

MTX 15VRS

NC Simulation Multitouch

Application Description
R911399601

Edition 02



Title	MTX 15VRS NC Simulation Multitouch
Type of Documentation	Application Description
Document Typecode	DOK-MTX***-NC*SIM*M*15-AP02-EN-P
Internal File Reference	RS-725519e2699cadafc0a802865c88864e-2-en-US-3
Change Record	Edition 02, 2020-07 Refer to tab. 1-1 "Change Record" on page 1
Copyright	© Bosch Rexroth AG 2020 All rights reserved, also regarding any disposal, exploitation, reproduction, editing, distribution, as well as in the event of applications for industrial property rights.
Liability	The specified data is intended for product description purposes only and shall not be deemed to be a guaranteed characteristic unless expressly stipulated in the contract. All rights are reserved with respect to the content of this documentation and the availability of the product.
Editorial Department	Engineering automation systems solution integration HMI, NC controls and support, BeZe (TaDo/MePe)

Table of Contents

	Page
1 About this documentation.....	1
1.1 Validity of the documentation.....	1
1.2 Notation, fonts, symbols.....	2
1.3 Prerequisites.....	2
1.4 Required and supplementing documentation MTX.....	2
1.4.1 Selection/compilation.....	2
1.4.2 Configuration.....	3
1.4.3 Commissioning.....	3
1.4.4 Operation.....	4
1.4.5 OEM Engineering.....	5
1.4.6 AddOns.....	5
1.5 Information representation.....	6
1.5.1 Using safety instructions.....	6
1.5.2 Names and abbreviations.....	8
1.6 Customer feedback.....	8
2 Important instructions on use.....	9
2.1 Intended use.....	9
2.1.1 Introduction.....	9
2.1.2 Areas of use and application.....	9
2.2 Unintended use.....	10
3 Overview.....	11
3.1 Function.....	11
3.2 Starting and exiting the simulation.....	13
3.3 Layout.....	15
4 Operation.....	19
4.1 Operating keys and corresponding dialogs.....	19
4.1.1 Shortcuts of the simulation.....	19
4.1.2 Extended operating keyboard of the simulation.....	20
4.2 Shortcuts.....	36
5 Commissioning.....	39
5.1 Commissioning procedure: Example of milling with a tool management.....	39
5.2 Machine definition.....	40
5.2.1 Machine model and kinematic model.....	40
5.2.2 Configuration file.....	42
5.3 Resource definition.....	48
5.3.1 Tool definition.....	48
5.3.2 Raw part definition.....	50
5.3.3 Clamping device definition.....	54
5.4 Program definition.....	55

Table of Contents

	Page
5.4.1 NC program.....	55
5.4.2 Zero offsets und placements.....	56
6 Recording and playing recordings.....	57
7 Program instructions for the simulation.....	59
7.1 General information on program instructions for the simulation.....	59
7.2 STOCK/BLOCK Rectangular raw part (milling)	59
7.3 STOCK/CYLINDER Cylindrical raw part (milling)	61
7.4 STOCK/BAR Bar section (turning).....	63
7.5 STOCK/POLYGON Polygon (turning).....	65
7.6 STOCK/SHEET Sheet (punching).....	68
7.7 STOCKIMP Import model.....	70
7.8 STOCK/REMOVE Remove workpieces/models.....	73
7.9 STOCK/RENEW Renew workpiece.....	74
7.10 STOCKSEL Select workpiece holder.....	76
7.11 STOCKADD Replace/add workpieces/models.....	77
7.12 TOOL/DRILL Drill, countersink, reamer.....	80
7.13 TOOL/CDRILL Step drill, centering drill.....	81
7.14 TOOL/BBAR Boring bar, backward counterbore.....	83
7.15 TOOL/MILL Cylindrical, angular and ball cutter.....	85
7.16 TOOL/CHAMFER Engraving milling cutter, chamfer milling cutter, face milling head.....	88
7.17 TOOL/CRMILL Quarter round profile milling cutter.....	91
7.18 TOOL/GRIND Grinding wheel.....	93
8 Configuration.....	105
8.1 Overview.....	105
8.2 Introduction to the tool graphic generation.....	108
8.3 Description of the configuration file MTX.Simulation.Options.XML or MTX.Simulation.Options.Temp.XML.....	120
8.3.1 File structure.....	120
8.3.2 Configuring the NC simulation.....	122
8.3.3 Simulation options.....	138
9 Integrating the MTX simulation control into third-party applications.....	147
9.1 Preparation.....	147
9.2 Behavior of the simulation control.....	147
9.3 Sample project.....	149
10 Service and support.....	153
Index.....	155

1 About this documentation

Editions of this documentation

Edition	Release date	Note
01	2019-07	First edition
02	2020-07	chapter 4.1.2 "Extended operating keyboard of the simulation" on page 20 : Keys added (WP-00076634) chapter 5.2.1 "Machine model and kinematic model" on page 40 supplemented chapter "Referencing the machine model" on page 42 modified chapter 8.3.2 "Configuring the NC simulation" on page 122 , subchapter "Machine" added chapter "Settings for display" on page 141 : Opening angle of the camera, "Transparency of the machine housing" supplemented chapter 9 "Integrating the MTX simulation control into third-party applications" on page 147 modified

Tab. 1-1: Change Record

1.1 Validity of the documentation

Purpose This documentation describes the operation of the standard user interface of the MTX NC control.

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

Overview on target groups and product phases

The following graphic refers to the bordered activities, product phases and target groups of the present documentation.

Example: The target group "Programmer" can execute the "Configuration" in the product phase "Engineering" using this documentation.

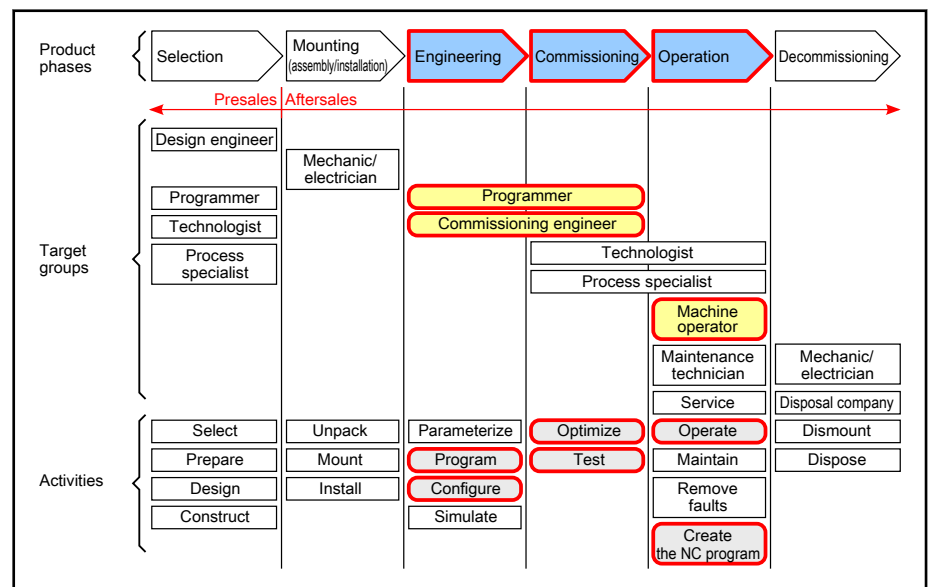


Fig. 1-1: Assigning this documentation to the target groups, product phases and target group activities

Target group This documentation is intended for users commissioning an MTX control. 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 during the following phases:

- Engineering
- Commissioning

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

1.2 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 have to be pressed **in sequence** are separated by a comma, e.g. <1>, <2>, <Enter>.

Keys that have to be 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 the 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 "Save" menu sub-item in the "File" menu: **File ▶ Save**.

1.3 Prerequisites

To operate the standard user interface of MTX according to this documentation, the control has to be configured and commissioned.

The default settings of the dialogs as delivered are described. For other configurations, refer to "MTX 15VRS Commissioning", DOK-MTX***-STARTUP*V15-CORS-EN-P.

1.4 Required and supplementing documentation MTX

1.4.1 Selection/compilation

Documentation titles with type codes and part numbers

MTX 15VRS System Description DOK-MTX***-SYS*DES*V15-PRRS-EN-P, R911397742 This documentation describes the MTX control. It includes the designs, technical data, interfaces as well as the configuration of the control components.
MTX 14VRS SafeLogic System Overview DOK-MTX***-SL**SYS*V14-PRRS-EN-P, R911341698 This documentation describes the use of the safety control SafeLogic in the MTX.
IndraControl XM42 Controls DOK-CTRL-XM4X*CTRL**-ITRS-EN-P, R911345566 This documentation describes the IndraControl XM42 controls.

RS Corresponding edition
Tab. 1-2: MTX documentation overview - Selecting/composing

1.4.2 Configuration

Documentation titles with type codes and part numbers

MTX 15VRS Machine Parameters DOK-MTX***-MA*PAR**V15-RERS-EN-P, R911400176 This documentation describes structure and the adjustment of available parameters of the MTX. It also includes the functions of the NC configurator and its operation.
MTX 15VRS PLC Interface DOK-MTX***-PLC*INT*V15-PRRS-EN-P, R911400172 This documentation describes the interface signals and the program function blocks for the integrated PLC.
MTX 15VRS Functional Description - Basics DOK-MTX***-NC*F*BA*V15-RERS-EN-P, R911401401 This documentation describes the basic functions of the MTX. The basic commissioning steps and the control functions are provided as description and handling instruction.
MTX 15VRS Functional Description - Extension DOK-MTX***-NC*F*EX*V15-RERS-EN-P, R911393316 This documentation describes the extended functions of the MTX. The basic commissioning steps and the control functions are provided as description and handling instruction.
MTX 15VRS Functional Description - Special Functions DOK-MTX***-NC*F*SP*V15-RERS-EN-P, R911393309 This documentation describes the special functions of the MTX. The basic commissioning steps and the control functions are provided as description and handling instruction.
MTX Free Form Surface Milling DOK-MTX***-FREEFORM***-APRS-EN-P, R911341435 This documentation describes the free form surface milling process with the MTX control. CNC programs generated by a CAD/CAM system are used as basis for the entire process. Overview over the following topics: Description of the MTX or CNC function for freeform surface milling, strategy during the NC parameterization process, boundary conditions when generating CNC programs with CAM software.
MTX Conversion of MTX Projects DOK-MTX***-PROCONV****-PRRS-EN-P, R911342484 This documentation provides support during the conversion of MTX 1.x projects to MTX 2G. This project planning manual provides information on how to convert a project and identifies potential difficulties during the conversion.

RS

Corresponding edition

Tab. 1-3:

MTX documentation overview - Configuring

1.4.3 Commissioning

Documentation titles with type codes and part numbers

MTX 15VRS Commissioning DOK-MTX***-STARTUP*V15-CORS-EN-P, R911393281 This documentation describes the commissioning of the 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.
IndraWorks 15VRS Basic Libraries, IndraLogic 2G DOK-IL*2G*-BASLIB**V15-LIRS-EN-P, R911398633 This documentation describes the system-comprehensive PLC libraries.

IndraWorks 15VRS Field Buses

DOK-IWORKS-FB*****V15-APRS-EN-P, R911393284

This documentation describes the field bus and local periphery connections supported by the MLC and MTX systems. The focus of this documentation lies in the configuration, parameterization, commissioning and diagnostics of different periphery connections. It is the basis for the online help.

IndraWorks 14VRS WinStudio 7.4

DOK-IWORKS-WINSTUD*V14-APRS-EN-P, R911341585

This "User Manual and Technical Reference Book" supports the user in achieving the best results with the "Rexroth WinStudio"™ software. The document provides technical information and step-by-step instructions to create web-enabled HMI/SCADA programs.

IndraWorks 15VRS Software Installation

DOK-IWORKS-SOFTINS*V15-CORS-EN-P, R911393450

This documentation describes the IndraWorks installation.

IndraWorks 15VRS Engineering

DOK-IWORKS-ENGINEE*V15-APRS-EN-P, R911393303

This documentation describes the use 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.

IndraWorks 15VRS PLC Programming System IndraLogic 2G

DOK-IWORKS-IL2GPRO*V15-APRS-EN-P, R911396137

This documentation describes the PLC programming tool IndraLogic 2G and its use. The documentation includes the basic use, first steps, visualization, menu items and editors.

IndraWorks 15VRS HMI

DOK-MLC***-HMI*****V15-APRS-EN-P, R911399270

This documentation describes the functions, configuration and operation of the user interfaces IndraWorks HMI Engineering and IndraWorks HMI Operation.

MTX 15VRS Shape Cutting Technology

DOK-MTX***-TECHCUT*V15-CORS-EN-P, R911399147

This documentation describes the shape cutting technology used by the machines to cut materials using an NC-controlled tool head.

RS

Corresponding edition

Tab. 1-4:

MTX documentation overview - Commissioning

1.4.4 Operation

Documentation titles with type codes and part numbers

MTX 12VRS Block Pre-Run

DOK-MTX***-BLK*RUN*V12-APRS-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.

MTX 15VRS Programming Manual

DOK-MTX***-NC**PRO*V15-RERS-EN-P, R911393318

This documentation describes the standard programming of the MTX control. Apart from the basics of NC programming, the use of NC functions according to DIN 66025 as well as the NC functions with high-level language syntax and CPL functions are described.

MTX 15VRS Standard NC Operation DOK-MTX***-NC*OP***V15-APRS-EN-P, R911393314 This documentation describes the operation of the standard user interface of the NC control of the MTX. It includes the operation of the interface, the NC program development as well as the tool management.
MTX 15VRS Multitouch DOK-MTX***-MULTI***V15-APRS-EN-P, R911393311 The Multitouch user interface of the NC control MTX is described in this documentation.
MTX 15VRS Standard NC Cycles DOK-MTX***-NC*CYC**V15-PRRS-EN-P, R911394940 This documentation describes the application of the standard cycles of the different technologies for the MTX control.
MTX 15VRS NC Simulation DOK-MTX***-NC*SIM**V15-APRS-EN-P, R911393273 This documentation describes the NC simulation for the MTX control.
MTX 15VRS Measuring Functions DOK-MTX***-MES*FUN*V15-APRS-EN-P, R91194938 This documentation describes the measuring cycles of the MTX control.

RS Corresponding edition
Tab. 1-5: MTX documentation overview - Operating

1.4.5 OEM Engineering

Documentation titles with type codes and part numbers

MTX 13VRS Automation Interface DOK-MTX***-AUT*INT*V13-APRS-EN-P, R911337274 This documentation describes the script-based access to IndraWorks project data via the interface of the Automation Interface. Different objects including code examples are described. The Automation Builder is also described in this manual.
MTX 15VRS OPC Communication DOK-MTX***-OPC*COM*V15-PRRS-EN-P, R911399272 This documentation describes the syntax and the structure of the items for the communication with Bosch Rexroth devices.
IndraWorks OPC UA Communication DOK-IWORKS-OPC*UA*****-APRS-EN-P, R911379309 This documentation describes the OPC UA communication of the MLC and MTX control systems.

RS Corresponding edition
Tab. 1-6: MTX documentation overview - OEM engineering

1.4.6 AddOns

Documentation titles with type codes and part numbers

MTX 11VRS 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.
MTX 15VRS Efficiency Workbench MTX cta, MTX ega DOK-MTX***-EWB*****V15-APRS-EN-P, R911400178 This documentation describes the mode of operation and the area of application of the analysis tool MTX cta and MTX ega.

MTX Remote Condition Monitoring

DOK-MTX***-RCM*****V01-APRS-EN-P, R911334383

This documentation describes the operation of the Remote Condition Monitoring System.

MTX visIREC User Documentation

DOK-MTX***-VISIREC*V01-APRS-EN-P, R911344242

This documentation describes the analysis tool visIREC used to optimize the free form surface milling process. 2D or 4D display of path-related data. 2D or 4D display of coordinate-related data. Analysis of critical areas (path and orientation deviation). Comparing the programmed NC blocks to the interpolated NC blocks.

xx / RS

Corresponding edition

Tab. 1-7:

MTX documentation overview - AddOns

1.5 Information representation

1.5.1 Using safety instructions

Structure of the safety instructions

The safety instructions are structured as follows:

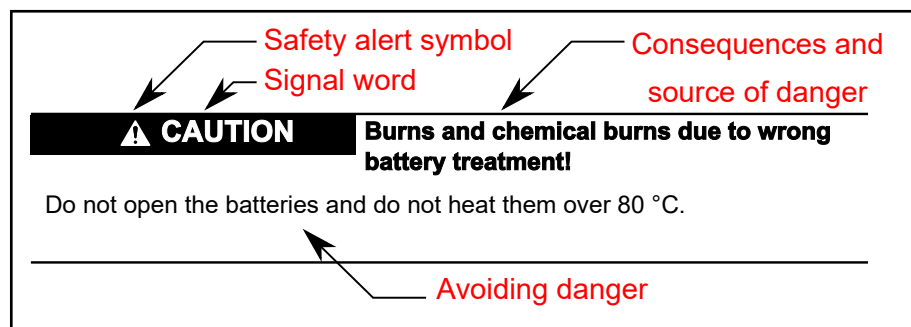


Fig. 1-2: Structure of the safety instructions

Explaining signal words and safety alert symbol

The safety instructions in this documentation contain specific signal words (danger, warning, caution, notice) and, if necessary, a safety alert symbol (according to ANSI Z535.6-2006).

The signal word draws attention to the safety instruction and indicates the risk potential.

The safety alert symbol (triangular safety reflector with exclamation marks), preceding the signal words Danger, Warning, Caution indicates hazards for persons.

⚠ DANGER

In case of non-compliance with this safety instruction, death or serious injury **will** occur.

⚠ WARNING

In case of non-compliance with this safety instruction, death or serious injury **can** occur.

⚠ CAUTION

In case of non-compliance with this safety instruction, minor or moderate injury can occur.

NOTICE

In case of non-compliance with this safety instruction, material damage can occur.

Symbols used

Pointers are displayed as follows:



This is a note.

Tips are displayed as follows:



This is a tip.

Explaining the signal alert symbol on the device



If this symbol is on your device, you have to observe the documentation on the device. The respective documentation informs on the type of hazard as well as the steps required to avoid this hazard.

1.5.2 Names and abbreviations

Term	Explanation
Zeroing	Setting the value to 0
CANopen	Field bus
CNC	Computerized Numerical Control
CPL	Customer Programming Language
DeviceNet	Field bus
Ethernet	Communication interface
GNP	Graphical NC programming
IWE	IndraWorks Engineering
IWO	IndraWorks Operation
OWG	Optical waveguide
NC	Numerical Control
OEM	Original Equipment Manufacturer
PROFIBUS-DP	Field bus
Sercos	Field bus

Tab. 1-8: Terms and abbreviations

1.6 Customer feedback

Customer requests, comments or suggestions for improvement are of great importance to us. Please email your feedback on the documentations to Feedback.Documentation@boschrexroth.de. Directly insert comments in the electronic PDF document and send the PDF file to Bosch Rexroth.

2 Important instructions on use

2.1 Intended use

2.1.1 Introduction

Bosch Rexroth products are developed and manufactured according to the state-of-the-art. The products are tested prior to delivery to ensure operating safety and reliability.

The products may only be used as intended. If they are not used as intended, situations occur that result in damage to property or injury to persons.



Bosch Rexroth shall not assume any warranty, liability or payment of damages in case of damage resulting from a non-intended use of the products; the use shall solely bear all risks from unintended use of the products.

Before using Bosch Rexroth products, the following requirements have to be met to guarantee the intended use of the products:

- Anybody handling Bosch Rexroth products in any way is obliged to read and consent to the relevant safety instructions and the intended use.
- Hardware products may not be altered and have to remain in their original state; i.e. no structural changes are permitted. It is not permitted to decompile software products or alter source codes.
- Do not install damaged or defective products or use them in operation.
- It has to be ensured that the products have been installed as described in the relevant documentation.



Ensure that the data present in the control or entered or read in by the user is correct before applying it to exclude unwanted axis motion. It can be the following invalid or old data:

- Part programs
- ZO tables
- Compensation tables
- Tool tables
- Permanent CPL variables
- Remanent PLC data
- Permanent system data

2.1.2 Areas of use and application

The MTX control is used to

- program contour and machining technology (path feed, spindle speed, tool change) of a workpiece.
- guide 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 - together with the actual CNC - control the machining process as a whole and also monitors this process with regard to technical safety.

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

The CNC system MTX provides the perfect CNC system solution for cutting and forming for the following technologies:

- Rotate
- Milling
- Drilling
- Grinding
- Bending
- Nibbling
- Punching
- Contour cutting
- Handling

2.2 Unintended use

Using the CNC system MTX outside the above-referenced areas of application or under operating conditions other than the conditions described in the documentation and the specified technical data, is defined as "unintended use".

The CNC system MTX must not be used if ...

- it is subjected to operating conditions not corresponding to the specified ambient conditions. Operation under water, under extreme temperature fluctuations or under extreme maximum temperatures is prohibited.
- Furthermore, the CNC system MTX shall not be used for applications not expressly approved of by Bosch Rexroth. Therefore, please read the information given in the general safety instructions!
- The CNC system MTX may not be used in systems or machines connected to the internet via an unsecure network connection. Otherwise, malfunctions or a control failure can result due to unauthorized access.

3 Overview

3.1 Function

Objective	The NC simulation is a graphic simulation for CNC-controlled machine tools with the following functions: • Material removal and material application simulation • Displaying the path of the programmed tool point as polyline (tool path) • Simulation of the machine tool motions including collision detection
	Initially, the simulation has been developed as online simulation and visualizes the running process in each NC operation mode. Only the running NC program can be simulated at primary processing time.
Software options	The simulation can be used in different configuration levels:
	1. Basic license: The basic license facilitates the workpiece simulation as material removal simulation and as tool path graphic. The definition of the raw part geometry is limited to semi-finished products such as cuboids, sheets and cylinders. The tool geometry is calculated based on their technology data (generically generated tool models). An accurate tool representation based CAD models as well as the representation of parts or the entire tool machine is not possible. A collision can only be detected between a workpiece and the currently displayed tool (its non-cutting part). The basic simulation license is part of the MTX interface software from 15V02. 2. Simulation virtual machine: The configuration level "Simulation virtual machine" extends the functionality by the use of CAD models. Using CAD models, workpieces of any form, tools or tool holders, fixtures up to an entire machine (machine simulation) can be represented in detail and then material removal and collision analysis can be executed under realistic conditions. The single-user license "Virtual Machine Simulation" is available under SWS-MTX***-RUN-NNVRS-D0-VMS, R911374010. 3. Virtual Machine Builder The "Virtual Machine Builder" is an additional program for a user-guided and graphically supported creation or to edit machine models installed and used as stand-alone program (without IndraWorks context). The license (dongle license) can be ordered under SWS-MTX***-ENG-01VRS-D0-VMB-DGL, R911394064.
IndraWorks project without simulation	As the simulation is automatically ready in the IndraWorks project, options have to be modified in cases without a simulation being available including the depending functions (refer to chapter "General information" on page 122).
Machining method and kinematics	The NC simulation as tool path graphic can be used for any machining method. It is limited to the following methods and it will be extended to more methods: • Material removal simulation: Boring, milling, punching, beam cutting • Line graphic: Rolling (forming) and signing • Task simulation: 3D print, welding and others • Auxiliary processes, e.g. measurement

Simulation principle

The number of NC axes that can be simulated simultaneously is not strictly limited; it is, however, limited due to performance reasons.

The kinematic type is limited to Cartesian-allocated NC axes. The linear main axes form a right-handed coordinate system. Special kinematics such as parallel and wedge kinematics cannot be simulated.

The machine model consists of the following two components:

- Machine definition

This is a description of the machine structure on how it is composed of moving and stationary elements, where and in which direction the NC axes move (kinematic tree) and where the reference points for tools (tool nodes) and workpieces (stock nodes) are.

- A CAD model set (optional)

The files have to be in STL format (**ST**ereo **L**ithography). They define the physical design of machine parts, tools, clamping devices and workpieces.

Generally, each axis of the machine definition corresponds to a machine tool drive. The simulation is informed periodically in a time interval (multiple of the interpolator cycle) on the current command positions of each drive via a communication interface (OPD UA) and corrects the axes of the model respectively. The reference point of the workpiece or the tool moves along a short polygon edge. The accuracy of the simulated path increases with a decreasing feed velocity and with the number of transmitted coordinates per time interval.



The motions represented in the simulation correspond to reality if the stored machine model exactly represents the real kinematics. At corner transitions, the approximation of the simulated path by polygon edges results in the generation of chamfers/bevels. These chamfers are a result of this principle and do not constitute a deficiency.

Additionally, the following data can be used for the motion:

- SD data values
- Virtual axes positions
- Punching signal for punching stroke



- The SD data change is not reported synchronously with the coordinates from the interpolator. System data can thus only be used for uncritical and rarely changing coordinates (e.g. the position of clamping devices).
- Special kinematics can for example be mapped using virtual axes.
- The simulated punching head position is calculated using the sheet thickness and the punching signal and does not correspond to the actual position in this case.

States of the NC block processing

In automatic mode, the following NC kernel states affect the graphic:

- Starting an NC program results in the deletion of the tool path of the respective channel.
- Depending on the option, starting the NC program triggers a renewal of the workpieces machined by the respective channel.

- After reaching the regular program end and depending on the options, a workpiece preview (see ["Representation of finished part" on page 16](#)) is created with the current workpiece state.
- When selecting an NC program, a preview can be shown depending on the option selected.

Feature

The online simulation is provided as .NET control which is displayed in the NC screen of the "Machine" operating area by default. This control may also be integrated in a third party application in IndraWorks. The following operating description applies only analogously (see [\[External link could not be resolved.\]](#)).

3.2 Starting and exiting the simulation

Start Immediately after starting IndraWorks Operation, the NC Simulation can detect each machine motion during the entire session - irrespective of its visibility.

The simulation can temporarily be enabled and disabled using a key. See [chapter 3.3 "Layout" on page 15](#).

Displays In each operation mode, the simulation can be displayed in the additional operating area of IndraWorks Operation. Enable the "NC Simulation" tab



In the operating area "Machine", the NC Cockpit provides another option to display the NC Simulation. If this visualization method is used, the complete application window can be used to display the NC Simulation.

Therefore, it is required to call the function in the Home Control tab in the additional operating area via the tile <NC Cockpit>.

If the NC cockpit display is active, the content of the "NC Simulation" tab is toggled to the "Position and block progress display" shown on the tab using



the icon

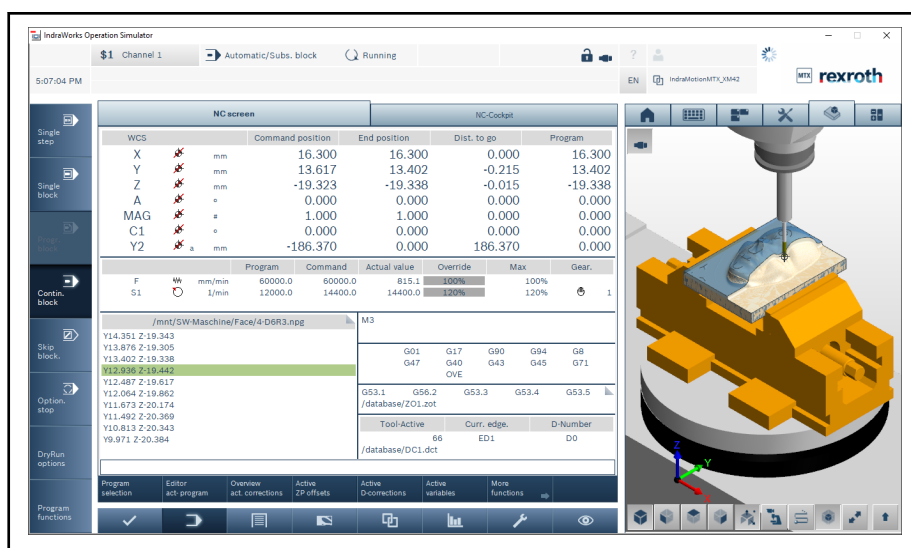


Fig. 3-1: NC simulation in the additional operating area

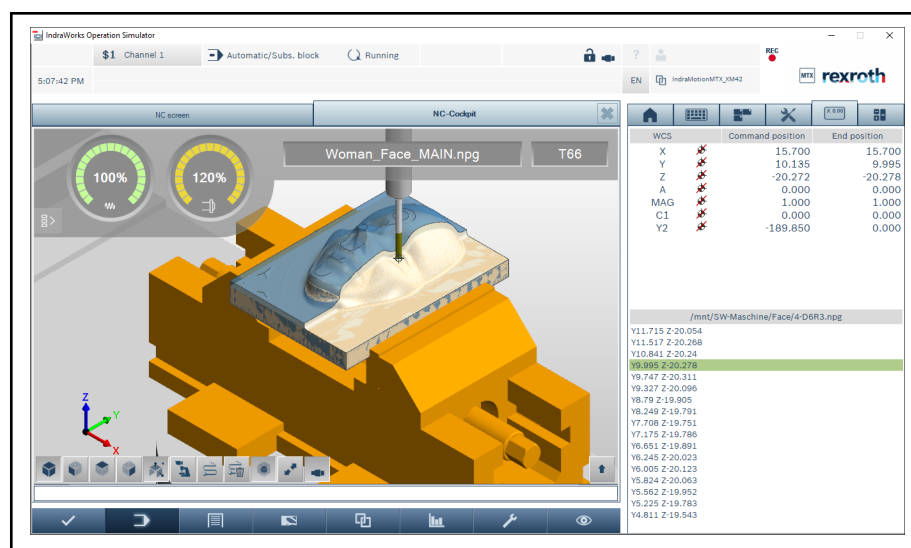
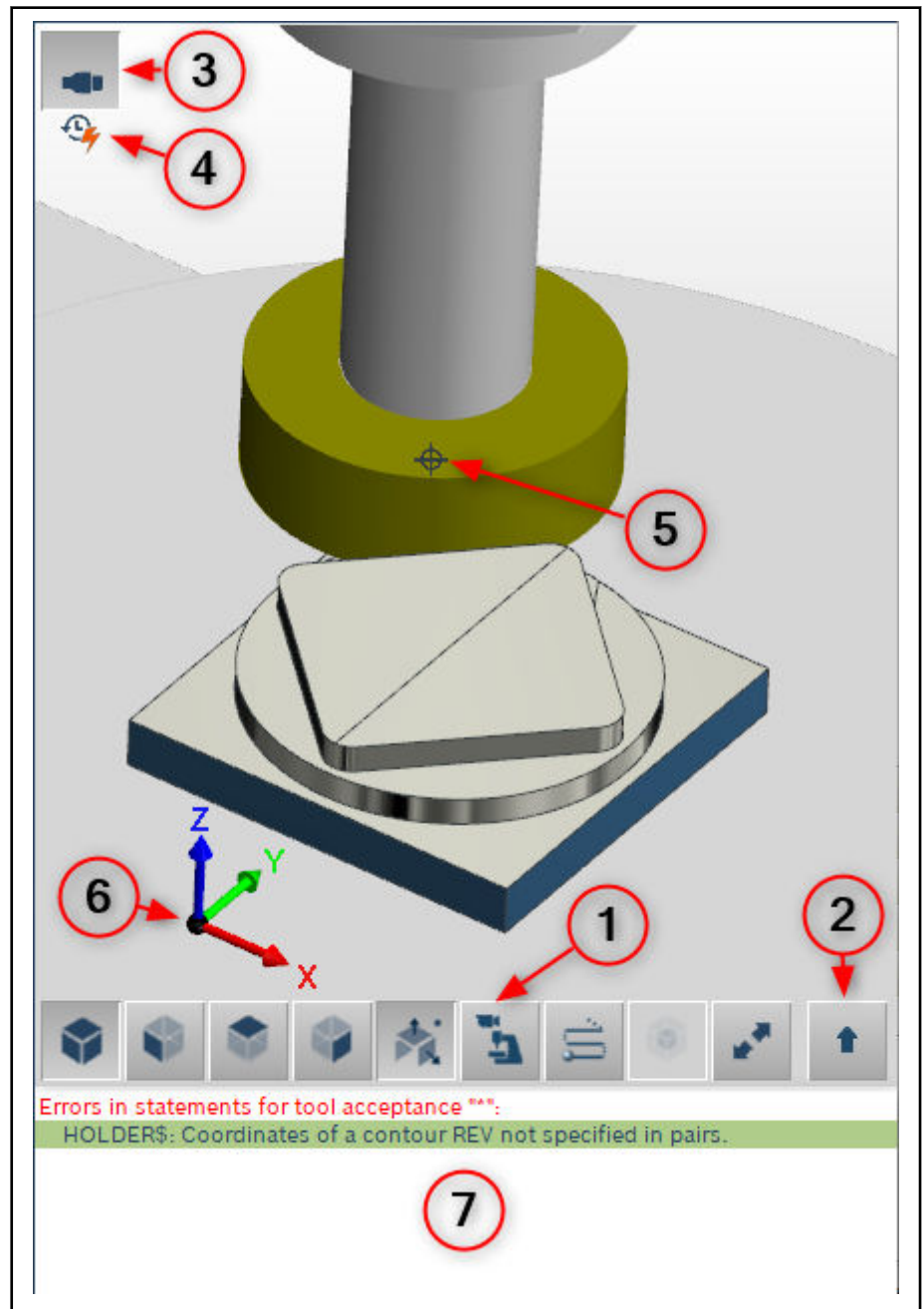


Fig. 3-2: NC simulation in the "Cockpit" display

Exit When exiting IndraWorks Operation, the current settings and, depending on the selected options, the machining state of the workpiece are saved to retain the settings for the next session. The last image section and the camera position are part of it.

3.3 Layout

Screenshot



- 1 Shortcuts
- 2 Showing and hiding additional operating keys
- 3 Operating key and status display to connect to the NC, to disable or start after a collision
- 4 Displaying that at least one collision was registered in the running programs
- 5 Generating point of tool path graphic at tool
- 6 Coordinate system
- 7 Output window for errors and warnings

Fig. 3-3: NC simulation layout

Shortcuts Use the shortcuts to immediately make frequently used settings without showing the additional keyboard. The status "selected yes/no" and the

Connection to the NC and disabling of the simulation

changing symbol display the current status. Tool tips provide additional information on the status.

The symbol ( - OK,  - instable,  - faulty,  - stopped after collision) indicates whether a stable connection is available to transmit axis coordinates and other simulation input data. If one of the symbols is visible, it is tried to reconnect when pressing once, to reestablish the connection or to start the simulation after the collision.

To reduce the system load, the simulation can be disabled permanently. The dialog "The simulation is active. Stop now?" is displayed if the key is pressed multiple times (at least five times) successively. After actuating <Yes>, the

simulation is disabled. The  symbol indicates the disabled state.

For reactivation, press the confirmation prompt and confirm the prompt.

Output window for errors and warnings

The output window is only displayed if errors and warnings are reported. The background is highlighted in red as soon as at least one error is listed.

Representation of finished part

A command geometry of the workpiece at the end of the machining cycle is referred to as finished part. There are two options to load the geometry to the simulation:

- By CAD file import
- By loading the machining status of the workpiece and the tool path present when the regular NC program end was reached during the last program run

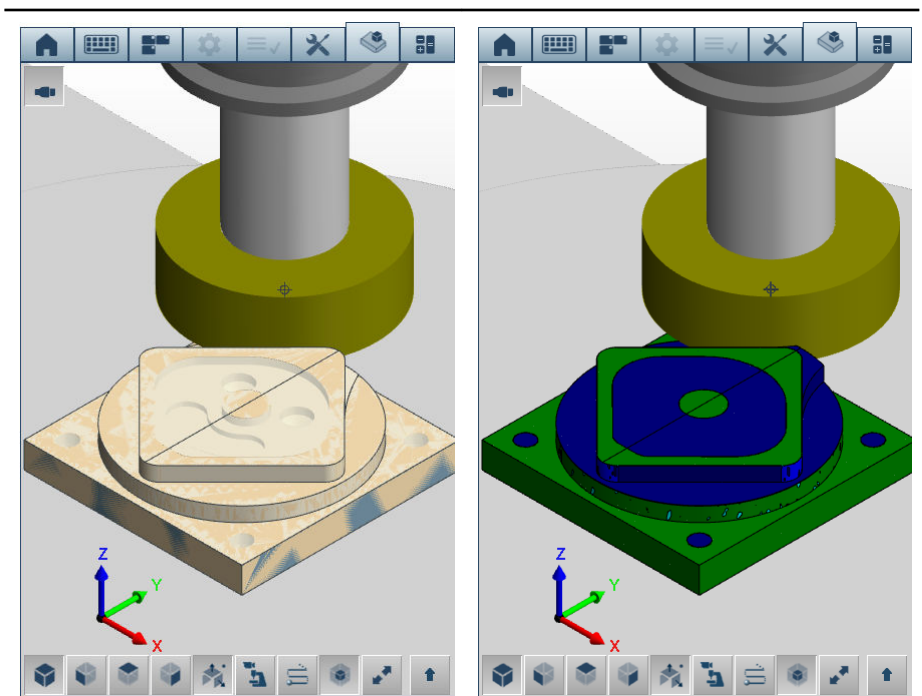
This geometry is saved in the main program folder under the program name with the extension ".p" and it is then connected to the main program.

The finished part geometry generated by material removal executed by the simulation and the corresponding tool path is referred to as **Workpiece preview**.

The workpiece preview is automatically generated/displayed and has to be enabled in the options of the simulation. The preview is displayed directly upon the program selection if no imported finished part has been loaded and the file is up-to-date and/or not read-only.

There are two options to map the finished part in the workpiece representation:

- As (usually internal) representation in the transparent workpiece
- As target-actual comparison with the workpiece. Different colors are allocated to the workpiece surfaces. The colors are schematically allocated to the dimension deviation in certain intervals. This representation mode is also referred to as "Deviation mode"



Blue shades Machining allowance available
Green shades Finished dimension reached
Red shades Finished dimension not reached (not displayed)

Tab. 3-1: Finished part representation

4 Operation

4.1 Operating keys and corresponding dialogs

4.1.1 Shortcuts of the simulation



The shortcuts listed in the following are partially placed option-dependently on the simulation control.

Key/function		Explanation	
3D view		"3D view" sets an inclined camera position with preset angles	
Front view		"Front view" shows the front view	
Top view		"Top view" shows the top view	
Right view		"Right view" shows the right side	
Hide		"Hide" to toggle between machine and workpiece simulation in several steps by hiding/showing machine parts and tool holders	Refer to chapter "Button block Representation" on page 21
Camera position (Point of view of camera)		Machine: This stationary reference is the position of the observer/operator	Refer to chapter "Button block Details" on page 23
		Tool: The camera position moves with the tool, but does not rotate with the rotary axes swiveling the tool. This setting is recommended for large workpieces with small, distributed machining steps	
		Workpiece: The camera position moves and rotates together with the workpiece. This setting is recommended for dynamic workpieces (XY-sliding table kinematics, rotary axes at machine table)	
Tool path		"Tool path" activates/deactivates the tool path representation	Refer to chapter "Button block Representation" on page 21
Finished part		"Finished part" includes the finished part geometry in their different variants	Refer to chapter "Button block Representation" on page 21
Improve workpiece ¹		"Improve workpiece" increases the representation quality	Refer to chapter "Button block Details" on page 23

Key/function		Explanation	
Full screen		Overall view is optimally adjusted to screen size	Refer to chapter 4.1.2 "Extended operating keyboard of the simulation" on page 20
Connection status		Connection OK	The operator can start and stop the connection using this key (refer to chapter 3.3 "Layout" on page 15). The key starts the simulation after a collision.
		Connection interfered	
		Insecure connection	
		Connection stopped by operator	
		Simulation stopped after collision	
Show extended operating keyboard		The extended operating keyboard is hidden.	
		The shortcuts become the button block "Display status".	

1

Only active if "HighQualityZooming" was enabled in the options

Tab. 4-1: Shortcuts

4.1.2 Extended operating keyboard of the simulation

Button block "Display status"

Key/function		Explanation	
Hide		"Hide" to toggle between machine and workpiece simulation in several steps by hiding/showing machine parts and tool holders	The key corresponds to the "Hide" key in the second line of keys See chapter "Button block Representation" on page 21
Camera position		Camera position "Machine"	The key corresponds to the "Camera position" key in the second line of keys See chapter "Button block Details" on page 23
		Camera position "Tool"	
		Camera position "Workpiece"	
Finished part		"Finished part" includes the finished part geometry in their different variants	Refer to chapter "Button block Representation" on page 21
Axes/scales		Shows and hides a coordinate tripod or scales	Refer to chapter "Button block Display status" on page 20

Key/function		Explanation
Hide extended operating keyboard		The extended operating keyboard is hidden.
		Only the shortcuts are visible.

Tab. 4-2: Button block "Display status"

Button block "Representation"

Key/function		Explanation
Hide		"Hide" to toggle between machine and workpiece simulation in several steps by hiding/showing machine parts and tool holders.
Tool path		"Tool path" activates/deactivates the tool path representation
Remove tool path		"Remove tool path" removes the current tool path graphic
Edges		"Edges" draws the object edges with dark lines or suppresses retracing
Axes/scales		"Axes/scales" alternately or simultaneously show the coordinate tripod and the scales
Finished part		"Finished part" includes the finished part geometry in their different variants
Central/parallel projection		Toggling the view between the central projection (perspective view) and parallel projection

Tab. 4-3: Button block "Representation"

Hide Press the "Hide" key to toggle between machine and workpiece simulation in several stages by hiding/showing machine parts and tool holders.

The following stages are available depending on the machine model and settings:

- Hide OFF, hood closed
- Hood open. If a transparency is set for the machine housing, the unmovable machine parts are shown as transparent (see "[Transparency of the machine housing](#)" on page 143).
- Hide all except for active elements. With this setting, all parts not moved by NC axes and not being part of the tools, workpieces or clamping devices are hidden.
- Hide all except for the workpiece, tool and clamping device. Tool holders are not displayed for a better monitoring of the tool operation.

Tool path The key "Tool path" activates/deactivates the tool path representation in the following stages:

- Hide tool path,
- Always show tool path (on top)
- Show tool path if it is not covered by the foreground.

The function is only available if the tool path representation has been enabled in the options.

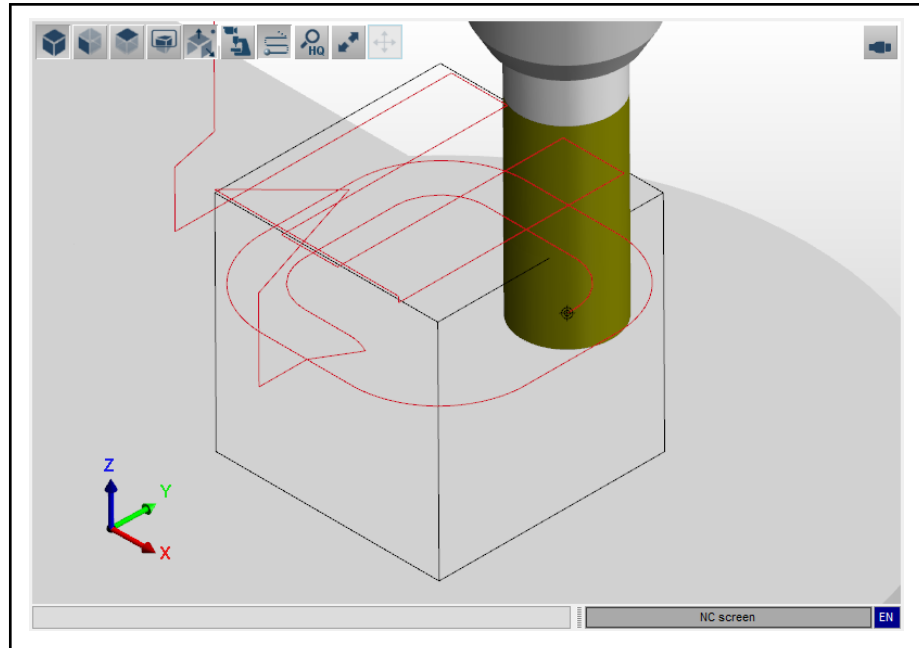


Fig. 4-1: Simulation with tool path "On Top"

Edges The key "Edges" draws the object edges with black lines or suppresses re-tracing. In the options, it is specified whether the edges are to be drawn at all objects or only at the raw part. The edge representation cannot be hidden in drawings or if the transparency limit is exceeded.

Finished part Press the "Finished part" key and the finished part geometry is included in the representation with its different variants. If the finished part was created as workpiece preview, the tool path preview can also become visible with <F3> "Tool path". The key is only available if the finished part geometry is available.

There are three stages:

- No finished part visible
- Finished part visible (in transparent workpiece)
- Target/actual comparison of the workpiece to the finished part. For different intervals of the dimensional deviation, different colors are used allowing conclusions on waste, finished dimension reached or remaining machining allowance (see also [chapter 3.3 "Layout" on page 15](#)).

The target-actual comparison requires significant CPU resources and can thus be enabled separately by selecting a specific setting in the options.

All finished part geometries that can be shown are applicable in this context:

- Via a mesh (STL file) loaded using a dialog or cycle with the import target "Finished part"
- Via a two-dimensional CAD drawing (DXF file) loaded using a dialog or cycle with the import target "Finished part"
- A mesh (STL file) automatically loaded upon program selection. Its file name is equal to the name of the main program (extension not considered)

- A part program preview generated during program run. The tool path can be displayed additionally
- A two-dimensional CAD drawing automatically loaded upon program selection. Its file name is equal to the name of the main program (extension not considered)
- The enveloping body from the raw part dimensions around the future workpiece in the additive machining

Button block "View"

Key/function		Explanation
3D view (1)		"3D view (1)" sets an inclined camera position with preset angles
3D view (2)		"3D view (2)" sets an another camera position with preset angles
Front view		"Front view" shows the front view
Rear view		"Rear view" shows the rear view
Right view		"Right view" shows the right side
Left view		"Left view" shows the left side
Top view		"Top view" shows the top view.
Bottom view		"Bottom view" shows the bottom view

Tab. 4-4: Button block "View"

Button block "Details"

Key/function		Explanation
Improve workpiece		"Improve workpiece" allows to improve the representation quality and to return to the standard quality
Full screen		"Full screen" scales the graphic to fill the complete screen
Camera position		"Camera position" determines to which object the camera position is applied to

Key/function		Explanation
Collisions		"Collisions" opens a dialog used to affect the simulation response in case of a collision, to monitor registered collision events and to open the NC program editor if required
Collision events		"Collision events" opens the list of all detected collision events and allows to access already saved collision reports.

Tab. 4-5: Button block "Details"

Improving the workpiece

The key "Improve workpiece" allows to improve the representation quality and to return to the standard quality.

There are two methods to improve the workpiece:

- **Dynamic zoom**
After zooming, the previous machining is internally repeated and displayed in a higher resolution for the zoomed out workpiece of the image section. If the improve function is active, the improvement is executed after each zooming. After changing the image section, the improvement is lost.
- **Auto refine**
If permitted by the simulation load, the representation is improved continuously. The improved quality is maintained and can be seen at the complete workpiece.



Improving with dynamic zoom takes time. Meanwhile, the simulation should not be operated and no axis motions should take place. Otherwise, the standard quality is displayed again. The pending improvement is indicated by a countdown display. During this period, the image section can still be changed.

The "Improve workpiece" function or the method to be used has to be enabled in the simulation options.

Full screen

Press the "Full screen" key and the currently shown scene is fitted to the complete area.

If the option "Automatic zoom" is enabled, this key is provided with two additional states:

- **Off:**
The current screen is scaled in full size. There is no subsequent automatic change in the image section.
- **On:**
The current screen is scaled in full size. If, during the following machining, the bounding box increased by all representing elements, the graphic is scaled again for a new full screen. It is also rescaled if another image section was selected in the meantime.

Camera position

The function "Camera position" determines to which object the camera position is applied to.

Depending on the machine model (kinematic tree and availability of CAD models), there are up to three reference options:

- Machine: This stationary reference is the position of the observer/opera-

tor. Shortcut symbol:



- Tool: The camera position moves with the tool, but does not rotate with the rotary axes swiveling the tool. This setting is recommended for large workpieces with small, distributed machining steps. Shortcut symbol:



- Workpiece: The camera position moves and rotates together with the workpiece. This setting is recommended for dynamic workpieces (XY-sliding table kinematics, rotary axes at machine table). Shortcut symbol:



Collisions

The "Collisions" function intended to control the simulation response during a collision and to analyze current and preceding collisions is only available if the collision detection is enabled in the options. An additional dialog is displayed below the simulation.

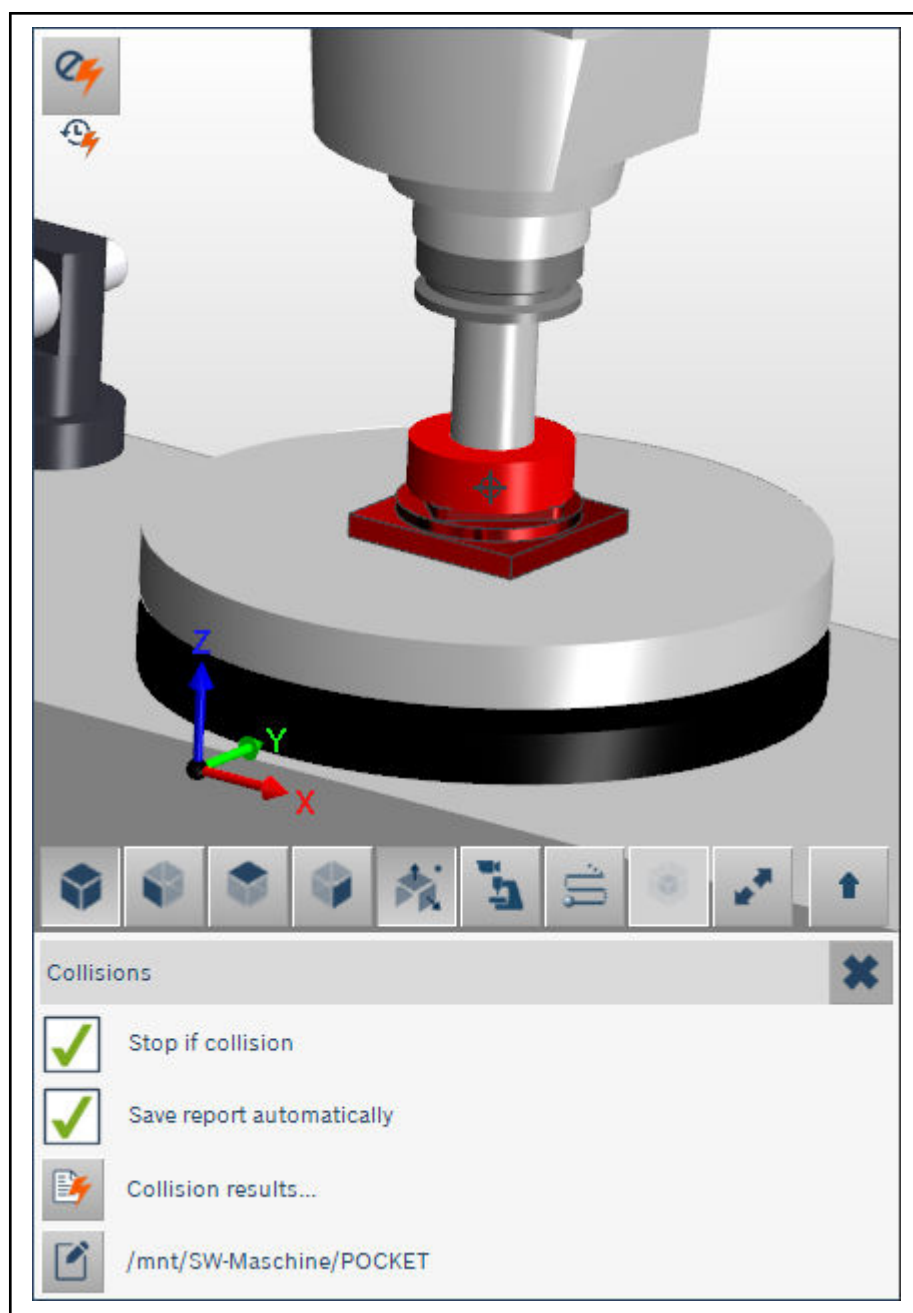


Fig. 4-2: "Collisions" dialog after a stop in an editable program

There are the following menu items in the "Collisions" dialog:

- Stop if collision: If "Stop if collision" is enabled, the simulation goes off-

line and stops (indicated with the symbol  on the "Connection status" key) while the program continues. In the stopped state, the cause or the NC block can be identified immediately. Press the "Connection status" button and the simulation is online again (indicated with the sym-

bol ). The simulation clears the back log as fast as possible with regard to the real program progress and stops at the next detected collision.



The simulation backlog after stopping continuously consumes RAM during program progress. To avoid a memory overflow, disable "Stop if collision" if the simulation runs for a longer period without being monitored.

If "Stop if collision" is deselected, the simulation continuous. Collision events can be analyzed subsequently.

- **Save report automatically:** If "Save report automatically" is enabled, the number of collisions associated with the main program and the respective channel is saved in a file called collision report after reaching the end of the program. The name is composed of the main program name (shortened, if required), the channel and the saving time.

e.g. "MyProg_02_20190711_133815.crp" with the main program MyProg.npg, channel \$2 and point in time 11.07.2019 13:38:15.

The extension is ".crp". The file storage folder can be changed in the options of the simulation.



- Collision events  :

Opens a dialog that lists all registered collision events connected to the running programs. The list is overwritten with "Current collision events".



Press to restore or refresh it.

Upon program start, the collision events of a channel are deleted from the current collision events.

Current collision events					
Point in time	\$	Program	NC block/addr.		
7/16/2019 6:55:33 AM	1	STC_G266	@1C02		
7/16/2019 6:55:35 AM	1	STC_G266	@207B		
7/16/2019 6:55:35 AM	1	STC_G266	@207B		
7/16/2019 6:55:35 AM	1	GesichtCo.npg	G0 Z150		
7/16/2019 6:55:35 AM	1	GesichtCo.npg	N60		
7/16/2019 6:55:35 AM	1	STC_PIN	N1020		

Collision partner: tool - ImportedFixt000, tool - ImportedFixt001					
Path: /mnt/SW-Maschine/GesichtCo.npg					
NC block: G0 Z150					

Fig. 4-3: Dialog "Current collision events"

To alternatively open and display saved collision reports, press



key. The title then changes to "Collision report [xyz.crp]". Press  or double-click on a line in the list to start the NC editor with the currently selected program and to navigate to the respective NC block.

- Editor  : Opens the NC editor and navigates to the NC block where the simulation just stopped due to a detected collision. The key is only available if the simulation is stopped and if the program is neither read-only nor encoded.

Button block "Tools"

Key/function		Explanation
Reset raw part...		"Reset raw part..." opens a dialog to renew the workpiece and to delete graphics
Cuboid, sheet...		"Cuboid/sheet..." opens a dialog to define a prismatic workpiece
Cylinders...		"Cylinder..." is used to define a cylindrical raw part, optionally with inner bore
Import...		"Import..." to import a CAD model for different targets
Export...		"Export..." writes the current workpiece geometry or workpiece-related geometries to a file
Online/offline		"Online/offline" allows to switch the simulation using a fast emulation to create a preview. The function is being planned.
Options...		"Options" facilitates the display and change of the current options/settings of the simulation in a text editor

Tab. 4-6: Button block "Details"

Reset raw part The function "Reset raw part..." opens a dialog to renew the workpiece and to delete graphics.

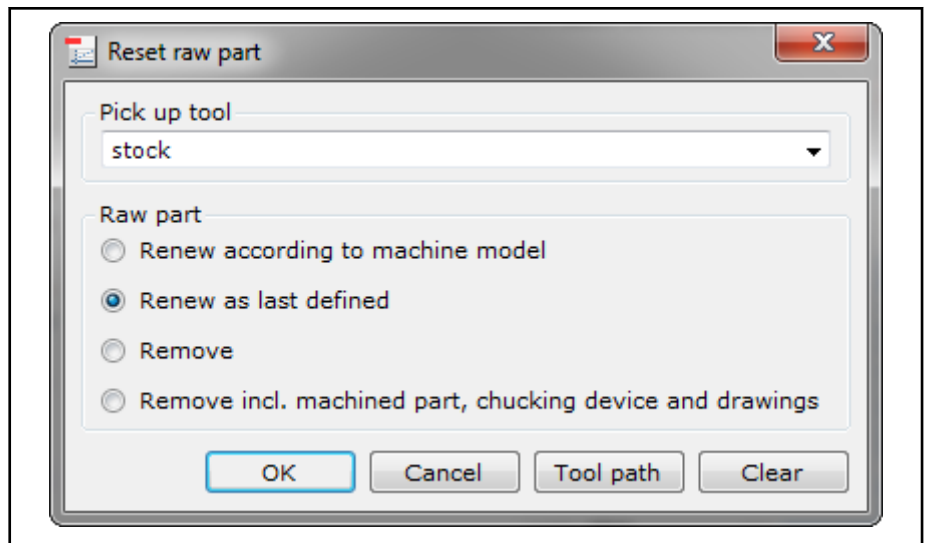


Fig. 4-4: "Reset raw part" dialog

The dialog has the following elements:

- Selection list "Workpiece holder": The selection determines to which workpiece holder (stock node) the reset is applied to. The box is only displayed if the machine model has more than one workpiece holder.
- "Raw part" selection specifies the action to be executed:
 - Renew according to machine model:
If the machine model contains a CAD model for the workpiece for the selected workpiece holder, the workpiece is subsequently displayed with the corresponding clamping device.
 - Renew as last defined:
If you defined a workpiece, it is displayed in its raw state.
 - Remove:
Removes the workpiece.
 - Remove including finished part, clamping device and drawings:
Empties the workpiece holder completely.

A finished part is a geometry corresponding to the machining end. It can either be a command geometry as mesh or a two-dimensional drawing imported as finished part.

Drawings are representations of object edges from meshes without considering the collision detection. Alternatively, two-dimensional drawings can be imported from DXF files.

<OK> to execute the selected action and to close the dialog.

- <Cancel> closes the dialog without action.
- <Tool path> to delete the tool path assigned to the selected workpiece holder.
- <Clear> deletes all saved raw parts and partially machined parts as well as all clamping devices, drawings and workpiece previews. Use this function only in case of repeated problems with the workpiece representation that cannot be solved differently. The deleted data is lost and cannot be recovered. Thus, a safety prompt is displayed before data is deleted.

Cuboid/sheet The "Cuboid/sheet..." function opens a dialog to define a prismatic workpiece as semi-finished parts, a cuboid during milling, a polygon during turning and a sheet during punching.

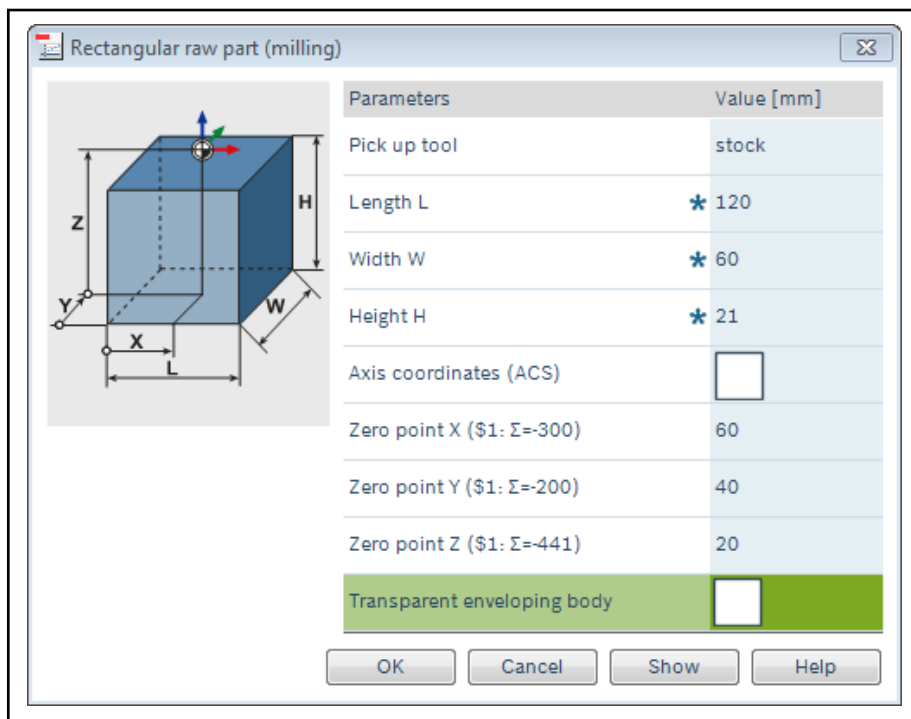


Fig. 4-5: "Rectangular raw part (milling)" dialog

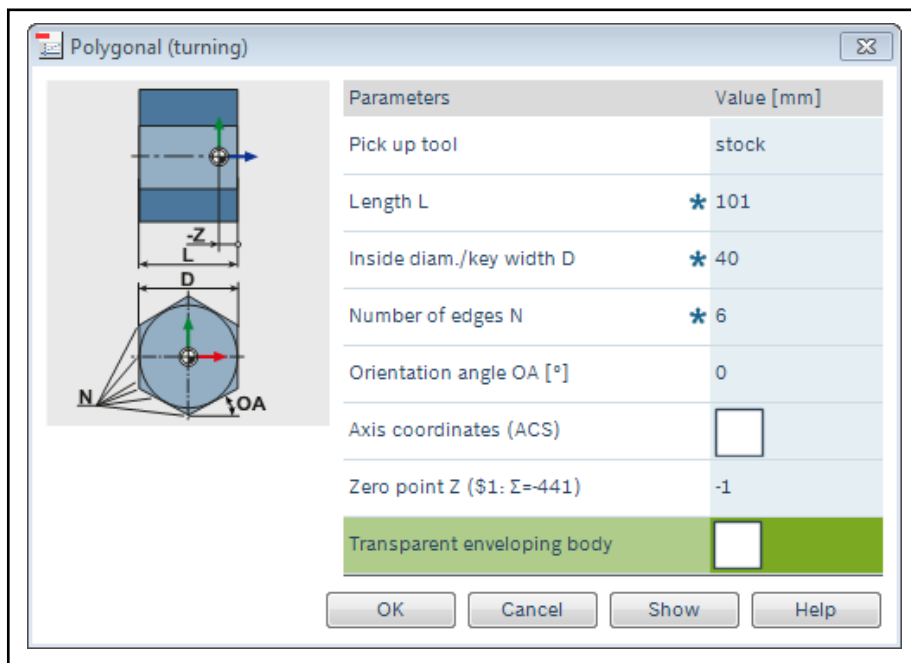


Fig. 4-6: "Polygon (turning)" dialog

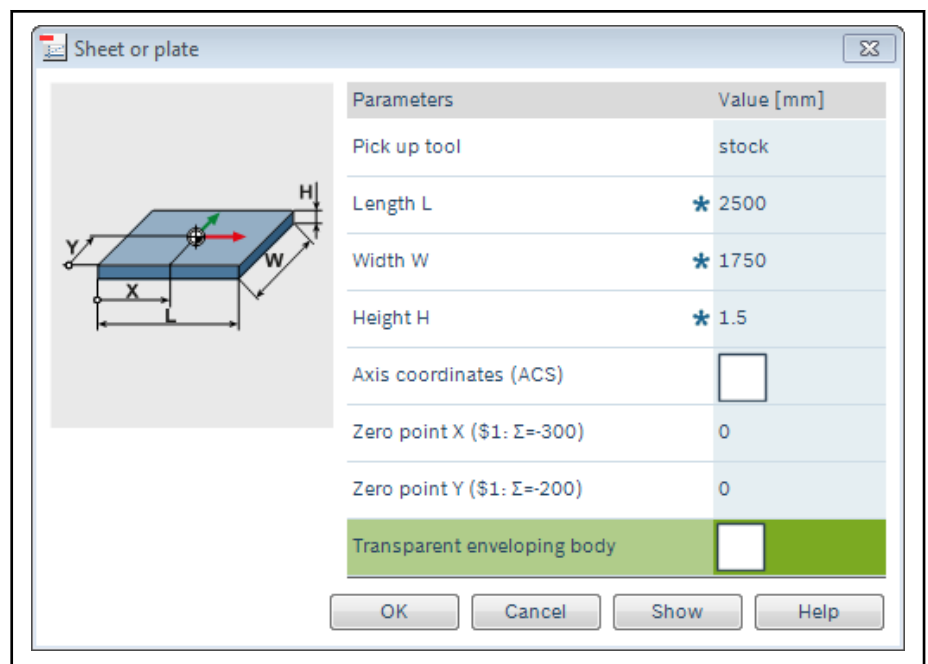


Fig. 4-7: "Sheet or plate" dialog

The first parameter, "Workpiece holder" is used to specify where to create the new raw part. For this purpose, the name of the workpiece node in the machine model is used.

The respective dimensions follow.

The "Axis coordinates (ACS)" checkbox specifies whether the sum of zero point offsets and the placements in the channel is to be added internally to the following parameters, the zero points. For information purposes, the resulting sum of linear offsets is displayed in the parameter names after the sum character. If there are - in addition to linear main axes - further rotary or linear axes (rotary table, telescope axis) implemented in the kinematic chain of the machine model, their actual position is considered when positioning the workpiece in the working area. Position and alignment of the workpiece are oriented at the current workpiece coordinate system. If ACS is selected, the mentioned transformations are not executed and the workpiece is directly defined in the coordinate system of the workpiece holder.

If "Transparent enveloping body" is selected, the raw part is displayed as edge model without collision check and without possible material removal. The tool path representation is activated automatically. Use this setting for more complex workpieces you do not have a CAD model for.

Press the <Display> key and the new raw part is flashing indicating to check the inputs. The dialog is converted to a non-modal dialog to allow the operation of the simulation.

The <Help> key opens the help chapter on all functions of the "Tools" menu level. The input dialog for the raw part is located there.



If - in the options - , several workpieces can be defined per holder, the definition results in the replacement of the present workpiece or, otherwise, in the addition to the existing workpieces. Remove the workpiece using "Reset raw part" function to replace the workpiece.

Cylinder The function "Cylinder..." is used to define a cylindrical raw part, optionally with an inner bore. In the main machining process "Turning", a dialog is displayed to define a cylindrical bar section.

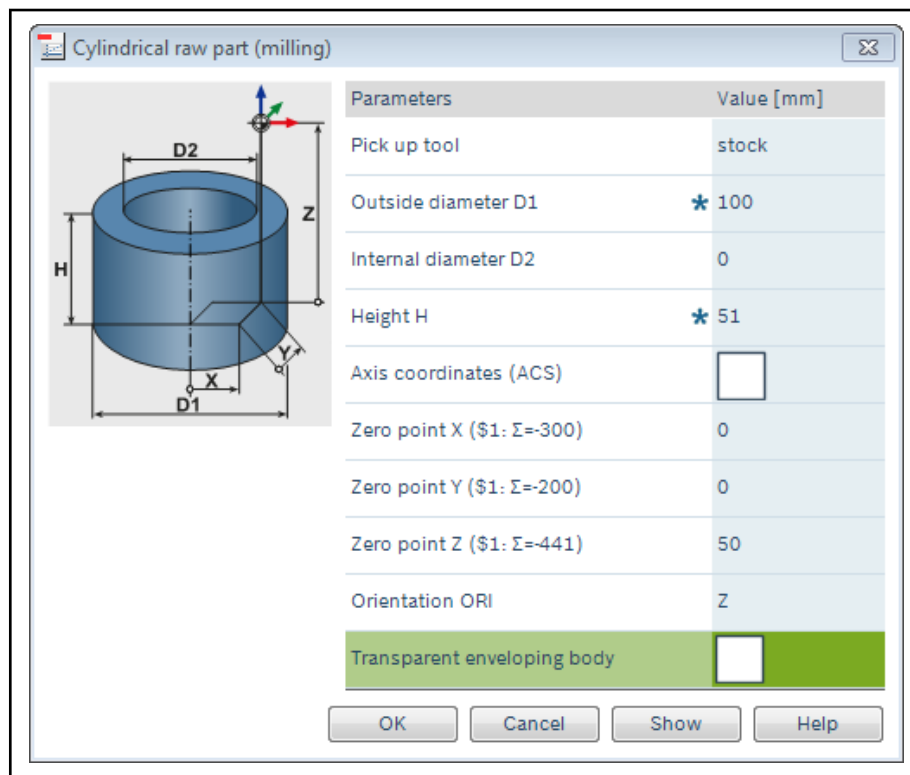


Fig. 4-8: "Cylindrical raw part (milling)" dialog

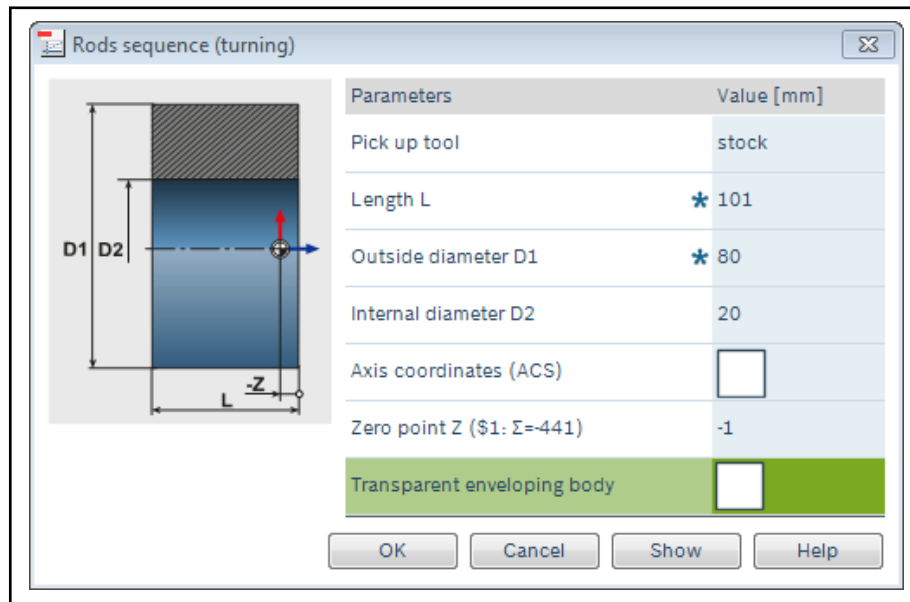


Fig. 4-9: "Bar section (turning)" dialog

The first parameter, "Workpiece holder" is used to specify where to create the new raw part. For this purpose, the name of the workpiece node in the machine model is used.

The respective dimensions follow.

The "Axis coordinates (ACS)" checkbox specifies whether the sum of zero point offsets and the placements in the channel is to be added internally to the following parameters, the zero points. For information purposes, the resulting sum of linear offsets is displayed in the parameter names after the sum character. If there are - in addition to linear main axes - further rotary or linear axes implemented in the kinematic chain of the machine model, their actual position is considered when positioning the workpiece in the working area. Position and alignment of the workpiece orient at the current workpiece coordinate system. If ACS is selected, the mentioned transformations are not executed and the workpiece is directly defined in the coordinate system of the workpiece holder.

If "Transparent enveloping body" is selected, the raw part is displayed as edge model without collision check and without possible material removal. The tool path representation is activated automatically. Use this setting for more complex workpieces you do not have a CAD model for.

Press the <Display> key and the new raw part is flashing indicating to check the inputs. The dialog is converted to a non-modal dialog to allow the operation of the simulation.

The <Help> key opens the help chapter on all functions of the "Tools" menu level. The input dialog for the raw part is located there.



If - in the options - , several workpieces can be defined per holder, the definition results in the replacement of the present workpiece or, otherwise, in the addition to the existing workpieces. Remove the workpiece using "Reset raw part" function to replace the workpiece.

Import

Use the "Import..." function to import a CAD model for different targets. The underlying coordinate system corresponds to the one at the workpiece holder and can deviate from the workpiece coordinate system.

Initially, a file selection dialog is displayed. Select the source file in this dialog. It can be saved in the MTX file system and in the file system of the local PC.

To toggle in the file selection from opening from the local PC to the MTX file system:

1. In the selection list for the file type (bottom right), select the last entry "Change to MTX file system (*.*)".
2. Click on any file and press <Open>.

The file selection dialog is closed and the file selection dialog of MTX is displayed.

To toggle in the file selection from opening from the MTX file system to the local PC:

1. Select the last entry "Change to local PC (*.*)" from the "File type" selection list.
2. Click on <Cancel> or on any file and press <Open>.

The file selection dialog is closed and the file selection dialog for the local PC file system is displayed.

The following file types are provided in the file selection dialog. The file type selected restricts the possible import targets:

- STL file: All import targets
- DXF file: Import target is either drawing or workpiece preview

- Job setup file (*.mwjs, *.zip): Raw parts, clamping media and tools are imported for a complete workpiece program. Select the file. No further entries are required. It is still a prototype function.

Click <OK> and the model to be imported is flashing and displayed in an initial position at the workpiece holder. Additionally, a dialog is shown in horizontal or vertical alignment used to control the import:

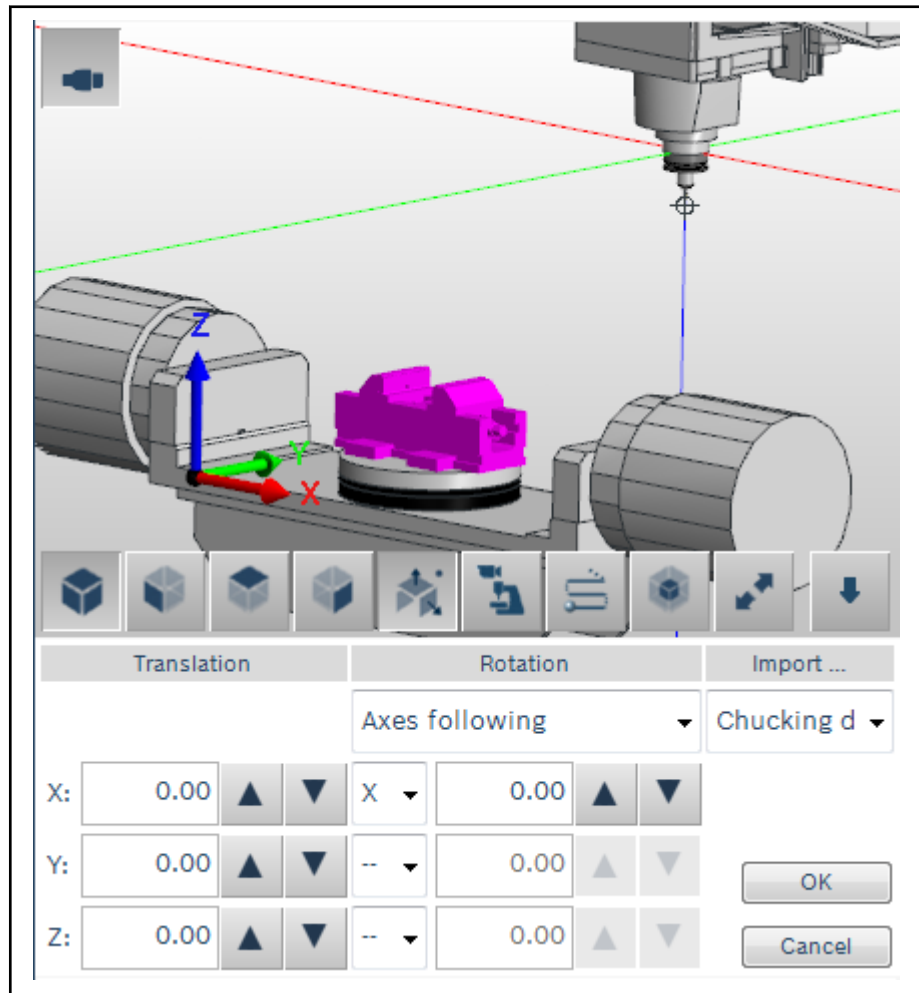


Fig. 4-10: Import (dialog in horizontal arrangement)

- The import target is specified below the "Import..." label. Available for selection:
 - Raw part
 - Finished part
 - Clamping device
 - Drawing
 - Export

Clamping devices and drawings are always added to the existing devices and drawings. Raw parts are added depending on the options. The finished part geometry is replaced by the existing geometry, if available.

During the export, the imported model is written as file. A file selection dialog is then displayed. This function can be used to adjust an existing CAD model by rotating or moving.

If the machine model has more than one workpiece holder, there is another selection list below the aforementioned one to select the workpiece holder to which the import is to be made.

- "Translation": Move the imported model in X, Y or Z.
- "Rotation": Rotate the model around the Cartesian axes in up to three individual steps. Each subsequent rotation is executed around stationary axes or around the axes resulting from a previous rotation ("Axes following").

The rotary axis is specified using the selection lists before the angles of rotation.

- <OK> executes the import and closes the import dialog.
- <Cancel> exits the import without action.

Export

The "Export..." function writes the current workpiece geometry to a file, except enveloping bodies. If the clamping device or the drawing consists of several parts, the clamping device/the drawing can also be exported to a file.

If the machine model has more workpiece holders that are not empty or different geometries resulting from the import target, a dialog is displayed to select the workpiece holder and the geometry used for the export.

Select the target file for the export from a file selection list.

To toggle in the file selection from "Save as" from the local PC to the MTX file system:

1. In the selection list for the file type (bottom right), select the last entry "Change to MTX file system (*.*)".
2. Click on any file and press <Save>.
3. Confirm the subsequent prompt with <Yes> to replace the file (decision not performed).

Instead, the file selection dialog is closed and the MTX file selection dialog is displayed.

To toggle in the file selection from "Save as" from the MTX file system to the local PC:

1. Select the last entry "Change to local PC (*.*)" from the "File type" selection list.
2. Click on <Cancel> or on any file and press <Save>.
3. If <Save> has been pressed in the previous step, confirm the subsequent prompt with <Yes> to replace the file (decision is not performed).

The file selection dialog is closed and the file selection dialog for the local PC file system is displayed.



Two-dimensional drawings cannot be exported into DXF files.

Options

The "Options" functions allows to display and change the current options/settings of the simulation in a text editor.

The query is displayed "Do you want to open and change the options in the editor? <Yes>/<No>". If the query is confirmed, a text editor is started (Notepad++, if installed or Windows Wordpad).



The XML file opened in the editor is a temporary file. If the XML syntax is violated by editing, the simulation settings cannot be damaged. Deleted settings or incorrectly formatted settings are reset to default values. Format errors or value range exceedances may cause loss of data. After such case, download the visualization data again (IndraWorks Engineering, context menu of the node for the visualization device).

For more information on the data structure and on the configuration of the simulation, refer to [chapter 8 "Configuration" on page 105](#).

After saving the changes and closing the editor, confirm or reject the box "Apply options? Press <Yes>/<No>". Press <Yes> to apply the options.

To also apply the changes to the visualization project, execute the "Get changes from operating station" function in IndraWorks Engineering on the node of the visualization device.

4.2 Shortcuts

Simulation shortcuts



The shortcuts of the simulation are only applied if the focus is on the simulation control.

Shortcuts	Simulation function
<C>	Establishing a connection to the NC
<N>	Define new raw part as semi-finished product: • Milling, generative processes: rectangular raw part • Punching: Sheet • Rotate: Bar section
+	Zoom in
-	Zoom out
0	Full screen
<Del>	Reset raw part
<Esc>	Cancels the import and other dialogs, empties the output window
<Cursor key>	Moves the image section
<Shift>+<Cursor key>	Rotates the camera position
<Ctrl>+<C>	Copies the transformation matrix (4x4) from the syntax of the machine definition to the clipboard. Applicable in the import dialog when focusing on an input element.
<Shift>+<F10>	Displays the context menu to record the simulation. Only available in debug mode.
<Left mouse key>	Rotates the camera position when the mouse is moved with the left mouse button pressed

Shortcuts	Simulation function
<Right mouse key>	Moves the image section when the mouse is moved while the right mouse button is pressed. In debug mode, the menu selection to control the recording is displayed after a short delay.
<Mouse wheel>	Scrolling up zooms in and scrolling down zooms out the image centrally to the mouse position.

Tab. 4-7: Shortcuts – simulation functions

5 Commissioning

5.1 Commissioning procedure: Example of milling with a tool management

1. Specifying the kinematic model

Use the Virtual Machine Builder to create the kinematic model of the machine (refer to [chapter 5.2.1 "Machine model and kinematic model" on page 40](#)).

Take all motion units to be simulated into account by specifying the respective traversing ranges.

Ideally, STL resources are additionally assigned to these traversing ranges.

Furthermore, define them in the tool and workpiece holder.

The machine model is stored as XML file with all relevant STL files in the model directory.

2. Adjusting the configuration file

(refer to [chapter "Defining the machine type" on page 42](#))

1. Defining the machine type (see [chapter "Defining the machine type" on page 42](#))
2. Referencing the machine model (refer to [chapter "Referencing the machine model" on page 42](#))
3. Referencing the axes from the kinematic model (refer to [chapter "Referencing axes from kinematic model" on page 43](#))
4. Referencing the "Tool" node (see [chapter "Referencing the ToolNodes" on page 45](#))
5. Definition of the tool holder geometry (optional) (refer to [chapter "Definition of the tool holder geometry \(optional\)" on page 47](#))

The configuration file is configured completely.

Traversing motions of the machine axes are correctly simulated in the simulation control.

All machine parts additionally defined in the model are displayed correctly as well.

3. Resource definition

(refer to [chapter 5.3 "Resource definition" on page 48](#))

1. Description of the tools used by means of correction data and technology data in the tool data set (refer to [chapter "Tool definition via tool management" on page 48](#))
2. If required, i.e. if not yet defined in the model, create a raw part using the definition dialog (refer to [chapter "Raw part definition via definition dialog" on page 51](#))
3. If required, define the clamping devices using the STL file import function (refer to [chapter "Clamping device definition via STL file import" on page 55](#))

The active tool is correctly displayed including all relevant geometries.

The raw part is at the correct position.

Clamping devices are displayed correctly and they are applied in the collision monitoring.

4. Program definitions

(refer to [chapter 5.4 "Program definition" on page 55](#))

1. If there is a normal NC program processing using a real control or the EMU, no specific features have to be considered.

When using the EMU with a fast NC program processing, modify the tool change program and ensure that the simulation can read changes made to the active tool. The standard tool change cycle TCH provides a template (refer to ["Special feature when using the fast EMU" on page 55](#)).

2. Zero offsets or placements have to be considered when defining the workpiece.

Tool changes are correctly registered in the simulation.

The position of the workpiece matches the machining program or the preset zero offsets.

5.2 Machine definition

5.2.1 Machine model and kinematic model

General information

The machine model includes a description of the machine structure and how it is composed of moving and stationary elements. It includes information on where, in which direction and to which traversing ranges the NC axes are applied to. This part is also called **Kinematic model**. The references for tools ("Tool" nodes) and workpieces ("Stock" nodes) are also defined in the machine model.

The physical design of machine parts, tools, clamping devices and workpieces is optionally subject to CAD models in the STL format.

Creating a machine model

The "Virtual Machine Builder" is a suitable medium to create the machine model. For more information on the Virtual Machine Builder, refer to the documentation "Virtual Machine Builder", DOK-GENERL-VMB*****-APRS-EN-P.

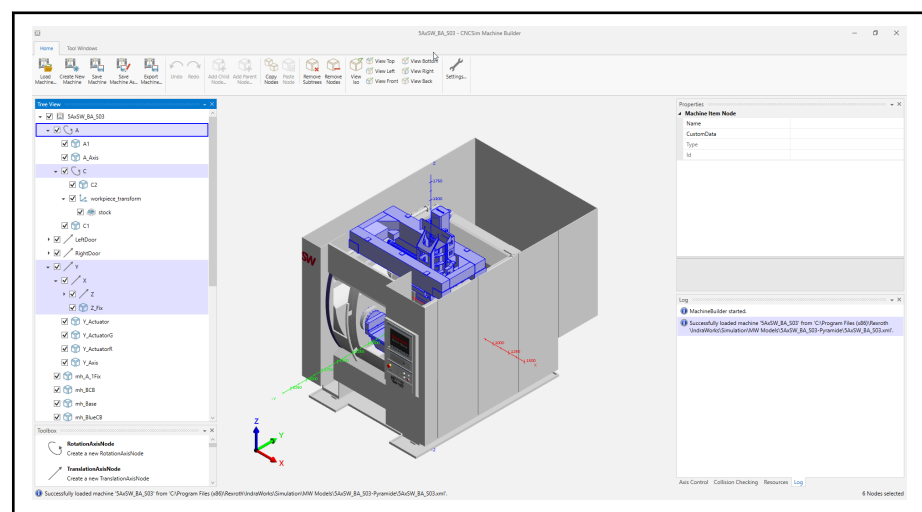


Fig. 5-1: Virtual Machine Builder with kinematic model

Data storage and activation of the simulation models

The machine model for the simulation in IndraWorks Operation is provided differently depending on the installed IndraWorks version.

Data storage for 15V06 and older

Copy the complete folder of the model created with the Virtual Machine Builder to the folder "Simulation\MW Models" below the IndraWorks installation folder using the File Explorer or a comparable tool or export the machine model in the Virtual Machine Builder directly to the mentioned folder. Both variants require administrator rights

Data storage for 15V08 and later

Simulation data is managed in IndraWorks Engineering in the Project Explorer below the "Models" node.

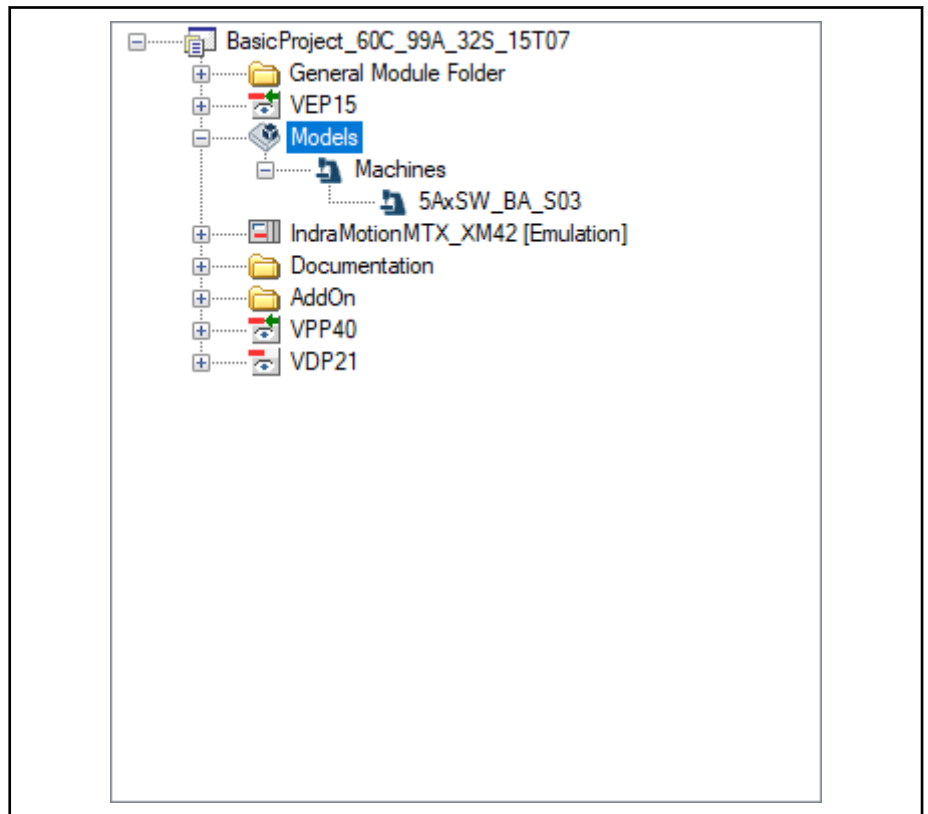


Fig. 5-2: "Models" node in the Project Explorer

The "Models" always includes a "Machines" child node. One or multiple machine models can be managed below the child node.

Select the menu function **Add** to add a model created with the Virtual Machine Builder. Select the storage folder of the machine model in the open dialog and open it with <Open>. The relevant XML file is selected and can also be opened with <Open>. The machine model is then displayed under the "Machine" node.

Go to the newly created node of the machine model and select the menu item **Apply** to provide the machine model for its use in IndraWorks Operation. The result of the transfer is shown in the message window under "Apply simulation model"



Only one model can be used in IndraWorks Operation.

When starting IndraWorks Operation or at its runtime, a dialog is displayed that informs on the provision of the new machine model. To apply the new machine model, press **Yes**.

5.2.2 Configuration file

General information

The NC Simulation is configured using the configuration file "**MTX.Simulation.Options.xml**" in the user project.



For editing, open this file with <Options> in the user interface.



After changing the settings in IndraWorks Operation, the changes made in IndraWorks Engineering have to be applied to the user project on the node of the visualization device using the function **Visualization data ▶ Load changes from operating station**.

Defining the machine type

The following types can be selected when defining the machine type:

Value	Machining method
Mill	Boring, milling, shape cutting
Turn	Turning (being prepared)
Punch	Punching
AM	Additive machining

Tab. 5-1: Machining method in the NC simulation
Machine type definition

```
...
<!--*****-->
<!--***** Machine *****-->
<!--*****-->
<!--MachiningType: Possible values are Mill, Turn, Punch and AM for additive manufacturing-->
<MachiningType>Mill</MachiningType>
...
```

Referencing the machine model

Referencing the machine model for 15V06 and older

Specify the name of the machine model in the file "**MTX.Simulation.Options.xml**". It is expected in the folder with the same name below the IndraWorks installation folder in the "Simulation\MW Models" subfolder

Referencing the machine model for 15V08 and following

It is referenced by applying the machine model to IndraWorks Engineering and the subsequent "Apply" request in IndraWorks Operation. If no machine model name is entered into the file "**MTX.Simulation.Options.xml**", the latest applied model or the standard model is shown in the simulation. If a machine model name was defined, the transferred machine model is only accepted if the name is the same. Otherwise, if the referenced machine model is different from the applied machine model, a message is displayed in the window stating that the referenced machine model is missing.

Machine model in the configuration

```
...
<!--*****-->
<!--***** Machine *****-->
<!--*****-->
<!--MachiningType: Possible values are Mill, Turn, Punch and AM for additive manufacturing-->
```

```
...  
<MachineModel>5AxSW_BA_S03</MachineModel>  
...
```

Referencing axes from kinematic model

The axis ID in the machine model has to be referenced in pairs with the respective axis numbers of the machine configuration (MACODA) in the configuration file.

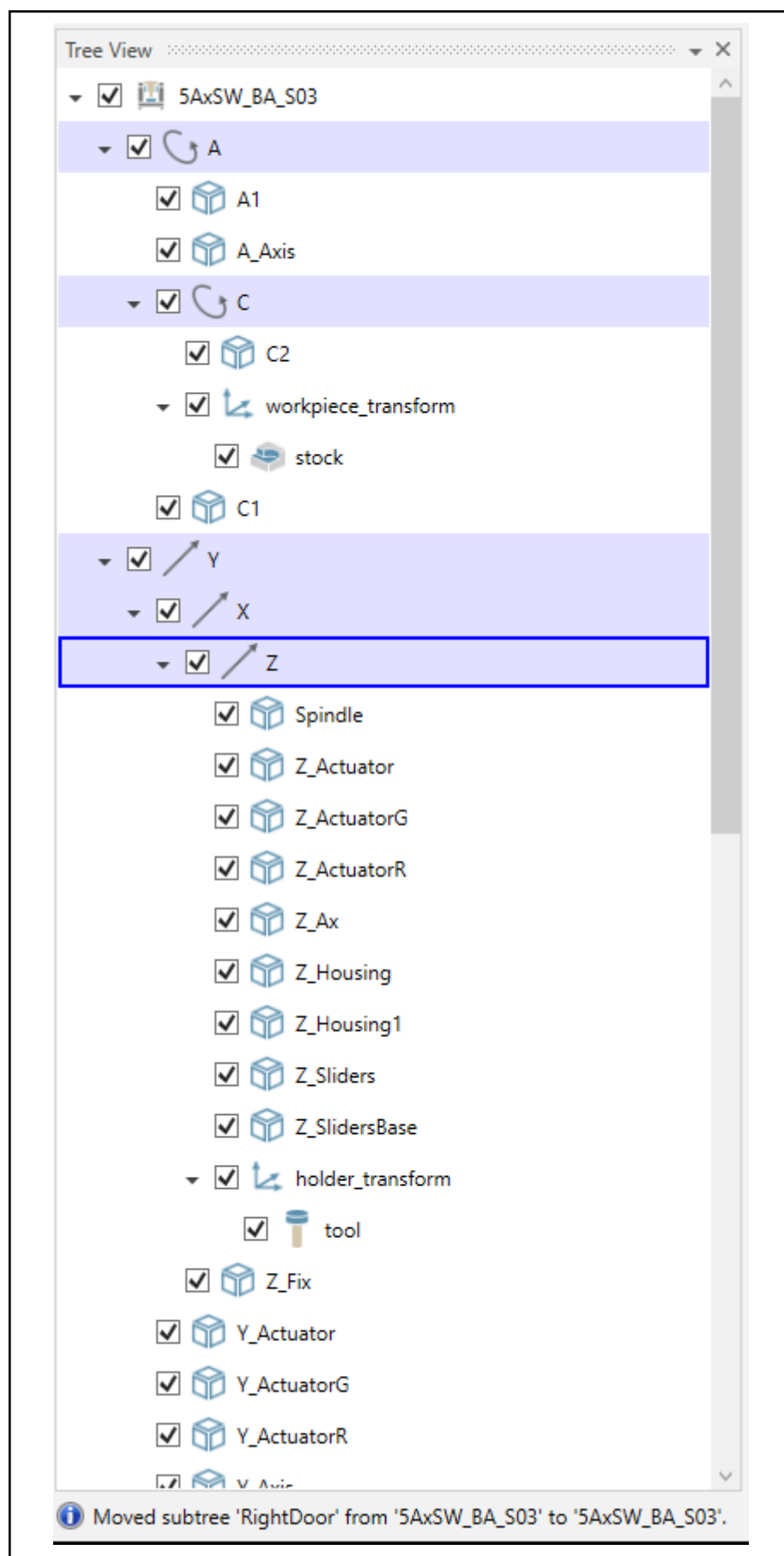


Fig. 5-3: Axis ID from the machine model

Parameter ID	Value
MAIN/Dr1/SysDrName	X1
MAIN/Dr2/SysDrName	Y1
MAIN/Dr3/SysDrName	Z1
MAIN/Dr4/SysDrName	A1
MAIN/Dr5/SysDrName	C1
MAIN/Dr6/SysDrName	MAG
MAIN/Dr7/SysDrName	Y2
MAIN/Dr8/SysDrName	C

Fig. 5-4: Axis numbers from the machine configuration data
Axis definition in the configuration

...

```

...
<AxisConnection>
  <!--Define the drive number [1..NofDr] as value input for each simulated axis (attribute Name).
    Use PUNCH for punch head, PUNCHDOWN for punch down axis and DOOR for door opening function.-->
  <!--The axis value and the visibility of child geometry elements could be defined individually
    in inner text of Axis item. Sample (CPL) :-->
  <!--<Axis Name="YClamp2">
  IF (SD.SysARA[2].invalid=1)AND(SD.SysARA[2].AreaType=1)
  THEN VALUE=SD.SysARA[2].Pos2 : VISIBLE?=TRUE ELSE VALUE=-9999 : VISIBLE?=FALSE ENDIF</Axis>-->
  <Axis Name="A" Drive="4"></Axis>
  <Axis Name="B" Drive="undefined"></Axis>
  <Axis Name="C" Drive="8"></Axis>
  <Axis Name="LeftDoor" Drive="DOOR"></Axis>
  <Axis Name="RightDoor" Drive="DOOR"></Axis>
  <Axis Name="X" Drive="1"></Axis>
  <Axis Name="Y" Drive="2"></Axis>
  <Axis Name="Z" Drive="33"></Axis>
</AxisConnection>
...

```

Referencing the ToolNodes

The ID of the ToolNode in the machine model has to be referenced in the configuration file.

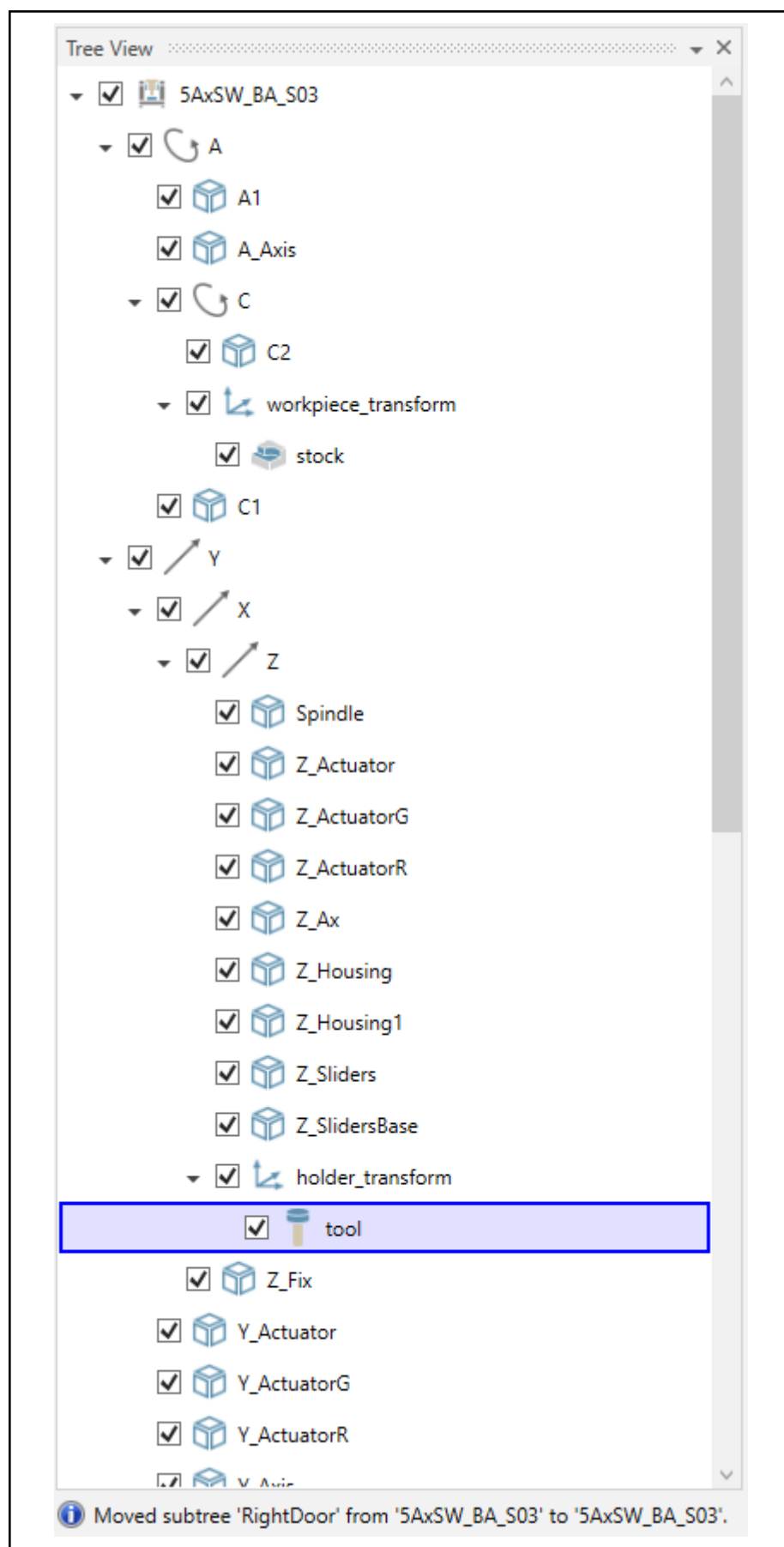


Fig. 5-5: ID of the ToolNode in the machine model

ToolNode of the machine model in the configuration

```
...
<ToolConnection>
  <!--Define how each tool node Name is connected with the NC.
    The attribute "Name" may contain wildcards like '*' and '?'.-->
  <ToolNode Name="tool" Channel="1" Function="Regular">
...
  </ToolNode>
</ToolConnection>
...
```

Definition of the tool holder geometry (optional)

Optionally, a tool holder geometry can be created in the tag <ToolConnection><ToolNode><Tool> in the configuration file using the variable "HOLDER\$" (see "Syntax of CUTT\$, SHAFT\$, ARBOR\$ and HOLDER\$" on page 112).

Example of a tool holder definition in the configuration

```
...
<Tool>
  HOLDER$="REV,63,0,63,4,55,6,55,10,63,12,63,26"
  REM ARBOR$="D60 L20 CHL3"
</Tool>
...
```

Fig. 5-6: Note: Chuck definition is commented

Technology-dependent settings

Defining the tool activation

For the machining technologies "Shape cutting" and "Additive machining", it might be required to activate or to deactivate the tool by switching on or off a laser beam for example. The simulation has to know the control signals used.

Example of a tool activation in the configuration

```
...
<IsToolActive>ACTIVE?=(SD.SysAdvDistSig[1]SignalStatus AND 128)<>0</IsToolActive>
...
```

Fig. 5-7: Example of a switch signal: Bit 8 of the variable SD.SysAdvDistSig[1]SignalStatus

Setting to remove cut-out parts

For the machining technology "Shape cutting", it might be preferred to remove completely cut-out parts (waist or finished part) from the raw part (sheet). Define the <ChunkDetection> variable.

Example of a ChunkDetection setting

```
...
<ChunkDetection>Timer</ChunkDetection>
<ChunkDetectionInterval>2500</ChunkDetectionInterval>
...
```

Fig. 5-8: Example of ChunkDetection via timer

Settings for punching

For punching applies that there is often a big difference in size between the workpiece and the tool. I.e. very big sheets (workpiece) are machined using relatively small punching or signing tools.

To obtain a good simulation result, it might be required to increase the value of the variable <ModelSamplesPerDiagonal>.

Example of a punching setting

...

```
<ModelSamplesPerDiagonal>800</ModelSamplesPerDiagonal>  
...
```

Fig. 5-9: Example of increased representation accuracy



When punching, further settings can be required in the configuration file as well as in the machine configuration data.
e.g. parameters: "EnablPunch"; MACODA no.: 8001 00010 = Yes

Memory-critical machinings

Runtime-intense NC programs, big workpieces with high resolution and multi-piece clampings can cause a high RAM consumption and performance losses or the user interface can even be aborted. When configuring, note which options are actually required or useful and deselect all other features.

Reasons of highest energy consumption:

- Representation accuracy `ModelSamplesPerDiagonal`.
- High quality zooming: Set `<HighQualityZooming>None</HighQualityZooming>`
- Collision detection: Set `<CollisionDetection>NoMachiningHistory</CollisionDetection>`
- Tool path: Set `<ToolpathEnabled>None</ToolpathEnabled>`

5.3 Resource definition

5.3.1 Tool definition

General information

There are the following options to define the machining tool in the NC Simulation:

Tool management	If the tool management is active
STL file to describe the tool geometry	For more complex tool geometries that are not explained in the tool catalog and only if the tool management is active
NC command	Using D-corrections if no tool management is active
Configuration file	To modify and supplement already defined tools

Tab. 5-2: Sources of the tool definition

Tool definition via tool management

The tool is described in the tool geometry data and in the tool technology data of the tool data set.

The active tool data is always used in the simulation. If no tool is active, no tool is displayed.

In the operating area "Tool management", in the additional operating area of the multi-touch interface in the tab "Tool preview", the selected tool is visualized by analyzing the relevant tool data as in the NC Simulation.

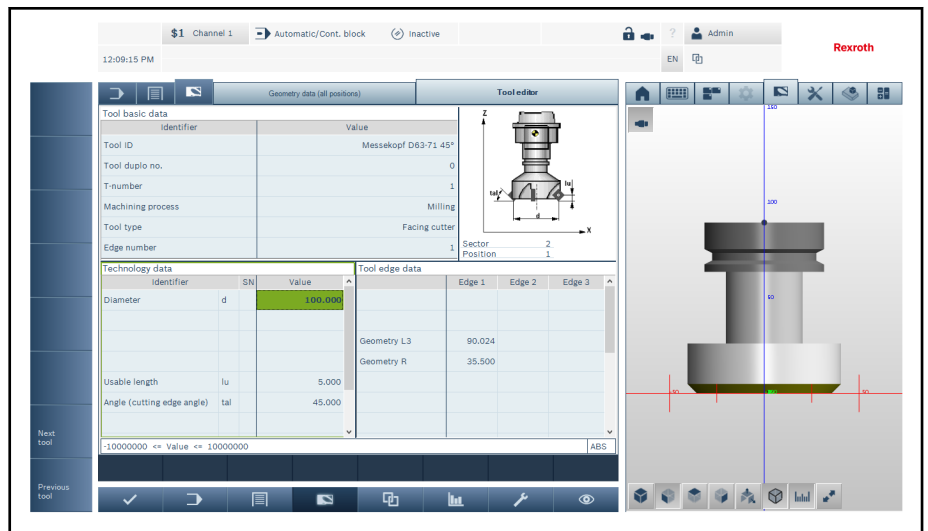


Fig. 5-10: Tool geometry data and tool technology data with tool preview in the tool editor

The colors to represent the individual sections in the configuration file **MTX.Simulation.Options.xml** can be defined for the following tools:

- Cutting edge
- Shaft
- Tool holder (chuck, collet, etc.)
- Holder

Program:

```
...
<Colors>
<!--*****-->
<!--***** Colors *****-->
<!--*****-->
<Background>#F6F6F6</Background>
<CuttingPart>#8C8C00</CuttingPart>
<Shaft>#F6F6F6</Shaft>
<Arbor>#D3D3D3</Arbor>
<Holder>#808080</Holder>
...
```

More complicate tool geometries that cannot be described using the tool technology data can also be defined via an STL file.

It is required that there is an STL description file for the respective tool in the installation directory "<IndraWorks>\Simulation\MW Models\Tools".

The format of the STL file name should be as follows: Tool ID + "_Extension".

"Extension" = cutt, shaft, arbor, holder

Example: "Tool ID_cutt.stl" for milling

Tool definition via NC command

In the NC program, the NC cycle can be used to define a tool.

This is especially required if there is no tool management.

Cycle	Syntax	Description
TOOL/DRILL	TOOL("DRILL",,,) (refer to chapter 7.12 "TOOL/DRILL Drill, countersink, reamer" on page 80)	Drilling tool, countersinking cutter and reamer
TOOL/CDRILL	TOOL("CDRILL",,,,,) (refer to chapter 7.13 "TOOL/CDRILL Step drill, centering drill" on page 81)	Center drill and step drill
TOOL/BBAR	TOOL("BBAR",,,,,) (refer to chapter 7.14 "TOOL/BBAR Boring bar, backward counterbore" on page 83)	Boring bar and backward counterbore
TOOL/MILL	TOOL("MILL",,,) (refer to chapter 7.15 "TOOL/MILL Cylindrical, angular and ball cutter" on page 85)	Cylindrical angle milling and spherical cutters
TOOL/CHAMFER	TOOL("CHAMFER",,,) (refer to chapter 7.16 "TOOL/CHAMFER Engraving milling cutter, chamfer milling cutter, face milling head" on page 88)	Engraving milling cutter, chamfer milling cutter and face milling head
TOOL/CRMILL	TOOL("CRMILL",,,,,) (refer to chapter 7.17 "TOOL/CRMILL Quarter round profile milling cutter" on page 91)	Quarter-round profile milling cutter
TOOL/GRIND	TOOL("GRIND",,,) (refer to chapter 7.18 "TOOL/GRIND Grinding wheel" on page 93)	
	TOOL()	Remove tool

Tab. 5-3: NC command "TOOL"



The tool graphic remains until cancelation by the command "TOOL" or the command "TOOL ()". Then, the active tool is shown using the respective data set of the tool management or no tool is shown at all anymore.

5.3.2 Raw part definition

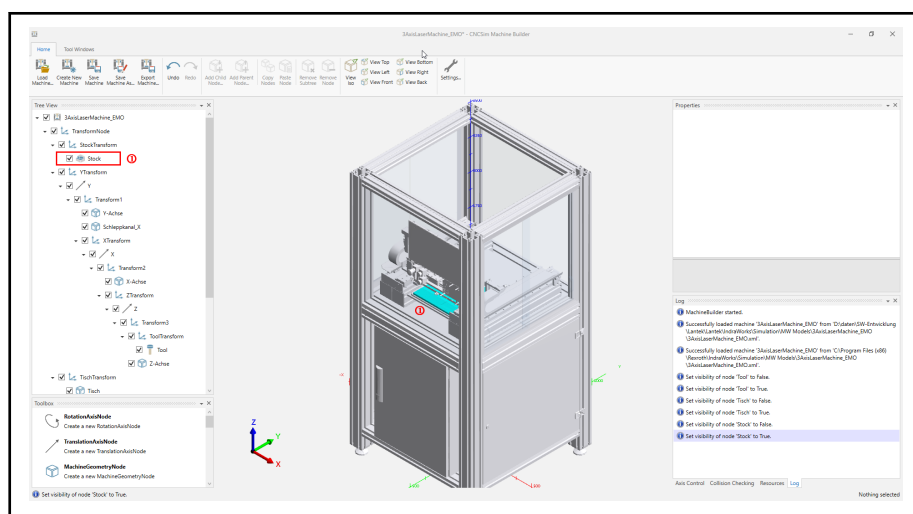
General information

There are the following options to define the raw part in the NC Simulation:

Machine model	The same raw part is always used
Definition dialog	Manual definition of a cubical or cylindrical raw part Mostly in test mode
NC command	Suitable for the simulation at runtime with different raw parts
STL file import	Suitable for already machined parts in test mode

Tab. 5-4: Sources of the raw part definition

Raw part definition via the machine model



1 Raw part definition and representation in the model

Fig. 5-11: Raw part definition via machine model

Raw part definition via definition dialog

Specify the raw part in the definition dialog. To open the definition dialog, go

to <Cuboid, sheet...>  or <Cylinder> .

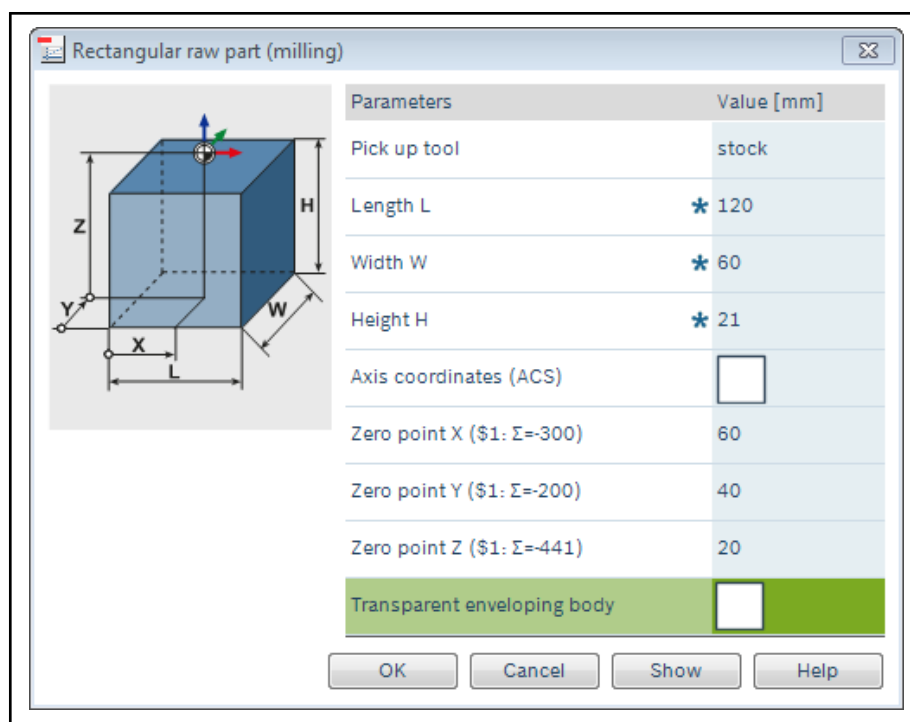


Fig. 5-12: Definition of a rectangular raw part during milling

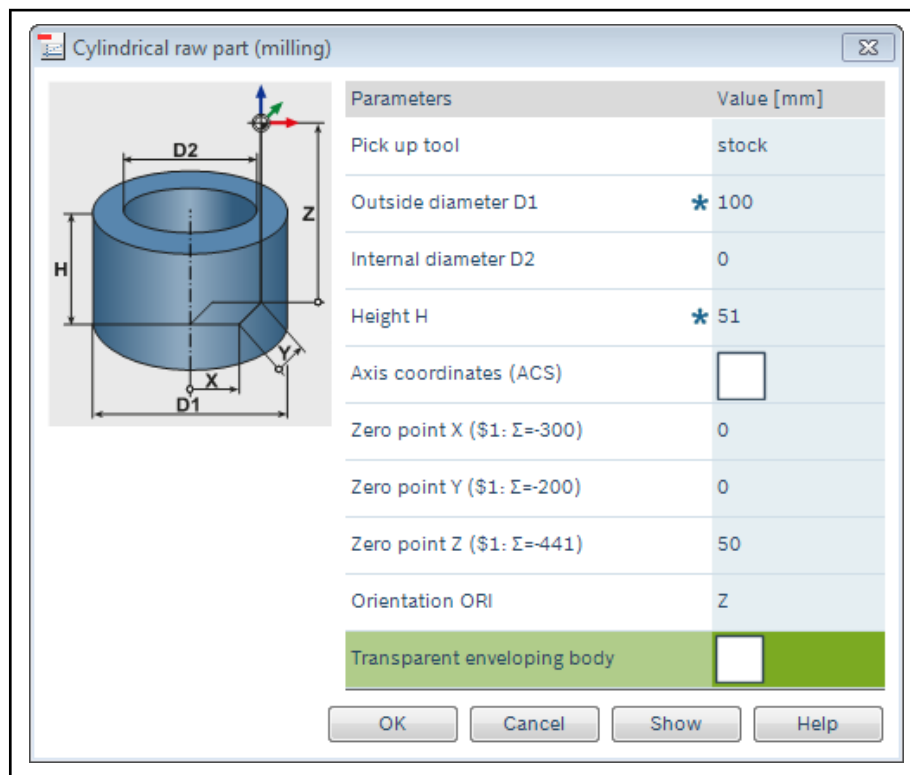


Fig. 5-13: Definition of a cylindrical raw part during milling

Raw part definition via NC command

In the NC program, the NC cycle can be used to define a raw part.

STOCK/BLOCK	STOCK("BLOCK",,,) (refer to chapter 7.2 "STOCK/BLOCK Rectangular raw part (milling) " on page 59)	Rectangular raw part	Milling
STOCK/CYLINDER	STOCK("CYLIN- DER",,,,,) (refer to chapter 7.3 "STOCK/CYLINDER Cylindrical raw part (milling) " on page 61)	Cylindrical raw part	Milling
STOCK/BAR	STOCK("BAR",,,,,) (refer to chapter 7.4 "STOCK/BAR Bar section (turning)" on page 63)	Bar section	Rotate
STOCK/POLYGON	STOCK("POLY- GON",,,) (refer to chapter 7.5 "STOCK/POLYGON Polygon (turning)" on page 65)	Multi-edge	Rotate
STOCK/SHEET	STOCK("SHEET",,,) (refer to chapter 7.6 "STOCK/SHEET Sheet (punching)" on page 68)	Sheet	Punch- ing
STOCK/REMOVE	STOCK() (refer to chapter 7.8 "STOCK/REMOVE Remove workpieces/ models" on page 73)	Remove workpiece	
STOCKIMP	STOCKIMP() (refer to chapter 7.7 "STOCKIMP Import model" on page 70)	Import model	

Tab. 5-5: NC command "STOCK"

Raw part definition via STL file import

It is also possible to define the raw part via an import dialog. Go to <Im-

port...>  .

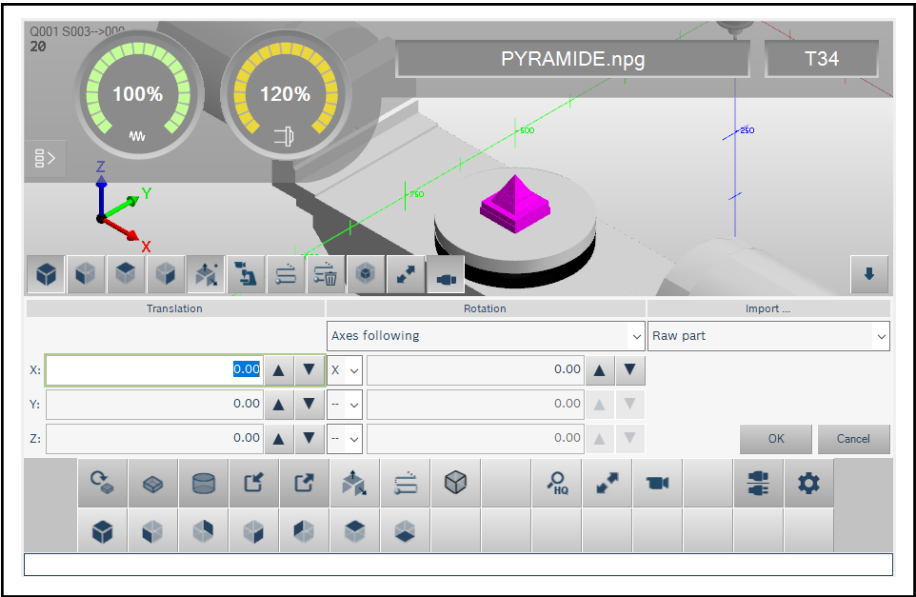


Fig. 5-14: Raw part definition via “Import” dialog

5.3.3 Clamping device definition

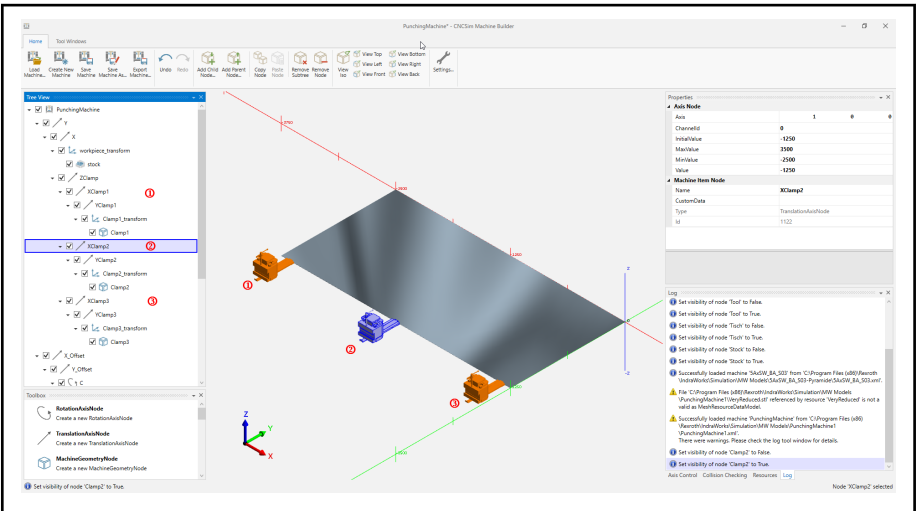
General information

There are the following options to define the clamping devices in the NC Simulation:

Machine model	The same clamping device is always used
NC command	Suitable for the simulation at runtime with different clamping devices
STL file import	Suitable for clamping devices during preparation

Tab. 5-6: Sources of the clamping device definition

Clamping device definition via the machine model



1...3 Clamping device definition and representation in the model

Fig. 5-15: Clamping device definition via machine model

Clamping device definition via NC command

In the NC program, the NC cycle can be used to define a clamping device.

STOCKIMP	STOCKIMP()	Import model	
----------	------------	--------------	--

Tab. 5-7: NC command "STOCKIMP"

Clamping device definition via STL file import

It is also possible to define the clamping device via an import dialog. Go to

<Import...>  .

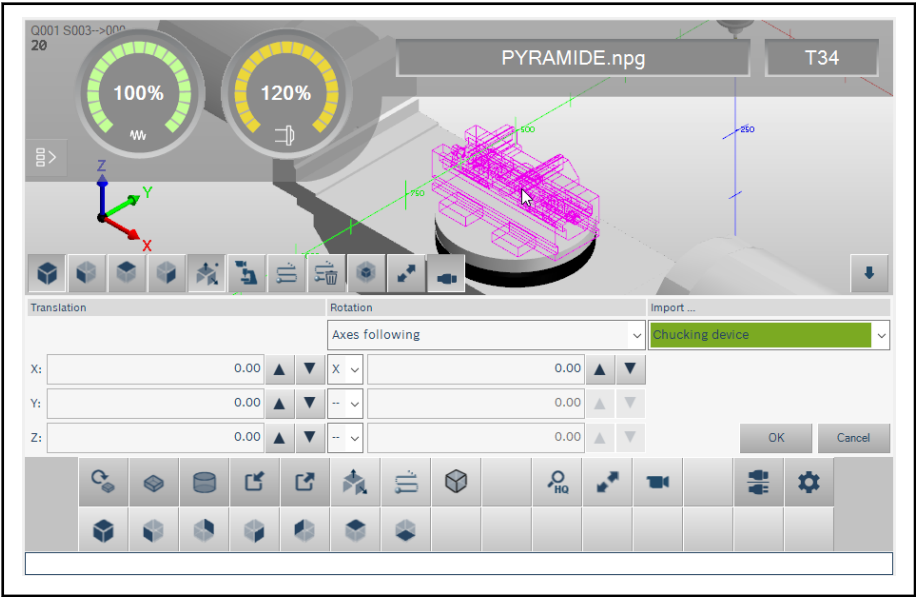


Fig. 5-16: Clamping device definition via "Import" dialog

5.4 Program definition

5.4.1 NC program

Program processing with tool management

If there is a normal NC program processing using a real control or the EMU, noting has to considered. When using the tool change, e.g. TCH, the simulation automatically receives all information on the active tool.

Special feature when using the fast EMU

When using the EMU with a fast NC program processing, modify the tool change program to ensure that the simulation can read changes made to the active tool.

The standard tool change cycle TCH provides a template.

Modifications in the tool change program

```
; ----- Werkzeug aufnehmen
...
; Synchronisation mit der Simulation
10 IF SD.Sys_Simulation THEN
20   CALL STP_SYNCSIM
30 ENDIF
...
```

Program processing without tool management

If the tool management is not used, the D-corrections of the NC have to be provide the tool correction data. In this case, it is required to provide the description of the active tool to the simulation via a respective TOOL command. It is to be provided in the NC program instead of the tool change cycle (refer to [chapter 5.3.1 "Tool definition" on page 48](#)).

The raw part can also be defined in the NC program using the STOCK command. It is important that the previously defined zero offsets are applied to the raw part definition in addition to the position definitions in the STOCK command. If the raw part is defined in the machine model or by using a pre-definition in the definition dialog or an STL file import, no NC program instructions are required (refer to [chapter 5.3.2 "Raw part definition" on page 50](#)).

5.4.2 Zero offsets und placements

Zero offsets and placements in the NC program especially affect the raw part definition for the NC Simulation.

When using the NC command STOCK and as well as the raw part definition via the definition dialog or the STL file import, offsets can be entered to determine the position of the raw part which can either be applied absolutely or additively.

These offsets are applied until the raw part is removed or renewed via <Re-

set raw part...>  (refer to [chapter "Button block Tools" on page 28](#)).

6 Recording and playing recordings

Objective	Using the recorder included in the simulation, motion and operating sequences can be recorded. The scene can be played at any time (also outside of the IndraWorks installation) using a player. The recorder can be used for presentations, testings and to reproduce errors. Each recording contains all relevant files for the simulation which considerably facilitates the handling.
Recording with the simulation	To record data, proceed as follows: 1. Activate the debug mode in the options, <i>Optimizing</i> substructure. Refer to chapter 4.1 "Operating keys and corresponding dialogs" on page 19, key  "Options". For information on the XML tag, refer to chapter "General information" on page 122. 2. Activate the context menu of the display surface of the simulation and activate "Start recording". 3. Simulate the sequence to be recorded and enable the context menu again. Click on "Complete record". 4. A file selection dialog is displayed in which the created recording file (".rec" extension) is selected. The file can now be renamed or copied. Click on <Open> to immediately open the recording with the player. If you do not want to play back the recording immediately, click <Cancel>.
Providing the runtime environment of the player	To play the recording outside IndraWorks installation, create a multi-part archive under the folder MyDocuments\SimulationPlayer with the name MTX.Simulation.TestAppRuntime.zip.001 to <n> for all required files by executing <pre><IndraWorks installation directory> \MTX.Simulation.TestApp.CreateRuntimeArchive.bat</pre> with admin rights. Unzip the archive in the execution folder of the player.
Starting the player	If the player is not started directly after the recording has been completed, start the application "MTX.Simulation.TestApp.exe" via the function "Simulation player" in the Windows start menu (via "All programs ► Rexroth ► IndraWorks_xx.yy.zzz.a ► MTX Tools ► Simulation player").

Player layout

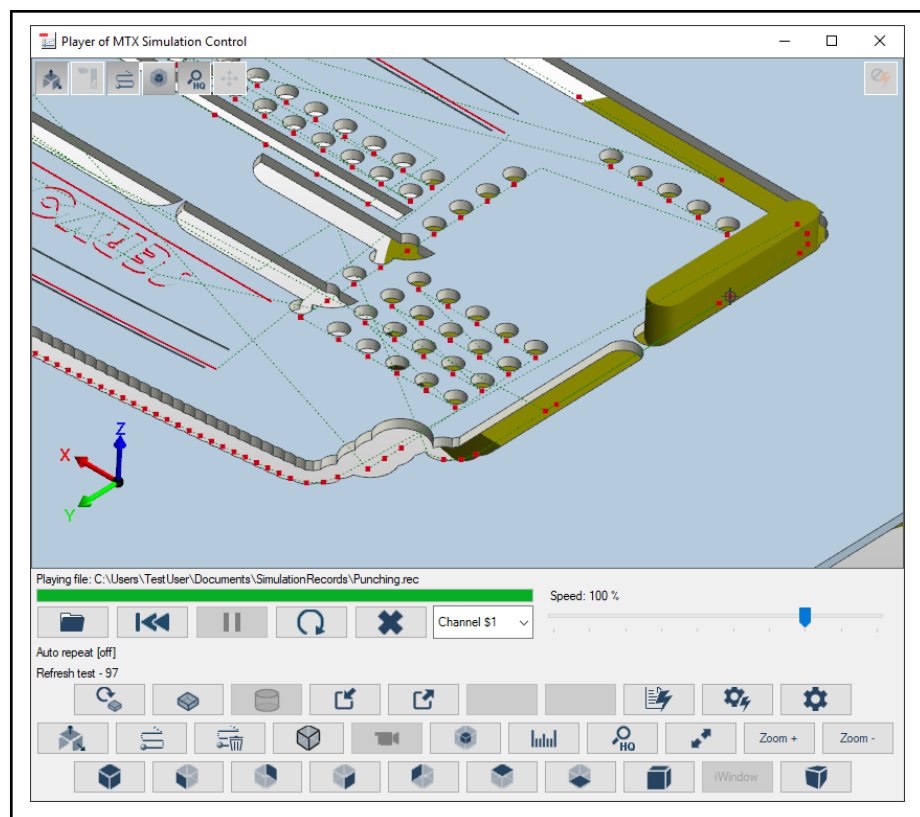


Fig. 6-1: Simulation player

Operating the player

-  displays a file selection dialog used to select and open the recording file
-  starts the animation
-  cancels the animation
-  resets the replay
-  causes a continuous playback until it is pressed again. In the label, the repetitions are counted with "Auto repeat".
-  <Close> closes the player
- The drop-down list with "Channel \$..." corresponds to the selection of the active channel in IndraWorks Operation
- Use "Speed" to accelerate or decelerate the animation
- All other operating keys correspond to the known simulation operating keys



It is not possible to change a recording by user input.

If the options are changed in the player, the change applies until the next opening of a recording.

If raw part, tool path, etc. are changed, the change applies until reset to default by the  key.

7 Program instructions for the simulation

7.1 General information on program instructions for the simulation

Functional scope	The instructions for the simulation are cycle subroutines providing the simulation with information on the graphic. They can be inserted into NC programs and also be used in the manual data input (MDI). The cycles cover the following functions: • Defining/importing raw parts and other geometry types • Handling geometries such as deletion or renewal • Tool descriptions
Tool description	The use of tool descriptions is one possible alternative to the tool graphic as it is based on the information from the MTX tool management. Tool descriptions can be used • if no tool management is used. The tool is compensated via D-corrections • if, for example, no matching workpiece list DBT1 is present when creating the program • if the tool management is not yet completely and functionally configured (auxiliary means to commission the simulation) Tool descriptions are made for simple drilling and milling tools. Only the dimensions of the cutting edge part are usually defined. If possible, the simulation will amend the shaft, the chuck and the holder.
Input masks	For the instructions, cycle masks are provided in the NC editor for entries under <Input assistance> --> <Simulation instructions>. For correction, use <Correction cycle/instruction>.

7.2 STOCK/BLOCK Rectangular raw part (milling)

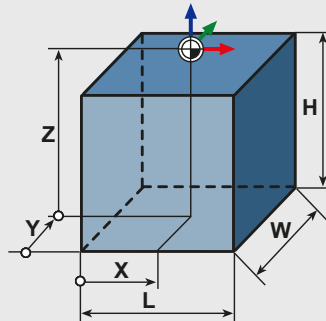
Syntax	STOCK("BLOCK",<L>,<W>,<H>,<X>,<Y>,<Z>,<V>)
Brief description	A new rectangular raw part is generated at or added to the selected workpiece holder. It is intended for the machining simulation of prismatic parts.
Overview figure	

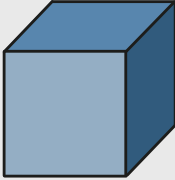
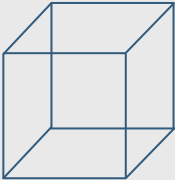
Fig. 7-1: STOCK/BLOCK instruction "Rectangular raw part"

Program instructions for the simulation

Parameter

Parameter	Name	Properties	Values	Pre-assignment
L	Length	Mandatory	>0	
W	Width	Mandatory	>0	
H	Height	Mandatory	>0	
X	Zero point in X	Optional	Incremental	0
Y	Zero point in Y	Optional	Incremental	0
Z	Zero point in Z	Optional	Incremental	0
V	Modeling	Optional		0

Tab. 7-1: Parameter call

V	Effect
0	Volume
1	Enveloping body
V=0  V=1 	

Tab. 7-2: Description of the parameter V - Modeling

Description

L Length

Defines the X-expansion of the cuboid.

W Width

Defines the Y-expansion of the cuboid.

H height

Defines the Z-expansion of the cuboid.

X, Y, Z zero point

Specifies the distance of the current workpiece zero point from the front left and the bottom corner.

V modeling

Specifies whether a volume model for material removal simulation or an enveloping body is to be drawn. The latter is an alternative to the import of a CAD model in a complex geometry.

- Special features**
- The parameters are position parameters without an address
 - In the workpiece coordinate system, the active placements and the zero offsets in the linear main axes are taken into account. It can be required to "retract" the offsets in rotary axes and telescope axes before the start
 - G53 combined with G153 arranges the geometry in the coordinate system of the tool holder. It can deviate from the machine coordinate system. The positions of further axes in the kinematic chain have no effect
 - Cannot or not completely be programmed if there is an active axis transformation

The parameters are position parameters without an address.

Procedure/processing The workpiece is displayed as matching the previously activated workpiece zero point. The tool path is deleted.

Example Cuboid in volume representation, 100 in length, 70 in width and 30 in height. The zero point of the workpiece (G54) should be positioned in the center of the upper side.

Program:

```
N10 G54 STOCK("BLOCK", 100, 70, 30, 50, 35, 30)
```

7.3 STOCK/CYLINDER Cylindrical raw part (milling)

Syntax STOCK("CYLINDER", <D1>, <D2>, <H>, <X>, <Y>, <Z>, <V>)

Brief description A new cylindrical raw part, optionally with continuous inner boring, is generated at or added to the selected workpiece holder. It is intended for the machining simulation on milling machines.

Overview figure

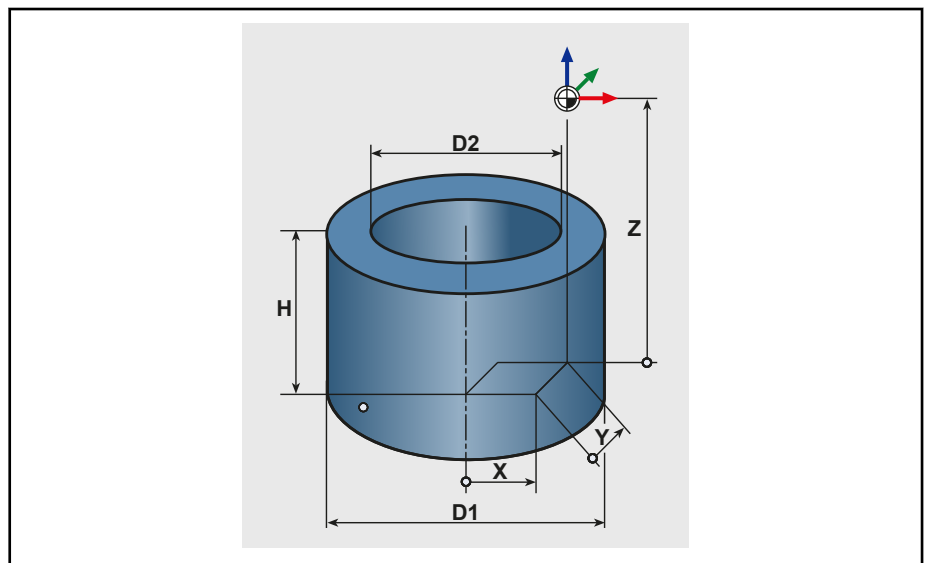


Fig. 7-2: STOCK/CYLINDER instruction "Cylindrical raw part"

Parameter

Parameter	Name	Properties	Values	Pre-assignment
D1	Outer diameter	Mandatory	>0	
D2	Inner diameter	Optional	>=0	0
H	Height	Mandatory	>0	
X	Zero point in X	Optional	Incremental	0

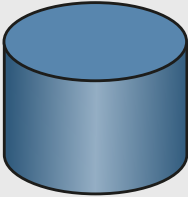
Program instructions for the simulation

Y	Zero point in Y	Optional	Incremental	0
Z	Zero point in Z	Optional	Incremental	0
V	Modeling and orientation	Optional		0

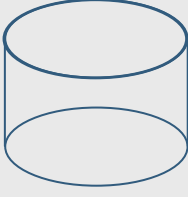
Tab. 7-3: Parameter call

V units digit	Effect
0	Volume
1	Enveloping body

V=x0



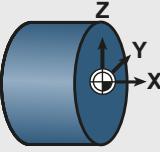
V=x1



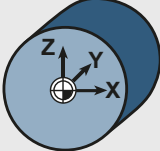
Tab. 7-4: Description of the units digit parameter V - modeling

V tens digit	Effect
0	Center axis Z-parallel
1	Center axis Y-parallel
2	Center axis X-parallel

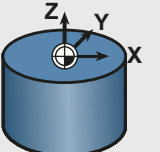
V=2x



V=1x



V=0x



Tab. 7-5: Description of the tens digit of parameter V - orientation

Description**D1 Outer diameter**

Defines the outer diameter of the cylinder.

D2 Inner diameter

In case of values greater than 0, it defines a continuous inner boring.

H height

Defines the height of the cuboid.

X, Y, Z zero point

Specifies the distance of the current workpiece zero point from the center point of the left, front or bottom base.

V modeling and orientation

The units digit specifies whether a volume model for material removal simulation or an enveloping body is to be drawn. The latter is an alternative to the import of a CAD model in a complex geometry.

The tens digit affects the center axis orientation.

Special features

- The parameters are position parameters without an address
- In the workpiece coordinate system, the active placements and the zero offsets in the linear main axes are taken into account. It can be required to "retract" the offsets in rotary axes and telescope axes before the start
- G53 combined with G153 arranges the geometry in the coordinate system of the tool holder. It can deviate from the machine coordinate system. The positions of further axes in the kinematic chain have no effect
- Cannot or not completely be programmed if there is an active axis transformation

Procedure/processing

The workpiece is displayed as matching the previously activated workpiece zero point. The tool path is deleted.

Example

Hollow cylinder in volumetric display with a diameter of 100, an inner diameter of 40 and a height of 30. The center line has to be parallel to Y. The workpiece zero point (G54.2) is to be positioned in the center of the Y-positive upper side.

Program:

```
N10 G54.2 STOCK("CYLINDER",100,40,30,,30)
```

7.4 STOCK/BAR Bar section (turning)

Syntax STOCK("BAR",<L>,<D1>,<D2>,<Z>,<V>)

or

STOCK(<L>,<D1>,<D2>,<Z>,<V>)

Brief description

A new cylindrical raw part, optionally with inner boring, is generated at or added to the selected workpiece holder. It is intended for turning.

Overview figure

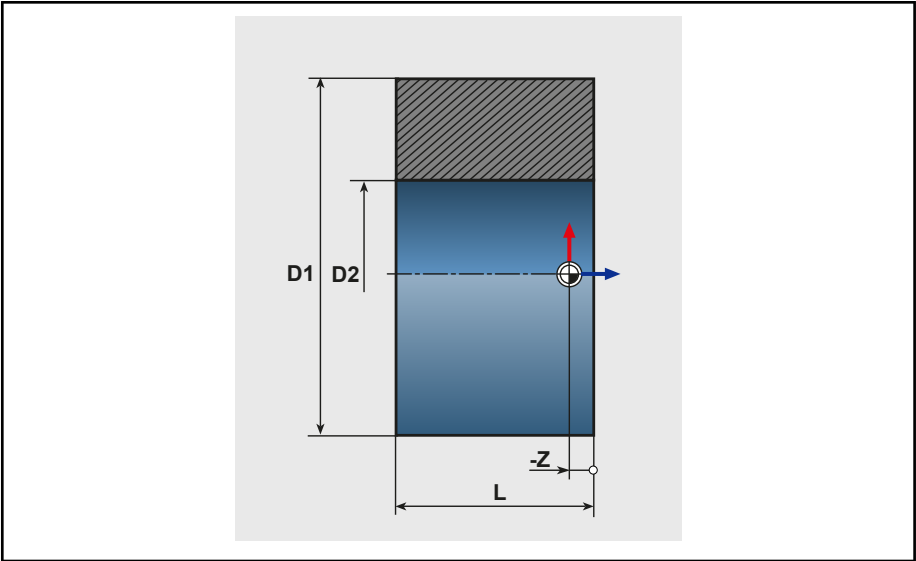


Fig. 7-3: STOCK/BAR instruction “Bar section”

Parameter

Parameter	Name	Properties	Values	Pre-assignment
L	Length	Mandatory	>0	
D1	Outer diameter	Mandatory	>0	
D2	Inner diameter	Optional	>=0	0
Z	Zero point in Z	Optional	Incremental	0
V	Modeling	Optional		0

Tab. 7-6: Parameter call

V modeling	Effect
0	Volume
1	Enveloping body

V=0

V=1

Tab. 7-7: Description of the units digit parameter V - modeling

Description

L Length

Defines the Z-expansion.

D1 Outer diameter

Defines the outer diameter.

D2 Inner diameter

In case of values greater than 0, it defines a continuous inner boring.

Z zero point offset

Specifies the distance of the current workpiece zero point from the right limiting surface (usually negative).

V modeling

Specifies whether a volume model for material removal simulation or an enveloping body is to be drawn. The latter is an alternative to the import of a CAD model in a complex geometry.

Special features

- The parameters are position parameters without an address
- In the workpiece coordinate system, the active placements and the zero offsets in the linear main axes are taken into account. It can be required to "retract" the offsets in rotary axes and telescope axes before the start
- G53 combined with G153 arranges the geometry in the coordinate system of the tool holder. It can deviate from the machine coordinate system. The positions of further axes in the kinematic chain have no effect
- Cannot or not completely be programmed if there is an active axis transformation

Procedure/processing

The workpiece is displayed as matching the previously activated workpiece zero point. The tool path is deleted.

Example

Bar section in volumetric display with an outer diameter of 60 and a length of 102. The workpiece zero point (G54.3) should be located 2 mm to the left of the right limiting surface.

Program:

```
N10 G54.3 STOCK("BAR",102,60,, -2)
```

7.5 STOCK/POLYGON Polygon (turning)

Syntax STOCK("POLYGON",<L>,<D>,<N>,<Z>,<OA>,<V>)

Brief description A new raw part is generated at or added to the selected workpiece holder as polygon. It is intended for turning.

Overview figure

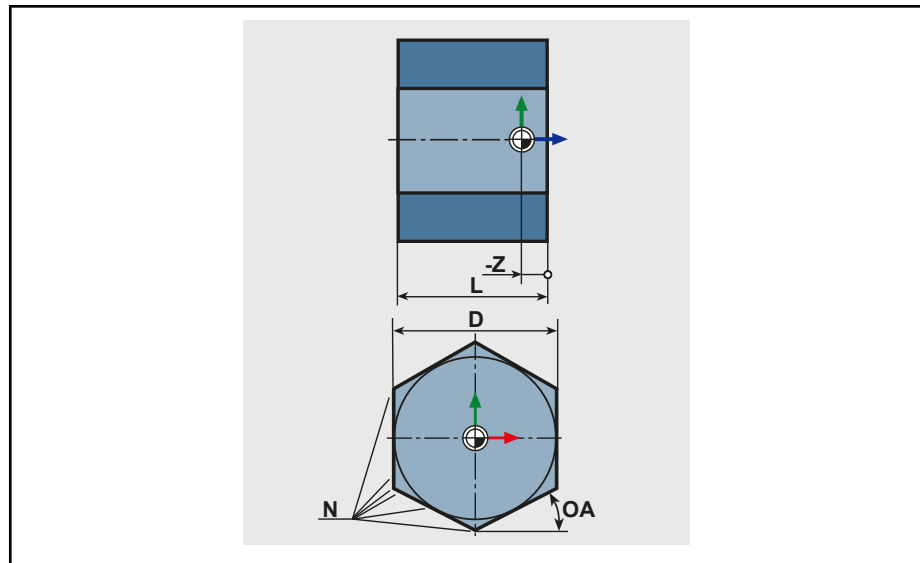


Fig. 7-4: STOCK/POLYGON instruction "Polygon"

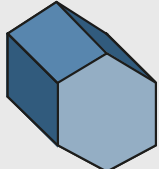
Parameter

Parameter	Name	Properties	Values	Pre-assignment
L	Length	Mandatory	>0	
D	Diameter (wrench size)	Mandatory	>0	
N	Number of edges	Mandatory	>2	
Z	Zero point in Z	Optional	Incremental	0
OA	Orientation an- gle	Optional	Degree	0
V	Modeling	Optional		0

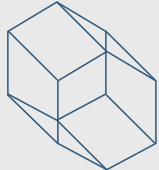
Tab. 7-8: Parameter call

V modeling	Effect
0	Volume
1	Enveloping body

V=0



V=1



Tab. 7-9: Description of the parameter V - Modeling

Description

L Length

Defines the Z-expansion.

D Diameter

Defines the diameter as wrench size.

N Number of edges

Defines the number of edges.

Z zero point

Specifies the distance of the current workpiece zero point from the right limiting surface (usually negative).

OA Orientation angle

Specifies the angle offset of a profile edge in the angle of rotation C.

V modeling

Specifies whether a volume model for material removal simulation or an enveloping body is to be drawn. The latter is an alternative to the import of a CAD model in a complex geometry.

Special features

- The parameters are position parameters without an address
- In the workpiece coordinate system, the active placements and the zero offsets in the linear main axes are taken into account. It can be required to "retract" the offsets in rotary axes and telescope axes before the start
- G53 combined with G153 arranges the geometry in the coordinate system of the tool holder. It can deviate from the machine coordinate system. The positions of further axes in the kinematic chain have no effect
- Cannot or not completely be programmed if there is an active axis transformation

Procedure/processing

The workpiece is displayed as matching the previously activated workpiece zero point. The tool path is deleted.

Example Hexagon in volumetric display with a diameter of 60 and a length of 82. The workpiece zero point (G54.4) should be located 2 mm to the left of the right limiting surface. There has to be a profile edge at angle C90.

Program:

```
N10 G54.4 STOCK("POLYGON",82,60,6,-2,30)
```

7.6 STOCK/SHEET Sheet (punching)

Syntax STOCK("SHEET",<L>,<W>,<H>,<X>,<Y>,,<V>)
Brief description A new raw part is generated at or added to the selected workpiece holder as rectangular plate. It is intended for the machining simulation of sheet machining on machine tools without z-axis (e.g. punching).

Overview figure

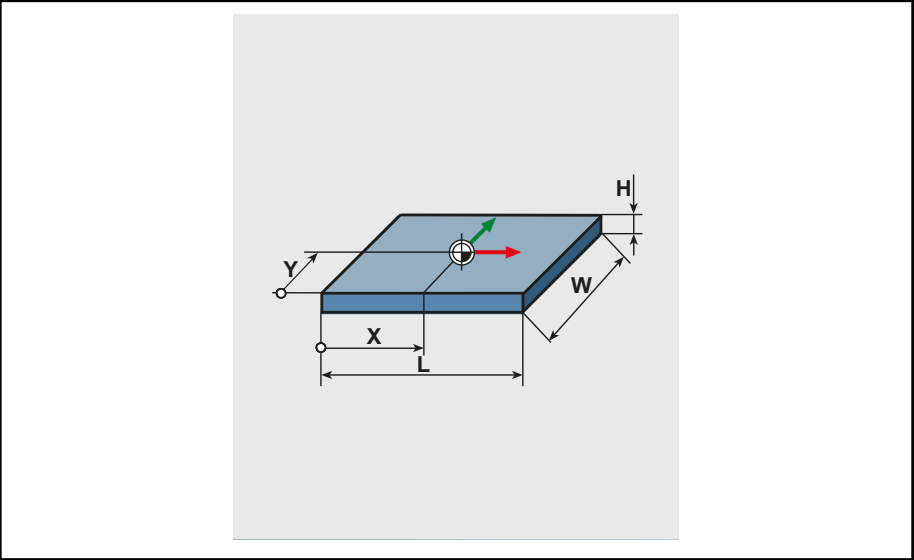


Fig. 7-5: STOCK/SHEET instruction "Sheet"

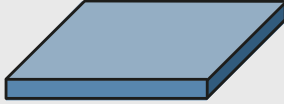
Parameter

Parameter	Name	Properties	Values	Pre-assignment
L	Length	Mandatory	>0	
W	Width	Mandatory	>0	
H	Thickness	Mandatory	>0	
X	Zero point in X	Optional	Incremental	0
Y	Zero point in Y	Optional	Incremental	0
V	Modeling	Optional		0

Tab. 7-10: Parameter call

V	Effect
0	Volume
1	Enveloping body

V=0



V=1



Tab. 7-11: Description of the parameter V - Modeling

Description**L Length**

Defines the X-expansion of the sheet.

W Width

Defines the Y-expansion of the sheet.

H Thickness

Defines the sheet thickness.

X, Y zero point

Specifies the distance of the current workpiece zero point from the front left corner.

V modeling

Specifies whether a volume model for material removal simulation or an enveloping body is to be drawn. The latter is an alternative to the import of a CAD model in a complex geometry.

Special features

- The parameters are position parameters without an address
- In the workpiece coordinate system, the active placements and the zero offsets in the linear main axes are taken into account. It can be required to "retract" the offsets in rotary axes and telescope axes before the start
- G53 combined with G153 arranges the geometry in the coordinate system of the tool holder. It can deviate from the machine coordinate system. The positions of further axes in the kinematic chain have no effect
- Cannot or not completely be programmed if there is an active axis transformation

Procedure/processing

The workpiece is displayed as matching the previously activated workpiece zero point. The tool path is deleted.

Example

Plate in volumetric display with 1,000 in length, 500 in width and 2.5 in thickness. The workpiece zero point (G54.5) has to be in the center.

Program:

N10 G54.5 STOCK ("SHEET",1000,500,2.5,500,250)

7.7 STOCKIMP Import model

Syntax STOCKIMP(P, TGT,X,Y,Z,RS,A1,A2,A3)

Brief description Imports a CAD model displayed at a workpiece holder.

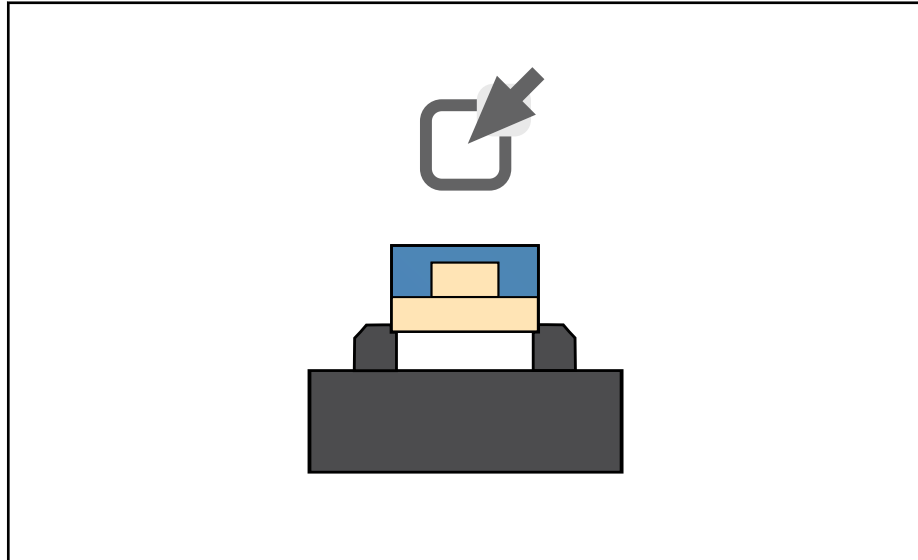
Overview figure


Fig. 7-6: STOCKIMP instruction "Import model"

Parameter

Parameter	Name	Properties	Values	Pre-assignment
P	File name	Mandatory	Max. 255 characters	
TGT	Import target	Optional		1
X	Translation X	Optional	Incremental	0
Y	Translation Y	Optional	Incremental	0
Z	Translation Z	Optional	Incremental	0
RB	Position of the rotary axes	Optional		1
RS	Order of the rotary axes	Optional	Max. 3 characters	ZYZ
A1	Angle of rotation first rotation	Optional	Degree	0
A2	Angle of rotation second rotation	Optional	Degree	0
A3	Angle of rotation third rotation	Optional	Degree	0

Tab. 7-12: Parameter call

TGT	Effect
1	Imports a raw part.
2	Imports a finished part.
3	Imports a clamping device.
4	Imports a drawing.

Tab. 7-13: Description of parameter TGT - Import target

RB	Effect
1	Fixed in space The rotary axes keep their initial position.
2	consequently The rotary axes rotate synchronously to the model.

Tab. 7-14: Description of parameter RB - Position of the rotary axes

Description

P File name

Meshes and (extension ".stl") as well as two-dimensional drawings (extension ".dxf") can be used as file names.

The extension ".stl" can be omitted. Path names of the local PC can also be used as path names of the MTX file system.

If only one file name is specified without path, it is searched for the file according to the search strategy for subroutines, starting from the main program folder.



- As an alternative to a file, definitions of geometries to be generated generically can be transferred using parameter P. The syntax is shown in [chapter 8 "Configuration" on page 105](#).
- For the import target "Raw part" (TGT1) applies that only a simply interrelated mesh may be imported.

TGT Import target

Determines the type of model use.

For DFX file, only the import targets "Finished part" (TGT2) and "Drawing" (TGT4) are allowed.

X, Y, Z translation

Specifies an offset of the coordinate origin of the imported model in the work-piece coordinate system.

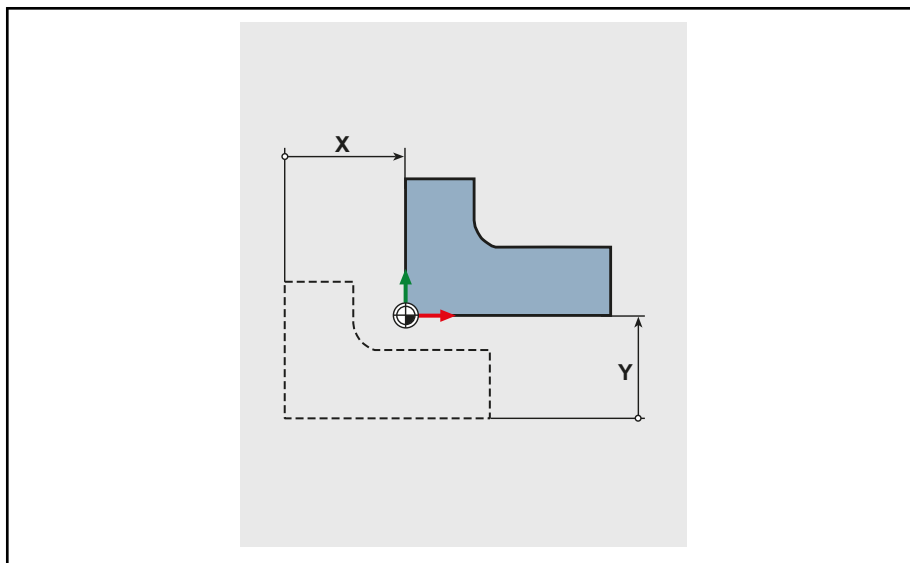


Fig. 7-7: Translation in X and Y. The target position is displayed in color.



The parameter Z is ignored for DXF files with the import target TGT2 (finished part). The applied offset in Z corresponds to the upper side of the workpiece.

RB Position of the rotary axes

Specifies whether subsequent rotations refer to the initial position of the rotary axes.

RS Order of the rotary axes

Specifies the rotary axes to be used subsequently, one after another in each rotation starting with the first character of parameter RS. Only the characters X, Y and Z are permitted.

A1, A2, A3 angle of rotation

Angle of rotation for the first, second and third rotation; corresponding to RS.

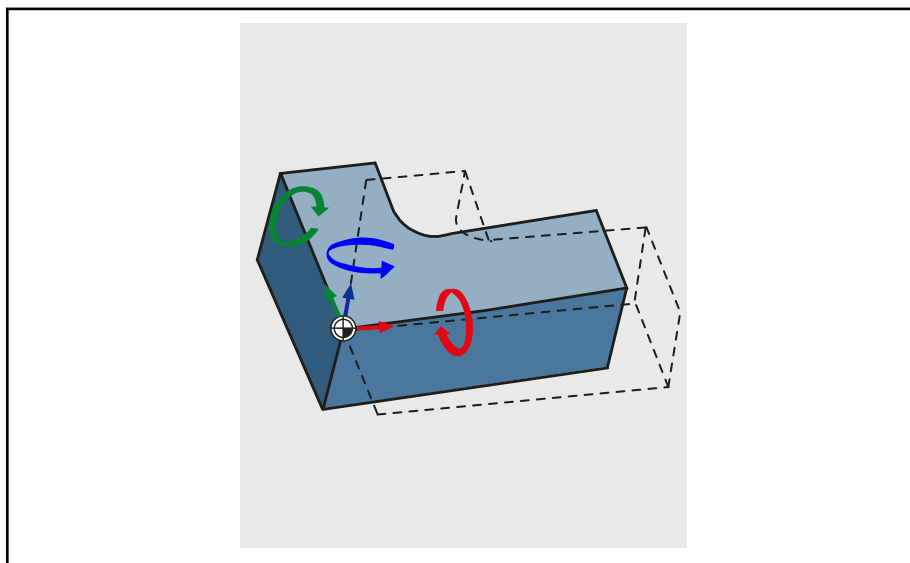


Fig. 7-8:

- Special features**
- The parameters have addresses

- In the workpiece coordinate system, the active placements and the zero offsets in the linear main axes are taken into account. It can be required to "retract" the offsets in rotary axes and telescope axes before the start
- G53 combined with G153 arranges the geometry in the coordinate system of the tool holder. It can deviate from the machine coordinate system. The positions of further axes in the kinematic chain have no effect
- Cannot or not completely be programmed if there is an active axis transformation

Procedure/processing The model is displayed matching the previously activated workpiece zero point. The tool path is deleted upon the import as workpiece.

Example Import the "finalwp.stl" model in the folder of the main program as finished model. One rotation around the axes fixed in space is required : X at 180° and Z at 90°. To correctly position the model with regard to the workpiece zero point (G55), an additional offset by Y-55 is required.

Program:

```
N10 G55 STOCKIMP(P finalwp, TGT2, Y-55, RS XZ, A1 180, A2 90)
```

7.8 STOCK/REMOVE Remove workpieces/models

Syntax STOCK("REMOVE",<TGT>)

or

STOCK()

Brief description Removes a workpiece or another geometry from the selected workpiece holder.

Overview figure

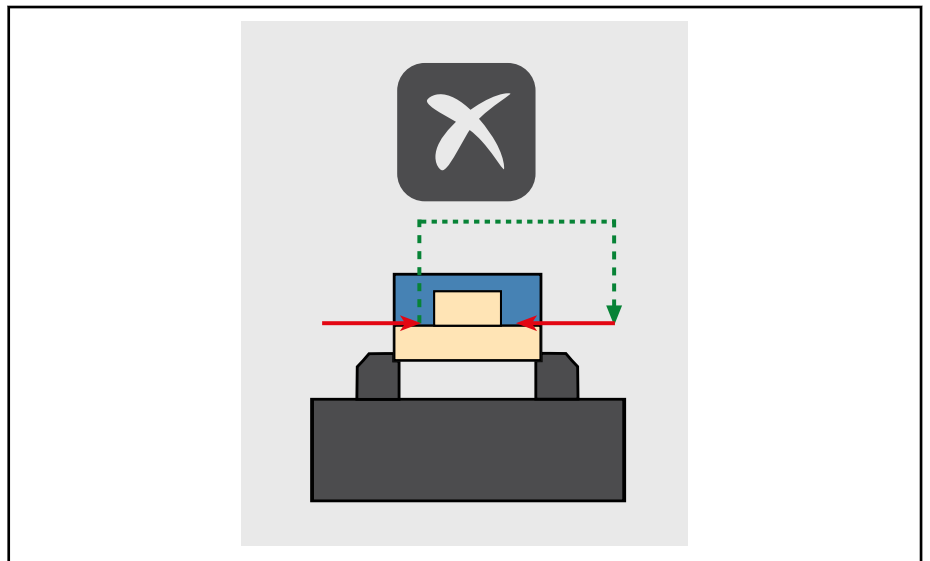


Fig. 7-9: STOCK/REMOVE instruction "Remove workpieces/models"

Parameter

Parameter	Name	Properties	Values	Pre-assignment
TGT	Geometry to be removed	Optional		0

Tab. 7-15: Parameter call

TGT	Effect
0	Complete Removes all geometries of all workpiece holders.
1	Removes a workpiece/raw part and a tool path of the selected workpiece holder.
2	Removes the finished part of the selected workpiece holder.
3	Removes the clamping device of the selected workpiece holder.
4	Removes the drawing of the selected workpiece holder.
5	Removes the tool path of the selected workpiece holder.

Tab. 7-16: Description of parameter TGT - Geometry to be removed

Description	TGT Geometry to be removed The entire parameter list can be omitted if the default value TGT is equal to 0.
Special features	The parameter does not have an address. An automatically loaded workpiece preview (TGT=0 or 2) cannot be deleted. It can only be overwritten by an import or removed using the "Reset raw part" dialog (see "Reset raw part" on page 28 , menu item "Remove incl. machined part, clamping device and drawings").
Procedure/processing	The selected geometry is deleted.
Example	Delete workpiece, tool path and finished part. <i>Program:</i> <pre>N10 STOCK("REMOVE",1) ;removes workpiece and tool path N20 STOCK("REMOVE",2) ;removes finished part</pre>

7.9 STOCK/RENEW Renew workpiece

Syntax	STOCK("RENEW",<TN>)
Brief description	Renews the workpiece of the respective workpiece holder or all workpieces by replacing the workpiece(s) with the last defined raw part geometry. The function also deletes the tool path. It is irrelevant whether the workpiece has previously been machined or removed.

Overview figure

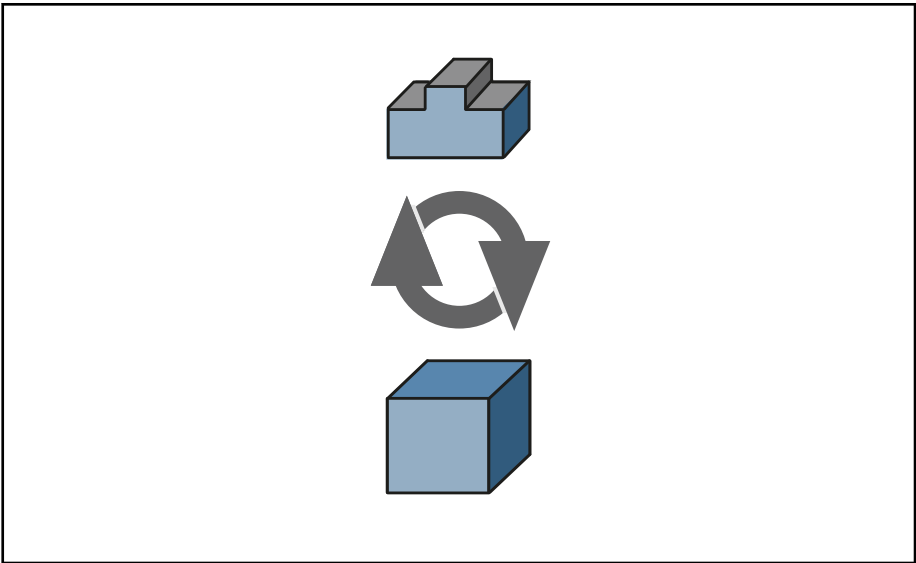


Fig. 7-10: STOCK/RENEW instruction “Renew workpiece”

Parameter

Parameter	Name	Properties	Values	Pre-assignment
TN	Affected work-piece holder	Optional		0

Tab. 7-17: Parameter call

TN	Effect
0	Renews all workpieces.
1	Renews the workpiece at the selected workpiece holder.

Tab. 7-18: Description of parameter TN - affected workpiece holder

Description

TN respective workpiece holder

Specifies whether the renewal refers to the selected workpiece holder or to all workpiece holders.

Special features

The parameter does not have an address.



Generally, the result is the same when a new workpiece is defined in the program every time instead of renewing it. In case of complex, imported geometries, the performance may be higher if you choose renewal.

There is a setting in the simulation options that automatically executes the renewal of the workpiece/workpieces machined by the relevant channel upon program start.

Procedure/processing

The workpiece geometry is restored as defined last.

Example

All workpieces have to be renewed.

Program:

N10 STOCK ("RENEW")

7.10 STOCKSEL Select workpiece holder

Syntax STOCKSEL(<NAM>)

Brief description Selects one or multiple workpiece holder(s) using the name. All subsequent calls of the STOCK and STOCKIMP instructions in the same channel refer to this/these workpiece holder(s). This instruction is required if a tool machine has several workpiece holders or in case of multi-piece clampings.

Overview figure

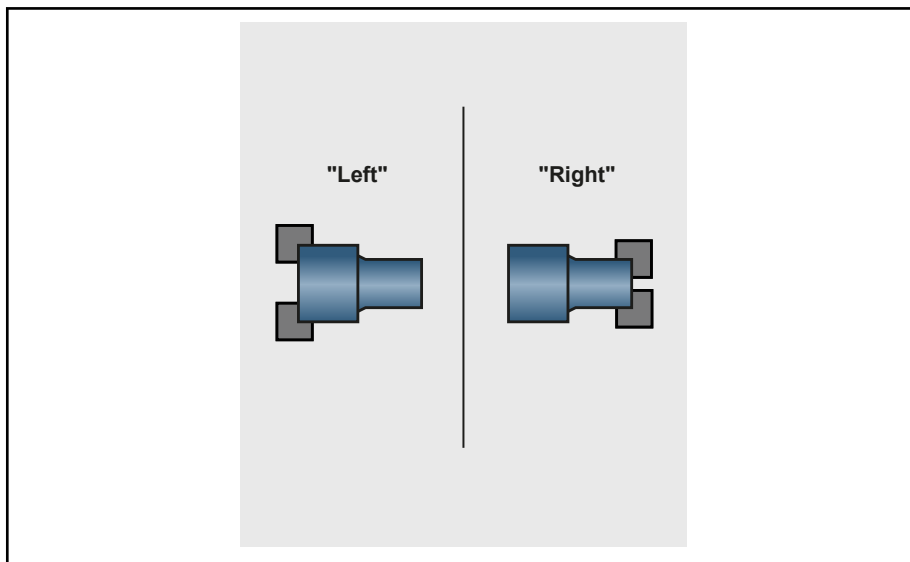


Fig. 7-11: STOCKSEL instruction "Select workpiece holder"

Parameter

Parameter	Name	Properties	Values	Pre-assignment
NAM	Name of workpiece holder	Optional	Max. 32 characters	First workpiece holder of the machine model

Tab. 7-19: Parameter call

Description

NAM name of the workpiece holder

Defines the workpiece holder to be selected.

The name can be composed of wildcards together with the goal that the following workpiece instructions affect multiple workpieces or workpiece holders:

- ? (question mark) labels any character
- * (asterisk) labels none or any number of any characters

Multi-piece clampings with volume models of the workpieces are implemented internally by temporarily created workpiece holders. Their names are composed of the name of the workpiece holder from the machine model, of a # and of an index (≥ 1). The name of the tool holder from the machine model refers to all parts of the multi-piece clamping. If the name ends with a hash and an index, only a specific part of the multi-piece clamping is triggered. The index 0 triggers the holder from the machine model.

Special features The parameter does not have an address.

Procedure/processing The new selection is applied immediately in the current channel until the selection is canceled.

Example Two workpiece holders are available in one machine model, whereas "Left" should be the first one in the machine model. In the "Left" holder, the workpiece is to be deleted, in the "Right" one, a bar section is to be defined.

Program:

```
N10 STOCKSEL()           ; Select "Left"
N20 STOCK("REMOVE",1)    ; Remove workpiece
N30 STOCKSEL("Right")    ; Select "Right"
N40 STOCK("BAR",61,50,, -60); Define bar section
```

7.11 STOCKADD Replace/add workpieces/models

Syntax STOCKADD(<ADD>)

Brief description The following raw part definitions and model imports replace the existing geometry or are added to the existing geometry.

Parameter

Parameter	Name	Properties	Values	Pre-assignment
ADD	Effect of the raw part definitions/imports	Optional		0

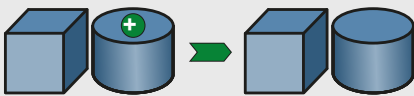
Tab. 7-20: Parameter call

ADD	Effect
0	Replace
1	Add

ADD=0



ADD=1



Tab. 7-21: Description of parameter ADD

Description

ADD effect of raw part definitions/imports

Defines whether the following geometry definitions should eventually replace the geometries at the import targets or whether they should be added.

Special features

- The parameter does not have an address
- Multiple raw parts as volume model and finished part geometries as mesh (e.g. imported from an STL file) cannot be assigned directly to a workpiece holder from the machine model. Thus, when adding tempora-

Program instructions for the simulation

ry workpiece holders, their names end with a hash and an index (1...999). If the workpiece holder selected with STOCKSEL corresponds to a name from the machine model, it is searched for the first currently free holder with an unassigned import target or a new holder is created

- The last call of STOCKADD has no effect if a name with hash and index was programmed at the end in STOCKSEL. In this case, the following geometry definitions are replaced. If there is no temporary workpiece holder, it is created
- Raw parts as enveloping bodies, clamping media and drawings can be added as required. They are always assigned to a workpiece holder from the machine model. If a temporary workpiece holder has been selected, the index is ignored and the geometry is inserted at the holder from the machine model

The parameter does not have an address.

Procedure/processing

The new selection is applied immediately in the current channel until the selection is canceled.

Example 1

Import a clamping device consisting of two parts.

Program:

```

N10 STOCKADD()          ; subsequently results in replacement of existing clamping
                        ; device
N20 STOCKIMP("CLAMP1",3); loads CLAMP1.stl as clamping device
N30 STOCKADD(1)         ; switch to add
N40 STOCKIMP("CLAMP2",3); adds CLAMP2.stl to clamping devices

```

Example 2

Write a program to equip a clamping tower that defines the fixture as well as multiple raw parts (cuboid) and finished parts (final.stl) in different positions for the simulation. The clamping tower with a zero point correlating with the symmetry axis should be at X-300 and Y-200. The clamping tower has four sides. There are two workpieces each in two clamping positions above each other. Use the cycle G230 "Swiveling" to position the workpieces.

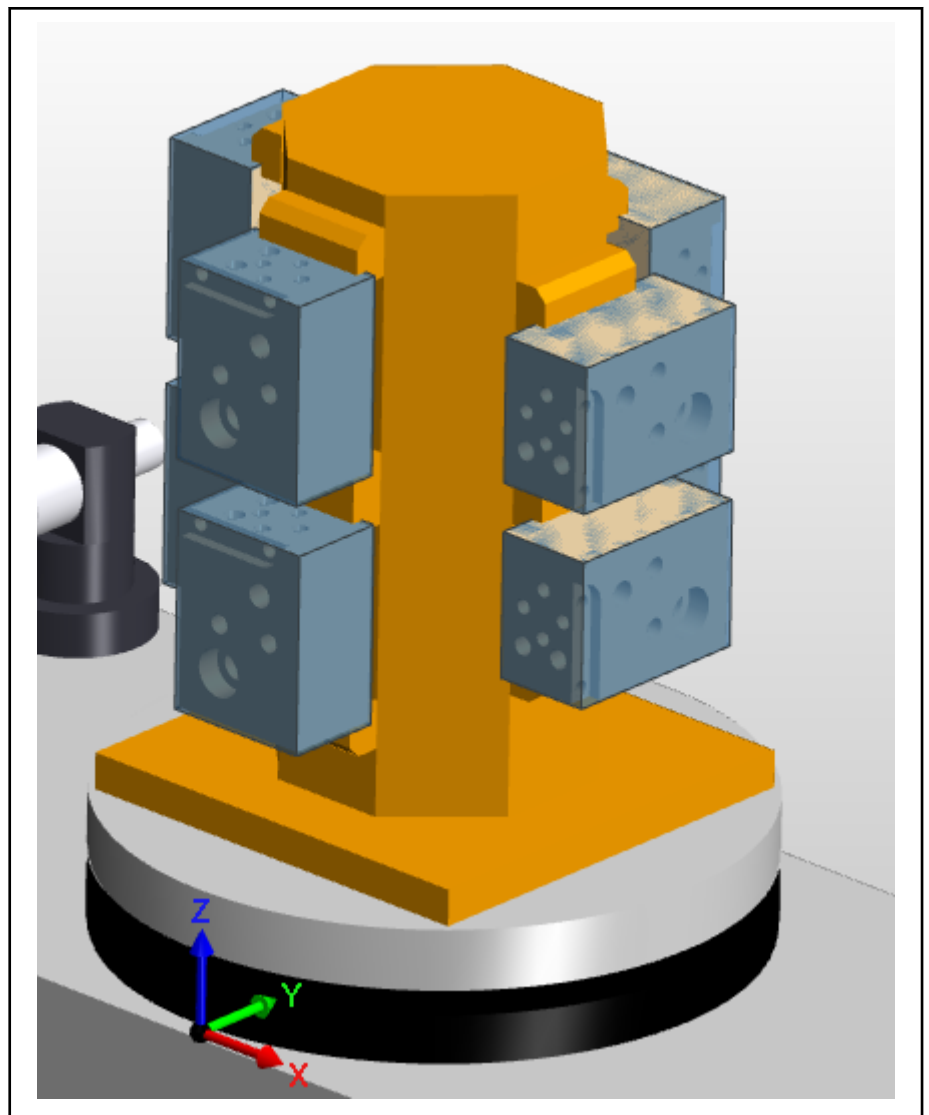


Fig. 7-12: Example of a clamping tower

Program:

```

N100 G230() F1000
N110 G0 X0 Y0 Z0 C0 A0          ; Initial position of the axes
N120 G4 F2
N130 STOCK
N140 STOCKSEL("stock")          ; Delete all geometries
N150 STOCKADD(1)                 ; Select workpiece holder stock
N160 G52.1(X-300, Y-200, Z-460) ; Select "Add"
N170 STOCKIMP(P clamping tower.stl, TGT3); Activate zero point for clamping tower center
                                   ; Import clamping medium
                                   ; Swiveling on side 1, 1. Clamping
N180 G230 (TNR3, OB2, RS "ZX", A2 90, AX-30, AY139, AZ110, OM3, DIR1)
N190 G4 F2
N200 STOCK("BLOCK",62,82,41,1,81,40); Raw part at stock
N210 STOCKIMP(P final.stl, TGT2) ; finished part at stock
N220 G230()
N230 G52.1(X-300, Y-200, Z-460)
                                   ; Swiveling on side 1 1, 2. Clamping
N240 G230 (TNR3, OB2, RS "ZX", A1 0, A2 90, AX-30, AY241, AZ110, OM3)
N250 STOCKIMP(P final.stl, TGT2) ; Raw part at stock#1
N260 STOCK("BLOCK",62,82,41,1,81,40); Finished part at stock#1
N270 G230()
N280 G52.1(X-300, Y-200, Z-460)
                                   ; Swiveling on side 2, 1. Clamping
N290 G230 (TNR3, OB2, RS "ZX", A1 90, A2 90, AX-30, AY241, AZ110, OM3)
N300 G4 F2

```

Program instructions for the simulation

```
N310 STOCK("BLOCK",62,82,41,1,81,40); Raw part at stock#2
N320 STOCKIMP(P final.stl, TGT2) ; Finished part at stock#2
N330 G230()
N340 G52.1(X-300, Y-200, Z-460)
; Swiveling on side 2, 2. Clamping
N350 G230 (TNR3, OB2, RS "ZX", A1 90, A2 90, AX-30, AY140, AZ110, OM3)
N360 STOCKIMP(P final.stl, TGT2) ; Raw part at stock#3
N370 STOCK("BLOCK",62,82,41,1,81,40); Finished part at stock#3
N380 G230()
N390 G52.1(X-300, Y-200, Z-460)
; Swiveling on side 3, 1. Clamping
N400 G230 (TNR3, OB2, RS "Y", A1 -90, AY40, AX80, AZ110, OM3)
N410 G4 F2
N420 STOCK("BLOCK",60,82,41,,81,40) ; Raw part at stock#4
N430 STOCKIMP(P final.stl, TGT2) ; Finished part at stock#4
N440 G230()
N450 G52.1(X-300, Y-200, Z-460)
; Swiveling on side 3, 2. Clamping
N460 G230 (TNR3, OB2, RS "Y", A1 -90, AY40, AX160, AZ110, OM3)
N470 STOCKIMP(P final.stl, TGT2) ; Raw part at stock#5
N480 STOCK("BLOCK",60,82,41,,81,40) ; Finished part at stock#5
N490 G230()
N500 G52.1(X-300, Y-200, Z-460)
; Swiveling on side 4, 1. Clamping
N510 G230 (TNR3, OB2, RS "XZ", A1 -90, A2 270, AY40, AX80, AZ110, OM3)
N520 G4 F2
N530 STOCK("BLOCK",60,82,41,,81,40) ; Raw part at stock#6
N540 STOCKIMP(P final.stl, TGT2) ; Finished part at stock#6
N550 G230()
N560 G52.1(X-300, Y-200, Z-460)
; Swiveling on side 4, 2. Clamping
N570 G230 (TNR3, OB2, RS "XZ", A1 -90, A2 270, AY40, AX160, AZ110, OM3)
N580 STOCKIMP(P KKB final.stl, TGT2); Raw part at stock#7
N590 STOCK("BLOCK",60,82,41,,81,40) ; Finished part at #7
N600 G4 F2
N610 G0 A0 C0 G230() ; Initial position
```

Each workpiece added to N250 creates a new temporary workpiece holder. When importing the finished part geometry, it is detected that this geometry target is empty since the latest created holder and that it is applied to it. In this example, raw part definition and finished part definition can be interchanged.

7.12 TOOL/DRILL Drill, countersink, reamer

Syntax	TOOL("DRILL",<D>,<A>,<H>)
Brief description	Generates the graphic for a twist drill, tapping device, reamer, face or flat countersink (without spigot) irrespective of the changed tool in the simulation.
Overview figure	



Fig. 7-13: TOOL/DRILL tool description "Drill, countersink, reamer"

Parameter

Parameter	Name	Properties	Values	Pre-assignment
D	Diameter	Mandatory	>0	
A	Apical angle	Optional	0<A<=180 degree	118
H	Length	Mandatory	>0	

Tab. 7-22: Parameter call

Description

D Diameter

Determines the drilling diameter.

A Apical angle

Defines the taper angle at the tool tip.

H Length

Defines the length of the cutting edge part.

Special features

The parameters do not have any address.

The generation of the tool graphic is based on the modal states of the NC at the time of the cycle processing. Thus, it is mandatory to activate the tool corrections and the correct interpolation plane **before** calling the cycle. The tool tip is positioned at the tool zero point.

The orientation of the longitudinal tool axis corresponds to the position of the drilling axis of the interpolation level. In case of a negative direction of length correction in the axis running parallelly to the longitudinal tool axis (e.g. using L3 G47 (, , -Z)), the simulation will turn the tool orientation over.

Procedure/processing

The tool description superimposes the simulation of the standard tool change via the tool change subroutine and generates a new graphic at the tool node that is assigned to the calling channel. The generation does not depend on the last tool.

The tool graphic is remains until it is revoked with the cycle call without parameter. (TOOL or TOOL ()).

Example

A tool description is required for a twist drill with a diameter of 10 mm.

Program:

```

N10 G17 G47 D1 S800 F300 M13 ; Interpolation level and tool correction
N20 TOOL("DRILL",10,,60) ; Description twist drill D10
N30 G81 (SL2, DT-50, RL100) ; Machining cycle
N40 G0 X25 Y30 ; Approaching and drilling
N50 G80 M5 ; Deselect modal cycle, spindle stop
N60 TOOL ; Deactivate tool description

```

7.13 TOOL/CDRILL Step drill, centering drill

Syntax TOOL("CDRILL",<D1>,<A1>,<L>,<D2>,<A2>,<H>)

Brief description Generates the graphic for a subland twist drill or centering drill irrespective of the changed tool in the simulation.

Overview figure

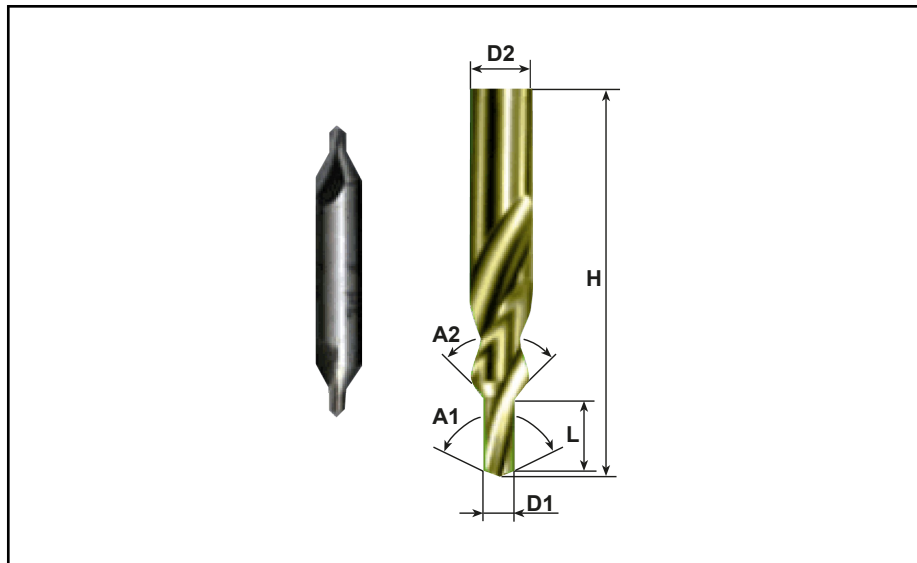


Fig. 7-14: TOOL/CDRILL tool description "Step drill, centering drill"

Parameter

Parameter	Name	Properties	Values	Pre-assignment
D1	Small diameter	Mandatory	>0	
A1	Apical angle	Optional	$0 < A \leq 180$ degree	118
L	Step length	Mandatory	<>0	
D2	Large diameter	Mandatory	$D2 > D1$	
A2	Countersink angle	Mandatory	$0 < A \leq 180$ degree	
H	Total length	Mandatory	>0	

Tab. 7-23: Parameter call

Description

D1 Small diameter

Determines the drilling diameter.

A1 Apical angle

Defines the taper angle at the tool tip.

L Step length

Length of the cylindrical part with the small diameter.

D2 Large diameter

Determines the countersink diameter.

A2 Countersink angle

Defines the taper angle at the tool tip.

H Total length

Defines the length of the cutting edge part.

Special features

The parameters do not have any address.

The generation of the tool graphic is based on the modal states of the NC at the time of the cycle processing. Thus, it is mandatory to activate the tool corrections and the correct interpolation plane **before** calling the cycle. The tool

tip is positioned at the tool zero point if the "L" parameter is positive. If the "L" parameter is negative, the tool zero point is positioned at the complete countersink diameter.

The orientation of the longitudinal tool axis corresponds to the position of the drilling axis of the interpolation level. In case of a negative direction of length correction in the axis running parallel to the longitudinal tool axis (e.g. using L3 G47 (, , -Z)), the simulation will turn the tool orientation over.

Procedure/processing

The tool description superimposes the simulation of the standard tool change via the tool change subroutine and generates a new graphic at the tool node that is assigned to the calling channel. The generation does not depend on the last tool.

The tool graphic is remains until it is revoked with the cycle call without parameter. (TOOL or TOOL ()).

Example

Use a centering drill with a diameter of 2.5 mm for a centering with a depth of 9 mm. For this purpose, its use length of 10 mm is sufficient. It is a centric bore at a lathe. Usually, plane G18 is activated. Thus, it is programmed in the simulation configuration that the orientation plane G17 is set for drilling tools and for the active plane G18.

Program:

```
N10 G18 DIA G47 D2 S2500 F150 M13 ; Interpolation level and tool correction
N20 TOOL("CDRILL",2.5,,4,8,60,10) ; Description centering drill D2.5
N30 G81 (IX Z, SL2, DT-9) ; Machining cycle
N40 G0 X0 Z10 ; Approaching and drilling
N50 G80 X200 Z100 M5
N60 TOOL ; Deactivate tool description
```

7.14 TOOL/BBAR Boring bar, backward counterbore

Syntax TOOL("BBAR",<D1>,<W>,<H>,<L>)

Brief description Generates the graphic for a boring bar (also referred to as reaming rod) or for a backward counterbore irrespective of the changed tool in the simulation. Both types are boring tools with one cutting edge in which the tool rotates.

Overview figure

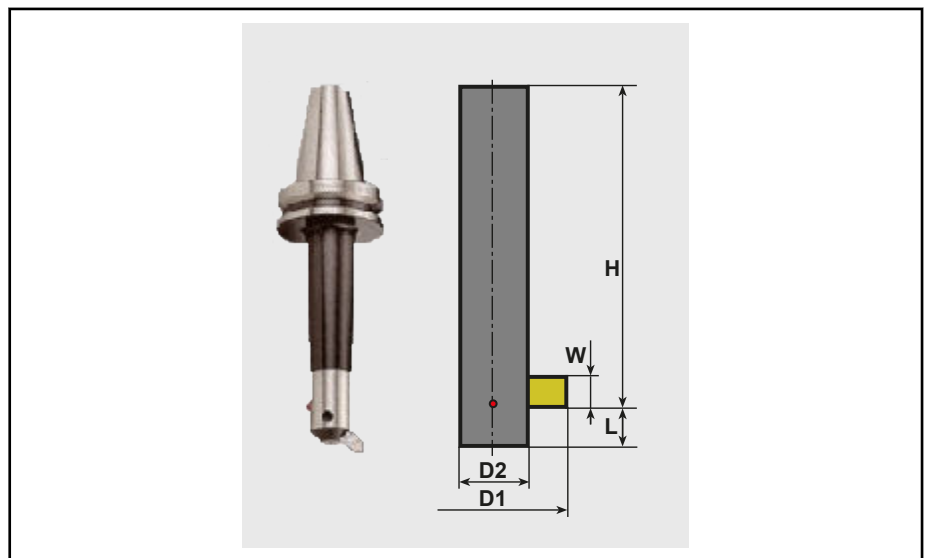


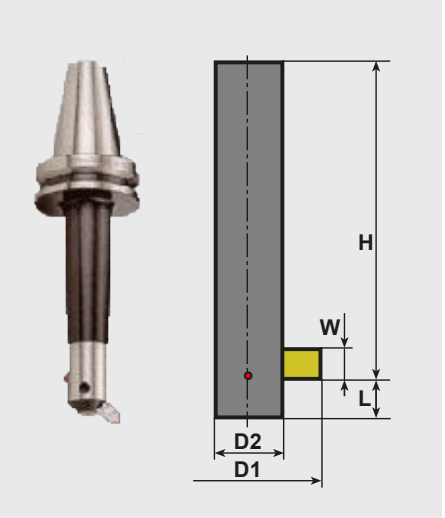
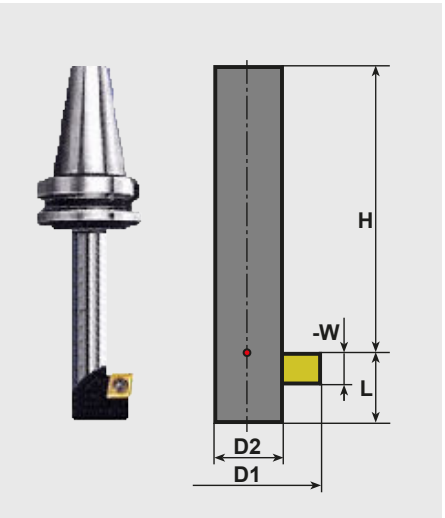
Fig. 7-15: TOOL/BBAR tool description "Boring bar"

Program instructions for the simulation

Parameter

Parameter	Name	Properties	Values	Pre-assignment
D1	Diameter	Mandatory	>D2	
W	Cutting edge width	Mandatory		
D2	Shaft diameter	Mandatory	>0	
H	Shaft length	Mandatory	>0	
L	Loss length	Optional	≥ 0	0 if $W > 0$, -W if $W < 0$

Tab. 7-24: Parameter call

W	Effect
>0	Boring bar 
<0	Backward counterbore 

Red point Position of the tool zero point

Tab. 7-25: Description of the parameter W

Description**D1 Diameter**

Determines the drilling diameter.

W Cutting edge width

Defines the width of the chisel and uses the sign to define whether it is a boring bar or a backward counterbore.

D2 Shaft diameter

Determines the diameter of the shaft.

H Shaft length

Defines the useful length of the tool.

L Loss length

Defines a protruding spigot.

Special features

The use of boring bars and backward counterbores is limited in the simulation as no collision-free, eccentric retraction and extension with oriented spindle (command Mx19) can be simulated.

The parameters do not have any address.

The generation of the tool graphic is based on the modal states of the NC at the time of the cycle processing. Thus, it is mandatory to activate the tool corrections and the correct interpolation plane **before** calling the cycle.

The orientation of the longitudinal tool axis corresponds to the position of the drilling axis of the interpolation level. In case of a negative direction of length correction in the axis running parallel to the longitudinal tool axis (e.g. using L3 G47 (, , -Z)), the simulation will turn the tool orientation over.

Procedure/processing

The tool description superimposes the simulation of the standard tool change via the tool change subroutine and generates a new graphic at the tool node that is assigned to the calling channel. The generation does not depend on the last tool.

The tool graphic is remains until it is revoked with the cycle call without parameter. (TOOL or TOOL ()).

Example

At a housing, a pre-processed bore (30 mm deep) is to be opened by boring using a boring bar with a diameter of 40 (without loss length). Afterwards, make a backward counterbore with a diameter of 44 mm (26 mm deep). The backward counterbore has a loss length of 25 mm and a shaft diameter of 32 mm, which allows for a radial move-out of 3 mm (1 mm radial safety distance).

Program:

```

N10 G17 G47 G54 D3 S1100 F300 M13 ; Interpolation level and tool correction
N20 TOOL("BBAR",40,3,30,100)      ; Description boring bar D40
N30 G86 (SL2, DT-32, SX X)        ; Machining Cycle boring
N40 G0 X57.5 Y0 Z2                ; Approaching and boring
N50 G48 G53 G80 Y-250 Z500 M5     ; Leaving and spindle stop
N60 M0                            ; Stop for manual tool change
N70 G17 G47 G54 D4 S750 F210 M13 ; Interpolation level and tool correction
N80 TOOL("BBAR",44,-3,32,100,25) ; Description backward counterbore D44
N90 G87 (SL-32, DT-26, DW0.5, SX X, RD3) ; Machining cycle back boring
N100 G0 X57.5 Y0 Z27              ; Approaching and countersinking
N110 G48 G53 G80 Y-250 Z500 M5    ; Leaving and spindle stop
N120 TOOL                        ; Deactivate tool description

```

7.15 TOOL/MILL Cylindrical, angular and ball cutter

Syntax TOOL("MILL",<D>,<R>,<H>,<A>)

- Brief description**
- Generates the graphic for a milling tool of the following types irrespective of the changed tool in the simulation:
- End milling cutter, groove milling cutter,
 - Plain and shell end milling cutters,
 - Dove tail milling cutter,
 - Angle milling cutter (with a defined small diameter),
 - Side milling cutters and circular saw blades,
 - T-groove milling cutters,
 - Die-sinking cutters,
 - Ball and ball-end cutters,
 - Copy and square shoulder milling head.

Overview figure

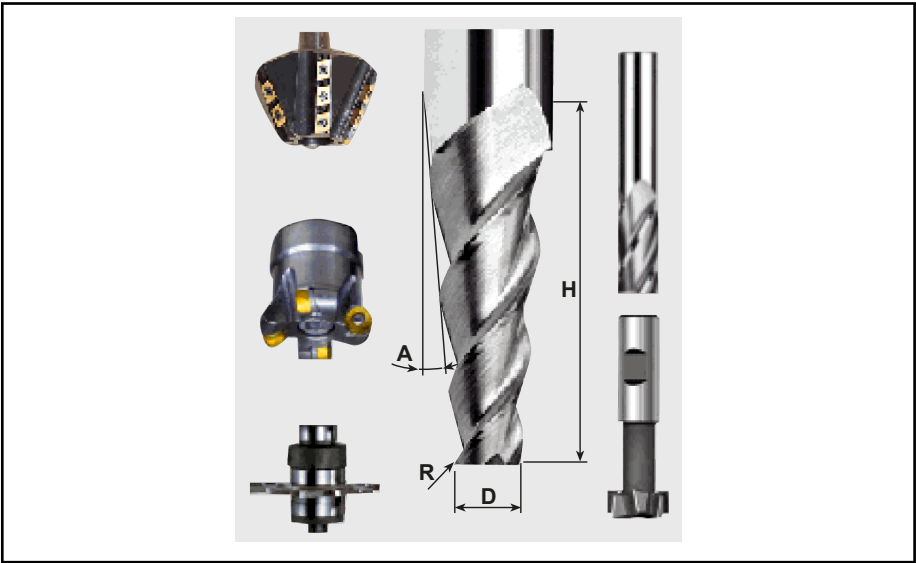


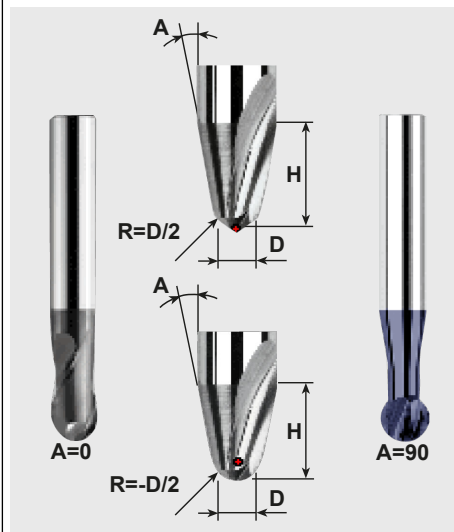
Fig. 7-16: TOOL/MILL tool description "Cylindrical or bevel cutter with optional corner radius"

Parameter

Parameter	Name	Properties	Values	Pre-assign-ment
D	Diameter	Optional	>0	Double current tool radius cor- rection
R	Corner/ball ra- dius	Optional		0
H	Length	Mandatory	>0	
A	Taper angle	Optional	-90<A<=90	0

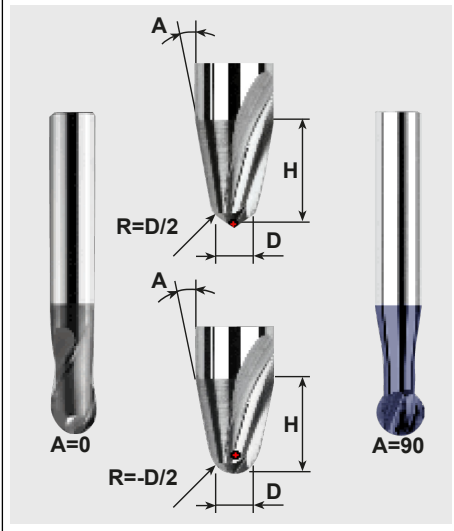
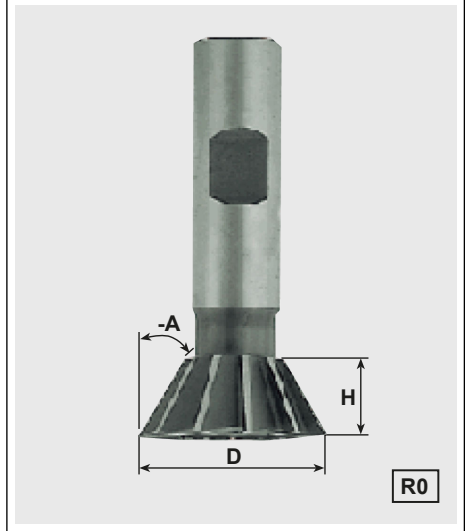
Tab. 7-26: Parameter call

R	Effect
0	No corner radius. With $H \leq D$ side milling cutter, saw blade or T-groove milling cutter.
$0 < R < D/2$	Corner radius
$D/2$	Ball cutter zeroed at the tool tip

R	Effect
-D/2	Ball cutter zeroed at ball center
	

Red point Position of the tool zero point

Tab. 7-27: Description of parameter R

A	Effect
0	Cylindrical milling cutter
0<A<90	Angular milling cutter
90	Ball-end cutter
-90<A<0	Dove tail milling cutter. Parameter R is ignored.
 	

Tab. 7-28: Description of parameter A

Description

D Diameter

Determines the diameter at the front.

R Ball/corner radius

	Defines the corner radius and uses the sign to define whether a ball cutter is zeroed in the ball center.
	H Length
	Determines the length of the cutting edge.
	A Taper angle
	Defines the taper angle defines whether it is a ball-end cutter or dove tail milling cutter.
Special features	The parameters do not have any address. The generation of the tool graphic is based on the modal states of the NC at the time of the cycle processing. Thus, it is mandatory to activate the tool corrections and the correct interpolation plane before calling the cycle. Depending on the sign of parameter "R", the tool zero point is positioned at the tool tip or in the ball center. The orientation of the longitudinal tool axis corresponds to the position of the drilling axis of the interpolation level. In case of a negative direction of length correction in the axis running parallelly to the longitudinal tool axis (e.g. using L3 G47 (, , -Z)), the simulation will turn the tool orientation over. If H is smaller or equal to D in a cylindrical tool. the shaft is displayed at 25% of diameter D. The slimmer shaft corresponds to the shaft of a side milling cutter or a T-slot cutter. Pivots protruding at the front or tightening nuts cannot be mapped using this form of description.
Procedure/processing	The tool description superimposes the simulation of the standard tool change via the tool change subroutine and generates a new graphic at the tool node that is assigned to the calling channel. The generation does not depend on the last tool. The tool graphic is remains until it is revoked with the cycle call without parameter. (TOOL or TOOL ()).
Example	Create a rectangular pocket using a groove milling cutter with a diameter of 6 mm and a use length of 20 mm. The D-correction includes the current milling cutter radius. Thus, the diameter D can be omitted in the tool description. <i>Program:</i>

```

N10 G17 G47 D5 S1800 F420 M13 ; Interpolation level and tool correction
N20 TOOL("MILL",,,20)        ; Description groove milling cutter D6
                               ; Machining cycle rectangular pocket
N30 G271 (LE28, WI20, R4, UL0, DT-5, CD3, RSW0.1, RSL0.1, FF280)
N40 G0 X0 Y0 Z2               ; Approaching and milling
N50 G80 Z500 M5               ; Leaving and spindle stop
N60 TOOL                      ; Deactivate tool description

```

7.16 TOOL/CHAMFER Engraving milling cutter, chamfer milling cutter, face milling head

Syntax	TOOL("CHAMFER",<D>,<A>,<H>,<L>)
Brief description	Generates the graphic for a milling tool with chamfer or tip irrespective of the changed tool in the simulation.

Overview figure

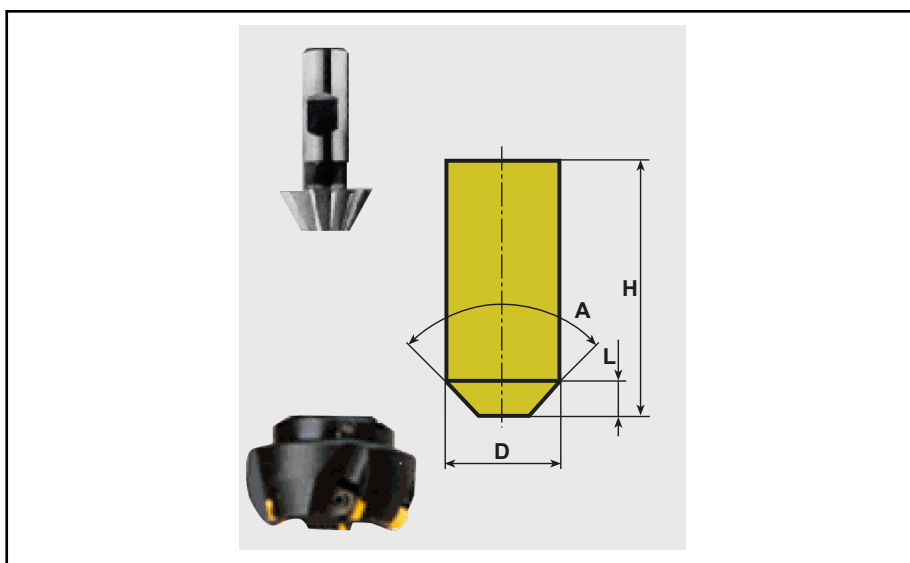


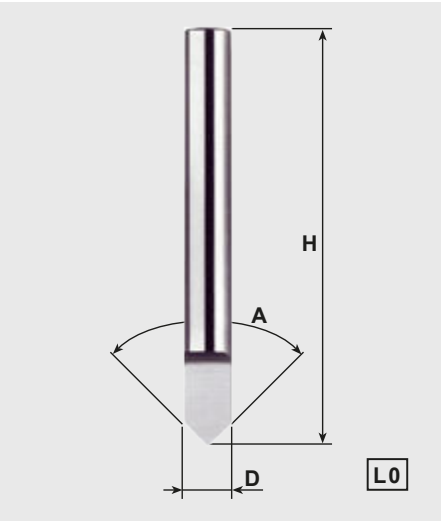
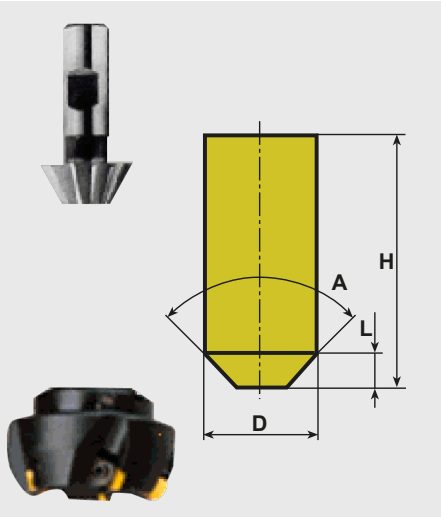
Fig. 7-17: TOOL/CHAMFER tool description "Milling cutter with chamfer"

Parameter

Parameter	Name	Properties	Values	Pre-assignment
D	Diameter	Mandatory if $L=0$, otherwise optional	>0	With $L>0$, calculated from A, L and current tool radius correction.
A	Taper angle	Optional	$0 < A < 180$	90
H	Length	Mandatory	>0	
L	Chamfer length	Optional	≥ 0	0

Tab. 7-29: Parameter call

Program instructions for the simulation

L	Effect
0	Pointed tool 
>0	Tool with chamfer 

Tab. 7-30: Description of parameter L

Description	D Diameter Defines the outer diameter. A Taper angle Defines the angle of the tip or the chamfer. H Length Determines the length of the cutting edge. L Chamfer length Defines the size of the chamfer or whether it is a pointed tool.
Special features	The parameters do not have any address.

The generation of the tool graphic is based on the modal states of the NC at the time of the cycle processing. Thus, it is mandatory to activate the tool corrections and the correct interpolation plane **before** calling the cycle.

The orientation of the longitudinal tool axis corresponds to the position of the drilling axis of the interpolation level. In case of a negative direction of length correction in the axis running parallel to the longitudinal tool axis (e.g. using L3 G47 (, , -Z)), the simulation will turn the tool orientation over.

Procedure/processing

The tool description superimposes the simulation of the standard tool change via the tool change subroutine and generates a new graphic at the tool node that is assigned to the calling channel. The generation does not depend on the last tool.

The tool graphic is remains until it is revoked with the cycle call without parameter. (TOOL or TOOL()).

Example

A rectangular area has to be machined using a face milling head with a diameter of 125 mm. The adjustment angle of the cutting edge is 45°, its height is 5.5 mm. The D-correction includes the current milling cutter radius at the front. The cycle can use it to automatically determine the outer diameter.

Program:

```
N10 G17 G47 D6 S740 F650 M13 ; Interpolation level and tool correction
N20 TOOL("CHAMFER",,,63,5.5) ; Description face milling head D125
N30 G266 (LE320, WI240, UL2, DT0) ; Machining cycle rectangular area
                                ; Approaching and milling
N40 G0 X100 Y0                  ; Leaving and spindle stop
N50 G80 Z200 M5
N60 TOOL                        ; Deactivate tool description
```

7.17 TOOL/CRMILL Quarter round profile milling cutter

Syntax TOOL("CRMILL",<D1>,<D2>,<R>,<H>,<L>)

Brief description Generates the graphic for a profile cutter with concave rounding irrespective of the changed tool in the simulation.

Overview figure

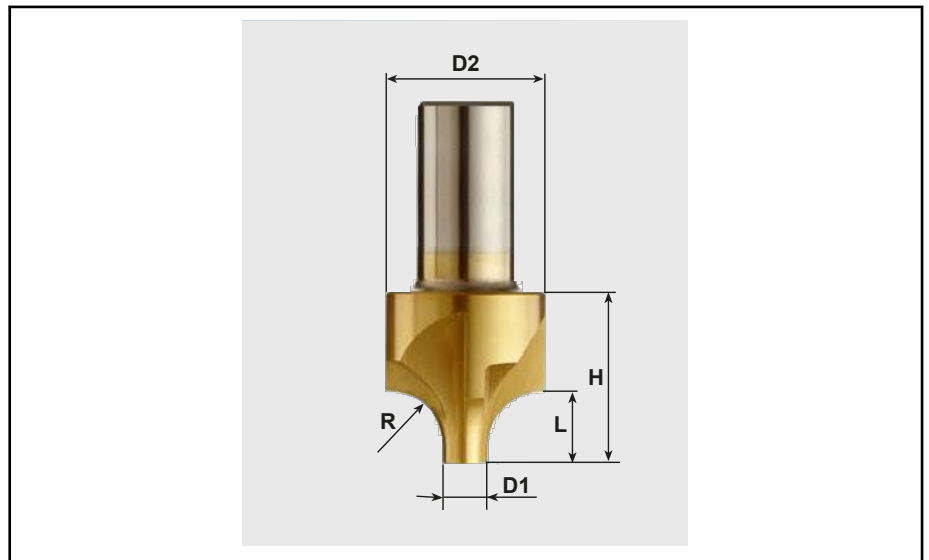


Fig. 7-18: TOOL/CRMILL tool description "Quarter round profile milling cutter"

Parameter

Parameter	Name	Properties	Values	Pre-assignment
D1	Small diameter	Mandatory	>0	

Program instructions for the simulation

D2	Large diameter	Mandatory	>D1	
R	Profile radius	Optional	>=0	Minimum from L and (D2 - D1) / 2
H	Total length	Mandatory	>L	
L	Length, profile	Mandatory	>0	

Tab. 7-31: Parameter call

Description**D1 Small diameter**

Determines the smaller diameter at the front.

D2 Large diameter

Defines the outer diameter.

R Profile radius

Defines the radius of the quarter round profile.

H Total length

Determines the length of the cutting edge.

L Length profile

Defines the axial dimension of the profile.

Special features

The parameters do not have any address.

The generation of the tool graphic is based on the modal states of the NC at the time of the cycle processing. Thus, it is mandatory to activate the tool corrections and the correct interpolation plane **before** calling the cycle.

The orientation of the longitudinal tool axis corresponds to the position of the drilling axis of the interpolation plane. In case of a negative direction of length correction in the axis running parallelly to the longitudinal tool axis (e.g. using L3 G47 (, , -Z)), the simulation will turn the tool orientation over.

Procedure/processing

The tool description superimposes the simulation of the standard tool change via the tool change subroutine and generates a new graphic at the tool node that is assigned to the calling channel. The generation does not depend on the last tool.

The tool graphic is remains until it is revoked with the cycle call without parameter. (TOOL or TOOL()).

Example

The opening of a bore with a diameter of 40 mm (level of the upper side Z0) has to be provided with a quarter round profile with a radius of 6 mm. The other dimensions of the tool are D1=8, D2=20, H=12 and L=6. The milling cutter radius correction value is 4. Consequently, a circle with a radius of 20 can be traveled if the milling cutter radius correction is active.

Program:

```

N10 G17 G47 D7 S2000 F400 M13 ; Interpolation level and tool correction
N20 TOOL("CRMILL",8,20,6,12,6); Description profile cutter
N30 G0 G41 X-4 Y0 ; Starting point run-in cycle
N40 Z-6 ; Infeed deeper by L=6
N50 G3 X20 I12 J0 ; Run-in circle
N60 I-20 J0 ; Milling profile at diameter 40
N70 X8 Y12 I-12 ; Run-out cycle
N80 G0 Z100 M5 ; Retract and spindle stop
N90 G40 (NOM) TOOL ; Tool description and deactivating FRK

```

7.18 TOOL/GRIND Grinding wheel

Syntax TOOL("GRIND",<TY>,<P>,<D1>,<D2>,<W>,<A>,<H1>,<R1>,<CS1>,<CA1>,<AX1>,<AZ1>,<A1>,<H2>,<R2>,<CS2>,<CA2>,<AX2>,<AZ2>,<A2>)

Brief description Generates the graphic for the layer of a grinding wheel irrespective of the changed tool in the simulation. Parameters can be used to determine the profile (left and right one, if applicable).

Overview figure

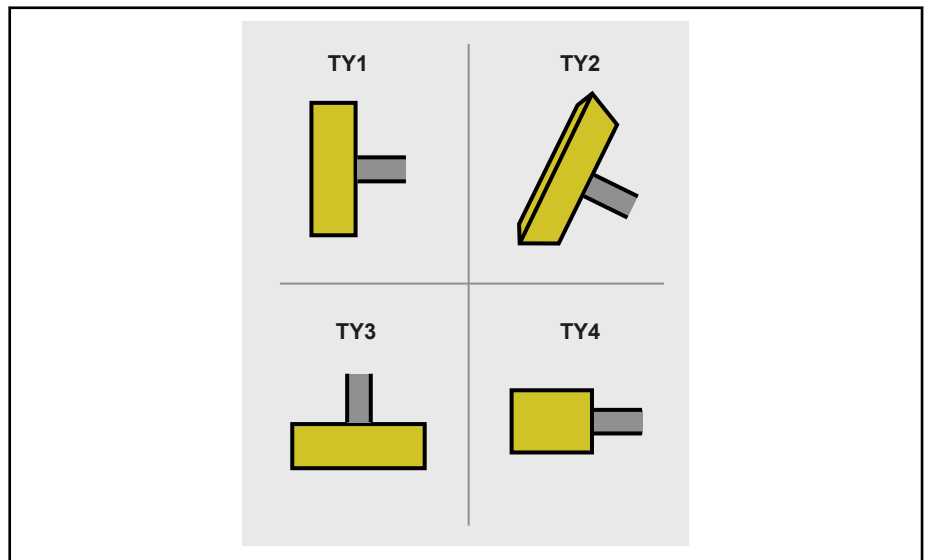
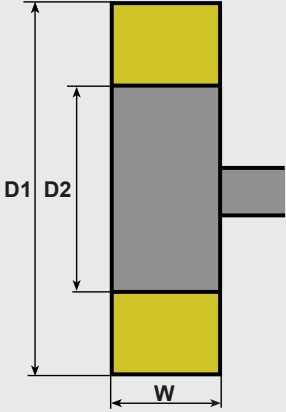
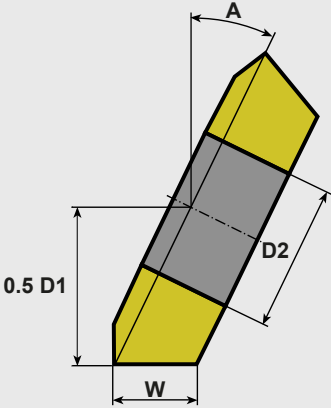


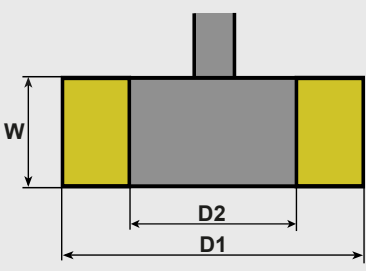
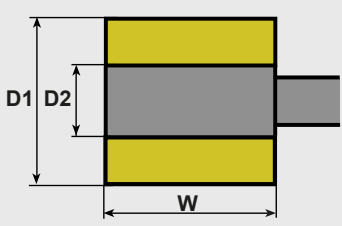
Fig. 7-19: TOOL/GRIND tool description "Grinding wheel"

Geometry parameters **Geometry parameters**

Parameter	Name	Properties	Values	Pre-assignment
TY	Wheel type	Optional		1
P	Profile number	Optional		for TY1: 2, for TY2: 201, for TY3: 301, for TY4: 401,
D1	Outer diameter	Optional	<>0	Double length correction value in Z or X
D2	Inner diameter, layer	Optional	>0	Complete grinding body
W	Wheel width	Mandatory	>0	
A	Angle of inclination	Mandatory for TY2	0<A<90	

Tab. 7-32: Call of the parameters defining the geometry

TY	Effect
1	Straight wheel 
2	Inclined wheel 

TY	Effect
3	Face plate 
4	Internal wheel 

Tab. 7-33: Description of parameter TY and specification of parameters D1, D2, W and A

Description of the geometry parameters

TY wheel type

The wheel type determines the position of the rotary axes.

P Profile number

Defines the wheel form and the possible parameters to define the dimensions of the profile and/or the left and right profile.

D1 Outer diameter

Diameter measured parallelly to the axis. If the opposite point is programmed as tool point compared to the following figures regarding the wheel forms, specify it as negative point.

D2 Inner diameter, layer

Defines the inner diameter of the grinding layer.

W Wheel width

Defines the width of the wheel.

A Angle of inclination

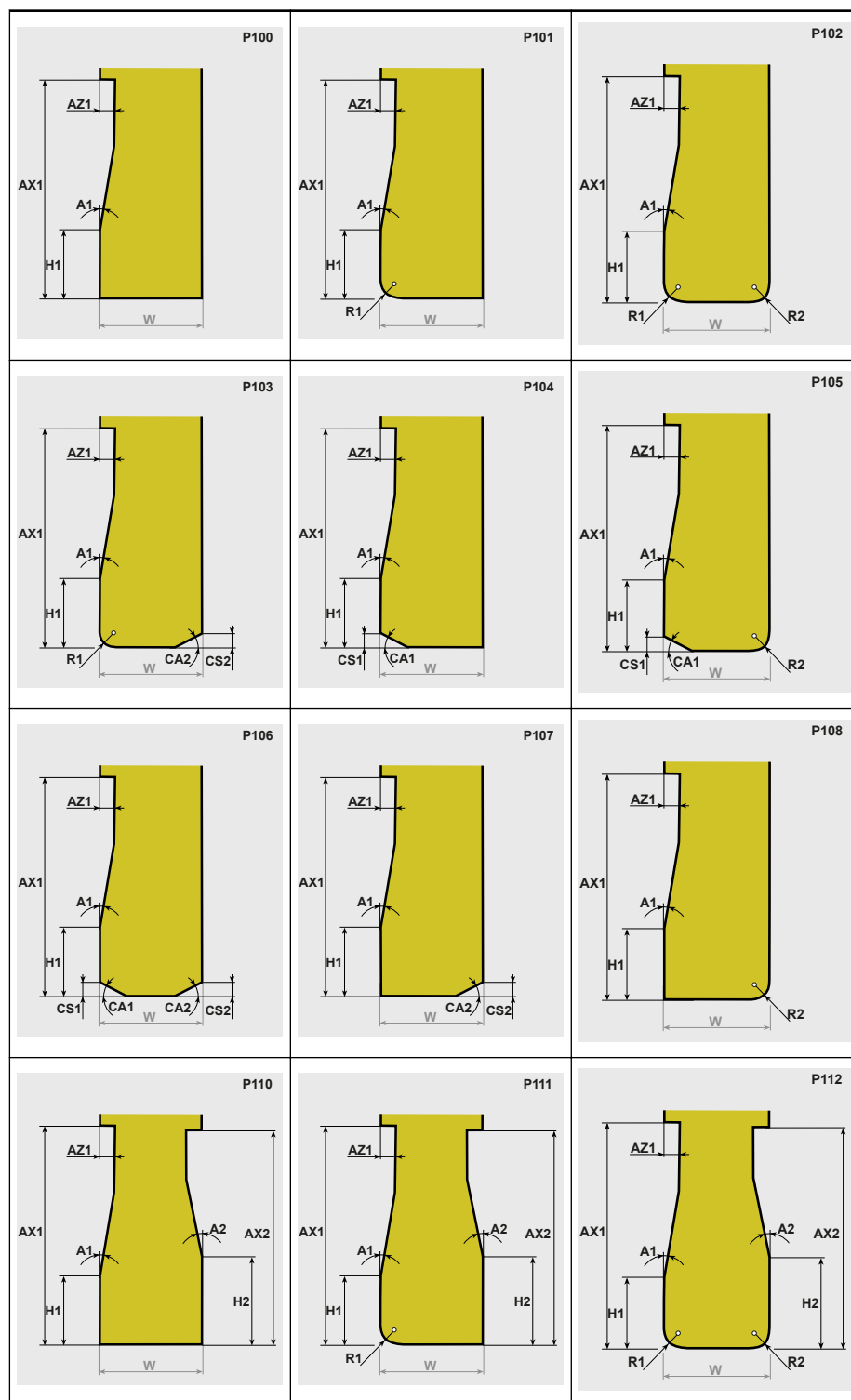
Defines the inclination of an inclined wheel.

Wheel forms for straight wheels TY1 part 1

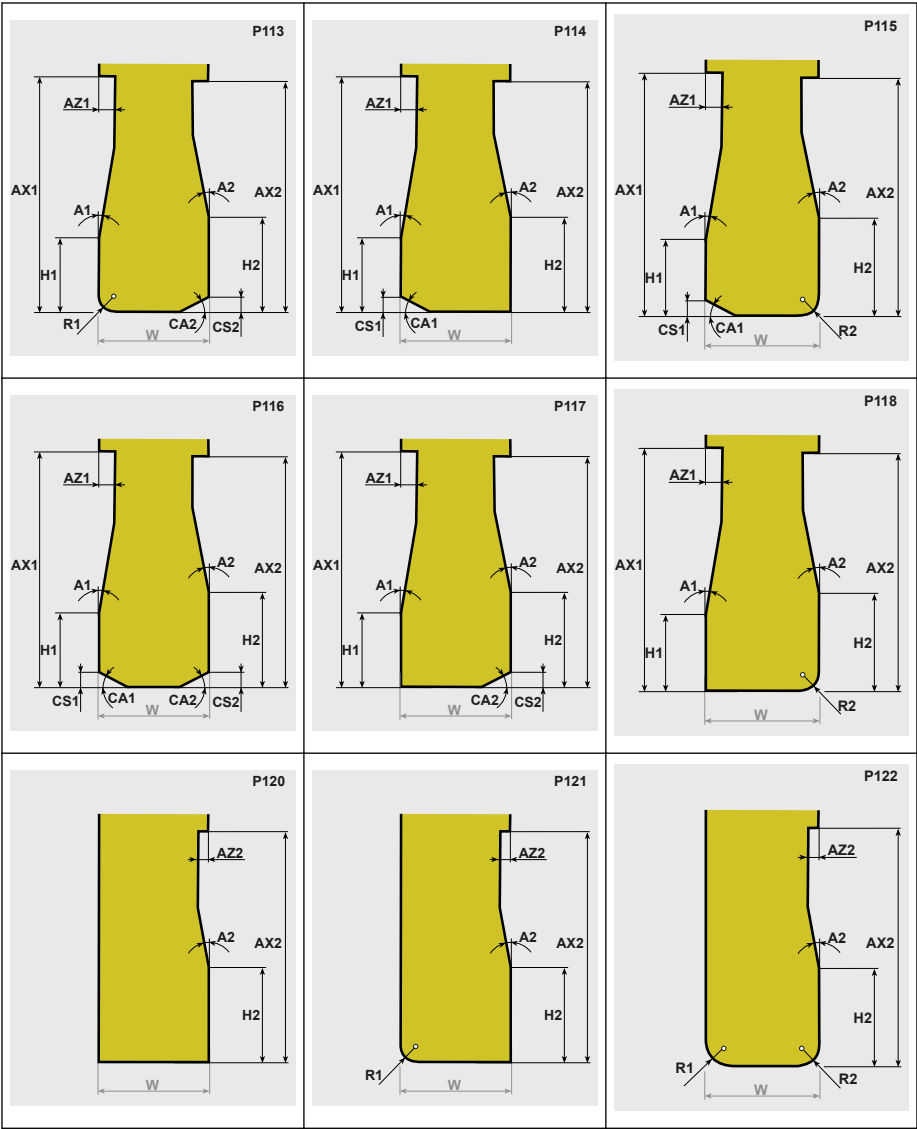


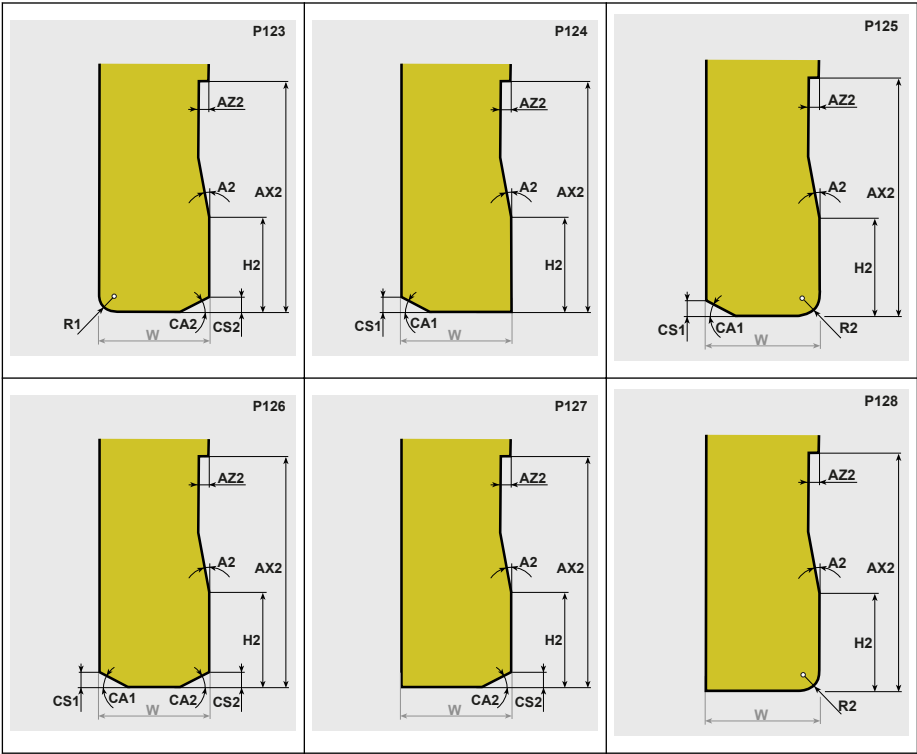
Tab. 7-34: Wheel forms for profile number P from 1 to 25

Wheel forms for straight wheels TY1 part 2



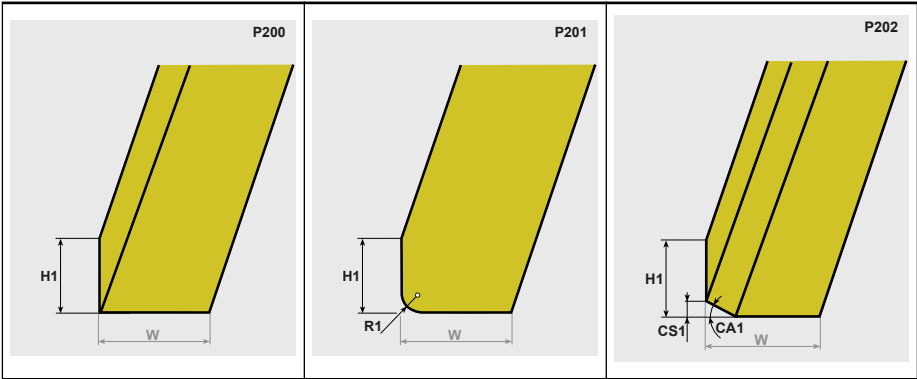
Program instructions for the simulation





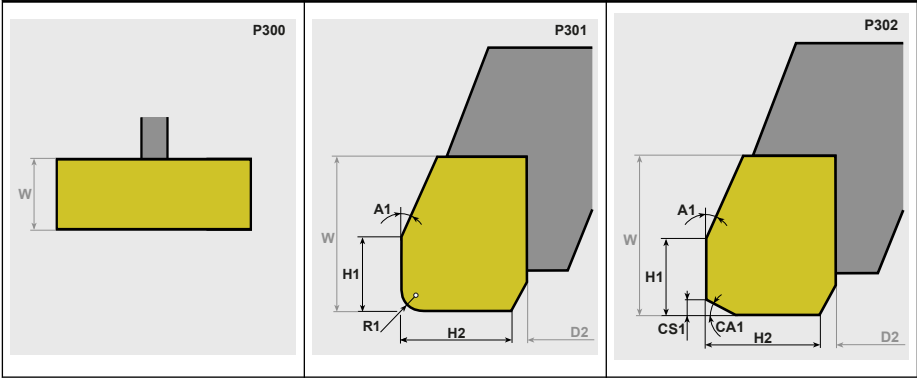
Tab. 7-35: Wheel forms for profile number P from 100 to 128

Wheel forms for inclined wheels TY2



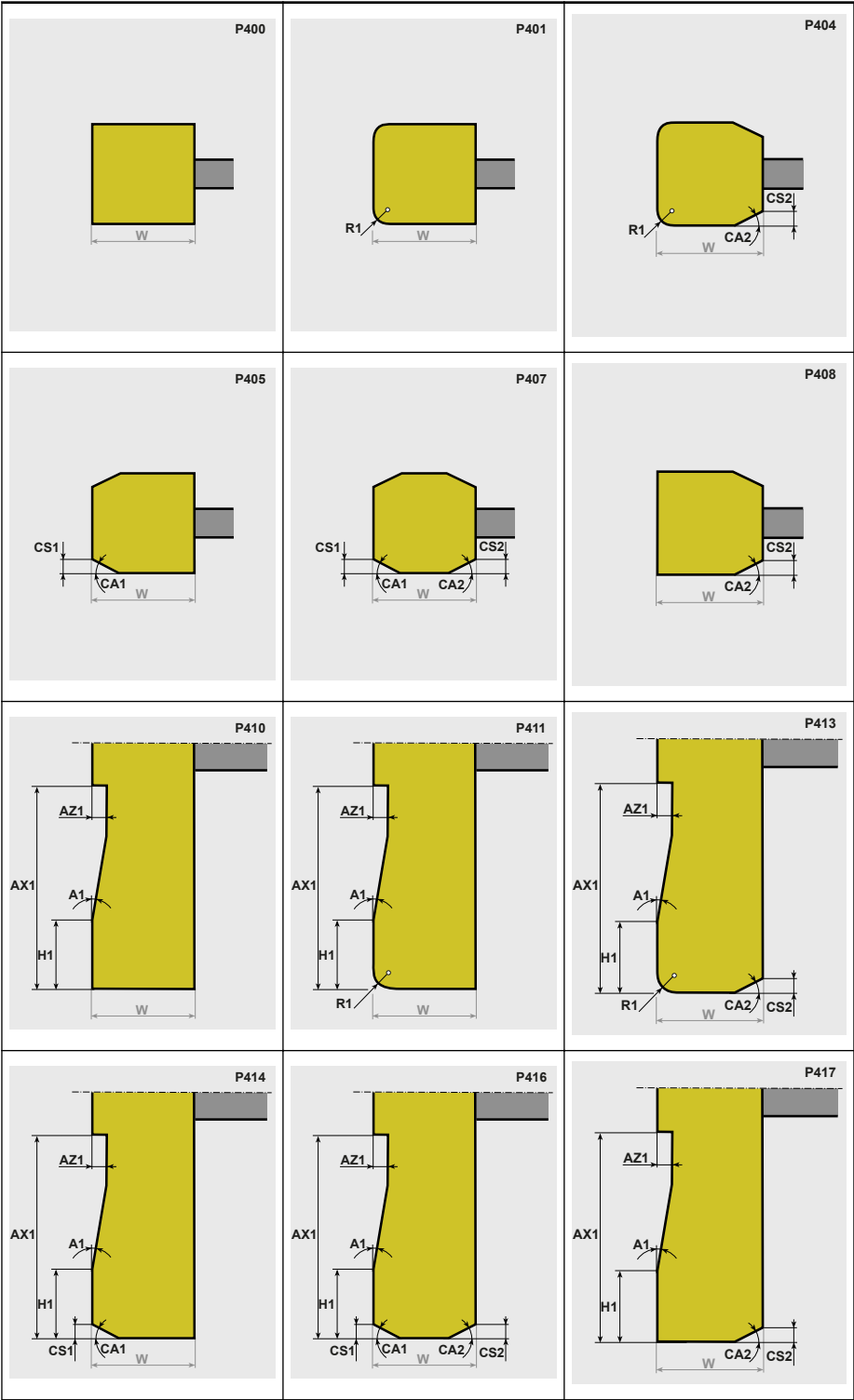
Tab. 7-36: Wheel forms for profile number P from 200 to 202

Wheel forms for face plates TY3



Tab. 7-37: Wheel forms for profile number P from 300 to 302

Wheel forms for internal wheels TY4



Tab. 7-38: Wheel forms for profile number P from 400 to 417

Parameters for the (left) profile

Parameters for the profile or the (left) profile

Parameter	Name	Properties	Values	Pre-assignment
H1	Shoulder height in X	Optional	≥ 0	0
R1	Radius	Optional	≥ 0	Current cutting edge radius correction
CS1	Width of the chamfer in X	Optional	≥ 0	0
CA1	Angle of the chamfer	Optional	$0 \leq CA1 < 90$	0
AX1	Back-slope amount in X	Optional	≥ 0	0
AZ1	Back-slope amount in Z	Optional	≥ 0	0
A1	Back-slope angle	Optional	$0 \leq A1 < 90$	0
H2	Face width	Optional	≥ 0	$(D1I-D2)/2$

Tab. 7-39: Call of the parameters defining the (left) profile

Description of the parameters for the (left) profile

H1 Shoulder height in X

Determines the height of the shoulder.

The lower limit of the shoulder height is formed by CS1 and/or by the tangential transition from the radius to the back-slope. If the minimum shoulder height is not reached, the value is automatically corrected to the minimum shoulder height.

R1 Radius

Determines the corner radius of the profile.

CS1 Width of the chamfer in X

Value of the chamfer in X in radius dimension.

CA1 Angle of the chamfer

Determines the angle of the chamfer.

AX1 Back-slope amount in X

Defines the smallest back-slope diameter. To be specified in radius dimension.

AZ1 Back-slope amount in Z

Depth of the back-slope in Z.

A1 Back-slope angle

Defines the back-slope angle relatively to the edge.

H2 Face width

Defines the profile width in Z at the face of a face plate.

Parameters for the (left) profile

Parameters for the right profile

Parameter	Name	Properties	Values	Pre-assignment
H2	Shoulder height in X	Optional	≥ 0	0
R2	Radius	Optional	≥ 0	R1
CS2	Width of the chamfer in X	Optional	≥ 0	0
CA2	Angle of the chamfer	Optional	$0 \leq CA2 < 90$	0
AX2	Back-slope amount in X	Optional	≥ 0	0
AZ2	Back-slope amount in Z	Optional	≥ 0	0
A2	Back-slope angle	Optional	$0 \leq A2 < 90$	0

Tab. 7-40: Call of the parameters defining the right profile

Description of the parameters for the right profile

H2 Shoulder height in X

Determines the height of the shoulder.

The lower limit of the shoulder height is formed by CS2 and/or by the tangential transition from the radius to the back-slope. If the minimum shoulder height is not reached, the value is automatically corrected to the minimum shoulder height.

R2 Radius

Determines the corner radius of the profile.

CS2 Width of the chamfer in X

Value of the chamfer in X in radius dimension.

CA2 Angle of the chamfer

Determines the angle of the chamfer.

AX2 Back-slope amount in X

Defines the smallest back-slope diameter. To be specified in radius dimension.

AZ2 Back-slope amount in Z

Depth of the back-slope in Z.

A2 Back-slope angle

Defines the back-slope angle relatively to the edge.

Special features

The parameters do not have any address.

Parameters that - according to the profile shape - are irrelevant and have no effect, are subject to a value range check.

The generation of the tool graphic is based on the modal states of the NC at the time of the cycle processing. Thus, it is mandatory to activate the tool corrections and the correct interpolation plane **before** calling the cycle.

Procedure/processing The tool description superimposes the simulation of the standard tool change via the tool change subroutine and generates a new graphic at the tool node that is assigned to the calling channel. The generation does not depend on the last tool.

The tool graphic remains until it is revoked with the cycle call without parameter. (TOOL or TOOL ()).

Example Create the tool description for a conical cup wheel according to the drawing.

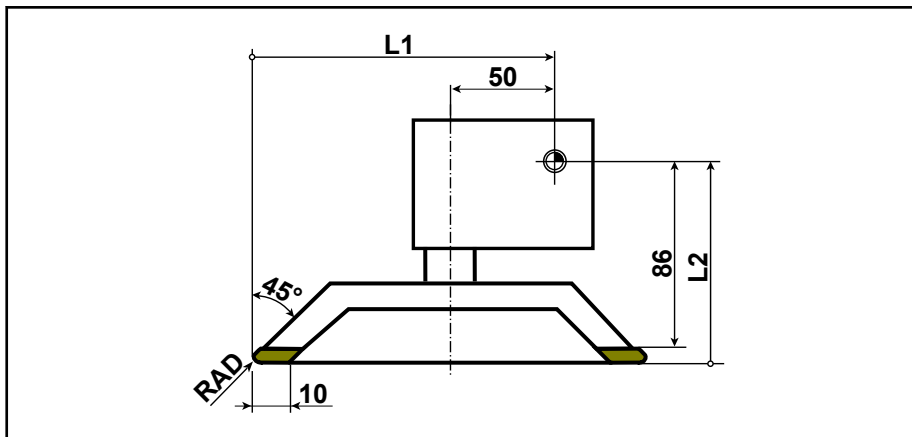


Fig. 7-20: Example: tool description for a cup wheel

Program:

```

100 L1=TCV("L1")           : REM correction value in Z
110 L2=TCV("L2")           : REM correction value in X
120 D1=2*(L1-50)           : REM outer diameter from L1
130 W=L2-86                : REM wheel width from L2
N10 TOOL("GRIND",3,,D1,,W,,,,,45,10) ; Tool description

```


8 Configuration

8.1 Overview

Contents The configuration contains important information to ensure the basic functioning of the simulation usually created during its commissioning. The information defines the machine model to be used and the link of the model components and other details with the NC kernel. Settings/options constitute a second data group adjusting the simulation to the hardware and the user requirements.

Creating a file For a new IndraWorks project, the "MTX.Simulation.Options.xml" file is created at the latest when the simulation becomes visible for the first time and it is filled with default values. The simulation tries to connect the available drives to the axes of the referenced machine model. If the model is not available or if no model is referenced, the simulation uses a minimized model according to the machining method.



Always provide the machine model first and then start the simulation. Otherwise, data can be lost when the simulation tries to connect the drives to the axes of the standard machine model. If it happens unintentionally (as you have the machine model and a suitable configuration in the project and you just did not provide the model correctly), transmit and enable the visualization data again after the model import (IndraWorks Engineering, context menu of the visualization device).

Editing A text editor changes the "MTX.Simulation.Options.xml". The interface creates a temporary file with the current settings and configurations that can be directly changed in the XML format. For more detailed information on option changing, refer to the operating instructions of the simulation.

If contents are deleted, invalid values are entered or the XML syntax was violated, the default values are applied.

The element names usually reflect their meaning. There are English comments above multiple elements describing their meaning and/or the value range.



Within text blocks (between beginning and end of tag) and attributes, some characters are not allowed. Replace them as follows:

- "Less than" character < in text blocks by XML notation <
- "Greater than" character > in text blocks by XML notation >
- & character in text blocks by XML notation &
- " quotation marks in attributes by XML notation "

Example: Text block IF A\$<>" THEN in XML notation IF A\$<>" THEN.

Instructions similar to CPL

Within individual text blocks of the "Machine" section, program parts can be stored in a CPL-based language (Custom Programming Language, see documentation "MTX Programming Manual", chapter "Basics of NC programming"). They are processed interpretatively if, e.g., the tool is changed or the position value of an axis. Thus, the simulation can be adjusted applicatively to multiple conditions.

The simulation initializes input and output variables context-dependently, which can be changed in the instruction section using auxiliary variables, if required. Afterwards, the simulation uses the last assigned values of the input and output variables.

All assignments are only valid for the current call and do not affect subsequent calls.

The program lines are not introduced by a number (CPL block number). Separate multiple instructions in one line by a colon.

Errors in programming (syntax, value range violation, unknown variable or instruction, etc.) are displayed in an output window below the simulation.

Instruction on program structuring

Program branches with `IF <Condition> THEN <Block> {ELSE <alternative block>} ENDIF`.

Simple variables

Simple variables are created either by initializing the input and output variables or upon their first assignment (auxiliary variables). The names consist of capital letters, numbers and underscore followed by a type-dependent postfix.



Arrays of simple variables cannot be created. Compared to CPL, there is no DIM instruction.

- Integers with postfix %. Example: `COUNT%`
- Boolean variables with postfix ?, values `TRUE` and `FALSE`. Example: `EMPTY?`
- Real numbers, without postfix. Example `POSITION`
- Strings with postfix \$. Example `NAME$`.

Structured variables

Structured variables are created context-dependently and only by the system as input and output variables. The variables are read-only and sometimes provided with a write access.

In all contexts, SD variables can be used, however, only with read-access, whereas indices may only be written as constants.



System data changes are not transmitted to the simulation synchronously to the interpolator cycle.

Another important structure is the DBT1 tool data record which is only provided in the "Tool graphic generation" context. The `Rec` substructure is optional.

Examples:

```
PX=SD.SysARA[1].Pos1 : REM read-only access to a system date
```

```
TNAME$=DBT1.Rec.Hd.SKQ : REM read-only access to the ident in the tool data record
```

Operators

Operators	Instruction
Assignment	=
Addition (arithmetic and connection of strings)	+
Subtraction	-
Multiplication	*
Division	/
Modulo (modulo operation)	%
Power	^
Faculty	!
AND (Boolean expression and bit-wise integer operation)	AND
OR (Boolean expression and bit-wise integer operation)	OR
Exclusive OR (Boolean expression and bit-wise integer operation)	XOR
Negation (Boolean expression and bit-wise integer operation)	NOT
	equal to
	greater than/equal to
Comparisons of figures, character strings and Boolean variables or expressions	greater than
	unequal to
	less than/equal to
	less than

Tab. 8-1: Operator list



Expressions using `NUL` (undefined variable in CPL) are not allowed.

Functions

The arguments are enclosed by round brackets. Separate multiple arguments by commas.

Functions	Instruction code
Trigonometric functions	Sine
	Cosine
	Tangent
	Arcus sine
	Arcus cosine
	Arcus tangent
Addition (arithmetic and connection of strings)	+
Absolute value	ABS(<Digit>)
Conversion of a floating-point number into an integer by cut-off	INT(<Digit>)

Functions	Instruction code
Conversion of a floating-point number into an integer by rounding up or down	ROUND(<Digit>)
Square root	SQRT(<Digit>)
Supply ASCII code of a character as integer	ASC(<Character>)
Conversion of a floating-point number into an integer by cut-off	CHR\$(<Digit>)
Convert number into a string	STR\$({<Format-String>,}<Value>)
Conversion of a string into a number	VAL(<StringExpression>)
Removal of leading or trailing spaces	TRIM\$(<StringExpression>{,"L" "R"})
Reading characters from a string	MID\$(<StringExpression>,<StartIndex>{,<NumberOfCharacters>})
MID\$ must not stand on the left side of an assignment.	
Conversion of a string into capital or lowercase letters	MCASE\$(<StringExpression>,"U" "L")

Tab. 8-2: Function list

Comment

Comment form	Instruction code
Commenting the complete line	; {<Comment>}
Comment to the next colon or line end	REM {<Comment>}

Tab. 8-3: List of comment instructions

8.2 Introduction to the tool graphic generation

Tools in the configuration of the simulation

The tools are largely generated automatically according to implemented rules. If the rules are violated or special features have to be implemented, knowledge of the generation and influence on the generation are required. The following section also provides information on the import of STL models for tools.

The applicative influencing of the tool graphics calculation is effected is executed by a program section. Its use - described in detail in [chapter "Machine" on page 123](#) - with regard to contextual and geometric aspects is addressed in the following.

How is a tool implemented into the simulation?

Normally, the programmed tool is exchanged during a tool change. In a machine tool with revolver, the place and/or the sector of the programmed tool changes. Its combination always refers to the data record of the new tool (data record key). During the tool change at a machine with tool magazine and when editing a displayed tool in the tool editor, the tool data record below a specific place/sector changes automatically. The simulation registers the change in the data record key or the data record, initializes the input and output parameters, starts the mentioned program section and calculates the tool graphic from the return parameters. Thus, the physical tool change also triggers a change in the simulation graphic.

The programmed tool is able to generate a tool path graphic. If there are several exchanged tools in one channel, only one of them generates the tool path.

There two more options for the simulation to display a tool graphic:

- For revolvers, the access to tool data records for the remaining (not exchanged) tools can be defined via database queries. This provides the tool information for the entire revolver fitting.
- The programmed tool whose graphic is computed from the respective data record of the tool management can be temporarily superimposed over the `TOOL` cycle using a tool description.

How is the tool graphic generated?

The tool graphic (here so-called meshes) can be generated as follows:

- by analyzing the relevant geometry and technology values of the tool data record
- by **importing an STL file**
- by **generating meshes** using one of the following methods:
 - **Rotating** a contour (revolved contour).
A rotation body is described by rotating a two-dimensional shell contour around a rotary axis
 - **Extruding** a contour (extruded contour).
A body is described by the linear movement of a closed and not overlapping, two-dimensional outline contour along the vertical (height)

The contour can be generated as follows:

- by directly specifying segments and arcs
- by parameters which the simulation includes in the contour computation. To the parameters apply context-dependent default values or dependencies in relation to other variables. Thus, they do not have to be specified in the configuration



When generating meshes from parameters, assumptions or simplifications are often applied. Consequently, the geometry is not exactly foreseeable and unsuitable for an exact collision consideration.



Provide STL models to be imported for tools in the "IndraWorks \MW Models\Tools" installation directory.

Parts of a tool from the simulation view

The graphic of a tool can consist of up to four parts. Every part can be represented by up to one body and is usually described separately from the other parts.

In the program part, there is one string variable as input and output parameter per part. Use this variable to reference to an STL file using a file name. Alternatively, define the contour to be rotated/extruded and/or the parameters.

- **Cutting edge** (cutting part, variable `CUTT$`):

This part can remove or apply material from/to the workpiece model or shapen the workpiece if the tool is active.

With drilling and milling tools for example, the cutting edge part is formed by the enveloping body of the rotating cutting edges; with grinding tools by the grinding layer. With jet cutting, the cutting edge is the

cutting jet. With a tool for generative production, the cutting edge geometry is used to define the torus cross-section of the applied material. It is invisible itself.

- **Shaft** (non-cutting part, variable `SHAFT$`):

The non-cutting part is mostly formed by a fixed unit with the cutting part.

With drilling and milling tools with indexable inserts, this is the part of the basic body that is not fitted with inserts.

With drilling and milling tools with a spigot (e.g. counterbore) or protruding tightening nut, this part is considered as shaft. The remaining part of the (physical) shaft above the cutting edge is then assigned to the chuck.

- **Chuck** (arbor, variable `ARBOR$`):

All tool clamps and fixtures such as chucks, collet chucks, shell-type cutter arbors, are attributed to the chuck. The chuck may also form a physical unit with the holder, but still be defined as separate mesh.

- **Holder** (holder, variable `HOLDER$`):

The holder establishes the connection of the tool-chuck combination with the fixture at the spindle or the revolver.

It is mostly a steep-angle taper for drilling and milling machines. It is sufficient to only model the part protruding from the spindle.

During punching, the holder can be the tool guide on which the plunger acts.

Except for the cutting edge, all tool parts are functionally equal. They differ in color and their the assembly to the tool-chuck combination. Which of the specified parts are actually considered in a certain tool depends on the tool type.

Special characteristics when adding components

Generic Revolved Tool (GRT) for rotation-symmetric tools (except for grinding tools):

With rotation-symmetric tools, the meshes of **cutting edge** and **shaft** are stacked like a tower. The penetration of both meshes or an integrated assembly is not possible. Usually, the upper limitation of the cutting edge and the lower limitation of the shaft are on the same height. When importing meshes, it might be necessary to omit internal parts or to adjust the geometries using a CAD system. If you only have one single model comprising cutting edge and shaft, import it as cutting edge and define the cutting part thereof using the `lu` (useful length) technology parameter in the tool editor.

The imported meshes for the cutting edge may be of any shape (e.g. the individual inserts of a hedgehog end mill). The cutting edge geometry in the simulated tool is calculated using the enveloping body of the rotating mesh. The tool is always shown as currently rotating.

With rotation-symmetric tools, the meshes of **chuck** and **holder** are also stacked like a tower. It is checked whether the tool length is sufficient to represent the chuck or the holder.

The combination of holder and arbor and the combination of shaft and cutting edge may penetrate each other.

Generic Tool (GT) for grinding, turning and special tools:

The arrangement of the tool components should ensure that the zero points of the meshes are located at the tool zero point. Penetration is possible.

Generic Extruded Tool (GET) for punching tools:

With punching and forming tools, only cutting edge and holder can be specified which are stacked like a tower, depending on the function (upper/lower die). The generic creation of the cutting edge geometry is based on an offset of a plane contour longitudinally to its vertical height called extruding.

Tool components according to tool types

Name of type/ group	Type number (1) / type in tool description (2)	Cutting edge CUTT\$	Shaft SHAFT\$	Chuck ARBOR\$	Holder HOLDER\$
Standard tool	0	When importing or generically creating from a contour interpretation of the meshes for GT. Generically from technology and cutting edge data for GRT	Import or generic	Import or generically from parameters in ARBOR\$	Import or generic
Drilling tools	1000..1016	Import, rotated contour or generic from technology and cutting edge data Is always created GRT	Import, rotated contour or generic Is only created if the length is sufficient and with a cutting edge, generically generated from technology data	Import, rotated contour or generic from parameters in ARBOR\$ Is only created if the length is sufficient	Import or generic Is not automatically generated. Only if the length is sufficient
Milling tools	2000..2017				
Turning tools	3000..3027	Not implemented GT	--	--	--
Measuring probe	5001..5003	Balls (ruby) Import or generically from the cutting edge data GT	Handle Import or generically from the cutting edge data	Import, rotated contour or from parameters in ARBOR\$ Is only created if the length is sufficient	Import or generic Is not automatically generated and only created if the length is sufficient
Special tool: gripper	6000	Import GT	Import	Import	Import
Jet cutting tool	6001	Only generically, from technology and cutting edge data GRT	Results in a transparent "flame" in cutting edge color Not available by default Only import or rotated contour possible	Import or rotated contour It is not automatically generated	

Configuration

Name of type/ group	Type number (1) / type in tool description (2)	Cutting edge CUTT\$	Shaft SHAFT\$	Chuck ARBOR\$	Holder HOLDER\$
AM tool	6002	Torus cross-section of the material to be applied Import, rotated contour or from technology data Is always created GRT	Import, rotated contour or generically, from technology data Is only created if the length is sufficient	Import, rotated contour or from parameters in ARBOR\$ Is only created if the length is sufficient	Import or generic Is not automatically generated and only created if the length is sufficient
Punching tools	7000..7013	Import, extruded contour (in case of lower part without hole) or from technology data Is always created GET	Not available	Not available	Import or generic
Drilling tools	"DRILL", "CDRILL", "BBAR"	Only generically from the parameters of the tool description and the tool correction GRT	Only generically suitable for the cutting edge.	Import, rotated contour or from parameters in ARBOR\$	Import or generic Is not automatically generated and only created if the length is sufficient
Milling tools	"MILL", "CHAMFER", "CRMILL"			Is only created if the length is sufficient	
Grinding tools	"GRIND"	Only generically from the parameters of the tool description and the tool correction GT	Only import or rotated contour possible It is not automatically generated		

- (1) The type number is in the tool data record in the DBT1.Rec.Hd.IKQ2 element
- (2) The type in the tool description is a string corresponding to the first parameter of the TOOL cycle and which is accessed in the input/output structure for tool descriptions using the DESCR.TYPE element

Tab. 8-4: Tool components depending on the tool type

Syntax of CUTT\$, SHAFT\$, ARBOR\$ and HOLDER\$

When assigning variables for cutting edge, shaft, chuck and holder, observe the subsequently listed syntactic rules:

- **Import of an STL model:**
Specify the file name with ".stl" extension and without path.
Example: HOLDER\$="Holder-2000-03.stl"
- **Rotated contour:**
After REV, followed by a comma or a colon, alternately specify a diameter coordinate followed by a length coordinate per corner point (without address). A command is used as separator between the figures. The comma may be preceded or followed by any number of spaces. Constant coordinates do not have to be written.

The contour must always run in the positive diameter range. Usually, it starts at length 0 and is continued in positive length direction. By default, the starting point is [0,0]. These zeros can be omitted.

If the end point of a contour element is reached via an arc, the radius has to be specified as follows in front of the end point coordinates, separated by a separator:

- Arc in clockwise direction: CW<Radius>
- Arc in counterclockwise direction: CCW<Radius>
- If the opening angle of the arc is greater than 180°, specify the radius with a negative sign

If the first or the last element is a plane surface, they do not have to be specified in the contour path.

By default, the orientation of the rotary axis is aligned parallelly to Z. Change the orientation with a self-sustaining parameter ORI that can be listed at any position, separated by comma, within or at the end of the contour.

- ORI0: rotary axis, longitudinal Z (default)
- ORI1: rotary axis, longitudinal Y
- ORI2: rotary axis, longitudinal X

By default, the rotary axis goes through point [0,0,0]. Change the position using the self-sustaining parameters X, Y, Z to be inserted at any position, separated by comma, within or at the end of the contour.

Example:

The electronic part of a measuring probe has the following dimensions and is to be shown as chuck:

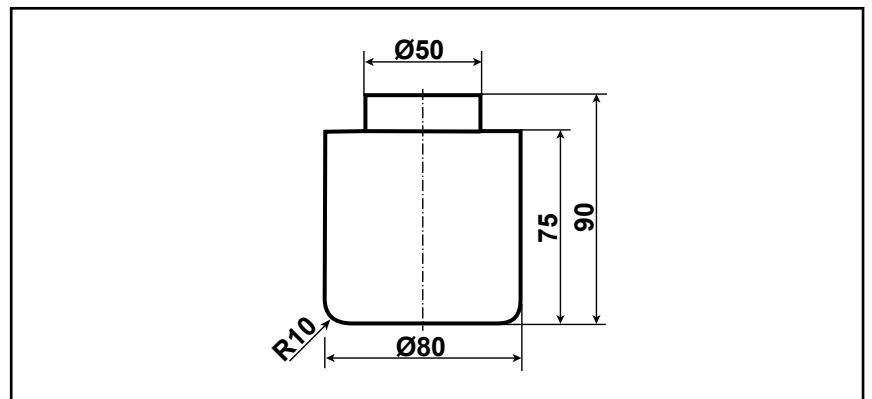


Fig. 8-1: Example: Housing of a touch probe

```
ARBOR$="REV:60,0,CCW10,80,10,80,75,50,75,50,90" : REM
All coordinates written
```

```
ARBOR$="REV,60,,CCW10,80,10,,75,50,,,90" : REM Re-
peating coordinates and start 0 omitted.
```

- **Extruded contour:**

After EXT, specify the contour with abscissa and ordinate alternately, analogously to the rotated contour, however both coordinates in radius dimension. Deviating from the rotated contour, there are no limitations regarding the value range of the coordinates.

Any non-closed contour is automatically closed by the attachment of a section.

The thickness is specified by an addressed parameter within the point list: TH<Thickness>.

Without further specification, the contour is created at Z0 and moved longitudinally at Z by the thickness TH. To select a different height direction, supplement parameter ORI:

- ORI0: offset, longitudinal Z (default)
- ORI1: Offset, longitudinal Y
- ORI2: Offset, longitudinal X

Change the position of the base with the self-sustaining parameters X, Y, Z to be listed at any position, separated by comma, within or at the end of the contour.

Example:

The drawing shows the dimensions of the cutting edge of a punching tool (upper part). For its thickness extending in Z-direction, use the CUTTHEIGHT variable.

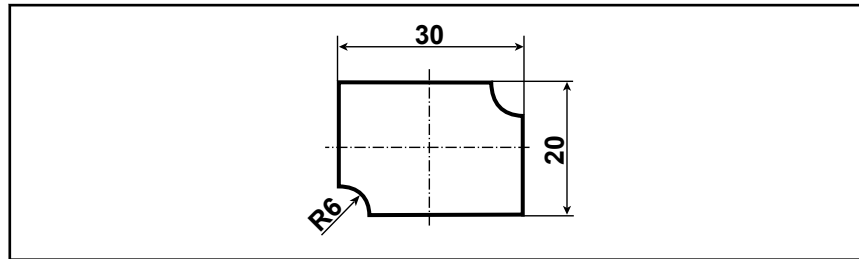


Fig. 8-2: Example: Punching tool

CUTT

```
$="EXT:-9,-10,15,,,4,CCW6,9,10,-15,-4,CCW6,-9,-10,TH
"+STR$(CUTTHEIGHT)
```



As an open body cannot be generated by extrusion, use a circular disk for the lower die of a punching tool instead. Alternatively, a lower die can result by extruding two contours from two halves. Separate the contours from each other using commas.

- **Cuboid:**

Describe an axis-parallel cuboid with CUB followed by two of its corner points (minimum and maximum coordinate values). Syntax "CUB:<X1>,<Y1>,<Z1>,<X2>,<Y2>,<Z2>"

Example:

```
HOLDER$="CUB,-50,-40,-30,50,40,0"
```

- **Cylinder:**

Write a cylinder with optional internal boring with CYL. Syntax "CYL:<D1>,{<D2>},<H>,{<X>},{<Y>},{<Z>}{,ORI<ori>}" with:

- D1 outer diameter
- D2 Inner diameter (default 0)
- H Height
- X, Y, Z Position of the lower base center (default 0)

- ORI Rotary axis orientation (ORI0 longitudinal Z (default), ORI1 longitudinal Y, ORI2 longitudinal X)

Example:

```
CUTT$="CYL:20,,30,,,,"
```

- **Taper:**

Taper and truncated cones are defined using CON. Syntax "CON: {<D1>}, {<D2>}, <H>, {<X>}, {<Y>}, {<Z>} {, ORI<ori>}" with:

- D1 Diameter of lower base (default 0)
- D2 Diameter of upper base (default 0)
- H Height
- X, Y, Z Position of the lower base center (default 0)
- ORI Rotary axis orientation (ORI0 longitudinal Z (default), ORI1 longitudinal Y, ORI2 longitudinal X)

Example:

```
CUTT$="CON:,20,30,,,,-30"
```

- **Polygon:**

Polygon parts are described with POL. Syntax "POL: <D>, {<A>}, <H>, {<X>}, {<Y>}, {<Z>} {, ORI<ori>} {, N<n>}" with:

- D Diameter/wrench size
- A orientation angle (default 0)
- H Height
- X, Y, Z Position of the lower base center (default 0)
- ORI Longitudinal axis orientation (ORI0 longitudinal Z (default), ORI1 longitudinal Y, ORI2 longitudinal X)
- N Number of edges (default 6)

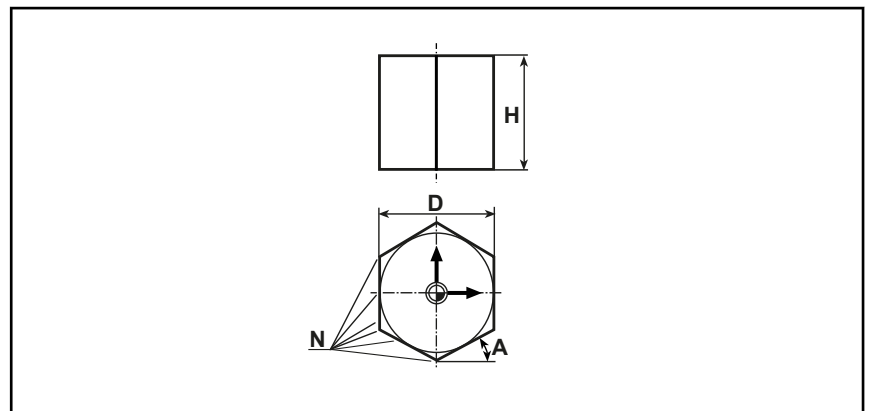


Fig. 8-3: Parameter of a polygon to be generated generically

Example:

```
CUTT$="POL:16,30,12,,,ORI1,N8"
```

- **Ball:**

A ball is defined using BAL. Syntax "BAL: <D>, {<X>}, {<Y>}, {<Z>} {, ORI<ori>}" with:

- D Diameter

- X, Y, Z Position of the ball center (default 0)
- ORI Rotary axis orientation (ORI0 longitudinal Z (default), ORI1 longitudinal Y, ORI2 longitudinal X)

Example:

CUTT\$="BAL:6,, ,3"

- **Parameters for a cylindric chuck:**

If, for a tool type, a rotation-symmetric arbor geometry is considered, a cylinder with a chamfer at the front face is generated automatically if neither a suitable contour was defined nor a suitable STL model was imported and if no empty string was assigned to the ARBOR\$ variable.

For this case, the following parameters are provided:

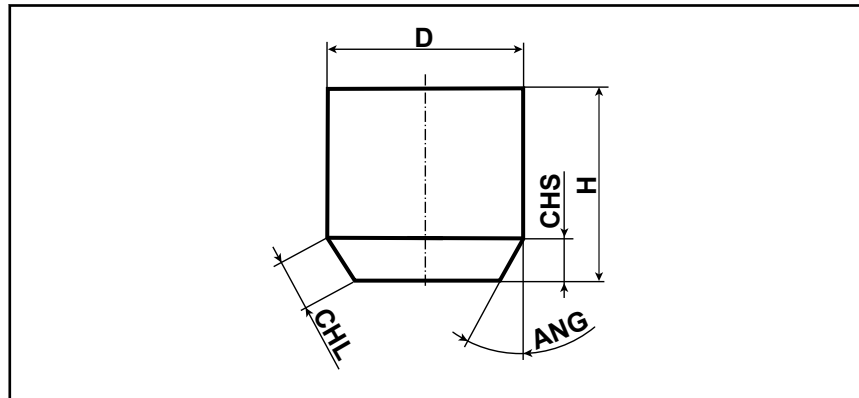


Fig. 8-4: Parameters for a cylindric chuck

- Total length H: The default value depends on the lengths of the remaining tool components
- Diameter D: The default value is usually greater than that the shoulder diameter (shaft) and slightly smaller than the holder diameter
- Chamfer angle ANG: Default value 45°
- Chamfer length CHL or alternatively the chamfer section CHS: Without this specification, the chamfer size is adjusted to the cylinder length and the diameters to shoulder and holder

The parameters can be assigned to the ARBOR\$ variable individually or combined in groups (separated by comma). It is also possible to define these parameters within a contour. If the body from the contour does not satisfy the spatial conditions available, a cylindric chuck according to the standard is inserted and you can affect the latter using the parameters.

Example 1:

A cylindric chuck is to be displayed with a 30° chamfer

ARBOR\$="ANG30"

Example 2:

The formation of a chamfer is to be prevented

ARBOR\$="CHL0"

Example 3:

No chuck geometry is to be generated

ARBOR\$=""

Tool length correction and orientation, position of the mesh zero points for Generic Revolved Tools

The generation of the tool shape has been explained up to now. Another important criterion for a correct graphic is its orientation in space.

Orientation means the alignment of a tool in the space coordinates X, Y and Z. With a rotation-symmetric tool (except for grinding), consider its rotary axis which has to run parallel to a spatial axis of the coordinate system at the tool node of the machine model (e.g. radial or axial drilling tool at a one-spindle lathe, orientation in the spatial directions X or Z). Tools may also be oriented in the opposite axis direction, e.g. at a lathe with opposite spindle on the second side. With lathes, the orientation of the tools corresponds to the interpolation plane (usually G18). This is also true for grinding, whereas here, the torus cross-section of the grinding layer in the interpolation plane is the decisive factor.

When programming or importing tool models for rotation-symmetric tools, define **cutting edge, shaft and chuck** in a coordinate system in which Z runs parallel to the rotary axis and the positive Z-direction points from the tip to the shaft. The definition of the specified geometries does not depend on the orientation and on the **length corrections** (e.g. with an angle head). Only **the zero point of the cutting edge in longitudinal direction has to match the position of the tool zero point**. The simulation then transforms the geometry according to the orientation and the length corrections.

The following applies to the selection of the **shaft geometry zero point**:

- If, in longitudinal direction, the lower limitation of the mesh is in the negative area, the geometry is arranged below the cutting edge (spigot or loss length). The chuck geometry has to include the actual shaft, if applicable
- In all other cases, the shaft is "put" onto the cutting edge (normal case)

When orienting the **holder**, the mentioned rules for cutting edge, shaft and chuck apply.

Regarding its zero point in longitudinal direction, the following applies:

- If the zero point is at the lower limitation of the holder geometry, the zero point at the tool holder is aligned to the top limitation of the holder (normal case)
- In all other cases, the zero point of the holder geometry is on one level with the zero point at the tool holder

The zero point of the holder in the level vertical to the rotary axis always corresponds to the zero point at the tool holder.

Complete examples with rotated and extruded contours

The following examples show how to adjust and extend the generation of tool graphics of Generic Revolved Tools.

Example:

Double angle cutter

Completely implement the generation of the tool graphic for a double angle cutter. The implementation is an extension of the "angle cutter" tool type and depends on whether the ident starts with the "DbIAng" string. The variable diameter of the milling arbor should be 80% of the cutter face diameter, but not exceed 50 mm. At the lower side, there is a tightening nut (spigot) with a height of 12 mm.

For all tools, the holder geometry should be analogously to HSK 63.

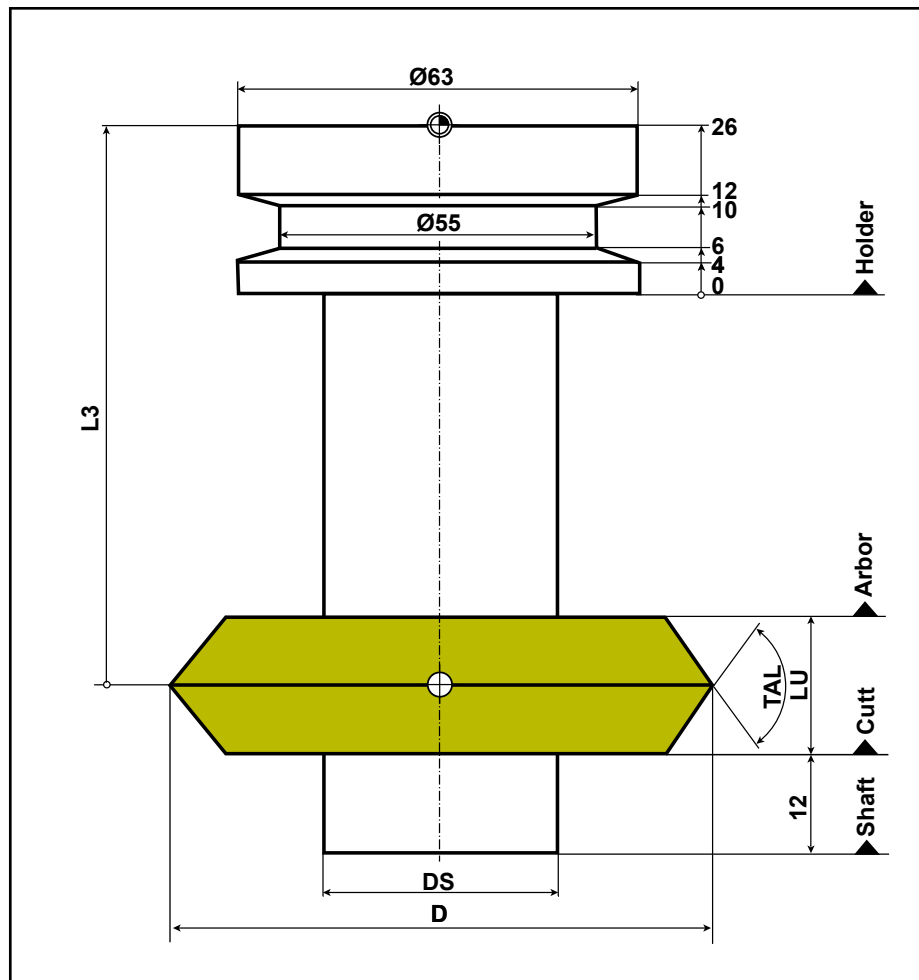


Fig. 8-5: Double angle cutter with holder HSK 63 (approximated geometry)



As the spigot is to be placed below the cutting edge section, the contour is developed with negative longitudinal coordinates.

The holder cone is omitted as it will remain invisible within the headstock. For modelling, the zero length coordinate of the holder contour has to be on the zero point of the tool holder.

Program:

```
TNAME$=DBT1.Hd.SKQ           : Get REM ident from tool dataset
IF TNAME$<>" " THEN           : REM tool is no empty tool
                                ; Holder contour HSK 63 for all tools
    HOLDER$="REV,63,,,4,55,6,,10,63,12,,26"
    TY%=DBT1.Hd.IKQ2          : Get REM tool type number
    IF (TY%=2010)AND(MID$(TNAME$,1,6)="DblAng")AND(EDREF%<0) THEN
                                ; User implementation for double angle
                                ; cutter
        D=DBT1.UD.Tl.Tech.Tech1 : Get REM outer cutter diameter
        LU=DBT1.UD.Tl.Tech.Tech5 : Get REM cutting edge height
        TAL=DBT1.UD.Tl.Tech.Tech6 : Get REM angle
        L_3=DBT1.UD.Ed[EDREF%].Geo.L3 : Get REM length correction L3
        IF (TAL<=0)OR(TAL>170) THEN
            TAL=90                : REM Use standard angle if required
        ENDIF
        IF LU<=0 THEN LU=10 ENDIF : REM Set standard cutting edge height if
                                ; required
        D1=D-LU/TAN(0.5*TAL)      : REM Calculate face diameter of cutting
                                ; edge
                                ; Validate remain. length and face diam.
```

```

IF (D1>0)AND(L_3-26-0.5*LU>0) THEN
  LL=12                                : REM spigot length 12 mm
  DS=D1*0.8                            : REM Determine/limit milling arbor diam.
  IF DS>50 THEN DS=50 ENDIF
  ; Def. contours for shaft, cutting edge, arbor
  SHAFT$="REV,"+STR$(DS)+",""+STR$(-LL)+",""+0"
  CUTT$="REV,"+STR$(D1)+",""+STR$(-0.5*LU)+",""+0"
  CUTT$=CUTT$+STR$(D)+",""+0,""+STR$(D1)+",""+STR$(0.5*LU)
  ARBOR$="REV,"+STR$(DS)+",""+STR$(L_3-26-0.5*LU)
ENDIF
ENDIF
ENDIF

```

Example:

Radially set form countersink at a lathe (angle head)

Define the generic graphic for a radial angle head including collet chucks.

Implement the tool graphic for a form countersink as per the drawing. The implementation is to be an extension of the "general tool" tool type and depends on whether the ident matches the "FormCounterSink 12x6" string.

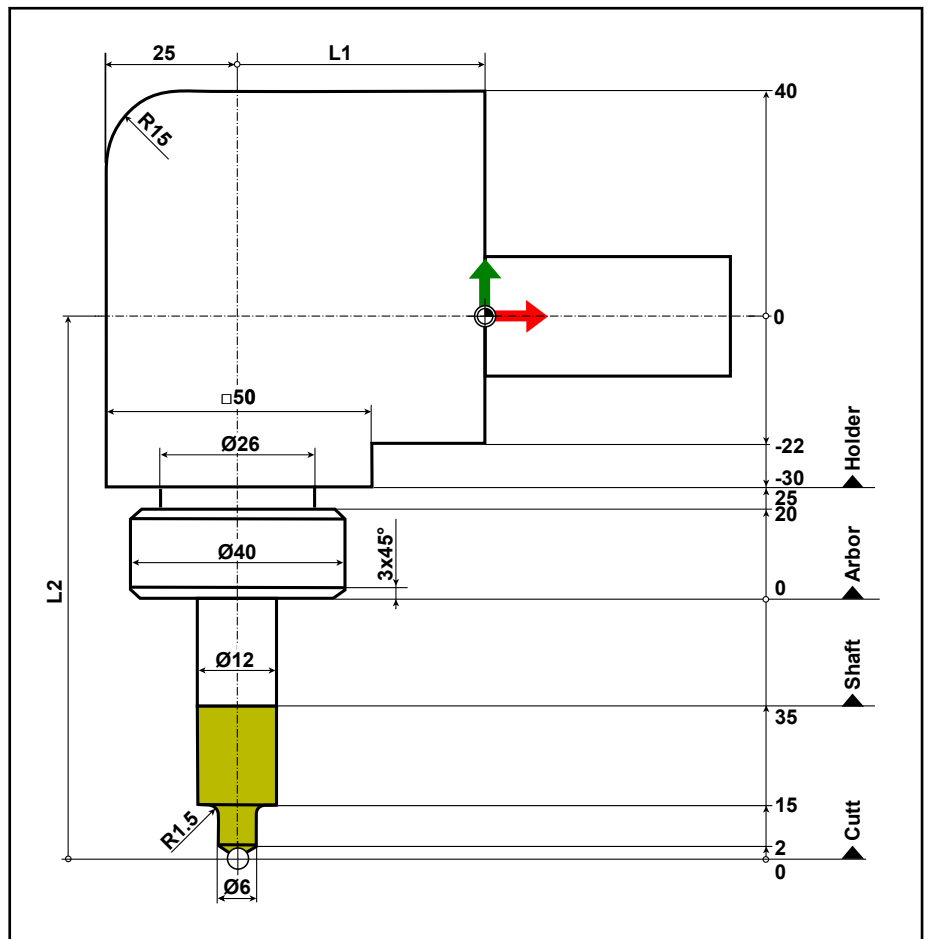


Fig. 8-6: Form countersink 12x6 with radial angle head



Generate the angle head mesh by extrusion. The abscissa of the coordinate system for this contour is marked by a red arrow, the ordinate by a green arrow. The zero point of the contour has to comply with the zero point at the tool holder, as the geometry in longitudinal tool direction exceeds the zero point at the tool holder. In this case, the direction of extruding is Y. Write OR11.

Configuration

Program:

```

TNAME$=DBT1.Hd.SKQ           : REM Get ident from tool dataset
IF (TNAME$&lt;&gt;"")AND (ORIPLANE$="G19")AND (EDREF&gt;0) THEN
    ; Tool not empty and radially oriented.
    L_1=DBT1.UD.Ed[EDREF%].Geo.L1 : REM Get length correction L1
    L_2=DBT1.UD.Ed[EDREF%].Geo.L2 : REM Get length correction L2
    ; Validation for angle head incl. collet chucks
    IF (L_1&gt;50)AND (L_2&gt;55) THEN
        ; Contours for angle head and collet chucks
        HOLDER$="EXT,0,-22,,40,"+STR$(-L_1-10)+",,CCW15,"+STR$(-L_1-25)
        HOLDER$=HOLDER$+",25,,30,"+STR$(-L_1+25)+",,,-22,Y-25,TH50,ORI1"
        ARBOR$="REV,34,,40,3,,17,34,20,26,,25"
        IF (TY%=0)AND (TNAME$="FormCounterSink 12x6") THEN
            ; Branching for form countersink
            SHAFTLEN=L_2-30-25-35 : REM remaining shaft length
            IF SHAFTLEN&gt;=0 THEN : REM shaft length sufficient?
                ; Contours for shaft and cutting edge
                SHAFT$="REV,12,,,"+STR$(L_2-35)
                CUTT$="REV,,,6,2,,13.5,CW1.5,9,15,12,,,35"
            ENDIF
        ENDIF
    ENDIF
ENDIF

```

8.3 Description of the configuration file MTX.Simulation.Options.XML or MTX.Simulation.Options.Temp.XML

8.3.1 File structure

Header and main element The file can start with a header with a description of the valid XML-version and the XML-format in UTF-8 coding.

The main element <SimulationOptions> follows. There are several sections (settings divided in topic groups) below the main element.

Sections

There are the subsequently listed XML tags (sections). The function of every section is described and it is labeled whether it is an option configuration data to be specified for the commissioning.

- Tag <General> **Optimization** (--> configuration)

Includes settings to completely remove the simulation from IndraWorks Operation, for the compatibility and tests, fault analysis and recordings on the fault reproduction.
- Tag <Optimizing> **optimization** (--> Options)

The simulation is adjusted to the performance of the visualization unit (CPU, graphics) and the NC control (interpolator clock frequency) as well as the part spectrum (workpiece size). These settings define the graphic quality and the simulation performance.
- Tag <Machine> **machine** (--> configuration)

Defines which machine model is used and how the simulation is connected to the NC kernel:

 - The main machining process which affects the dialog control
 - Which axis of the model is moved by which drive?
 - Which tool node is operated by which channel due to the tool change?
 - How are tools activated/deactivated?
 - Which criteria are used for the revolver fitting?
 - How are tools and their holders to be dimensioned and oriented?
- Tag <View> **settings for display** (--> options)

Defines all settings for the graphical characteristics and the menu scope except for the colors

- Tag <Colors> **colors** (--> options)

Assigns special colors to the areas and lines. There are also color schemes allowing to assign a color with regard to a variable

- Tag <Punching> **punching** (--> configuration)

Special settings for the punching simulation

- Tag <AdditiveManufacturing> **generative manufacturing** (--> configuration)

Special settings for generic production methods (e.g. 3D printing)

- Tag <iWindow> **camera settings** (--> configuration)

The simulation graphic can be superimposed by a camera image. The camera settings for this special view for augmented reality are made in this section

File structure *Program:*

```
<?xml version="1.0" encoding=UTF-8"?>
<SimulationOptions>
  <General>
    <!--*****-->
    <!--***** General *****-->
    <!--*****-->
    <!--
    ...
    -->
  </General>
  <Optimizing>
    <!--*****-->
    <!--***** Optimizing *****-->
    <!--*****-->
    <!--
    ...
    -->
  </Optimizing>
  <Machine>
    <!--*****-->
    <!--***** Machine *****-->
    <!--*****-->
    <!--
    ...
    -->
  </Machine>
  <View>
    <!--*****-->
    <!--***** View Settings *****-->
    <!--*****-->
    <!--
    ...
    -->
  </View>
  <Colors>
    <!--*****-->
    <!--***** Colors *****-->
    <!--*****-->
    <!--
    ...
    -->
  </Colors>
  <Punching>
    <!--*****-->
    <!--***** Punching *****-->
    <!--*****-->
    <!--
    ...
    -->
  </Punching>
  <AdditiveManufacturing>
    <!--*****-->
    <!--***** Additive Manufacturing *****-->
    <!--*****-->
```

```

<!--
...
-->
</AdditiveManufacturing>
<iWindow>
<!--*****-->
<!--***** Camera View Settings (iWindow) *****-->
<!--*****-->
<!--
...
-->
</iWindow>
</SimulationOptions>

```

8.3.2 Configuring the NC simulation

General information

Availability of the simulation	Tag <IsSimulationAvailable>: The availability of the simulation allows to completely remove the simulation from the project. Value range: • True (default) The simulation becomes visible as control in IndraWorks Operation and can be operated. • False The simulation is not visible in IndraWorks Operation or in third-party applications without content. Simulation-dependent controls, such as workpiece preview and tool display are not available
Compatibility setting for offsets	Tag <OffsetCompatibility>: Up to 14V22, geometries for workpieces, clamping media and drawings were directly positioned in the coordinate system of the workpiece holders irrespective of whether the point in time matches the workpiece coordinate system. From 15V02, a transformation is inserted between the workpiece holder and the workpiece geometries guarantying the congruence between workpiece coordinate system and coordinate system used to embed the geometry. The new functionality requires a machine model without transformation node in the kinematic chain. Thus, changes in the machine model might be required. The Compatibility settings for offsets allows to return to the old workpiece geometry definition. The old workpiece geometry definition does not depend on the compatibility setting at G53 combined with G153. Value range: • False (default) The coordinate system to define the workpiece geometry becomes congruent with the current workpiece coordinate system due to a transformation. • True The new workpiece geometry is directly placed at the workpiece holder in the coordinate system.
Compatibility setting for STL files	Tag <StlFileCompatibility>: From 15V02, STL files with names (extension not considered) matching the main program, are automatically accepted as finished part geometry at the first workpiece holder of the machine model upon program selection. Thus, naming STL files is restricted. Disable this automatism with the Compatibility setting for STL files. Value range:

	• <code>True</code> (default) No automatic import of finished part geometries from STL files. • <code>False</code> STL file names matching the main program names are automatically loaded as finished part geometry.
OPC UA real-time item	Tag <code><NC.DataRT></code>: Determines if the newer OPC UA item is to be used with more information for the data transfer from the NC kernel to the simulation. Value range: <code>True</code> - yes (default), <code>False</code> - no  This setting is only intended for test purposes and not included in the file in case of the standard value.
Problem analysis (debug mode)	Tag <code><Debug></code>: This setting activates different functions for the problem analysis: • Context menu to create simulation records • Display of stack size, assumed item responses per second and simulated item responses per second • Use the dialogs to also import and export STL, DFX and binary files • Dumps of the machine model can be executed in the "Export" dialog • STL models of the tools Value range: <code>False</code> - no analysis function (default), <code>True</code> - analysis function active
Machine	
Main machining method	Tag <code><MachiningType></code>: The main processing method affects the dialog control, particularly which raw part types are provided in the menu. Possible values: • <code>Mill</code> (default) Mill machining center, e.g. machine tools for prism-shaped parts machining (e.g. laser machining). Provided raw parts: Cuboids and cylinders • <code>Turn</code> Lathe, e.g. machine tools for rotation-symmetric parts machining. Provided raw parts: Bar section and polygon • <code>Punch</code> Punching machine. Provided raw part: Sheet The punching machine cannot be declared explicitly. This process is automatically set if the value of the "Punches applied" function is "1" in Macoda 8001 00010 • <code>AM</code> Tool machine tool for generative manufacturing methods. Provided raw parts: Cuboids and cylinders With rectangular and cylinder-shaped raw parts in the volume presentation, only the base areas are shown as "base". The raw part dimensions describe the space within which the workpiece is generated.

- undefined
Machining method not assigned

Machine model **Tag** <MachineModel>:

A **machine model** consists of one XML file which defines the kinematics tree and more data, if applicable, used by the geometry elements as models (STL files) and for other purposes. The XML file starts with the main element tag <machine_definition>. All specified files are stored in one single folder as created in the virtual machine builder. To reference a machine model, specify the folder name as text. If there are several XML files with the specified main element, add the file name following a backslash or slash if required.

Without valid reference to a machine model, a milling machine or punching machine uses a standard model which can usually only move the linear axes X, Y and Z correctly.

The reference to the machine model and the storage location of the model are handled version-specifically.

- MTX version 15V06 and older:
Provide the machine model in a separate folder below the installation directory "IndraWorks\Simulation\MW Models". Enter the name of this folder into the `MachineModel` tag. If required, also add the names of the XML files.
- MTX 15V08 and following:
Machine models can be integrated directly into the IndraWorks project in the mentioned version and above. Right-click on the node of a machine model (project tree in IndraWorks Engineering) and select the "Applying..." menu item to load this machine model to the operating station. A specific machine model is selected for the simulation upon loading. If the `MachineModel` tag is empty, the latest machine model applied to the simulation is used. If a machine model name is entered into the `MachineModel` tag, the simulations will show this machine model if it was applied before the IndraWorks Operation start. If this model was not applied, it is searched in the IndraWorks installation folder and the standard machine model is used at the end. The tag `MachineModel` has only a check function and does not have to be filled in.

Example:

The machine model should be defined in the file "\\MachineCenter500\MachineCenter500.xml".

```
<MachineModel>MachineCenter500</MachineModel>
```

or

```
<MachineModel>MachineCenter500\MachineCenter500.xml</MachineModel>
```

By default, the entry is empty (line break).



When entering the machine model, ensure that there is no line break between the pointed brackets.

Axis connection **Tag** <AxisConnection>:

Below the **axis connection** tag, define for every axis of the kinetic tree (machine model) which drive moves this axis.

For this purpose, use the `<Axis>` tag multiple times with the following attributes:

- **Name:**
The name used for the axis in the machine model. Wild cards not allowed
- **Drive:**
The drive moving the axis. For NC axes, specify the drive number (>0). Apart from the drive numbers, there are the following possible values:
 - "undefined" - No drive assigned
 - "PUNCH" - The motion of the upper die of a punching machine is simulated
 - "PUNCHDOWN" - The motion of the lower die of a punching machine is simulated
 - "DOOR" - The motion is simulated by the "Open door" function, an intermediate level while hiding machine parts

In the element content of the `<Axis>` tag, a program section can be stored that affects the movement patterns of the axis and/or the visibility of the subordinate machine elements (kind elements) in the kinematic tree.

Variables of the "Axis" program section

Variable name	Type	Value range		Pre-assignment	Description
VALUE	real			Next coordinate value	Defines the axis position to be approached
VISIBLE?	boolean	TRUE	Visible	TRUE	Defines the visibility of the child elements
		FALSE	Invisible		
SUSPEND?	boolean	TRUE	Ignore coordinate	FALSE	Defines whether the coordinate is considered in the simulated movements
		FALSE	Use coordinate		
SD	Structure				Input variable with read-only access
STOCK	Structure			Current raw part data	Input variable with read-only access to raw part dimensions

Tab. 8-5: Input/output variables of the "Axis" program part

SD system data:

Elements from one array may only be addressed with their simple, constant and integer index (e.g. correct: `SD.Sub[1].XY`, wrong: `SD.Sub[I%+1].XY`).

Raw part dimensions `STOCK`:

The `STOCK` variable is an array providing the raw part dimensions, but not the current workpiece dimensions. Use a string with the name of the workpiece holder in square brackets as index. The index has to be a text constant, no expression. If no index is specified, the first workpiece holder of the machine model is addressed.

STOCK variable

Element name	Type	Value range		Description
EMPTY	boolean	FALSE	Workpiece available	Returns the availability of a workpiece
		TRUE	No workpiece	
DMIN	real	>=0		Inner diameter of a cylindrical raw part, otherwise 0
DMAX	real	>=0		Outer diameter of a cylindrical raw part, otherwise 0
HEIGHT	real	>= 0		Z-expansion
XMIN	real			Lower limitation in X
XMAX	real			Upper limitation in X
YMIN	real			Lower limitation in Y
YMAX	real			Upper limitation in Y
ZMIN	real			Lower limitation in Z
ZMAX	real			Upper limitation in Z

Tab. 8-6: Elements of the STOCK input variable



The interpretation of the program sections negatively affects the simulation performance. Consequently, store programs only rarely, especially when connecting NC axes and only for rarely moved drives (e.g. axes for the workpiece infeed).

Example:

A punching machine uses the linear axes "X" and "Y" (of the model) which includes the drives 1 and 2 to move the sheet over the machine table. The sheet is clamped by a manually operated clamping claw. The NC is informed on its X- and Y-position (in the model "XClamp" and "YClamp") using SD variables for the area monitoring (SD.SysARA[1].Pos1 or VSD.SysARA[1].Pos2). If SD.SysARA[1].IsValid and SD.SysARA[1].AreaType do not indicate a valid dead area, the clamping claw is to be invisible and be moved out of the working area to avoid collision warnings. Otherwise, correct the Y-position by half the Y-expansion of the dead area (SD.SysARA[2].Ext2/2). The Z-position of the clamping claw (in the "ZClamp" model) should match the upper sheet side.

The model axis "Z" that of the lower part by "Z2" is responsible for the stroke movement of the upper punching part. The tool is moved by a rotary axis "C" (drive 3) which simulated position should always be 0 if the ident of the active tool starts with the string "MT" (label of a multi-tool).

The "Open door" function operates the "LeftDoor" and "RightDoor" model axes.

Program:

```

<AxisConnection>
  <Axis Name="X" Drive="1"/>
  <Axis Name="Y" Drive="2"/>
  <Axis Name="C" Drive="3">
    IF MID$(SD.SPINDLE.Hd.SKQ,1,2)="MT" THEN
      VALUE=0
    
```

```

ENDIF
<Axis>
<Axis Name="XClamp" Drive="undefined">VALUE=SD.SysARA[1].Pos1</Axis>
<Axis Name="YClamp" Drive="undefined">
  IF (SD.SysARA[1].IsValid=1)AND(SD.SysARA[1].AreaType=1) THEN
    ; Valid dead area
    VALUE=SD.SysARA[1].Pos2-SD.SysARA[1].Ext2/2
    ; Position from area monitoring
    VISIBLE?=TRUE : REM support bracket visible
  ELSE
    VALUE=-9999 : REM remove from working area
    VISIBLE?=FALSE : REM support bracket invisible
  ENDIF
</Axis>
<Axis Name="ZClamp" Drive="undefined">VALUE=STOCK.ZMAX</Axis>
<Axis Name="Z" Drive="PUNCH"/>
<Axis Name="Z2" Drive="PUNCHDOWN"/>
<Axis Name="LeftDoor" Drive="DOOR"/>
<Axis Name="RightDoor" Drive="DOOR"/>
</AxisConnection>

```

Tool connection Tag <ToolConnection>:

Below the tag of the **tool connection**, the following can be defined for every tool node of the kinematic tree (machine model):

- How to create the tool graphic
- What is the query for the database query for revolver fitting
- When the tool activated (switched on) for machining

For this purpose, the <ToolNode> tag is to be used several times, if necessary, with the following attributes:

- Name:

The name of the tool node in the machine model to which the "tool" element is to refer to.

Wild cards are allowed:

- "*" (asterisk) stands for any and also an empty character string
- "?" (question mark) stands for exactly one character

Example: At one machine, there are two revolvers with the tool nodes called "Rev101" at "Rev112" at the first revolver and "Rev201" and "Rev212" at the second revolver. To assign all tool nodes on revolver 2 to one tool element, the following attribute values apply: "Rev2*", "Rev2??", "*2???", "????2*". "*2*", however, is wrong as it also includes "Rev112".

- Channel:

Specifies the number of the channel responsible for the tool change at this/these tool node(s)

- Function:

Specifies the function of the tool node. Possible values:

- "undefined" - The tool element is deactivated
- "Regular" - The tool element describes a normal tool node or, in case of punching, the upper part
- "DownDie" - The tool element describes the lower part of a punching machine. The tool node cannot generate a tool path graphic

- ToolpathPartnerChannels:

By default, only the tool path of the channel active at the interface is visible. Use the parameter to determine more channels (comma-separated

digits) whose tool path is to be displayed simultaneously with the `Channel`. Use this attribute if more channels simultaneously machine a workpiece.

Example: If channel 1 is active, the tool path of the channels 2 and 3 is also to be displayed. `ToolpathPartnerChannels="2,3"`

As element content of the `<ToolNode>` tag, three tags are possible having the subsequently listed functions, for which one program section each can be stored:

- `<Query>`:
Compiles the query for the database query for the revolver fitting
- `<IsToolActive>`:
Determines whether the tool is activated for machining (switched on)
- `<Tool>`:
This program section affects the graphic design of the tool

Tag `<Query>`

Variables of the "ToolNode/Query" program section

Variable name	Type	Value range	Pre-assignment	Description
QUERY\$	String		Empty string	Output variable with database query for this tool node
NAME\$	String		Name of the tool node in the model	Input variable
CUSTOMDATA\$	String		Value of the <code>Custom-Data</code> attribute of the tool node	Input variable
SD	Structure			Input variable with read-only access

Tab. 8-7: Input/output variables of the "Query" program section

If the QUERY\$ variable is empty or the database query does not find a tool, the tool node is not fitted outside the program processing.



For spindle places (pickup automatic tool changer or manual change), instructions may not be defined in the program section "Query" or an empty string has to be assigned to the output variable "QUERY\$".

For the syntax of the query string, go to the documentation "MTX 15VRS commissioning", chapter "Control commissioning", "Definition of the list content".

Example:

At a revolver machine, provide the database query using the name of the tool node. The name consists of "Rev" followed by the place number. The sector is constantly equal to 1.

Program:

```

<ToolConnection>
  <ToolNode Name="*" Channel="1" Function="Regular">
    <Query>
      QUERY$="K1=1,K2="+MID$(NAME$,4,LEN$(NAME$)-3)
    </Query>
    <IsToolActive/>
  </ToolNode>
</ToolConnection>

```

Tag <IsToolActive>**Variables of the "ToolNode/IsToolActive" program section**

Variable name	Type	Value range		Pre-assignment	Description
ACTIVE?	boolean	TRUE	on	Last state	Tool input/output variable activated
		FALSE	off		
NAME\$	String			Name of the tool node in the model	Input variable
CUSTOMDATA\$	String			Value of the CustomData attribute of the tool node	Input variable
SD	Structure				Input variable with read-only access.

Tab. 8-8: Input/output variables of the "IsToolActive" program section

The activation state of the tool is generally ON. For generative processes and jet cutting (e.g. laser machining), it is essential to control the activation state.



The SD.SysAdvDistSig[1].SignalStatus system date is the only one that is transmitted to the simulation synchronously with the interpolator cycle and should be used here.

Example:

At a machine, the activation state of a laser can be seen at bit 1 (starting from 0) of the SD.SysAdvDistSig[1].SignalStatus variable.

Program:

```

<ToolConnection>
  <ToolNode Name="*" Channel="1" Function="Regular">
    <Query/>
    <IsToolActive>
      ACTIVE?=(SD.SysAdvDistSig[1].SignalStatus AND 2)&lt;&gt;0
    </IsToolActive>
  </ToolNode>
</ToolConnection>

```

Tag <Tool>

Configuration

Variables of the "ToolNode/Tool" program section

Variable name	Type	Value range	Pre-assignment	Description
ARBOR\$	String	STL file name	<DBT1.Hd.SKQ>+"_arbor.stl" if available, otherwise "?"	Input/output variable for the chuck. No chuck if string is empty. In case of question mark, generation according to standard
		Contour definition and/or parameters		
		Empty string		
		Question mark		
CUSTOMDATA\$	String		Value of the CustomData attribute of the tool node	Input variable
CUTT\$	String	STL file name	<DBT1.Hd.SKQ>+"_cutt.stl" if available, otherwise " ?"	Input/output variable for the cutting edge part
		Contour definition		
		Empty string		
		Question mark		
CUTTDIA	real	>=0	0, i.e. definition according to a rule	Outer diameter of the cutting edge part of a punching tool (die) in mm
CUTTHEIGHT	real	>=0	Turning tool: 2.38 Grinding tool: 20 (1) Punching tool: by sheet thickness otherwise 0	Height of the cutting edge part of the specified tool groups in mm
DBT1	Structure		Current tool data	Input variable. Output variable if no tool description is available
DESCR	Structure		Data transmitted with TOOL cycle or empty	Input and output variable with valid tool description
EDREF%	Integer	0..16	0 for tool description, otherwise 1	Reference cutting edge number to which the tool is calculated relatively. If =0, then with regard to L1, L2, L3, RAD and ORI%.
HOLDER\$	String	STL file name	<DBT1.Hd.SKQ>+"_holder.stl" if available, otherwise "?"	Input/output variable for the holder. No holder if string is empty. In case of question mark, generation according to standard
		Contour definition		
		Empty string		
		Question mark		
L1	real		Correction amount L1 in case of tool description, otherwise 0	Input/output variable for the correction L1 for the tool description
L2	real		Correction amount L2 in case of tool description, otherwise 0	Input/output variable for the correction L2 for the tool description

Variable name	Type	Value range		Pre-assignment	Description
L3	real			Correction amount L3 in case of tool description, otherwise 0	Input/output variable for the correction L3 for the of tool description
NAME\$	String			Name of the tool node in the model	Input variable
ORI%	Integer	0..9		Cutting edge orientation for the tool description, otherwise 0	Input/output variable for cutting edge orientation for the tool description
ORIPLANE\$	String	"G17", "G18", "G19"		After DBT1.Hd.ToolOri. Otherwise "G18" for grinding and turning tools, "G17" for the rest	Input and output variable for the tool orientation
SD	Structure				Input variable with read-only access
SHAFT\$	String	STL file name		<DBT1.Hd.SKQ> + "_shaft.stl" if available, otherwise "?"	Input/output variable for the tool shaft. In case of question mark, generation according to standard
		Contour definition			
		Empty string			
		Question mark			
STOCK	Structure			Current raw part data	Input variable with read-only access to raw part dimensions
SUSPEND?	Boolean	TRUE	No graphic tool change	FALSE	Output variable to control whether to simulate the tool change at the tool node NAME\$ (2)
		FALSE	Graphic tool change		
TOFFL\$	String	"ActPlane", string of X,Y,Z, separated by comma and minus sign, if applicable (e.g. "Z, , -X")		"ActPlane", in the tool description, always a list of axis names	Input/output variable for the assignment of the lengths L1..L3 to the spatial axes X,Y,Z analogously to G78
RAD	real	=>0		Radius correction amount for the tool description, otherwise 0	Input/output variable for the cutting edge radius for the tool description

- (1) CUTTHEIGHT for grinding tool: Length of the replacement presentation in the workpiece simulation as profile bar.
- (2) SUSPEND? For a machine with tool turret: Assign FALSE to this variable if the place number in the current tool data (DBT1.Hd.K2) does not correlate with the name of the tool node NAME\$ or does not match the included place number.

Tab. 8-9: Input/output variables of the "Tool" program section

DESCR variable for drilling tools

Index (type name)	Element name	Type	Value range	Description
--	DEF	String	"/TOOL/<TYPE>/ <P1>{,<P2>..{<Pn>}})\\"	All parameters in one string
			Empty string for no tool description	
--	TYPE	String	"BBAR", "CDRILL", "DRILL", ""	Type name
P1 (DRILL)	D	real	>0	Diameter
P1 (CDRILL)	D1			Small diameter
P1 (BBAR)				Diameter
P2 (DRILL)	A	real	0<A<=180	Apical angle
P2 (CDRILL)	A1		0<A1<=180	
P2 (BBAR)	W		<>0	Cutting edge width
P3 (DRILL)	H	real	>0	Length
P3 (CDRILL)	L		<>0	Step length
P3 (BBAR)	D2		>0	Shaft diameter
P4 (CDRILL)	D2	real	>0	Large diameter
P4 (BBAR)	H			Shaft length
P5 (CDRILL)	A2	real	0<A2<=180	Countersink angle
P5 (BBAR)	L		>=0	Loss length
P6 (CDRILL)	H	real	>0	Total length

Tab. 8-10: Elements of the DESCR input variables for drilling tools

DESCR variable for milling tools

Index (type name)	Element name	Type	Value range	Description
--	DEF	String	"/TOOL/<TYPE>/ <P1>{,<P2>..{<Pn>}})\\" Empty string for no tool description	All parameters in one string
--	TYPE	String	"CHAMFER", "CRMILL", "MILL", ""	Type name
P1 (MILL)	D	real	>0	Diameter
P1 (CHAMFER)				
P1 (CRMILL)	D1			Small diameter
P2 (MILL)	R	real	>=0	Taper angle
P2 (CHAMFER)	A		0<A<=180	Corner/ball radius
P2 (CRMILL)	D2		>0	Large diameter

Index (type name)	Element name	Type	Value range	Description
P3 (MILL)	H	real	>0	Length
P3 (CHAMFER)				
P3 (CRMILL)	R		=>0	Profile radius
P4 (MILL)	A	real	-90<A<90	Taper angle
P4 (CHAMFER)	L		>=0	Chamfer length
P4 (CRMILL)	H		>0	Total length
P5 (CRMILL)	L	real	>0	Length, profile

Tab. 8-11: Elements of the DESCR input variables for milling tools

DESCR variable for grinding tools

Index	Element name	Type	Value range	Description
--	DEF	String	"/TOOL/<TYPE>/ <P1>{,<P2>...{<Pn>}})\\" Empty string for no tool description	All parameters in one string
--	TYPE	String	"GRIND", ""	Type name
--	TY	Integer	1 straight wheel, 2 inclined wheels, 3 face wheels 4 internal wheel	Wheel type
P1	P	Integer	1..402 with interruptions	Profile number
P2	D1	real	<>0	Outer diameter
P3	D2	real	>0	Inner diameter, layer
P4	W	real	>0	Wheel width
P5	A	real	0<A<90	Angle of inclination (TY=2)
P6	H1	real	>=0	Shoulder height left
P7	R1	real	>=0	Radius left
P8	CS1	real	>=0	Width of the chamfer in X left
P9	CA1	real	0<=CA1<90	Angle of the chamfer left
P10	AX1	real	>=0	Back-slope amount in X left
P11	AZ1	real	>=0	Back-slope amount in Z left
P12	A1	real	0<=A1<90	Back-slope angle left
P13	H2	real	>=0	Shoulder height right, face width at TY3
P14	R2	real	>=0	Radius right

Index	Element name	Type	Value range	Description
P15	CS2	real	≥ 0	Width of the chamfer in X right
P16	CA2	real	$0 \leq CA2 < 90$	Angle of the chamfer right
P17	AX2	real	≥ 0	Back-slope amount in X right
P18	AZ2	real	≥ 0	Back-slope amount in Z right
P19	A2	real	$0 \leq A2 < 90$	Back-slope angle right

Tab. 8-12: Elements of the DESCR input variables for grinding tools

The general operating principle of this program section is described in [chapter 8.2 "Introduction to the tool graphic generation" on page 108](#).



With cylindrical tools, it is advisable to assign a reasonable value to the HOLDER\$ variable, as the radial dimension of the holder determines the diameter of generically generated chucks. A valid assignment of the holder variables at runtime is one of the important activities in the simulation application.

Example:

Support of generic graphics and imports

At a milling machining center, use the "Description" element in the DBT1.UD.TI.Descr tool data set to specify STL files, contours or parameters for the tool components in the tool editor:

1. If the description ends with "_h.stl", the model is interpreted as holder including chuck. Cutting edge and shaft are generically created from technology data
2. In all other cases, ".stl" extensions assume that the model includes the entire tool
3. If the description starts with "REV" (followed by "," or ":"), use the contour for the rotary generation of the cutting edge part. It is left to the system to amend the chuck
4. In all other cases, it is assumed that the description contains the parameters to generate the chuck
5. For the holder, declare a shell contour analogously to HSK 63



In case 2, assign the CUTT\$ variable. The system separates the cutting edge part from the mesh using the useful length (technology data) and the remaining part as shaft geometry.

The simulation amends chuck or holder only if the lengths of cutting edge and shaft are not sufficient to fill the length correction amount and if this gap is at least as high as the relevant geometry. Thus, handle the cases 2 and 3 analogously.

Program:

```
<ToolConnection>
  <ToolNode Name="" Channel="1" Function="Regular">
    <Query/>
    <IsToolActive/>
```

```

<Tool>
  IF DBT1.Hd.SKQ<>" THEN      : REM tool not empty
    DESCR$=TRIM$(DBT1.UD.Tl.Descr) : REM description from tool dataset
    IF MCASE$(MID$(DESCR$,LEN(DESCR$)-3,4),"L")=".stl" THEN
      IF MCASE$(MID$(DESCR$,LEN(DESCR$)-5,2),"L")="_h" THEN
        HOLDER$=DESCR$ : ARBOR$="" : REM point (1)
      ELSE
        CUTT$=DESCR$      : REM point (2)
      ENDIF
    ELSE IF (MID$(DESCR$,1,4)="REV:")OR(MID$(DESCR$,1,4)="REV,") THEN
      CUTT$=DESCR$      : REM point (3)
    ELSE IF DESCR$<>" THEN
      ARBOR$=DESCR$      : REM point (4)
    ENDIF ENDIF ENDIF
    ; Holder contour HSK 63, point (5)
    HOLDER$="REV,63,,,4,55,6,,10,63,12,,26"
  ENDIF
</Tool>
</ToolNode>
</ToolConnection>

```

*Example:***Importing different holders at one punching machine**

At a punching machine, there are different holder sizes from A to E to be taken from the "DBT1.UD.PD.Spring_Mode" string variable. For each size, send the suitable STL file for upper and lower part for the holder and assign the relevant value to the CUTTDIA cutting edge diameter.

Program:

```

<ToolConnection>
  <ToolNode Name="ToolUp" Channel="1" Function="Regular">
    <Query/>
    <IsToolActive/>
    <Tool>
      ; Upper tool part
      IF DBT1.UD.PD.Spring_Mode="A" THEN      : REM size A
        HOLDER$="Spring_part_up-die_A.stl" : CUTTDIA=25
      ELSE IF DBT1.UD.PD.Spring_Mode="B" THEN : REM size B
        HOLDER$="Spring_part_up-die_B.stl" : CUTTDIA=47.5
      ELSE IF DBT1.UD.PD.Spring_Mode="C" THEN : REM size C
        HOLDER$="Spring_part_up-die_C.stl" : CUTTDIA=88
      ELSE IF (DBT1.UD.PD.Spring_Mode="D")OR(DBT1.UD.PD.Spring_Mode="E") THEN
        ; Sizes D and E
        HOLDER$="Spring_part_up-die_D.stl" : CUTTDIA=125
      ENDIF ENDIF ENDIF ENDIF
    </Tool>
  </ToolNode>
  <ToolNode Name="ToolDown" Channel="1" Function="DownDie">
    <Query/>
    <IsToolActive/>
    <Tool>
      ; Lower tool part
      IF DBT1.UD.PD.Spring_Mode="A" THEN      : REM size A
        HOLDER$="Spring_part_down-die_A.stl" : CUTTDIA=25
      ELSE IF DBT1.UD.PD.Spring_Mode="B" THEN : REM size B
        HOLDER$="Spring_part_down-die_B.stl" : CUTTDIA=47.5
      ELSE IF DBT1.UD.PD.Spring_Mode="C" THEN : REM size C
        HOLDER$="Spring_part_down-die_C.stl" : CUTTDIA=88
      ELSE IF (DBT1.UD.PD.Spring_Mode="D")OR(DBT1.UD.PD.Spring_Mode="E") THEN
        ; Sizes D and E
        HOLDER$="Spring_part_down-die_D.stl" : CUTTDIA=125
      ENDIF ENDIF ENDIF ENDIF
    </Tool>
  </ToolNode>
</ToolConnection>

```

*Example:***Tool orientation for centric drilling tools**

At a lathe, change the tool orientation from G18 to G17 for drilling tools defined using a tool description. Thus, the interpolation plane G18 can be used for centric drilling, but nevertheless, the graphical orientation is correct. As centric drilling tools, use the "DRILL" and "CDRILL" types.

Configuration



As for tool descriptions, the `TOFFL$` string variable does not receive the "ActPlane" content, the change in the orientation plane never affects the assignment of the length corrections to the spatial directions X, Y and Z.

Program:

```
<ToolConnection>
  <ToolNode Name="*" Channel="1" Function="Regular">
    <Query/>
    <IsToolActive/>
    <Tool>
      IF DESCR.TYPE<>" THEN : REM tool description active
      IF ((DESCR.TYPE="DRILL")OR(DESCR.TYPE="CDRILL"))AND(ORIPLANE$="G18") THEN
        ORIPLANE$="G17" : REM orientation for centric tool
      ENDIF
    ENDIF
  </Tool>
</ToolNode>
</ToolConnection>
```

Punching

Factor for cutting edge height	Tag <PunchToolLenFact>: The factor for the cutting edge height is one part of the cutting edge part thickness of punching tools by multiplying this factor with the sheet thickness. Value range: ≥ 0 . Default is 1.25
Constant cutting edge height	Tag PunchToolLenConst>: The constant cutting edge height in mm is the second part of the cutting edge part thickness of punching tools. Value range: ≥ 0 . Default 20 mm With punching tools, the <code>CUTTHEIGHT</code> variable is initialized with the total of both cutting edge heights.
Z-offset for tool path	Tag ZFightingOffset: The Z-offset for the tool path in mm is a displacement of the line contour over the sheet surface and is used for the better graphical separation. Value range: ≥ 0 . Default 0.03 mm

Generative manufacturing

Activate dimension additions for generative processes	Tag <AddMaterial>: The feature for the activation of size additions for generative production processes defines the validity of the subsequently specified additions for the bounding box of an imported raw part. A material application can only be simulated within the (possibly extended) cuboid. Value range: <code>False</code> - no extension (default), <code>True</code> - additions active
Dimension addition at top	Tag <BoundingBoxExtendTop>: Extension of space for the material application in Z-positive direction in mm. Value range ≥ 0 , default 100 mm
Dimension addition at bottom	Tag <BoundingBoxExtendBottom>: Extension of space for the material application in Z-negative direction in mm. Value range: ≥ 0 , default 0
Dimension addition at the left	Tag <BoundingBoxExtendLeft>: Extension of space for the material application in X-negative direction in mm.

Dimension addition at the right	Value range: ≥ 0, default 0 Tag <BoundingBoxExtendRight>: Extension of space for the material application in X-positive direction in mm. Value range: ≥ 0, default 0
Dimension addition at front	Tag <BoundingBoxExtendFront>: Extension of space for the material application in Y-positive direction in mm. Value range: ≥ 0, default 0
Dimension addition at rear	Tag <BoundingBoxExtendRear>: Extension of space for the material application in Y-negative direction in mm. Value range: ≥ 0, default 0

Camera settings



Use of the augmented reality within the simulation is not generally available and a feature that cannot be purchased. It requires a special DLL. In the officially marketed versions, the subsequently listed settings cannot be used.

Availability of the augmented reality	Tag <Enabled>: Specifies whether the view for augmented reality is applied. Value range: False - no (default), True - yes
Transparency with augmented reality	Tag <Transparency>: Transparency of the simulation graphics in percent shown in the camera image. Value range: 0..100%, default 50%
Background	Tag <Background>: Addresses the camera or specifies a locally stored image or video. Syntax: - File name - the file (image or video) is searched in the folder containing the machine model, e.g. <Background>myStream.jpg</Background> - File name with path specification - the string refers to an image or video file on the local computer - http://<IPv4>:<Port> or tcp://<IPv4>:<Port> - IP address depending on camera type, e.g. <Background>http://10.152.234.231:80</Background> - Device number. Value range ≥ 0, default 0
Lens viewing angle	Tag <VerticalViewingAngle>: Vertical viewing angle of the camera lens in degrees. Value range: $0 < \text{OpeningAngle} < 180$, default 45°
Camera vectors	Tag <CamVectors>: Specifies the camera position and the alignment of the camera (world coordinates). Syntax: <CamX>, <CamY>, <CamZ>, <CenterX>, <CenterY>, <CenterZ>, <UpX>, <UpY>, <UpZ> with CamX - Camera position in X

- CamY - Camera position in Y
- CamZ - Camera position in Z
- CenterX - X-position of a point on the jet from the camera position to the image center
- CenterY - Y-position of a point on the jet from the camera position to the image center
- CenterZ - Z-position of a point on the jet from the camera position to the image center
- UpX - X-component of the unit vector pointing to the image height direction
- UpY - Y-component of the unit vector pointing to the image height direction
- UpZ - Z-component of the unit vector pointing to the image height direction

Example with image center equal to world coordinate origin and upright camera assembly:

```
<CamVectors>-500,-1500,1500,0,0,0,0,0,1</CamVectors>
```

Horizontal shearing

Tag <XShearing>:

Factor of the horizontal image shearing.

Value range: -1..1, default 0

Vertical shearing

Tag <YShearing>:

Factor of the vertical image shearing.

Value range: -1..1, default 0



The setup of the camera also includes a camera calibration file. Save it as "camera_calibration_file.xml" in the folder with the machine model.

8.3.3 Simulation options

Optimization

Sampling interval

Tag <SamplingInterval>:

Specifies the duration between two axis coordinate value pairs to be simulated in milliseconds.

Value range: >0, default 40 ms

Reduce sampling rate automatically?

Tag <AutoAdaptSamplingRate>:

Specifies whether individual axis coordinate value pairs can be excluded from the simulation in an online simulation to keep the simulation lag below one second. Higher computer load can cause a simulation tool path considerably deviating from reality.



Generally, assign `False` to `AutoAdaptSamplingRate`. Try to reduce the lagging using other performance-improving actions.

Value range: `True` - minimize lagging, `False` - simulate all path points (default)

Render interval

Tag <RenderInterval>:

Specifies the duration between two image refreshes in milliseconds.

	Value range: >0, default 50 ms
Triangulation interval	Tag <TriangulationInterval>: Specifies the period between the triangle areas to be rendered from the workpiece models in milliseconds. Value range: >0, default 100 ms
Number of pixels in space	Tag <ModelSamplesPerDiagonal>: Specifies the number of room pixels per diagonal of the cuboid enveloping the raw part. This setting decisively affects the graphical quality, the memory consumption and the performance of the simulation. Value range: 5..800, for sheets up to 1500, standard 400 (for medium quality)
	 Note that the memory requirements can increase with the cube of the pixels in space and that the simultaneous display of multiple workpieces can also increase the memory consumption. Thus, set the setting to the lowest value, but ensure an acceptable quality, especially if an OutOfMemory exception was output.
Display quality of the workpiece surface	Tag <StockMeshQualityDuringSimulation>: For the graphical presentation (rendering), the workpiece model is converted into triangles (meshes). The setting specifies the quality level of this process. Value range: 1 - highest graphical quality (default), 2 - medium quality, 3 - fastest rendering
Time interval to improve the display quality of the workpiece surface	Tag <StockMeshQualityAutoRefineDuration>: Use a time interval > 0 to change the quality level of the graphic representation of the workpiece model (specified by tag <StockMeshQualityAutoRefineDuration>) to the highest quality level. During the movement, the quality is reduced again. If 0 is specified, it is drawn immediately with the highest quality.
Facet tolerance	Tag <FacetTolerance>: Converting a bent area into a mesh necessarily results in a deviation between the ideal geometry and the model geometry. The setting specifies the maximum deviation in mm. Value range: >0, default 0.05 mm
Facet tolerance during the workpiece export	Tag <ExportTolerance>: When exporting the workpiece geometry as STL file, the deviation of the mesh from the internal representation of the workpiece model may be changed to simplify the model. The setting specifies the maximum deviation in mm. Value range: >=0, default 0.05 mm
Tool path tolerance	Tag <ToolpathTolerance>: To save memory space and to increase the performance, subsequent tool path sections running almost parallel can be combined in one section. The setting specifies the maximum deviation between the "original" tool path and the summarized sections in mm. Value range: >=0, default 0.01 mm
Number of stored, block-related information	Tag <ToolpathMaximumMarkerNumber>: For every section of the tool path, information on the NC block can be saved. The setting specifies the maximum number of stored block information. When the maximum number is exceeded, the oldest information is lost.

Configuration

	Value range: >0, default 100,000
Show edges everywhere?	Tag <ShowEdgesEverywhere>: The presentation of body edges reduces the performance. If the edge presentation is enabled (menu function), the setting determines whether with the edges are shown at all geometry elements or whether the edge presentation is limited to the workpiece. Value range: <code>False</code> - Edges only visible at the workpiece (default), <code>True</code> - Show edges at all elements
Removing separated volumes	Tag <ChunkDetection>: The simulation can optionally remove completely separated areas of the workpiece (waste or finished workpieces). The setting defines when separated volumes are to be removed. Value range: • <code>None</code> - never (default) • <code>ZoomHQ</code> - manually when improving the workpiece (has to be applied) • <code>Auto</code> - automatically (only suitable for turning) • <code>Timer</code> - in fixed time intervals
Time interval to remove separated volumes	Tag <ChunkDetectionInterval>: Specifies the related time interval for <ChunkDetection><code>Timer</code></> <ChunkDetection> in milliseconds. Value range: >0, default 2,500 ms
Renew workpiece upon program start?	Tag <RenewStockAtProgStart>: Specifies whether - upon program start - workpieces machined or created by the channel are replaced automatically by unmachined workpieces. Value range: <code>False</code> - do not renew (default), <code>True</code> - renew
Save workpiece at the end of the session?	Tag <SaveMachinedStock>: Specifies whether the current processing state of the workpieces is saved when exiting IndraWorks Operation to continue with the latest machining state upon the interface restart. With complex workpieces, saving delays exiting the interface. Value range: <code>True</code> - save (default), <code>False</code> - unmachined workpieces after the interface start
Multiple workpieces per workpiece holder	Tag<AllowMultipleStocks>: This setting defines whether this workpiece is replaced by the available one or added using the dialog-guided definition or the import of a workpiece. This setting does not refer to the import or the definition using cycles and does not refer to other geometry types. Value range: <code>False</code> - Replace workpiece (default), <code>True</code> - Add workpiece
Collision detection	Tag <CollisionDetection>: Specifies whether the simulation indicates collisions by coloring the collision partners in red and issuing a message in the status bar. Value range: • <code>None</code> - No collision detection • <code>NoMachiningHistory</code> - simple collision detection (default setting). The collision partners become temporarily red. A message is displayed in the status bar and the simulation can optionally be stopped automatically after the collision

- **Full** - Highest configuration level of the collision detection. Machined workpiece areas are additionally and permanently colored in red at rapid traverse



Full saves the complete preceded program sequence up to the workpiece renewal (machining history) and consumes significantly more memory than **NoMachingHistory**. Thus, avoid this setting for the part program spectrums with long runtimes, especially when an **OutOfMemory** exception results.

Folder for collision report files

Tag `CollisionReportsFolder`:

Names the folder used to automatically store collision reports.

Value range: Folder names in the file system of the NC or on the local PC.

Default: `/mnt/CollisionReports`

Change in view with full-screen presentation

Tag `<FitViewWhileChanging>`:

Enabling this setting provides a full-screen presentation of the graphic after every view toggling using a menu function.

Value range: `True` - full screen (default), `False` - enlargement remains

Locking orthogonal views

Tag `<LockOrthogonalViews>`:

This setting blocks the orthogonal views (top view, from right, etc.) against twisting and is especially recommended for sheet machining.

Value range: `False` - Twisting always permitted (default), `True` - Twisting only possible in 3D view.

Automatic zooming

Tag `<AutoZoom>`:

With this setting, it is changed automatically to full screen, once the bounding box around the scene to be shown became bigger with program progress

and if the key  (with two switching positions) is pressed.

This setting is recommended for projects that to not use workpiece definitions, i.e. with a preferred use of the tool path representation.

Value range: `False` - No automatic zoom (default), `True` - Automatic zoom can be enabled using the full screen key

Settings for display

Initial setting axes/scales

Tag `<CoordSystemVisible>`:

Represents the initial setting on the visibility of a coordinate system. The display is switched using the "Axes/scales" menu function.

Value range:

- 0 - No coordinate system visible
- 1 - Show coordinate tripod for the visualization of the axis directions (default)
- 2 - Show scales for the dimensional orientation
- 3 - Show coordinate tripod and scales

Main view

Tag `<MainView>`:

The "Main view" setting is currently not used.

Value range:

- `Iso` - Non-orthogonal view

Configuration

	• Front - Front view • Rear - Rear view • Left - View from left • Right - View from the right • Top - Top view • Bottom - Bottom view • Iso - Second non-orthogonal view
Opening angle of the camera	Tag <VerticalViewingAngle>: Defines the camera angle (opening angle of the objective) in the horizontal of the image for the perspective view, also called parallel projection. Value range: >0, default 40°.
Workpiece preview	Tag <WorkpiecePreview>: Specifies the behavior of the workpiece preview. Value range: • No - No workpiece preview (default) • CreateOnly - The preview is only used for display during the selection of a program in the navigator and in the program selection • ToolpathOnly - The workpiece preview includes only the toolpath and cannot be displayed in the IndraWorks Operation navigator. • Full - The preview is moreover automatically provided and/or displayed in the simulation as finished part geometry in the program selection and thus replaces the import via dialog or the STOCKIMP cycle
Create workpiece preview after each program end?	Tag <CreateWorkpieceOnce> Specifies whether a workpiece preview is created after each successful program run. This process negatively affects the simulation performance. Value range: False - always create again (default), True - create only if the preview is not available anymore or if it is older than the main program.
Tool path	Tag <ToolpathEnabled>: Defines the visibility of the tool path graphic. Value range: • None - No tool path visible • Tracking - The tool path is visible as line draft (default) • TrackingAndBlockinfo - Program name and NC set to the tool path graphic are saved and used for the display a tool tip and/or for opening the editor This setting has not been implemented yet. Its effect is analogous to tracking.
	 The tool path increases the memory requirement and reduces the performance of the graphical representation. Thus, avoid the tool path graphic for the part program spectrums with high runtimes, especially when an OutOfMemory exception results. In these part programs, the tool path graphic is mostly insignificant, as the colored area is shown.
Improve workpiece	Tag <HighQualityZooming>: Provides one option to subsequently improve the workpiece graphic.

Value range:

- **None** - The "Improve workpiece" function is not available
- **DynamicZoom** - When zooming, the simulation of the previous machining for the shown section is repeated internally with higher resolution. The smaller the size of the section, the higher the resolution and the better the quality. After changing the image section, the improvement is lost and does not require memory anymore. The performance load is thus reduced
- **AutoRefine** - Accompanying the simulation and depending on the load, the simulation kernel determines workpiece areas that need to be improved and calculates a more detailed graphic for them. The improvements are permanent and visible at the entire workpiece, even in case of full-screen presentation. The achieved quality improvement is limited



The high quality zooming saves the complete preceded program sequence up to the workpiece renewal (machining history) and consumes significantly more memory. Thus, avoid this setting for the part program spectrums with long runtimes for multiple hours, especially when an OutOfMemory exception results. Additionally, the time required to repeat the program increases proportionally to the program runtime.

Color gradient on reflecting workpiece surface

Tag <ReflectionBitmap>:

The reflection of incident ambient light through the workpiece surface leads to different brightness levels and colors. These color gradients at the workpiece can be mapped by a superimposition with a bitmap file. A reflectance greater than zero is assumed.

An activation reduces the color contrast and may be unfavorable under certain circumstances.

Value range: **True** - simulate color gradient due to reflection of the ambient light (default), **False** - uniform color gradient on a plane surface

Reflectance

Tag <StockReflectivity>:

Defines the portion of the light (shine) reflected by the workpiece surface in percent.

Value range: 0..100, default 25%

Workpiece transparency

Tag <StockTransparency>:

Defines the workpiece transparency in percent. The setting does not apply for Augmented Reality and not for the simultaneous presentation of the finished part geometry or command/actual comparison.

Value range: 0..100, default 25%

Transparency of the workpiece with internal finished part

Tag <StockTransparencyShowingTarget>:

Defines the workpiece transparency in percent if the finished part geometry is visible at the same time.

Value range: 0..100, default 30%

Transparency of the body edges

Tag <EdgeTransparency>:

Defines the transparency of the lines for body edges in percent. Too "dominant" body edges can be avoided with this setting as it can be observed with some graphic cards.

Value range: 0..99, default 0%

Transparency of the machine housing

Tag HousingTransparency:

Defines the transparency of all unmovable machine parts and doors applied at the first stage of hiding.

Value range: 0..100, default 0%

Colors

Syntax for colors Colors are defined via their red, green and blue contents without alpha value (transparency). There are two equal syntax variants:

- Hexadecimal: #<RR><GG><BB> with RR, GG, BB for red, green and blue in the value range from 00 (no illumination) to FF (maximum illumination). Specify every color component as two-digit figure and with a leading zero, if required.

Example: #0AEF67

- Decimal: RGB(<r>,<g>,) with r, g, b for red, green and blue in the value range from 0 to 255. No leading zeros are required.

Example: RGB(10,239,103)

Background color **Tag** <Background>:

Specifies the color of the background in the image center. The color gradient cannot be modified.

Default: #F6F6F6

Color of the tool edge **Tag** <CuttingPart>:

Specifies the color of the cutting edge part of the tools if the scheme for changing colors was enabled.

Default #8C8C00

Color of the tool shaft **Tag** <Shaft>:

Specifies the color of the tool shaft.

Default #A9A9A9

Color of the tool chuck **Tag** <Arbor>:

Specifies the color of the chuck of the tools.

Default #D3D3D3

Color of the tool holder **Tag** <Holder>:

Specifies the color of the tool holder.

Default #808080

Color for tool path rapid traverse **Tag** <ToolpathRapid>:

Specifies the color of the tool path at rapid traverse (G0) or passive movements during punching.

Default RGB(0,130,0)

Color for tool path feed **Tag** <ToolpathFeed>:

Specifies the color of the tool path in case of feed movements or forming movements during punching.

Default RGB(206,0,24)

Color of tool path preview **Tag** <ToolpathPreview>:

Specifies the color of the tool path as part of the workpiece preview.

Default #CDAA7D

Color for workpiece clamping fixtures **Tag** <Fixture>:

Specifies the color of workpiece clamping fixtures.

	Default #FFA500
Color for highlighting	Tag <Highlighting>: Specifies the color used to graphically highlight or mark the user interaction elements. Default #FF00FF
Raw part color	Tag <Stock>: Specifies the color for non-machined raw part surfaces except for the target/actual comparison. Default #4682B4
Color for machined surfaces	Tag <RemovedMaterial>: Specifies the color of the machined workpiece surfaces, except for the target/actual comparison. Default #FFFFFF0
Finished part color	Tag <TargetGeometry>: Specifies the color of the finished part. Default #FFE4B5
Finished part drawing	Tag <TargetDrawing> Specifies the color for two-dimensional drawings imported as finished part from DXF files. Default #303030
Color for body edges	Tag <Edges>: Specifies the color for body edges and symbols. Default #373F43
Color for wire models	Tag <Wireframe>: Specifies the color of raw parts as enveloping body and drawings. Default #0000FF
Color for cutting areas	Tag <SectionFace>: Specifies the color of cutting areas (sectional view). The color is not used yet.
Color scheme for target/actual comparison	Tag <DeviationColorScheme>: Target/actual comparison between the workpiece and the finished part geometry significantly increases the CPU load. Enable it with the attribute <code>EnableDeviationMode="True"</code> . For the target/actual comparison of the workpiece with the finished part geometry (deviation mode), one color each is assigned to the different intervals of the dimensional deviation. The color value specifies whether the command geometry was violated (waste), reached or whether there is an allowance. The upper limit of every interval is defined in the <code>deviation</code> attribute. The upper limit of the highest interval is always evaluated as plus infinite. The lowest interval starts at minus infinite. All other intervals start at the value of the <code>deviation</code> attribute of the previous interval (with the next smaller attribute). Syntax: <code><SchemeColor Deviation="<Upper limit>"><Color></SchemeColor></code> with upper limit equal to upper interval limit in mm and the color value <code><Color></code> in decimal or hexadecimal presentation.

Configuration

The intervals do not have to be specified by size. A valid color scheme consists of at least three intervals.

Default:

Program:

```
<DeviationColorScheme EnableDeviationMode="False">
  <SchemeColor Deviation="-0.1">#800000</SchemeColor>
  <SchemeColor Deviation="-0.05">#FF0000</SchemeColor>
  <SchemeColor Deviation="-0.025">#FF8000</SchemeColor>
  <SchemeColor Deviation="-0.01">#FFFF00</SchemeColor>
  <SchemeColor Deviation="0.01">#008000</SchemeColor>
  <SchemeColor Deviation="0.025">#00FFFF</SchemeColor>
  <SchemeColor Deviation="0.05">#0080FF</SchemeColor>
  <SchemeColor Deviation="0.1">#0000FF</SchemeColor>
  <SchemeColor Deviation="9999.999">#000080</SchemeColor>
</DeviationColorScheme>
```

Scheme for changing cutting edge colors

Tag <CutterColorScheme>:

Alternatively to the fixedly specified colors for the tool edge and for machined areas, changing colors can be activated with are used for the cutting edge and the machined areas. The colors at the workpiece allow conclusions on the tool used. To activate the color change, set the attribute UseCutterColors="True".

At least three colors are required for a valid color scheme. Their order determines the order of their use.

Syntax: <SchemeColor><Color></SchemeColor> with the color value <Color> in decimal or hexadecimal presentation.

Default:

Program:

```
<CutterColorScheme UseCutterColors="False">
  <SchemeColor>#EEE5DE</SchemeColor>
  <SchemeColor>#97FFFF</SchemeColor>
  <SchemeColor>#FFFF00</SchemeColor>
  <SchemeColor>#FFB6C1</SchemeColor>
  <SchemeColor>#54FF9F</SchemeColor>
  <SchemeColor>#F5F5DC</SchemeColor>
  <SchemeColor>#FFBFFF</SchemeColor>
  <SchemeColor>#ADFF2F</SchemeColor>
  <SchemeColor>#FFA07A</SchemeColor>
  <SchemeColor>#F0E68C</SchemeColor>
</CutterColorScheme>
```

9 Integrating the MTX simulation control into third-party applications

9.1 Preparation

Contents Using a sample application, the following section describes the possibility to integrate the MTX simulation control into third-party applications. The related C# project is part of the IndraWorks installation and can be used for the integration in a customer application by modifying or copying code. It implements all relevant operating functions.

- Work process**
- *An IndraWorks installation "OPD" or "Workstation" of the version MTX 15V02 or higher is available*
 - *You must have administrator rights*
 - *For the test, the Microsoft Visual Studio 2012 or higher must be installed*
 - *If not available, please install an archiving program, e.g. <http://www.7-zip.de/download.html>*
 - *Ensure that the current visualization data have been transmitted (IndraWorks Engineering ► Visualization device node ► Context menu ► Transmit visualization data)*
 - *The simulation control uses the ComponentOne C1FlexGrid. If the definition of simple raw parts via the control-inherent dialogs is to be used in your application, your application has to be built with a valid developer license (<http://www.componentone.com>). Otherwise, the dialog will be preceded by a message box which does, however, basically not impair the function*
1. Click on the shortcut "_LinkToCommonAppData" in the IndraWorks installation folder in the file explorer. Unzip the archive "Simulation\MTX.Simulation.SampleAppProject.zip" below the IndraWorks installation folder to the folder "CommonAppData\Simulation". If required, create that folder.
You will get a C# project with the complete build environment for a stand-alone application embedding the online simulation
 2. Open the file "IndraWorks\CommonAppData\Simulation\MTX.Simulation.SampleAppCustom.sln" with the Microsoft Visual Studio
 3. Build the application "MTX.Simulation.SampleApp.exe".
The specified application is stored in the IndraWorks installation directory
 4. For the debugging, enter the program "IndraWorks\MTX.Simulation.SampleApp.exe" that is to be started externally in the program properties (page "Debug"). The IndraWorks installation directory is the working directory

9.2 Behavior of the simulation control

Multiple instances One process can only be visualized by one simulation at a time. This limitation is due to the fact that the NC kernel is only able to communicate with one single client via an OPC UA real-time item.



The OPC UA connection to the NC kernel prevents you from using IndraWorks Operation with applied simulation and its interface or the sample application at the same time.

It is, however, still possible to instantiate several instances of the `Rexroth.MTX.SimulationACI.Ctrl.SimulationACI.Ctrl` control (e.g. with a Windows MDI application). The actual simulation control inside which is not available via interfaces, is then changed back and forth between the different simulation ACI controls, depending on which one is currently visible.

To activate a control, set the property of the simulation control `HMI.ACI.Interface.IACI.Active=true`. When it is called for the first time, this property triggers the generation of the internal simulation control and establishes the connection to the NC kernel. If this property is set to `true` in another simulation control, the internal simulation is transmitted to this control which enables the drawing (triangulation).

`IACI.Active=false` disables the triangulation. Please use this property if the embedding window is invisible in order to keep the influence of the simulation on the performance to a minimum.

Communication with the NC kernel

The control automatically maintains the communication with the NC. So before the first activation, you have to set the Device-Guid with `IACI.Device`.

Operating commands

Operation is effected using the `IACI.InvokeCommand()` method, to which strings have to be transferred in the argument. The stock of commands is defined in the `SimulationACI.Ctrl.FunctionName` class.

Status information

The simulation informs the client about texts to be displaced (e.g. on a status bar) and the status of menu functions. To this end, register with the EventHandler `SimulationACI.Ctrl.Notify(string sMethod, string[] sParameters)`.

The feedbacks are text-based:

1. Argument `sMethod = "SetStatusbarMessage"`:

Display in the status bar

- The `sParameters[0]` argument specifies the message type:
 - `"SetStatusText"`: Status message (can remain)
 - `"SetErrorText"`: Error message (has to be deleted if required)
 - `"SetInformationText"`: Information (should be deleted after expiry of a time interval)
- The `sParameters[1]` argument contains a message text.
The text is displayed in the current national language of the interface.

2. Argument `sMethod = "FunctionStateChanged"`:

Information on the change in state of a command to adjust the menu buttons.

- The `sParameters[0]` argument contains the command name.
The string corresponds to the definitions in the `FunctionName` class.
- `sParameters[0]` argument informs on the new state:
 - String corresponding to the definitions in the `SimulationACI.Ctrl.FunctionState` class.

- The most important states are `FunctionState.ON`, `FunctionState.OFF` and `FunctionState.DISABLED` – Function active/pressed, function inactive/off and function not available
- Some functions have more than two switching positions and they are fed back using corresponding strings (defined in the `FunctionState` class). The strings can be used for changing key labels, graphics or other status displays
- The initial state of every function is `FunctionState.OFF` and is not notified

Hiding the control-inherent shortcuts

To do so, use either the `IACI.InvokeCommand()` method with the `FunctionName.SHOW_UI_ELEMENTS` method and moreover include the visibility in a command argument.

Exiting the application and connection interruption

In order to give the internal control of the simulation the possibility to close the current threads and save the current simulation state, you should call the `InvokeCommand()` method using the `FunctionName.TERMINATE` command when closing your application.

If the connection is interrupted, also call this command. Upon re-establishment of the connection, the internal control is newly generated with `IACI.Active=true`.

9.3 Sample project

Screenshot

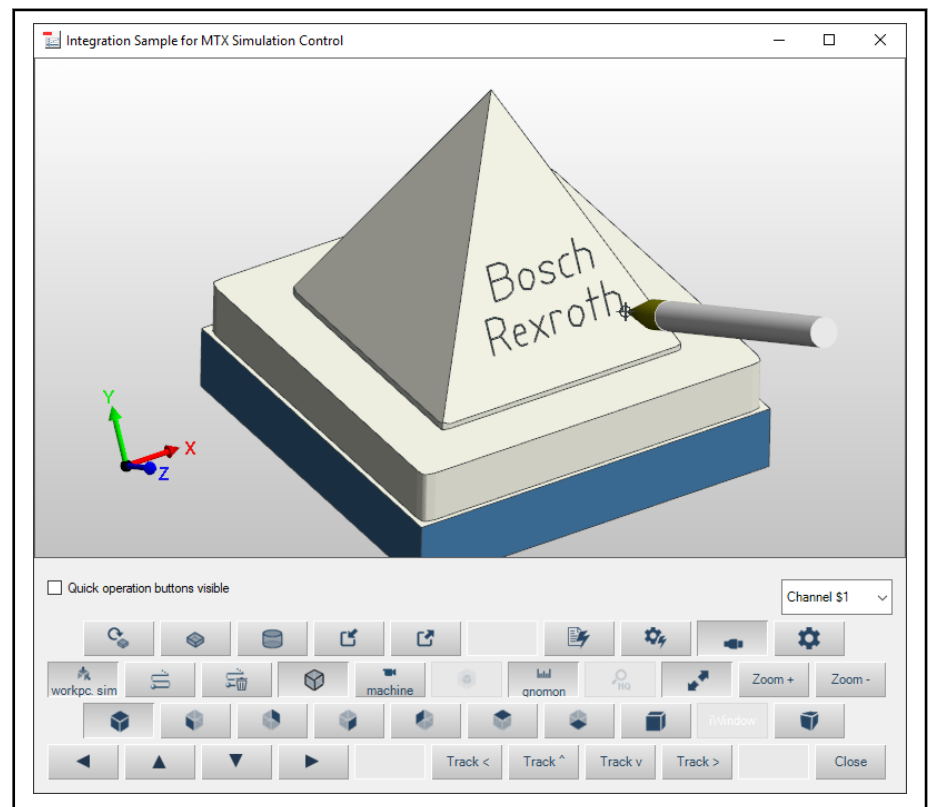


Fig. 9-1: User interface of the sample application for integrating the MTX simulation control

DeviceConnection.cs

The `DeviceConnection` class ensures that the sample application can be executed in the IndraWorks environment.

Essential functions:

- Determines the Device-Guid of the device to be simulated
- Informs on the connection status, enables all resources if the connection is interrupted
- Informs on the channel active at the interface and sets it

The code is not important for the integration but only for the completion of the example to a working application. As developer of your own user interface, you will certainly already have an adequate implementation for this part and thus, the code is not addressed in more detail here.

SampleForm.cs

Here, the exemplary integration of the simulation control into a Windows form takes place.

The following **namespaces** are required:

Program:

```
using Rexroth.MTX.SimulationACICtrl;
using Rexroth.HMI.ACI.Interface;
```

MainForm_Load() contains all relevant commands including a short explanation. In this connection, refer to the initialization of the string[] command field at the beginning of the method.

These instructions are of major importance in the load event for instantiating the simulation code:

Program:

```
_simCtrl = new SimulationACICtrl(); // Instantiation as online simulation
_simCtrl.Notify += _simCtrl.Notify; // Login for status messages
pnlSim.Controls.Add(_simCtrl);      // Set the simulation control as
                                     // child of another control
_simCtrl.Dock = DockStyle.Fill;
```

It is recommended to arrange the simulation control as (only) child of a control (here panel).

The commands are subsequently stored in the tag of very menu button to easily determine the related command in the click event from the transmitter. Thanks to the _buttonPerCommand member variable, it will later be possible to draw conclusions from the FunctionName name to the related button.

In the SampleForm_Shown() call-back, the following is programmed which leads to the instantiation of the internal simulation control and the establishment of the connection:

Program:

```
_simCtrl.Active = true;
```



It is important that you only set `_simCtrl.Active = true` when the main window of your application has already been created and a handle has been assigned.

The following methods demonstrate what has to be done in case of connection interruption and in case the application is closed (e.g. executing the FunctionName.Terminate command).

Program:

```
void MainForm_FormClosing(object sender, FormClosingEventArgs e)
void Terminate()
void DeviceConnection_DeviceConnected(object sender, EventArgs e)
void DeviceConnection_DeviceDisconnected(object sender, EventArgs e)
```

The following method shows how the suitable command is called for the simulation from a click event by means of the `_buttonPerCommand` variable. The method to execute the command is called `IACI.InvokeCommand()`.

Program:

```
void btnMenu_Click(object sender, EventArgs e)
```

The following line contains the implementation for the visible/invisible setting of the shortcuts of the control.

Program:

```
void checkBoxBtnVisibility_CheckedChanged(object sender, EventArgs e)
```

This method demonstrates how from the notify event of the simulation, the appearance of the **menu** buttons can be updated and the output of status texts can be realized.

Program:

```
void _simCtrl_Notify(string sMethod, string[] sParameters)
```

10 Service and support

Our worldwide service network provides an optimized and efficient support. Our experts offer you advice and assistance should you have any queries. You can contact us **24/7**.

Service Germany Our technology-oriented Competence Center in Lohr, Germany, is responsible for all your service-related queries for electric drive and controls.

Contact the **Service Hotline** and **Service Helpdesk** under:

Phone: **+49 9352 40 5060**
Fax: **+49 9352 18 4941**
E-mail: service.svc@boschrexroth.de
Internet: <http://www.boschrexroth.com>

Additional information on service, repair (e.g. delivery addresses) and training can be found on our internet sites.

Service worldwide Outside Germany, please contact your local service office first. For hotline numbers, refer to the sales office addresses on the internet.

Preparing information To be able to help you more quickly and efficiently, please have the following information ready:

- Detailed description of malfunction and circumstances
- Type plate specifications of the affected products, in particular type codes and serial numbers
- Your contact data (phone and fax number as well as your e-mail address)

Index

A

About this documentation.....	1
Information representation.....	6
Validity of the documentation.....	1
ANSI Z535.6-2006.....	6

B

Bar section (turning).....	63
Boring bar, backward counterbore.....	83
Button block "Details".....	23
Button block "Display status".....	20
Button block "Representation".....	21
Button block "Tools".....	28
Button block "View".....	23

C

Clamping device definition.....	54
Clamping device definition via NC command.....	55
Clamping device definition via STL file import....	55
Clamping device definition via the machine model.....	54
Commission.....	39
Commissioning	
Procedure.....	39
Commissioning the NC Simulation	
Machine definition.....	40
Resource definition.....	48
Tool definition.....	48
Complaints.....	8
Configuration file.....	42
Criticism.....	8
Customer Feedback.....	8
Cylindrical raw part (milling).....	61
Cylindrical, angular and ball cutter.....	85

D

Defining the machine type.....	42
Definition of the tool holder geometry (optional).....	47
Documentation	
Change record.....	1
Drill, countersink, reamer.....	80

E

Engraving milling cutter, chamfer milling cutter, face milling head.....	88
Extended operating keyboard of the simulation.....	20

F

Feedback.....	8
---------------	---

G

Grinding wheel.....	93
---------------------	----

H

Hazard warnings.....	6
Helpdesk.....	153
Hotline.....	153

I

Import model.....	70
Information representation	
Names and Abbreviations.....	8
Intended use	
Areas of application.....	9
Introduction.....	9
Use cases.....	9

K

Kinematic model.....	40
----------------------	----

M

Machine definition.....	40
Machine model.....	40

N

NC program.....	55
NC simulation	
Hide machine parts.....	143
Integration into third-party applications.....	147
NC Simulation	
Auto refine.....	24, 143
Automatic zoom.....	24
Automatic zooming.....	141
Basic license.....	11
Collision detection.....	25, 140
Collisions.....	25
Configuration.....	35, 105
Cylindrical raw part.....	32
Dynamic zoom.....	24, 143
Edges.....	22, 140, 143, 145
Export.....	35
Finished part.....	16, 22, 143
Full screen.....	24
Hide machine parts.....	21
Import.....	33
Improve workpiece.....	24, 142
Layout.....	15
Lock orthogonal views.....	141
Operation.....	19
Options.....	35, 105
Perspective view.....	21, 142
Player.....	57
Problem analysis (debug mode).....	57, 123
Raw part cuboid/sheet.....	30
Recording and playing recordings.....	57
Reset raw part.....	28, 36, 140
Shortcuts.....	36
Simulation virtual machine.....	11

Index

Target-actual comparison.....	16	Symbols.....	8
Target/actual comparison.....	145	T	
Tool path.....	11, 21, 109, 136, 139, 142, 144	Target groups.....	1
Transparent enveloping body as raw part.....	31	Technology-dependent settings.....	47
Workpiece preview.....	16, 142, 144	Tool definition.....	48
O		Tool definition via NC command.....	49
Operation.....	19	Tool definition via tool management.....	48
Overview.....	11	TOOL/BBAR.....	83
P		TOOL/CDRILL.....	81
Placement.....	56	TOOL/CHAMFER.....	88
Polygon (turning).....	65	TOOL/CRMILL.....	91
Program definition.....	55	TOOL/DRILL.....	80
Program processing with tool management.....	55	TOOL/GRIND.....	93
Program processing without tool management.....	56	TOOL/MILL.....	85
Q		U	
Quarter round profile milling cutter.....	91	Unintended use.....	10
R		Consequences, non-liability.....	9
Raw part definition.....	50	W	
Raw part definition via definition dialog.....	51	Warnings.....	6
Raw part definition via NC command.....	52	Z	
Raw part definition via STL file import.....	53	Zero offset.....	56
Raw part definition via the machine model.....	51		
Rectangular raw part (milling).....	59		
Referencing axes from kinematic model.....	43		
Referencing the machine model.....	42		
Referencing the ToolNodes.....	45		
Remove workpieces/models.....	73		
Renew workpiece.....	74		
Replace/add workpieces/models.....	77		
Resource definition.....	48		
S			
Safety instructions.....	6		
Select workpiece holder.....	76		
Service hotline.....	153		
Sheet (punching).....	68		
Shortcuts of the simulation.....	19		
Signal alert symbol.....	6		
Signal words.....	6		
Step drill, centering drill.....	81		
STOCK/BAR.....	63		
STOCK/BLOCK.....	59		
STOCK/CYLINDER.....	61		
STOCK/POLYGON.....	65		
STOCK/REMOVE.....	73		
STOCK/RENEW.....	74		
STOCK/SHEET.....	68		
STOCKADD.....	77		
STOCKIMP.....	70		
STOCKSEL.....	76		
Suggestions.....	8		
Support.....	153		

Notes

Bosch Rexroth AG

P.O. Box 13 57

97803 Lohr a.Main, Germany

Bgm.-Dr.-Nebel-Str. 2

97816 Lohr a.Main, Germany

Phone +49 9352 18 0

Fax +49 9352 18 8400

www.boschrexroth.com/electrics



R911399601