

IndraMotion MTX 15VRS

NC Simulation

Application Description
R911393273

Edition 01



Title	IndraMotion MTX 15VRS NC Simulation
Type of Documentation	Application Description
Document Typecode	DOK-MTX***-NC*SIM**V15-AP01-EN-P
Internal File Reference	RS-c0451d84a49cd816c0a802867a8c1037-1-en-US-5
Change Record	Edition 01, 2019-03 Refer to tab. 1-1 "Change Record" on page 3
Copyright	© Bosch Rexroth AG 2019 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 (MS/PiaSt)

Table of Contents

	Page
1 About this documentation.....	3
1.1 Validity of the documentation.....	3
1.2 Notation, fonts, symbols.....	4
1.3 Requirements.....	4
1.4 Required and supplementing documentation MTX.....	4
1.4.1 Selection/Compilation.....	4
1.4.2 Configuration.....	5
1.4.3 Commissioning.....	5
1.4.4 Operation.....	6
1.4.5 OEM Engineering.....	7
1.4.6 AddOns.....	7
1.5 Information representation.....	8
1.5.1 Using safety instructions.....	8
1.5.2 Names and abbreviations.....	10
1.6 Customer feedback.....	10
2 Important instructions on use.....	11
2.1 Intended use.....	11
2.1.1 Introduction.....	11
2.1.2 Areas of use and application.....	11
2.2 Unintended use.....	12
3 Function overview.....	13
4 Starting and exiting the simulation.....	17
5 Layout.....	19
6 Operation with keys and corresponding dialogs.....	21
6.1 General information.....	21
6.2 Main level of the simulation.....	21
6.3 Views menu.....	21
6.4 Rotate menu.....	22
6.5 Move menu.....	22
6.6 Details menu.....	23
6.7 Display menu.....	24
6.8 Tools menu.....	25
7 Shortcuts.....	35
8 Recording and playing recordings.....	37

Table of Contents

	Page
9 Program instructions for simulation.....	39
9.1 General information about program instructions for simulation.....	39
9.2 STOCK/BLOCK Cuboid raw part (milling)	39
9.3 STOCK/CYLINDER Cylindrical raw part (milling)	41
9.4 STOCK/BAR Bar section (turning).....	43
9.5 STOCK/POLYGON Polygon (turning).....	45
9.6 STOCK/SHEET Sheet (punching).....	47
9.7 STOCKIMP Import model.....	48
9.8 STOCK/REMOVE Remove workpieces/models.....	51
9.9 STOCK/RENEW Renew workpiece.....	53
9.10 STOCKSEL Select workpiece holder.....	54
9.11 STOCKADD Replace/add workpieces/models.....	55
9.12 TOOL/DRILL Drill, countersink, reamer.....	56
9.13 TOOL/CDRILL Step drill, centering drill.....	57
9.14 TOOL/BBAR Boring bar, backward counterbore.....	59
9.15 TOOL/MILL Cylindrical, angular and ball cutter.....	61
9.16 TOOL/CHAMFER Engraving milling cutter, chamfer milling cutter, face milling head.....	64
9.17 TOOL/CRMILL Quarter round profile milling cutter.....	67
9.18 TOOL/GRIND Grinding wheel.....	69
10 Configuration.....	81
10.1 Overview.....	81
10.2 Introduction to the tool graphic generation.....	84
10.3 Description of the configuration file MTX.Simulation.Options.XML or MTX.Simulation.Options.Temp.XML.....	96
10.3.1 File structure.....	96
10.3.2 Configuration of the NC simulation.....	98
10.3.3 Simulation options.....	113
11 Integrating the MTX simulation control into third-party applications.....	121
11.1 Preparation.....	121
11.2 Behavior of the simulation control.....	121
11.3 Sample project.....	123
12 Service and support.....	127
Index.....	129

1 About this documentation

Editions of this documentation

Edition	Release Date	Note
01	2019-03	First edition

Tab. 1-1: Change Record

1.1 Validity of the documentation

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

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

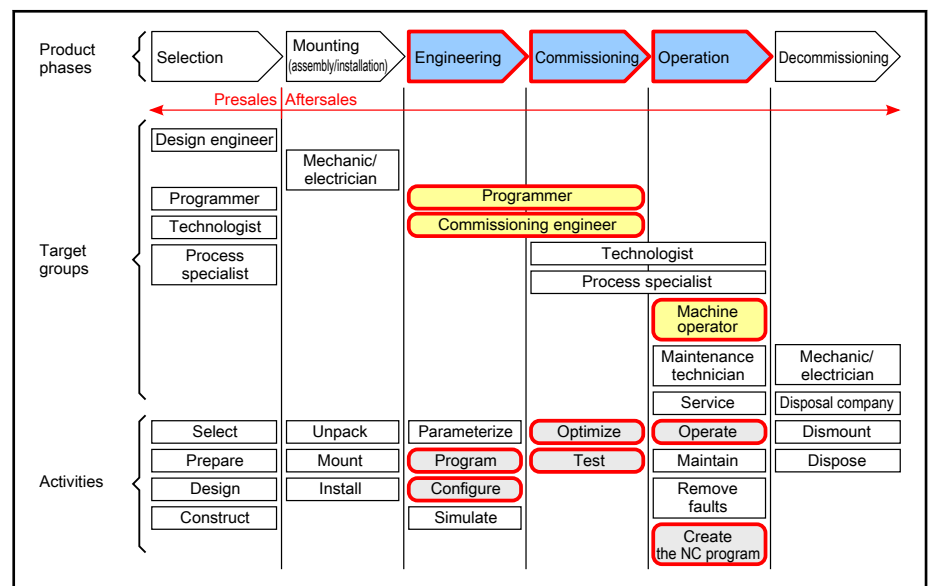


Fig. 1-1: Assigning this documentation to the target groups, product phases and target group activities

Target group	This documentation is intended for users commissioning an IndraMotion 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.
------------------	--

About this documentation

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 Requirements

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

The default settings of the dialogs as delivered are described. For further configurations, refer to the "IndraMotion MTX 14VRS Commissioning", DOK-MTX***-STARTUP*V14-CORS-EN-P documentation.

1.4 Required and supplementing documentation MTX

1.4.1 Selection/Compilation

Documentation titles with type codes and part numbers

IndraMotion MTX 14VRS System Description DOK-MTX***-SYS*DES*V14-PRRS-EN-P, R911342640 This documentation describes the Rexroth IndraMotion MTX control. It includes the designs, technical data, interfaces as well as the configuration of the control components.
IndraMotion 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 IndraMotion MTX.
IndraControl XM42 Controls DOK-CTRL-XM4X*CTRL**-ITRS-EN-P, R911345566 This documentation describes the IndraControl XM42 controls.

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

1.4.2 Configuration

Documentation titles with type codes and part numbers

IndraMotion MTX 14VRS Machine Parameters DOK-MTX***-MA*PAR**V14-RERS-EN-P, R911342630 This documentation describes structure and the adjustment of available parameters of Rexroth IndraMotion. It also includes the functions of the NC configurator and its operation.
IndraMotion MTX 14VRS SPS-Interface DOK-MTX***-PLC*INT*V14-PRRS-EN-P, R911342622 This documentation describes the interface signals and the program function blocks for the integrated PLC.
IndraMotion MTX 14VRS Functional Description - Basics DOK-MTX***-NC*F*BA*V14-RERS-EN-P, R911342626 This documentation describes the basic functions of Rexroth IndraMotion MTX. The basic commissioning steps and the control functions are provided as description and handling instruction.
IndraMotion MTX 15VRS Functional Description - Extension DOK-MTX***-NC*F*EX*V15-RERS-EN-P, R911393316 This documentation describes the extended functions of Rexroth IndraMotion MTX. The basic commissioning steps and the control functions are provided as description and handling instruction.
IndraMotion MTX 15VRS Functional Description - Special Functions DOK-MTX***-NC*F*SP*V15-RERS-EN-P, R911393309 This documentation describes the special functions of the Rexroth IndraMotion MTX. The basic commissioning steps and the control functions are provided as description and handling instruction.
IndraMotion 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.
Rexroth IndraMotion MTX Converting MTX Projects DOK-MTX***-PROCONV****-PRRS-EN-P, R911342484 This documentation provides support during the conversion of IndraMotion 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.

Tab. 1-3: MTX documentation overview - Configuration

1.4.3 Commissioning

Documentation titles with type codes and part numbers

IndraMotion MTX 15VRS Commissioning DOK-MTX***-STARTUP*V15-CORS-EN-P, R911393281 This documentation describes the commissioning of the IndraMotion MTX control. Apart from a complete overview, commissioning and configuration of the axes and the user interface as well as the PLC data are described.
IndraWorks 14VRS Basis Libraries IndraLogic 2G DOK-IL*2G*-BASLIB**V14-LIRS-EN-P, R911343920 This documentation describes the system-comprehensive PLC libraries.

About this documentation

IndraWorks 15VRS Field Buses DOK-IWORKS-FB*****V15-APRS-EN-P, R911393284 This documentation describes field bus connections and local periphery connections supported by the systems IndraLogic XLC, IndraMotion MLC and IndraMotion MTX. 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 14VRS HMI DOK-IWORKS-HMI*****V14-APRS-EN-P, R911343569 This documentation describes the functions, configuration and operation of the user interfaces IndraWorks HMI Engineering and IndraWorks HMI Operation.

Tab. 1-4: MTX documentation overview - Commissioning

1.4.4 Operation

Documentation titles with type codes and part numbers

Rexroth IndraMotion 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.
IndraMotion MTX 15VRS Programming Manual DOK-MTX***-NC**PRO*V15-RERS-EN-P, R911393318 This documentation describes the standard programming of the Rexroth IndraMotion 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.
IndraMotion 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 Rexroth IndraMotion MTX. It includes the operation of the interface, the NC program development as well as the tool management.
IndraMotion MTX 15VRS Multitouch DOK-MTX***-MULTI***V15-APRS-EN-P, R911393311 In this documentation, the operation of the multitouch user interface of the NC control Rexroth IndraMotion MTX is described.

About this documentation

IndraMotion 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 Rexroth IndraMotion MTX control.
IndraMotion MTX 15VRS NC Simulation DOK-MTX***-NC*SIM**V15-APRS-EN-P, R911393273 This documentation describes the NC simulation for the Rexroth IndraMotion MTX control.
IndraMotion MTX 15VRS Measuring Functions DOK-MTX***-MES*FUN*V15-APRS-EN-P, R91194938 ???

xx Corresponding edition
Tab. 1-5: MTX documentation overview - Operation

1.4.5 OEM Engineering

Documentation titles with type codes and part numbers

Rexroth IndraMotion 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.
Rexroth IndraMotion MTX 14VRS OPC Communication DOK-MTX***-OPC*COM*V14-PRRS-EN-P, R911342636 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 IndraMotion MLC and IndraMotion MTX control systems.

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

1.4.6 AddOns

Documentation titles with type codes and part numbers

Rexroth IndraMotion 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.
Rexroth IndraMotion MTX 14VRS Efficiency Workbench MTX cta, MTX ega DOK-MTX***-EWB*****V14-APRS-EN-P, R911343916 This documentation describes the mode of operation and the area of application of the analysis tool IndraMotion MTX cta and IndraMotion MTX ega.

About this documentation

Rexroth IndraMotion MTX Remote Condition Monitoring DOK-MTX***-RCM*****V01-APRS-EN-P, R911334383 This documentation describes the operation of the Remote Condition Monitoring System.
Rexroth IndraMotion 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 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:

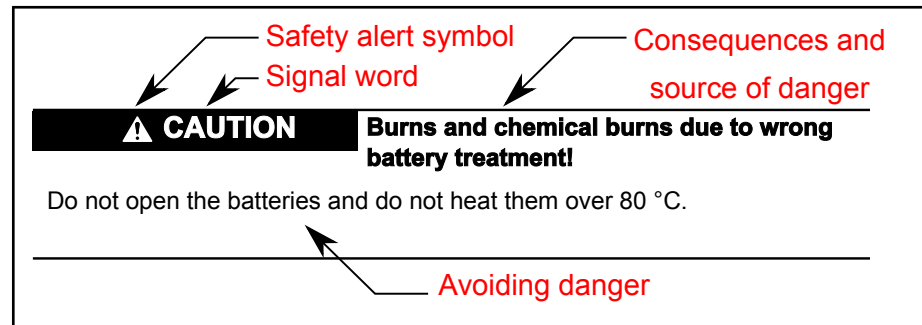


Fig. 1-2: Structure of the safety instructions

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.

[About this documentation](#)

⚠ 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.

About this documentation

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
OWL	Optical waveguide
NC	Numerical Control
OEM	Original Equipment Manufacturer
PROFIBUS-DP	Field bus
Sercos	Field bus

Tab. 1-8: Names and abbreviations used

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 dealing with 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. The decompilation of software products or the alteration of source codes is not permitted.
- Do not install or operate damaged or faulty products.
- 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 Rexroth IndraMotion 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.

Important instructions on use



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

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

2.2 Unintended use

The use of the Rexroth IndraMotion MTX in application areas other than those specified or described in the documentation and technical data is considered as "unintended".

The Rexroth IndraMotion 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.
- the intended applications have not expressly been released by Bosch Rexroth. Therefore, please read the information given the general safety instructions!
- The Rexroth IndraMotion 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 Function overview

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 operating mode. It is not possible to simulate the running NC program parallel to peak times.

Software options The simulation can be used in two configurations:

1. Basic license:

The basic license facilitates the workpiece simulation as material removal simulation and the tool path graphics. The definition of the raw part geometry is limited to semi-finished products such as cuboids, sheets and cylinders. The tool geometry is calculated on the basis of their technology data (generically generated tool models). An accurate tool representation on the basis of 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 any new installation of the following products from release MTX 14V12:

- Single-user license - IndraMotion MTX Operation 2.0 14VRS+higher
SWL-IWORKS-MTX-NNVRS-D0-OPD2, R911374093
- Single-user license - Operation 2.0 14VRS+higher (multipack)
SWL-IWORKS-MTX-NNVRS-D0-OPD2*M25, R911374094
- Single-user license - Workstation 2.0 14VRS+higher
SWL-IWORKS-MTX-NNVRS-D0-WORKSTATION2, R911374095
- Single-user license - Workstation 2.0 14VRS+higher (multipack)
SWL-IWORKS-MTX-NNVRS-D0-WORKSTATION2*25,
R911374096

One of the listed licenses is required when upgrading older versions and wanting to use the simulation.

2. Simulation virtual machine:

The configuration "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 accurately represented and 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. This license is required from version MTX 14V14.

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 is extended to more methods:

- Material removal simulation: Boring, milling, punching, beam cutting
- Line graphic: Rolling (forming) and signing

Function overview

- Task simulation: 3D print, welding and others
- Auxiliary processes, e.g. measurement

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 aligned NC axes. Special kinematics such as parallel and wedge kinematics cannot be simulated.

Simulation principle

The machine model comprises the following two components:

- Machine definition

This is a description of the machine structure that 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 that must exclusively be available in STL format (**ST**ereo **L**ithography) define the physical design of machine parts, tools, chucking devices up to workpieces.

Generally, each axis of the machine definition corresponds to a machine tool drive. The simulation is periodically (in a time interval: multiple of the interpolator cycle) informed about the current command position of each drive via a communication interface (OPD UA) and the axes of the model are corrected to the command position. 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 coordinates that are rarely changed (e.g. the position of clamping devices).
- Special kinematics can e.g. be mapped using virtual axes.
- The simulated punching head position is calculated using the sheet thickness and the punching signal and does in this case not correspond to the actual position.

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 affected channel.
- Depending on the option, starting the NC program triggers a renewal of the workpieces machined by the channel.

Function overview

- After reaching the end of program, a workpiece look-ahead (see ["Representation of finished part" on page 20](#)) is created with the current workpiece state, depending on the options.
- When selecting an NC program, a workpiece preview is shown, depending on the options, if the available preview file is younger than the main program.

Feature

The online simulation is provided as .NET control which by default is displayed in the NC screen of the Machine operating area. This control may also be integrated in a third party application in an IndraWorks context. The following operating description only applies analogously (see [chapter 11 "Integrating the MTX simulation control into third-party applications" on page 121](#)).

4 Starting and exiting the simulation

Start The NC simulation, if configured in the IndraWorks project, is active after starting IndraWorks Operation during the entire session and records each machine motion, irrespective of its visibility.

The simulation can be enabled and disabled by using a key. See [chapter 5 "Layout" on page 19](#).

Displays The NC simulation can have been configured as first display in an NC screen container or as second display. To visualize the simulation, use <F7> "Next window" to set the focus on the container under which the simulation is expected and press <F3> "Next display" until the simulation is displayed.

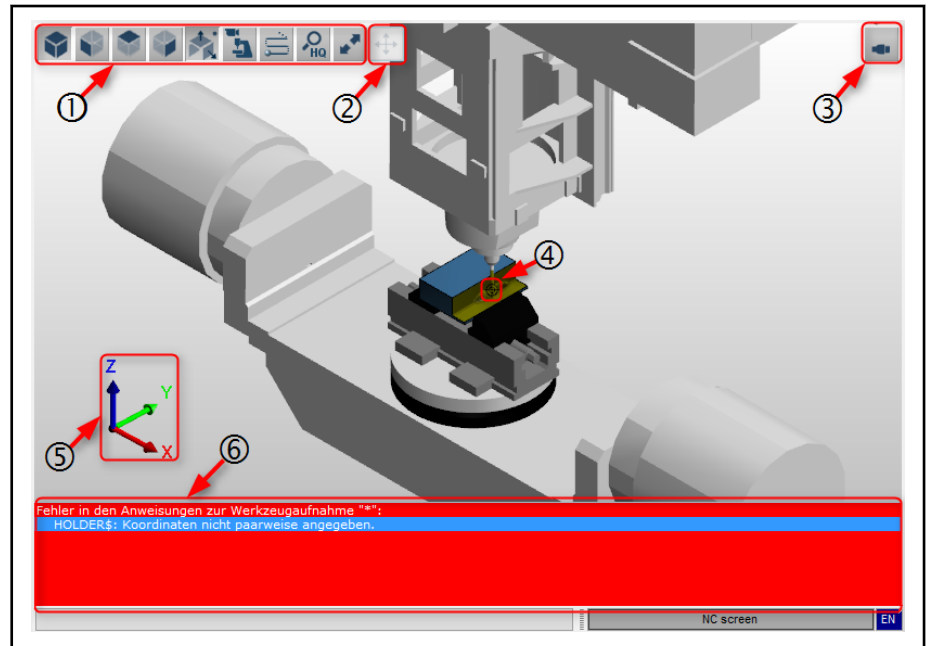


Fig. 4-1: NC simulation in NC screen (Automatic mode)

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 screen section and the camera position are part of it.

5 Layout

Screenshot



- 1 Shortcuts
- 2 Move/rotate status display active
- 3 Operation key and status display to connect to the NC and to disable
- 4 Generating point of tool path graphic at tool
- 5 Coordinate system
- 6 Output window for errors and warnings

Fig. 5-1: NC simulation layout

Shortcuts

By using the shortcuts, frequently used settings can be changed immediately without changing the F-key level if a mouse is available. The status 'selected yes/no' and the changing symbol display the current status. Tool tips provide additional information about the status.

Move/rotate status

The status display shows if the partial image can be moved using the cursor

keys () or if the view is rotated (). The display changes when keeping the <Shift> key pressed.

Connection to the NC and disabling the simulation

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

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

simulation is disabled. The  symbol indicates the disabled state.

Reactivation is executed by pressing the confirmation prompt and confirming 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.

Layout

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, the status at the time of reaching the regular NC program end during the last program run

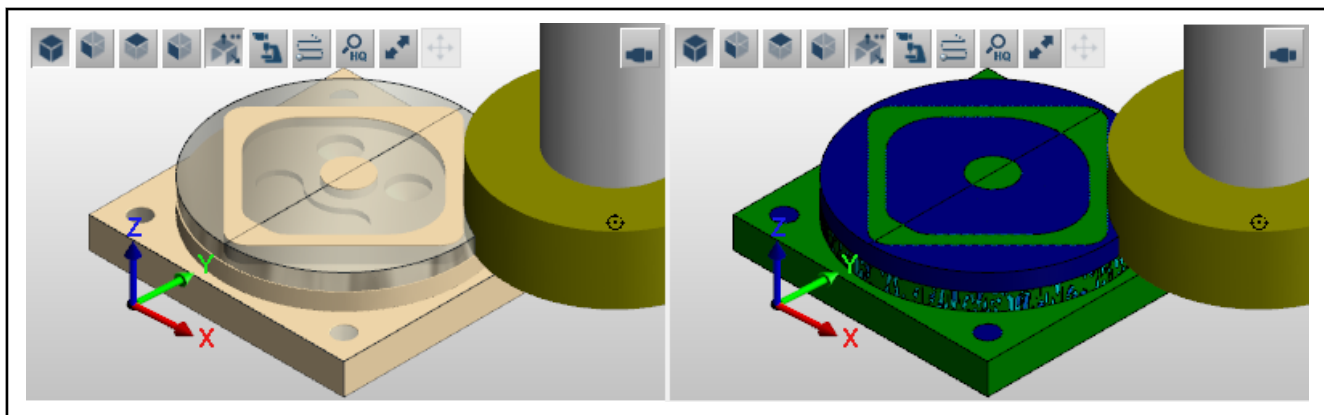
This geometry is saved in the main program folder under the program name with the extension ".p" and is then connected to the main program.

The finished part geometry generated by material removal and the corresponding tool path is referred to as **workpiece look-ahead**.

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

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

- as (usually internal) representation in the transparent workpiece
- as command-actual comparison with the workpiece. Different colors are allocated to the workpiece surfaces. The colors are schematically allocated to the 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)

Fig. 5-2: Internal representation of the finished part (left) and command-actual comparison (right)

6 Operation with keys and corresponding dialogs

6.1 General information

In the following, the function key level structure of the simulation is described. The levels can only be accessed if the focus is on the simulation control (focus frame).

6.2 Main level of the simulation

F2	F3	F4	F5	F6	F7	F8	F9
Views	Next display	Rotate/move	Details	Zoom +/-	Next window	Tools	Back

Tab. 6-1: F-keys of the main level of the simulation

Views	<F2> "Views" is a subordinate F key level of the "Views" menu.
Next display	<F3> "Next view" activates the next control and is only available if several views have been configured for this container. Actuation results in hiding of the simulation. <F3> "Next view" is not available in zoomed state.
Rotate/move	<F4> "Rotate/move" accesses two subordinate F-key levels. Here, the camera position can be changed by rotation and movement using F keys.
Details	<F5> "Details" accesses subordinate F-key levels where zooming is possible (menu "Details") or display modes can be changed (menu "Display").
Zoom +/-	<F6> "Zoom +" enlarges the simulation by hiding all other ACI containers. "Zoom -" reduces the simulation and displays all other ACI containers again.
Next window	<F7> "Next window" sets the focus on the next container of the NC screen. <F7> "Next window" is not available in zoomed state.
Tools	<F8> "Tools" activates the subordinate F-key level (menu "Tools") to influence the raw part and to change the options.
Back	<F9> "Back" displays the F-key main level of the NC screen.

6.3 Views menu

The "Views" menu can be accessed from the main menu of the simulation using <F2>"Views".

F2	F3	F4	F5	F6	F7	F8	F9
							Back
Front view	Rear view	Right view	Left view	Top view	Bottom view	3D view	

Tab. 6-2: F-keys of the "Views" menu

The F-keys for the views set a new camera position. Also "full screen", depending on the options. Subsequently, the position can be changed again by using cursor keys and the mouse.

The view can also be changed using shortcuts. The key state "pressed yes/no" indicates the state of the current view.

Front view	<F2> "Front view" shows the front view.
Rear view	<F3> "Rear view" shows the rear view.

Operation with keys and corresponding dialogs

Right view	<F4> "Right view" shows the right side.
Left view	<F5> "Left view" shows the left side.
Top view	<F6> "Top view" shows the top view.
Bottom view	<F7> "Bottom view" shows the bottom view.
3D view	<F8> "3D view" sets an inclined camera position with preset angles.
Back	<F9> "Back" activates the F-key main level of the simulation.

6.4 Rotate menu

The "Rotate" menu can be accessed from the main menu of the simulation using <F4> "Rotate/move" or from the "Move" menu with <F2>.

F2	F3	F4	F5	F6	F7	F8	F9
Offset	▶ Ro- tate right	◀ Ro- tate left	▲ Ro- tate top	▼ Ro- tate bot- tom			Back

Tab. 6-3: F-keys of the "Rotate" menu

The F-keys to rotate change the viewing angle. Optionally, the cursor keys and the mouse can be used.

Offset	<F2> "Move" accesses the next menu to move the partial image.
Rotate right	<F3> "Rotate right" moves the scene to the right or the camera position to the left.
Rotate left	<F4> "Rotate Left" moves the scene to the left or the camera position to the right.
Rotate top	<F5> "Rotate top" moves the scene to the top or the camera position to the bottom.
Rotate bottom	<F6> "Rotate bottom" moves the scene to the bottom or the camera position to the top.
Back	<F9> "Back" activates the F-key main level of the simulation.

6.5 Move menu

The "Rotate" menu can be accessed from the "Rotate" menu with <F2> "Move".

F2	F3	F4	F5	F6	F7	F8	F9
Rotate	▶ Move right	◀ Move left	▲ Move to top	▼ Move to bottom			Back

Tab. 6-4: F-keys of the "Move" menu

The F-keys for moving change the viewing angle. Optionally, the cursor keys and the mouse can be used.

Rotate	<F2> "Rotate" leads back to the menu to rotate the camera position.
Move right	<F3> "Move right" moves the scene to the right or the partial image to the left.
Move left	<F4> "Move left" moves the scene to the left or the partial image to the right.

Operation with keys and corresponding dialogs

Move to top	<F5> "Rotate top" moves the scene to the top or the partial image to the bottom.
Move to bottom	<F6> "Rotate bottom" moves the scene to the bottom or the partial image to the top.
Back	<F9> "Back" activates the F key main level of the simulation.

6.6 Details menu

The "Details" menu can be accessed from the main menu of the simulation using <F5>"Details".

F2	F3	F4	F5	F6	F7	F8	F9
	 Improve work-piece	Zoom in	Zoom out	 Full screen	 Camera position	more functions >>	Back

Tab. 6-5: F keys of the "Move" menu

The representation scale can also be changed using the mouse wheel or shortcuts.

Improve workpiece <F3> "Improve workpiece" allows to improve the representation quality and returning 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 partial image. If the improve function is active, the improvement is executed after each zooming process. After changing the image section, the improvement is lost.
- **Auto refine**
If permitted by the simulation utilization, 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 partial image can still be changed.

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

Zoom in	<F4> "Zoom in" increases the object-to-image ratio centrally to the image center.
Zoom out	<F5> "Zoom out" decreases the object-to-image ratio.
Camera position	<F6> "Camera position" determines to which object the camera position is pointing at.

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

Operation with keys and corresponding dialogs

- Machine: This stationary reference is the position of the observer/operator. Shortcut symbol: 
- Tool: The camera position moves with the tool, does, however, 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: 

More functions <F8> "More functions" switches to the second menu level "Details", referred to as "Representation menu".

Back <F9> "Back" activates the F key main level of the simulation.

6.7 Display menu

The display menu can be accessed from the main menu of the simulation by pressing <F5> "Details", <F8>"More functions".

F2	F3	F4	F5	F6	F7	F8	F9
 Hide	 Tool path	 Edges	 Finished part	 Axes/ scales		Back <<	

Tab. 6-6: F keys of the "Representation" menu

- Hide** Press <F2>"Hide" to toggle between machine and workpiece simulation in several steps by hiding/showing machine parts and tool holders.
- The following steps are available depending on the machine model:
- Hide OFF, hood closed
 - Hood open
 - Hide all except for active elements. In this setting, all parts not moved by NC axes and not part of tools, workpieces or clamping devices are hidden.
 - Hide all except for workpiece, tool and clamping device. Tool holders are not displayed to be able to better monitor the tool operation.
- Tool path** <F3> "Tool path" activates/deactivates the tool path representation in the following steps:
- 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.

Operation with keys and corresponding dialogs

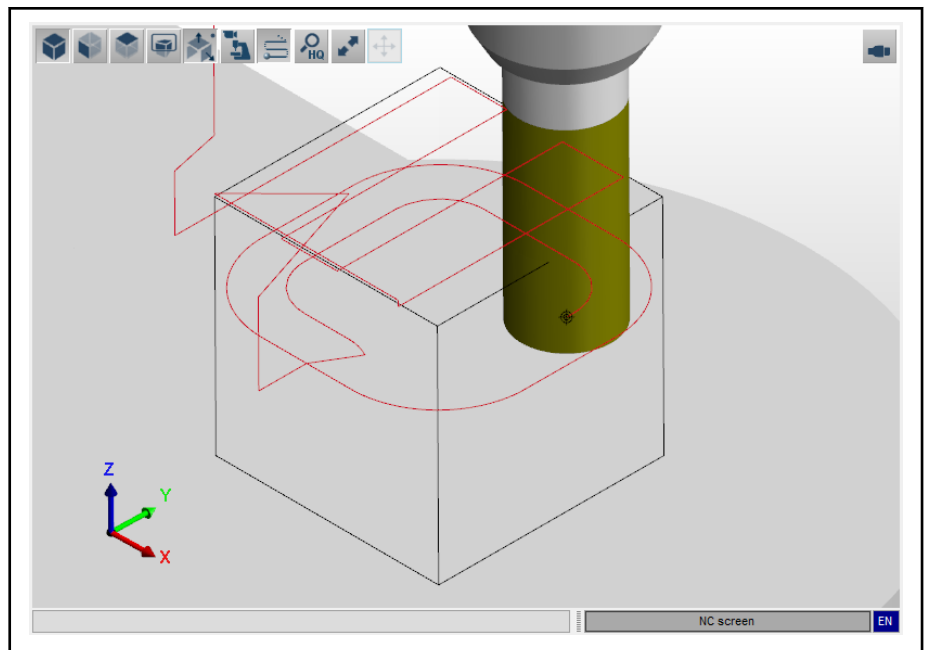


Fig. 6-1: Simulation with tool path on top

Edges	<F4> "Edges" draws the object edges with black lines or suppresses retracing. 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 case of drawings or in case the transparency limit is exceeded.
Finished part	With <F5> "Finished part", 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 be made visible with <F3> "Tool path". The key is only available if the finished part geometry is available. There are three levels: • No finished part visible • Finished part visible (in transparent workpiece) • Command/actual comparison of the workpiece to the finished part. For different intervals of the dimensional deviation, different colors are used allowing for conclusions on waste, finished dimension reached or remaining machining allowance (see also chapter 5 "Layout" on page 19)
Axes/scales	<F6> "Axes/scales" subsequently shows the coordinate tripod or scales with numeric values in the following stages: • No coordinate system visible • Coordinate tripod visible • Axis scales visible
Back	<F9> "Back" activates the F key main level of the simulation.

6.8 Tools menu

The "Tools" menu can be accessed from the main menu of the simulation using <F8>"Tools".

Operation with keys and corresponding dialogs

F2	F3	F4	F5	F6	F7	F8	F9
							Back
Reset raw part...	Cuboid, sheet...	Cylinder...	Import...	Export...		Options...	

Tab. 6-7: F keys of the "Tools" menu

Reset raw part

<F2>"Reset raw part..." opens a dialog to renew the workpiece and to delete graphics.

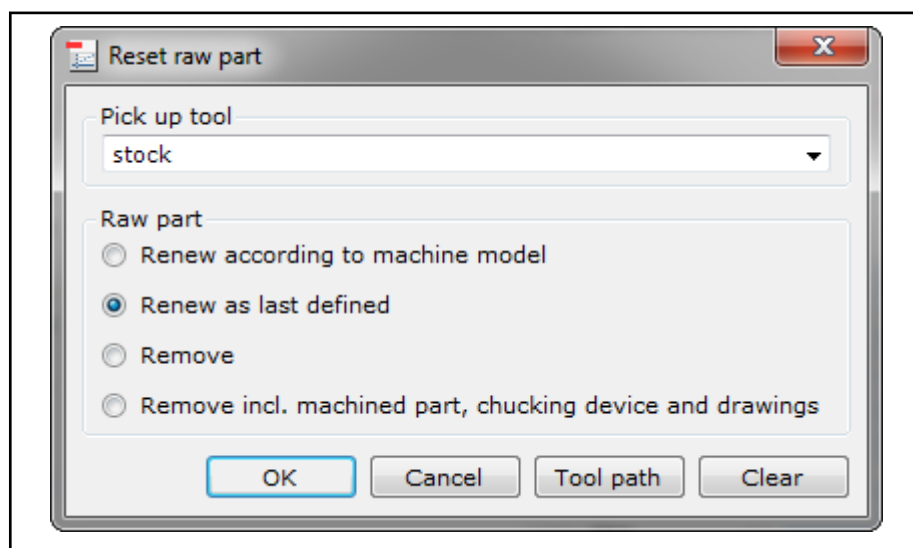


Fig. 6-2: "Reset raw part" dialog

The dialog has the following elements:

- Selection list "Workpiece holder": The selection determines to which workpiece holder (stock node) resetting 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 of 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 yourself, it is displayed as raw part.
 - Delete:
Removes the workpiece.
 - Remove including finished part, clamping device and drawings:
Empties the workpiece holder completely.
A finished part means a geometry corresponding to the machining end, represented within the (transparent) workpiece.
Drawings are representations of object edges without taking the collision detection into consideration.
- <OK> executes the selected action and closes the dialog.
- <Cancel> closes the dialog without action.

Operation with keys and corresponding dialogs

- Use <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. Only use this functions in case of repeated problems with the workpiece representation that cannot be solved in any other way. The deleted data will be irrecoverably lost. That is why a safety prompt is displayed before data is deleted.

Cuboid/sheet <F3> "Cuboid/sheet..." 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.

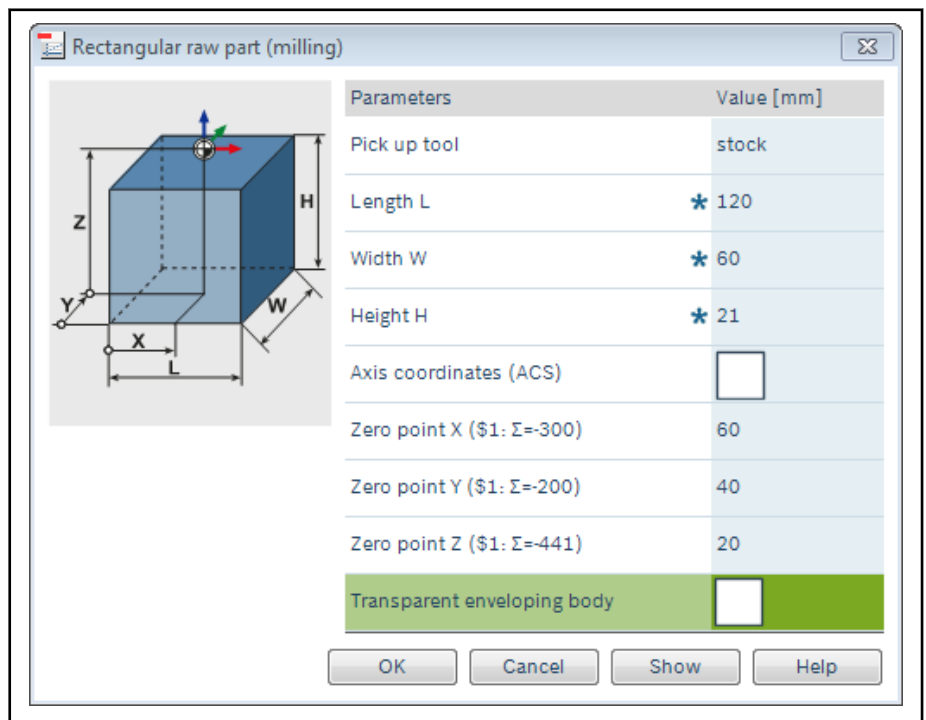


Fig. 6-3: "Cuboid raw piece (milling)" dialog

Operation with keys and corresponding dialogs

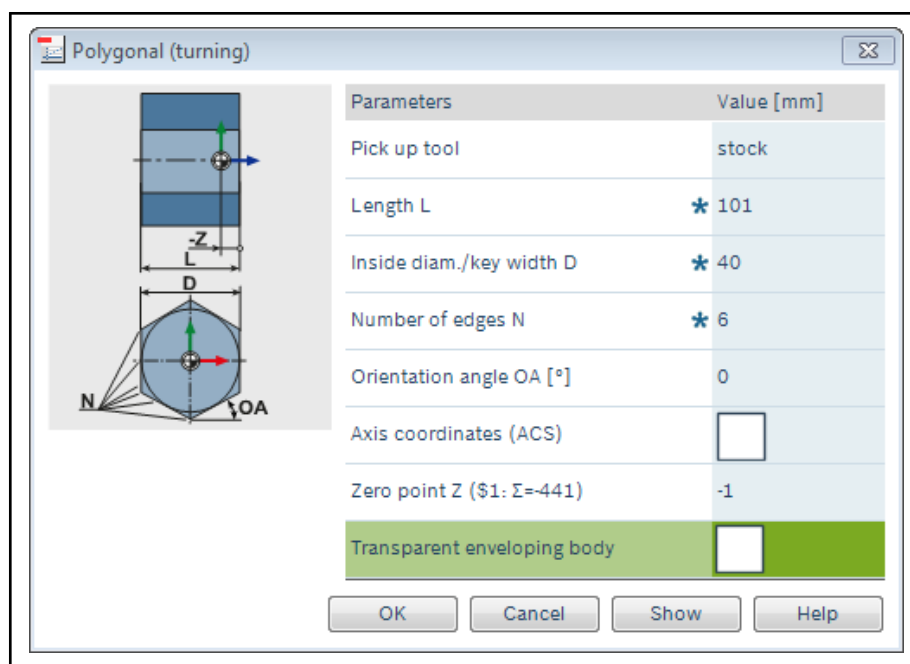


Fig. 6-4: "Polygon (turning)" dialog

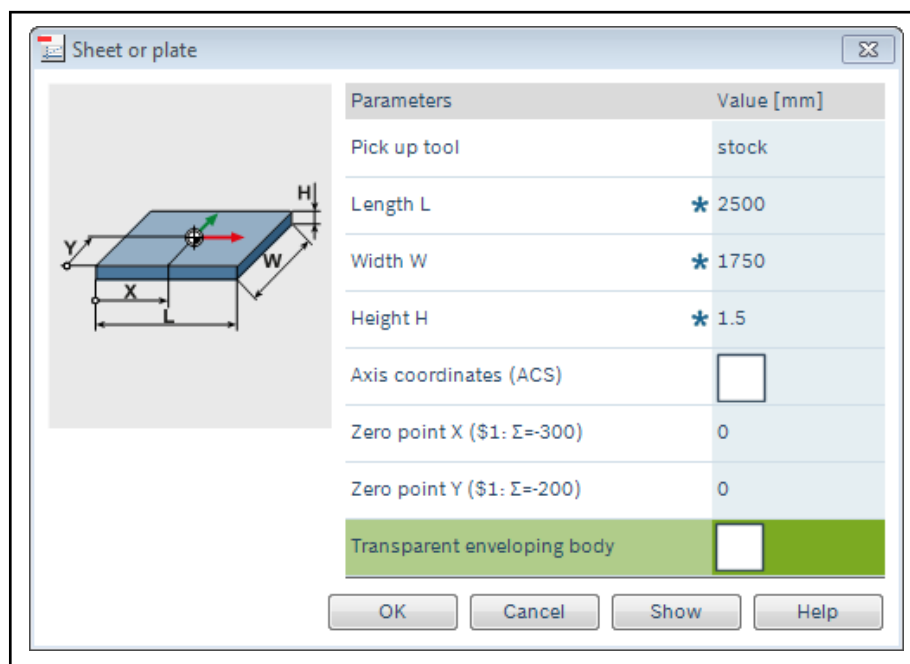


Fig. 6-5: "Sheet or plate" dialog

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

The respective dimensions follow.

The "Axis coordinates (ACS)" check box specifies whether the sum of zero point offsets in the channel is to be added internally to the following parameters, the zero points. For information purposes, the sum in the axes is displayed in the parameter names after the sum character.

If "transparent enveloping body" is ticked, the raw part is displayed as edge model without collision check and without possible material removal. The tool

Operation with keys and corresponding dialogs

path representation is automatically activated. Use this setting for more complex workpieces you do not have a CAD model for.

If you operate the <Display> key, the new raw part is shown flashing, to check the inputs. The dialog is converted into a non-modal one 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. In the latter case, remove the workpiece using "Reset raw part" if you want to replace the workpiece.

Cylinder

<F4> "Cylinder..." is used to define a cylindrical raw part, optionally with internal bore. In the main machining process Turning, a dialog appears to define a cylindrical bar section.

Parameters	Value [mm]
Pick up tool	stock
Outside diameter D1	* 100
Internal diameter D2	0
Height H	* 51
Axis coordinates (ACS)	
Zero point X (\$1: Σ=-300)	0
Zero point Y (\$1: Σ=-200)	0
Zero point Z (\$1: Σ=-441)	50
Orientation ORI	Z
Transparent enveloping body	☒

Fig. 6-6: "Cylindrical raw part (milling)" dialog

Operation with keys and corresponding dialogs

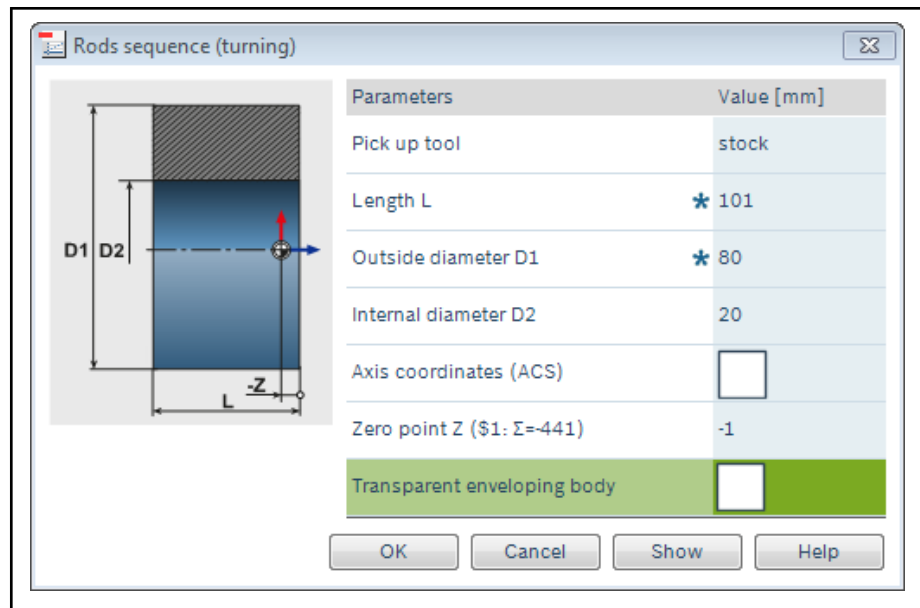


Fig. 6-7: "Bar section (turning)" dialog

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

The respective dimensions follow.

The "Axis coordinates (ACS)" check box specifies whether the sum of zero point offsets in the channel is to be added internally to the following parameters, the zero points. For information purposes, the sum in the axes is displayed in the parameter names after the sum character.

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

If you operate the <Display> key, the new raw part is shown flashing, to check the inputs. The dialog is converted into a non-modal one 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. In the latter case, remove the workpiece using "Reset raw part" if you want to replace the workpiece.

Import Use <F5> "Import..." to import a CAD model for different targets.

Initially, a file selection dialog is displayed from which the source file is to be selected. The latter can be saved in the MTX file system and in the file system of the local PC.

To change in the file selection for opening from the local PC to the MTX file system proceed as follows:

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>.

Operation with keys and corresponding dialogs

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

To change in the file selection for opening from the MTX file system to the local PC proceed as follows:

1. In the "File type" selection list, select the last entry "Change to local PC (*.*)".
2. Click on <Cancel> or optionally click 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.

After clicking on <OK>, the model to be imported is displayed flashing in an initial position at the workpiece holder. Additionally, a dialog is shown in horizontal or vertical alignment in which the import process can be controlled:

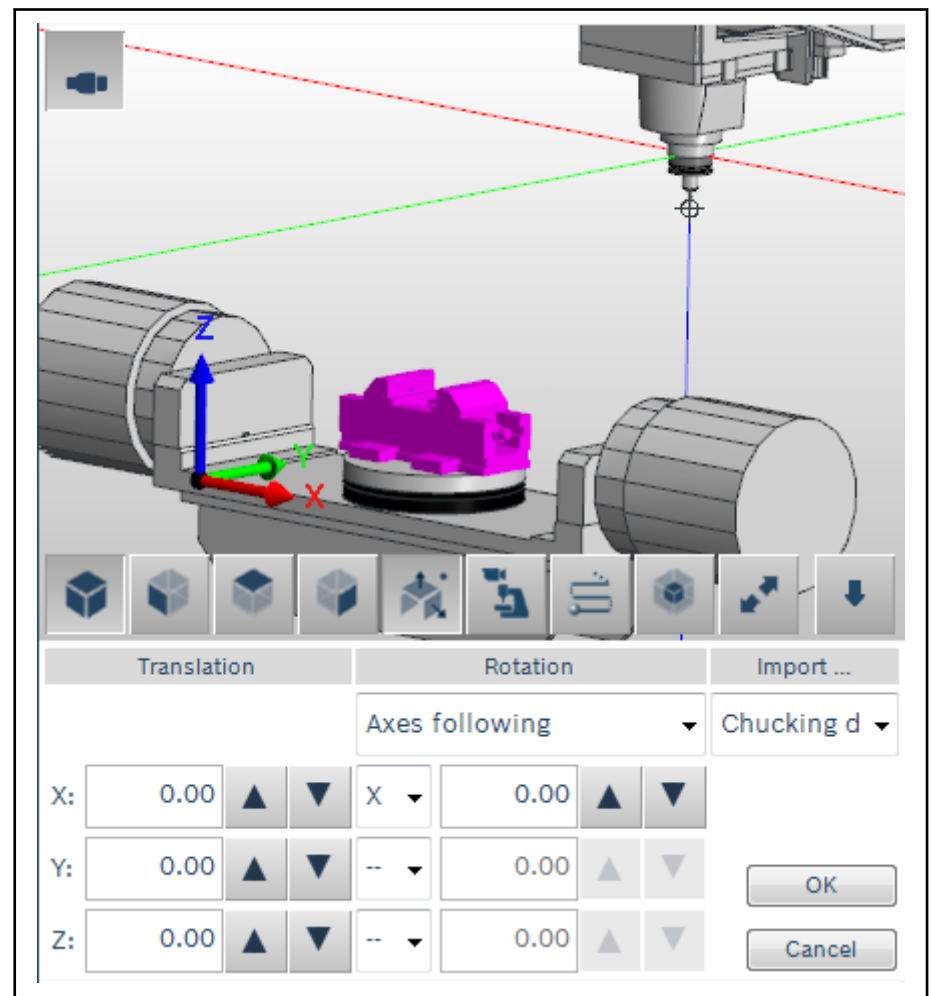


Fig. 6-8: Import (dialog in horizontal arrangement)

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

Operation with keys and corresponding dialogs

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 subsequently displayed. This function can be used if an existing CAD model is to be adjusted by rotating or moving.

If the machine model has more than one workpiece holder, there is another selection list below the afore-mentioned one which is to be used to select the workpiece holder to which the import is to be effected.

- "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 with the selection lists before the angles of rotation.

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

Export <F6> "Export..." writes the current workpiece geometry to a file, except for 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, a dialog to select the workpiece holder is displayed to select which holder is to be exported.

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

To change in the file selection for Save as from the local PC to the MTX file system proceed as follows:

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 if you want to replace the file with <Yes> (is not executed).

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

To change in the file selection for Save as from the MTX file system to the local PC proceed as follows:

1. In the "File type" selection list, select the last entry "Change to local PC (*.*)".
2. Click on <Cancel> or optionally click on any file and press <Save>.
3. If <Save> has been pressed in the previous step, confirm the subsequent prompt if you want to replace the file with <Yes> (is not executed).

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

Options <F8>"Options" facilitates displaying and changing of the current options/settings of the simulation in a text editor.

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

Operation with keys and corresponding dialogs



The XML file opened in the editor is a temporary file. If the XML syntax is damaged 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 data loss. After such case, please download the visualization data once 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 10 "Configuration" on page 81](#).

After saving the changes and closing the editor, confirm or reject the box "Apply options? <Yes>/<No>". If you select <Yes>, the options are applied.

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.

Back <F9> "Back" displays the F key main level of the NC screen.

7 Shortcuts

Simulation shortcuts



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

Shortcut	Simulation function
<C>	Establish connection to the NC
<N>	Define new raw part as semi-finished product: • Milling, generative processes: cuboid-shaped 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 partial image, indicated by the  symbol
<Shift>+<Cursor key>	Rotates the camera position, indicated by the  symbol
<Ctrl>+<C>	Copies the transformation matrix (4x4) in the syntax of the machine definition to the clipboard. Applicable in the import dialog in case of focus on an input element.
<Shift>+<F10>	Displays the context menu for recording of the simulation. Only available in debug mode.
<Left mouse key>	Rotates the camera position when the mouse is moved while the left mouse key is pressed
<Right mouse key>	Moves the partial image when the mouse is moved while the right mouse key is pressed. In debug mode, the menu selection to control the recording is displayed after a short delay.
<Mouse wheel>	Scrolling up zooms in, scrolling down zooms out, centrally to the mouse position.

Tab. 7-1: Shortcuts – simulation functions

8 Recording and playing recordings

Objective	Using the recorder contained 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, testing and reproducing 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. See chapter 6 "Operation with keys and corresponding dialogs" on page 21, menu "Tools" - <F8> "Options". For information on the XML tag, refer to chapter "Optimization" on page 113. 2. Activate the context menu of the display surface of the simulation and activate "Start recording". 3. Simulate the sequence to be recorded and reactivate the context menu. Now 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	If the recording is to be played outside the IndraWorks installation, all required data can be added to a multi-part archive with the name C:\Users\<Username>\AppData\Local\Temp\Rexroth\IndraWorks\MTX.Simulation.TestAppRuntime.zip.001 bis <n> by executing <IndraWorks installation directory>\MTX.Simulation.TestApp.CreateRuntimeArchive.bat with admin rights. Unzip the archive in the execution folder of the player.
Starting the player	If the player is not directly started 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").

Recording and playing recordings

Player layout

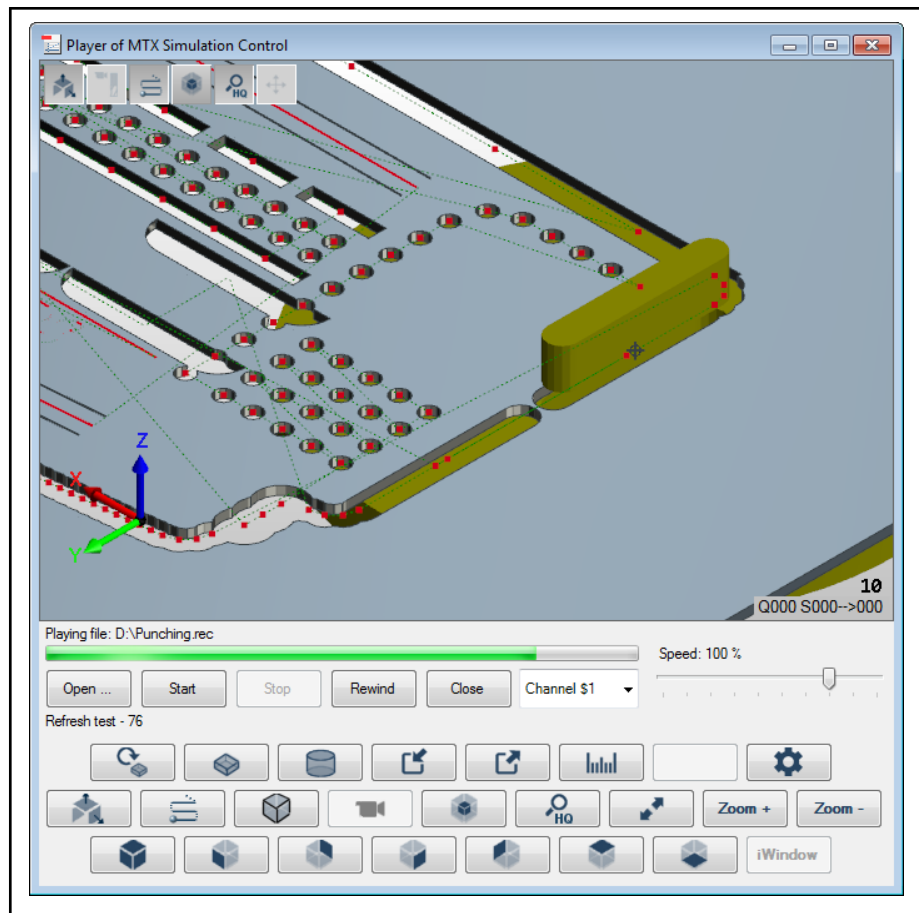


Fig. 8-1: Simulation player

Operating the player

- <Open...> displays a file dialog in which the recording file is selected and opened
- <Start> starts the animation
- <Stops> cancels the animation
- <Rewind> resets the replay
- <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 in the player are changed, 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 <Rewind>.

9 Program instructions for simulation

9.1 General information about program instructions for simulation

Functional scope	The instructions for the simulation are cycle subroutines providing the simulation with information about the graphic. They can be inserted into NC programs and also be used in the manual entry (MDI). With the cycles, the following functions are covered: • 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 in principle, 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 projected (auxiliary means to commission the simulation) Tool descriptions are made for simple drilling and milling tools. In this connection, only the dimensions of the cutting edge part are usually defined. If possible, the simulation will amend the shaft, the arbor and the holder.
Input masks	Cycle masks are provided in the NC Editor for the instructions. They are available for input under <F5> "Input assistance" -> <F5> "Simulation instructions". The correction is possible with <F3> "Correction NC cycle".

9.2 STOCK/BLOCK Cuboid raw part (milling)

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

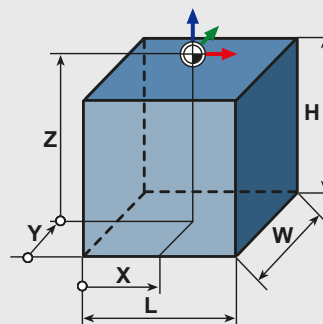


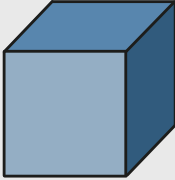
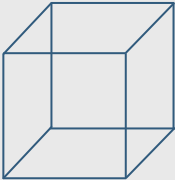
Fig. 9-1: STOCK/BLOCK instruction cuboid raw part

Program instructions for 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. 9-1: Parameter call

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

Tab. 9-2: Meaning of 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 removal simulation or an enveloping body is to be drawn. The latter is an alternative to import a CAD model in a complex geometry.

Program instructions for simulation

- Special features** The parameters are position parameters without an address.
- Procedure/processing** The workpiece is displayed as compatible with 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)
```

9.3 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 internal boring, is generated at or added to the selected workpiece holder. It is intended for the machining simulation on milling machines.

Overview

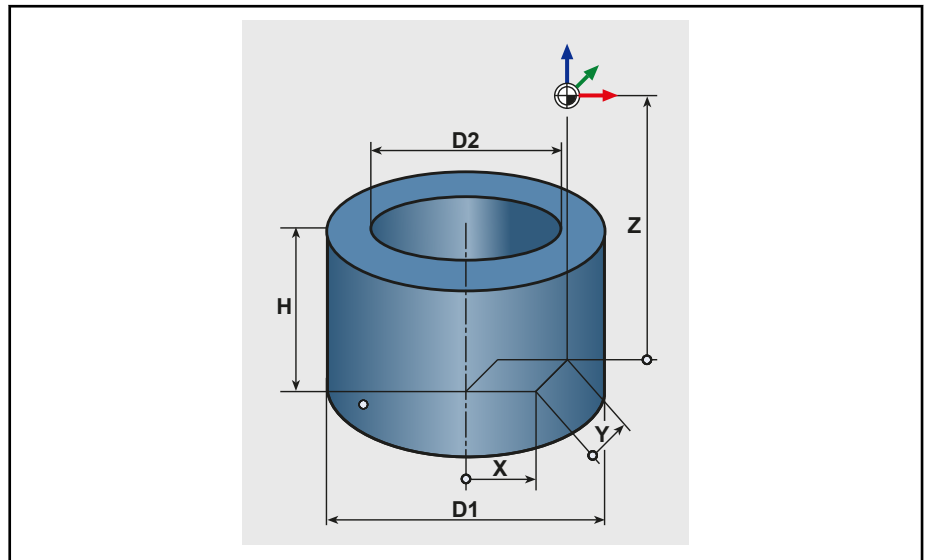


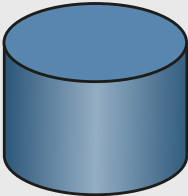
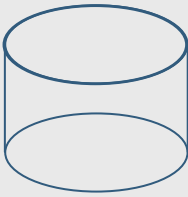
Fig. 9-2: STOCK/CYLINDER instruction cylindrical raw part

Parameter

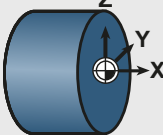
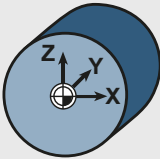
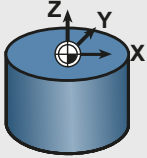
Parameter	Name	Properties	Values	Pre-assignment
D1	Outer diameter	Mandatory	>0	
D2	Inside diameter	optional	>=0	0
H	Height	Mandatory	>0	
X	Zero point in X	optional	incremental	0
Y	Zero point in Y	optional	incremental	0
Z	Zero point in Z	optional	incremental	0
V	Modeling and orientation	optional		0

Tab. 9-3: Parameter call

Program instructions for simulation

V units digit	Effect
0	Volume
1	Enveloping body
V=x0  V=x1 	

Tab. 9-4: Meaning of the units digit of 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. 9-5: Meaning 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, defines a continuous internal boring.
- H height**

Program instructions for simulation

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 removal simulation or an enveloping body is to be drawn. The latter is an alternative to import a CAD model in a complex geometry.

The tens digit influences the center axis orientation.

Special features

The parameters are position parameters without an address.

Procedure/processing

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

Example

Hollow cylinder in volumetric display with a diameter of 100, an inside 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)
```

9.4 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 internal bore, is generated at or added to the selected workpiece holder. It is intended for turning.

Overview

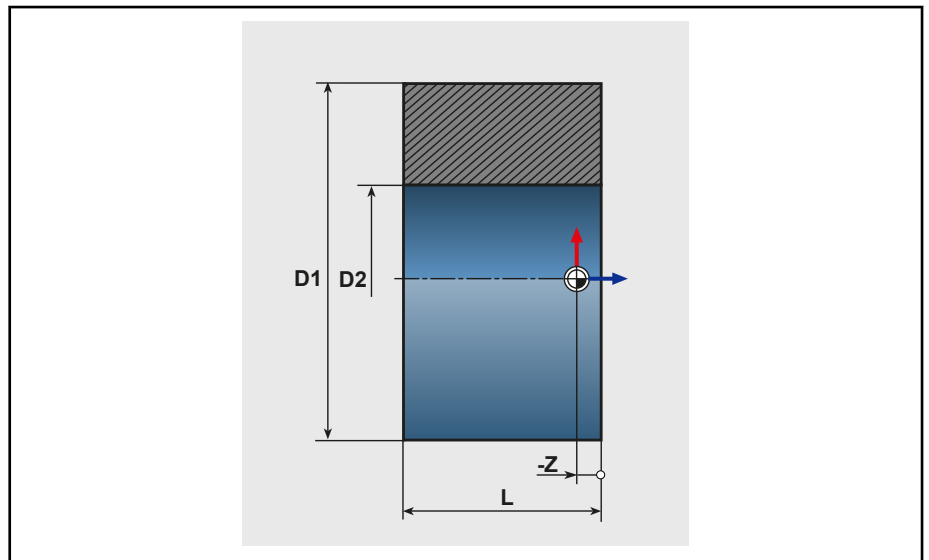


Fig. 9-3: STOCK/BAR instruction bar section

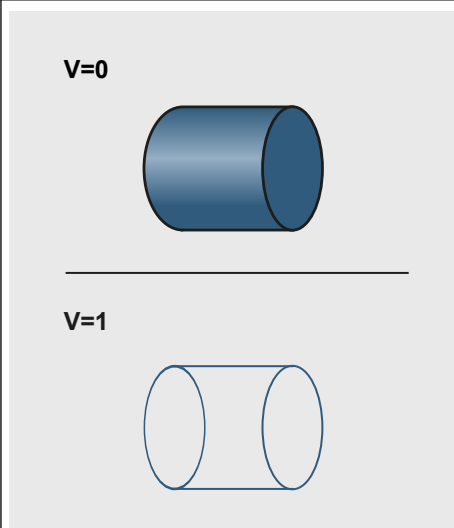
Parameter

Parameter	Name	Properties	Values	Pre-assignment
L	Length	Mandatory	>0	
D1	Outer diameter	Mandatory	>0	

Program instructions for simulation

D2	Inside diameter	optional	≥ 0	0
Z	Zero point in Z	optional	incremental	0
V	Modeling	optional		0

Tab. 9-6: Parameter call

V modeling	Effect
0	Volume
1	Enveloping body
	

Tab. 9-7: Meaning of the units digit of 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, defines a continuous internal boring.

Z zero point offset

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

V modeling

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

Special features

The parameters are position parameters without an address.

Procedure/processing

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

Example

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

Program:

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

9.5 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

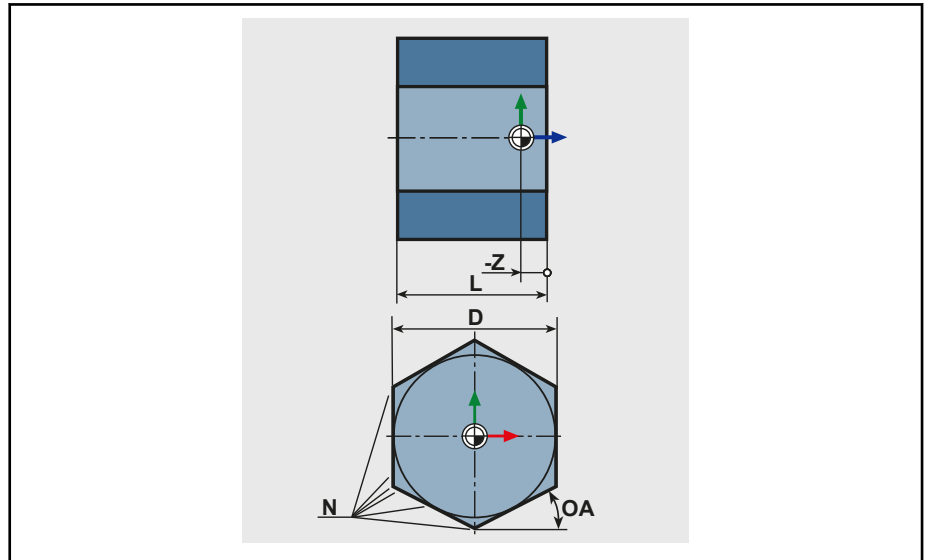


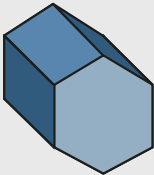
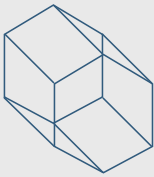
Fig. 9-4: STOCK/POLYGON instruction polygon

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 angle	optional	Degree	0
V	Modeling	optional		0

Tab. 9-8: Parameter call

Program instructions for simulation

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

Tab. 9-9: Meaning of 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 limiting surface to the right (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 removal simulation or an enveloping body is to be drawn. The latter is an alternative to import a CAD model in a complex geometry.
Special features	The parameters are position parameters without an address.
Procedure/processing	The workpiece is displayed as compatible with the previously activated work-piece zero point. The tool path is deleted.
Example	Hexagon in volumetric display with a diameter of 60, a length of 82. The workpiece zero point (G54.4) is to be located 2 mm to the left of the limiting surface to the right. There has to be a profile edge at angle C90. <i>Program:</i> N10 G54.4 STOCK("POLYGON",82,60,6,-2,30)

9.6 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

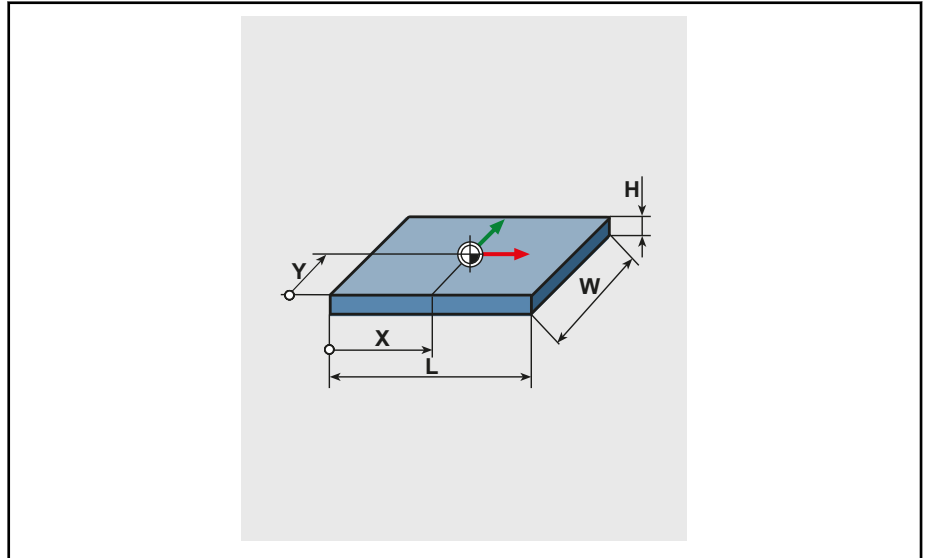


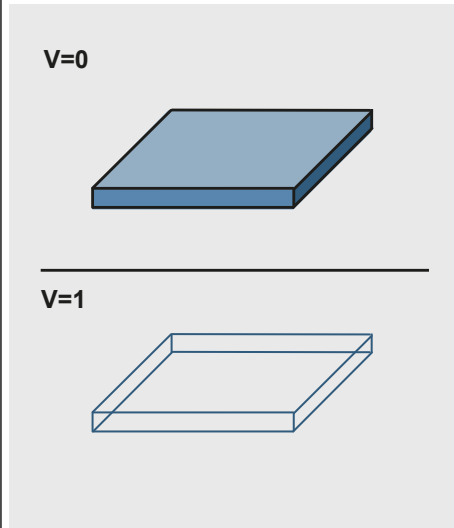
Fig. 9-5: STOCK/SHEET instruction sheet

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. 9-10: Parameter call

Program instructions for simulation

V	Effect
0	Volume
1	Enveloping body
	

Tab. 9-11: Meaning of 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 removal simulation or an enveloping body is to be drawn. The latter is an alternative to import a CAD model in a complex geometry.
Special features	The parameters are position parameters without an address.
Procedure/processing	The workpiece is displayed as compatible with the previously activated workpiece zero point. The tool path is deleted.
Example	Plate in volumetric display with 1000 in length, 500 in width and 2.5 in thickness. The workpiece zero point (G54.5) has to be in the center.
	<i>Program:</i> <pre>N10 G54.5 STOCK("SHEET",1000,500,2.5,500,250)</pre>

9.7 STOCKIMP Import model

Syntax	STOCKIMP(P, TGT,X,Y,Z,RB,RS,A1,A2,A3)
Brief description	Imports a CAD model displayed at a workpiece holder.

Program instructions for simulation

Overview

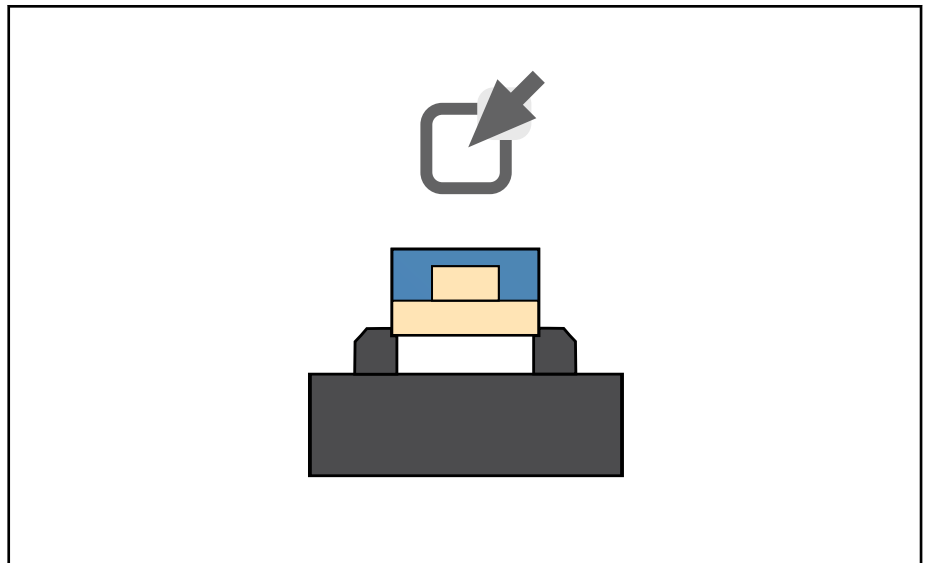


Fig. 9-6: *STOCKIMP instruction Import model*

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 axes of rotation	optional		1
RS	Order of the axes of rotation	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. 9-12: *Parameter call*

TGT	Effect
1	Imports a raw part.
2	Imports a finished part.

Program instructions for simulation

TGT	Effect
3	Imports a clamping device.
4	Imports a drawing.

Tab. 9-13: Meaning of parameter TGT - Import target

RB	Effect
1	fixed in space The axes of rotation maintain their initial position.
2	consequently The axes of rotation rotate synchronously with the model.

Tab. 9-14: Meaning of parameter RB - Position of the axes of rotation

Description

P File name

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 a file name is specified without path, the file is searched for 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 displayed in [chapter 10.2 "Introduction to the tool graphic generation" on page 84](#).

TGT Import target

Specifies the type of model use.

X, Y, Z translation

Specifies an offset of the coordinate origin of the imported model.

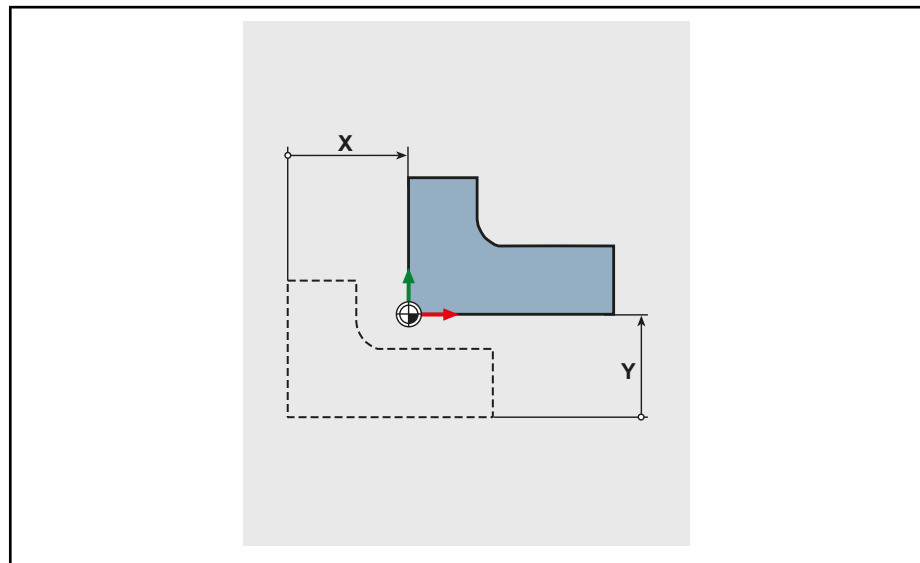


Fig. 9-7: Translation in X and Y. The target position is displayed in color.

RB Position of the axes of rotation

Program instructions for simulation

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

RS Order of the axes of rotation

Specifies the axes of rotation that are to be used 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 with RS.

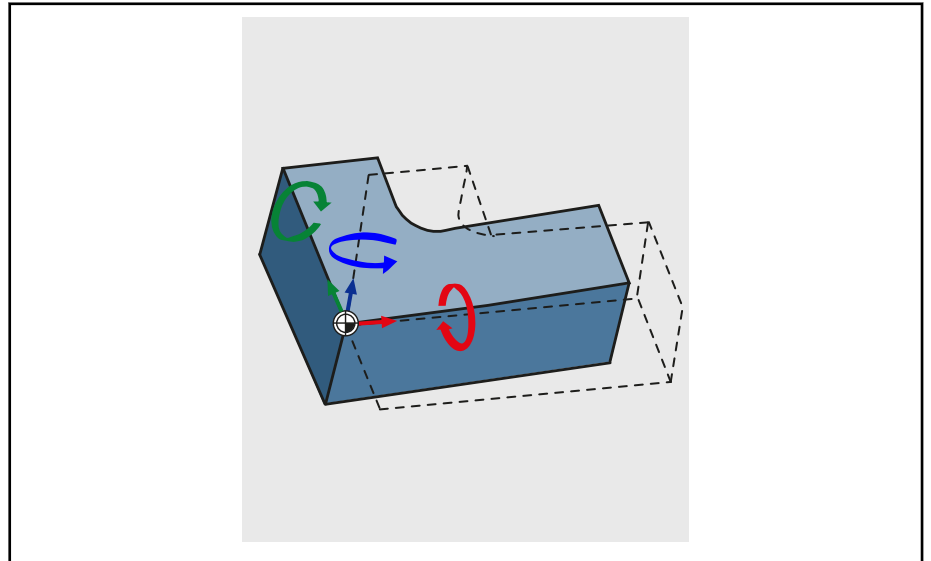


Fig. 9-8:

Special features

The parameters have addresses.

Procedure/processing

The model is displayed as compatible with the previously activated workpiece zero point. The tool path is deleted upon the import as workpiece.

Example

The "finalwp.stl" model contained in the folder of the main program is to be imported as finished model. To do so, execute one rotation around the axes fixed in space: X at 180° and Z at 90°. To position the model in the correct position in relation 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)
```

9.8

STOCK/REMOVE Remove workpieces/models

Syntax

STOCK("REMOVE",<TGT>)

or

STOCK()

Brief description

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

Program instructions for simulation

Overview

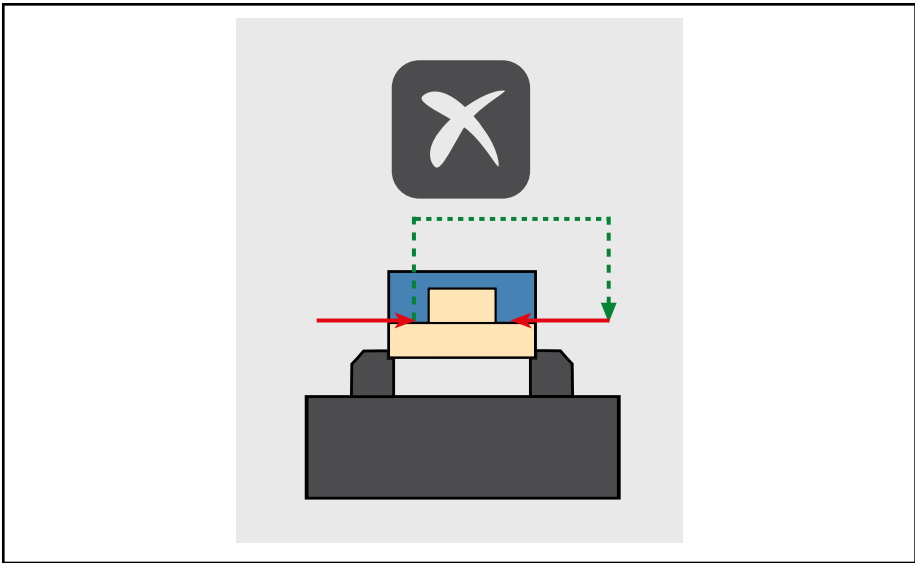


Fig. 9-9: STOCK/REMOVE instruction Remove workpieces/models

Parameter

Parameter	Name	Properties	Values	Pre-assignment
TGT	geometry to be removed	optional		0

Tab. 9-15: Parameter call

TGT	Effect
0	complete Removes all geometries of all workpiece holders.
1	Removes workpiece/raw part and 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. 9-16: Meaning of parameter TGT - Geometry to be removed

Description

TGT Geometry to be removed

The entire parameter list can be omitted in case of default value TGT=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 26, menu item "removing incl. finished part, clamping device and drawings").

Procedure/processing

The selected geometry is deleted.

Example Workpiece, tool path and finished part are to be deleted.

Program:

```
N10 STOCK("REMOVE",1) ;removes workpiece and tool path
N20 STOCK("REMOVE",2) ;removes finished part
```

9.9 STOCK/RENEW Renew workpiece

Syntax STOCK("RENEW",<TN>)

Brief description Renews the workpiece of the affected workpiece holder or all workpieces by replacing the workpiece(s) by the last defined raw part geometry. The function also deletes the tool path. In this connection, it is not relevant whether the workpiece has previously been machined or removed.

Overview

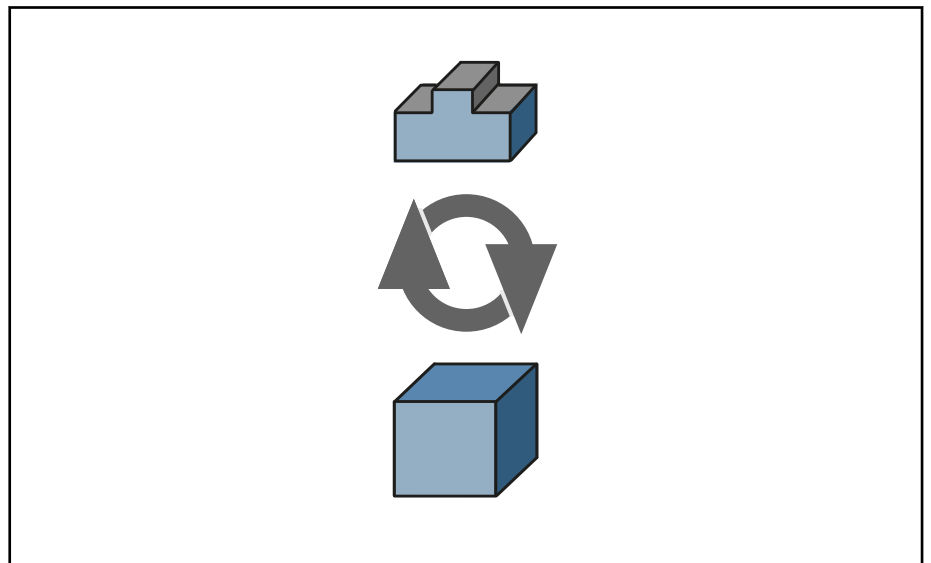


Fig. 9-10: STOCK/RENEW instruction Renew workpiece

Parameter

Parameter	Name	Properties	Values	Pre-assignment
TN	affected workpiece holder	optional		0

Tab. 9-17: Parameter call

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

Tab. 9-18: Meaning of parameter TN - affected workpiece holder

Description

TN affected workpiece holder

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

Special features The parameter does not have an address.

Program instructions for simulation



Generally, the result is the same when you define a new workpiece 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 renewal of the workpiece/workpieces machined by the relevant channel upon the program start.

Procedure/processing	The workpiece geometry is restored as last defined.
Example	All workpieces have to be renewed. <i>Program:</i> <pre>N10 STOCK("RENEW")</pre>

9.10 STOCKSEL Select workpiece holder

Syntax	STOCKSEL(<NAM>)
Brief description	Selects a workpiece holder via its name. All subsequent calls of the STOCK and STOCKIMP instructions in the same channel refer to this workpiece holder. This instruction is required if a tool machine has several workpiece holders.

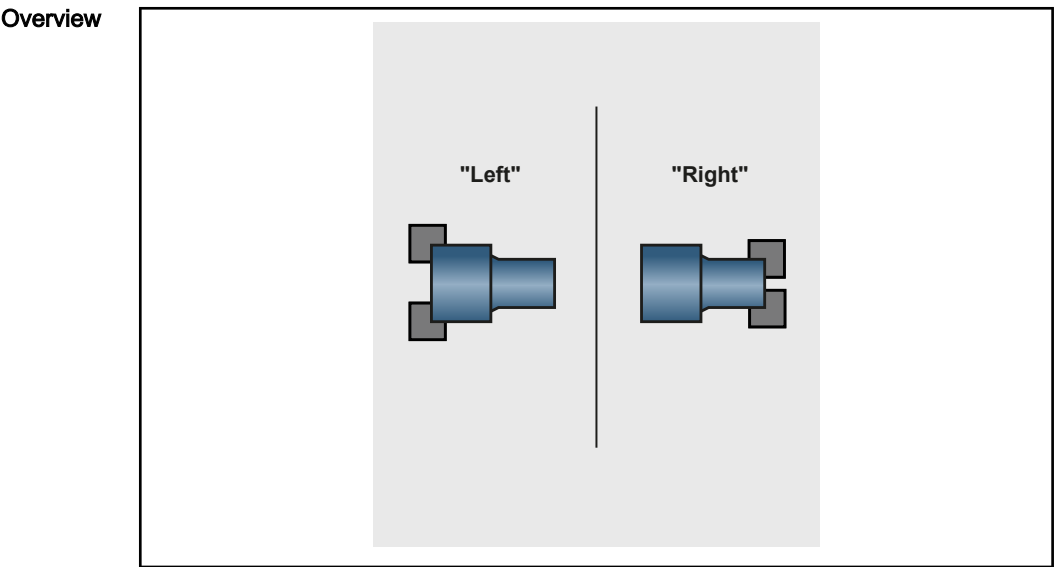


Fig. 9-11: STOCKSEL instruction Select workpiece holder

Parameter

Parameter	Name	Properties	Values	Pre-assignment
NAM	Name of workpiece holder	optional	max. 32 characters	first workpiece holder of the machine model

Tab. 9-19: Parameter call

Description	NAM name of the workpiece holder Defines the workpiece holder to be selected.
Special features	The parameter does not have an address.
Procedure/processing	The new selection is applied immediately in the current channel until cancellation of the selection.

Example Two workpiece holders are available in a 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
```

9.11 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

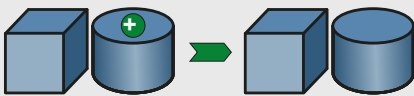
Tab. 9-20: Parameter call

ADD	Effect
0	Replace
1	Add

ADD=0



ADD=1



Tab. 9-21: Meaning of parameter ADD

Description

ADD Effect of the raw part definitions/imports

Defines the workpiece holder to be selected.

Special features

The parameter does not have an address.

Program instructions for simulation



In the present version, adding cannot neither be applied to workpieces in volume modeling nor to finished parts.

If "Adding" is active, a replacement can also be achieved by deleting the geometry concerned in advance.

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

Example A clamping device consisting of two parts is to be imported.

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
```

9.12 TOOL/DRILL Drill, countersink, reamer

Syntax TOOL("DRILL",<D>,<A>,<H>)

Brief description Generates the graphic in the simulation for a twist drill, tapping device, reamer, face or flat countersink (without spigot), independent of the exchanged tool.

Overview



Fig. 9-12: 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. 9-22: Parameter call

Description

D diameter

Determines the drilling diameter.

A Apical angle

Program instructions for simulation

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 modal states of the NC at the time of the cycle processing. Thus, it is imperatively necessary to activate the tool corrections and the correct interpolation level **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 negative 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 round.

Procedure/processing

The tool description superimposes the simulation of the standard tool exchange via the tool change subroutine and - depending on the last tool - generates a new graphic at the tool node that is assigned to the calling channel.

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

Example

For a twist drill, diameter 10 mm, a tool description is to be used.

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
```

9.13 TOOL/CDRILL Step drill, centering drill

Syntax

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

Brief description

Generates the graphic in the simulation for a multicut step drill or centering drill, independent of the exchanged tool.

Overview

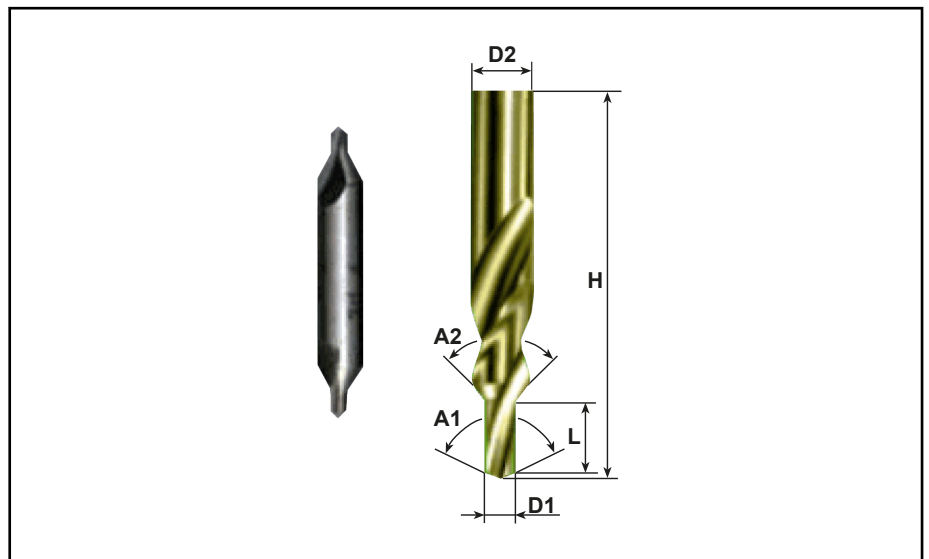


Fig. 9-13: TOOL/CDRILL tool description Step drill, centering drill

Program instructions for simulation

Parameter

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

Tab. 9-23: 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 modal states of the NC at the time of the cycle processing. Thus, it is imperatively necessary to activate the tool corrections and the correct interpolation level **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 negative 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 round.

Procedure/processing

The tool description superimposes the simulation of the standard tool exchange via the tool change subroutine and - depending on the last tool - generates a new graphic at the tool node that is assigned to the calling channel.

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

Example

A centering drill, diameter 2.5 mm, is to be used for a centering with 9 mm depth. For this purpose, its useful length of 10 mm is sufficient. It is a centric bore at a lathe; for it, you usually activate level G18. In the simulation configuration, you therefore have to define by means of programming that with drilling tools and active level G18, the orientation level G17 is set.

Program instructions for simulation

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
```

9.14 TOOL/BBAR Boring bar, backward counterbore

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

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

Overview

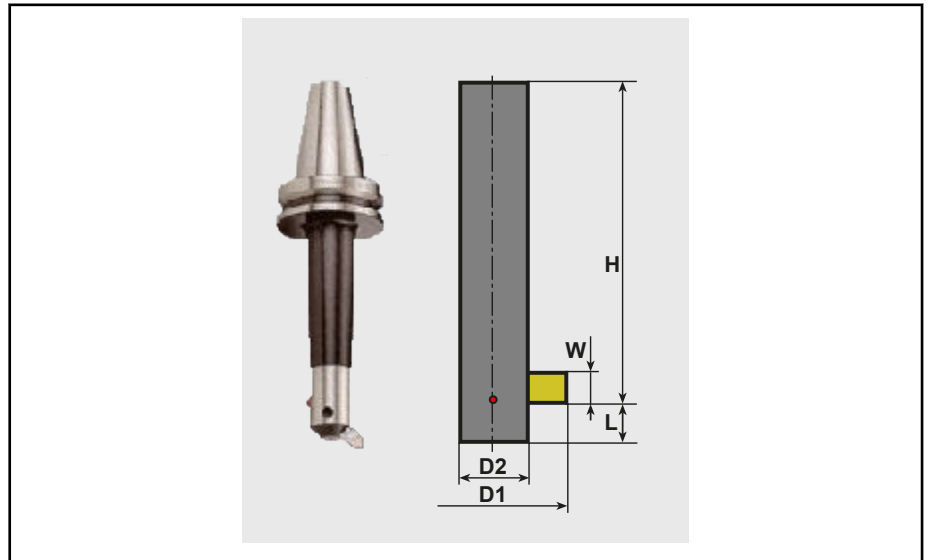


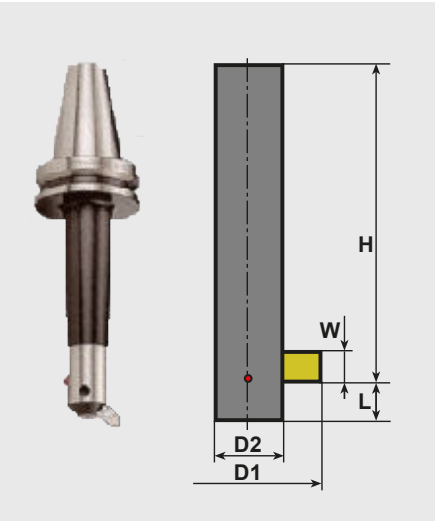
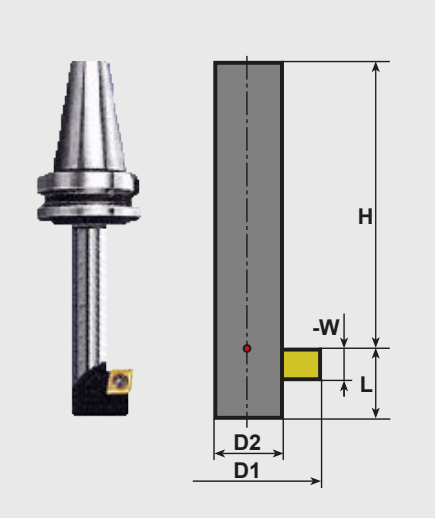
Fig. 9-14: TOOL/BBAR tool description Boring bar

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. 9-24: Parameter call

Program instructions for simulation

W	Effect
>0	Boring bar 
<0	Backward counterbore 

Red point Position of the tool zero point

Tab. 9-25: Meaning of parameter W

Description

D1 Diameter

Determines the drilling diameter.

W Cutting edge width

Defines the width of the chisel and defines by means of the sign whether a boring bar or a backward counterbore is concerned.

D2 Shaft diameter

Defines the diameter of the shaft.

H Shaft length

Defines the useful length of the tool.

L Loss length

Defines a protruding spigot.

Program instructions for simulation

Special features



Boring bars and backward counterbores can only be used in the simulation with limitations 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 modal states of the NC at the time of the cycle processing. Thus, it is imperatively necessary to activate the tool corrections and the correct interpolation level **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 negative 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 round.

Procedure/processing

The tool description superimposes the simulation of the standard tool exchange via the tool change subroutine and - depending on the last tool - generates a new graphic at the tool node that is assigned to the calling channel.

The tool graphic is maintained 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 drilled open using a boring bar with diameter 40 (without loss length). Afterwards, a backward counterbore with a diameter of 44 mm (26 mm deep) is to be produced. 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
```

9.15 TOOL/MILL Cylindrical, angular and ball cutter

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

Brief description

Generates the graphic in the simulation for a milling tool of the following types, independent of the exchanged tool:

- End milling cutters, groove milling cutters,
- Plain and shell end milling cutters,
- Dove tail milling cutters,
- Angle milling cutter (with 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.

Program instructions for simulation

Overview

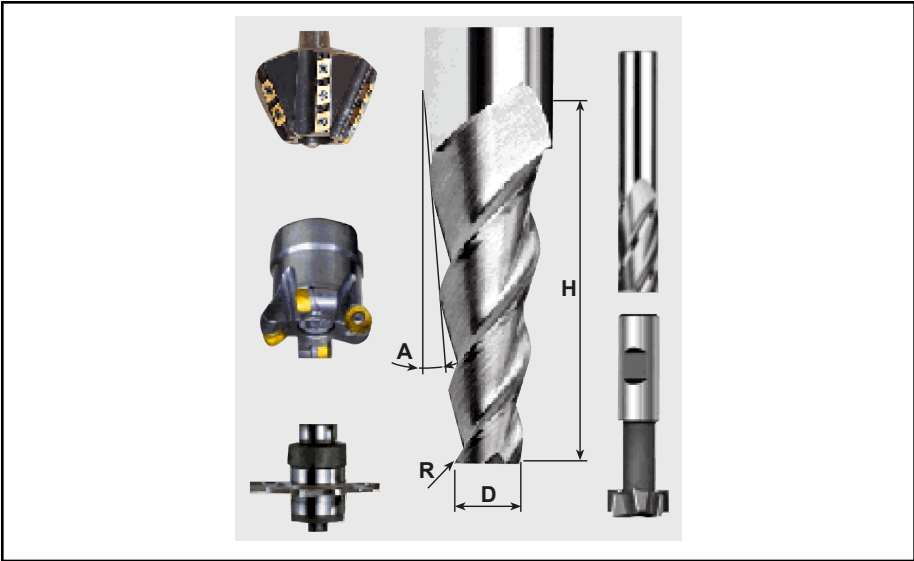


Fig. 9-15: TOOL/MILL tool description cylindrical or bevel cutter with optional corner radius

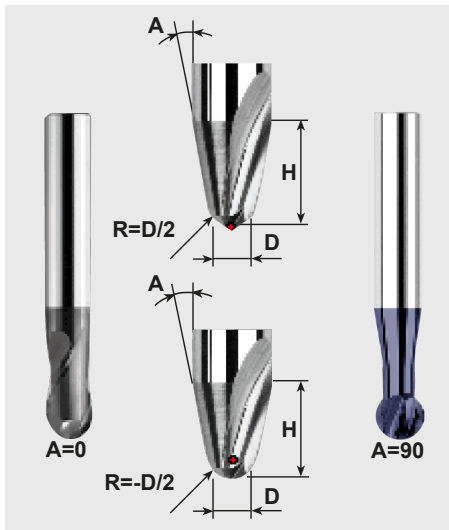
Parameter

Parameter	Name	Properties	Values	Pre-assignment
D	Diameter	optional	>0	double current tool radius correction
R	Corner/ball radius	optional		0
H	Length	Mandatory	>0	
A	Taper angle	optional	-90<A<=90	0

Tab. 9-26: Parameter call

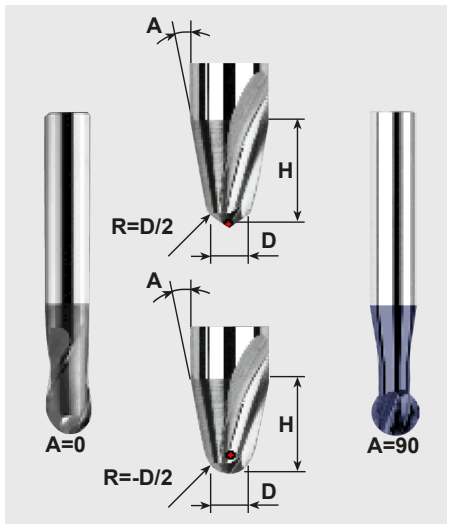
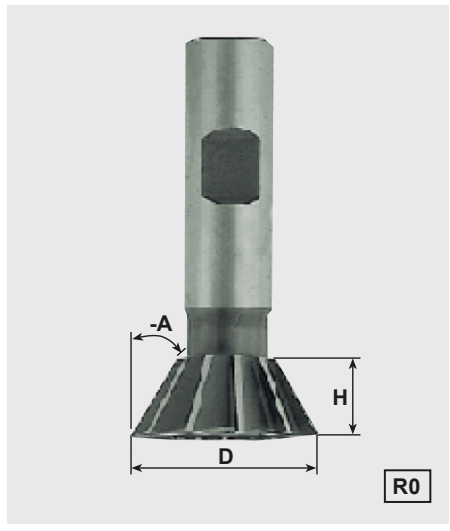
R	Effect
0	No corner radius. With H<=D side milling cutter, saw blade or T-groove milling cutter.
0<R<D/2	Corner radius
D/2	Ball cutter zeroed at the tool tip.

Program instructions for simulation

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

Red point Position of the tool zero point

Tab. 9-27: Meaning 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. 9-28: Meaning of parameter A

Description

D diameter

Determines the diameter at the front.

R Ball/corner radius

Program instructions for simulation

	Defines the corner radius and defines by means of the sign whether a ball cutter is zeroed in the ball center.
	H Length
	Defines the length of the cutting edge.
	A Taper angle
	Defines the taper angle and/or defines whether a ball-end cutter or dove tail milling cutter is at hand.
Special features	The parameters do not have any address. The generation of the tool graphic is based on modal states of the NC at the time of the cycle processing. Thus, it is imperatively necessary to activate the tool corrections and the correct interpolation level 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 negative 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 round. 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 exchange via the tool change subroutine and - depending on the last tool - generates a new graphic at the tool node that is assigned to the calling channel. The tool graphic is maintained until it is revoked with the cycle call without parameter. (TOOL or TOOL ()).
Example	A rectangular pocket is to be produced using a groove milling cutter diameter 6 mm x 20 mm useful length. The D correction comprises the current milling cutter radius; thus, the diameter D in the tool description can be omitted. <i>Program:</i> <pre> 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 </pre>

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

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

Program instructions for simulation

Overview

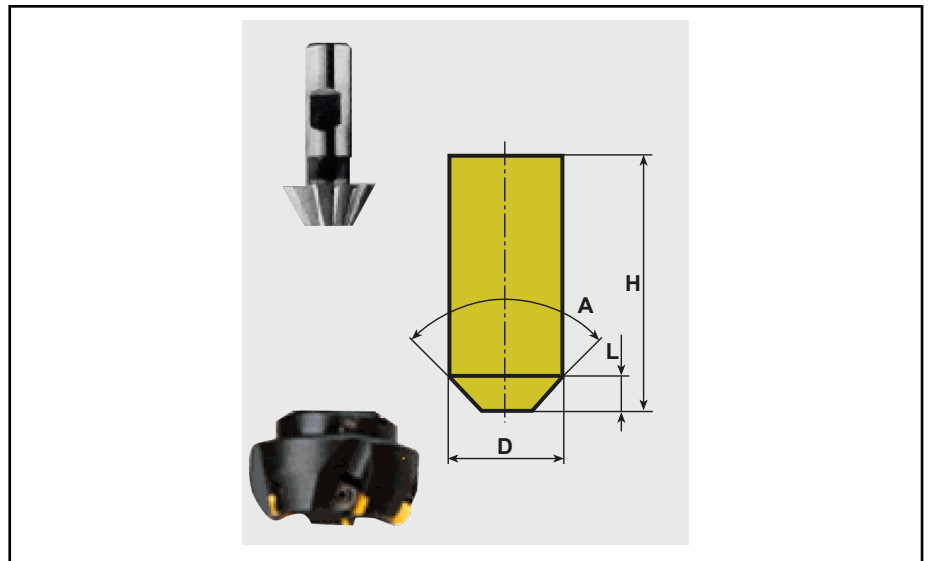


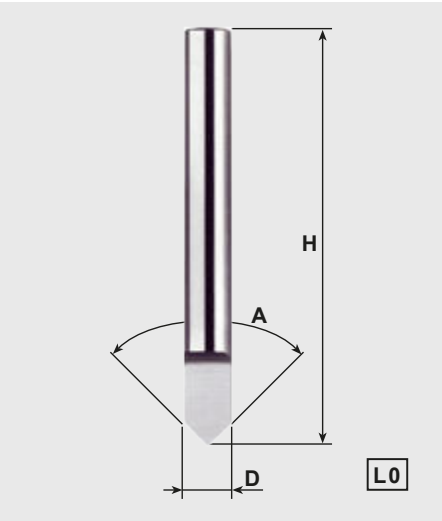
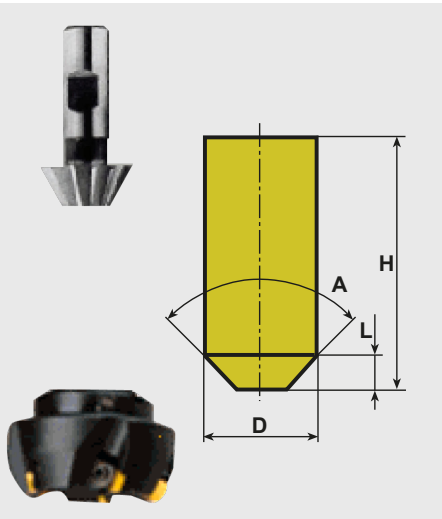
Fig. 9-16: TOOL/CHAMFER tool description Milling cutter with chamfer

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. 9-29: Parameter call

Program instructions for simulation

L	Effect
0	Pointed tool 
>0	Tool with chamfer 

Tab. 9-30: *Meaning of parameter L*

Description

- D diameter**
Defines the outer diameter.
- A Taper angle**
Defines the angle of the tip or the chamfer.
- H Length**
Defines the length of the cutting edge.
- L Chamfer length**
Defines the size of the chamfer and/or defines whether a pointed tool is at hand.

Special features The parameters do not have any address.

Program instructions for simulation

The generation of the tool graphic is based on modal states of the NC at the time of the cycle processing. Thus, it is imperatively necessary to activate the tool corrections and the correct interpolation level **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 negative 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 round.

Procedure/processing

The tool description superimposes the simulation of the standard tool exchange via the tool change subroutine and - depending on the last tool - generates a new graphic at the tool node that is assigned to the calling channel.

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

Example

A rectangular area is to be machined by means of a face milling head with 125 mm diameter. The adjustment angle of the cutting edge is 45°, its height 5.5 mm. The D correction comprises the current milling cutter radius at the front which can be used by the cycle 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
N40 G0 X100 Y0 ; Approaching and milling
N50 G80 Z200 M5 ; Leaving and spindle stop
N60 TOOL ; Deactivate tool description
```

9.17 TOOL/CRMILL Quarter round profile milling cutter

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

Brief description Generates the graphic in the simulation for a profile cutter with concave rounding, independent of the exchanged tool.

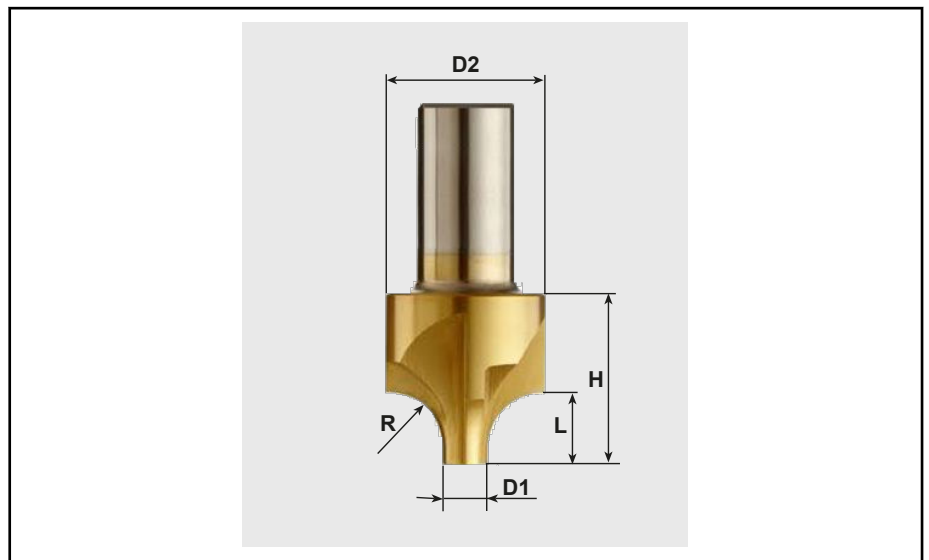
Overview

Fig. 9-17: TOOL/CRMILL tool description Quarter round profile milling cutter

Parameter

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

Program instructions for 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. 9-31: 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

Defines 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 modal states of the NC at the time of the cycle processing. Thus, it is imperatively necessary to activate the tool corrections and the correct interpolation level **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 negative 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 round.

Procedure/processing

The tool description superimposes the simulation of the standard tool exchange via the tool change subroutine and - depending on the last tool - generates a new graphic at the tool node that is assigned to the calling channel.

The tool graphic is maintained 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) is to be provided with a quarter round profile, radius 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 with active milling cutter radius correction.

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

```

9.18 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 in the simulation for the layer of a grinding wheel, independent of the exchanged tool: The profile (left and right one, if applicable) can be determined by means of parameters.

Overview

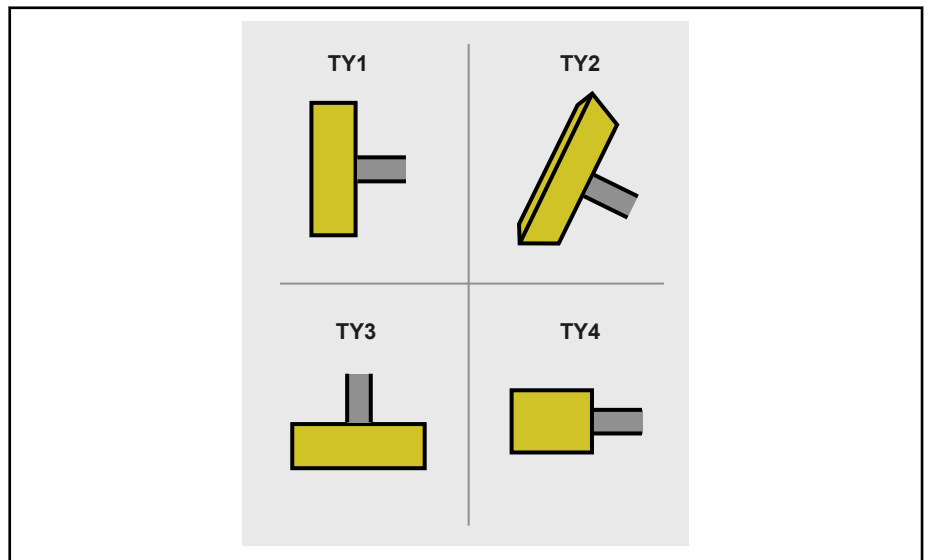


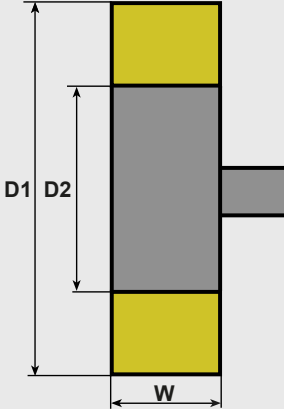
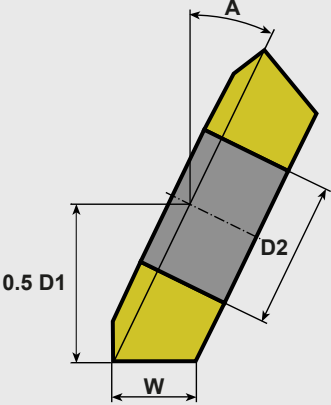
Fig. 9-18: TOOL/GRIND tool description Grinding wheel

Geometry parameters

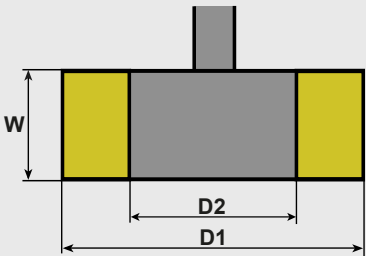
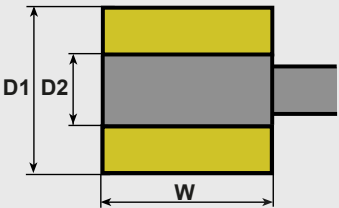
Parameter	Name	Properties	Values	Pre-assignment
TY	Wheel type	optional		1
P	Profile number	optional		for TY1: 2, for TY2: 201, for TY3: 301, for TY4: 401,
D1	Outer diameter	optional	<>0	double length correction value in Z or X
D2	Inside diameter, layer	optional	>0	complete grinding body
W	Wheel width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

Tab. 9-32: Call of the parameters defining the geometry

Program instructions for simulation

TY	Effect
1	Straight wheel 
2	Inclined wheel 

Program instructions for simulation

TY	Effect
3	Face plate 
4	Internal wheel 

Tab. 9-33: Meaning 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 axes of rotation.

P Profile number

Defines the wheel form and the possible parameters for the definition of the dimensions of the profile and/or the left and right profile.

D1 outer diameter

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

D2 Inside diameter, layer

Defines the inside diameter of the grinding layer.

Program instructions for simulation

W Wheel width

Defines the width of the wheel.

A Angle of inclination

Defines the inclination of an inclined wheel.

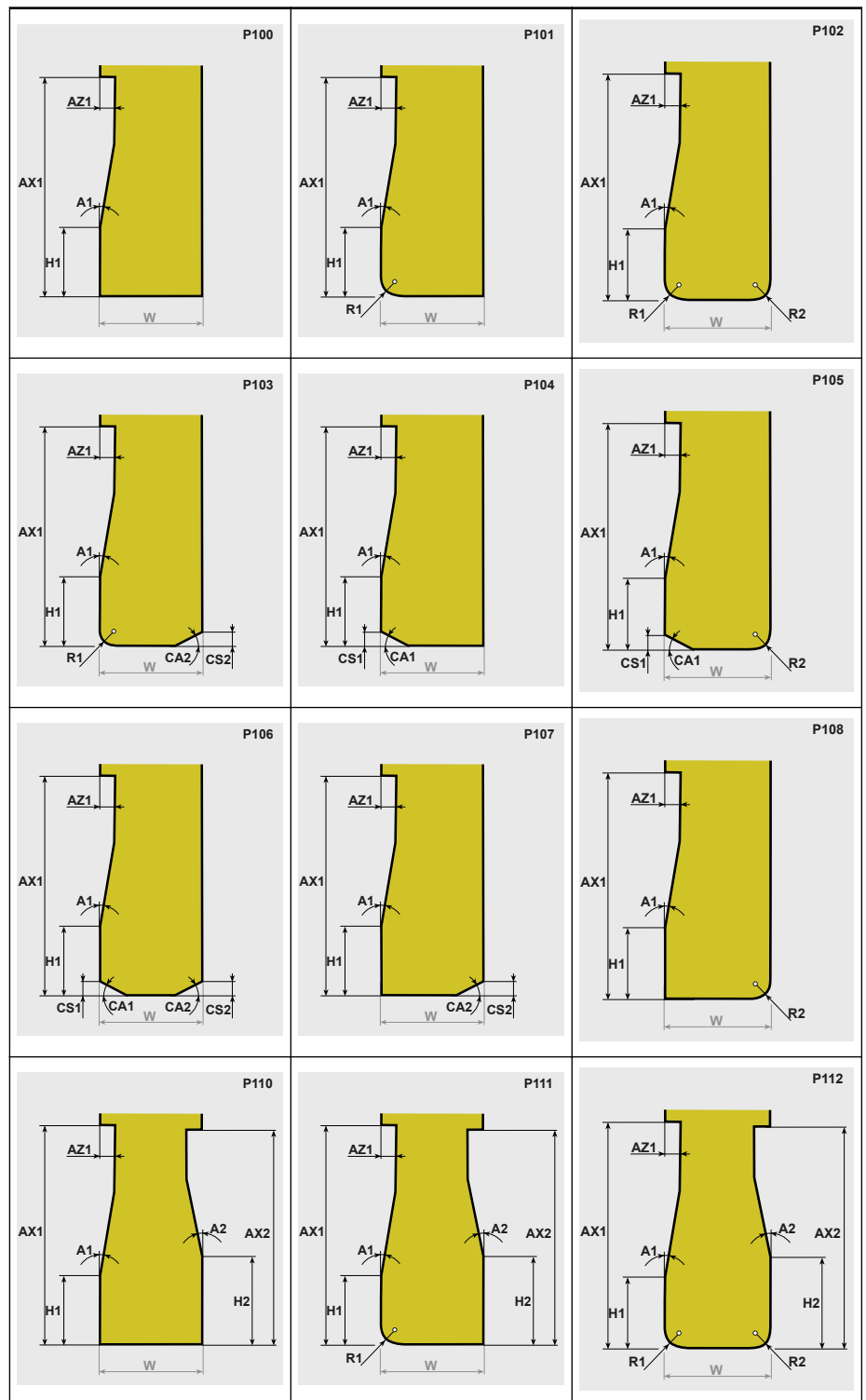
Wheel forms for straight wheels TY1 part 1



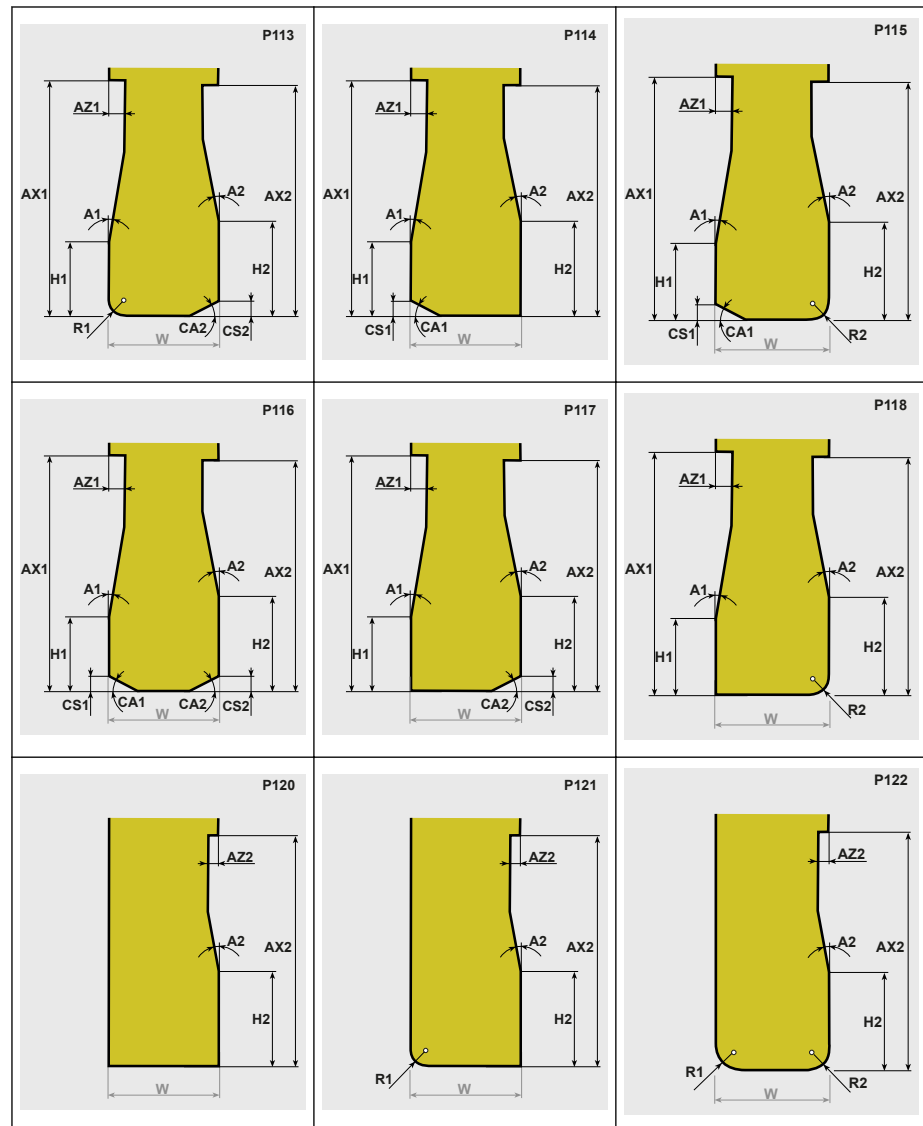
Tab. 9-34: Wheel forms for profile number P from 1 to 25

Program instructions for simulation

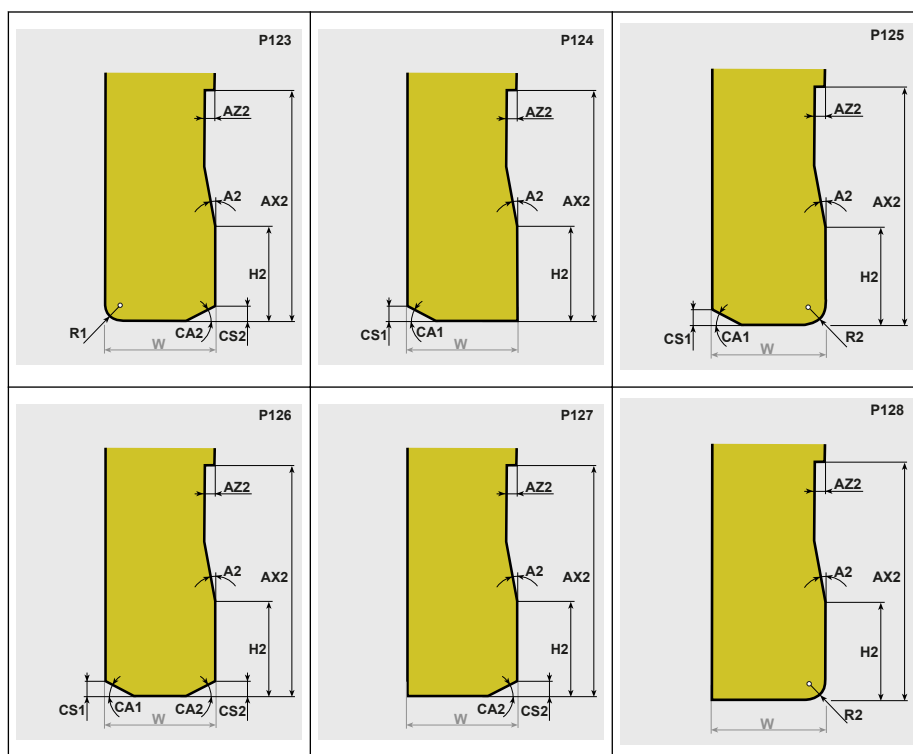
Wheel forms for straight wheels TY1 part 2



Program instructions for simulation

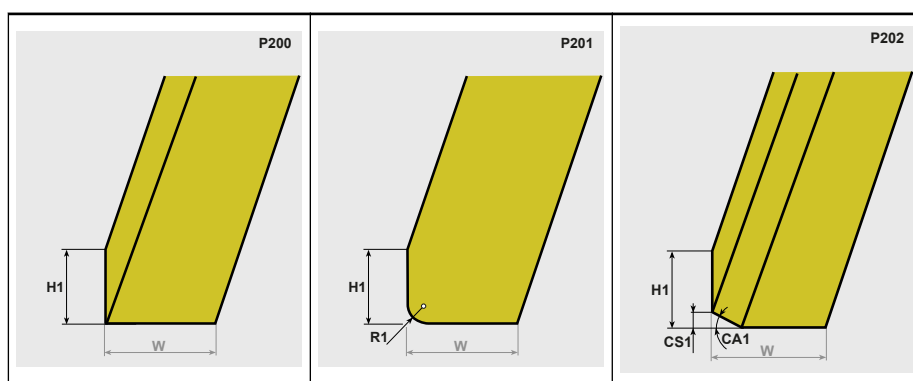


Program instructions for simulation



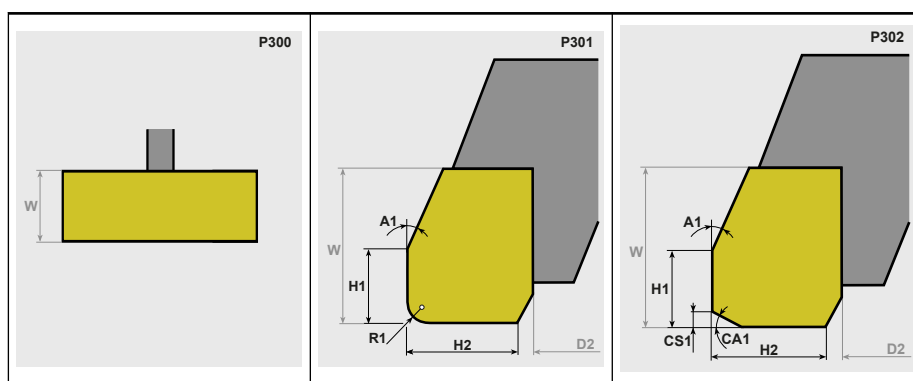
Tab. 9-35: Wheel forms for profile number P from 100 to 128

Wheel forms for inclined wheels TY2



Tab. 9-36: Wheel forms for profile number P from 200 to 202

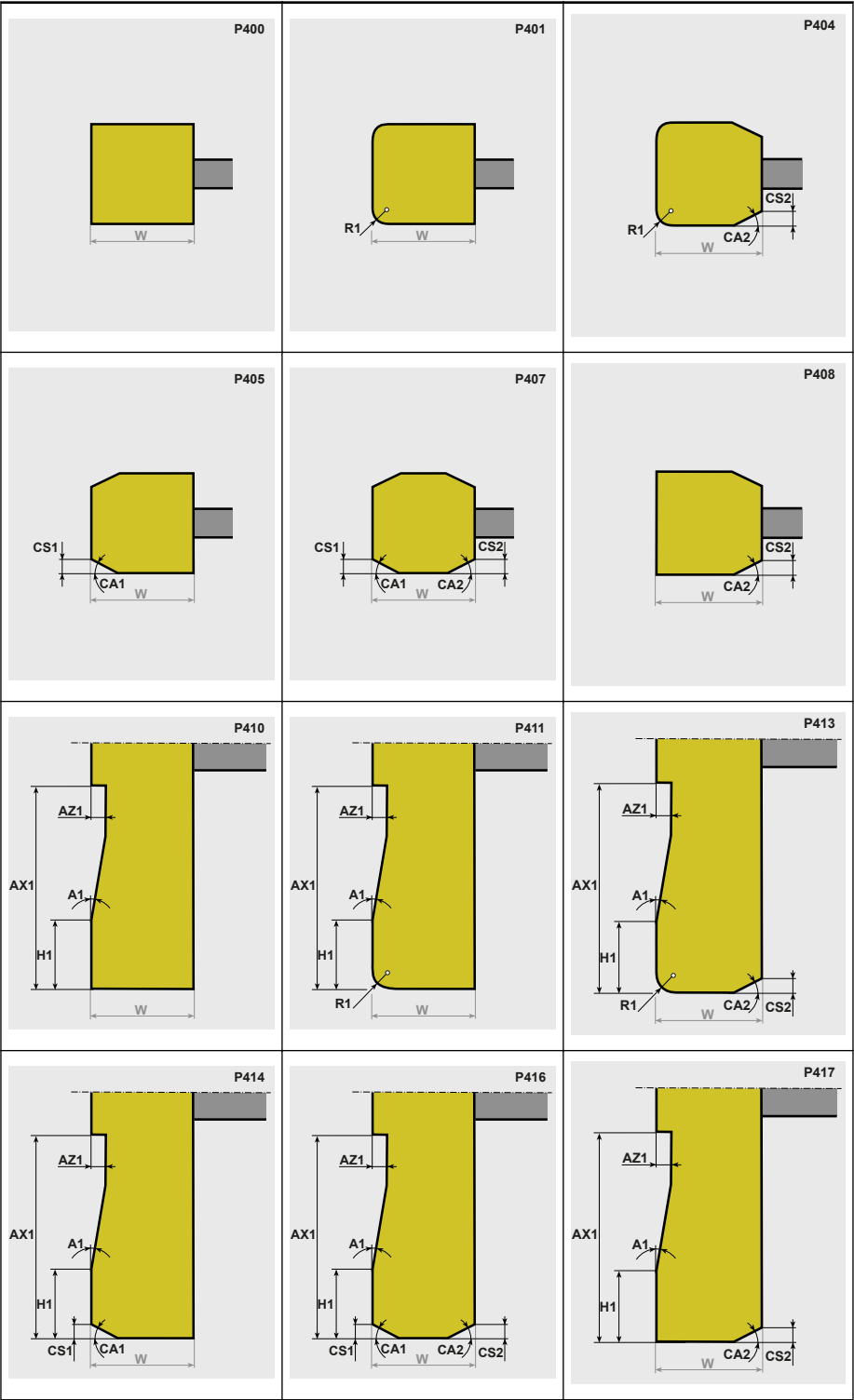
Wheel forms for face plates TY3



Tab. 9-37: Wheel forms for profile number P from 300 to 302

Program instructions for simulation

Wheel forms for internal wheels TY4



Tab. 9-38: Wheel forms for profile number P from 400 to 417

Program instructions for simulation

Parameters for (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. 9-39: Call of the parameters defining the (left) profile

Description of the parameters for the (left) profile

H1 Shoulder height in X

Defines 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 will automatically be corrected to the minimum shoulder height.

R1 Radius

Defines 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

Defines 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.

Program instructions for simulation

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. 9-40: Call of the parameters defining the right profile

Description of the parameters for the right profile

H2 Shoulder height in X

Defines 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 will automatically be corrected to the minimum shoulder height.

R2 Radius

Defines 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

Defines 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 form - are not relevant do not have any effect, they are, however, subject to a value range check.

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

Program instructions for simulation

Procedure/processing The tool description superimposes the simulation of the standard tool exchange via the tool change subroutine and - depending on the last tool - generates a new graphic at the tool node that is assigned to the calling channel.
 The tool graphic is maintained until it is revoked with the cycle call without parameter. (TOOL or TOOL ()).

Example The tool description for a conical cup wheel according to the drawing is to be prepared.

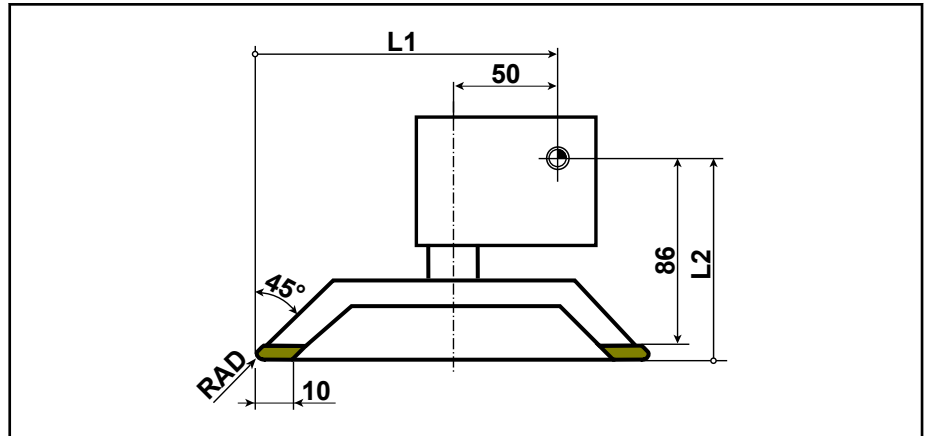


Fig. 9-19: Example: tool description for a cup wheel

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
```


10 Configuration

10.1 Overview

Contents The configuration contains important information necessary for the basic functioning of the simulation, which is usually to be generated in the application (commissioning of the simulation). 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's requirements.

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



You should always provide the machine model first and then start the simulation. Otherwise, there is the risk of data loss when the simulation tries to connect the axes of the standard machine model. If this happens unintentionally as you have the machine model and a suitable configuration in the project and you only did not provide the model correctly, please transmit the visualization data after the import of the model once again and activate them (IndraWorks Engineering, context menu of the visualization device).

Editing The "MTX.Simulation.Options.xml" file is changed using a text editor. To do so, the interface creates a temporary file with the current settings and configurations that can be directly changed in the XML format. More detailed information for changing the options can be found in the operating instructions of the simulation, [chapter 6.8 "Tools menu" on page 25](#).

If contents are deleted, inadmissible values are entered or the XML syntax was violated, default values take effect.

The element names usually reflect their meaning. Over many elements, there are English comments describing their meaning and/or the value range.



Within text blocks (between beginning and end of tag) and attributes some characters are not admissible which are to be replaced 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 language based on CPL (Custom Programming Language, see documentation "IndraMotion MTX Programming Manual", chap. "Basics of NC programming"), which are then processed interpretatively if, e.g., the tool

Configuration

is changed or an axis is provided with a new position value. In this way, the simulation can be adjusted to manifold operations in an applicative way.

To this extent, the simulation initializes input and output variables in a context-dependent manner, which can be changed in the instruction part by using auxiliary variables, if applicable. Afterwards, the simulation uses the lastly assigned values of the input and output variables.

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

The program lines are not introduced by a number (CPL block number). Several instructions in one line are to be separated by colon.

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

Instruction regarding the program structuring

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

Simple variables

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



No arrays can be created of simple variables. As opposed 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 in a context-dependent form as input and output variables, only by the system; to these variables, there is read-access and sometimes also write-access.

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



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

Another important structure is the DBT1 tool data record which is only provided in the "Tool graphic generation" context. Here, the `Rec` substructure can be optionally specified.

Examples:

```
PX=SD.SysARA[1].Pos1 : REM read-access to a system date
TNAME$=DBT1.Rec.Hd.SKQ : REM read-access to the ident in
the tool data record
```

Operators

Configuration

Operators	Instruction
Assignment	=
Addition (arithmetic and connection of strings)	+
Subtraction	-
Multiplication	*
Division	/
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	>=
greater than	>
unequal to	<>
less than/equal to	<=
less than	<

Tab. 10-1: Operator list



Here, expressions using `NUL` (undefined variable in CPL) are not admissible.

Functions

The arguments are enclosed by round brackets. Multiple arguments are separated by commas.

Functions	Instruction code
Sine	SIN(<Digit>)
Cosine	COS(<Digit>)
Tangent	TAN(<Digit>)
Arcus sine	ASIN(<Digit>)
Arcus cosine	ACOS(<Digit>)
Arcus tangent	ATAN(<Digit>)
Addition (arithmetic and connection of strings)	+
Absolute value	ABS(<Digit>)
Conversion of a floating-point number into an integer by cut-off	INT(<Digit>)
Conversion of a floating-point number into an integer by rounding up or down	ROUND(<Digit>)

Configuration

Functions	Instruction code
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\$ must not stand on the left side of an assignment.	MID\$(<StringExpression>,<StartIndex>{,<NumberOfCharacters>})
Conversion of a string into capital or lowercase letters	MCASE\$(<StringExpression>,"U" "L")

Tab. 10-2: Function list

Comment

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

Tab. 10-3: List of comment instructions

10.2 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 not complied with or special features are to be realized, knowledge of the generation and influencing of the generation are necessary. The following section also provides information that is necessary for the import of STL models for tools.

The applicative influencing of the tool graphics calculation is effected by a program section the use of which is described in more detail in [chapter "Machine" on page 98](#), while in the following, content-related and geometric aspects are addressed.

How is a tool integrated into the simulation?

Normally, the programmed tool is exchanged while a tool change is completed. In a machine tool with revolver, the place and/or the sector of the programmed tool change, the combination of which always refers to the data record of the new tool (data record key). In the tool change at a machine with tool magazine and when editing a displayed tool in the tool editor, the tool data record under a certain place/sector is changed automatically. The simulation registers the change in the data record key and/or the data record, initializes the input and output parameters, starts the mentioned program section and calculates the tool graphic from the return parameters. In this way, 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 a channel, only one of them will generate the tool path.

There are also two other possibilities to display a tool graphic, however only for the simulation:

- For revolvers, the access to tool data records for the remaining (not exchanged) tools can be defined via database queries; the latter provide the tool information for the entire revolver fitting.
- The programmed tool the graphic of which is calculated from the related data record of the tool management can be temporarily superimposed over the `TOOL` cycle by means of 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 an axis of rotation
 - **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 in the following ways:

- by directly specifying segments and arcs
- by parameters which the simulation considers in the contour calculation. For the parameters, there are context-dependent default values and/or dependencies with regard to other variables so that they principally don't have to be specified in the configuration



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



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

Parts of a tool from the simulation's point of view

The graphic of a tool may consist of a maximum of four parts. Every part may be represented by a maximum of 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; by means of this variable, reference can be made to an STL file by means of a file name. Alternatively, you can define the contour to be rotated/extruded and/or the parameters there.

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

This part is only able to remove and/or apply material from/to the workpiece model or to form 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, which is invisible itself.

Configuration

- **Shaft** (non-cutting part, variable `SHAFT$`):
The non-cutting part is mostly formed by a fixed unit with the cutting part.
With drilling and milling tools with indexable inserts, this is the part of the basic body that is not fitted with inserts.
With drilling and milling tools with a spigot (e.g. counterbore) or protruding tightening nut, this part is considered as shaft. The remaining part of the (physical) shaft above the cutting edge is then assigned to the arbor.
- **Arbor** (arbor, variable `ARBOR$`):
All tool clamps and fixtures such as chucks, collet chucks, shell-type cutter arbors, are attributed to the arbor section. The arbor 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.
With drilling and milling machines, this is mostly a steep-angle taper. It is sufficient to only model the part protruding from the spindle.
In punching processes, the holder may be the tool guide on which the plunger acts.

All tool parts except for the cutting edge are functionally equal. They differ in their color and as regards 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

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. So usually, the upper limitation of the cutting edge and the lower limitation of the shaft are on the same level. When you import meshes, it may therefore be necessary to omit internal parts and/or to adjust the geometries accordingly 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.

With rotation-symmetric tools, the meshes of **arbor** and **holder** are also stacked like a tower.

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

Grinding tools:

The tool components are arranged so that the zero points of the meshes face in the direction of the rotation axis at the tool zero point. Penetration is also possible.

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).

Tool components according to tool types

Configuration

Name of type/ group	Type number (1) / type in tool description (2)	Cutting edge CUTT\$	Shaft SHAFT\$	Arbor ARBOR\$	Holder HOLDER\$
Standard tool	0	Import, rotated contour or generic from technology and cutting edge data. Is always created	Import, rotated contour or generic. Is only created if the length is suffi- cient and with a cutting edge, ge- nerically generated from technology data	Import, rotated contour or generic from parameters in ARBOR\$. Is only created if the length is suffi- cient	Import or generic. Is not automatically generated and only created if the length is sufficient
Drilling tools	1000..1016				
Milling tools	2000..2017				
Turning tools	3000..3027	Not implemented	--	--	--
Measuring probe	5001..5003	Not available	Spheres (ruby) and handle. Only generically from the cutting edge data	Import, rotated contour or from pa- rameters in AR- BOR\$. Is only created if the length is suffi- cient	Import or generic. Is not automatically generated and only created if the length is sufficient
Special tool gripper	6000	Not implemented	--	--	--
Jet cutting tool	6001	Only generically, from technology and cutting edge data	Results in a trans- parent "flame" in cutting edge color. Not available in the standard. Only import or rota- ted contour possi- ble	Import or rotated contour. Is not automatically generated	
AM tool	6002	Torus cross-sec- tion of the material to be applied. Import, rotated contour or from technology data. Is always created	Import, rotated contour or generi- cally, from technol- ogy data. Is only created if the length is suffi- cient	Import, rotated contour or from pa- rameters in AR- BOR\$. Is only created if the length is suffi- cient	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 section with- out hole) or from technology data. Is always created	Not available	Not available	Import or generic

Configuration

Name of type/ group	Type number (1) / type in tool description (2)	Cutting edge CUTT\$	Shaft SHAFT\$	Arbor ARBOR\$	Holder HOLDER\$
Drilling tools	"DRILL", "CDRILL", "BBAR"	Only generically from the parame- ters of the tool de- scription and the tool correction	Only generically suitable for the cut- ting edge.	Import, rotated contour or from pa- rameters in AR- BOR\$.	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 suffi- cient	
Grinding tools	"GRIND"		Only import or rotated contour possible. Is not automatically generated		

- (1) The type number is contained 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 by means of the DESCR.TYPE element

Tab. 10-4: Tool components depending on the tool type

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

In the assignment of the variables for cutting edge, shaft, arbor and holder, the subsequently listed syntactic rules are to be observed:

- **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). As separator between the figures, the comma is used whereas the comma may be preceded or followed by any number of spaces. Constant coordinates don't have to be written.
The contour must always run in the positive diameter range, usually 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 means of 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 don't have to be specified in the contour path.
By default, the orientation of the rotary axis is aligned in parallel to Z. The orientation can be changed with a self-sustaining parameter ORI that has to be listed at a user-defined position, separated by comma, within or at the end of the contour.
 - ORI0: rotary axis, longitudinal Z (default)
 - ORI1: rotary axis, longitudinal Y

Configuration

- ORI2: rotary axis, longitudinal X

By default, the rotary axis goes through point [0,0,0]. The position can be changed with self-sustaining parameters X, Y, Z that are to be inserted at a user-defined position, separated by comma, within or at the end of the contour.

Example:

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

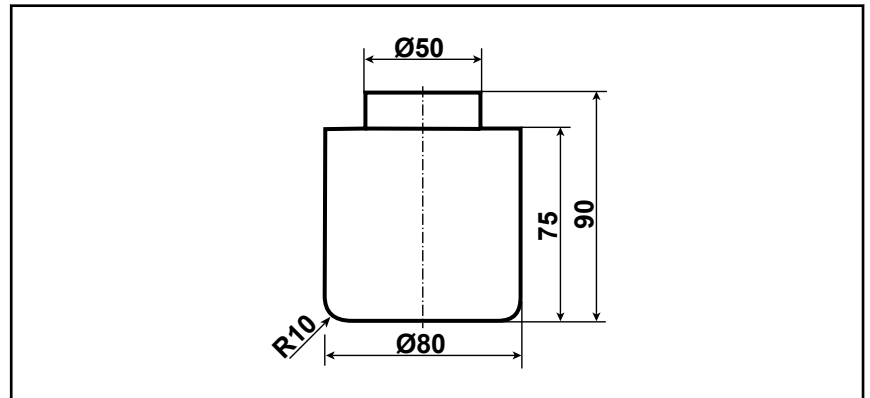


Fig. 10-1: Example: Housing of a measuring probe

```
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 development 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. If you want to select a different height direction, supplement parameter ORI:

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

The position of the base can be changed with self-sustaining parameters X, Y, Z that are to be listed at a user-defined position, separated by comma, within or at the end of the contour.

Example:

The cutting edge of a punching tool (upper part) has the dimensions according to the drawing. Its thickness extending in Z direction is to be taken from the **CUTHEIGHT** variable.

Configuration

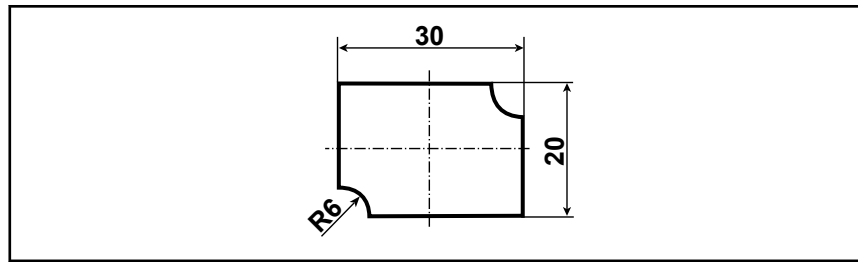


Fig. 10-2: Example: Punching tool

CUTT

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



As it is not possible to generate an open body by extrusion, please use a circular disk for the lower die of a punching tool instead.

- **Cuboid:**

An axis-parallel cuboid can be described by means of 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:**

A cylinder with optional inner boring can be written 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 center of the lower base (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 center of the lower base (default 0)
- ORI Rotary axis orientation (ORI0 longitudinal Z (default), ORI1 longitudinal Y, ORI2 longitudinal X)

Example:

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

Configuration

- **Multi-edge:**

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

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

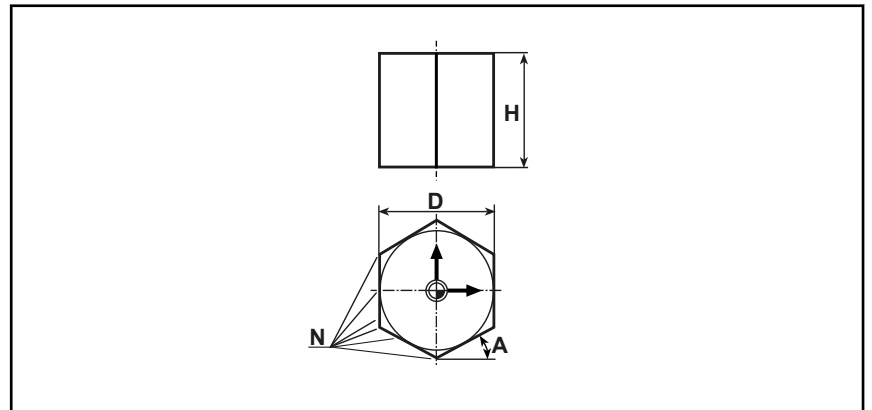


Fig. 10-3: Parameter of a multi-edge to be generated generically

Example:

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

- **Sphere:**

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

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

Example:

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

- **Parameters for a cylindrical arbor part:**

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

For this case, the following parameters are provided:

Configuration

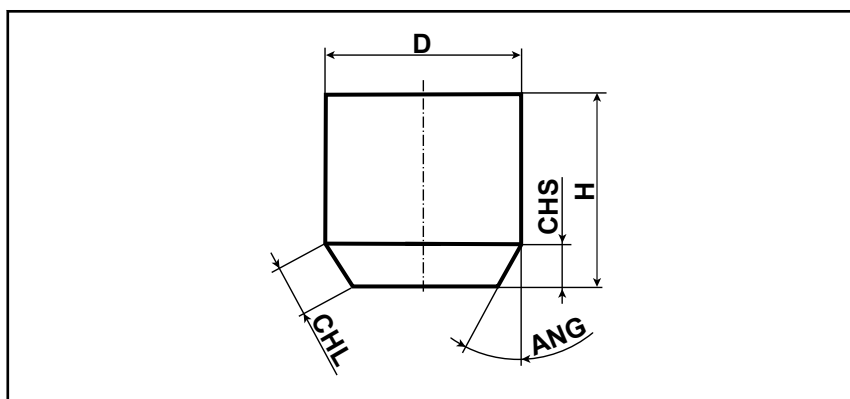


Fig. 10-4: Parameters for a cylindrical arbor

- 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 diameter of the holder.
- Chamfer angle **ANG**: Default value 45°
- Chamfer length **CHL** or alternatively chamfer section **CHS**: Without this specification, the chamfer size is adjusted to the cylinder length and the diameters at 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 cylindrical arbor part according to the standard is inserted and you can influence the latter using the parameters.

Example 1:

A cylindrical arbor 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 arbor geometry is to be generated

```
ARBOR$=""
```

**Tool length correction and orientation,
position of the mesh zero
points**

Until now, the generation of the tool form has been explained. 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), its axis of rotation is to be considered 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 spatial direction 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 level (usually G18). This is also true for grinding whereas here, the torus cross-section of the grinding layer in the interpolation level is decisive.

When programming and/or importing tool models for rotation-symmetric tools, **cutting edge, shaft and arbor** are to be defined in a coordinate system in which Z runs parallel to the axis of rotation and the positive Z direction points from the tip to the shaft. The definition of the specified geometries is also independent of the orientation and also of the **length corrections** (e.g. with an angle head). Only **the zero point of the cutting edge in longitudinal direction must comply with the position of the tool zero point**. The simulation will then transform the geometry according to the orientation and the length corrections.

The following applies for 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 will be arranged below the cutting edge (spigot or loss length). The arbor geometry must then include the actual shaft, if applicable
- In all other cases, the shaft is "put" onto the cutting edge (normal case)

To the orientation of the **holder**, the afore-mentioned rules for cutting edge, shaft and arbor 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 will be aligned with the top limitation of the holder (normal case)
- In all other cases, the zero point of the holder geometry itself will be on one level with the zero point at the tool holder

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

Complete examples with rotated and extruded contours

The following examples are to show how the generation of tool graphics can be adjusted and extended.

Example:

Double angle cutter

The generation of the tool graphic for a double angle cutter is to be completely implemented. The implementation is to be realized as extension of the "angle cutter" tool type and is to be dependent on the question whether the ident starts with the "DblAng" string. The milling arbor is to get a variable diameter which amounts to 80% of the cutter face diameter but is not larger than 50 mm. At the lower side, there is a tightening nut (spigot) the height of which is to amount to 12 mm.

For all tools, a holder geometry analogously to HSK 63 is to be agreed.

Configuration

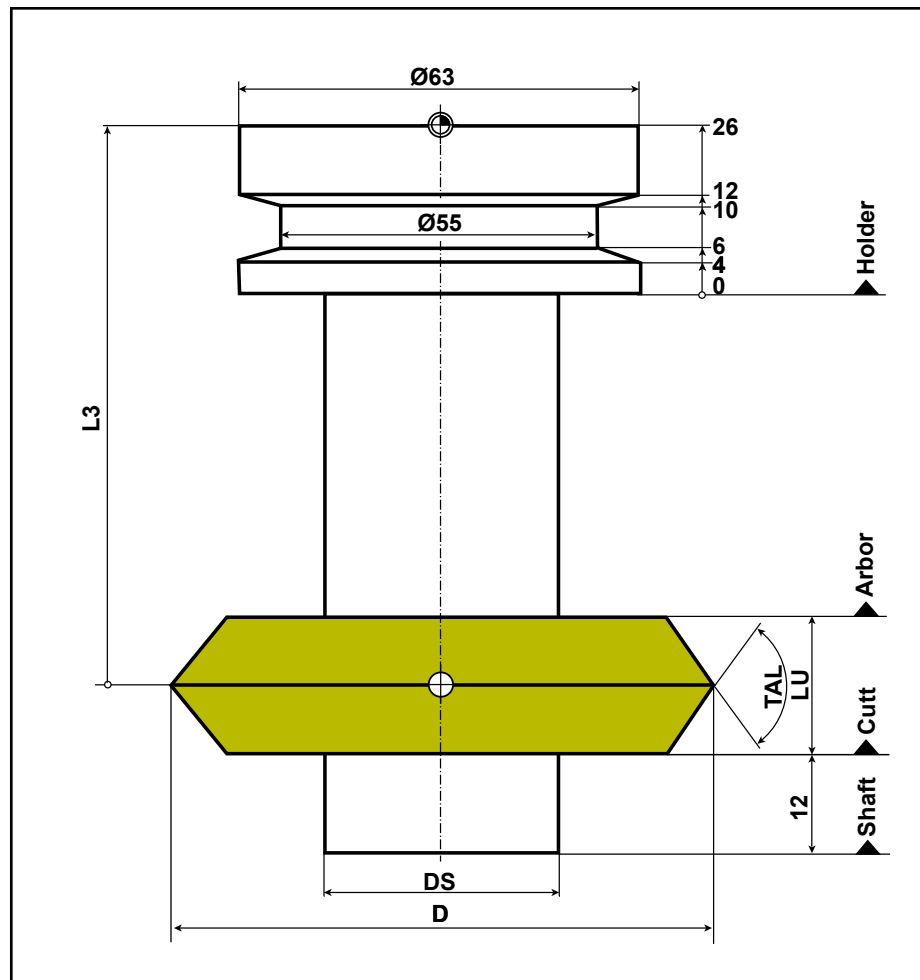


Fig. 10-5: Double angle cutter with holder HSK 63 (approximated geometry)



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. If you want to model it nevertheless, the zero length coordinate of the holder contour must lie at 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
Dl=D-LU/TAN(0.5*TAL)          : REM Calculate face diameter of cutting
                                ; edge

```

Configuration

```

; 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)+", "
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)

The generic graphic for a radial angle head including collet chucks is to be defined.

Apart from that, the tool graphic for a form countersink is to be implemented according to the drawing. The implementation is to be realized as extension of the "general tool" tool type and is to be dependent on the question whether the ident complies with the "FormCounterSink 12x6" string.

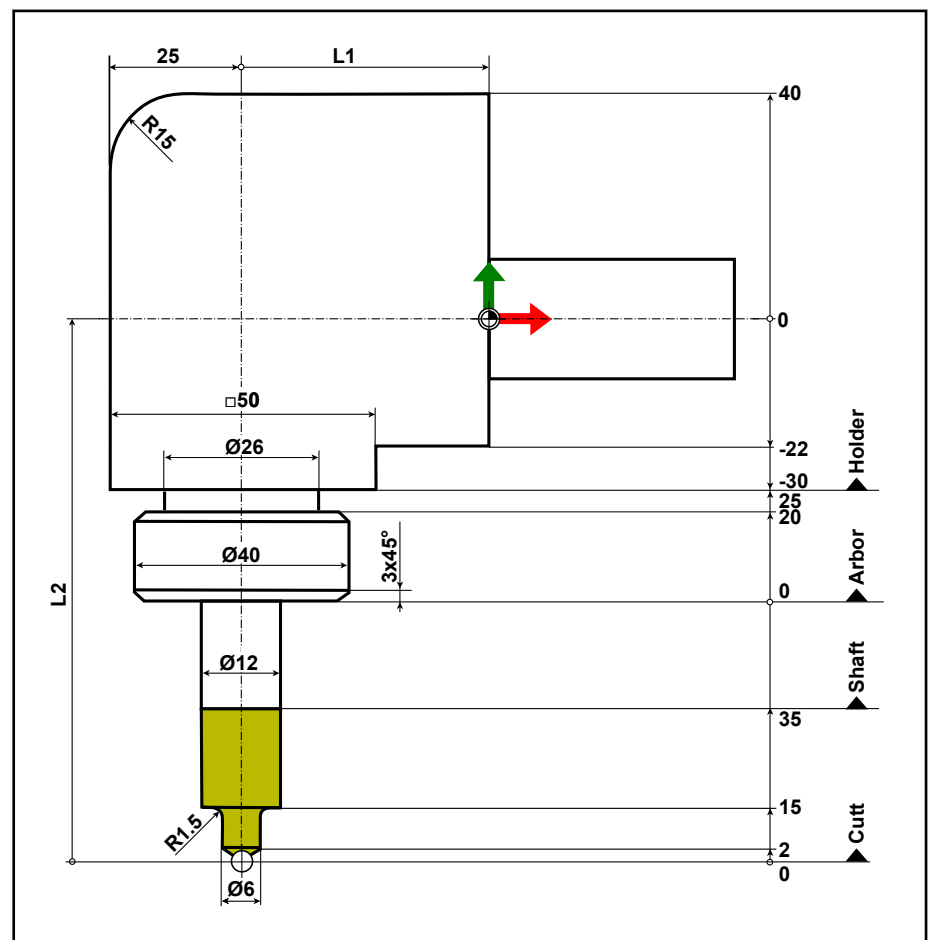


Fig. 10-6: Form countersink 12x6 with radial angle head

Configuration



The angle head mesh is to be generated 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 must 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.

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$(-L_1+25)+",,,-22,Y-25,TH50"
        ARBOR$="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
```

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

10.3.1 File structure

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

The former is followed by the <SimulationOptions> main element; below it, there are several sections (settings divided in topic groups).

Sections

Below the main element, there are the subsequently listed XML tags (sections). The function of every section is described and it is indicated whether pure options are concerned or configuration data that is to be specified in the commissioning.

- Tag <Optimizing> **optimization** (--> Options)

Adjustment of the simulation 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. Apart from that, there is a setting for tests, error analysis and records regarding error reproduction and settings for the dialog control.

- 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 effects 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?

Configuration

- How are tools activated/deactivated?
- Which criteria form the basis of 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. Apart from that, there are color schemes allowing for color assignment depending on a variable
- Tag <Punching> **punching** (--> configuration)
Special settings for the punching simulation
- Tag <AdditiveManufacturing> **generative production** (--> configuration)
Special settings for generic production methods (e.g. 3D printing)
- Tag <iWindow> **camera settings** (--> configuration)
The simulation graphics 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>
  <Optimizing>
    <!--*****-->
    <!--***** Optimizing *****-->
    <!--*****-->
    <!--
    ...
    -->
  </Optimizing>
  <Machine>
    <!--*****-->
    <!--***** Machine *****-->
    <!--*****-->
    <!--
    ...
    -->
  </Machine>
  <View>
    <!--*****-->
    <!--***** View Settings *****-->
    <!--*****-->
    <!--
    ...
    -->
  </View>
  <Colors>
    <!--*****-->
    <!--***** Colors *****-->
    <!--*****-->
    <!--
    ...
    -->
  </Colors>
  <Punching>
    <!--*****-->
    <!--***** Punching *****-->
    <!--*****-->
    <!--
    ...
    -->
  </Punching>
  <AdditiveManufacturing>
    <!--*****-->
    <!--***** Additive Manufacturing *****-->
```

Configuration

```

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

```

10.3.2 Configuration of the NC simulation

Machine

Main processing method Tag <MachingType>:

The **main processing method** influences the dialog control, particularly the question which raw part types are provided in the menu.

Possible values:

- Mill (default)
Mill machining center, e.g. machine tools for processing prism-shaped parts (e.g. laser processing). Provided raw parts: Cuboids and cylinders
- Turn
Lathe, e.g. machine tools for processing rotation-symmetric parts. Provided raw parts: Bar section and polygon
- Punch
Punching machine. Provided raw part: Sheet
The punching machine cannot be explicitly agreed. This process is automatically set if the "Punches applied" function in Macoda 8001 00010 has the value "1"
- AM
Machine tool for generative production processes. Provided raw parts: Cuboids and cylinders
With cuboid- and cylinder-shaped raw parts in the volume presentation, only the base areas are shown as "base". The raw part dimensions will describe the space within which the workpiece is generated.
- undefined
Processing 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, which are 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 contained in one single folder under the "IndraWorks\Simulation\MW Models" installation directory. In order to reference a machine model, specify the folder name as text. If there are several XML files with the specified main element, you can add the file name following a backslash.

Example:

The machine model is to be defined in the "IndraWorks\Simulation\MW Models\MachineCenter500\MachineCenter500.xml" file.

Configuration

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

or

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

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



When entering the machine model make sure that no line break is maintained between the pointed brackets.

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

Axis connection

Tag <AxisConnection>:

Below the **axis connection** tag, it can be defined for every axis of the kinetic tree (machine model) which drive is to move this axis.

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

- **Name:**
The name used to designate the axis in the machine model. Wild cards are not admissible
- **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 movement of the upper die of a punching machine is simulated
 - "PUNCHDOWN" - The movement of the lower die of a punching machine is simulated
 - "DOOR" - The movement is simulated by the "Open door" function, an intermediate level when hiding machine parts

In the element content of the <Axis> tag, you can store a program section influencing 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 part

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		

Configuration

Variable name	Type	Value range	Pre-assignment	Description
SD	Structure			Input variable with read access
STOCK	Structure		Current raw part data	Input variable with read access on the raw part dimensions

Tab. 10-5: 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 supplying the raw part dimensions, however not the current workpiece dimensions. As index, a string with the name of the workpiece holder in square brackets is to be used. The index must be a text constant, no expression. If no index is specified, the first workpiece holder of the machine model will be addressed.

STOCK variable

Element name	Type	Value range		Description
EMPTY	boolean	TRUE	Workpiece available	Feeds back the availability of a workpiece
		FALSE	No workpiece	
DMIN	real	>=0		Inside 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. 10-6: Elements of the STOCK input variable



The interpretation of the program sections negatively influences the simulation performance. Consequently, programs are only to be stored sparsely, particularly in the connection of 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 notified

Configuration

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 in order to avoid collision warnings. Otherwise, the Y position is to be corrected 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) is to correspond to the upper sheet side.

The stroke movement of the upper punching part is to be carried out by the model axis "Z", that of the lower part by "Z2". The tool is moved by a rotary axis "C" (drive 3) the simulated position of which is to be constantly 0 if the ident of the active tool starts with the string "MT" (sign of a multi-tool).

The "LeftDoor" and "RightDoor" model axes are operated by the "Open door" function.

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 the tool graphic is to be prepared
- What the query for the database query for the revolver fitting is
- When the tool is to be regarded as activated (switched on) for the processing

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.

In this connection, wild cards are admissible:

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

Configuration

Example: At one machine, there are two revolvers the tool nodes of which are "Rev101" to "Rev112" at the first revolver and "Rev201" to "Rev212" at the second revolver. In order to assign all tool nodes on revolver 2 to one tool element, the following attribute values would be applicable: "Rev2*", "Rev2???", "*2???", "????2*". "*2*", however, is wrong as it also includes "Rev112".

- **Channel:**
Specifies the number of the channel that is responsible for the tool change at this/these tool node(s)
- **Function:**
Specifies the function of the tool node. The following values are possible:
 - "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 is not able to generate a tool path graphic
- **ToolpathPartnerChannels:**
By default, only the tool path of the channel active at the interface. 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 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 for the processing is activated (switched on)
- `<Tool>`:
This program section influences 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

Configuration

Variable name	Type	Value range	Pre-assignment	Description
CUSTOMDATA\$	String		Value of the Custom-Data attribute of the tool node	Input variable
SD	Structure			Input variable with read access

Tab. 10-7: 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 will not be fitted outside the program processing.

The syntax of the query string is described in the documentation "IndraMotion MTX 14VRS commissioning", chap. "Control commissioning", "definition of the list content".

Example:

At a revolver machine, the database query is to be provided using the name of the tool node. The name comprises "Rev" followed by the place number. The sector is constant and 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 access.

Tab. 10-8: 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 processing), 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 in this place.

Configuration

Example:

At a machine, the activation state of a laser can be seen from bit 1 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>
    <Tool/>
  </ToolNode>
</ToolConnection>
```

Tag <Tool>**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 arbor part. If string is empty, no arbor part. 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

Configuration

Variable name	Type	Value range		Pre-assignment	Description
EDREF%	Integer	0..16		0 in case of tool description, otherwise 1	Reference cutting edge number relatively to which the tool is calculated. 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. If string is empty, no holder. 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 in case of tool description
L2	real			Correction amount L2 in case of tool description, otherwise 0	Input/output variable for the correction L2 in case of tool description
L3	real			Correction amount L3 in case of tool description, otherwise 0	Input/output variable for the correction L3 in case of tool description
NAME\$	String			Name of the tool node in the model	Input variable
ORI%	Integer	0..9		Cutting edge orientation in case of tool description, otherwise 0	Input/output variable for cutting edge orientation in case of tool description
ORIPLANE\$	String	"G17", "G18", "G19"		after DBT1.Hd.ToolOri. Otherwise "G18" with grinding and turning tools, "G17" for the rest	Input and output variable for the tool orientation
SD	Structure				Input variable with read 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 access on the raw part dimensions
SUSPEND?	Boolean	TRUE	no graphic tool change	FALSE	Output variable to the control whether the tool change is to be simulated
		FALSE	graphic tool change		

Configuration

Variable name	Type	Value range	Pre-assignment	Description
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 lengths L1..L3 to the spatial axes X,Y,Z analogously to G78
RAD	real	=>0	Radius correction amount in case of tool description, otherwise 0	Input/output variable for cutting edge radius in case of tool description

(1) CUTTHEIGHT with grinding tool: Length of the replacement presentation in the workpiece simulation as profile bar.

Tab. 10-9: Input/output variables of the "Tool" program section

DESCR variable with 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. 10-10: Elements of the DESCR input variables with drilling tools

DESCR variable with 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

Configuration

Index (type name)	Element name	Type	Value range	Description
--	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
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. 10-11: Elements of the DESCR input variables with milling tools

DESCR variable with 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 plate, 2 inclined plates, 3 face plates 4 internal plates	Wheel type
P1	P	Integer	1..402 with interruptions	Profile number
P2	D1	real	<>0	Outer diameter
P3	D2	real	>0	Inside 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

Configuration

Index	Element name	Type	Value range	Description
P9	CA1	real	$0 \leq CA1 < 90$	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	$0 \leq A1 < 90$	Back-slope angle left
P13	H2	real	≥ 0	Shoulder height right, face width at TY3
P14	R2	real	≥ 0	Radius right
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. 10-12: Elements of the DESCR input variables with grinding tools

The general operating principle of this program section is described in [chapter 10.2 "Introduction to the tool graphic generation" on page 84](#).



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 arbor parts. So an assignment of the holder variable which is valid during the run-time is one of the important activities in the simulation application.

*Example:***Support of generic graphics and imports**

At a milling machining center, the "description" element in the DBT1.UD.TI.Descr tool data set is to be used in order to be able 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 will be interpreted as holder including arbor. Cutting edge and shaft are generically created from the technology data
2. In all other cases ending with ".stl", it is assumed that the model comprises the entire tool
3. If the description starts with "REV" (followed by "," or ":"), the contour is to be used for the rotary generation of the cutting edge part. It is left to the system to amend the arbor section
4. In all other cases it is assumed that the description contains the parameters for the generation of the arbor section
5. For the holder, a shell contour analogously to HSK 63 is to be agreed

Configuration



In case 2, the CUTT\$ variable is to be assigned. The system will separate the cutting edge part from the mesh using the useful length (technology data) and use the remaining part as shaft geometry.

The simulation will only amend arbor and/or holder if the lengths of cutting edge and shaft are not sufficient in order to fill the length correction amount and if this gap is at least as high as the relevant geometry. Against this background, cases 2 and 3 are to be treated identically.

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 that are to be taken from the "DBT1.UD.PD.Spring_Mode" string variable. Suitable for every size, the suitable STL file for upper and lower part is in each case to be transmitted for the holder and the relevant value is to be assigned 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
```

Configuration

```

        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, the tool orientation is to be changed from G18 to G17 for drilling tools that have been defined by means of a tool description. In this way, the interpolation level G18 can be used for centric drilling but nevertheless, you still get the correct graphical orientation. As centric drilling tools, the "DRILL" and "CDRILL" types are to come into question.



As with tool descriptions, the `TOFFL$` string variable should not get the "ActPlane" content, the change in the orientation level will never influence 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** forms one part of the cutting edge part thickness of punching tools by multiplying this factor by the sheet thickness.

Value range: ≥ 0 . Default is 1.25

Constant cutting edge height

Tag <PunchToolLenConst>:

The **constant cutting edge height** in mm forms the second part of the cutting edge part thickness of punching tools.

Value range: ≥ 0 . Default 20 mm

With punching tools, the `CUTTHEIGHT` 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 acts as displacement of the line draft over the sheet surface and serve the better graphical separation.

Value range: ≥ 0 . Default 0.03 mm

Generative manufacturing

Activate size 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: `False` - no extension (default), `True` - additions active

Size addition at top

Tag <BoundingBoxExtendTop>:

Extension of the space for the material application in Z-positive direction in mm.

Value range: ≥ 0 , default 100 mm

Size addition at bottom

Tag <BoundingBoxExtendBottom>:

Extension of the space for the material application in Z-negative direction in mm.

Value range: ≥ 0 , default 0

Size addition on the left

Tag <BoundingBoxExtendLeft>:

Extension of the space for the material application in X-negative direction in mm.

Value range: ≥ 0 , default 0

Size addition on the right

Tag <BoundingBoxExtendRight>:

Extension of the space for the material application in X-positive direction in mm.

Value range: ≥ 0 , default 0

Size addition at front

Tag <BoundingBoxExtendFront>:

Extension of the space for the material application in Y-positive direction in mm.

Value range: ≥ 0 , default 0

Size addition at rear

Tag <BoundingBoxExtendRear>:

Extension of the 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 no generally available and unsalable feature. It requires a special DLL. In the officially marketed versions, the subsequently listed settings are thus ineffective.

Availability of the augmented reality

Tag <Enabled>:

Specifies whether the view for the augmented reality is applied.

Value range: `False` - no (default), `True` - yes

Transparency with augmented reality

Tag <Transparency>:

Transparency of the simulation graphics in percent that are shown in the camera image.

Value range: 0..100%, default 50%

Background

Tag <Background>:

Configuration

Addresses the camera or specifies a locally stored image and/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 in the image height direction
- UpY - Y component of the unit vector pointing in the image height direction
- UpZ - Z component of the unit vector pointing in the image height direction

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

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

Horizontal shearing

Tag `<XShearing>`:

Factor for the horizontal image shearing.

Value range: $-1..1$, default 0

Vertical shearing

Tag `<YShearing>`:

Factor for the vertical image shearing.

Value range: $-1..1$, default 0



Set-up of the camera also includes the provisioning of a camera calibration file. The latter is to be stored under the name "camera_calibration_file.xml" in the folder containing the machine model.

10.3.3 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 in an online simulation, individual axis coordinate value pairs can be excluded from the simulation in order to keep the extended simulation running to below one second. Higher computer utilization may result in a simulation tool path considerably deviating from reality. Value range: True - minimized overtravelling, 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
Number of room pixels	Tag <ModelSamplesPerDiagonal>: Specifies the number of room pixels per diagonal of the cuboid enveloping the raw part. This setting decisively influences the graphical quality, the memory utilization and the performance of the simulation. Value range: 5..1500, default 400
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 for 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 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.
Facet tolerance	Tag <FacetTolerance>: Converting a bent area into a mesh necessarily results in a deviation between ideal geometry and model geometry. The setting specifies the maximum deviation in mm. Value range: >0, default 0.05 mm
Facet tolerance in 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

Configuration

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, set-related pieces of information	Tag <ToolpathMaximumMarkerNumber>: For every section of the tool path, you can save information on the NC set. The setting specifies the maximum number of stored pieces of set information. When the maximum number is exceeded, the oldest pieces of information will be lost. Value range: > 0, default 100000
Show edges everywhere?	Tag <ShowEdgesEverywhere>: The presentation of body edges compromises the performance. The setting determines whether with activated edge presentation (menu function) the edges are shown at all geometry elements or whether the edge presentation is to be 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 for removing separated volumes	Tag <ChunkDetectionInterval>: Specifies the related time interval for <ChunkDetection>Timer</> in milliseconds. Value range: > 0, default 2,500 ms
Renew workpiece at program start?	Tag <RenewStockAtProgStart>: Specifies whether upon program start, workpieces processed and/or created by the channel concerned are to be exchanged against unprocessed 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 to be saved upon termination of IndraWorks Operation in order to be able to continue with the latest processing state upon restart of the interface. With complex workpieces, saving will delay the termination of the interface. Value range: <code>True</code> - save (default), <code>False</code> - unprocessed workpieces after start of the interface
Several workpieces per workpiece holder	Tag <AllowMultipleStocks>: This setting defines whether with dialog-guided definition or import of a workpiece, this workpiece is to be replaced by the available one or whether it is to

be added. This setting does not refer to the import or the definition via cycles and does not refer to other geometry types.



Currently, the clamping of several pieces is only possible with workpieces as enveloping bodies (wire model).

Value range: `False` - Replace workpiece (default), `True` - Add workpiece

Collision monitoring

Tag `<CollisionDetection>`:

Specifies whether the simulation is to indicate collisions by coloring the collision partners in red and issuing a message in the status bar.

Value range: `True` - Monitoring on (default), `False` - No monitoring

View change with full-screen presentation

Tag `<FitViewWhileChanging>`:

Activation of this setting will provide for the full-screen presentation of the graphic after every view change using a menu function.

Value range: `True` - full-screen (default), `False` - enlargement is maintained

Real-time OPC UA item

Tag `<NC.DataRT>`:

Determines if the newer item is to be used with more information for data transfer from the NC kernel to the simulation.

Value range: `True` - yes (default), `False` - no (lower quality)

Problem analysis (debug mode)

Tag `<Debug>`:

This setting activates different functions for the problem analysis:

- Context menu for creating simulation records
- Display of stack size, assumed item responses per second and simulated item responses per second
- In the dialogs, you can import and export both, STL and binary files.
- STL models of the tools

Value range: `False` - no analysis function (default), `True` - analysis function active.

Settings for display

Initial setting axes/scales

Tag `<CoordSystemVisible>`:

Represents the initial setting regarding the visibility of a coordinate system. The display is switched by means of 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

Main view

Tag `<MainView>`:

The "main view" setting is currently not used.

Value range:

- `Iso` - Non-orthogonal view
- `Front` - Front view
- `Rear` - Rear view
- `Left` - View from the left
- `Right` - View from the right

Configuration

	• Top - Top view • Bottom - Bottom view
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 selection of a program in the navigator and in the program selection. • 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
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 realized yet and has the same effect as tracking.
Improve workpiece	Tag <HighQualityZooming>: Provides one possibility for the subsequent improvement of the workpiece graphic. Value range: • None - The "Improve workpiece" function is not available • DynamicZoom - When zooming, the simulation of the previous processing for the shown section is internally repeated, 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; it will, however, no longer compromise the memory and the performance • AutoRefine - Accompanying the simulation and depending on the utilization, 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
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 means of superimposition with a bitmap file whereas reflectance greater than zero is assumed. An activation reduces the color contrast and may be unfavorable under certain circumstances. Value range: <code>True</code> - simulate color gradient due to reflexion of the ambient light (default), <code>False</code> - uniform color gradient on a level surface

Configuration

Reflectance	Tag <StockReflectivity>: Defines the share of the light (shine) reflected by the workpiece surface in percent. Value range: 0..100, default 25%
Workpiece transparency	Tag <StockTransparenzy>: Defines the workpiece transparency in percent. The setting is not effected with Augmented Reality and not in case of simultaneous presentation of the finished part geometry and/or command/actual comparison. Value range: 0..100, default 25%
Transparency of the workpiece with internal finished part	Tag <StockTransparenzyShowingTarget>: Defines the workpiece transparency in percent if at the same time, the finished part geometry is visible. Value range: 0..100, default 30%
Transparency of the body edges	Tag <EdgeTransparenzy>: Defines the transparency of the lines for body edges in percent. By means of the setting, too "dominant" body edges can be avoided how they can be observed with some graphic cards. Value range: 0..99, default 0%
Colors	
Syntax for colors	Colors are defined via their red, green and blue shares without alpha value (transparency) whereas 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). Every color component is to be specified as two-digit figure and with a leading zero, if applicable. 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 has been activated. Default #8C8C00
Color of the tool shaft	Tag <Shaft>: Specifies the color of the shaft of tools. Default #A9A9A9
Color of the tool arbor	Tag <Arbor>: Specifies the color of the arbor part of the tools. Default #D3D3D3
Color of the tool holder	Tag <Holder>: Specifies the color of the holder of tools.

Configuration

	Default #808080
Color for tool path rapid traverse	Tag <ToolpathRapid>: Specifies the color of the tool path in case of rapid traverse (G0) and/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 and/or forming movements during punching. Default RGB (206, 0, 24)
Color for 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 and/or mark the user interaction elements. Default #FF00FF
Raw part color	Tag <Stock>: Specifies the color for non-machined raw part surfaces except for in the command/actual comparison. Default #4682B4
Color for machined surfaces	Tag <RemovedMaterial>: Specifies the color of the machined workpiece surfaces, except for in the command/actual comparison. Default #FFFFFF0
Finished part color	Tag <TargetGeometry>: Specifies the color of the finished part. Default #FFE4B5
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 sectional areas	Tag <SectionFace>: Specifies the color of sectional areas (sectional view). The color is not used yet.
Color scheme for command/actual comparison	Tag <DeviationColorScheme>: For the command/actual comparison of the workpiece with the finished part geometry (deviation mode), different intervals of the dimensional deviation are in assigned one color each. The color value specifies whether the com-

Configuration

mand geometry was violated (waste), the command geometry was achieved or whether there is an allowance.

The upper limit of every interval is defined in the `deviation` attribute whereas the upper limit of the highest interval is always evaluated as plus infinite. The lowest interval starts at minus infinite, all others at the value of the `deviation` attribute of the predecessor (with the next-smaller attribute).

Syntax: `<SchemeColor Deviation="<Upper limit>"><Color></SchemeColor>` with upper limit equal to upper interval limit in mm and the color value `<Color>` in decimal or hexadecimal presentation.

The intervals don't have to be specified ordered by size. A valid color scheme comprises at least three intervals.

Default:

Program:

```
<DeviationColorScheme>
  <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 may be activated that are used for both, the cutting edge and for the machined areas. The colors at the workpiece will then allow for conclusions regarding the tool used. To activate the color change, set the attribute `UseCutterColors="True"`.

For a valid color scheme, at least three colors are required. 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>
```

11 Integrating the MTX simulation control into third-party applications

11.1 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 "OPD2" or "Workstation2" of version MTX 14V20 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. Unpack the archive "IndraWorks\CommonAppDataAllInst\Simulation\MTX.Simulation.SampleAppProject.zip" in the same 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\CommonAppDataAllInst\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

11.2 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.

Integrating the MTX simulation control into third-party applications

It is, however, still possible to instantiate several instances of the `Rexroth.MTX.SimulationACICtrl.SimulationACICtrl` 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 `SimulationACICtrl.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 SimulationACICtrl.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 `SimulationACICtrl.FunctionState` class.
 - The most important states are `FunctionState.ON`, `FunctionState.OFF` and `FunctionState.DISABLED` – Function active/pressed, function inactive/off and function not available

Integrating the MTX simulation control into third-party applications

- 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 short-cuts

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 `IA-CI.Active=true`.

11.3 Sample project

Screenshot

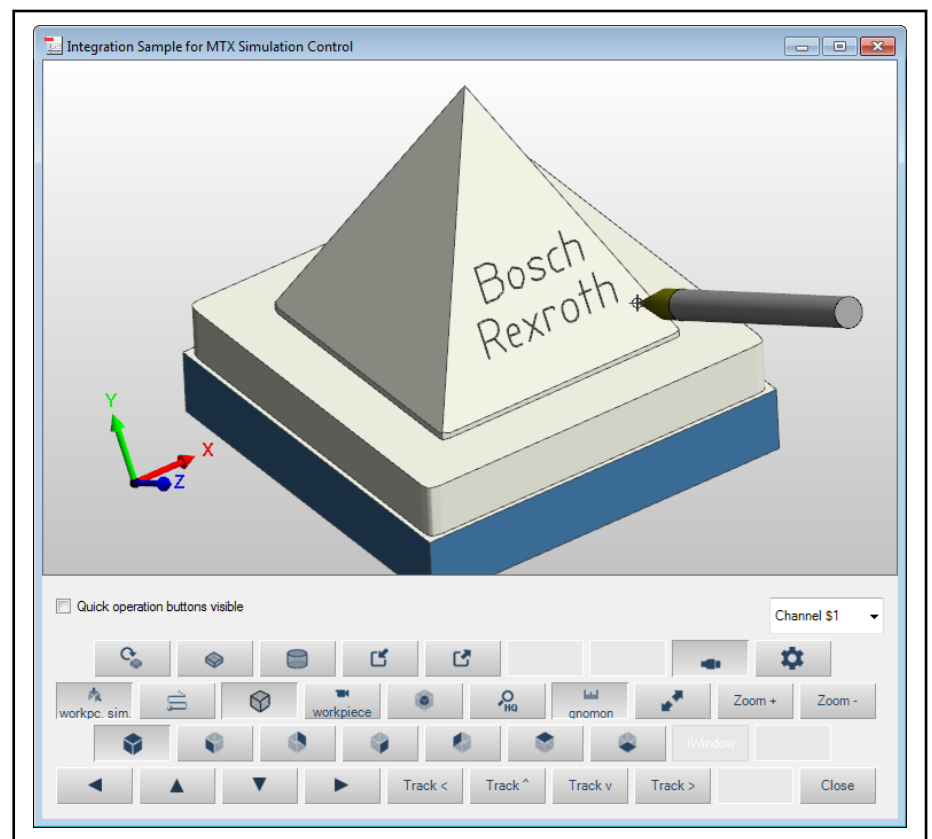


Fig. 11-1: User interface of the sample application for integrating the MTX simulation control

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

Integrating the MTX simulation control into third-party applications

- 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)
```

Integrating the MTX simulation control into third-party applications

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)
```

12 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.....	3
Information representation.....	8
Validity of the documentation.....	3
ANSI Z535.6-2006.....	8

B

Bar section (turning).....	43
Boring bar, backward counterbore.....	59

C

Complaints.....	10
Criticism.....	10
Cuboid raw part (milling).....	39
Customer Feedback.....	10
Cylindrical raw part (milling).....	41
Cylindrical, angular and ball cutter.....	61

D

Documentation:	
Record of revision.....	3
Drill, countersink, reamer.....	56

E

Engraving milling cutter, chamfer milling cutter, face milling head.....	64
--	----

F

Feedback.....	10
---------------	----

G

Grinding wheel.....	69
---------------------	----

H

Hazard warnings.....	8
Helpdesk.....	127
Hotline.....	127

I

Import model.....	48
Information representation	
Names and Abbreviations.....	10
Intended use	
Areas of Application.....	11
Introduction.....	11
Use cases.....	11

N

NC simulation	
Auto refine.....	23, 116
Basic license.....	13
Command-actual comparison.....	20
Command/actual comparison.....	118
Configuration.....	32, 81

Cylindrical raw part.....	29
Dynamic zoom.....	23, 116
Edges.....	25, 114, 117, 118
Export.....	32
Finished part.....	20, 25, 117
Hiding machine parts.....	24
Import.....	30
Improve workpiece.....	23, 116
Integration into third-party applications.....	121
Layout.....	19
Operation.....	21
Options.....	32, 81
Player.....	37
Problem analysis (debug mode).....	37, 115
Raw part cuboid/sheet.....	27
Recording and playing recordings.....	37
Reset raw part.....	26, 35, 114
Shortcuts.....	35
Simulation virtual machine.....	13
Tool path.....	13, 24, 84, 110, 114, 116, 118
transparent enveloping body as raw part.....	28
Workpiece look-ahead.....	20
Workpiece preview.....	116, 118

P

Polygon (turning).....	45
------------------------	----

Q

Quarter round profile milling cutter.....	67
---	----

R

Remove workpieces/models.....	51
Renew workpiece.....	53
Replace/add workpieces/models.....	55

S

Safety instructions.....	8
Select workpiece holder.....	54
Service hotline.....	127
Sheet (punching).....	47
Signal alert symbol.....	8
Signal words.....	8
Step drill, centering drill.....	57
STOCK/BAR.....	43
STOCK/BLOCK.....	39
STOCK/CYLINDER.....	41
STOCK/POLYGON.....	45
STOCK/REMOVE.....	51
STOCK/RENEW.....	53
STOCK/SHEET.....	47
STOCKADD.....	55
STOCKIMP.....	48
STOCKSEL.....	54
Suggestions.....	10
Support.....	127

Index

Symbols..... 10

T

Target groups..... 3
TOOL/BBAR..... 59
TOOL/CDRILL..... 57
TOOL/CHAMFER..... 64
TOOL/CRMILL..... 67
TOOL/DRILL..... 56
TOOL/GRIND..... 69
TOOL/MILL..... 61

U

Unintended use..... 12
Unintended Use
 Consequences, non-liability..... 11

W

Warnings..... 8

Notes

Bosch Rexroth AG

Electric Drives and Controls

P.O. Box 13 57

97803 Lohr, Germany

Bgm.-Dr.-Nebel-Str. 2

97816 Lohr, Germany

Phone +49 9352 18 0

Fax +49 9352 18 8400

www.boschrexroth.com/electrics



R911393273