description" cannot be modified for standard groups.

Group Members The assigned users will be shown for the selected group and can be modified.

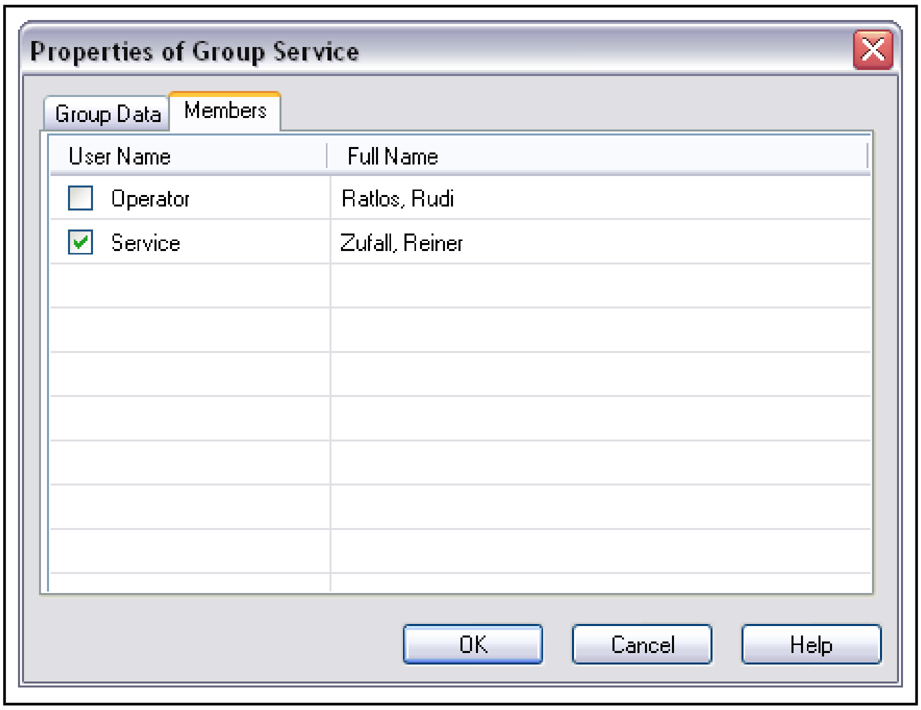


Fig. 15-10: Editing group members

Copying During "Copy", the groups will be copied with their privileges.

Delete "Delete" removes the group with its privileges permanently. Deleting standard groups is not possible.

15.6 Assignment of Privileges

Which functions can be executed by the user are assiged via the group to which the user is assigned. The user receives the privileges which are determined for this group.

If the user has the privilege "X", the users which are a member of this group can execute the function "X".

Context	Group	Function	L1	L2	L3	L4	L5	L6	L7	L8
Machine	NC screen	Configuration			x	x	x	x	x	x
		MDI		x	x	x	x	x	x	x
		ChannelReset		x	x	x	x	x	x	x
Program	NC editor	EditNpg		x	x	x	x	x	x	x
		SelectNpg	x	x	x	x	x	x	x	x
		SearchBlock		x	x	x	x	x	x	x
		EditDefaultCycles		x	x	x	x	x	x	x

Context	Group	Function	L1	L2	L3	L4	L5	L6	L7	L8
Program	ZOT editor	Edit		x	x	x	x	x	x	x
		Configuration			x	x	x	x	x	x
	DCT editor	Edit		x	x	x	x	x	x	x
	PMT editor	Edit		x	x	x	x	x	x	x
	MTX navigator	Configuration		x	x	x	x	x	x	x
		DeleteFile		x	x	x	x	x	x	x
		RenameFile		x	x	x	x	x	x	x
		PasteFile		x	x	x	x	x	x	x
		NewFile		x	x	x	x	x	x	x
		EditFileOptions		x	x	x	x	x	x	x
	CPL editor	Edit		x	x	x	x	x	x	x
		EditActive		x	x	x	x	x	x	x
		WriteValue		x	x	x	x	x	x	x
Tools	General	Import		x	x	x	x	x	x	x
		Export	x	x	x	x	x	x	x	x
		IncrementalInput	x	x	x	x	x	x	x	x
		AbsoluteInput		x	x	x	x	x	x	x
		ToolInsert		x	x	x	x	x	x	x
		ToolDelete		x	x	x	x	x	x	x
		ToolMove		x	x	x	x	x	x	x
		ToolPos	x	x	x	x	x	x	x	x
		ListConfiguration			x	x	x	x	x	x
	Tool editor	Invoke	x	x	x	x	x	x	x	x
		Configuration			x	x	x	x	x	x
IMT	General	Invoke	x	x	x	x	x	x	x	x
	MTX configuration	Invoke			x	x	x	x	x	x

Fig. 15-11: Editing privileges/functions

The assignment of group functions is made by the manufacturer. The table shows this assignment.

The groups assigned with L1 up to L8 are the standard groups:

- L1: Operator
- L2: Setter
- L3: Maintenance
- L4: Commissioning instruction
- L5: Machine manufacturer
- L6: Service
- L7: Developer
- L8: Administrator

The assignment is effected in the "Indraworks.Userman.Plugin_Mtx.dll" by GUIDs or public variables.

The complete overview with the access GUIDs or access variables can be found in "V06_IndraWorksMTX_Authorisation_FunctionID.xls".

For standard groups, these functions can be displayed via "Privileges" but can not be modified.

With this dialog box, the allowed functions are determined for the freely configurable groups. This is only allowed for a member or standard group "Administrator".

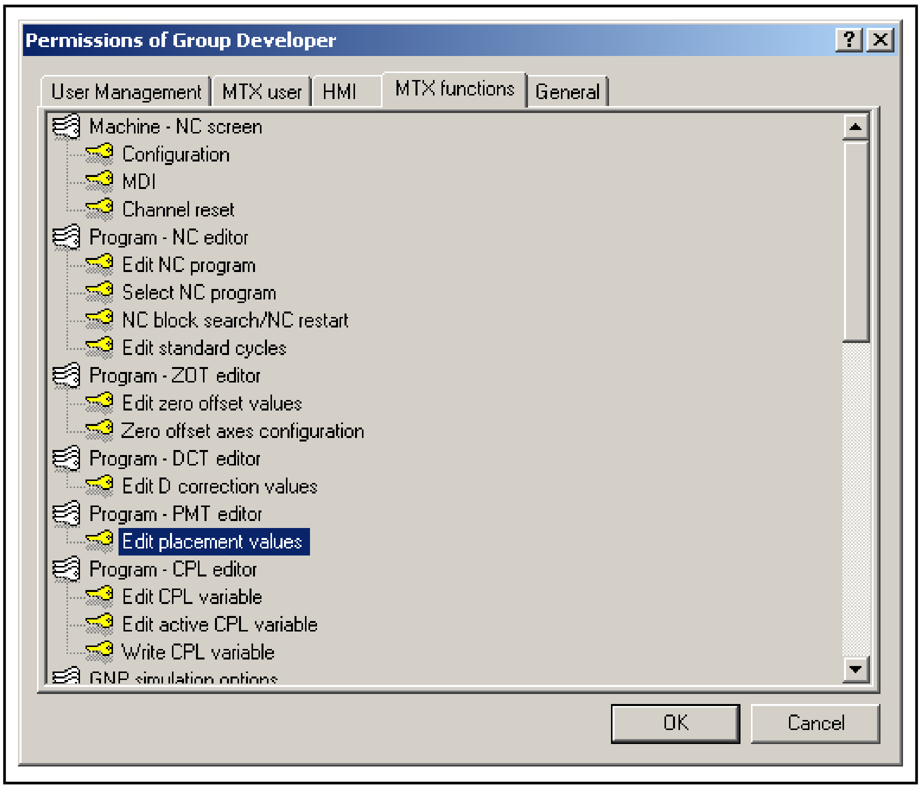


Fig. 15-12: Editing privileges/functions

15.7 Configuration of Settings

Additional specifications for user management can be set via the **Settings** page.

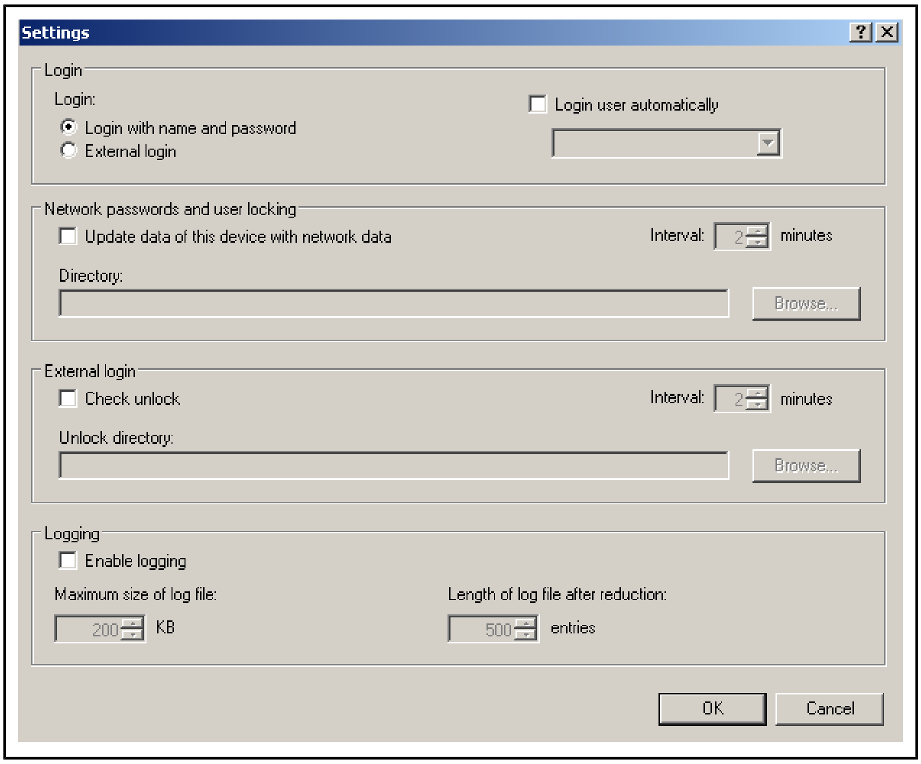


Fig. 15-13: Edit settings

To implement an "external log-on procedure", additional hardware or special directories are requested.

15.8 Export/Import

Users or groups are saved in a compressed file via "Export". The storage location and name can be selected freely.

The saved data are made accessible again via "Import" and selection of the file created in the export.

Users from versions older than MTX06VRS can also be accepted. For this, the data file of this (older) user management is selected as an import file e.g.: Acc001.dat.

15.9 Methods of User Management

Particularly "CheckPermissionWithLogin" is significant for the public methods of user management (IndraWorks.Userman.dll) in order to utilize Rexroth IndraMotion MTX. The return value is "True" if the logged-on user has the requested privileges.

If the privileges are insufficient, a log-on window for the user log-on is opened after a message. The return value is "False" until an authorized user logs on. Access to the protected function can be decided in the program.

Further methods such as CheckPermission, GetCurrentUserID und GetCurrentUserName supply return values without opening a log-on window.

The methods described here are available both for C# and for COM (Indraworks.Userman.tlb).

In the protected program parts with data input, it is a good idea to utilize user management both during a function call and during saving.

User management methods are currently not available for external applications (within the MTX).

15.10 Call from the Program Side

Call in C#

The recommended call is effected via the public variable. Direct access to the GUID is also possible.

1. Adding references

   `Indraworks.Userman.dll`

   `Indraworks.Userman.Plugin_Mtx.dll`

2. Adding namespace

   `Using Rexroth.Indraworks.Userman;`

3. Variable declaration (example)

   `Private IUsermanClient _usermanClient =null;`

   `Private Guid GuidZOTEditorEdit;`

4. Instances (example)

   `this._usermanClient = UsermanClient.GetInstance();`

   `GuidZOTEditorEdit = Rexroth.IndraWorks.Userman.MtxPermissions.GuidProgramZOTEditorEdit;`

5. Call (example)

   `bool boAuth = this._usermanClient.CheckPermissionWithLogin(this, GuidZOTEditorEdit );`

   `if( !boAuth )…`

Call in COM (Example for VB)

For COM applications, calling is possible only directly via the GUID.

1. Adding references

   `Indraworks.Userman.tlb`

2. Variable declaration (example)

   `Private IUsermanClient as New IndraWorks_Userman.UsermanClientCCW`

3. Call (example)

   `Dim bool as boolean`

   `bool=IUsermanClient.CheckPermission(„{A59F589B-D497-4519-B014-3F0401EEF768}“)`

   `If bool = False Then …`

16 Configuring the Tool Management

16.1 Basics

16.1.1 Tool Corrections

The tool data management is of highest importance for controlling machine tools.

The tool data comprises, for example, data on:

- Tool identification
- Description of tool geometry (tool correction data)
- Tool life management
- Description of tool location
- Description of tool type
- Status information
- etc.

In addition to these so-called external tool corrections, D-corrections can be used.

The following figure explains the mode of operation of external tool corrections and D-corrections.

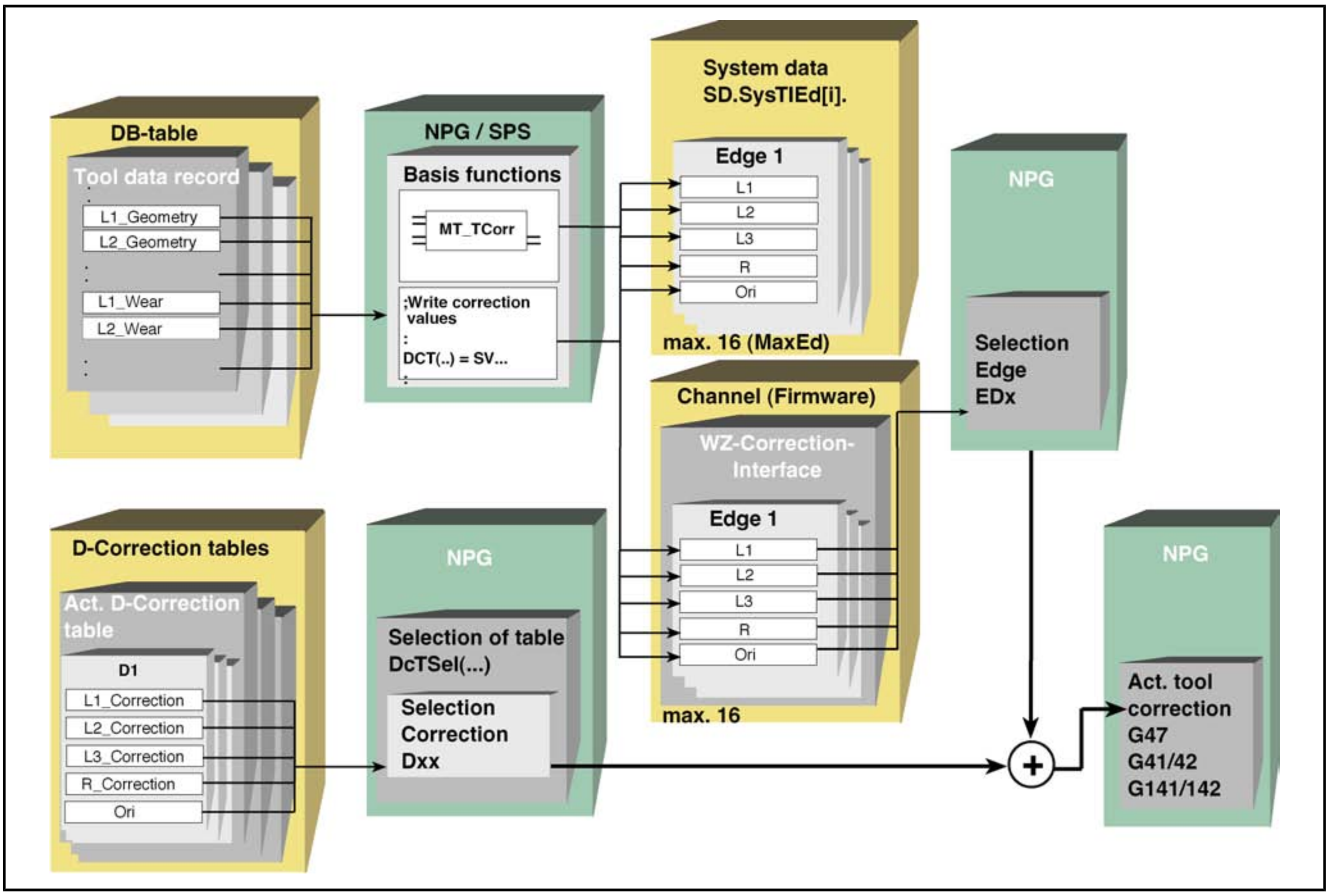


Fig.16-1: Principle of D-corrections and external tool corrections

The block preparation of the MTX takes the tool corrections stored in the tool correction interface into account. The tool correction interface can be written via NC (via CPL) or PLC. Within this interface, corrections for up to 16 tool edges can be stored. A correction set includes correction values L1, L2, L3, R as well as the edge position Ori and orientation angles Phi, Theta and Psi. By means of the orientation angle, the tool can be positioned at an inclined angle in space. G47 actives the MTX tool length correction, and G48 deactivates it. Use ED (cyclic duration) to switch the edge or the correction data set within the tool correction interface.

The assignment of tool length corrections L1, L2, and L3 to the axes or coordinates where the length corrections are to take effect is performed via the configuration data. Irrespective of this setting, the effect of the tool length corrections can be changed via G47 during operation. G47 simultaneously activates tool correction for the edge last preselected using ED.

16.1.2 Data Management

The data management of the external tool corrections is managed in a database. The database is subdivided into database tables. Presently, 2 database tables are available. By default, database table 1 is used for managing the data of the external tool corrections. Accordingly, database table 2 is available to manage pallet data for example.

The following figure provides an overview on the database structures and its exemplary usage:

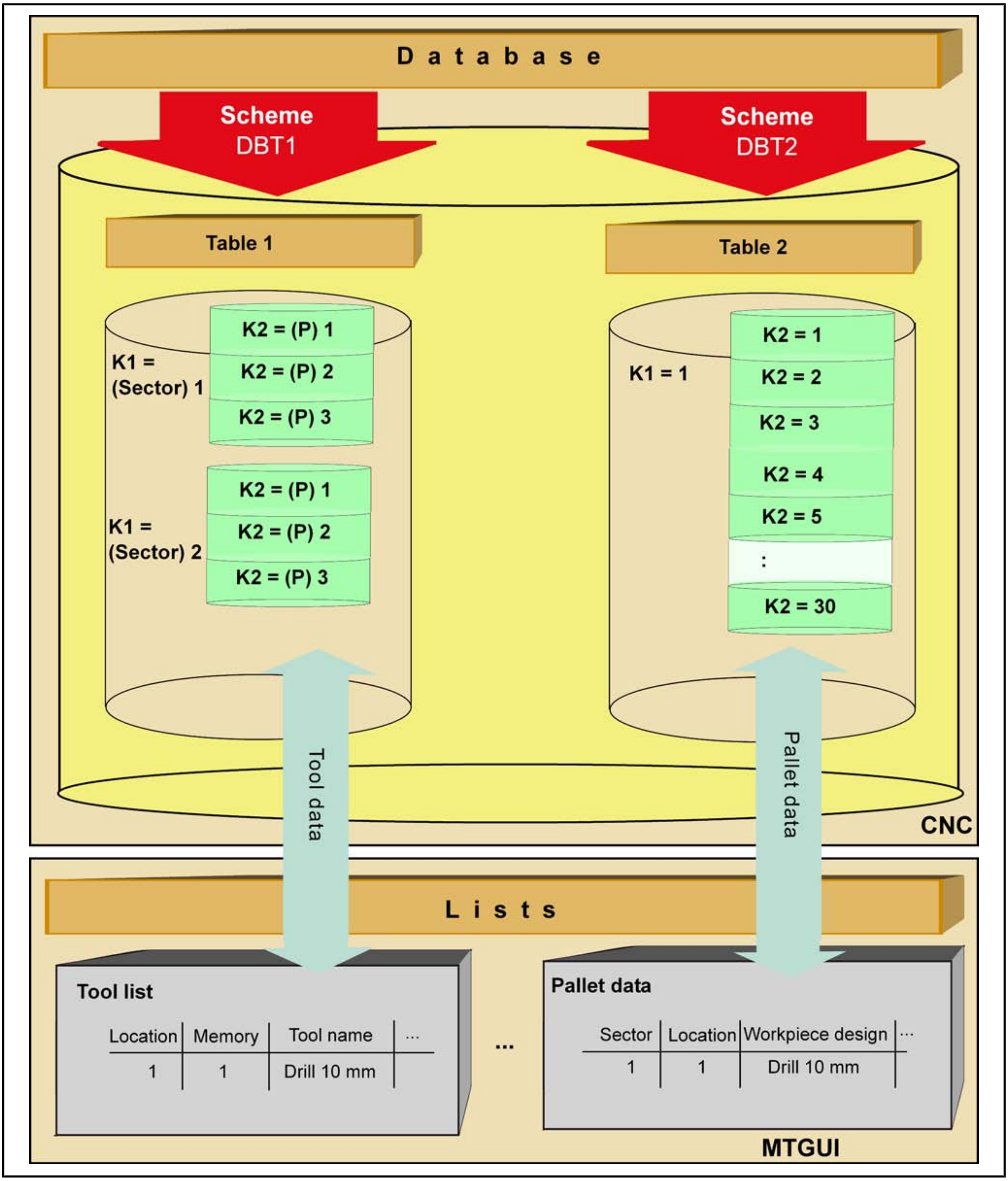


Fig. 16-2: Database structure

The tool database can be compared to a container in which necessary tool data can be stored in a structured manner.

The database consists of individual data records. Their structure can be configured related to the user.

Array structure of a data record:

Array/element	Identifier	Type	Meaning	Comment
1	K1	Integer	Sector	DS key
2	K2	Integer	Place	
3	SKQ	String	ID	Tool identification
4	IKQ1	Integer	Duplo no.	
5	IKQ2	Integer	Type	Type identification
6	IKQ3	Integer	T. No.	Tool identification
7	IQ1	Integer	Reserve	-
8	IQ2	Integer	Reserve	
9	IQ3	Integer	Reserve	
10	BQ1	Bit array (32)	P-status	Place status
11	BQ2	Bit array (32)	T-status	Tool status
12	BQ3	Bit array (32)	Technology	Type identification
13	aaa	1. Freely configurable data element		
:	:	:		
n	zzz	nth freely configurable data element		

Fig. 16-3: Data record structure

A data record is uniquely identified by the two arrays "Storage"/"K1" and "Place"/"K2".

A single data record always corresponds to a location which can receive a tool. This can be specific places in the machine (e.g. spindle, gripper, charging place, discharging place) or in a magazine.

When tools are changed into a place or transferred to other locations, this causes the relevant tool data to be copied and transmitted in the concerned data records.

The number of data records available after re-initialization of the database can be configured as well as the display of tool data in tool lists and tool editors.

Presently, the size of a data record is restricted to 4 kB.

16.2 General Configuration Tools

16.2.1 General

To configure tool data management, follow the configuration steps described below in the specified sequence:

1. Specification of the database size or the number of data records (chapter 16.4 "Configuring the Database" on page 341).
2. Definition of the database structure (sector / location; chapter 16.4.2 "Defining the Sector and Location Assignment of the Database Table" on page 342)
3. Adjustment of the database schema (chapter 16.4.3 "Configuration of Data Records" on page 344)
4. Optional extension of the tool catalog (chapter 16.5 "Tool Catalog" on page 370)

5. Configuration of the user interface (chapter 16.6 "User Interface" on page 385)

16.2.2 Definition of Terms

ULC **"ULC"** stands for "**U**niversal **L**ist **C**ontrol", which is the central element of tool list and container configuration.

Sublist **"Sublist"** is a central term in schema definition for the ULC. A ULC is a table editor which may be able to multiply the configured line-column definition for the presentation of a (tool) data record according to the number of data records. In this case, a sublist stands for the presentation of a data record. Therefore, only one sublist needs to be defined for a list of several tool data records.

This principle can also be applied to certain parts of the (tool) data record (e.g. tool edge data, etc.).

16.2.3 Schema Editor

General

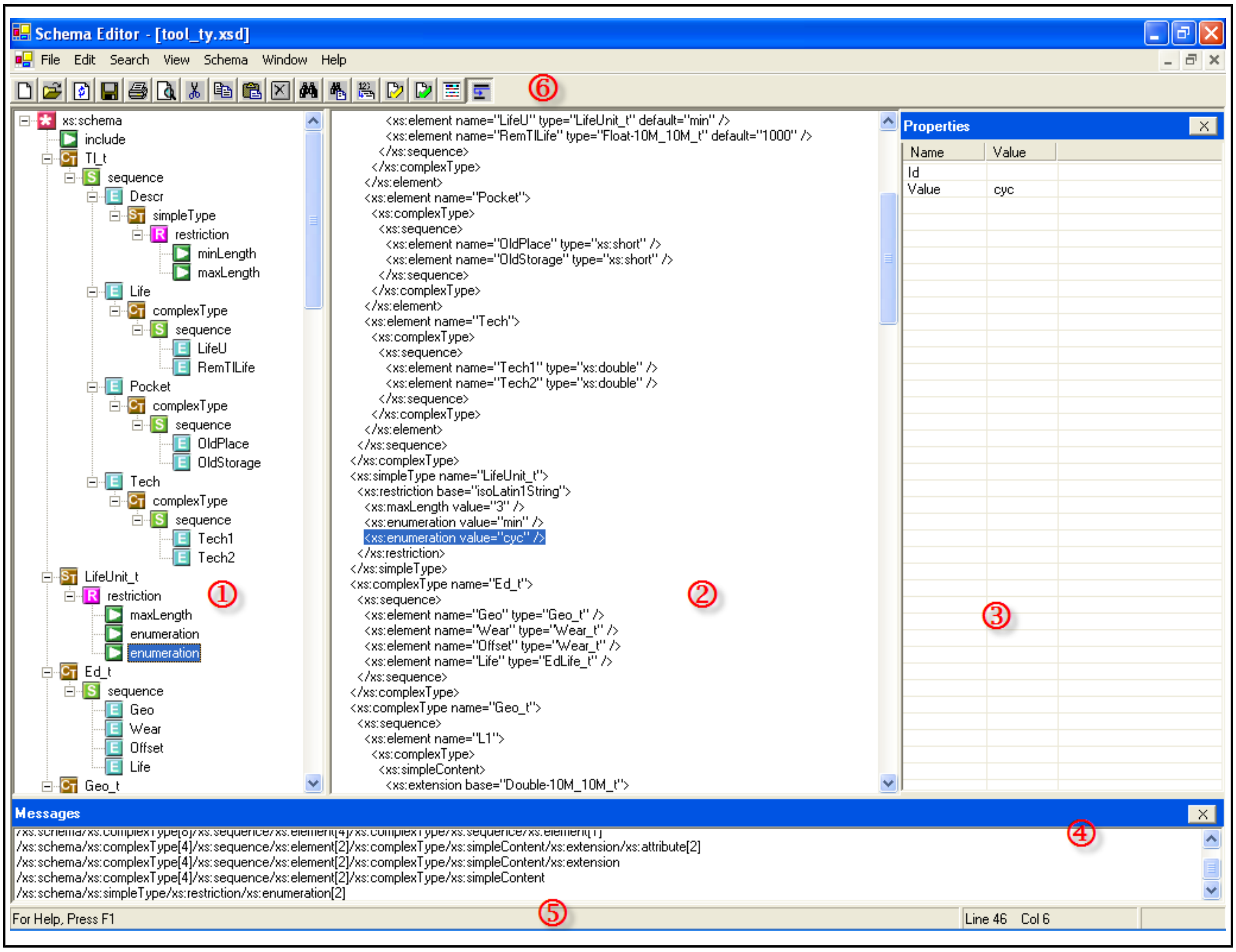
A schema editor is provided especially so that the user can see and change data record schemas.

In addition to the possibilities of schema validation and of the "style" test, this schema editor permits schema files to be clearly displayed and modified within various windows.

The figure below shows the various areas of the schema editor.

- Tree display (always present)
- Text display (always present)
- Properties window (can be optionally hidden)
- Message window (can be optionally hidden)
- Status and tool bar (can be optionally hidden)

Configuring the Tool Management



1	Tree representation
2	Text display
3	Properties window
4	Message window
5	Status line
6	Toolbar

Fig. 16-4: MTX schema editor

Graphical Element	Meaning
	Root node in xsd files
	Simple type
	Complex type
	Element
	Attributes
	AnyAttribute
	AttributeGroup
	All
	Any

Graphical Element	Meaning
C	Choice
G	Group
L	List
R	Restriction
S	Sequence
a	Annotation
d	Documentation
U	Union
	Gen. symbol for: • minLength • maxLength • Include • #text • simple Content • extension • enumeration • ...

Editing in the Tree View

- **Erasing**:

  Select the node and use the <DEL> key or click with the right mouse button on **Delete**
- **Paste**:

  Right-click with the mouse, pop-up **Add child.**. This menu item provides a selection of all schema elements that can be pasted under the currently selected element.

The pasted elements are immediately visible in the Text Editor. In order to change the elements, the Properties windows or the Text Editor can now be used.

Editing in the Properties Window

The properties listed in this window always belong to the node selected in the tree representation. Different properties are possible from node to node. If only a limited range exists for the value of a property, this range is provided in combo boxes.

After the node has been changed, the other two views (tree and text) are updated.

Editing in the Text Editor

Here, anything can be entered without any limitations. By means of the button <F5> "Synchronize Tree", the tree is updated. Any errors are detected during the update and are shown in the "Messages" window. Furthermore, you can check your text for "Good style" <F7> and for compliance with the schema conventions <F8>.

Entries Menu

File menu

New <Ctrl>+<N>:	Create new file; A selection can be made between XML and schema and between various Unicode character sets.
Open <Ctrl>+<O>:	Open an existing file
Close:	Close the currently open file
Close all:	Close all open files
Save <Ctrl>+<S>:	Save the current file
Save as:	Save the current file under a different name
Print <Ctrl>+<P>:	Directly print the current file
Print preview:	Display a print preview
Print setup:	Set up the page to be printed
Recent files:	List of the files opened last
Exit:	Exit the schema editor

Edit Menu

Cut <Ctrl>+<X>:	Cut the highlighted characters in the text editor
Copy <Ctrl>+<C>:	Copy the highlighted characters in the text editor
Paste <Ctrl>+<V>:	Paste the highlighted characters in the text editor
Delete text:	Delete the highlighted characters in the text editor
Delete node:	Delete the highlighted node in the tree view
Delete messages:	Delete the messages in the messages window
Select all <Ctrl>+<A>:	If the text editor is active, mark the entire text from top to bottom.
Go to line/char <Ctrl>+<G>:	Open an input window to enter the line and character number and to focus the specified character in the text editor.
Set font:	Change the font of the text editor
Format text:	Change the color of key words

Search Menu

Find <Ctrl>+<F>:	Open the "find" window
Replace <Ctrl>+<Shift>+<H>:	Open the "find and replace" window

View Menu

Highlight text <Ctrl>+<H>:	Specifically switch "highlight key words" on or off. Documents up to a certain size are highlighted when opened. For performance reasons, larger files are not highlighted when opened.
WordWrap:	Switch the automatic line break on or off
Expand tree view <Ctrl>+<Shift>+<E>:	Expand the tree and all the nodes
Collapse tree view <Ctrl>+<Shift>+<C>:	Collapse the tree and all the nodes

Schema view <Ctrl>+<Shift>+<S>:	Switch on or off schema view The schema view hides certain structure nodes, e.g. the nodes of the type "simpleContent" and "complexContent" are hidden.
Synchronize tree <F5>:	Scan text and update the tree view. Scanning errors are shown in the message box.
Schema <F4>:	The schema that describes the permitted elements of the schema that is currently being processed is displayed.
Customize...:	Show/hide toolbar buttons
Properties <Ctrl>+<Shift>+<P>:	Show/hide properties window
Messages <Ctrl>+<Shift>+<P>:	Show/hide messages window
Tool bar <Ctrl>+<Shift>+<P>:	Show/hide toolbar
Status bar <Ctrl>+<Shift>+<P>:	Show/hide status bar

Schema Menu

Check style <F7>:	Checks the "style" of the document. For example, missing parentheses are found here.
Validate <F8>:	Checks the document against the schema on which it is based.

Window Menu

Cascade:	Cascade the windows
Tile horizontally:	Arrange the windows under one another
Tile vertically:	Arrange the windows next to one another

Help Menu

Help topics <F1>:	Show the "Help" dialog
About the schema editor:	

16.2.4 ULC Configurator

General

The ULC Configurator has 2 types of visual presentation which, however, use the same database (configuration file), i.e changes in one of the configurators will have the same effect on the other configurator.

1. Configurator for common configuration steps.
2. XML editor for special settings which cannot be made by means of the configurator (see chapter 16.2.5 "XML File Editor" on page 329).

This chapter especially deals with the ULC configurator.

Opening or Creating a Configuration

Opening an Existing Configuration

The ULC configurator is either called by the list configurator via the configuration key  or the function "Open" on the relevant entry in the branch "Form Configuration".

Creating a New Configuration

Via **New** under the node **List Controls** or **Container Controls** either a new list configuration or a new container configuration can be created for the tool editor (see fig. 16-6 "Template for a new container configuration" on page 313). For this purpose, the relevant basic configuration is referred to for initial definition. Another option is to set a similar configuration as basis via **Duplicate (Copy &**

Paste) for a new configuration file. After having created such a configuration, load it into the editor as described above.

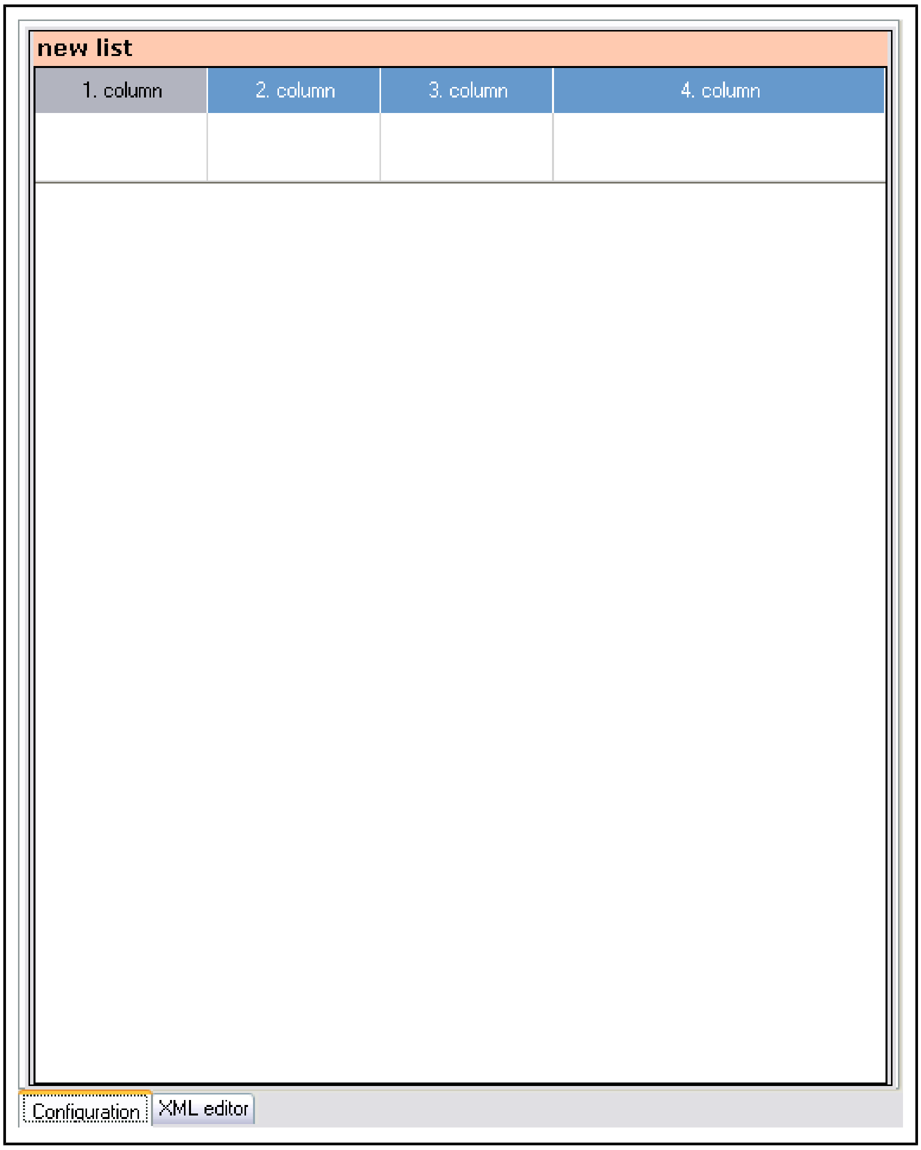


Fig.16-5: Template for a new list configuration

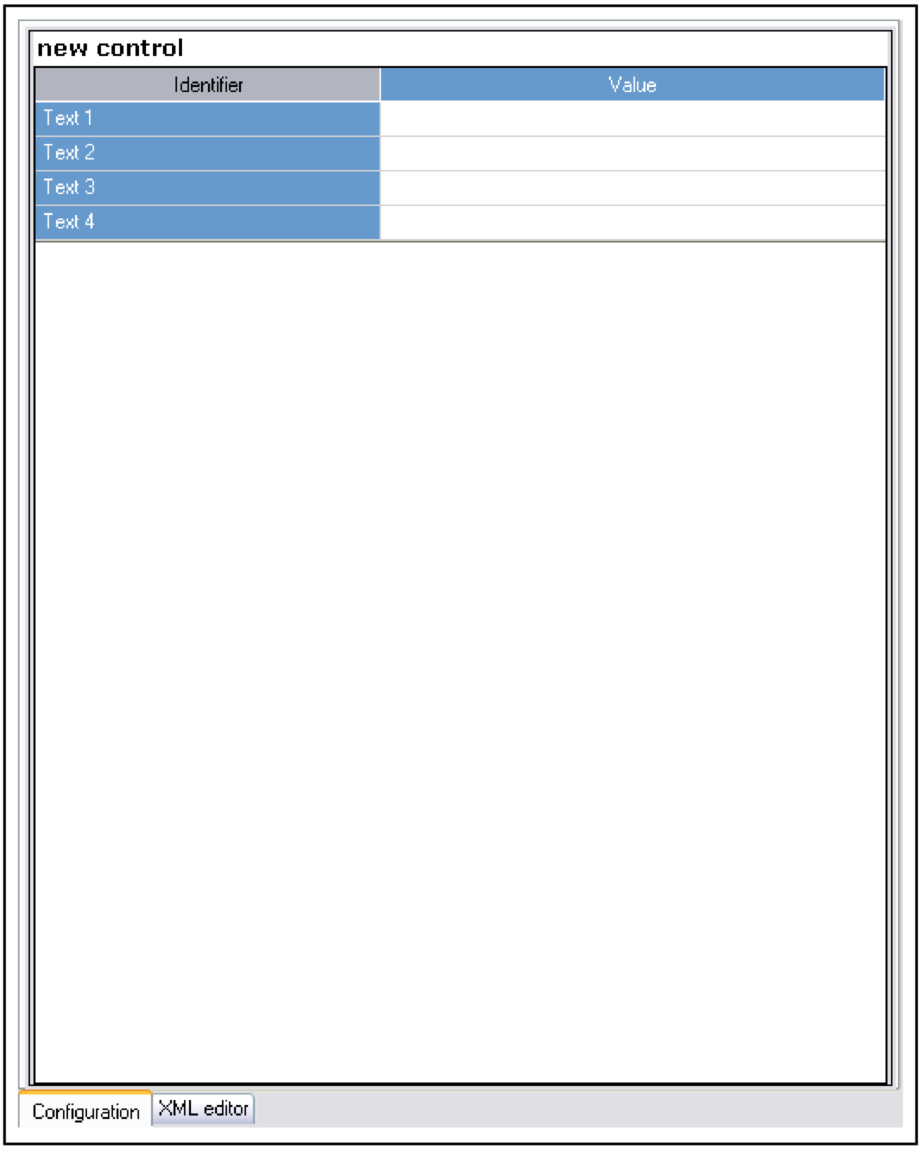


Fig. 16-6: Template for a new container configuration

Saving the Changed Configuration

If the configuration has been changed, the user will be asked whether the changes are to be saved or not when exiting the configuration dialog. It is also possible to save the current state via **Save** <Ctrl>+<S> in the pull down menu.

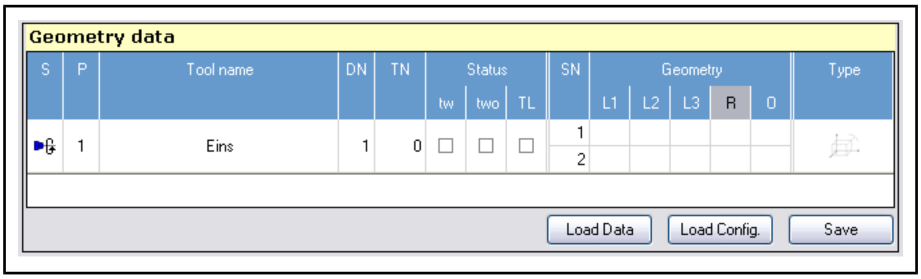


Fig. 16-7: ULC preview with function buttons

Creating the Desired Number of Lines and Columns

To obtain the desired number of lines and columns, the initial configuration must be modified. To do this, call the corresponding commands using the right mouse button and the pop-up menu. For example, to insert a column at the front, use the right mouse button to click the first column and execute command **Insert ▸ InsertColumn ▸ Before**

The following commands exist:

- Insert
  - Insert column
    - Before
    - Behind
  - Insert subcolumn
    - Before
    - Behind
  - Insert row
    - Above
    - Below
- Delete
  - Column
  - Subcolumn
  - Row

In contrast to the command "Insert Column", the command "Insert Subcolumn" does not generate a complete column; instead, it generates a subcolumn that shares the topmost heading with the initial column.

Opening the Cell Editor

The cell editor is opened by double-clicking a cell or via the pop-up menu. When the editor is open, click another cell using the mouse and edit its data. The data of the cell that was edited previously is saved in the clipboard. The modified data record is added to the preview by pressing "Accept" or "OK".

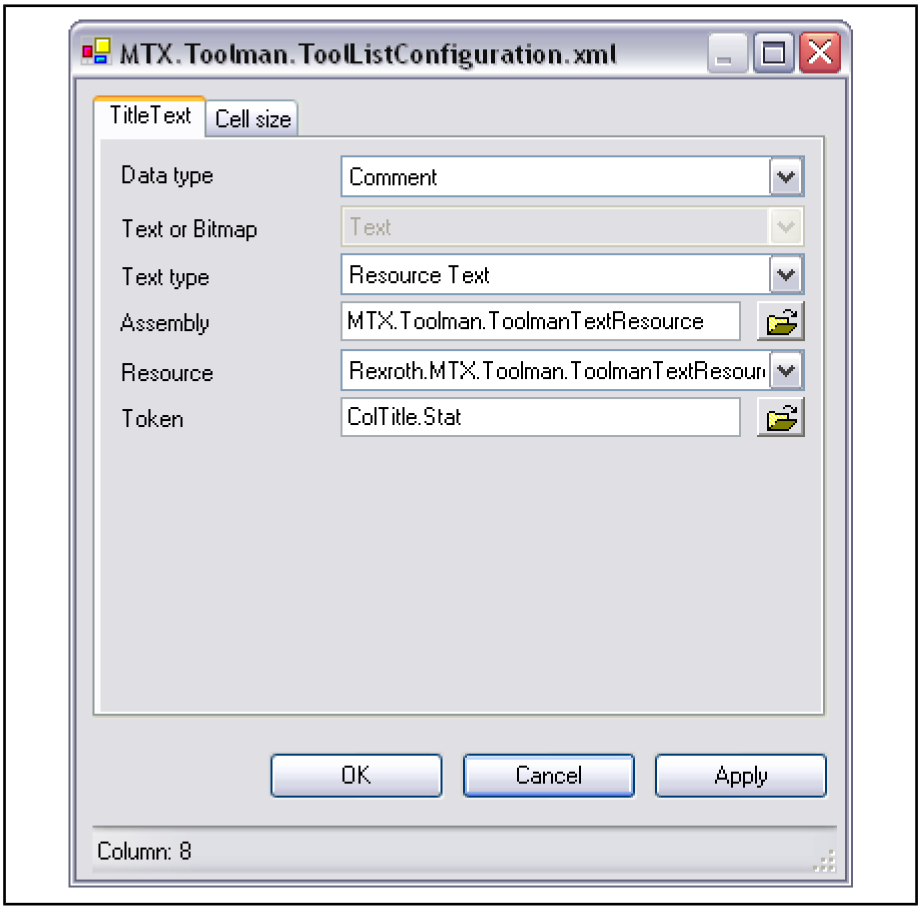


Fig. 16-8: Dialog: MTX.Toolman (editor for cells, columns, titles and global list properties)

Setting the Width and Height of Rows and Columns

The width and height can be set in two different ways. To be able to make all the settings using the mouse, activate the edit mode: right mouse button -> pop-up menu **Edit column width / row height**. Now the width and height of every line and column can be modified using the mouse.

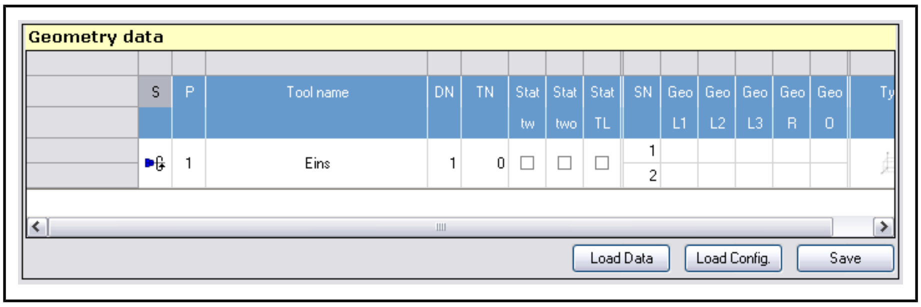


Fig. 16-9: The row height and column width can be modified directly using the mouse in this preview

The second method is using the editor. Open the editor as described above. Make sure that the cursor is not positioned over the title row, but that the row/column to be modified is selected. Open the second tab page **"CellSize"** and enter the desired values in the fields Height / Width.

Creating the Contents of Individual Cells

Logic of the Cell Contents

In order to edit the content of individual cells, open the editor again, as described above. Ensure that only the cell whose data is to be modified is selected.

If you press "Apply", the modified data is applied to the preview, but are not permanently saved – instead, it is saved only in RAM and applied to the preview.

In general, a cell can have three states:

1. It can be empty, this means "data type = **empty cell**".

   Nothing else can be set in this case.
2. The cell can contain a comment (**comment**).
3. The cell can reflect the value of a process variable (**process variable**).

Empty Cell

The cell is empty; it does not display a comment or a process value. Nothing else can be set here.

Comment

The cell displays a fixed comment. This comment can consist of a bitmap or text. In turn, the text can be permanently entered or can be language-dependent. Language-dependent texts can originate from a resource file or can be selected from a CSV file. CSV files can be edited by the user in this dialog.

If the cell is defined as a comment cell, there are several input methods:

Text or Bitmap

There are 3 methods here:

Text	Only text is displayed
Bitmap	Only a bitmap is displayed
Text and bitmap	Both text and a bitmap can be specified and selected. The corresponding style can be used to specify how the text and the bitmap are displayed, i.e. on top of or next to one another. Currently, the data of the style must still be modified directly in the configuration file using an XML editor.

Text type

The following three selection possibilities exist:

User text	This reads out the CSV file mentioned above. If a user text is used, the elements "list name" and "token" need to be specified in addition.
Text resource	The comment to be displayed is read from a resource file with text. File name and text name must be specified.
Fixed comment	In this case, the text can be entered directly in field "comment".

List name

This field is available only if "User text" has been selected for the text type. Either a list can be selected or a new one can be generated with "New".

⚠ CAUTION

⇒ Although this field can be adjusted in every cell, it is global and thus always changes the content for every field!

Token (User Text) A token can be entered directly. However, it is a good idea to select the token from the list. To do this, press the "Open" button. Now a box with the list of all texts that have already been defined opens. This list can be used to not only select texts, but also to generate new ones. A description of this list can be found in "Edit Text List".

Assembly The field "Assembly" is provided for the text type "Resource Text". The content of this field can be selected using a file dialog that opens when you press "Open" on to the right of the input field.

Resource If you have selected an assembly, all the resources in this assembly are contained in the Resource combo box, where they can be selected.

Token (Resource) If a resource that contains the texts has been selected, a token can be selected from the contained token/text pairs. Start the text list using the "Open" button. The description of this list can be found in "Edit Text List".

Comment This field is available for the text type "Fixed comment". The text input here will be entered later in the cell.

Assembly of Image If Bitmap was selected in field "Text or Bitmap", the fields for selecting a bitmap are accessible. As has already been the case for the text assembly, a file dialog is used in "Assembly of image" to select an assembly that contains a resource with images.

Image Resource If an assembly has been selected, this combo box contains the resources of this assembly so that you can select the suitable one.

Image Name If the selected resource contains images, they can now be selected using the text list. This text list contains the names of all the images. The description of this list can be found in "Edit Text List".

Process Variable

The cell displays a process-dependent value. This value is addressed using an XPath expression. The value can be displayed as text and/or as a bitmap. It can be formatted as text and be edited in the cell. Make the following entries to set all these options:

Process variable The XPath for the process variables is entered here.

Text or Bitmap There are three possibilities:

Text	Only text is displayed
Bitmap	Only a bitmap is displayed
Text and bitmap	Both text and a bitmap can be specified and selected. The corresponding style can be used to specify how the text and the bitmap are displayed, i.e. on top of or next to one another. Currently, the data of the style must still be modified directly in the configuration file using an XML editor.

Text Format

Numerical	The text is displayed as a number that can be formatted.
Text	The text is displayed as a string. Using this setting, the process value is displayed directly, without any formatting.

Numerical Type If the format "Numerical" has been selected for the text, the number can be specified in more detail now. "Numerical type" is used to define how the number is to be displayed:

Integer	The number is displayed as a whole number
Float	A number with positions after the decimal point is displayed

Bit	Checkboxes are displayed
Use format string	The representation of the number is specified in more detail in the field "format string".

Edit Mask The field "EditMask" is available for the numerical types Integer, Float and "Use format string". This field can be used to specify which form the user's entry must satisfy. Fixed characters can be assigned to the Edit field of the cell. The syntax is described in chapter "Syntax of Edit Mask" on page 325.

Digits Digits can only be set if "Numerical type" = "Integer". The number entered here indicates the minimum number of digits with which the number is displayed. Missing digits are supplemented by leading zeros.

ProcVar Type The following types can be selected:

- System.Double
- System.Single
- System.Decimal
- System.Integer

Decimal points The minimum number of positions after the decimal point is specified here. Missing digits are filled by zeros. This field exists only if "Numerical type" = "Float".

Bit no. (0..32) This field is visible only if "Numerical type" = "Bit". Here, the bit is defined by an integer whose value you want to display as a checkbox.

Format String This field is available only if "Numerical type" = "Used format string". The entry in this field determines how the number is displayed in the cell.

The syntax is as follows:

C or c	Display as a localized currency
E or e	Display as an exponential number
F or f	Fixed number of positions after the decimal point
N or n	Like F, but with separator symbols for thousands
P or p	Number is multiplied by 100 and displayed as a percentage
0	Placeholder for numeral. If no numeral is located at this position, one is inserted.
#	Placeholder for numeral. A numeral in this location is displayed; if there is no numeral in this location, it is not filled by 0.
.	The character "." indicates the position of the decimal separator symbol.

Editable There are 4 possible values for Editable **(EditStatus)**:

0	**Yes**	Cell can be edited
1	**No**	Cell cannot be edited
2	**Call-back of the application**	The call-back mechanism is used to query the calling application whether the currently selected cell can be edited.
3	**Depends on process variables**	Important for conditional editability

Edit Type There are 3 options for the edit type **(EditTypeSelection)**:

Text editor	An input field is available in the edit mode.
Enum combobox	If a cell is focused, it becomes a combo box whose entries must be available in the schema for the data element to be defined.
User combobox	When the cell is focused, it becomes a combo box. The entries in the combo box list are queried by the embedded application using call-back.

Copying Cell Definitions

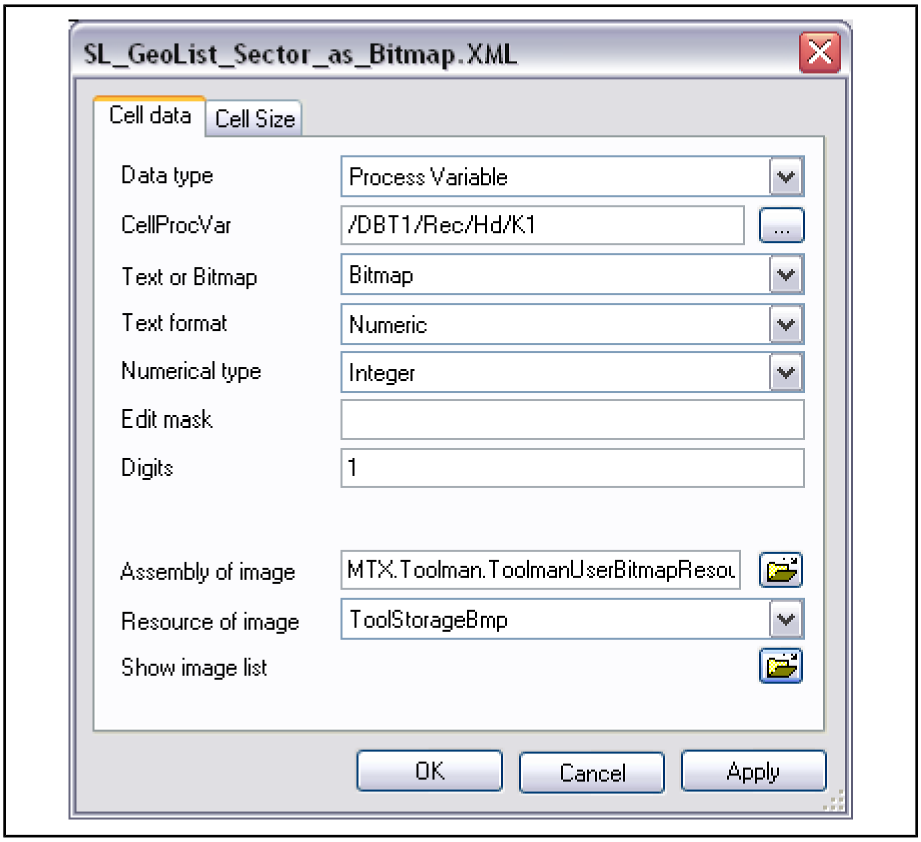
Complete cell definitions can be transferred from a selected cell to another cell by means of the function **"Copy cell data"** and/or **"Paste cell data"**.

Merging Cells

Vertical cell rows can be merged by highlighting them and summarizing them in one cell system using the function **Merge Cells** o be called via the pop-up menu. Via **Split Cells** A highlighted cell system can be split again.

Creating Process-dependent Bitmap Selection

Screens can also be shown subject to process variables. To this effect, a list with process values and the corresponding screen is being defined. In the configurator, there is an own configuration dialogue for the definition of the Bitmap list. If you choose the display mode "Bitmap", you at first will have to choose Dll and Resource which you would like to choose screens from. Afterwards, define a "key - value - pair" using the dialog below. In this case, the value corresponds to the value of the variable which is linked to this cell. The value is the name of the screen from the resource. You can also indicate whether the value has to match the table value exactly or whether it has to be lower or greater.



Assembly of the Bitmaps	Dialogue for selection of the DLL with the screens
Screen resource	Combobox with all resources, the Dll
Show screen list	Opens the dialogue with the list of the values and the screens

Fig. 16-10: Configuration dialogue for creating the Bitmap list

List	The value pairs of process value and screen name Furthermore the screen itself is shown
Selection of radio button	Here it is determined how the process value has to be compared with the value from the list. The images are only displayed in the image above if the process value is equal to one of the values specified.
Add	The list will be extended by one entry at the end of the list.
Delete	The selected entry is deleted from the list.
Image	Another dialogue opens in order to select a screen from the list of all screens contained in the resource.
Ok	List is being accepted.
Cancellation	Settings are being rejected.

Fig. 16-11: Dialog: List of keys/value pairs

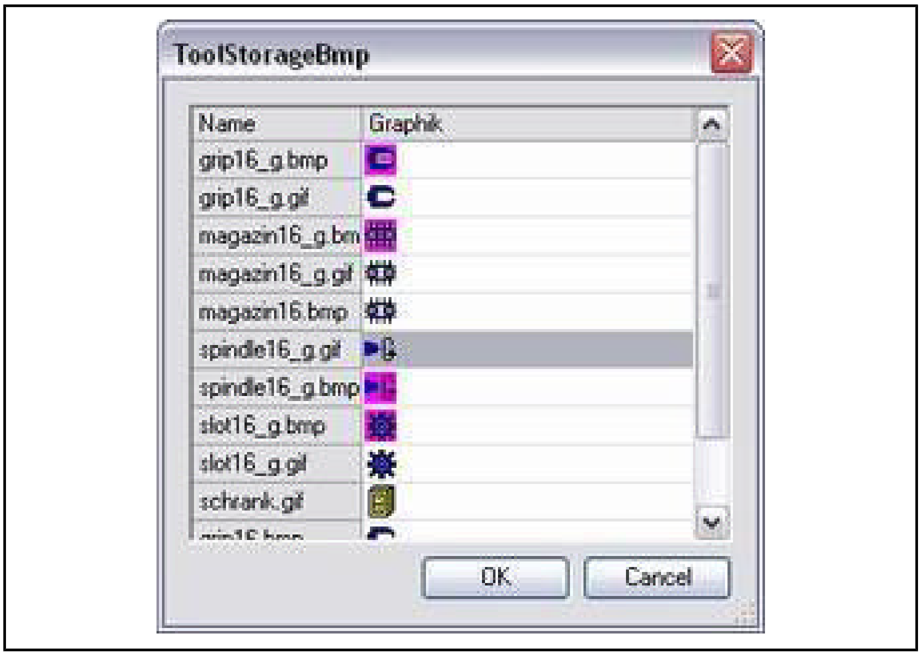


Fig.16-12: List of the screens in the selected resource

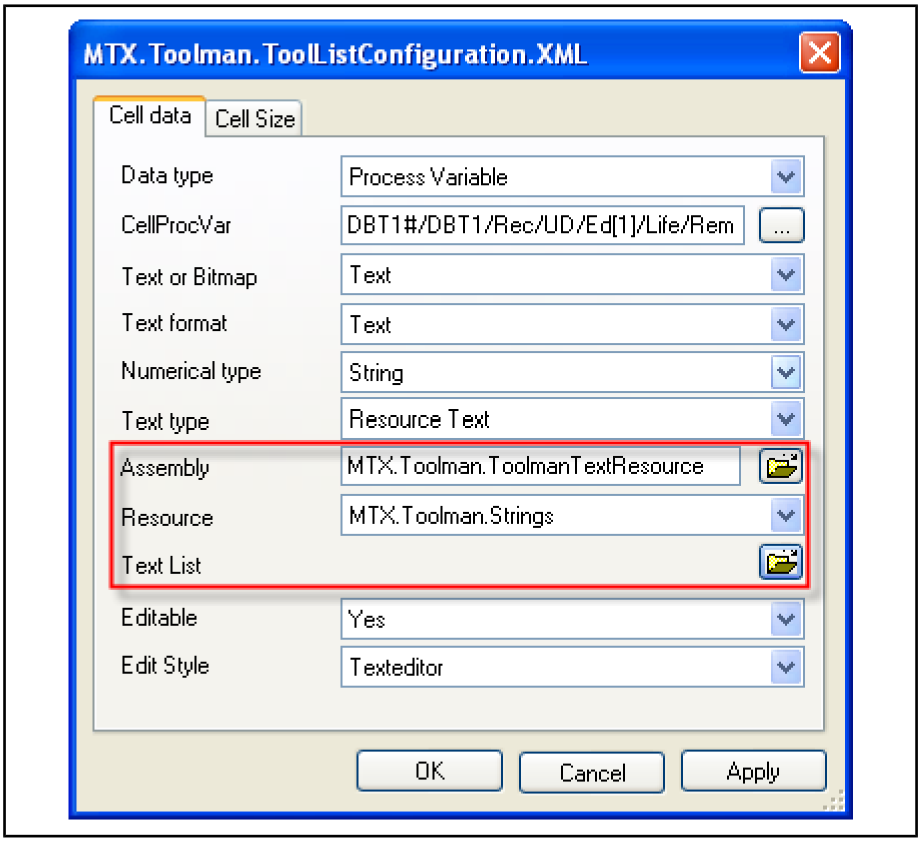
In order to select a screen, place the cursor on the desired screen and click "Ok".

Creating Process-dependent Text Selection

Texts can also be shown depending on the process variables. Therefore, a process value list and the respective text is defined. In the configurator, there is an own configuration dialog for the text list definition. Select the text format "Text" (instead of "Numerical") to enter a list with value pairs. First, it is to be decided from where the texts are taken. There are three options (as for all text definitions): 1. Use text directly, 2. Use text from a DLL (resource text) or 3. Use text from user text file.

If the resource or the user text list should be selected, the "Text List" button opens the dialog for the list creation.

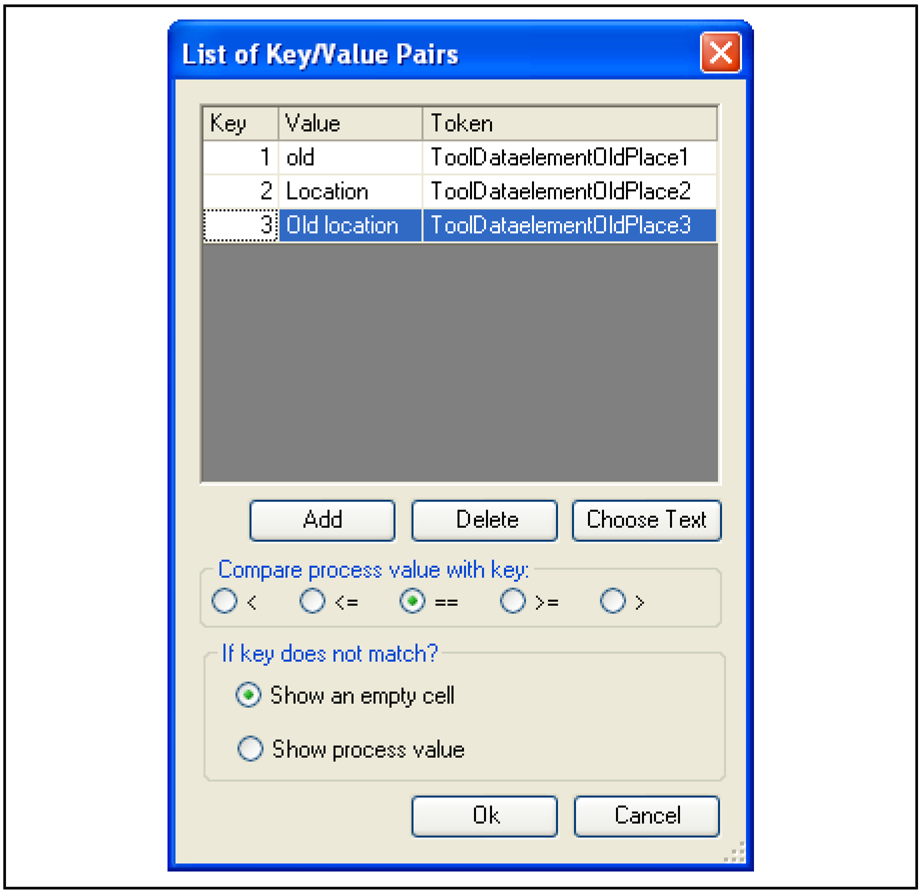
Afterwards, define a "key - value - pair" using the dialog below. In this case, the value corresponds to the value of the variable which is linked to this cell. The value is the text name from the resource, the user text list or the text directly. You can also indicate whether the value has to match the table value exactly or whether it has to be lower or greater.



Assembly	Dialog to select the DLL with the texts
Resource	Combobox with all resources, the Dll
Show text list	Opens the dialog with the list of values and screens

Fig. 16-13: Configuration dialog to create the text list

The dialog to create a text list looks as follows:



Add	Add new line
Delete	Deleting a line
Select text	Select text from resource or from user text
Upper row of the button	Here it is determined how the process value has to be compared with the value from the list. The images are only displayed in the image above if the process value is equal to one of the values specified.
Button "If key does not fit?"	If the process value does not correspond to any of these values, either an empty line or the process value itself can be displayed.

Fig.16-14: Dialog: List of keys/value pairs (text list creation)

Editing a Text List

Text lists are used to select texts from a list and to generate new ones. The texts that are selected can originate from a resource file and can be either texts or images; they can also come from a user text list. The list can be supplemented only if the texts come from a user text list.

Selecting Text A text is selected by double-clicking the mouse.

Adding Text To add a new text, position the cursor over the last cell and enter a token in the first column and a text in the second column.

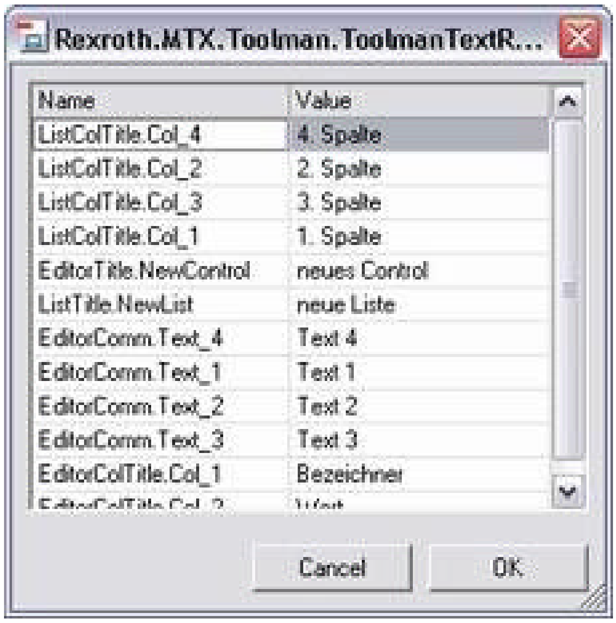


Fig. 16-15: Text list for selecting texts

Handling Instruction: Define a New User Text

The following handling instruction describes the procedure for defining a new user text in several languages.

Creating a User Text in Several Languages

When creating new texts, the project language should always be the master language.

1. Create a new text (see chapter "Editing a Text List" on page 324)
2. **File ▸ Save All**

   A new text token is created.
   If the set project language was not the master language, the text token for the master language is generated automatically with the prefix "@@@@". Otherwise, the tokens for additional installed project languages remain empty; they can be generated externally using the functions **Project ▸ Language ▸ Export translation file...** and finally **Import translation file...** .

3. In order to generate a variant of the new text in another language within IW Engineering, the project language must be switched to the desired language.
4. The text for the new text token is displayed in the master language in the text list. The text can now be modified.
5. **File ▸ Save All**

   The new text is now available in two languages.

Screen		Documentation
Documentation:	MTX Functional Description	

Syntax of Edit Mask

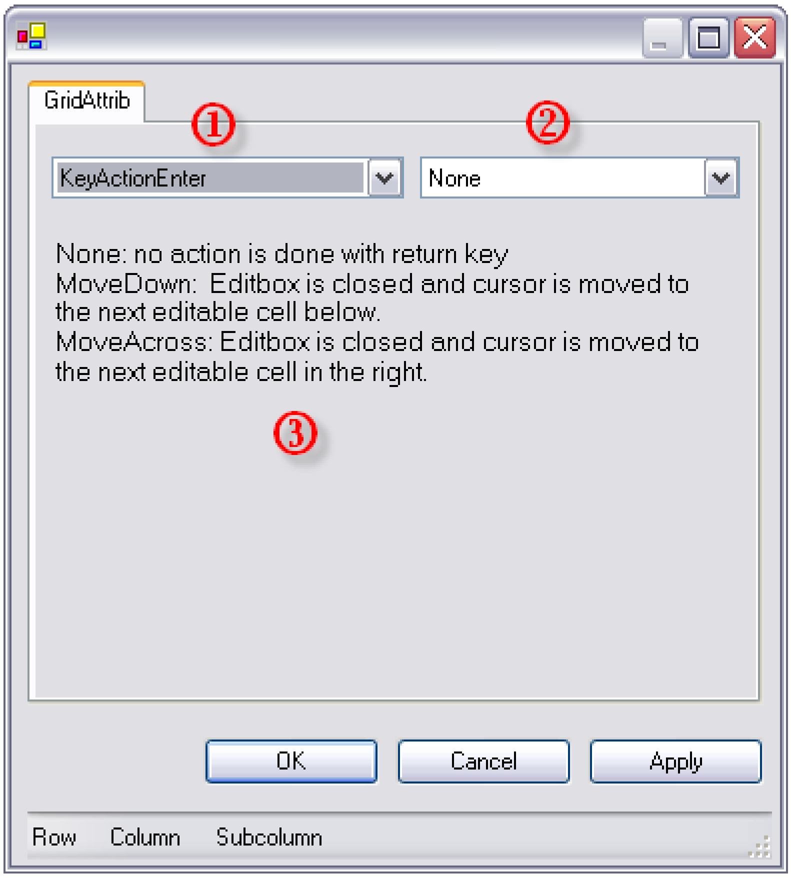
The edit mask must consist of the following characters:

1) Wildcards	
0	Number
9	Number or space
#	Number or character
L	Letter
?	Letter or space
A	Letter or number
a	Letter, number or space
&	Any character
2) Localized characters	
.	Localized decimal point
,	Localized separating symbol for thousands
:	Localized separating symbol for time
/	Localized separating symbol for date
3) Command characters	
****	The next character will be interpreted as literal (directly as a character and not in its meaning)
>	Turn letter into capital letter
<	Turn letter into lowercase letter
4) Placeholder definition	
;	The next character will be used as a placeholder (default: underline)

Example: EditMask = 000.00

Sets a mask that forces a three-digit number with two positions after the decimal point.

Setting of General Table Properties



1 Process parameter
2 Value
3 Comment explaining the setting options

Fig. 16-16: Setting dialog for table attributes

General definitions (GridAttributes):

Element	Description
AllowFocusOnNonEditableCells	True: Cursor jumps to all cells False: Cursor skips all non-editable cells
AllowFreezingWithMouse	Shifting the frozen area: 0 = none 1 = Columns 2 = Rows 3 = both
AllowMerging	True: Allow merging of cells False: Do not allow merging of cells

Element	Description
AllowResizing	0 = none 1 = Columns 2 = Rows 3 = Both 4 = RowUniform 5 = BothUniform
AutoResize	Column width is set automatically when data ex-ist.
AutoSearch	0 = none 1 = FromTop 2 = FromCursor
AllowSorting	0: Sorting not allowed 1: Single column sorting 2: Multi-column sorting
BorderStyle	1 = None 2 = FixedSingle 3 = Fixed3D 4 = Light3D
CheckRowVisibilityAtBeginning	No user setting required or reasonable
CursorKeyCanCloseEditmode	Cursor movements beyond the edge of the edit window terminates the edit mode
DrawTextFlexgridOrg	True: FlexGrid indication mode for texts and bitmaps False: ULC indication mode for texts and bitmaps
ExtendLastCol	True: The last column fills the entire area
FilenameOfDefaultValues	Name of the default setting file
GridHighlight	Defines when selected cells are highlighted: 0: Never 1: Always 2: With focus
KeyActionEnter	Cursor movements after <ENTER>: 0 = none 1 = down 2 = to the right
KeyActionTab	Cursor movement after <TAB>: 0 = none 1 = down 2 = to the right

Element	Description
ListBackGroundColor	Background color of the list (visible only if the indicated list area is smaller than the control).
PageDownTrack	0: Data of the next page are **not** updated while the user presses <PgDn>. 1: Data of the next page are updated while the user presses <PgDn>.
ScrollTrack	0: The list is **not** updated while the user moves the scroll bar. 1: The list is updated while the user moves the scroll bar.
ShowDebugMessages	Trace message output in the debug window
ShowTraceMessages	Display of error messages
SortEmptyRowsToEnd	1: Empty entries are allocated at the end of the list; irrespective of the sorting. Default value: 0
Stripline	Separating line between fixed and scrollable area
SuppressComma	If UseNumberDecimalSeparator = False: True: Comma is replaced by point
TabCanCloseEditMode	TAB terminates edit mode
TestDOMToConfig	No user setting necessary
TestSOMToConfig	No user setting necessary
UseFixedNumberOfSubRows	1: The first sublist is checked and then used to generate the number of all SubRows. Default value: 0
UseNumberDecimalSeparator	True: The separator defined in the current country setting is used False: Point is used generally.

Fig. 16-17: GritAttributes

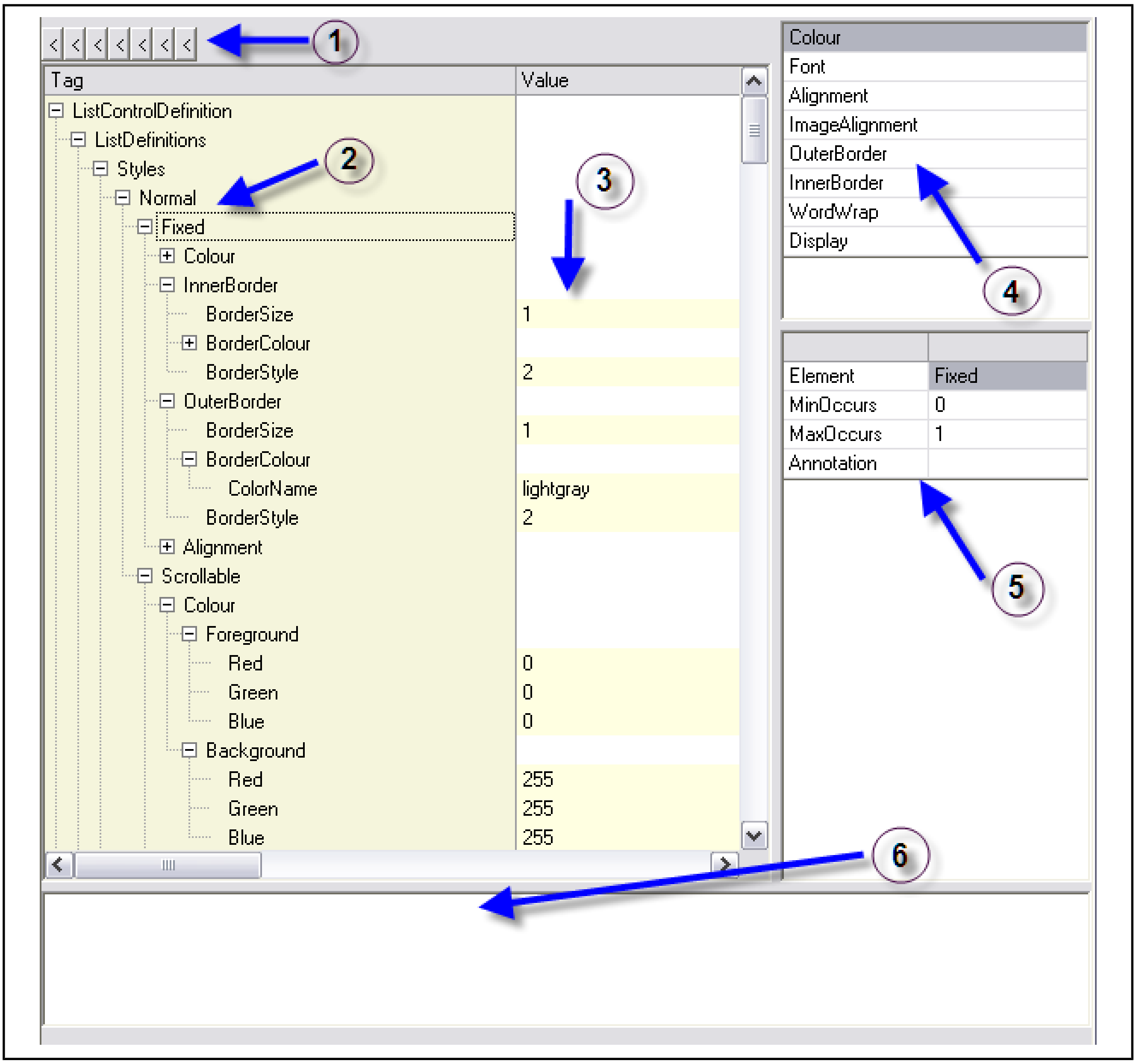
16.2.5 XML File Editor

General

The XML file editor can be accessed via the tab page "XML Editor" of the ULC configurator and can serve as a supplementary view of the tool list and tool editor control configurator (also see chapter 16.2.4 "ULC Configurator" on page 311).

The XML file editor is provided especially for schema-supported editing. The available version is opened only to configure tool lists and tool editors.

The advantage of schema support is that nodes defined in the schema can be selected from a list; they do not have to be entered separately. In addition, values that are entered are checked whether they are appropriate for the schema.



1	Button for expanding/collapsing nodes
2	Node tree
3	Value list
4	List of possible children of the selected node
5	Attributes and comments for the selected node
6	Output window

Fig. 16-18: The tool list XML editor

The editor provides the following possibilities:

Pasting Known Nodes

When an element in the list of possible node children (4) is double-clicked, this element is pasted as a child in the selected node.

Editing a Value

1. In the list of values (3), select a cell and double-click or press <Return> to switch to the edit mode.
2. Enter the value and confirm with <Return>.

If the selected node is defined in the schema, the entry is monitored. Only valid values may be entered.

Search for Strings

Press <Ctrl>+<F> or click with the right mouse button on "Find" to activate the search window, with which you can search for strings in the document.

Copying Nodes

Select a node and, with the right mouse button, click "Copy Node". Now the node, with all the subnodes, is copied as a partial XML tree to the clipboard. From there, one can make further use of it. For example, you can copy the node into a Text Editor, modify it there, copy it back to the clipboard and paste it back into the document.

Pasting Nodes

If a node was copied to the clipboard previously, it can be pasted anywhere. This procedure is not monitored by the schema! Select the node which is supposed to become a "parent" and, with the right mouse button, open the pop-up menu **Paste Node**

Moving Nodes Up

Select the node, press the right mouse button and select "Move Node up" in the pop-up menu.

Moving Nodes Down

Select the node, press the right mouse button and select "Move Node Down" in the pop-up menu.

Deleting Nodes

Select the node and use the <Del> key or press the right mouse key and select the "Delete Node" in the pop-up menu.

Pasting Nodes with Freely Definable Names

Select the node, press the right mouse key and select "Add New Node" in the pop-up menu. This pastes a node whose name is freely definable. This name does not have to be defined in the schema and can lead to errors during validation.

Pasting a Text Node

A value can be assigned to nodes that do not have subnodes as children themselves (end nodes). This occurs using a text node. Select the node, press the right mouse key and select "Add New Text Node" in the pop-up menu. The text node that was just created can now be filled with any text.

Copying XPath to the Clipboard

Select the node, press the right mouse key and select "Copy XPath to Clipboard" in the pop-up menu. The path from the basic node to the selected object can now be found in the clipboard.

Opening and Closing the Individual Node Levels

For every node level, a button with which the corresponding node level can be opened or closed is created when the XML document is loaded. If a subordinate level is expanded, all superordinated nodes are also opened.

16.3 General DB Configuration

16.3.1 Number of the DBT Used

General

All general settings for the database visualizations (e.g. tool management) are made via the "Properties" dialog which can for instance be called via **DBTx Screens (tool management) ▸ Properties...**

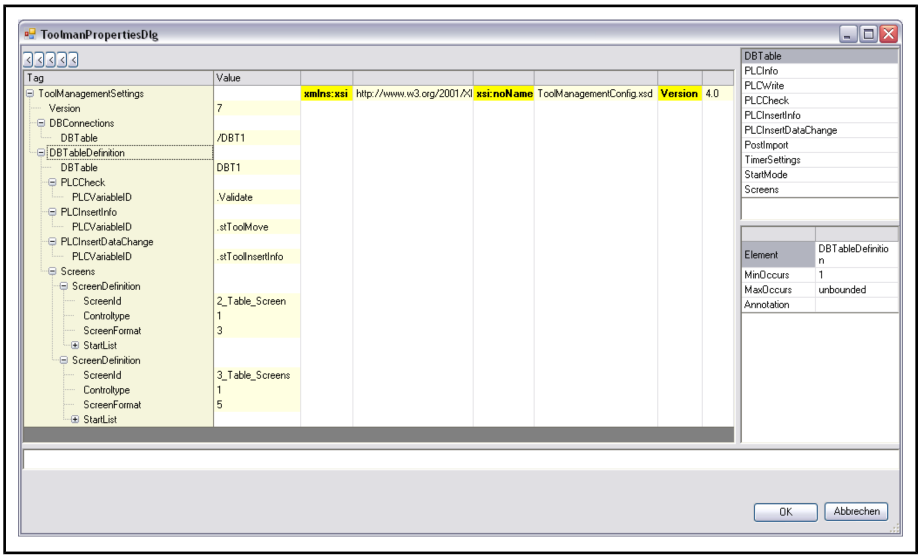


Fig. 16-19: Tool management

2. Activating database table Currently, a maximum of two database tables can be used.

If the second database table should be used, respective settings have to be made in the file "ToolManagementConfig.xml" .

First, activate the communication in the second database table:

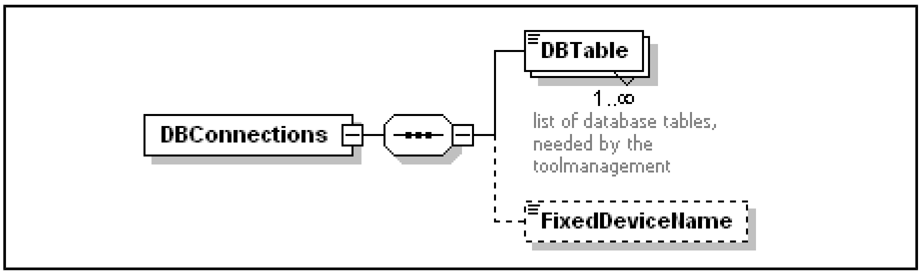
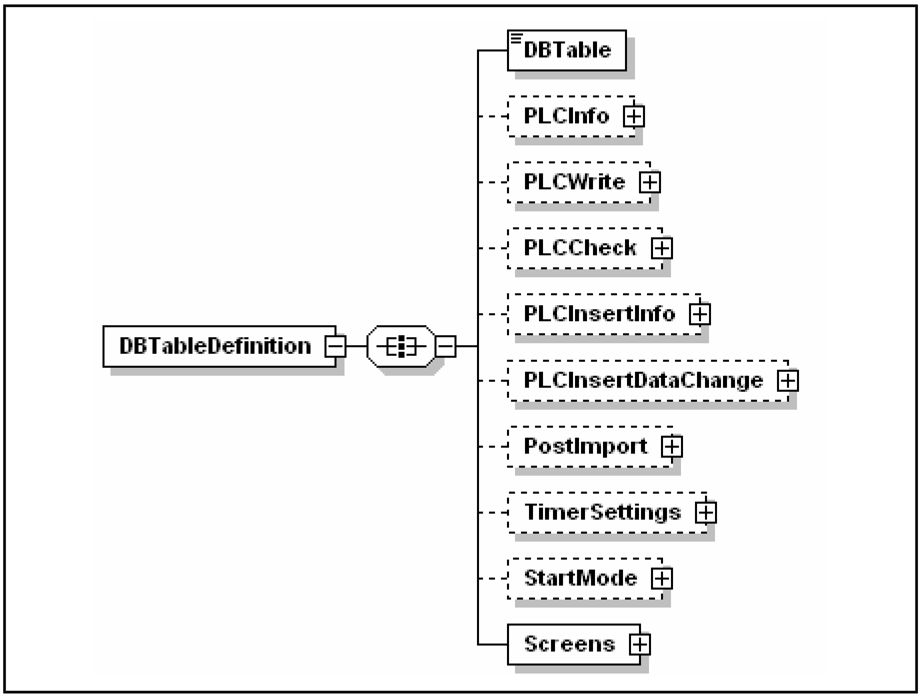


Fig. 16-20: Activating the second database table

If these entries are missing under **DBConnections**, it is only communicated with the database table 1.

Furthermore, if applicable, all functional basic settings should be made for DBT2 as made for DBT1.



DBTable	ID entry
PLCInfo	Definition of the PLC variable name for the Info function
PLCWrite	Definition of the PLC variable name for the Write function
PLCCheck	Definition of the PLC variable name for the Validate function
PLCInser-tInfo	Definition of the PLC variable name for the InsertInfo function
Post import	Definition of the Post Import function
TimerSet-tings	Low-pass filter configuration:
StartMode	Setting the edit mode used for starting
Screens	Contains screen definitions for DBT

Fig.16-21: Structure diagram 1: ToolManagementConfig.xml

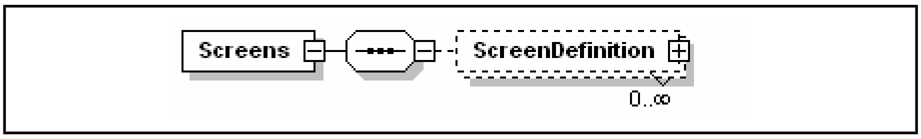
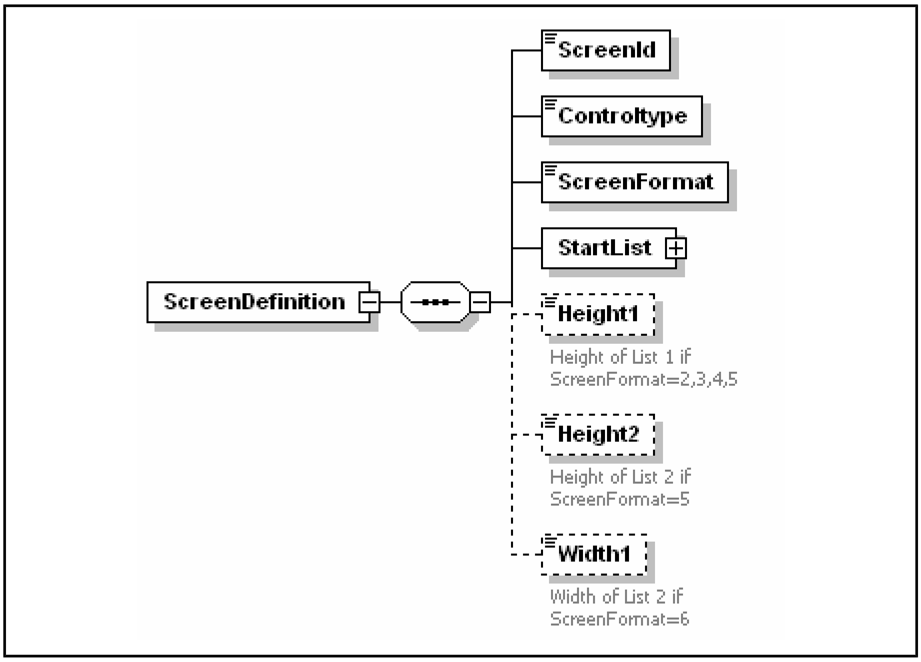


Fig.16-22: Structure diagram 2: ToolManagementConfig.xml



ScreenID	is automatically specified
Controltype	is automatically specified (see chapter "General" on page 335)
ScreenFormat	is automatically specified (see chapter "General" on page 335)
StartList	defines the starting list of the screen
Height1	Height of list 1 (relevant for manual size setting of the lists for ScreenFormat 2,3,4,5)
Height2	Height of list 2 (relevant for manual size setting of the lists for ScreenFormat 5)
Width1	Width of list 1 (relevant for manual size setting of the lists for ScreenFormat 6)

Fig. 16-23: Structure diagram 3: ToolManagementConfig.xml

Creating a DBT2 screen in IndraWorks Engineering

To visualize the DBT2 data in the IWO, it is possible to define an own visualization screen. Furthermore, it is still possible to call and visualize DBT2 lists within the operating area tool management.

Handling Instruction: Configuration to Use the Second Database Table

The following handling instruction describes the process to be followed when further database tables are to be used.

Activate communication with the second database table

1. Open the "Properties" dialog of the database applications via **DBT1 Screens (tool management) ▸ Properties...**.
2. Add the entry for the second database table under the node **"<DBConnections>"** (refer to the following Fig.).

Program:

```
<DBConnections>
    <DBTable>/DBT1</DBTable>
    <DBTable>/DBT2</DBTable>
</DBConnections>
```

Basic setting definitions of the management for the second database table

1. Open the "Properties" dialog of the database applications via **DBT1 Screens (tool management) ▸ Properties...**.

2. Add another node for the DBT2 in the same way as for partial tree **"<DBTableDefinition>"** for the DBT1 and enter all necessary entries for the second database table under this node. (see example in following Fig.)

Program:

```
<DBTableDefinition>
    <DBTable>DBT2</DBTable>
    <PostImport>
     <Action>
      <ProcVariableID>Hd/BQ2</ProcVariableID>
      <Operation>AND</Operation>
      <Value>FFFFFF7F</Value>
  </Action>
 </PostImport>
    <PLCCheck>
     <PLCVariableID>.Validate</PLCVariableID>
 </PLCCheck>
...
</DBTableDefinition>
```

Activation of DBT2 in the project and setting up the user interface configuration

1. Close the project to apply the changes.
2. Open the project again in IndraWorks Engineering.

   The database table 2 in the project tree does now have an individual node. Under this node all interface settings for the DBT2 can be made.

	Documentation	
Documentation:	MTX Functional Description	Transferring the new data structure

16.3.2 Configuration of Several DBT Visualizations

General

There is the possibility of visualizing the database tables in one or in several applications. This applies both for database table 1 which is exclusively intended for the management of tool data and for database table 2 which also allows for the management of other user data.

The configuration of the database is the same for all database tables (refer to chapter 16.4 "Configuring the Database" on page 341).

The visualization of the data of one database table requires at least one visualization screen per DBT. For the DBT1 there is always a "Default ToolmanScreen". In the tree its node is presented in gray, since the settings for this screen cannot be changed. Further screens can be defined for a DB table via the menu function **New Screen...**. Depending on the node under which the new screen is created, the DBT allocation of the latter takes place.

In the following dialog all further configuration steps for the screen definition are made:

- Specifying the application type
- Specifying the list number and the display format
- Specifying the start lists

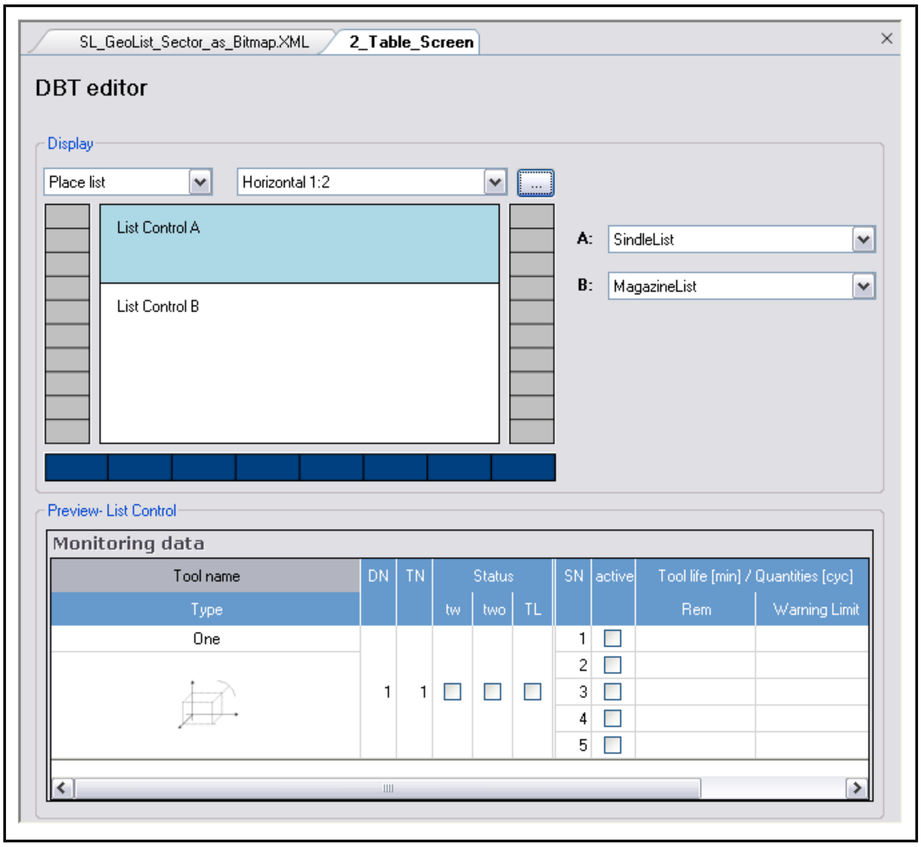
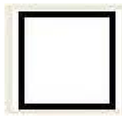
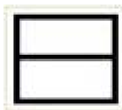
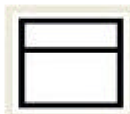
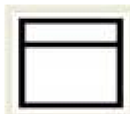


Fig. 16-24: Screen configuration

Display format	Number of lists	Properties	ScreenFormat
Default	1		1
horizontal 1:1	2		2
horizontal 1:2	2		3
horizontal 1:3	2		4

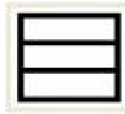
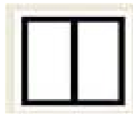
Display format	Number of lists	Properties	ScreenFormat
horizontal 1:1:1	3		5
Vertical 1:1	2		6

Fig.16-25: Display format

If the database tables are used in different operating areas - which in turn requires the definition of different applications or screens - these applications can have the following properties.

Application type	Example application	Properties
as place-related data record management **Controltype = 1**	Tool management Workpiece management	• SKQ as a criterion for assigned location • derived from this, the F-key and menu functions are placed actively or inactively • Evaluation of the PD attribute in the schema
as universal data record management **Controltype = 2**	any data container (does only make sense for DBT2)	• All data elements are equal without downstream display and function logic. • The attributes "UA","SE" and "PD" in the schema do not have any function. • The data element BQ3 for the type-related display control is free and can be made further use of as bit array. • As opposed to the system data, the DBT2 has the effect of a data container in sectors with predefined memory and access functions typically for a database.

Fig.16-26: Application types for database tables

There is still the possibility of visualizing both database tables via different lists in one screen.

Handling Instruction: Configuration of Several Different Applications

The following handling instruction describes the process to be followed when visualizations of database tables in different operating areas are to take place.

Creating a new DBT Screen

1. Open IndraWorks Engineering
2. Select the node **"Screens"** in the project tree under the HMI device.
3. Select **New screen...** in the menu.
4. Define the screen name in the following dialogue.

   A new node will appear in the tree with the name of the new screen.

Setting the Properties of the new DBT Screen

1. Select the node of the new screen in the tree and press <ENTER> or double-click on this node.

   The screen editor for the new picture is opened.

2. In the selection box "Screen type" the type **"MTX ToolList"** is selected.
3. In the selection box "Table" the application type is selected (refer to fig. 16-26 "Application types for database tables" on page 337).

   Subject to the selection, the following appears in the field "Table":

Selection	Entry
DBT1 place list	DBT1 ;1
DBT2 place list	DBT2 ;1
DBT1 general list	DBT2 ;2

4. In the selection box "Operating area" the application is allocated to an operating area.

F or M-key configurations should not be made, since these take place when defining the list.

Call Definition via F-key or M-key

1. Selection of the corresponding F-or M-keypad of the operating area which the application has been allocated to and which contains the key which is provided for the call of the application
2. Opening the selected panel.

   F or M-key configurator is being opened.

3. Selection of the corresponding key and definition of the key labeling.
4. In the selection box "Function" **"Screen change"** is selected.

   The selection box "Screen name" becomes active.

5. The name of the newly created DBT application is selected in the selection box "Screen name".
6. Close editor and save change.

	Documentation	
Documentation:	MTX Functional Description	Transferring the new data structure

Configuring Multiple-table Screens

Handling Instruction: Creating a multiple-table screen

Creating a multiple-table screen

None

1. Create a new tool screen or DBT screen via <New Screen>.

   Depending on the node under which the new screen is created, the DBT assignment of the latter takes place.

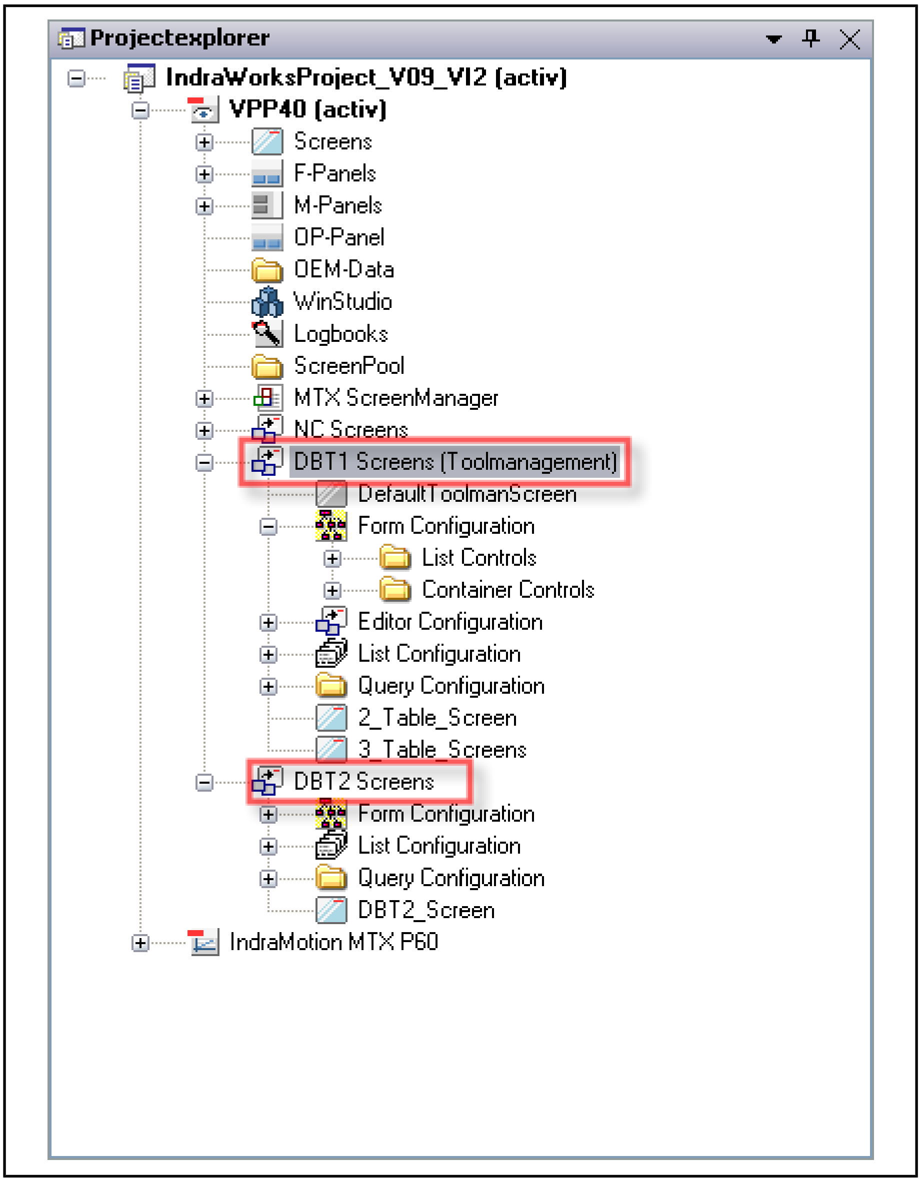
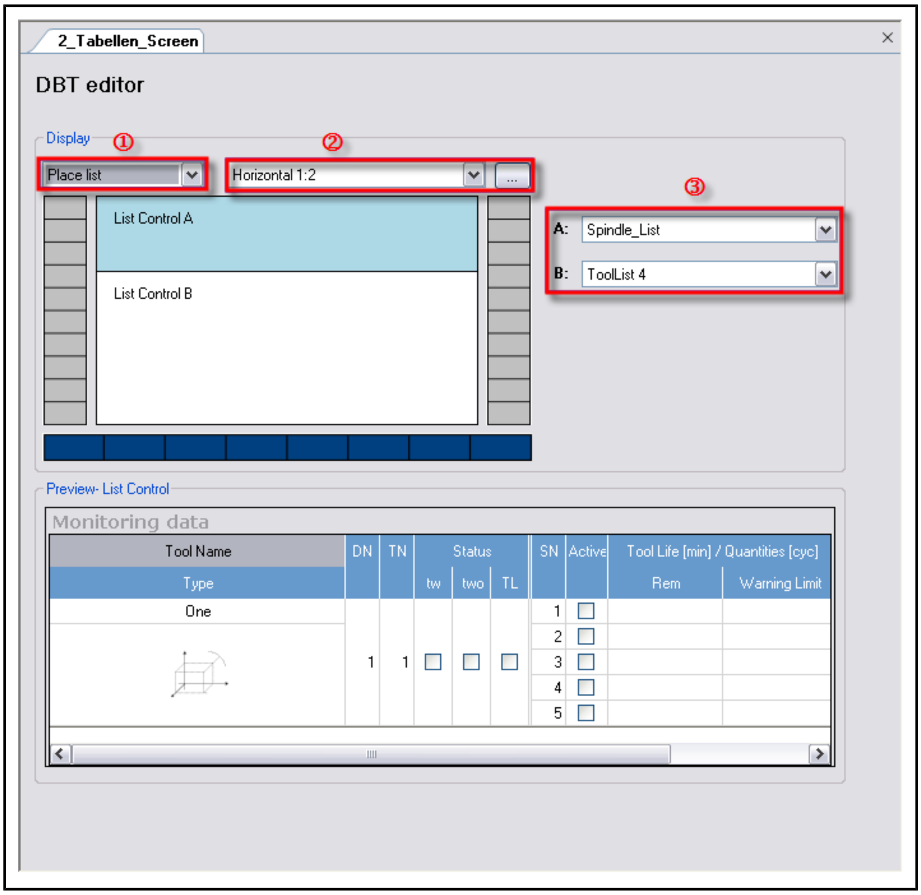


Fig. 16-27:

New entry under "DBT1 Screens (Tool Management)" or "DBT2 Screens"

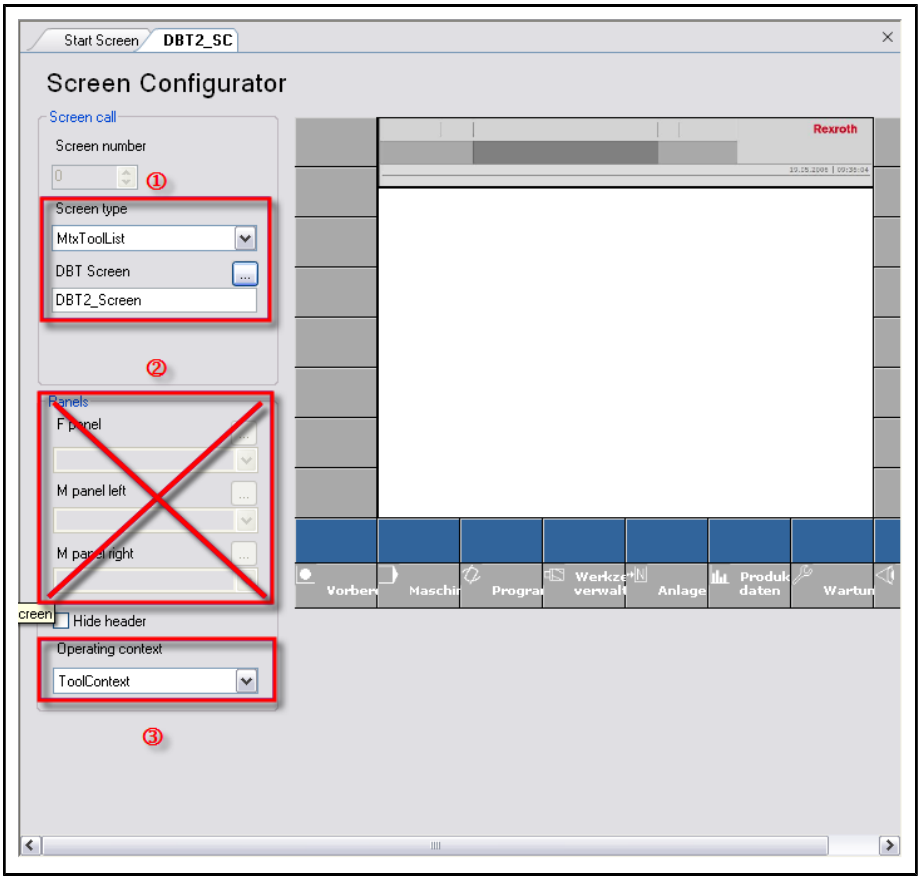
2. Configure new screen



1: Determine list type
2: Determine display format
3: Specify starting list(s)

Fig. 16-28: Configure new screen

3. Identify new screen for HMI device and assign to operating area.



1: Specify image type
2: Please do not execute any panel definitions since these are executed when defining the list
3: Assign operating area

Fig. 16-29: Screen definition

4. Opening a new screen via an F-key in the selected operating area.

16.4 Configuring the Database

16.4.1 General

Settings for control reset in case of changes made to the database configuration:

Changes in:	Necessary action:
dbtxprms.dat	Control reset with restart mode 6
Changing the element structure in the tool data schema	Control reset with restart mode 6
Changing attributes/restrictions in the tool data schema	Control reset with restart mode 0

Fig. 16-30: Reset settings

☞ A control reset with the restart mode 6 and control reset with the restart mode 0 and parameter 2 = 6 in the file "dbtxprms.dat" always leads to the deletion of all data records of the database table.

16.4.2 Defining the Sector and Location Assignment of the Database Table

General

The sector-place structure of the database table is specified as follows in the configuration file. A specific file exists for each database table: for DBT1 the file **"dbt1prms.dat"**, for DBT2 the file **"dbt2prms.dat"**.

Program:

```
; -----------------------------------------------------------
;       Configuration file for the DB table DBT1
; -----------------------------------------------------------
; general design:
; P0 P1 P2 P3 P4 P5 P6
;  |  |  |  |  |  |  |
;  |  |  |  |  |  |  +--- string: may also contain " " and \t
;  |  |  |  |  |  |               End identification is "line end"
;  +--+--+--+--+--+------ 6 int values
; P0 describes the object type and may not be modified
; Empty string marked with "."
; ";" marks comment lines At the end of a data line
; No comment must be added
;
;       ****************************
;       Configuration of the DB table
;       ****************************
;
; Control parameter
; P0: 0
; P1: Controls the creation of the data records in the database
;       0  Standard operation
;          Data records are only created when the tool database
;          is empty
;       6  All existing data records are deleted first and;
;          then created again according to the
;          sector configuration
;
0 0
; Sector configuration
; P0: 1-99 (corresponds to sectors 1 - 99)
; P1: Number of places in the sector
1 5
2 5
```

Changing the sector and place distribution becomes only effective after a control reset with mode 6.

The existing content of the database is deleted and cannot be restored! If the existing tool data will be required in the future, export the data before making any change to the database. If the number of data records is smaller than that of the data records included in the export file, this file must be adjusted by means of a suitable XML editor.

Handling Instruction: Defining the Sector and Location Assignment of a Database Table

This handling instruction describes how to modify the sector and location division for a database table.

IW Operation / program: Edit the File "dbt?prms.dat"

1. Copy the file **"dbt?prms.dat"** (?:= 1 [DBT1] or 2 [DBT2]) to the mount directory (\mnt) in the control directory **"\usrfep"** or **"\feprom"**.
2. Edit the file via the editor; enter the desired values for the number of sectors and locations (see the program in chapter "General" on page 342).
3. Save file.
4. Copy the file to the **"\usrfep"** control directory.

	Flowchart	Documentation
Flowchart		Flowchart fig. 1-7
Documentation:	MTX Functional Description	Define data records

NC: Data Transfer

1. Close IW Operation.
2. Start the control with "start up mode 6". To do so, start the MTX control.

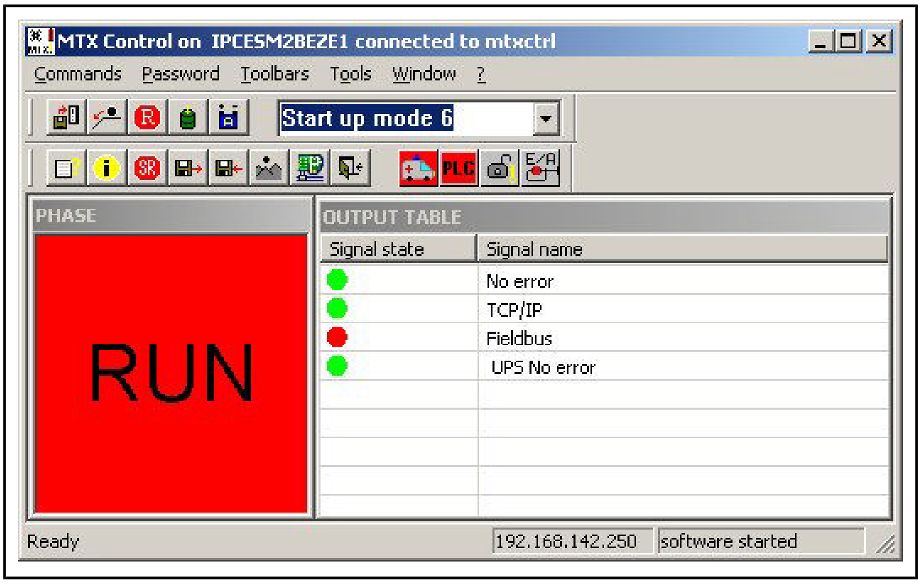


Fig. 16-31: Dialog: MTX Control (Start up mode 6)

- Before the next start, set "start up mode 0" in the MTX control.
- All database contents are deleted in the case of "start up mode 6".

Startup with mode 6 for the MTX Emulation:

- Exit emulation.
- Delete file "typ3ram.pxf" in the working directory of the emulation.
- Start emulation.

⚠ WARNING

⇒ It is absolutely necessary to execute an NC data backup and to reimport the data afterwards.

	Flowchart	Documentation
Flowchart		Flowchart fig. 1-7
Documentation:	MTX Functional Description	Define data records

16.4.3 Configuration of Data Records

General

Since the current version does not yet have a "Settings" dialog to configure the user interface, the corresponding schema or XML document files must be adjusted by means of a suitable editor (for example, see chapter 16.2.3 "Schema Editor" on page 307). This documentation proceeds on the assumption that the universally available Notepad editor is used.

For describing the tool data record schema, the following XSD schema files are installed by default in the FEPROM/schema range of the PNC. To adjust them, these schema files should be copied into the Userfep/Schema directory. If there is no such directory yet, the user should create it at this point.

The files can only be edited in the Mount directory or another Windows directory. Then the changed schema files should be copied into the Userfep/Schema directory. The changes take effect in the control kernel after control restart.

The tool data structure is described by means of a hierarchically structured system of XSD files:

dbt1sd.xsd	Tool data system structure
dbt1ud.xsd	Tool user data structure
tool_ty.xsd	Tool data type collection
basic_ty.xsd	Basic data type collection

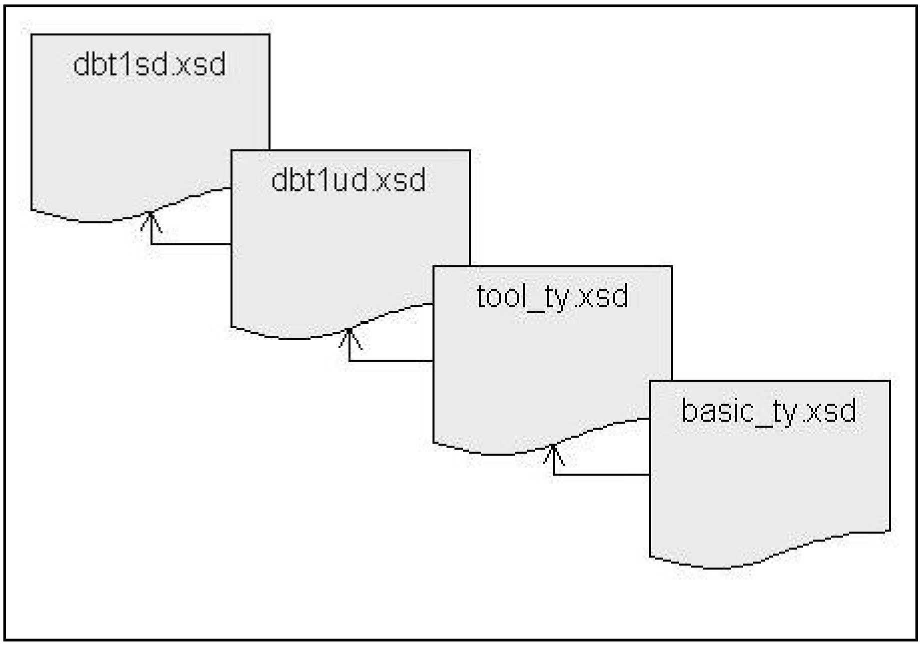


Fig.16-32: Schema configuration (delivery state)

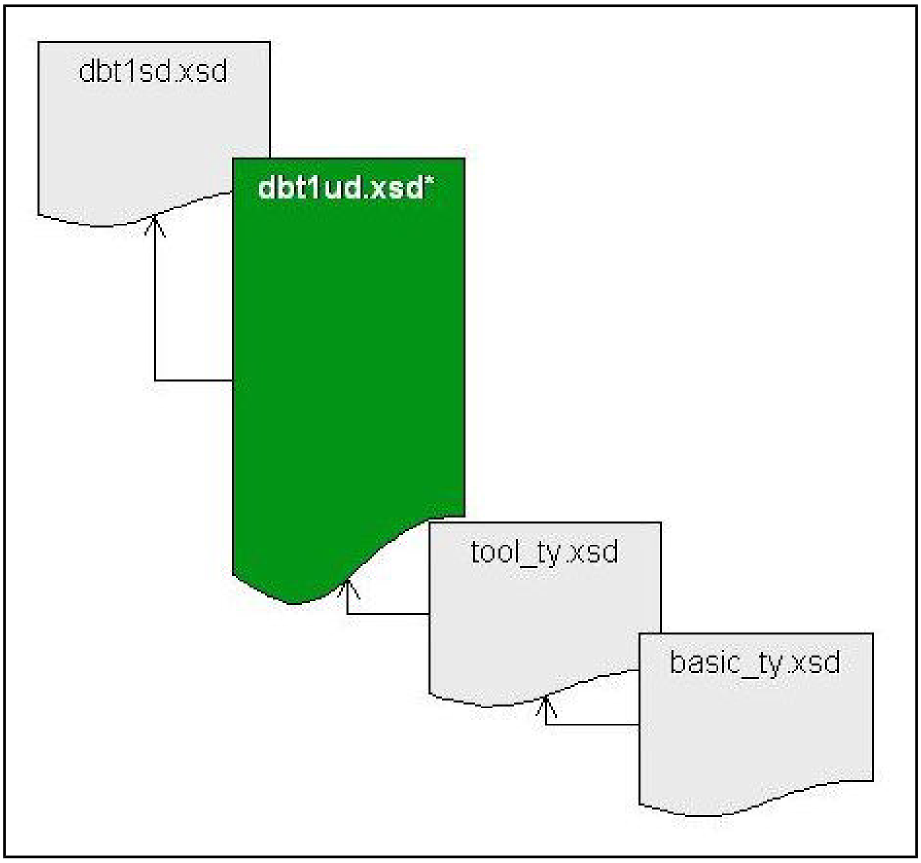


Fig. 16-33: Modified schema configuration (recommended variant)

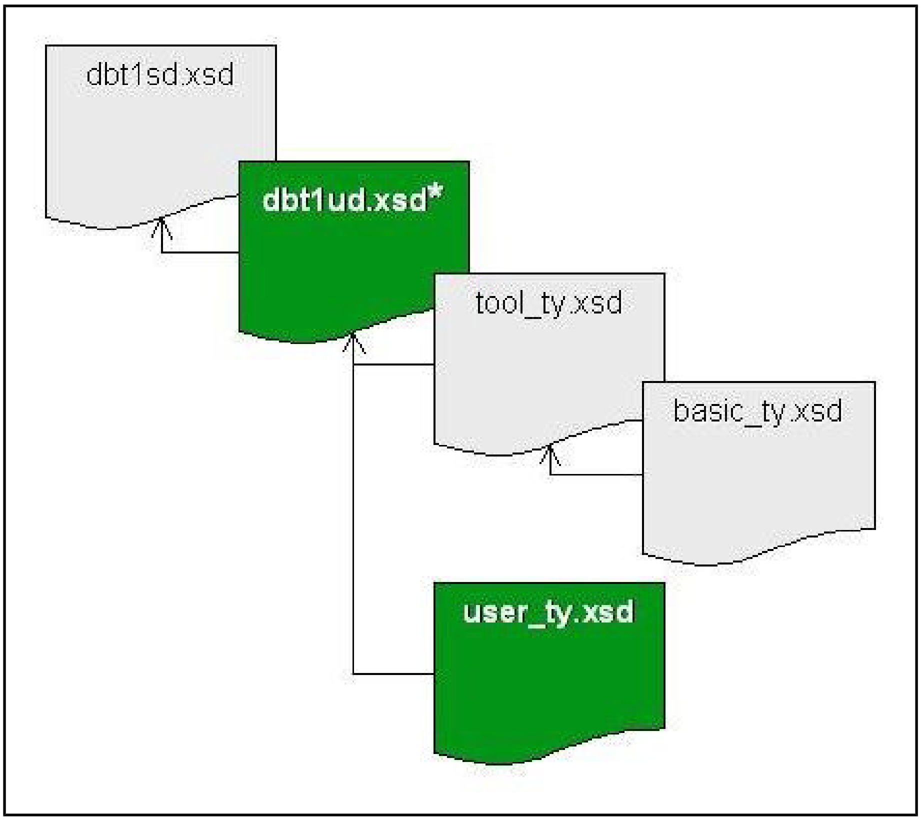


Fig. 16-34: Modified schema configuration

Ideally, the user just has to extend the user data structure in the "dbt1ud.xsd" file.

Tool System Data Structure (dbt1sd.xsd)

This file describes the system structure permanently preset in the MTX for all tool management systems implemented with database table 1. The "DBT1Hd_t" data type comprised in this file is a standard data type preset by the system. It contains a series of basic tool data which are used by the system functions, e.g. the tool catalog, and standard functions, such as location and tool search functions, etc. For this reason, the user must not change the data structure of this schema.

Include	**loc:dbt1du.xsd**
element	DBT1
complexType	DBT1Rec_t
complexType	DBT1Hd_t

Fig. 16-35: Contents of the system structure

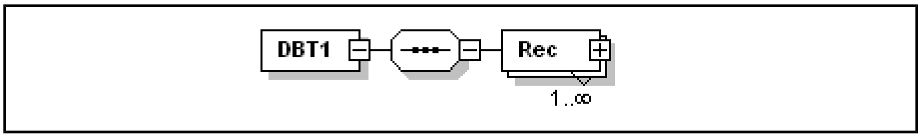


Fig. 16-36: Data record schema (1)

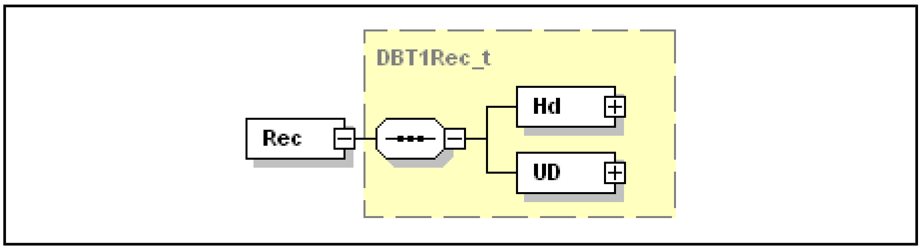


Fig. 16-37: Data record schema (2)

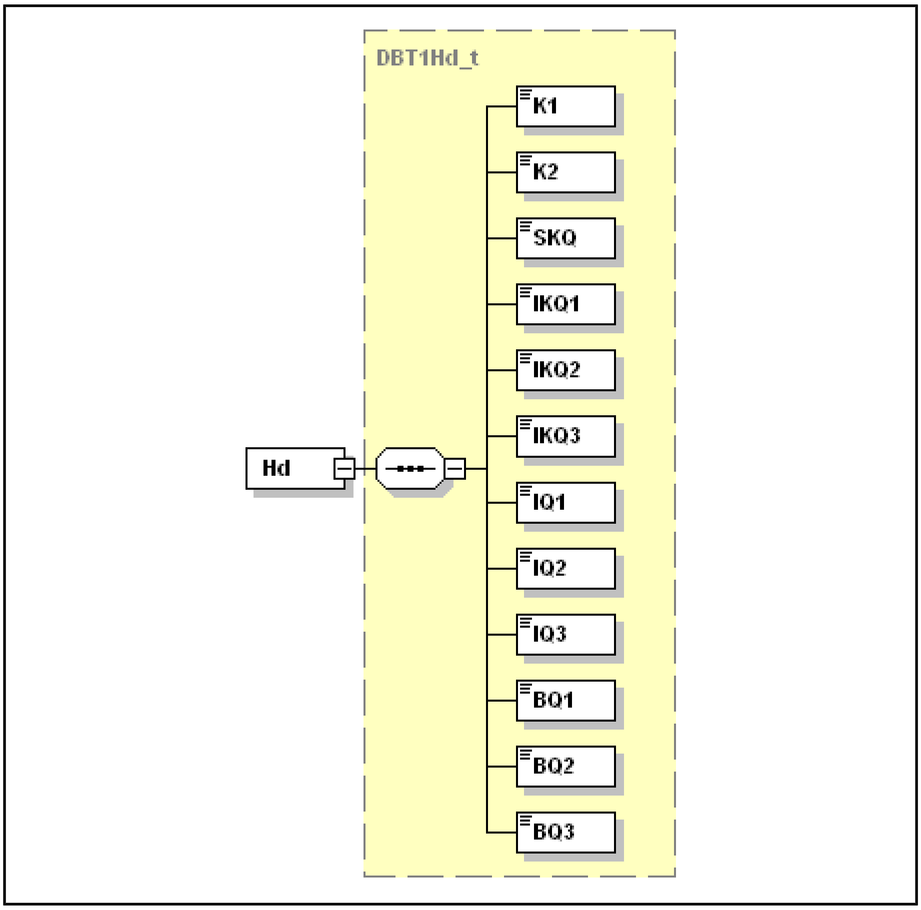


Fig. 16-38: Data record schema (3) (header schema)

The significance of the following header data elements is permanently defined for the tool data record:

K1:	Sector (memory)	Integer
K2:	Place	Integer
SKQ:	Tool ID	String
IKQ1:	Duplo number	Integer
IKQ2:	Tool type	Integer
IKQ3:	Tool number	Integer
IQ1:	Free (reserved for multi-table system)	Integer
IQ2:	Free (reserved for multi-table system)	Integer
IQ3:	Free (reserved for multi-table system)	Integer
BQ1:	Location status	Bit array
BQ2:	Tool status	Bit array
BQ3:	Tool type description	Bit array

Tool User Data Structure (dbt1sd.xsd)

Using the parts structures predefined in the tool data type collection, this file describes the parts of the tool data structure the user needs to modify. In the current case, the user structure consists of:

		Type > V09:	Type > V04	Type in V02
Tool basic data structure	Tl	Tl_V09_t	Tl_V04_t	Tl_t
Tool edge data structure	Ed	Ed_ V09_t	Ed_ V04_t	Ed_t

Fig. 16-39: Tool user data structure

These types are defined in type collection "tool_ty.xsd".

By means of the entry <xs:element name="Ed" maxOccurs="2"> the maximum number of tool edge data structures and, thus, the maximum tool edge number is defined in the data record.

Include	loc:tool_ty.xsd
complexType	DBT1Ud_t

Fig. 16-40: Contents of the user data structure

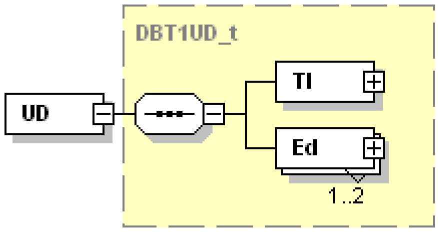


Fig. 16-41: Data record schema (4) (user data structure when delivered)

Changing the Number of Tool Edges

The tool edge number can be changed by means of the following input (maxOccurs) in this schema file:

Example:

Number of tool edges = 2:

Program:

```
<xs:complexType name="DBT1UD_t">
  <xs:sequence>
    <xs:element name="Tl" type="Tl_t"/>
    <xs:element name="Ed" type="Ed_t" maxOccurs="2"/>
  </xs:sequence>
</xs:complexType>
```

The default tool data record was defined for two cutting edges max. If this number should be changed, the following system data setting must be adjusted in the **SDDat.xml** file to ensure a correct display of the active tool correction in the operating area "Machine".

Program:

```
...
  <SysMaxEd>
    <MaxEd>2</MaxEd>
  </SysMaxEd>
...
```

Tool Data Type Collection (tool_ty.xsd)

This schema file includes a collection of tool data part structures, predefined in the MTX as a standard, which act as devices for defining the tool user data structure. Please note that all data elements in the MTX not contained in the header must be viewed as user data. This type collection has been extended in the course of further developing the MTX version 04; for reasons of compatibility, it contains redundant data types, which are marked in the following table by "for V02 default projects only".

If legacy projects are used without modifications, the following modification must be made in file "dbt1ud.xsd":

The following lines

...

<xs:element name="Tl" type=**"Tl_V09t"**/>

<xs:element name="Ed" type=**"Ed_V09_t"** maxOccurs="2"/>

...

are replaced by

...

<xs:element name="Tl" type=**"Tl_V04_t"**/>

<xs:element name="Ed" type=**"Ed_V04_t"** maxOccurs="2"/>

...

or

...

<xs:element name="Tl" type=**"Tl_t"**/>

<xs:element name="Ed" type=**"Ed_t"** maxOccurs="2"/>

...

Now the modified file must be located in directory usrfep\schemas.

To use new functions, the schemas must be supplemented in accordance with the standard of version 04.

But if the new structure is used, any PLC and CPL programs from version 02VRS must be adapted.

Include	loc:basic_ty.xsd	Remarks
complexType	Tl_t	For V02 default projects only
complexType	Tl_V04_t	New default type as of V04
complexType	Tl_V09_t	New default type as of V09
simpleType	LifeUnit_t	
complexType	Ed_t	For V02 default projects only
complexType	Ed_V04_t	New default type as of V04
complexType	Ed_V09_t	New default type as of V09
complexType	Geo_t	For V02 default projects only
complexType	Geo_V04_t	New default type as of V04
complexType	Wear_t	
complexType	EdLife_t	For V02 default projects only
complexType	EdLife_V04_t	New default type as of V04
complexType	Limits_V04_t	New default type as of V04

Fig. 16-42: Contents of the tool data type collection

Only the data structure of the new default projects as of V04 is described below.

Configuring the Tool Management

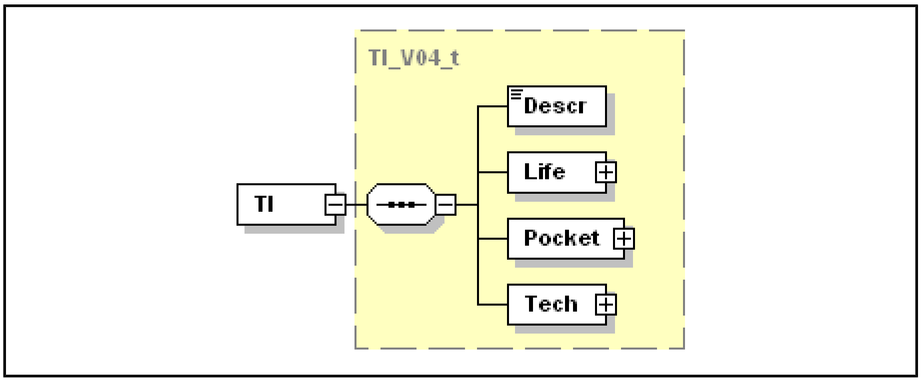


Fig.16-43: Data record schema (5) (tool data upon delivery)

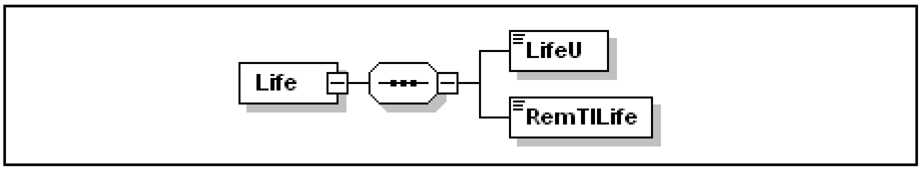


Fig.16-44: Data record schema (6) (tool life data)

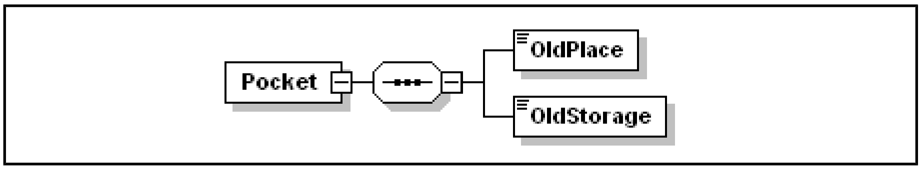


Fig.16-45: Data record schema (7) (tool data)

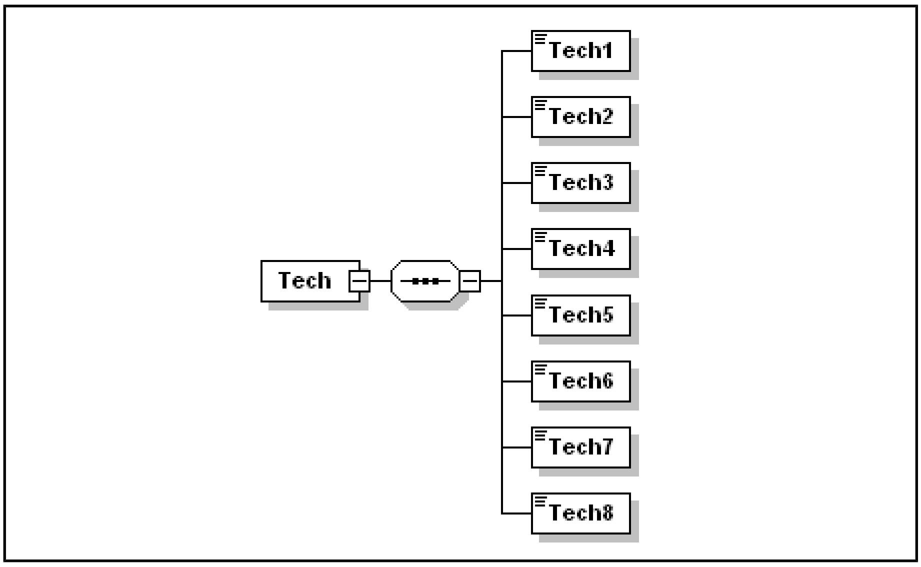


Fig.16-46: Data record schema (8) (tool technology data)

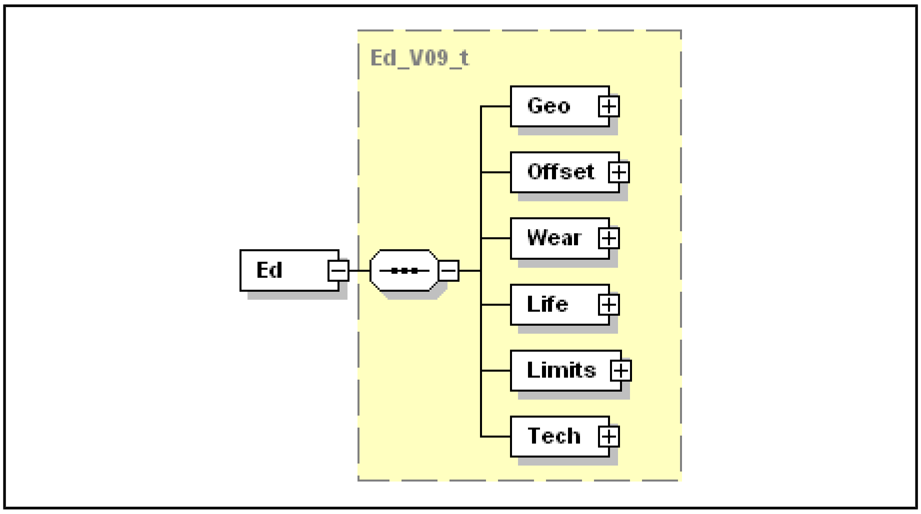


Fig.16-47: Data record schema (9) (tool edge data upon delivery)

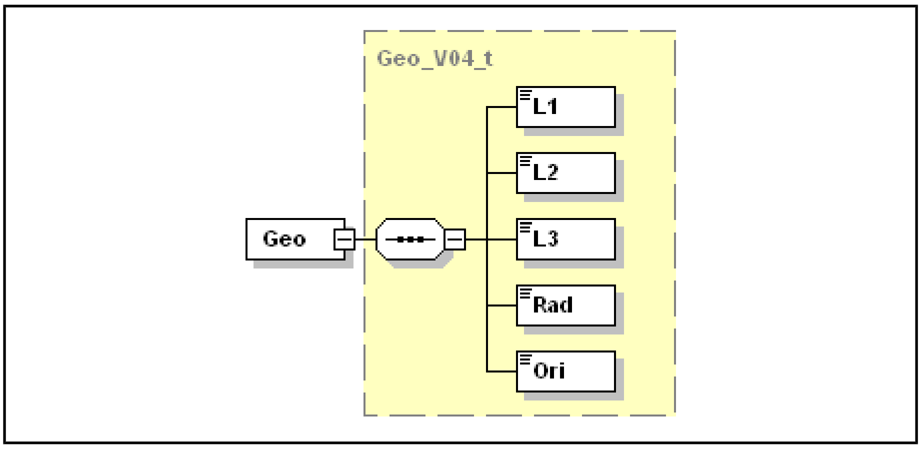


Fig.16-48: Data record schema (10) (geometry data upon delivery)

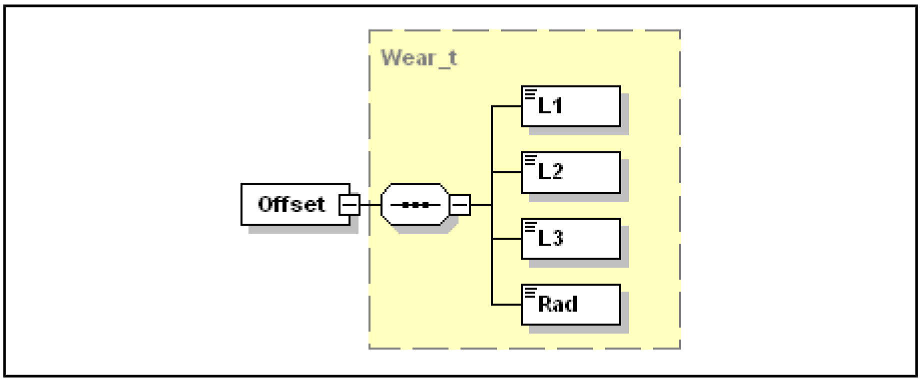


Fig.16-49: Data record schema (11) (offset data upon delivery)

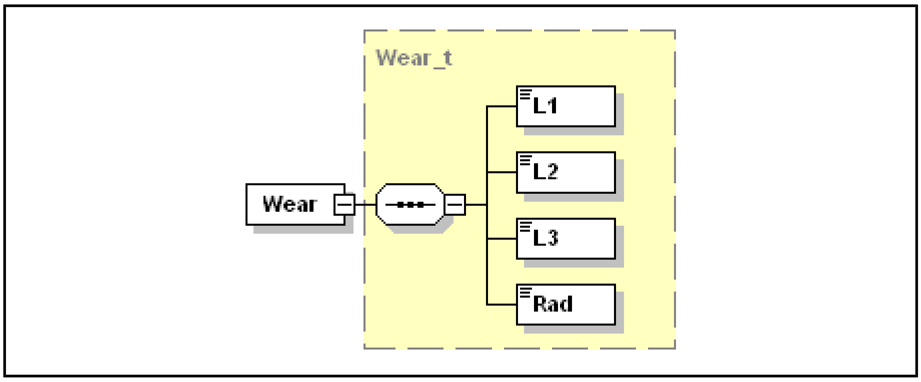


Fig. 16-50: Data record schema (12) (wear data upon delivery)

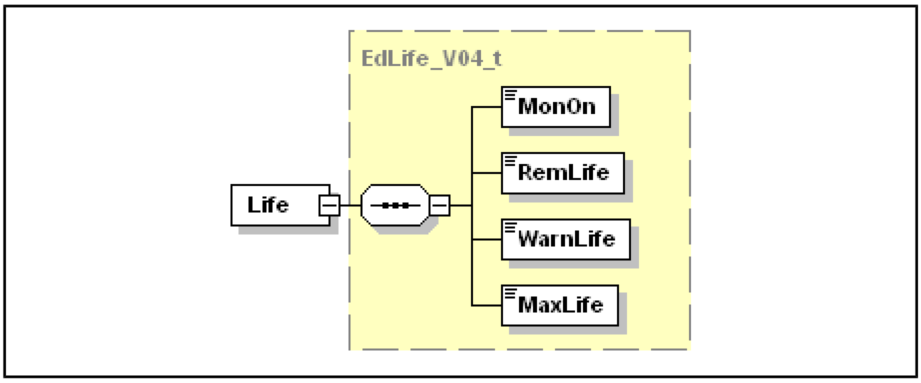


Fig. 16-51: Data record schema (13) (life data upon delivery)

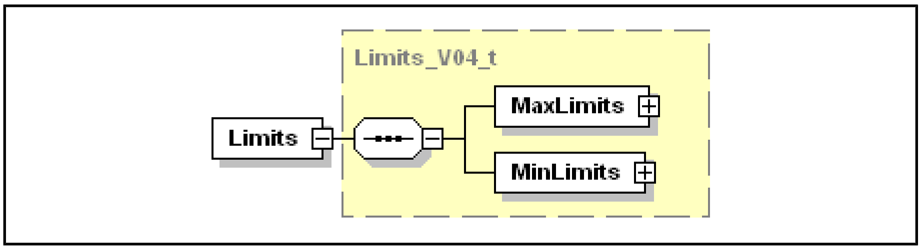


Fig. 16-52: Data record schema (13) (limit value data)

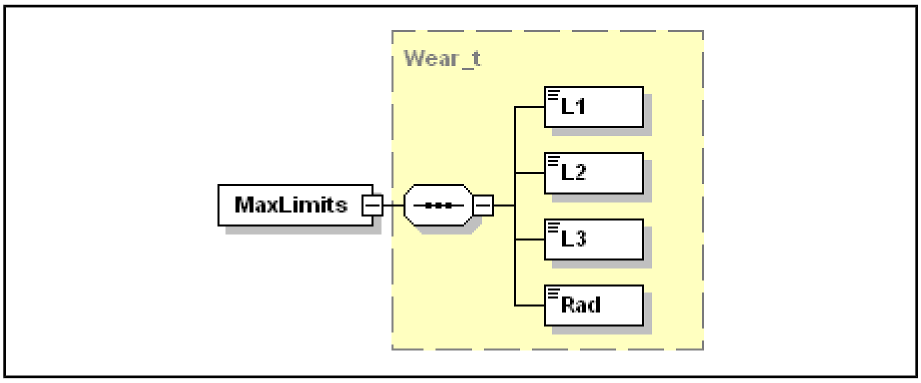


Fig. 16-53: Data record schema (14) (max. limit value data)

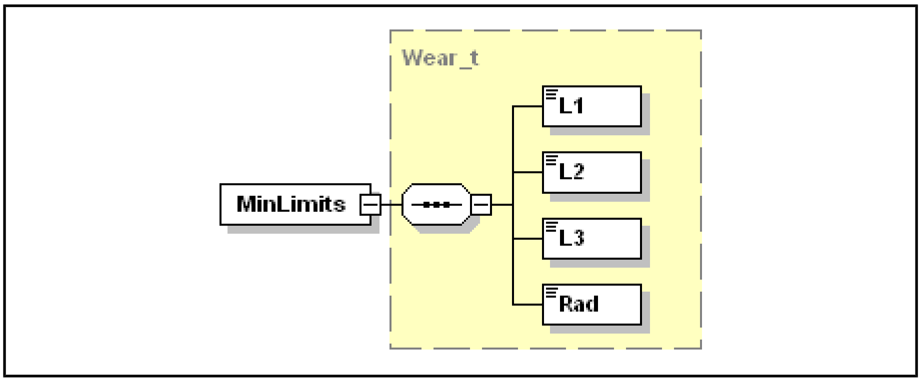


Fig. 16-54: Data record schema (15) (min. limit value data)

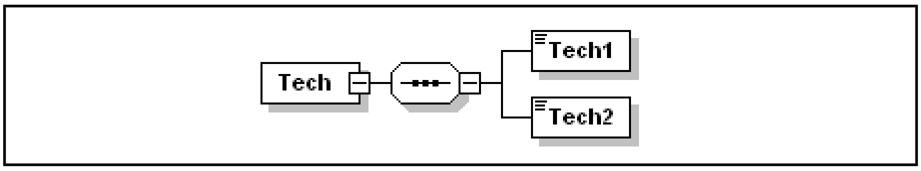


Fig. 16-55: Data record schema (16) (cutting technology data)

Basis Data Type Collection (basic_ty.xsd)

The schema file is available for all data schema definitions used by the MTX. It cannot be changed by the user. However, the user can use these basic types in the new data structures generated e.g. in the schema file **dbt1ud.xsd**.

Contents of the basic data type collection, refer to chapter 25 "Annex" on page 575

Definition of Basic Settings of Tool Management

The following settings are made using the attribute definitions in the schema files:

- Definition of the user rights
- Additive value input for certain data elements yes/no
- Limits for value increase with additive value entry
- Data element type definition (location or tool data element)
- Activation of tool-specific limit value check of input values in the PLC
- Activating the message to the PLC regarding a value change in the data record
- Activation of a value change in the data record by the PLC
- Activating the message to the PLC regarding inserting, deleting or moving a data record using the interface

Definition					Note			Usage	
			Example						
Name	Type	Use	Default	Fixed	Description	Poss. values	Meaning	Nodes	Element
L1	xs:string	Optional		R	User rights for L1 users	R	Read only	x	x
						RW	Read and write		
L2	xs:string	Optional		R	User rights for L2 users	R	Read only	x	x
						RW	Read and write		

Configuring the Tool Management

Definition					**Note**			**Usage**	
			Example						
Name	**Type**	**Use**	**Default**	**Fixed**	**Description**	**Poss. values**	**Meaning**	**Nodes**	**Element**
L3	xs:string	Optional		R	User rights for L3 users	R	Read only	x	x
						RW	Read and write		
L4	xs:string	Optional		RW	User rights for L4 users	R	Read only	x	x
						RW	Read and write		
L5	xs:string	Optional		RW	User rights for L5 users	R	Read only	x	x
						RW	Read and write		
ETA	xs:string	Optional		IA	Permitted type of editing	Without	Only absolute without diameter		x
						IA	Incremental and absolute without diameter		
						DI	Incremental with diameter and absolute without diameter		
MaxIncIn	xs:double	Optional	0.5		Maximum growth <= Limit value				x
MaxIncEx	xs:double	Optional	0.5		Maximum growth < Limit value				x
DataType	xs:string	Optional		PD	Data type attribute	PD	Place datum		x
						- or TD	Tool / workpiece date		
PLCCheck	xs:string	Optional		Lim	Type of communication with the PLC during editing	Lim	Limit value check by PLC		x
						Write	Checked and written by PLC		
						Info	Information that writing occurred to PLC		

Fig. 16-56: User-definable data element attributes for database tables

For the sake of completeness, the following two tables describe additional attribute definitions; however, these cannot be changed by the user during the configuration of the data record.

Definition					Note		
			Example				
Name	**Type**	**Use**	**Default**	**Fixed**	**Description**	**Poss. values**	**Meaning**
UA	xs:string	Optional		ID	Application attribute: • Application should be unique for the entire schema, i.e. each value may be used only once in the schema. • The attribute is required as an information source for system routines	TSt	Sector
						TPl	Place
						TID	ID element
						TT	Type element
						TC	Type code el.
						TS	Tool/part status
						PS	Location status
						TN	Number element
						DN	DuploNo element
						Rec	Record node
						Hd	Header node
						Ud	User data node
						Ed	Tool edge data node
SE	xs:boolean	Optional		1	Significant data element • Useful only for list display • If used together with UA=ID in the tool list, this causes write protection on the data element in the list.	1	DS ID valid / place occupied
U	xs:string	Optional	mm		Unit attribute	mm	
						inch	
						Nm	
						%	
Class	xs:string	Optional		DBTAB	Assignment attribute		
V	xs:string	required		00T00	Version attribute		

Fig. 16-57: System-specific data element attributes for database tables

Definition					Note	
			Example			
Name	**Type**	**Use**	**Default**	**Fixed**	**Description**	**Meaning**
DbL1.1	xs:string	Optional		DBT2/Rec/Hd/K1	Link element 1 for DBTab1	currently not used
DbL1.2	xs:string	Optional		DBT2/Rec/Hd/K2	Link element 2 for DBTab1	currently not used

Definition					Note	
			Example			
Name	**Type**	**Use**	**De-fault**	**Fixed**	**Description**	**Meaning**
DbL2.1	xs:string	Option-al		DBT3/Rec/Hd/K1	Link element 1 for DBTab2	currently not used
DbL2.2	xs:string	Option-al		DBT3/Rec/Hd/K2	Link element 2 for DBTab2	currently not used
TbL1.1	xs:string	Option-al		Root1/Rec/Hd/K1	Link element 1 for XMLTab1	currently not used
TbL1.2	xs:string	Option-al		Root1/Rec/Hd/K2	Link element 2 for XMLTab2	currently not used
TbL2.1	xs:string	Option-al		Root2/Rec{3}	Link element 1 for XMLTab2 (DS Index)	currently not used
TbL2.2	xs:string	Option-al		...	Link element 2 for XMLTab2	currently not used

Fig. 16-58: Link attributes for database tables (in preparation)

Increase Definition

The maximum permissible value increase for additive value increase for the corresponding data element is specified by the data attribute **MaxIncIn** or **MaxIncEx**.

Definition of the Input Type

When defining the input type, the **ETA** attribute can be used to select one of the options:

Without	Only absolute value input possible
IA	Additive and absolute value input
DI	Additive diameter input (absolute value input without diameter), i.e. input value / 2 is added

Definition of Data-Related User Privileges

Using the data attributes **L1** to **L5**, each user level can be assigned data element-relevant read and write rights.

These attributes can be defined for nodes and for elements. The attribute definitions can be structure-overridden. Then the definition which is closest to the element in the tree is valid.

Definition of Limit and Enumeration values

Data element-relevant limit value and enumeration definitions are implemented by defining a basic type with corresponding restrictions.

Example limit value definition:

1 ⇐ double value ⇐ 10 000 000

Program:

```
<xs:simpleType name=“Double1_10M_t”>
     <xs:annotation>
       <xs:documentation>Type Double 1 - 10 Mio</xs:documentation>
     </xs:annotation>
     <xs:restriction base=“xs:double”>
       <xs:minInclusive value=“1”/>
       <xs:maxInclusive value=“10000000”/>
     </xs:restriction>
</xs:simpleType>
```

Example enumeration definition:

0 ⇐ byte value ⇐ 3 with possible values: 1;2;3

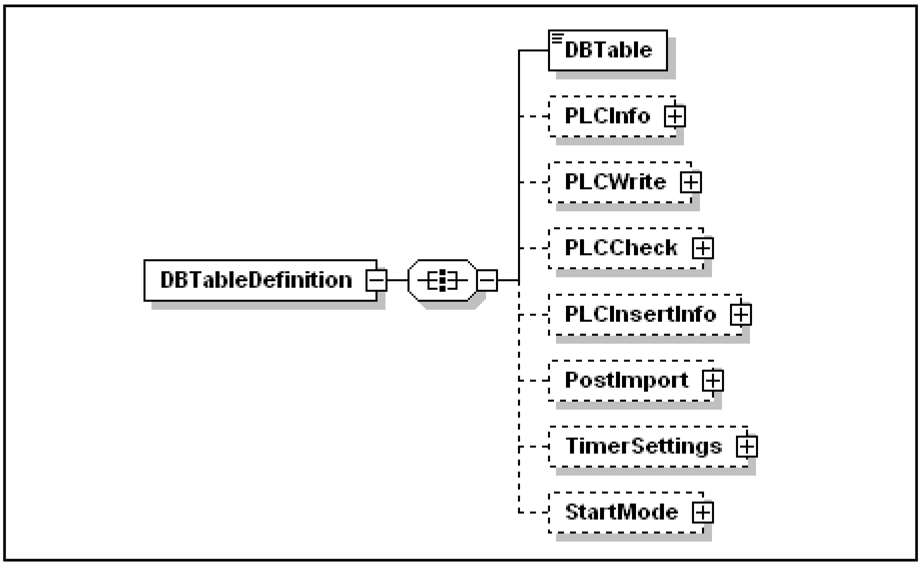
Program:

```
<xs:simpleType name="Byte0_3_t">
     <xs:annotation>
       <xs:documentation>Byte 0 - 3</xs:documentation>
     </xs:annotation>
     <xs:restriction base="xs:byte">
       <xs:minInclusive value="0"/>
       <xs:maxInclusive value="3"/>
       <xs:enumeration value="0"/>
       <xs:enumeration value="1"/>
       <xs:enumeration value="2"/>
       <xs:enumeration value="3"/>
     </xs:restriction>
</xs:simpleType>
```

Check of the Input Value via PLC

The following settings can be made in the file **"ToolManagementConfig.xml"** in the project directory.

This file is structured as follows:



PLCCheck	Definition of the PLC variable name for the Validate function
PLCInfo	Definition of the PLC variable name for the Info function
PLCWrite	Definition of the PLC variable name for the Write function
Post import	Definition of the post import function
DBConnections	Activating several DB tables
TimerSettings	Low-pass filter configuration:

Fig. 16-59: Structure diagram 1: ToolManagementConfig.xml

The data attribute **PLCCheck** of the data schema can be used to activate the input value check in terms of tool-specific limit value exceedance in the PLC for every data element via the entry**"Lim"**.

Example: The activation of limit value monitoring in the PLC in the data structure "Geo_V04_t" of the schema definition file "tool_ty.xsd":

Program:

```
<xs:element name="Rad">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="Double-10M_10M_t">
                <xs:attribute name="ETA" type="xs:string"
                use="optional" fixed="IA"/>
                <xs:attribute name="PLCCheck" type="xs:string"
                use="optional" fixed="Lim"/>
            </xs:extension>
        </xs:simpleContent>
```

```
    </xs:complexType>
</xs:element>
```

This function cannot be used for the bit fields (BQ1-3) and the string elements (SKQ).

Sequence:

1. Enter a value in the editor or in the list.
2. After input is completed, e.g. by pressing <ENTER>, the interface (Act = TRUE) activates the PLC via the structure variable and transfers the entered value.
3. The PLC checks the value to see if it exceeds the defined limit value.
4. The PLC sends an acknowledgement to the interface (Act = FALSE).

   The interface expects acknowledgement by the PLC within 500 ms (5 times the 100 ms framework). If this acknowledgement is missing, the value is not transferred to the DB. An error message is then output in the status line.

   The editing can only be closed when entering either a correct value or <ESC>.
5. In the case of positive feedback (result = 0) from the PLC, the interface writes the value change to the DB. Otherwise, the interface outputs an error message in the status line and the value is not written to the DB.

The PLC expects the following return values (applies for all following PLC functions):

Return Value	Description	Comment
<0	Reserved	Special error
0	Value is OK	
1	Value was rejected by the PLC	Normal error without description of cause
2	The selected position does not contain a tool	Special error
3	The tool database is blocked	Special error
4-10	Reserved	
>10	The value was rejected by the PLC with a number (e.g. tool edge number; number of correction value, etc.)	Normal error with description of cause The error number is output to describe the cause

For all return values ≠ 0, the entered value is NOT transferred. Negative return codes less than -1 may nor be returned by the PLC because these are used internally by the interface.

Structure of PLC var:

TYPE stValidate:			
STRUCT			
	Place:	INT;	Place
	Storage:	INT;	Sector

Act:	BOOL;	Activation
XPath:	STRING(79);	Data element
Value:	LREAL;	Input value
EditType:	INT;	Additive / absolute - not relevant!
Result:	INT;	Error return

END_STRUCT

END_TYPE

The name of the PLC variable can be specified in the "ToolManagementConfig.xml" file under **PLCCheck/PLCVariableID** in the project directory.

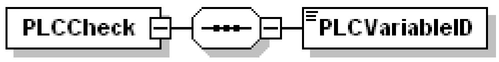


Fig. 16-60: Structure diagram 2: ToolManagementConfig.xml

Program:

```
...
    <PLCCheck>
        <PLCVariableID>.Validate</PLCVariableID>
    </PLCCheck>
    <PLCInfo>
        <PLCVariableID>.TLInfo</PLCVariableID>
    </PLCInfo>
    <PLCWrite>
        <PLCVariableID>.TLDataWrite</PLCVariableID>
    </PLCWrite>
...
```

Message to PLC Regarding a Value Change in the Data Record

The data attribute **PLCCheck** of the data schema can be used to activate the message of changes and the input value check for every data element to the PLC via the entry **"Info"**.

For example, activation of PLC info function in the data structure "Geo_V04_t" of the schema definition file "tool_ty.xsd":

Program:

```
<xs:element name="Rad">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="Double-10M_10M_t">
                <xs:attribute name="ETA" type="xs:string"
                use="optional" fixed="IA"/>
                <xs:attribute name="PLCCheck" type="xs:string"
                use="optional" fixed="Info"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
```

For string elements (SKQ) this function is also applicable, however, the variable "Value" is not assigned with a value in this case.

Sequence:

1. Enter a value in the editor or in the list.
2. The interface writes the value change into the database.
3. The interface activates the PLC via a structure variable and transfers the input value.
4. The PLC **can** react accordingly to the messages from the user interface.

Structure of PLC var:

```
TYPE stChangeInfo:
STRUCT
        Place:          INT;            Place
        Storage:        INT;            Sector
        Act:            BOOL;           Activation
        XPath:          STRING(79);     Data element
        Value:          LREAL;          Input value
        EditType:       INT;            Additive / absolute - not relevant!
        Result:         INT;            Error return - not relevant !
END_STRUCT
END_TYPE
```

The name of the PLC variable can be specified in the "ToolManagementConfig.xml" file under **PLCInfo/PLCVariableID** in the project directory.

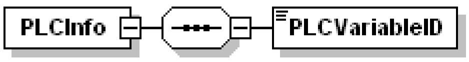


Fig. 16-61: Structure diagram 3: ToolManagementConfig.xml

Program:

```
...
        <PLCCheck>
        <PLCVariableID>.Validate</PLCVariableID>
    </PLCCheck>
    <PLCInfo>
        <PLCVariableID>.TLInfo</PLCVariableID>
    </PLCInfo>
    <PLCWrite>
        <PLCVariableID>.TLDataWrite</PLCVariableID>
    </PLCWrite>
...
```

Activation of a Value Change in the Data Record by the PLC

The data attribute **PLCCheck** of the data schema can be used to activate the change message and the input value check for every data element to the PLC via the entry **"Write"**.

Example:

The activation of limit value monitoring in the PLC in the data structure "Geo_V04_t" of the schema definition file "tool_ty.xsd":

Program:

```
<xs:element name="Rad">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="Double-10M_10M_t">
                <xs:attribute name="ETA" type="xs:string"
                use="optional" fixed="IA"/>
                <xs:attribute name="PLCCheck" type="xs:string"
                use="optional" fixed="Write"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
```

Sequence:

1. Enter a value in the editor or in the list.
2. After input is completed, e.g. by pressing <ENTER>, the interface (Act = TRUE) activates the PLC via the structure variable and transfers the entered value.
3. The PLC checks the value to see if it exceeds the defined limit value.

4. If the test was successful, the PLC writes the value change to the database.
5. The PLC sends an acknowledgement to the interface.

   The PLC sends an acknowledgement to the interface (Act = FALSE).

   The interface expects acknowledgement by the PLC within 500 ms (5 times the 100 ms framework). If this acknowledgement does not occur, the value is transferred to the DB by the interface. An error message is then output in the status line.
6. The interface issues a message regarding the success (or lack thereof) of the writing procedure in the status line.

Structure of PLC var:

TYPE stTLDataWrite:

STRUCT

Place:	INT;	Place
Storage:	INT;	Sector
Act:	BOOL;	Activation
XPath:	STRING(79);	Data element
Value:	LREAL;	Input value
EditType:	INT;	Additive / absolute
Result:	INT;	Error return

END_STRUCT

END_TYPE

The name of the PLC variable can be specified in the "ToolManagementConfig.xml" file under **PLCWrite/PLCVariableID**.

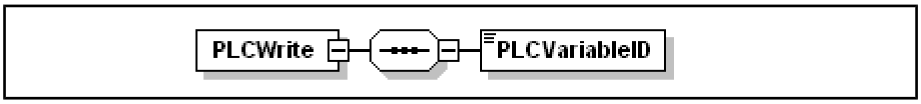


Fig. 16-62: Structure diagram 4: ToolManagementConfig.xml

Program:

```
...
    <PLCCheck>
        <PLCVariableID>.Validate</PLCVariableID>
    </PLCCheck>
    <PLCInfo>
        <PLCVariableID>.TLInfo</PLCVariableID>
    </PLCInfo>
    <PLCWrite>
        <PLCVariableID>.TLDataWrite</PLCVariableID>
    </PLCWrite>
...
```

This function cannot be used for string elements (SKQ).

Message to the PLC Regarding Inserting, Deleting or Moving a Data Record

The function can be activated and the name of the PLC variable can be specified in the "ToolManagementConfig.xml" file under **PLCInsertInfo/PLCVariableID**.

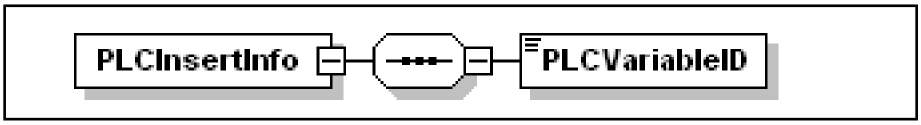


Fig.16-63: Structure diagram 5: ToolManagementConfig.xml

Program:

```
...
    <PLCCheck>
        <PLCVariableID>.Validate</PLCVariableID>
    </PLCCheck>
    <PLCInfo>
        <PLCVariableID>.TLInfo</PLCVariableID>
    </PLCInfo>
    <PLCInsertInfo>
        <PLCVariableID>.TLDeleteMoveInsertInfo</PLCVariableID>
    </PLCInsertInfo>
...
```

Structure of PLC variables

TYPE stInsertTool:

STRUCT

SrcPlace :	INT;	Location (source)
SrcStorage :	INT;	Sector (source)
DstPlace :	INT;	Location (target)
DstStorage :	INT;	Sector (target)
Act:	BOOL;	Activation
Value:	INT;	Function ID

END_STRUCT

END_TYPE

Function ID:

Value	**Function**
1	Tool has been inserted
2	Tool has been deleted
3	Tool data records have been imported
4	Tool was displaced

Sequence for tool list import (inserting/overwriting several data records - no single tool import):

1. Activating and executing tool list import
2. The interface activates the PLC using the structure variable:
   - SrcPlace = 0 (not relevant)
   - SrcStorage = 0 (not relevant)
   - DstPlace = 0 (not relevant)
   - DstStorage = 0 (not relevant)
   - Act = 1
   - Value = 3
3. PLC sets Act = 0
4. The PLC **can** react accordingly to the message from the user interface.

Sequence for inserting or copying a tool or importing a single tool:

1. Activating and executing tool import or the copy or insert function
2. The interface activates the PLC using the structure variable:
   - SrcPlace = n (n = number of the place to which insertion occurred)
   - SrcStorage = m (m = number of the sector to which insertion occurred)
   - DstPlace = 0 (not relevant)
   - DstStorage = 0 (not relevant)
   - Act = 1
   - Value = 1
3. PLC sets Act = 0
4. The PLC **can** react accordingly to the message from the user interface.

Sequence during tool deletion:

1. Activating and executing tool deletion function
2. The interface activates the PLC using the structure variable:
   - SrcPlace = n (n = number of the place on which deletion occurred)
   - SrcStorage = m (m = number of the sector on which deletion occurred)
   - DstPlace = 0 (not relevant)
   - DstStorage = 0 (not relevant)
   - Act = 1
   - Value = 2
3. PLC sets Act = 0
4. The PLC **can** react accordingly to the message from the user interface.

Sequence during tool movement:

1. Activating and executing tool movement function
2. The interface activates the PLC for the first time using the structure variable:
   - SrcPlace = k (k = number of the place on which deletion occurred)
   - SrcStorage = l (l = number of the sector in which deletion occurred)
   - DstPlace = n (n = number of the place to which insertion occurred)
   - DstStorage = m (m = number of the sector to which insertion occurred)
   - Act = 1
   - ID = 4
3. PLC sets Act = 0
4. The PLC **can** react accordingly to the message from the user interface.

Communication Between the User Interface and the PLC During the Inserting Process

Via the entry **PLCInsertDataChange/PLCVariableID** in the file "ToolManagementConfig.xml" there is the possibility to inform the PLC about the change of data elements while inserting a tool data record.

During this process, PLC and interface correspond next to the PLC variable structure for exchanging the handshake signals via the system data structure "**SD.SysToolInsert**". This system variable structure is of the type "DBT1Rec_t" and thus corresponds to the a complete tool data record.

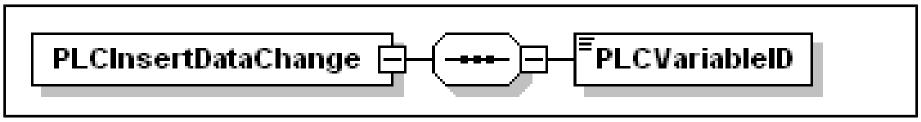


Fig.16-64: Structure diagram 6: ToolManagementConfig.xml

Program:

```
...
    <PLCCheck>
        <PLCVariableID>.Validate</PLCVariableID>
    </PLCCheck>
    <PLCInfo>
        <PLCVariableID>.TLInfo</PLCVariableID>
    </PLCInfo>
    <PLCInsertDataChange>
        <PLCVariableID>.TLInsertDataSet</PLCVariableID>
    </PLCInsertDataChange>
...
```

Structure of PLC variables

TYPE stInsertDataChange:

STRUCT

Place:	INT;	Place
Storage:	INT;	Sector
Act:	BOOL;	Activation
XPath:	STRING(79);	Data element
Value:	LREAL;	Input value
EditType:	INT;	Not relevant
Result:	INT;	Error return

END_STRUCT

END_TYPE

Procedure in the case of the the offline editing of a tool data record to be inserted

1. Transferring the data record of the insert position from the DB in SD.SysToolInsert.
2. The interface activates the PLC using the structure variable:
   - Place = n (n = number of the place to which insertion occurs)
   - Storage = m (m = number of the sector to which insertion occurs)
   - Act = 1
   - XPath = '' (not relevant)
3. PLC sets Act = 0
4. Change in the editor
5. Transferring the change in the SD.SysToolInsert
6. Message to the PLC
   - Place = n (n = number of the place to which insertion occurs)
   - Storage = m (m = number of the sector to which insertion occurs)
   - Act = 1
   - XPath = Modified data element
7. PLC reads SD.SysToolInsert
8. PLC sets Act = 0

9. PLC modifies SD.SysToolInsert
10. Message to the interface
    - Place = n (n = number of the place to which insertion occurs)
    - Storage = m (m = number of the sector to which insertion occurs)
    - Act = 1
    - XPath = Modified data element
11. Interface read modified variable SD.SysToolInsert
12. Interface sets Act = 0
13. Transferring the data record in the DB as completion of the inserting process

Post Import Function

In addition to the variable name definition described above, post importing is configured using file "ToolManagementConfig.xml".

If an action entry is defined in this file, a post-import function automatically follows a list or data record import.

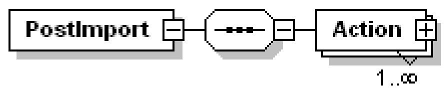


Fig. 16-65: Structure diagram 7ToolManagmentConfig.xml

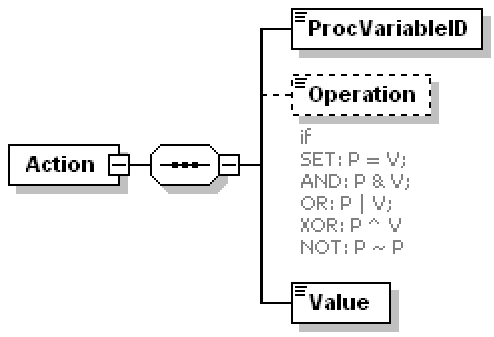


Fig. 16-66: Structure diagram 8: ToolManagmentConfig.xml

Explanation of abbreviations:

P:	Process variable	(ProcVariableID)
V:	Value	(Value)

Permitted operations

SET:	Process variable receives value
AND:	Process variable is logically AND-linked with value
OR:	Process variable is logically AND-linked with value
XOR:	Process variable is logically AND-linked with value
NOT:	The value of the process variable is negated

Fig. 16-67: Abbreviations and permitted operations

Data import procedure with post-import function active:

1. Import the tool list or the tool data record
2. Apply the Post import function on all imported data records

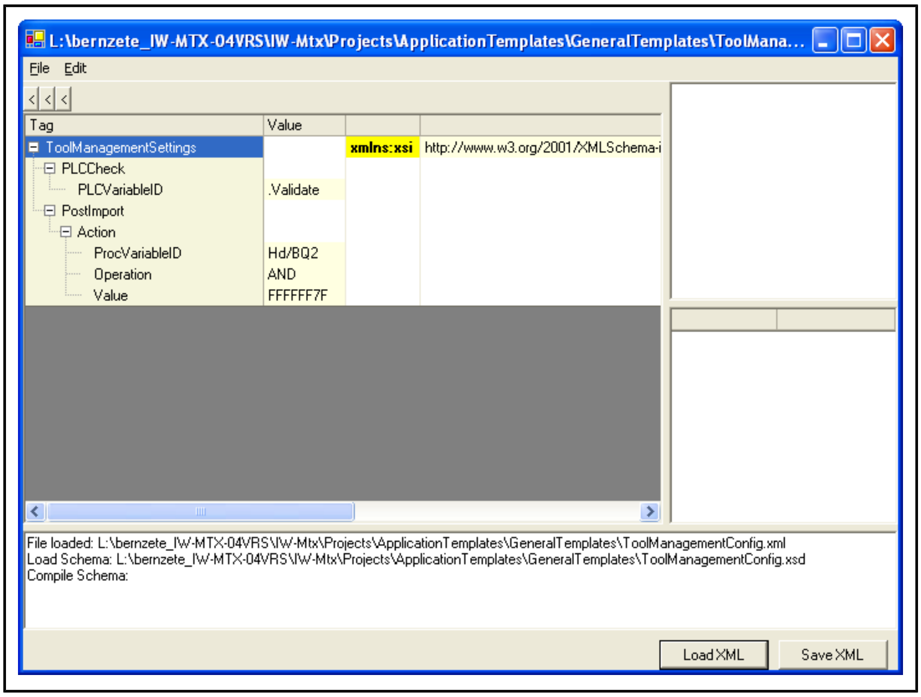


Fig. 16-68: Configuration of basic tool management settings

Example:

ProcVariableID	Hd/BQ2
Operation	AND
Value	FFFFFF7F

In the example, the "Tool broken" flag (TD) is reset in all imported data records after the tool list has been imported.

2. Activate 2nd Database Table

File "ToolManagementConfig.xml" is also used to activate communication with the 2nd database table.

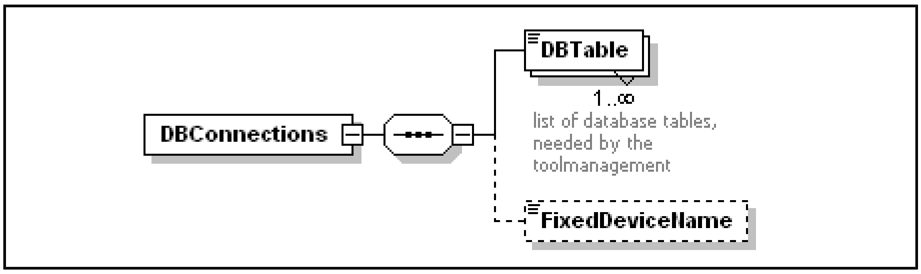


Fig. 16-69: Activating the 2nd database table

Program:

```
<DBConnections>
    <DBTable>/DBT1</DBTable>
    <DBTable>/DBT2</DBTable>
</DBConnections>
```

If these entries are missing, communication occurs only with database table 1.

If the database table is also to appear in IW Engineering as a separate configuration node, this must be configured by adding the following entry before creating the project in file **MTX.MnoApplicationConfig.xml** in the **runtime directory \Library\Devices\Visualization\.ApplicationTemplates\Visualization device (e.g. BTV40)**.

Program:

```
...
<HmiMnO ID="MTX DB SCREENS" Text="Pallet Screens" Category="MTXPALLETSCREENS"
Assembly="MTX.Toolman.MnOPackage"
Factory="Rexroth.MTX.Toolman.MnOPackage.ToolmanMnOHandlerFactory"/>
...
```

Low-Pass Filter for Change Events

To prevent blocking of communication by, for example, the PLC if the database is modified several times in sequence very quickly, a low-pass filter for change events can be activated and dimensioned. This is also carried out using file "ToolManagementConfig.xml".

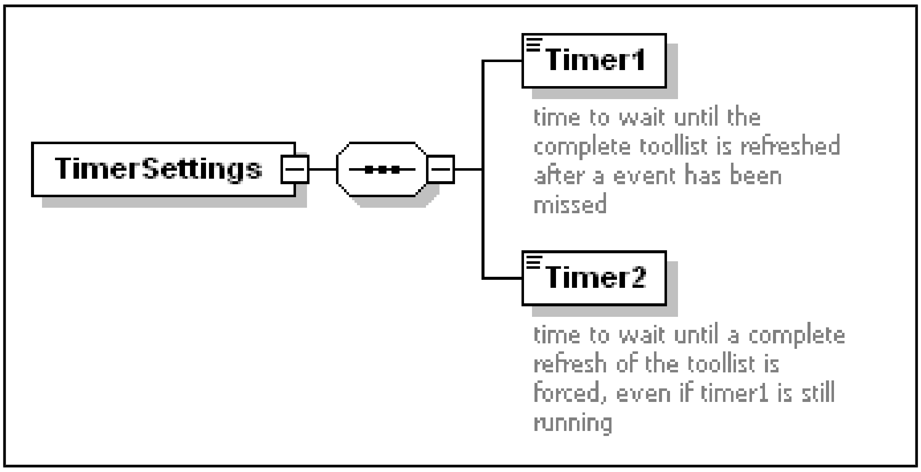


Fig.16-70: Low-pass filter for change events

Timer 1:

This timer is started if it is ascertained that a change event has been lost. If another event is lost during this time, the timer is restarted and no tool data can be read. Otherwise, all the data in the DB table are read after the timer elapses.

Timer 2:

If timer 1 never counts down completely to subsequently read all data read, a forced reading of all data is carried out after timer 2 elapses. Therefore, timer 2 is activated the first time that timer 1 is started and is stopped each time that all the data in the DB table are read.

Timer 1:	(small timer)	e.g. 500 ms
Timer 2:	(large timer)	e.g. 20 s

Setting the change events to be buffered in the control

The tool list display is informed via changes in the database tables to visualize the current state. Due to performance reasons, every single change is identified by the user interface.

In case of a fast sequence of writing operations by the PLC or CPL, the user interface can only be informed completely if a sufficient number of change events are buffered in the MTX.

The number of change events to be buffered can be defined via the configuration parameter "/NCO/DBTables/DBTEventBuffSize".

Default Values for Data Records

When creating data records, deleting their contents and displacing them, all data elements are initialized by default with the value 0 or an empty string. If other default values are desired, create one of the database-specific files "dbt1dat.xml" or "dbt2dat.xml". This file must contain a data record with the default values of the respective database table and must be located in the "\usrfep“ directory.

Handling Instruction: Modify a Data Record Schema

The following handling instruction describes the process to be followed when the data record of a database table is to be modified.

W Operation / Program: Edit the Schema File "dbt?ud.xsd"

1. Copy the file "dbt?ud.xsd" (?:= 1 [DBT1] or 2 [DBT2]) in the control directory "\usrfep\schema" or "\feprom\schema" or "\root\schema" to the mount directory (\mnt).
2. Edit file with schema editor.
3. Save file.
4. Copy the file back to the control directory "\usrfep" or "\root".

Screen		Documentation
Screen		Screen: Data record schema
Documentation	MTX Functional Description	Edit schema file

NC: Transferring the new data structure

1. Complete IW Operation
2. Reset the control

		Documentation
Documentation	MTX Functional Description	Transferring the new data structure

Handling Instruction: Modify Table-Specific Data Element-Relevant Limit Values

The following handling instruction describes the process to be followed when the limit values for the individual data elements of a tool data record (DBT1, DBT2) are to be modified.

IW Operation / program: Edit the Schema File "dbt?ud.xsd"

1. Copy the file "dbt?ud.xsd", "dbt?sd.xsd" (?:= 1 [DBT1] or 2 [DBT2]) or "tool_ty.xsd" in the control directory "\usrfep\schema" or "\feprom\schema" or "\root\schema" to the mount directory (\mnt).
2. Value range of the respective elements via

   <xs:minInclusive value = ".."/>

   <xs:maxInclusive value = ".."/>
3. Save file.
4. Copy the file into the control directory "\usrfep\schemas" or "\root\schemas".

Screen		Documentation
Screen		Screen: Data record schema
Documentation	MTX Functional Description	Edit schema file

NC: Transferring the new data structure

1. Complete IW Operation
2. Reset the control

		Documentation
Documentation	MTX Functional Description	Transferring the new data structure

Handling Instruction: Activate Tool-Specific Limit Value Monitoring in the User Interface

The following handling instruction describes the procedure for activating tool-specific limit value monitoring for the individual data elements in the editors of the user interface.

IW Operation / program: Edit the Schema File "dbt?ud.xsd"

For this function, it is necessary that limit value data elements are contained in the tool data record.

1. Call the tool list configuration in the current IndraWorks Engineering project
2. "Show XML"
3. In the XML editor, search for the corresponding CellDef entry for the data element that is to be edited using limit value monitoring.
4. Search/insert: "RepresentationDefinitions"
5. Search/insert: "Dependency_for_Representation"
6. Search/insert: "Validation"
7. Search/insert: "DepProcessVariableID"
8. In the Value column, define the process variable that contains the limit value.
9. Search/insert "Operation"
10. In the Value column, define the test condition (you can find help in the Properties window)
11. If further test conditions are to be defined: Search/insert "FollowingDepCondition" value = 1
12. Search/insert "Condition"
13. Continue with step 7 if the condition under step 11 has been fulfilled.
14. Save file.

Screen		Documentation
Screen		Screen: Data record schema (3) (header schema)
Documentation	MTX Functional Description	Tool-specific limit values

User interface: Call the modified list display again

Screen		Documentation
Documentation	MTX Functional Description	Tool-specific limit values

Handling Instruction: Create file with default values for a database table

This handling instruction can be used to create a new file with default values for a database table.

IW Operation / tool management: Create "dbt?dat.xml" file

1. Enter the desired default values into any data record of the respective database table

2. Export this data record into the file **"dbt?dat.xml"** (?:= 1 [DBT1] or 2 [DBT2]) into the mount directory (\mnt).

		Documentation
Documentation	MTX Functional Description	Create file dbt?dat.xml

IW Operation / program: Adapt file "dbt?dat.xml"

1. Open file in editor
2. Replace content of data element SKQ with an empty string.
3. Save file.
4. Copy the file to the **"\usrfep"** control directory.

		Documentation
Documentation	MTX Functional Description	Adapt file dbt?dat.xml

NC: Apply default values

1. Complete IW Operation
2. Reset the control

		Documentation
Documentation	MTX Functional Description	Apply default values

16.5 Tool Catalog

16.5.1 Definition of Terms and General Explanations

In dealing with the tool management of the MTX, the tool catalog has the following tasks:

- Tool type administration
- Administration of basic and master data

It follows that two types of data administration can be differentiated.

Tool type administration:	ToolCatalog.xml
Basic data administration:	ToolBDSCatalog.xml

The following illustration explains the terms

- Tool technology
- Tool type
- Tool class
- Alternate tool
- Master data record
- Basic data record

for the descriptions below.

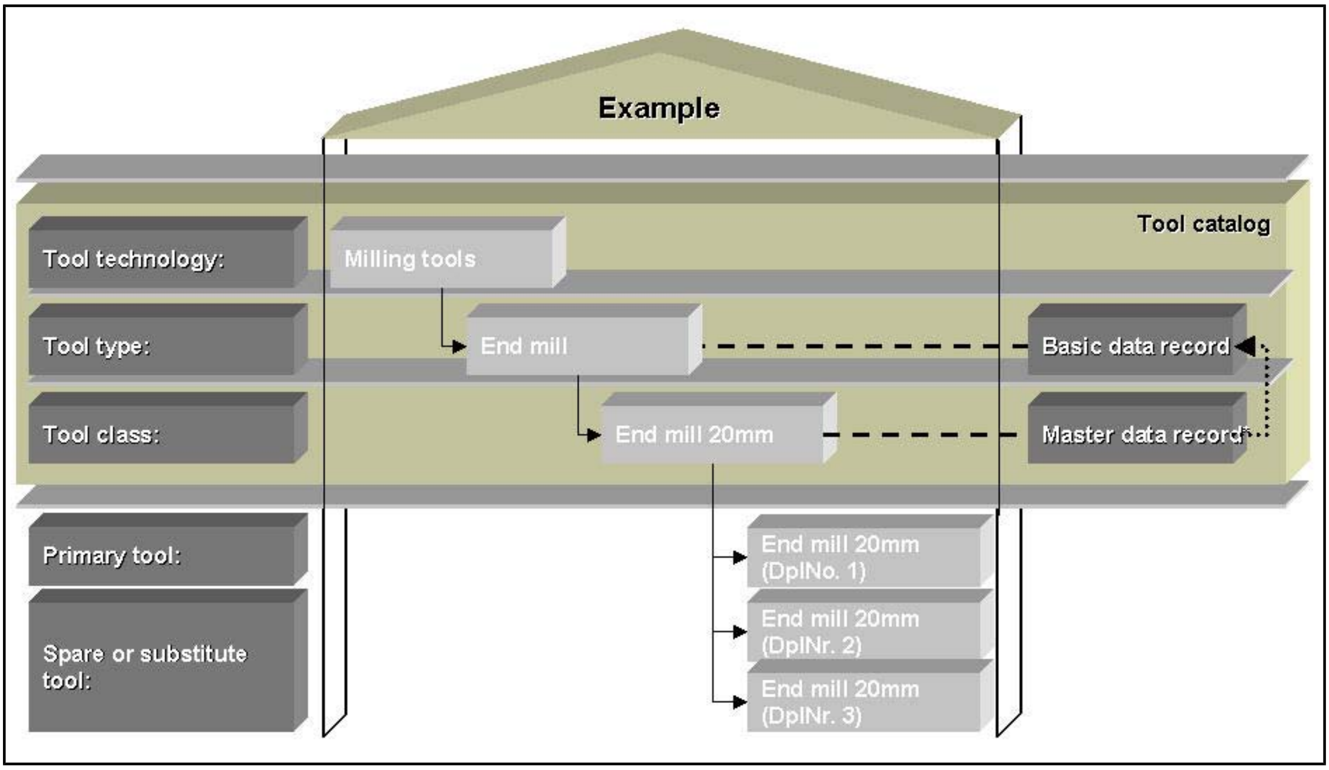


Fig.16-71: Explanation of tool catalog terms

Tool Technology

Comprises all tool types belonging to a machining technology.

The tool catalog includes the following machining technologies:

- General machining
- Drilling
- Milling
- Turning
- Special-purpose tools

Tool Type

Describes the total of all tool classes with identical type-defining properties. The individual tool classes of a tool type differ by the tool ID. Together with the tool technologies, the tool types form the basis of the tool catalog and, thus, the basis of master tool data administration. Furthermore, bitmaps are provided to go with the tool types in the tool catalog.

Tool Class

The amount of all tools with the same ID is described as a tool class. The individual tools of a class differ by their index number.

Replacement or Alternate Tool

A replacement tool belongs to a tool class and can replace the respective primary tool if necessary. It is, for example, not blocked or worn.

Primary Tool

A primary tool is the tool of a tool class which will be used for machining at the next call-up via the T word. All other tools of the tool class are called secondary tools.

Basic Data Record

The basic data record exists exactly once for each tool type when a tool has already been defined for this type and when a basic data record has been explicitly created in this context. If an additional tool (tool class) of this type is created, the basic data record is used to initialize the data record.

Master Data Record (in Preparation)

The master data record exists exactly once for each tool class when a tool has already been defined for this class. If another tool (alternate tool) is created, the master data record is used for data record initialization. If a tool is created for a class for the first time, the data record of this tool is turned into the master data record for the tool class.

Master data administration is still in preparation.

16.5.2 Predefined Tool Types (Standard Types)

For the machining technologies

- Drilling
- Milling
- Turning

the user may resort to pre-defined tool types in order to create the tool data records.

These tool types are stored in the tool catalog and are characterized by the following tool properties:

Designation of the element in the tool catalog		**Meaning**	**Element in the DS header**
TypeNo		Type number	IKQ2
TypePic		Name of the bitmap file	-
CoTe (bits 0 - 3)	EdNo	Edge number	BQ3
Relevance of the correction values (correction type):			BQ3
CoTe (bits 8 - 14)	L1	**1st length correction value**	
	L2	**2nd length correction value**	
	L3	**3rd length correction value**	
	R1	**big radius**	
	R2	**small radius**	
	O	**Edge position**	
	DIA	**Tool with diameter information**	

Designation of the element in the tool catalog		Meaning	Element in the DS header
Relevance of the technology data:			BQ3
CoTe (bits 15 - 23)	T1	**dortw**	
	T2	**ds**	
	T3	**lfc** or **ll** or **tw** or **tc**	
	T4	**tl**	
	T5	**lu**	
	T6	**tp** or **ta** or **tal**	
	T7	**tac**	
	T8	**nt**	
	ET1	**de**	
	ET2	**lfc**	

Fig.16-72: Type-defining tool data

The technology data bits T1 and T8 mentioned refer to the tool technology data in the standard data record. The technology data bit ET1 marks the relevance of the tool edge technology date in the default data record.

As can be seen from the table, the technology data in the standard data record can have a different meaning subject to the tool type.

Technology data

Abbreviation	Meaning
d respectively **de**	
	Diameter
	Greater diameter
	Outer diameter
	outer limitation of the cutting diameter area
	Edge radius
ds respectively **de**	Minor diameter
	Shank diameter
	Drilling diameter
	Smallest diameter that can be machined
	inner limitation of the cutting diameter area
lfc	Chamfer length
	(General) length
ll	Loss length
tw	Tool edge width
	Width of trapezoid thread
tc	Corner radius
	Tool tip size

Abbreviation	Meaning
tl	Max. cutting height
	Max. thread height
lu	Usable length
tp	(Thread)lead
ta	Setting angle
tal	Orientation angle
	Angle (edge angle)
	Tool setting angle
	Angle (kappa)
	(General) angle
tac	Apical angle
	Countersink angle
	Corner angle
	Flank angle
nt	Number of teeth

Fig. 16-73: Meaning of the technology data

Obviously, these technology data may be assigned another meaning in case of self-defined tool types.

For users who wish to use only a part or none of the tool catalog functions, the tool type "general tool" with the machining technology "general machining" as well as a general drilling, milling or turning tool with a variable tool edge number for the individual technologies has been introduced. Furthermore, the two "UsedFlag" data elements can be used to deselect both individual types and technologies in the tool catalog. Then these will not be offered for selection any more in the tool definition.

☞ If a tool data record is created using the PLC or CPL with reference to the tool catalog, data elements IKQ2 and BQ3 must be written in accordance with the catalog definition. If a tool data record is created using the user interface, these data elements are written automatically.

Create Data Record via PLC

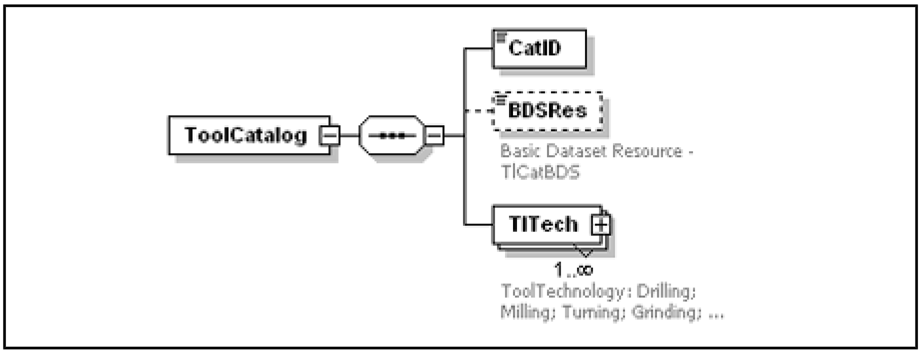
If the PLC is to create tool data records which are compatible with the default tool catalogue - e.g. by means of a Balluff BIS identification system - and which are to be entered into the database, it is recommended to determine the corresponding value pairs IKQ2 - BQ3 via a prepared system data structure and the corresponding function module.

16.5.3 Optional Extension or Modification of the Tool Type Catalog

In MTX tool management, the user is provided with a default tool catalog which he can supplement by tool types of his own. It is also possible to deactivate default tool types or tool technologies which are not required.

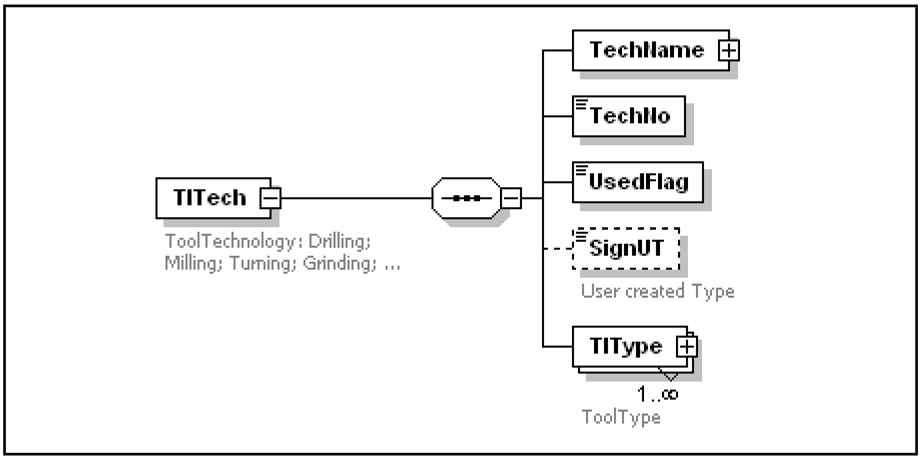
For this purpose, the XML document file "ToolCatalog.xml" (included in the delivery) must be changed.

This file is based on the following data scheme.



TlTech Tool technology

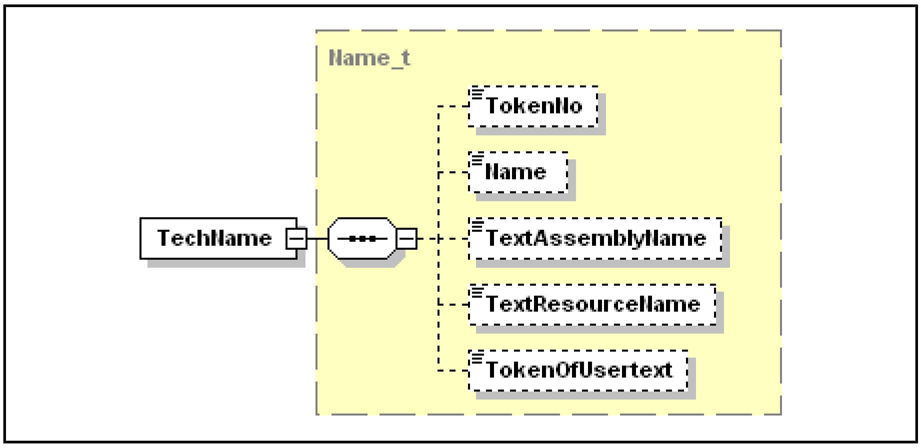
Fig.16-74: Data scheme of tool type catalog (1)



TechName Designation of the tool technology
TechNo Number of the tool technology
UsedFlag Utilization sign
SignUT Indication of the user type
TlType Tool type structure

Fig.16-75: Data scheme of tool type catalog (2)

By setting / resetting the UsedFlag, the user can hide tool technologies.

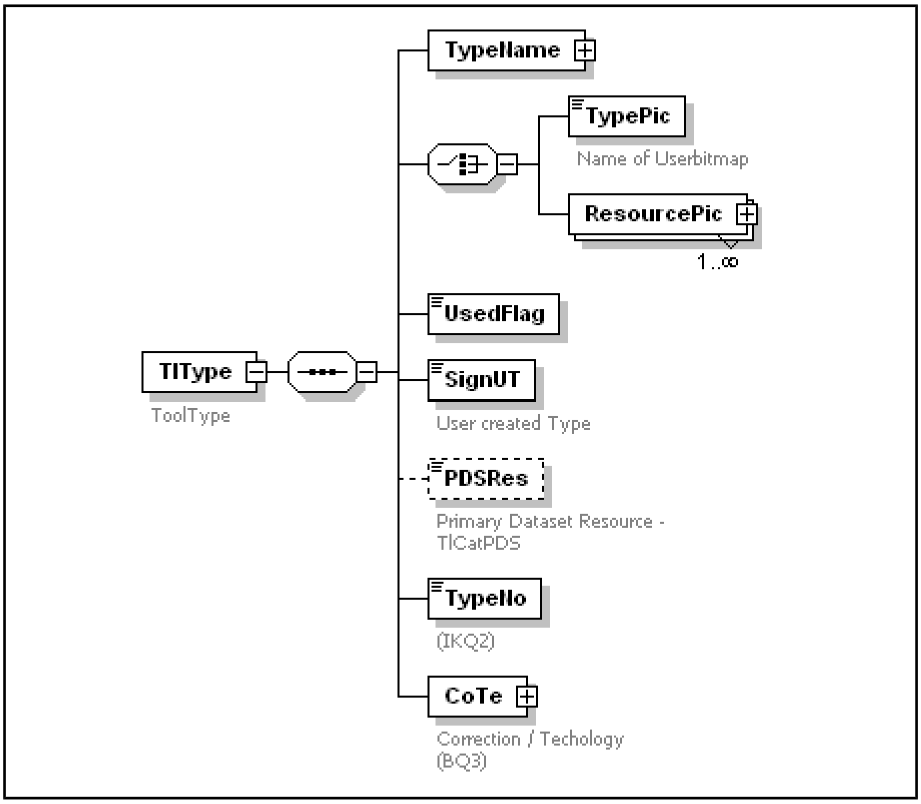


TokenNo	Text token (for resource texts)
Name	Fixed text (language-independent)
TextAssem-blyName	Assembly name (for resource texts)
TextResour-ceName	Resource name (for resource texts)
TokenOfU-sertext	Text token (for user texts)

Fig. 16-76: Data scheme of tool type catalog (3)

If all elements are used for text definition, the search proceeds according to the following criteria:

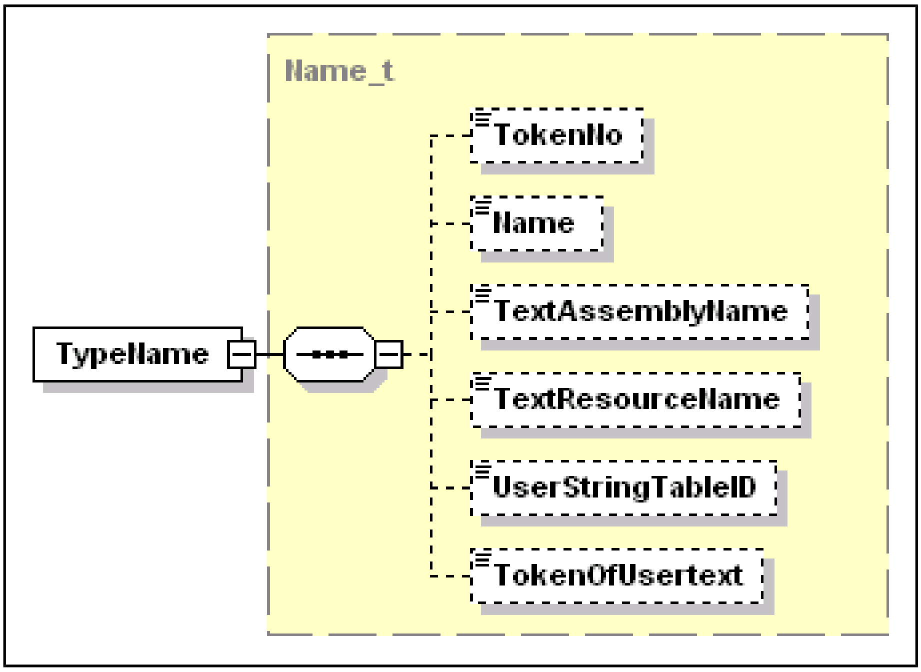
Priority	Text type
1	Resource text
2	User text
3	Fixed text



TypeName	Designation of the tool type
TypePic	Name of the type image (for user images)
ResourcePic	Definition of the image resource (if bitmap libraries are used)
UsedFlag	Utilization sign
SignUT	Indication of the user type
PDSRes	Link to the master data record (currently not yet in use)
TypNo	Tool type number (IKQ2)
CoTe	Code for the presentation of technology data (BQ3)

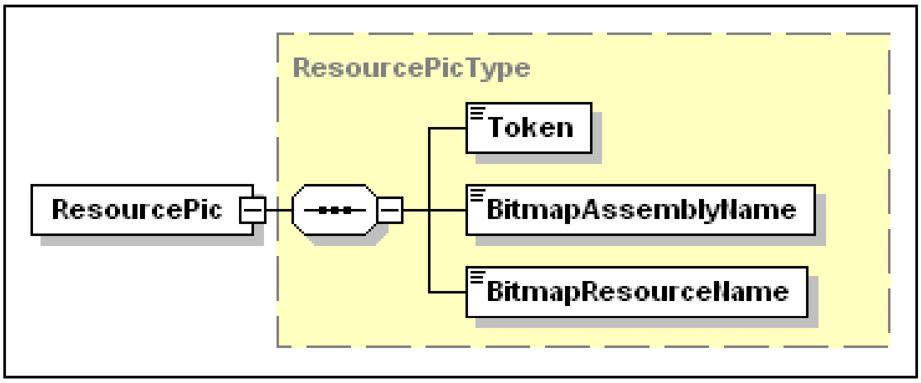
Fig.16-77: Data scheme of tool type catalog (4)

☞ New user tool type images have to be stored under the project directory **...\[Project Name]\[Visualization Device]\user\Config**.



TokenNo	Text token
Name	Fixed text
TextAssem-blyName	Name of the resource DLL
TextResour-ceName	Name of the text resource
UserStringTa-bleID	ID of the user text table
TokenOfU-serText	Token of the user text

Fig. 16-78: Data scheme of tool type catalog (5)



Token	Image token (for resource bitmaps)
BitmapAs-semblyName	Assembly name (for resource bitmap)
BitmapRe-sourceName	Resource name (for resource bitmap)

Fig. 16-79: Data scheme of tool type catalog (6)

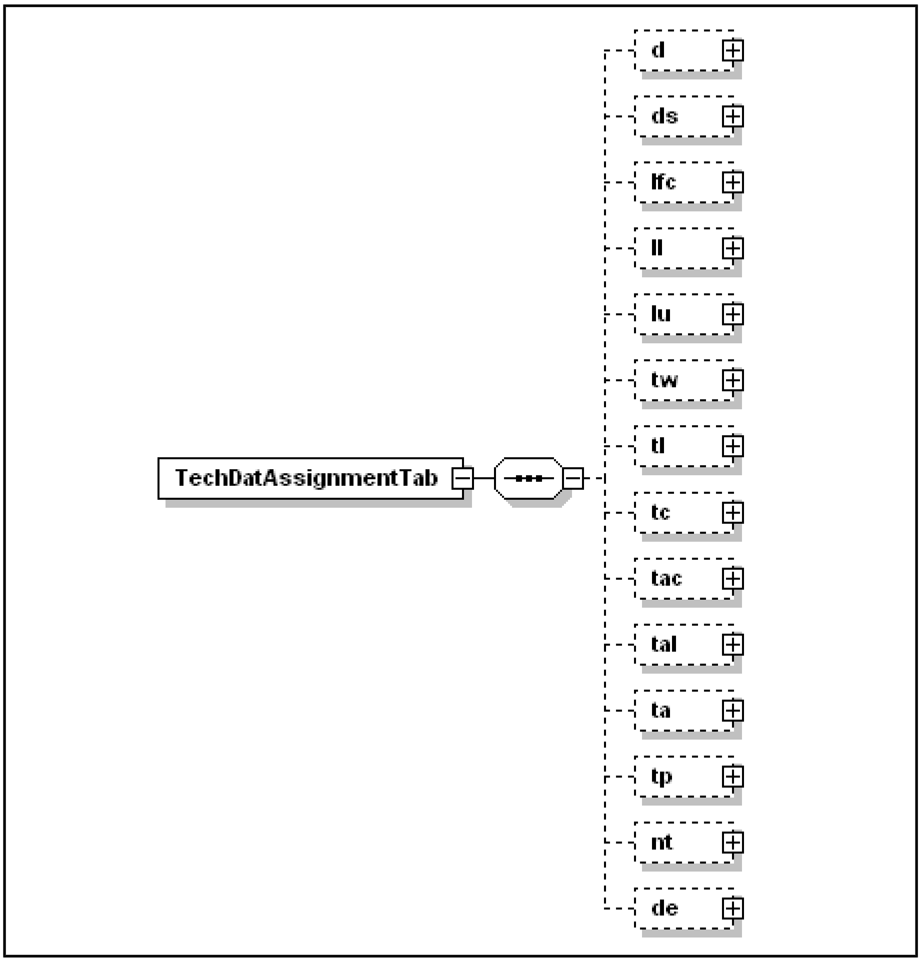
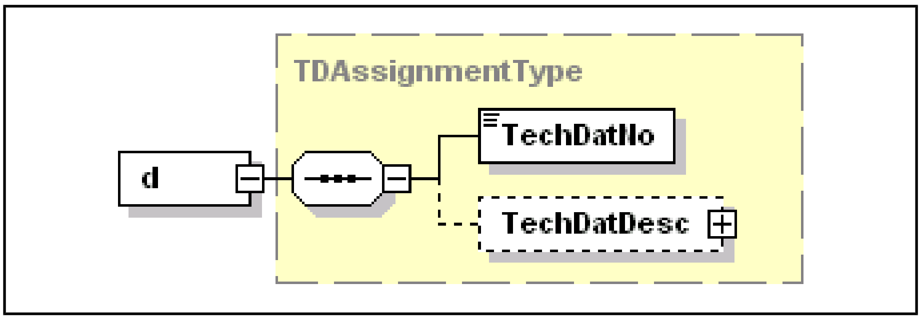


Fig. 16-80: Data scheme of tool type catalog (7)

Using the example of the technology date "d", the next screen shows how a data element in the tool data record is allocated to the corresponding technology date - dependent on the tool type. Furthermore, via "TechDatDesc" this data can also be given a corresponding name text which, for instance, is then shown in the tool editor.



TechDatNo	Number of the technology data element (1 ≙ T1 ... 8 ≙ T8 ; 11 ≙ ET1 ; 12 ≙ ET2 ; 0 ≙ not relevant)
TechDatDesc	currently only for internal usage

Fig. 16-81: Data scheme of tool type catalog (8)

For reasons of completeness, the next figure shows the structure of the technology data utilization (CoTe).

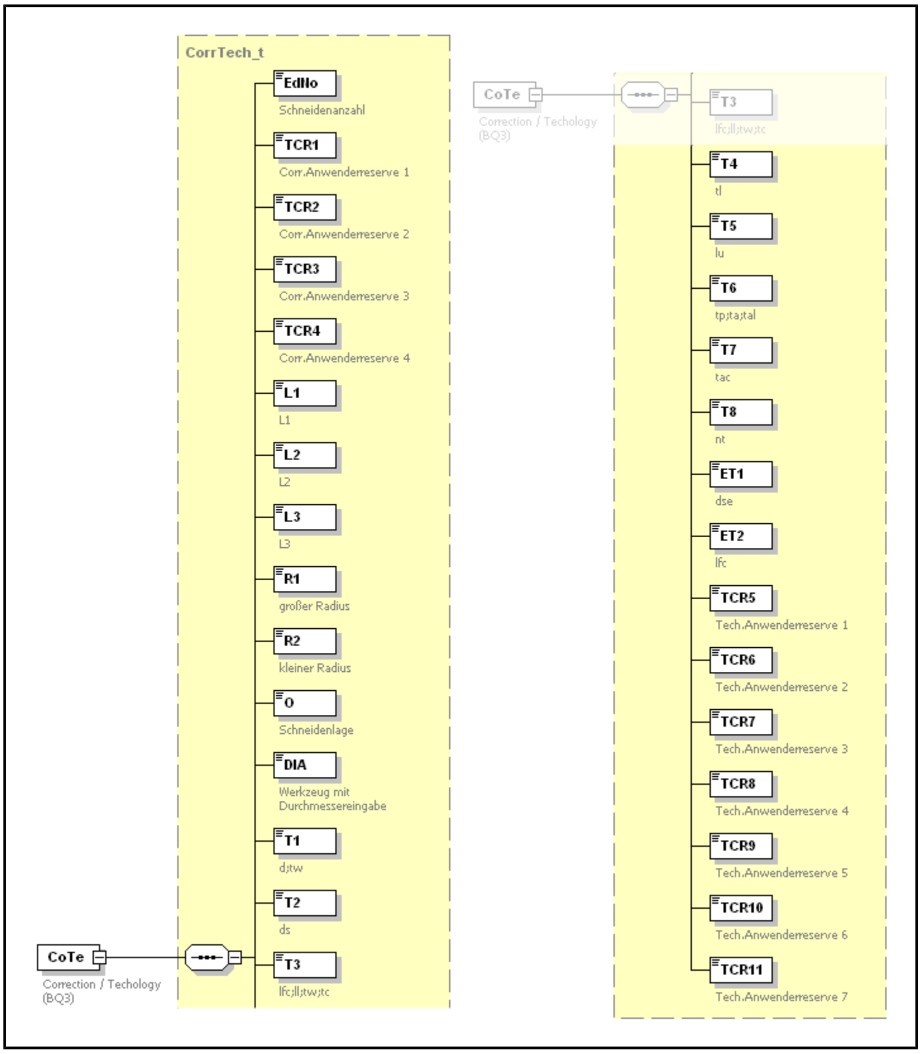


Fig. 16-82: Data scheme of tool type catalog (9)

For an explanation of CoTe, see fig. 16-72 " Type-defining tool data" on page 372.

By setting / resetting the UsedFlag, the user can hide tool types. To add a new tool type of his own, the user needs to create a complete TIType structure and insert it.

These changes can be effected via any XML editor.

16.5.4 Basic Data Administration

General

In the insertion editor, the "Store basic tool data" <F5> function can be used any time to create a basic data record for the currently selected tool type, or to overwrite an existing basic data record. If another tool of this type has been created later, the basic data record is used for data record initialization. All basic data records are stored in the "ToolBDSCatalog.xml" file.

The basic data record only comprises the user data of the tool data record.

Initialization of a Tool Data Record During the Creation of a Tool

Field / element	Identifier	Meaning	Source of initialization
1	K1	Storage	No initialization
2	K2	Place	
3	SKQ	ID	
4	IKQ1	Duplo no.	
5	IKQ2	Type	ToolCatalog.xml: **TypeNo**
6	IKQ3	T. No.	No initialization
7	IQ1	Reserve	ToolBDSCatalog.xml: **BaseDS**
8	IQ2	Reserve	
9	IQ3	Reserve	
10	BQ1	P status	No initialization
11	BQ2	T status	
12	BQ3	Technology	ToolCatalog.xml: **CoTe**
13	aaa	1. Freely configurable data element	ToolBDSCatalog.xml: **BaseDS**
:	:	:	
n	zzz	Nth freely configurable data element	

Fig. 16-83: Tool data record initialization using the basic data record

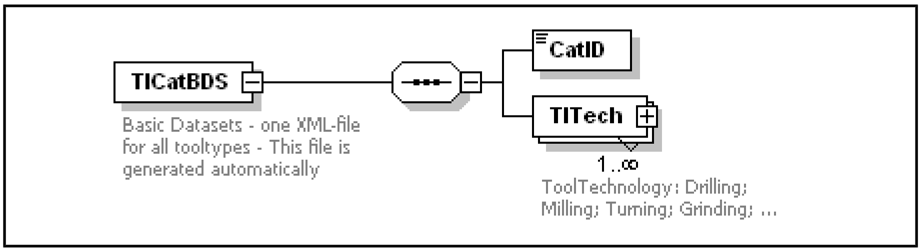


Fig. 16-84: Data scheme of tool basic data catalog (1)

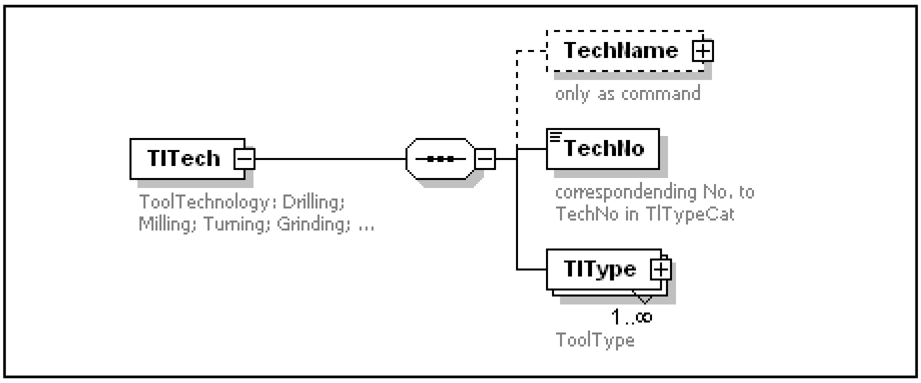


Fig. 16-85: Data scheme of tool basic data catalog (2)

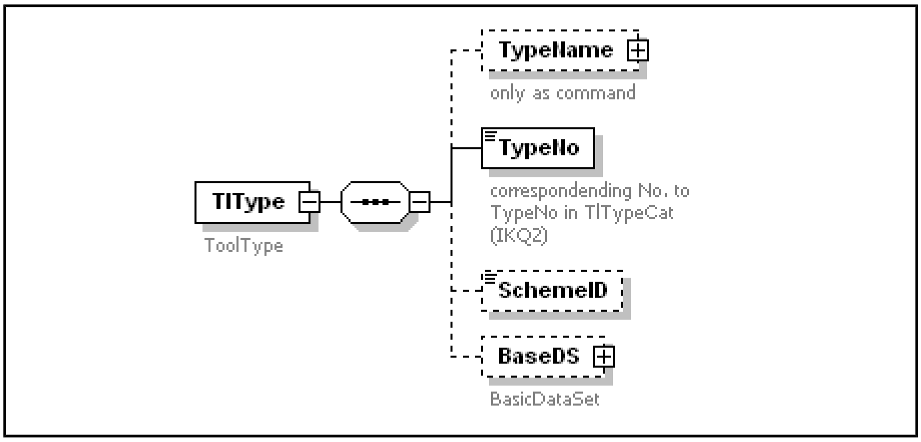


Fig.16-86: Data scheme of tool basic data catalog (3)

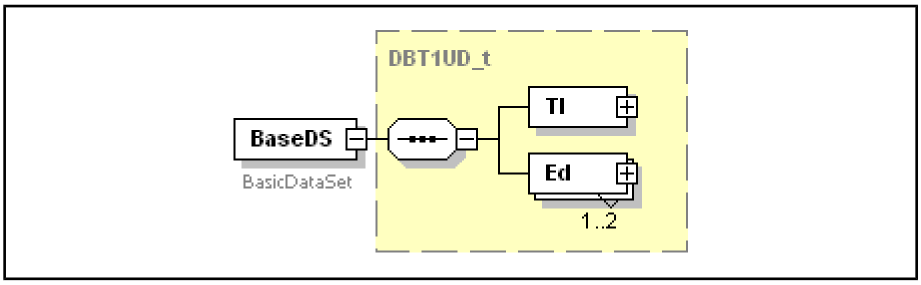


Fig.16-87: Data scheme of tool basic data catalog (4)

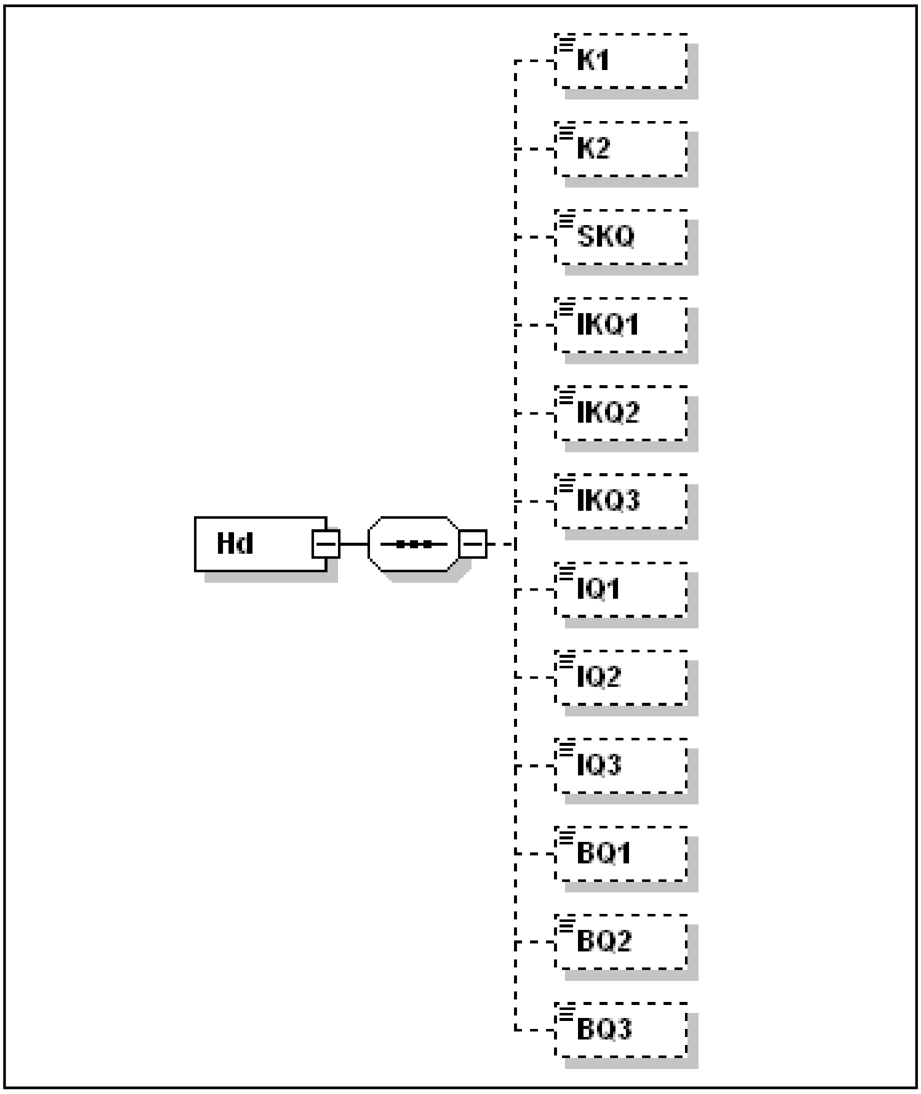


Fig. 16-88: Data scheme of tool basic data catalog (5)

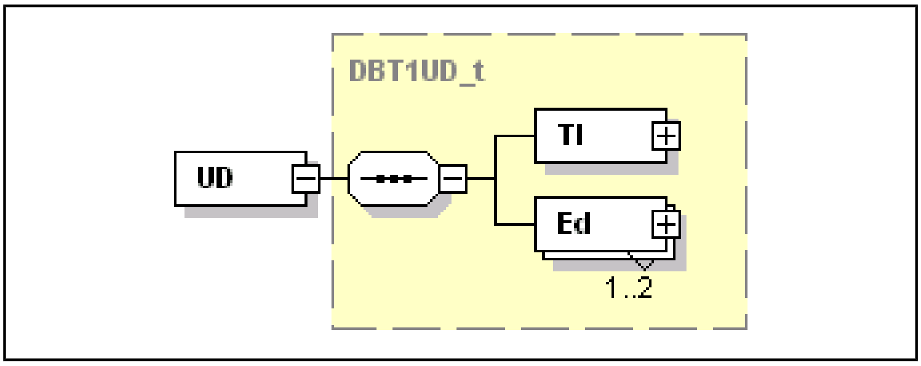


Fig. 16-89: Data scheme of tool basic data catalog (6)

The data type "DBT1UD_t" used in the catalog scheme is the same data type which is used in the tool data scheme. This is ensured by the include instruction <xs:include schemeLocation="dbt1sd.xsd"/>.

Handling Instruction: Add Tool Types to the Catalog

The following handling instruction describes the process to be followed when a new tool type is to be inserted into the tool catalog.

PC / XML Editor: Edit ToolCatalog.xml

1. Open the file **"ToolCatalog.xml"** in the current project directory **...\[Project name]\[Visualization device]\user\Config** via the XML editor.
2. Search the tag **<TlTech>** with the corresponding **<TechName>**.
3. Create a new structure **<TlType>**.
4. Adjust the data element as desired.
5. Save the file.
6. New user tool type images must be saved in the project directory under: **...\[project name]\[visualization device]\user\Config**.

Figure		Documentation
Figure		Screen: tool scheme
Documentation	MTX Functional Description	Edit schema file

IW Engineering: Data Transfer

1. Save the project

		Documentation
Documentation	IndraWorks HMI	Data Transfer

Handling Instruction: Activating/Deactivating the Tool Type in the Catalog

The following handling instruction describes the process to be followed when a tool type is to be shown/hidden within the tool catalog and thus within the tool management user interface.

PC / XML Editor: Edit "ToolCatalog.xml"

1. Open "ToolCatalog.xml" file in the current project directory

   ...\[project name]\[visualization device]\user\Config

   with the XML editor.
2. Search the tag **<TlTech>** with the corresponding **<TechName>**.
3. Search the tag **<TlType>** with the corresponding **<TypeName>**.
4. Set the tag **<UsedFlag>** to the value "true"/ "false".
5. Save the file.

Figure		Documentation
Figure		Screen: tool scheme
Documentation	MTX Functional Description	Edit schema file

IW Engineering: Data transfer

Save the project

		Documentation
Documentation	IndraWorks HMI	Data Transfer

Handling Instruction: Activate/Deactivate Tool Technology in the Catalog

The following handling instruction describes the process to be followed when a technology is to be shown/hidden within the tool catalog and thus within the tool management user interface.

PC / XML Editor: Edit "ToolCatalog.xml"

1. Open the file "ToolCatalog.xml" in the current project directory **...\[Project name]\[Visualization device]\user\Config** via the XML editor.
2. Search the tag **<TlTech>** with the corresponding **<TechName>**.
3. Set the tag **<UsedFlag>** to the value "true"/ "false".
4. Save the file.

Figure		Documentation
Figure		Screen: tool scheme
Documentation	MTX Functional Description	Edit schema file

IW Engineering: Data transfer

Save the project

		Documentation
Documentation	IndraWorks HMI	Data Transfer

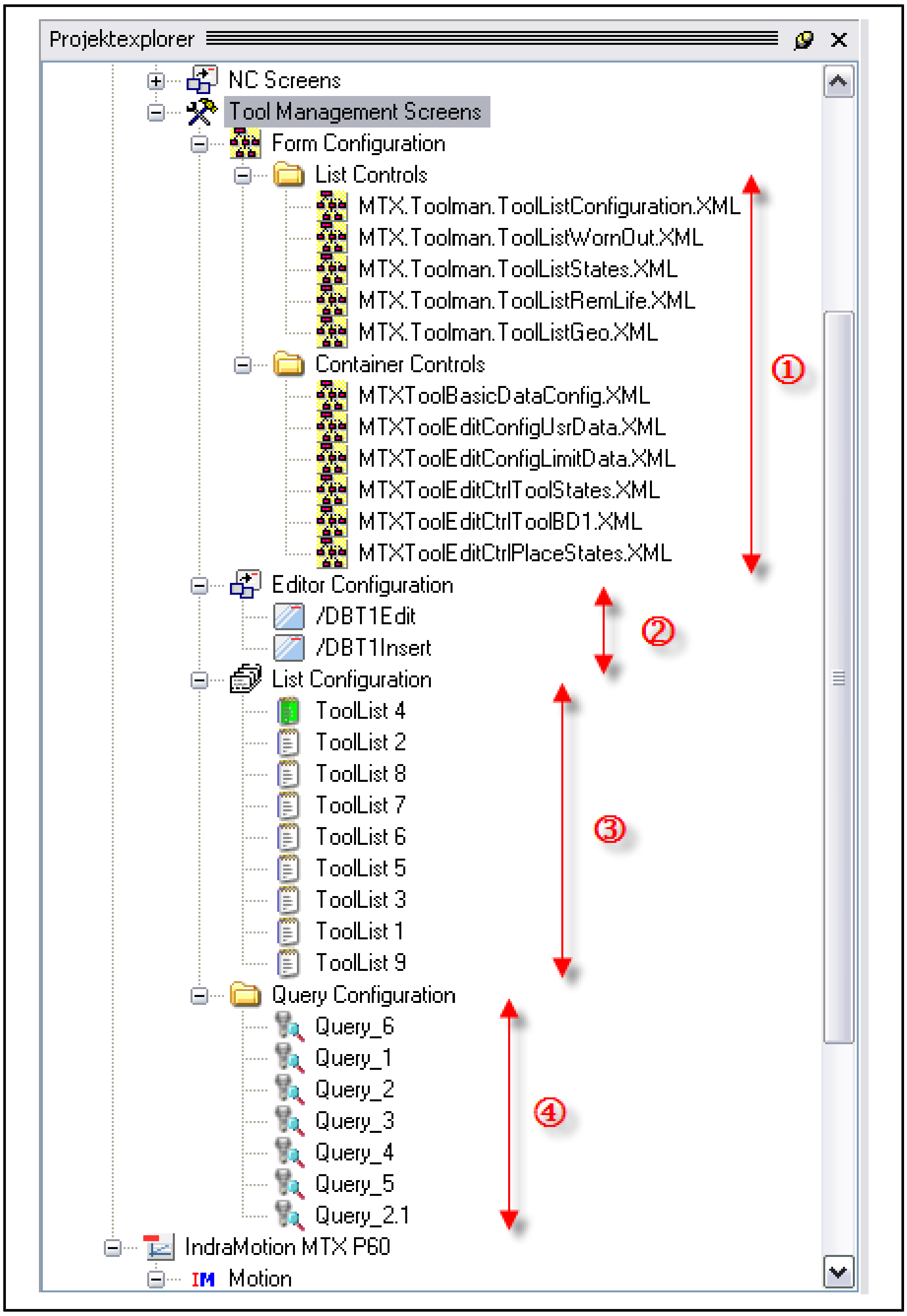
16.6 User Interface

16.6.1 General

The configuration of the tool management user interface is carried out in IndraWorks Engineering.

After an HMI device has been created, a **"Tool Management Screens"** project node with the following structure appears in the project tree:

Configuring the Tool Management



1	Predefined ULC configurations
2	Tool editors
3	Tool lists
4	Query definitions
Fig.16-90:	*Project Explorer (tool management)*

Called editors

1. Forms
   - List Controls

     XML file editor

     (refer to chapter 16.2.5 "XML File Editor" on page 329)

     or

     ULC configurator

(refer to chapter 16.2.4 "ULC Configurator" on page 311)

- Container Controls
  XML file editor
  (refer to chapter 16.2.5 "XML File Editor" on page 329)
  or
  ULC configurator
  (refer to chapter 16.2.4 "ULC Configurator" on page 311)

2. Editor configuration
   Editor configurator
   (refer to chapter 16.6.3 "Configuration of Tool Editors" on page 408)
3. List configuration
   List configurator
   (refer to chapter 16.6 "User Interface" on page 385)
4. Query configuration
   Query configurator
   (refer to chapter "Definition of List Content" on page 388)

16.6.2 Configuring Tool Lists

General

In the "Tool management" operating range, various tool lists can optionally be defined with different contents and forms of representation.

Double-click with the mouse or use the pop-up menu function **Open** on a list configuration to call the following "Settings" dialog.

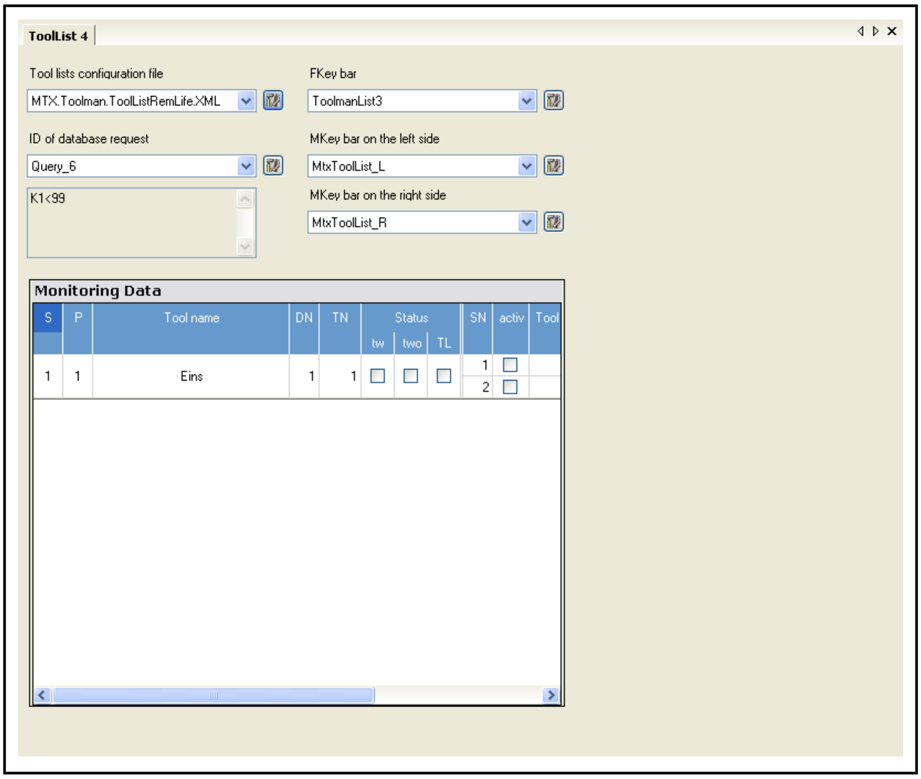


Fig. 16-91: List configurator

The lower part of the list configurator shows a preview of the list. In addition, the relevant configuration dialogs, which are described in detail in the following chapters, can be called via the configuration keys .

The **"ToolManagementApp.xml"** contains all the required list information.

The query comments are language dependent and can - as it is the case for all other language-dependent texts - be defined in one of the following formats in the tool management:

- A resource text to be defined in a resource DLL or as
- user text in the user text file (MLS_StringTables.csv)

.

Information on List Call (List Identification)

Various list views can also be called via F-keys or M-keys in the MTX. For this purpose, there are 16 defined list call functions for the list ID "ToolList 1 - 16"(without stating the list name). Further lists can be viewed by stating the list name in the parameter field above the function "ToolList".

Definition of List Content

Here, 'list content' is understood to be the selection of the indicated tool data records.

The query definitions in the file specify which data records are to be shown in a tool list.

The key "Configuration" can be used to directly call the query configurator for modifying the selected query definition.

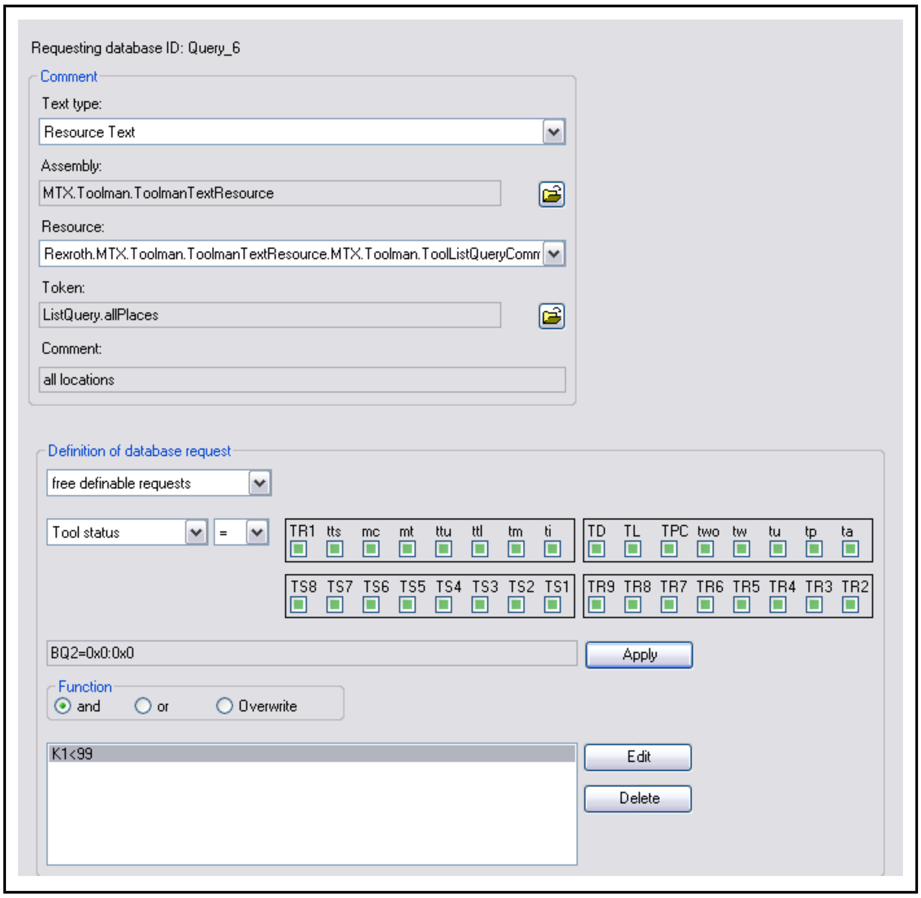


Fig. 16-92: Query configurator

The query editor allows, in addition to the actual query definition, a query comment in the "Comment" dialog. This comment is displayed next to the list title in the headline row. Both elements are linked via a query ID in the list configuration.

Furthermore, the query configurator allows to define the database query in a user-friendly manner in the "Definition of the database query" area of the dialog.

Here, the user can decide in the first selection box whether he wants to use predefined queries or to create a freely defined query. A freely defined query is created on the screen using data element "Tool status (BQ2)".

The following default predefined queries have been supplied:

Description	Query text
All worn tools	BQ2=0x10:0x10
All sectors	K1<99
Sector 1	K1=1
Sector 2	K1=2
Sector 3	K1=3
All tools	SKQ<>''

Channel-dependent database query

The tool list can be visualized with regard to the channel.

Therefore, the default query with the ID "ChannelQuery" exists in a newly created project.

This query definition is already provided with a predefined, but changeable title "**active channel**" that appears additionally in the list title as for each query comment.

As shown in the following figure, the query configurator for this query type has an additional input box to select the channel.

Configuring the Tool Management

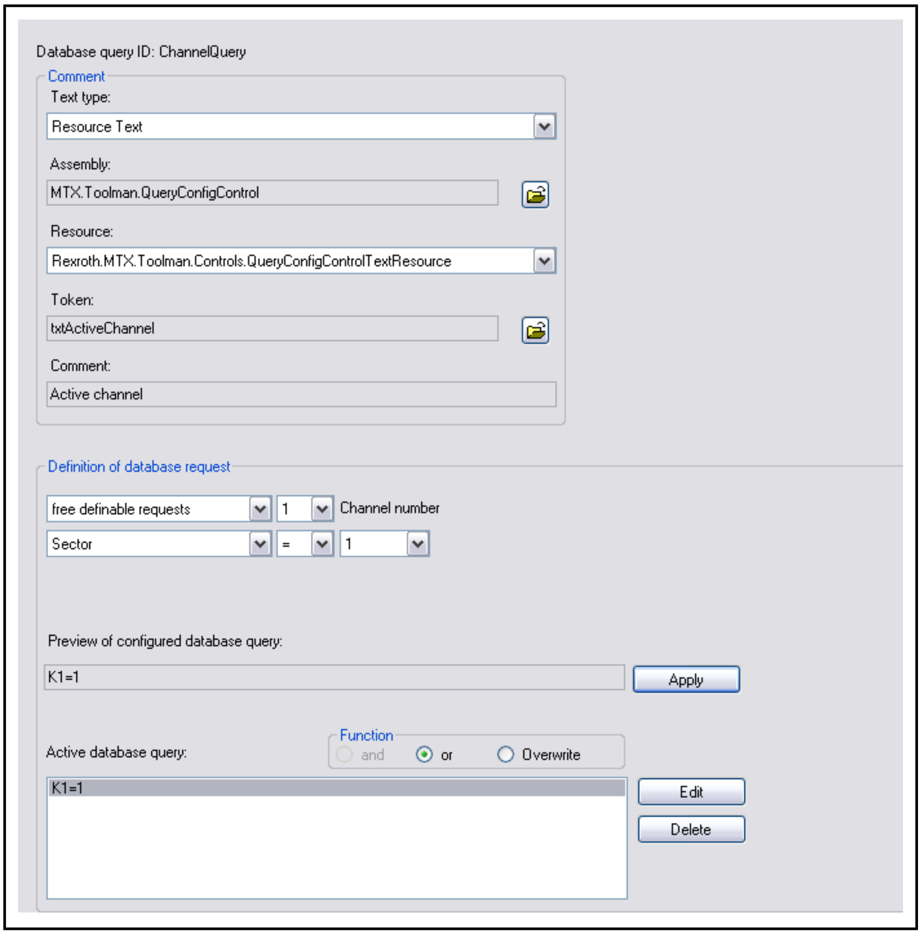


Fig. 16-93: Channel-dependent query definition

This channel-dependent query definition is saved in a channel-dependent system variable "**SD.SysChannelQuery**" which is analyzed in the tool list display at runtime.

Database Queries

Using the <Edit button>, the experienced user can enter the query condition for defining the list contents with no restraints and without any configuration help.

The following compare operators are admissible. Meaning of the data elements:

Operator	Syntax	Note
Equals	<header component> = <value>	
Greater than	<header component> > <value>	Not for SKQ
Greater than or equal to	<header component> >= <value>	Not for SKQ
Less than	<header component> < <value>	Not for SKQ
Less than or equal to	<header component> <= <value>	Not for SKQ
Not equal to	<header component> <> <value>	
Bit mask	<header component> = <value> : <mask>	Not for SKQ

Fig. 16-94: Syntax for query string

Example:

K1=1, IKQ2=1001, BQ2=0x10:0x10

In example 1, all worn twist drills are shown in sector 1.

Example:

SKQ <> “

In example 2, all tools in the database are shown.

For interlinked query conditions, several QueryStr's can be defined for a list.

F Key Panel Definition

F-key bar

The valid call panel for the tool list to be defined can be selected from the F key configurations contained under the project node "F panels".

The key "Configuration" can be used to directly call the F key configurator for modifying the desired F key bar.

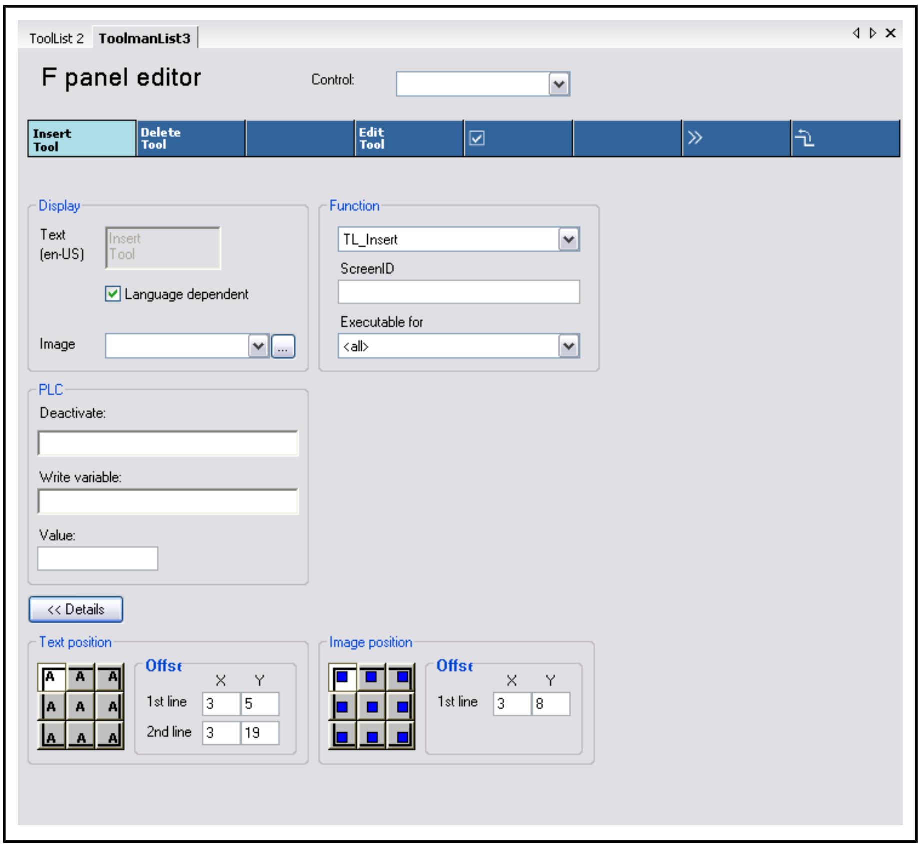


Fig. 16-95: F-key Configurator

Tool management makes the following functions available for the available F key/M key configuration:

Name	Description	Parameter / notes
TL_Copy	Copy tool	-
TL_Delete	Delete tool	-
TL_Edit	Edit tool	-
TL_Edit_Additive	Switching additive ↔ absolute	-
TL_Export	Export list etc.	-

Configuring the Tool Management

Name	Description	Parameter / notes
TL_Import	Import list	-
SingleTool_Import	Import single tool	-
TL_Insert	Insert tool	-
TL_Move	Move tool	-
TL_Editor_Command	Only applicable to M key definitions in the tool editor	(refer to fig. 16-97 "M key definitions in the tool editor" on page 392)
TL_ChangeConfig	Modification of the list configuration	Name of the configuration file
TL_ChangeQuery	Modification of the list query	Query ID
TL_ActivateNextList	Switching to the next list if several lists are displayed	-
TL_SPS_Copy	Copy tool using PLC	PLC var
TL_SPS_Delete	Delete tool using PLC	PLC var
TL_SPS_Move	Move tool using PLC	PLC var
ToolCursor	Indicating the cursor position to the PLC	PLC var
ToolList	Universal list call	ListID
ToolList 1	Call A_ToolList_1 (predef.)	-
:	:	:
ToolList 16	Call A_ToolList_16 (predef.)	-

Fig.16-96: Function table for F and M key configuration

Command strings for the function TL_Editor_Command

Name	Description
Delete_BasicData_ToolType	Delete the basic data record
Save_BasicData_ToolType	Save the basic data record
Edit_Next_Tool	Selection of the next tool
Edit_Prev_Tool	Selection of the previous tool

Fig.16-97: M key definitions in the tool editor

The following PLC data structures are required for the functions with a PLC connection:

TL_SPS_Copy

Sequence:

1. Select the tool to be copied (cursor position)
2. Enter the target place using the selection dialog
3. The interface activates the PLC using the structure variable (Act = TRUE).
4. The PLC executes the copy function (writes to the database)
5. The PLC returns an acknowledgement to the interface; in a negative case, this results in an error being output in the status bar. The interface expects this answer from the PLC within 500 ms. If no answer arrives, an error message is also issued.

Structure of PLC var:

```
TYPE stCopy:
STRUCT
```

```
        SrcPlace :      INT;        Location (source)
        SrcStorage :    INT;        Sector (source)
        DstPlace :      INT;        Location (target)
        DstStorage :    INT;        Sector (target)
        Act:            BOOL;       Activation
        Value:          INT;        Error return
END_STRUCT
END_TYPE
```

TL_SPS_Delete

Sequence:

1. Select the tool to be deleted (cursor position)
2. Safety prompt via the dialog "Delete: yes/no"
3. The interface activates the PLC using the structure variable (Key = TRUE).
4. The PLC executes the delete function (writes to the database)
5. The PLC returns an acknowledgement to the interface; in a negative case, this results in an error being output in the status bar. The interface expects this answer from the PLC within 500 ms. If no answer arrives, an error message is also issued.

Structure of PLC var:

```
TYPE stDelete:
STRUCT
        K1:             INT;        Sector
        K2:             INT;        Place
        Key:            BOOL;       Activation
        Value:          INT;        Error return
END_STRUCT
END_TYPE
```

TL_SPS_Move

Sequence:

1. Select the tool to be moved (cursor position)
2. Enter the target place using the selection dialog
3. The interface activates the PLC using the structure variable (Act = TRUE).
4. The PLC executes the copy function (writes to the database)
5. The PLC returns an acknowledgement to the interface; in a negative case, this results in an error being output in the status bar. The interface expects this answer from the PLC within 500 ms. If no answer arrives, an error message is also issued.

Structure of PLC var:

```
TYPE stMove:
STRUCT
        SrcPlace :      INT;        Location (source)
        SrcStorage :    INT;        Sector (source)
        DstPlace :      INT;        Location (target)
        DstStorage :    INT;        Sector (target)
```

Act:	BOOL;	Activation
Value:	INT;	Error return
END_STRUCT		
END_TYPE		

ToolCursor

Sequence:

1. Select the tool location (cursor position) in the structure variables (K1 and K2).
2. The interface activates the PLC using the structure variable (Key = TRUE).
3. The PLC executes the specified function.

Structure of PLC var:

TYPE stCurPos:		
STRUCT		
K1:	INT;	Sector
K2:	INT;	Place
Key:	BOOL;	Activation
END_STRUCT		
END_TYPE		

M Key Panel Definition

M Key Bar on the Left and on the Right

The valid left and right M key panels for the tool list to be defined can be selected from the M key configurations contained under the project node "M panels".

The key "Configuration" can be used to directly call the M key configurator for modifying the desired M key bar.

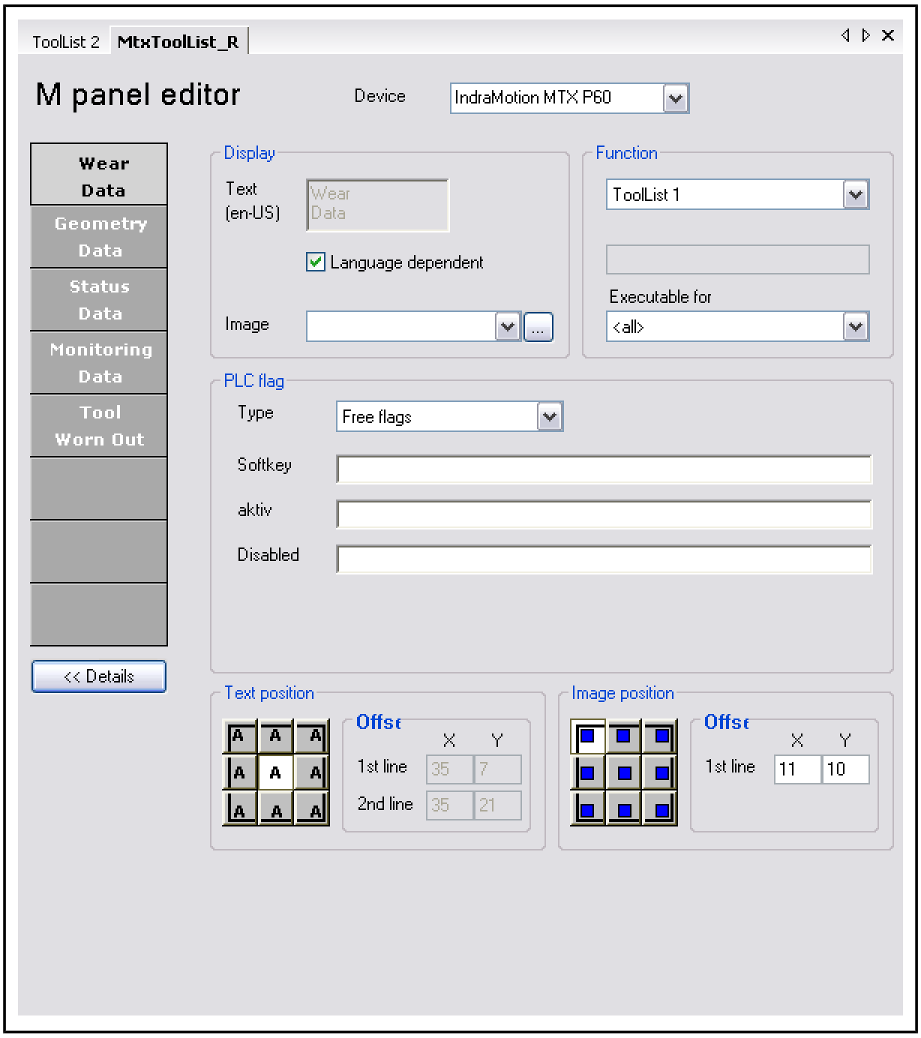


Fig. 16-98: M key configurator

For the functions, see fig. 16-96 " Function table for F and M key configuration" on page 391.

M key bars can be assigned to the tool lists using the following 2 methods:

Prio	Variant	Implementation
1	List-specific (local) M keys	As described above, M key bars are assigned to the tool list in the list configurator.
2	Global M keys	M key bars are not assigned to either the list or the screen.

Fig. 16-99: Variants of M key configuration

Definition of List Display

Tool Lists Configurator

There are two ways of viewing the tool lists configurator. However, they use the same database (configuration file), i.e. changes made in the configurator have the same effect on the XML editor and vice versa, i.e changes made in the XML editor have the same effect on the configurator.

1. Configurator for common configuration steps
2. XML editor for special settings which cannot be made by means of the configurator.

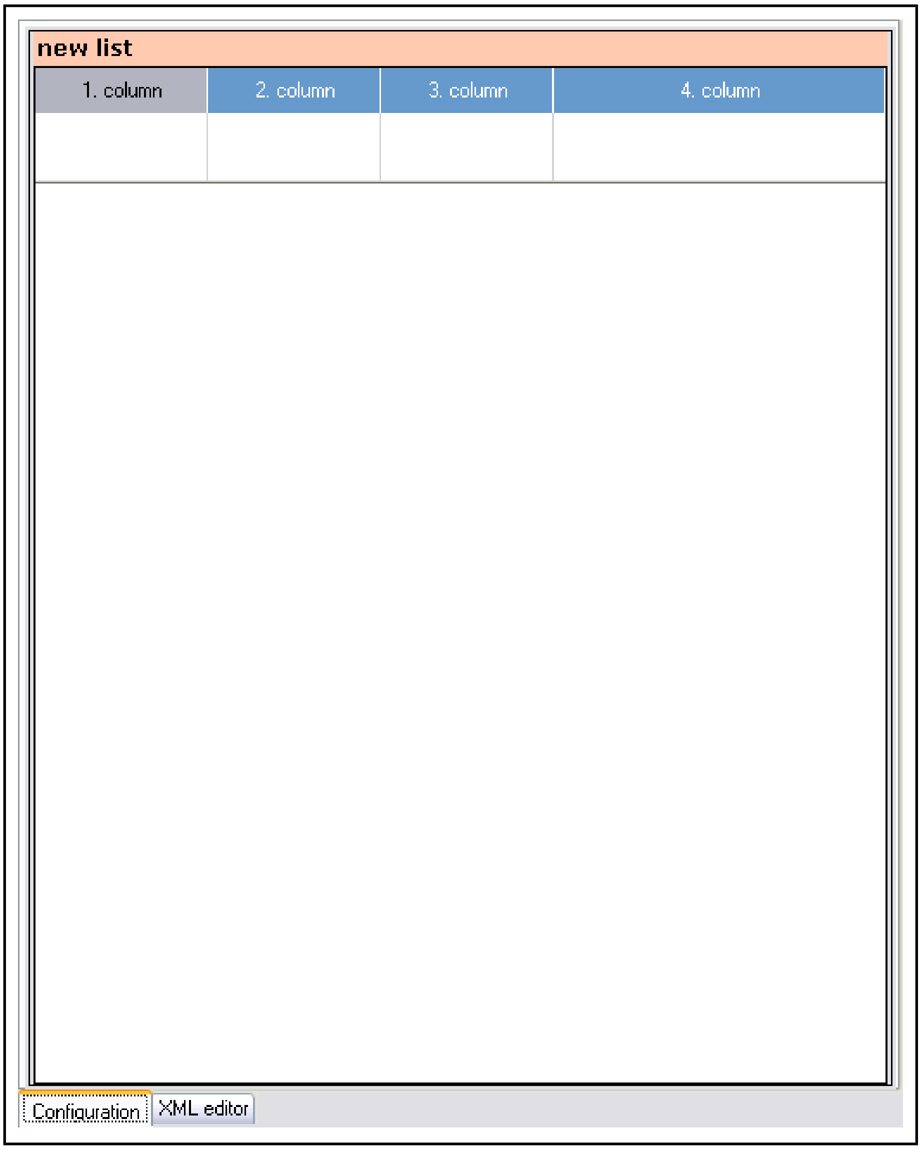


Fig.16-100: Tool lists configurator

Most of the settings can be made using the ULC configurator (configuration) (see chapter 16.2.4 "ULC Configurator" on page 311). Only in exceptional cases do special changes need to be made directly in this configuration file using the XML editor (see chapter 16.2.5 "XML File Editor" on page 329). These cases are described in the next section.

Tool Lists Configuration File

The valid configuration file defining the list structure for the tool list to be defined can be selected from the controls list contained under (1).

The list structure is defined by the tool list configuration file assigned to the tool list.

All tool list configuration files are based on the schema file **"UniversalListcontrol.xsd"**.

The following overview in the appendix chapter 25.2 "Tool Lists Configuration File" on page 581 shows whether the list/editor control configuration file can be opened using the ULC configurator for all process parameters.

Description of application-specific display modifications outside of the ULC configurator

Setting the Compressed Display Mode

If work is carried out with a list configuration that displays the data record in several partial lists (e.g. using **MTX.Toolman.ToolListConfiguration.xml**), the following handling instruction can be used to switch the display to a compressed version:

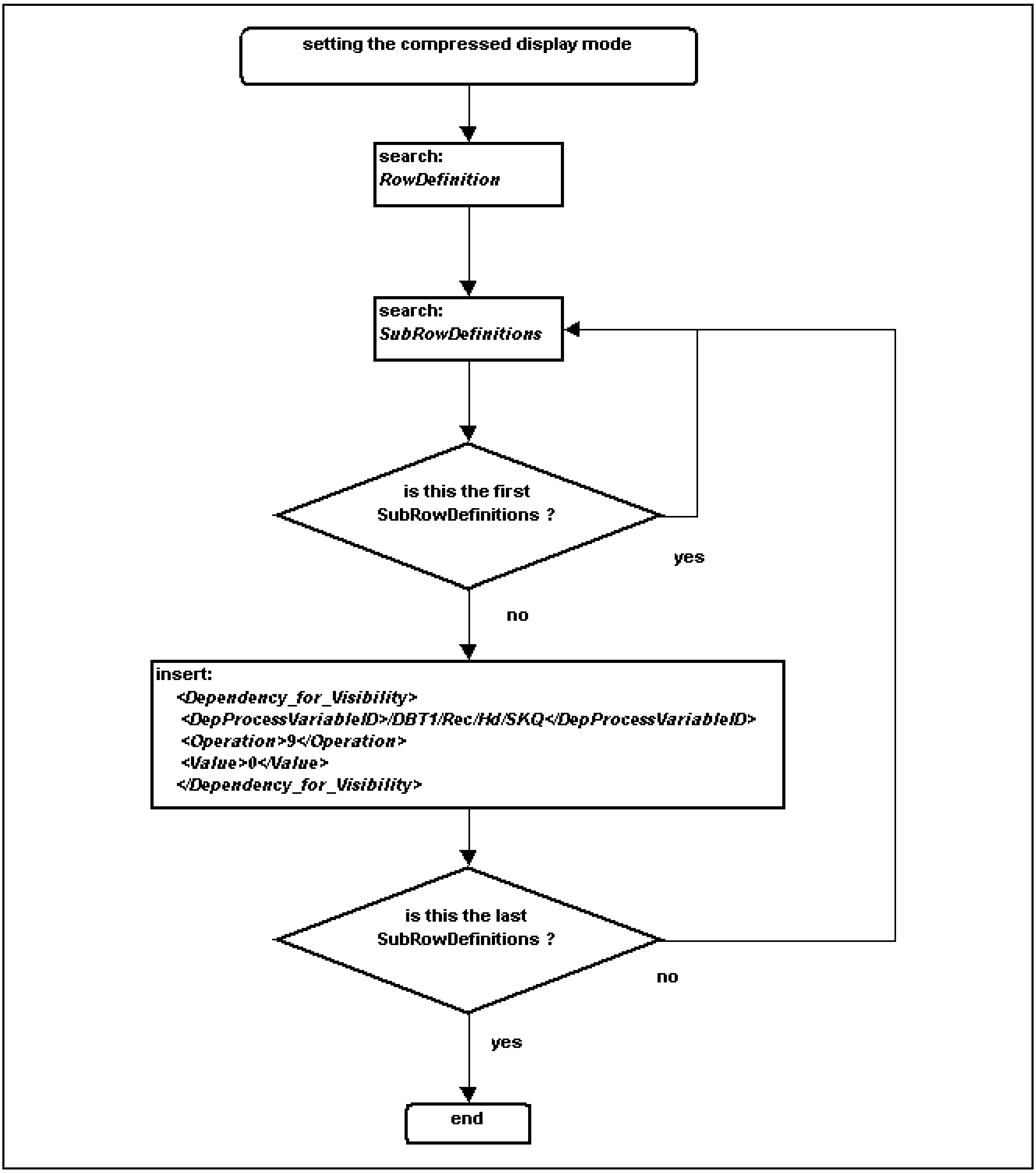


Fig. 16-101: Setting the compressed display mode

Configuring the Tool Management

Conditional Display of Cell Contents

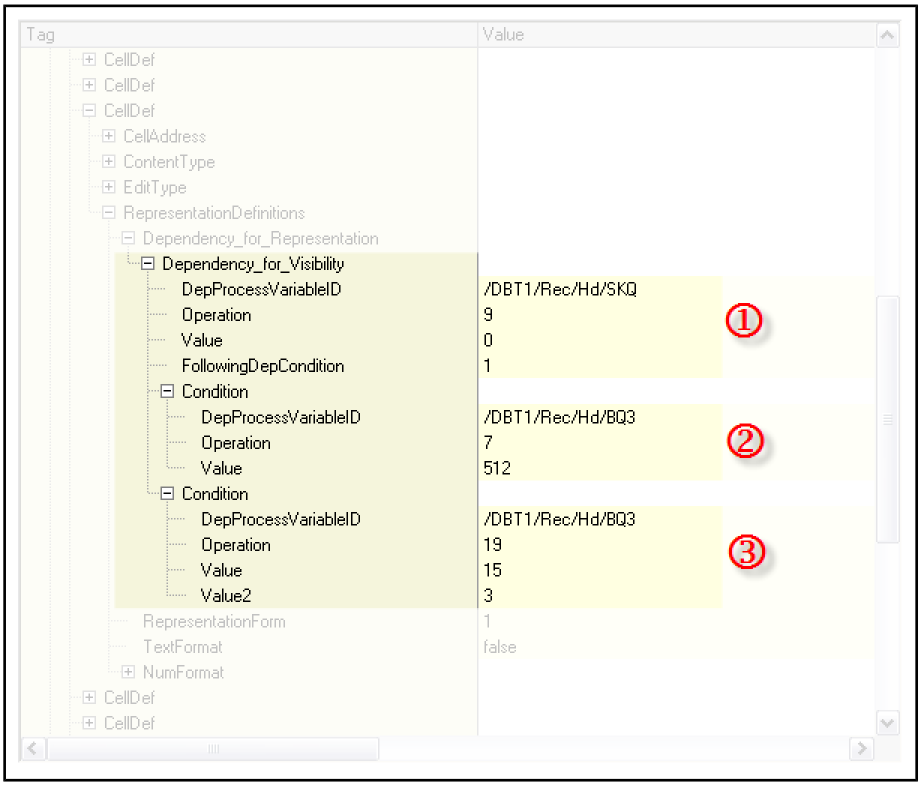
During the editor/list configuration, it is possible to define the editability of a data element depending on an additional process variable.

The setting is made in the configuration file in configuration node **"Dependency_for_Visibility"** (see P1 - P5).

☞ The requirement for this function is that the edit status = 3 for this cell.

Example:

The value of the tool edge-dependent data element "Geometry value L2" is displayed depending on data elements SKQ (tool name) and BQ3 (tool technology code) (determined by the CellDef node to which the "Dependency_for_Visibility" definition belongs).



1 1. Display condition
2 2. Display condition
3 3. Display condition

Fig. 16-102: Example: Conditional display of cell contents

1. Display condition:	P.stringlength longer V	Tool name may not be an empty string, i.e. the location must contain a tool.
2. Display condition:	P AND V != 0	Bit 10 in the technology code word must be set, i.e. the L2 correction value must be relevant for the type of the displayed tool data record (see tool catalog).
3. Display condition:	(P AND V) higher or equal {MN}	The no. of the subordinate column is less than or equal to the number of tool edges of the tool data record, i.e. the tool edge datum is relevant for the number of tool edges of the data record
Explanation for MN:	MN = 0	Partial list

MN = 1	Subordinate line
MN = 2	Column
MN = 3	Subordinate column

Setting a Process-Dependent Bitmap Display

It is possible to display bitmaps from resource files depending on the value of a process variable in the tool list.

Since the bitmap files must be a component of a resource DLL, a default resource DLL (assembly file) is included; in turn, this contains the following bitmap resource files (resx files):

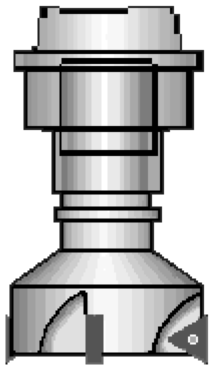
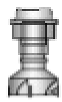
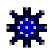
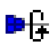
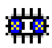
AssemblyName	BitmapResourceName	Contents	Example (original size)
MTX.Toolman.ToolmanUser-BitmapResource	ToolTypes_MediumSizeBmp.resx	Tool type	
	ToolTypes_SmallSizeBmp.resx	Tool type	
	ToolTypes_IconSizeBmp.resx	Tool type	
	ToolStorageBmp.resx	Storage type	
	ToolTecTypesBmp.resx	technology type	

Fig. 16-103: Supplied bitmap libraries

Example:

Storage Type Bitmap

A bitmap is assigned to the value of K1 (sector number) using an assignment table.

Value of K1	Bitmap	Meaning
1		Spindle
2		Grippers
3		Magazine feeding attachment

Configuring the Tool Management

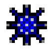
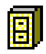
Value of K1	Bitmap	Meaning
4		Turret
99		Tool cabinet

Fig.16-104: Example storage type bitmaps

Settings in the configuration file:

1. Setting: Display of the process variables as a bitmap.

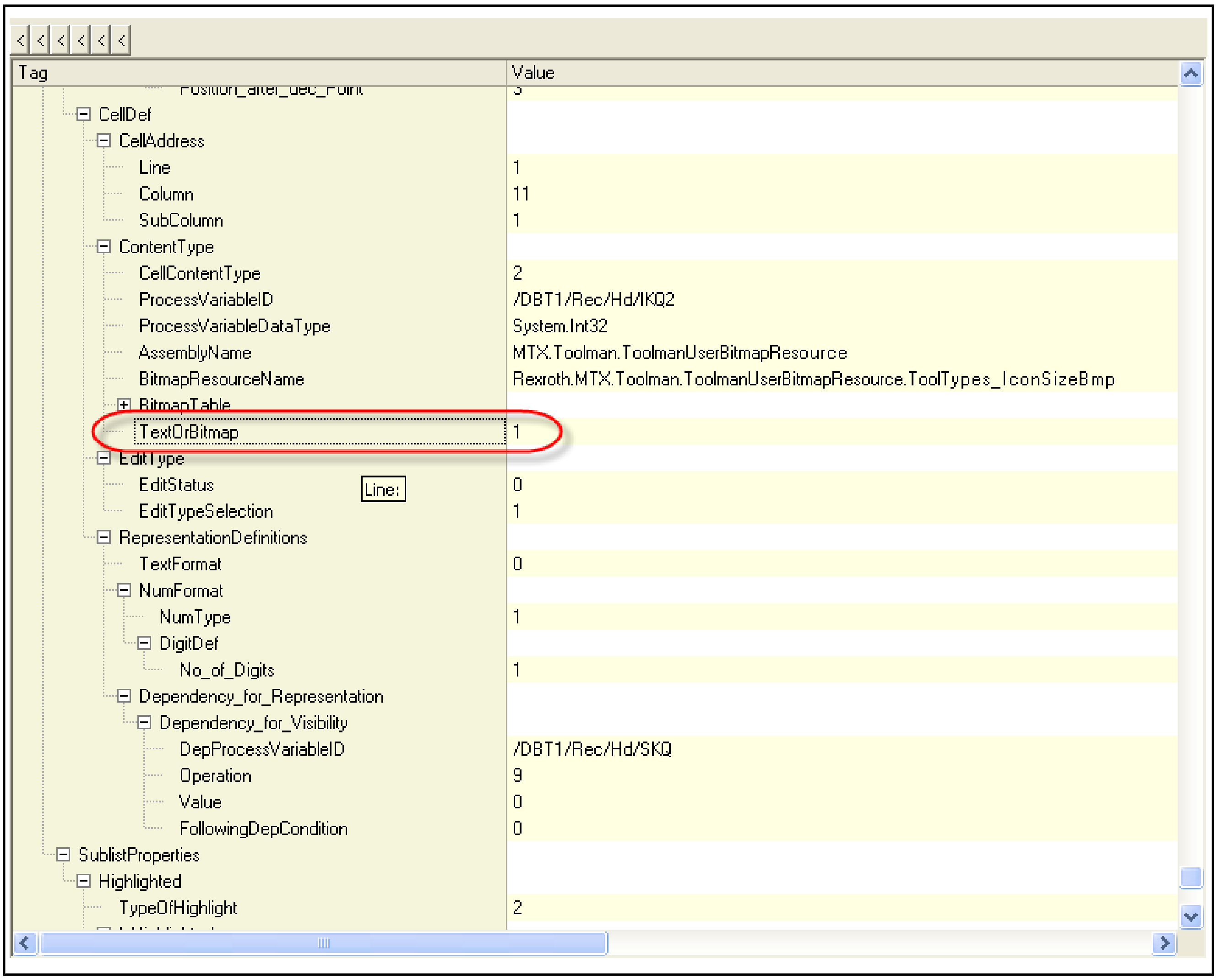


Fig.16-105: Setting the display of the process variables as a bitmap

2. Define AssemblyName, BitmapResourceName and the bitmap table (see Q1 - Q2).

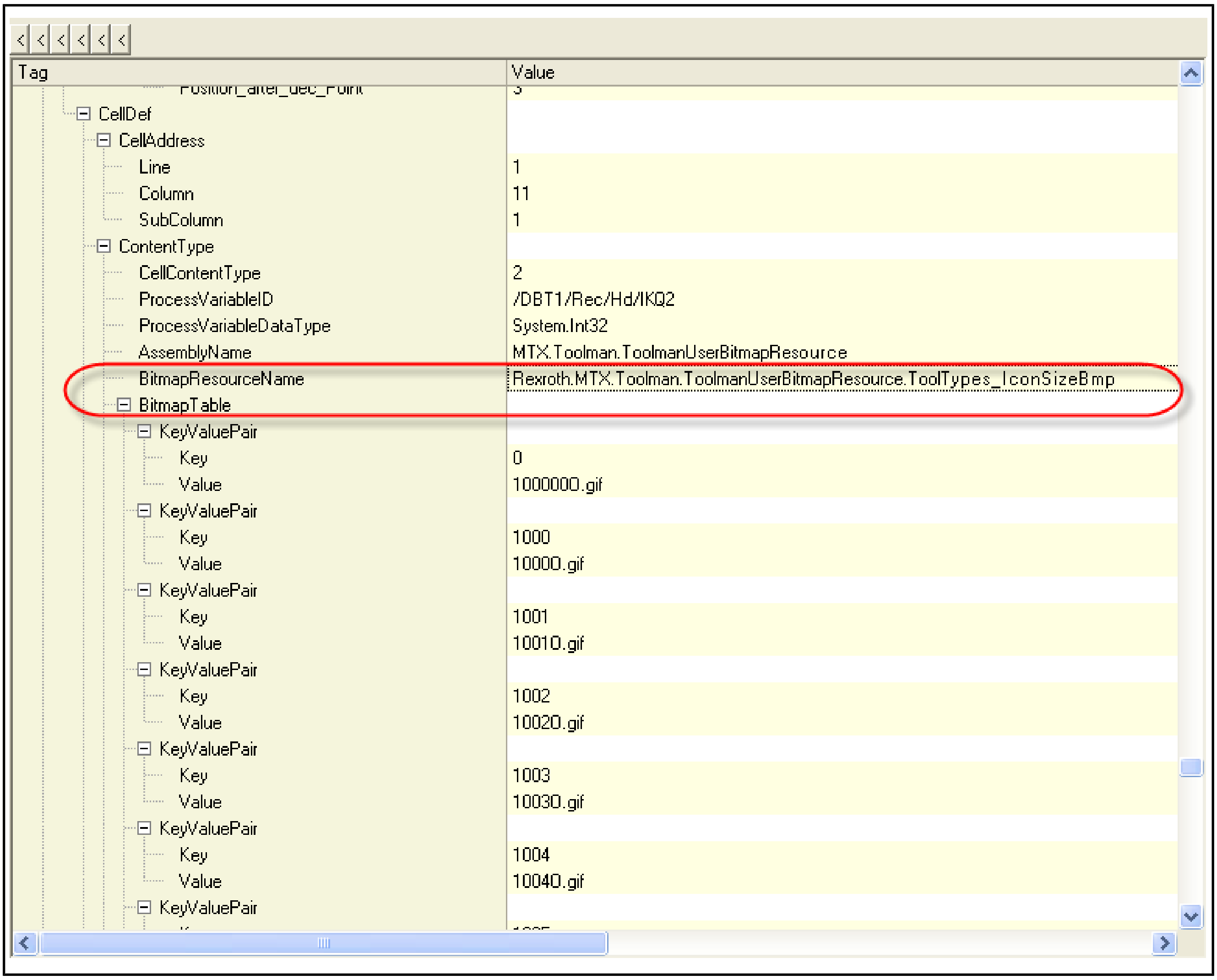


Fig.16-106: Bitmap links

Setting a Process-dependent Text Display

It is possible to display bitmaps and texts from resource files depending on the value of a process variable in the tool list.

The texts can either be firm texts or can originate from a text resource DLL or from a user text file.

Settings in the configuration file:

1. Setting: Display of the process variables as a text.
2. Via "**TextTableUse**" it can be set if the text is to be a determined text (value = 0), a resource text (value = 1) or a user text (value = 2).
3. Definition of AssemblyName, TextResourceName (in the case of resource texts being used) and the text table (see Q 1 - Q2).

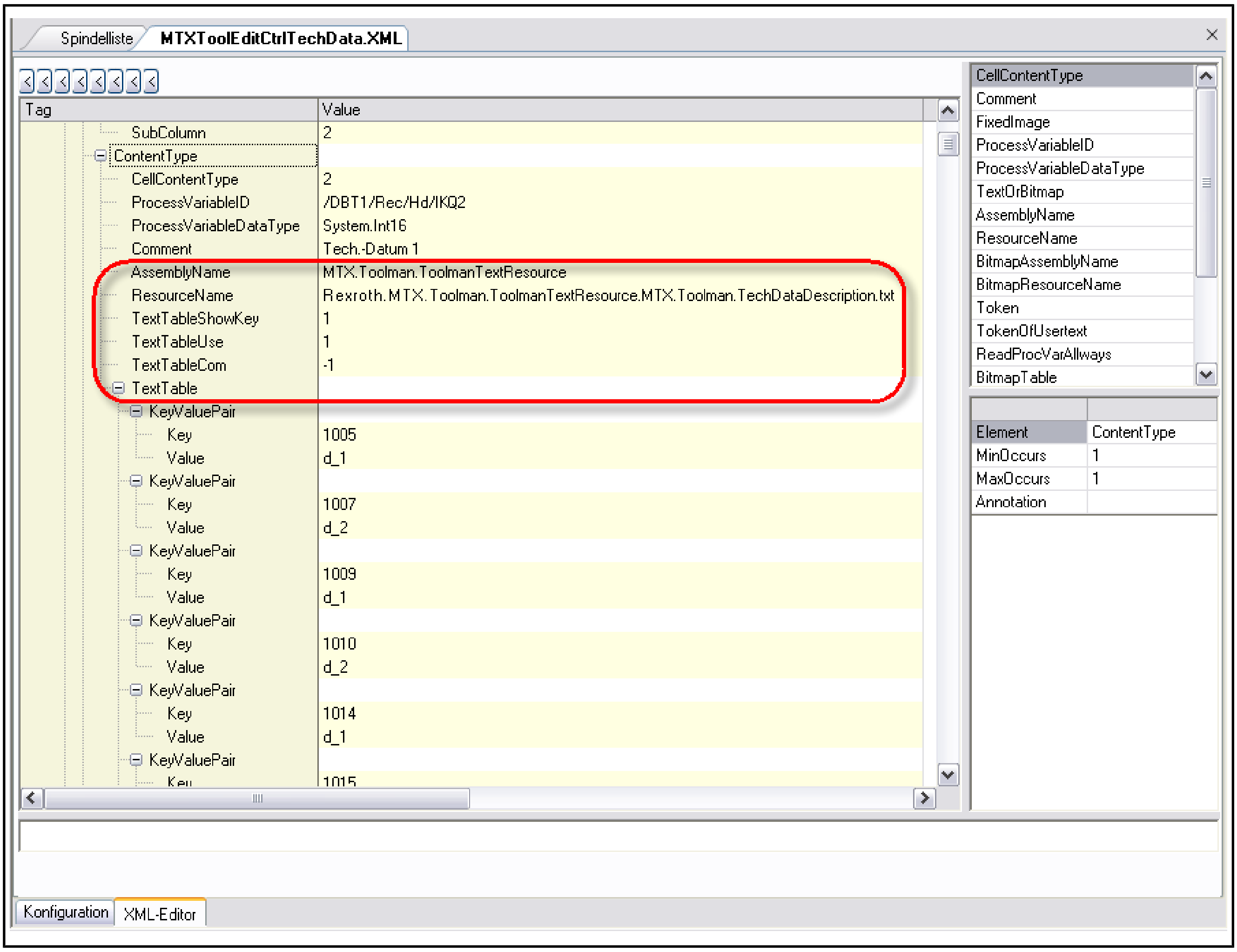


Fig. 16-107: Text references

4. Via "**TextTableShowKey**" it can furthermore be set how the display reacts in the case of the defined display conditions of the text table not being fulfilled. is to be (), a resource text (value = 1) or a user text (value = 2).

   - Value = 0: the cell is displayed empty
   - Value = 1: the cell is displayed as if the process value = 0

Conditional Editability of Cell Contents

During the editor/list configuration, it is possible to define the editability of a data element depending on an additional process variable.

The setting is made in the configuration file in the configuration node **"EditDepend"** (see P1 - P5).

Example:

Conditional Editability

The editing of a selected tool datum is permitted depending on data element BQ2 - bit 7 (Tool locked).

BQ2 AND 64 != 0

(Operation 7: P AND V != 0)

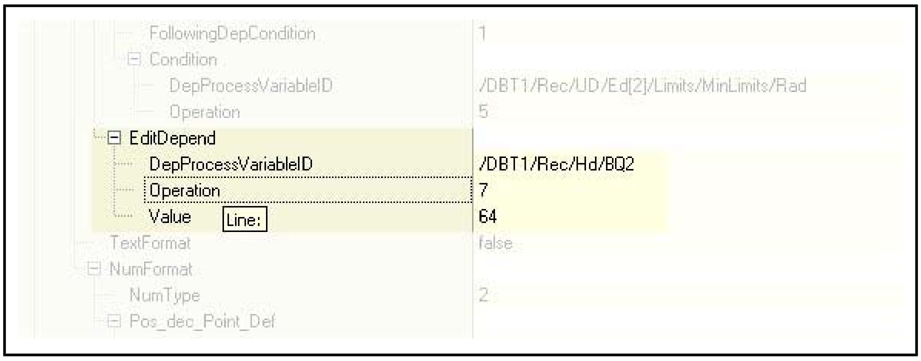


Fig. 16-108: Example of conditional editability

☞ The requirement for this function is that the **edit status = 3** in the edit type definition.

Highlighted Definition of Data Records

It is possible to highlight tool data records in the tool list depending on a process variable. As many displays as desired can be implemented.

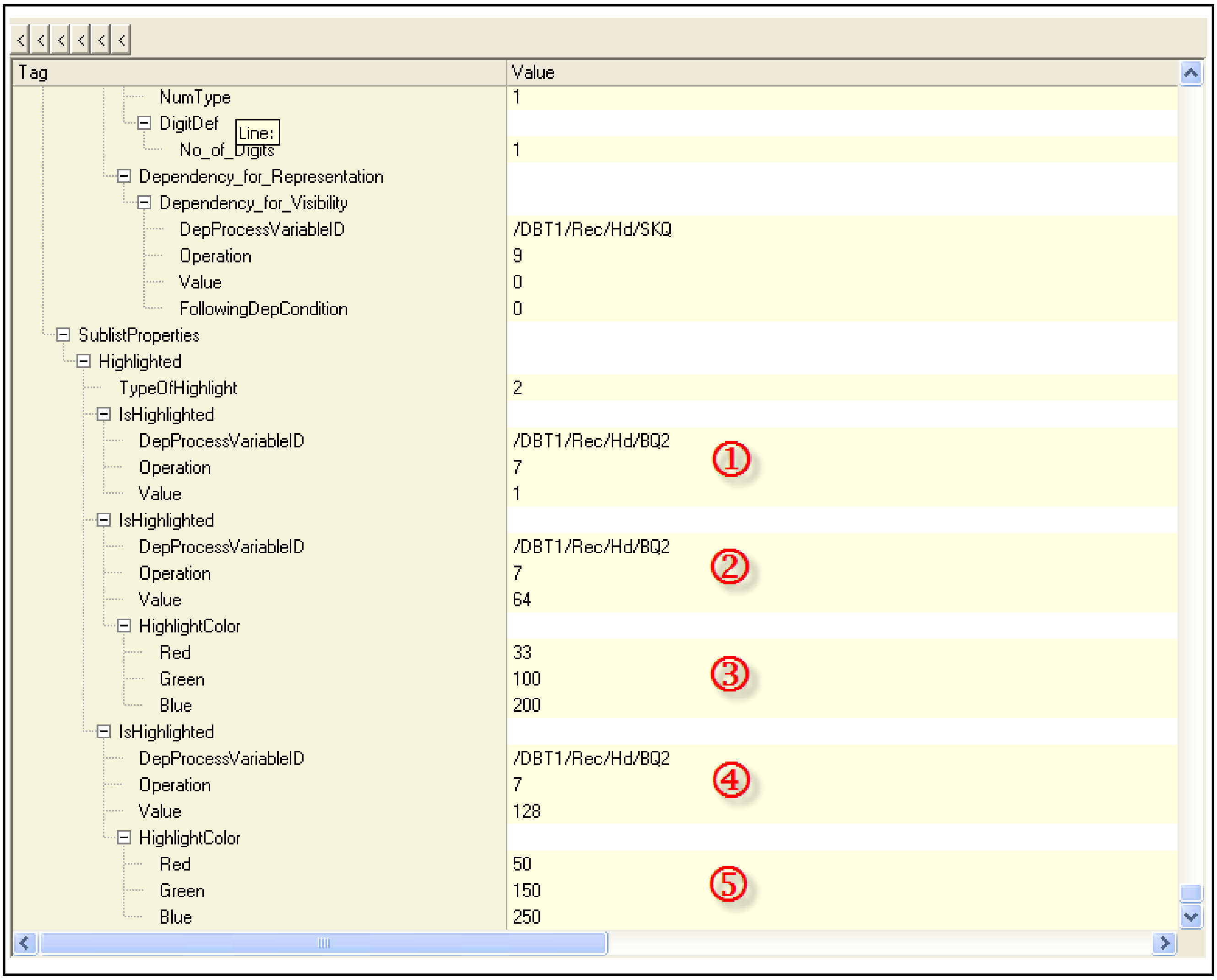
Example:

Highlighted Definitions

BQ2 - bit 0 (Tool active) → background color 1

BQ2 - bit 4 (Tool worn) → background color 2

Configuring the Tool Management



1	1. Highlighted condition
2	2. Highlighted condition
3	Color setting for the background color of the 2nd definition
4	3. Highlighted condition
5	Color setting for the background color of the 3rd definition

Fig. 16-109: Example for several highlighted definitions

☞ Color settings, including the background color for the 1st definition, are always located in the configuration node **Styles/Highlighted**; the foreground color that is defined there applies to all highlighted definitions.

Definition of Value Range Checks During the Entry of Values

In addition to limit value monitoring in the PLC, it is possible to configure the tool list / tool editor in such a manner that only the entry editor carries out a value range check during entry.

If the value range is exceeded, an error message is displayed in the status bar.

As is the case for the check using the PLC, it is required that limit values be defined in the data record (see fig. 16-51 "Data record schema (13) (life data upon delivery)" on page 352).

The setting for the check condition is made in the configuration file in configuration node **"Validation"** (see P1 - P5).

Here, you can also write:

instead of /DBT1/Rec/...

/DBT1/Rec[{0}]/...

Example:

Value Range Check during Entry in the Editor

(value of the limit R_{max} of tool edge 2) ≥ input value ≥ (value of the limit R_{min} of tool edge 2)

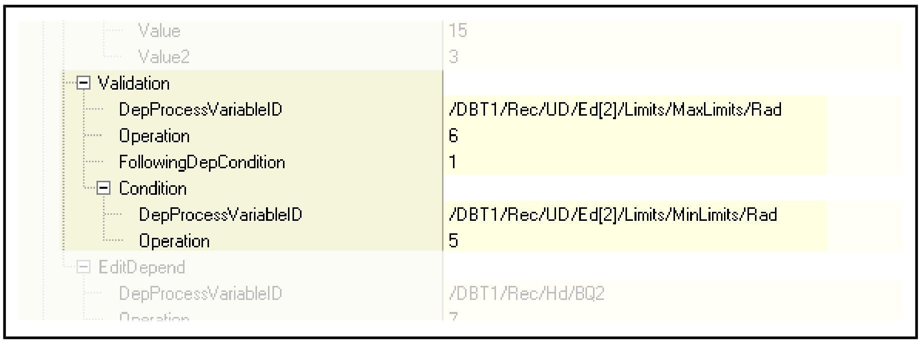


Fig.16-110: Example of value range inspection during entry in the editor

Possible operations (generally applies to the DependencyType):

Explanation of abbreviations:		
P	Process variable	(ProcVariableID)
V	Value	(Value)
V2	2. Value	(Value 2)
Explanation for MN:	Value is preset by V2	
	MN = 0	Partial list
	MN = 1	Subordinate line
	MN = 2	Column
	MN = 3	Subordinate column
Permitted operations		
	The value is written if:	
1	P==TRUE	The process variable log. is TRUE.
2	P<V	The process variable is less than the specified value.
3	P>V	The process variable is greater than the specified value.
4	P==V	The process value is equal to the specified value.
5	P≤V	The process value is less than or equal to the specified value.
6	P≥V	The process value is greater than or equal to the specified value.
7	P∧V≠FALSE	The process value is linked to the specified value with log. AND and the result is log. TRUE.

8	˜P∧V≠FALSE	The neg. process variables are linked to the specified value with log. AND and the result is log. TRUE.
9	strlen(P)<V	The string length of the process variables is higher than the specified value (P of the string type, e.g. SKQ).
10	(P∧V)<V2	The process value is linked to the specified 1. value with log. AND and the result is less than the specified 2. value.
11	(P∧V)>V2	The process value is linked to the specified 1. value with log. AND and the result is greater than the specified 2. value.
12	(P∧V)==V2	The process value is linked to the specified 1. value with log. AND and the result is equal to the specified 2. value.
13	(P∧V)≤V2	The process value is linked to the specified 1. value with log. AND and the result is less than or equal to the specified 2. value.
14	(P∧V)≥V2	The process value is linked to the specified 1. value with log. AND and the result is greater than or equal to the specified 2. value.
15	(P∧V)<MN[V2]	The process value is linked to the specified 1. value with log. AND and the result is less than the value of the MN process variable specified by V2.
16	(P∧V)>MN[V2]	The process value is linked to the specified 1. value with log. AND and the result is greater than the value of the MN process variable specified by V2.
17	(P∧V)==MN[V2]	The process value is linked to the specified 1. value with log. AND and the result is equal to the value of the MN process variable specified by V2.
18	(P∧V)≤MN[V2]	The process value is linked to the specified 1. value with log. AND and the result is equal to or less than the value of the MN process variable specified by V2.
19	(P∧V)≥MN[V2]	The process value is linked to the specified 1. value with log. AND and the result is greater than or equal to the value of the MN process variable specified by V2.

Handling Instruction: Configuring the list display

This handling instruction refers to the procedure described for configuring the display form of a tool list.

Configuring the list display

The following subtasks can be accomplished:

1. Adding/deleting a column
2. Adding/deleting a subcolumn
3. Changing a column or subcolumn title
4. Changing the column or subcolumn width
5. Setting the editing properties

See chapter 16.2.4 "ULC Configurator" on page 311.

		Documentation
Documentation	MTX Functional Description	Tool list configuration

IW Engineering: Data Transfer

1. Save the configuration
2. Save and, if necessary, activate the project

		Documentation
Documentation	IndraWorks HMI	Data Transfer

Handling Instruction: Configuring List Content

The following handling instruction describes the process to be followed to edit the displayed content of a list.

IW Engineering / Tool List Configuration: Enter or Double-Click a List Configuration

1. Selecting of list configuration files
2. Select the F key panel and configure it, if necessary
3. Select the right and left M key panels and configure them, if necessary
4. Define the list query.

		Documentation
Documentation	MTX Functional Description	Tool list configuration

IW Engineering / Tool list configuration: Configuring the list definition

		Documentation
Documentation	MTX Functional Description	Data Transfer

IW Engineering: Data Transfer

1. Save the configuration
2. Save and, if necessary, activate the project

		Documentation
Documentation	IndraWorks HMI	Data Transfer

Handling Instruction: Create New List Configuration

The following handling instruction describes the process to be followed to create a new list configuration

IW Engineering / Tool list configuration: Pop-up or menu function "New"

A dialog to enter/select the list name appears. This name can be entered during the F/M key configuration while entering the parameter **ListID** in the function **ToolList**.

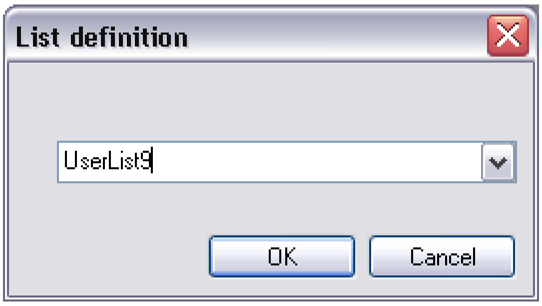


Fig.16-111: Creating a new tool list definition

		Documentation
Documentation	MTX Functional Description	Tool list configuration

IW Engineering / Tool list configuration: Configuring the list display

		Instruction chapter "Handling Instruction: Configuring the list display " on page 406
Instruction:	IndraWorks Commissioning	Configuring the list display

IW Engineering: Data transfer

Execute an HMI download

		Documentation
Documentation	IndraWorks HMI	Data Transfer

16.6.3 Configuration of Tool Editors

General

Brief Description

Like the NC main screens, the tool editors function according to the ACI principle (**A**ctive **C**ontainer **I**nterface).

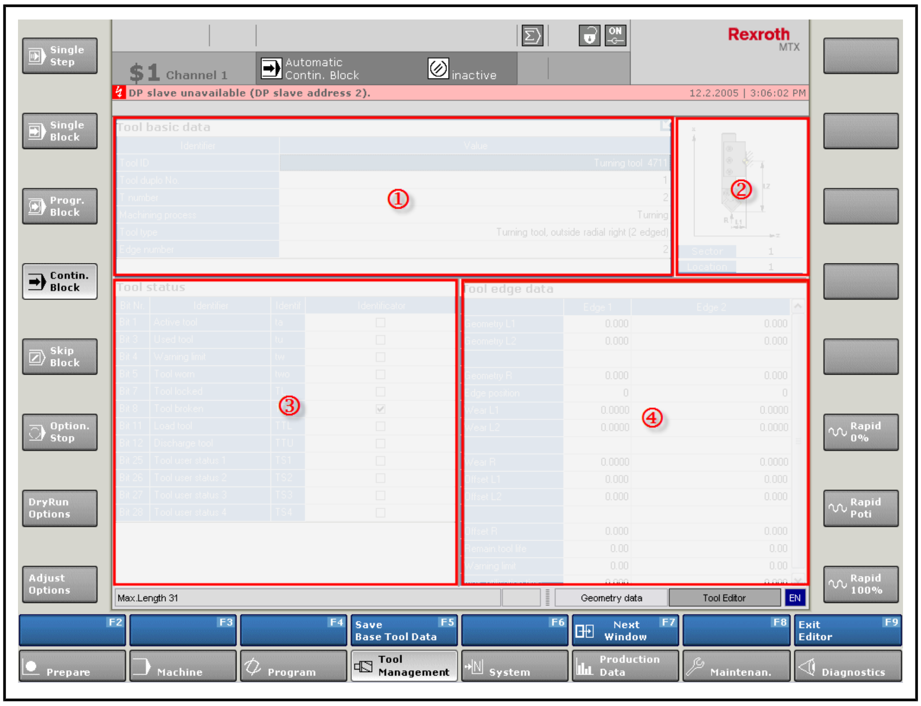
This is to say:

- They can be freely configured.
- They contain several data containers which in turn can comprise several containers.
- The controls themselves can also be configured.
- Navigation of the cursor, and thus editing, always takes place only within the focused container.

Description

The various groups of tool data are shown using separate displays (e.g. tool edge data). These are called controls.

When it is delivered, the Tool Editor has the following screen division.



1	Basic data container
2	Bitmap container
3	General tool data container
4	Tool edge data container

Fig. 16-112: Container distribution of the tool editor

The number and assignment of containers can be changed by the user. This information is stored in the frame configuration file "MTXToolEditor_Data.xml". The control configurations of the containers are assigned as follows in a standard project.

Container No. in the figure	Listcontrol	Name of the configuration file
1	Basic data control	MTXToolBasicDataConfig.XML
2	Bitmap control	Cannot be configured and focused
3	Tool status control	MTXToolEditCtrlToolStates.XML
3	Place status control	MTXToolEditCtrlPlaceStates.XML
3	Technology data control	MTXToolEditCtrlTechData.XML
3	Tool user data control	MTXToolEditCtrlToolBD1.XML
4	Tool edge data control	MTXToolEditConfigUsrData.XML
4	Geometry limit values control	MTXToolEditConfigLimitData.XML

Fig. 16-113: Overview on all editor configuration files

If there are several controls in one container (as for the default configuration), these can be selected via <F3><Next display>.

The bitmap container is an exception. It cannot be focused and the displayed tool screen is switched via the type of the respectively activated display control.

Display of a coordinate system in the bitmap container

It is possible to store the tool screen for a better illustration in a coordinate system.

This display is controlled via the system variable **SD.SysCoordSystem**:

SD.SysCoordSystem.	**Value**	**Meaning**
Value	See table	
Calc	1	Automatic determination of the coordinate system
	0	No automatic determination of the coordinate system
Horizontal	1	Abscissa in positive direction
	0	Abscissa in negative direction
Vertical	1	Ordinate in positive direction
	0	Ordinate in negative direction
X_axis_pref	1	The X axis is the axis preferred to display the coordinate systems for milling tools
Y_axis_pref	1	The Y axis is the axis preferred to display the coordinate systems for milling tools
Z_axis_pref	1	The Z axis is the axis preferred to display the coordinate systems for milling tools

Fig.16-114: Meaning of SD.SysCoordSystem

If no automatic determination of the coordinate system is set (SD.SysCoord-System.Calc = 0), the stored coordinate system can be set as follows via the variable **SD.SysCoordSystem.Value**:

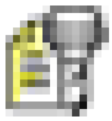
SD.SysCoordSystem	**Abscissa**	**Ordinate**
	Direction	**Direction**
0	-	-
1	-	X
		Positive
2	-	X
		Negative
3	-	Y
		Positive
4	-	Y
		Negative
5	-	Z
		Positive
6	-	Z
		Negative
10	Z	X
	Positive	Positive

SD.SysCoordSystem	Abscissa	Ordinate
	Direction	Direction
11	Z	X
	Negative	Positive
12	Z	X
	Negative	Negative
13	Z	X
	Positive	Negative
20	Y	X
	Positive	Positive
21	Y	X
	Negative	Positive
22	Y	X
	Negative	Negative
23	Y	X
	Positive	Negative
30	X	Y
	Positive	Positive
31	X	Y
	Negative	Positive
32	X	Y
	Negative	Negative
33	X	Y
	Positive	Negative
40	Z	Y
	Positive	Positive
41	Z	Y
	Negative	Positive
42	Z	Y
	Negative	Negative
43	Z	Y
	Positive	Negative
50	X	Z
	Positive	Positive
51	X	Z
	Negative	Positive
52	X	Z
	Negative	Negative

SD.SysCoordSystem	Abscissa	Ordinate
	Direction	**Direction**
53	X	Z
	Positive	Negative
60	Y	Z
	Positive	Positive
61	Y	Z
	Negative	Positive
62	Y	Z
	Negative	Negative
63	Y	Z
	Positive	Negative

Fig.16-115: Control of the stored coordinate system via SD.SysCoordSystem.Value

As a rule, there are two groups of tool editors that can be managed separately in the project tree of Engineering Desktop under project node **"Editor Configuration"**:

Designation		Method of operation	Symbol	Example
Insert type	Offline editor:	Editor for inserting a tool into the tool list		DBT1Insert
Edit type	Online editor::	Editor for modifying a tool data record		DBT1Edit

Newly created tool editors are always assigned to one of the two editor groups during creation by selecting the menu function **New Screen: Edit type** or **New Screen: Insert type** and displayed in the tree with the appropriate symbol. However, this assignment does not yet apply to new editors from legacy projects. But this assignment can be made later on using function **Assign to Edit Screens** or **Insert Screens**.

The distribution, selection and arrangement of the tool data displays of the existing tool editors can be changed using the "Configuration" dialog. New tool editors can be created and ones that are no longer needed can be deleted.

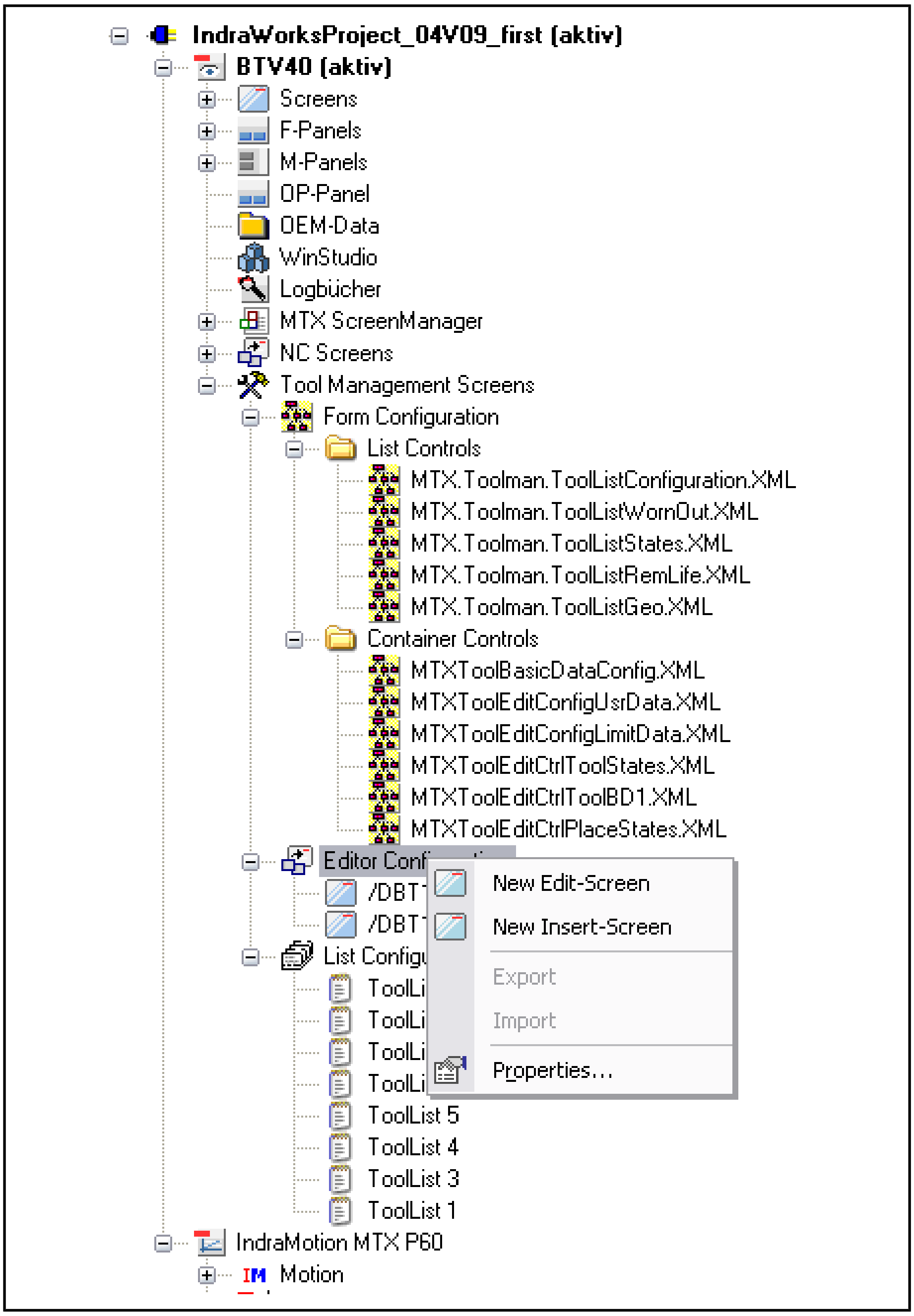


Fig. 16-116: Context menu: Editor configuration

The dialogs and functions to configure properties shared by all the tool editors as well as those to create a new tool editor, can be opened via the context menu of project node **"Editor Configuration"**.

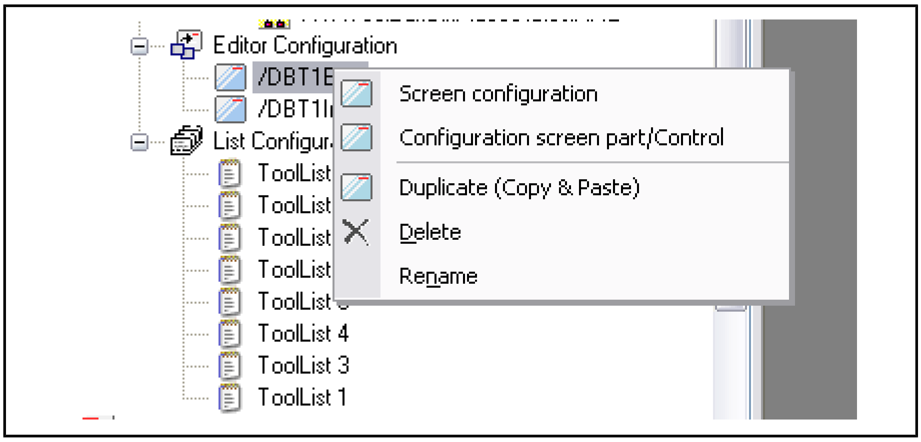


Fig.16-117: Context menu: Configuration (tool editor)

Dialog "New Edit/Insert Screen"

A new editor is created via **New Screen** that is opened in the context menu of the project node **"Editor Configuration"** with the entry **New Screen: Edit type** or **New Screen: Insert type** .

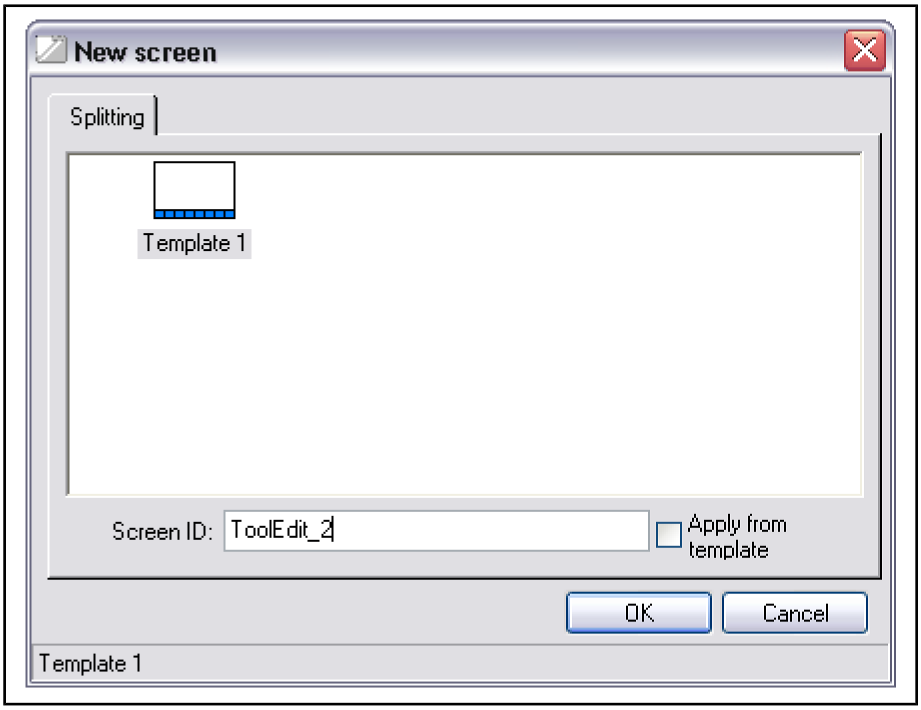


Fig.16-118: Dialog: New screen (with layout templates)

To create a new screen, it is necessary to select a layout template and to enter a unique screen ID. The screen ID can be entered in the input field or transferred to the layout template by selecting the control box.

The layout and content of the screen segment of the screen that is generated can be adapted in the dialog "Configuration screen" and "Configuration screen segment/control".

☞ It is not possible to create a new screen with a name (screen ID) that is already in use. If you attempt to do this, a corresponding error message appears.

Dialog "Properties of All Screens"

This dialog is opened with the context menu item **Properties...** of the project node **"Editor Configuration"**.

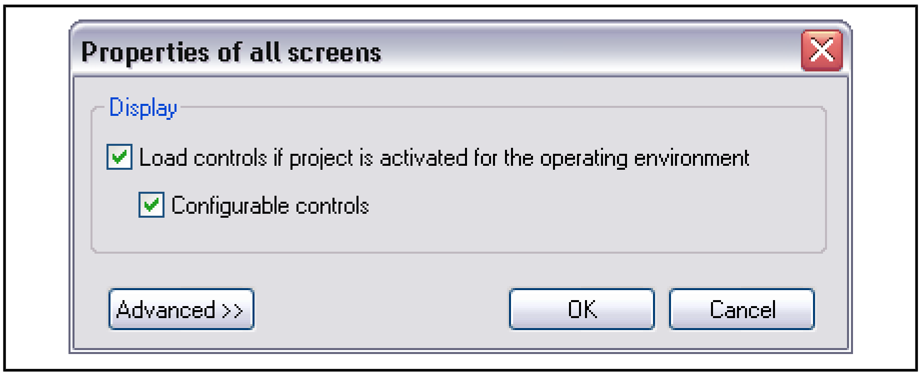


Fig. 16-119: Dialog: Properties of all screens

The "Display" group indicates whether the controls have been initialized in dialogs "Configuration screen" and "Configuration screen segment/control" if the current project has been activated for Operation Desktop or if, instead, only the name of the controls is shown as their placeholders. Furthermore, you can specify whether a "Configuration control" button is to be displayed in dialog "Configuration screen segment/control". This button calls the configuration dialog of the focused control if the control has been initialized and if it has its "Configuration" dialog.

☞ This dialog can be called under the "EditorConfiguration" node, but has no effect, because, in general, only the names of the controls are indicated as far as tool editors are concerned.

- **Advanced >>**:

  Shows the advanced dialog.

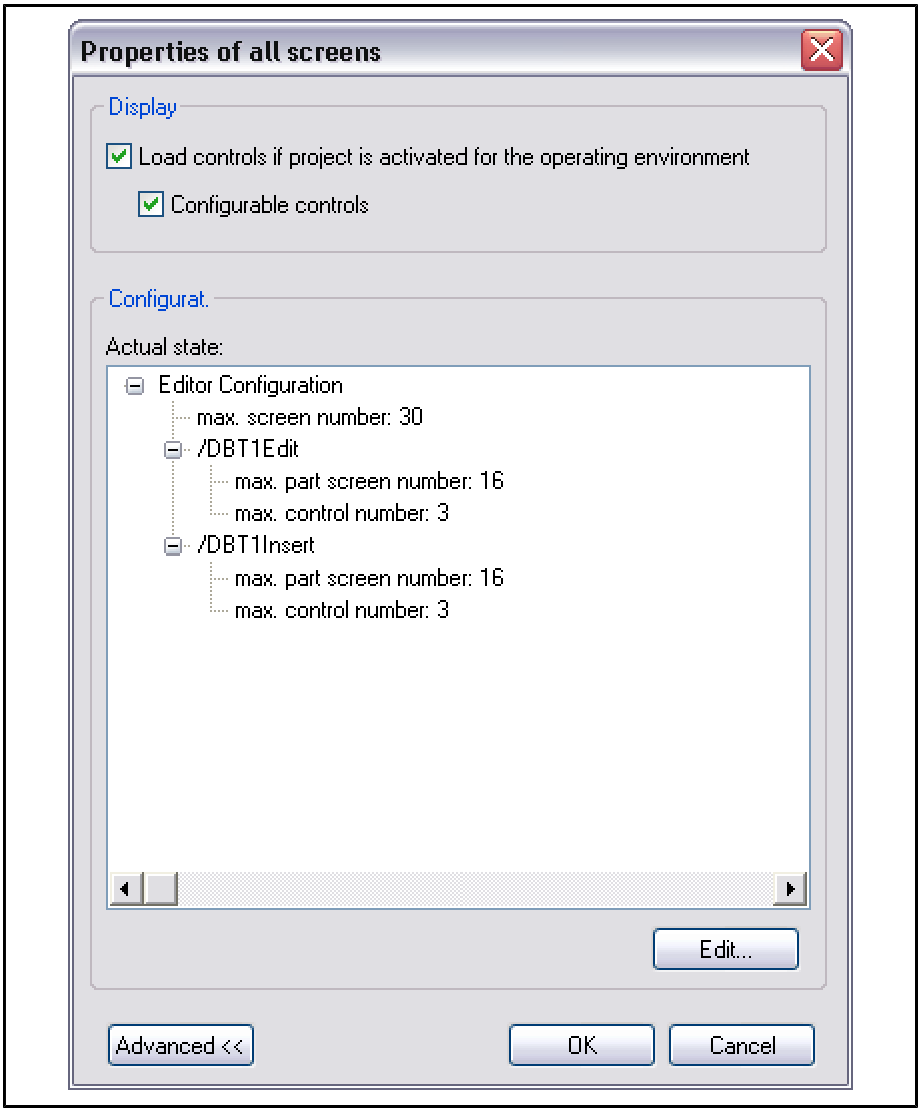


Fig.16-120: Dialog: "Properties of all screens" (advanced)

The advanced dialog also shows the current configuration of project node **"Editor configuration"** in terms of the maximum number of NC screens to be managed, as well as the screen segments that can be created in an NC screen and the controls that can be inserted into a screen segment.

☞ In the dialog "Configuration screen segment/control", the specified maximum number of controls is taken into account in the function "Insert control...".

If the maximum number of controls that are to be inserted in a screen segment is reduced, the controls that are already contained in the screen segment but which, as a result, exceed the maximum number are not removed. The same applies to the screens and the screen segments if their maximum number is reduced.

- **Modifying...**:

  In terms of their maximum number of screens, screen segments and controls, the configuration of the NC screens can be adapted by the user in a separate "Configuration - change" dialog called via the "Change..." button.

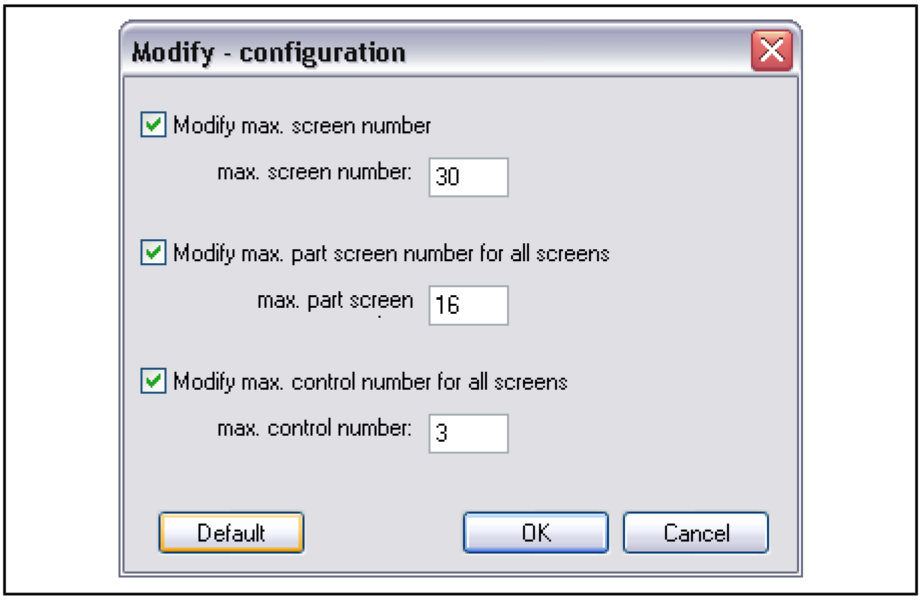


Fig.16-121: Dialog: Configuration - Modify

- **Default**:

  The values of the default configuration are automatically entered in the input fields.

Max. number of screens	30
Max. number of screen segments	16
Max. number of controls	3

☞ Values between 1 and 9999 can be entered for the maximum number of screens, screen segments and controls.

Continue to: Dialog "Properties of all Screens"

When the values in dialog "Configuration – Change" have been confirmed with "OK", the dialog "Properties of all Screens" shows the command status instead of the actual status and the modified values are highlighted.

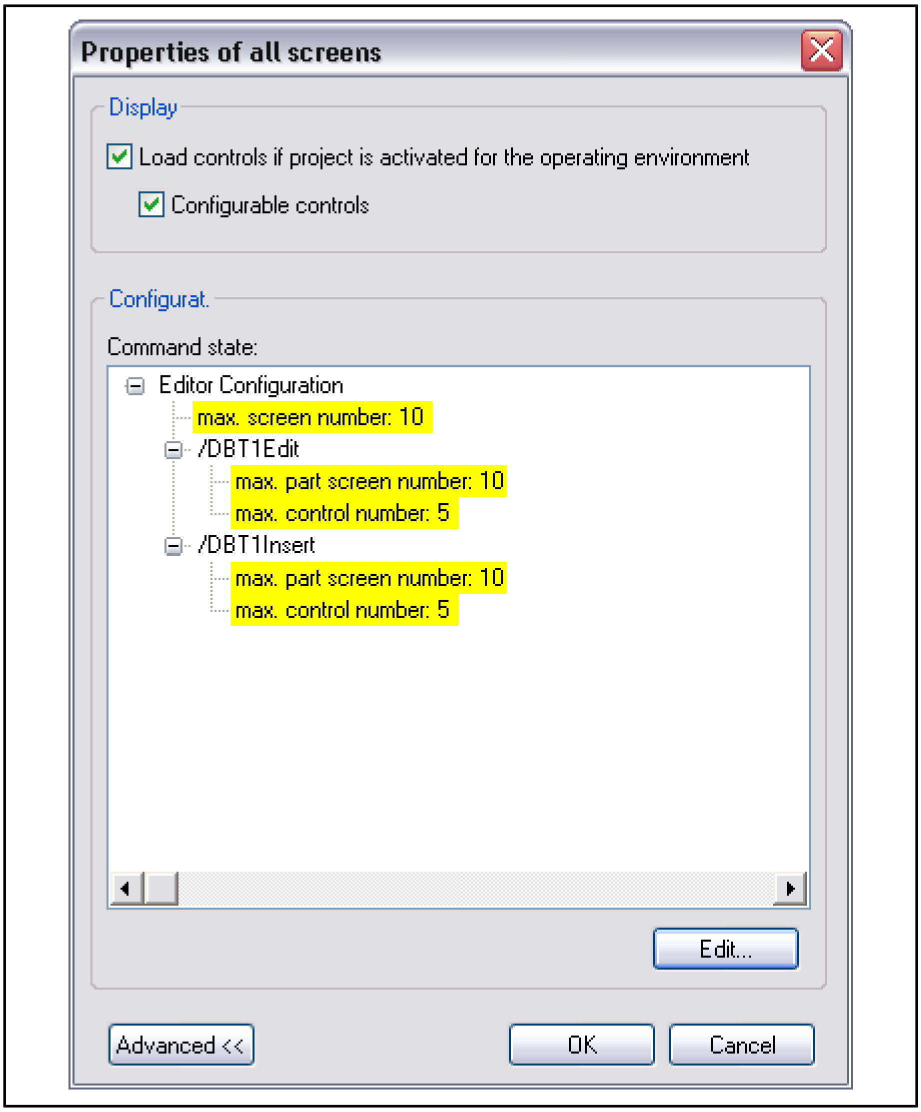


Fig. 16-122: Dialog: Properties of all screens (after modifying the configuration values)

The modified values for the NC screens are saved only after they have been confirmed with "OK" in the dialog "Properties of all screens".

- **Advanced <<**:

  Hides the advanced dialog.

Dialog "Screen Configuration"

The dialog **"Configuration screen"** is called via the context menu item of the same name in the project node of the NC screen to be modified. The division of the screen into screen segments, along with their number and position, can be changed in this dialog.

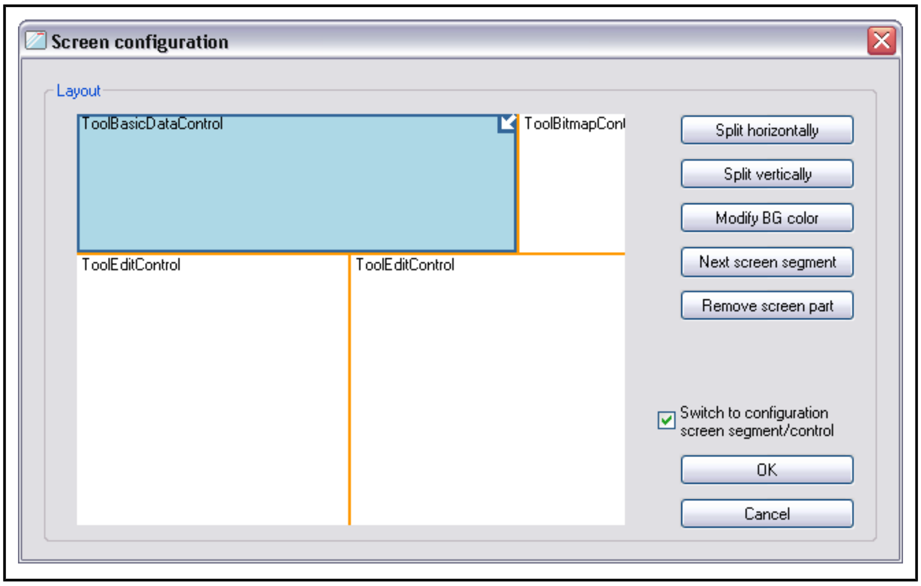


Fig. 16-123: Dialog: "Screen configuration" (with controls without process connection inserted into screen segments)

The following functions are available to design the editor layout as the division of the screen into screen segments:

- **Divide horizontally / vertically**:

  The focused screen segments is divided horizontally or vertically through the middle.

If this division results in the maximum number of screen segments for this screen being exceeded, a corresponding error message appears in the status line and division is cancelled.

The maximum number of screen segments into which the screen can be divided can be displayed and modified in dialog "Properties of all screens".

- **Change BG color**:

  To improve the visual separation of the screen segments, the background color (**B**ack**G**round) of the focused screen segment can be chosen from two shades of gray.

- **Next screen segment**:

  In the screen, the screen segment that follows the currently focused screen segment in the list of screen segments is focused.

The screen segments are arranged in a specific sequence in a list of screen segments. This sequence is specified by the position of the screen segments on the screen - according to the rule "from top to bottom, starting at the extreme left" – and cannot be changed.

Screens segments can also be focused directly regardless of the sequence in the list of screen segments by moving the cursor to the screen segment and clicking with the left mouse button. This method of focusing does not work if the screen segment contains a control that is not correctly programmed.

- **Delete screen segment**:

The focused screen segment is deleted and the dialog for filling the removed screen segment is adapted. In this adapted dialog, fill the vacated location of the deleted screen segment by increasing the size of neighboring screen segments. A corresponding handling instruction summary in the dialog explains the steps required.

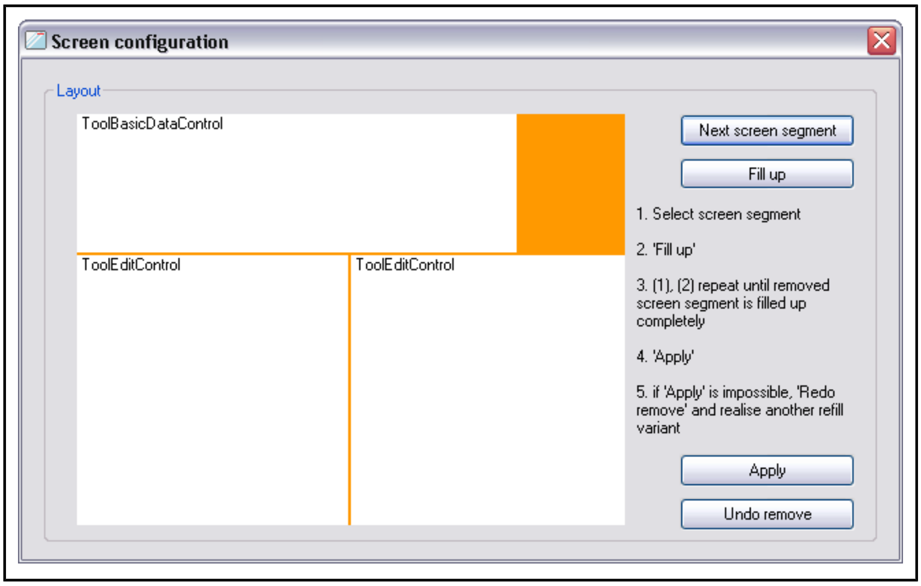


Fig.16-124: Dialog: "Screen configuration" (after calling the function: delete screen segment)

- **Next screen segment:**

  See above
- **Fill**:

  The focused screen segment is expanded into the vacant area of the deleted screen segment.
- **Accept**:

  The filling variant that is used is accepted. The display returns to the main dialog.
- **Undo delete**:

  The filling variant that is used is undone and the deleted screen segment is regenerated. The display returns to the main dialog.

☞ To ensure that the screen is divided into screen segments consistently, the free space of the deleted screen segment must be filled in completely before another division can be accepted. If the division cannot be accepted after the filling procedure has been carried out, undo the deletion, repeat it and use another filling variant.

The steps required for a filling procedure are listed in the dialog.

The division of a screen segment is undone by deleting the screen segment and expanding the adjacent screen segment(s) into the free space.

Continue to: Dialog "Configuration screen"

- **Dragging the screen segment dividing line**:

  The size of the screen segments can be changed by moving their dividing lines. To do this, move the cursor to the dividing line; the cursor icon changes. While holding the left mouse button down, you can now drag the dividing line using the mouse.

- **Change to screen segment/control configuration**:

  With "OK", the modified data is accepted and the dialog is closed. The checkbox can specify whether the "Configuration screen segment/control" dialog for the same screen segments is opened automatically.

Dialog "Configuration screen segments/control"

The **"Configuration screen segments/control"** dialog, which can be called using the corresponding context menu item in the project node of the associated NC screen, can be used to change the type, the number and the sequence of the controls in the screen segments and their associated F-panel, as well as certain focusing properties of the screen segments.

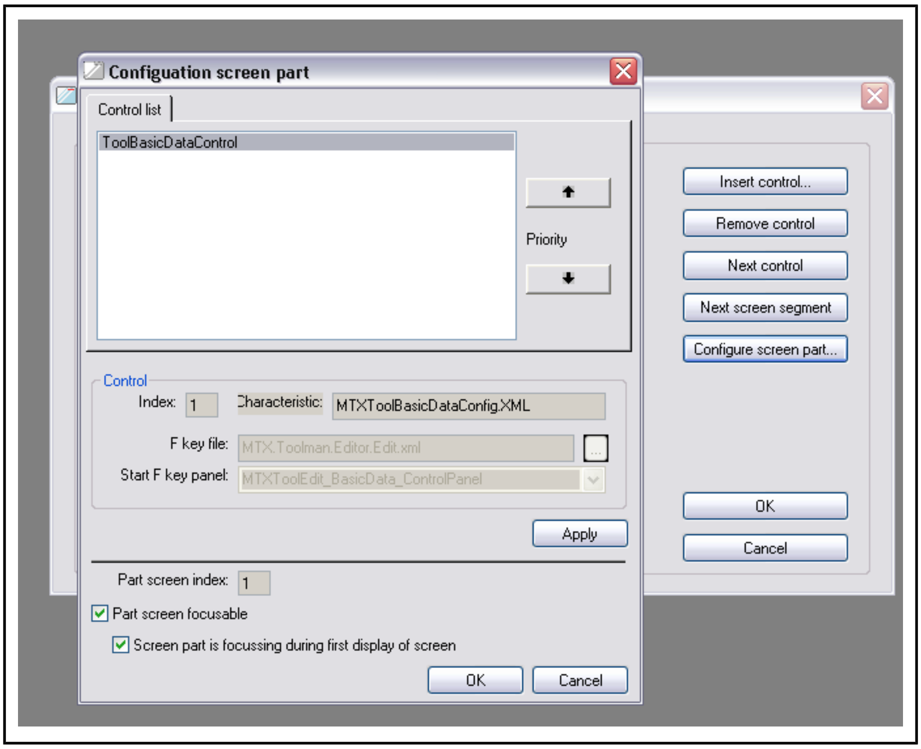


Fig. 16-125: Dialog: Configuration screen segment (with controls without process connection)

Controls can be inserted as content for the individual screen segments. A specific F-panel can be assigned to each control. The F-panel is shown in the user interface when the control is focused.

It is possible to insert several controls into a screen segment. If this is done, they are positioned congruently one in front of the other in the screen segment so that only the uppermost control is visible.

- **Insert control...**:

  The control to be inserted into the focused screen segments can be selected from the list of "Known controls" via the dialog "Insert control". In its original state, this "Known controls" list contains default entries for controls that are frequently used in NC screens. The "add..." button can be used to supplement entries for any additional controls that are needed.

Configuring the Tool Management

☞ If this insertion results in the maximum number of controls for this screen segment being exceeded, a corresponding error message appears in the status line and insertion is cancelled, i.e. the dialog is not called.

The maximum number of controls that can be inserted into the screen segment can be displayed and modified in the dialog "Properties of all screens".

The default entries in the "Known Controls" list may vary from software version to software version in order to fulfill the current requirements for the displays in NC screens. The control entries that have been subsequently added to the "Known Controls" list are retained if the software is updated.

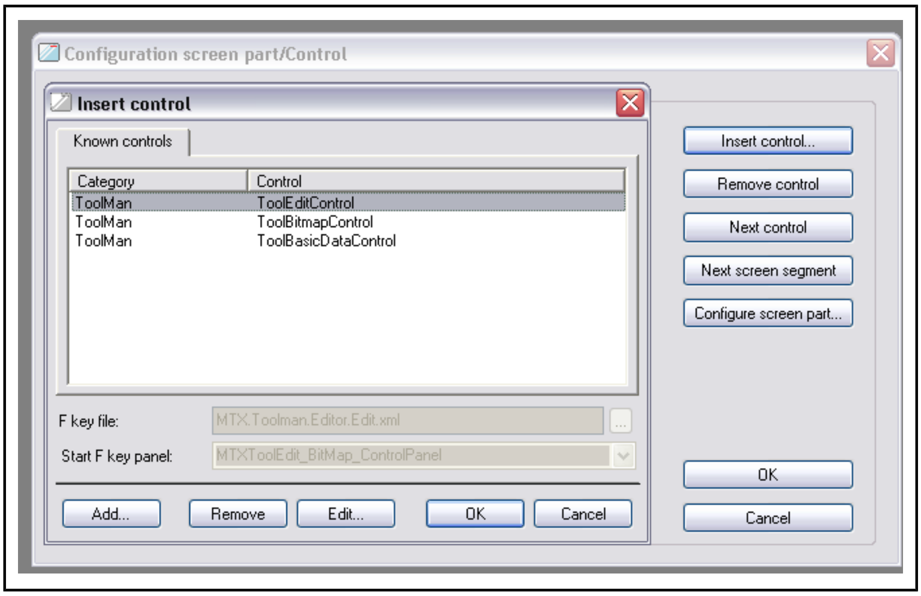


Fig. 16-126: Dialog: "Insert control"

- **Add...**:

  Using the "Add control" dialog (fig. 16-127 "Dialog: "Add control" (after calling the function Add...)" on page 423) that has been called, you can search the computer for "*.ocx" and "*.dll" files. If this file contains information about controls, these controls are incorporated into the "Known controls" list. The list entries consist of the category - which is used mainly as an arrangement criterion to assign the control to a certain group - and the control name, which is displayed in a screen segment if the control cannot be loaded into it. Both list entries can be freely assigned in the "Add control" dialog. However, entering a control name that differs from that read from the *.ocx/*.dll file is only useful if the file contains information for only one control. Otherwise all controls in this file appear with the same name in the list. But it is still possible to change the control names for each individual control at a later point in time.

When ActiveX controls are added, they are automatically entered in the registry (Windows database) using program "regsvr32.exe". For .NET controls, the program "regasm.exe" is searched for on the computer. This may take a few minutes. Prior to the search, a corresponding message appears pointing out the possibility for canceling the addition of the control. If a "*.tlb" file with the same name exists in the directory of the "*.dll" file in the case of .NET controls, the control information is extracted from this file and no registration is carried out.

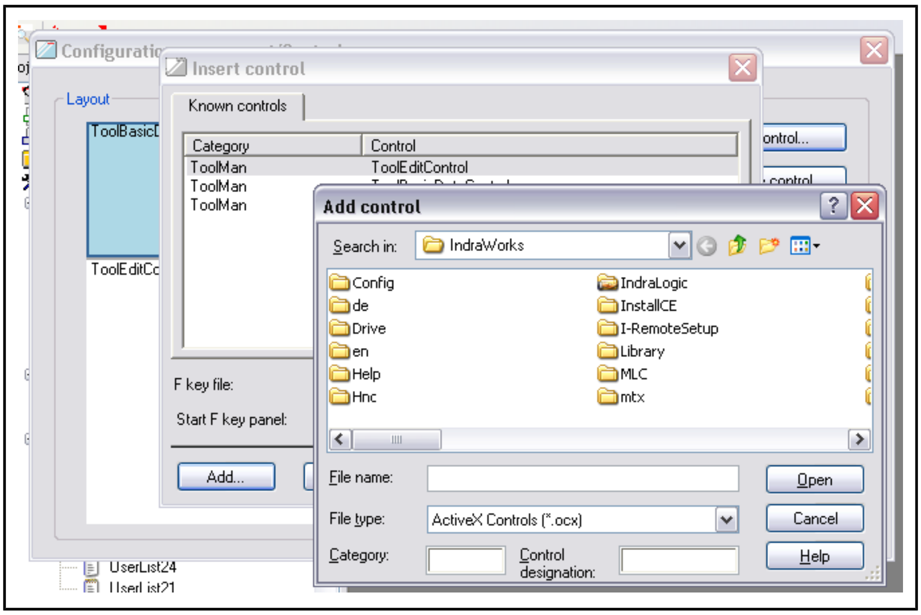


Fig. 16-127: Dialog: "Add control" (after calling the function "Add...")

Continue to: Dialog "Insert control"

- **Delete**:

  The selected entry is deleted from the list of "Known controls".

- **Modifying...**:

  The entries of the list of "Known controls" are specified in detail using the "change properties: Control entry" dialog - opened using the "change..." function - and can be changed there:

  – The control name, which has either been read from the "*.ocx"/"*.dll" file or has been defined in dialog "Add control..." when a control is added, can be renamed later or, by entering the corresponding token number, can be read from a text file. In this way, it can even be made language-dependent. The text file which is used for language dependency must be arranged according to a certain structure and must have the abbreviation of the regional schema set in the system as a supplement to the file name. For example, see the file "MtxNcScreen_Controls_de.txt" in the installation directory, which is used for the controls that are inserted by default in the regional schema "German" (DE). To provide language dependency, it is also required that all the controls added to the "Known controls" list use the same language file. This can be entered in the input field only after selecting the check box "change for all added control entries".

  – The same applies to renaming and the language-dependent specification of the category to which the control was assigned.

– A separate F-panel can be assigned to every control added to the "Known controls" list; this is displayed when the control is focused in the NC screen of Operation Desktop. To do this, select the appropriate F key file and enter the name of the F panel. The controls contained in the "Known controls" list by default have a fixed assignment to a specific F-panel. The check box "assignment can be changed" can be used to cancel this fixed assignment and to change the F panel.

☞ In case of a tool editor configuration, the following panels can be used, depending on whether an online or an offline editor is to be configured:

For an online editor:

- F key file: **MTX.Toolman.Editor.Edit.xml**
- possible start F key panel:
  - **MTXToolEdit_BasicData_ControlPanel**
  - MTXToolEdit_BasicData_ControlPanel2 (to be used only internally)
  - **MTXToolEdit_ToolStates_ControlPanel**
  - **MTXToolEdit_UserData_ControlPanel**
  - MTXToolEdit_UserData_ControlPanel2 (to be used only internally)
  - **MTXToolEdit_BitMap_ControlPanel**

For an offline editor:

- F key file: **MTX.Toolman.Editor.Insert.xml**
- possible start F key panel:
  - **MTXToolInsert_BasicData_ControlPanel**
  - **MTXToolInsert_ToolStates_ControlPanel**
  - **MTXToolInsert_UserData_ControlPanel**
  - **MTXToolInsert_BitMap_ControlPanel**

Online Editor

F-key level	<F2>	<F3>	<F4>	<F5>	<F6>	<F7>	<F8>	<F9>
1. Level				Save basic data	Modify type on/off	Next window		Close editor

Fig.16-128: MTXToolEdit_BasicData_ControlPanel

F-key level	<F2>	<F3>	<F4>	<F5>	<F6>	<F7>	<F8>	<F9>
1. Level		Next display				Next window		Close editor

Fig.16-129: MTXToolEdit_ToolStates_ControlPanel

F-key level	<F2>	<F3>	<F4>	<F5>	<F6>	<F7>	<F8>	<F9>
1. Level		Next display			Entry additive	Next window		Close editor

Fig.16-130: MTXToolEdit_UserData_ControlPanel

F-key level	<F2>	<F3>	<F4>	<F5>	<F6>	<F7>	<F8>	<F9>
1. Level						Next window		Close editor

Fig. 16-131: MTXToolEdit_BitMap_ControlPanel

Offline editor

F-key level	<F2>	<F3>	<F4>	<F5>	<F6>	<F7>	<F8>	<F9>
1. Level				Save basic data		Next window		Close editor

Fig. 16-132: MTXToolInsert_BasicData_ControlPanel

F-key level	<F2>	<F3>	<F4>	<F5>	<F6>	<F7>	<F8>	<F9>
1. Level		Next display				Next window		Close editor

Fig. 16-133: MTXToolInsert_ToolStates_ControlPanel

F-key level	<F2>	<F3>	<F4>	<F5>	<F6>	<F7>	<F8>	<F9>
1. Level		Next display				Next window		Close editor

Fig. 16-134: MTXToolInsert_UserData_ControlPanel

F-key level	<F2>	<F3>	<F4>	<F5>	<F6>	<F7>	<F8>	<F9>
1. Level		Next display				Next window		Close editor

Fig. 16-135: MTXToolInsert_BitMap_ControlPanel

- If a control contains the implementation of the MTXACIInterface.dll interface, the configuration parameter that can be entered in field "instance" can be transferred to it. If the control can display different process data or if it has various types of displays, this configuration parameter can be used during the initial instancing of the control to determine which of these instances the control should use.

☞ The following configurations can be used in case of a tool editor configuration:

- possible type:
  - **ToolTypeName.typ**
  - **ToolTypeNameO.typ**
  - **MTXToolBasicDataConfig.XML**
  - **MTXToolEditConfigLimitData.XML**
  - **MTXToolEditConfigUsrData.XML**
  - **MTXToolEditCtrlPlaceStates.XML**
  - **MTXToolEditCtrlToolBD1.XML**
  - **MTXToolEditCtrlToolStates.XML**

  ...or further special configurations.

- Controls that have been added later to the "Known controls" list can be easily deleted from the list via "Delete" in the "Add control" dialog. On the contrary, the default entries are protected against accidental deletion:

However, they can be enabled for deletion by activating the checkbox "control entry can be deleted from the 'Known controls' list".

The language file can be changed only for all control entries that were added to the default entries in the "Known Controls" list.

If an F-panel is not assigned to a control, a default F-panel is displayed when the control is focused in the NC screen of Operation Desktop. This applies to all NC screens; it cannot currently be reconfigured.

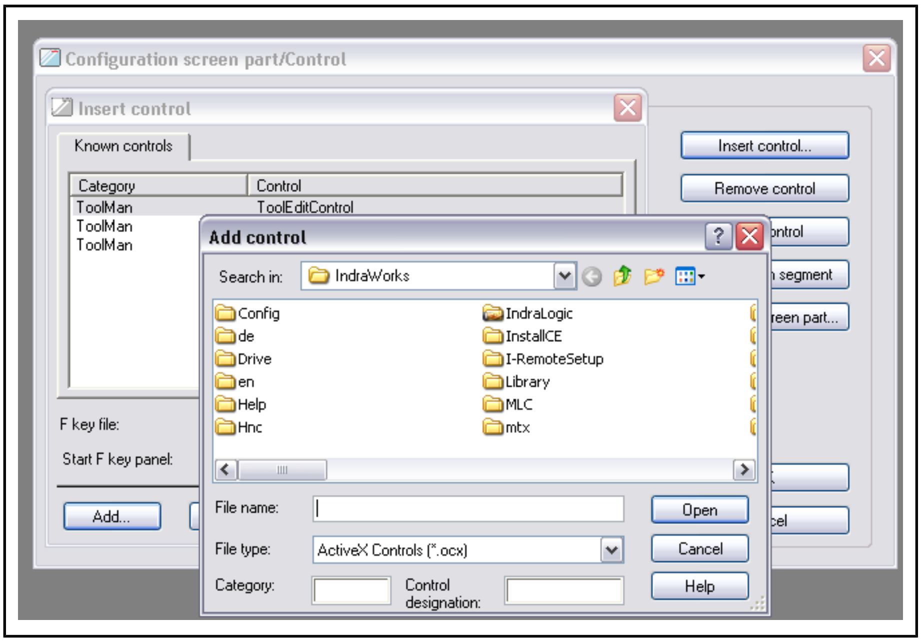


Fig. 16-136: Dialog: "Change properties: Control Entry (after calling function "Change..."

Continue to: Dialog "Configuration screen segment/control"

- **Delete control**:

  The visible control of the focused screen segment is deleted after a security query is confirmed.
- **Next control**:

  The control that follows the currently visible control in the list of controls of the screen segment is shown in the focused screen segment.
- **Next screen segment**:

  See the dialog "Configuration screen"
- **Configure screen segment**:

  The dialog "Configuration screen segment" is called for the focused screen segment. This contains the following information and setting options:

  1. The controls of the screen segment are listed in the display order in this dialog. The "up" and "down" buttons can be used to change the priority of a control, i.e. its position (index) in the display order.
  2. In addition, the instance – transferred as a configuration parameter – and the assigned F key file and F panel are displayed for a control that is selected in the list of controls. The F-key file and bar can be changed if the associated option "Allocation can be changed" is selected for the control in the dialog "Change properties: Control en-

try" (set by default for controls added to list "Known controls" in the dialog "Add control" - see fig. 16-126 "Dialog: "Insert control"" on page 422).

3. Furthermore, additional options can be set for focusing the screen segment in the NC screen of Operation Desktop . The checkboxes "Screen segment is focusable" and "Screen segment focused the first time that the screen is displayed" can be used to determine whether the screen segment can be focused in the first place and whether this screen segment should be immediately focused automatically when the screen is displayed for the first time after starting Operation Desktop. In Operation Desktop, pressing F key "next window" switches to the next screen segment in a specified sequence. This is focused. The screen segment index listed in dialog "Configuration screen segment" indicates the position of the screen segment in this focusing sequence.

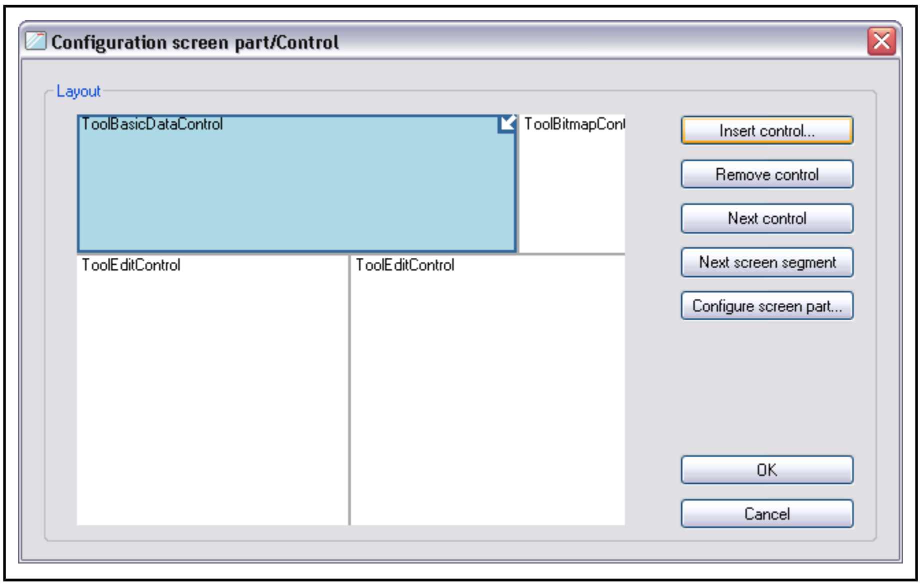


Fig. 16-137: Dialog: Configuration screen segment/Control

Function "Duplicate"

The function **Duplicate** is called using the corresponding context menu in the project node of the associated NC screen; its effect is that of **Copy & Paste**.

The selected screen is copied. In the **"Duplicate (copy & paste)"** input box which appears then, enter the screen ID that is to be used for the new screen to be created by pasting.

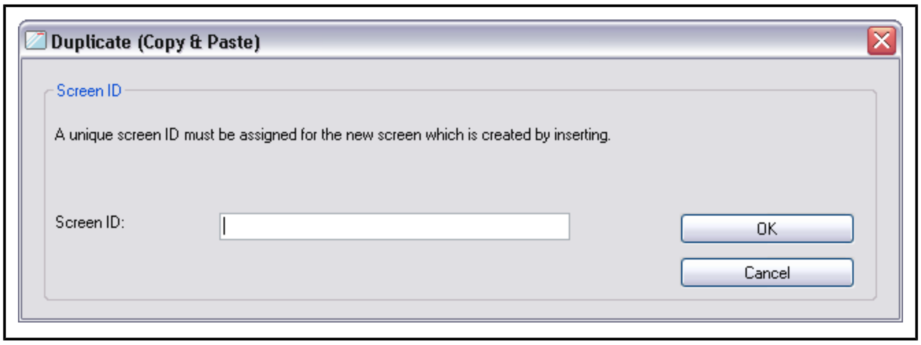


Fig. 16-138: Dialog: Duplicate (Copy & Paste)

The layout and content of the screen segments of the screen that is generated can be adapted in the dialogs "Configuration screen" (fig. 16-146 "Dialog:

"Screen configuration" (for new tool editor)" on page 432) and "Configuration screen segment/control" (fig. 16-151 "Dialog: "Configuration screen segment/control" (for the new tool editor)" on page 435).

It is not possible to paste a new editor with a name (screen ID) that is already in use. If you attempt to do this, a corresponding error message appears.

Function "Delete"

With the respective context menu item in the project node of the corresponding NC screen, **Delete** is called and the respective screen is deleted after confirming the security prompt.

Function "Rename"

The name (screen ID) of an NC screen can be changed directly in the associated project node.

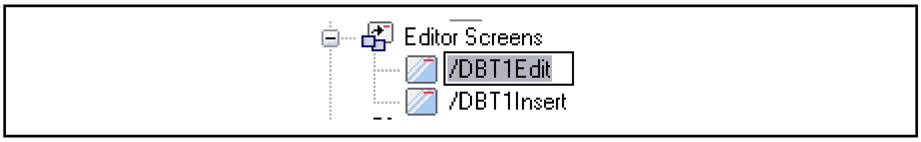


Fig.16-139: Renaming a tool editor in the project node

Rename is called via the respective context menu item in the project node of the NC screen. You can also click on the project node again after selecting it with the mouse.

It is not possible to rename an editor using a name (screen ID) that is already in use. If you attempt to do this, a corresponding error message appears.

Configuration of the Tool Editor Controls

The configuration of the following tool editor controls:

- MTXToolBasicDataConfig.XML
- MTXToolEditCtrlToolStates.XML
- MTXToolEditCtrlPlaceStates.XML
- MTXToolEditCtrlToolBD1.XML
- MTXToolEditConfigUsrData.XML
- MTXToolEditConfigLimitData.XML

can be carried out using the ULC configurator in a manner similar to tool list configuration. As opposed to the list configuration, however, only the data of one tool are displayed.

There is no way to directly call the ULC configurator from the editor configuration. Via **Form Configuration ▸ Container Controls** can be selected.

Configuration of the M Key Bars in the Tool Editor

The tool editors can receive their own M key bars specific to their type with the following fixedly determined panel names:

Editor type	M key bar	M panel name
Insert type	Right	ToolInsertEditorLocalRight
	Left	ToolInsertEditorLocalLeft
Edit type	Right	ToolEditorLocalRight
	Left	ToolEditorLocalLeft

Fig. 16-140: Names of the M key panels of the tool editors

Said M key bars can be configured by the user in the M key configurator.

For this, all keys should be defined by means of the **"TL_Editor"** function using the following subfunctions in **"CommandString"**:

Name of the function	CommandString	Description
TL_Editor	Edit_Prev_Tool	Search for the previous tool
	Edit_Next_Tool	Search for the next tool
	Save_BasicData_ToolType	Save the current data record as a basic data record
	Delete_BasicData_ToolType	Delete the basic data record

Fig. 16-141: Commands for the M key functions in the tool editor

☞ The use of other functions for the M key configuration is possible; however, it is not recommended, since these usually do not result in a reasonable behavior of the tool editor (e.g. Delete tool).

Handling Instruction: Creating and Configuring Tool Editors

Using an example, the creation and configuration of a new tool editor will be carried out stepwise.

IW-Engineering / configuration tool editor: Create new tool editor

1. In the open portion of the **"Tool Management Screens"** project tree, click node **"Editor Configuration"** using the right mouse button.

   The context menu of the **"Editor Configuration"** opens.

2. Click in the context menu on **New Edit or Insert Screen** .

Configuring the Tool Management

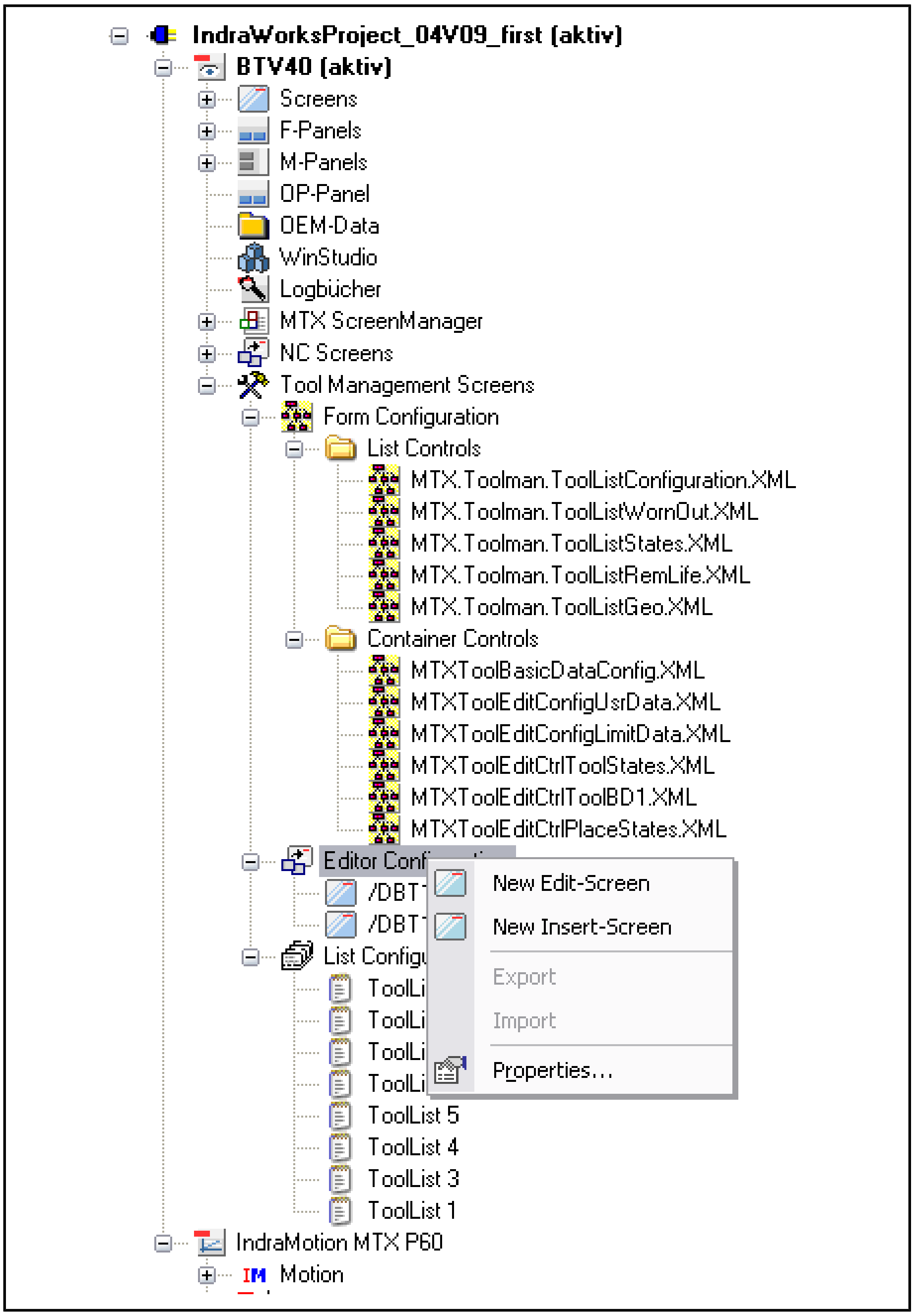


Fig.16-142: Context menu: Editor configuration

The **"New screen"** dialog is displayed.

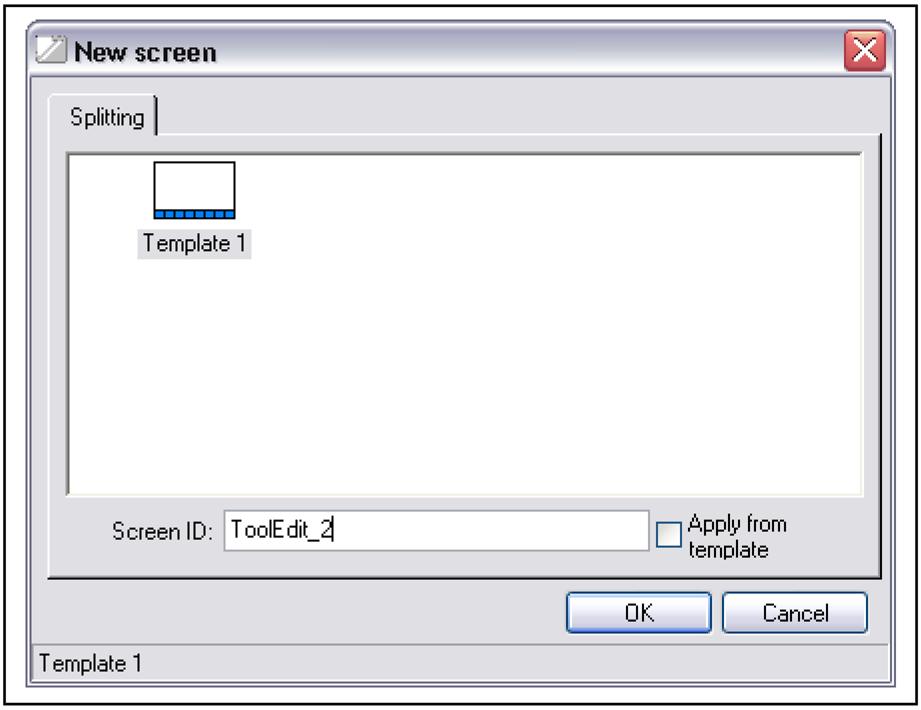


Fig.16-143: Dialog: "New screen" (with selected template)

3. In the "New Screen" dialog, click on the template and select the checkbox **"Transfer from Template"**.

   The screen division and screen ID of the template are used for the new screen.

4. Confirm the entries made in the dialog **"New Screen"** with "OK".

   The name of the new screen is displayed under the project node **"Editor Configuration"**.

Fig.16-144: Project node: "Editor configuration" (with new tool editor)

IW Engineering / Tool Editor Configuration: Changing the Screen Division of the New Tool Editor

1. Click the node of the new NC screen using the right mouse button.

   The context menu of the NC screen opens.

Configuring the Tool Management

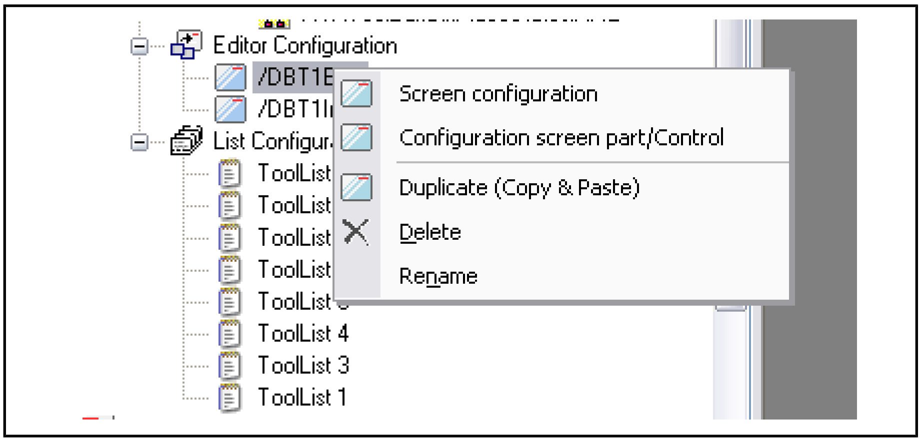


Fig. 16-145: Context menu: New tool editor

2. Click in the context menu on **Configuration Screen** .

   The **"Configuration Screen"** dialog is displayed.

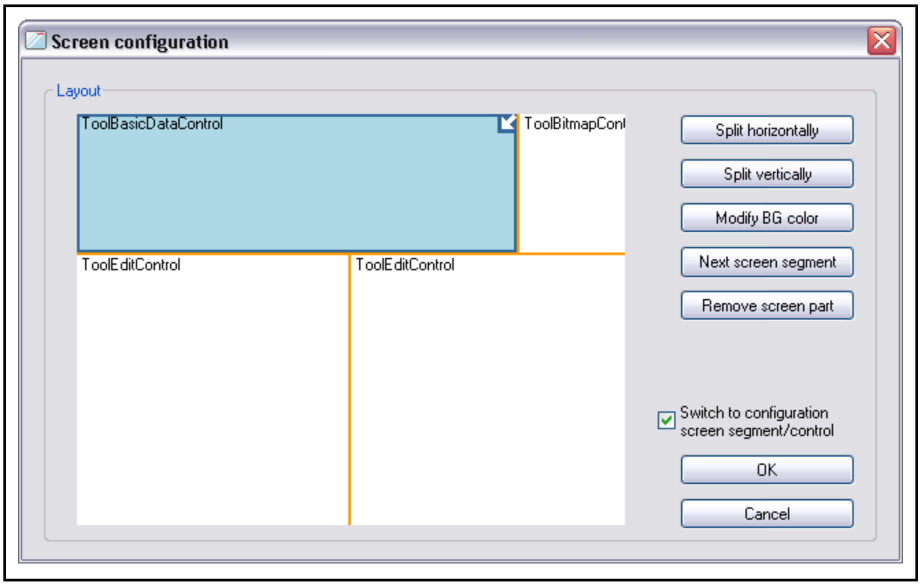


Fig. 16-146: Dialog: "Screen configuration" (for new tool editor)

3. Press **"Divide horizontally"**.
4. Press **"Change BG color"** twice.
5. Press **"Next screen segment"**.

   The focused screen segment is divided horizontally and the background color of the newly focused screen segment changes first to white, then to gray. Then, the display switches to the next screen segment, which is divided vertically.

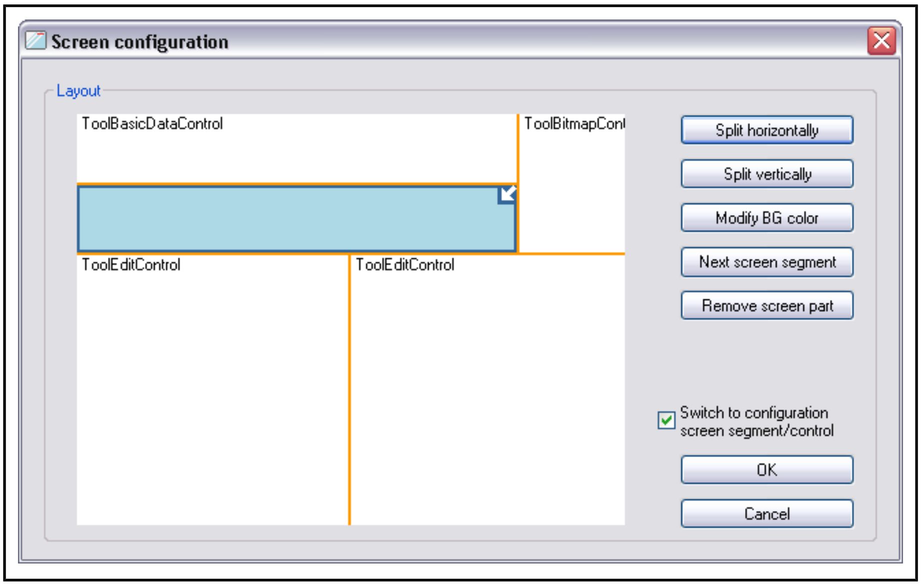


Fig. 16-147: Dialog: Screen configuration (division of the new tool editor after dividing)

6. Press **"Delete screen segment"**.

   The focused screen segment is deleted and the dialog to fill the removed screen segment is adapted.

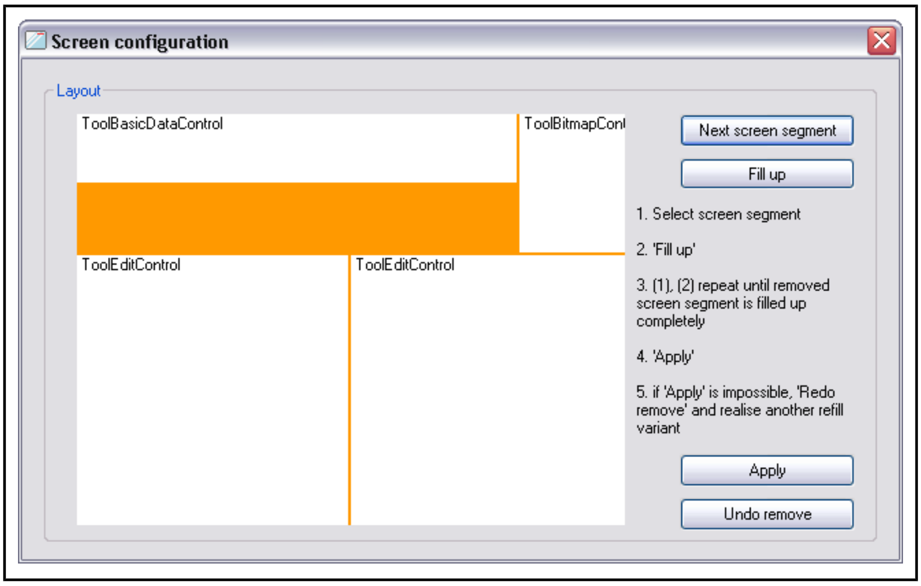


Fig. 16-148: Dialog: "Screen configuration" (for filling the deleted screen segment)

7. Press **"Next screen segment"** thrice.
8. Press **"Fill"**.
9. Press **"Accept"**.

   The focus moves to the screen segment to the left of the one that was deleted. This screen segment is expanded to the position of the deleted screen segment and filling is accepted.

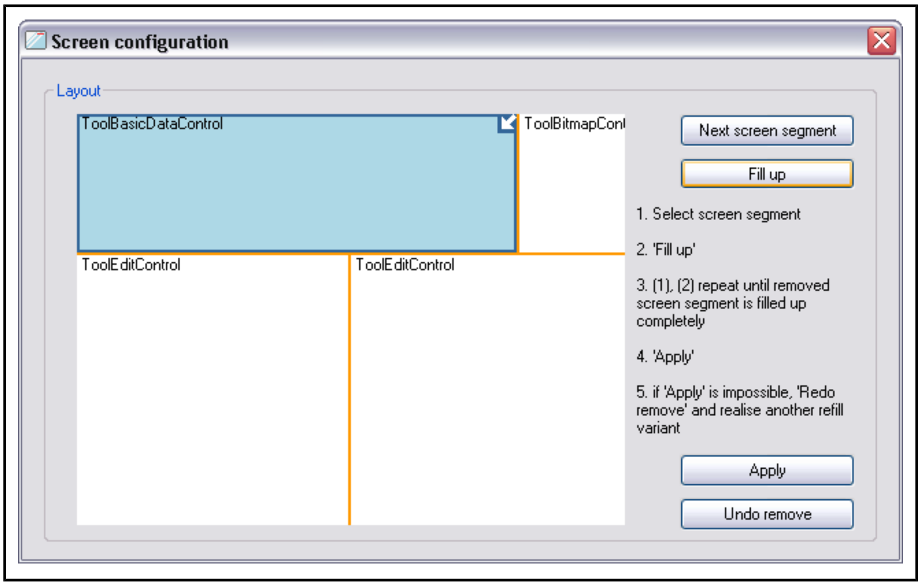


Fig. 16-149: Dialog: "Screen Configuration" (after deleting and filling the screen segment)

☞ The deletion and filling of the screen segment undoes the vertical division.

10. Move the cursor to the uppermost horizontal screen segment divider.

    The cursor symbol becomes a double arrow.

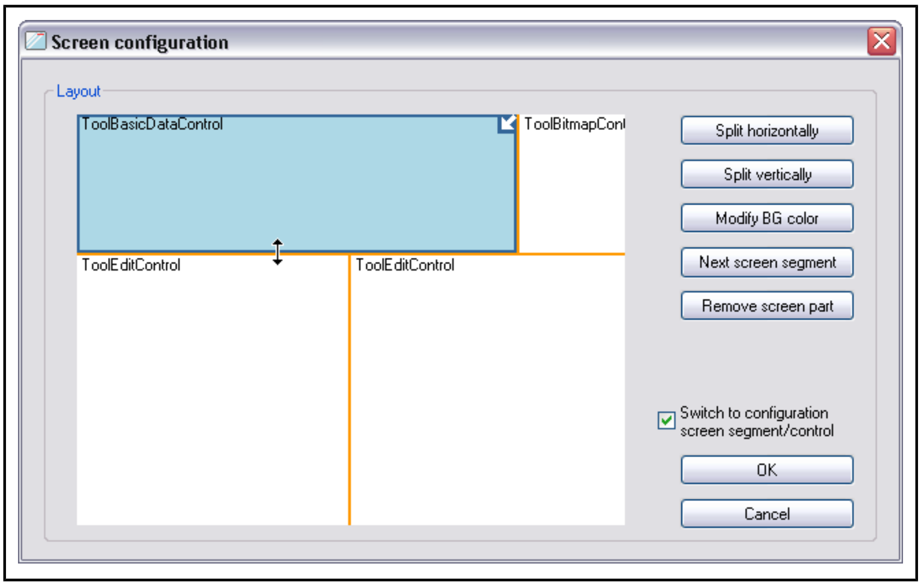


Fig. 16-150: Dialog: "Screen configuration" (cursor symbol indicates the possibility for moving the screen segment divider)

11. Press and hold the left mouse button and move the divider somewhat using the mouse.

    The size of the adjacent screen segment changes accordingly.

12. Press **"OK"**.

The modified division of the new NC screen is accepted and the dialog is closed.

☞ If the checkbox "Switch to Screen Segment/Control Configuration" is activated, the dialog "Screen Segment/Control Configuration" is opened automatically after the dialog has been closed.

IW-Engineering / configuration tool editor: Insert Controls in the New Tool Editor

1. In the dialog **"Configuration screen"**, activate the checkbox **"Switch to Configuration screen segment/control"** and exit the dialog with **"OK"**.

   – or –

   Click the node of the new NC screen using the right mouse button.

   The context menu of the NC screen opens.

2. Click in the context menu on **Configuration screen segment/Control** .

   The **"Configuration screen segment/control"** dialog is displayed.

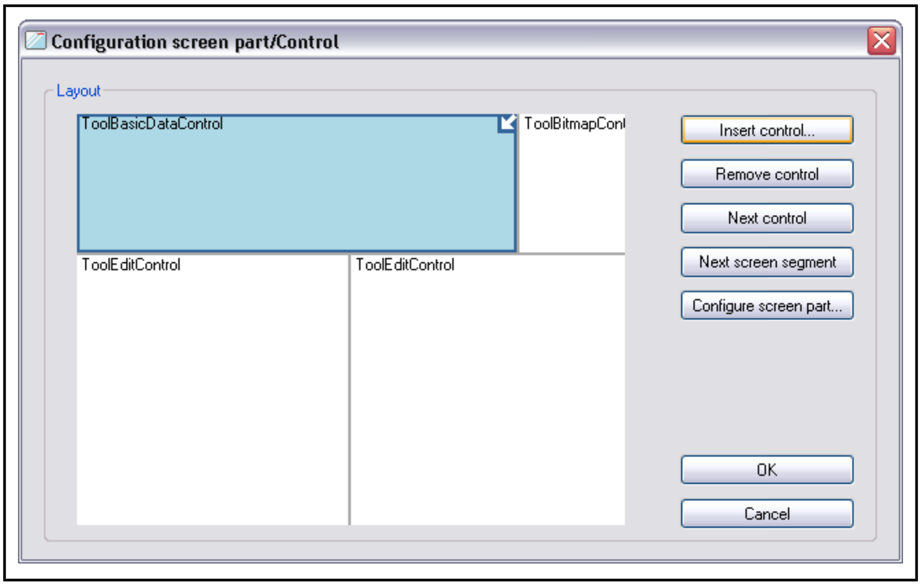


Fig.16-151: Dialog: "Configuration screen segment/control" (for the new tool editor)

3. Press **"Insert Control..."**.

   The **"Insert Control"** dialog is displayed.

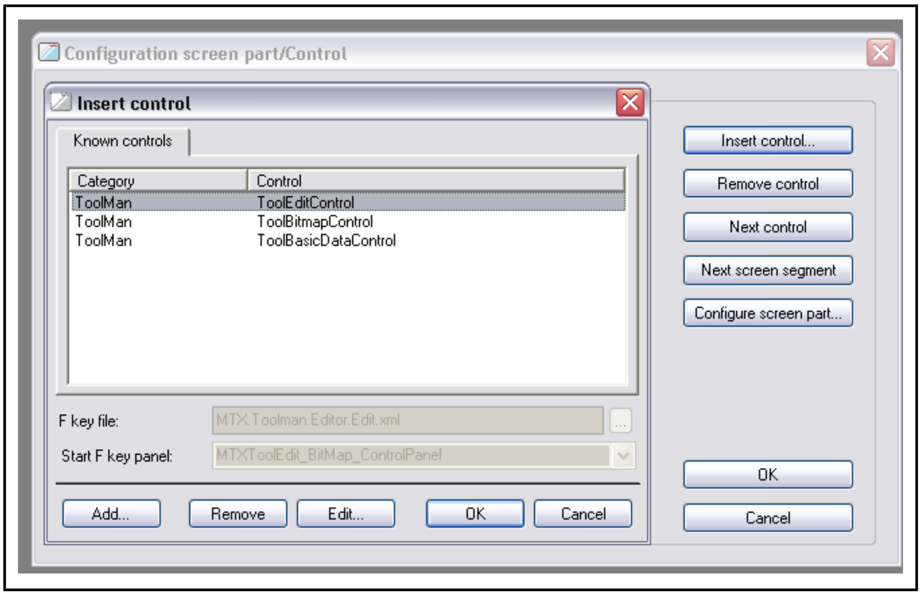


Fig.16-152: Dialog: "Insert control"

4. Select a control and click **"OK"** to confirm.

   The selected control is inserted into the focused screen segment.

5. Insert another control into the screen segment or proceed in the same manner in the other screen segments.
6. Press **"OK"**.

   The modified configuration of the new NC screen with controls is accepted and the dialog is closed.

IW-Engineering / configuration tool editor: Rename the New Tool Editor

1. Click the node of the new NC screen using the right mouse button.

   The context menu of the NC screen opens.

2. Click in the context menu on **Rename** .

   The name (screen ID) of the new NC screen can be changed directly in the associated project node.

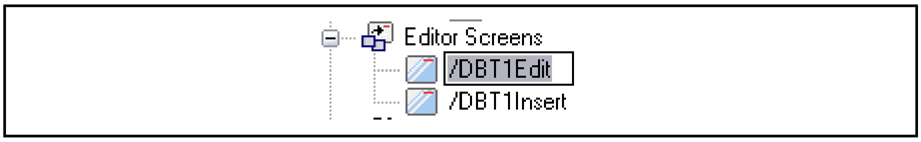


Fig.16-153: Renaming the new tool editor in the project node

IW Engineering: Data Transfer

1. Save the project
2. Insert the icon

Handling Instruction: Configuration of the Tool Editor Controls

The configuration of the following tool editor controls:

- MTXToolBasicDataConfig.XML
- MTXToolEditCtrlToolStates.XML

- MTXToolEditCtrlPlaceStates.XML
- MTXToolEditCtrlToolBD1.XML
- MTXToolEditConfigUsrData.XML

takes place in the same way as tool list configuration, the only difference being that the data of only one tool are indicated in these lists. This way, it is ensured that the entry **<ShowMultipleSublists>** = FALSE is set and that the entry **<PathOfMultiplicator>** remains empty.

IW Engineering: Data Transfer

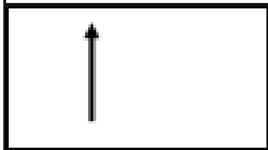
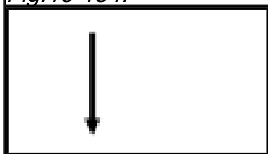
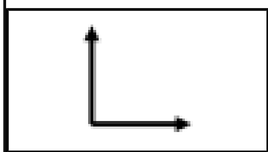
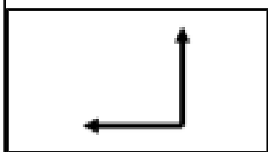
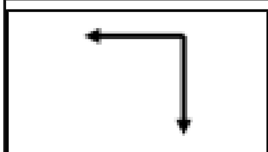
1. Save the project
2. Insert the icon

Defining the Display of the Coordinate System in the Tool Bitmap Control

A coordinate system establishing an axis reference of the correction values is stored in the displayed tool bitmap.

This axis reference can be set via the system date **SD.SysCoordSystem.Value**.

Bitmap control

Coordinate system	Ordinate	Abscissa	SD.SysCoordSystem.Value
Without	-	-	0
 *Fig.16-154:*	X	-	1
	Y	-	3
	Z	-	5
 *Fig.16-155:*	X	-	2
	Y	-	4
	Z	-	6
 *Fig.16-156:*	X	Z	10
	X	Y	20
	Y	X	30
	Y	Z	40
	Z	X	50
	Z	Y	60
 *Fig.16-157:*	X	Z	11
	X	Y	21
	Y	X	31
	Y	Z	41
	Z	X	51
	Z	Y	61
 *Fig.16-158:*	X	Z	12
	X	Y	22
	Y	X	32
	Y	Z	42
	Z	X	52
	Z	Y	62

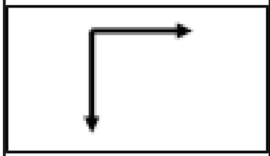
Coordinate system	Ordinate	Abscissa	SD.SysCoordSystem.Value
	X	Z	13
	X	Y	23
	Y	X	33
	Y	Z	43
	Z	X	53
Fig.16-159:	Z	Y	63

Fig.16-160: Coordinate system

The displayed coordinate system can also be determined automatically. But presettings have to be made once.

Specifications whether the value SD.SysCoordSystem.Value should be determined automatically:

SD.SysCoordSystem.Calc: 1 - Automatic value determination
0 - No automatic value determination

Specifications for the machine configuration:

Machine coordinate system:

SD.SysCoordSystem.horizontal: 1 - Abscissa in positive direction
0 - Abscissa in negative direction

SD.SysCoordSystem.vertikal: 1 - Ordinate in positive direction
0 - Ordinate in negative direction

Master axis meaning:

SD.SysCoordSystem.X_axis_pref: 1 - X-axis exists and is preferably displayed
0 - X-axis does not exist

SD.SysCoordSystem.X_axis_pref: 1 - Y-axis exists and is preferably displayed
0 - Y-axis does not exist

SD.SysCoordSystem.Z_axis_pref: 1 - Z-axis exists and is preferably displayed
0 - Z-axis does not exist

Fig.16-161: SD.SysCoordSystem

That means if the value of the variables SD.SysCoordSystem.Value should be determined automatically, 1 has to be assigned to SD.SysCoordSystem.Calc. it is only then required to set the machine configuration.

16.6.4 Configuration of User Management

General

MTX user management is classified into

- data-related user management and
- functional user management.

Data-relevant user management allows for assignment of data element-relevant read and write privileges to the individual user groups. As described above, this is realized via attribute definitions (L1, L2, L3, L3, L5) in the data record schema.

> ☞ Presently, the functional user management cannot be configured by the user.

Handling Instruction: Defining Data-Relevant User Privileges

The following handling instructions describe the process to be followed when the user privileges for the individual data elements are to be modified.

IW Operation / program: Edit the Schema File "dbt?ud.xsd"

1. Copy the file "dbt?ud.xsd", "dbt**?**sd.xsd" (**?**:= 1 [DBT1] or 2 [DBT2]) or "tool_ty.xsd" in control directory "\usrfep\schema" or "\feprom\schema" or "\root\schema" to the mount directory (\mnt).
2. Edit user rights L1 to L5 of the respective nodes or elements in the schema file by means of the XSD editor (preferably with Altova SPY).
3. Save file.
4. Copy the file back to the control directory "\usrfep" or "\root".

Figure		Documentation
Figure:		Image of data record schema
Documentation	MTX Functional Description	Edit schema file

NC: Transferring the new data structure

1. Complete IW Operation
2. Reset the control

Figure		Documentation
Documentation	MTX Functional Description	Transferring the new data structure

16.7 Interfaces

16.7.1 CPL Interfaces

The MTX provides the following NC commands for accessing the tool database:

For the detailed syntax of the NC commands described below, please see the documentation "Rexroth IndraMotion MTX Programming Manual". Here, only the names of the NC commands are mentioned, each illustrated by one example.

TCV

Supplies the last-programmed tool compensation values, either as a sum (D-correction + external correction memory) or as a single value.

Example: Reading the L2 tool length of the last-programmed ED correction.

Program:

```
110 TCV(2,"E")
```

DCT

Read and write access to tool compensation values in any D-correction tables as well as to external correction values (ED correction). During writing, incremental modifications can also be preset.

Example:

Inch access to the "L2" correction of data block 2 of the external tool compensation.

Program:

```
110 DCT("L2" ,2 ,0 ,"INCH")
```

DBSEA

Search for data records within a database table.

The CPL function returns the header of the first data block that satisfies the search condition. In this case, the variable returns a value of 1. The data block search starts with the data block defined using `<StartKey1>` and `<StartKey2>`. If one of the two start keys has a value of -1, the search starts at the first data block of the database table.

Example:

Search for the tool with T-number 5 in the tool memory and save the data record that is found in structured variable "SV.Tool". Using DBSEA, only the "Header" data of a data record are read.

Program:

```
110 SV.Tool.Hd = DBSEA(”/DBT1”,-1,-1 ,”IKQ3=5”,RECFOUND%,ERR%)
```

DBSEAX Searches for one or more data blocks in a tool database table.

The command supplies a list of data block headers that satisfy the search criterion. The headers are stored in a system data array `<HeaderArr>` that is sorted according to K1 and K2. Parameter `<ErgSize>` is used to specify the maximum number of data block headers to be sought. The return value of DBSEAX supplies the number of data blocks found.

DBTAB Reading or writing of data elements and data records

Example:

Reading the data record of sector 3, location 34 into the structured variable SV.Tool.

Program:

```
110 SV.Tool = DBTAB(”/DBT1”,3,34 ,ERR%)
```

DBTABX This can read a complete data block or a substructure of a tool database table into a CPL variable or write from the variable back to the data block. Parameter <Mode> can be used to control whether all the data of the data block or only tool- or location-specific data are written.

Example:

1. Reading all data of the data block of sector 3, location 34 into the structured variable SV.Tool.
2. Writing all tool data of the data block of sector 3, location 34 of the structured variable "SV.Tool". I.e. no location-specific data are adopted.

Program:

```
110 SV.Tool = DBTABX("DBT1.Rec",3,34)
120 DBTABX("DBT1.Rec",3,34,1,ERRNO) = SV.Tool
```

DBTABXL Similar to DBTABX, but the data block is locked during read access; this lock is removed when the data block is written.

DBMOVE Moving data records within a database table.

Example:

The contents of the data record (1,1) are moved into the data record (2,2).

Program:

```
130 DBMOVE(”/DBT1”,1,1,2,2)
```

DBLOAD Via "DBLOAD", parts of a database table or a complete database table can be read from a file into the database.

Example:

The content of the data record (1,1) is read from the file "dbdaten.txt" into the database table.

Program:

```
140 DBLOAD("/DBT1",1,1,"/dbdaten.txt")
```

DBSAVE Via "DBSAVE", parts of a database table or a complete database table can be saved to a file.

Example: The content of the data record (1,1) is moved to the "dbdaten.txt" file.

Program:

```
150 DBSAVE("/DBT1",1,1,"/dbdaten.txt")
```

16.7.2 PLC Interfaces

General

To access the tool database, the MTX offers the PLC function components and structures described in the following; these are summarized in PLC library "MT_MTX.lib".

☞ For a detailed syntax of the NC commands described below, please see the documentation "Rexroth IndraMotion MTX PLC Interface". Only the names of the PLC blocks are mentioned here.

Function Modules

MT_DbData The "MT_DbData" program component offers various functions for reading and changing tool data records.

MT_DbRecList The program component "MT_DbRecList" offers the possibility to search for data records within a database table or to delete data records according to a list.

MT_DbLoad By means of this program component, a database table can be read in from an XML file in whole or in part.

MT_DbSave By means of program component "MT_DbSave", a database table can be read into a file. Here, there are the options

- to delete the file first and then to save the data records

– or –

- to add the data records at the end of the file.

Structures

By means of the structures predefined in the PLC library "MT_MTX.lib", all data records and data elements of the tool database can be processed in the PLC user program. The names of the PLC structures follow the names of the data types defined in the scheme files.

The following structures currently exist:

- MT_DbRecListCond_t
- MT_DBT1Rec_t
- MT_DbT1Hd_t
- MT_DBT1Ud_t
- MT_Ed_t
- MT_EdLife_t
- MT_Geo_t
- MT_GeoExtended_t
- MT_Tl_t

- MT_Wear_t

16.7.3 State Upon Delivery

Database

In the delivery state, the database tables are configured as follows:

Database size

- DBT1: 10 data records
- DBT2: 10 data records

Table division

DBT1:

- Sector 1: 5 locations
- Sector 2: 5 locations

DBT2:

- Sector 1: 5 locations
- Sector 2: 5 locations

Data scheme

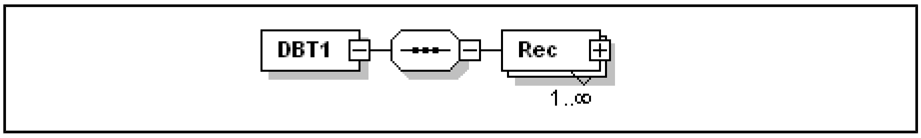


Fig. 16-162: Default data scheme configuration (1)

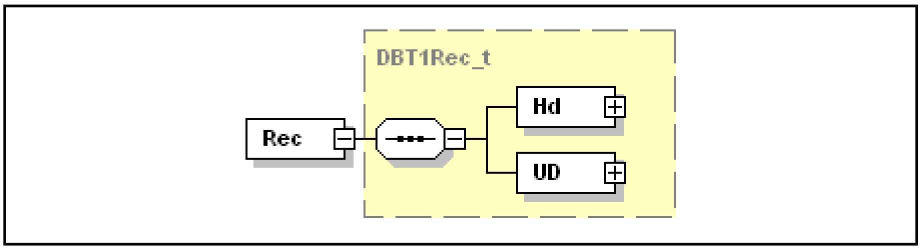


Fig. 16-163: Default data scheme configuration (2)

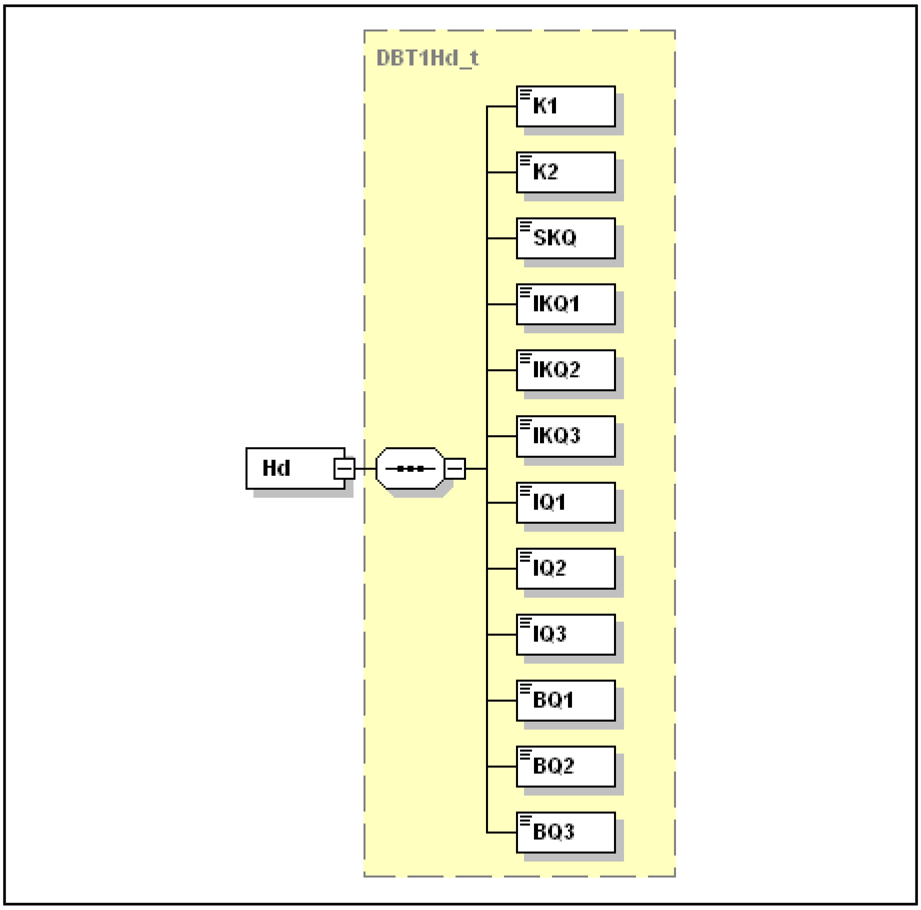


Fig. 16-164: Default data scheme configuration (3)

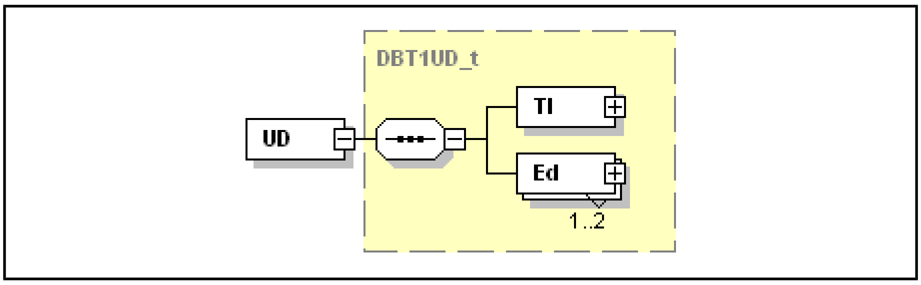


Fig. 16-165: Default data scheme configuration (4)

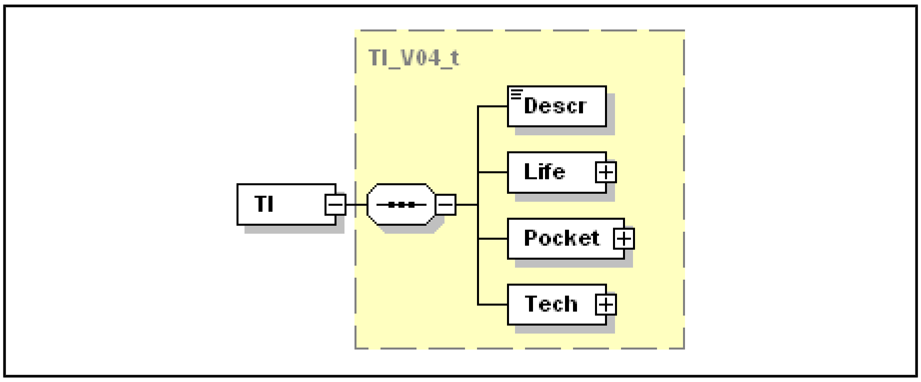


Fig. 16-166: Default data scheme configuration (5)

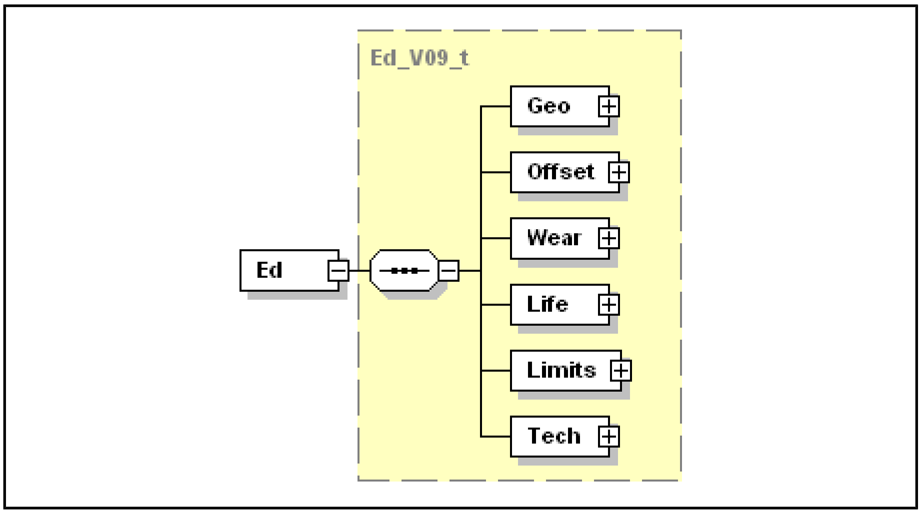


Fig. 16-167: Default data scheme configuration (6)

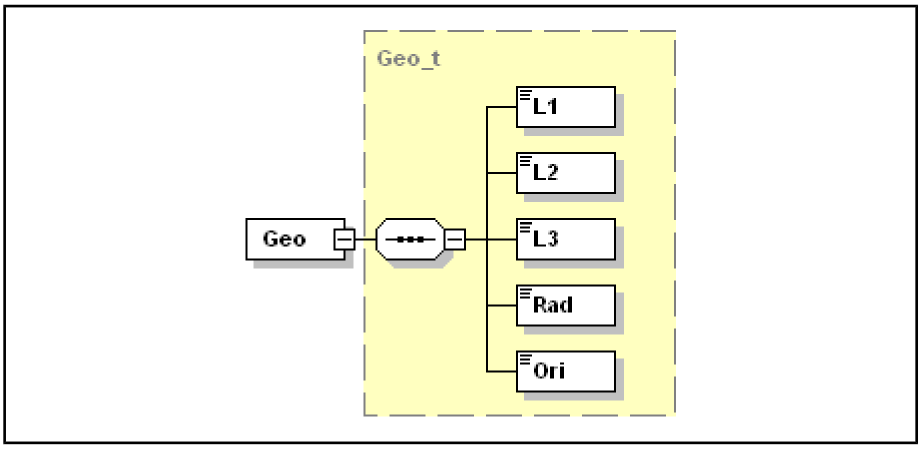


Fig. 16-168: Default data scheme configuration (7)

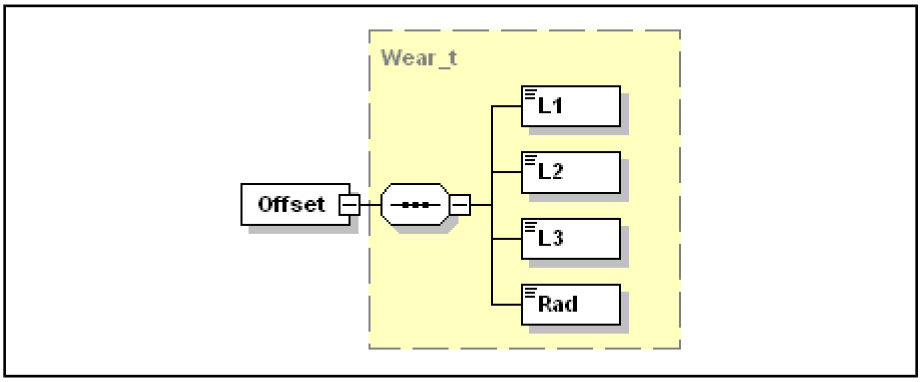


Fig. 16-169: Default data scheme configuration (8)

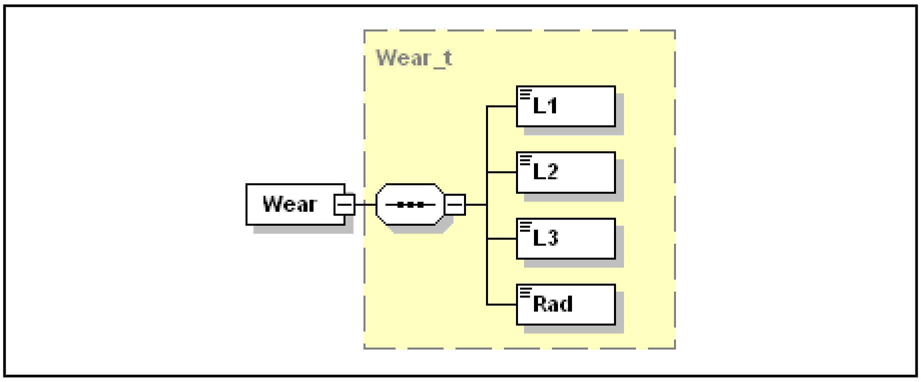


Fig. 16-170: Default data scheme configuration (9)

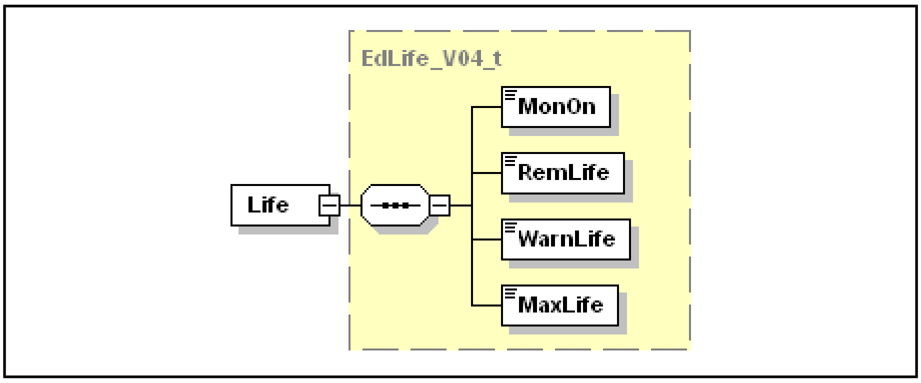


Fig. 16-171: Default data scheme configuration (10)

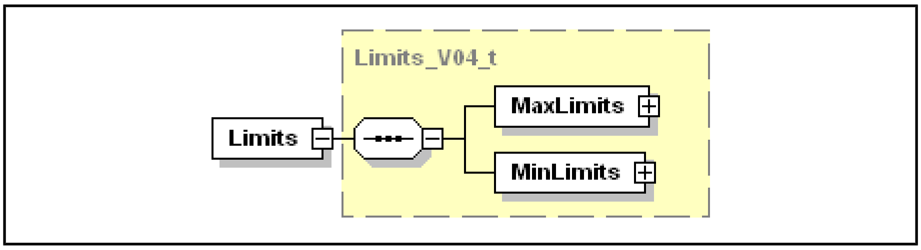


Fig. 16-172: Default data scheme configuration (11)

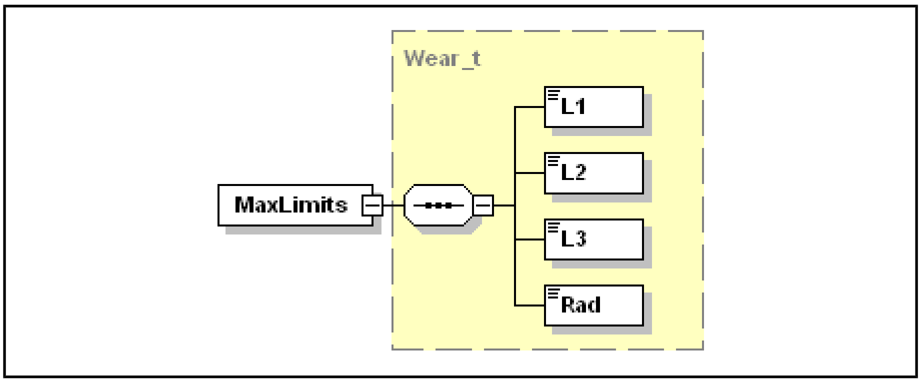


Fig. 16-173: Default data scheme configuration (12)

Status bits

chapter 25.3.1 "Status Bits" on page 591

Tool Catalog

chapter 25.3.2 "Tool Catalog" on page 594

Bitmap libraries

chapter 25.3.3 "Bitmap Libraries" on page 661

User Interface

Tool lists

The following lists are provided to the user in the supplied condition for direct use or to configure tool management:

- Geometry list
- Wear list
- Tool life list
- Status list
- List of all worn tools (same layout as tool life list)

Geometry List

Column division:		Header
Sector		S
Location		P
Tool name		Tool name
Duplo number		DN
T number		TN
Status		Status
	Warning limit reached	tw
	Tool worn	Two
	Tool locked	TL
Tool edge number		SN
Geometry		Geometry
	L1 value	L1
	L2 value	L2
	L3 value	L3
	Radius	R
	Edge position	O
Tool type		Type
Row division:		
One line per tool edge - irrelevant lines are hidden.		

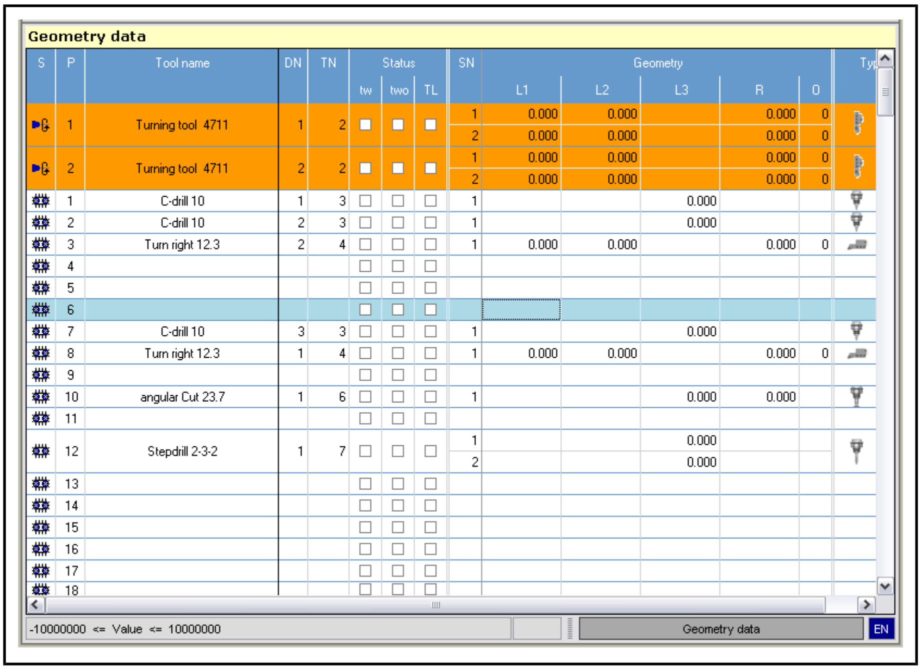


Fig. 16-174: Geometry list

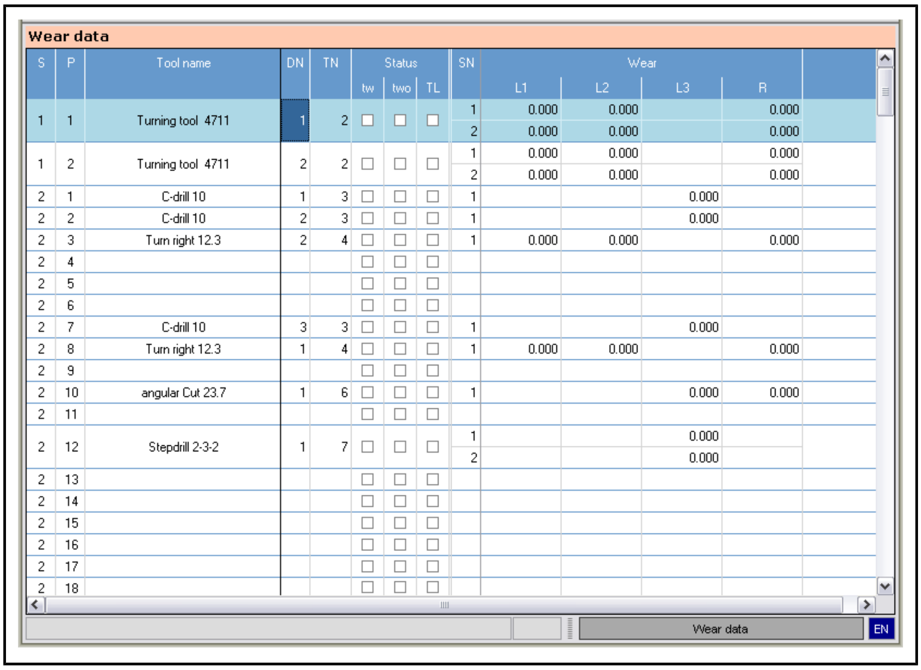
Wear List

Column division:	**Header**
Sector	S
Location	P
Tool name	Tool name
Duplo number	DN
T number	TN
Status	Status
Warning limit reached	tw
Tool worn	Two
Tool locked	TL
Tool edge number	SN
Wear	Geometry
L1 value	L1
L2 value	L2
L3 value	L3
Radius	R

Row division:

One line per tool edge - irrelevant lines are hidden.

Configuring the Tool Management



Wear data

S	P	Tool name	DN	TN	Status tw	Status two	Status TL	SN	Wear L1	Wear L2	Wear L3	Wear R
1	1	Turning tool 4711	1	2	☐	☐	☐	1	0.000	0.000		0.000
								2	0.000	0.000		0.000
1	2	Turning tool 4711	2	2	☐	☐	☐	1	0.000	0.000		0.000
								2	0.000	0.000		0.000
2	1	C-drill 10	1	3	☐	☐	☐	1			0.000	
2	2	C-drill 10	2	3	☐	☐	☐	1			0.000	
2	3	Turn right 12.3	2	4	☐	☐	☐	1	0.000	0.000		0.000
2	4				☐	☐	☐					
2	5				☐	☐	☐					
2	6				☐	☐	☐					
2	7	C-drill 10	3	3	☐	☐	☐	1			0.000	
2	8	Turn right 12.3	1	4	☐	☐	☐	1	0.000	0.000		0.000
2	9				☐	☐	☐					
2	10	angular Cut 23.7	1	6	☐	☐	☐	1			0.000	0.000
2	11				☐	☐	☐					
2	12	Stepdrill 2-3-2	1	7	☐	☐	☐	1			0.000	
								2			0.000	
2	13				☐	☐	☐					
2	14				☐	☐	☐					
2	15				☐	☐	☐					
2	16				☐	☐	☐					
2	17				☐	☐	☐					
2	18				☐	☐	☐					

Wear data EN

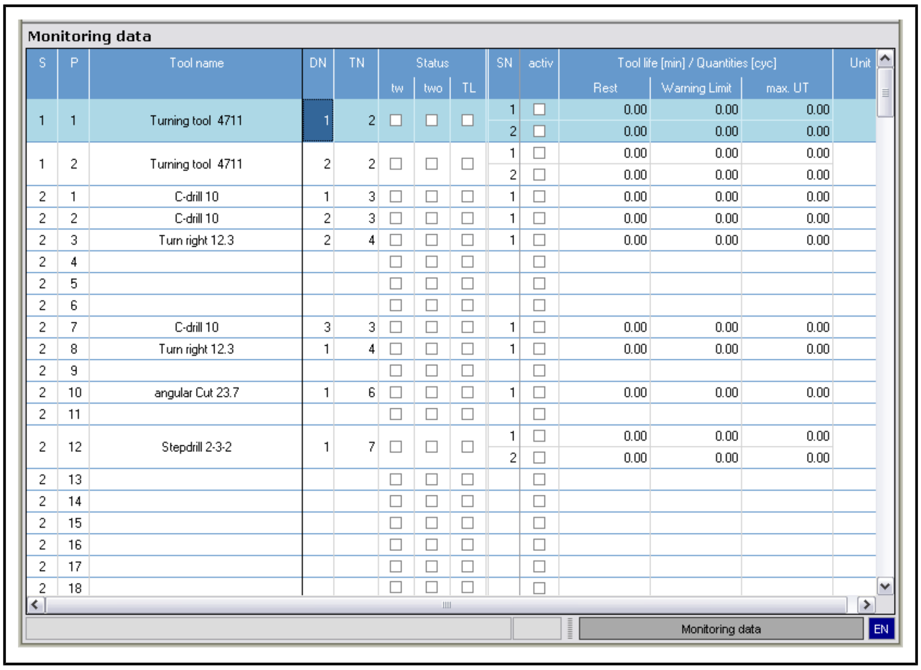
Fig.16-175: Wear list

Tool Life List

Column division:		**Header**
Sector		S
Location		P
Tool name		Tool name
Duplo number		DN
T number		TN
Status		Status
	Warning limit reached	tw
	Tool worn	Two
	Tool locked	TL
Tool edge number		SN
Monitoring status		Active°
Tool life		Tool life[min] / no. of pieces[cyc]
	Remain. tool life	Rest
	Warning limit	Warn. lim.
	Maximum utilization time	Max. UT
Time unit		Unit

Row division:

One line per tool edge - irrelevant lines are hidden.



Monitoring data

S	P	Tool name	DN	TN	Status tw	Status two	Status TL	SN	activ	Tool life [min] / Quantities [cyc] Rest	Warning Limit	max. UT	Unit
1	1	Turning tool 4711	1	2	☐	☐	☐	1	☐	0.00	0.00	0.00	
								2	☐	0.00	0.00	0.00	
1	2	Turning tool 4711	2	2	☐	☐	☐	1	☐	0.00	0.00	0.00	
								2	☐	0.00	0.00	0.00	
2	1	C-drill 10	1	3	☐	☐	☐	1	☐	0.00	0.00	0.00	
2	2	C-drill 10	2	3	☐	☐	☐	1	☐	0.00	0.00	0.00	
2	3	Turn right 12.3	2	4	☐	☐	☐	1	☐	0.00	0.00	0.00	
2	4				☐	☐	☐		☐				
2	5				☐	☐	☐		☐				
2	6				☐	☐	☐		☐				
2	7	C-drill 10	3	3	☐	☐	☐	1	☐	0.00	0.00	0.00	
2	8	Turn right 12.3	1	4	☐	☐	☐	1	☐	0.00	0.00	0.00	
2	9				☐	☐	☐		☐				
2	10	angular Cut 23.7	1	6	☐	☐	☐	1	☐	0.00	0.00	0.00	
2	11				☐	☐	☐		☐				
2	12	Stepdrill 2-3-2	1	7	☐	☐	☐	1	☐	0.00	0.00	0.00	
								2	☐	0.00	0.00	0.00	
2	13				☐	☐	☐		☐				
2	14				☐	☐	☐		☐				
2	15				☐	☐	☐		☐				
2	16				☐	☐	☐		☐				
2	17				☐	☐	☐		☐				
2	18				☐	☐	☐		☐				

Fig. 16-176: Tool life list

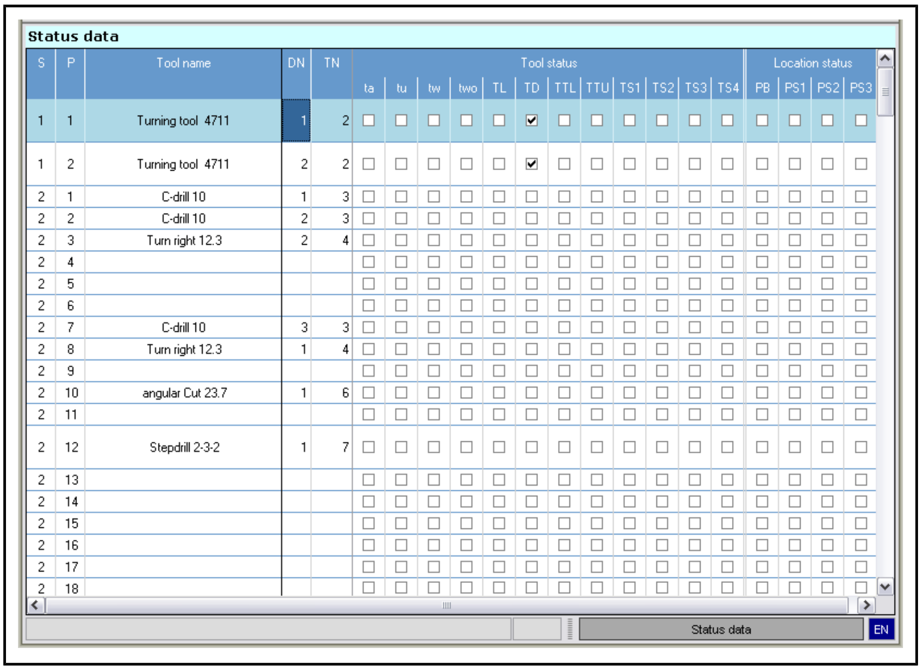
Status List

Column division:		Header
Sector		S
Location		P
Tool name		Tool name
Duplo number		DN
T number		TN
Tool status		Tool status
	Tool active	ta
	Tool used	tu
	Warning limit reached	tw
	Tool worn	Two
	Tool locked	TL
	Tool broken	TD
	Tool loaded	TTL
	Tool unloaded	TTU
	Tool user status 1	TS1
	Tool user status 2	TS2
	Tool user status 3	TS3
	Tool user status 4	TS4
Empty column		
Location status		Location status
	Location blocked	PB
	Location user status 1	PS1
	Location user status 2	PS2

Column division:	**Header**
Location user status 3	PS3

Row division:

One line per tool edge - irrelevant lines are hidden.



Status data

S	P	Tool name	DN	TN	Tool status ta	tu	tw	two	TL	TD	TTL	TTU	TS1	TS2	TS3	TS4	Location status PB	PS1	PS2	PS3
1	1	Turning tool 4711	1	2	☐	☐	☐	☐	☐	☑	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
1	2	Turning tool 4711	2	2	☐	☐	☐	☐	☐	☑	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	1	C-drill 10	1	3	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	2	C-drill 10	2	3	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	3	Turn right 12.3	2	4	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	4				☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	5				☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	6				☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	7	C-drill 10	3	3	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	8	Turn right 12.3	1	4	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	9				☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	10	angular Cut 23.7	1	6	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	11				☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	12	Stepdrill 2-3-2	1	7	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	13				☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	14				☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	15				☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	16				☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	17				☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
2	18				☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐

Status data EN

Fig.16-177: Status list

Tool editor

The default configuration of the tool editor provides for division into 4 containers:

- Container 1 comprises:
  - Basic tool data control
- Container 2 comprises:
  - Location and tool status control
  - Tool data control
- Container 3 comprises:
  - Edge data control
  - Limit value data control
- Container 4 comprises:
  - Bitmap control (with dimensions)
  - Bitmap control (without dimensions) - only in Insert editor

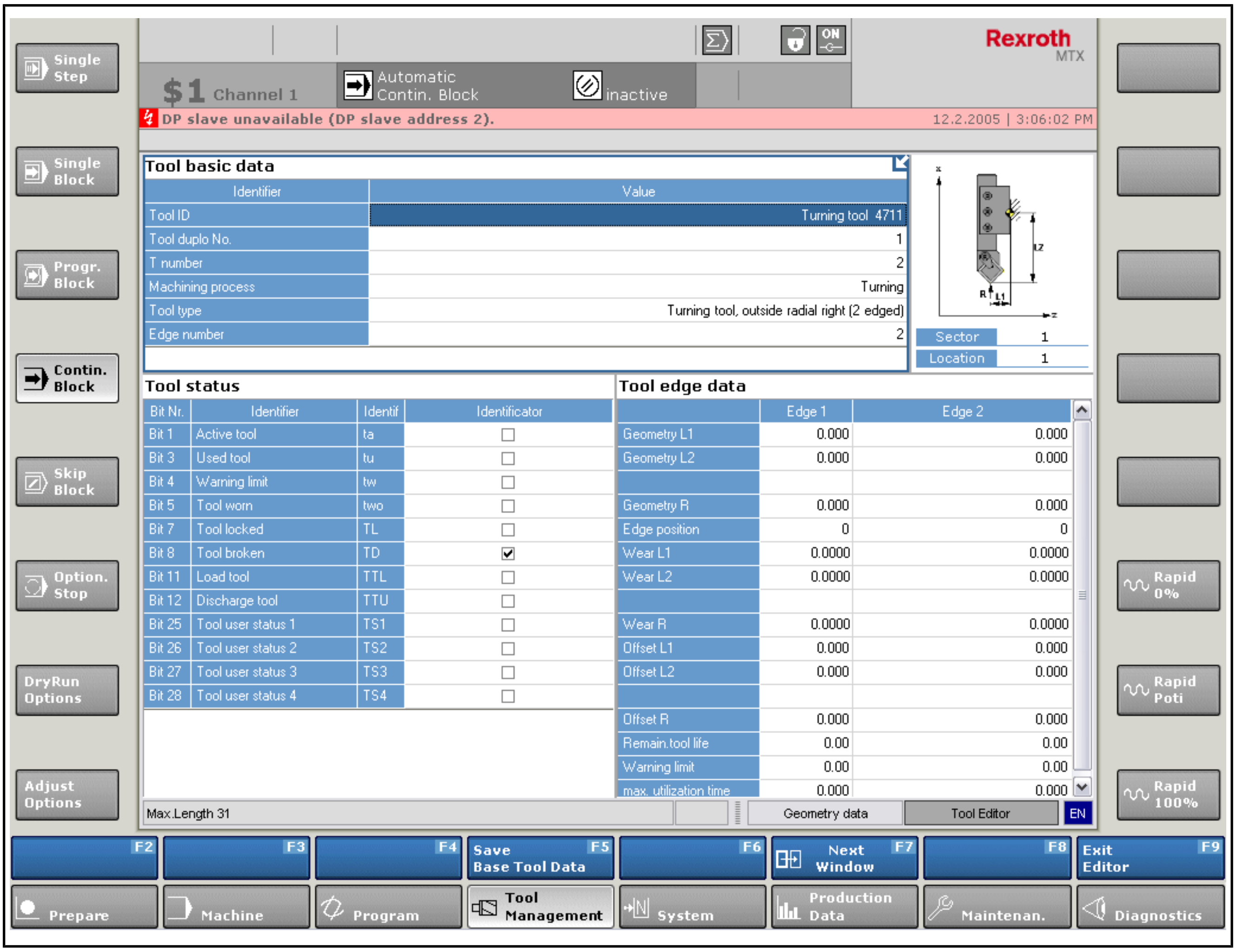


Fig. 16-178: Default tool editor configuration

16.8 Commissioning Simple Tool Management

Example for a system

- 1 spindle location
- 2 gripper locations
- 12 magazine locations

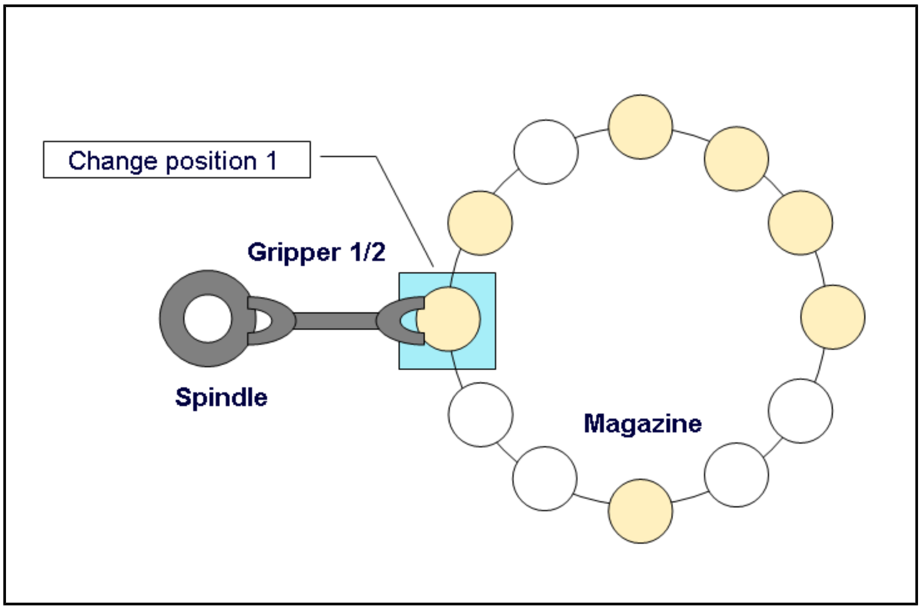


Fig.16-179: Example for a system

This instruction is not intended to be used for a real application, which is why it is not complete.

It merely serves for giving an impression of the MTX tool management.

Ideally, one of the already installed example projects with the respective PLC program should be used as configuration basis.

Configuring the Database

Configuration of the Database

These steps are carried out in IW Operation.

1. Copy dbt1prms.dat from FEPROM to mnt.
2. Carry out the red changes in the copied dbt1prms.dat (see the following program listing).
3. Copy the changed dbt1prms.dat from mnt to usrfep.

   If there is another file with the same name, rename it.

Program:

```
...
; then regenerated according to sector configuration
0 0
; sector configuration
; P0: 1-99 (corresponds to sectors 1 - 99)
; P1: number of places in the sector
1 1
2 2
3 12
```

Display configuration in the operation area "Machine"

There are two possibilities to display the active tool in the "Machine" operating area:

- Displaying via the CPL variable
- Displaying via the SD variable

The control for the display of the active tool has to be set accordingly.

When using system data to display the active tool, no additional settings are required.

If displaying is carried out via a CPL variable, the following settings have to be carried out:

CPL Variable Definition for Displaying the Active Tool in the Operating Area "Machine"

1. Create a new text file and enter the following (see `Program:`)
2. Save the file to mnt using the name wmhperm.dat.
3. Copy wmhperm.dat from mnt to usrfep.

   If there is another file of the same name in usrfep or root, rename it.

Program:

```
DEF INT @ACTTOOL01;
DEF INT @ACTTOOL02;
DEF INT @ACTTOOL03;
DEF INT @ACTTOOL04;
DEF INT @ACTTOOL05;
DEF INT @ACTTOOL06;
DEF INT @ACTTOOL07;
DEF INT @ACTTOOL07;
DEF INT @ACTTOOL08;
DEF INT @ACTTOOL09;
DEF INT @ACTTOOL10;
DEF INT @ACTTOOL11;
DEF INT @ACTTOOL12;
DEF INT @PRETOOL01;
DEF INT @PRETOOL02;
DEF INT @PRETOOL03;
DEF INT @PRETOOL04;
DEF INT @PRETOOL05;
DEF INT @PRETOOL06;
DEF INT @PRETOOL07;
DEF INT @PRETOOL08;
DEF INT @PRETOOL09;
DEF INT @PRETOOL10;
DEF INT @PRETOOL11;
DEF INT @PRETOOL12;
```

Defining system data for the NC program

Defining a System Variable

These steps are carried out in the scheme editor.

1. Define System Variables

   Calling the scheme editor with: **Start Program FilesRexrothIndraWorksMTX ToolsScheme Editor**
2. Menu bar: **FileNew** select an **XML Document** and press **"OK"**.
3. Enter the following text in the center column: (see `Program:`)
4. Save the file in mnt by selecting **FileSave As** and use the name SDDefMTB.dat.
5. Exit scheme editor.
6. Copy SDDefMTB.dat from mnt to usrfep.
7. If there is another file of the same name in usrfep or root, rename it.

Program:

```
<?xml version="1.0" encoding="UTF-8"?>
<SDDEF>
   <Variable Storage="volatile">
      <Name>ToolStr</Name>
      <Type>DBT1Rec_t</Type>
   </Variable>
</SDDEF>
```

Carrying out a Data Backup

These steps are carried out in IW Engineering.

1. Exit IW operation.
2. Create tar file
3. Back up the PLC program

Carrying out a System Restart

These steps are carried out in IW Engineering.

1. Below the control node **MTX System Status** <NC Restart...> Startup mode 6 - Bootstrapping and waiting till RUN appears
2. Restore the tar file.
3. Below the control node **MTX System Status** <NC Restart...> Startup mode 0 - Standard operation and waiting till RUN appears
4. Load the PLC Program.

   This step is also executed in IW Engineering.

Preparing tool data

Entering Tools in the Database

These steps are carried out in IW Operation.

1. Open OP5 in IW Operation.
2. Enter the following tools in the magazine:

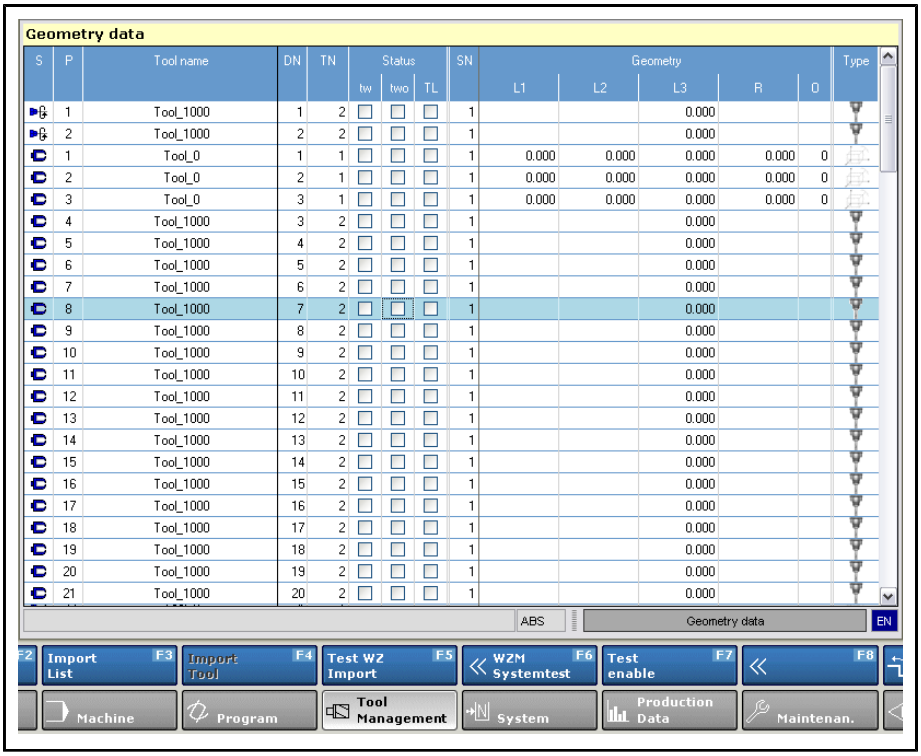


Fig. 16-180: Entering tools in the database.

Creating the NC program

Creating a Tool Exchange Subroutine (in CPL)

Open OP4 in IW Operation.

1. Create the following NC subroutine: (see `Program:`)
2. Save the program using the name ToolChange.

Program:

```
; ToolChange
;
; subroutine for replacing the tool on the magazine place P1%
; by the tool in the spindle
; via the gripper places 1 and 2 as intermediate stations
;
11 PLACE%=P1%
12 DBMOVE("/DBT1",3,PLACE%,2,1) : REM magazine place P1%
            -> gripper place 1
13 DBMOVE("/DBT1",1,1,2,2)      : REM spindle
            -> gripper place 2
14 DBMOVE("/DBT1",2,1,1,1)      : REM gripper place 1
            -> spindle
15 DBMOVE("/DBT1",2,2,3,PLACE%) : REM gripper place 2
            -> magazine place P1%
M30
```

1. Creating the 3rd Tool Change Program

1. Create the following NC program: (see `Program:`)
2. Save the program in the same directory as the NC subroutine using the name Test1.npg.
3. Let the NC program run through the program block while observing the tool list (OP5).

Program:

```
P ToolChange(7)
P ToolChange(4)
P ToolChange(11)
M30
```

2. Creating the 3rd Tool Change Program

1. Create the following NC program: (see `Program:`)
2. Save the program in the same directory as the other programs using the name Test2.npg.
3. Let the NC program run through the following block while observing the tool list (OP5).

Program:

```
21 FOR I%=1 TO 12
22   SD.ToolStr = DBTAB("DBT1.Rec",3,I%,RESULT%)
23   IF RESULT% = 0 THEN
24    IF SD.ToolStr.Hd.IKQ3 = 1 THEN
25     IF SD.ToolStr.Hd.IKQ1 <> 3 THEN
26        SD.ToolStr.Hd.BQ2 = SD.ToolStr.Hd.BQ2 OR 16
27       DBTAB("DBT1.Rec",3,I%) = SD.ToolStr
28     ENDIF
29    ENDIF
30   ENDIF
31 NEXT I%
```

The program will set the bit "Tool worn" with all twist drills (tool number 1), except for the twist drill with the duplo number 3.

3. Creating the 3rd Tool Change Program

1. Create the following NC program: (see `Program:`)
2. Save the program in the same directory as the other programs using the name Test3.npg.
3. Let the NC program run through the following block while observing the tool list (OP5):

The following program example applies for the display variant via the CPL variable.

Program:

```
40 SD.ToolStr.Hd = DBSEA
                   (“DBT1”,-1,-1,”BQ2=0:16,IKQ3=1,K1=3”,Found%)
41 IF Found%=1 THEN
      P ToolChange(SD.ToolStr.Hd.K2)
42 ENDIF
43 @ACTTOOL01% = SD.ToolStr.Hd.IKQ3
M30
```

☞ The only non-worn twist drill will be installed in the spindle.

The content of the CPLvariables @ACTTOOL01% will be displayed in the operating area Machine under "Tool active" (by default, at the bottom right of the operation screen). In the current case, line 43 contains the tool number for the replaced twist drill, i.e. number 1.

17 Configuring System Data Displays

17.1 General

17.1.1 Purpose of this Documentation

It is possible to visualize individual system data structures in freely definable list displays in IndraWorks Operation.

These list displays are configured in IndraWorks Engineering under the "SystemData Screens" configuration node.

17.1.2 Involved Components in IndraWorks

Screen Configurator

The screen is defined in the screen screen configurator below the "Screens" configuration node in the IndraWorks Engineering, see HMI documentation on page 499.

System Data Screen Configurator

A list configuration can be created in IndraWorks Engineering to visualize a system data structure like for the tool lists via the "SystemData Screens" configuration node.

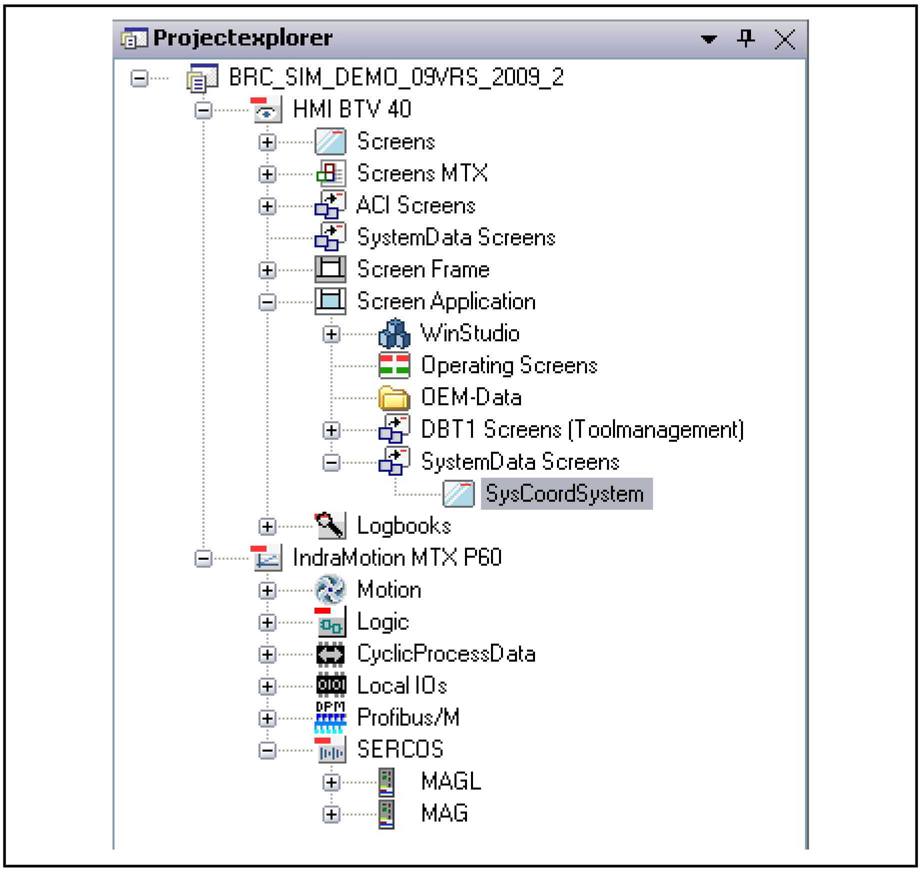


Fig.17-1: SystemData Screen in the Project Explorer

Only one SD variable can be visualized in one system data screen.

F-key Configurator

see chapter chapter 6.5 "Creating F-Keys " on page 110

17.1.3 Commissioning Tools

ULC configurator

see chapter chapter 16.2.4 "ULC Configurator" on page 311

XML Editor

see chapter chapter 16.2.5 "XML File Editor" on page 329

Schema Editor

see chapter chapter 16.2.3 "Schema Editor" on page 307

System Data Configurator

To specify the system data definitions in IndraWorks Operation, a system data configurator is available.

17.1.4 Definition of Terms

ULC **"ULC"** stands for "**U**niversal **L**ist **C**ontrol", which is the central element of the system data list configuration.

Sublist **"Sublist"** is a central term in schema definition for the ULC. A ULC is a table editor which may be able to multiply the configured line-column definition for the presentation of an array of system data structures according to the number of array elements. In this case, a sublist stands for the presentation of a system data structure. Thus, only one sublist has to be defined to display a multidimensional system data structure.

17.2 Handling Instruction: Creating a System Data Screen

Creating a System Data Screen

None

1. Create an SD list configuration

   New entry under "SystemData Screens"

2. Create / edit the F-key panel of the SD visualization
3. Create a screen definition:
   1. <New Screen>
   2. Select "MtxSystemDataList" image type
   3. Configure the list definition
   4. It is not required to define the F-panel. The predefined F-panel is always used.
   5. Assign screen to an operating area
4. Define the call of the SD.Screens in the operating area panel:
   1. Call the F-key configurator of the respective operating area panel
   2. Select "Image change" for the respective function key
   3. The newly created SD screen can be selected in the "Image name" box.
   4. Specify F-key labeling if required

18 Configuring the NC Program Editor/Text Editors

18.1 Overview on the configurations steps

Editor types

To edit text files and NC programs, two editor types are available in IndraWorks Operation:

- NC program editor
- Text editor

The NC program editor provides additional functions for the NC programming while the text editor is more simple and also suitable for bigger files.

☞ The following steps can be made either in the NC program editor or in the text editor. Thus, it is simply called "editor" in the following description.

Configuration steps

The same possibilities to adjust and change options are provided for both editor types. During installation, each editor is provided with all functionalities and a basic layout so that no configuration steps are required when working with each editor. Special user or machine manufacturer requirements can be configured. The following configurations are distinguished:

Options

The editor behavior and its layout can be set under **Tools ▸ Options ▸ Editor Options...**:

- Criteria when to use the editor types
- Tab positions
- Font type
- Behavior when assigning block numbers.

The options of the editor are modified in IndraWorks Operation. The settings are assigned to the Windows user currently logged in and may only be changed in exceptional cases during commissioning. Thus, these settings are not described in detail in the commissioning manual. Instead, it is referenced to the chapter "Options - Options Editor..." in the manual "Rexroth IndraMotion MTX Standard NC Operation".

Data for input support

Both editors are provided with identical input supports. These are integrated tools that insert individual words or complete sections and partially also correct them. This procedure is dialog-guided.

Data can be provided for the following parts of the input support:

- The **NC block sequences** are programming templates
- The **input masks** are input tables to edit parameter values to instructions

The input masks of the user and machine manufacturer, the information on hidden input masks as well as the NC block sequences are part of the IndraWorks project.

F-keys of the input support

The function keys to insert commands/calls via input mask can be changed in their allocation. These settings are performed in IndraWorks Engineering.

Two configurable F-panels are available. The following section describes the particularities when adjusting these panels.

The F-keys of the editor are part of the IndraWorks project.

18.2 NC Block Sequences

The **NC block sequences** are a collection of templates for certain instructions, program sections or even up to complete programs that can be selected by the respective operation and added to the NC program.

Configuring

After installation, this collection is empty at first and has to be filled with own contents by collecting or importing if necessary. This configuration step is made in IndraWorks Operation and described in the chapter "Input Help - NC Block Programming" in the documentation "Rexroth IndraMotion MTX Standard NC Operation".

The section 18.5.2 NC Block Sequences, page 464 is a handling instruction on filling block sequences.

18.3 Input Masks

The **input masks** are tables for comfortable editing of program sections in which changeable parameters play a major role. They are mainly used to program cycle calls.

Configuring

After installation, IndraWorks Operation is already provided with a high number of input masks for standard cycles and other instructions. For individual cycles or instructions to be programmed often, it can be required to self-define the input masks (with an external text editor, see chapter 19 Definition of Input Dialogs, page 471) or to import existing input masks.

The chapter 18.5.3 Input Masks, page 465 comprises a handling instruction on how to create input masks.

Setting up cycles on the NC

Due to the principal application of the input masks, there is a close connection to the cycles. Thus, it is noted here that is might be necessary to set up cycles in the control. Refer to the chapter 20 Setting up NC Cycles, page 497.

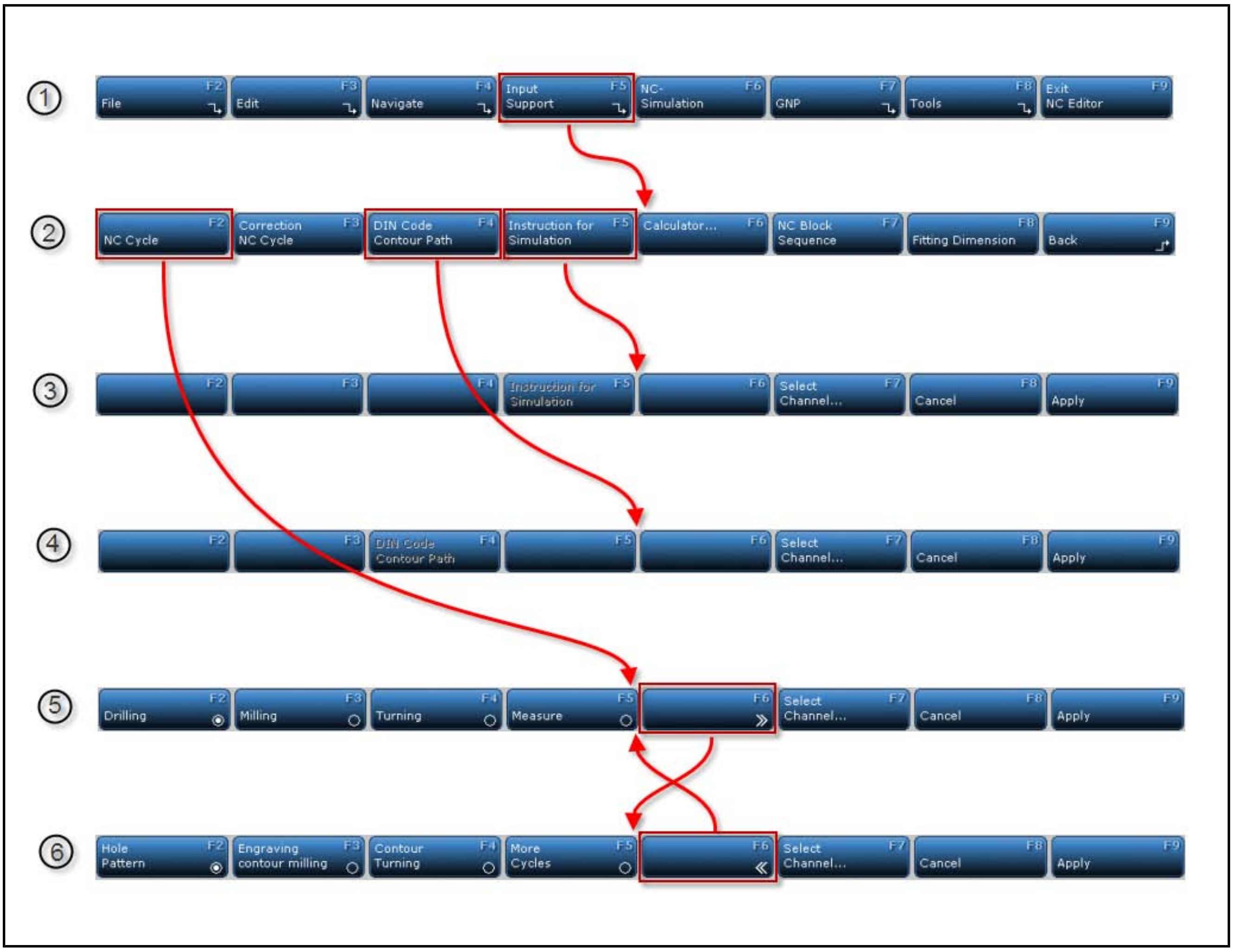
18.4 F-Keys for Input Support

18.4.1 General Information on F-Panels

F-panels

In total, there are four F-panels to insert instructions via input masks:

1. For the configurable **"left part of the menu"**, also refer to legend 5 in the following figure. This panel named "InsertCycle1", also called left half of the menu, is the entry level for inserting cycle calls. On this panel, group filters and different characteristics can be combined for each F-key. When pressing this key, the provided mask scope is reduced to those that are provided with at least one of the characteristics listed in the filter. Switching between different group filters is illustrated by selection buttons for these keys. If so many different filter are desired that they do not fit on one panel, the key for switching to a subsequent level is provided
2. The subsequent level is called **"right half of the menu"** and designated with **"InsertCycle2"**. This panel can also be configured and has to be provided with a key for level switching back to the left menu level
3. The non-configurable panel with **instructions for simulation**
4. The non-configurable panel for **DIN instructions and basic contour elements**



(1) Main Level
(2) "Input support" panel
(3) "Instruction for Simulation" panel
(4) "DIN Code Contour Path" panel
(5) "Left half of the menu" panel (name InsertCycle1)
(6) "Right half of the menu" panel (name InsertCycle2)

Fig. 18-1: Overview on F-panels for insertion via input masks (default assignment shown)

F-keys

Five special functions are allowed on the configurable F-panels "InsertCycle1" and "InsertCycle2":

- **"Set Cycle Group Editor"** to define a group filter
- **"Level Switching Editor"** to switch between the left and the right half of the menu
- **"Channel Selection Editor"** for additional limitation of the menu to one channel
- **"Cancel Entries Editor"** to discard entries
- **"Apply Parameters Editor"** to add the call to the NC program

F-panel editor

The panels are adjusted in the F-panel editor in IndraWorks Engineering.

☞ The F-panels "InsertCycle1" and "InsertCycle2" may not be deleted.

When modifying the F-panel, the following rules are to be observed (IndraWorks Engineering does not check whether they are observed) apart from the exclusive usage of the five named functions:

18.4.2 Function "Set Cycle Group Editor"

Set Cycle Group Editor

The "Set Cycle Group Editor" function allows the key to enable a specified group filter. The "Group" box lists all pre-defined group characteristics. Select a group or enter several group named separated by spaces. Individual group names can also be used.

The chapter Properties of an Input Mask (Grouping), page 480 describes how the input masks are assigned to the individual groups and the meaning of the predefined groups.

Rules

The following rules and particularities apply:

- F-keys with this function are automatically subject to a selection functionality
- Each key can be assigned to one image. Its position (preferred orientation on the bottom right, X=5 and Y=3) is used to set the selection button image
- F-keys, whose group filters specify an empty set are hidden and do not have to be deleted
- Two group filters without an empty set are to be specified at least on each panel. Alternatively, all group filters may specify an empty set on the panel "InsertCycle2" or all keys with the "Set Cycle Group" function may be deleted. Both conditions cause that the "InsertCycle2" panel does not appear in the dialog
- A group name can be used an unlimited number on different F-keys
- One F-key should always be intended with the group filter "_OTHERGROUPS". This special filters are for groups that are not selected on other F-keys by a filter. If such input masks that are not assigned, exist, these are hidden and cannot be inserted
- The F-keys with group filters should be assigned consecutively

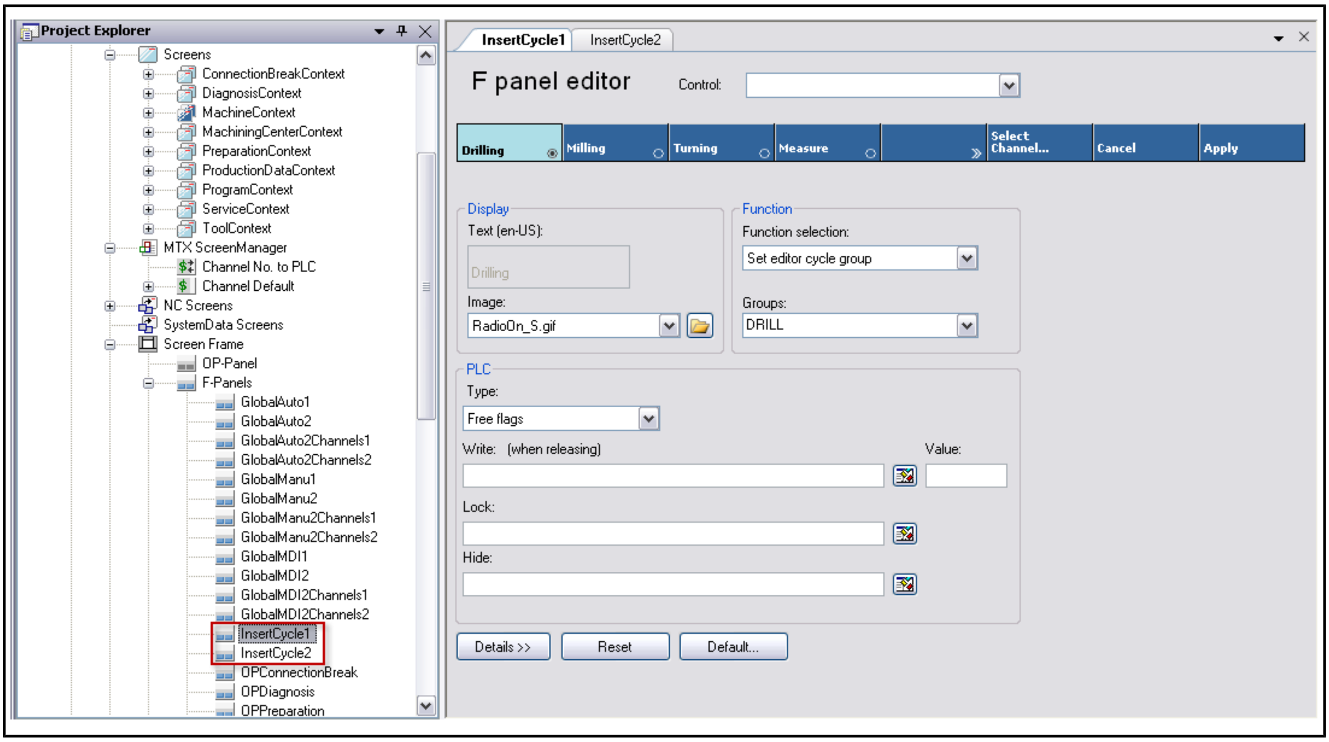


Fig. 18-2: Configuring F-keys of the input support "Set Cycle Group Editor"

18.4.3 "Level Switching Editor" Function

Level Switching Editor

The "Level Switching" function switches between the left and the right half of the menu. Enter the next F-panel name under "Level Name".

Rules

The following rules and particularities apply:

- The panel name always has to be the counterpart from "InsertCycle1" and "InsertCycle2" of the panel currently edited
- If all group filters of the "InsertCycle2" label empty sets, the key to switch to the level "InsertCycle2" is hidden. In this case, it may be deleted or overwritten with the "Set Group Editor" function
- The "Level Switching Editor" function may be allocated at any position on the panel

18.4.4 "Channel Selection Editor" Function

Channel Selection Editor

The "Channel Selection" function enables a dialog in which a channel or the setting "channel-comprehensive" can be selected.

Rules

The following rules and particularities apply:

- If there is only one channel or the channel-related hiding of input masks is not used, the key with the "Channel Selection" function is hidden. In this case, it may be deleted or overwritten with the "Set Group Editor" function
- The "Channel Selection Editor" function may be allocated at any position on the panel. It is recommended to keep <F7>, since the channel selection is located on other non-configurable editor panels on <F7>

18.4.5 "Cancel Entries Editor" Function

Cancel Entries Editor

The "Cancel Entries" function quits the input mask and returns to the editor view.

Rules

The following rules and particularities apply:

- This function has to be present on each panel
- The function may be allocated at any position on the panel. It is recommended to keep <F8>, since "Cancel Entries Editor" is located on other non-configurable editor panels on <F8>

18.4.6 "Apply Parameters Editor" Function

Apply Parameters Editor This function applies the edited instruction to the editor and returns to the editor view.

Rules The following rules and particularities apply:

- This function has to be present on each panel
- The function may be allocated at any position on the panel. It is recommended to keep <F9>, since "Apply Parameters Editor" is located on other non-configurable editor panels on <F9>

18.5 Handling Instruction: Configuring Editor - Input Support

18.5.1 Creating Basic Conditions

☞ Configuring the input support of the editor is optional.

Basic conditions Ensure or create the following basic conditions:

1. An IndraWorks project is created
2. A visualization device is created in the project
3. Visualization data is transmitted and activated

18.5.2 NC Block Sequences

Applying NC Block Sequences by Importing

The configuration is made in IndraWorks Operation.

Enabling block sequences The block sequences are integrated into the editor. To enable them, the following operating steps are required:

1. Switch to the "Program" operating area
2. Open the editor To do so, select an existing file in the file list of the navigator and open it with <F6 Edit> or via the context menu. Alternatively, a new file can also be created instead. Select a directory in the directory tree and press <F2 New> → <F3 NC Program> or <F4 Text File>. The context menu can also be used instead
3. <F5 Input Support> → <F7 NC Block Sequence>

Importing block sequences If block sequences are present from another project, proceed as follows:

1. <F5 File handling> → <F6 Import...>
2. Select the file to be imported (extension ".rc") and click **Open**
3. Press <F9 Return>

Creating New Block Sequences

The configuration is made in IndraWorks Operation.

Enabling block sequences The block sequences are integrated into the editor. To enable them, the following operating steps are required:

1. Switch to the "Program" operating area

2. Open the editor To do so, select an existing file in the file list of the navigator and open it with <F6 Edit> or via the context menu. Alternatively, a new file can also be created instead. Select a directory in the directory tree and press <F2 New> → <F3 NC Program> or <F4 Text File>. The context menu can also be used instead
3. <F5 Input Support> → <F7 NC Block Sequence>

New block sequence

Proceed as follows to create a new block sequence:

1. Click on <F7 Select Channel...>. Select the channel for which the sequence should be valid in the dialog
2. <F2 New NC Sequence>
3. Enter name and content. Consider the options of multi-language support (**Info**) and confirm with **OK**

18.5.3 Input Masks

Applying Input Masks by Importing Cycles

☞ The cycle import comprises the input masks as well as the cycle subroutines stored in the NC file system. Thus, there is a partial overlapping with the chapter 497Setting up NC Cycles, page 497.

The configuration is made in IndraWorks Operation.

Applying Input Masks by Importing Cycles

If input masks are present from another project, proceed as follows:

1. Switch to the "program" operating area
2. Select an existing file with the extension ".npg" in the file list of the navigator Alternatively, a new NC program can also be created instead. Select a directory in the directory tree, press <F2 New> → <F3 NC Program>, enter any short text and complete the editor with <F9>. In this case, also select the new NC program
3. <F8 NC Prog. Functions> → <F5 Import Cycles ...>
4. Select the import file (extension ".ncc") and press **Open**
5. Select the cycle package (origin) to be imported:
   - USER cycles are input masks and cycle files of the user
   - OEM cycles are input masks and cycle files of the machine manufacturer

   Press **OK**.

Creating Input Masks

To create an input mask, external programs of the Windows operating system or similar programs such as the file explorer, text editor and graphic editor are used. The input masks are tested in IndraWorks Operation.

Examples as templates

For MTX 12VRS or higher, examples for input masks are provided when creating a new project. These input masks are suitable for familiarizing with the mask definition technique. This is the most effective procedure to define individual masks from already existing ones.

Activating examples

The familiarization with the examples shows the possibilities of input masks and helps to specify the requirements on the input masks. The example input masks are normally hidden and can be shown as follows:

1. Switch to the "Program" operating area

2. Open the editor To do so, select an existing file in the file list of the navigator and open it with <F6 Edit> or via the context menu. Alternatively, a new file can also be created instead. Select a directory in the directory tree and press <F2 New> → <F3 NC Program> or <F4 Text File>. The context menu can also be used instead
3. <F8 Tools> → <F8 Options> → <F4 Adjust Input Support...>
4. Select the "Examples" node on the "Availability" tab and press **OK**
5. <F9 Return>

Testing examples

Test the example input masks by editing all parameters and inserting calls into the NC program. The following examples with increasing complexity are provided:

- Example 1: Simple example with position parameters without foreign language support.
- Example 2: Simple example with pre-assigned address parameters, two global images and foreign language support.
- Example 3: Advanced example with mandatory, standard and alternative programmed parameters or parameters programmed in pairs. Individual image per parameter.
- Example 4: Multi-line example for DIN programming.

Work with input masks as follows:

1. <F5 Input Support>
2. <F2 NC Cycle> → <F6 >>> → <F5 More Cycles> (This key sequence is the standard key assignment for input help)
3. Select an example (SAMPLE1 ... SAMPLE4) and enter the parameter values. Use the online help as well with <F1> to find explanations on the examples.
4. Apply with <F9> and continue with the next example starting at point 2.

Defining individual input mask

Define the first input mask as follows:

1. Create a copy of the file C:\Documents and Settings\All Users\application Data\Rexroth\IndraWorks\Project\<Visualization Device>\user\config\cycles\SampleMasks.cyc in the same directory. The copy can have any name, but the extension ".cyc" has to be kept.
2. Open the cyc file with a text editor. Select the most suitable example mask.
3. Edit the individual elements of the mask definition one after the other. First, do not use any tokens for multilingualism, but texts in your language. The line "//%LANG%" has to be converted to a comment using a semicolon in front.
4. Creates the images in a size of 223 x 263 pixels in the same directory using an external graphic editor and enters the name to the cyc file (behind %GF%).
5. Assign the mask to one or several groups (//%GROUP%).
6. Delete the masks that are not required and save the cyc file.
7. Test your input masks in the editor of IndraWorks Operation and correct them if required. The mask definition saved in the editor is always updated after closing all editors.
8. To support several languages, copy SampleMasks_EN.txt and create one text file each for your native language and at least one more for English. Fill out the token and write the token number %TNxxxx% to the cyc file. Enable the //%LANG% instruction (remove preceding semicolon) and re-

place the name by your text file (without country code and extension). Test the multilingualism of the input mask in the editor. Note that the multilingual support of the modified texts is only displayed after an IndraWorks Operation restart.

9. If there are online helps, copy them to the subfolder Help\<Country Code> and adjust them to the instructions //%HELPFILE% and //%HELPTOKEN %. Otherwise, the mentioned lines have to be deleted or converted to a comment using a semicolon.
10. One user mask is now defined. Any further number of input masks can be added to this or another cyc file. As machine manufacturer, move all files belonging to the just defined masks to the folder C:\Documents and Settings\All Users\Application Data\Rexroth\IndraWorks\Project\<Visualization Device>\OEM\config\cycles.

Hiding input masks

Machine manufacturers often have to remove input masks of some standard cycles, since the machine is not suitable for certain technologies or there are no axes for certain motions. In principal, input masks can be hidden in the options of the editor (IndraWorks Operation). The option is available for machine manufacturers and users if the respective user permission is granted. To remove certain input masks from this adjustment dialog, proceed as follows:

1. Create a copy of the file C:\Documents and Settings\All Users\Application Data\Rexroth\IndraWorks\Project\<Visualization Devices>\user\config\cycles\SampleMasks.cyc in the directory C:\Documents and Settings\All Users\Application Data\Rexroth\IndraWorks\Project\<Visualization Device>\OEM\config\cycles\. The copy can have any name, but the extension ".cyc" has to be kept.
2. Open the cyc file with a text editor. Delete all example masks except the last one (SAMPLE 5).
3. Open the file IndraWorks\config\cycles\MTX.Cycles.Canned.cyc with an editor in the installation directory.
4. Copy all lines of the input mask to be hidden beginning with //%N% to the clipboard.
5. Replace the existing //%N% line with the content of the clipboard.
6. If the input mask to be suppressed is located in the line //%NAME% of an instruction %SIGN%, also transmit this line to the new cyc file.
7. Any further number of input masks to be suppressed can be added to this or another cyc file.

Example:

Example for hiding an input mask with additional signature

This is the definition of the input mask that is to be hidden:

```
//%CHBEGIN%
//%LANG% MTXCyclesCanned
//%NAME% %TN801% %SIGN%_RptPnt
//%GROUP% DIN
//%GF% Contour_Chamfer_Round_Base.gif %TN809%
//%P% %@%X REAL %TN802%
//%P% %@%Y REAL %TN803%
//%P% %@%Z REAL %TN804%
```

```
//%P% %@%CHL REAL[0..9999] %TN805% %GF% Contour_Chamfer_Length.gif %TN809% %VALID%ALT%P5%%P6%
//%P% %@%CHS REAL[0..9999] %TN812% %GF% Contour_Chamfer.gif %TN809%
//%P% %@%RND REAL[0..9999] %TN806% %GF% Contour_Round.gif %TN809%
//%P% %@%FL REAL[0.0001..99999] %TN807%
//%C% %TN811%
//%C% %TN1000%
//%C% %TN1001%
//%N% G0 (%P4%%P5%%P6%%P7%)%P1%%P2%%P3%
//%HELPFILE% MTX_StandardNCCycles.chm
//%HELPTOKEN% ncc.ContourBezelCurve.htm
//%CHEND%
```

Apply the call signature from //%N% (parameters can be omitted) and set the group characteristic "_HIDDEN". Since an additional signature was arranged with %SIGN%, add this together with the line //%NAME%.

```
//%CHBEGIN%
//%NAME% %TN801% %SIGN%_RptPnt
//%GROUP% _HIDDEN
//%N% G0
//%CHEND%
```

Loading Data from Operating Station

To backup input masks or block sequences in the "IndraWorks Engineering" project, start IndraWorks Engineering, right click on the node of the visualization device and start the process under **Visualization Data ▸ Load changes from the operating station**.

18.5.4 F-Keys of the Input Support

Menu Design

At a new project, the IndraWorks Operation editor is provided with a menu structure to select groups of input masks that covers all existing and not permanently hidden input masks. There is no channel-specific restriction. If groups are irrelevant for the machine, it can be required to replace menu keys without function with menu keys with functions or to omit the "right half of the menu".

Preliminary considerations

Ask the following questions:

1. Which different group names are interesting for my project? Which groups should not be hidden? Determine the group names known by installed and individual input masks as follows:
   - Start the IndraWorks Operation desktop and switch to the "Program" operating area.
   - Open the editor To do so, select an existing file in the file list of the navigator and open it with <F6 Edit> or via the context menu. Alternatively, a new file can also be created instead. Select a directory in the directory tree and press <F2 New> → <F3 NC Program> or <F4 Text File>. The context menu can also be used instead

- <F8 Tools> → <F8 Options> → <F4 Adjust Input Support...>
- All groups are listed first on the "Availability" tab. The tooltips of the group nodes indicate the respective group names.
- After pressing the key **F-Key View**, the current menu structure and the groups and input masks hidden below can be seen.

2. Is a channel-dependent reduction of the scope of the provided input masks required? No if there is only one existing channel.

Determining the menu design

Specify the menu structure for the two configurable panels "InsertCycle1" and "InsertCycle2". The two groups "DIN" (DIN programming, contour element) and "SIM" (instructions for simulation) are located on panels that cannot be changed. These do not have to be considered. The groups can be distributed on the F-keys as follows:

- No channel-related menu restriction is required.
  – If limited to one panel, six F-keys for group selection are available. Further keys are "Cancel" and "Apply".
  – In case of two panels, ten F-keys are available in total for group selection. Three keys each are reserved per panel for level switching, "Cancel" and "Apply".
- A channel-specific menu restriction is required. It applies that the scope of the provided masks can vary channel-specifically, but not the menu design.
  – If limited to one panel, five F-keys are available for group selection. The remaining three keys of the panel are required for channel selection, "Cancel" and "Apply".
  – In case of two panels, eight F-keys for group selection are available. The remaining four keys per panel are required for level switching, channel selection, "Cancel" and "Apply".

Think about the future menu structure without considering the groups to be hidden. Each function key can be assigned to several groups. Several F-keys can also be assigned to groups.

☞ In the standard menu structure, the group filter "_OTHERGROUPS" is on the F5-key of the panel "InsertCycle2". This name stands for the remaining groups that are not selected as group filters by other F-keys. This filter should absolutely be assigned to a function key, since all new groups added at a later point in time appear under this filter. If no key is unassigned, assign this filter to a key already considered for another assignment.

Configuring Function Keys

This configuration measure is listed in IndraWorks Engineering.

Configuring in the F-panel editor

The configurable panels of the input support in the editor can be adjusted as follows:

1. Select the node **Visualization Device ▸ ScreenFrame ▸ F-Panels ▸ InsertCycle1** in the Project Explorer. That is the panel of the left half of the menu. Open the node via the context menu or double-click.
2. First, edit the keys to set the group filters:
   - If a new labeling is required, delete the key first
   - If a key for level or channel selection should be used for group selection, delete this key before as well

- Enter a new text if necessary
- Select an image (e.g. RadioOff_S.gif) and correct its position if required. It has to be located on the bottom right, at the position X=5 and Y=3
- Select the function "Set Cycle Group Editor"
- Determine a group filter under "Groups". Several group names are separated by a space.
- Repeat step 2 for all other keys for group selection. Delete keys that are not required.

3. If the key for level selection is required but different compared to the standard, delete the future key and provide it with the respective image (Next_Level_S.gif or Last_Level_S.gif). The function is set to "Level Switching Editor". Select the opposite of the currently edited panel under "Panel Name".
4. Delete the key for channel selection if still available and not required.

Keys without function are hidden. They do not have to be deleted.

5. To support multiple project languages, enable more languages under **Project ▸ Language ▸ Select Project Language...** and label the keys again.
6. Repeat the points 2 to 5 for the panel "InsertCycle2". If the right half of the menu is not required, delete all keys with the function "Set Cycle Group Editor" on this panel.

Transmitting and activating data

If IndraWorks Operation is still running, close this application now.

To activate the panels in the editor, right-click on the node of the visualization device and start the process under **Visualization Data ▸ Transmit and Activate**.

Hiding Individual Groups and Input Masks

The configuration step is made in IndraWorks Operation.

Hiding and Sequence

Proceed as follows to hide specified input masks:

1. Switch to the "Program" operating area.
2. Open the editor To do so, select an existing file in the file list of the navigator and open it with <F6 Edit> or via the context menu. Alternatively, a new file can also be created instead. Select a directory in the directory tree and press <F2 New> → <F3 NC Program> or <F4 Text File>. The context menu can also be used instead
3. <F8 Tools> → <F8 Options> → <F4 Adjust Input Support...>
4. Deselect all nodes on groups and masks that are not required anymore. Go down to the channel node for a channel-specific menu restriction.
5. The key **F-Key View** can check the assignment to the keys
6. To adjust the sequence of the masks, enable the "Sequence" tab and move the masks accordingly.
7. Press **OK**.
8. Check the menu design via <F9 Return> → <F5 Input Support> → <F2 NC Cycle>.

Loading Data from Operating Station

To backup menu adjustments in the project, start IndraWorks Engineering, right click on the node of the visualization device and start the process under **Visualization Data ▸ Load changes from operating station**.

19 Definition of Input Dialogs

19.1 Objectives and Terminology

19.1.1 Input Mask

Definition

An "input mask" is a dialog with a determined layout, in which values (parameters) are recorded in tabular form (insertion of new calls into an NC program) or are provided for correction.

Input masks also provide information on an output format. The output format establishes a connection between the program text and a specific input mask and between syntactic units and the individual lines of the input table (see chapter 19.3.7 Output Format, page 494).

Objective

Input masks enable a guided editing of selected parts of the NC program text without the help of a user documentation. The programmer is supported by graphics, texts and integral value validation mechanisms. The input masks can be directly accessed from the NC Program Editor (see "Rexroth IndraMotion MTX Standard NC Operation", "Inserting Functions ...").

NC blocks can be added and corrected with the help of input masks. Moreover, comments initiated with a semicolon and functional comments on the NC simulation can also be added and edited. However, editing CPL blocks is not possible.

The following chapters provide a definition of input masks. The syntax on which the definition is based, enables the user and machine manufacturer to create their individual masks without any knowledge of a higher programming language. The visible dialog is generated from the mask definition.

The following sections contain useful information on maintenance and error analysis.

19.1.2 The Signature of Calls and the Overlapping of Input Masks

Components of a call

It is differed between the following components of a DIN block when a call is compiled automatically:

- Unchangeable call names (subroutine names and DIN commands) followed by a parameter list,
- Parameters (either in brackets as parameter list or without brackets as DIN words),
- Other unchangeable block components without a parameter list (outside the brackets) and
- Comments (also comment lines, initiated with a semicolon).

☞ The term "parameter" has a more comprehensive meaning with regard to input masks compared to the NC point of view. As seen from the input masks, each block component with an input option is changeable and therefore a parameter. From the NC point of view, a parameter can also only be a command word, such as a parameter "Spindle command" with the values M3, M4, M5 etc.

Signature

The unchangeable block components, which are call names and constant DIN words outside the brackets, form the signature of a call. Special characters are not considered.

Example:

Signatures

- The first example includes the signature "G81".
  `G81 (IX X, SL2, DT-30, RL15); Twist drilling`
- The second example includes the signature "T M6".
  `T[4] M6`
- In the third example, the signature is "TOOL STANDARD".
  `//TOOL/STANDARD,93,55,0.8,10,3\\ [""+NCF("G90")]`

Signature function for the correction

The signature makes an unique call in the NC Editor. In the default case for input masks, there is a unique assignment of unchangeable block components to a mask. The signature property is used since they are unambiguous. This means that there is normally only one input mask for every signature. This is carried out by the NC program editor. However, it is also possible for the engineer to avoid the rule of unambiguous assignment via the signature in several ways. Thus, several masks can be provided for a signature.

☞ If a call has a signature, only the input masks are considered for correction since their signature is at least partially in the program line on which the current cursor position indicates. If no signature part is in the current line, corrections via input mask cannot be started from there (comment line, blank line, line without call word and without unchangeable block components). In this case, go to a line containing at least one word from the signature.

Signature function in case of overlapping input masks

As it is guaranteed by the system by default that only one input mask exists for each known signature, the definition of an input mask with a signature already registered, leads to the overwriting of the known input mask. This principle is used to provide the option of replacing standard cycles and their associated input masks with own ones to the machine manufacturer. Moreover, the user is also in a position to replace Bosch Rexroth input masks or input masks provided by the machine manufacturer with self-defined input masks.

Overlapping concept

In the overlapping concept, the input masks are provided in a definite hierarchy according to their origin. The following sources are distinguished:

- User-specific input masks,
- Tool machine-specific input masks and
- Default input masks (Bosch Rexroth)

In terms of overlapping, the user masks have the highest priority. The machine manufacturer's masks take priority over those of Bosch Rexroth. This overlapping takes place where the input masks are provided with an identical signature.

The overlapping concept presented is the basis for the input mask adaptation with regard to its content and partially availability.

Input masks without signature

The system also permits input masks to be defined to a limited extent, which only have changeable components in their output format. These are calls only consisting of parameters and possibly comments. During correction, the assignment of an NC block to an input mask is no longer unique and is performed by taking the parameters into account. During corrections, an input mask is provided. This input mask has the analogy with the parameters found.

☞

1. Input masks without a signature are not subject to overlapping, that is each input mask without a signature continually increases the stock of input masks.
2. Only parameters with preceding addresses can be used in input masks without signature. The listing is permitted as a single parameter type without address.
3. As only "soft" criteria are used in the assignment of NC blocks to input masks without signatures, the probability is significantly greater that several input masks with the same degree of correlation are considered for correction. Always enter as many parameters possible to make the mask assignment unique for the correction in case of input masks without signature.

19.1.3 Advanced Systematization of Calls

In addition to the differentiation criteria of calls based on the existence of a signature explained in the previous section, the following should also be taken into account:

Single line calls

All of the fixed block components and all of the parameters are in one line in case of a single line call. In addition, only comment lines initiated with a semicolon can be part of the call format.

Single line calls are normal.

Multiple line calls

With multiple line calls, the unchangeable block components and parameters are distributed across several lines. The call can be supplemented by comment lines initiated with a semicolon.

Multiple line calls are primarily used for Graphic NC Programming (GNP), specifically with geometry definitions (see "Rexroth IndraMotion MTX Workshop Programming Turning and Milling ", chapter "Geometry Definitions").

Parameter composition

Parameters can be specified in different ways:

1. As DIN parameter list.

The list is enclosed by round brackets. Parameters are separated by a comma. The DIN syntax applies. That is for example the expectation of a character string without inverted commas. For further information on parameter lists, refer to "Rexroth IndraMotion MTX Programming Manual ", chapter "Parameter Transfer to Subroutines". CPL expressions and the reference to CPL variables are only possible by enclosing the expression in square brackets (see "Rexroth IndraMotion MTX Programming Manual ", chapter "Labeling CPL Elements within a Part Program"). Such CPL elements within a DIN line are also referred to as **Inline CPLs** .

The functional comments on the NC simulation only initiated with "//", should be considered in this context like a DIN parameter list, although the Inline CPL is not allowed. Refer to "Rexroth IndraMotion MTX Workshop Programming Turning and Milling ", chapter "Additional Simulation Information in the Part Program" for functional comments on the simulation.

2. As CPL parameter list

   The entire parameter list appears as Inline CPL in square brackets. The parameters are separated by a comma. Each parameter value can be formulated as a CPL expression without additional brackets. A character string constant should be enclosed with inverted commas.

3. As a succession of DIN words.

   The parameters are directly in the line and are not summarized in a list. The parameters are separated by blank spaces, special characters (";", "(", ")", "[", "]") or with a letter following a character that is not a letter. These parameters are subject to DIN syntax. The value assignment of addressed parameters can also be performed by CPL elements within a DIN block (Inline CPL).

Address parameters

The parameters can be provided with addresses. The importance of a value is defined by the preceding address name. The programming of an address parameter is carried out as the assignment of a value to a variable. Separator characters (spaces or assignment operators) should optionally be inserted between the address and the value. Address parameters can occur in address parameter lists and in the line as a DIN word (e.g. a coordinate).

☞ The address names have to be unique within and outside a parameter list. Only for address names "G" and "M", duplications of the address name are allowed if additional value range limits are defined. This is required to be able to program different G-commands or M-commands, which belong to different modal G-groups or M-groups, in a line (using the parameter types for integers and real numbers).

Example:

Address parameters

```
G1(RND2.5, FL150) X200 Y-26
```

If all numbers of the example are seen as parameters and thus as changeable, "RND", "FL", "X" and "Y" act as parameter addresses.

Parameters without addresses

Parameters without addresses can be used in a parameter list. The meaning of a parameter is generally defined by its position within the list (counting from left to right starting from the open bracket). Therefore, these parameters are also known as **position parameters**. If individual parameters are not assigned, several commas have to be written without parameter value to maintain the assignment of subsequent parameters according to their position.

Parameters without addresses can also occur outside a parameter list. These are only DIN commands. The NC gets the meaning from the word as usual.

Example:

Parameters without addresses

Position parameters: `MYCYC["X",2,,,-3.75]`

The parameters 1, 2 and 5 are assigned. The parameters 3 and 4 are not assigned.

Parameters without address outside a parameter list:

`G97 S2500 M3`

If "G97" is seen as parameter and the G-group is assigned for spindle velocity, "G96" is also added to the value range. As "G" is included in the value, there is no address left. This parameter is then without an address outside a parameter list. The same could be done with spindle control command "M3". This parameter for the spindle command would get, for example, the value range M3, M4, M13 and M14.

No value range listing of possible values can be specified for the S value. The parameter for cutting velocity or speed is therefore converted into an address parameter.

19.2 File Structure

19.2.1 Overlapping Concept in the File Structure

File structures are provided to implement the overlapping concept for input masks. These structure provide a storage location for each origin named.

User input masks

All files defining user masks and all further files which can be used as resources (data sources), are filed in the IndraWorks Project in the <VisualizationDevice>\user\config\cycles folder.

Input masks of the machine manufacturer

The definition of the machine manufacturer's input masks and the resources are filed in the IndraWorks Project in the <VisualizationDevice>\OEM\config\cycles folder.

Input masks of Bosch Rexroth

Bosch Rexroth input masks are provided by being installed in the Rexroth\IndraWorks\config\cycles folder. The text files to support multiple languages reside in the Rexroth\IndraWorks\config folder.

Search strategy for resource files

Access to files, such as graphics, texts etc., is based on a specified search strategy. The file is initially searched for under

<VisualizationDevice>\user\config\cycles.

If the respective file is not found, it is searched in the folder

<VisualizationDevice>\OEM\config\cycles folder and finally in the particular folders for the installed input masks. This file search strategy enables individual resources to be specifically overlapped or existing resources to be used again in user masks.

Online helps are searched in the language-dependent subfolders of the named folders

(<VisualizationDevice>\user\config\cycles\Help\<CountryCode>, etc.).

19.2.2 Mask Definition Files

Function of the mask definition file

The mask definition file specifies the visible content of one or several input masks (texts, graphics and helps), the behavior (checking the type and the value range of the parameters, default values etc.,) as well as the output format of the call.

A mask definition file is an ASCII text file with the extension ".cyc", which has to be sufficient for the the syntax for input masks described below.

Considered mask definition files

Within the reserved folders for machine manufacturer's and user's masks, all files with the extension ".cyc" are considered.

The Bosch Rexroth input masks are taken from the following files of the Rexroth\IndraWorks\config\cycles folder:

- MTX.Cycles.Canned.cyc:

  Masks for standard cycles and others,
- MTX.Cycles.GnpMill.cyc:

  Masks for GNP programming, parts milling and drilling,
- MTX.Cycles.GnpTurn.cyc:

  Masks for GNP programming, parts turning and
- MTX.Cycles.Simulation:

  Programming of functional comments on the NC simulation.

Creating and editing mask definition files

The mask definition files can be edited using a simple external text editor such as WordPad.exe. Please consider that the editor can process ASCII files and that they use this file format only while saving. The NC Text Editor of the IndraWorks Operation can also be used if the file has previously been imported and then re-exported (file handling functions in the Project Navigator).

How many masks should be combined in a definition file?

Any number of input masks can be combined in a cyc-file. If masks are to be transferred independently from others to another tool machine, the masks to be transferred together should be in one file. It is recommended that several thematically-related masks are combined in one file each.

19.2.3 Graphic Files

Graphic formats

The following file formats are supported for image files:

- *.jpg,
- *.bmp,
- *.gif and
- *.tif.

A ".gif" format is recommended for drawings and ".jpg" photos.

A standardized size of 223 x 262 pixels (width x height) is provided for an optimum display of the graphics. Sections of larger images are cut.

The graphics can be created and edited using any graphic software which supports the aforementioned file formats.

No graphics defined

If there is no graphics file or if a graphics file was not provided for an input mask, the IndraWorks\config\C0.jpg will be displayed. This file can also be used as template for own images.

19.2.4 Support of Multiple National and Regional Languages

The texts for headings, parameter titles and explanatory information within the input mask can be provided in one or several languages, that is in multiple national or regional languages. Changing the languages is executed when changing the interface language.

Basically, the online helps can be provided with several languages.

Monolingual texts are directly written into the mask definition file. The possible available languages are significantly limited as this file is in ASCII format. For this reason, language-dependent texts should be preferably used.

Accessing a language-dependent text

Each input mask can be assigned to a text file containing all the input mask texts and all the comments to be added. The reference to a text to be represented/added text is carried out via a **token**. A token is the unique reference to a specific text in a language-dependent text file. In this case, it consists of a number greater than zero with one to four digits. In the mask definition file, it is accessed with "%TNxxxx%" placeholders. xxxx stands for the maximum four-digit token number.

Language variants

A text file can be in multiple languages. A file has to be provided for each language required. The name structure is subject to the following rule:

<langfile>_<CountryAbbreviation>.txt with

<langfile> constant part of the text file name (specified in the input mask definition) and

<CountryAbbreviation> two letters for the language variant according to ISO 639-1 e.g.: "DE" for German, "EN" for English, "SV" for Swedish, "ZH" for Chinese"KO" for Korean, etc.

At least one text file has to be provided for English (<langfile>_EN.text). English is the default language if there is no text file for the currently set interface language.

Text file not available in the current language

First, the text file is specified with regard to the current interface language according to the search strategy (see chapter19.2.1 Overlapping Concept in the File Structure, chapter 475). If the text file is not available in the current language, the respective English text file and then the German text file is searched in the same way.

File structures for language-dependent texts

Language-dependent texts should be provided in an ASCII file or in an UNICODE file.

Each token is initiated as follows:

```
|<tno>\----\0000\
```

including:

<tno> Token number (up to four digits, preceding zeros permitted).

The token numbers do not have to be ordered in an ascending order and may have gaps.

The respective text should be provided in the following line. If it is a multiline text, one line in the text file should be provided for each line of the text.

At the start of the text file, the token number 0 should be displayed and used in the event of an error if the addressed token does not exist.

Example:

File for language-dependent texts (in this case the German language variant "MyText_DE.txt")

```
|0000\----\0000\
No text defined
|0001\----\0000\
Retract height
|0002\----\0000\
```

```
Line 1 of a multiline text
Line 2 of a multiline text
```

Creating and editing text files

Language-dependent text files are created and changed by the user or by the machine manufacturer using a simple, external text editor (e.g. WordPad.exe). An independent provision mechanism exists for Bosch Rexroth text files.

How many text files should be prepared?

Theoretically, an individual text file can be provided for every input mask. It is the recommended method as there is rarely an opportunity to use texts with masks again. Provide at least one text file for every cyc-file in order not to unnecessarily limit the possibility of transferring single or thematically related input masks to other machines.

Help files in several languages

The assignment of help files to the individual language is carried out with a single mechanism as for the text files. Here, the country codes are not contained in the filename. Instead, subfolders are provided which are named according to the country code (<VisualizationDevice>\user\config\cycles\Help\<CountryCode> and <VisualizationDevice>\\OEM\config\cycles\Help\<CountryCode>).

If the help file is not available in the respective language, first the English file is used and then the German one.

19.3 Mask Definition Syntax

19.3.1 Design of a Mask Definition

Bracketing

Each input mask is initiated with a line

```
//%CHBEGIN%
```

and ended with a line of the content

```
//%CHEND%
```

.

Mask elements

Mask elements (graphics, parameters, output formats, explanatory texts, titles) are to be defined between these two lines, whereby each of the lines relevant to the definition begins with "//%" and describes one mask element. All other lines are not considered. Preferably initiate comments with a semicolon as in an NC program.

The sequence of mask elements in a category determines the sequence within the input mask. For example, parameters are listed in the input table from top to bottom as they are listed in the definition file.

Mask elements of different categories can be listed in any order and mixed line by line with other categories.

Sequence of input masks

☞ The sequence of mask elements of the same type is decisive for the structure of the input mask, but the sequence of the masks in the definition file has no effect on the sequence provided in the selection list while inserting. The default sequence is defined by the alphabetically ascending sorting of the signatures, wherein G- and M- commands are sorted in a numerically ascending order. The sequence can be modified in a dialog for the adjustment of the input support by shifting.

Basic information on syntax

The syntax fundamentally consists of reserved language words and placeholders enclosed in percentage symbols and constants (values, names). The reserved language words in percentage symbols determine the interpretation of the subsequent words to a large extent. At least one separating space is required between individual words, although not before and after reserved words in percentage symbols.

Names with spaces to be understood as a word (e.g. file names) can be provided with inverted commas.

19.3.2 Reference to a Language-dependent Text File

If an input mask should support several languages, a reference to a text file has to be included in the mask definition.

Syntax `//%LANG% <langfile>`

including:

<langfile> Name of the text file without two-digit language code according to ISO 639-1, without extension and without path specification. Enclose the name with inverted commas if the name contains spaces.

Example:

Language-dependent text file

Reference to the MyText_EN.txt, MyText_DE.txt, MyText_RU.txt etc. text files.

```
//%LANG% MyText
```

☞ There is also an older manner to define language-dependent text files which are still supported due to compatibility reasons.

```
//%ID% <langfile>
```

19.3.3 Title and Properties

Title

A title should be defined in each mask definition.

Syntax `//%NAME% <title>{ %SIGN% <sign>{ <sign2>...}}`

including:

<title> Title of a cycle.

The title is displayed for selection when adding and should therefore be unambiguous for the NC programmer. A language-dependent title can be achieved by specifying a token instead of the text. A token is specified using `%TN<tno>%`, where <tno> is the token number of up to four-digits in the language file (e.g. `%TN0123%`, see chapter19.2.4 Support of Multiple National and Regional Languages, page 476).

<sign> <sign2>... Apart from the call signature specified in the output format, the signature can also be extended. <sign> is a word consisting of letters, digits and "_". Several additions should be separated from each other by spaces. For more information on the signature in general can be found in 19.1.2 The Signature of Calls and the Overlapping of Input Masks, page 472.

What is the purpose of the signature addition? The unique assignment of input masks to a program text via the signature is the basis for mask overlapping . Sometimes, it may be desired, that several input masks with the same signature have to be parallely kept. This is used

- if several masks with different parameters are to be declared to take different cases into account, but where the call names are identical (e.g. different masks with G1 for two-contour and three-contour lines).
- for masks without signature. In case of cycles without signature, the alphabetical order is solely defined by the signature addition.

The signature addition is subordinated to the call signature. Therefore, the alphabetical sorting via the additional signature can only be changed to a minimum extent for masks with signature.

☞ Use only character strings not allowed as DIN words as additional signatures. Combine the signature with "_".

Example:

Title

```
//%NAME% G81 Twist drilling
```

monolingual title

```
//%NAME% %TN0678%
```

multilingual title with token specification

```
//%NAME% G1 - Three-point cycle %SIGN% 3_PNT_DRAFT
```

Signature addition

Properties of an Input Mask (Grouping)

Objective Each input mask can be provided with properties used by the system when creating the list of the masks available for insertion (menu scope). The specification of one and the same property word in several masks declares all of these masks to be elements of a specific subset, the group.

properties are differentiated allowing a grouping with regard to

- thematic (contentual) or technological point of view,
- origin (user, manufacturer, standard cycle or software option) and
- availability of the input mask when inserting and correction

The property on the origin is automatically specified by the system and does not have to be programmed.

Syntax

```
//%GROUP% <property>{ <property2>{ <property...>}}
```

including:

<property> <property2>... Word consisting of letters, digits and "_", the group name standing for a specific property. According to the table below, there are multiple pre-defined group names in the system. The machine manufacturer and the user can introduce any number of new group names using %GROUP% instructions that are always understood as name of a thematic/technological group. Separation of several properties by spaces.

Property table **List of predefined properties**

Group name	Automatic assignment	Property
CANNED	Yes, when reading from IndraWorks \config\cycles\MTX.Cycles.Canned.cyc	Origin "Standard cycle" (Bosch Rexroth)
DIN	No	Topical grouping "DIN instructions" (masks supporting DIN programming do not represent cycle calls)
DRILL	No	Grouping for "drilling" technology
GNP	Yes, when reading from IndraWorks \config\cycles\MTX.Cycles.GnpMill.cyc and MTX.GnpTurn.cyc	Assignment to the software options of the Graphic NC Programming. Type code SWS-MTX***-RUN-NNVRS-D0-BAZ1 or SWS-MTX***-RUN-NNVRS-D0-TURN1
GRAVE	No	Grouping for "engraving" technology

Group name	Automatic assignment	Property
`MILL`	No	Grouping for "milling" technology (except engraving and contour milling)
`MILLCONT`	No	technology grouping for "contour milling"
`OEM`	Yes, when reading from the project folder <VisualizationDevice>\OEM \config\cycles	Origin of "Machine manufacturer"
`PATT`	No	Thematic grouping for "bore pattern"
`PROBE`	No	Thematic grouping for "measurement"
`SAMPLE`	No	Thematic grouping for samples
`SIM`	Yes, when reading from IndraWorks \config\cycles\MTX.Cycles.Simulation.cyc	Input masks for functional simulation comments belonging to the NC simulation software options.
`TURN`	No	Grouping for "turning" technology (except contour turning)
`TURNCONT`	No	technology grouping for "contour turning"
`USER`	Yes, when reading from the project folder <VisualizationDevice>user \config\cycles	Origin of "User"
`_HIDDEN`	No	Hides an input mask. This mask is neither available for inserting nor deleting.
`_NOCORR`	No	Hides a mask for correction. This mask is only available for inserting.
`_NOINSERT`	No	Hides a mask for inserting. This mask is only available for correction. (*)
`_NOGROUP`	Yes	This group gets each input mask automatically if it does not belong to any contentual/technological group Never include in a %GROUP% instruction.
`_OTHERGROUPS`	No	This group name represents all groups not explicitly controlled via an F-key in the input support. This group name is used as filter for the function key "More cycles...". Never include in a %GROUP% instruction.

(*) Assign this property only to all masks for which there are newer calls with other masks (e.g. for syntax change). This ensures that all newly inserted calls are based on the new pattern and the old masks are still available for correction.

Fig.19-1: List of predefined properties

Example:

Properties (Grouping)

The properties of a user cycle for drilling should be defined.

```
//%GROUP% DRILL
```

As the cyc-file is saved in the <VisualizationDevice>\user\config\cycles project folder, the property "USER" for "User cycle" is automatically assigned to the mask.

Example:

Hiding the input mask

A machine manufacturer likes to hide G111 Hole circle rotary axis, since the machine is not provided with rotary axes. Therefore, the cycle mask of the standard cycle has to be hidden.

```
//%CHBEGIN%
//%GROUP% _HIDDEN
//%N% G111(%P%)
//%CHEND%
```

The machine manufacturer adds this mask definition to a cyc file store in the project folder <VisualizationDevice>\OEM\config\cycles.

19.3.4 Global Mask Elements

Graphics

Global mask elements are graphic dialog elements generally displayed as long as the assigned input mask is visible. Only the global graphics can be temporarily overlapped by a parameter-specific graphic.

Objective The graphic has particular significance for the mask design, as it is a key to explain the function of a cycle and the parameters. In principle, it is not possible to incorporate long explanatory texts into an input mask. Therefore, all of the relevant information possible has to be provided graphically.

Multiple global graphics It is possible to assign multiple global graphics to an input mask. Use this option to display different machining cases (e.g. internal and external drilling) or dimensioning variants separately and thus more clearly.

☞ The display of graphics is provided independently of the language set. For this reason, texts should be avoided in the graphic, except the parameter names and possibly an explanatory program code.

Syntax

```
//%GF% <graphicfile>{ <graphictitle>}
```

including:

<graphicfile> Name of a graphic file without path including the extension ".bmp", ".jpg", ".gif" or ".tif". Size of the images 223 x 262 pixels (width x height) . File names with spaces should be enclosed by inverted commas.

<graphictitle> If several graphics are displayed, a title has to be assigned to each graphic. This title is displayed or offered for selection in the list above the graphic. The title includes the entire text after <graphicfile> up to the end of the line. To support multilingualism, a token can also be specified ("%TNxxxx%", see Support of Multiple National and Regional Languages, page 476).

If only one graphic is provided for a mask, it is recommended to omit the graphic title.

Example:

Global graphics

Two global graphics should be assigned to one cycle - one graphic for the internal machining and one graphic for the external machining.

```
//%GF% TurnOutside.gif external machining
//%GF% "Turn Inside.gif" %TN0346%
```

Explanatory Texts

Explanatory texts can be displayed in the lower section of the input mask.

Objective The explanations provide an overview on the syntax and on the effect of the cycle. Special features should also be described there, such as for the default parameters.

Syntax `//%C% <explanation>`

including:

<explanation> Content of an explanatory line or a token. A monolingual text starts at the first character different to the space following //%C% and extends to the end of the line. Thus, no indentation can be programmed. A language-dependent text referenced via token can also comprise several lines and indentations (also refer to Support of Multiple National and Regional Languages, page 476).

Example:

Explanatory texts

```
//%C% %TN0672%
```

```
//%C% default for parameter <ax>: Drilling axis for cur-
rent interpolation plane
```

Online Help

Help file Each input mask can be assigned to one help file only (preferably HTML help). The help can be provided in several language variants. For information on storage location and support of several language, see Help Files in Several Languages, page 478.

Syntax `//%HELPFILE% <helpfile>`

including:

<helpfile> Name of a help file including extension and excluding path specification. The file name extends to the end of the line so that no inverted commas are required for spaces in the name.

Help token A help token can optionally be specified for each input mask. A help token is the reference to a specific section in the online help.

Syntax `//%HELPTOKEN% <helptoken>`

including:

<helptoken> Identifier of a reference to a specific section in the help file. This can either be a TopicID such as "G86Boring.htm" or a keyword (search term). If your help is translated into other languages, note that the selected keyword is identical for all languages and thus not translatable.

Example:

Online help

The help file "MyCycles.chm" should be displayed in an input mask. The respective section should be reachable via the keyword "CYCLE2".

```
//%HELPFILE% MyCycles.chm
```

```
//%HELPTOKEN% CYCLE2
```

19.3.5 Default Parameter Types

Syntax

The following describes the parameter declarations as to be found in almost every input mask. There are also special types subject to extra syntax and therefore documented separately.

Syntax

```
//%P%{%@%<pname>}{ <type>{ [<range>]}}<ptitle>
{ %D%{<behavior>}<default>}{ %GF%<graphicfile>{ <graphic-
title>}}
{ %VALID%<validinfo>}
```

The syntax is explained in the following sections. Note that each parameter description has to be written into one line even if the syntax appears across several lines due to space reasons.

- Define the sequence of the parameter definition based on a reasonable procedure and not on the sequence of the position parameters. Parameters directly related or dependent on each other are defined in succession.
- The method of defining parameters up to now using "//%V%" is continuously supported for reasons of compatibility.

Parameter Address and Title

<ptitle>

A title is assigned to each parameter. This title must include the parameter description in plain text along with any possible supplementary information on the unit of measurement etc. The parameter title can specify multiple languages as token for support purposes ("%TNxxxx%", see Support of Multiple National and Regional Languages, page 476). No percentage sign can be used for a monolingual text. If you nevertheless need it, for example as a unit of measurement, enclose it with round or square brackets.

A constant title text extends to the end of the line or to the next percentage sign outside a bracket.

<pname>

If the parameter is provided with an address, the address name should preferably be given directly behind "//%P%". The first word after "%@%" is accepted as address name. The address name is displayed in the first column of the input table, otherwise, a continuous index appears.

Address names may only consist of letters, digits and "_", but not of special characters. If NC notes are to be programmed via an input mask (character string parameter type, see also Character String STRING, page 487, the following syntax variants can only be used:

- Channel-specific note (`MSG<NoteText>`):

  Address name "MSG" or
- Channel-comprehensive note (`GMSG<NoteText>`):

  Address name "GMSG".

For NC notes, refer to "Rexroth IndraMotion MTX Programming Manual", chapter "Notes in the User Interface"

Variable address name (GNP)

If an output format begins with a parameter of the special type "DCS", (Description Coordinate System from Graphic NC Programming, which defines the coordinate axes active in the cycle) the meanings of the axis names specified in the DCS can be used again in the subsequent parameters. The axis names become variable. That is achieved by the following placeholders instead of the constant address name:

%A1%

Name of the master axis of the plane (abscissa).

%A2%

Name of the slave axis of the plane (ordinate).

%A3%

Name of the drilling axis of the plane (applicate).

For more detailed information on the description coordinate system, refer to the "Rexroth IndraMotion MTX Programming Manual ", chapter "Geometry Definitions".

Example:

Parameter address and title

Position parameters with monolingual title

```
//%P% Contact width [%]
```

The coordinates of the first DCS master axis with a multilingual title should be programmed as reference coordinate of a geometry definition.

```
//%P% %@%%A1% %TN3742%
```

Parameter Types and Value Ranges

A data type **<type>** can be optionally assigned to each parameter. The specification of a data type can - and has to for the enumeration type - follow the specification of a value range `<range>` in square brackets.

The data type should follow directly after the address name or after "//%P%".

The following data types are possible:

Real number REAL

The "real number" parameter type is programmed using "REAL". This is also the default data type if a parameter declaration does not include any type specification.

The optional value range specification **<range>** is composed of a lower value range limit followed by two points and the upper value range limit.

☞ The value range limits are always within the value range. If a value range limit is not within the range, switch to a fraction number close to the real range limit. If the value range limit is an integer, enter as limitation a value either 0.01, 0.001 or 0.0001 smaller or greater than the integer already outside the value range.

If the lower or the upper value range limit is not required (value range limited on one side), change to a very small or very large number. The lower value range limit is canceled out with -999, -9999 or -99999, the upper one with 999, 9999 or 99999.

Example:

Parameters of type "REAL"

A real parameter ANG should have the value range 0<=ANG<=360.

```
//%P% %@%ANG REAL[0..360] angle [deg]
```

A real position parameter should be greater than 0.

```
//%P% REAL[0.0001..99999] distance
```

A real parameter A should move in the range 90<A<180.

```
//%P% REAL [90.001..179.999] angle [deg]
```

Integer INT

For the "integer" parameter type, "**INT**" is to be specified.

The optional value range specification **<range>** is composed of a lower value range limit followed by two points and the upper value range limit.

Example:

Parameters of type "INT"

An integer parameter TEC should be provided with the value range 0<=TEC<=9.

```
//%P% %@%TEC INT[0..9] Edge position
```

An integer number position parameter should be greater than 0.

```
//%P% INT[1..9999] Number of bores
```

Binary type BOOL

The "binary type" represents a logic statement which can only accept two values:

"FALSE" or "TRUE".

This parameter type should be declared with "`BOOL`". It can only be used in Inline CPL parameter lists (parameter list in square brackets) and accepts the values "TRUE" and "FALSE" according to CPL syntax.

The options value range specification **<range>** is used to display plain text for the meanings "FALSE" and "TRUE". First, the text is given for "FALSE" and then, separated by a comma, for "TRUE". The texts can be defined as token or monolingual text, although no comma and no "]" are permitted for the latter.

Without any range specification, "TRUE" and "FALSE" appear as possible values in the input table.

Example:

Parameters of type "BOOL"

For binary parameters, the text should appear "on the right" in case of "TRUE" and "on the left" in case of "FALSE".

```
//%P% BOOL[let, right] direction of rotation
```

The same in the multilingual version:

```
//%P% BOOL[%TN0560%, %TN0561%] %TN0562%
```

ENUM enumeration type

If a finite number of words, numbers or strings belong to the value range of a parameter, the "`ENUM`" enumeration type is often used. In this case, each element of the value range is defined separately including an assigned plain text. The value should be given first for each element followed by a plain text separated by a space. The plain text can be declared by a token or can be monolingual, although no comma and no "]" may be included in the text. A comma is the separator between two elements.

- If the enumeration contains integers (see Integer INT, page 486), consider whether the type "INT" is not more suitable, because the editing takes place in a selection list which is harder to operate without a mouse than a simple text field.

  The following criteria are in favor of ENUM:

  – The value range has interruptions

    – or –

  – the values are not assigned to memorable meanings (e.g. machining variants).

  The following criteria are in favor of INT:

  – The value range has no gaps and

  – the value meaning can be represented more graphically than with a short plain text.

- The "ENUM" type is the only data type permissible for parameters without address that appear outside a parameter list. In this case, each value has to start with a letter.
- The type "ENUM" may - as an exception - only be provided with one single element in the value range. Use this parameter definition if multiple input masks should be provided for one cycle. By specifying only one single element in the value range, the input mask is assigned to this parameter value. In this case, the respective parameter is not displayed in the input table.

Example:

Parameters of type "ENUM"

A position parameter (Inline CPL) has the character string type, whereby only two values, "CCW" and "CW" are accepted.

```
//%P% ENUM["CCW" counterclockwise, "CW" clockwise] Direc-
tion of rotation
```

A parameter without an address outside a parameter list should be used to program the spindle control commands M3, M4, M5 and M19.

```
//%P% ENUM[M3 right, M4 left, M5 stop, M19 orientate]
Spindle command
```

A position parameter should accept the values 0, 1, 10 and 11 (similar to the binary representation of a number). It is not possible to provide the meaning as short plain text. Thus, the individual values are illustrated in a graphic. The element value is therefore repeated in plain text.

```
//%P% ENUM[0 0, 1 1, 10 10, 11 11] Variant
```

A parameter with the address D is to be assigned with the abbreviations for weekdays (two characters) The parameter is provided multilingually.

```
//%P% %@%D [Mo %TN0501%, Tu %TN0502%, We %TN0503%, Th
%TN0504%, Fr %TN0505%, Sa %TN0506%, Su %TN0507%] %TN0500%
```

Character String STRING

Character stings are declared using the type specification "`STRING`".

The range specification `<range>` is always omitted in case of character strings.

Example:

Parameters of type "STRING"

A position parameter should accept a character string.

```
//%P% STRING engraving text
```

Default Values and Pre-assignment

Optional pre-assignments can be implemented for each parameter when inserting a call or a default parameter behavior. The syntax elements that follow "%D%" <behavior> and <default> are therefore provided.

Pre-assignment

The pre-assignment enters an initialization value when inserting a call and before the mask dialog becomes visible. Pre-assignment is not important for corrections.

For the pre-assignment, the initialization value is only programmed at the position of <default> (not the substitute representation in plain text for the types and BOOL!). Only the first word after %D% is accepted as initialization value. Enclose the character string with inverted commas.

Use the pre-assignment option if the same value is used frequently, particularly if this is an obligatory parameter.

Example:

Parameter pre-assignment

The direction of rotation is nearly always programmed on the right with a spindle control command.

```
//%P% [M3 right, M4 left, M5 stop, M19 orientate] Spindle
command %D%M3
```

Default parameters

A "default parameter" is an optional cycle parameter, for which - if it is not programmed - a specific value from the value range of the parameter is used program-internally (e.g. 60 ° thread edge angle if the parameter is not assigned). This differs from a similar case, in which an unassigned parameter leads to the omission of a certain function or process (e.g. no chip breakage). The latter is not a default parameter, because in this case the default is "unassigned" and applies to each parameter.

For default parameters, "unassigned" stands for a value. This value can constantly, depending on options or via the context of the subroutine call, be changed (depending on the previously activated interpolation plane).

You cannot convert default parameters - depending on the call context - in the input masks. Such a condition could, for example, be the interpolation plane activated before the cycle. Document the behavior in an explanatory text (see Explanatory Texts, page 483).

Constant default parameters

If the default value is constant, implement the default parameter behavior into the input mask. Enter "=" instead of <behavior>. "=" is followed by the default value (<default>).

With an unassigned default parameter, the default value always appears in the input mask when inserting and correcting. When transferring a call to the NC program, the respective parameter is omitted if its value corresponds to the default value.

Example:

Default parameters

In a thread turning cycle, the edge angle is to be transferred via a parameter. The default value of this parameter is 60°.

```
//%P% %@%FA REAL[0..179.9999] %TN7321% %D%=60
```

Option-dependent default parameters (GNP)

There is an extension contrary to the aforementioned default parameters for input masks of the Graphic NC Programming. The default parameter is not a constant, but it is taken from a current "Option" setting (see "Rexroth IndraMotion MTX Workshop Programming Turning and Milling", chapter "GNP Options"). This Options setting is accessed via placeholders which are formed from the address (enclosed in percentage symbols) of a value from the file in project folder <VisualizationDevice>\user\ config\MTXGnpConfig_000_<channel>.ini, Section "[Tech_Settings]". The structuring of this file is documented in the Manual "Rexroth IndraMotion MTX Workshop Programming Turning and Milling", chapter "Options and User Settings".

Example:

Option-dependent default parameters

An address parameter AT should be provided with the default value set in the GNP options for the tangential approach length when turning.

```
//%P% %@%AT REAL[0..9999] %TN7327% %D%=%Siabst_Tan%
```

It should be assigned to the program in channel 1. The file MTXGnpConfig_000_001.ini contains:

```
[Tech_Settings]
..
Siabst_Tan = 2,
```

The default value of the parameter is 2.

Option-dependent default parameters with output condition (GNP)

Option-dependent default parameters conceal the risk that another program sequence results from modification of the options or after program transfer to another machine. For this reason, it is not desired that a parameter with the default value set in the "Options", is transferred to the source code. Whether the default value is transferred can also be set using the "GNP Options" (see "Rexroth IndraMotion MTX Workshop Programming Turning and Milling", section "Tabs - General" in the chapter "GNP Options"). The following settings in file <VisualizationDevice>\user\config\ MTXGnpConfig_000_000.ini, Section "[View]", control the output (see also "Rexroth IndraMotion MTX Workshop Programming Turning and Milling", chapter "Options and User Settings"):

- `"ForceAT"`

  Responsible for the approach when turning,
- `"ForceCV"`

  Responsible for the cutting variant when turning,
- `"ForceTR"`

  Responsible for the edge radius when turning and
- `"ForceTRV"`

  Responsible for the tool retract movement at the starting point before the cycle.

If the value stored there is "1", each value is transferred to the NC program. If the value is "0", normal behavior applies, like an option-dependent default parameter. Thus, the parameter appears only if it is different to the default parameter.

To formulate the output condition with an option-dependent default parameter, enter "!<placeholder_outputCondition>" for <behavior> and the placeholder "<placeholder_tech_setting>" for `default`. Each placeholder should be enclosed by percentage signs.

Example:

Option-dependent default parameter with output condition

An address parameter TRV should control the retract movement to the starting point of a cycle. The default parameter should be taken from the "GNP Options", value address "Tool_Rev_T_Out". If the parameter value corresponds to the default value, it should be transferred to the NC program depending on the options (output condition "ForceTRV").

```
//%P% %@%TRV [0 without, 1 inclined, 2 X-Z, 3 Z-X] Retract
movement %D%!%ForceTRV%%Tool_Rev_T_Out%
```

Only if in the file MTXGnpConfig_000_000.ini

```
[View]
..
ForceTRV = 1
```

is the parameter transferred to the NC program without conditions.

Parameter-related Graphics

Using an addition initiated by "`%GF%`", every parameter in an input mask can have its own graphic assigned to it, which temporarily hides the global graphic Global mask elements, page 482). A parameter-related graphic is recommended if the effect of a parameter cannot be clearly represented in one or more global graphics.

<graphicfile> Name of a graphic file without path including the extension ".bmp", ".jpg", ".gif" or ".tif". Size of the images 223 x 262 pixels (width x height) . File names with spaces should be enclosed by inverted commas.

<graphictitle> As an input mask automatically owns several graphics due to a parameter-related graphic, a title should be assigned to each graphic. These titles are displayed in the selection list above the graphic. The title includes the entire text following <graphicfile> to the end of the line or to the next reserved word enclosed by a percentage sign. To support multilingualism, a token can also be specified ("`%TNxxxx%`", see 19.2.4 Support of Multiple National and Regional Languages, page 476).

☞ Behavior towards parameter-related graphics has changed towards an incompatible way. From version MTX 10VRS onwards, there is now an actual overlapping of the global graphic by the parameter-related graphic, whereas it was necessary up to now to activate the global graphic again in the following parameter using %GF%.

Adapt your input masks from older versions accordingly.

Example:

Parameter-related graphics

A cycle consists of two global graphics and a parameter-related graphic (to the 2nd parameter).

```
//%P% Parameter 1
//%P% Parameter 2 %GF%GraphPar3.gif Effect of parameter 2
//%P% Parameter 3
```

```
//%GF% GraphGlob1.gif 1st variant
//%GF% GraphGlob1.gif 2nd variant
```

Validation Information

<validinfo>

The validation of parameter values is defined irrespective of other parameters by specifying a parameter type (see 19.3.5 Standard Parameter Types, page 484). This section shows how to provide further information (initiated via "`%VALID%`") which describes dependencies between individual parameters and how to declare a parameter to an obligatory parameter.

The validation information can trigger system-internal tests and lead to desirable, error-avoiding input obligations. Other relations are simply visualized graphically to illustrate a programming rule about one or more parameters to the programmer at a glance.

> Only simple standard cases can be dealt with using the validation information. You have to represent more complex relations between individual parameters in an explanatory text.

Obligatory parameters

The declaration as obligatory parameter initially results in the parameter being labeled in the input table using the symbol . There is no test when transferring a call to ascertain whether an obligatory parameter has actually been filled out.

From the functional side, it is only achieved that when an unassigned obligatory parameter occurs in an output line, this line is inserted, even if no single parameter in this line is assigned. A line without an assigned parameter would be omitted without obligatory parameter labeling. If this is an obligatory parameter in a parameter list, the call name and the brackets around the parameter list also appear. With position parameters, empty parameters are optionally filled so that the parameter list has the corresponding length for the subsequent acceptance of all obligatory parameters.

Instead of `<validinfo>` "`MAND`" is programmed for an obligatory parameter.

Example:

Validation information of obligatory parameters

The parameter DT is an obligatory parameter.

```
//%P% %@%DT %TN2266% %VALID% MAND
```

Parameter pair

There are optional parameters which can only be programmed with another or with several other parameters. Parameters to be programmed in pairs can be identified by a symbolic string in the input mask connecting the affected parameters to each other. It is therefore necessary to define all parameters of a pair amongst each other.

Instead of `<validinfo>` "`PAIR`" is written in a parameter pair, to which a list of placeholders is connected indicating other parameters of the pair. These placeholders are provided in the form `%P<pno>%`, whereby <pno> is the continuous parameter number from top to bottom beginning with 1. When defining a pair, it is freely selectable on which parameter of the pair the references to the parameters belonging to the pair are entered.

Example:

Validation information of parameter pairs

A cycle has 6 parameters. The parameters 2 to 5 are optional and should transfer the 4 corner coordinates of a machining window.

```
//%P% %@%ID INT[1..999] %TN0711%
```

```
//%P% %@%WHT %TN0712% %VALID% PAIR%P3%%P5%
//%P% %@%WHD %TN0713%
//%P% %@%WVL %TN0714%
//%P% %@%WVR %TN0715% %VALID% PAIR%P4%
//%P% %@%F REAL[0.001..99999] %TN0716%
```

In the example above, the PAIR entries have been consciously incorporated in different parameters in order to show several options for a pair definition.

Param.	Comment	Value
ID	Ident of contour	1
WHT	Machining window outer border	50
WHD	Machining window inner border	20
WVL	Machining window left border	-60
WVR	Machining window right border	-25
F	Feed	

Fig. 19-2: Exemplary layout of the input table for the parameter pair

Alternative parameters

Many parameters exclude each other and cannot be programmed at the same time. This situation occurs occasionally with optional parameters, but is also possible with obligatory parameters. Within the input masks, only the case is supported that ensures at most one parameter from a number of parameters as high as possible can be assigned. Other conditions, for example two parameters out of three, are not met. The input table prevents the assignment of a second parameter from the group of alternative parameters by disabling all other input fields if a parameter has been provided with a value. While correcting, a multiple assignment is recognized and a corresponding error message is displayed. No additional symbols are displayed.

Instead of `<validinfo>` "ALT" is written for alternatively used parameters, to which a list of placeholders follows indicating the alternative parameters. These placeholders are provided in the form %P<pno>%, whereby <pno> is the continuous parameter number from top to bottom beginning with 1. When defining alternative parameters, it is also not important on which parameters of the alternative group the references to the other associated parameters are entered.

Example:

Alternative parameters

There is a turning cycle for which chip breakage can optionally be programmed. If chip breakage is to be activated, the distance CBD should be specified after which the chip is to be broken. In addition, either a dwell time DWT or the number of dwell revolutions should be programmed (combination of parameters programmed in pairs and alternatively).

```
//%P% %@%ZS %TN0801%
//%P% %@%ZE %TN0802%
//%P% %@%CBD REAL[0.0001..9999] %TN0803% %VALID% PAIR%P4%
%P5%
//%P% %@%DWT REAL[0.0001..9999] %TN0804% %VALID%ALT%P5%
//%P% %@%DWR INT[1..9999] %TN0805%
```

Param.	Comment	Value
ZS	Start coordinate	-20
ZE	End coordinate	-30
CBD	Distance per chip break	8
DWT	Dwell time during chip break	
DWR	Dwell revolutions during chip break	2

Fig. 19-3: Exemplary layout of the input table for the alternative parameters

19.3.6 Special Parameter Types

Description Coordinate System DCS

For the Graphic NC Programming, the parameter type Description Coordinate System (DCS) has been introduced. For more detailed information on the description coordinate system, refer to the "Rexroth IndraMotion MTX Programming Manual ", chapter "Geometry Definitions".

This parameter type never provided with an address, permits one of the planes declared in the GNP Options to be selected for further programming or a new plane to be specified in DIN syntax. The DCS type implements the necessary default parameters and validation behavior. If this parameter type is used, the variable parameter names can be used in this mask for coordinate axes (%A1%, %A2% and %A3%, (see chapter19.2.4 Multiple National and Regional Languages, page 476) which depend on the DCS selected. The parameter type DCS cannot yet be used for the programming of an unwinding.

Syntax

```
//%DCS% <ptitle> {%GF% <graphicfile>{ <graphictitle>}}
```

including:

<ptitle> A parameter title should be assigned to the DCS. The title can be specified as token ("%TNxxxx%") or as monolingual text.

A monolingual title text extends to the end of the line or to the next percentage sign outside a bracket.

<graphicfile> Name of a graphic file without path including the extension ".bmp", ".jpg", ".gif" or ".tif". Size of the images 223 x 262 pixels (width x height) . File names with spaces should be enclosed by inverted commas.

<graphictitle> As several graphics are "automatically possible" with a parameter-related graphic, a title should be assigned to each graphic. This title is displayed in the selection list above the graphic. The title includes the entire text following <graphicfile> to the end of the line or to the next reserved word enclosed by a percentage sign. To support multilingualism, a token can also be specified ("%TNxxxx%", see chapter 19.2.4 Support of Multiple National and Regional Languages, page 476).

Example:

Use of the DCS parameter type

An input mask is to be defined for the GNP geometry definition "Point Pattern on Part Circle" (PCI).

```
//%P% %@%ID INT[1..999] GeometryIdent %VALID% MAND
//%DCS% Description Coordinate System
//%P% %@%%A1% Center point of master axis %VALID% MAND
//%P% %@%%A2% Center point of slave axis %VALID% MAND
//%P% %@%R REAL[0.0001..9999] Radius %VALID% MAND
//%P% %@%NR INT[1..999] Number of points %VALID% MAND
```

```
//%P% %@%SA Angle of first point
//%P% %@%EA Angle of last point %VALID% ALT%P9%
//%P% %@%A Segmentation angle
```

Parameter with Variable Binding

In MTX 08VRS and below, a special parameter type has been supported, with which only CPL variables could be specified instead of a constant value. This data type which definition started with "//%VAR%"can no longer be used.

Replace this type in older input masks by a default type. To assign a value via a CPL variable or a CPL expression, enclose the expression/variable with square brackets in the input field as it is done for Inline CPL.

19.3.7 Output Format

Using the "output format", the part of the syntax of the call belonging to the input mask is defined. It incorporates more than the collection of the individual parameters. The output format includes the compilation of the parameters and their combination with other unchangeable block components and comments.

Objective The output format defines the syntax of a call, which, during correction, also permits the unique input mask identification which suits a piece of program text best. The format forms the link between the NC program and the input mask.

An output format can contain any number of characters.

Syntax `//%N% {<format>} {<comment>}`

including:

<format> The format of the call(s) is stored under `<format>` in the form in which it is ultimately to appear in the NC program. Enter the call names and the brackets enclosing the parameter lists. Unchangeable block components appear in their final form. Placeholders are provided for the parameters:

"%P%" stands for all parameters appearing in the order of their definition (see also chapter 19.3.5 Standard Parameter Types, page 484).

If the parameters are distributed across several parameter lists or if position parameters require a different order than given in the parameter definition, access each parameter individually. Placeholders in the form of "`%P<pno>%`" are again used, whereby <pno> is the continuous parameter number from top to bottom, beginning with 1.

No comma has to be inserted between the placeholders in a parameter list. The commas are automatically inserted. The insertion of spaces cannot be controlled by the specification of the format.

For simple and therefore single line cycle calls (see chapter19.1.3 Advanced Systematization of Calls, place 473), the specification of the brackets enclosing the parameter list and the placeholders can be omitted in this exceptional case. Only the call name has to be specified. For parameters without address, a parameter list in square brackets and for parameters with address, in round brackets follows automatically.

<comment> The comments initiated with a semicolon can either directly be specified or as a separate line after the call. In monolingual texts, the semicolon is written first and then the comment. If multiple languages should be supported, a semicolon is not to be specified, but only a token as follows `%TNxxxx%` (see chapter 19.2.4 Support of Multiple National and Regional Languages, page 476). Comments in round brackets are not supported except NC notes.

- If comment lines are to be added to multiline calls (via the output format or by subsequent editing), it should be noted that the number of comment lines between two DIN lines is limited to 4. Once this limit has been exceeded, not all of the lines belonging to the format can be found while correcting. This can result in parameter values being missing in the input table and double assignments when they are transferred to the program.
- Comments supported by multilingualism are only generated when inserting. During correction, comments remain untouched. There is also no translation into another language if a language setting is active that is different from the case during insertion when correcting the call.

Example:

Output formats

A cycle with position parameters is to be programmed. A monolingual comment is to be added to the call.

```
//%N% G84[%P%] ;Thread drilling
```

A cycle with address parameters is to be programmed. A multilingual comment is to be added to the call.

```
//%N% G722(%P%) %TN0722%
```

A functional comment to describe a tool for simulation is to be written. The definition sequence of the parameters is to be different to the sequence in the parameter list. At the end of the line, an unchangeable block component is to be added as Inline CPL.

```
//%N%            //TOOL/STANDARD,%P2%%P3%%P4%%P5%%P1%\\
[""+NCF("G90")]
```

A complete tool change (travel to the tool change point, tool change, a comment, an NC note and a master block) is to be provided for a lathe. The machine is provided with two spindles.

```
//%CHBEGIN%
//%NAME% Tool change
//%P% %@%X Tool change point X (diameter) %D%200
//%P% %@%Z Tool change point Z %D%250
//%P% ENUM[M105 Spindle 1, M205 Spindle 2] Spindle stop
before tool change
//%P% INT[1..12] T-number %VALID% MAND
//%P% %@%MSG STRING NC note
//%P% ENUM[DIA Diameter, RAD Radius] X-scaling %D%DIA
//%P% ENUM[G17 G17, G18 G18, G19 G19] Interpolation plane
%D%G18
//%P% %@%G REAL[54..59.6] Zero point displacement %D%54.1
//%P% ENUM[G97 Speed, G96 Cutting velocity] Spindle pro-
gramming constant ... %D%G97
```

```
//%P% %@%S1 REAL[0.001..99999] 1. Spindle value %VALID%
PAIR%P11%
//%P% ENUM[M103 right, M113 right KM, M104 left, M114 left
KM] 1st spindle command %D%M114
//%P% %@%S2 REAL[0.001..99999] 2nd spindle value %VALID%
PAIR%P13%
//%P% ENUM[M203 right, M213 right KM, M204 left, M214 left
KM] 2nd spindle command
//%N% G0 DIA G53 G90 G48 %P1% %P2% %P3%
//%N% ;------------ Tool change -----------
//%N% T[%P4%] M6
//%N% (%P5%)
//%N% %P6%%P7%G47%P8%%P9%%P10%%P11%%P12%%P13%
//%CHEND%
```

20 Setting up NC Cycles

20.1 Input Masks

To facilitate the programming of cycle calls, the input support of the editor is often used. Input masks have to be defined for cycles requiring this support. This is for the interface configuration and is not related to the function of cycles during program processing.

For the creation of masks, refer to Definition of Input Dialogs, page 471.

20.2 Parameter Settings for Cycle Calls

The non-modal standard cycles have already been entered. Non-modal user cycles can be added here.

The modal standard cycles have already been entered. Modal user cycles can be added here.

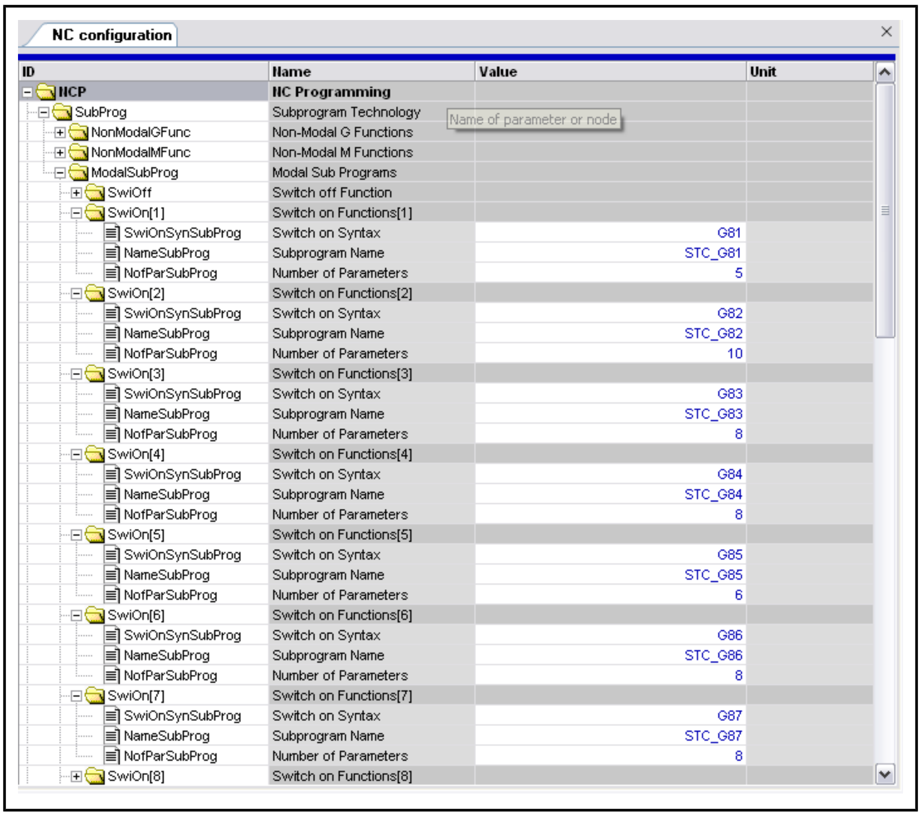


Fig.20-1: Editing IndraWorks Engineering parameters

20.3 Subroutines

The subroutines (cycles) of the machine manufacturer are saved in the control file system under "root\usr\mtb\cycles".

The subroutines (cycles) of the end user are saved in the file system of the control under "root\usr\user\cycles".

20.4 SD Variables

Some cycles can optionally operate with permanent channel-dependent SD variables. The assignment is listed in the respective cycle description. The SD variables used for the standard cycles are available. If user cycles should be created and implemented this way, the SD variables used have to be defined.

20.5 Usage of Existing Projects

This section relates only to projects created with versions < MTX09V06.

There is not complete compatibility. Therefore, the user has to decide whether to continue using the existing projects in their "old" form or whether to work with standard cycles in future. **A combination of these two variants should not be used!**

The usage of individual cycles from an old version does not require any further measures. However, it is not be possible to access the installed standard cycles.

The following modifications are required to use the standard cycles in existing projects (project version >=MTC09V06):

- Adapt the search path for the cycles in the machine parameters. "NC Optimization (NCO) -- FileOrg -- SrchPathSubProg". Add a new entry "/feprom/cycles" in front of the entry "/feprom".

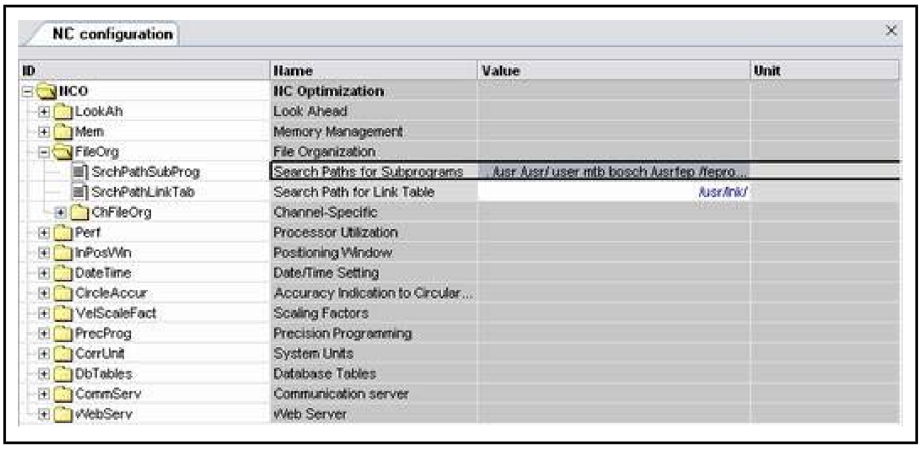


Fig.20-2: Extending the search path

- Make standard cycle entries in the machine parameters. "NC Programming(NCP) -- SubProg -- ModalSubProg -- SwiOn[n]". See chapter20.2 Parameter Settings for Cycle Calls, page 497. The required entries can be taken from the default values.

If, after conversion to the standard cycles, the existing user cycles have to be used as well, these should either be adapted to the addressed notation or their names should be changed in such a way that they match the intended range for user cycles. In doing so, cycle subroutines, cycle headers and definitions in the machine data also have to be adapted. It should also be noted that the cycle calls in the NC programs also have to be adapted to these changes.

21 Virtual Commissioning of the MTX

21.1 Installation of the MTX Emulation

21.1.1 General

Description

Brief Description The MTX Emulation is installed from the installation CD by following the relevant dialogs of the installation program. It is recommended that you accept the default values.

21.2 Configuring the MTX Emulation

21.2.1 Restoring an Existing Project

General

Brief Description With IndraWorks Engineering, existing projects can be restored which have been created earlier or on a different computer for real systems or the MTX Emulation.

Description In the following, you will find a description of how to transfer (=restore) an IndraWorks project from a real control to the MTX Emulation. Among other things, the IndraWorks project storage contains the control parameters, the Profibus configuration, the PLC program, the definition files for the M and F-keys, the logbooks and the user-defined screens.

Restoration of the project is carried out by using the function "Restore" in the menu "Project".

Handling Instruction: Restoration of an Existing Project for MTX Emulation

In the following, you will find a description of how to transfer (=restore) an IndraWorks project from a real control to the MTX Emulation.

IW-Engineering / Motion: Restore control data

Figure	Flowchart	Example	Instruction	Documentation
Instruction:			Restore data	
Documentation:	IndraWorks commissioning instructions		Data backup	

Fig.21-1: Link

21.3 Configuring the HMI

21.3.1 General

Description

Brief Description The configured visualization data become effective in the user interface only if they are transferred and activated and the operating station.

Handling Instruction: Configuring the HMI for MTX Emulation

It is described how the visualization data of the HMI is transferred and activated on the operating station.

IW Engineering / Project: Transferring and activating visualization data

- Select in the context menu of the visualization device the menu item **Visualization data ▸ Transfer and activate...** in IndraWorks Engineering.
- The dialog "Transferring and activating visualization data" opens. With <OK>, the visualization data is transferred and activated on the operating station.
- The visualization data is available at next start of the MTX user interface.

21.4 Configuring the NC Kernel

21.4.1 General

Description

Brief Description

Among other things, the NC kernel data contains the the usrfep and the root files, the machine data as well as the database tables for the system and tool data. The data of the NC kernel is located in the IndraWorks project tree below device node "IndraMotion MTX P60".

To restore the NC kernel data of a project, which has been created on a real control, you first have to adapt the "Properties of the IndraMotion MTX P60" and then activate the NC kernel data in two steps.

Handling Instruction: Configuring the NC Kernel

The handling instruction contains instructions for adaptations for the NC kernel specific to the use of MTX Emulation. These instructions refer to existing data which are to be activated by restoring them.

IW Engineering / IndraMotion MTX P60: Adapt the properties of the device "IndraMotion MTX P60"

- With the right mouse button, click the device node "IndraMotion MTX P60". The "Properties" dialog of the "IndraMotion MTX P60" opens.

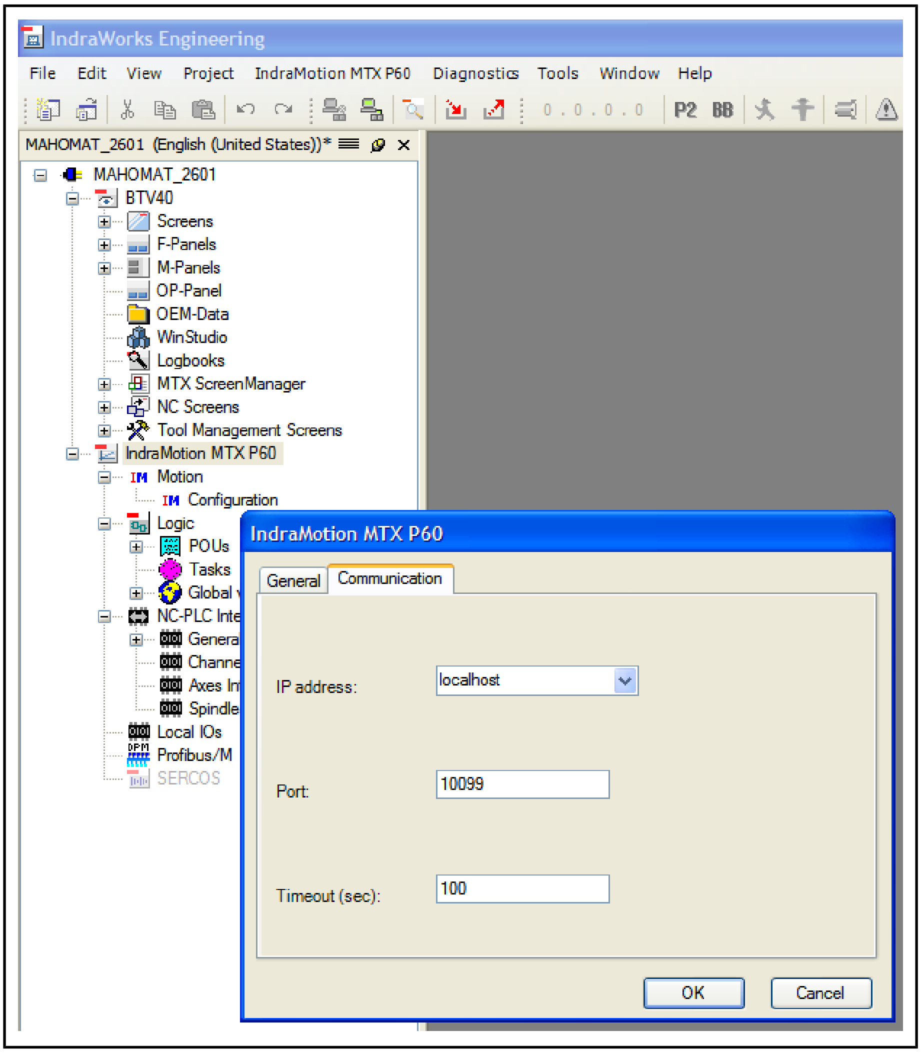


Fig.21-2: IndraMotion -

- Select the tab "communication" using the left mouse button.
- In the input field "IP address", select the entry "localhost" and exit the dialog with "OK".

IW Engineering / IndraMotion MTX P60: Restore NC kernel data

Procedure, see chapter 12.4 "Control Data" on page 270

Special Features when Working with MTX Emulation

MTX Emulation does not have a "CMP60 control" with which the startup modus is set, the startup phase is observed and different commands to save parameters, load firmware, start/stop PLC, etc. can be executed. The following sequence describes how this functions can be executed in MTX Emulation.

Emulation System Restart (Startup Mode 0)

It is necessary to restart the system to, for example, apply modifications for machine parameters (MACODA).

1. Exit IndraWorks.
2. Exit emulation in the DOS window by entering "14". The changed parameters are saved on the computer by writing the files "typ3ram.pxf" and "t3usrfep.pxf" on the PC; the changes become effective upon the next startup of emulation.
3. Restart emulation.

4. Restart IndraWorks.

Bootstrapping (Startup Mode 6)

Via bootstrapping, which can be compared to starting up the real MTX with startup mode 6, the RAM file system of the control is created again.

☞ A new root file system is created by bootstrapping. As a result, all data of the old file system is lost. If an intact user FEPROM file system exists, the PLC boot project and configuration data are loaded from there.

Bootstrapping is necessary to apply, for example, the modifications to the tool database size as well as the structure of tool data records.

1. Copy modifications to the tool data record into usrfep.
2. Exit IndraWorks.
3. Exit emulation in the DOS window by entering "14".
4. Delete the RAM file system. That is deleting the file "C:\My Documents\%user%\Local Settings\User Data\Rexroth\IndraWorks\%A%\MTX\Emu\typ3ram.pxf" [(where %user% = current user, %A% = IndraWorks installation) (to distinguish multiple installations)].
5. Restart emulation.
6. Restart IndraWorks.

Creating the User FEPROM File System (Startup Mode 7)

Creating a user FEPROM file system, which can be compared to execution of startup mode 7 of a real MTX, is necessary if all project-specific data can be deleted on a control or if the user FEPROM file system can is damaged.

☞ The user FEPROM is recreated by creating the user FEPROM file system. As a result, all data on the old file system is lost. The root file system is retained. The permanent CPL variables will be deleted.

---

1. Exit IndraWorks.
2. Exit emulation in the DOS window by entering "14".
3. Delete the USER FEPROM file system. That is deleting the file "C:\My Documents\%user%\Local Settings\User Data\Rexroth\IndraWorks\%A%\MTX\Emu\t3usrfep.pxf" [(where %user% = current user, %A% = IndraWorks installation) (to distinguish multiple installations)]
4. Restart emulation.
5. Restart IndraWorks.

21.5 Configuring the PLC

21.5.1 General

Description

Brief Description

First, the communication settings of the PLC project have to be changed if the IndraWorks PLC project comes from a real control or if the IndraWorks had been installed previously in a directory other than the directory of the active version. Then the PLC project can be activated.

Handling Instruction: Restoring the PLC Project

This handling instruction described the steps for commissioning the PLC project which had already been running on a real control.

IW Engineering / Logic: Restoring the PLC project

- Open the menu item "Properties" from the context menu at the "Logic" node below the MTX device node.
- In the "Properties" dialog, select the tab "Communication settings".

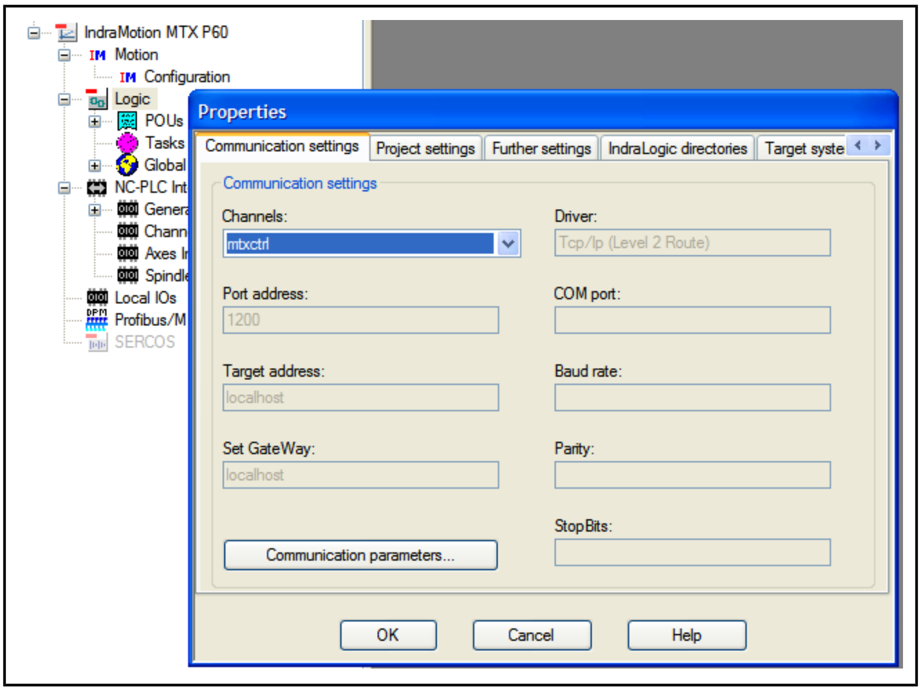


Fig.21-3: PLC communication parameters

- Press the button "Communication parameters...". The dialog "Communication Parameters" opens.
- Only one channel with the address "localhost" should exist in the "Communication parameter" dialog. If applicable, the existing settings have to be deleted first.

☞ A channel with an active connection cannot be deleted.

To delete this channel, close and restart IndraWorks and the MTX Emulation without executing an action which make changes in the communication settings of the IndraLogic.

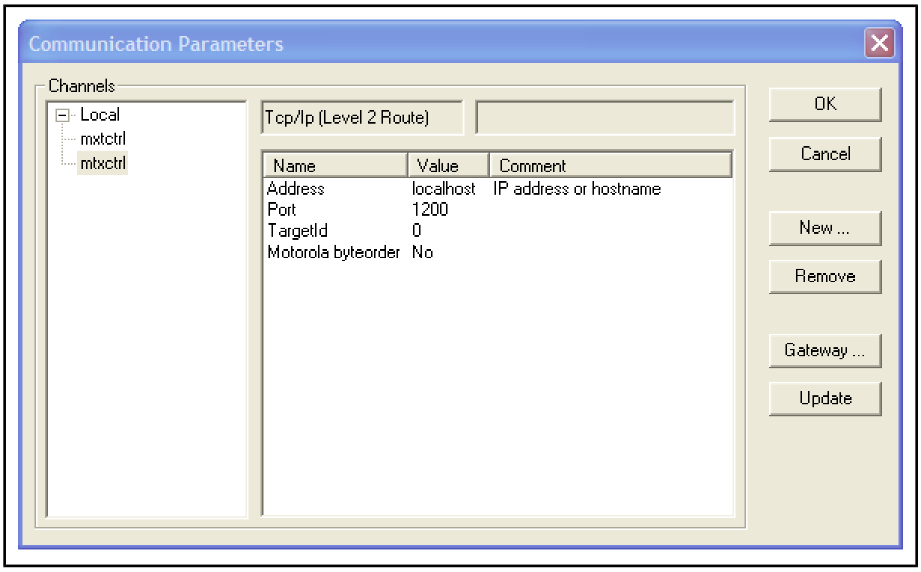


Fig.21-4: Settings in the dialog

- In the dialog "Properties" (see above), open the tab "IndraLogic directories" and, if applicable, adapt the path information to the installation paths of the MTX Emulation. Only one path entry may be maintained.

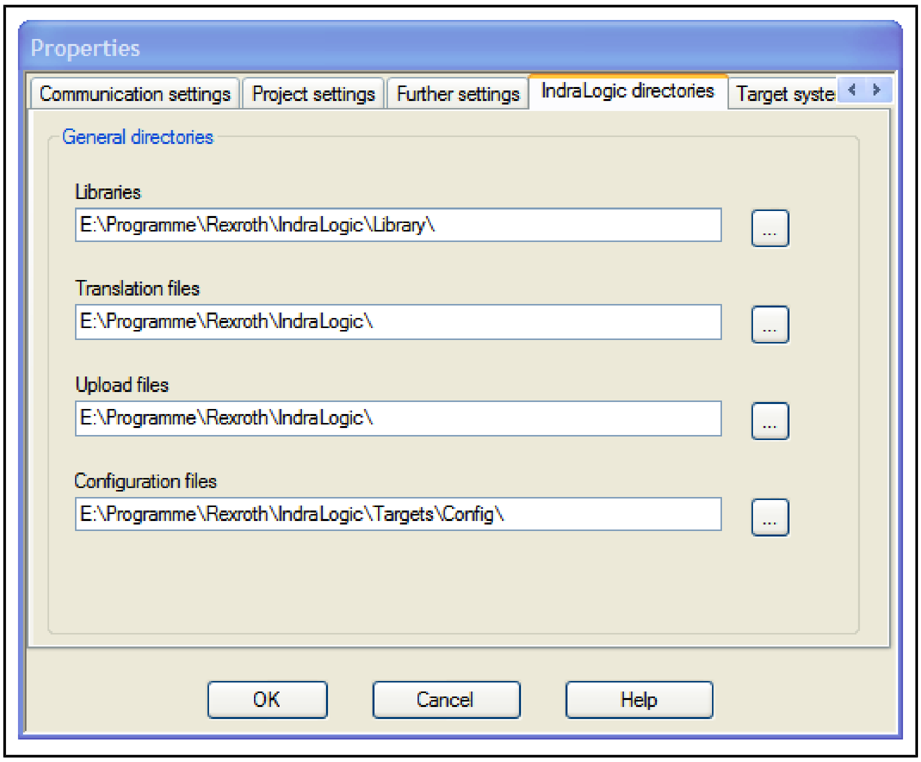


Fig.21-5: Settings in the dialog

- Exit the properties dialog by pressing "OK".
- In IndraWorks, click the node "logic" using the right mouse button and execute the command "update". The IndraLogic project has now been adapted to the "MTX Emulation" and can be started.
- IndraLogic can be started with a double-click on the "Logic" node. Then, the PLC project can be compiled and a download can be executed.

Figure	Flowchart	Example	Instruction	Documentation
Documentation:	IndraLogic system description		Working with Projects	

Fig.21-6: Link

21.6 Starting and Exiting MTX Emulation

21.6.1 General

Description

Brief Description

Emulation comprises both the NC kernel and the PLC, which are started and stopped together. For this purpose, the Windows application "Sco.exe" is available; its functioning during runtime can be checked via a DOS window.

To accept e.g. tool data configuration changes, it is necessary to restart the system.

Boundary Conditions

Emulation must always be started prior to working with the IndraWorks Engineering interface and the IndraMotion MTX user interface.

Handling Instruction: Starting Emulation

The handling instruction contains a description of how to start the emulation (CNC and PLC).

Windows: Starting Emulation

- Start the emulation with the command file under "C:\My Documents\%user%\Local Settings\User Data\Rexroth\IndraWorks\%A%\MTX\Emu\emu.bat" [(where %user% = current user, %A% = IndraWorks installation) (to distinguish multiple installations)].
- Trouble-free startup of emulation can be recognized by means of the feedback (see the below figure).

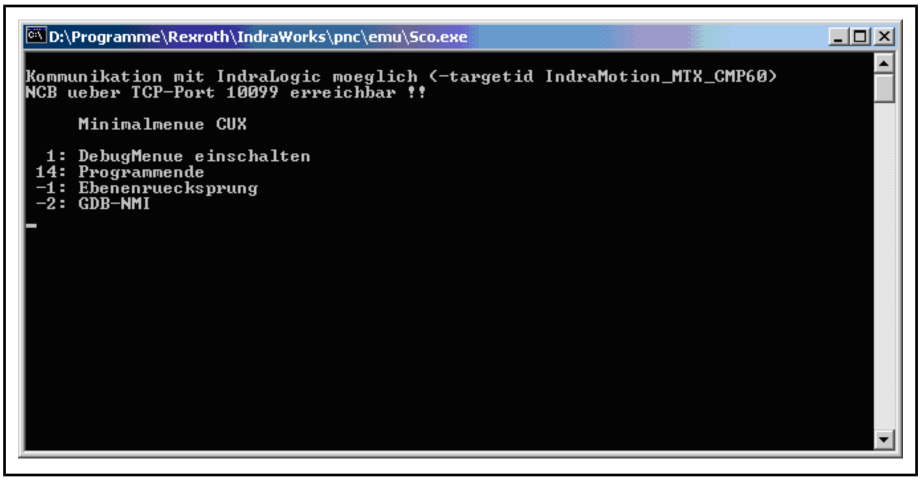


Fig.21-7: Start emulation

☞ Emulation must always be started prior to working with IndraWorks Engineering and IndraMotion MTX. It is recommended to create a link on the Windows desktop for starting the command file.

Virtual Commissioning of the MTX

Figure	Flowchart	Example	Instruction	Documentation
Documentation:	IndraWorks commissioning		Start emulation	

Fig.21-8: Link

Handling Instruction: Stopping Emulation

The handling instruction contains a description of how to stop the emulation (CNC and PLC).

Windows: Starting Emulation

- Emulation is closed by entering "14" in the DOS window.

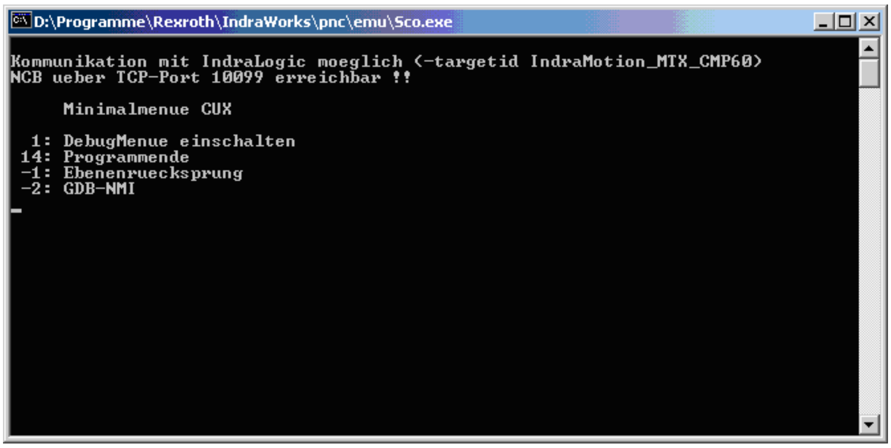


Fig.21-9: Stopping emulation

☞ Any changed data (e.g. machine parameters, tool data) will be saved only if Emulation is exited by entering "14" in the DOS window.

Figure	Flowchart	Example	Instruction	Documentation
Documentation:	IndraWorks commissioning		Stopping emulation	

Fig.21-10: Link

Handling Instruction: System Restart of Emulation

The handling instruction contains a description of how to restart the emulation (CNC and PLC). A system restart of Emulation is necessary e.g. to make changed machine data effective.

Windows: Restarting Emulation

- Exit IndraWorks Engineering Desktop and/or Operation Desktop.
- Exit emulation in the DOS window by entering "14". The changed parameters are saved on the computer by writing the files of file system "typ3ram.pxf" and user feprom "t3usrfep.pxf"; the changes become effective upon the next startup of emulation.
- Restart emulation.
- Restart IndraWorks Engineering Desktop and/or Operation Desktop.

Any changed data (e.g. MACODA, tool data) will be effective only if Emulation is exited by entering "14" in the DOS window. It is not necessary to set different startup modes for the MTX Emulation.

Figure	Flowchart	Example	Instruction	Documentation
Documentation:	IndraWorks commissioning		Emulation system restart	

Fig.21-11: Link

21.7 Starting the Virtual User Panel VAM 40

21.7.1 VAM Simulator

General Description

The VAM simulator is used as a replacement for the real VAM 40 and VAM 41 when you are working with MTX Simulator. The appearance and functions replicate the real VAMs. In the current version, the texts are available in English and German.

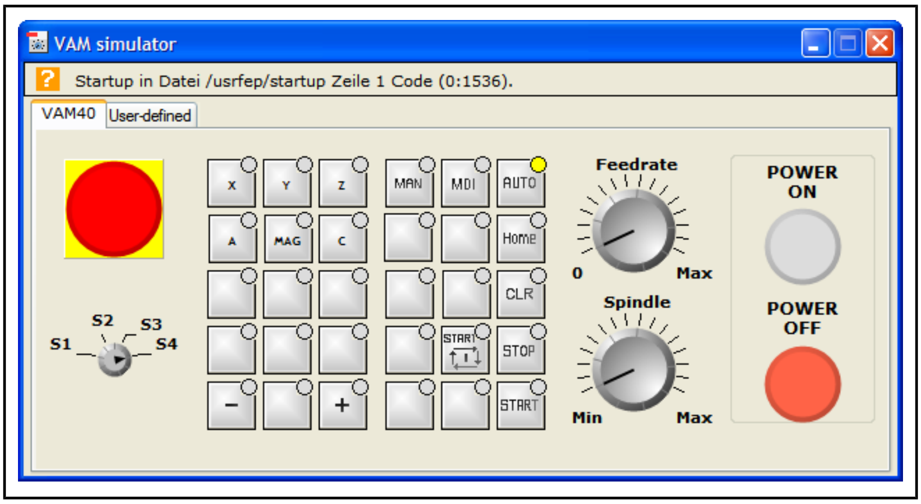


Fig.21-12: VAM simulator VAM 40

Configurator

The VAM simulator is configured in IndraWorks Engineering.

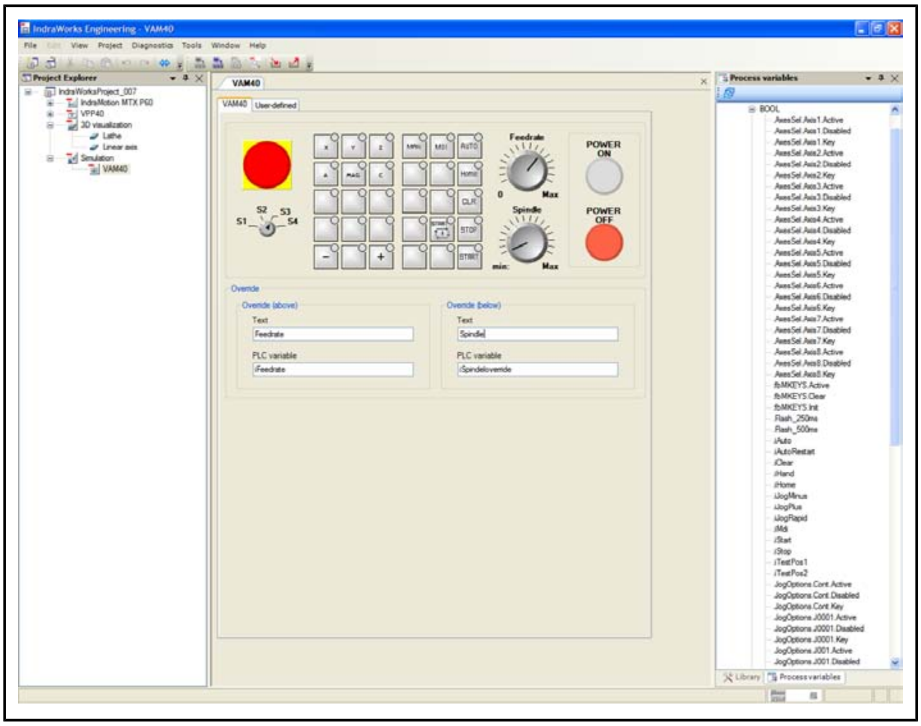


Fig.21-13: Configuration in IndraWorks

Project Node Simulation with Subnode VAM40 or VAM41

Double-clicking the project node of a virtual user panel calls the pages for configuration. The top half of the screen shows the image of the virtual user panel, while the configuration screen of the element selected in the image is shown in the bottom half.

The Configuration Page of the Emergency Stop

Only the PLC variable for later communication can be generated on the emergency stop page.

The Configuration Page of the Key-Operated Switch

The individual switch settings are labeled and the PLC variables are assigned on the configuration page of the key-operated switch.

The Configuration Page of the Override

The configuration page of the override is used to label and assign the PLC variables.

The Configuration Page of the Keypads

This page is used to assign the PLC variables and to label the keypads with texts or prepared images. Each individual key can be allocated by entering a text or by dragging and dropping an image from the symbol list.

☞ Depending on the selected type (VAM 40 or VAM 41), the following different functions are available.

The Configuration Page of the Quick-Stop Module

The configuration page of the quick-stop module can be used to make various settings for the two keys:

- Labeling the key
- Function

  Since the keys on the real VAM 40 are hardware-wired and the switches are equipped with make and break contacts, these settings can also be selected here.
- Assigning PLC variables

The Configuration Screen for the Freely Configurable Elements

In the configuration screen of the eight freely configurable elements, the following ones can be selected:

- LED

- Button with LED
- Button without LED
- Switch with 2 positions
- Switch with 3 positions

  The center position has no function.

These elements can be labeled and assigned PLC variables.

☞ Since the keys on the real VAM 41 are hardware-wired and the switches are equipped with make and break contacts, these settings can also be selected here.

The Application

The application is separate; it is used to control the PLC program in IndraMotion MTX Emulation. This can be started independently of IndraWorks Engineering or IndraWorks Operation. When the VAM simulator is started, the configuration is read out of the currently active project.

☞ The VAM simulator communicates directly with Emulation. IndraWorks Operation or IndraWorks Engineering do not need to be started.

21.7.2 Using the Virtual User Panel

Configuration in IndraWorks

Configuration of the Virtual User Panel

Creating a Virtual User Panel

In order to be able to configure a Virtual User Panel, a project must have been created in IndraWorks.

A virtual user panel is created in an existing IndraWorks project via **Project node** ▸ **right mouse button** ▸ **Add new element** ▸ **Virtual user panel** or using the library via tab**Simulation** ▸ **Virtual user panels**; then the panel is dragged to the project node and dropped there.

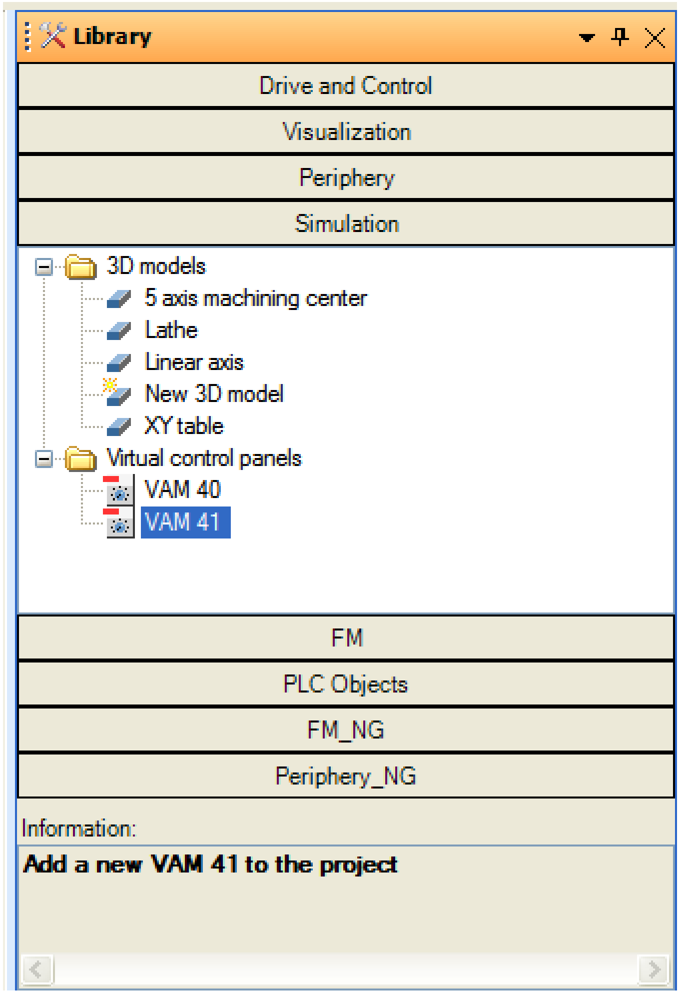


Fig.21-14: Simulation library

Then dialog box "Create VAM4x" appears, in which the following settings can be made:

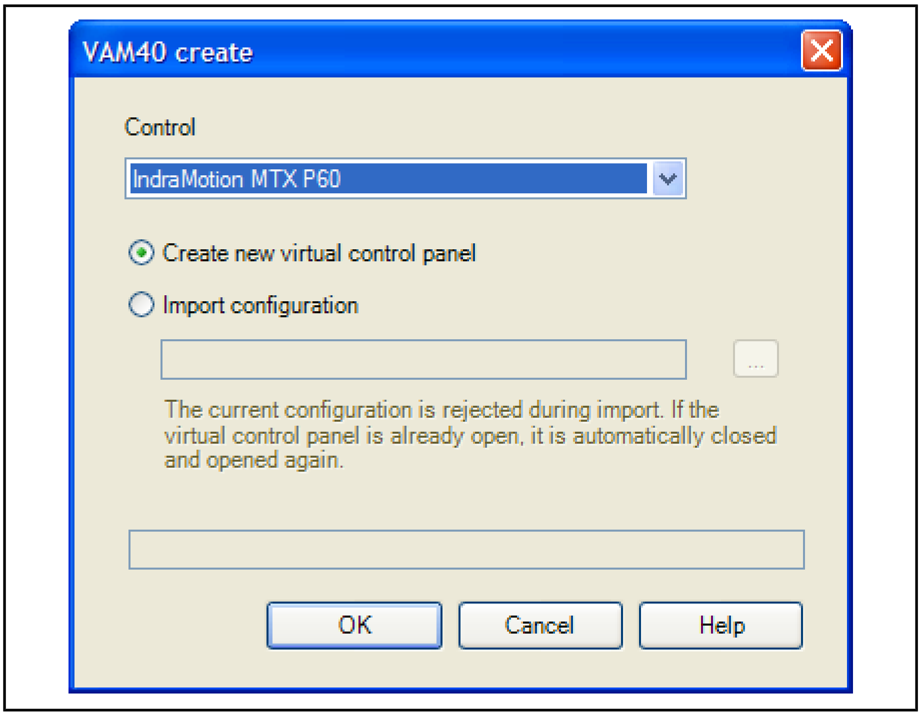


Fig.21-15: Dialog box "create new VAM simulator"

- Select the control with which the selected VAM is to communicate.
- Creating a new virtual user panel.

  A new VAM simulator is created by selecting**create new virtual user panel ▸ OK**.
- Creating a preconfigured user panel.

  Select function "import configuration" and then select a configured configuration file in the browser using the <...> button. Then confirm the selection and exit dialog box "create new VAM4x" by pressing <OK>.

"VAM" then appears as a subnode in the Simulation node.

The configuration pages open by <double-clicking> the node of the VAM (or clicking the **right mouse button on ▸ Open**).

Assigning Process Variables

Process variables are assigned by dragging and dropping them onto to the corresponding element in the VAM 40 or VAM 41 display. The MTX simulator must have been started in order for this list to be visible.

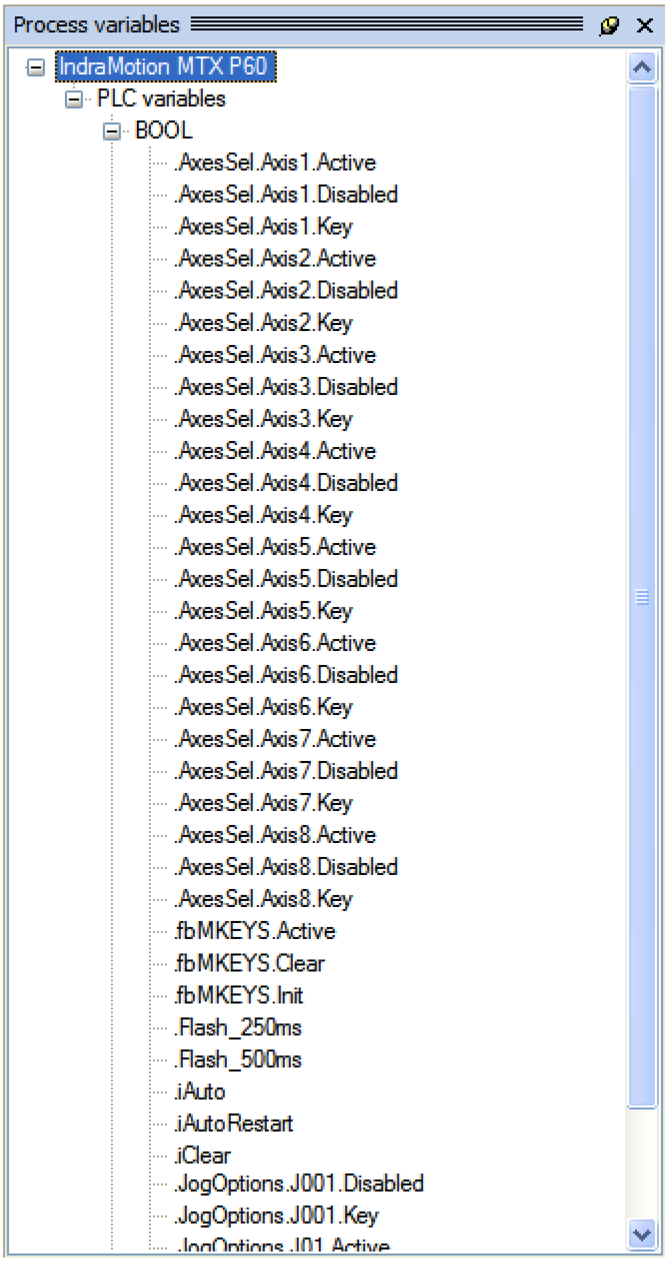


Fig.21-16: View of process variables

Configuring the Emergency Stop

There are two ways to assign a PLC variable to the emergency stop.

1. In the process variables window, select a BOOL variable and drag and drop it onto the display of the emergency stop in the VAM. After you release the mouse button, the configuration page of the emergency stop - with the PLC variable entered in the text field - opens automatically if it is not yet open.
2. Activate the configuration page by placing the focus on the emergency stop in the display of the virtual VAM. Then enter the variable with which the emergency stop is to communicate in the text field.

☞ The emergency stop can communicate only with BOOL variables!

Configuration of the Key-Operated Switch

Labeling the Key-Operated Switch By highlighting the key-operated switch in the display of the virtual VAM, its configuration screen opens. The switch positions are labeled in the individual text fields. The number of characters for the texts is unlimited; the new label appears immediately in the display of the VAM.

Assigning PLC Variables PLC variables can be assigned in two ways:

1. Select the PLC variable from the process variables tree and drag and drop it onto the display of the key-operated switch. Then a contextual menu opens in which selections can be made to specify which switch setting should later communicate with the variable. The assignment of positions 1 - 4 goes from left to right.
2. Enter the PLC variable in the appropriate text field. The configuration page of the key-operated switch is activated by highlighting the key-operated switch in the display of the VAM.

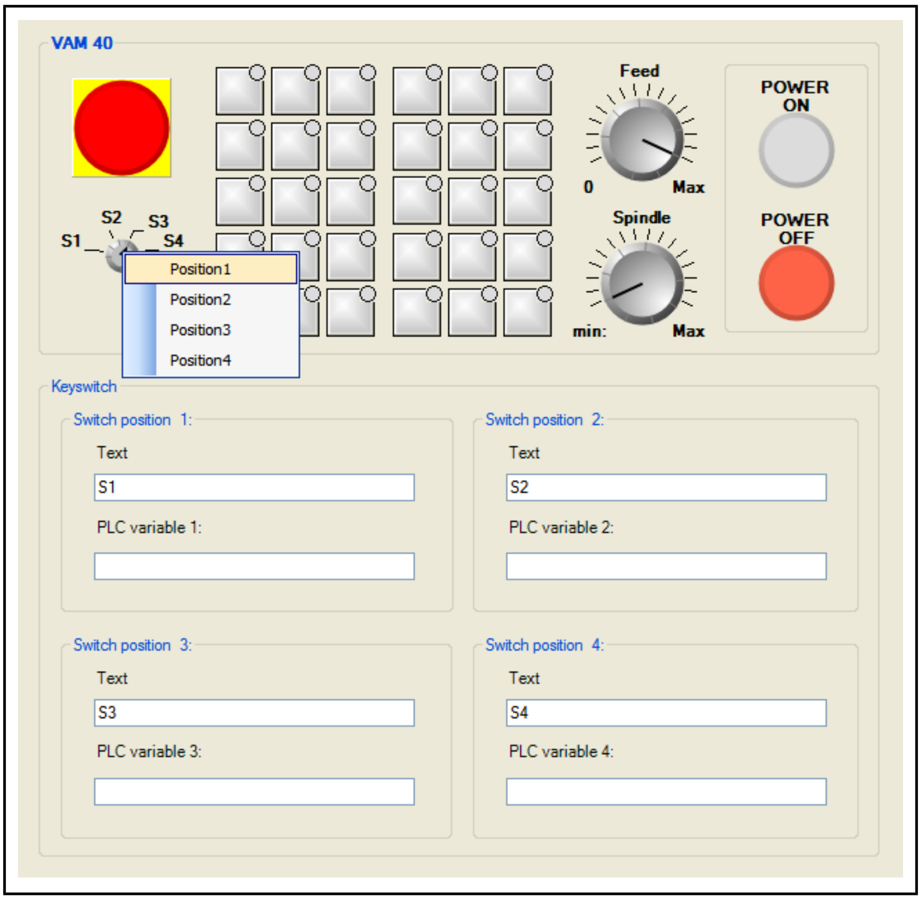


Fig.21-17: Configuration page of the key-operated switch

Only BOOL variables can be assigned to the key-operated switch.

Configuring the Overrides

Labeling the Overrides Highlighting one of the overrides in the display of the virtual VAM opens its configuration page. The overrides are labeled in the individual text fields. The number of characters for the texts is unlimited; the new label appears immediately in the display of the VAM.

Assigning PLC Variables As is the case for the other elements, PLC variables are assigned in two different ways:

1. Select the PLC variable from the process variables tree and drag and drop it onto the display of the override with which the variable is to be activated. The name of the variable can then be seen in the appropriate text field on the configuration page.
2. When you "go" to the display of an override, the configuration page of the override opens. The PLC variable can now be written in one of the text fields.

☞ The overrides are activated using BYTE variables.

Configuring the Keypads

The configuration page of the keypads is opened by highlighting a key on the keypad.

Assigning Text to the Keys

When labeling an individual key, first highlight the desired key on the keypad. Then switch to the text field of "Labeling the key" and enter the text.

Assigning Images to the Keys

The prepared images from the default page can be used, or you can add your own images using the user-defined page using Import. These are then available for every new virtual user panel.

The user-defined images are imported by pressing the <Import> button. This opens a dialog box in which the image with the suffix "*.bmp" can be searched for. Then the image appears on the tab and can be placed on the keys of the virtual user panels.

The Keys can be Assigned in Two Ways: Using Drag-and-Drop or, if the Key is Not in the Visible Area of the Windows Desktop, Using the Contextual Menu.

- Drag-and-drop

  First select any image in the "symbols" category. Then press the left mouse button and drag the cursor to the desired key to assign it.

- Contextual menu

  First select the key on which the image is to appear in the VAM image. Then select the image in the "symbols" category and press the right mouse button. This opens a contextual menu with the text "assign bitmap of key X3Y3". After the contextual menu item has been executed, the image appears on the desired key in the image of the virtual user panel.

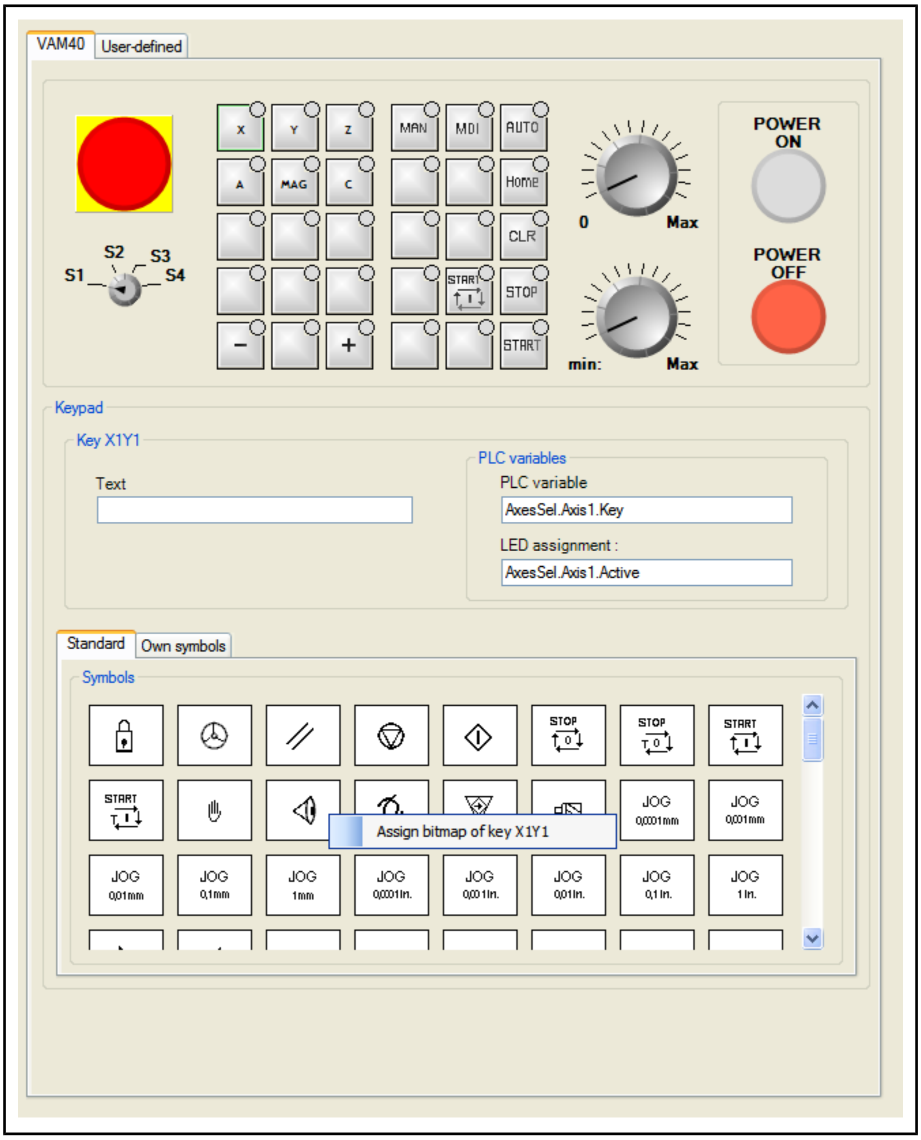


Fig.21-18: Assigning images using the contextual menu

Assigning PLC Variables to the Keys

As is the case for the emergency stop and the key-operated switch, PLC variables can be added in two different ways.

1. In the process variables window, select a PLC variable and drag and drop it onto the desired key. After you release the button, you must specify on what the variable should have an effect. The key and the associated LED can be selected to do this. The name of the variable can then be seen in the appropriate text field on the configuration page.
2. In the VAM display, highlight the button with which the PLC variable is to be addressed later. Then enter the PLC variable in the text field of the LED or key.

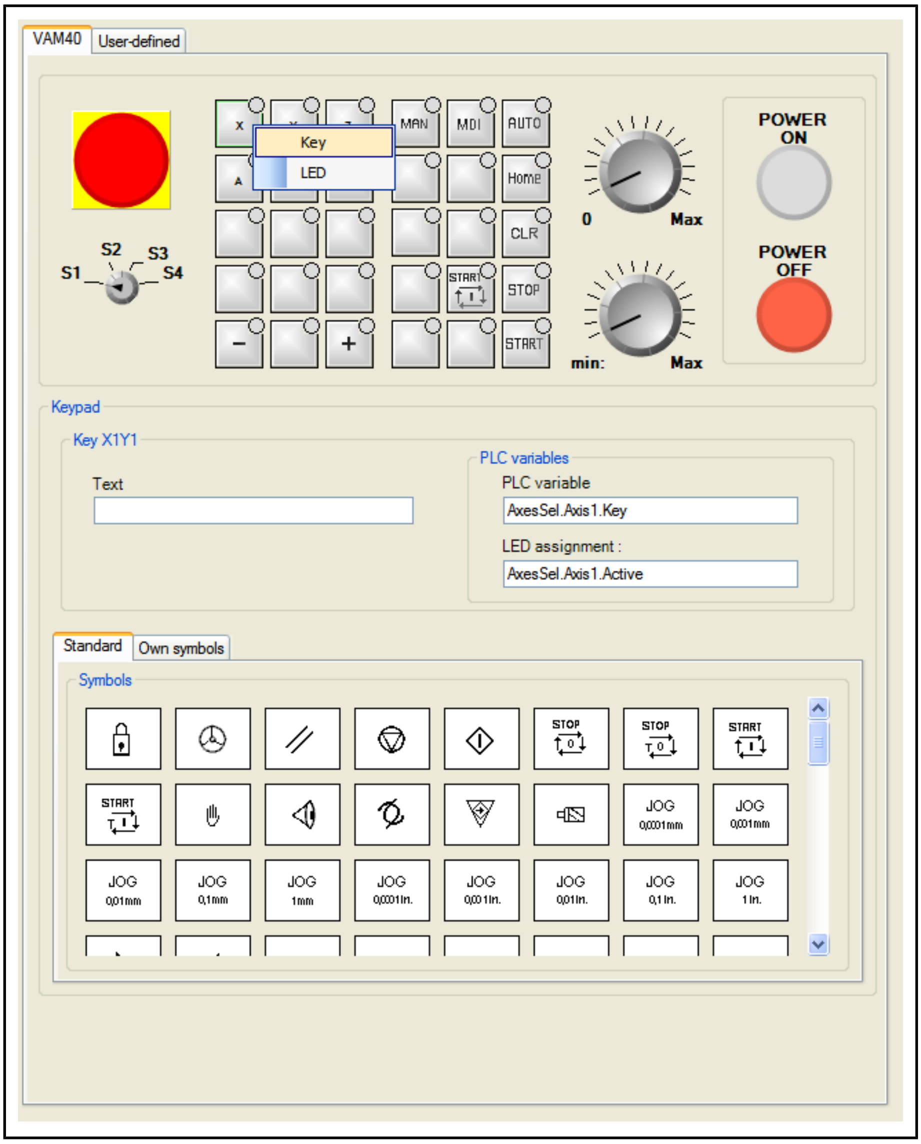


Fig.21-19: Configuration page of the keypads

☞ Only BOOL variables can be assigned to the LEDs and keys.

VAM 40-Specific Configuration Pages

Configuration of the Quick-Stop Module

The configuration page of the quick-stop module is opened by selecting the quick-stop module in the display of the virtual VAM 40.

Labeling the Keys

The new labels can be entered in the text fields of the "(Top) key " and "(Bottom) key". Line breaks and line lengths are not taken into account. The label is shown immediately in the VAM 40 display.

Functions of the Keys

Since the keys on the real VAM 40 are hardware-wired and the switches are equipped with make and break contacts, these settings must be selected here. This is accomplished by setting function "Break-contact" or "Make-contact".

Assigning PLC Variables

There are also two different ways to assign the PLC variables to the keys and the LED in the quick-stop module.

1. In the process variables window, select a PLC variable and drag and drop it onto the desired key. If the upper button is selected, a contextual menu appears in which the association of the PLC variable still remains to be clarified. The key and the associated LED can be selected to do this. There is no dialog box for the lower button because only the button can be assigned. The name of the variable can then be seen in the appropriate text field on the configuration page.
2. Highlight the quick-stop module in the display of the VAM 40. Then enter the PLC variable for the desired element in the text field of the configuration page.

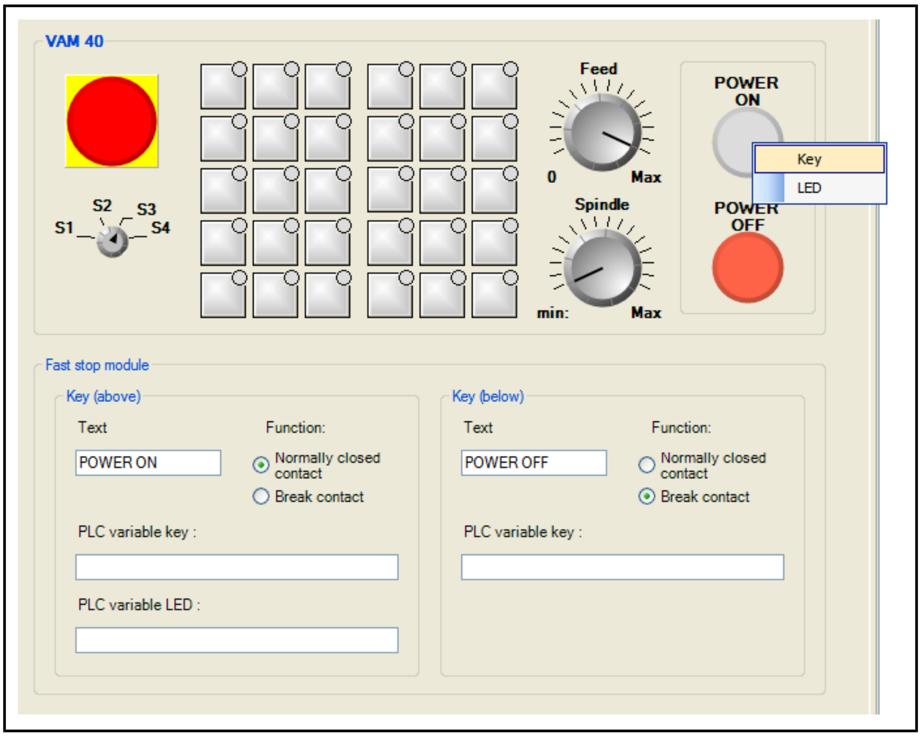


Fig.21-20: Configuration page of the quick-stop module

☞ Only BOOL variables can be assigned to the LED and keys.

VAM 41-Specific Configuration Pages

Configuration of the Eight Freely Configurable Elements

The Configuration page of the eight freely configurable elements is shown by selecting either the square boxes or the previously selected elements in the VAM 41 image.

Use "Function of the element" to choose among several elements.

- LED
- Button with LED
- Button without LED
- Switch with 2 positions
- Switch with 3 positions

  No PLC variable can be assigned to the center position.

These elements can be labeled and assigned PLC variables.

☞ Since the keys on the real VAM 41 are hardware-wired and the switches are equipped with make and break contacts, these settings can also be selected here.

Labeling the Elements

The individual elements can be labeled using the "Text" field on the Configuration page. In this case also, line breaks and character length are irrelevant; the label is immediately shown in the VAM 41 display.

Function of the Keys with and without LED

Since the keys on the real VAM 41 are hardware-wired and the switches are equipped with make and break contacts, these settings must also be selected here. This is accomplished by setting function "Break-contact" or "Make-contact".

Function of the Switches

In the case of the switch with 2 positions, the PLC variable is always set to logic one, according to the switch position. The PLC variable that was assigned the other position is set to logic zero.

The function of the switch with 3 positions is identical to that of the switch with 2 positions; however, both PLC variables are set to logic zero in the center position.

Assigning PLC Variables

As in the other configuration pages, there are two ways to assign the PLC variables to the elements.

1. In the process variables window, select a PLC variable and drag and drop it onto the desired element. If the element is a key with LED or a switch, a contextual menu appears in which the association of the PLC variables remains to be clarified. For "key with LED", you can choose between LED and key; for the "switches", you can choose between the left and right sides.
2. Highlight the module in the display of the VAM 41. Then enter the PLC variable for the desired element in the text field of the configuration page.

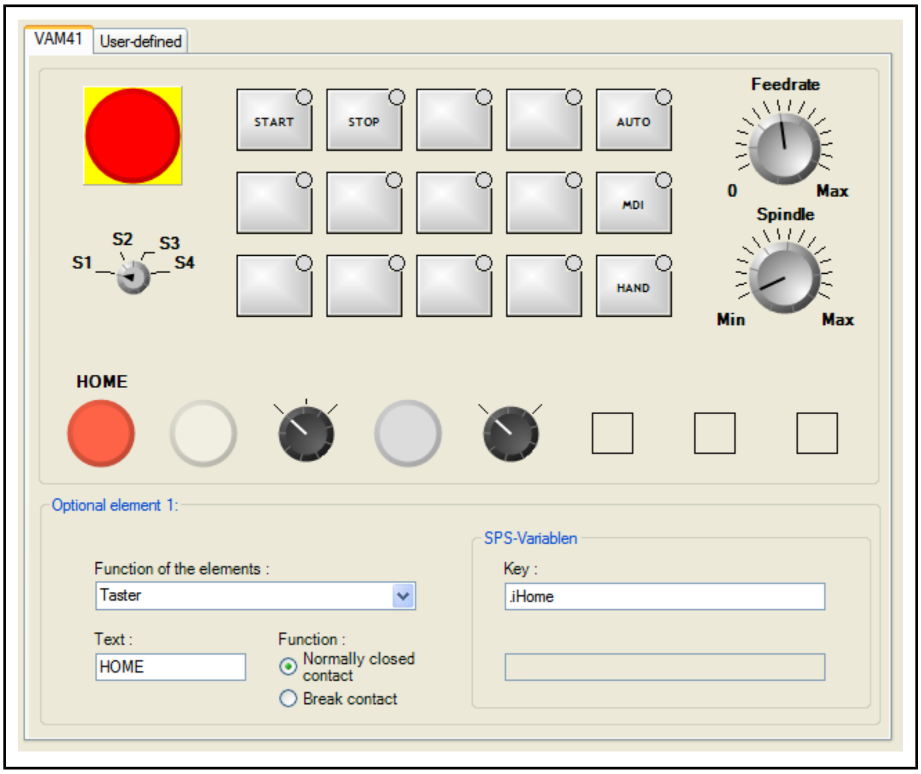


Fig.21-21: Freely configurable elements: VAM 41

> As for the keys and switches, only BOOL variables can be assigned to the LED.

Configuration of the Tab

Parallel to the VAM 40 / VAM 41, PLC variables can be freely assigned to the tab for space reasons or if they have a type other than "BOOL" and "BYTE". When a variable has been successfully assigned to a field, the data type-dependent value field appears.

> However, the current value of the variable is visible only in the application; it can also be influenced only there!

As is the case for the other configuration pages, PLC variables are assigned in two different ways:

1. In the process variables window, select a PLC variable and drag and drop it onto the desired field. The name of the variable is then in the text field and the suitable value field is shown.
2. Highlight one of the 14 fields and enter the PLC variable.

   When you exit the field, a check is made whether the entered PLC variable exists. If this is not the case, it is shaded red. The color disappears only after this has been corrected and the field has been exited again.

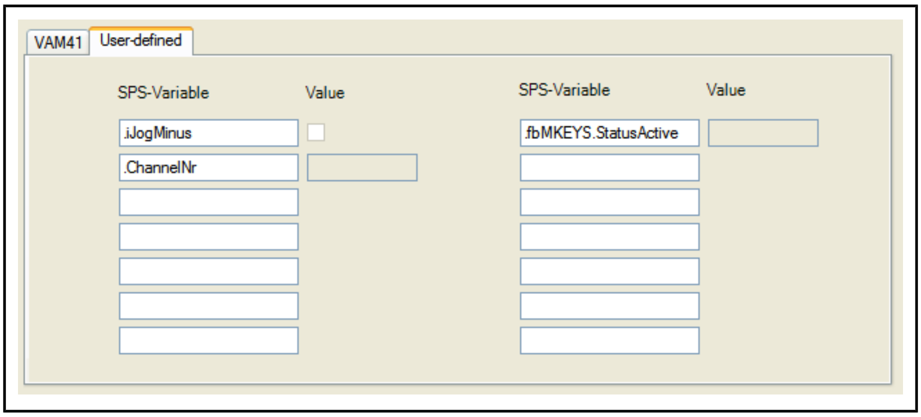


Fig.21-22: Configuring the tab in IndraWorks

Contextual Menu of the Virtual User Panel in Project Explorer

The contextual menu is called via **Project Explorer ▸ VAM4x ▸ right mouse button.**

The following functions can also be carried out:

Open The virtual user panels are opened using this function.

Import... During an import, the same dialog box opens as for the creation of a new user panel, but the function "Create new virtual control panel" is dimmed. The control with which communication is to occur later can be changed if necessary.

Then select the configuration to be imported using the <...> button in the browser and confirm it.

> The name of the configuration file must have the extension "*.vcp" or must be "VirtualControlPanel_VAM40.xml" (from version IW-Simulation-01VRS)!

Then confirm dialog box "Create new VAM4x" by pressing <OK>. If a VAM is open at this time, it is closed automatically; the previous configuration is discarded and the panel is opened again with the imported configuration.

A check is made whether the configuration to be imported corresponds to the type of the new virtual control panel. If this is not the case, the import is cancelled.

Export...

If menu item "Export" is selected, dialog box "Export" opens; the storage location and the name of the file can be specified here. After confirmation, a current copy of the configured control panel with the extension "*.vcp" exists in the target directory.

Assign Control...

Contextual menu item "Assign control..." is used to subsequently assign or change a control. When a selection has been made and the dialog box is confirmed by pressing <OK>, the virtual user panel closes when it is supposed to be open. Then it is automatically reopened.

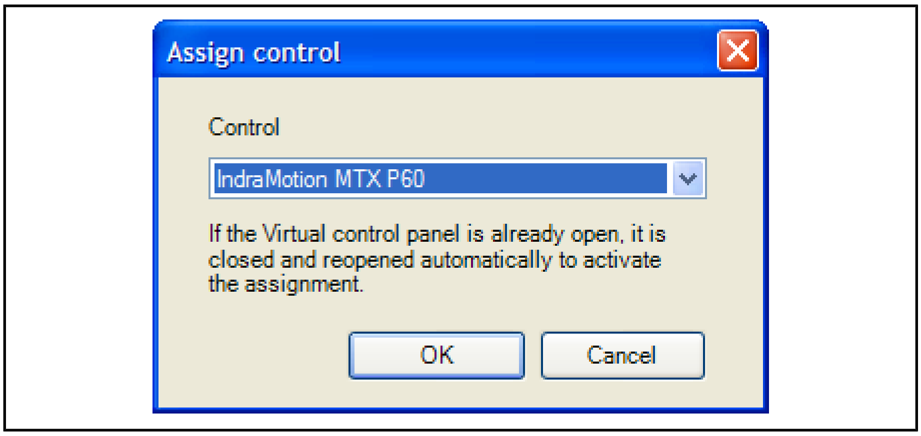


Fig.21-23: Assign control

Delete

If function "Delete" is selected, the virtual user panel is removed from the project. However, the simulation node is retained.

Connection to Virtual Control

Before the VAM simulator can be started successfully, the following must be carried out to be able to establish a connection to the virtual control:

Preparations in IndraWorks Engineering

1. Set the project in which the virtual user panel was configured to active. This is accomplished by selecting menu item **project ▸ activate for IndraWorks Operation**.
2. Start MTX emulation.

Preparations in IndraLogic

.

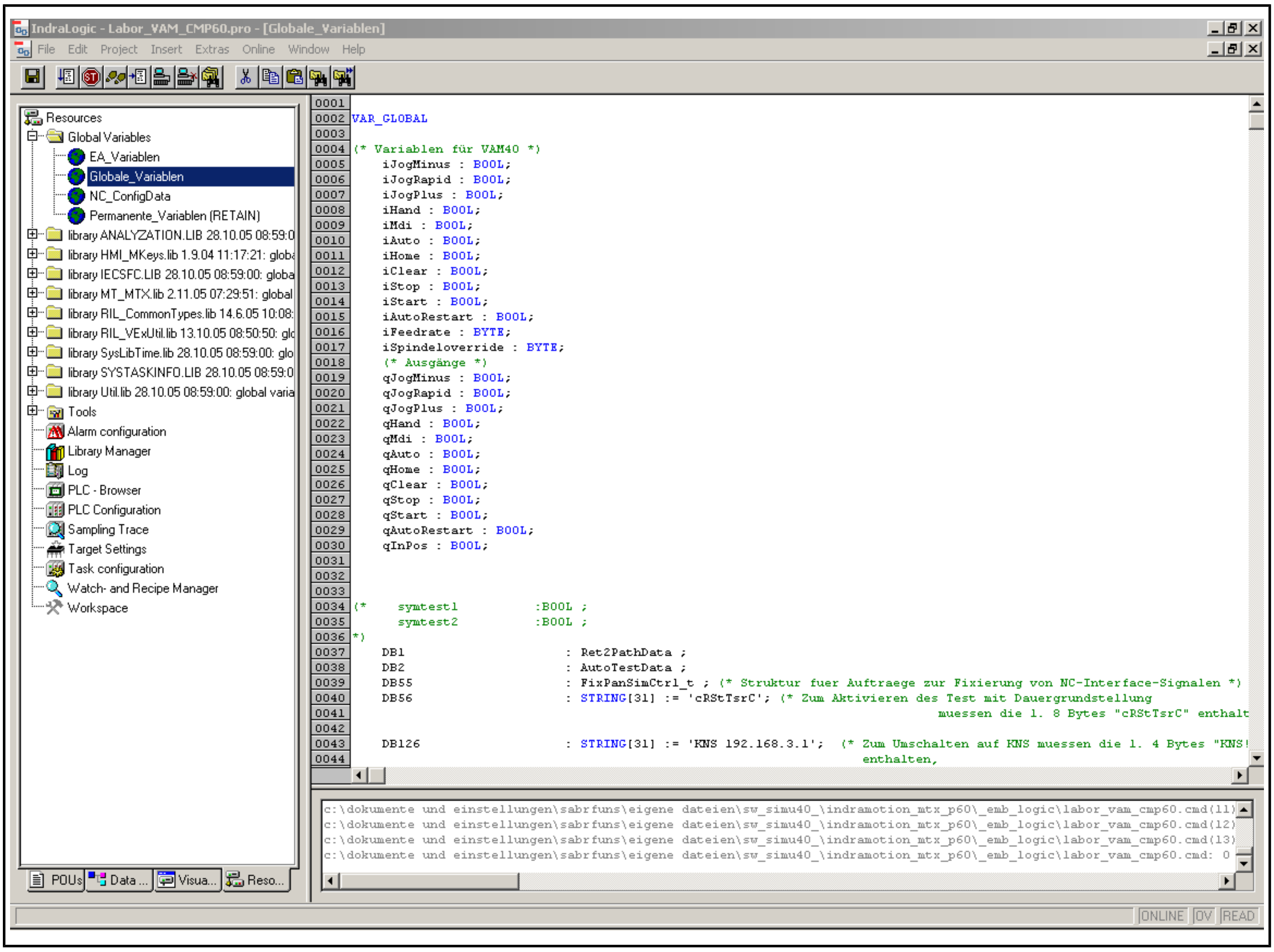


Fig.21-24: Preparations in IndraLogic

- In the PLC program, create all the variables that are to receive a connection with the VAM simulator as "global variables". These variables are used to activate the individual switch settings, buttons, lamps and overrides.

  These are created as follows: Variable name: BOOL, e.g. **iStart : BOOL;**

Do not assign an address; remove it if necessary!

- Reset the PLC program, log in and start.

Starting and Operation

Start

The VAM simulator is started via **Start ▸ Program Files ▸ IndraWorks ▸ Virtual User Panel** or using the "IndraWorks Virtual User Panel" icon on the desktop.

Operating the User-Defined Tab

All PLC variables that were added to the configuration can be neither deleted nor modified. Only the values of these variables can be influenced. Variables that are added during runtime are deleted after the VAM simulator is closed; these must be entered again at the next startup.

As is the case for the configuration, PLC variables can be added in two different ways.

1. Press the <...> button next to the desired PLC variable field. The process variable browser, in which the desired PLC variable can be selected, then

opens. This is entered in the field by pressing <OK>; the data-dependent value field then appears with the current value from the PLC.

2. Highlight one of the PLC variable fields that have not yet been assigned and enter the PLC variable in the field.

   When you exit the field, a check is made whether the PLC variable exists in the PLC. If this is not the case, it is shaded red.

The variable value is modified by entering the desired value in the Values field. The value is sent to the PLC by pressing <Enter> or by exiting the Values field.

If the value range is exceeded when modifying the PLC variable value and if this is sent to the PLC, an error message appears and reentry is required!

Contextual Menu in the VAM Simulator

If the VAM simulator has been started, three functions can be activated/deactivated using the **right mouse button**. These are saved so that they do not have to be set every time the VAM simulator is started.

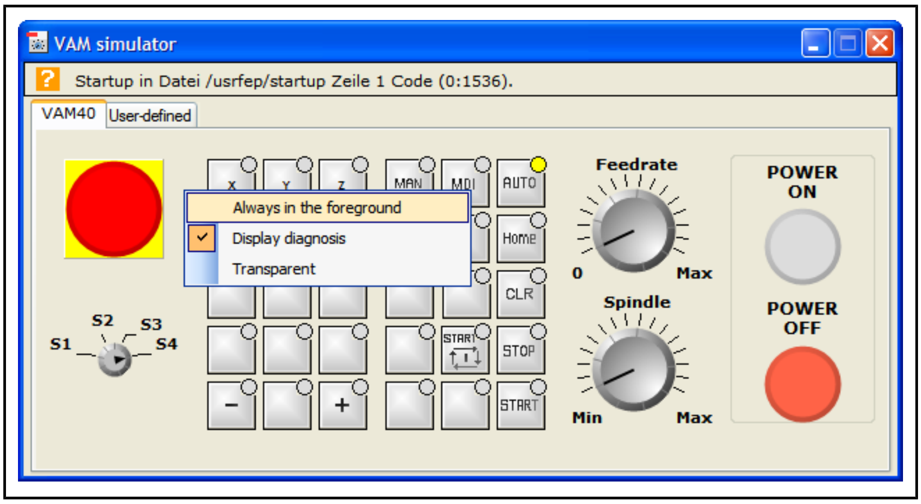


Fig.21-25: Contextual menu display in the virtual user panel

Function "Always in the Foreground"

Function "Always in the foreground" is used to always keep the VAM simulator in the foreground so that it does not disappear behind Operation Desktop when this is being used. It can be activated/deactivated at any time by selecting **right mouse button** ▸ **Always in the foreground**.

Function "Display Diagnosis"

The diagnosis display is used to show error, warning and info texts even if Operation Desktop is closed. It can be activated/deactivated at any time by selecting **right mouse button** ▸ **Display diagnosis**. The location for the display is always adapted automatically.

Only the message that is currently pending in the control is displayed!

Function "Transparent"

This function can be used to make the VAM transparent so that applications in the background can be seen. It can be activated/deactivated at any time by selecting **right mouse button** ▸ **Transparent**.

Error Messages and Remedies

Error Box "No Active Project Exists!"

Remedy: Open IndraWorks Engineering and set the project in which the Virtual User Panel was configured to Active. This is accomplished by selecting menu item **Project** ▸ **Activate for IndraWorks Operation**.

Then restart the "IndraWorks Virtual User Panel".

Error Box "No Virtual User Panel Configured!"

Remedy: Open IndraWorks Engineering and open the project in which the Virtual User Panel was configured. Within it, create a virtual user panel via **Project node** ▸ **right mouse button** ▸ **Add new element** ▸ **Virtual User Panel** or, for **Library** ▸ **Simulation**, select a **Virtual User Panel** ▸ **VAM4x** and move it to the project node using drag-and-drop. Then proceed according to chapter "VAM 40-Specific Configuration Pages" on page 516.

Error Box "Communication Could Not be Established!"

Remedy: Start emulation via **Start** ▸ **Program Files** ▸ **Rexroth** ▸ **IndraWorks** ▸ **MTX Emulation**

Error Box "Error Writing PLC Variables"

This error message can have various causes; therefore various remedies are available:

1. Check whether the PLC variables were stored under "global variables" and are of type BOOL. Exception: Override activation uses BYTE variables.
2. Check the communication to the PLC and reestablish the connection if necessary.

Error Box "No Control Assigned!"

Remedy: Open IndraWorks Engineering and the project in which the Virtual User Panel was configured. Then execute the following: **VAM** ▸ **right mouse button** ▸ **Assign control** and select the control with which communication is to be performed. Restart the VAM simulator.

21.8 Starting Operation Desktop

Brief Description

The MTX user interface is started in the same manner as the user interface of a real application. For this purpose, a link is created on the desktop ("IndraWorks HMI") when installing the MTX Emulation.

22 Operation Desktop

22.1 Zero Offsets

22.1.1 General

Description

Zero offsets (ZO) are used to displace (i.e. transform) the machine coordinate system (MCS) in the direction of the machine coordinates. The resulting offset coordinate system is called a "Local Machine Coordinate System" (LCS). If no coordinate transformation and no axis transformation is active when the ZO is called up, the offset machine zero point corresponds to the workpiece zero point.

The offset values of the machine coordinates are stored in zero offset tables.

Each zero point offset table comprises 5 zero point offset bases (groups) with 6 zero point offsets each:

- G54.1 - G59.1
- G54.2 - G59.2
- G54.3 - G59.3
- G54.4 - G59.4
- G54.5 - G59.5

22.1.2 Creating Zero Offsets

Description

In Operation Desktop, zero offsets can be created and edited using OP key 4 "Program". Zero offset tables have a file extension of "*.zot" and are stored in directory "\database" under the root directory. If the zero offset table has been saved in a user-defined directory, it must be specified when the table is activated.

A template can be used to generate a new zero offset table. The channel assignment can also be configured.

Fig.22-1: Creating a zero point table

22.1.3 Changing the Zero Offset Table

General

Description

A table with 6 zero points (G54.1 to G59.1) is available. This table can be supplemented by 4 additional correction pages. The number of axes can also be increased. Furthermore, a comment can be assigned to each zero point.

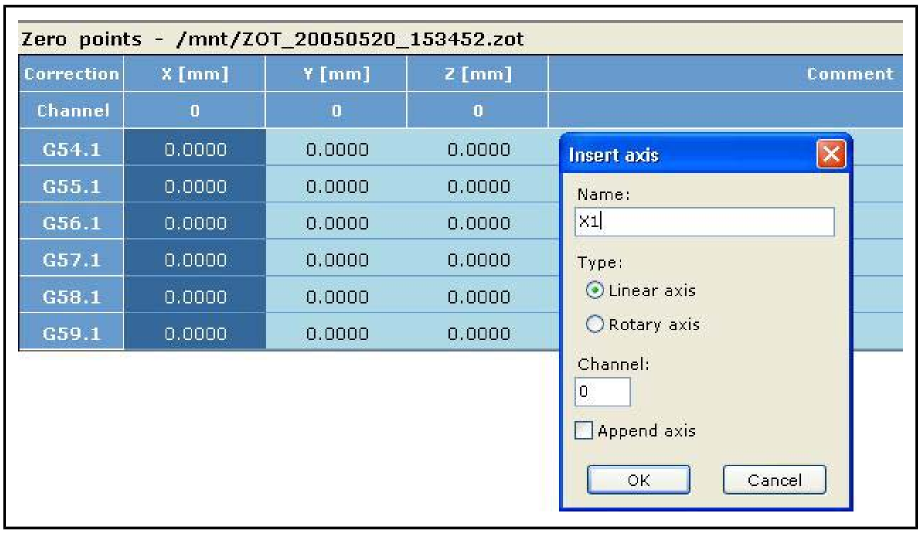


Fig.22-2: Zero offset table: adding axes

Any number of axes can be added Software version 04VRS]. No check is made to see if the axis has been configured.

Additional details regarding the activation and deactivation of zero points can be found in the "IndraMotion MTX" Functional Description.

Handling Instruction: Creating a Zero Offset (ZO) Table

Creating a ZO table in IW Operation.

IW Engineering / Configurator: Adapt display per NC parameter

		Instruction chapter "Handling Instruction: Zero Offsets (ZO) Table: Creating/Deleting/Editing an Axis" on page 532
Instruction		Editing parameters for ZO

IW-Operation/program: Create a ZO File

1. Start IW Operation
2. Select "Program" mode using OP key **Program (OP4)**.

Fig.22-3: OP key "program"

3. Navigate to the desired directory or create a new one if necessary.

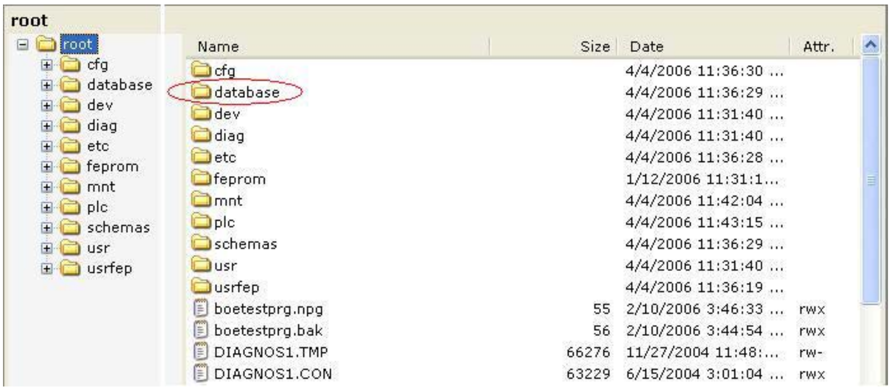


Fig.22-4: Path overview in mode "program"

4. Select the desired zero point file with the extension "*.zot".

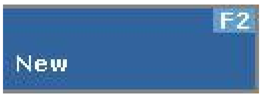


Fig.22-5: New key

5. Select "Zero points" <F7> from the F-keypad.

Fig.22-6: Zero points key

The "Create new zero point table" dialog appears.

Fig.22-7: Creating a zero point table

6. Assign a name for the new file, assign the desired channel and specify whether the zero point table applies to all the axes (checkbox "Assignment for all axes").

The zero point offset (ZO) editor opens.

		Documentation
Documentation	IndraWorks HMI	Zero point offsets

IW Operation / Program / ZO editor: Create/delete a correction page

		Instruction chapter "Handling Instruction: Zero Offsets (ZO) Table: Adding/ Deleting a Correction Page" on page 532
Instruction		Create/Delete a Correction Page

IW Operation / Program: Create / delete / edit an axis

		Instruction chapter "Handling Instruction: Zero Offsets (ZO) Table: Creating/ Deleting/Editing an Axis" on page 532
Instruction		Create/Delete/Edit an Axis

IW Operation / Program: Enter values

The desired zero point offset can be entered/modified by selecting the desired block.

Correction	X1 [mm]	Z1 [mm]	X2 [mm]
Channel	0	0	0
G54.1	10.0000	20.0000	0.0000
G55.1	20.0000	0.0000	0.0000
G56.1	0.0000	0.0000	0.0000
G57.1	0.0000	0.0000	0.0000
G58.1	0.0000	0.0000	0.0000
G59.1	0.0000	0.0000	0.0000

Fig.22-8: Zero point table overview in the ZO editor

- Offsets from different ZO pages always work additively.
- Offsets within a ZO page mutually overwrite each other.

		Documentation
Documentation	IndraWorks HMI	Zero point offsets

Handling Instruction: Zero offset table (edit ZO table)

Editing a ZO table in IW Operation.

IW Operation / Program: Open the file to be edited

1. Start IW Operation.
2. Select "Program" mode using OP key **Program (OP4)**.

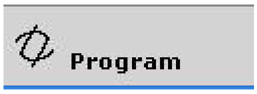


Fig.22-9: OP key "Program"

3. Navigate to the desired directory or create a new one if necessary.

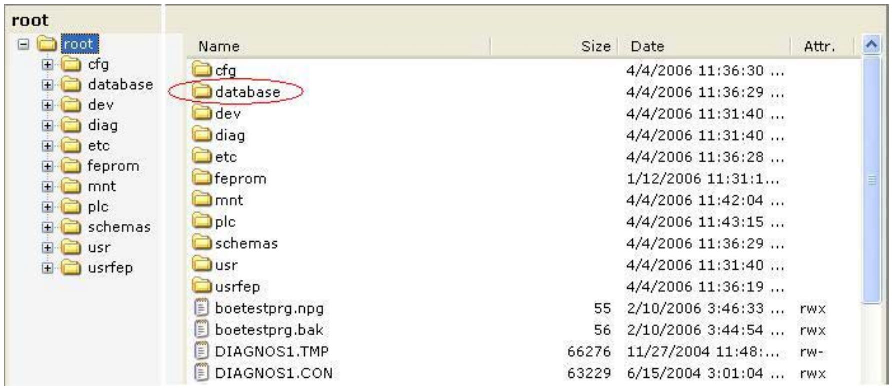


Fig.22-10: Path overview in the mode "Program"

4. Select the desired zero point file with the extension "*.zot".
5. Open the file by selecting F key "Edit" <F6>.

Fig.22-11: Edit key

– or –

6. Double-click the desired file to open it.

The zero offset (ZO) editor opens.

		Documentation
Documentation:	IndraWorks HMI	Zero offsets

IW Operation / Program / ZO editor: Create/delete a correction page

		Instruction chapter "Handling Instruction: Zero offset table (edit ZO table)" on page 528
Instruction:		Create/delete a correction page

IW operation / program: Create / delete / edit an axis

		Instruction chapter "Handling Instruction: Zero offset table (edit ZO table)" on page 528
Instruction:		Create/delete/edit an axis

IW operation / program: Enter values

The desired zero offset can be entered/modified by selecting the desired block.

Correction	X1 [mm]	Z1 [mm]	X2 [mm]
Channel	0	0	0
G54.1	10.0000	20.0000	0.0000
G55.1	20.0000	0.0000	0.0000
G56.1	0.0000	0.0000	0.0000
G57.1	0.0000	0.0000	0.0000
G58.1	0.0000	0.0000	0.0000
G59.1	0.0000	0.0000	0.0000

Fig.22-12: Zero point table overview in the ZO editor

- Offsets from different ZO pages always work additively.
- Offsets within a ZO page mutually overwrite each other.

		Documentation
Documentation:	IndraWorks HMI	Zero offsets

IW Operation / Program / ZO editor: Exit the ZO editor

1. If necessary, press the F key "Back" <F9> until "Exit zero points" is displayed on the F key <F9>.

Fig.22-13: Back key

2. Then exit the zero points editor by pressing F key "Exit zero points" <F9>.

Fig.22-14: Exiting the zero points editor

		Documentation
Documentation:	IndraWorks HMI	Zero offsets

Handling Instruction: Applying Zero Point Offset Parameter

Lists all parameters to be set regarding zero point offsets.

IW Engineering / Configurator: Adapt display per NC parameter

- Into the parameter **CorrUnit** "Unit for tables and corrections" (9020 00010) enter the desired table unit.
- Using the following parameters, the number of decimal positions for the different tables can be modified:
  - **PrecLinMetr** "Decimal position for linear axes in case of metric Display of table."
  - **PrecRotMetr** "Decimal position for rotary axes in case of metric Display of table." (6020 00021)

- The parameters **PrecLinInch** "Decimal position for linear axes in case of non-metric Display of table."
- **PrecRotInch** "Decimal positions for rotary axes in case of non-metric Display of table." (6020 00022)
- The parameter **NofZotSet** "Number of zero point offset blocks to be displayed" (6005 00061) defines, how many zero point offset blocks/ groups are displayed in the NC.

	Documentation	Instruction
Instruction:		Editing machine parameters
Documentation:	MTX Parameter Description	Zero point offset

Handling Instruction: Activate the zero offset (ZO)

Activation of the ZO from the NC program.

IW operation / program: If applicable, create a zero offset table (ZO table)

		chapter "Handling Instruction: Activate the zero offset (ZO)" on page 531
Instruction:		Creation of a ZO table

IW operation / program: If applicable, edit a zero offset table (ZO table)

		Instruction chapter "Handling Instruction: Activate the zero offset (ZO)" on page 531
Instruction:		Editing a ZO table

IW operation / NC programming: Activation of zero offsets

The following NC program line activates a zero offset table in the channel.

ZoTSel({<Path/}Table name>{.zot}) or **ZOS({<Path/}Table name>{.zot})** or

With MTX standard names according to name convention **(ZO<int value>.zot)**:

ZoTSel(int value) or **ZOS(int value)**

Example: For the zero point offset table 5

ZoTSel(zo5.zot) or **ZOTSel(5)**

Activation of the desired zero offset with the respective NC function **G54.1...G59.5**, with the short variant of the NC function **G54...G59**, the respective offset of the ZO base 1 is activated.

Tool must no longer be active.

Any other still active offsets must be taken into consideration when a zero offset is programmed as they may act additively or overwrite each other.

Offsets from different ZO pages always work additively.

Offsets within a ZO page mutually overwrite each other.

		Documentation
Documentation:	MTX NC Programming Instructions	Zero offset

IW operation / program: If applicable, deactivate / delete the zero offset (ZO)

		Instruction chapter "Handling Instruction: Activate the zero offset (ZO)" on page 531
Instruction:		Deactivation / Deletion of a ZO table

Handling Instruction: Zero Offsets (ZO) Table: Creating/Deleting/Editing an Axis

Creating/deleting/editing axes for zero offsets using the zero offsets editor.

IW-Operation/program: Create/Delete/Edit an Axis

1. Use the F-key ">>" <F8> to switch the F-key menu.

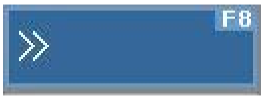


Fig.22-15: Next/More key

2. Select the F-key "Extras" <F7> to open the "Edit axes" menu.

Fig.22-16: Extras key

3. Use the F-key "Insert axis..." <F2> to add a new axis of the zero point offset page/group.

   – or –

4. Use the F-key "Delete axis..." <F3> to delete an axis from the zero offset page/group by selecting the desired axis.

   – or –

5. Use the F-key "Edit axis..." <F3> to edit an axis in the zero offset page/group by selecting the desired axis.

Fig.22-17: Creating/Deleting/Editing axis keys

		Documentation
Documentation	IndraWorks HMI	Zero point offsets

Handling Instruction: Zero Offsets (ZO) Table: Adding/Deleting a Correction Page

Adding/deleting a correction page for zero offsets using the zero offset editor.

IW Operation / Program / ZO Editor: Create/Delete a Correction Page

1. Use the F-key "Add correction page" <F2> to add a new zero point offset page/group.

   or

2. Use the F-key "Delete correction page" <F3> to delete a legacy zero offset page/group by selecting the desired page.

Fig.22-18: Adding/Deleting a correction page

		Documentation
Documentation:	IndraWorks HMI	Zero point offsets

Handling Instruction: Delete a Zero Offset (ZO) Table

Deleting a ZO table in IW Operation.

IW-Operation/program: Delete a File

1. Start IW Operation.
2. Select "Program" mode using OP key **Program (OP4)**

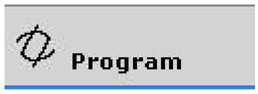


Fig.22-19: OP key “program”

3. Navigate to the directory that contains the zero offset file to be deleted.

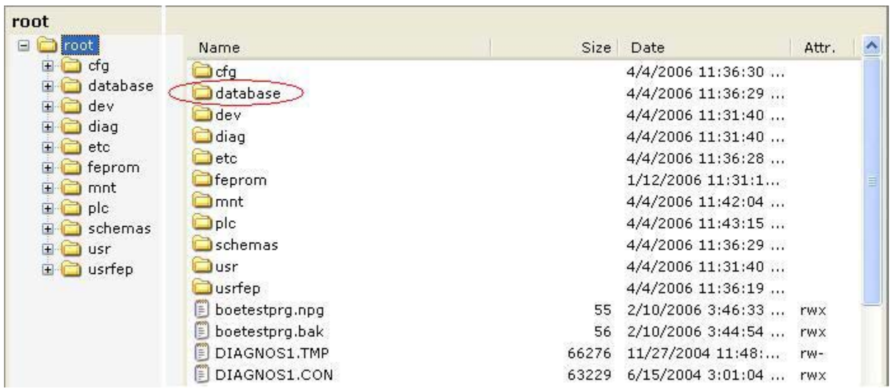


Fig.22-20: Path overview in mode “program”

4. Select the desired zero offset file with the extension "*.zot", but do not open it.
5. Select the F-key "Delete" (F3).

Fig.22-21: Delete key

The "MTX Navigator" dialog appears. In the dialog, check whether the correct file has been selected.
If "Yes", confirm the Delete procedure.
If not, cancel with "No" and select a different zero offset file.

Operation Desktop

Fig.22-22: Confirming deletion

		Documentation
Documentation	IndraWorks HMI	Zero point offsets

22.2 Placement Table

22.2.1 General

Brief Description

In addition to the zero point table, placement tables can be created. As opposed to zero offsets, placement corrections do not affect the machine coordinate system (MCS), but rather the basic coordinate system / workpiece coordinate system (BCS/WCS).

22.2.2 Creating a Placement Table

General

Description

In the first step, specify the name and determine whether a template is to be used. Placement tables have a file extension of "*.pmt" and are stored in directory "\database" under the root directory. The table can also be saved to any directory. The directory must be selected before the table is created.

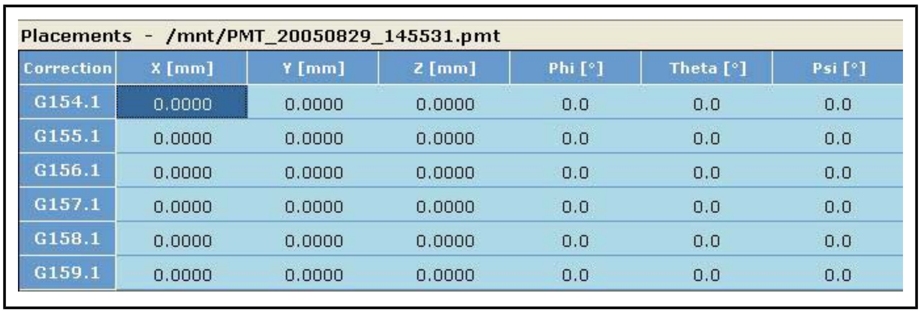
Fig.22-23: Creating a new placement table

The offset values and rotation angles for the inclined plane can be preset via so-called placement tables.

A maximum of 5 correction pages with 6 placements can be configured:

- G154.1 to G159.1
- G154.2 to G159.2
- G154.3 to G159.3

- G154.4 to G159.4
- G154.5 to G159.5



Placements - /mnt/PMT_20050829_145531.pmt

Correction	X [mm]	Y [mm]	Z [mm]	Phi [°]	Theta [°]	Psi [°]
G154.1	0.0000	0.0000	0.0000	0.0	0.0	0.0
G155.1	0.0000	0.0000	0.0000	0.0	0.0	0.0
G156.1	0.0000	0.0000	0.0000	0.0	0.0	0.0
G157.1	0.0000	0.0000	0.0000	0.0	0.0	0.0
G158.1	0.0000	0.0000	0.0000	0.0	0.0	0.0
G159.1	0.0000	0.0000	0.0000	0.0	0.0	0.0

Fig.22-24: Placement correction page

Additional information regarding the creation, activation and deactivation of placements can be found in the "IndraMotion MTX" Functional Description.

Handling Instruction: Placement Table

The placement "Inclined Plane" can shift and orientate the workpiece coordinate system anywhere in space. The inclined plane affects the coordinates with the meanings "X", "Y" and "Z" in the corresponding channel. A maximum of 5 correction pages with 6 placements can be configured.

Since there are 3 degrees of freedom for orientation, every orientation can be mathematically represented by 3 consecutive basic rotations.

IW-Operation/program: Create a placement file

1. Select "Program" mode using OP key **Program (OP4)**.

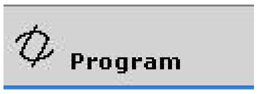


Fig.22-25: OP key "program"

2. Navigate to the desired directory or create a new one if necessary.

Fig.22-26: Root directory

3. Select the F-key "New" <F2>.

Fig.22-27: F-key

4. Select "Placements" <F8> from the F-keypad.

Fig.22-28: F-key

The "Create new placement table" dialog appears.

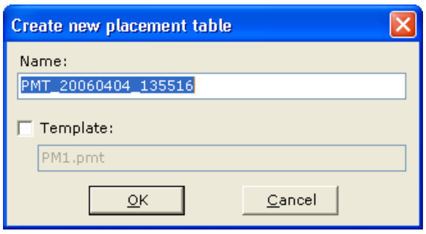


Fig.22-29: Dialog

5. Assign a name for the new file.

   The placement editor opens.

		Documentation
Documentation	MTX Standard NC Operation	Create a placement file

IW Operation / Program / Placement Editor: Create/Delete a Correction Page

1. Use the F-key "Add correction page" <F2> to add a new placement page/group.

   – or –

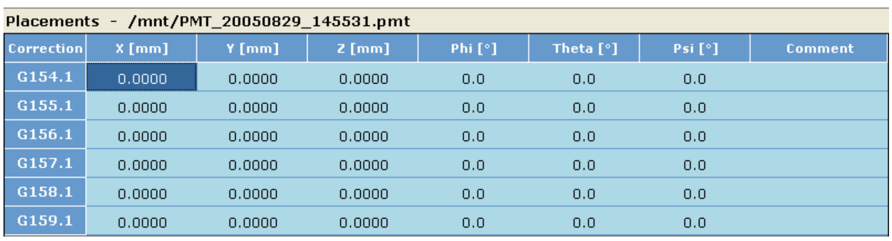
2. Use the F-key "Delete correction page" <F3> to delete a legacy placement page/group by selecting the desired page.

Fig.22-30: Adding/Deleting a correction page

		Documentation
Documentation	MTX Standard NC Operation	Create/Delete a Correction Page

IW Operation / Program / Placement editor: Enter values

The desired placement can be entered/modified by selecting the desired block.


Placements - /mnt/PMT_20050829_145531.pmt

Correction	X [mm]	Y [mm]	Z [mm]	Phi [°]	Theta [°]	Psi [°]	Comment
G154.1	0.0000	0.0000	0.0000	0.0	0.0	0.0	
G155.1	0.0000	0.0000	0.0000	0.0	0.0	0.0	
G156.1	0.0000	0.0000	0.0000	0.0	0.0	0.0	
G157.1	0.0000	0.0000	0.0000	0.0	0.0	0.0	
G158.1	0.0000	0.0000	0.0000	0.0	0.0	0.0	
G159.1	0.0000	0.0000	0.0000	0.0	0.0	0.0	

Fig.22-31: Placement table overview in the placement editor

- Offsets from different placement pages always work additively.
- Offsets within a placement page mutually overwrite each other.

		Documentation
Documentation	MTX Standard NC Operation	Enter values in a placement table

22.3 D-Corrections

22.3.1 General

Brief Description

Any number of D-correction tables can be stored within the file system of the MTX. Every table can consist of a maximum of 99 tool correction blocks with each correction page comprising 3 tool lengths "L1", "L2", "L3", the tool radius "Rad" as well as the tool edge orientation "Ori". The structure of the D-correction tables is XML-based. In the MTX user interface, the individual table elements can be comfortably edited with a Table Editor. Alternatively, the CPL command "DCT" offers the possibility of writing or reading individual table elements directly from the part program (documentation "Rexroth IndraMotion MTX Functional Description" and "NC Programming Instructions").

22.3.2 Creating a D-Correction Table

General

Description

Below the OP key 4 **Program** D-correction tables can be created.

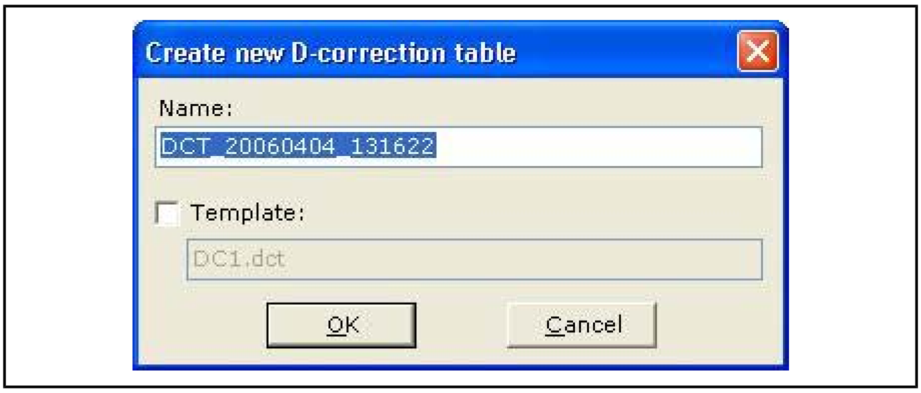


Fig.22-32: New D-correction table

The file extension of D-correction tables is "*.dct". The table is saved to directory "\database" under the root directory, but can be saved to any directory; however, the path must be specified when activating the table.

D-corrections - /mnt/DCT_20050829_154003.dct

Correction	L1 [mm]	L2 [mm]	L3 [mm]	R [mm]	Edge position
D1	0.0000	0.0000	0.0000	0.0000	0
D2	0.0000	0.0000	0.0000	0.0000	0
D3	0.0000	0.0000	0.0000	0.0000	0
D4	0.0000	0.0000	0.0000	0.0000	0
D5	0.0000	0.0000	0.0000	0.0000	0
D6	0.0000	0.0000	0.0000	0.0000	0
D7	0.0000	0.0000	0.0000	0.0000	0

Fig.22-33: D-correction table

Additional information regarding the creation, activation and deactivation of D-corrections can be found in the "IndraMotion MTX Functional Description".

Handling Instruction: Creating a D-Correction Table

D-corrections are equally suitable for drilling, milling, lathing and cross-staff tools. With a total of 3 offset values (L1, L2 and L3), you can perform both constant three-dimensional tool offsets for a tool and parallel length corrections of 3 different tools as a maximum.

IW-Operation/program: Create a D-Correction Table

1. Select "Program" mode using OP key **Program (OP4)**.

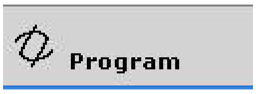


Fig.22-34: OP key "program"

2. Navigate to the desired directory or create a new one if necessary.

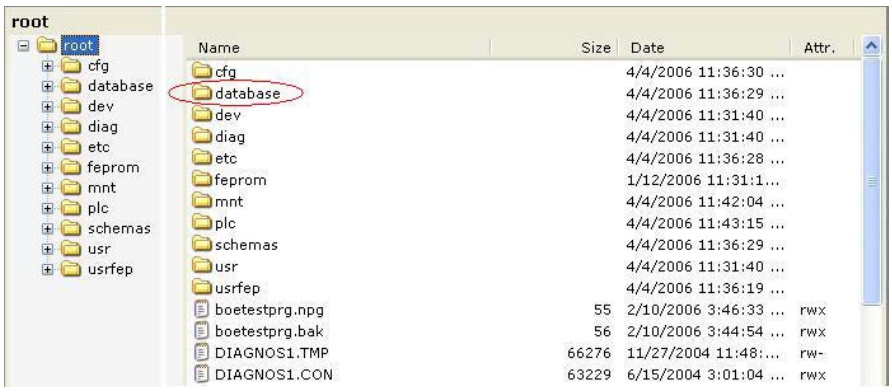


Fig.22-35: Root directory

3. Select the F-key "New" <F2>.

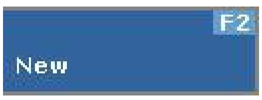


Fig.22-36: F-key

4. Select the "D-corrections" <F6> from the F-keypad.

Fig.22-37: F-key

The "Create new D-correction table" dialog appears.

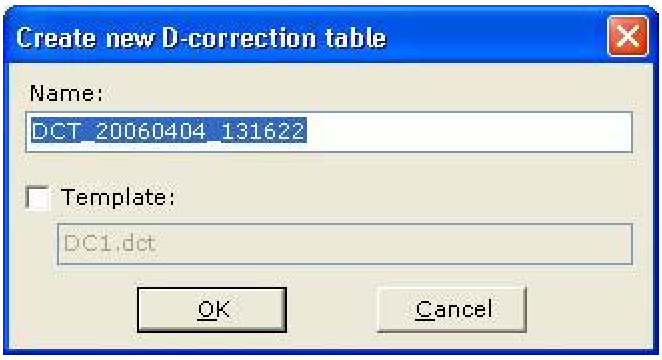


Fig.22-38: “Create new D-correction table” dialog

5. Assign a name for the new file

The D-correction editor opens.

		Documentation
Documentation	IndraWorks HMI	D-correction table

IW Operation / Program / D-Correction Editor: Create/Delete a Correction Page

1. Use the F-key "Add correction page" <F2> to add a new block (D-correction page/group)

   – or –

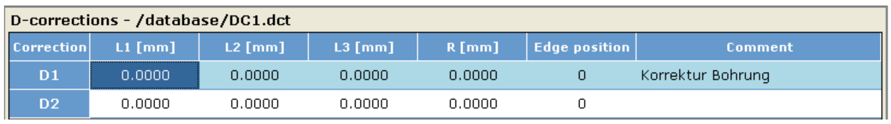
2. Use the F-key "Delete correction page" <F3> to delete a legacy block (D-correction page/group) by selecting the desired block.

Fig.22-39: Adding/Deleting a correction page

		Documentation
Documentation	IndraWorks HMI	D-correction table

IW Operation / Program / D-correction editor: Enter values

The desired D-correction can be entered/modified by selecting the desired block.


D-corrections - /database/DC1.dct

Correction	L1 [mm]	L2 [mm]	L3 [mm]	R [mm]	Edge position	Comment
D1	0.0000	0.0000	0.0000	0.0000	0	Korrektur Bohrung
D2	0.0000	0.0000	0.0000	0.0000	0	

Fig.22-40: Overview of D-correction tables in the D-correction editor.

- D-corrections overlap additively.
- A correction table can contain a maximum of 99 data records. Each data record contains the following correction values:
  - 3 tool lengths L1, L2, L3,
  - Tool edge radius RAD,
  - Tool edge position ORI.
- A D-correction may be programmed in the same block as other path conditions, traveling movements or auxiliary functions.
- The tool correction is calculated only if the corresponding NC function has been activated: G47, G41, G42, G141, G142.

		Documentation
Documentation	IndraWorks HMI	D-correction table

22.4 Variable List

22.4.1 General

Description

The "Variable List" function allows the user to form a list of variables that are required very often in the application out of the total number of variables that are available. This selection makes it possible to access the desired variables very quickly. If this selection did not exist, all the variables would have to be queried and displayed each time, which wastes time. The schema files that contain the variables can be edited with the MTX Schema Editor. The Schema Editor can be found in the IndraWorks installation directory (e.g. "...\Programs \Rexroth\IndraWorks\ MTX.Schema.Editor.exe").

22.4.2 Overview

Editor for the Variable List

The Variable Editor can be used to display and edit the following variables:

- Permanent CPL variables (@)
- System data (SD)

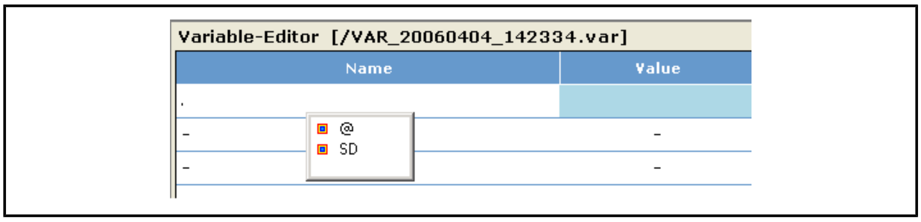


Fig.22-41: Variable editor

Screen Layout

Structure

The variable editor consists of the columns "Name", "Value" and "Comment".

Header

The name of the Variable Editor and the list name, including the path specification, are displayed in the title line.

If the list of all active variables is displayed, the list name is "ControlVariables".

Main Area

The variables are displayed in the main area of the Variable Editor.

Information Area Information texts and error messages can be displayed in the information area during operation.

Variable Lists

List filing and list access

There are two types of variable lists:

- Display of all variables that exist in the system.
- Display of a self-defined subset of variables.

Display of all variables

An internal list of all existing variables is created. It is called in the operating area **Machine <OP3>**.

Display of a subset of variables

If a subset of the existing variables is to be displayed, a list of variables is first edited. It is called in the operating area **Program OP4**, F-key **"New"**, F-key **"variable list"**. A self-edited list can be stored in the file system of the control. Use the F-key "Activate" to specify whether this list is to be declared as the **"active list"**.

The data type of a list can be identified by the file extension of the list name (*.var).

> ☞ An active list (all variables or a subset) is always called in the operating area **Machine** via the F-key **"Active Variables"**.

Creating/defining new variables:

Variable definition is carried out via the "wmhperm.dat" or the "anwperm.dat" file. Within this file, global permanent CPL variables are defined as follows:

DEF <variable type> **@**<variable name>

> ☞ For all variable names, the first 16 characters are significant. If the names of variables exhibit a difference only with the 17th character or later, CPL will interpret them as a single variable. After the global permanent CPL variables have been entered in the file, the variables are activated for the next control startup. The number of variables is unlimited.

Adding/editing new variables

After a new line has been added, either a CPL variable or a system variable (or a structured variable) can be added.

Use of the Intellisense control

The Intellisense control is used to add new CPL variables. The control appears when an **"@"** is entered. You can navigate within the control using the <Cursor up/down> or <Page up/down> keys or with the mouse. Confirm selections using <Enter> or by double-clicking. If the selected CPL variable is a field, an additional layer can be displayed by the Intellisense control by entering a <period>. If an additional layer does not exist, the control does not appear.

The Intellisense control is used to add new system variables. The control appears when **"SD."** is entered. You can navigate within the control using the <Cursor up/down> or <Page up/down> keys or with the mouse. Confirm selections using <Enter> or by double-clicking. If the selected system variable is a field or structure, an additional layer can be displayed by the Intellisense control by entering a <period>. If an additional layer does not exist, the control does not appear.

Special features: When entering related fields or structures, it is possible to enter only the first part of the name. The other elements of this variable or structure are then automatically entered in the list.

Example:

Using the Intellisense Control

Array name:

SD.Sys_VAR; the Intellisense control displays SD.Sys_VAR[1,1,1] and the following elements. The entire array is input by deleting the indices (new name: SD.Sys_VAR) and completing the entry.

Structure name:

SD.Sys_SearchRun[1]; this is also shown by the Intellisense control. Since this is the first part of the variables, it can be confirmed with <Enter>. All associated elements are inserted.

If a variable is not specified completely, a message appears in the status line and the value of the variable is filled with **"..."**. This value means that the variable was not specified completely. If the list is activated and displayed as an active list, a value cannot be displayed for this variable and it cannot be edited. Any attempt to carry out editing will be refused.

☞ If the help of the Intellisense control is not required, it is hidden by means of the Intellisense Control and you can continue editing.

Handling Instruction: Creating a Variable List

This handling instruction describes how to create a variable list.

IW-Operation/program: Open Variable Editor

1. Start IW Operation
2. Select "Program" mode using OP key **Program OP4**.

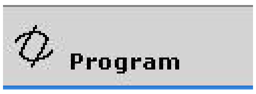


Fig.22-42: OP key “program”

3. Navigate to the desired directory or create a new one if necessary.

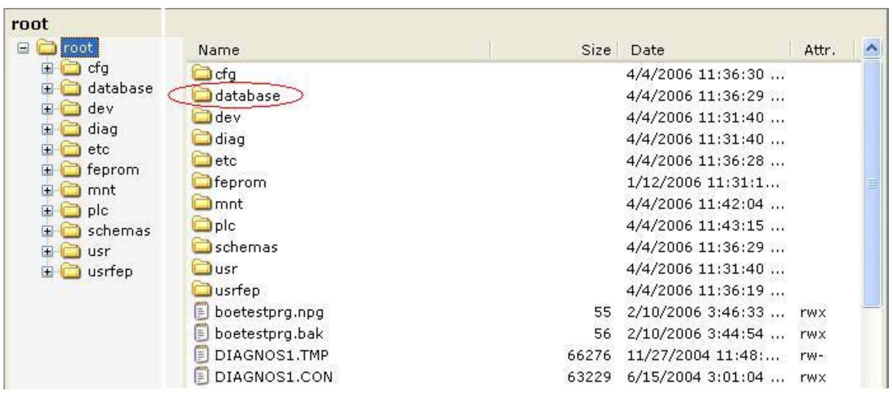


Fig.22-43: Path overview in mode “program”

4. Select the F-key "New" <F2>.

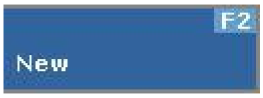


Fig.22-44: F-key

5. Select "Variable List" <F5> from the F-keypad.

Fig.22-45: F-key

The Variable Editor opens.

☞ Variable lists have the extension "*.var".

		Documentation chapter 22.4 "Variable List" on page 540
Documentation	IndraWorks Commissioning	Open variable editor

IW Operation / Program / Variable Editor: Add/delete/insert a line

		Instruction chapter "Handling Instruction: Variable Editor: Add/Delete/Insert a Line" on page 544
Instruction		Add/delete/insert a line

IW Operation / Program / Variable Editor: Entering a CPL variable

1. In column "Name", go to the desired field and enter the **"@"** symbol (<<Alt Gr + Q>>).

   The Intellisense control appears.

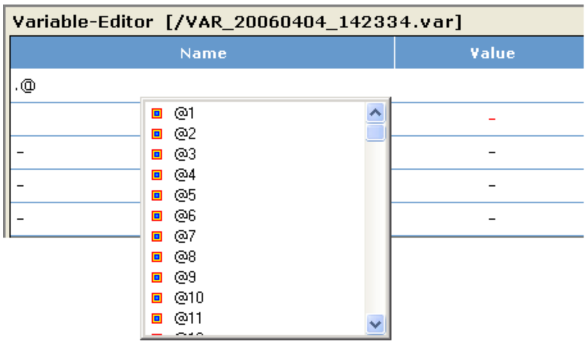


Fig.22-46: Entering a CPL variable

2. Use the keys <Cursor up/down>, <Page up/down> as well as the **mouse** to navigate in the Intellisense Control and select the desired CPL variable by pressing <Enter> or by **double-clicking**.

☞ If the selected CPL variable is a field, an additional layer can be displayed by the Intellisense control by entering a <period>. If an additional layer does not exist, the Intellisense control does not appear.

		Documentation
Documentation:	MTX: Standard NC Operation	Variable editor

IW Operation / Program / Variable Editor: Entering system data

1. In the column "Name", go to the desired field and enter **"SD" +** <period>.

   The Intellisense control appears.

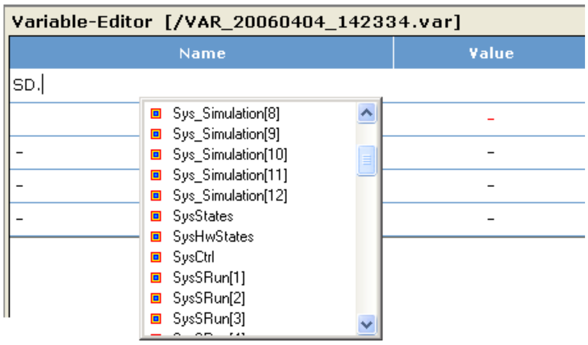


Fig.22-47: Entering system data

2. Use the keys <Cursor up/down>, <Page up/down> as well as the **mouse** to navigate in the Intellisense Control and select the desired CPL variable by pressing <Enter> or by **double-clicking**.

If the selected system data is represented by a field, an additional layer can be displayed by the Intellisense control by entering a <period>. If an additional layer does not exist, the Intellisense control does not appear.

The variable list is saved when the Variable Editor is exited.

		Documentation
Documentation:	MTX: Standard NC Operation	Variable editor

Handling Instruction: Variable Editor: Add/Delete/Insert a Line

This handling instruction describes how to add/delete/insert a line in the Variable Editor.

IW Operation / Program / Variable Editor: Add a line

Press the F-key "Add line" <F2> to insert a new variable line at the end.

Fig.22-48: F-keys for editing a line

		Documentation
Documentation:	MTX System Description	Add line

IW Operation / Program / Variable Editor: Delete line

1. Select the line that is to be deleted.

2. Delete the line using the F-key "Delete line" <F3>.

Fig.22-49: F-keys for editing a line

3. Check and confirm the "Delete" dialog.

Fig.22-50: “Confirm variable deletion” dialog

		Documentation
Documentation:	MTX System Description	Delete line

IW Operation / Program / Variable Editor: Insert line

1. Select the line after which a line is to be inserted.
2. Press the F-key "Insert line" <F4> to insert a new variable line after the selected variable line.

Fig.22-51: F-keys for editing a line

		Documentation
Documentation:	MTX System Description	Insert line

Handling Instruction: Activate/Deactivate the Variable Editor

This handling instruction describes how to activate and deactivate a desired variable list.

IW-Operation/program: Open a Desired Variable List

1. Start IW Operation
2. Select "Program" mode using OP key **Program (OP4)**.

Fig.22-52: OP key “program”

3. Navigate to the desired directory.
4. **Select** the desired variable list (with the extension *.var) and call it by **double-clicking** it or by pressing <Enter>.

Fig.22-53: Path overview in mode "program"

		Documentation
Documentation:	IndraWorks HMI	Open a variable list

IW Operation / Program / Variable Editor: Activate the desired list

In the Variable Editor, activate the opened list using the F-key "Activate" <F7>.

Fig.22-54: F-key

		Documentation
Documentation:	IndraWorks HMI	Activate variable list

IW Operation / Program / Variable Editor: Deactivate the desired list

In the variable editor, deactivate the opened list using the F-key "Deactivate" <F6>.

Fig.22-55: F-key

When selecting the active variable list in the operation mode "Machine", all variables are displayed. This may take a long time. We recommend that you always create a variable list with the desired variables and activate it.

		Documentation
Documentation:	IndraWorks HMI	Deactivate a variable list

Handling Instruction: Display an Active Variable List

This handling instruction describes how to display an active variable list in operation mode "Machine".

IW Operation / Program / Variable Editor: Activate a variable list (if necessary)

We recommend that you activate a variable list with a selection of the desired variables.

When selecting the active variable list in the operation mode **Machine** all variables are displayed. This might take a while. We recommend that you always create a variable list with the desired variables and activate it.

		Instruction chapter "Handling Instruction: Activate/ Deactivate the Variable Editor" on page 545
Instruction:		Activate variable list

IW Operation / Machine: Display an Active Variable List

1. Start IW Operation
2. Select "Program" mode via OP key **Machine (OP3)**.

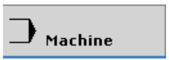


Fig.22-56: OP key

3. Display the active variable list with F-key "Active Variables" (F7).

Fig.22-57: F-key

4. Make the desired selection in the lower right corner to be able to switch back and forth between the NC screen and the Variable Editor.
5. The display of active variables can be exited by pressing F-key "Exit Var. Editor" <F9>.

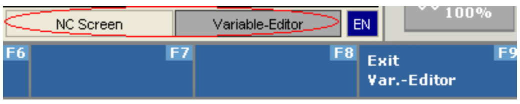


Fig.22-58: Switching between the NC screen and the variable editor

If no variable list was activated beforehand, all the variables are displayed; this may take a long time.

		Documentation
Documentation:	IndraWorks HMI	Display an active variable list

23 Drive-Integrated Safety Technology

23.1 Basic Method of Functioning

The commissioning of the integrated safety technology is briefly described in this chapter. An extensive explanation, with additional example applications, can be found in the documentation **"Rexroth IndraDrive Integrated Safety Technology – Description of Functions and Application"** (Mat. No. R911297838).

Description

As regards the use of safety technology, we distinguish between the

- **normal operation** and the
- **special mode of operation**.

In the special mode, the following is possible:

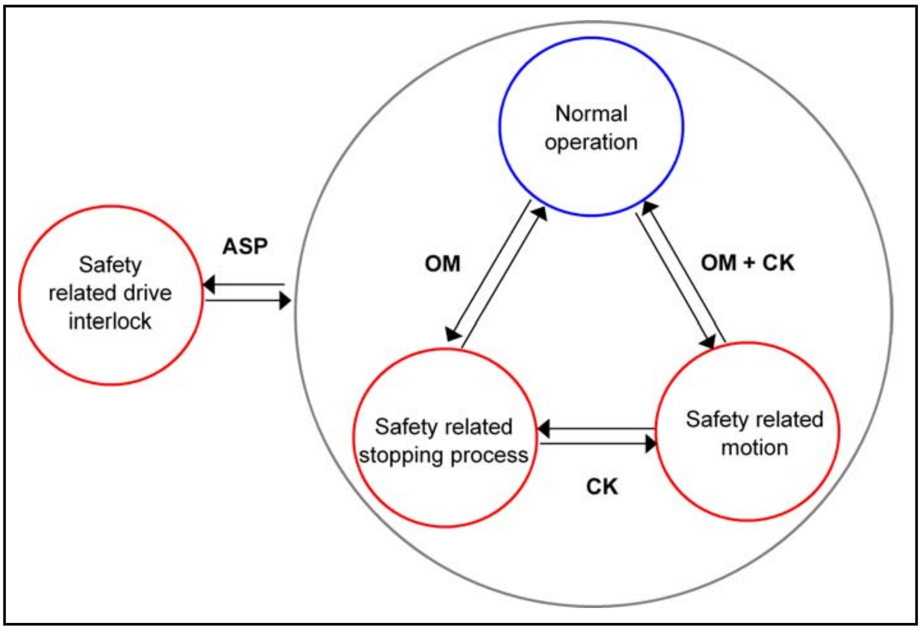
a special mode with standstill ("Safe standstill")

and of

a special mode with movement ("Safe movement")

"Safe stop" or "Safe operation stop" can be engineered for special mode with standstill.

Regardless of whether the operation operation is normal or special, **Safe drive lock** can also be used for Safe stop:



OM	Special operating mode
CK	Acknowledgement button
ASP	Drive lock
Fig.23-1:	*Safety technology operating modes*

23.2 Overview of the Operating Modes

23.2.1 "Safe Standstill" Safety Functions

Safe Stop

"Safe stop" corresponds to stop category 1 according to EN 60204-1.

The drive can not generate torque/force and thus can not generate dangerous movements. Monitoring is not active for “Safe stop”.

For the “Safe stop”, "SH" is shown on the display of the Rexroth IndraDrive control device.

Safe Operation Stop

"Safe stop" corresponds to stop category 2 according to EN 60204-1.

In safety function "Safe operation stop", a dual-channel monitor prevents the drive from carrying out dangerous movements due to errors.

For the “Safe operation stop”, "SBH" is shown on the display of the Rexroth IndraDrive control device.

Safe Drive Lock

"Safe drive lock" corresponds to stop category 1 according to EN 60204-1.

Safety function "Safe drive lock" is the same as "Safe stop"; however, it is not cancelled by pressing a consent device.

When the drive lock is active, "ASP" is shown on the display of the Rexroth IndraDrive control device.

This is used, for example, in spindle drives to exchange tools manually and to handle axes for movements by hand.

23.2.2 Safety Functions "Movement with Safe Velocity"

Safely Reduced Velocity

In safety function "Safely reduced velocity", a dual-channel monitor prevents the drive from exceeding the specified velocity limit values (P-0-3244, P-0-3254, P-0-3264, P-0-3274).

When the movement lock is active, "SBB" is shown on the display of the Rexroth IndraDrive control device.

Movement is enabled by a consent key (CK). The activation time of the consent device is monitored.

☞ All further safety functions are discussed in the documentation "Integrated Safety Technology" (Mat. No. R911297838).

For the special mode with movement, the user can switch between up to 4 safe movement operating modes during operation, using 2 process selection signals (= safety switches):

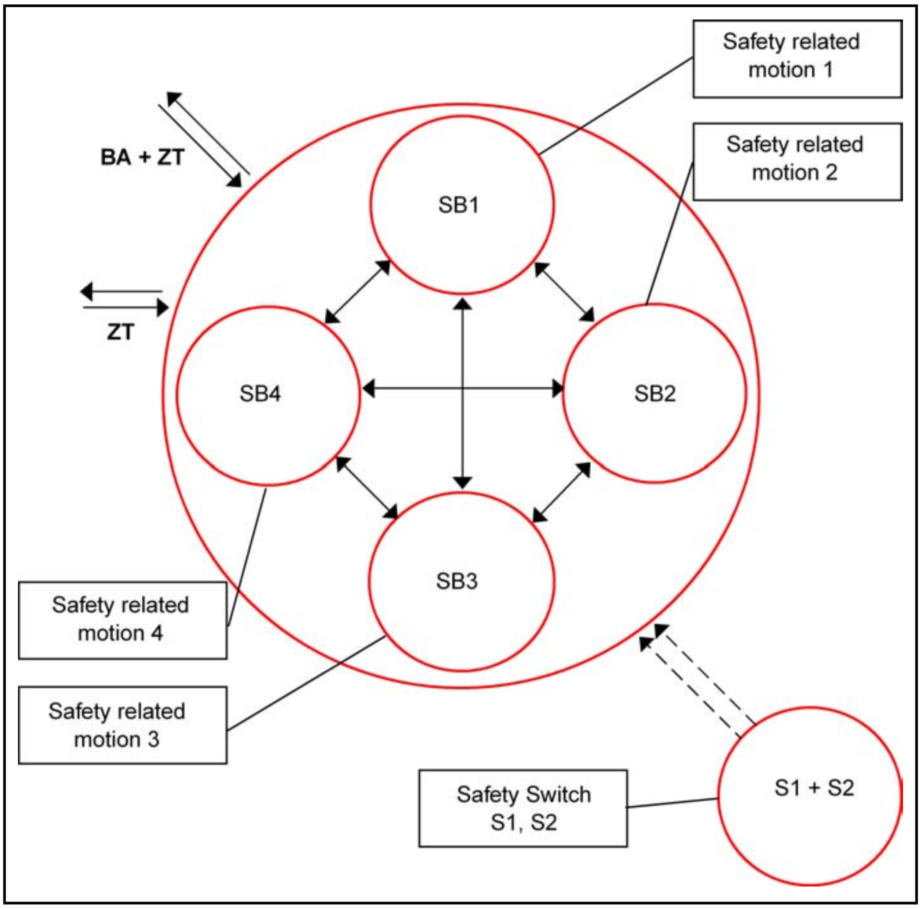


Fig.23-2: Subordinate operating modes – safe movement

23.3 Example: Installation and PLC Interface

23.3.1 Application Structure

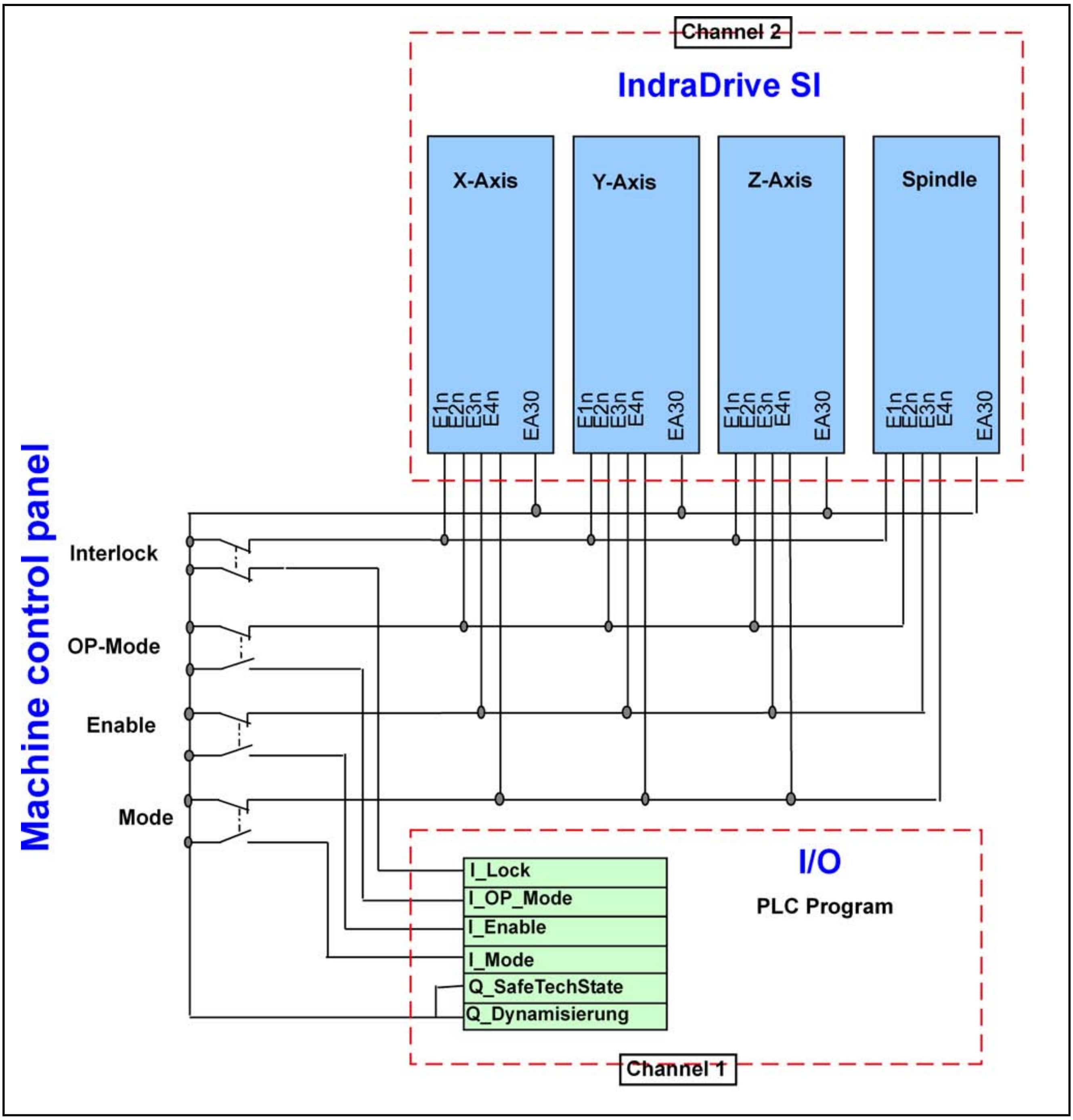


Fig.23-3: Wiring

Example The safe 2-channel activation of the safety technology occurs in channel 1 using a standard input of the PLC I/O level and in channel 2 directly by the safety module of the drive control devices (see previous figure).

Since the safety functions should be effective for all axes and spindles simultaneously, the input signals of all control devices (signals of safety module channel 2) are to be cross-connected.

For channel 1 (PLC I/O level), the safety signals can be passed on in the PLC user program. Therefore, a total of only one input per input signal is required for all the axes.

Due to the dynamization of the signals that is required for safety technology, all sensors for activating the safety functions are to be supplied using dynamization signal "Q_Dynamization".

The signal that is required for this purpose is to be generated in the PLC user program and provided for all drives via IO30.

When the dynamization signal is generated, observe the dynamization limit values parameterized in the IndraWorks Drive (period duration and pulse duration) under consideration of the runtimes in the PLC and peripherals. If the values are exceeded, the drives are switched off and an error is issued.

☞ It is recommended that dynamization be executed as "isolated dynamization". As a result, the dynamization pulse can be set to 50 ms, which shortens the reaction times.

23.3.2 PLC Program Part

Assign the signals as follows in the PLC user program:

Safety-oriented machine function	Abbr.	I/O signal
Safe drive lock (for EMERGENCY STOP functions)	ASP	qax_SafDrvLock
Special mode (for open safety equipment)	OM	qAx_SafOpModeSwitch
Consent key (for movement when safety equipment is open)	CK	qAx_SafEnablCtrl
Switching to safe movement	S1	qAx_SafSwitch1
Dynamization	DYN	qAx_SafTecState

Fig.23-4: Interface signals

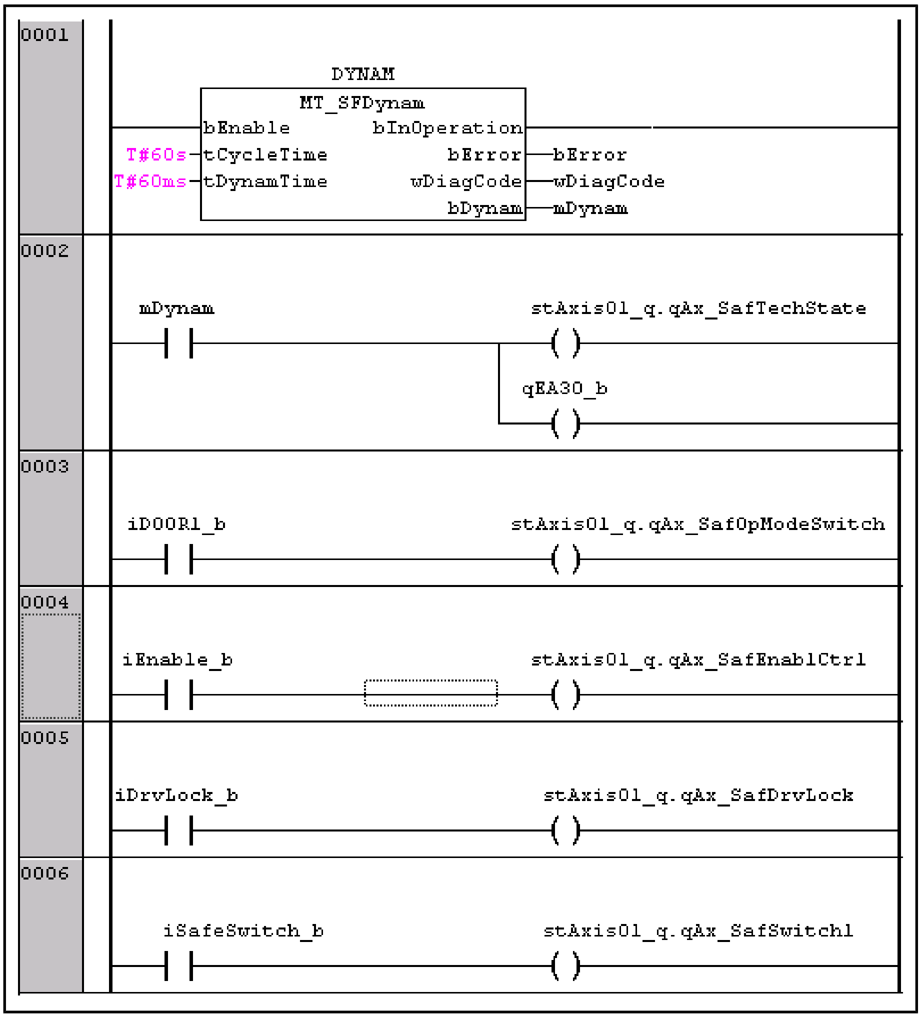


Fig.23-5: PLC program example

23.3.3 PLC Configuration

General

In the SCS files for axes with safety technology, make the divergent setting for the parameters for SERCOS communication depending on the axis type:

Excerpt from an SCS file:

Program:

```
;--------------------------------------------------------------
; configuration of cyclic telegram
S-0-16  = (S-0-51,S-0-144,P-0-3215)
;           |      |       -safety signals
;           |      -signal status word
;           -AT: actual position value 1
;
S-0-24  = (S-0-47,S-0-0145)
; MDT
;           |      -signal control word
;           -position command value
;
;--------------------------------------------------------------
; signal status word
S-0-26  = (S-0-403,P-0-3213,P-0-3214)
;           |       |        -SI-signal status word
;           |       -SI status
;           -reference bit
```

```
; Bit numbers
S-0-328  = (0,7,0)
;           | | -Bit number
;           | -status safe position
;           -Bit number
;
;------------------------------------------------------------
; signal control word
S-0-27  = (P-0-3212,P-0-3212,P-0-3212,P-0-3212,P-0-3212)
;          |        |        |        |        -denomination
;          |        |        |        -safety switch
;          |        |        -acknowledgement key
;          |        -drive lock
;          -operation mode
; Bit numbers
S-0-329  = (0,1,2,3,10)
;           | | | | -denomination
;           | | | -safety switch
;           | | -acknowledgement key
;           | -drive lock
;           -operation mode
;------------------------------------------------------------
```

Bold parameters must also be configured for safety technology

23.3.4 Handling Instructions

Handling Instruction: Commissioning of Safety Technology

Following is a description of the step-by-step commissioning of the Rexroth IndraDrive safety technology integrated into the drive using Rexroth IndraMotion MTX.

IW Engineering / IndraLogic: Adapting the PLC Program

		Instruction chapter "Handling Instruction: Adapting the PLC Program" on page 556
Instruction:		Adapting the PLC Program

IW Operation / Program: Configuring SCS files

		Instruction chapter "Handling Instruction: Configuration of SCS Files" on page 557
Instruction:		Configuration of SCS files

IW Engineering / Configuration: Setting the Machine Parameters

		Instruction chapter "Handling Instruction: Setting the Machine Parameters" on page 558
Instruction:		Setting the Machine Parameters

IW Engineering / SERCOS: Activating Safety Technology in the Drive

		Instruction chapter "Handling Instruction: Activating Safety Technology in the Drive" on page 558
Instruction:		Activating Safety Technology in the Drive

Handling Instruction: Adapting the PLC Program

This handling instruction describes the adaptation of the PLC program to activate the safety technology.

IW Engineering / IndraLogic: Insert Interface Signals and Dynamization

1. Double-click the "IndraLogic" node in Engineering Desktop.

   The PLC is opened.

We recommend that programming be carried out in a separate subroutine in programming language LD. This should be kept as brief as possible.

---

2. To generate the dynamization impulse, incorporate the block "FB Dynam" in the PLC program and activate it according to the times desired for dynamization.

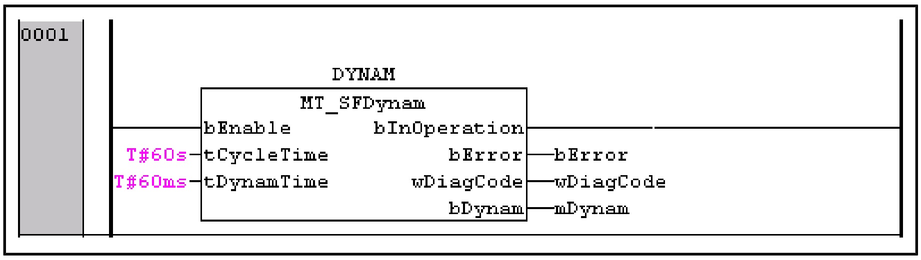


Fig.23-6: FB_Dynam

3. Furthermore, the interface signals for selecting the desired safety functions are to be wired in the PLC program.

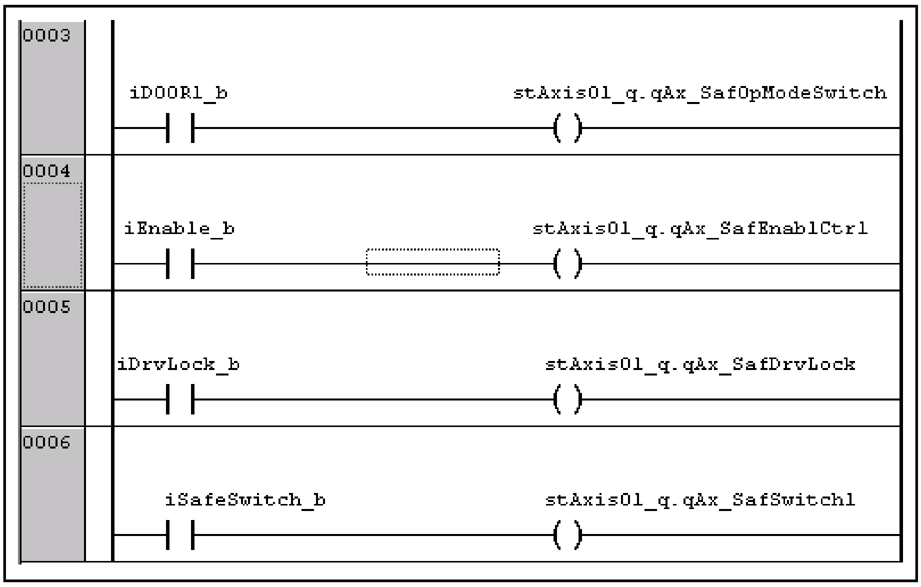


Fig.23-7: Interface signals

4. The dynamization impulse must also be transferred to the PLC interface. On the outside, the dynamization output serves as the supply for the selection signals of the 2nd channel.

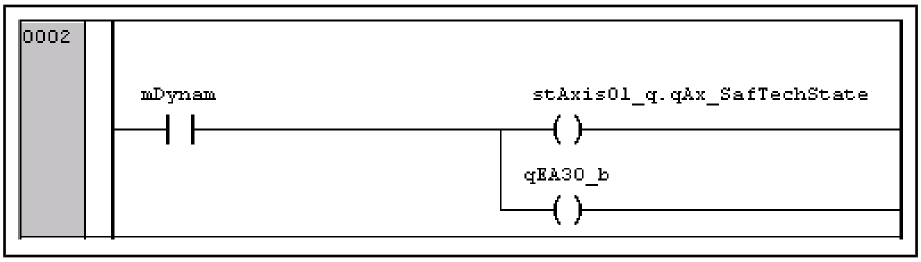


Fig.23-8: Dynamization interface

		Documentation
Documentation:	IndraLogic Programming	Safety systems

Handling Instruction: Configuration of SCS Files

This handling instruction describes the changes that must be made to the SCS files to activate the safety technology.

IW Operation Desktop / Program: Adapt SCS files

Open the SCS files and enter the parameters to be transferred.

☞ The SCS files are located in root or userfep.

Example excerpt from an SCS file:

Program:

```
;---------------------------------------------------------------
; configuration of cyclic telegram
S-0-16  = (S-0-51,S-0-144,P-0-3215)
;          |      |       -safety signals
;          |      -signal status word
;          -AT: actual position value 1
;
S-0-24  = (S-0-47,S-0-0145)
; MDT
;          |      -signal control word
;          -position command value
;
;---------------------------------------------------------------
; signal status word
S-0-26  = (S-0-403,P-0-3213,P-0-3214)
;          |       |         -SI-signal status word
;          |       -SI status
;          -reference bit
; Bit numbers
S-0-328  = (0,7,0)
;           | | -Bit number
;           | -status safe position
;           -Bit number
;
;---------------------------------------------------------------
; signal control word
S-0-27  = (P-0-3212,P-0-3212,P-0-3212,P-0-3212,P-0-3212)
;          |        |        |        |        -denomination
;          |        |        |        -safety switch
;          |        |        -acknowledgement key
;          |        -drive lock
;          -operation mode
; Bit numbers
S-0-329  = (0,1,2,3,10)
;           | | | | -denomination
;           | | | -safety switch
;           | | -acknowledgement key
```

```
;           | -drive lock
;           -operation mode
;-----------------------------------------------------------------
```

		Instruction chapter "Handling Instruction: Configuration of SCS Files" on page 557
Instruction:		Create SCS files

Handling Instruction: Setting the Machine Parameters

This handling instruction describes the configuration of the relevant machine parameters for safety technology.

IW Operation Desktop / Program: Adapt SCS Files

1. The safety technology function must be activated using parameter **EnablSafe** "Safety technology (SAFE)" in Setup (SUP).
2. Select **Select Data Group** and then the new item "Safety Technology (SAFE)".
3. If necessary, set the following parameters for each drive:
   - **EnablSafeTech** set "activate intelligent safety technology" (1001 00002) to "yes" for activation.
   - **SupprSafeTechNc** "suppress safety technology in the NC" (1001 00003)
   - **VelWeightFact** "scaling factor for safe velocities" (1001 00004)

		Documentation
Instruction:	Instruction chapter "Handling Instruction: Activating Safety Technology in the Drive" on page 558	Editing machine parameters
Documentation:	IndraDrive Integrated Safety Technology	Activating safety technology

Handling Instruction: Activating Safety Technology in the Drive

This handling instruction describes the activation of safety technology in the drive using Rexroth IndraWorks Engineering.

IW Engineering / SERCOS: Activate safety technology in the drive

In the last step, the safety technology is activated in the drive using IW Drive and the safety technology wizard.

1. After the drives have been switched online, open folder "Drive-integrated safety technology" in the corresponding drive node.

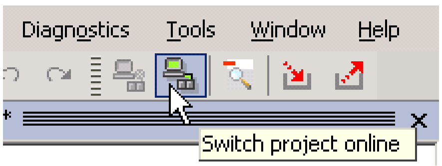


Fig.23-9: Switching online

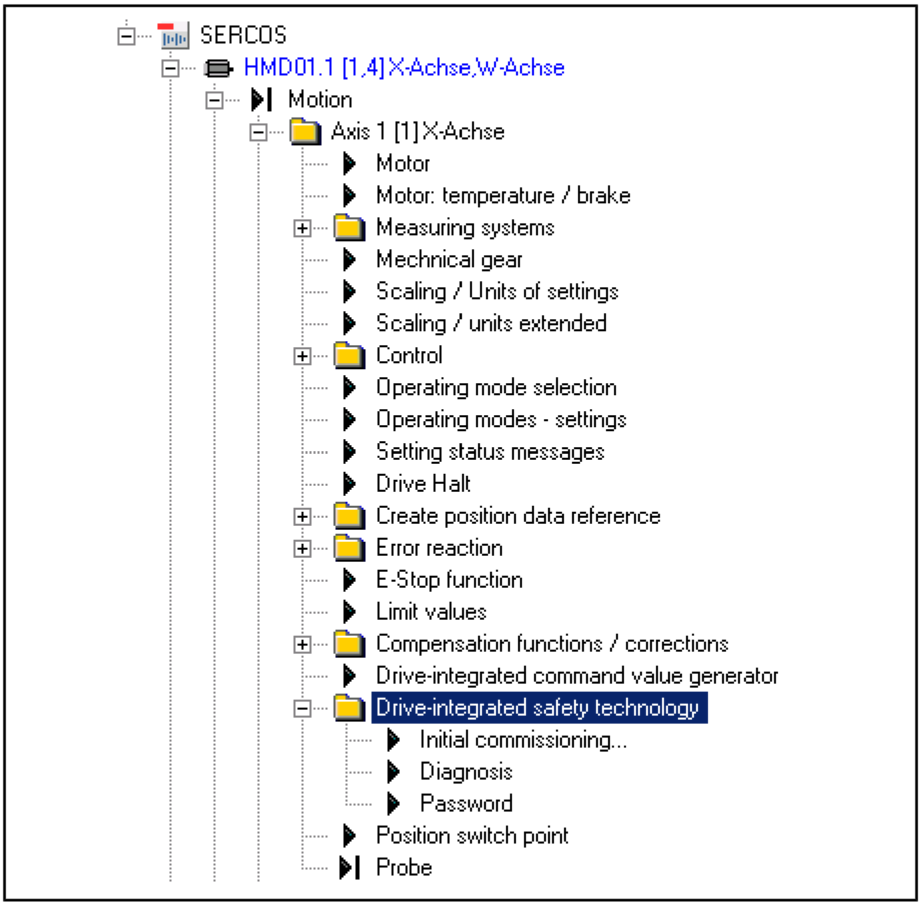


Fig.23-10: Drive-integrated safety technology

2. Double-click "Initial commissioning" to start the safety technology wizard of IW Drive. The wizard guides you through the configuration of the safety technology within the drive. The number of steps to be carried out depends on your specific settings.

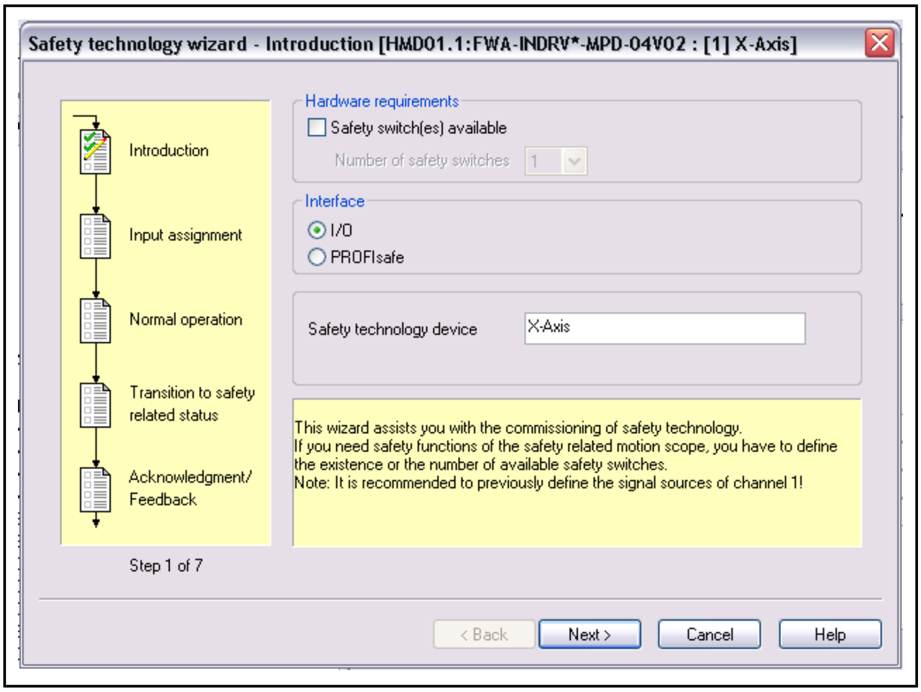


Fig.23-11: Safety technology wizard

		Documentation
Documentation:	IndraDrive Integrated Safety Technology	Activating safety technology

23.4 "NC Ready" Bit (P-0-3212 bit 11)

The transition times have to be long since axes / spindles are sometimes moved within a wide acceleration range / rpm range. To shorten this transition time in case of lower acceleration / revolutions, bit 11 of the P-0-3212 parameter can be used.

If this bit is set, the drive acknowledges a safe state immediately.

The "NC ready" bit is set from the PLC application. The interface signal on the axis interface / spindle interface is **"qAx_SafRedTransTime"** bzw. **"qSp_SafRedTransTime"**. The completion of the command value system adjustment is reported to the drive.

☞ The bit must be reset if the selected safety technology operating state is active or after a constant, application-dependent time.

⚠ WARNING

If the "NC ready" bit is set before the drive has reached its safe operating state selected, the drives are stopped and error messages of the safety technolgy appear.

Also refer to the description of functions and application"Rexroth IndraDrive Integrated Safety Technology" transfer in a safe state.

Using the "NC ready" bits

1. In order to transfer the bit from the control to the drive, a configuration of the signal control word (S-0-0027/S-0-0329) in the SCS file has to be executed.

   S-0-0027 = (....,P-0-3212,...)

   S-0-0329 = (.....,11,...)

2. The interface signal **"qAx_SafRedTransTime"** or **"qSp_SafRedTransTime"** is used to set the bit in the PLC.

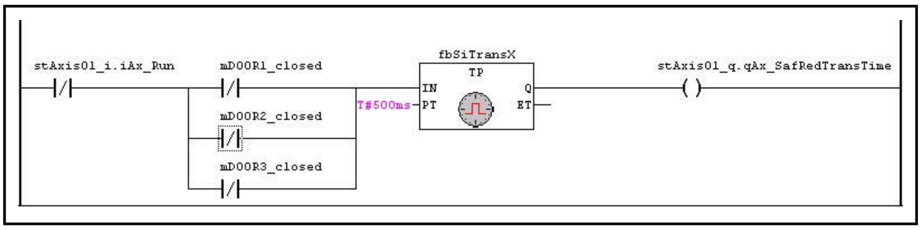


Fig.23-12: PLC program

Schematic representation

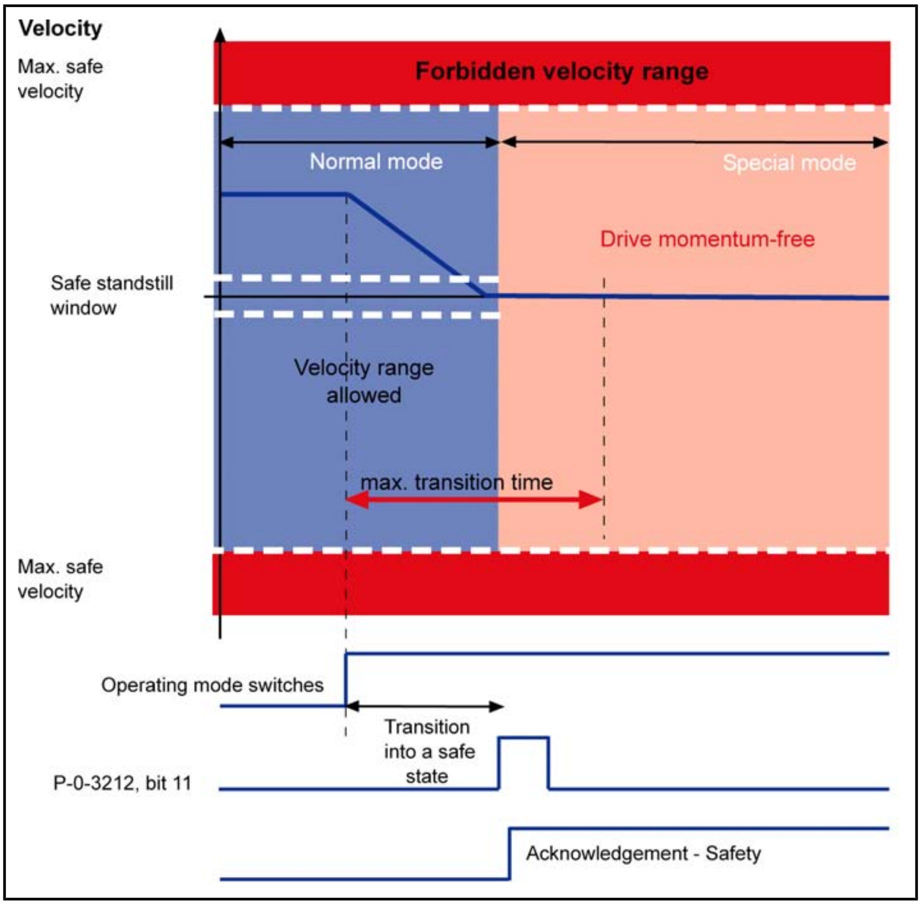


Fig.23-13: Flow chart

⚠ DANGER **Risk of injury**

If the bits are not used properly, dangerous situations might occur.

Example:

A fast rotating spindle should be switched into ASP. The bit is set while the spindle is still rotating with a dangerous velocity.

Consequence:

The drive acknowledges the safe state immediately. The safety technology detects the error, switches off the power and the spindle decelerates till standstill. Now, the guard door might be unlocked even though the spindle is still dangerous.

23.5 Parameterizing Safety Technology in the Drive

Typical report of an axis with safety technology

Normal operation				**OK**
P-0-3234	Safety related maximum speed	41000.000	mm/min	
Transition to safety related status				**OK**
P-0-3210	Transition to safety related status	NC-controlled		
P-0-3220	Tolerance time transition from normal operation	0.5	s	
P-0-3221	Max. tolerance time for different channel states	0.3	s	
P-0-3225	Tolerance time transition from safety rel. oper.	0.2	s	
Acknowledgment/ Feedback				**OK**
P-0-3210	Safety related feedback ...	for control PLC		
Dynamization				**OK**
P-0-3210	Dynamization source	Slave		
P-0-3210	Kind of dynamization	common source		
P-0-3223	Time interval for dynamization of safety function selection	60.0	s	
P-0-3224	Duration of dynamization pulse of safety function selection	0.2	s	
Error reaction				**OK**
P-0-3210	Reaction to F7 error	Velocity command value reset		
Drive interlock				**OK**
P-0-3233	Velocity threshold for safety related halt	10.000	mm/min	
Safety related operational stop				**OK**
P-0-3230	Monitoring window for safety related operational stop	1.0000	mm	
Safety related motion				**OK**
P-0-3239	Max. activation time of enabling control for safety related motions (SBBs)	common		
P-0-3222	Max. activation time of enabling control	30.0	s	
Safety related motion 1				**OK**
P-0-3244	Safety related reduced speed 1	2000.000	mm/min	

Fig.23-14: SI report

The safety technology report provides a clear overview of the safety relevant parameters which are set in the SI Wizzard. The following is a description of how to define useful values for these parameters.

See the description of the individual parameter for a more detailed explanation.

P-0-3234 Safe Maximal Velocity

The parameter defines a velocity limit which is applicable in regular mode and in special mode (SBB, SBH). Accordingly, the maximum permissible velocity for the drive is set here.

P-0-3210 Transition to Safe Status

This parameter is used to configure SI functions. When Safety Technology is used with the MTX, this parameter must be set to "NC controlled". The control reacts to mode changes by reducing velocity or stopping the drive.

P-0-3220 Tolerance Time Transfer from Regular Mode

The parameter defines the maximum time that may elapse before the drive's system of setpoint values must be adjusted to the new safety function for transition from regular mode to a safety function. This time can be calculated as follows:

$$\text{P-0-3220} > \left(\frac{\text{P-0-3234}}{\text{MP101000001}*1000}\right) + t_{control}$$

$$t_{control} = 2*\text{PLCScan} + 2*\text{IPO} + \text{IPO}*\text{SHAPE}$$

Fig.23-15:

Typical value for this parameter: 0.5s - 0.8s

P-0-3221 Max. Tolerance Time Different Channel Statuses

This parameter defines the maximum permissible time for which input and activation signals (SI statuses) of the two monitoring channels may diverge. When this time limit is exceeded, error message **"F3141 Plausibility error activation"** is generated.

The value of this parameter depends on the hardware used and the signal runtimes. The permissible values range is between 0.1s and 2s.

P-0-3225 Tolerance Time Transfer from Safe Mode

The parameter defines the maximum time that may elapse before the drive's system of setpoint values must be adjusted to the new safety function for transition from one safety function into another.

$$\text{P-0-3225} > \left(\frac{\text{P-0-3244}}{\text{ACCEL}*1000}\right) + t_{control}$$

$$t_{control} = 2*\text{PLCScan} + 2*\text{IPO} + \text{IPO}*\text{SHAPE}$$

Fig.23-16:

Typical value for this parameter: 0.2s - 0.5s

In this equation, ACCEL stands for axis acceleration (max. or jog.) The correct acceleration value can be selected after the following operation.

```
IF (MP1010 00001 < MP1010 00002) OR MP1010 00002=0 THEN

ACCEL = MP1010 00001

ELSE
```

```
ACCEL = MP1010 00002
ENDIF
```

MP1010 00001: Maximum axis acceleration (m/s^2)

MP1010 00002: Jog acceleration in (m/s^2)

P-0-3282 Safely Monitored Deceleration

With NC-controlled transitions from regular mode or Special Mode Motion (SBB) into Safe Stop (SH) or Safe Operating Stop (SBH) or Drive Inhibit Activation (ASP), the drive checks whether it can reach standstill or the velocity limit of the selected special mode within the "Tolerance time for transfer" (P-0-3220 or P-0-3225). To this end, it ensures that the deceleration ramp defined in "P-0-3282" is observed.

```
IF (MP1010 00001 < MP1010 00002) OR MP1010 00002=0 THEN
P-0-3282 = MP1010 00001
ELSE
P-0-3282 = MP1010 00002
ENDIF
```

P-0-3210 Source of Dynamization

Source for the dynamization signal. When the Safety Technology is used with the MTX, this parameter must be set to "Slave". The dynamization signal itself must be generated within the PLC.

P-0-3210 Type of Dynamization

The parameter "Type of dynamization" must be set to "Common source".

P-0-3223 Time Interval for Activation of Dynamization

This parameter defines the cycle time in which compulsory dynamization is executed. The value range of this parameter is between 1 and 3600 seconds.

A typical value for this parameter is 60s.

P-0-3224 Duration of the Dynamization Impulse Activation

The parameter defines the maximum duration of the dynamization impulse. The value range is between 0.1 and 2 s.

☞ An externally generated dynamization signal (as is typical for MTX) may be shorter but must not fall below the minimum pulse duration of 30ms.

With a typical value of 0.1 s in the drive, the duration of the impulse can be set to 50ms in the PLC.

The "Duration of the dynamization impulse activation" affects the system's reactivity, as the evaluation of the safety signals is interrupted during dynamization. For this reason, it is not advisable to select a duration value at random.

P-0-3233 Velocity Threshold Safe Stop

This parameter defines a velocity threshold for Special Mode Standstill or for Drive Inhibit Activation.

Typical values:

- Linear axis: 25 - 50 mm/min
- Spindle: 5 rpm

P-0-3230 Monitoring Window for Safe Operating Stop

This parameter defines the maximum permissible traversing path in respect of the actual value available at the time of Safe Operating Stop activation.

Typical values:

- Linear axis: 1mm
- Spindle: 1 degree

P-0-3222 Max. Confirmation Time

The numerical value entered in parameter "P-0-3222 Max. confirmation time" defines the maximum permissible time that may elapse before the confirmation device is operated. The value range is between 0 and 3600s.

Typical values: 30 - 60 s

P-0-3244 Safely Reduced Velocity 1

Parameter "P-0-3244 Safely reduced velocity 1" defines a velocity threshold (bipolar) which is activated at all times in special mode Safe Motion 1 (SBB1).

Typical values:

- Linear axis: 2000 mm/s
- Spindle: 50 rpm

☞ The same applies to the parameters

- P-0-3254 Safely Reduced Velocity 2
- P-0-3264 Safely Reduced Velocity 3
- P-0-3274 Safely Reduced Velocity 4

---

24 IndraLogic

24.1 General

This chapter contains general notes on the handling with the programmed logic control IndraLogic.

24.2 Import of GSD Files

"GSD" stands for **G**eneric **S**tation **D**escription (a.k.a. "device master data") and is a data format for PROFIBUS devices. A defined number of PROFIBUS devices is supplied for the installation of the MTX. These devices are listed in the device library under Peripherals.

If you want to provide a new PROFIBUS device from the device library of IndraWorks, the corresponding GSD file as well as the related bitmap file - if applicable - have to be imported.

Import

The function for importing GSD files can be found in the contextual menu of the Profibus master and is called "Importing GSD files...".. If this function is carried out, a file selection dialog box appears. Here, you can choose a GSD or a BMP file from the file system and launch the import procedure via "Open". After the import, an internal routine runs which interprets the imported files and then adapts the device library of Indra Works.

☞ The import of GSD files is only possible after an IndraWorks project, including a control with Profibus DP functions, has been created/ opened.

24.3 Integrating a PLC Library

You can integrate a series of PLC libraries in your PLC project, the modules, data types and global variables of which you can use in exactly the same manner as self-defined ones. The integration of PLC libraries is carried out via IndraLogic's library manager.

Library Manager

The library manager shows all libraries connected with the current project. The modules, data types and global variables of the libraries can be used just like self-defined modules, data types and global variables. The library manager is opened by means of the command "Window" "Library management" or by selection in the "Resources" tab. The information about the integrated libraries is saved together with the project and can be consulted via the command "Extras" "Properties" if the corresponding entry is marked in the library manager.

Insertion of Further Libraries

This command opens the dialog for opening a file. If the currently set directory does not contain the desired library, you can select a different directory in field "Library directory"; the library files stored there (file type "*.lib") are displayed. Choose the desired library/libraries - multiple selections are possible - and confirm with "OK". The dialog box closes and the library is added to the library manager. You can now use the objects of the library just like self-defined objects.

Library Paths

Please check which library directories are currently defined. If you insert a library from a directory which is not indicated there, the library is entered with the corresponding path indication.

When a project is opened, the libraries entered in the library manager are searched according to the entries available therein. For example, a library entered without any path data is searched for in those library directories that are defined in the project options.

If libraries are not found when a file is opened, you will first be asked if you want to jump to the directory set in the project options. If you do not wish to do this, a dialog box appears giving information highlighted in red on the libraries not found and on the corresponding entries in the library manager. In this case, is is possible to choose the command Search... in the contextual menu, provided a red entry is marked. Using this command, the dialog box for opening a file opens so that you can, if need be, directly load the missing library.

Licensing If you insert a library subject to licensing, you will be notified that the library is only available in the demo mode or that it is not valid for the currently set target system. You can ignore this message or immediately start corresponding measures as regards licensing. Invalid licenses generate an error when the project is transmitted ("Transmit" "project"). Double-clicking the error message or pressing <F4> opens a dialog box "License information", with which you can take appropriate measures with the help of the "wizard".

24.4 Creating a PLC Library

A PLC project can be stored as a library by means of the command "Save as..." in the **File** menu. The project itself remains unchanged; an additional file with the standard extension ".lib" is created and is available afterwards - e.g. the standard library - under the name entered.

In order to be able to use the modules of one project in other projects, it is saved as an internal library "*.lib". This - e.g. the "Standard.lib" - can then be integrated in a different project via the library manager.

☞ Check the possibility of defining via pragmas to what extent the declaration part of the library is displayed after the integration of the library in a project in the library manager ("hiding" variable declarations).

If you want to subject a library to a licensing obligation, press the button "License information" and enter the corresponding data in the "Edit information about licensing" dialog box. See also the description concerning the command "File" "Save as..." or concerning license management in IndraLogic.

24.5 Creating a PLC Task

General In addition to the special program "PLC_PRG", the processing of a project can also be controlled via task management.

- A task is a temporal process unit of the IEC program. It is defined by a name, a priority and a type which defines which condition triggers a start. This condition can be defined either temporally (cycle interval, free running) or by an internal or external event, which, when it occurs, triggers the execution of the task , e.g. the rising flank of a global project variable or an interrupt event of the control.
- A series of programs can be assigned to each task; these are processed when the task is carried out.
- The combination of priority and condition stipulates in which temporal succession the tasks are to be carried out.
- For each task, a timing supervision (watchdog) can be configured; which settings are possible depends on the target system.
- In the online mode, the processing of the tasks can be followed by means of a graphical representation.

- Furthermore, there is the possibility to directly couple system events (e.g. start, stop, reset) with the execution of a project module.

The control of PLC tasks is implemented in the task configuration. The task configuration is an object in the "Resources" tab. The task editor appears in a split window.

"Paste" "Paste a Task"

By means of this command, you can add a new task to the task configuration. Every entry consists of a symbol and the task name.

If a task entry or the entry "System events" in the configuration tree is selected, the command "Paste task" is available. The new task is pasted after the selected task. If the entry "Task configuration" is selected, the command "Attach task" is available and the new task is attached to the end of the existing list.

If a task is pasted, the dialog box for the definition of task properties opens.

Here, you can enter the desired attributes:

Name

A name for the task with which it appears in the configuration tree; the name can also be edited there by clicking or pressing the space bar in order to open an input field.

Priority (0-31)

A figure between 0 and 31, with 0 being the highest and 31 being the lowest priority.

Type

Cyclical:	The task is started cyclically according to the period of time entered in interval.
Free running:	The task starts with the program start and is restarted after each process. There are no instructions concerning the cycle.
Event-driven:	The task is started so that the variable entered in event receives a rising flank.
Externally event-driven:	The task is started as soon as the system event entered in event occurs. The supported events offered in the selection list are specific to the target system and are defined via the target file. **Note:** The system events of externally event-driven tasks should not be mixed up with SystemEvents.
Watchdog:	Activate this option if the task is to be exited with an error status as soon as the watchdog time indicated for the processing under "time" is exceeded (watchdog mechanism).

Time (e.g.: t#200ms): After this period of time has elapsed, the watchdog mechanism is activated if the task was not automatically ended. For the input unit, see above under "interval". It is possible that the target system demands that the watchdog time also be given in percent in relation to the task interval. In this case, the selection window for the unit is dimmed and contains "%".

Sensitivity: Number of overruns of the watchdog time that are accepted without switching the control to an error state.

"Paste" "Paste a Program Call"

Having pasted and defined the task, you can use this command to open the dialog box for the entry of a program call for a task in the task configuration. For "Paste program call", the new program call is pasted before the selected program call; for "Attach program call", the new program is attached to the end of the existing list of program entries.

Enter a valid program name from your project in the "Program field" field or open the input help by means of the "..." button or the <F2> key in order to select valid program names. The program name can also be changed in the configuration tree if the program entry is selected. For this, an editing field is opened either by a mouse-click on the name or by pressing the space bar. If the selected program requires input variables, indicate them in the usual form and of the declared type (e.g. prg(invar:=17)). The processing of the program calls will be carried out later in the online mode according to their order of arrangement from top to bottom.

☞ You should not use the same string functions in several tasks; in this case, processing involves the risk of overwriting elements.

24.6 Creating PLC Objects

Among the objects in IndraLogic are modules (programs, function blocks and functions), data types, visualizations and global variables.

"Project" "Paste an Object"

By means of this command, you can create a new object. The type of this object depends on the selected tab in the object organizer. Please note that a defined template for the selected object type is used for this. This is possible for objects of the type "Global variables", "File type", "Function", "Function module" or "Program".

Enter the name of the new object in the dialog box that opens.

Please note the following restrictions:

- The module name must not contain blanks.
- A module must not be given the same name as a different module or data type.
- A data type must not be given the same name as a different data type or module.
- A list of global variables must not be given the same name as a different list of global variables.

- An action must not be given the same name as a different action of the same module.
- A visualization must not be given the same name as a different visualization.

In all other cases, identical names are allowed. This means that actions of different modules as well as a visualization and a module may be given the same name.

If it is a module, the type of the module (program, function or function block) and the language in which it is to be programmed must also be chosen. The default value for the module type is "Program"; the default language of the module is the language of the last created module. If a module of the "Function" type is to be created, the desired data type must be entered in the text input field "Return type". For this, all elementary data types and defined data types (arrays, structures, enumerations, alias) are admissible. The input help (e.g. via <F2>) can be used.

After the input has been confirmed with "OK" - which is possible only if none of the the name provisions mentioned above is violated - the new object is created in the "Object Organizer" and the corresponding input window appears.

24.7 Transmitting and Activating a PLC Project

"Project" "Transmit"

By means of "Project" "Transmit", the project is compiled. Basically, the transmission process is incremental, i.e. only the changed modules are newly transmitted. A non-incremental transmission process can also be achieved by means of this command if the command "Project" "Debug all" has been executed beforehand.

The transmission run carried out with "Project" "Transmit" is carried out automatically if you log on to the control via "Online" "Log-on".

For the transmission, the message window is opened, indicating the progress of the transmission process, potential errors occurring during the transmission and warnings, as well as data concerning indices or memory consumption (all with number and percentage). Errors and warnings are marked with numbers. Via <F1>, you receive further information on the error currently marked.

Individual or several objects can be excluded from the transmission option using the contextual menu command "Exclude from transmission" or using a corresponding configuration ("Exclude objects") in the transmission options.

☞ The cross-references are produced during the compilation and are stored together with the transmission information. In order to be able to use the commands "Output call tree", "Output cross-reference list", "Unused variables", "Concurring access" and "Multiple writing on output" of the "Project" "Check" menu and to receive all current results, the project must be transmitted again after a change.

"Project" "Transmit All"

As opposed to incremental transmission ("Project" "Transmit"), the project is completely compiled again in case of "Project" "Transmit all". However, the download information is not rejected, as is the case with the command "Debug all". Note the possibility of excluding objects from transmission.

"Online" "Log-on"

This command connects the programming system with the control (or starts the simulation program) and switches to the online mode.

If the current project has not been transmitted since the opening or the last change, it will now be transmitted (just as in the case of "Project" "Transmit").

If errors occur during the transmission, IndraLogic does not switch to the online mode.

If the current project has been changed since the last download, yet was not closed, and if the last download information was not deleted by means of the command "Project" "Debug all", a dialog box will be opened after the command "Log-on" asking:

"The program has been changed. Should the changes be loaded? (Online change)".

By clicking "Yes", you confirm that the changed parts of the project are to be loaded to the control during the log-on process (see the notes on online change below).

By clicking "No", you log on without loading the changes carried out after the last download to the control.

By clicking "Cancel", you cancel the command.

By clicking "Load all", the entire project is loaded again to the control.

If, in the project options, the option "Online operation in the safety mode" is activated in the category Working Area and if the target system supports the function, the log-on dialog box also provides the project information of the project currently loaded to the programming system and already existing on the control. It can be closed via the button "Details <<". If the Working Area option is not activated, this project information can be explicitly opened via the button "Details >>".

> ☞ Please note that the default button, which is highlighted automatically, depends on the settings in the target system.

After a successful log-on, all online functions are available. In order to switch from the online mode back to the offline mode, use the command "Online" "Log-off".

Notes Concerning Online Change

- Online change is not possible in the following cases: after a change of the task configuration, after a change of the control configuration, after pasting a library or after the command "project" "debug all" (see below).
- If the download information (file <project name> <target identifier>.ri) created during the last loading of the project (can also be an online change) was deleted (e.g. via the command "debug all"), an online change is no longer possible unless the *.ri file was also stored in another location and under a different name and can therefore be reloaded by means of the command "load download information". For this, see "online change for a project running on several controls" below.
- There will be no new initialization for an online change, i.e. changes of the initialization values are not taken into account!
- Contrary to a new download of the project (see below, "online" "load"), retain variables keep their values during the online change.

24.8 Debugging a PLC Project

In case of a programming error, you can set break points. If the execution stops within such a break point, you can view the values of all project variables at this point in time. By means of gradual processing (individual steps), you can check the logical correctness of your program.

"Online" "Break Point On/Off"

This command sets a break point at the current position in the active window. If, in the current position, a different break point has already been set, the latter will be removed. The position on which a break point can be set depends on the language in which the module in the active window is written.

In the text editors (AWL, ST), the break point is set to the line in which the cursor is located, provided that this line is a break point position. A break point position can be identified by means of the dark gray (default setting) color of the row number field. In order to set or remove a break point in the text editors, you can also click the row number field.

In FUP and KOP, the break point is manually set to the currently marked network. In order to set or remove a break point in the FUP or DOP editor, you can also click the row number field.

In the AS, the break point is set to the currently marked step. In order to set or remove a break point in the AS, you can also use <Shift> with a double-click.

If a break point is set, the row number field or the network number field or the step is shown with a light blue (default setting) background color.

If program processing reaches a break point, the program stops and the corresponding field will be shown with a red (default setting) background color. In order to continue the program, use the commands "Online" "Start", "Online"" Individual step in" or "Online" "Individual step via".

"Online" "Break Point Dialog Box"

This command opens a dialog box for editing break points in the entire project. The dialog box also indicates all currently set break points.

In order to set a break point, select a module in the combobox "Module" and the row or the network where you would like to set the break point in the combobox "Location"; then press the button "Add". The break point is registered in the list.

In order to delete a break point, mark it while pressing the button "Delete".

Using the button "Delete all", all break points are deleted.

In order to go to the location in the editor where a certain break point has been set, mark it while pressing the button "Go to".

"Online" "Individual Step Via"

This command is used to carry out an individual step; if modules are called, the program stops only after it has been processed. In the AS, a complete action will be carried out.

If the current instruction is the call of a function or of a function block, the function or the function block is carried out completely. Use the command "Online" "Individual step in" in order to obtain the first instruction of a called function or function block.

After the last instruction is reached, the program goes on to the next instruction of the calling module.

"Online" "Individual Step In"

An individual step is carried out; if modules are called, the program stops before the execution of the module's first instruction. The system may switch to a called module.

If the current position is a call of a function or of a function block, the command goes on to the first instruction of the module called.

In all other situations, the command behaves exactly as described in "Online" "Individual step via".

"Online" "Individual Cycle"

This command carries out an individual control cycle and stops after said cycle. This command can be repeated without interruption in order to continue with the individual cycles.

The individual cycle ends when the command "Online" "Start" is executed.

25 Annex

25.1 Basic data type collection

SimpleType	Ann:	BaseType	Restrictions					
			Facets				Patterns	Enum.
			MinInc	MaxInc	Min. Length	Maximum Length		
isoLatin1String	Iso Latin 1 String	xs:string						
isoStr31_t	Iso Latin 1 String 0 - 31	isoLatin1String			0	31		
isoStr16_t	Iso Latin 1 String 0 - 16	isoLatin1String			0	16		
isoStr99_t	Iso Latin 1 String 0 - 99	isoLatin1String			0	99		
isoStr512_t	Iso Latin 1 String 0 - 512	isoLatin1String			0	512		
Str1_t	Type string 0 - 1	isoLatin1String			0	1		
Str2_t	Type string 0 - 2	isoLatin1String			0	2		
Str3_t	String 0 - 3	isoLatin1String			0	3		
Str6_t	String 0 - 6	isoLatin1String			0	6		
Str8_t	String 0 - 8	isoLatin1String			0	8		
Str8_NoUmlaut_t	String 0 - 8	isoLatin1String			0	8	[^äöüß]*	
Str16_t	String 0 - 16	isoLatin1String			0	16		
Str32_t	String 0 - 32	isoLatin1String			0	32		
Str80_t	String 0 - 80	isoLatin1String			0	80		
Str240_t	Type string 0 - 240	isoLatin1String			0	240		
Str240_ NoUmlaut_t	Type string 0-240 without äöüß	isoLatin1String			0	240	[^äöüß]*	
Byte_t	Type byte	xs:byte						
UnsignedByte_t	type unsigned byte	xs:unsignedByte						

Annex

SimpleType	Ann:	BaseType	Restrictions					
			Facets				Pat- terns	Enu m.
			MinInc	Max- Inc	Min. Length	Max- i- mum Leng th		
Byte0_1_t	byte 0 or 1	xs:byte	0	1				0 1
Byte0_2_t	Byte 0 - 2	xs:byte	0	2				0 1 2
Byte0_3_t	Byte 0 - 3	xs:byte	0	3				0 1 2 3
Byte0_4_t	Byte 0 - 4	xs:int	0	4				0 1 2 3
Byte0_5_t	Byte 0 - 5	xs:byte	0	5				0 1 2 3 4
Byte0_6_t	Byte 0 - 6	xs:byte	0	6				0 1 2 3 4 5 6
Byte0_7_t	Byte 0 - 7	xs:byte	0	7				0 1 2 3 4 5 6 7

SimpleType	Ann:	BaseType	Restrictions					
			Facets				Pat-terns	Enum.
			MinInc	Max-Inc	Min. Length	Max-i-mum Length		
Byte0_8_t	Byte 0 - 8	xs:byte	0	8				0 1 2 3 4 5 6 7 8
Byte0_32_t	Byte 0 - 32	xs:byte	0	32				
Byte1_2_t	byte 1 or 2	xs:byte						1 2
Byte1_3_t	Byte 1 - 3	xs:byte	1	3				1 2 3
Byte1_4_t	Byte 1 - 4	xs:byte	1	4				1 2 3 4
Byte1_8_t	Byte 1 - 8	xs:byte	1	8				1 2 3 4 5 6 7 8

Annex

SimpleType	Ann:	BaseType	Restrictions					
			Facets				Pat-terns	Enu m.
			MinInc	Max-Inc	Min. Length	Max-i-mum Leng th		
Byte1_12_t	Byte 1 - 12	xs:byte						1 2 3 4 5 6 7 8 9 10 11 12
Byte-1_7_t	To be omitted	xs:byte	-1	7				
Byte0_10_t	Byte 0 - 10	xs:byte	1	10				0
Byte0_99_t	Byte 0 - 99	xs:byte	0	99				
Byte1_99_t	Byte 1 - 99	xs:byte	1	99				
Byte0_100_t	Byte 0 - 100	xs:byte	0	100				
Short_t	Type short	xs:short						
Unsigned-Short_t	Type unsigned short	xs:unsigned-Short						
Int_t	Type Int	xs:int						
UnsignedInt_t	Type unsigned Int	xs:unsignedInt						
Int0_999_t	To be omitted	xs:int	0	999				
Int0_9999_t	To be omitted	xs:int	0	9999				
Int0_1T_t	Int 0 - 1000	xs:unsignedInt	0	1000				
Int0_10T_t	Int 0 - 10000	xs:int	0	10000				
Int0_100T_t	Int 0 - 100000	xs:int	0	100000				
Int0_32767_t	To be omitted	xs:int	0	32767				
Int0_65535_t	Int 0 - 65535	xs:int	0	65535				
Int0_1M_t	Type Int 0 - 1000000	xs:int	0	1000000				
Float_t	Type float	xs:float						
Float-10_10_t	Type float -10 - 10	xs:float	-10	10				

SimpleType	Ann:	BaseType	Restrictions					
			Facets				Pat-terns	Enum.
			MinInc	Max-Inc	Min. Length	Maximum Length		
Float0_100_t	Type float 0 - 100	xs:float	0	100				
Float-100_100_t	Type float -100 - 100	xs:float	-100	100				
Float-10M_10M_t	Type float -10 millions - +10 millions	xs:float	-10000000	10000000				
Double_t	Type double	xs:double						
Double0_100_t	Type double 0 - 100	xs:double	0	100				
Double0.01_100_t	Type double 0.01_100	xs:double	0.01	100				
Double0_1T_t	Type double 0 - 1000	xs:double	0	1000				
Double0_180_t	Type double 0 - 180	xs:double	0	180				
Double0_360_t	Type double 0 - 360	xs:double	0	360				
Double-180_180_t	Type double -180 - 180	xs:double	-180	180				
Double0_10T_t	Type double 0 -10000	xs:double	0	10000				
Double0_100T_t	Double 0 - 100000	xs:double	0	100000				
Double1_10M_t	Type double 1 - 10 millions	xs:double	1	10000000				
Double-10M_10M_t	Type double -10 millions - +10 millions	xs:double	-10000000	10000000				
Boolean_t	Type boolean	xs:boolean						
Ovr_t	Type override 0 - 150	xs:float	0	150				
Pos_t	Type position (-1 million - +1 million)	xs:double	-1000000	1000000				
Dist_t	Type position (0 - 1 million)	xs:double	0	1000000				
Vel_t	Type velocity 0 - 1000 million [axis velocity]	xs:double	0	1000000000				
SpSpeed_t	Spindle speed 0 - 100000	xs:double	0	100000				
Acc_t	Acceleration 0 - 1000 [axis acceleration]	xs:double	0	1000				
SpAcc_t	Spindle acceleration 0 - 100000	xs:double	0	100000				

Annex

SimpleType	Ann:	BaseType	Restrictions					
			Facets					
			MinInc	Max-Inc	Min. Length	Maximum Length	Patterns	Enum.
JumpVel_t	Jump velocity 0 - 100 000	xs:double	0	100000				
JumpAcc_t	Jump acceleration 0 - 200	xs:float	0	200				
Torq_t	Torque 0 - 1000 [%]	xs:float	0	1000				
Sp_t	Spindle 0 - 8	xs:byte	0	8				
ChSp_t	Spindle 0 - 8	xs:byte	0	32				
ChAx_t	Channel axis (0 - 8)	xs:unsignedByte	0	8				
Ax_t	Lin. and rot. axis (0 - 64)	xs:byte	0	64				
Dr_t	NC-controlled axis drives (0 - 64)	xs:byte	0	64				
Prec_t	Type precision (0 .. 7)	xs:byte	0	7				
Ch_t	Type channels (0-12)	xs:byte	0	12				
CS_t	Type coordinate system	isoLatin1String				3		WCS MCS BCS
AxFun_t	Type axis functionality (to be limited later to 0, 1, 2, 3 (designation main axis X, Y, Z))	xs:int						
SpGr_t	Type spindle group 0 - 4	xs:byte	0	4				
DigBuff_t	Type digitizing buffers 20 - 2000	xs:unsignedInt	20	2000				
ResDigBuff_t	Type reserved dig. Buffer 5 - 100	xs:unsignedInt	5	100				
Blk_t	Block 3 - 999	xs:int	3	999				
CpuTimeBlPrep_t	Cpu time for block preparation 50 - 100	xs:byte	50	100				
OpFiles_t	Open files 5 - 60	xs:int	5	60				
BuffNfs_t	NFS buffer 2048 - 4194304	xs:int	2048	4194304				
CplStack_t	CPL stack 1024 - 524288	xs:int	1024	524288				
BuffSizeExBl_t	Buffer size external block	xs:int	0	1048576				
MinLenCornRound_t	Min block length for corner rounding	xs:float	2	90				
MaxAngCornRound_t	Max angle for corner rounding	xs:double	0	45				

SimpleType	Ann:	BaseType	Restrictions					
			Facets				Pat-terns	Enum.
			MinInc	Max-Inc	Min. Length	Max-i-mum Length		
NofAuxFun_t	Number of auxiliary functions	xs:int	0	1536				
SercBaudeRate_t	Sercos Baud rate	xs:byte						2 4 8 16
TrTimeMdt_t	Transmission time MDT	xs:unsignedInt	62	65535				
CycTime_t	Sercos cycle time = IPO cycle time	xs:unsignedInt	100	64000				
TrTimeMdt_t	Transmission time MDT	xs:unsignedInt	62	65535				
CycTime_t	Sercos cycle time = IPO cycle time	xs:unsignedInt	100	64000				
DbHd_t	Data base header	isoLatin1String				4		IKQ1 IKQ2 IKQ3 IQ1 IQ2 IQ3
DECMV4_t	Decimal with sign (4 bytes)	xs:double	-2147483648	2147483648				
DECMV2_t	Decimal with sign (2 bytes)	xs:double	-32768	32768				
DECOV4_t	Decimal without sign (4 bytes)	xs:double	-4294967296	4294967296				
DECOV2_t	Decimal without sign (2 bytes)	xs:double	-65536	65536				
IDN_t	Sercos IDs	xs:string			8	8	[SP] [-] [0-9] [-] [0-9] (4.4)	

Fig.25-1: Contents of the basic data type collection

25.2 Tool Lists Configuration File

The following overview shows whether the list/editor control configuration file can be opened using the ULC configurator for all process parameters.

			Ⓐ	Ⓑ
ListControlDefinition	ListDefinitions	»1		

Annex

					Ⓐ	Ⓑ
		SubListDefinitions	»13			
1»	ListDefinitions	ListTitle	**TitleType**	»2		
		Styles	**comRepDefType**	»7		
		GridAttributes	»12			
2»	**TitleType**	TitleText			+	
		AssemblyName			+	
		ResourceName			+	
		Token			+	
		TokenOfUsertext			+	
		Alignment	**AlignmentType**	»3		
		Font	**FontType**	»4		
		Color	**CellColour**	»5		
		Height			-	A
3»	**AlignmentType**	Align_vertical			o	B1
		Align_horizontal			o	B2
4»	**FontType**	FontID			o	C1
		FontSize			o	C2
		FontStyle			o	C3
5»	**CellColour**	Foreground	**ColourType**	»6		
		Background	**ColourType**	»6		
6»	**ColourType**	Red			-	D1
		Green			-	D2
		Blue			-	D3
		ColorName			o	D4
7»	**comRepDefType**	Normal	**Style**	»8		
		Selected	**Style**	»8		
		Highlighted	**Style**	»8		

					Ⓐ	Ⓑ
		Empty	**Style**	»8		
		ColumnTitle	**ColumnTitle-Type**	»11		
		SubColumn-Title	**ColumnTitle-Type**	»11		
8»	**Style**	Fixed	**SubStyle**	»9		
		Scrollable	**SubStyle**	»9		
9»	**SubStyle**	Colour	**CellColour**	»5		
		Font	**FontType**	»4		
		Alignment	**Alignment-Type**	»3		
		ImageAlign-ment	**Alignment-Type**	»3		
		OuterBorder	**BorderType**	»10		
		InnerBorder	**BorderType**	»10		
		WordWrap			-	E
		Display			-	F
10»	**BorderType**	BorderSize			-	G
		BorderColour	**ColourType**	»6		
		BorderStyle			-	H
11»	**ColumnTitle-Type**	Colour	**CellColour**	»5		
		Font	**FontType**	»4		
		Alignment	**Alignment-Type**	»3		
		ImageAlign-ment	**Alignment-Type**	»3		
		InnerBorder	**BorderType**	»10		
		WordWrap			-	E
12»	GridAttributes	BorderStyle			+	
		Stripline	**BorderType**	»10		
		FilenameOf-DefaultValues			+	
		AllowFocu-sOnNonEdita-bleCells			+	
		ShowTrace-Messages			+	

Annex

					Ⓐ	Ⓑ
		ShowDebugMessages			+	
		AllowMerging			+	
		ListBackGroundColor	**ColourType**	»6		
		KeyActionEnter			+	
		KeyActionTab			+	
		AllowFreezingWithMouse			+	
		AllowResizing			+	
		AutoSearch			+	
		ExtendLastCol			+	
		AutoResize			+	
		UseNumberDecimalSeparator			+	
		SuppressComma			+	
		CursorKeyCanCloseEditmode			+	
		TabCanCloseEditMode			+	
		DrawTextFlexgridOrg			+	
		AllowSorting			+	
		ScrollTrack			+	
		PageDownTrack			+	
		TestDOMToConfig			+	
		TestSOMToConfig			+	
		UserStringTableID			-	J
		GridHighlight			+	
		CheckRowVisibilityAtBeginning			+	
13»	SubListDefinitions	ScopeDefinition	**ScopeDefinitionType**	»14		

					Ⓐ	Ⓑ
		ColumnDefinitions	**CommonColumnDefinitionsType**	»15		
		ColumnGeneralAttributes	»23			
		RowDefinition	»26			
		CellDefinitions	**CellDefinitionsType**	»29		
		SublistProperties	»41			
14»	**ScopeDefinitionType**	PathOfMultiplicator			-	K1
		ShowMultipleSublists			-	K2
		NoOfFixedColumn			-	K3
		NoOfScrollableColumn			-	K4
		NoOfFrozenColumn			-	K5
		NoOfFrozenRow			-	K6
15»	**CommonColumnDefinitionsType**	ColDef	**ColumnDefinitionType**	»16		
16»	**ColumnDefinitionType**	PathOfMultiplicator			-	L1
		ColTitle	**ColumnTitleStyle**	»17		
		SubColumnDefinitions	**SubColumnDefinitionsType**	»18		
17»	**ColumnTitleStyle**	TitleText			+	
		ProcessVariableID			+	
		AssemblyName			+	
		ResourceName			+	
		Token			+	

					Ⓐ	Ⓑ
		TokenOfUsertext			+	
18»	**SubColumnDefinitionsType**	SubColDef	»19			
19»	SubColDef	PathOfMultiplicator			-	L2
		SubColTitle	**ColumnTitleStyle**	»20		
		SubColWidth			+	
		CellRanges	»21			
		ColSortFlag			-	M
		SortRow			-	N
20»	**ColumnTitleStyle**	TitleText			+	
		ProcessVariableID			+	
		AssemblyName			+	
		ResourceName			+	
		Token			+	
		TokenOfUsertext			+	
21»	CellRanges	CellRange	»22			
22»	CellRange	FirstRangeCellNo			-	O1
		LastRangeCellNo			-	O2
23»	ColumnGeneralAttributes	ColumnTitleAtributes	»24			
		SubColumnTitleAtributes	»25			
24»	ColumnTitleAtributes	ColumnTitleHeight			+	

						Ⓐ	Ⓑ
25»	SubColumn-TitleAtributes	ColumnTitle-Height				+	
26»	RowDefinition	SubRowDefi-nitions	»27				
27»	SubRowDefi-nitions	PathOfMulti-plicator				-	L3
		SubRow-Height				+	
		Dependen-cy_for_Visibili-ty	**Dependency-Type**	»28			
28»	**Dependency-Type**	DepProcess-VariableID				-	P1
		Operation				-	P2
		Value				-	P3
		Value2				-	P4
		Following-DepCondition				-	P5
		Condition	**Dependency-Type**	»28			
29»	**CellDefinition-sType**	CellDef	»30				
30»	CellDef	CellAddress	**CellAddres-sType**	»31			
		ContentType	**CellContent-Type**	»32			
		EditType	**CellEditType**	»36			
		Representa-tionDefinitions	»37				
31»	**CellAddres-sType**	line				+	
		Column				+	
		SubColumn				+	
32»	**CellContent-Type**	CellContent-Type				+	
		Comment				+	
		FixedImage				+	

Annex

				Ⓐ			Ⓑ
		ProcessVariableID			+		
		ProcessVariableDataType			+		
		TextOrBitmap			+		
		AssemblyName			+		
		ResourceName			+		
		BitmapAssemblyName			+		
		BitmapResourceName			+		
		Token			+		
		TokenOfUsertext			+		
		ReadProcVarAllways			-	T	
		BitmapTable	»33				
		BitmapTableCom		+			
		BitmapTableShowKey		-			
		TextTable	»34				
		TextTableCom		-			
		TextTableUse		-			
		TextTableShowKey		-			
33»	BitmapTable	KeyValuePair	»35				
»34	TextTable	KeyValuePair	»35				
34»	KeyValuePair	Key			o		Q1
		Value			o		Q2
36»	**CellEditType**	EditStatus			o	U	
		EditTypeSelection			+		
37»	RepresentationDefinitions	Dependency_for_Representation	»38				

					Ⓐ	Ⓑ
		TextFormat			+	
		NumFormat	**NumFormatType**	»39		
38»	Dependency_for_Representation	Dependency_for_Visibility	**DependencyType**	»28		
		Validation	**DependencyType**	»28		
		EditDepend	**DependencyType**	»28		
39»	**NumFormatType**	NumType			+	
		DigitDef	»40			
		Pos_dec_Point_Def	»41			
		FormatString			+	
		EditMask			+	
40»	DigitDef	No_of_Digits			+	
		DepVar			+	
41»	Pos_dec_Point_Def	Position_after_dec_Point			+	
		DepVar			+	
42»	SublistProperties	Highlighted	»43			
		SublistActions	»45			
43»	Highlighted	TypeOfHighlight			-	R
	IsHighlighted	IsHighlighted	»44			
44»	IsHighlighted	**DependencyType**	»28			
		HighlightColor	**ColourType**	»6		
45»	SublistActions	Action	»46			

					Ⓐ	Ⓑ
46»	Action	TypeOfAction			-	S
		IsCondition	Dependency-Type	»28		

Explanations of the process parameters that cannot, or only to a degree, be opened using the ULC configurator (indicated by **-** or **o**).

Ⓑ		Explanation		Remarks
A		Height of table title line		
B	1	Horizontal arrangement		
	2	Vertical arrangement		
C	1	Font type		
	2	Font size		
	3	Font style		
D	1	Color setting	Red value	
	2		Green value	
	3		Blue value	
	4		Color name	Log. from color helper definition
E		Automatic line break		
F		Type of display		
G		Line width		
H		Line type		
J		Name of user text file		
K	1	Multiplication process variable		Multiplication of partial lists
	2	Display of partial lists		
	3	Number of fixed columns		
	4	Number of scrollable columns		Total number of main columns
	5	Number of frozen columns		
	6	Number of frozen lines		
L	1	Multiplication process variable		Multiplication of main columns
	2			Multiplication of subordinate col-umns
	3			Multiplication of subordinate lines
M		Column sorting permitted		
N		Line sorting permitted		
O	1	No. of 1st subordinate line		Of merge area
	2	No. of last subordinate line		
P	1	Dep. on process variable		

	Ⓑ	Explanation	Remarks
	2	Operation / condition	
	3	Value	
	4	2. Value	(multiplication process variable)
	5	Subsequent condition	
Q	1	Value of process variables	for bitmap table and text table
	2	Bitmap name or text token name	
R		Highlighted type	Currently not relevant
S		Action type	Currently not relevant
T		Control flag for reading the process variables	
U		This is the editability setting in the ULC configurator.	Mode 3 cannot be selected (necessary for conditional editing).

25.3 Interfaces

25.3.1 Status Bits

BQ2 - Tool status bits (tool identification)

Data element	Status group	Designation	Abbreviation		Bit no.	Description	Type	Preassignment	Can be changed in MTGUI?
BQ2	ToolStatus	ActiveTool	ta	0	1	active tool	BrcBoolean	0	
		PrimaryTool	tp		2	primary tool	BrcBoolean	0	
		UsedTool	tu		3	used tool	BrcBoolean	0	
		WarnLimit	tw		4	warning limit reached	BrcBoolean	0	
		WornOut	two		5	tool worn out	BrcBoolean	0	
		PlaceCoded	TPC		6	place-coded tool	BrcBoolean	0	yes
		Locked	TL		7	locked tool	BrcBoolean	0	yes
		Defective	TD	7	8	defective tool	BrcBoolean	0	yes
		Ignore	ti	8	9	ignore tool	BrcBoolean	0	
		Measured	tm		10	measure tool	BrcBoolean	0	
		ToLoad	ttl		11	load tool	BrcBoolean	0	
		ToUnload	ttu		12	unload tool	BrcBoolean	0	
		MasterTool	mt		13	master tool	BrcBoolean	0	
		ToolChange	tc		14	tool change	BrcBoolean	0	
		SubstituteTool	ts		15	substitute tool	BrcBoolean	0	
		Reserve 1	TR1	15	16	reserve	BrcBoolean	0	
		Reserve 2	TR2	16	17	reserve	BrcBoolean	0	
		Reserve 3	TR3		18	reserve	BrcBoolean	0	
		Reserve 4	TR4		19	reserve	BrcBoolean	0	
		Reserve 5	TR5		20	reserve	BrcBoolean	0	
		Reserve 6	TR6		21	reserve	BrcBoolean	0	
		Reserve 7	TR7		22	reserve	BrcBoolean	0	
		Reserve 8	TR8		23	reserve	BrcBoolean	0	
		Reserve 9	TR9	23	24	reserve	BrcBoolean	0	
		UserStatus 1	TS1	24	25	tool user status 1	BrcBoolean	0	yes
		UserStatus 2	TS2		26	tool user status 2	BrcBoolean	0	yes
		UserStatus 3	TS3		27	tool user status 3	BrcBoolean	0	yes
		UserStatus 4	TS4		28	tool user status 4	BrcBoolean	0	yes
		UserStatus 5	TS5		29	tool user status 5	BrcBoolean	0	yes
		UserStatus 6	TS6		30	tool user status 6	BrcBoolean	0	yes
		UserStatus 7	TS7		31	tool user status 7	BrcBoolean	0	yes
		UserStatus 8	TS8	31	32	tool user status 8	BrcBoolean	0	yes

= used at present

= not used at present

Fig.25-2: Tool status bits (BQ2)

BQ1 - Location status bits

Data element	Status group	Designation	Abbreviation		Bit no.	Description	Type	Preassignment	Can be changed in MTGUI?
BQ1	PlaceStatus	Reserve 1	PR1	0	1	free location	BrcBoolean	0	
		Blocked	PB		2	blocked location	BrcBoolean	0	yes
		Ignore	pi		3	ignore location	BrcBoolean	0	
		Reserved	pr		4	reserve location	BrcBoolean	0	
		ReservedLeft	prl		5	left half-location reserved	BrcBoolean	0	
		ReservedRight	prr		6	right half-location reserved	BrcBoolean	0	
		ReservedLeftLower	prll		7	left lower half-location reserved	BrcBoolean	0	
		ReservedRightLower	prrl	7	8	right lower half-location reserved	BrcBoolean	0	
		OccupiedLeft	pol	8	9	left half-location occupied	BrcBoolean	0	
		OccupiedRight	por		10	right half-location occupied	BrcBoolean	0	
		OccupiedLeftLower	poll		11	left lower half-location occupied	BrcBoolean	0	
		OccupiedRightLower	porl		12	right lower half-location occupied	BrcBoolean	0	
		PlaceTypeMonitoring	ptm		13	place type monitoring	BrcBoolean	0	
		DeviceTypeMonitoring	dtm		14	device type monitoring	BrcBoolean	0	
		FormTypeMonitoring	ftm		15	form type monitoring	BrcBoolean	0	
		DeviceInPlace	dip	15	16	device in place	0...99	0	
		Reserve 2	PR2	16	17	reserve	BrcBoolean	0	
		Reserve 3	PR3		18	reserve	BrcBoolean	0	
		Reserve 4	PR4		19	reserve	BrcBoolean	0	
		Reserve 5	PR5		20	reserve	BrcBoolean	0	
		Reserve 6	PR6		21	reserve	BrcBoolean	0	
		Reserve 7	PR7		22	reserve	BrcBoolean	0	
		Reserve 8	PR8		23	reserve	BrcBoolean	0	
		Reserve 9	PR9	23	24	reserve	BrcBoolean	0	
		PlaceUserStatus 1	PS1	24	25	place user status bits 1	BrcBoolean	0	yes
		PlaceUserStatus 2	PS2		26	place user status bits 2	BrcBoolean	0	yes
		PlaceUserStatus 3	PS3		27	place user status bits 3	BrcBoolean	0	yes
		PlaceUserStatus 4	PS4		28	place user status bits 4	BrcBoolean	0	yes
		PlaceUserStatus 5	PS5		29	place user status bits 5	BrcBoolean	0	yes
		PlaceUserStatus 6	PS6		30	place user status bits 6	BrcBoolean	0	yes
		PlaceUserStatus 7	PS7		31	place user status bits 7	BrcBoolean	0	yes
		PlaceUserStatus 8	PS8	31	32	place user status bits 8	BrcBoolean	0	yes

= used at present (yellow)

= not used at present (grey)

Fig.25-3: Location status bits (BQ1)

BQ3 - Technology status bits

Data element	Status group	Designation	Abbreviation		Bit no.	Description	Type	Preassignment	Can be changed in MTGUI
BQ3	ToolTechStatus	NoOfEdges_1	EdNo	0	1	no. of edges	0 -16	acc. to ToolType	
		NoOfEdges_2			2				
		NoOfEdges_3			3				
		NoOfEdges_4			4				
		Reserve 1	TCR1		5	reserve	BrcBoolean	0	
		Reserve 2	TCR2		6	reserve	BrcBoolean	0	
		Reserve 3	TCR3		7	reserve	BrcBoolean	0	
		Reserve 4	TCR4	7	8	reserve	BrcBoolean	0	
		L1	L1	8	9	L1 - correction valid	BrcBoolean	acc. to ToolType	
		L2	L2		10	L2 - correction valid	BrcBoolean	acc. to ToolType	
		L3	L3		11	L3 - correction valid	BrcBoolean	acc. to ToolType	
		R1	R1		12	R1 - correction valid	BrcBoolean	acc. to ToolType	
		R2	R2		13	R2 - correction valid	BrcBoolean	acc. to ToolType	
		O	O		14	edge position valid	BrcBoolean	acc. toToolType	
		Reserve 5	TCR5		15	reserve	BrcBoolean	0	
		Reserve 6	TCR6	15	16	reserve	BrcBoolean	0	
		TechReserve 1	TCR7	16	17	reserve for new technology data	BrcBoolean	0	
		TechReserve 2	TCR8		18	reserve for new technology data	BrcBoolean	0	
		TechReserve 3	TCR9		19	reserve for new technology data	BrcBoolean	0	
		Diameter	d		20	diameter	BrcBoolean	acc. to ToolType	
		DiameterSmall	ds		21	small diameter	BrcBoolean	acc. to ToolType	
		AngleDrillBit	adb		22	angle drill bit	BrcBoolean	acc. to ToolType	
		AngleSetting	as		23	setting angle	BrcBoolean	acc. to ToolType	
		TreadPitch	tp	23	24	tread pitch	BrcBoolean	acc. to ToolType	
		LengthFirstCut	lfc	24	25	length of first cut	BrcBoolean	acc. to ToolType	
		LengthLoss	ll		26	length loss	BrcBoolean	acc. to ToolType	
		LengthUsable	lu		27	usable length	BrcBoolean	acc. to ToolType	
		NumberOfTeeth	nt		28	number of teeth	BrcBoolean	acc. to ToolType	
		WidthOfEdge	we		29	width of edge	BrcBoolean	acc. to ToolType	
		LengthOfEdge	le		30	length of edge	BrcBoolean	acc. to ToolType	
		AngleMainCuttingEdge	ame		31	angle of main cutting edge	BrcBoolean	acc. to ToolType	
		AngleAuxiliaryEdge	aae	31	32	angle of auxiliary edge	BrcBoolean	acc. to ToolType	

= used at present

= not used at present

Fig.25-4: Technology status bits (BQ3)

25.3.2 Tool Catalog

Tool Catalog

Technology		General processing							
Tool type name		General tool							
Tool type	IKQ2	0000							
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5
		x	x	x	x	x	x	x	x

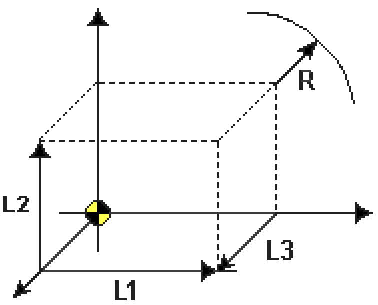
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	-	-	-	-	-
		x	x	x	x	x	x	x	x	x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x	x	x	x	x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
		x	x	x	x					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		can be set from 1 - 16								
EdNo = 1	BQ3(decimal value)	-16								
:		:								
EdNo = 16		-1								
Screen										

Fig.25-5: 00000

Technology		Drilling tools								
Tool type name		Drilling tool, general								
Tool type	IKQ2	1000								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	-	-	-	-	d
										x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					

Annex

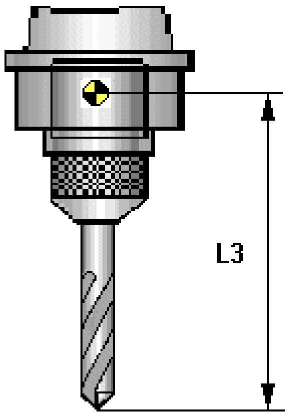
Edge number	BQ3 (Bit 1 ..4)	EdNo	
		can be set from 1 - 16	
EdNo = 1	BQ3(decimal value)	33792	
:		:	
EdNo = 16		33807	
Screen			

Fig.25-6: 1000

Technology		Drilling tools							
Tool type name		Twist drill							
Tool type	IKQ2	1001							
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	-	lu	-	lfc	-	d
				x		x		x		x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

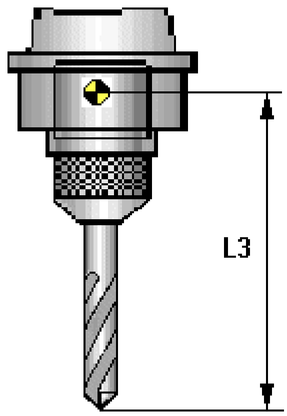
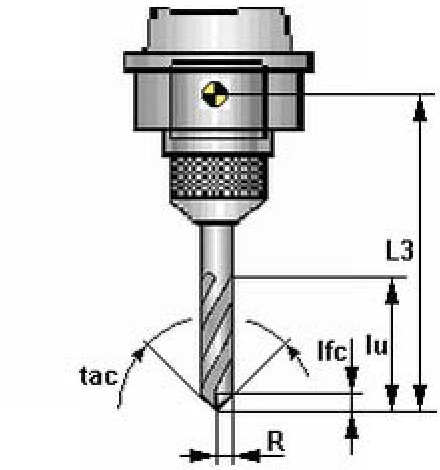
	BQ3(decimal value)	2786304	
Screen			

Fig.25-7: 1001

Technology		Drilling tools							
Tool type name		Center drill							
Tool type	IKQ2	1002							
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	-	-	-	-	d
										x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

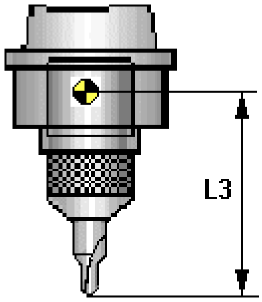
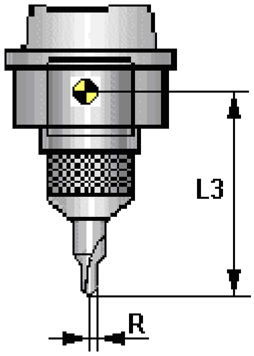
	BQ3(decimal value)	33792	
Screen			

Fig.25-8: 1002

Technology		Drilling tools							
Tool type name		NC start drill							
Tool type	IKQ2	1003							
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	-	-	-	lfc	-	d
				x				x		x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

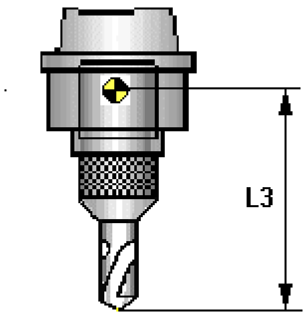
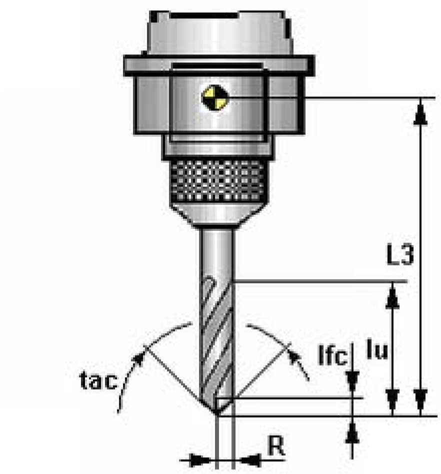
	BQ3(decimal value)	2262016	
Screen			

Fig.25-9: 1003

Technology		Drilling tools								
Tool type name		Reversible tip drill								
Tool type	IKQ2	1004								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	lu	-	-	-	d
						x				x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

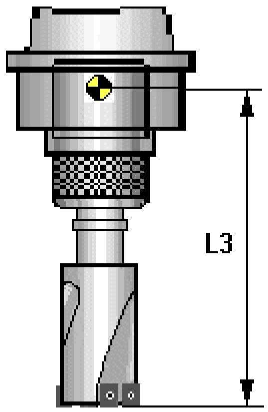
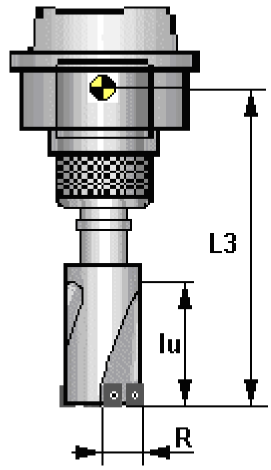
	BQ3(decimal value)	558080	
Screen			

Fig.25-10: 1004

Technology		Drilling tools								
Tool type name		Step drill 2 steps								
Tool type	IKQ2	1005								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	-	lu	-	lfc	ds	d
				x		x		x	x	x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		2								

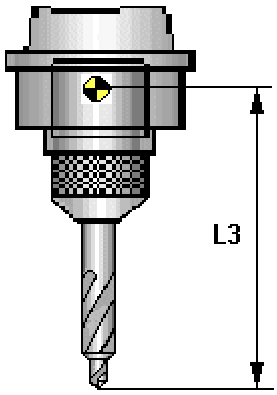
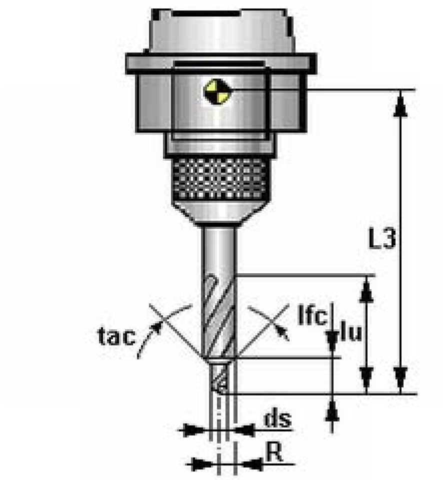
	BQ3(decimal value)	2851841	
Screen			

Fig.25-11: 1005

Technology		Drilling tools								
Tool type name		Step drill 3 steps								
Tool type	IKQ2	1006								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		ds	-	tac	-	lu	-	lfc	-	-
		x		x		x		x		
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		3								

Annex

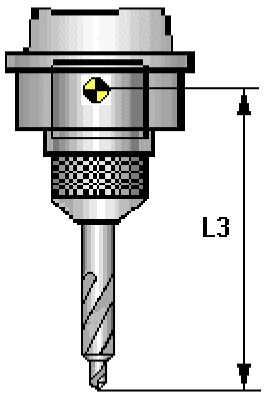
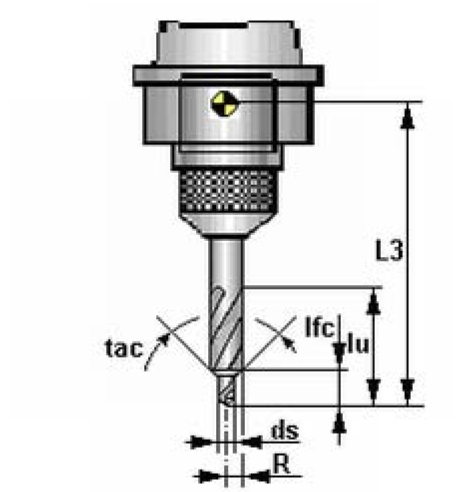
	BQ3(decimal value)	11174914	
Screen			

Fig.25-12: 1006

Technology		Drilling tools								
Tool type name		Countersink								
Tool type	IKQ2	1007								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	-	-	-	-	ds	d
				x					x	x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

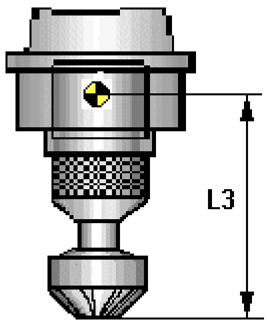
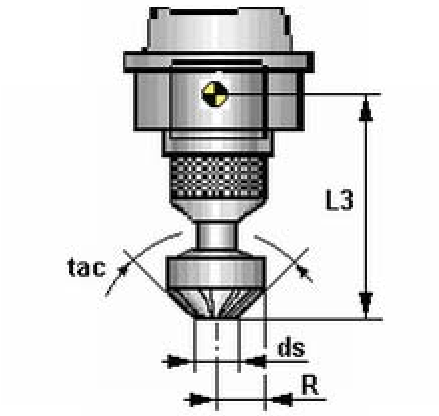
	BQ3(decimal value)	2196480	
Screen			

Fig.25-13: 1007

Technology		Drilling tools								
Tool type name		Plane countersink								
Tool type	IKQ2	1008								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	lu	-	-	-	d
						x				x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

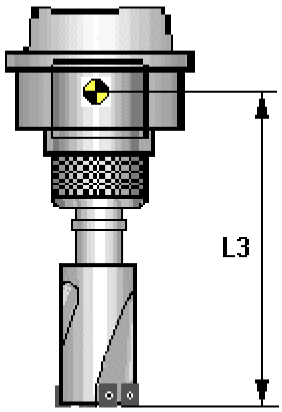
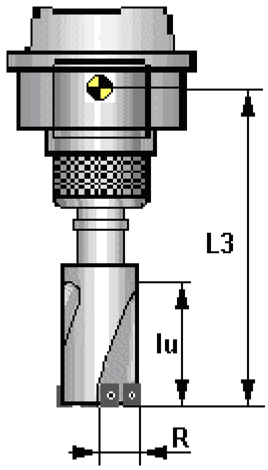
	BQ3(decimal value)	558080	
Screen			

Fig.25-14: 1008

Technology		Drilling tools								
Tool type name		Spiral countersink								
Tool type	IKQ2	1009								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	lu	-	lfc	-	d
						x		x		x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

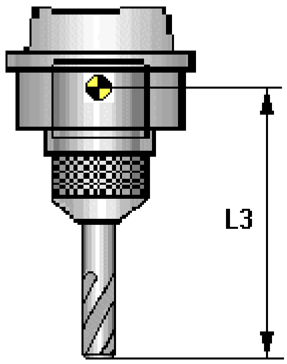
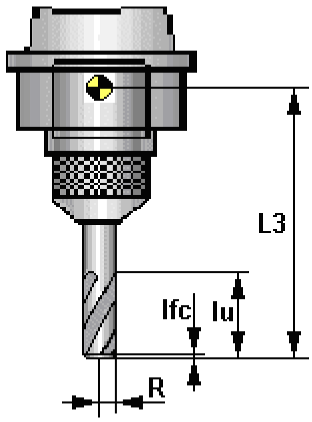
	BQ3(decimal value)	689152	
Screen			

Fig.25-15: 1009

Technology		Drilling tools								
Tool type name		Spot countersink								
Tool type	IKQ2	1010								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	lu	-	ll	ds	d
						x		x	x	x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

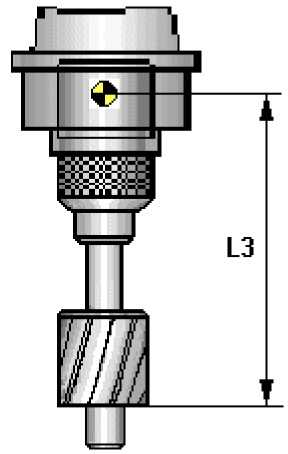
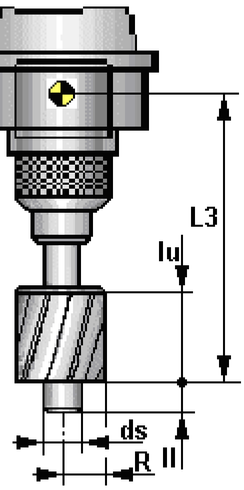
	BQ3(decimal value)	754688	
Screen			

Fig.25-16: 1010

Technology		Drilling tools								
Tool type name		Tap								
Tool type	IKQ2	1011								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	tp	lu	-	lfc	-	d
					x	x		x		x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

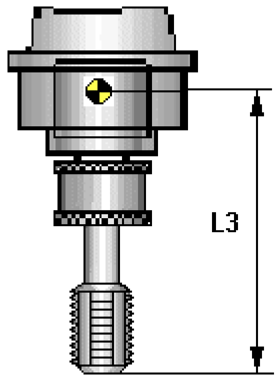
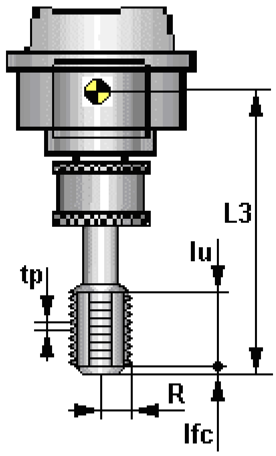
	BQ3(decimal value)	1737728	
Screen			

Fig.25-17: 1011

Technology		Drilling tools								
Tool type name		Boring bar								
Tool type	IKQ2	1012								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	ta	lu	-	ll	ds	d
					x	x		x	x	x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

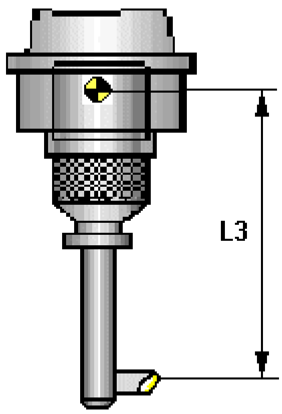
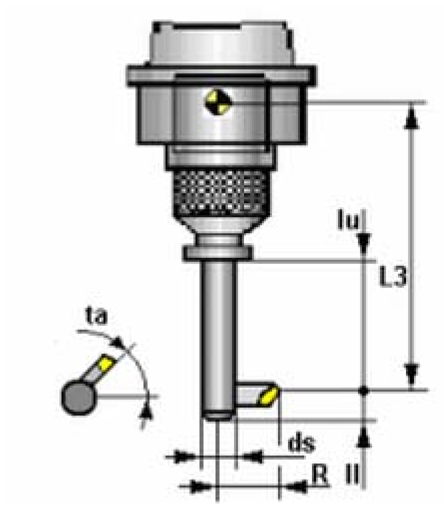
	BQ3(decimal value)	1803264	
Screen			

Fig.25-18: 1012

Technology		Drilling tools								
Tool type name		Reverse countersink								
Tool type	IKQ2	1013								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	ta	lu	-	ll	ds	d
					x	x		x	x	x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

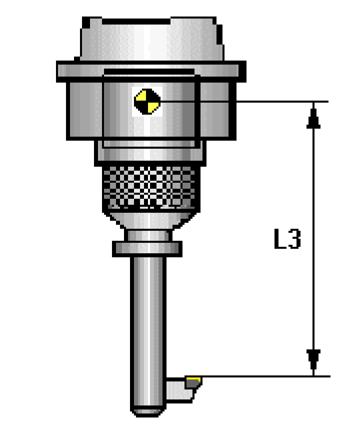
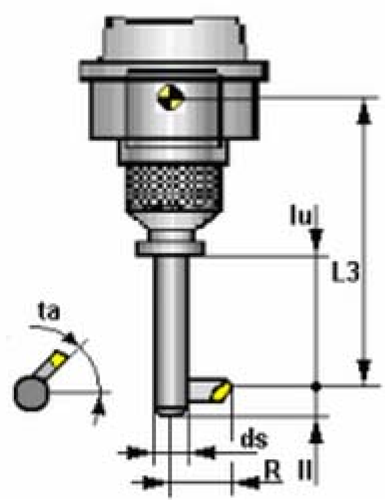
	BQ3(decimal value)	1803264	
Screen			

Fig.25-19: 1013

Technology		Drilling tools								
Tool type name		Reamer								
Tool type	IKQ2	1014								
Reserve	BQ3	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
	(Bit 32 ... 25)									
Technology data	BQ3	ET1	T8	T7	T6	T5	T4	T3	T2	T1
	(Bit 24 ... 16)	-	-	-	-	lu	-	lfc	-	d
						x		x		x
Correction type	BQ3	DIA	O	R2	R1	L3	L2	L1		
	(Bit 15 .. 0.9)					x				
Reserve	BQ3	TRC4	TRC3	TRC2	TRC1					
	(Bit 8 .. 0.5)									
Edge number	BQ3	EdNo								
	(Bit 1 ..4)	1								

Annex

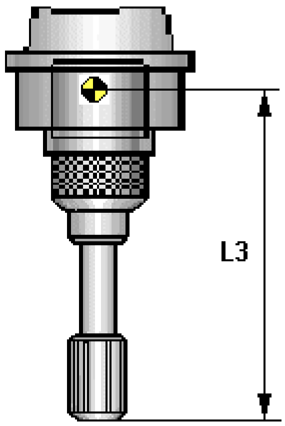
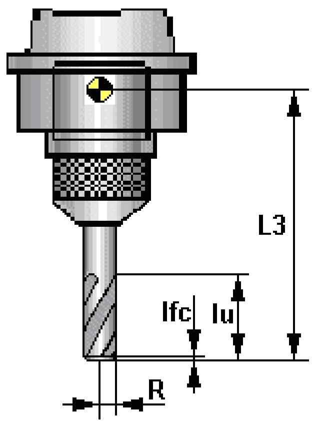
	BQ3(decimal value)	689152	
Screen			

Fig.25-20: *1014*

Technology		Drilling tools								
Tool type name		Thread drill mill								
Tool type	IKQ2	1015								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	tp	lu	-	ll	ds	d
					x	x		x	x	x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
					x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
		x	x	x	x					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

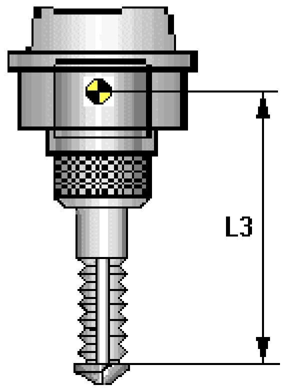
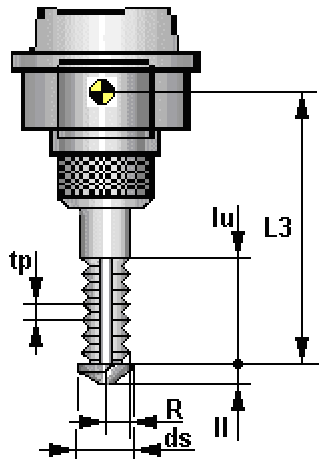
<table>
<tr><td></td><td>BQ3(decimal value)</td><td>1805312</td><td></td></tr>
<tr><td>Screen</td><td colspan="2">
</td><td>
</td></tr>
</table>

Fig.25-21: 1015

Technology		Drilling tools								
Tool type name		Special drilling tool								
Tool type	IKQ2	1016								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	-	-	-	-	d
										x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
					x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		can be set from 1 - 16								

Annex

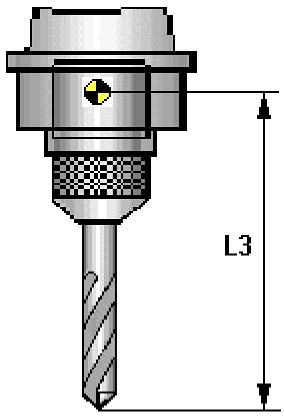
EdNo = 1	BQ3(decimal value)	35840	
:		:	
EdNo = 16		35855	
Screen			

Fig.25-22: 1016

Technology		Milling tools							
Tool type name		Milling tool, general							
Tool type	IKQ2	2000							
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	nt	-	-	lu	-	-	-	d
			x			x				x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1	
					x	x			
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1				
Edge number	BQ3 (Bit 1 ..4)	EdNo							
		can be set from 1 - 16							

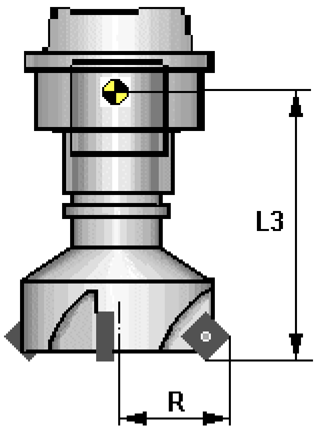
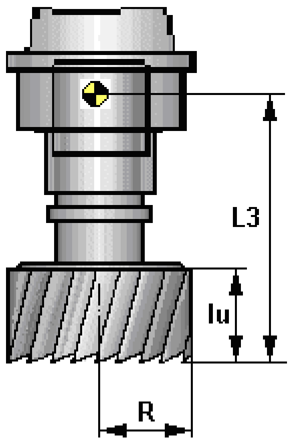
EdNo = 1	BQ3(decimal value)	4230144	
:		:	
EdNo = 16		4230159	
Screen			

Fig.25-23: 2000

Technology		Milling tools								
Tool type name		End milling cutter								
Tool type	IKQ2	2001								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	nt	-	-	lu	-	-	-	d
			x			x				x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
					x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

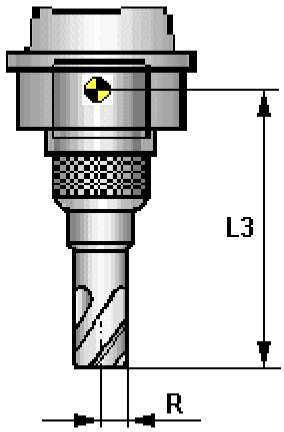
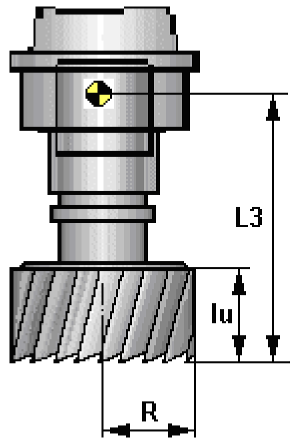
	BQ3(decimal value)	4754432	
Screen			

Fig.25-24: 2001

Technology		Milling tools								
Tool type name		Groove milling cutter								
Tool type	IKQ2	2002								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	nt	-	-	lu	-	-	-	d
			x			x				x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
					x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

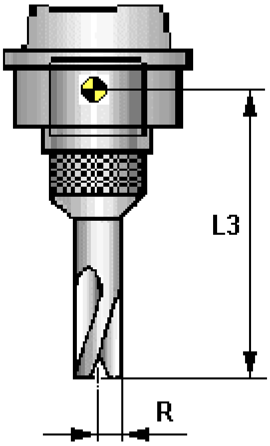
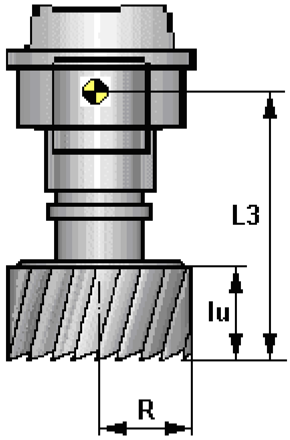
	BQ3(decimal value)	4754432	
Screen			

Fig.25-25: 2002

Technology		Milling tools								
Tool type name		Long-hole milling cutter								
Tool type	IKQ2	2003								
Reserve	BQ3	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
	(Bit 32 ... 25)									
Technology data	BQ3	ET1	T8	T7	T6	T5	T4	T3	T2	T1
	(Bit 24 ... 16)	-	nt	-	-	lu	-	-	-	d
			x			x				x
Correction type	BQ3	DIA	O	R2	R1	L3	L2	L1		
	(Bit 15 .. 0.9)				x	x				
Reserve	BQ3	TRC4	TRC3	TRC2	TRC1					
	(Bit 8 .. 0.5)									
Edge number	BQ3	EdNo								
	(Bit 1 ..4)	1								

Annex

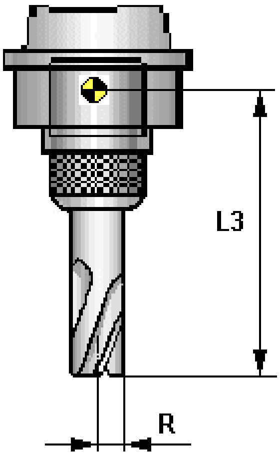
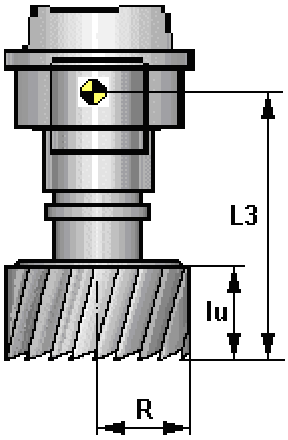
	BQ3(decimal value)	4754432	
Screen			

Fig.25-26: 2003

Technology		Milling tools								
Tool type name		Disk milling cutter								
Tool type	IKQ2	2004								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	nt	-	-	lu	-	ll	-	d
			x			x		x		x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
					x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

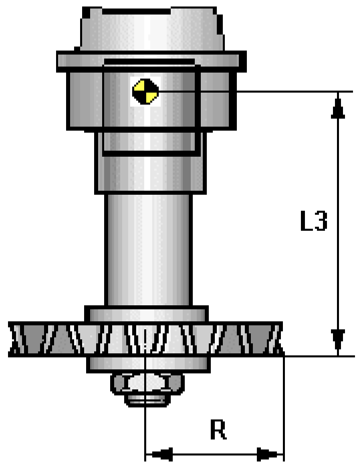
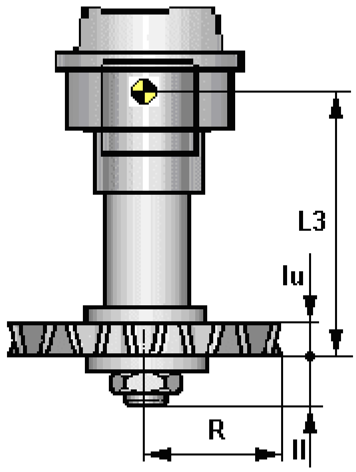
	BQ3(decimal value)	4885504	
Screen			

Fig.25-27: 2004

Technology		Milling tools								
Tool type name		Saw blade								
Tool type	IKQ2	2005								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	nt	-	-	lu	-	ll	-	d
			x			x		x		x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
					x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

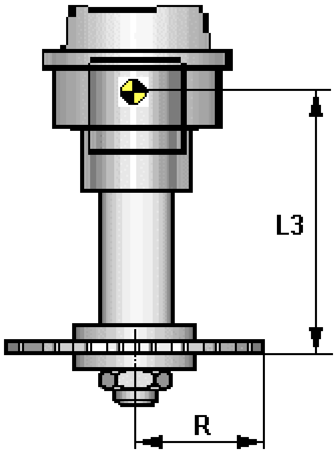
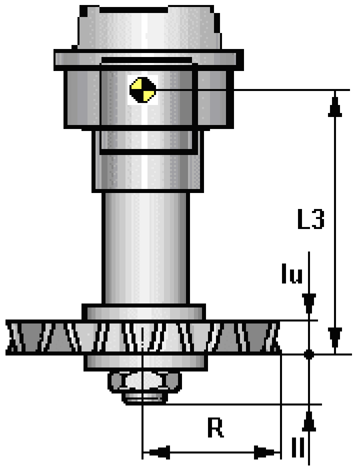
	BQ3(decimal value)	4885504	
Screen			

Fig.25-28: 2005

Technology		Milling tools								
Tool type name		Pacing cutter								
Tool type	IKQ2	2006								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	nt	-	tal	lu	-	-	-	d
			x		x	x				x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
					x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

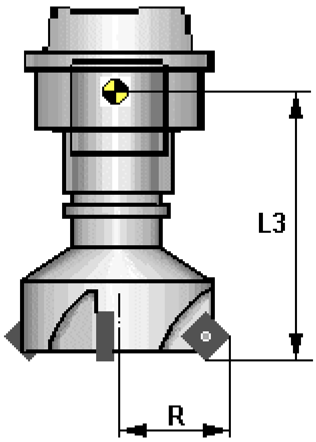
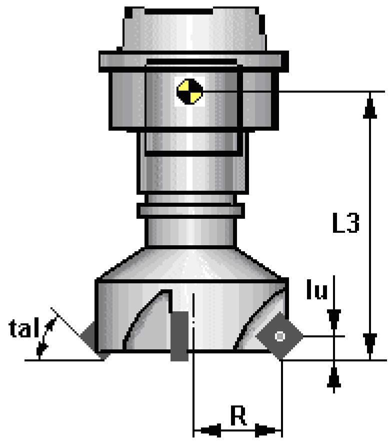
	BQ3(decimal value)	5803008	
Screen			

Fig.25-29: 2006

Technology		Milling tools								
Tool type name		Corner milling cutter								
Tool type	IKQ2	2007								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	nt	-	-	lu	-	-	-	d
			x			x				x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
					x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

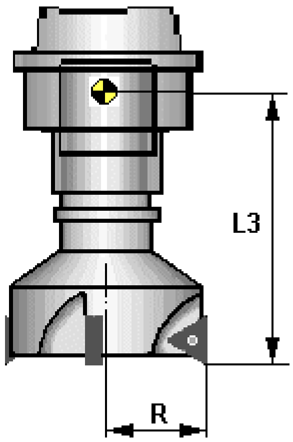
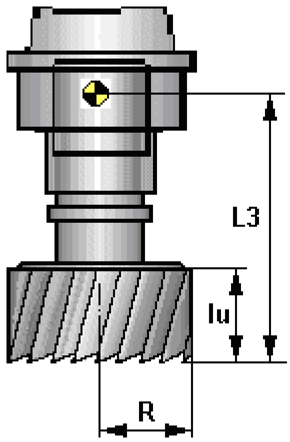
	BQ3(decimal value)	4754432	
Screen			

Fig.25-30: 2007

Technology		Milling tools								
Tool type name		Plain milling cutter								
Tool type	IKQ2	2008								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	nt	-	-	lu	-	ll	-	d
			x			x		x		x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
					x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

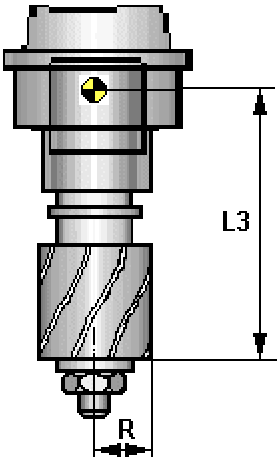
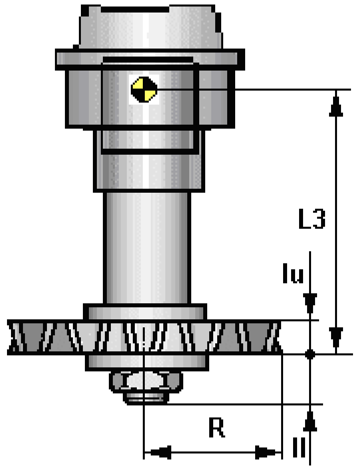
	BQ3(decimal value)	4885504	
Screen			

Fig.25-31: 2008

Technology		Milling tools								
Tool type name		Shell end mill								
Tool type	IKQ2	2009								
Reserve	BQ3	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
	(Bit 32 ... 25)									
Technology data	BQ3	ET1	T8	T7	T6	T5	T4	T3	T2	T1
	(Bit 24 ... 16)	-	nt	-	-	lu	-	-	-	d
			x			x				x
Correction type	BQ3	DIA	O	R2	R1	L3	L2	L1		
	(Bit 15 .. 0.9)				x	x				
Reserve	BQ3	TRC4	TRC3	TRC2	TRC1					
	(Bit 8 .. 0.5)									
Edge number	BQ3	EdNo								
	(Bit 1 ..4)	1								

Annex

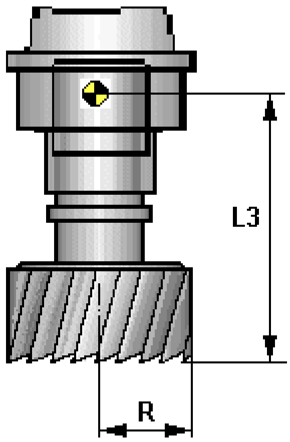
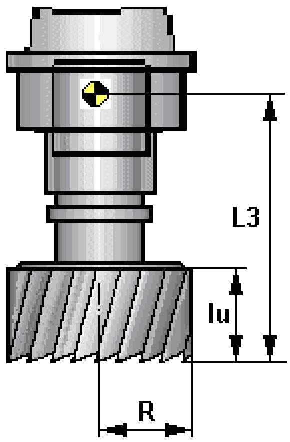
	BQ3(decimal value)	4754432	
Screen			

Fig.25-32: 2009

Technology		Milling tools								
Tool type name		Angular milling cutter								
Tool type	IKQ2	2010								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	nt	-	tal	lu	-	-	-	d
			x		x	x				x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
					x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

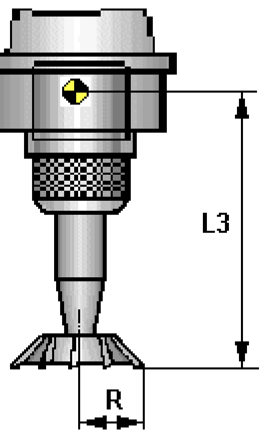
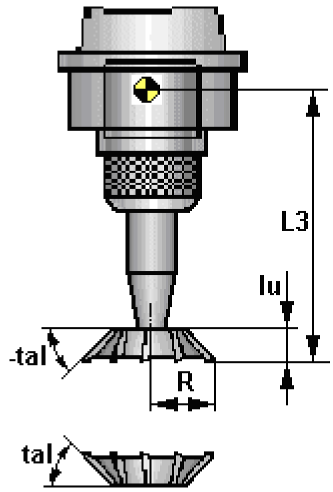
	BQ3(decimal value)	5803008	
Screen			

Fig.25-33: 2010

Technology		Milling tools								
Tool type name		T groove cutter								
Tool type	IKQ2	2011								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	nt	-	-	lu	-	-	-	d
			x			x				x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
					x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

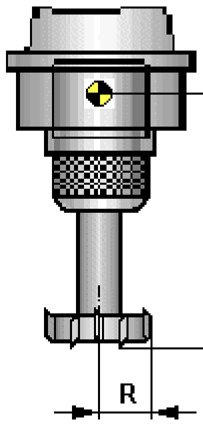
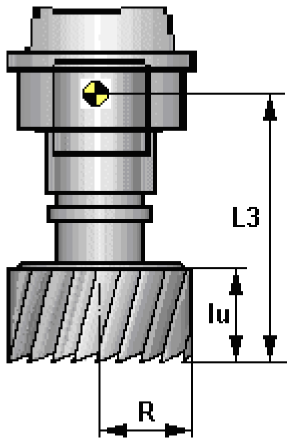
	BQ3(decimal value)	4754432	
Screen			

Fig.25-34: 2011

Technology		Milling tools								
Tool type name		Diesinking cutter								
Tool type	IKQ2	2012								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	nt	-	tal	lu	-	tc	-	d
			x		x	x		x		x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
				x	x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

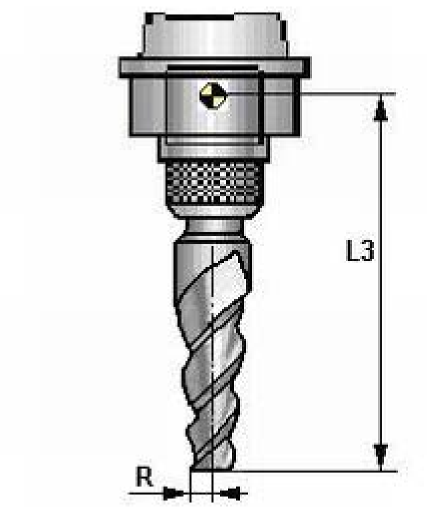
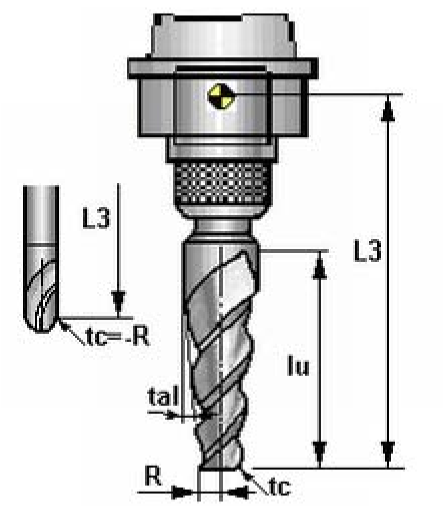
	BQ3(decimal value)	5938176	
Screen			

Fig.25-35: 2012

Technology		Milling tools								
Tool type name		Thread milling cutter								
Tool type	IKQ2	2013								
Reserve	BQ3	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
	(Bit 32 ... 25)									
Technology data	BQ3	ET1	T8	T7	T6	T5	T4	T3	T2	T1
	(Bit 24 ... 16)	-	nt	-	tp	lu	-	-	-	d
			x		x	x				x
Correction type	BQ3	DIA	O	R2	R1	L3	L2	L1		
	(Bit 15 .. 0.9)				x	x				
Reserve	BQ3	TRC4	TRC3	TRC2	TRC1					
	(Bit 8 .. 0.5)									
Edge number	BQ3	EdNo								
	(Bit 1 ..4)	1								

Annex

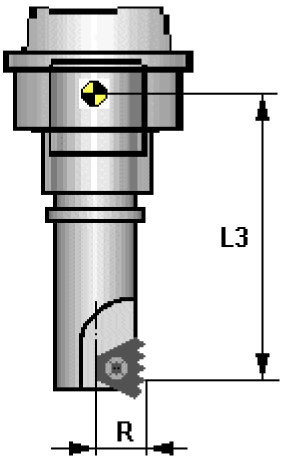
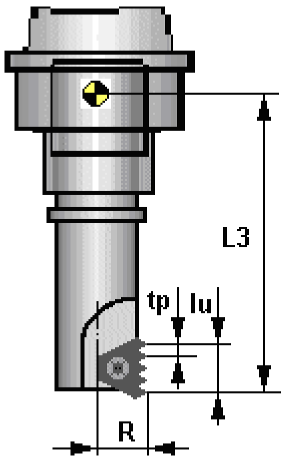
	BQ3(decimal value)	5803008	
Screen			

Fig.25-36: *2013*

Technology		Milling tools							
Tool type name		Engraving tool							
Tool type	IKQ2	2014							
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	nt	tac	-	lu	-	-	-	d
			x	x		x				x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
				x	x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

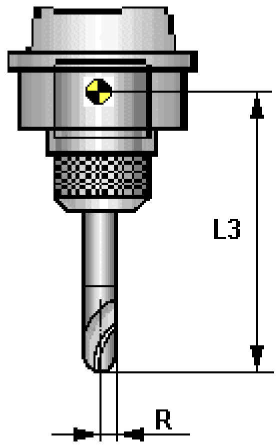
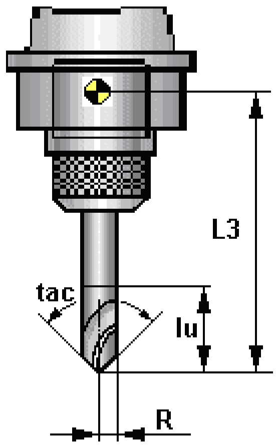
	BQ3(decimal value)	6331392	
Screen			

Fig.25-37: 2014

Technology		Milling tools								
Tool type name		Special milling tool								
Tool type	IKQ2	2015								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	nt	-	-	-	-	-	-	d
			x							x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
				x	x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		can be set from 1 - 16								

Annex

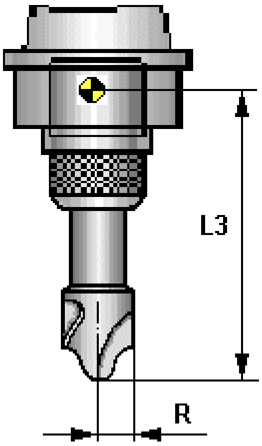
EdNo = 1	BQ3(decimal value)	4234240	
:		:	
EdNo = 16		4234255	
Screen			

Fig.25-38: 2015

Technology		Turning tools								
Tool type name		Turning tool, general								
Tool type	IKQ2	3000								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	-	-	-	-	-
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		can be set from 1 - 16								

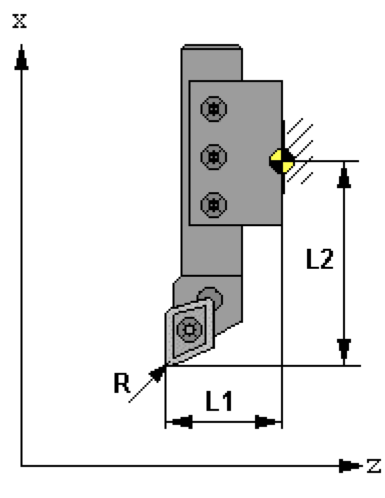
EdNo = 1	BQ3(decimal value)	27392
:		:
EdNo = 16		27407
Screen		

Fig.25-39: 3000

Technology		Turning tools								
Tool type name		Turning tool, outside radial right								
Tool type	IKQ2	3001								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tal	-	-	tc	-	-
				x	x			x		
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

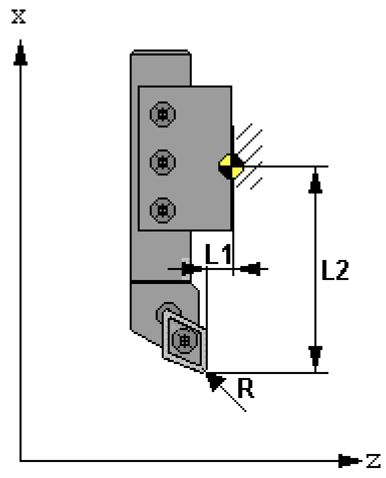
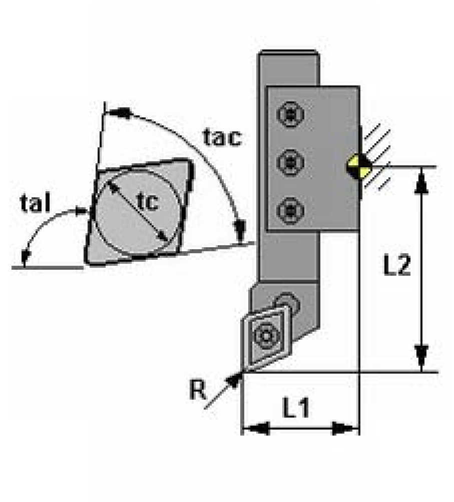
	BQ3(decimal value)	3304192	
Screen			

Fig.25-40: 3001

Technology		Turning tools								
Tool type name		Turning tool, outside radial left								
Tool type	IKQ2	3002								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tal	-	-	tc	-	-
				x	x			x		
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

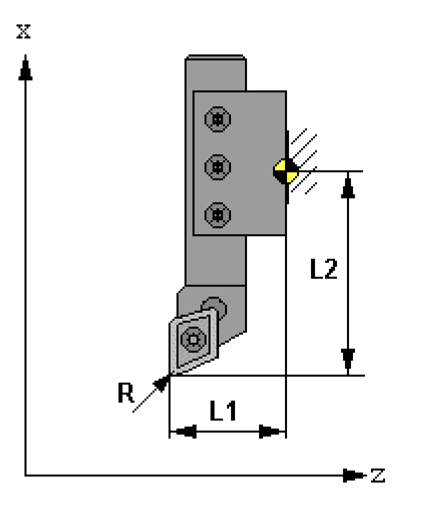
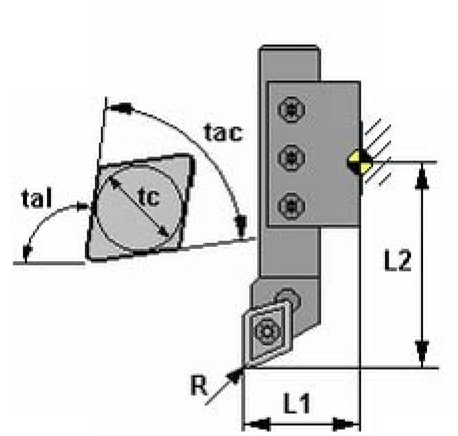
	BQ3(decimal value)	3304192	
Screen			

Fig.25-41: 3002

Technology		Turning tools								
Tool type name		Turning tool, outside axial right								
Tool type	IKQ2	3003								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tal	-	-	tc	-	-
				x	x			x		
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

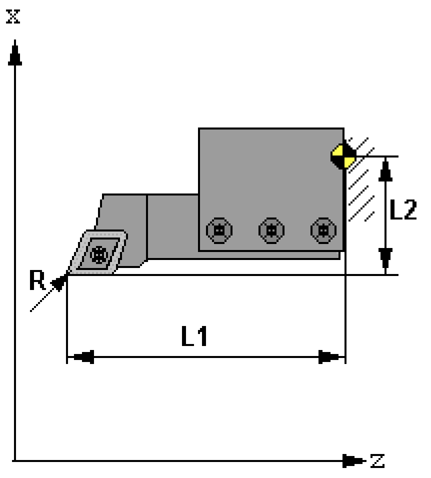
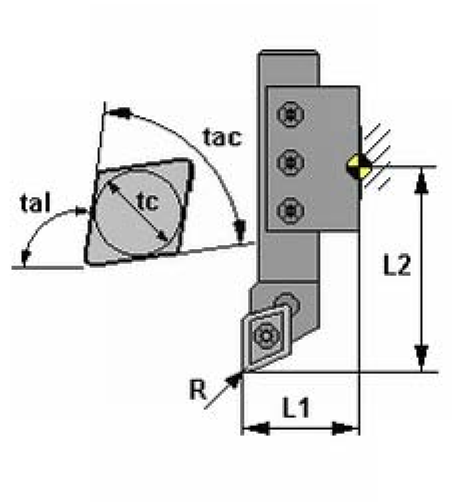
	BQ3(decimal value)	3304192	
Screen			

Fig.25-42: 3003

Technology		Turning tools								
Tool type name		Turning tool, outside axial left								
Tool type	IKQ2	3004								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tal	-	-	tc	-	-
				x	x			x		
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

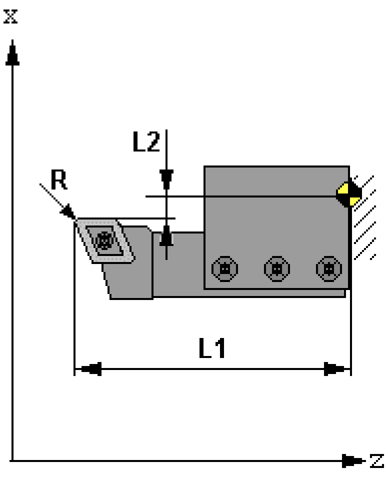
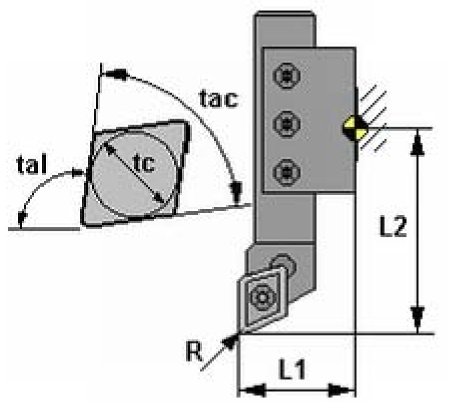
	BQ3(decimal value)	3304192	
Screen			

Fig.25-43: 3004

Technology		Turning tools								
Tool type name		Turning tool, inside right								
Tool type	IKQ2	3005								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tal	lu	-	tc	ds	-
				x	x	x		x	x	
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

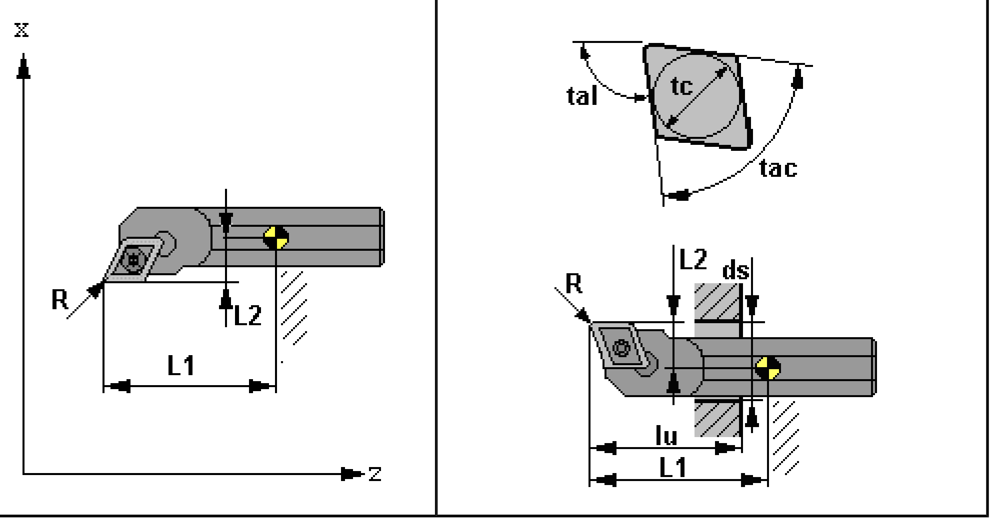
	BQ3(decimal value)	3894016	
Screen			

Fig.25-44: 3005

Technology		Turning tools								
Tool type name		Turning tool, inside left								
Tool type	IKQ2	3006								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tal	lu	-	tc	ds	-
				x	x	x		x	x	
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

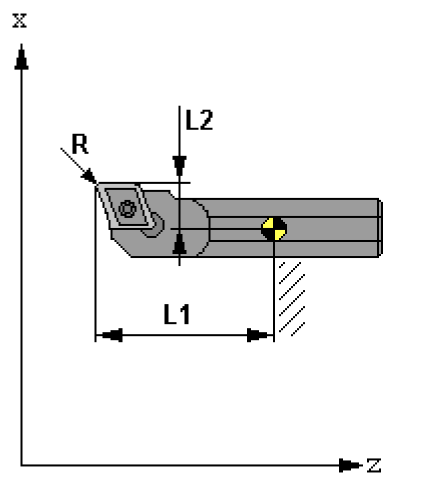
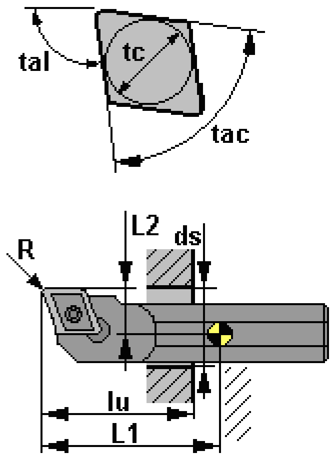
	BQ3(decimal value)	3894016	
Screen			

Fig.25-45: 3006

Technology		Turning tools								
Tool type name		Plunging turn tool, outside right								
Tool type	IKQ2	3007								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	lu	tl	tw	-	-
						x	x	x		
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		2								

Annex

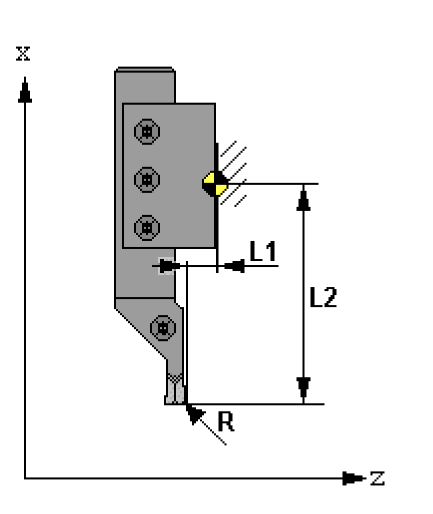
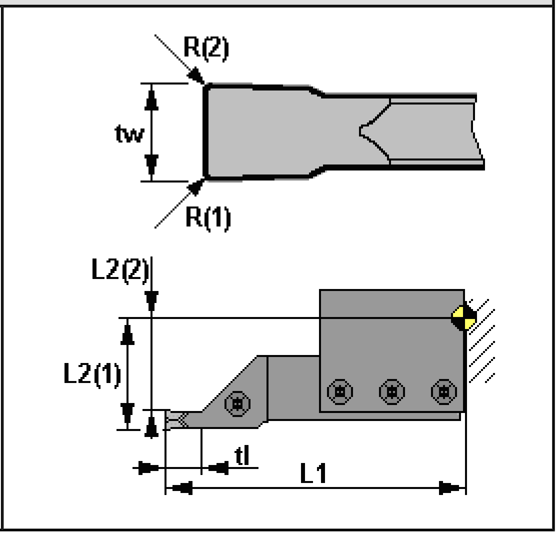
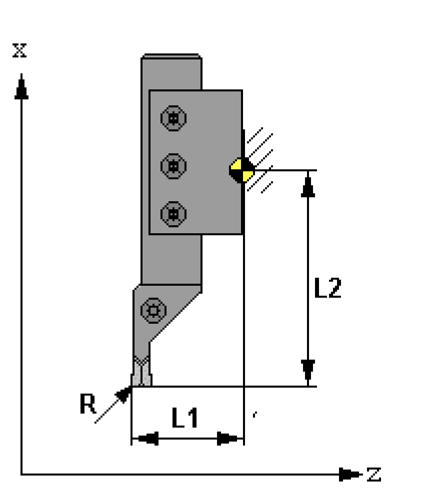
	BQ3(decimal value)	944897	
Screen			

Fig.25-46: 3007

Technology		Turning tools								
Tool type name		Plunging turn tool, outside left								
Tool type	IKQ2	3008								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	lu	tl	tw	-	-
						x	x	x		
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		2								

	BQ3(decimal value)	944897	

Screen



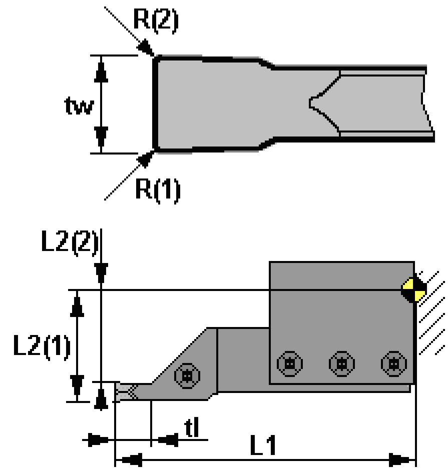


Fig.25-47: 3008

Technology		Turning tools								
Tool type name		Plunging turn tool, inside right								
Tool type	IKQ2	3009								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	lu	tl	tw	ds	-
						x	x	x	x	
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		2								

Annex

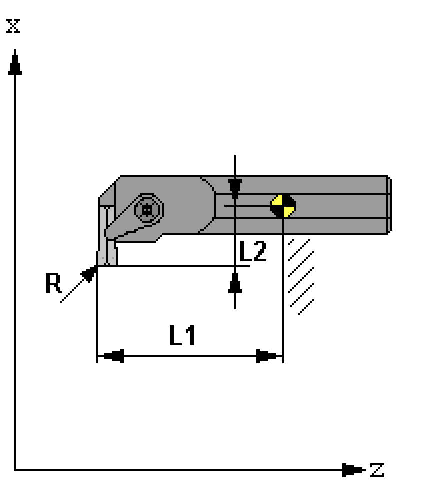
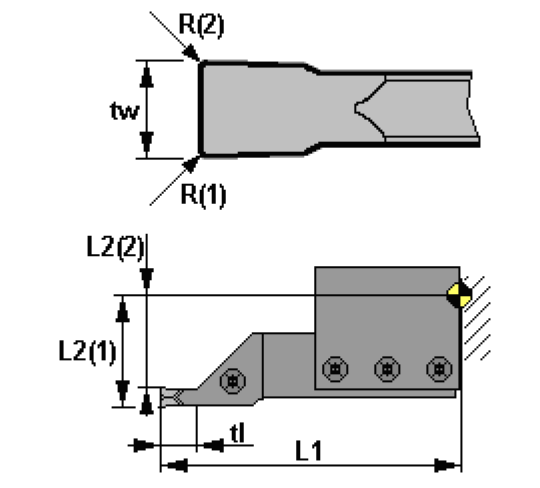
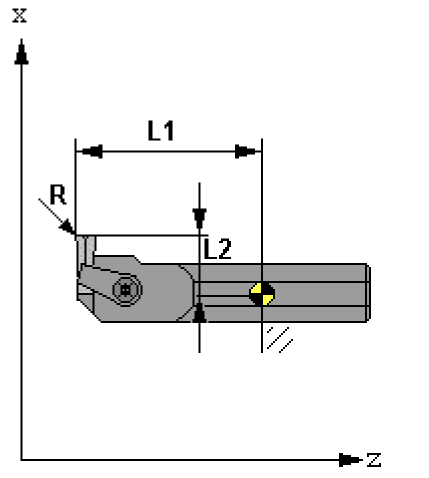
	BQ3(decimal value)	1010433	
Screen			

Fig.25-48: *3009*

Technology		Turning tools								
Tool type name		Plunging turn tool, inside left								
Tool type	IKQ2	3010								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	lu	tl	tw	ds	-
						x	x	x	x	
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		2								

	BQ3(decimal value)	1010433	
Screen			



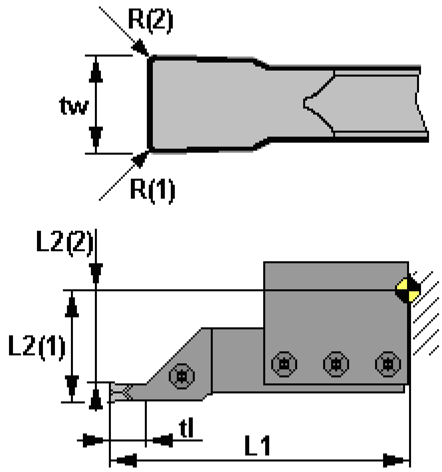


Fig.25-49: 3010

Technology		Turning tools								
Tool type name		Plunging turn tool, axial right								
Tool type	IKQ2	3011								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	lu	tl	tw	ds	d
						x	x	x	x	x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		2								

Annex

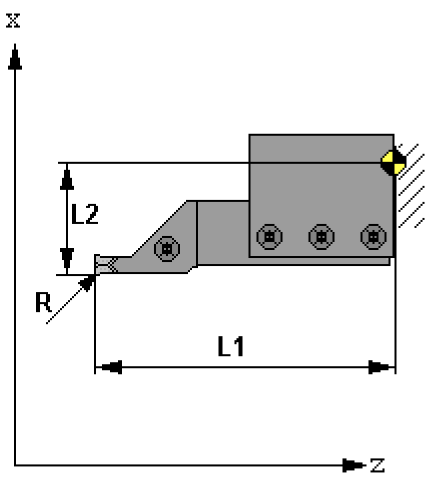
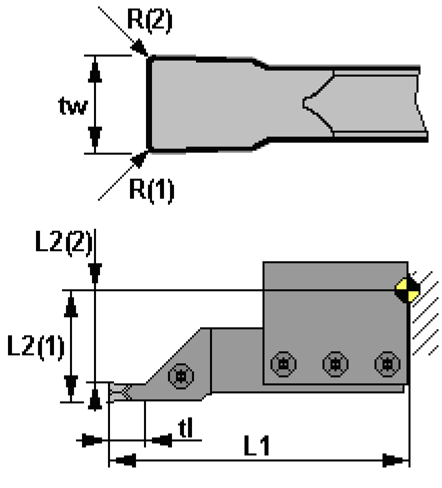
	BQ3(decimal value)	1043201	
Screen			

Fig.25-50: 3011

Technology		Turning tools								
Tool type name		Plunging turn tool, axial left								
Tool type	IKQ2	3012								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	lu	tl	tw	ds	d
						x	x	x	x	x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		2								

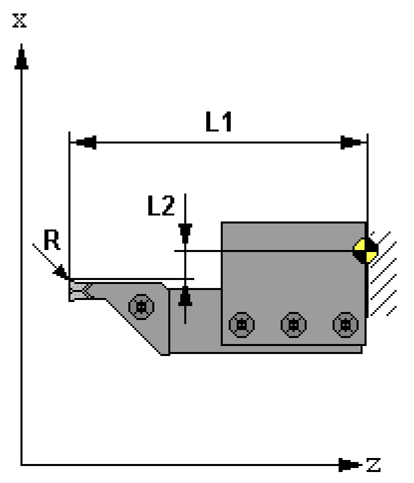
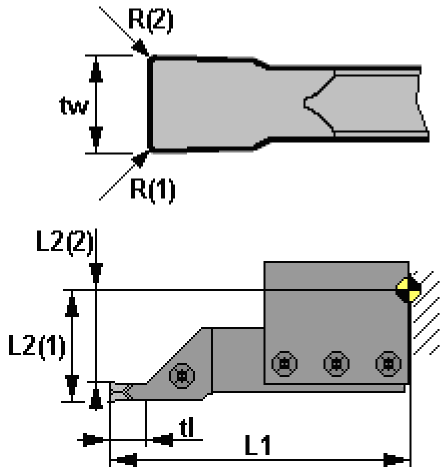
	BQ3(decimal value)	1043201	
Screen			

Fig.25-51: 3012

Technology		Turning tools							
Tool type name		Cut-off tool							
Tool type	IKQ2	3013							
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	tal	-	tl	tw	-	-
					x		x	x		
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

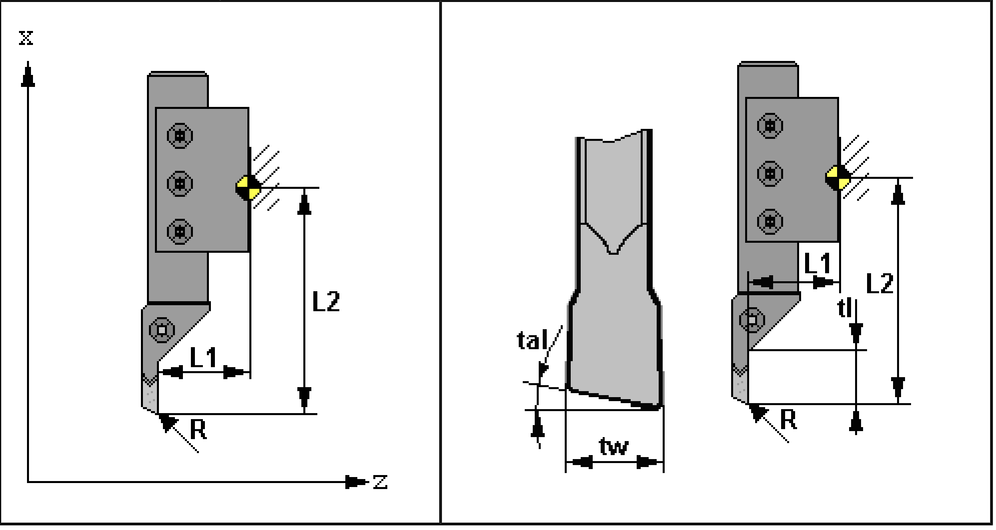
	BQ3(decimal value)	1469184	
Screen			

Fig.25-52: 3013

Technology		Turning tools								
Tool type name		Form turning tool, right								
Tool type	IKQ2	3014								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	tal	-	-	-	-	d
					x					x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

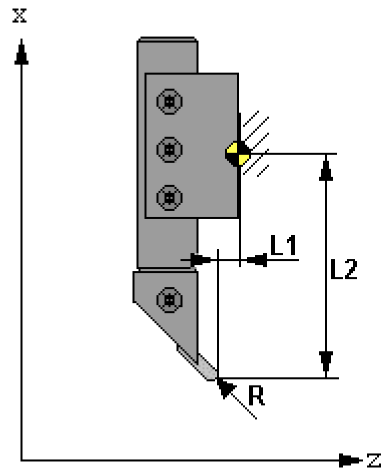
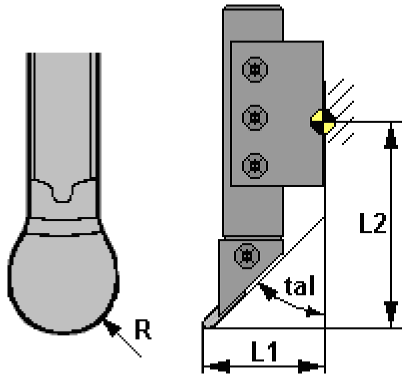
	BQ3(decimal value)	1108736	
Screen			

Fig.25-53: 3014

Technology		Turning tools								
Tool type name		Form turning tool, left								
Tool type	IKQ2	3015								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	tal	-	-	-	-	d
					x					x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

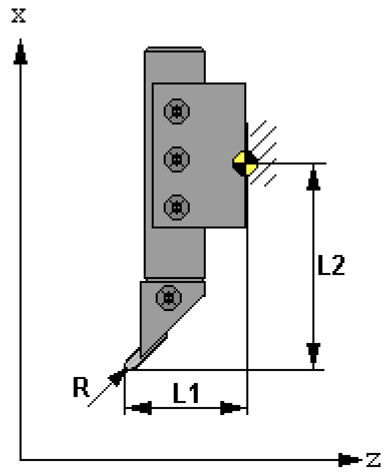
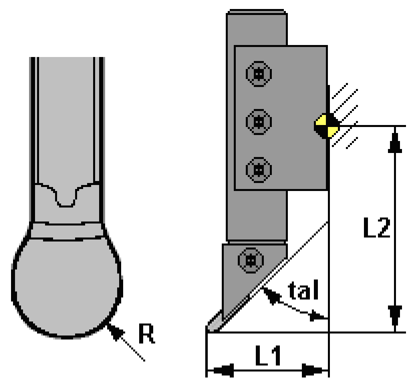
	BQ3(decimal value)	1108736	
Screen			

Fig.25-54: 3015

Technology		Turning tools								
Tool type name		Thread turning tool, outside right								
Tool type	IKQ2	3016								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tp	lu	tl	tw	ds	-
				x	x	x	x	x	x	
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

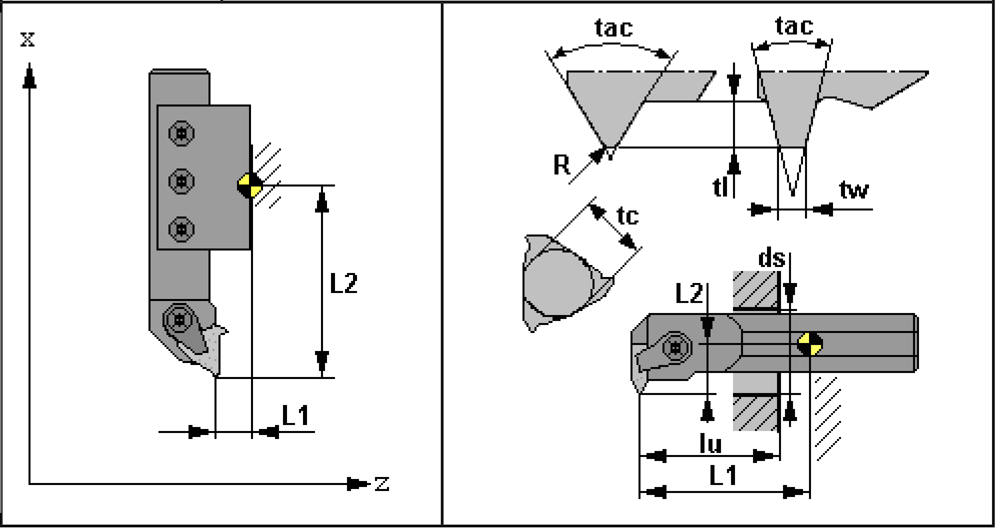
	BQ3(decimal value)	4156160	
Screen			

Fig.25-55: 3016

Technology		Turning tools								
Tool type name		Thread turning tool, outside left								
Tool type	IKQ2	3017								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tp	lu	tl	tw	ds	-
				x	x	x	x	x	x	
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

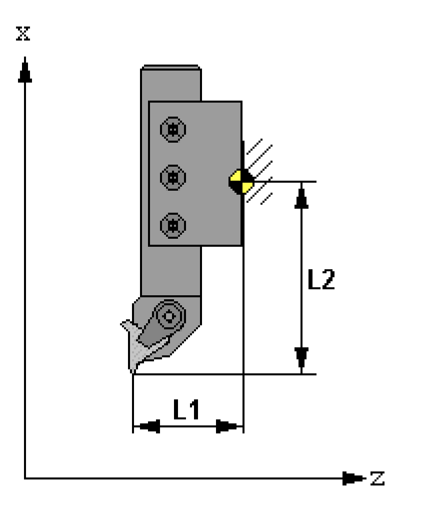
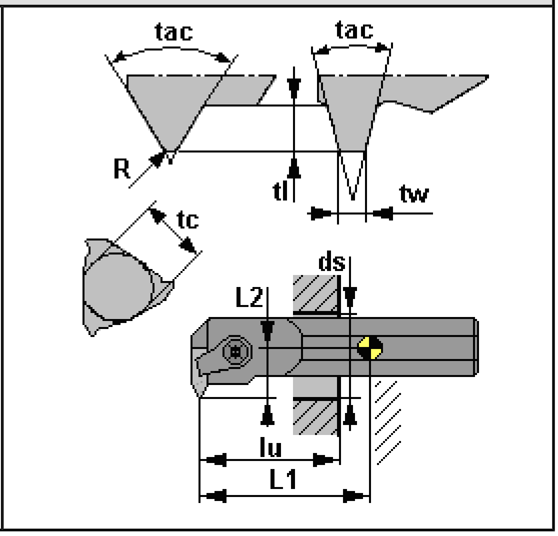
	BQ3(decimal value)	4156160	
Screen			

Fig.25-56: 3017

Technology		Turning tools								
Tool type name		Thread turning tool, inside right								
Tool type	IKQ2	3018								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tp	lu	tl	tw	ds	-
				x	x	x	x	x	x	
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

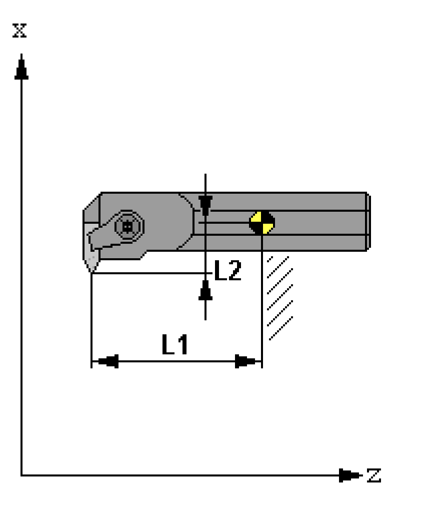
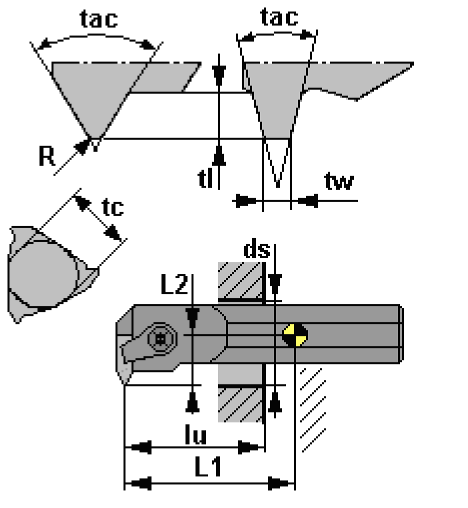
<table>
<tr><td></td><td>BQ3(decimal value)</td><td>4156160</td><td></td></tr>
<tr><td>Screen</td><td></td><td>
</td><td>
</td></tr>
</table>

Fig.25-57: 3018

Technology		Turning tools								
Tool type name		Thread turning tool, inside left								
Tool type	IKQ2	3019								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tp	lu	tl	tw	ds	-
				x	x	x	x	x	x	
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

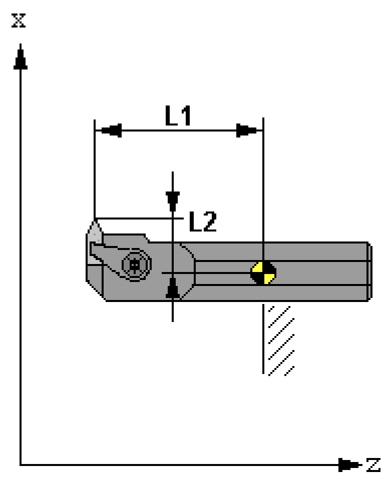
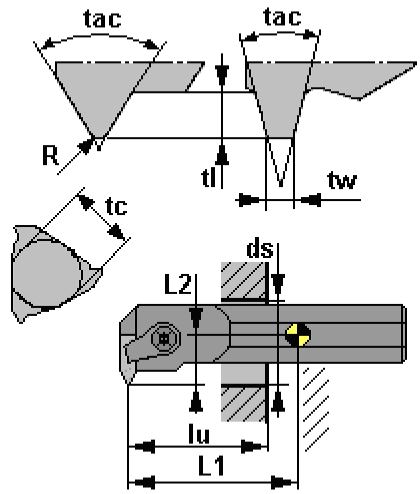
	BQ3(decimal value)	4156160	
Screen			

Fig.25-58: 3019

Technology		Turning tools								
Tool type name		Special turning tool								
Tool type	IKQ2	3020								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	-	-	-	-	-
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		can be set from 1 - 16								

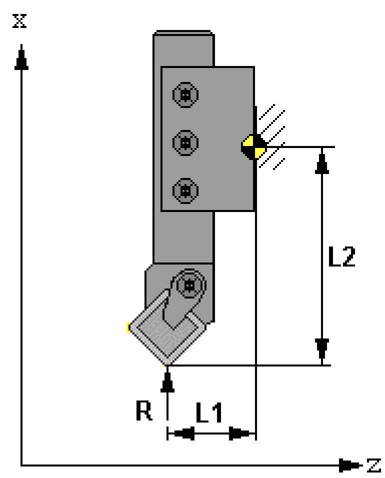
EdNo = 1	BQ3(decimal value)	27392	
:		:	
EdNo = 16		27407	
Screen			

Fig.25-59: 3020

Technology		Turning tools								
Tool type name		Turning tool, outside radial right (2S)								
Tool type	IKQ2	3021								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tal	-	-	-	-	-
				x	x					
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		2								

Annex

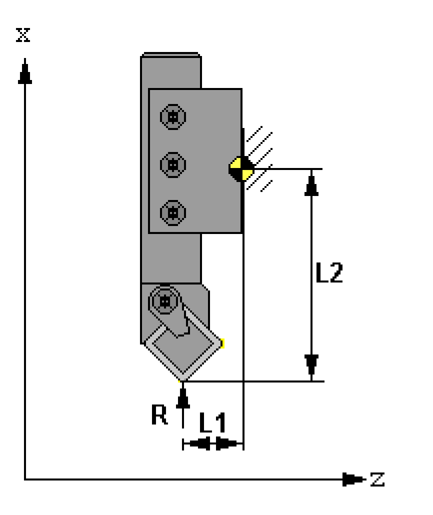
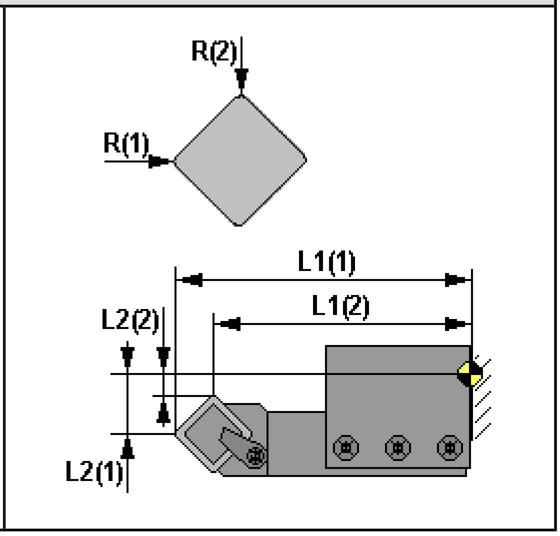
	BQ3(decimal value)	3762945	
Screen			

Fig.25-60: 3021

Technology		Turning tools								
Tool type name		Turning tool, outside radial left (2S)								
Tool type	IKQ2	3022								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tal	-	-	-	-	-
				x	x					
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		2								

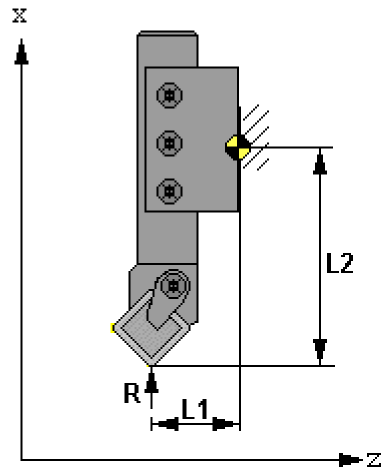
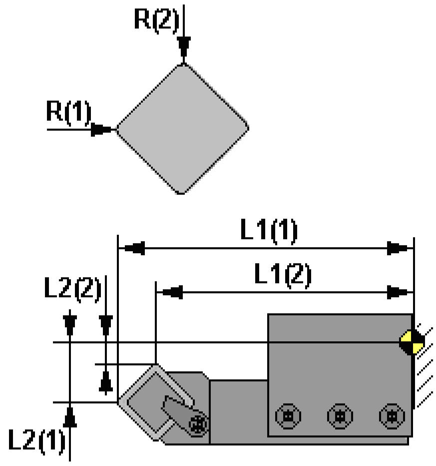
	BQ3(decimal value)	3762945	
Screen			

Fig.25-61: 3022

Technology		Turning tools								
Tool type name		Turning tool, outside axial right (2S)								
Tool type	IKQ2	3023								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tal	-	-	-	-	-
				x	x					
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		2								

Annex

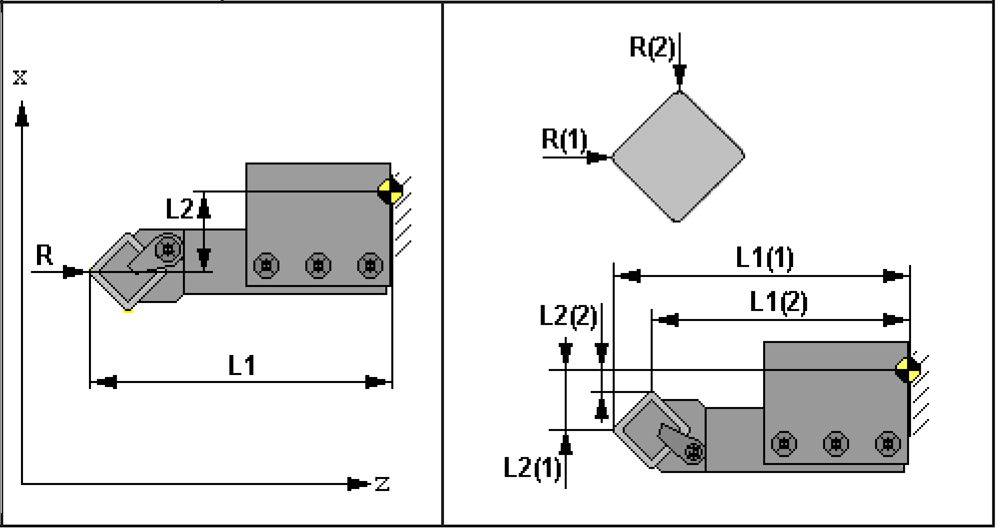
	BQ3(decimal value)	3762945	
Screen			

Fig.25-62: 3023

Technology		Turning tools								
Tool type name		Turning tool, outside axial left (2S)								
Tool type	IKQ2	3024								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tal	-	-	-	-	-
				x	x					
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		2								

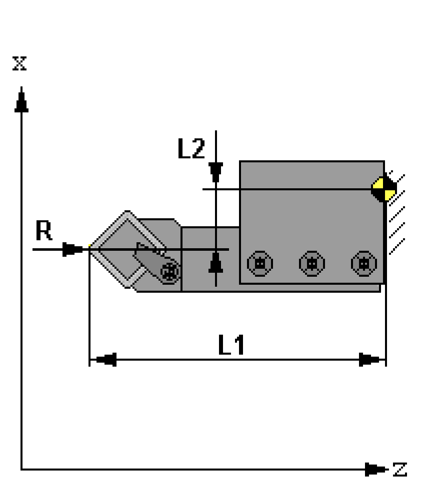
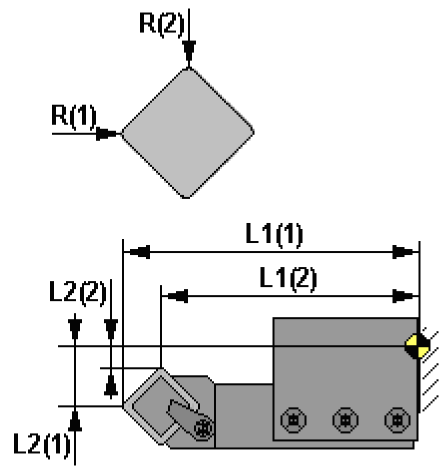
	BQ3(decimal value)	3762945	
Screen			

Fig.25-63: 3024

Technology		Turning tools								
Tool type name		Turning tool, inside right (2S)								
Tool type	IKQ2	3025								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tal	lu	-	-	ds	-
				x	x	x			x	
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		2								

Annex

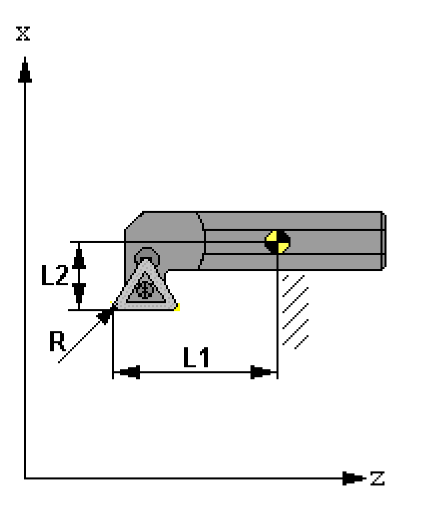
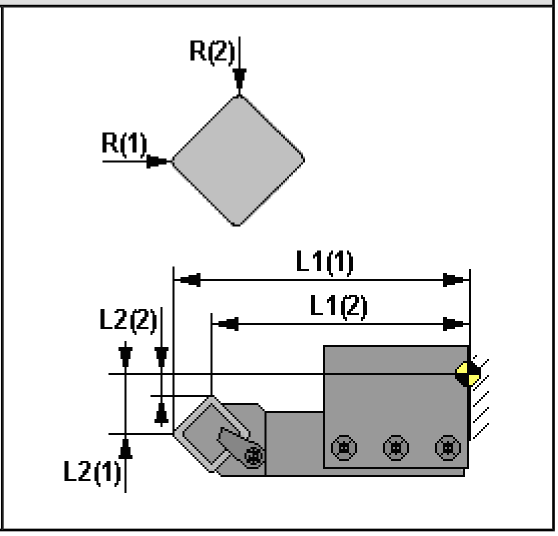
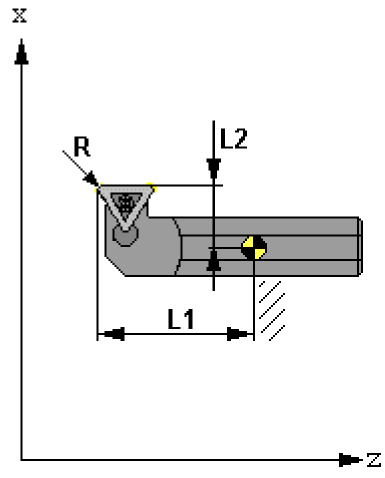
	BQ3(decimal value)	3762945	
Screen			

Fig.25-64: 3025

Technology		Turning tools								
Tool type name		Turning tool, inside left (2S)								
Tool type	IKQ2	3026								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	tac	tal	lu	-	-	ds	-
				x	x	x			x	
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		2								

	BQ3(decimal value)	3746561	

Screen



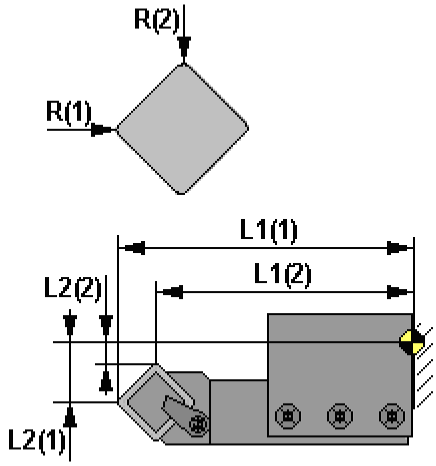


Fig.25-65: 3026

Technology		Turning tools							
Tool type name		EcoCut							
Tool type	IKQ2	3027							
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	lu	-	-	ds	-
						x			x	
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
		x	x		x		x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

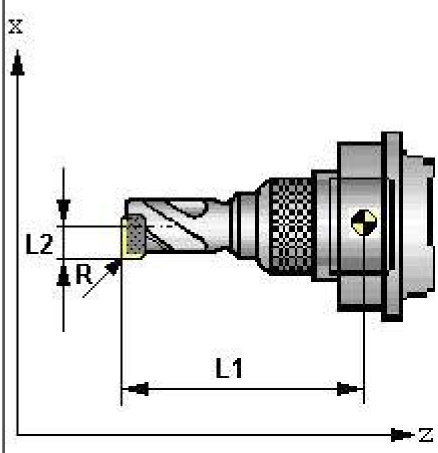
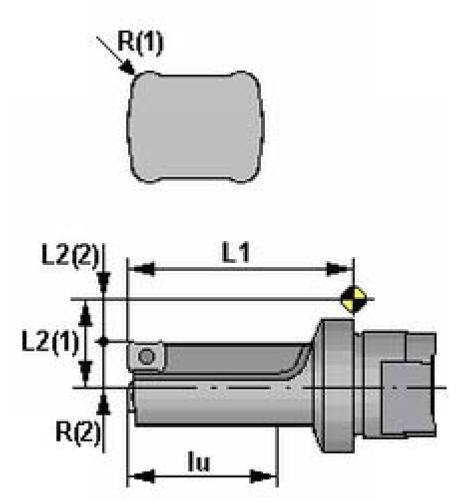
	BQ3(decimal value)	2648832	
Screen			

Fig.25-66: 3027

Technology		Special tools								
Tool type name		Grippers								
Tool type	IKQ2	5000								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	-	-	-	-	-
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
						x	x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

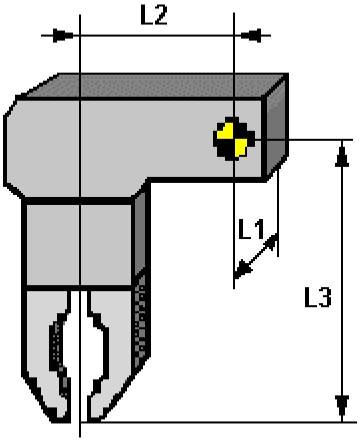
	BQ3(decimal value)	1792	
Screen			

Fig.25-67: 5000

Technology		Special tools								
Tool type name		Probe, general								
Tool type	IKQ2	5001								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	-	-	-	-	-
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
					x	x				
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

Annex

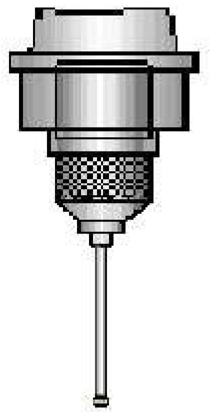
	BQ3(decimal value)	3072	
Screen			

Fig.25-68: 5001

Technology		Special tools								
Tool type name		3D probe								
Tool type	IKQ2	5002								
Reserve	BQ3 (Bit 32 ... 25)	TRC12	TRC11	TRC10	TRC9	TRC8	TRC7	TRC6	TRC5	
Technology data	BQ3 (Bit 24 ... 16)	ET1	T8	T7	T6	T5	T4	T3	T2	T1
		-	-	-	-	-	-	-	-	d
										x
Correction type	BQ3 (Bit 15 .. 0.9)	DIA	O	R2	R1	L3	L2	L1		
					x	x	x	x		
Reserve	BQ3 (Bit 8 .. 0.5)	TRC4	TRC3	TRC2	TRC1					
Edge number	BQ3 (Bit 1 ..4)	EdNo								
		1								

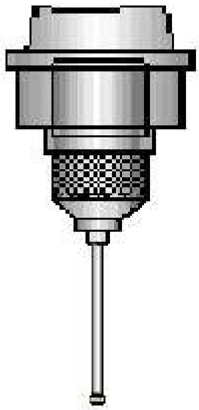
	BQ3(decimal value)	36608	
Screen			

Fig.25-69: 5002

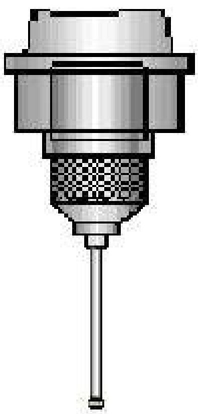
Technology	Special tools												
Tool type name	3D probe												
Tool type	IKQ2	5002											
Technology data	aae	ame	le	we	nt	lu	ll	lfc	tp	as	adb	ds	d
													x
Correction type	DIA	O	R2	R1	L3	L2	L1						
				x	x	x	x						
Edge number	EdNo												
	1												
BQ3 value	528128												
Screen													

Fig.25-70: 5002

Annex

Element	tech. label	Meaning	Type dependency of the meaning
L1		1. Length correction value	-
L2		2. Length correction value	-
L3		3. Length correction value	-
R1		big radius	-
R2		small radius	-
O		Edge position	-
DIA		Tool with diameter information	-
T1	d	Diameter	Drilling and milling tools
		Greater diameter	
		Milling diameter	
		Outer diameter	
		outer limitation of the cutting diameter area	Turning tools
		Edge radius	
T2	ds	Minor diameter	Drilling tools
		Shank diameter	
		Drilling diameter	
		Smallest diameter that can be machined	Turning tools
		inner limitation of the cutting diameter area	
T3	lfc	Chamfer length	Drilling and milling tools
	ll	Loss length	
	tw	Tool edge width	div. turning tools
		Width of trapezoid thread	
	tc	Corner radius	Div. milling tools
		Tool tip size	div. turning tools
T4	tl	Max. cutting height	Turning tools
		Max. thread height	
T5	lu	Usable length	-
T6	tp	(Thread)lead	Drilling, milling or turning tools
	ta	Setting angle	div. drilling tools
	tal	Orientation angle	div. turning tools
		Angle (edge angle)	div. milling or turning tools
		Tool setting angle	div. turning tools
		Angle (kappa)	Div. milling tools
		Angle	

Element	tech. label	Meaning	Type dependency of the meaning
T7	tac	Apical angle	Drilling and milling tools
		Countersink angle	
		Corner angle	Turning tools
		Flank angle	
T8	nt	Number of teeth	Milling tools
ET1	ds	Small diameter (edge-dependent)	Step drill

Fig.25-71: Tool catalog key

25.3.3 Bitmap Libraries

The following bitmap libraries are contained when delivered:

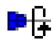
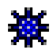
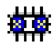
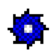
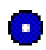
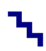
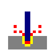
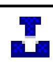
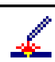
BitmapResourceName	Bitmap name (value)	Example (original size)	Explanation
ToolStorageBmp.resx	spindle16_g.gif		Spindle
	slot16_g.gif		Turret
	magazin16_g.gif		Magazine feeding attachment
	grip16_g.gif		Grippers
	schrank.gif		Tool cabinet

Fig.25-72: Bitmap library of tool memory

BitmapResourceName	Bitmap name (value)	Example (original size)	Explanation
ToolTecTypesBmp.resx	drill16_g.gif		Drilling
	mill16_g.gif		Milling
	turn16_g.gif		Turning
	grind16_g.gif		Grinding
	cut16_g.gif		Tool edges
	form16_g.gif		Reshaping
	erode16_g.gif		Eroding
	nibble16_g.gif		Nibbling
	punch16_g.gif		Punching
	screw16_g.gif		Screwing
	weld16_g.gif		Welding

Annex

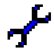
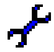
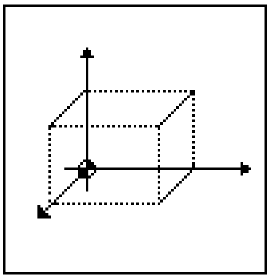
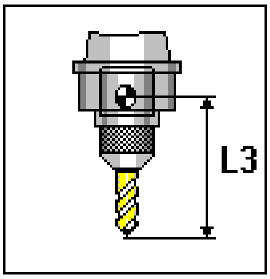
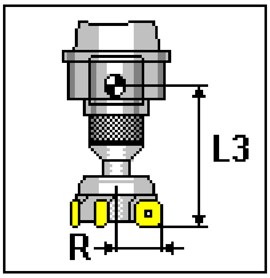
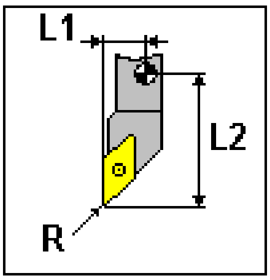
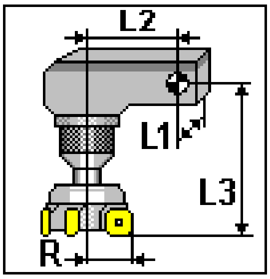
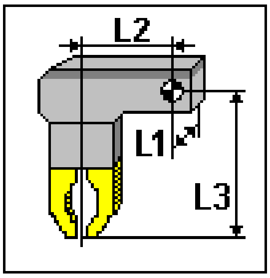
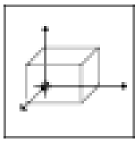
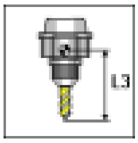
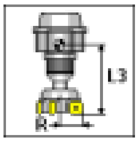
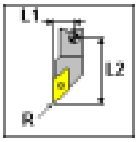
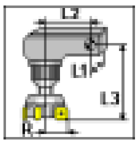
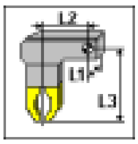
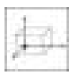
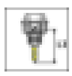
BitmapResourceName	Bitmap name (value)	Example (original size)	Explanation
	wrench16_g.gif		-
	mtctds_TOOL_g.gif		-
	special16_g.gif		-
	tool16_g.gif		-

Fig.25-73: Bitmap library of tool technology

BitmapResourceName	Bitmap name (value)	Example (original size)	Explanation
ToolCorrTypeBmp.resx	CorrType_0.gif		Correction type 0 (gen. tool type)
	CorrType_1.gif		Correction type 1 (drilling tools)
	CorrType_2.gif		Correction type 2 (milling tools)
	CorrType_3.gif		Correction type 3 (turning tools)

BitmapResourceName	Bitmap name (value)	Example (original size)	Explanation
	CorrType_4.gif		Correction type 4 (special milling tools)
	CorrType_5.gif		Correction type 5 (special tools)
	CorrType_0_k.gif		Correction type 0 (gen. tool type)
	CorrType_1_k.gif		Correction type 1 (drilling tools)
	CorrType_2_k.gif		Correction type 2 (milling tools)
	CorrType_3_k.gif		Correction type 3 (turning tools)
	CorrType_4_k.gif		Correction type 4 (special milling tools)
	CorrType_5_k.gif		Correction type 5 (special tools)
	CorrType_0_kk.gif		Correction type 0 (gen. tool type)
	CorrType_1_kk.gif		Correction type 1 (drilling tools)

Annex

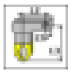
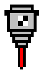
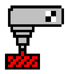
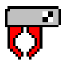
BitmapResourceName	Bitmap name (value)	Example (original size)	Explanation
	CorrType_2_kk.gif		Correction type 2 (milling tools)
	CorrType_3_kk.gif		Correction type 3 (turning tools)
	CorrType_4_kk.gif		Correction type 4 (special milling tools)
	CorrType_5_kk.gif		Correction type 5 (special tools)
	Kt1k.gif		Correction type 1 (drilling tools)
	Kt2k.gif		Correction type 2 (milling tools)
	Kt3k.gif		Correction type 3 (turning tools)
	Kt4k.gif		Correction type 4 (special milling tools)
	Kt5k.gif		Correction type 5 (special tools)

Fig.25-74: Bitmap library of tool correction types

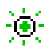
BitmapResourceName	Bitmap name (value)	Example (original size)	Explanation
ToolStatusBmp.resx	checkbox_false.gif		
	checkbox_true.gif		
	hookred.gif		
	hookwhite.gif		
	ledgreen.gif		
	ledred.gif		
	ledyellow.gif		
	stateundefined16.gif		Undefined
	statealarm16.gif		Alarm
	statedisabled16.gif		Inactive
	stateok16.gif		OK

Fig.25-75: Bitmap library of status messages

☞ In addition, there are 3 more bitmap libraries (as in fig. 16-103 "Supplied bitmap libraries" on page 399) for displaying the tool types in different sizes.

26 Service and Support

Our service helpdesk at our headquarters in Lohr, Germany and our worldwide service will assist you with all kinds of enquiries. You can reach us **around the clock - even on weekend and on holidays**.

	Helpdesk	Service Hotline Worldwide
Phone	+49 (0) 9352 40 50 60	Outwith Germany please contact our sales/service office in your area first. For hotline numbers refer to the sales office addresses on the Internet.
Fax	+49 (0) 9352 40 49 41	
E-mail	service.svc@boschrexroth.de	
Internet	http://www.boschrexroth.com You will also find additional notes regarding service, maintenance (e.g. delivery addresses) and training.	

Preparing Information

For quick and efficient help please have the following information ready:

- Detailed description of the fault and the circumstances
- Information on the type plate of the affected products, especially type codes and serial numbers
- Your phone, fax numbers and e-mail address so we can contact you in case of questions.

Index

Index

C

C

C

D

Index

D

E

E

F

G

G

H

H

Index

Index

O

P

Index

X

X

Z

Notes

Bosch Rexroth AG
Electric Drives and Controls
P.O. Box 13 57
97803 Lohr, Germany
Bgm.-Dr.-Nebel-Str. 2
97816 Lohr, Germany
Tel. +49 (0)93 52-40-0
Fax +49 (0)93 52-48 85
www.boschrexroth.com/electrics

R911334377

Printed in Germany
DOK-MTX***-STARTUP*V12-CO01-EN-P