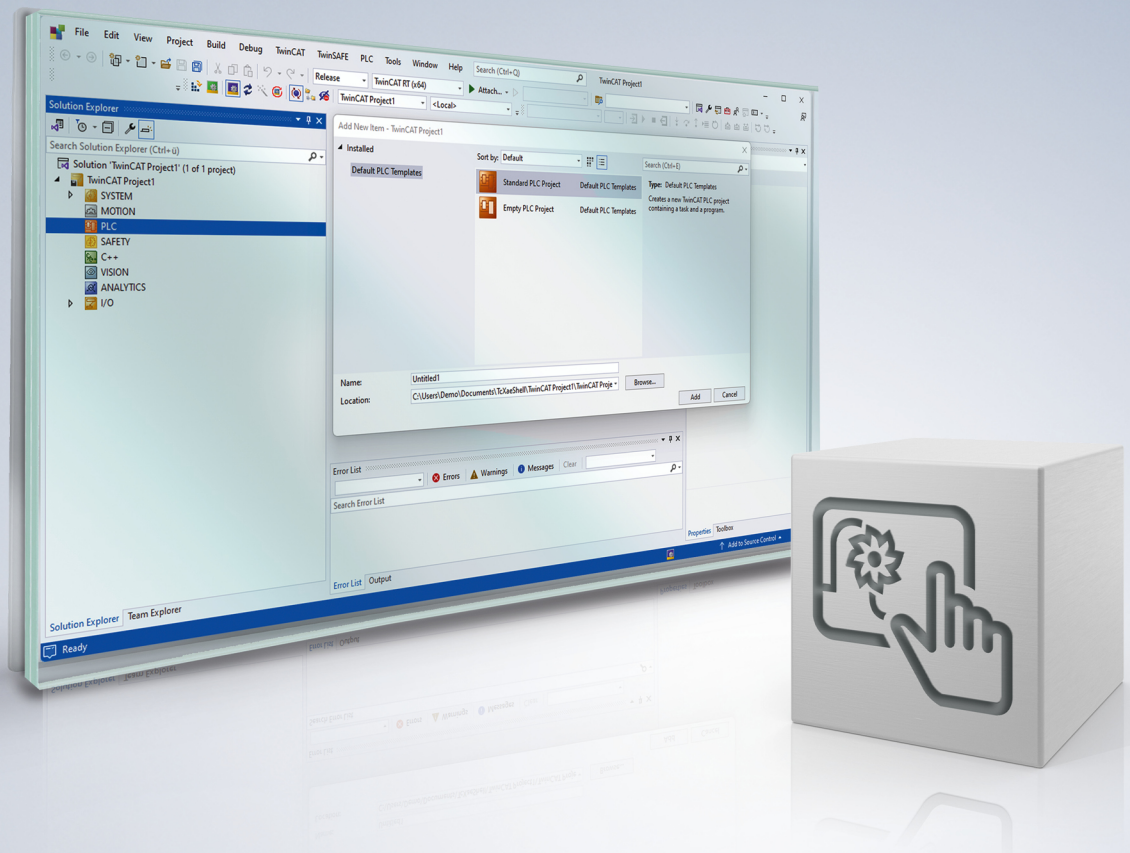
BECKHOFF New Automation Technology

Manual | EN

TF5310

TwinCAT 3 | CNC HMI Base



2026-11-23 | Version: 1.0.0

Table of contents

1 Foreword 5
1.1 Notes on the documentation 5
1.2 For your safety 6
1.3 Notes on information security 7
1.4 Documentation issue status 8
2 Product overview 9
3 Installation 10
3.1 System requirements 10
3.2 Installation 10
3.2.1 Windows 10
3.3 Licensing 13
4 Product components 15
4.1 CNC controls 15
4.2 Server extension 15
4.3 CNC HMI Template 16
4.4 Styling and themes 25
4.5 User management 29
4.6 Internationalization 32
4.7 Virtual directories 32
5 Control elements 33
5.1 TcHmiCncAdsState 33
5.2 TcHmiCncAxisList 36
5.3 TcHmiCncButton 40
5.4 TcHmiCncChannelStatus 42
5.5 TcHmiCncCombobox 44
5.6 TcHmiCncConfigurableButton 47
5.7 TcHmiCncContourWorkbench 60
5.8 TcHmiCncDetachableTabNavigation 69
5.9 TcHmiCncDiagData 72
5.10 TcHmiCncFileExplorer 73
5.11 TcHmiCncIpcDiagnosis 77
5.12 TcHmiCncLayoutManager 79
5.13 TcHmiCncLinearOverride 86
5.14 TcHmiCncMessageLine 93
5.15 TcHmiCncMessageList 95
5.16 TcHmiCncPriorityGrid 97
5.17 TcHmiCncScaleRegions 99
5.18 TcHmiCncSettings 101
5.19 TcHmiCncSimulation 107
5.20 TcHmiCncTechnoData 109
5.21 TcHmiCncTextEditor 111
5.22 TcHmiCncToolList 126
5.23 TcHmiCncTreeview 130

5.24 TcHmiCncVariableView 132
5.25 TcHmiCncZeroOffsets 134

6 Appendix 137
6.1 Third-party components 137
6.2 Support and Service 138

1 Foreword

1.1 Notes on the documentation

This description is intended exclusively for trained specialists in control and automation technology who are familiar with the applicable national standards.
The documentation and the following notes and explanations must be complied with when installing and commissioning the components.
The trained specialists must always use the current valid documentation.

The trained specialists must ensure that the application and use of the products described is in line with all safety requirements, including all relevant laws, regulations, guidelines, and standards.

Disclaimer

The documentation has been compiled with care. The products described are, however, constantly under development.
We reserve the right to revise and change the documentation at any time and without notice.
Claims to modify products that have already been supplied may not be made on the basis of the data, diagrams, and descriptions in this documentation.

Trademarks

Beckhoff®, ATRO®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, MX-System®, Safety over EtherCAT®, TC/BSD®, TwinCAT®, TwinCAT/BSD®, TwinSAFE®, XFC®, XPlanar® and XTS® are registered and licensed trademarks of Beckhoff Automation GmbH.
If third parties make use of the designations or trademarks contained in this publication for their own purposes, this could infringe upon the rights of the owners of the said designations.





EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.
The distribution and reproduction of this document, as well as the use and communication of its contents without express authorization, are prohibited.
Offenders will be held liable for the payment of damages. All rights reserved in the event that a patent, utility model, or design are registered.

Third-party trademarks

Trademarks of third parties may be used in this documentation. You can find the trademark notices here: https://www.beckhoff.com/trademarks.

1.2 For your safety

Safety regulations

Read the following explanations for your safety.
Always observe and follow product-specific safety instructions, which you may find at the appropriate places in this document.

Exclusion of liability

All the components are supplied in particular hardware and software configurations which are appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

Personnel qualification

This description is only intended for trained specialists in control, automation, and drive technology who are familiar with the applicable national standards.

Signal words

The signal words used in the documentation are classified below. In order to prevent injury and damage to persons and property, read and follow the safety and warning notices.

Personal injury warnings

⚠ DANGER
Hazard with high risk of death or serious injury.

⚠ WARNING
Hazard with medium risk of death or serious injury.

⚠ CAUTION
There is a low-risk hazard that could result in medium or minor injury.

Warning of damage to property or environment

NOTICE
The environment, equipment, or data may be damaged.

Information on handling the product



This information includes, for example:
recommendations for action, assistance or further information on the product.

1.3 Notes on information security

The products of Beckhoff Automation GmbH & Co. KG (Beckhoff), insofar as they can be accessed online, are equipped with security functions that support the secure operation of plants, systems, machines and networks. Despite the security functions, the creation, implementation and constant updating of a holistic security concept for the operation are necessary to protect the respective plant, system, machine and networks against cyber threats. The products sold by Beckhoff are only part of the overall security concept. The customer is responsible for preventing unauthorized access by third parties to its equipment, systems, machines and networks. The latter should be connected to the corporate network or the Internet only if appropriate protective measures have been set up.

In addition, the recommendations from Beckhoff regarding appropriate protective measures should be observed. Further information regarding information security and industrial security can be found in our https://www.beckhoff.com/secguide.

Beckhoff products and solutions undergo continuous further development. This also applies to security functions. In light of this continuous further development, Beckhoff expressly recommends that the products are kept up to date at all times and that updates are installed for the products once they have been made available. Using outdated or unsupported product versions can increase the risk of cyber threats.

To stay informed about information security for Beckhoff products, subscribe to the RSS feed at https://www.beckhoff.com/secinfo.

1.4 Documentation issue status

Version	Changes
1.0.0	First release

2 Product overview

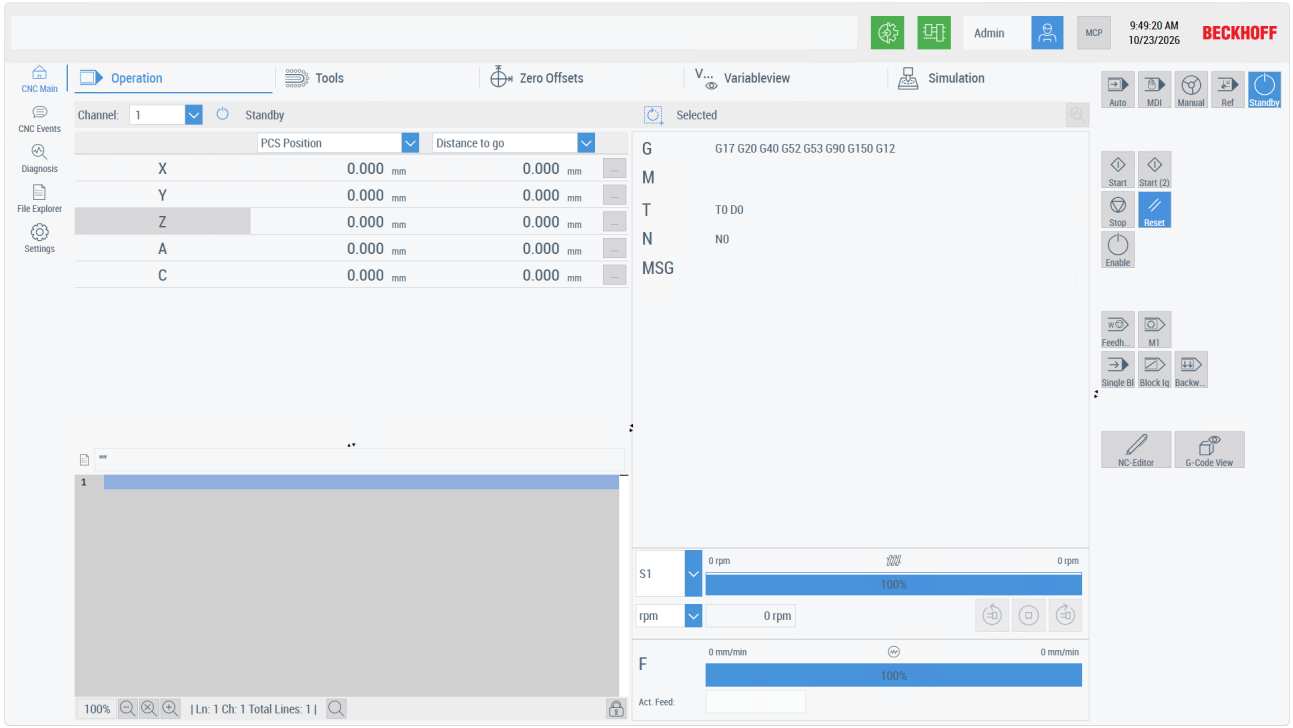
The TF5310 | TwinCAT 3 CNC HMI Base is a modular system for creating modern user interfaces for CNC machines. The modular system includes a range of CNC-specific HMI controls.

They are based entirely on the TwinCAT 3 HMI products:

- TE2000 | TwinCAT HMI Engineering
- TF2000 | TwinCAT HMI Server

The controls provided integrate seamlessly into the existing TwinCAT 3 HMI product family.

The controls can be used either as a predefined template for a CNC HMI or for developing a customer-specific HMI. The real-time connection is established via a dedicated TwinCAT HMI server extension, which aggregates data from the TwinCAT PLC and TwinCAT CNC and makes it available to the HMI client.



3 Installation

3.1 System requirements

Version 1.0.0

Technical data	TF5310 | TwinCAT 3 CNC HMI Base
Min. TwinCAT version	3.1.4026.30
Min. TwinCAT level	TF5200 | TwinCAT 3 CNC TF5210 | TwinCAT 3 CNC-E
Min. TwinCAT HMI Version	14.18
TcPkg Workload	TF5310.HMI.CNC.Base.XAE
Operating system	Windows 10, Windows 11

3.2 Installation

Depending on the operating system, there are different ways to install the TF5310 product.

3.2.1 Windows

To install the TF5310 on Windows-based operating systems, use the TwinCAT Package Manager.

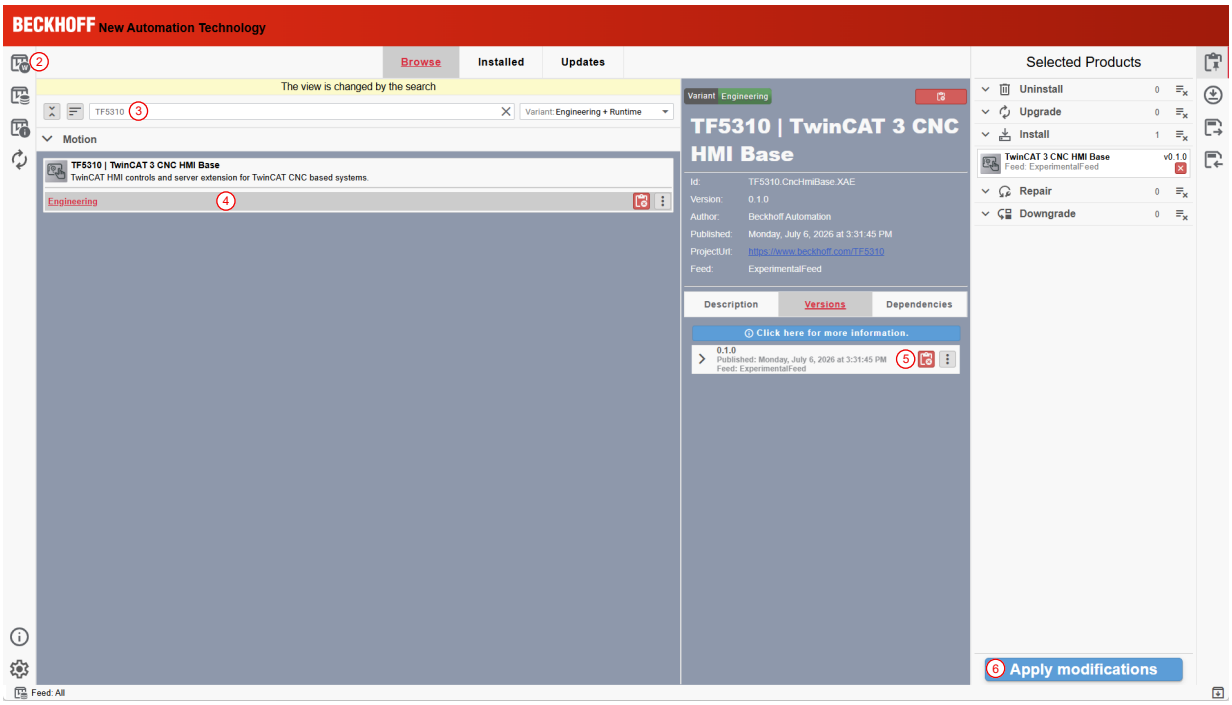
TwinCAT Package Manager (Command line)

1. Open PowerShell.
2. Enter the command:
   ```
   tcpkg install TF5310.CNC.HMI.Base.XAE
   ```
3. Please confirm any notes.

⇨ The TF5310 function has been successfully installed.

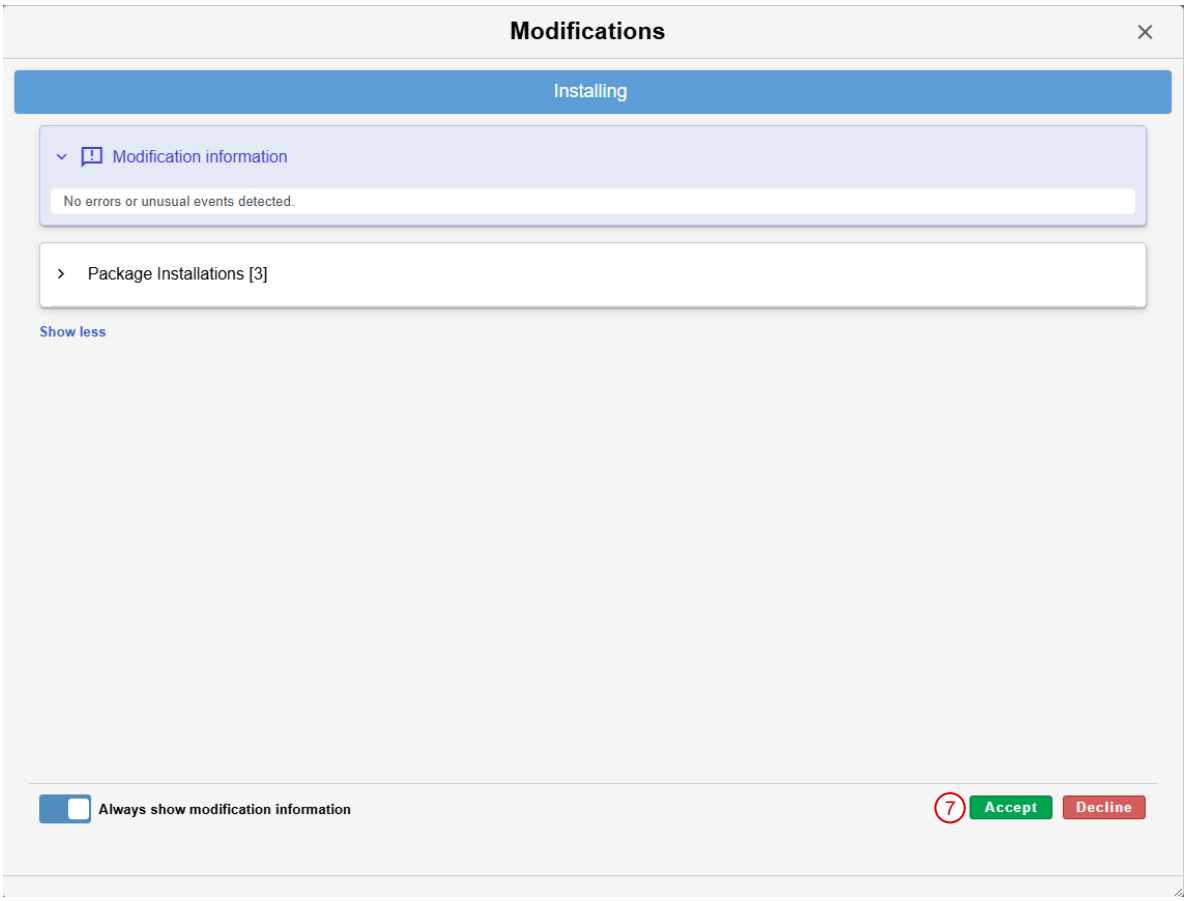
TwinCAT Package Manager (UI)

1. Open the TwinCAT Package Manager.



2. Under “Package View,” select the “Workloads” section.
3. Search for **TF5310**.
   ⇨ The workload of the TF5310 appears.
4. Select the TF5310 workload to see all available versions.
5. Select the required version.
   ⇨ The version has been marked for installation in the right-hand pane.
6. Click on **Apply modifications**.

⇨ A window will open with further installation instructions.

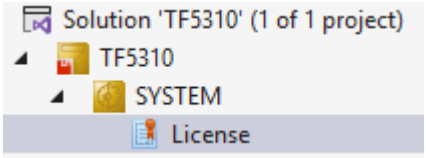


7. Click the **Accept** button to start the installation.

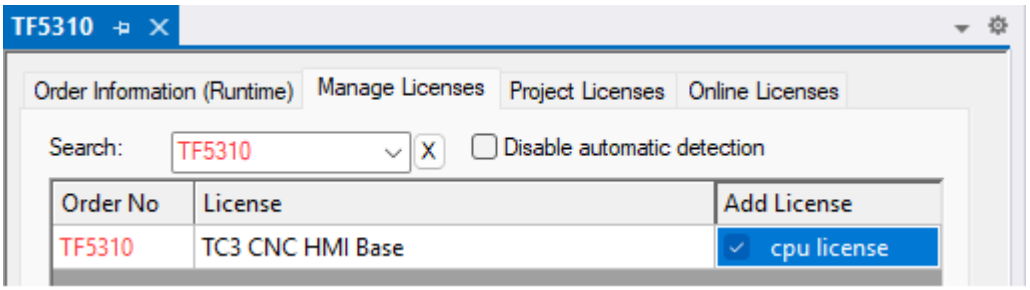
⇨ The TF5310 function has been successfully installed.

3.3 Licensing

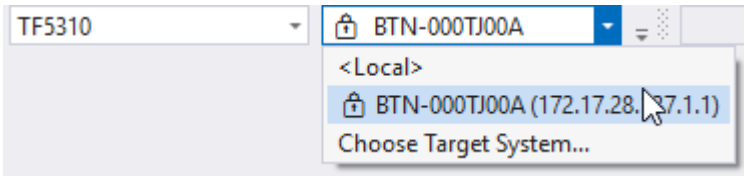
1. Start TwinCAT XAE.
2. Open an existing TwinCAT 3 project or create a new project.
3. In the Solution Explorer, navigate to the entry SYSTEM/License.



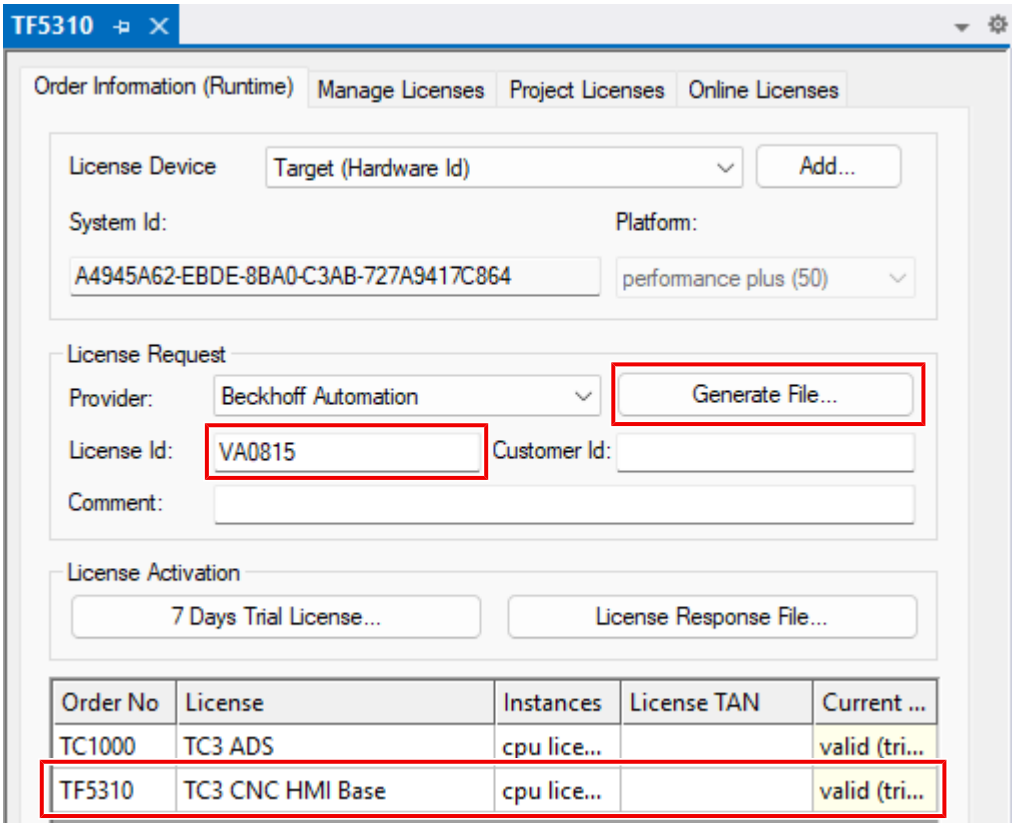
4. Open the **Manage Licenses** tab and add the following licenses to your project:



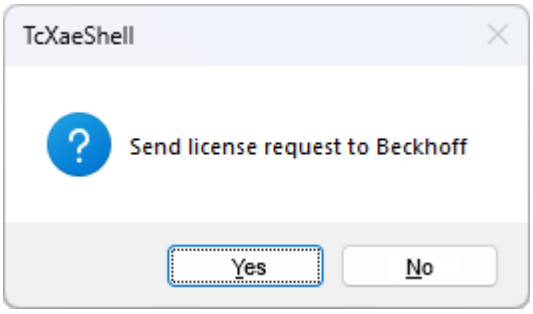
5. Optional: If you wish to add the license for a remote device, you must first connect to this device.



6. Open the **Order Information (Runtime)** tab.
7. Enter the License ID from your order, and then click **Generate File**. If you do not know the license ID, please contact your representative at Beckhoff sales.



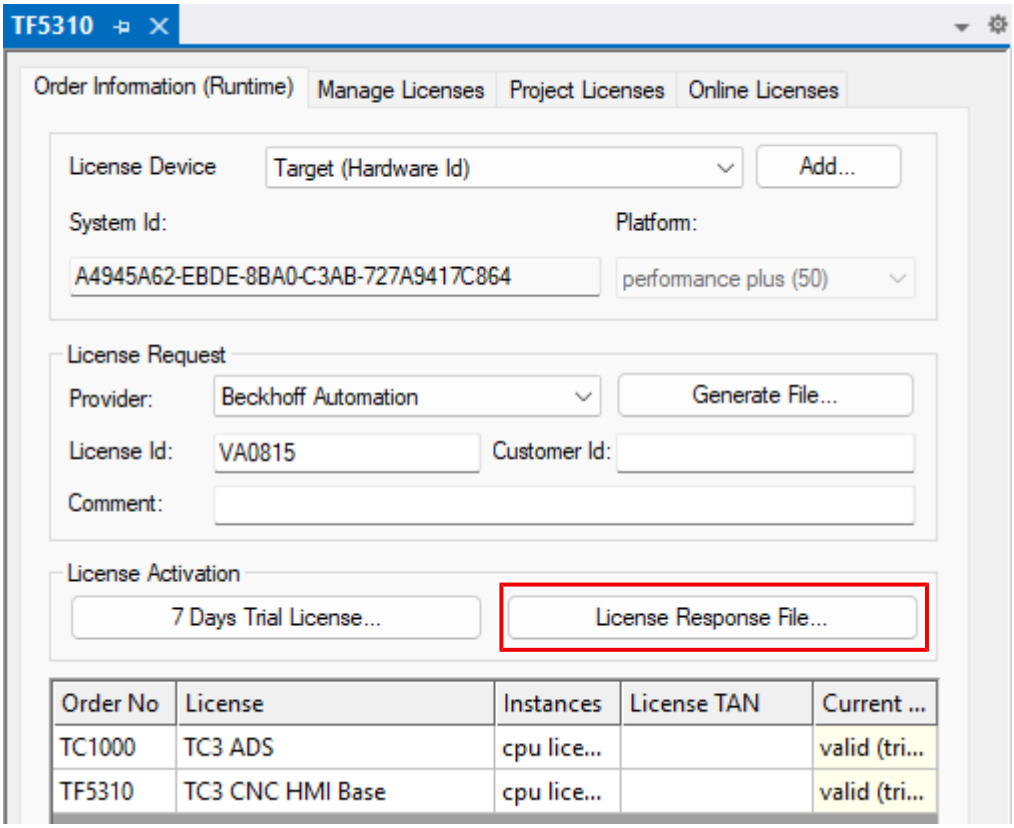
8. Save the License Request File. The system then asks whether the file should be sent by email to the Beckhoff license server:



9. If you confirm the dialog with **Yes**, your default email client opens and creates a new email for tclicense@beckhoff.com containing the license request file.
10. Send the license request file to Beckhoff.

The License Response File is sent to the same email address that sent the License Request File.

11. Shortly thereafter, you will receive a license file from the Beckhoff license server. Import it using the **License Response File** button to activate the product.



12. Restart TwinCAT to activate the license.

⇨ The license file is automatically copied to your local hard disk under ...\TwinCAT\3.1\Target\License.

4 Product components

The TF5310 product includes CNC-specific controls, an associated server extension, and a predefined CNC HMI template. This chapter provides a conceptual overview of the product's components.

4.1 CNC controls

The TF5310 product includes CNC-specific controls for an HMI that provide typical functions. These include, among others:

- Displaying axis and channel data
- NC program editor
- Diagnostic functions
- Zero offset table
- Tool management
- Cycle programming

You can use the controls both as a predefined template for a CNC HMI and for developing a customer-specific HMI. A more detailed description of the controls can be found in chapter Control elements [▶ 33].

As a rule, the controls require data binding to the server extension. The Server extension [▶ 15] is part of the TF5310 product. The data bindings for the controls are already preconfigured in the template. You can control the appearance of the controls using a set of CSS variables, which are described in the chapter Styling and themes [▶ 25]. In addition, you can use a User management [▶ 29]. The appearance and behavior of the controls depend on the access rights of individual users.

File formats

The controls use the following file formats.

Control	File formats
TcHmiCncContourWorkbench [▶ 60]	.cnccontour, .dxf
TcHmiCncFileExplorer [▶ 73]	Any file formats, including .pdf
TcHmiCncSettings [▶ 101]	.cncsettings
TcHmiCncTextEditor [▶ 111]	.nc, .cyc
TcHmiCncToolList [▶ 126]	.cnctools
TcHmiCncZeroOffsets [▶ 134]	.cnczo

4.2 Server extension

The TF5310 product includes a server extension that aggregates data from the TwinCAT PLC and the TwinCAT CNC and makes it available to the controls in the HMI.

Configuration

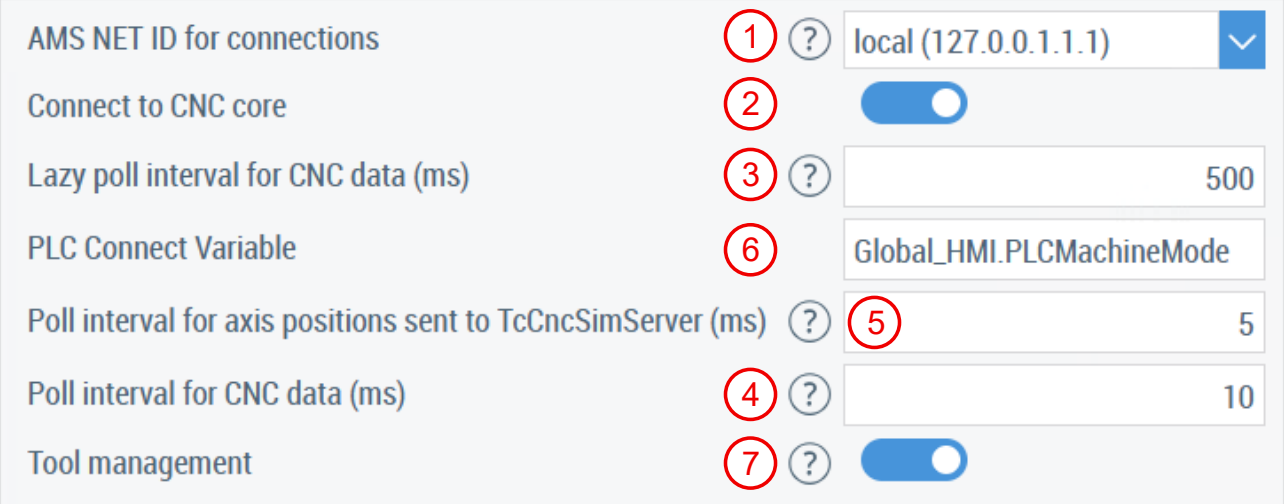
You can configure the server extension at runtime in the HMI using the control TcHmiCncSettings [▶ 101].

Identical target systems

The Server extension and the TwinCAT CNC run-time must be located on the same target system. It is not possible to use different target systems (e.g., on two different Industrial PCs).

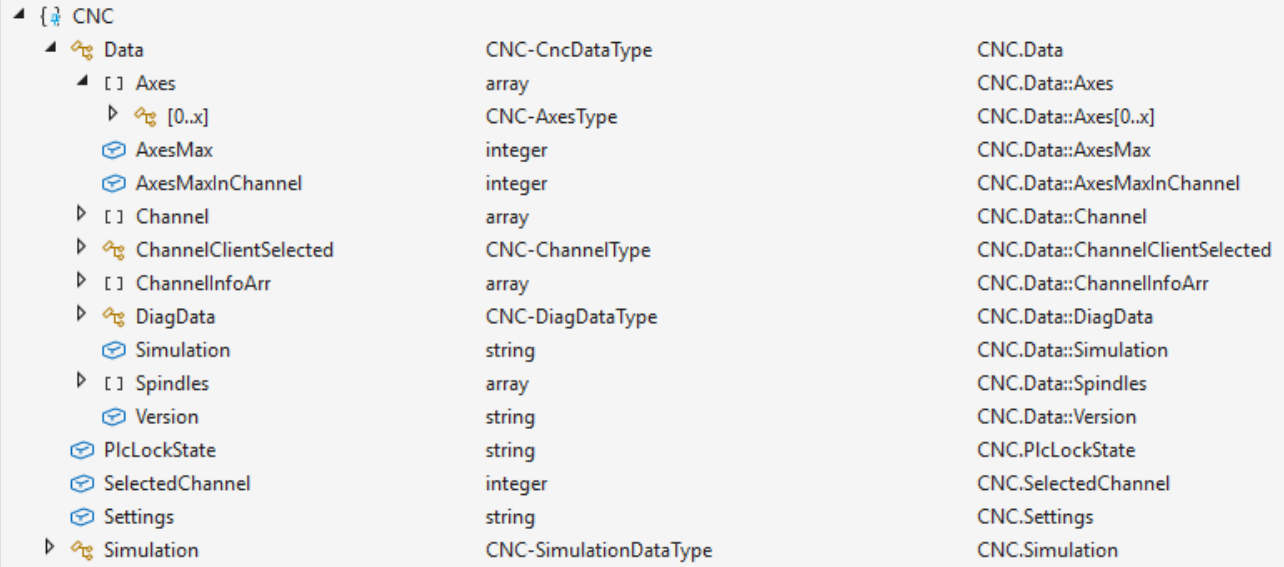
1. The server extension establishes a connection to TwinCAT CNC using the configured AMS Net ID. Connect to TwinCAT CNC either in a real-time runtime or in a user mode runtime.
2. Use the slider to set whether a connection to the TwinCAT CNC is established.

3. The server extension retrieves data from the CNC to display it in the HMI. Set the polling interval for data retrieved slowly. Data retrieved at a low frequency includes, for example, zero offset tables, tool data, and error messages.
4. Set the polling interval for data retrieved at a high frequency. Data retrieved at a high frequency includes, for example, axis and channel data.
5. If you are using the TF5320 product: Set the interval at which new axis positions are sent to the simulation server.
6. The CNC is controlled by a PLC. Define a PLC variable to which the server extension establishes a connection.
7. Use the slider to select whether you want to use tool management.



Server symbols

The server extension provides symbols for data bindings that you can use with the individual controls. The symbols required for data binding are described for each control in the chapter Control elements [▶ 33].

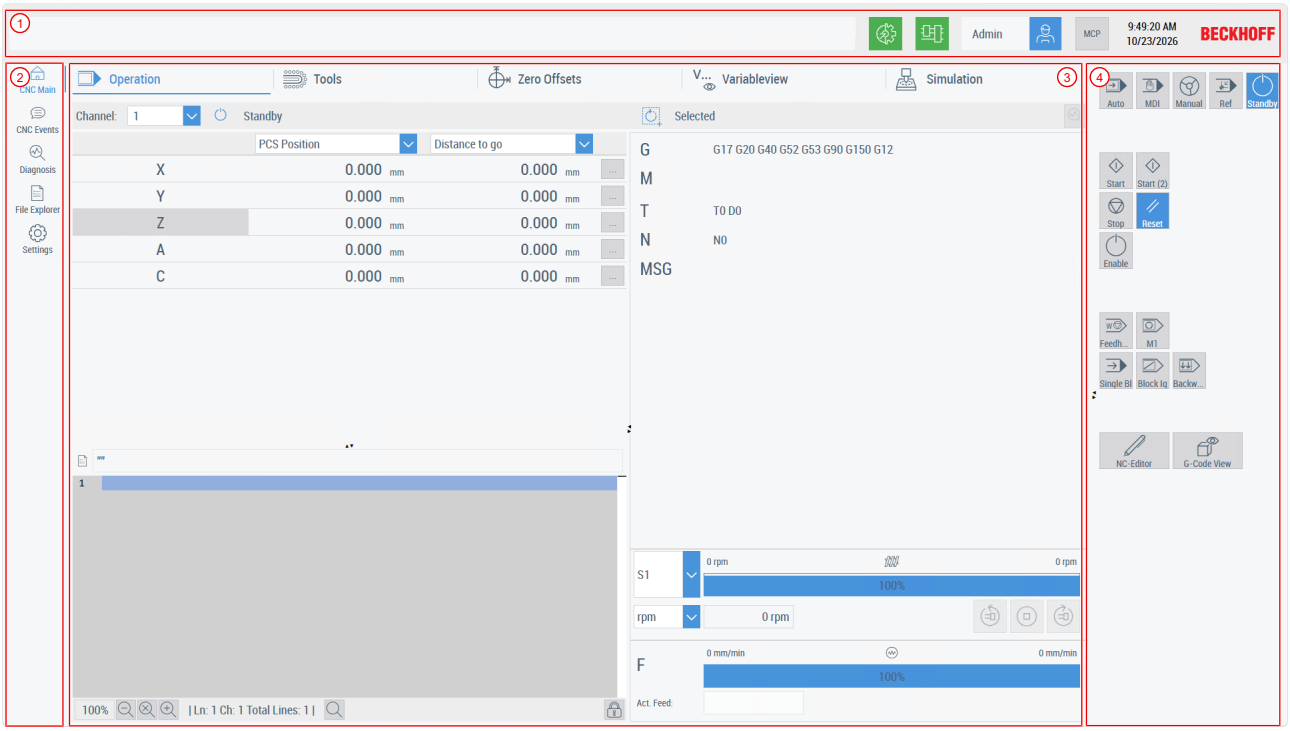


4.3 CNC HMI Template

The TF5310 product includes a predefined template for a CNC HMI, which consists of CNC-specific controls and maps out typical CNC use cases. This chapter introduces the key functions of the template. A more detailed description of the controls can be found in chapter Control elements [▶ 33]. The template is available for download on GitHub.

CNC HMI Layout

The HMI layout is divided into four sections.



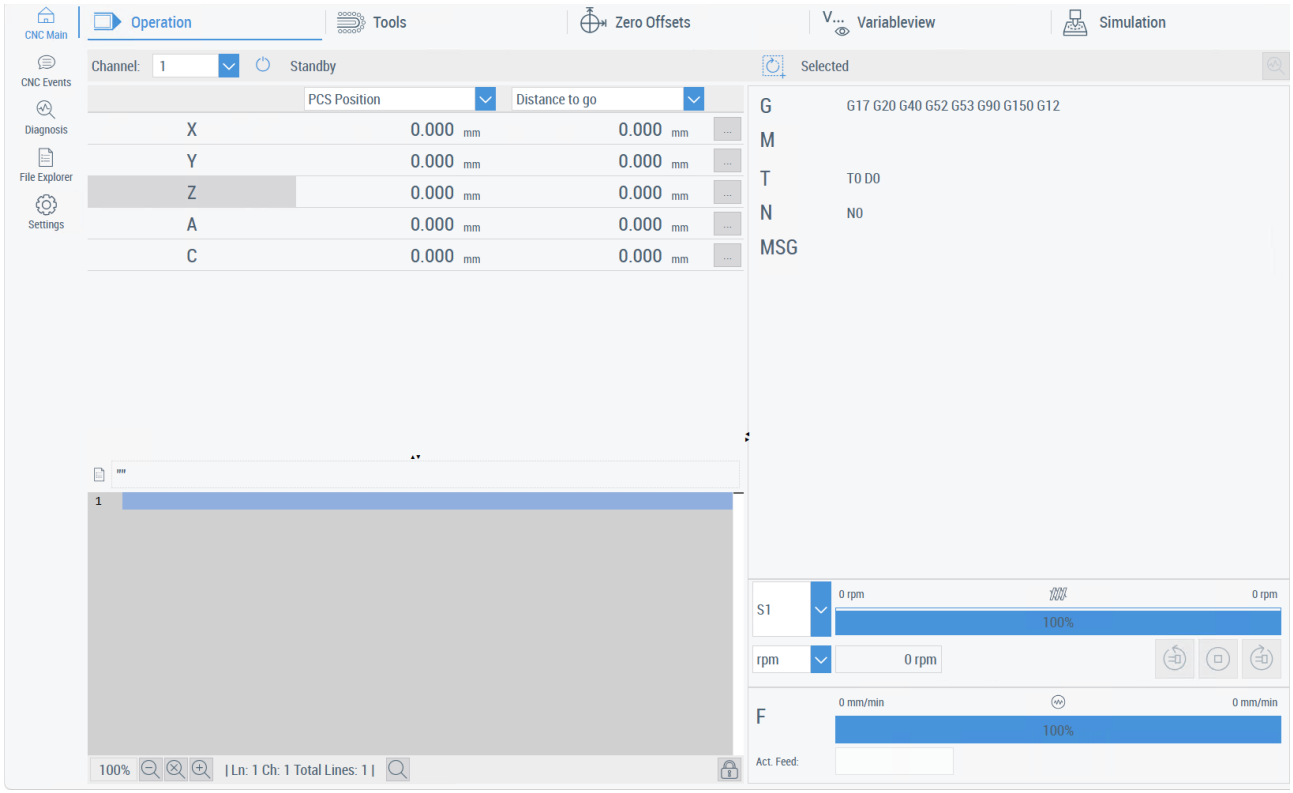
No.	Description
1	**Header:** This section displays status information as well as warning and error messages. In addition, you can switch between different users from the User management [▶ 29]. You can also customize the Machine Control Panel using the MCP button.
2	**Navigation bar:** You can use the navigation bar to switch between different main content areas.
3	**Main Content Area:** The main interaction between the user and the HMI takes place in the main content area. Using the Main Content Area, you can, for example, move axes, select an NC program, create a tool, or manage your zero offset shifts.
4	**Machine Control Panel:** The Machine Control Panel features several buttons for controlling the CNC, which you can customize and expand as needed.

Content in the Main Content Area

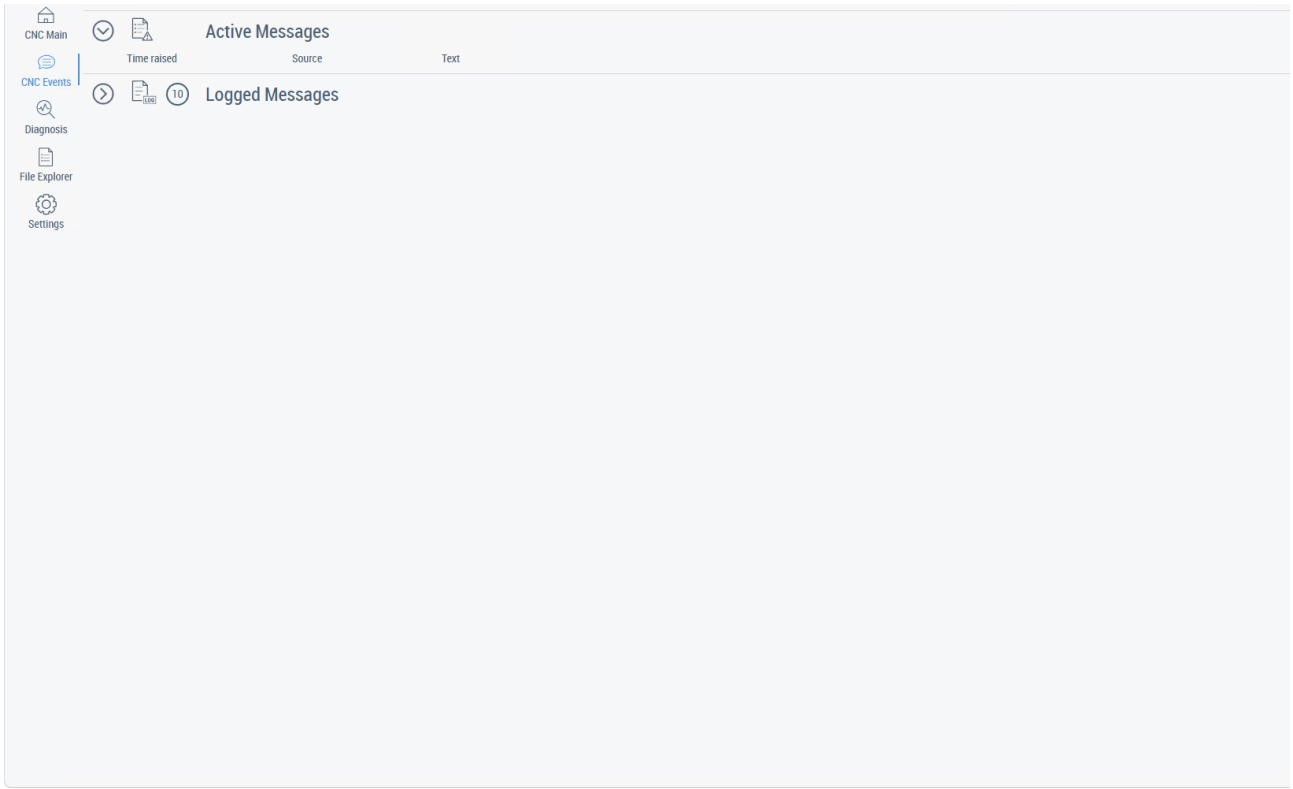
You can use the navigation bar to switch between different main content areas.

The following are available:

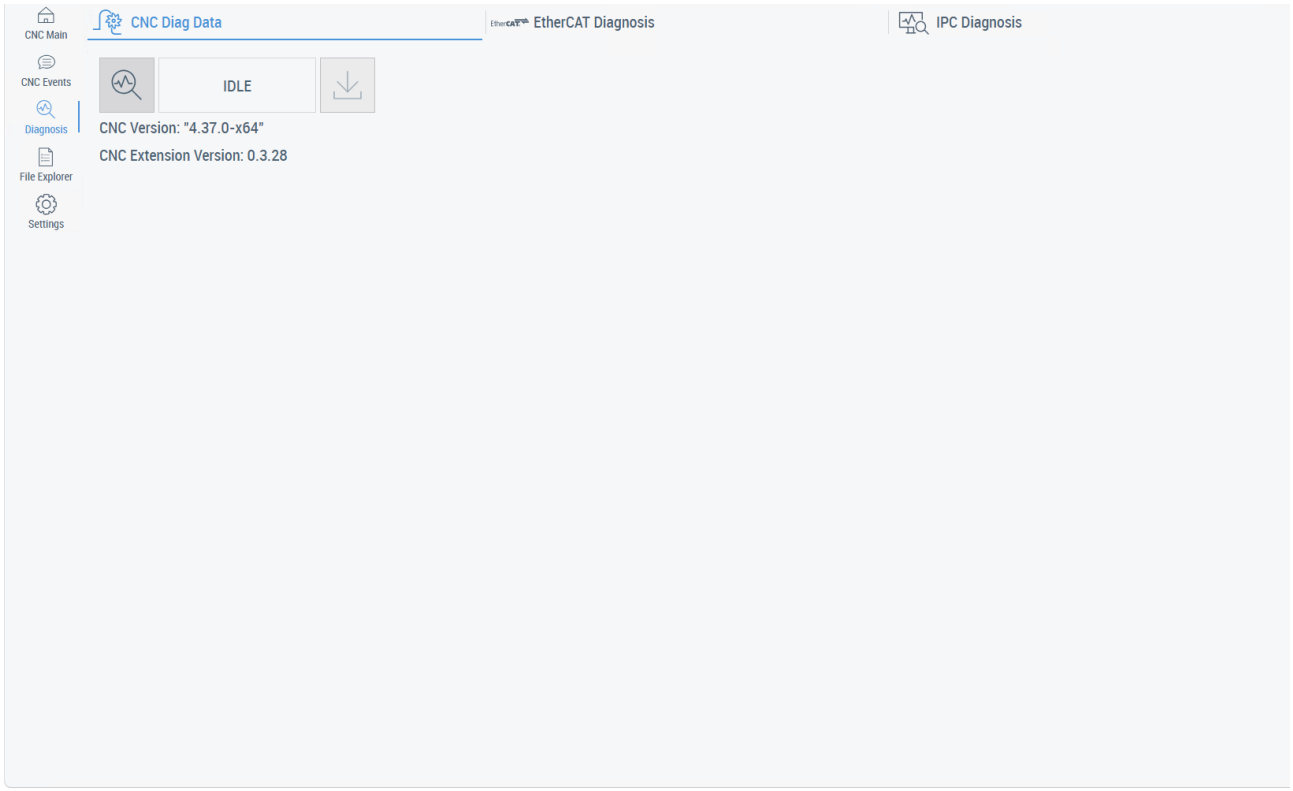
CNC Main



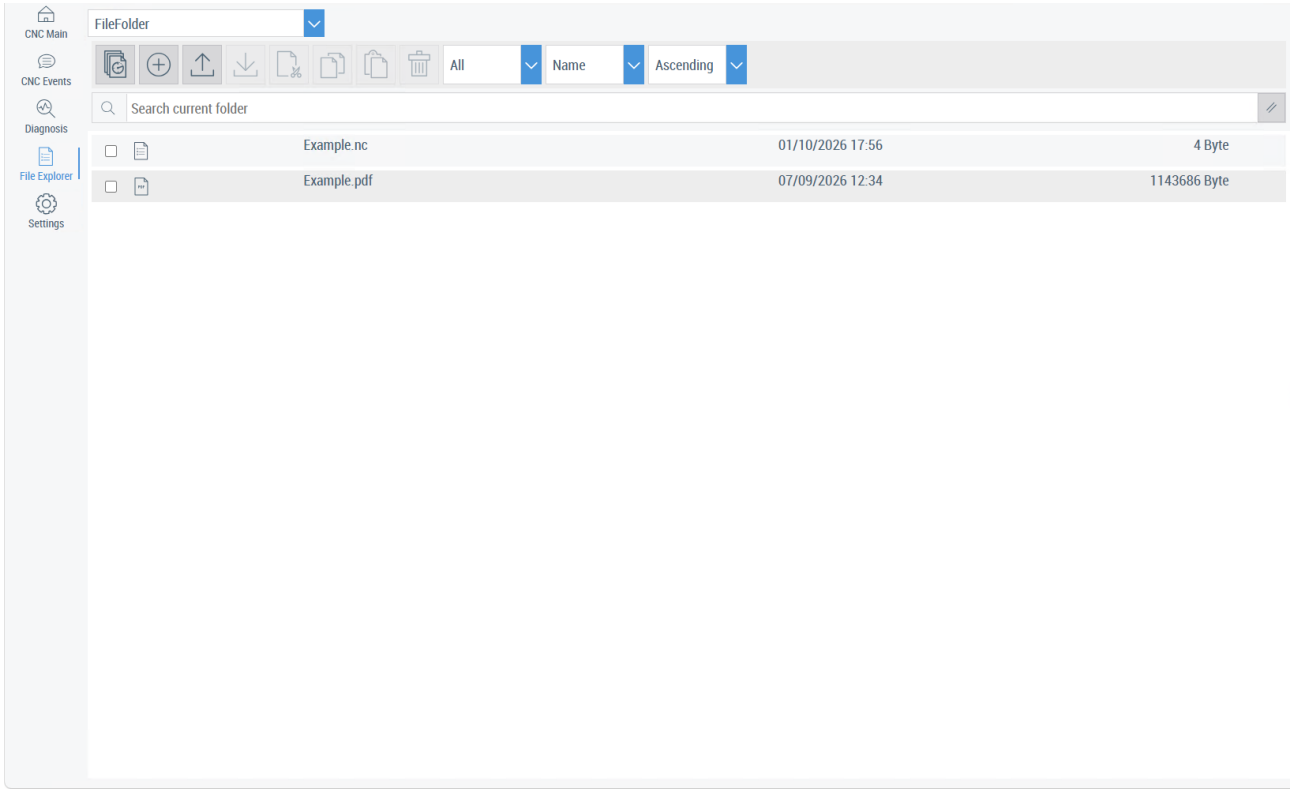
CNC Events



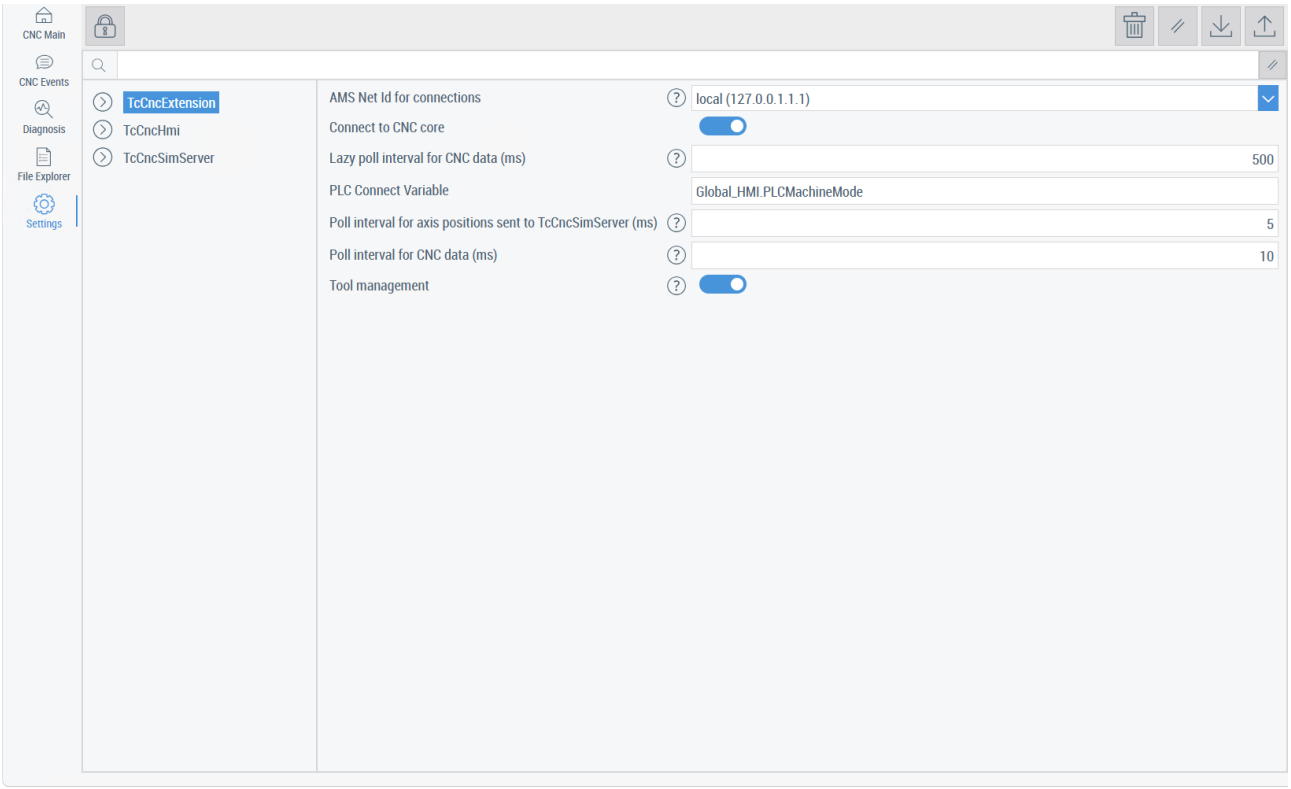
Diagnosis



File Explorer



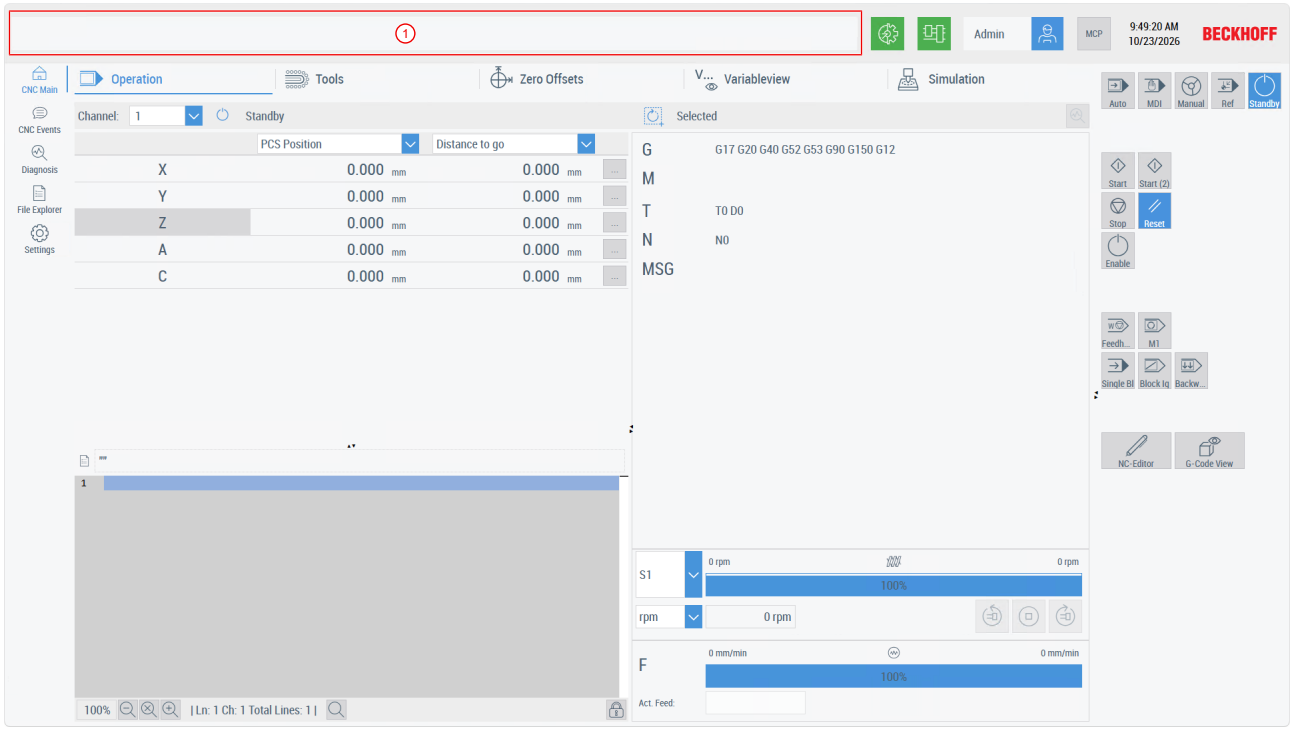
Settings



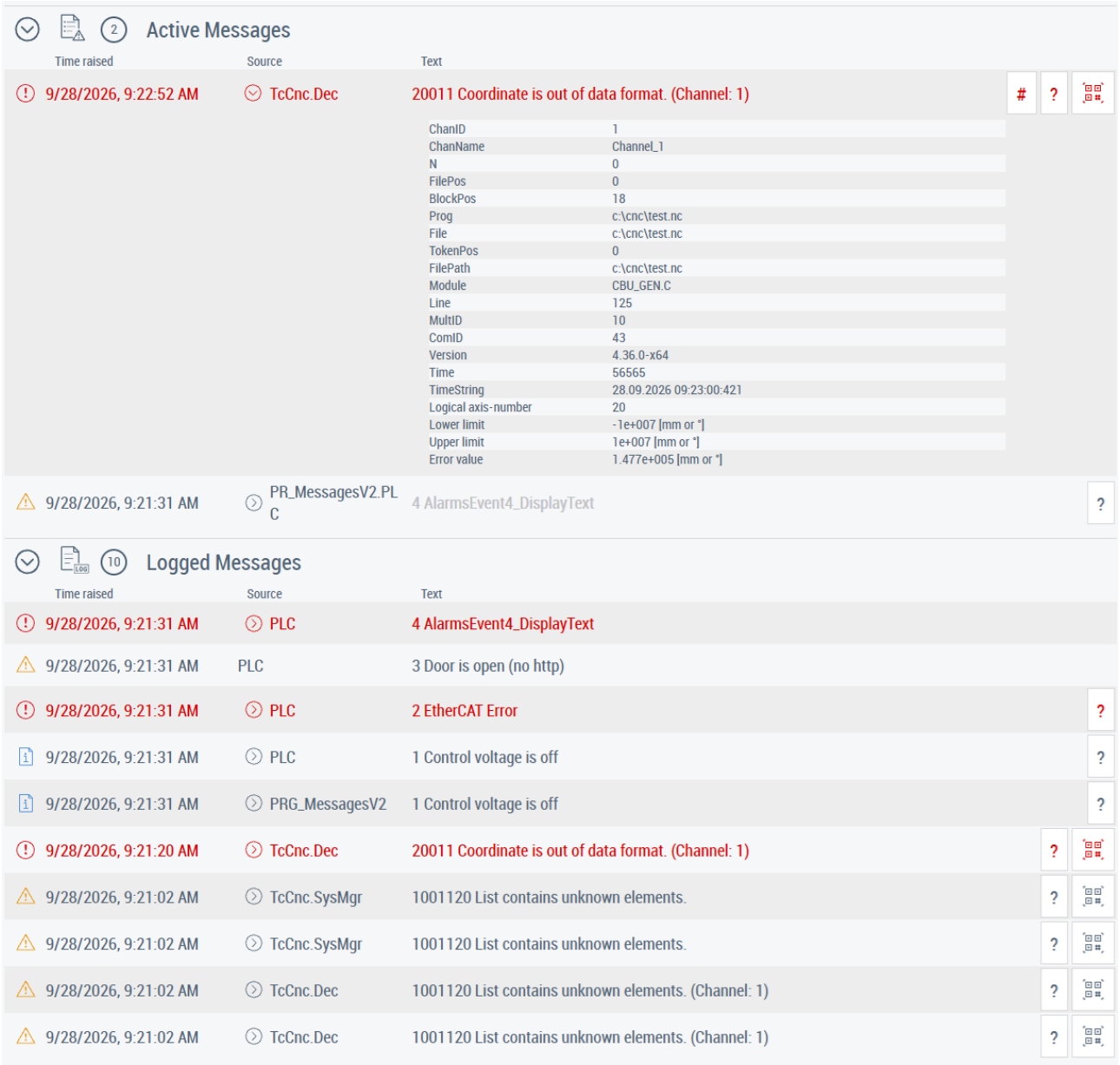
Main Content Area	Description
CNC Main	In the “CNC Main” main content area, you can, among other things: • Display the axis positions and move the axes manually (jog mode) • Display selected CNC functions (e.g., selected G and M commands) • Select and edit an NC Program • Adjust the spindle or feed override • Create tools or edit tool data • Set and manage zero offset shifts • Display CNC variables (e.g., V.E.)
CNC Events	The “CNC Events” main content area displays details about CNC messages that have not yet been acknowledged (i.e., active messages) as well as details about messages from the event logger.
Diagnosis	In the “Diagnosis” main content area, you can generate and export diagnostic data from the CNC. In addition, the TE2000’s EtherCAT diagnostics are integrated. In addition, diagnostic data from the IPC is displayed.
File Explorer	You can use File Explorer to manage your files for the CNC.
Settings	In the “Settings” main content area, you can configure settings that directly affect the HMI.

Warnings and error messages

Warning and error messages are displayed in the header via the TcHmiCncMessageLine [▶ 93] control.



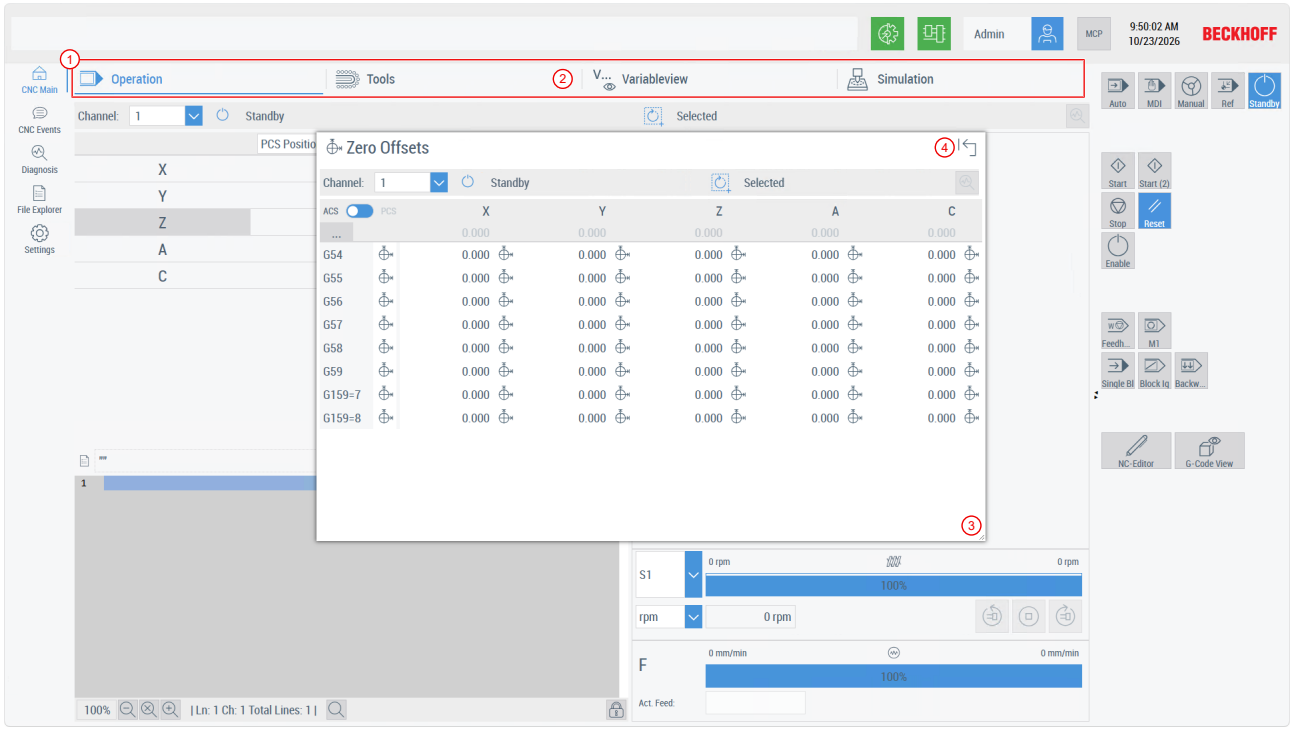
Double-click the warning or error message (1) to go to the “CNC Events” main content area. The control TcHmiCncMessageList [▶ 95] lists all unacknowledged (i.e., active) events from the TwinCAT Event logger. In addition, a configurable number of historical messages from the event logger will be displayed.



Detachable tab navigation in the main content area

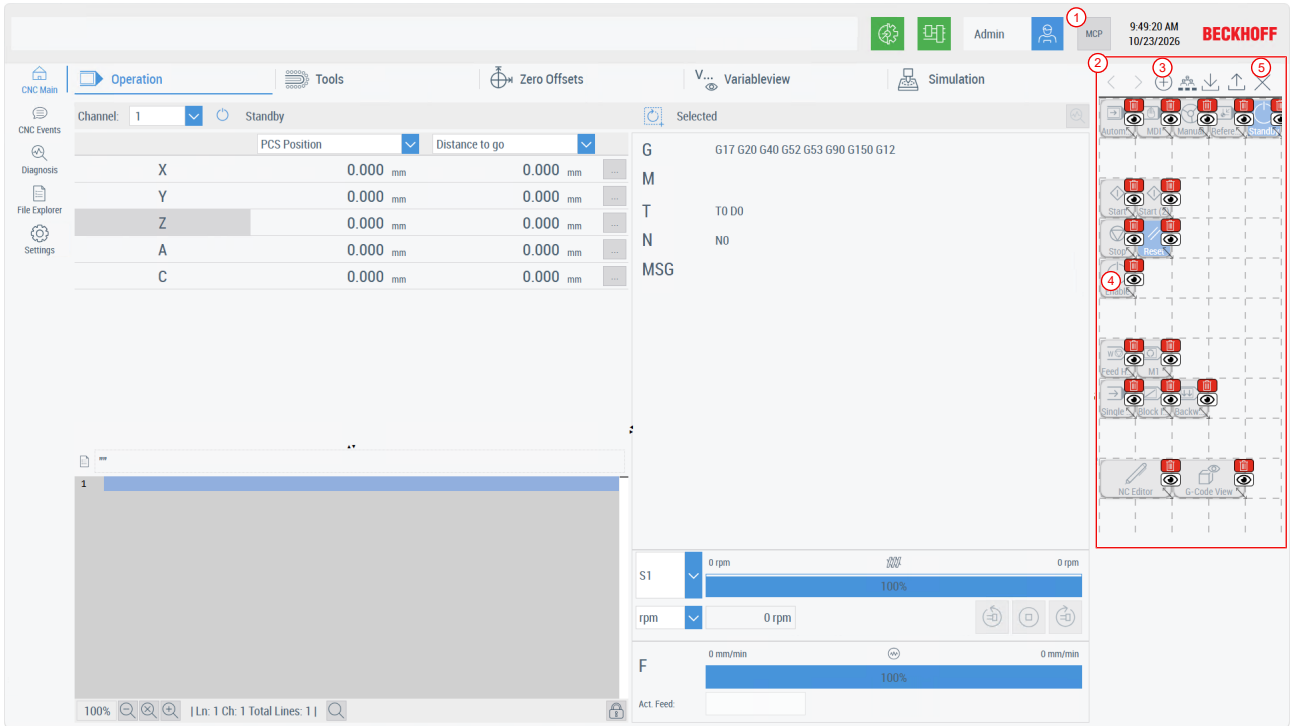
Within the main content area, there are several tabs in the navigation bar that you can switch between using the TcHmiCncDetachableTabNavigation [▶ 69] control.

1. Click one of the tabs in the navigation bar to switch between tabs.
2. Drag a tab to an area outside the navigation bar.
   ⇨ In the image, the “Zero Offsets” tab has been detached from the navigation bar. The pop-up window can be moved freely within the HMI.
3. If necessary, resize the pop-up window.
4. You can use the icon (back arrow) to move the tab back into the navigation bar.



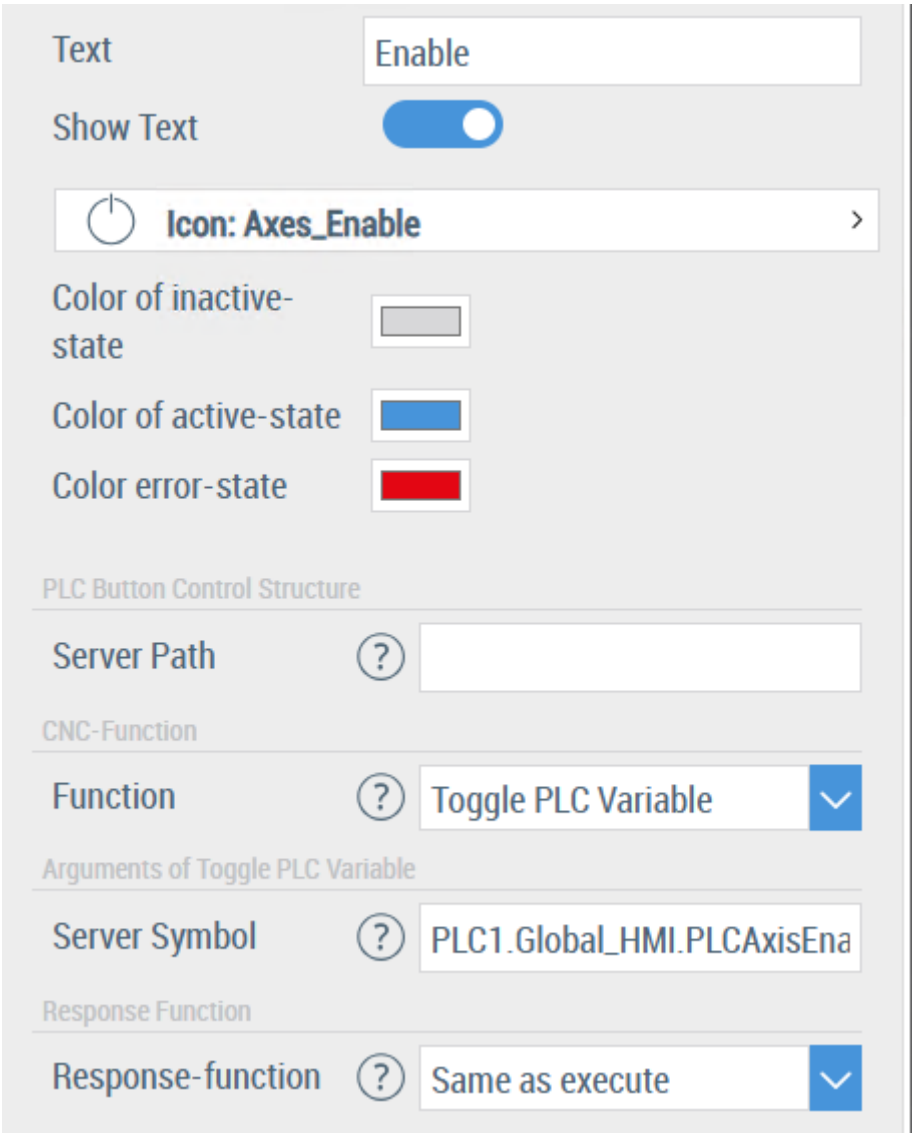
Dynamically generated buttons in the machine control panel

The buttons in the machine control panel are dynamically generated by the control TcHmiCncLayoutManager [▶ 79]. This allows you to change the layout of the buttons during the runtime of the HMI. In particular, you can also create new buttons. Proceed as follows:



1. Click the **MCP** (**M**achine **C**ontrol **P**anel) button.
2. You will then see the MCP layout.
3. If needed, use the **+** button to add a new button. Place the button in the desired position in the layout and adjust its size.
4. Click a button to configure it.

⇨ An additional dialog box appears, allowing you to configure the button. The following example shows a button for enabling the axes. The button sets the PLC variable “Global_HMI.PLCAxisEnable” to `true`.



5. Save the new layout by clicking the MCP button again or close the layout by clicking the X icon.

Display resolution

The CNC HMI template is optimized for a display resolution of 1920x1080 pixels with a 16:9 aspect ratio. In addition, you can also use the HMI at the lower resolutions listed below. However, as the resolution decreases, certain parts of the HMI may be displayed with increasingly limited detail. In particular, at resolutions below 1250x600 pixels, you should expect limited usability of the HMI. Therefore, use the CNC HMI template on a display with a resolution of at least 1920x1080 pixels to ensure unrestricted viewing.

Resolution	TFT display	Aspect ratio	Beckhoff Panel (Example)
1366x768	15.6"	16:9	CPx916
1366x768	18.5"	16:9	CPx918
1280x1024	19.0"	5:4	CPx919
1920x1080	21.5"	**16:9**	CPx921
1920x1080	24.0"	**16:9**	CPx924

4.4 Styling and themes

A number of CSS variables are defined within the TF5310 product, which ensure a consistent appearance for the CNC-specific controls. The appearance is based on the predefined “Base” and “Base-Dark” themes from the TwinCAT HMI product family. The CNC-specific controls are designed for these two themes. To do this, existing CSS variables (e.g., default colors) from both themes are used.

The individual CSS variables in TF5310 are grouped internally into CSS classes. CSS variables are inherited by the controls via CSS classes.

Customizing the theme’s appearance

You can globally customize the appearance of individual controls in the theme using the CSS variables described in the following sections. Proceed as follows:

1. Open the .css file for the theme whose CSS variables you want to customize. For the “Base” theme, for example, this is the “BaseStyle.css” file.
2. Navigate to the “:root” CSS pseudoclass in the file
3. Enter the appropriate CSS variables here and assign the desired values to them.

Example:

```
:root {
    --CNC-HMI-Button-Background: #424242;
    --tchmi-cnc-background-color-1: rgba(66,66,66,1);
}
```

The controls use default values for the individual CSS variables if you do not specify a value in the theme. With the default settings, the appearance matches the “Base” theme.

Standard colors

TF5310 uses standard colors from the “Base” and “Base-Dark” themes, with an alias variable assigned to each standard color. When switching between the two themes, the respective default color is assigned to an alias variable, which is then used in the individual controls. You can therefore use the alias variables to customize the appearance globally. The following table lists the default colors used in the “Base” and “Base-Dark” themes, along with their general uses.

Alias variable	“Base” theme	“Base-Dark” theme	Use
--tchmi-cnc-background-color-1	--tchmi-background-color-1	--tchmi-background-color-1	Main background (pages, dialogs)
--tchmi-cnc-background-color-2	--tchmi-background-color-2	--tchmi-background-color-2	Secondary Background (Navigation, Sidebar, Buttons)
--tchmi-cnc-background-color-3	--tchmi-background-color-3	--tchmi-background-color-3	Tertiary background (header, footer, table header)
--tchmi-cnc-background-color-4	--tchmi-background-color-4	--tchmi-background-color-4	Input fields
--tchmi-cnc-foreground-color-1	--tchmi-foreground-color-1	--tchmi-foreground-color-3	Regular text, icons
--tchmi-cnc-foreground-color-2	--tchmi-foreground-color-2	--tchmi-foreground-color-1	Highlighted text (active elements)
--tchmi-cnc-foreground-color-3	--tchmi-foreground-color-3	--tchmi-foreground-color-3	Reduced / lighter text, border
--tchmi-cnc-foreground-color-4	--tchmi-foreground-color-4	--tchmi-foreground-color-4	Another foreground
--tchmi-cnc-foreground-color-5	--tchmi-foreground-color-5	--tchmi-foreground-color-5	Very faint foreground (pressed buttons, disabled)
--tchmi-cnc-foreground-color-6	-	--tchmi-foreground-color-2	Frame color (for “Base Dark” only)
--tchmi-cnc-highlight-color-1	--tchmi-highlight-color-1	--tchmi-highlight-color-1	Primary accent color (active, selected)
--tchmi-cnc-highlight-color-2	--tchmi-highlight-color-2	--tchmi-highlight-color-2	Secondary accent color (hover)
--tchmi-cnc-shadow-color-1	--tchmi-shadow-color-1	--tchmi-shadow-color-1	Shadow
--tchmi-cnc-shadow-color-2	--tchmi-shadow-color-2	--tchmi-shadow-color-2	Dividing lines, splitters
--tchmi-cnc-error-color	--tchmi-error-color	--tchmi-error-color	Error color
--tchmi-cnc-warning-color	--tchmi-warning-color	--tchmi-warning-color	Warning color

Layout dimensions

You can control the basic dimensions of the individual controls using the following CSS variables.

Variable name	Meaning	Default value
--CNC-HMI-FixedInputControls-width	Standard width	50px
--CNC-HMI-FixedInputControls-height	Standard height	50px
--CNC-HMI-FixedMidInputControls-width	Width for medium-sized controls	35px
--CNC-HMI-FixedMidInputControls-height	Height for medium-sized controls	35px
--CNC-HMI-FixedSmallInputControls-width	Width for Small controls	30px
--CNC-HMI-FixedSmallInputControls-height	Height for Small controls	30px
--CNC-HMI-Padding	Global padding	5px
--CNC-HMI-Checkbox-width	Width of Checkbox controls	60px
--CNC-HMI-Table-Row-Height	Row Height in Tables	--CNC-HMI-FixedMidInputControls-height
--CNC-HMI-Table-Header-Height	Height of Table Headers	--CNC-HMI-FixedMidInputControls-height
--CNC-HMI-Table-Header-Line-Thickness	Line Weight Table Header	2px
--CNC-HMI-Table-Line-Thickness	Line Weight of Table Cells	1px

Fonts

You can control the fonts and font sizes used with the CSS variables listed below.

Variable name	Meaning	Default value
--CNC-HMI-Font	Font	RobotoCondensed
--CNC-HMI-HeadlineFontSize	Headings	26px
--CNC-HMI-TitleFontSize	Title	24px
--CNC-HMI-SubTitleFontSize	Subtitle	20px
--CNC-HMI-BodyFontSize	Body text, properties window	18px
--CNC-HMI-LabelFontSize	Labels, button labels	14px
--CNC-CycleSelectionTextSize	Font size in the cycle selection	--CNC-HMI-BodyFontSize

Colors

Texts

You can control the color of text using the following CSS variables.

Variable name	Meaning	Default value
--CNC-HMI-NormalFontColor	Standard text	--tchmi-cnc-foreground-color-1
--CNC-HMI-HighlightFontColor	Highlighted text	--tchmi-cnc-foreground-color-2
--CNC-HMI-LightFontColor	Weaker text	--tchmi-cnc-foreground-color-3
--CNC-HMI-DisabledFontColor	Disabled elements	--tchmi-cnc-foreground-color-5
--CNC-HMI-WarnFontColor	Warning message text	--tchmi-cnc-foreground-color-1
--CNC-HMI-AlarmFontColor	Alarm message text	--tchmi-cnc-error-color
--tchmi-icons-color	Color for SVG icons	--tchmi-cnc-foreground-color-1

Buttons

You can control the color of buttons using the following CSS variables.

Variable name	Meaning	Default value
--CNC-HMI-Button-Background	Default background	--tchmi-cnc-background-color-2
--CNC-HMI-Button-Background-active	Active state	--tchmi-cnc-highlight-color-1
--CNC-HMI-Button-Background-disabled	Disabled state	--tchmi-cnc-background-color-3
--CNC-HMI-Button-Background-error	Error state	--tchmi-cnc-error-color
--CNC-HMI-Button-Background-pressed	Pressed state	--tchmi-cnc-foreground-color-5 (Theme “Base”) --tchmi-cnc-background-color-4 (Theme “Base-Dark”)
--CNC-HMI-Button-Border	Border	--tchmi-cnc-foreground-color-3 (Theme “Base”) Black (Theme “Base-Dark”)
--CNC-HMI-Hover-Color	Hover state	--tchmi-cnc-highlight-color-2

Input fields

You can control the color of input fields using the following CSS variables.

Variable name	Meaning	Default value
--CNC-HMI-Input-Background	Default background	--tchmi-cnc-background-color-4
--CNC-HMI-Input-Background-active	Focused input field	--tchmi-cnc-highlight-color-1
--CNC-HMI-Input-Background-error	Faulty input	--tchmi-cnc-error-color
--CNC-HMI-Input-Background-warning	Missing input	--tchmi-cnc-warning-color
--CNC-HMI-Input-disabled	Disabled input field	--tchmi-cnc-background-color-3

Layout backgrounds

You can control the color of backgrounds in layouts using the following CSS variables.

Variable name	Meaning	Default value
--CNC-HMI-Main-Background	Main Content Background	--tchmi-cnc-background-color-1
--CNC-HMI-Header-Background	Headers, dialogs	--tchmi-cnc-background-color-3
--CNC-HMI-Footer-Background	Footers	--tchmi-cnc-background-color-3
--CNC-HMI-Navigation-Background	Navigation Bars	--tchmi-cnc-background-color-2
--CNC-HMI-Navigation-HighlightFontColor	Active navigation entry	--tchmi-cnc-highlight-color-1
--CNC-HMI-Sidebar-Background	Sidebars	--tchmi-cnc-background-color-2
--CNC-HMI-Dialog-Background	Dialog Background	--tchmi-cnc-background-color-1
--CNC-HMI-Dialog-Header-and-Border	Dialog header and border	--tchmi-cnc-foreground-color-3
--CNC-HMI-Splitter-Background	Dividing lines between layout areas	--tchmi-cnc-shadow-color-2
--CNC-HMI-Standard-Border	Border	--tchmi-cnc-background-color-2 (Theme “Base”) --tchmi-cnc-foreground-color-6 (Theme “Base-Dark”)
--CNC-HMI-ToolTip-Background	Tooltip background	--CNC-HMI-Button-Border
--CNC-HMI-ToolTip-Highlight	Highlighted tooltip text	--tchmi-cnc-highlight-color-1

Tables

You can control the color of tables using the following CSS variables.

Variable name	Meaning	Default value
--CNC-HMI-Table-Background	Table Background	--tchmi-cnc-background-color-1
--CNC-HMI-Table-Header-Background	Background: Table Header	--tchmi-cnc-background-color-3
--CNC-HMI-Selected-Background	Selected table row	--tchmi-cnc-highlight-color-1

Additional CSS variables

Variable name	Meaning	Default value
--CNC-HMI-Checkbox-Background	Background of checkboxes	--tchmi-cnc-background-color-2
--CNC-HMI-Checkbox-Background_active	Active checkbox	--tchmi-cnc-highlight-color-1
--CNC-HMI-Checkbox-Dot-Background	Check mark in the checkbox	--tchmi-cnc-foreground-color-2 (Theme “Base”) --tchmi-cnc-foreground-color-3 (Theme “Base-Dark”)
--CNC-HMI-Combobox-active	Open combo box	--tchmi-cnc-background-color-3
--CNC-HMI-Tab-Header-Item-active	Active tab header	--tchmi-cnc-background-color-2
--CNC-HMI-Tab-Header-Span-Background	Inactive Tab Header	--tchmi-cnc-background-color-3

4.5 User management

The TE2000 product includes an authorization system to control user-specific access at the control level. Based on this authorization system, the template provides you with predefined users for the HMI. In addition to the predefined users, you can create your own users and user groups and configure access rights for the individual controls. For each control, the chapter Control elements [▶ 33] describes how the control behaves depending on the access right.

Default user

The template uses the four user groups “Admins,” “AdvancedOperators,” “Operators,” and “Viewers”. Each predefined user is assigned to one of these user groups and inherits the access rights of that user group. The standard users are listed in the following table, along with their key restrictions regarding HMI use:

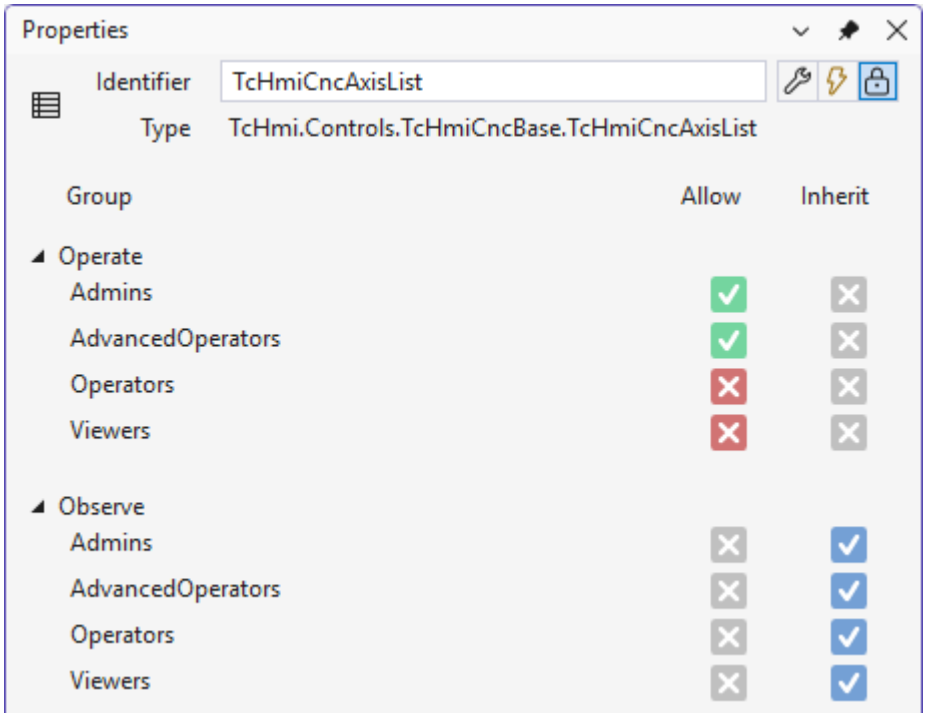
User	User group	Default password	Description
__SystemAdmin	__Systemadministrators	1	Full access.
Admin	Admins	1	Restricted by: • No user management
AdvancedOperator	AdvancedOperators	1	Restricted by: • No user management • No rights management in TcHmiCncSettings
Operator	Operators	1	Restricted by: • No user management • No rights management in TcHmiCncSettings • No changes to the Machine Control Panel (MCP)
Viewer	Viewers	1	HMI display only, without CNC control

Changing default passwords

The template comes with default passwords. Change the default passwords before commissioning your machine.

Control access rights

The controls use the basic access right “Operate” from the TE2000 product. The appearance and behavior of the controls depend on this access right. For each user group, configure the access permission in each control you use to “Allow,” “Disallow,” or “Inherit”. The template includes preconfigured access rights for the four user groups.



Control in the template	Operate			
	Admins	AdvancedOperators	Operators	Viewers
TcHmiCncAxisList	Allow	Allow	Disallow	Disallow
TcHmiCncButton	Inherit	Inherit	Inherit	Inherit
TcHmiCncChannelStatus	Inherit	Inherit	Inherit	Inherit
TcHmiCncCombobox	Inherit	Inherit	Inherit	Inherit
TcHmiCncConfigurableButton	Allow	Allow	Disallow	Disallow
TcHmiCncContourWorkbench	Allow	Allow	Disallow	Disallow
TcHmiCncDetachableTabNavigation	Allow	Allow	Allow	Disallow
TcHmiCncDiagData	Inherit	Inherit	Inherit	Inherit
TcHmiCncFileExplorer	Allow	Allow	Allow	Disallow
TcHmiCncIpcDiagnosis	Inherit	Inherit	Inherit	Inherit
TcHmiCncLayoutManager	Allow	Allow	Allow	Disallow
TcHmiCncLinearOverride	Allow	Allow	Allow	Disallow
TcHmiCncMessageLine	Inherit	Inherit	Inherit	Inherit
TcHmiCncMessageList	Inherit	Inherit	Inherit	Inherit
TcHmiCncPriorityGrid	Inherit	Inherit	Inherit	Inherit
TcHmiCncScaleRegions	Allow	Allow	Allow	Disallow
TcHmiCncSettings	Allow	Disallow	Disallow	Disallow
TcHmiCncSimulation	Inherit	Inherit	Inherit	Inherit
TcHmiCncTechnoData	Inherit	Inherit	Inherit	Inherit
TcHmiCncTextEditor	Allow	Allow	Disallow	Disallow
TcHmiCncToolList	Allow	Allow	Disallow	Disallow
TcHmiCncTreeview	Inherit	Inherit	Inherit	Inherit
TcHmiCncVariableView	Inherit	Inherit	Inherit	Inherit
TcHmiCncZeroOffsets	Allow	Allow	Disallow	Disallow

In addition to the “Operate” access right, there is an additional “Editable” access right for the following controls.

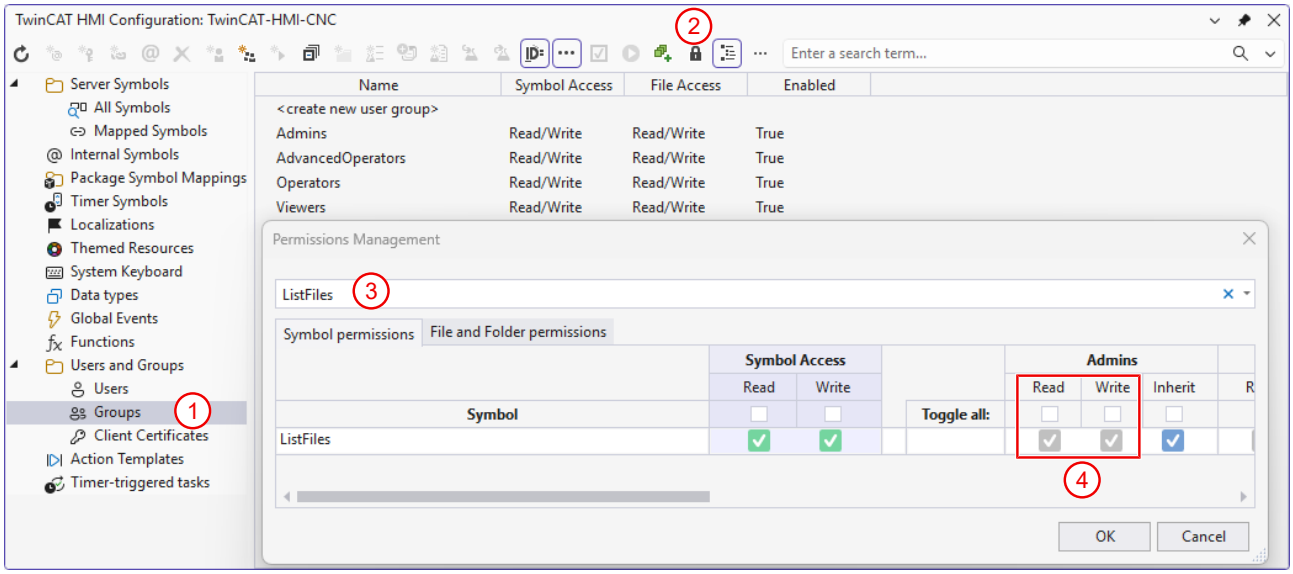
Control	Editable			
	Admins	AdvancedOperators	Operators	Viewers
TcHmiCncConfigurableButton	Allow	Allow	Disallow	Disallow
TcHmiCncLayoutManager	Allow	Allow	Disallow	Disallow

Settings in permissions management for new user groups

In addition to the predefined user groups, you can create your own user groups. New user groups are assigned a set of default permissions. Make sure that the following rights have also been granted to the user group:

- AddSymbol
- ListFiles
- ListUserGroup

Without these rights, the controls may have limited functionality.



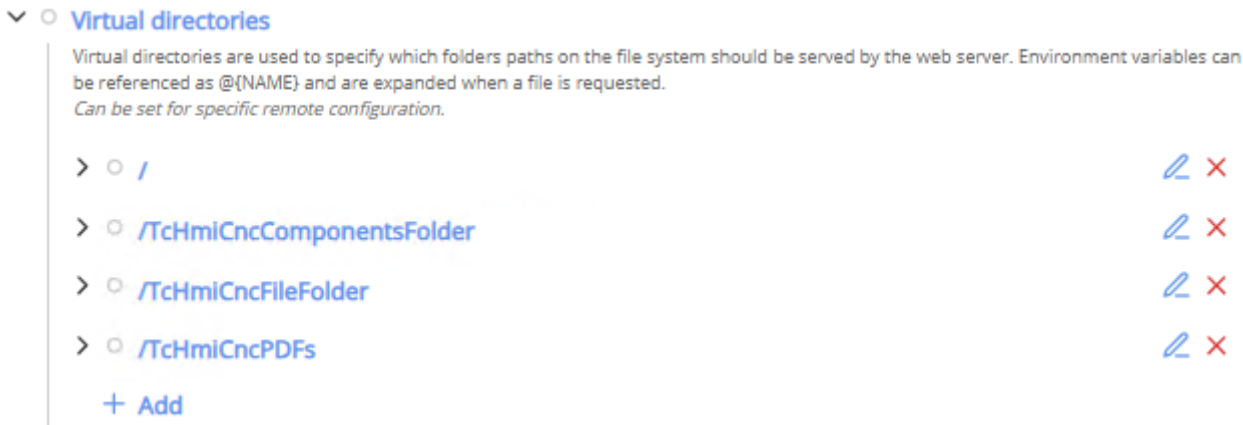
In the TwinCAT HMI Configuration, first click (1), and then open the rights management screen by clicking (2). Search for the appropriate permission in (3) and make sure that “Read” and “Write” are enabled for the user group (4). These settings are already preconfigured in the template.

4.6 Internationalization

The controls use the TE2000’s features for language switching. Language switching in TwinCAT HMI is implemented with keys, which are managed in language tables (localization files). Using the language table, you can change the text displayed in the controls based on the keys. A language table for English is included with the TF5310 product. For other languages, you can configure your own language tables. Use the Localization Editor to manage your language tables.

4.7 Virtual directories

The controls access virtual directories. These include, for example, the control TcHmiCncFileExplorer [▶ 73]. Use TE2000 to create the following virtual directories:



Directory	Default directory
TcHmiCncComponentsFolder	C:\Program Files (x86)\Beckhoff\TwinCAT\3.1\Components\Mc\Cnc
TcHmiCncFileFolder	C:\cnc
TcHmiCncPDFs	C:\Program Files (x86)\Beckhoff\TwinCAT\3.1\Components\Mc\Cnc\Documents

5 Control elements

5.1 TcHmiCncAdsState

Component: `TcHmiCncAdsState`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

The control `TcHmiCncAdsState` displays the ADS state using configurable background colors. In addition, you can display an icon as an SVG file in the control.



Operation

The control `TcHmiCncAdsState` does not have any special operating functions.

Configuration

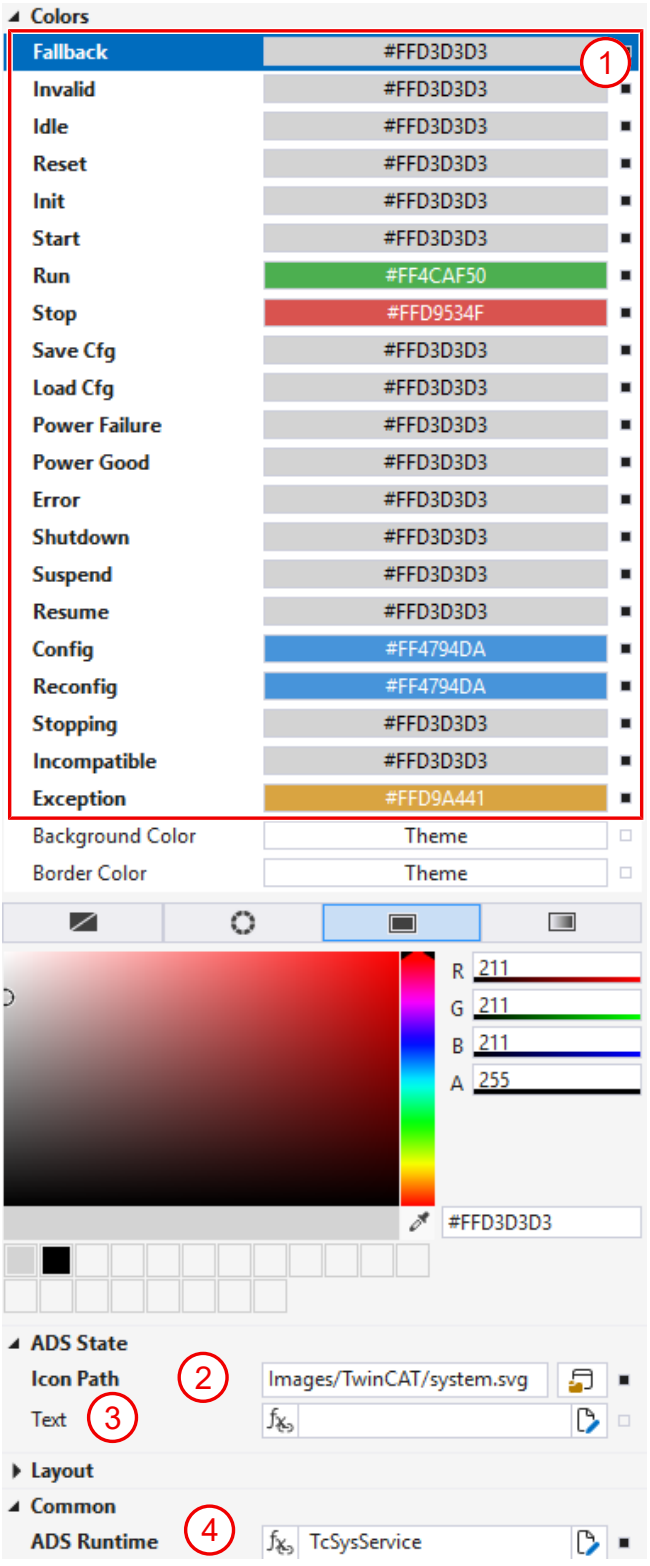
✓ Configure the control `TcHmiCncAdsState` as follows.

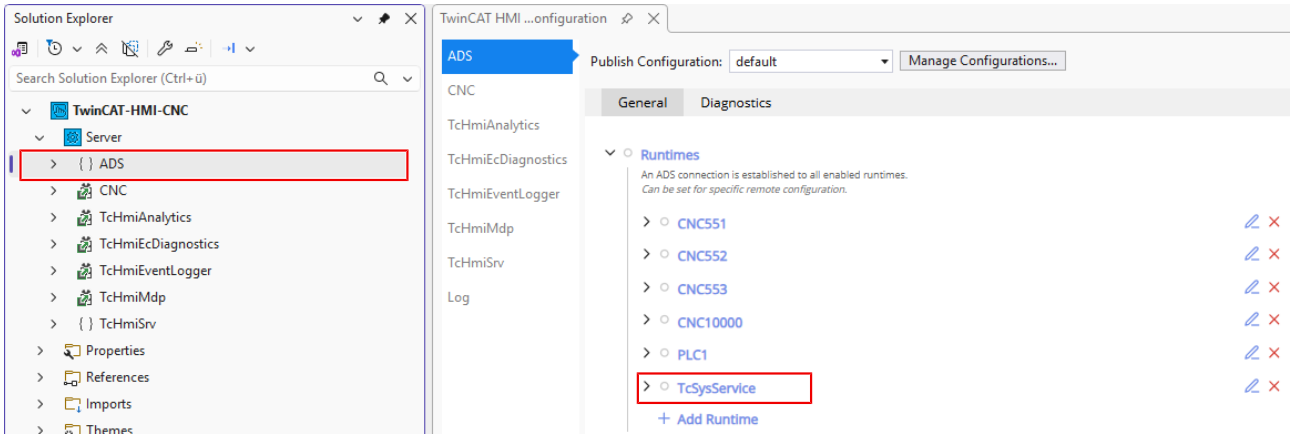


You do not need any special data binding with the TcHmiCnc extension.

1. In the “Colors” context menu, configure the background color of the control that should be displayed for each ADS state.
2. Optionally, under “Icon Path,” specify an SVG file to be displayed as an icon in the control.
3. Optionally, you can also define text to be displayed in the control.
4. Specify the runtime for which the ADS state should be displayed. You can find the available ADS runtimes in the server configuration.

⇨ In the configuration example, the control has a connection to the ADS runtime “TcSysService”.


Colors
Fallback #FFD3D3D3
Invalid #FFD3D3D3
Idle #FFD3D3D3
Reset #FFD3D3D3
Init #FFD3D3D3
Start #FFD3D3D3
Run #FF4CAF50
Stop #FFD9534F
Save Cfg #FFD3D3D3
Load Cfg #FFD3D3D3
Power Failure #FFD3D3D3
Power Good #FFD3D3D3
Error #FFD3D3D3
Shutdown #FFD3D3D3
Suspend #FFD3D3D3
Resume #FFD3D3D3
Config #FF4794DA
Reconfig #FF4794DA
Stopping #FFD3D3D3
Incompatible #FFD3D3D3
Exception #FFD9A441
Background Color Theme
Border Color Theme
R 211
G 211
B 211
A 255
#FFD3D3D3
ADS State
Icon Path Images/TwinCAT/system.svg
Text
Layout
Common
ADS Runtime TcSysService
1
2
3
4



User settings via TcHmiCncSettings

The control `TcHmiCncAdsState` does not have any customizable settings in the control `TcHmiCncSettings`.

Keyboard shortcuts

The control `TcHmiCncAdsState` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncAdsState` has a public API that allows you to use the following functions.

Function	Description
setIconPath() / getIconPath()	Set or read the path of the displayed SVG file.
setText() / getText()	Set or read text.
set<ADS-state>Color() / get<ADS-state>Color()	Set or read the color for the current state, e.g., setRunColor() / getRunColor().

Impact of user management

The control `TcHmiCncAdsState` does not exhibit any specific change in behavior as a result of the “Operate” access right.

5.2 TcHmiCncAxisList

Component: `TcHmiCncAxisList`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

The control `TcHmiCncAxisList` displays an overview of the CNC axes for a channel.



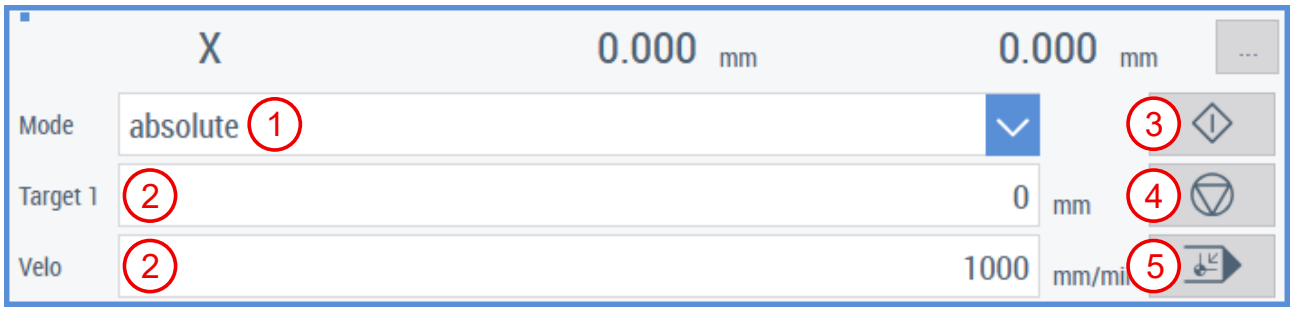
No.	Description
1	You can switch the position display between axis coordinate system (ACS) and part coordinate system (PCS) using the combo box.
2	Using the combo box, you can switch between displaying “Distance to go,” “Active Feedrate,” “Position Lag,” and “Torque”.
3	A dot indicates whether the axis is enabled.
4	A gray background in the axis name indicates that the axis is not referenced.
5	Click the button to view the online data for the axis.
6	An axis in an error state is indicated by red lines above and below the axis name.

Operation

In “Automatic”, “Reference”, and “Standby” operation modes, the control `TcHmiCncAxisList` provides you with display functions only. In “MDI” and “Manual” operation modes, the control provides additional buttons that you can use to move an axis.

✓ You are in “MDI” operation mode and have selected an axis by clicking on the desired axis.

1. Use the combo box to select one of the predefined motion patterns.
2. Configure the motion parameters.
3. Click this button to start the movement.
4. Use this button to stop the movement.
5. You can use this button to start a homing.



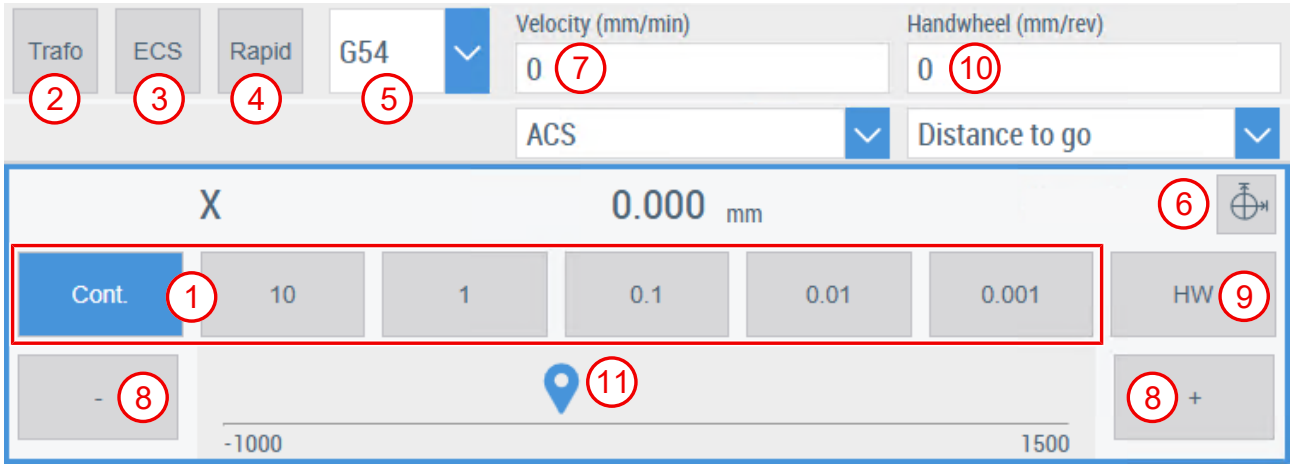
Overwriting the MDI string

An MDI string is automatically generated for the movement, which overwrites an existing entry.

✓ You are in “Manual” operation mode and have selected an axis by clicking on the desired axis.

1. Specify whether the axis should move continuously or in fixed increments.
2. Specify whether a kinematic transformation (“trafo”) is used during the operation.
3. Specify whether to use the effector coordinate system (“ECS”).
4. Set whether to use the “Rapid” mode.
5. Use the combo box to select which zero offset to use.
6. If necessary, set a new zero offset. This button is hidden when zero offset G53 is active.
7. Set the feed rate for the axis movement.
8. Use the + and – buttons to move the axis.
9. Optional: Move the axis using a hand wheel.
10. Optional: Set the handwheel resolution.

⇨ For referenced axes, the axis position is displayed relative to the software end positions (11)

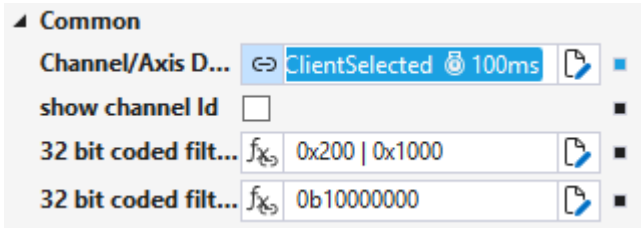


Configuration

The control `TcHmiCncAxisList` requires a data binding to a symbol from the TcHmiCnc extension. You can use three different axis lists:

1. Axes of the currently selected channel.
2. Axes of a specific channel. The axis list is retained even when you select a different channel.
3. All Axes

Axes with the string “_dx” in their axis names will not be displayed in the list in any of the three cases.



No.	Data Binding	Symbol
1	Channel/Axis Data	CNC.Data.ChannelClientSelected
2	Channel/Axis Data	CNC.Data.Channel.<CHANNEL> (For CNC.Data.Channel.2, only the axes in this channel are displayed)
3	Channel/Axis Data	CNC.Data.Axes

Axes not assigned to a channel

Please note that when all axes are displayed, axes that are not assigned to a channel are also shown. You cannot move axes that are not assigned to a channel. In this case, you cannot use the context menus described for moving the axes in “MDI” and “Manual” operation modes.

User settings via TcHmiCncSettings

You can adjust the following settings via the control `TcHmiCncSettings`:

Section	Setting	Standard	Description
TcCncHmi	Decimal places	3	Decimal places in the axis position display
TcCncHmi	Decimal places for feed rate display	0	Decimal places in the feed rate display
TcCncHmi.Axis List view	Automatically change the axis name when #TRAFO is ON/OFF	`"false"`	In `"true"`, either the axis name in the channel or the actual axis name is displayed, depending on the transformation status.

Keyboard shortcuts

You can use the following keyboard shortcut in the control `TcHmiCncAxisList`:

Shortcut	Action
Enter	The value entered in the input field is accepted and written to the controller.

API functions

The control `TcHmiCncAxisList` has a public API that allows you to use the following functions:

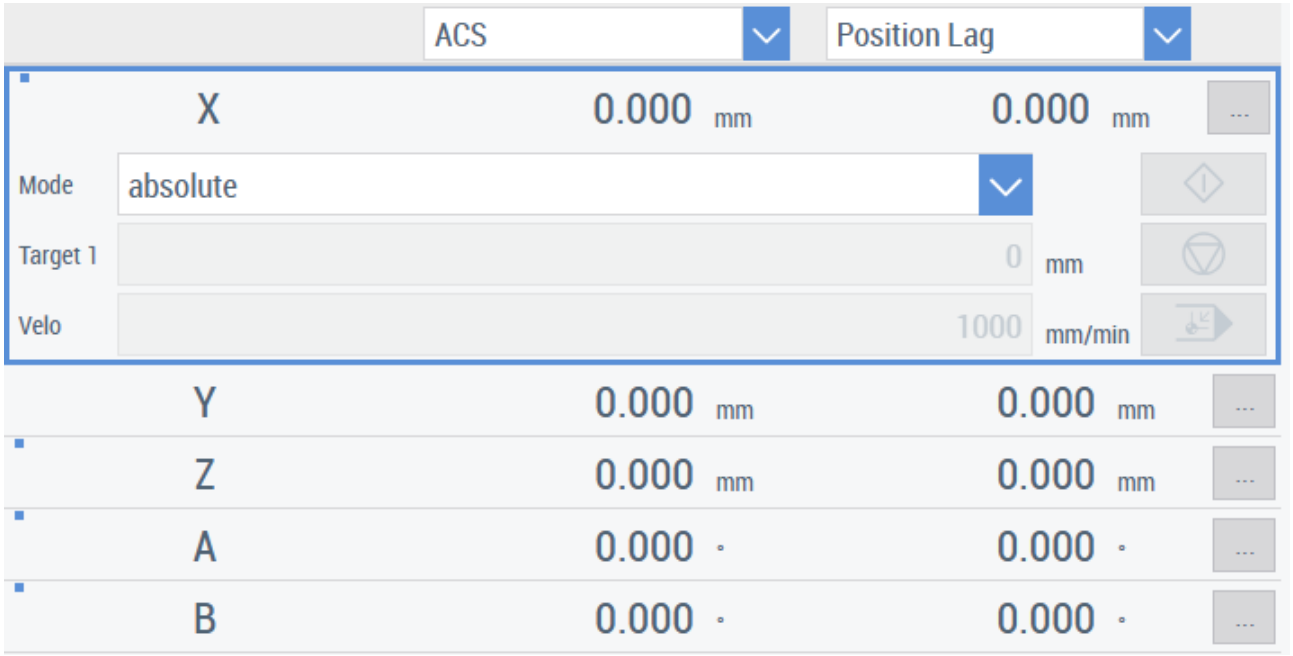
Function	Description
setDataSymbol() / getDataSymbol()	Replaces the data icon. Channel, axis, and spindle symbols are all accepted and automatically recognized.
setShowChannelID() / getShowChannelId()	Turn the channel ID display on or off.
setFilterStopConditions1() / getFilterStopConditions1()	Set or read the filter for stop condition 1.
setFilterStopConditions2() / getFilterStopConditions2()	Set or read the filter for stop condition 2.
setShowVerticalLines() / getShowVerticalLines()	Turn vertical dividing lines on or off.
setShowHorizontalLines() / getShowHorizontalLines()	Turn horizontal dividing lines on or off.
setRowHeight() / getRowHeight()	Specify the line height in pixels.
setAxisArrayConnect() / getAxisArrayConnect()	Transfer axis data directly rather than retrieving it via the data symbol.
setChannelMode() / getChannelMode()	Set the channel mode to control which panel is displayed.
setManualOperation() / getManualOperation()	Set or read the manual operation state – jog mode, increment, rapid, transformation, ECS, and selected zero offset.

Impact of user management

The control `TcHmiCncAxisList` evaluates the “Operate” access right. For “Allow,” the control behaves as usual.

Restrictions of the control on the “Disallow” access right:

- All axis values and states are *displayed* in read-only mode.
- Both combo boxes remain functional because they only toggle the display.
- You can still open the detailed view of an axis.
- In MDI operation mode, the combo box for selecting the motion remains operable, since the motion is not executed until you click Start.



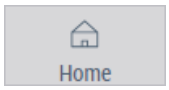
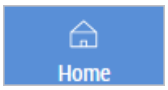
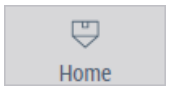
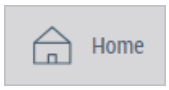
If you have not set up user management, “Operate” is considered “Allow,” and the control behaves as usual.

5.3 TcHmiCncButton

Component: `TcHmiCncButton`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

The control `TcHmiCncButton` is used as the default interaction button in the CNC HMI. The control is a generic button in which you can display an icon as an SVG file. Optionally, you can include a text label in the button.

Inactive	Active	Error	Pressed	Hover	Rotates	Horizontal layout
						
--CNC-HMI-Button-Background	--CNC-HMI-Button-Background-active	--CNC-HMI-Button-Background-error	--CNC-HMI-Button-Background-pressed	--CNC-HMI-Hover-Color	“Width Scale” = -1 “Height Scale” = -1	“Horizontal Layout” = `true`

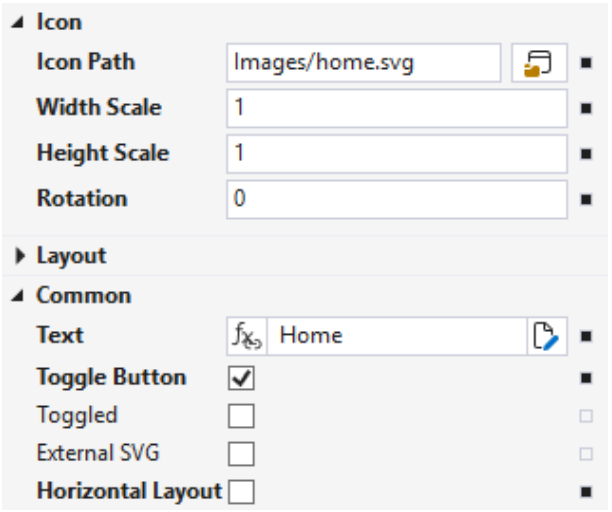
You can use the control `TcHmiCncButton` as a pushbutton or a switch. You can use CSS variables to configure different background colors for the button’s four states: “inactive,” “active,” “error,” and “pressed”. You can also set the background color that appears when you “hover” over the button. You can rotate the icon displayed on the button and set the icon and the text label to be displayed horizontally next to each other.

Operation

The control `TcHmiCncButton` does not have any special operating functions.

Configuration

You can configure the control `TcHmiCncButton` as follows. You do not need any special data binding with the TcHmiCnc extension.



No.	Description
1	You can specify the path to the icon as an SVG file here.
2	If you set “Width scale” or “Height scale” to “-1,” the icon will be rotated 180°.
3	You can optionally enter a text label for the button here.
4	`False` (default): The button functions as a pushbutton. `True`: The button functions as a switch.
5	`True`: The button is active by default.
6	`True`: Mark the SVG file as an external resource.
7	`True`: The icon and the text label are arranged horizontally relative to each other.

User settings via TcHmiCncSettings

The control `TcHmiCncButton` does not have any customizable settings in the control `TcHmiCncSettings`.

Keyboard shortcuts

The control `TcHmiCncButton` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncButton` has a public API that allows you to use the following functions.

Function	Description
setText() / getText()	Set or read a text label.
setIconPath() / getIconPath()	Set or read the path to the SVG file.
setIconWidthScale() / getIconWidthScale()	Set or read the horizontal scaling factor.
setIconHeightScale() / getIconHeightScale()	Set or read the vertical scaling factor.
setToggleButton() / getToggleButton()	Switch between using the button as a pushbutton or a switch.
setToggled() / getToggled()	Set or check whether the button is active as a toggle in its initial state.
setExternalSVG() / getExternalSVG()	Mark the SVG file as an external resource.
setHorizontalLayout() / getHorizontalLayout()	Switch between vertical and horizontal layouts.
setActiveColor() / getActiveColor()	Set or read the color of the active state.
setInactiveColor() / getInactiveColor()	Set or read the color of the inactive state.
setErrorColor() / getErrorColor()	Set or read the error status color.

Impact of user management

The control `TcHmiCncButton` evaluates the “Operate” access right. For “Allow,” the control behaves as usual. When set to “Disallow,” the control is limited to the following:

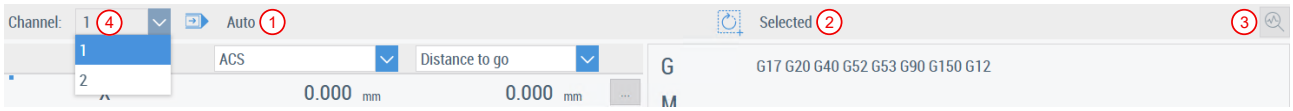
- The button remains visible, but no event is triggered.

5.4 TcHmiCncChannelStatus

Component: `TcHmiCncChannelStatus`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

The control `TcHmiCncChannelStatus` displays status information for a CNC channel.



No.	Description
1	Current operating mode
2	Current operating state
3	Button to display active stop conditions in a pop-up window. This button is only available if at least one stop condition (e.g., “Feedhold”) is set.
4	Combobox for switching between CNC channels. This combo box appears only when there is more than one channel.

The following states are distinguished as operating modes and operating states and are indicated by an icon.

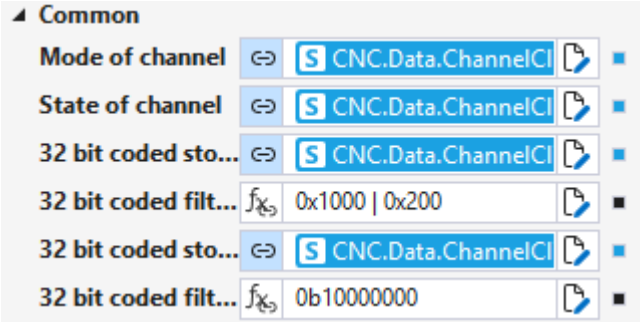
Value	Operating mode	Icon	Value	Operating state	Icon
0	Not Ready		1	Deselected	
1	Standby		2	Selected	
2	Automatic		3	Ready	
3	MDI		4	Active	
4	Manual		5	Hold	
5	Reference		6	Error	
			15	Resetting	

Operation

The control `TcHmiCncChannelStatus` does not have any special operating functions.

Configuration

Link the `TcHmiCncChannelStatus` control to the following symbols from the TcHmiCnc extension:



Data Binding	Symbol
Mode of channel	CNC.Data.ChannelClientSelected.Mode
State of Channel	CNC.Data.ChannelClientSelected.State
32-bit-coded stop conditions 1	CNC.Data.ChannelClientSelected.StopConditions1
32-bit-coded stop conditions 2	CNC.Data.ChannelClientSelected.StopConditions2

User settings via TcHmiCncSettings

The control `TcHmiCncChannelStatus` does not have any customizable settings in the control `TcHmiCncSettings`.

Keyboard shortcuts

The control `TcHmiCncChannelStatus` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncChannelStatus` has a public API that allows you to use the following functions:

Function	Description
setChannelMode() / getChannelMode()	Set or read the operation mode.
setChannelState() / getChannelState()	Set or read the operating state.
setStopConditions1() / getStopConditions1()	Set or read stop condition 1.
setStopConditions2() / getStopConditions2()	Set or read stop condition 2.
setFilterStopConditions1() / getFilterStopConditions1()	Set or read the filter for stop condition 1.
setFilterStopConditions2() / getFilterStopConditions2()	Set or read the filter for stop condition 2.

Impact of user management

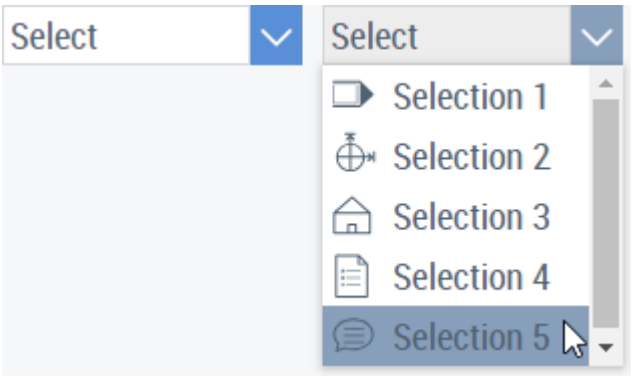
The control `TcHmiCncChannelStatus` does not exhibit any specific change in behavior as a result of the “Operate” access right.

5.5 TcHmiCncCombobox

Component: `TcHmiCncCombobox`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

The control `TcHmiCncCombobox` extends the control TcHmi.Controls.Beckhoff.TcHmiCombobox so that you can display additional icons as SVG files alongside the combo box's selection options.



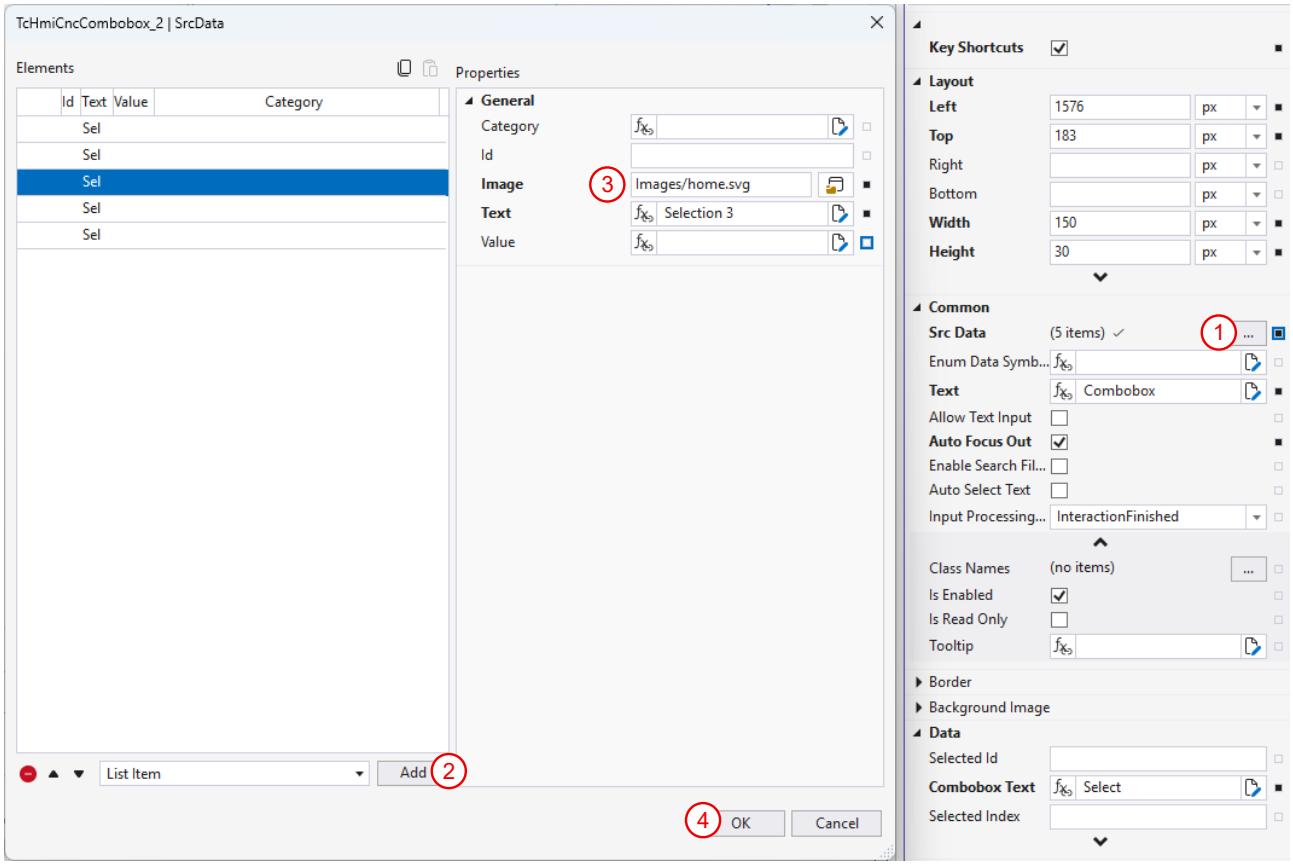
You can customize the icon's color using the CSS variable `--tchmi-icons-color`.

Operation

The operation is identical to that of the control TcHmi.Controls.Beckhoff.TcHmiCombobox.

Configuration

The configuration of the `TcHmiCncCombobox` control is identical to that of the TcHmi.Controls.Beckhoff.TcHmiCombobox control. You do not need any special data binding with the TcHmiCnc extension. You can add an icon as an SVG file using the selection options as follows.



1. Open “Src Data”.
   ⇨ An additional window appears.
2. Use the **Add** button to add an additional element.
   ⇨ Each element is displayed as an additional selection option in the combo box. The screenshot shows five selection options that have been added.
3. Under “Image,” enter the path to the SVG file.
4. Confirm your configuration with OK.

⇨ You have added an icon.

User settings via TcHmiCncSettings

The control `TcHmiCncCombobox` does not have any customizable settings in the control `TcHmiCncSettings`.

Keyboard shortcuts

You can use the following keyboard shortcuts in the control `TcHmiCncCombobox`.

Shortcut	Action
Up/Down Arrow	Navigation in the combo box
Esc	Close the combo box

API functions

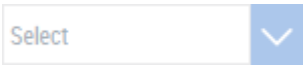
The control `TcHmiCncCombobox` has a public API that allows you to use the following functions.

Function	Description
setDataHeight() / getDataHeight()	Set or read the height of a selection option in the combo box.
setShortcuts() / getShortcuts()	Enable or disable keyboard shortcuts.

Impact of user management

The control `TcHmiCncCombobox` evaluates the “Operate” access right. For “Allow,” the control behaves as usual. When set to “Disallow,” the control is limited to the following:

- Display the most recently selected option. You cannot change the selection option.



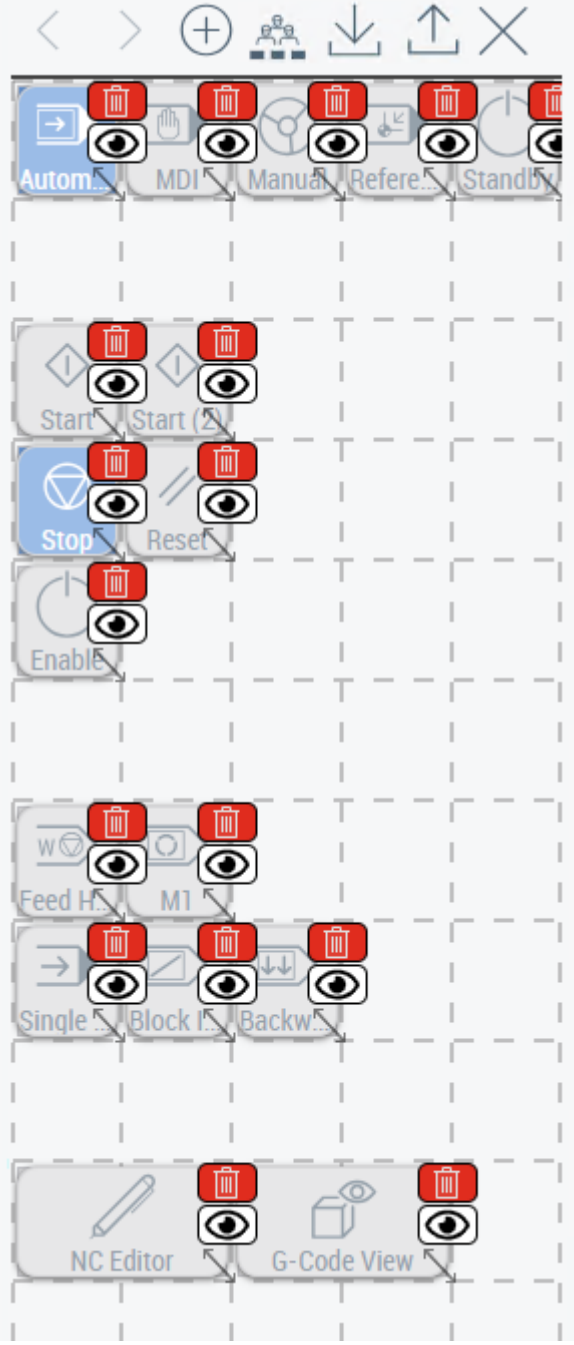
If you have not set up user management, “Operate” is considered “Allow,” and the control behaves as usual.

5.6 TcHmiCncConfigurableButton

Component: `TcHmiCncConfigurableButton`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

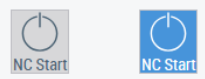
Description

The control `TcHmiCncConfigurableButton` is a button that you can configure dynamically at runtime in the HMI. In the CNC-HMI template, the button is used in the Machine Control Panel [▶ 23] (MCP). The control's full range of functions is only available when used in conjunction with the TcHmiCncLayoutManager [▶ 79] control.



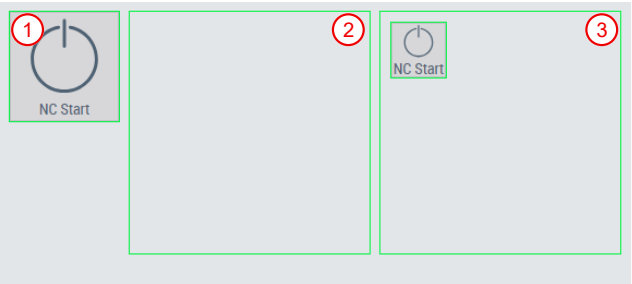
Operation

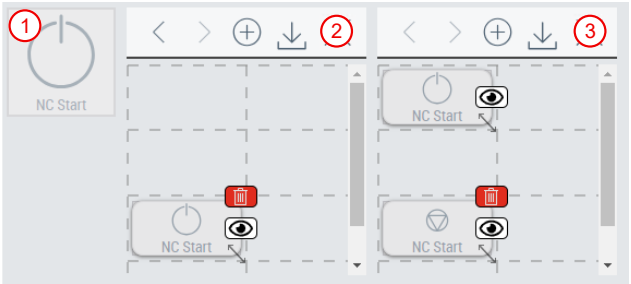
Each button has a configurable function that is triggered when you click the button.



Configuration

You can configure the button in both the HMI and TwinCAT XAE. Please note the restrictions described below regarding the creation and deletion of the button.





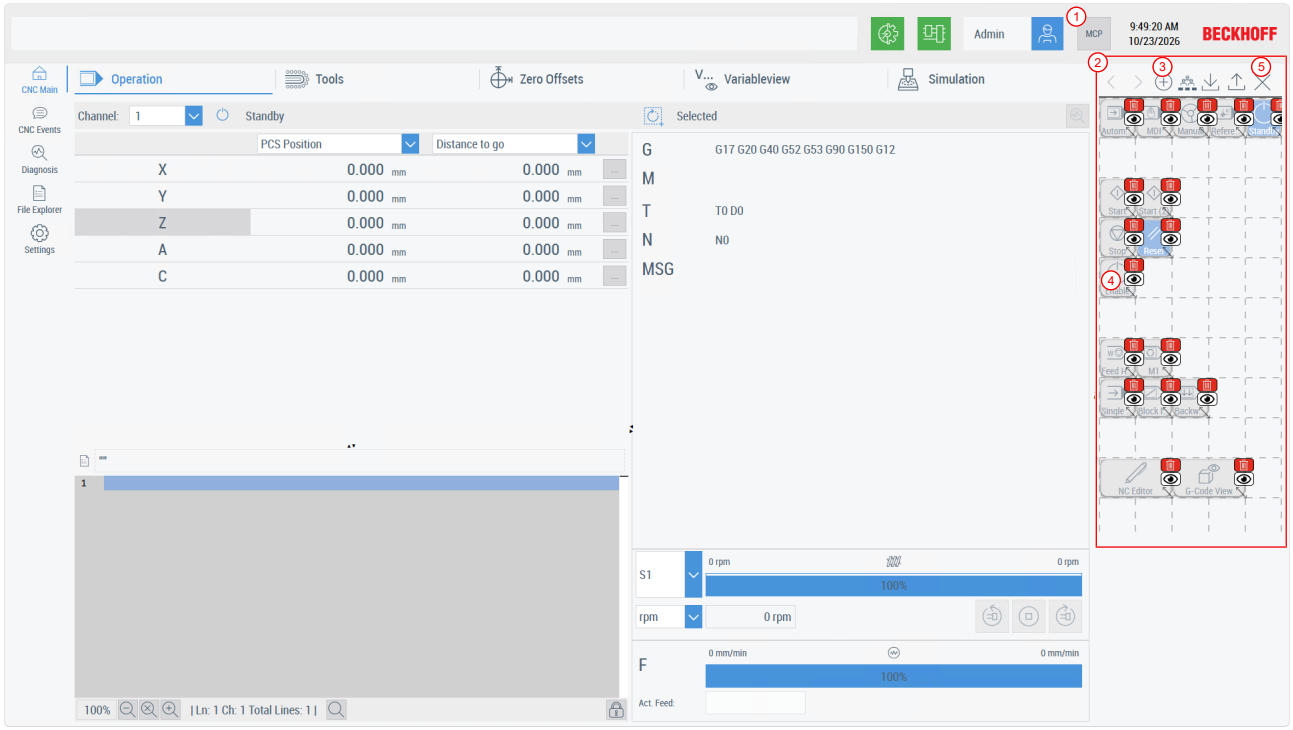
You can place the button on its own (1), place only the `TcHmiCncLayoutManager` control (2), or place the button within the `TcHmiCncLayoutManager` control (3). These three combinations give you three different ways to create and delete a button.

No.	Create a button in the HMI	Delete a button in the HMI
1	No	No
2	Yes	Yes
3	Yes	Yes: Buttons created in HMI. No: Buttons created in XAE.

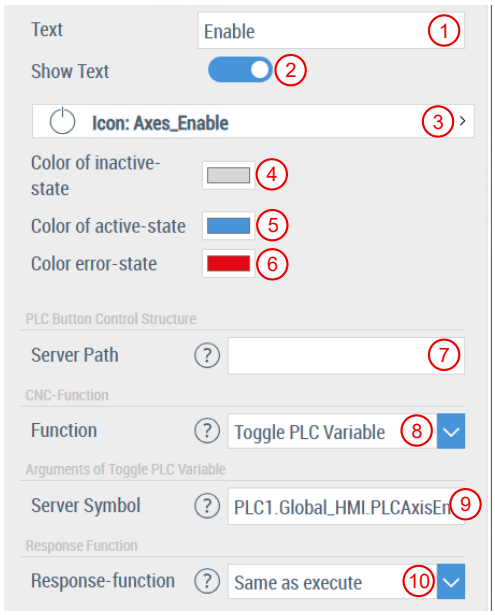
Configure the CNC-HMI Template

In the CNC-HMI template, you can access the dialog for configuring the button as follows:

1. Click the **MCP (M**achine **C**ontrol **P**anel) button.
2. You will then see the MCP layout.
3. If needed, use the **+** button to add a new button. Place the button in the desired position in the layout and adjust its size.
4. Click one of the buttons to configure it.
   ⇨ An additional dialog box appears, allowing you to configure the button.
5. Save the new layout by clicking the MCP button again or close the layout by clicking the X icon.



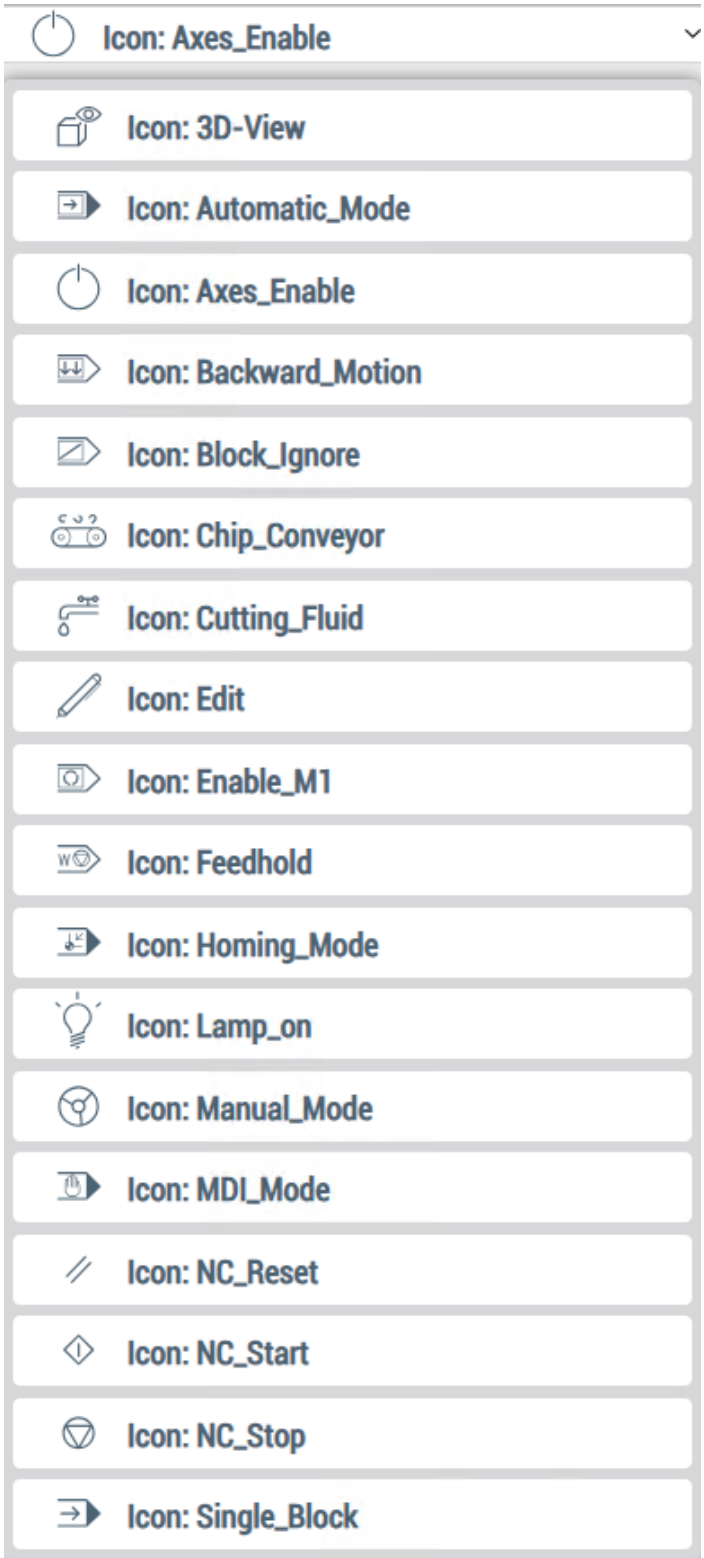
When configuring the button, the following settings are available:



No.	Description
1	Button text label.
2	Specifies whether the text label is displayed.
3	Button icon.
4	Background color of the button in the “inactive” state.
5	Background color of the button in the “active” state.
6	Background color of the button in the “Error” state.
7	Server path for controlling the button via the PLC.
8	Function that is executed when the button is clicked.
9	Context menu depending on the selected function.
10	Response Function: The return value of the Response Function controls the button’s background color through the three states: “inactive,” “active,” and “error”. The response function is available only for the following functions: • Execute CNC function • Toggle PLC variable • Set/Reset PLC variable • Dispatch custom event

Button icon

You can customize the button icon. The following icons are available:



Using the control TcHmiCncLayoutManager [▶ 79], you can upload user-specific icons as SVG files.

Server Path

Using the Server Path, you can optionally reference a variable of type “Structure” in the PLC, which is read by the button. This allows you to control the button from the PLC. Currently, it is possible to change the text label and disable the button. One example of this is selectively disabling individual buttons when the associated function cannot be performed by the actual machine (e.g., NC start when the machine door is open).

Example

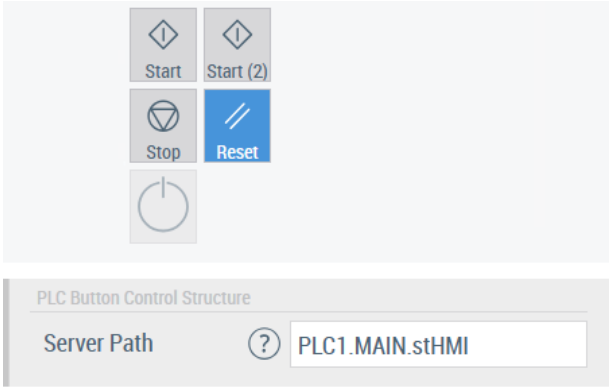
Create the following structure in the PLC.

```
TYPE ST_HMI: //Individual name possible
STRUCT
    bEnable : BOOL; //FALSE disables button. Exactly same variable name required.
    sText : STRING; //Text label of button. Exactly same variable name required.
END_STRUCT
END_TYPE
```

Next, call the structure in the PLC's "MAIN" program and assign the value `bEnable` to `false`.

```
VAR
    stHMI : ST_HMI;
END_VAR
stHMI.bEnable := FALSE;
```

In the "Enable" button, set the server path to `PLC1.MAIN.stHMI`. This button is now disabled via the structure and can no longer be used by the user.



Triggered Function

The following functions are available to you and are triggered when you click the button:

- Execute CNC function
- Toggle PLC variable
- Set/Reset PLC variable
- Open NC editor
- Open contour workbench
- Open G-code viewer
- ShowFileExplorer
- Open PDF viewer
- Dispatch custom event

Execute CNC function

This function allows you to execute the following CNC functions on a specific channel:

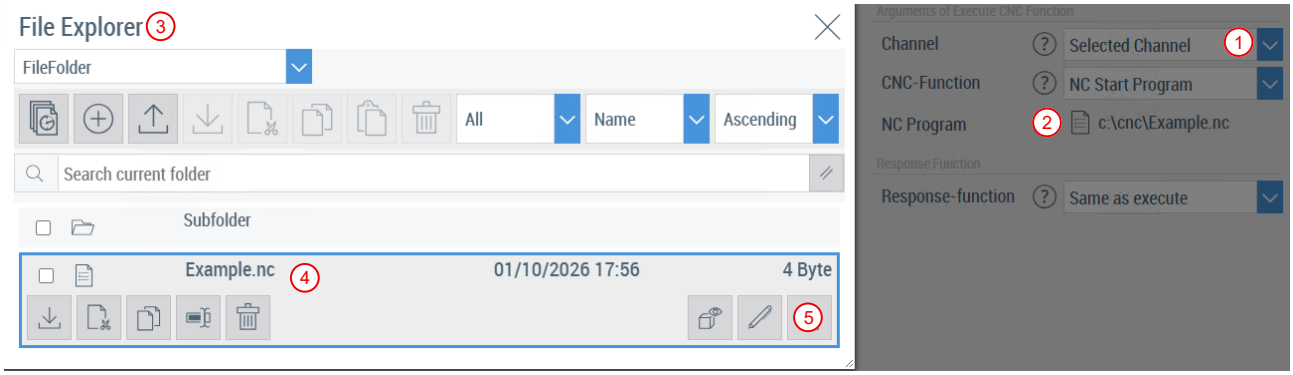
- NC Start Program
- NC Start
- NC Stop
- NC Reset
- Automatic Mode
- MDI Mode
- Manual Mode
- Reference Mode
- Standby Mode
- Block Ignore

- Feedhold
- Enable M1
- Single Block
- Backward Motion

With “NC Start Program,” you can run a predefined NC program.

1. First, use the “Channel” combo box to select the channel on which the program is to be run.
2. Next, click the icon in the “NC Program” section.
3. File Explorer opens, displaying the available NC programs.
4. Click on the NC program you want.
5. Click here to select the NC program and close the dialog box.

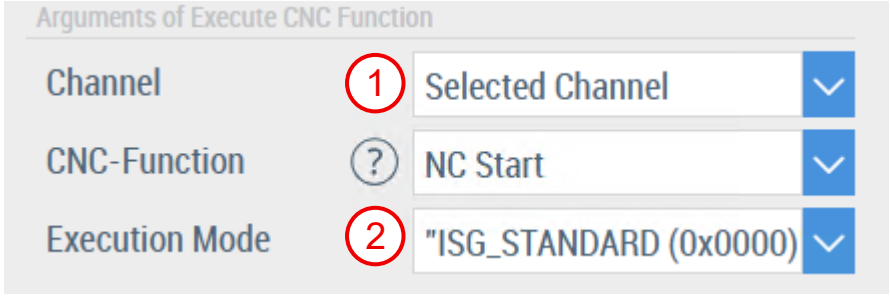
⇨ The selected NC program will run when you click the button configured in this way.



With “NC Start,” you can run the currently selected program in the channel.

1. First, use the “Channel” combo box to select the channel to which the start command should be applied.
2. Under “Execution Mode,” also select the mode in which the command should be executed. The selection options correspond to the modes described for the HLI.

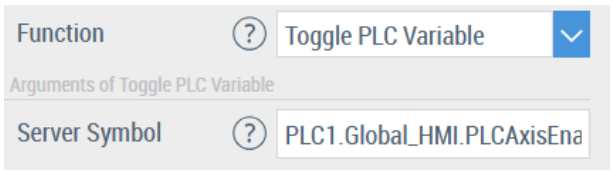
⇨ An NC program selected in the channel will start when you click the button configured in this way.



The other CNC functions are executed immediately when you click the button. You do not need any special configuration for this.

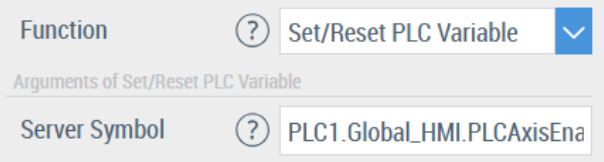
Toggle PLC variable

This function allows you to write to a BOOL-type variable in the PLC. The button acts as a toggle switch; in other words, once you click the button, the value is retained. If the button is active, the associated variable is set to `true`. In the following example, the button is linked to the variable PLC1.Global_HMI.PLCAxisEnable.



Set/Reset PLC variable

This function allows you to write to a BOOL-type variable in the PLC. The button acts as a pushbutton; that is, the variable is set to `true` only while the button is being clicked, and then reverts to `false`. In the following example, the button is linked to the variable PLC1.Global_HMI.PLCAxisEnable.

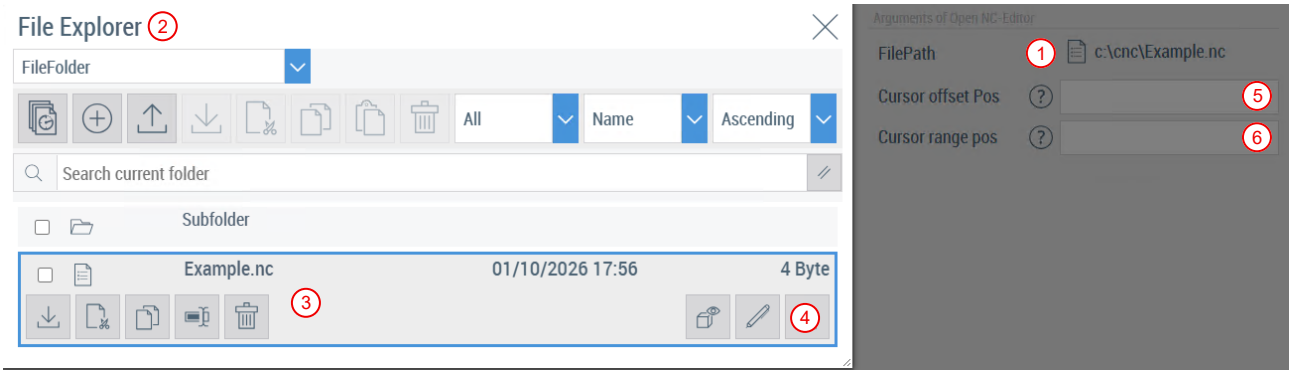


Open NC editor

This function allows you to open the NC editor [▶ 111]. Optionally, you can also define an NC program, which will open directly in the NC editor.

1. To do this, first click the icon in the “FilePath” section.
2. File Explorer opens, displaying the available NC programs.
3. Click on the NC program you want.
4. Click here to select the NC program and close the dialog box.
   ⇨ The NC program opens in the NC editor when you click the button configured in this way.
5. Optional: You can also specify the cursor position.

⇨ The area between the cursor position and the position defined in (6) is highlighted in the NC program.

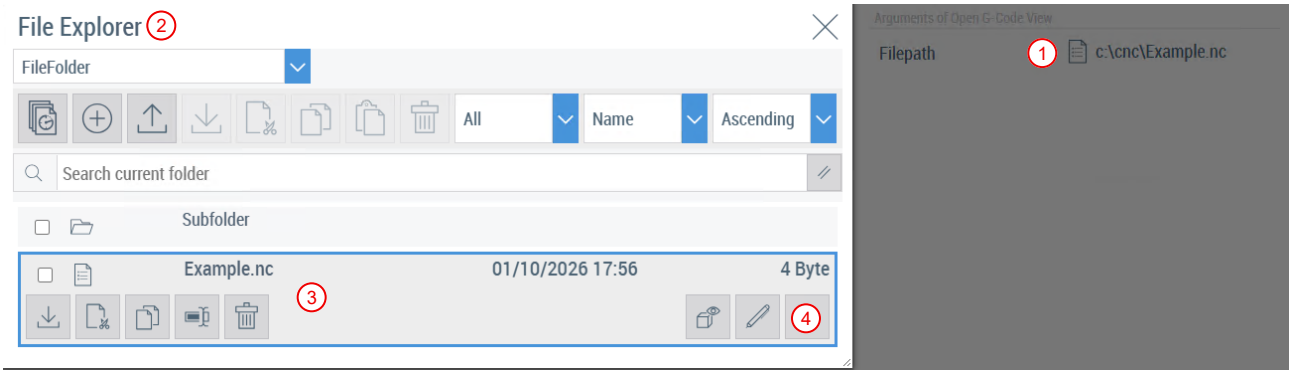


Open G-code viewer

This function allows you to open the G-Code viewer [▶ 111]. Optionally, you can define an NC program that will open directly in the G-Code viewer.

1. To do this, first click the icon in the “Filepath” section.
2. The File Explorer displaying the available NC programs opens.
3. Click on the NC program you want.
4. Click here to select the NC program and close the dialog box.

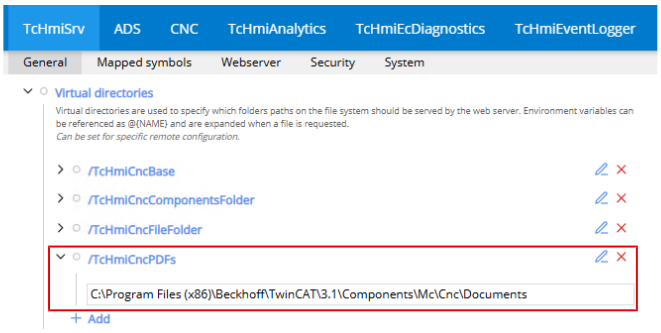
⇨ The NC program opens in the G-Code viewer when you click the button configured in this way.

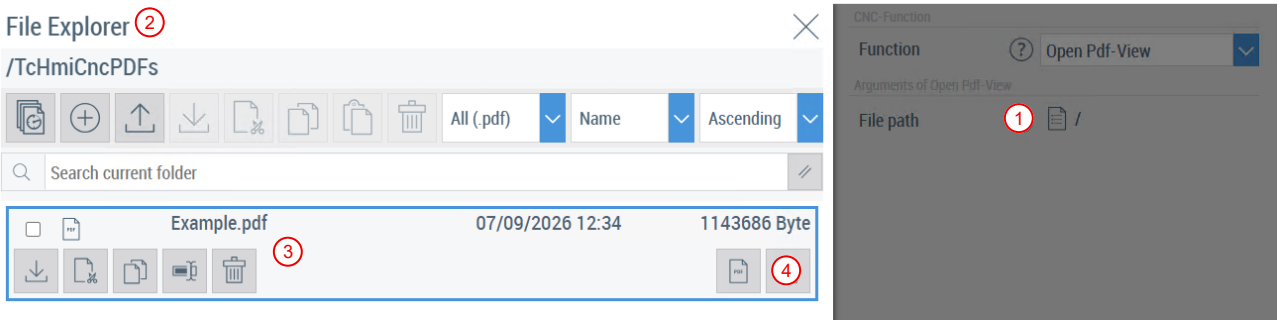


Open PDF viewer

This function allows you to open a predefined PDF file in the PDF viewer. To do this, the PDF file must be located in the “TcHmiCncPDFs” virtual directory.

1. Click the icon in the “File path” section.
2. File Explorer opens, displaying the available PDF files.
3. Click on the PDF file you want.
4. Click here to select the PDF file and close the dialog box.



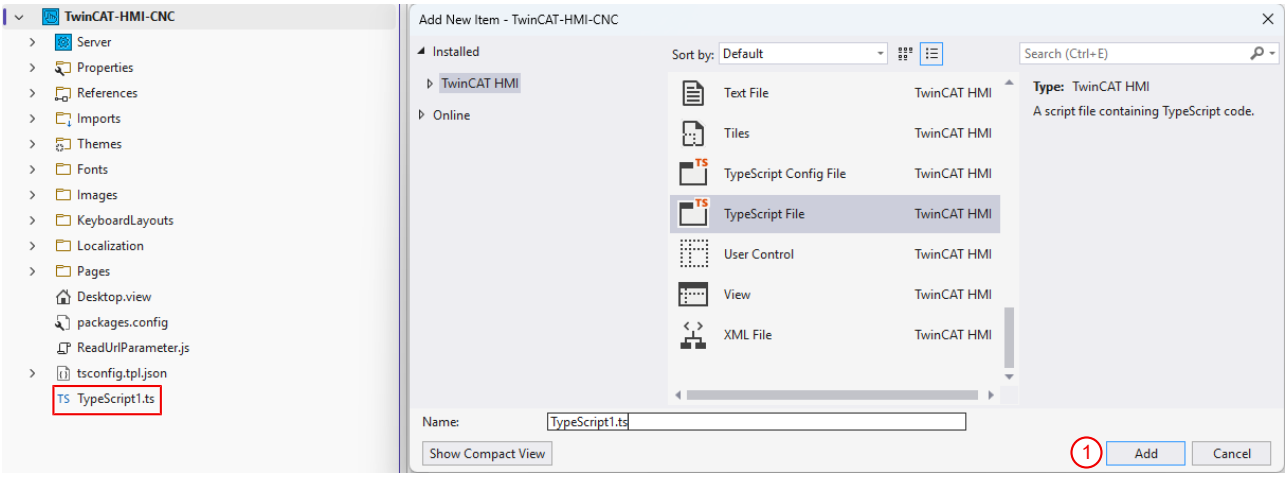


ShowFileExplorer

This function allows you to open the File Explorer [▶ 73] .

Dispatch custom event

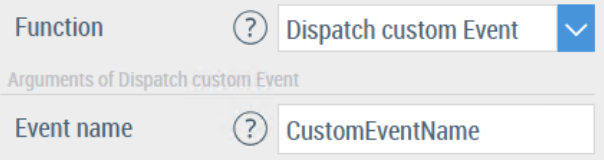
This function allows you to trigger a user-defined event that you have programmed in TypeScript. To do this, use the **Add** button (1) in your HMI project to create a new TypeScript file.



In the TypeScript file, you can then program virtually any action to be triggered when the button is clicked. In the following example, the HMI is reloaded in the browser.

```
addEventListener("CustomEventName", () => {
    //Insert below your custom event
    window.location.reload();
});
```

When configuring the button, specify the exact same event name that you used in the TypeScript file.



Response Function

The button's background color changes based on its three states: "inactive," "active," and "error". You use the return value of the response function to control which of the three background colors the button uses. However, the Response Function is only available for the following functions:

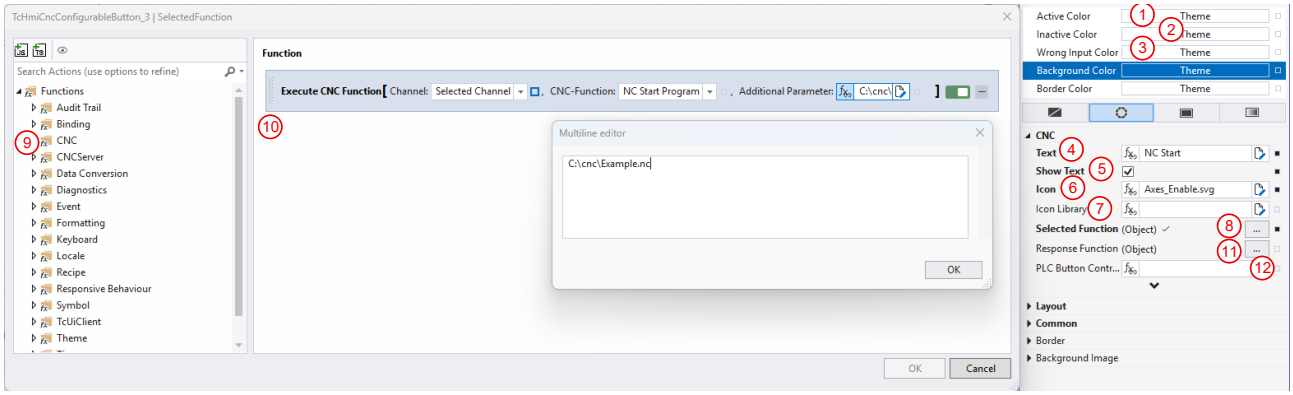
- Execute CNC function
- Toggle PLC variable
- Set/Reset PLC variable
- Dispatch custom event

The following settings are available for the response function.

Setting	Description
Same as execute	The background color is controlled by the return value of the selected function.
Execute CNC function	The background color is controlled by the "Execute CNC Function" command.
Read Server Symbol	The background color is controlled by the state of a server symbol. For example, using a variable in the PLC: PLC1.MAIN.bExample

Configuration in TwinCAT XAE

In TwinCAT XAE, you can configure the button through its properties. Configuration via TwinCAT XAE is similar to configuration in the HMI.



1. Define the background color for the "active" state.
2. Define the background color for the "inactive" state.
3. Define the background color for the "Error" state.
4. Set the text label for the button.
5. Specify whether the text label is displayed ("Show Text").
6. Specify an icon as an SVG file for the button.
   ⇨ You can change the icon later in the HMI. A user-specific icon is not automatically available for other buttons. You can make user-specific icons available for all buttons managed by the control using the TcHmiCncLayoutManager [▶ 79] control.
   ⇨ In addition, you can also make multiple user-specific icons available for a specific button.
7. To do this, specify a path to a folder containing user-specific SVG files ("Icon Library").

⇨ The SVG files must be located at the top level of the hierarchy. Subfolders are not included. If the folder contains filenames that are identical to those you use in the control `TcHmiCncLayoutManager`, the file will be used by the control `TcHmiCncConfigurableButton`.

8. Configure the function that is triggered when the button is clicked.
9. In the “CNC” folder in the left-hand pane, select the function you want.
10. Use drag and drop to move the function to the center of the screen.
    ⇨ Configure the function in the same way as you did in the HMI. In this example, the “Execute CNC Function” option was selected.
11. Define the response function.
    ⇨ The response function can only be used with the following functions:
    - Execute CNC function
    - Toggle PLC variable
    - Set/Reset PLC variable
    - Dispatch custom event
12. Optionally, specify a server path to control the button via the PLC.

User settings via TcHmiCncSettings

The control `TcHmiCncConfigurableButton` does not have any customizable settings in the control `TcHmiCncSettings`.

Keyboard shortcuts

The control `TcHmiCncConfigurableButton` does not have any specific keyboard shortcuts.

API functions

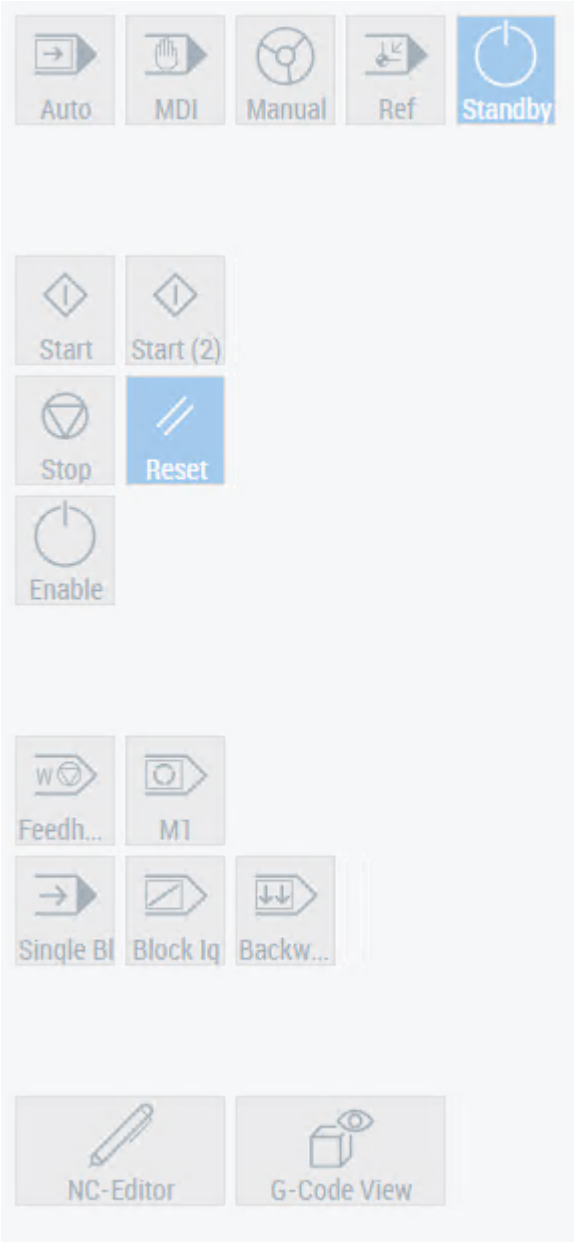
The control `TcHmiCncConfigurableButton` has a public API that allows you to use the following functions.

Function	Description
setText() / getText()	Set or read the button’s text label.
setShowText() / getShowText()	Show or hide the button’s text label.
setIcon() / getIcon()	Set or read an icon.
setIconLibrary() / getIconLibrary()	Set or read the directory for user-specific icons
setActiveColor() / getActiveColor()	Set or read the background color for the “Active” state
setInactiveColor() / getInactiveColor()	Set or read the background color for the “Inactive” state
setWrongInputColor() / getWrongInputColor()	Set or read the background color for invalid entries
setSelectedFunc() / getSelectedFunc()	Set or read the function that is executed when the button is clicked.
setResponseFunc() / getResponseFunc()	Set or read the response function
setFunctionList() / getFunctionList()	Set or Read the Selection List of Functions
setResponseFunctionList() / getResponseFunctionList()	Set or read the selection list of response functions
setPlcButtonControlStructure() / getPlcButtonControlStructure()	Set or Read the Server Path
ExecuteFunction()	Triggers the configured function without the button being actuated.

Impact of user management

The control `TcHmiCncConfigurableButton` evaluates the “Operate” access right. For “Allow,” the control behaves as usual. When set to “Disallow,” the control is limited to the following:

- Display of the button. The button itself does not work.



The control `TcHmiCncConfigurableButton` evaluates the “Editable” access right. For “Allow,” the control behaves as usual. If “Disallow” is selected, the button cannot be configured in the HMI.

With the “Editable” access right, you allow the user group to map to all PLC variables.

If you have not set up user management, “Operate” is considered “Allow,” and the control behaves as usual.

5.7 TcHmiCncContourWorkbench

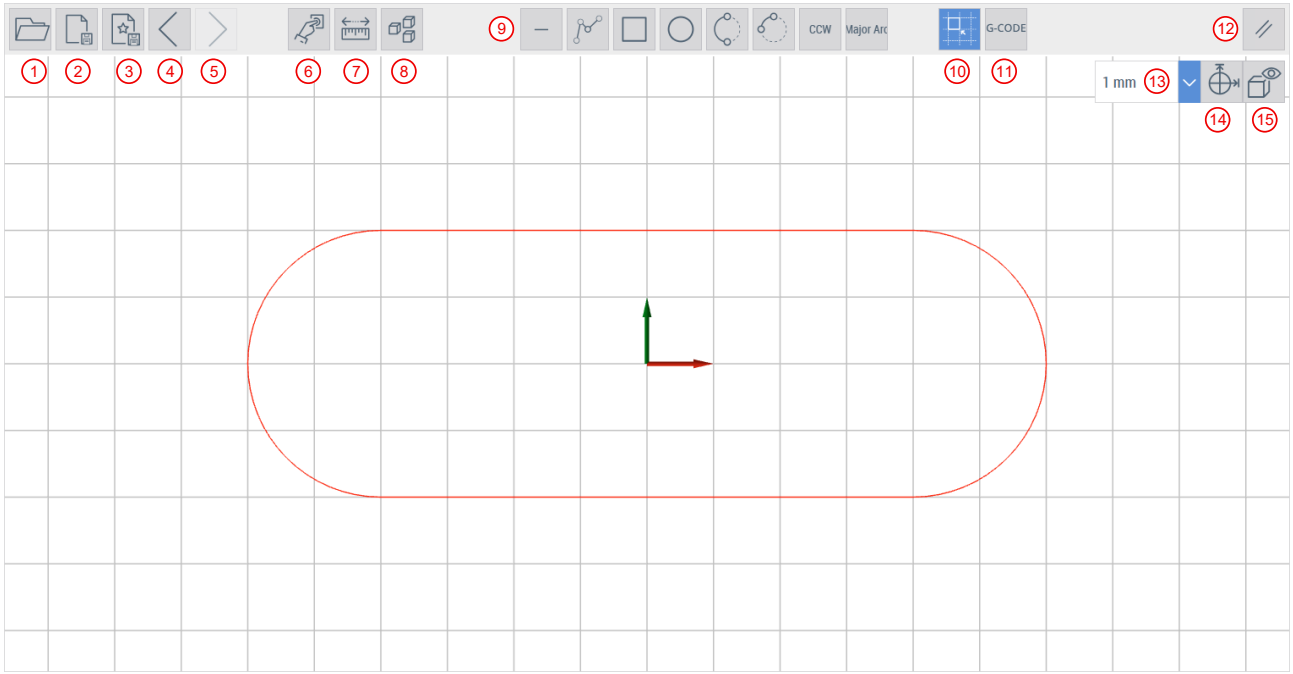
Component: `TcHmiCncContourWorkbench`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

The control `TcHmiCncContourWorkbench` is a graphical editor for creating 2D contours, which you can then export as G-Code. The control is particularly useful for generating G-Code for contour milling with the TF5293. Using the editor, you can draw lines, polygons, rectangles, circles, arcs, and semicircles. In addition, the control displays the motion path for the generated G-Code in the so-called G-Code view. The G0, G1, G2, and G3 commands are supported. You can save and edit contours in the .cnccontour and .dxf file formats.

Operation

The editor provides a menu bar with buttons that allow you to use the editor's various functions.

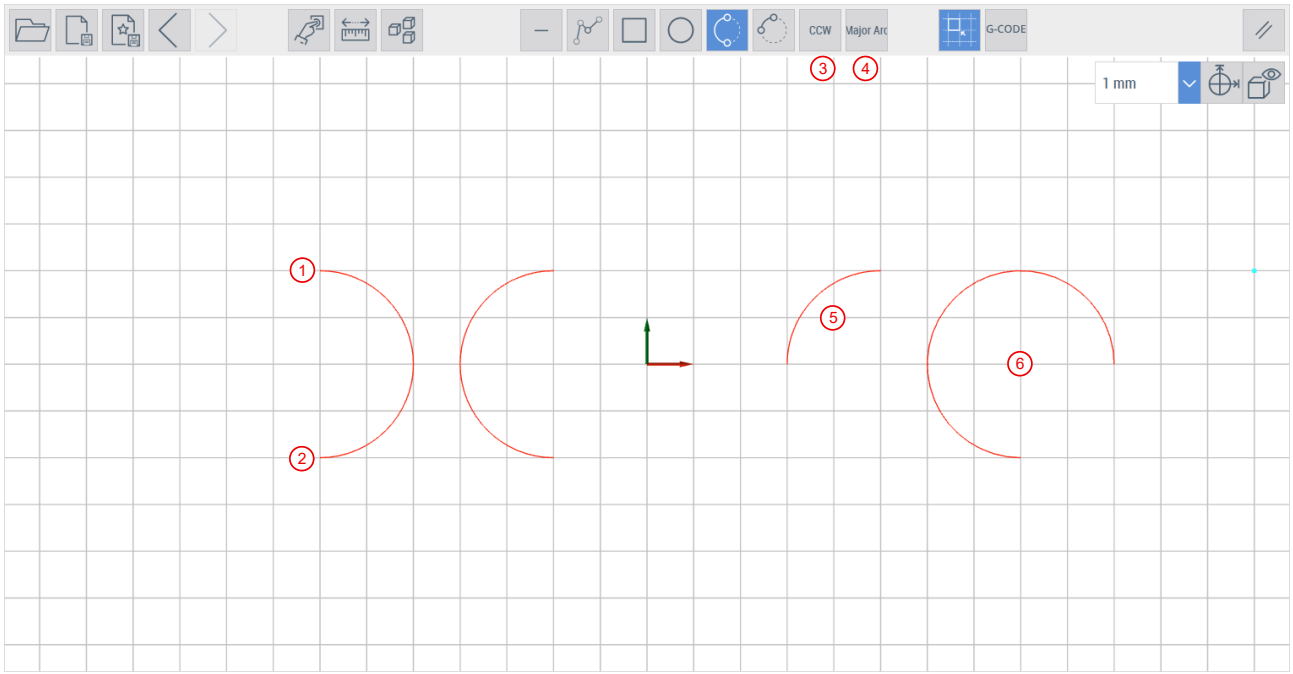


No.	Description
1	Opening an existing contour
2	Save contour
3	Create a copy of a contour
4	Undo change
5	Repeat change
6	Selecting a contour using the cursor
7	Measuring distances
8	Group contours
9	Buttons for drawing a contour
10	Enable snapping to grid intersections when measuring and drawing. The endpoints of a contour are always snapped.
11	Generate G-code
12	Reset editor
13	Adjust the spacing of the grid lines
14	Set view to coordinate origin
15	Fit the drawn contour into the window

Draw a contour

With the editor, you can draw lines, polygons, rectangles, circles, arcs, and semicircles. To do this, select the corresponding button in the menu bar and place the contour on the drawing area.

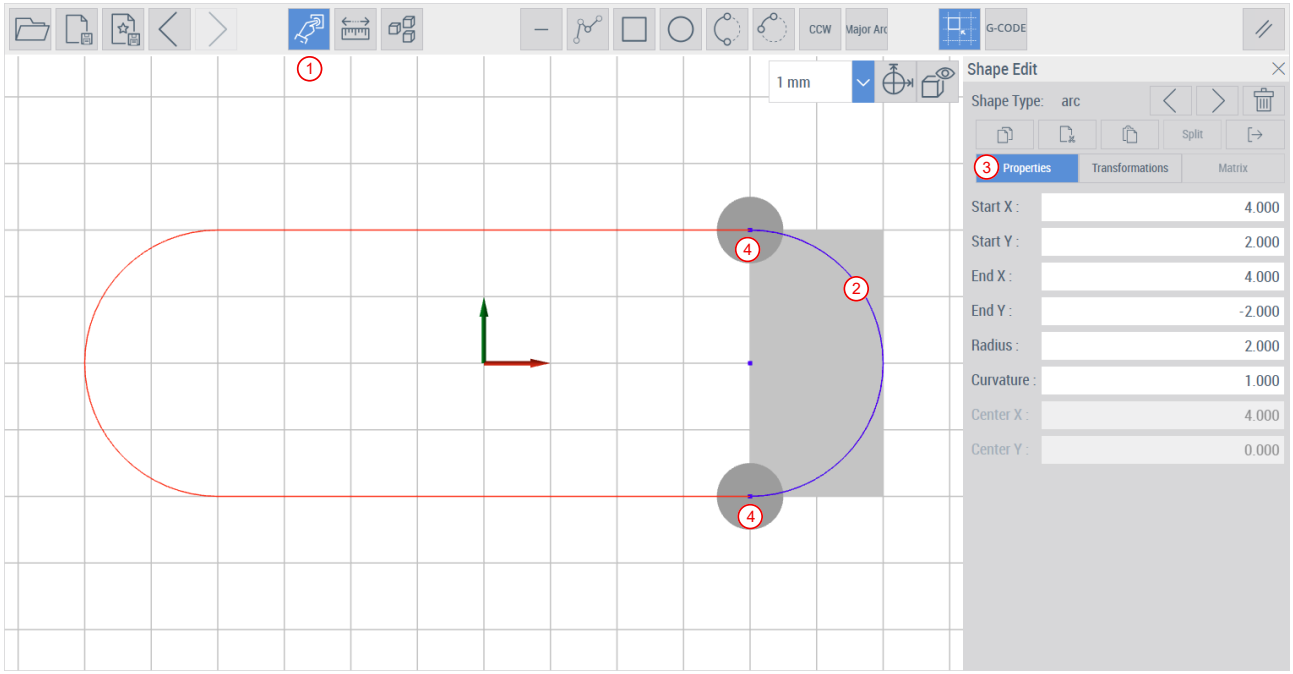
1. Place each contour using a starting point.
2. Place an endpoint.
   ⇨ From these two points, there are two possible representations for each of the semicircles and circular arcs.
3. For semicircles, use the button to specify whether the semicircle should be drawn clockwise (CW) or counterclockwise (CCW).
4. For circular arcs, specify whether the two points should be connected by a shorter (5) or longer (6) circular arc.



Adjust contour size

You can adjust the size of an existing contour.

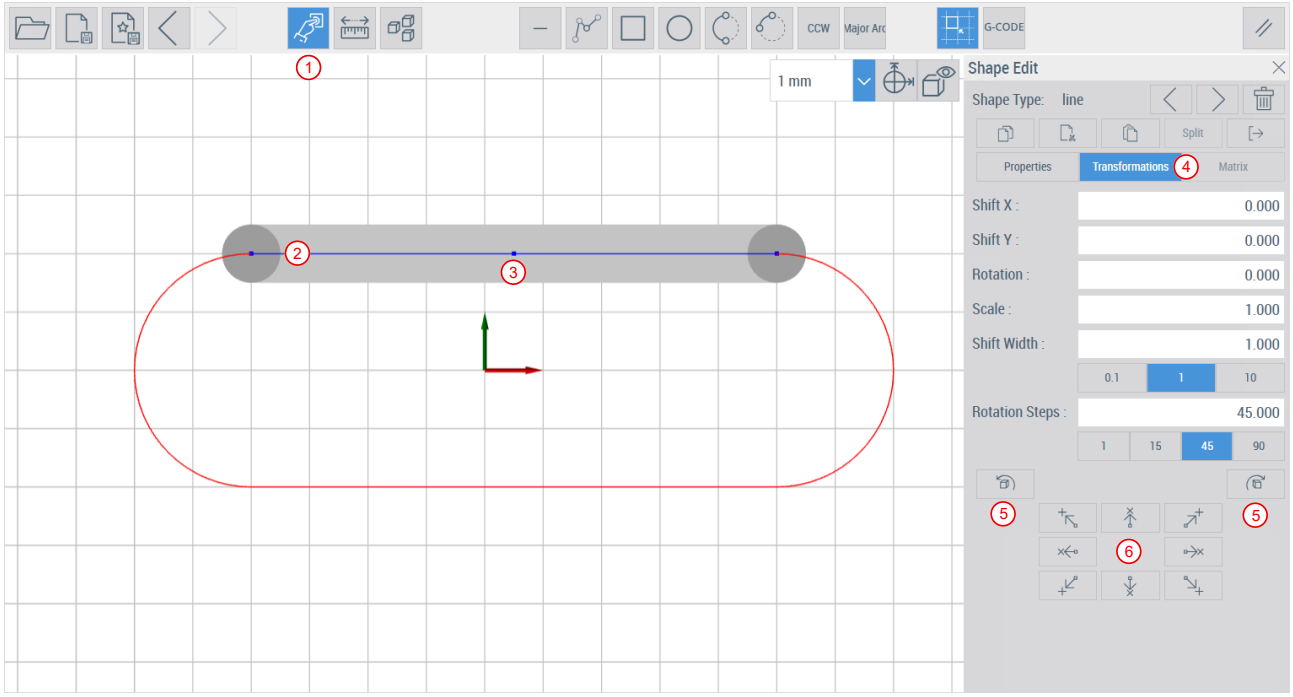
1. To do this, first click on the icon (1).
2. Next, click on the contour you want to change.
   ⇨ An additional window appears showing the contour parameters. To change the size of the contour, follow these steps:
3. Click “Properties” and adjust the contour using your parameters.
4. Alternatively: Move the start or end point of the contour. To do this, click one of the ends of the contour and move the point.



Move contour

You can move a contour.

1. To do this, first click on the icon (1).
2. Next, click on the shape you want to move.
3. Drag the center point of the contour to a new position.
4. Alternatively: Move the contour using the “Transformations” context menu.
5. Rotate a contour or
6. Move a contour using the arrows.

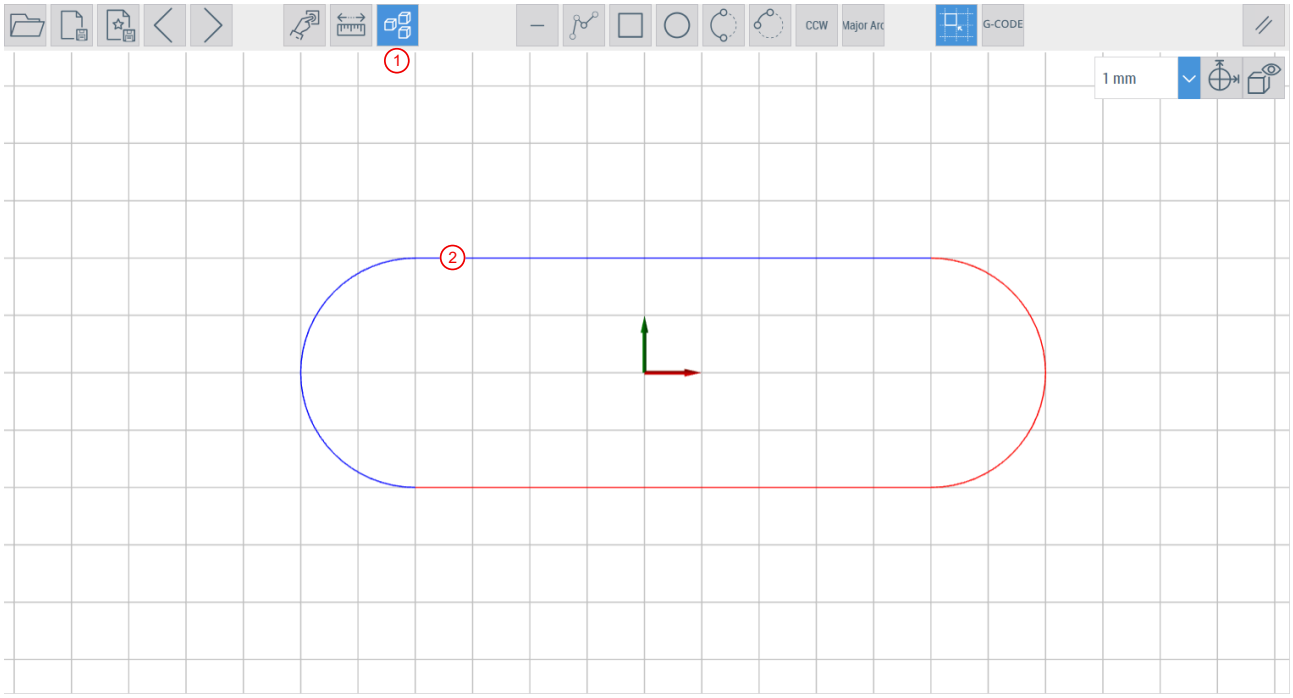


Group and duplicate a shape

You can combine multiple contours into a group.

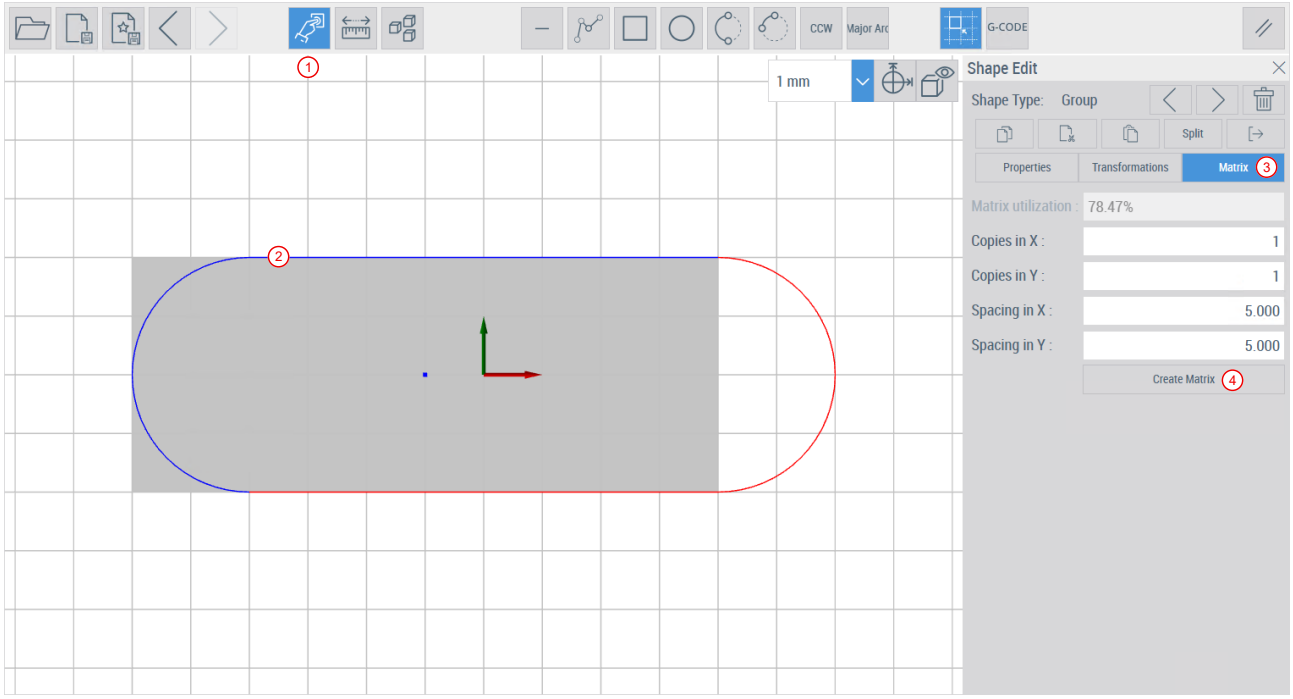
1. Click the icon (1)

2. Next, click on the shapes you want to combine into a group. The selected contours are highlighted in blue.
3. Click (1) again to confirm the grouping.



You can duplicate a group.

1. To do this, click the icon (1).
2. Next, click on the group you want to duplicate.
3. Click “Matrix” and configure the matrix you want to use to duplicate the group.
4. Click “Create Matrix” to create the matrix.



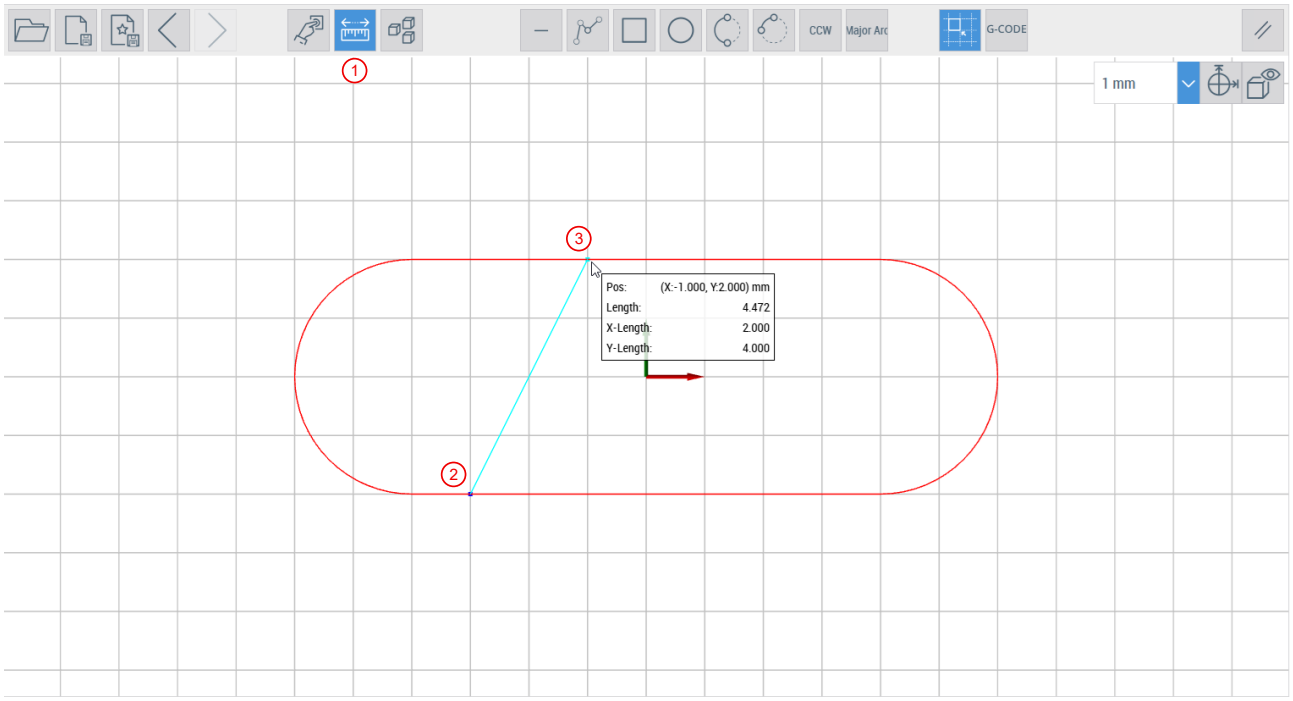
Measuring distances

You can measure the distance between two points.

5. Click the icon (1).
6. Click the starting point from which you want to measure.

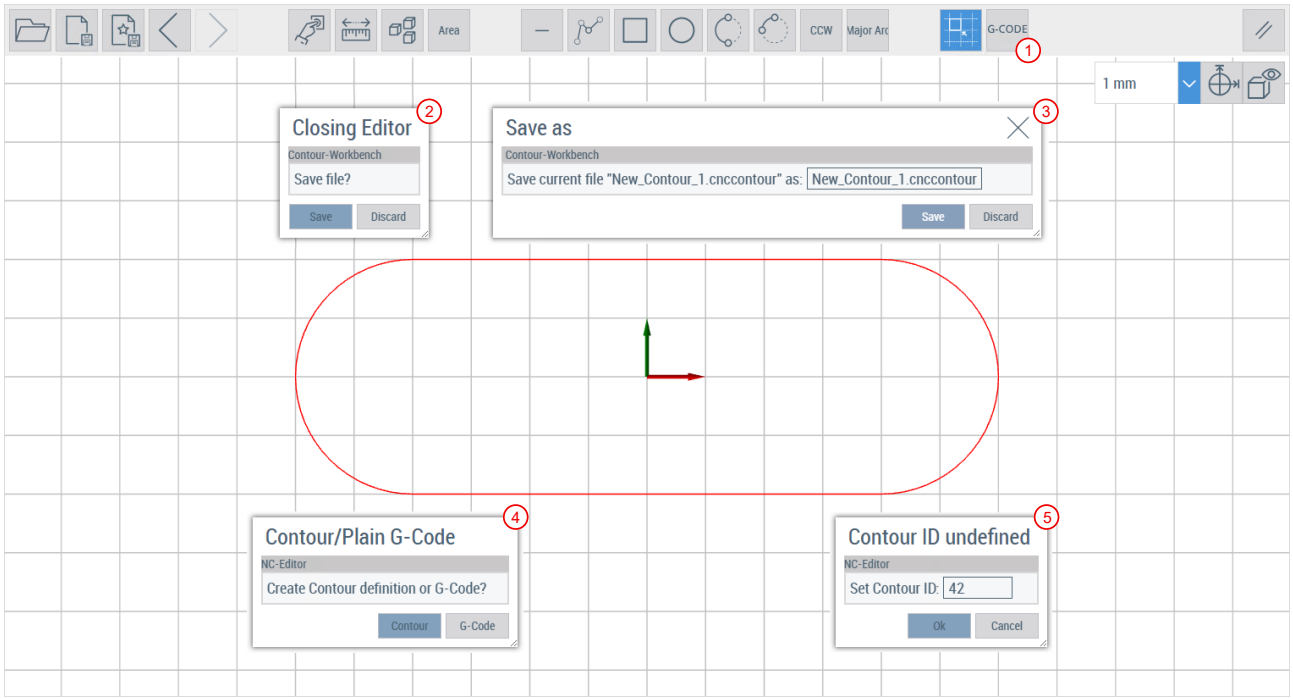
7. Click on the endpoint you want to measure to.

⇨ The result will appear next to your cursor.



Generate G-code

You can automatically generate the G-Code for your contour.



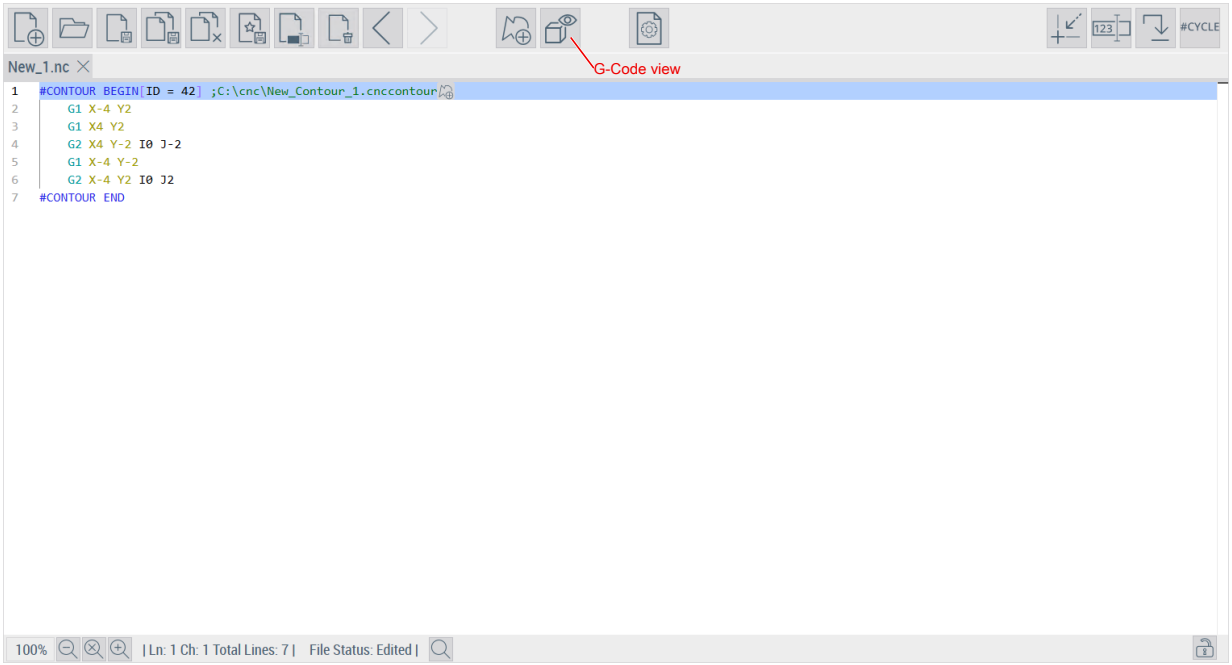
1. Click the icon (1).

   ⇨ A popup window opens.
2. Optional: Save the contour.
3. Optional: Enter a filename for the contour.

   ⇨ Another pop-up window appears.
4. Optional: Decide whether you want to include a #CONTOUR DEFINITION in the G-Code.

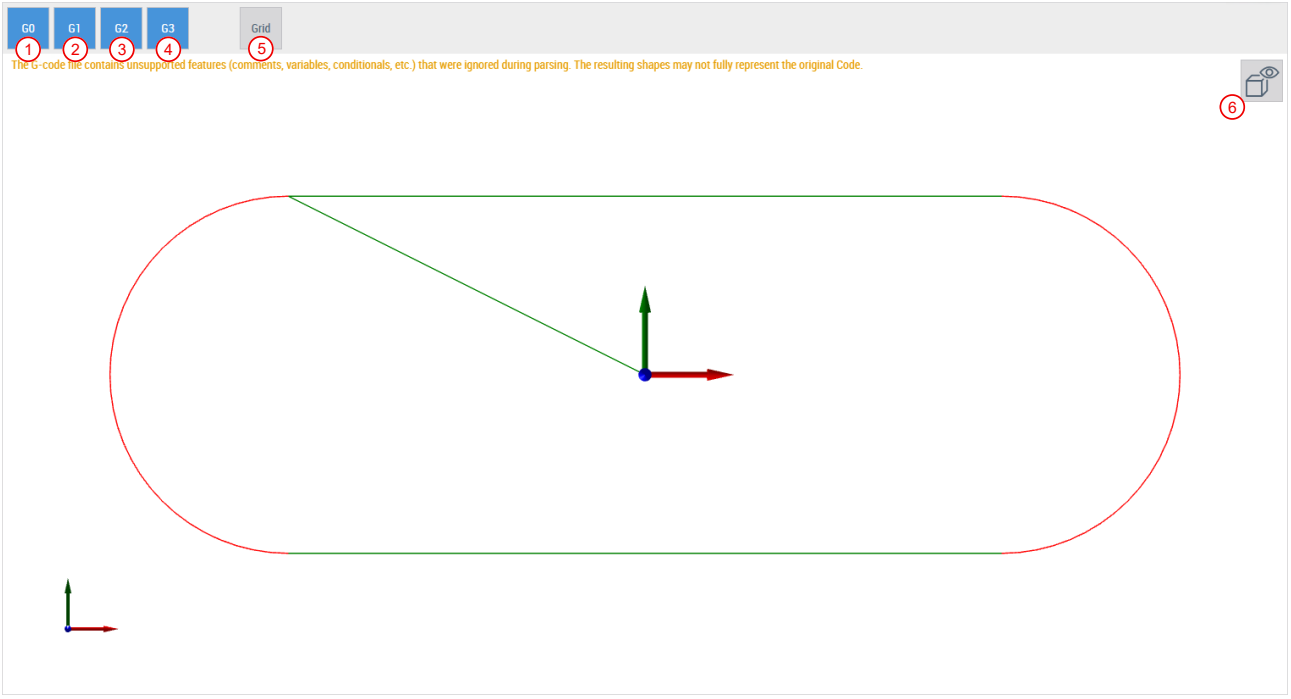
   ⇨ Another pop-up window appears.
5. Assign an ID to #CONTOUR DEFINITION

⇨ The generated G-Code is displayed in the NC editor [▶ 111].



G-Code view

You can view the resulting toolpath generated from the G-Code in the G-Code view. The G-Code view is part of the `TcHmiCncContourWorkbench` control. Open the G-Code view using the corresponding button in the NC Editor [▶ 111]. You can also open the G-Code view directly using the control TcHmiCncConfigurableButton [▶ 47].



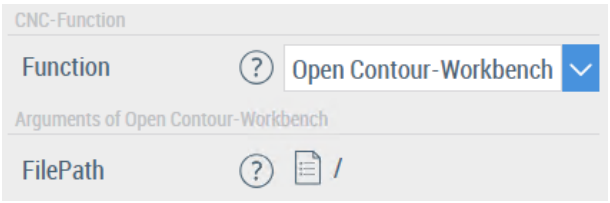
No.	Description
1	Filter by G0 commands
2	Filter by G1 commands
3	Filter by G2 commands
4	Filter by G3 commands
5	Turn gridlines on or off
6	Fit the displayed motion path into the window

Configuration

The control `TcHmiCncContourWorkbench` does not require any special data binding with the TcHmiCnc extension.

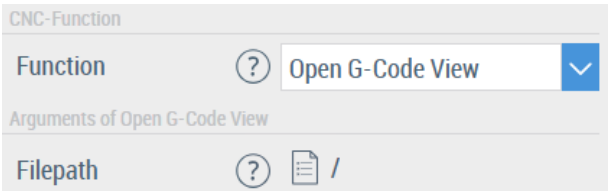
Opening the Editor Using TcHmiCncConfigurableButton

You can open the editor directly using the control TcHmiCncConfigurableButton [▶ 47]. To do this, create a button and configure “Open Contour-Workbench” as the function to be executed by the button. Here, you can specify a file to open. If “FilePath” is left blank, an empty editor will open, where you can create a new contour right away.



Opening the G-Code view using TcHmiCncConfigurableButton

You can open the G-Code view directly using the control TcHmiCncConfigurableButton [▶ 47]. To do this, create a button and configure “Open G-Code” as the function to be executed by the button. Here, you can specify a file to open. If “FilePath” is left blank, the NC program you have selected in the CNC channel will open.



Custom G-Code generator

You can create a custom G-Code generator and use it in the control `TcHmiCncContourWorkbench`. To do this, create a TypeScript file in your TwinCAT project and define a new class for your generator within it. In the following example, the class GCodeGenerator_1 is created, which uses the user-defined function `Shapes_to_GCode` to convert the contours to G-Code.

```
namespace TcHmi {
    // If you want to unregister an event outside the event code you need to use the return value of
the method register()
    let destroyOnInitialized = TcHmi.EventProvider.register('onInitialized', (e, data) => {
        // This event will be raised only once, so we can free resources.
        // It's best practice to use destroy function of the event object within the callback
function to avoid conflicts.
        e.destroy();
        // ----------------------
        // Place your code here!
        // ----------------------
        (TcHmi.Functions.TcHmiCncBase as any).SettingsWrite(
            {
                error: (data) => { console.log(data)},
                success: (data) => { console.log(data) }
            },
            {
                section: "TcCncHmi.Contour Editor",
                ident: "GCodeGenerator",
                name: "G-Code Generator",
                type: "selection",
                defaultValue: "Default",
                value: "Default",
                selectionItems: ["Default",
"TcHmi.Functions.TcHmiCncBase.externalModules.GCodeGenerator_1"],
                description: "When generating G-Code by the Contour-Editor, this setting determines
whether to use the default generator or a custom one."
            }
        );
    });
}
type Point = { X: number, Y: number }; //in TcGL scale 1000 = 1mm
```

```
type LineShape = { type: 'line', start: Point, end: Point };
type RectangleShape = { type: 'rectangle', start: Point, end: Point, angle: number };
type CircleShape = { type: 'circle', center: Point, radius: number, StartEndPos?: Point };
type ArcShape = { type: 'arc'; start: Point; end: Point; bulge: number; center: Point; radius:
number; startAngle: number; endAngle: number; };
type PolylineShape = { type: 'polyline', points: Point[] };
type Shape = LineShape | RectangleShape | CircleShape | ArcShape | PolylineShape;

interface IGCodeGenerator {
    /**
     * Converts an array of Shape objects into a G-Code string.
     *
     * @param shapesRaw - an array of Shape instances to convert
     * @returns the resulting G-Code as a string
     */
    Shapes_to_GCode(shapesRaw: Shape[]): string;
}

namespace TcHmi {
    export namespace Functions {
        export namespace TcHmiCncBase {
            export namespace externalModules {
                export class GCodeGenerator_1 implements IGCodeGenerator {
                    constructor() { }

                    public Shapes_to_GCode(ShapesRaw: Shape[]): string {
                        console.log("testing, no implementation done");
                        return "testing, no implementation done";
                    }
                }
            }
        }
    }
}
```

User settings via TcHmiCncSettings

You can adjust the following settings via the control `TcHmiCncSettings`:

Section	Setting	Standard	Description
TcCncHmi.Contour Workbench	G-code generator	—	TypeScript class name of the custom G-Code generator (empty = predefined generator)
TcCncHmi	Decimal places	3	Decimal places for measurement results and coordinate display

Keyboard shortcuts

You can use the following keyboard shortcuts in the control `TcHmiCncContourWorkbench`:

Menu Bar

Shortcut	Action
`Alt+O`	Open file.
`Ctrl+S`	Save file.
`Ctrl+Alt+S`	Save as
`Ctrl+Z`	Undo
`Ctrl+Y`	Redo

After selecting the contour

Shortcut	Action
`Ctrl+D`	Delete selected contour (with confirmation dialog).
`Ctrl+C`	Copy contour.
`Ctrl+X`	Cut out the contour.
`Ctrl+V`	Insert a contour.
`Ctrl+E`	Export the contour.
`Ctrl+←`	Select the previous contour.
`Ctrl+→`	Select the next contour.

API functions

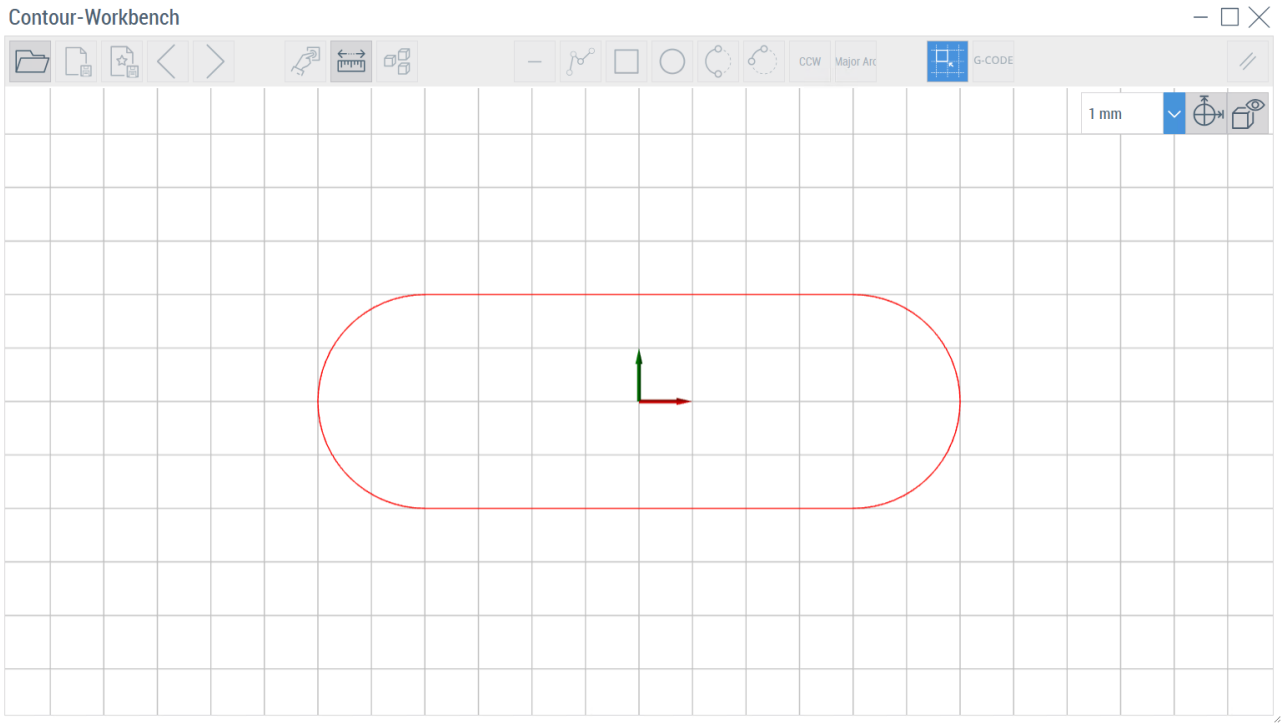
The control `TcHmiCncContourWorkbench` has a public API that allows you to use the following functions.

Function	Description
loadFile()	Loads a contour from a file path and/or a G-Code string.
setGCodeGenerator() / getGCodeGenerator()	Set or read the name of the user-specific G-Code generator class.
setGCodeView() / getGCodeView()	Enable or disable the G-Code view.

Impact of user management

The control `TcHmiCncContourWorkbench` evaluates the “Operate” access right. For “Allow,” the control behaves as usual. When set to “Disallow,” the control is limited to the following:

- Opening an existing contour.
- Measuring distances.
- Changing the view (e.g., the spacing of the gridlines).



If you have not set up user management, “Operate” is considered “Allow,” and the control behaves as usual.

5.8 TcHmiCncDetachableTabNavigation

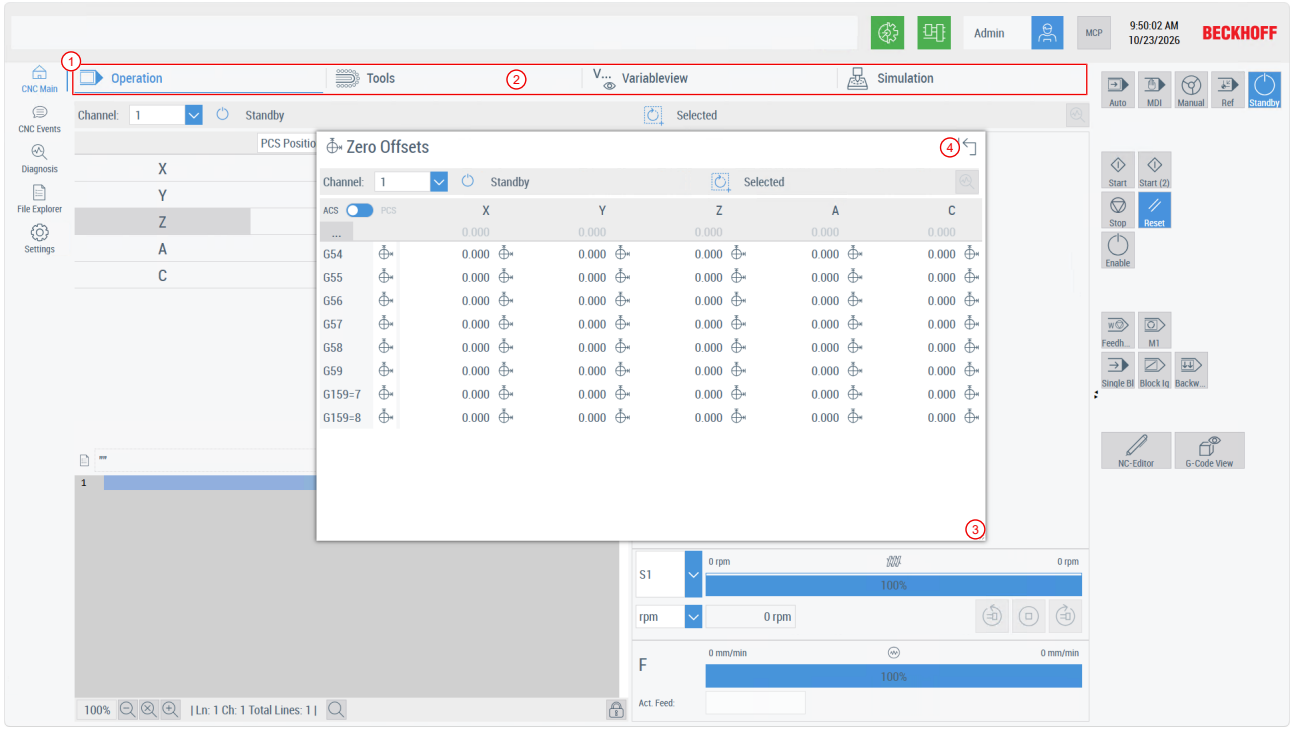
Component: `TcHmiCncDetachableTabNavigation`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

The control `TcHmiCncDetachableTabNavigation` is a navigation bar with tabs that you can switch between. You can detach each tab from the navigation bar and place it wherever you like. You can also change the order of the tabs in the navigation bar.

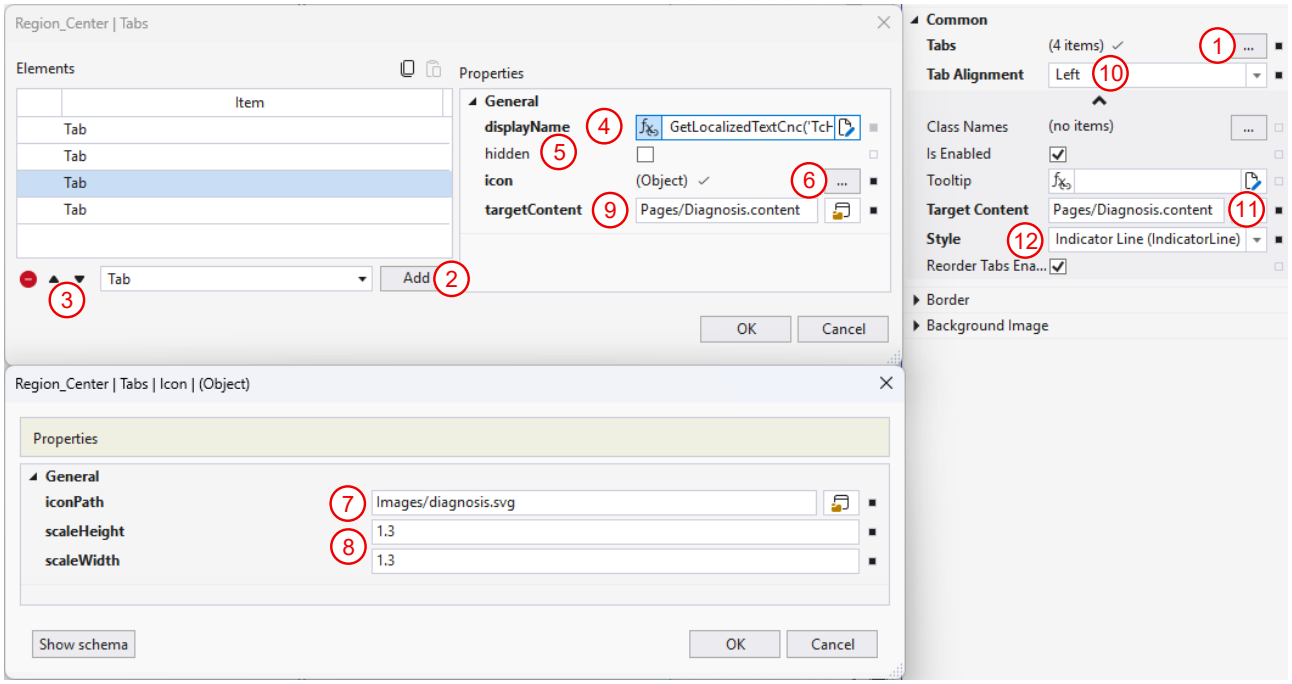
Operation

1. Click one of the tabs in the navigation bar to switch between tabs.
2. Drag a tab to an area outside the navigation bar.
   ⇨ In the image, the “Zero Offsets” tab has been detached from the navigation bar. The pop-up window can be moved freely within the HMI.
3. If necessary, resize the pop-up window.
4. You can use the icon (back arrow) to move the tab back into the navigation bar.



Configuration

You can configure the control `TcHmiCncDetachableTabNavigation` as follows. You do not need any special data binding with the TcHmiCnc extension.



1. Click on the button "**…**".
   ⇨ An additional window titled "Region_Center | Tabs" opens.
2. Click the **Add** button to create a new tab.
3. Use the up and down arrows to adjust the order of the tabs in the navigation bar.
4. Set a display name for each tab.
5. The tab is hidden when you set the "hidden" checkbox to `true`.
6. You can select an icon as an SVG file to be displayed in the tab. Click on the button "…".
   ⇨ An additional window titled "Region_Center | Tabs | Icon" opens.
7. Specify the path to the SVG file.
8. Define the dimensions of the icon.
9. Use "targetContent" to specify a content page to be displayed in the tab.
   ⇨ If there is no content page, the tab remains hidden.
10. Use "Tab Alignment" to specify on which side of the control the tabs should be arranged.
11. If you use a content page under "Target Content" that you have also assigned to one of the tabs, that content page will be displayed by default.
12. Use "Style" to set the display style of the navigation bar.

User settings via TcHmiCncSettings

The control `TcHmiCncDetachableTabNavigation` does not have any customizable settings in the control `TcHmiCncSettings`.

Keyboard shortcuts

The control `TcHmiCncDetachableTabNavigation` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncDetachableTabNavigation` has a public API that allows you to use the following functions:

Function	Description
setTabs() / getTabs()	Setting or reading a tab.
setTabAlignment() / getTabAlignment()	Set or read which side of the control the tabs are located on.
setTargetContent() / getTargetContent()	Set or read the path to the content page.
setStyle() / getStyle()	Set or read the display style.

Impact of user management

The control `TcHmiCncDetachableTabNavigation` evaluates the “Operate” access right. For “Allow,” the control behaves as usual. When set to “Disallow,” the control is limited to the following:

- Switch between different tabs. It is not possible to undock the tab from the navigation bar.
- Tabs that have already been undocked can still be moved and reintegrated into the navigation bar.

If you have not set up user management, “Operate” is considered “Allow,” and the control behaves as usual.

5.9 TcHmiCncDiagData

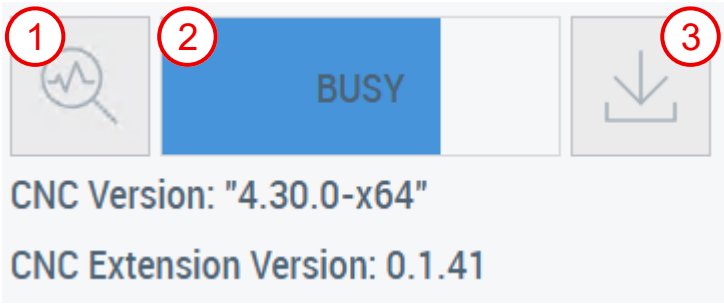
Component: TcHmiCncDiagData
Namespace: TcHmi.Controls.TcHmiCncBase
Available since version: 1.0.0

Description

Using the control TcHmiCncDiagData, you can trigger the creation of a diagnostic file for the CNC and then download the file.

Operation

1. Click the button to start creating a diagnostic file.
   ⇨ The diagnostic file is created in the background, so you can continue to use the HMI at the same time.
2. A progress bar shows you the current status.
   ⇨ During this process, the control cycles through the “IDLE,” “BUSY,” and “DONE” states, as well as “ERROR” if the diagnostic file cannot be created. In the state “DONE,” the diagnostic file is ready for download.
3. Then download the diagnostic file as diag_data.txt.



Configuration

The control TcHmiCncDiagData does not require any special data binding with the TcHmiCnc extension.

User settings via TcHmiCncSettings

The control TcHmiCncDiagData does not have any customizable settings in the control TcHmiCncSettings.

Keyboard shortcuts

The control TcHmiCncDiagData does not have any specific keyboard shortcuts.

API functions

The control TcHmiCncDiagData has a public API that allows you to use the following function:

Function	Description
btDownLoadPressed()	Triggers the download of the diagnostic file.

Impact of user management

The control TcHmiCncDiagData evaluates the “Operate” access right. For “Allow,” the control behaves as usual. When set to “Disallow,” the control is limited to the following:

- Simply displays the control. You cannot create or download a diagnostic file.

If you have not set up user management, “Operate” is considered “Allow,” and the control behaves as usual.

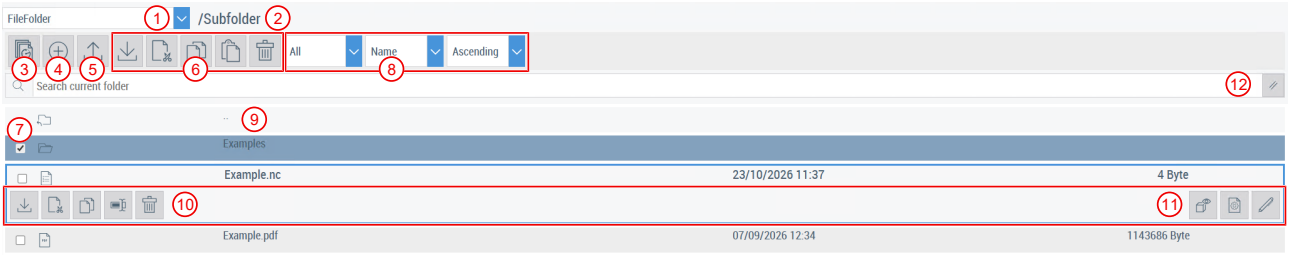
5.10 TcHmiCncFileExplorer

Component: `TcHmiCncFileExplorer`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

The control `TcHmiCncFileExplorer` is a file browser that allows you to access the CNC controller's file system. File Explorer only has access to the virtual directories that you have defined. You can also use a .hmiignore file to specifically hide individual files and folders.

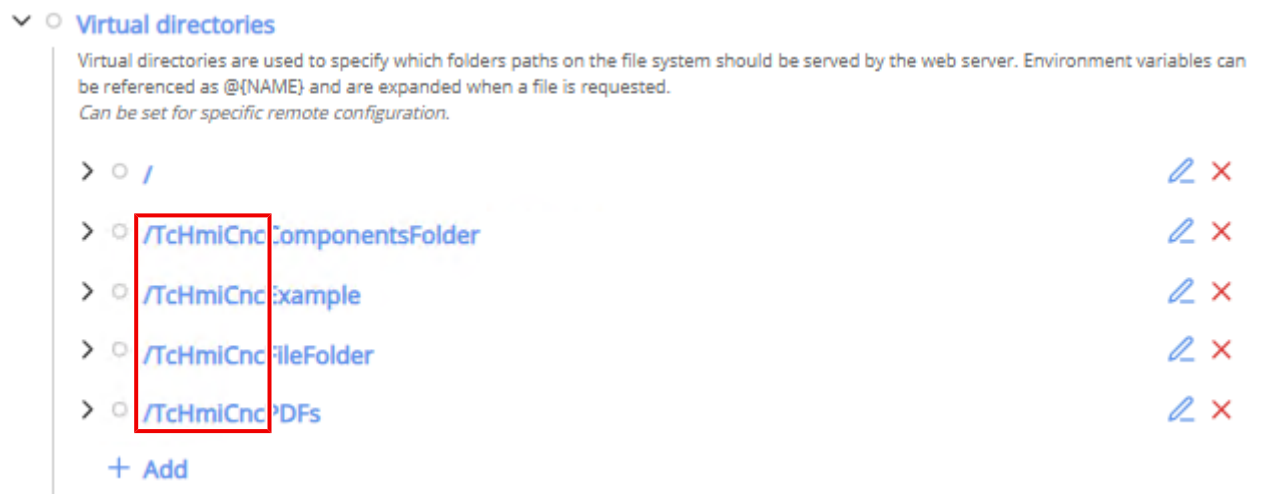
Operation



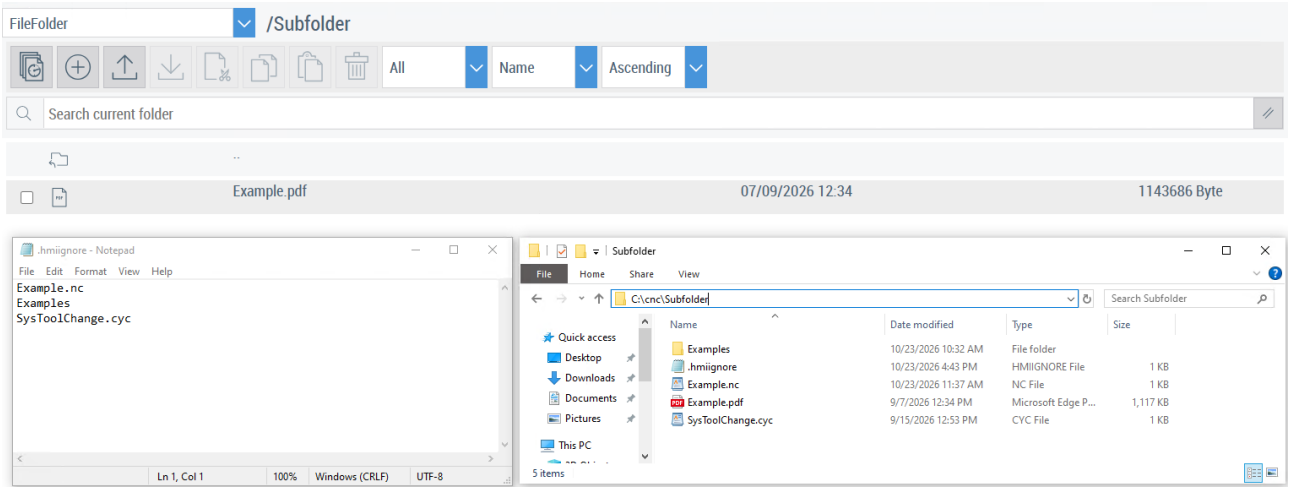
No.	Description
1	A combo box for selecting the virtual directory that is displayed in File Explorer.
2	Current folder path in the virtual directory
3	Button for displaying recently used files
4	Button to add a folder
5	Button to add a file from your local file system
6	Buttons for file operations. The buttons become active as soon as you select at least one file or folder.
7	Selecting a file or folder.
8	Combo boxes for filtering files and folders in the virtual directory.
9	Double-clicking this button will take you back to the parent folder.
10	Buttons for performing file operations on a specific file. The buttons appear after you click on the file.
11	Buttons for opening the file directly in the appropriate editor or viewer.
12	Search function in the current folder

Configuration

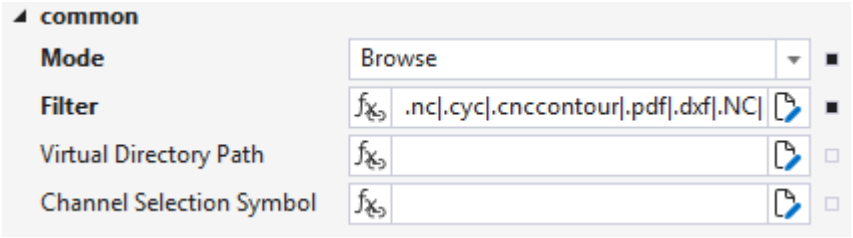
The control `TcHmiCncFileExplorer` allows you to access virtual directories that you created using TE2000. You can access virtual directories whose names begin with "TcHmiCnc".



For virtual directories, the TcHmiCNC extension evaluates an existing .hmiignore file. The files and folders listed in the file are not visible to the user. The .hmiignore file applies only to the folder in which it is located. In the following example, only the Example.pdf file is still visible in File Explorer. The other files and folders were hidden by the .hmiignore file.



You can configure the control `TcHmiCncFileExplorer` as follows.



Mode

This setting allows you to specify the default action that occurs when you double-click a file. The following modes are available:

Mode	Description
Browse	Open the file in the appropriate editor or viewer.
SelectProgram	Selecting the NC program. This action is available only for files with the .nc file extension.

Filter

This setting lets you specify which file extensions are displayed in File Explorer. Separate the file extensions with a vertical bar. For example:

.nc|.cnccontour|.pdf|.dxf|.NC|.CNCCONTOUR|.PDF|.DXF

Virtual directory path

This setting allows you to explicitly specify a virtual directory. For example:

/TcHmiCncFileFolder

If you explicitly specify a virtual directory, you will no longer be able to switch between different directories using the combo box in the HMI.

Channel Selection Symbol

For File Explorer, you can use this setting to specify the channel in which an NC program is selected by default. Data bindings to “ProgName” of the “CNC-ChannelType” data type are permitted. For example:

- CNC.Data.ChannelClientSelected.ProgName
- CNC.Data.Channel.0.ProgName

If no data binding is configured, a pop-up window asks the user in which channel the NC program should be selected.

User settings via TcHmiCncSettings

You can adjust the following settings via the control `TcHmiCncSettings`:

Section	Setting	Standard	Description
TcCncHmi	Recent file limit	20	Number of saved files that were last opened.

Keyboard shortcuts

The control `TcHmiCncFileExplorer` does not have any specific keyboard shortcuts.

API functions

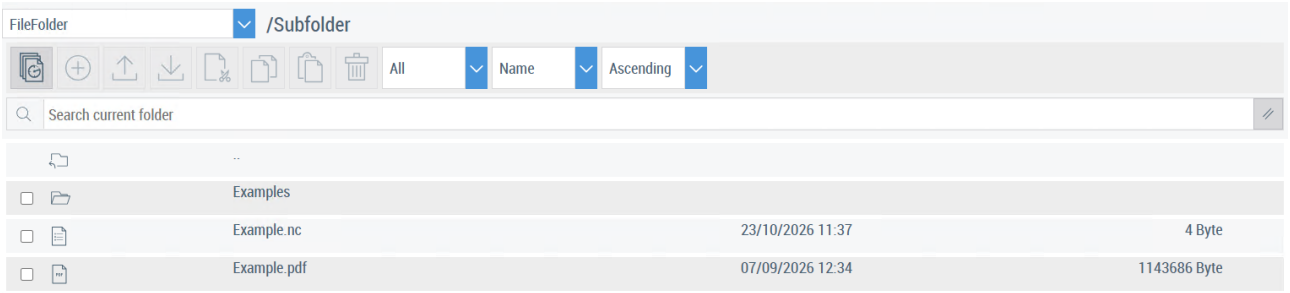
The control `TcHmiCncFileExplorer` has a public API that allows you to use the following functions:

Function	Description
getMode() / setMode()	Read or set the mode.
getFilter() / setFilter()	Read or set filters for file extensions.
getDirVirtual() / setDirVirtual()	Read or set a virtual directory.
getChannelSelectionSymbol() / setChannelSelectionSymbol()	Set or read the symbol for the channel in which the NC program is selected.
getSelectedFile()	Absolute file path to the selected NC program.

Impact of user management

The control `TcHmiCncFileExplorer` evaluates the “Operate” access right. For “Allow,” the control behaves as usual. When set to “Disallow,” the control is limited to the following:

- Switching Between Virtual Directories
- Searching for and Filtering Files
- Open files in the appropriate editor/viewer
- Show recently used files
- Select NC programs in a channel



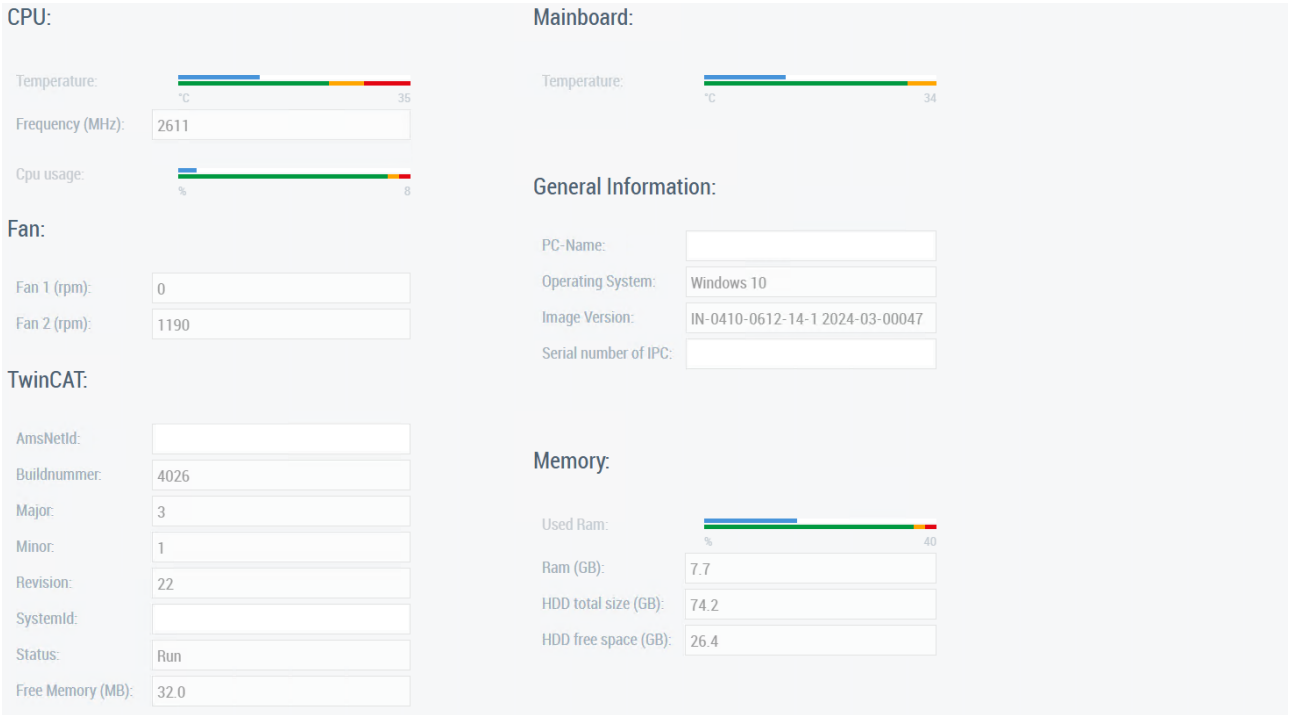
If you have not set up user management, “Operate” is considered “Allow,” and the control behaves as usual.

5.11 TcHmiCncIpcDiagnosis

Component: TcHmiCncIpcDiagnosis
Namespace: TcHmi.Controls.TcHmiCncBase
Available since version: 1.0.0

Description

The control TcHmiCncIpcDiagnosis uses the Modular Device Profile (MDP) extension to retrieve information about the Beckhoff Industrial PC and display it to you. For example, the control displays the CPU temperature, usage, and frequency, as well as fan speeds. In addition, information is displayed regarding the motherboard temperature, RAM and router memory usage, hard drive capacity, TwinCAT version and status, AMS Net ID, operating system, image version, and the serial number of the Beckhoff Industrial PC.

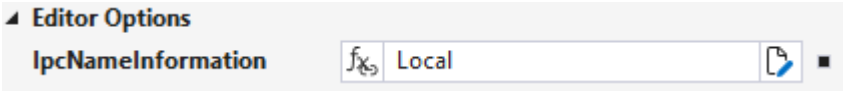


Operation

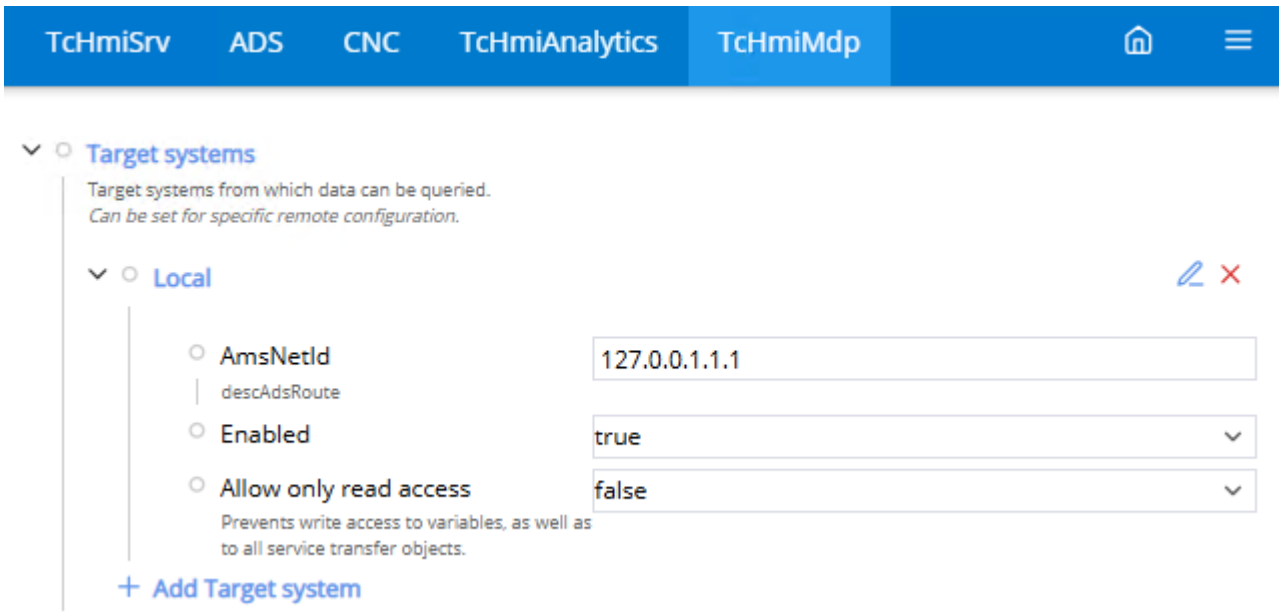
The control TcHmiCncIpcDiagnosis does not have any special operating functions.

Configuration

You can configure the control TcHmiCncIpcDiagnosis as follows. You do not need any special data binding with the TcHmiCnc extension. Define the MDP instance used to display information about the Beckhoff Industrial PC. The following example uses the “Local” MDP instance.



You can configure the MDP instance in the corresponding TcHmiMdp extension.



User settings via TcHmiCncSettings

The control `TcHmiCncIpcDiagnosis` does not have any customizable settings in the control `TcHmiCncSettings`.

Keyboard shortcuts

The control `TcHmiCncIpcDiagnosis` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncIpcDiagnosis` has a public API that allows you to use the following function:

Function	Description
setIpcName() / getIpcName()	Set or read the name of the MDP instance.

Impact of user management

The control `TcHmiCncIpcDiagnosis` does not exhibit any specific change in behavior as a result of the “Operate” access right.

5.12 TcHmiCncLayoutManager

Component: `TcHmiCncLayoutManager`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

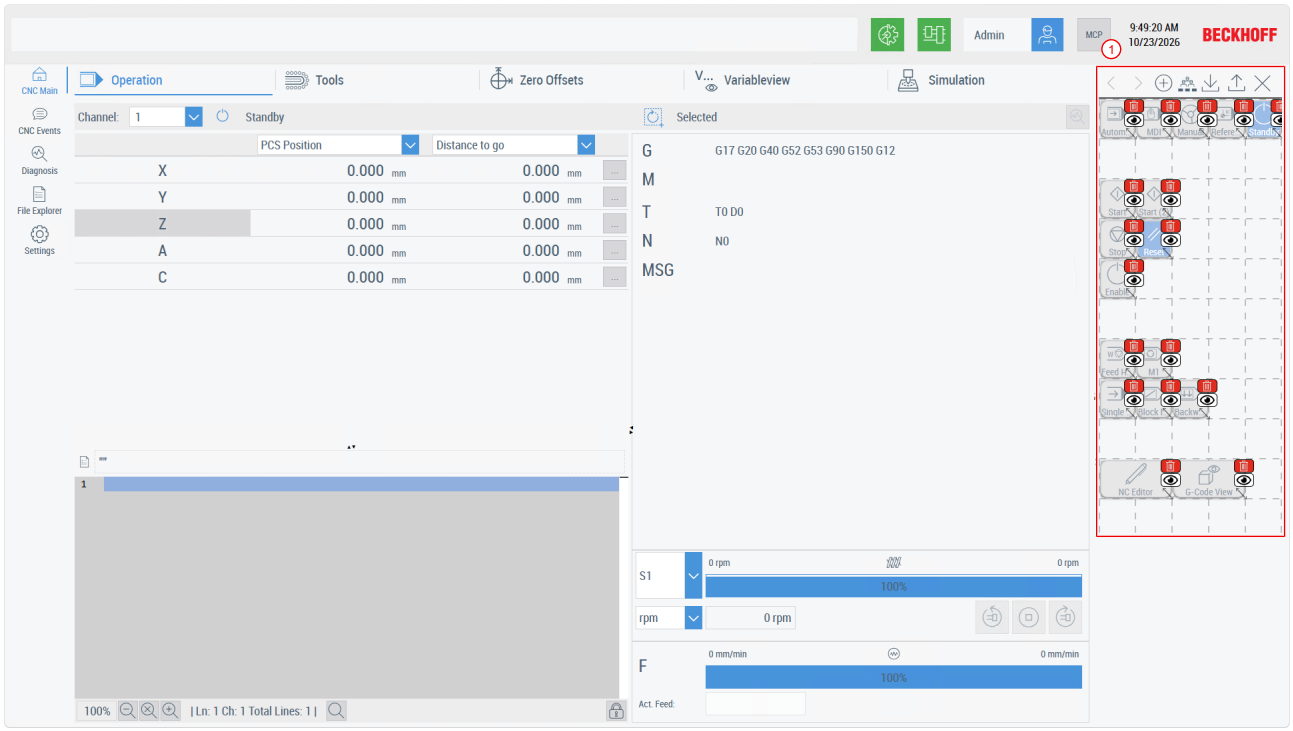
The control `TcHmiCncLayoutManager` is a CNC-specific extension of the control TcHmi.Controls.Analytics.LayoutManager. The control provides a grid in which controls of type TcHmiCncConfigurableButton [▶ 47] can be placed. The controls are dynamically created and managed by the `TcHmiCncLayoutManager` during HMI runtime.

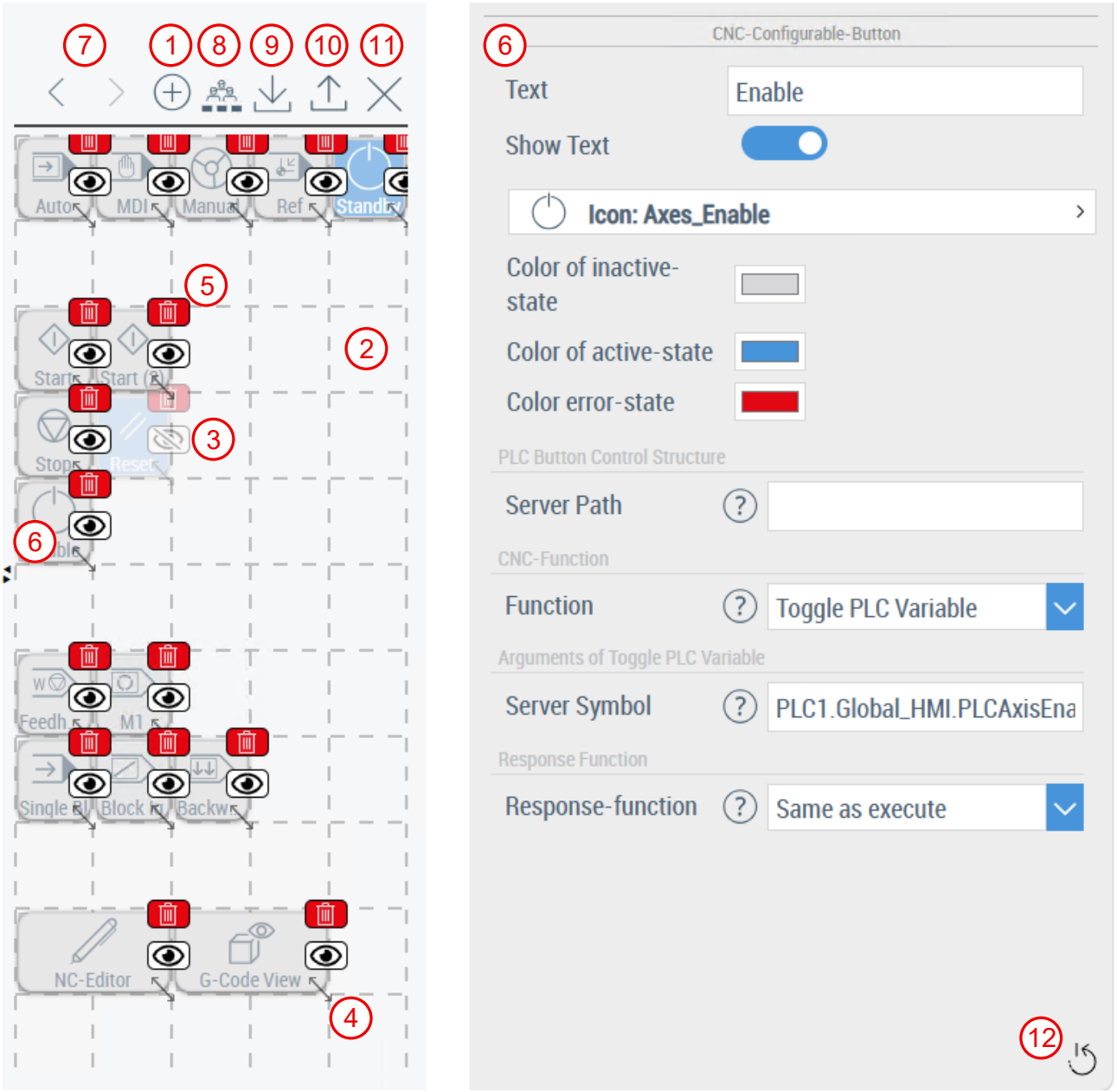
Operation

Using the `TcHmiCncLayoutManager` control, you can create and configure a new button at runtime in the HMI. In the CNC-HMI template, you can access the dialog for configuring the button as follows:

1. Click the **MCP** (**M**achine **C**ontrol **P**anel) button.

⇨ The layout manager opens, allowing you to create and configure a new button.





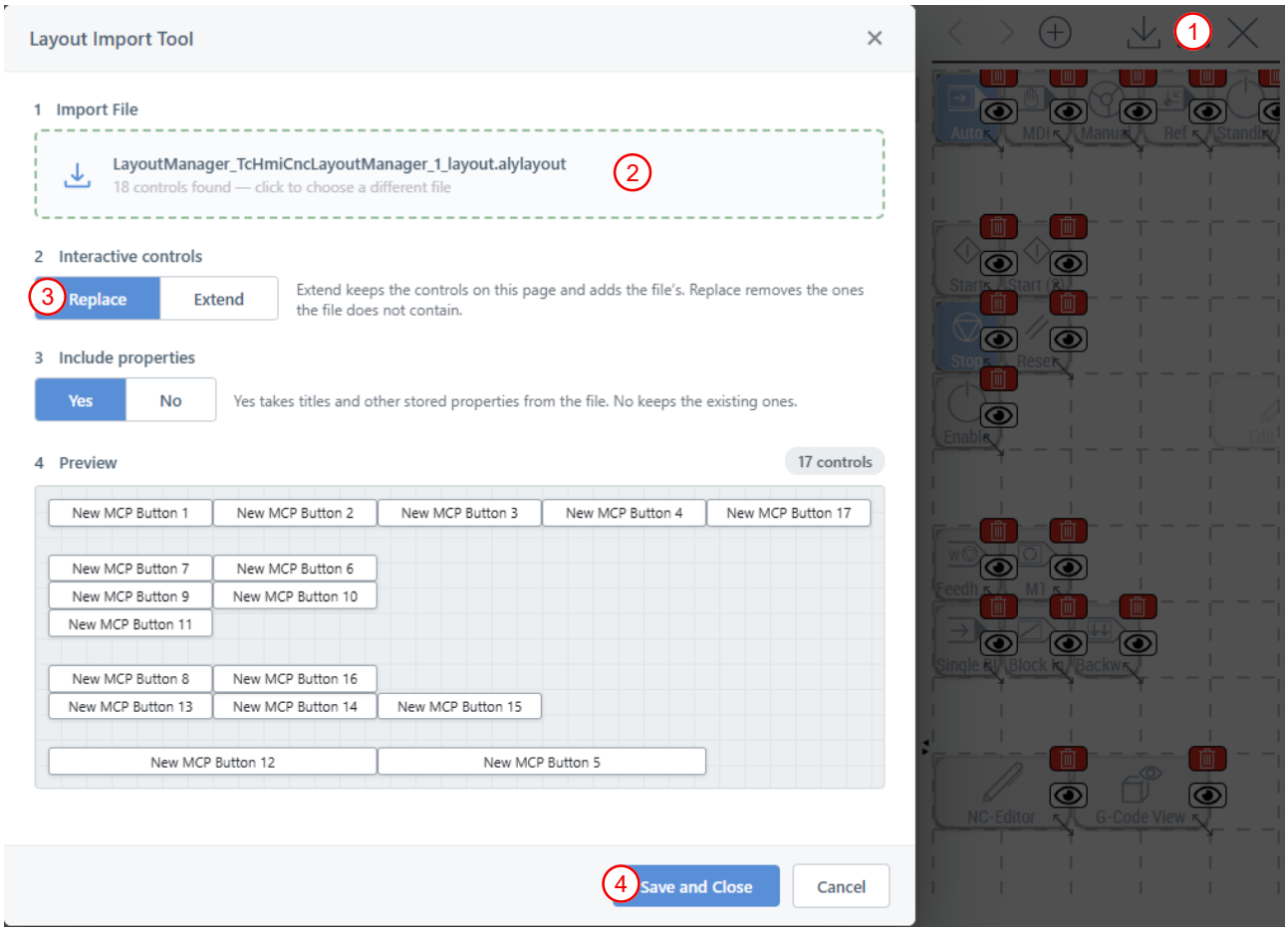
No.	Description
1	Creating a new button
2	Grid against which the buttons are aligned. You can move a button using drag and drop.
3	Hiding a button
4	Changing the size of a button
5	Deleting a button
6	Dialog box for configuring a button. The dialog box appears as an overlay above the buttons when you click one of them. The configuration of a button is described in the control TcHmiCncConfigurableButton [▶ 47].
7	Buttons for undoing or redoing changes.
8	Transferring a layout to another user.
9	Download layout (export).
10	Upload layout (Import).
11	Closing the layout manager. Changes to the layout are automatically saved when you close the layout manager.
12	Resetting a button to its default values.

Import layout

✓ A previously exported layout is available, and the layout manager is open.

1. Click the button to import a layout.
   ⇨ The “Layout Import Tool” pop-up window appears.
2. In the “Import File” section, select a layout that you exported earlier.
3. In the “Interactive controls” section, choose whether you want to replace or extend the existing layout.
4. Click the “ **Save and close**” button to save your changes.

⇨ The layout has been imported.



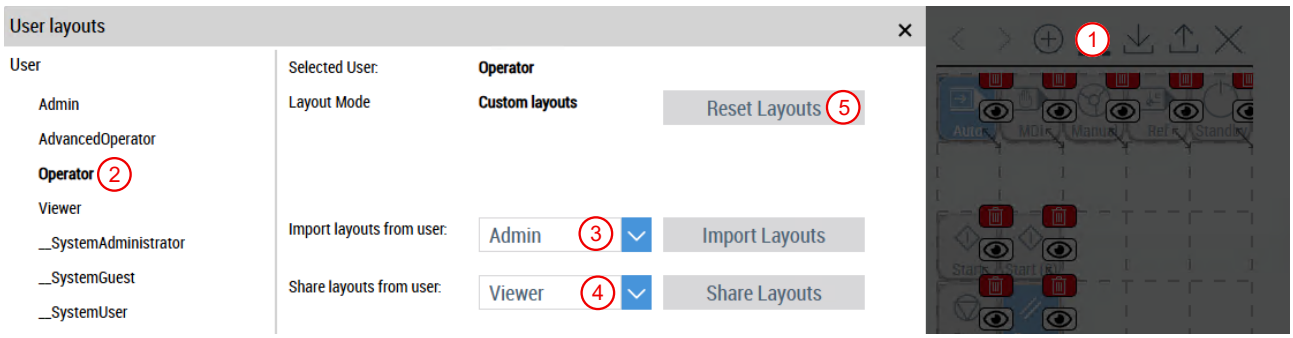
Transferring a layout between users

You can transfer a layout from one user to another. One example of this would be an administrator who configures the layout for a user “Operator” who, in turn, is not allowed to customize the layout.

✓ At least one user has a layout configured and the layout manager open.

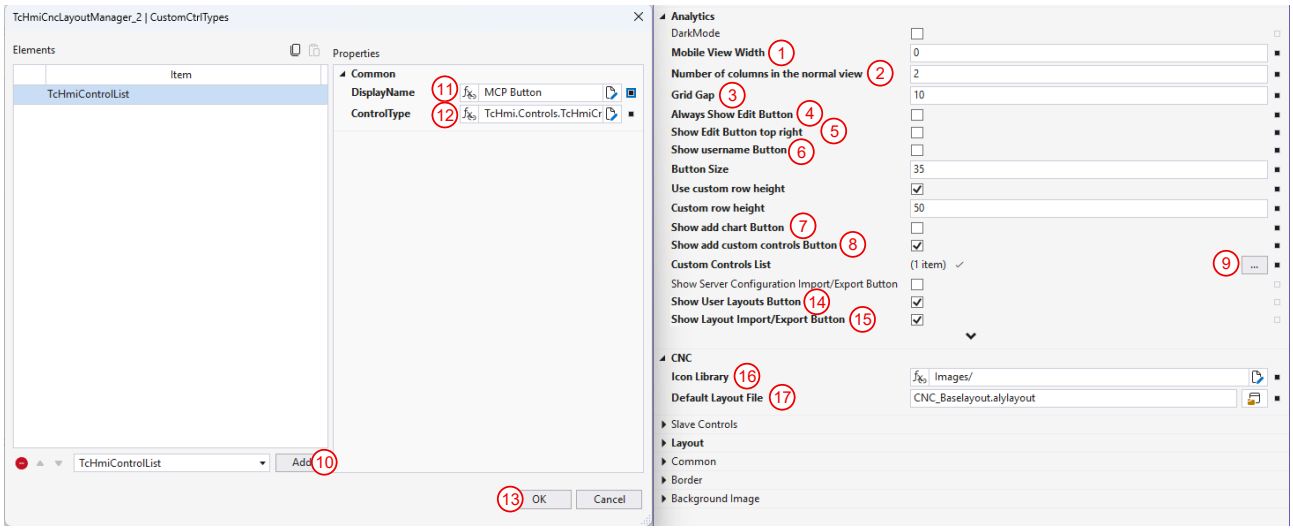
1. Click the button to transfer layouts.
   ⇨ The “User layouts” pop-up window appears.
2. Select a user for whom you want to transfer a layout. In the image, the user “Operator” was selected.
3. From the combo box, select the user whose layout you want to transfer to the “Operator” user, and then click “Import Layouts”. In the image, the layout of the “Admin” user was transferred to the “Operator” user.
4. Alternatively: Select a user from whom to inherit the layout, and then click “Share Layouts”. Changes to the layout will take effect for both users. In the image, the “Viewer” user’s layout was inherited by the “Operator” user
5. Optional: If necessary, set the layout to a default layout.

⇨ You have transferred a layout between two users.



Configuration

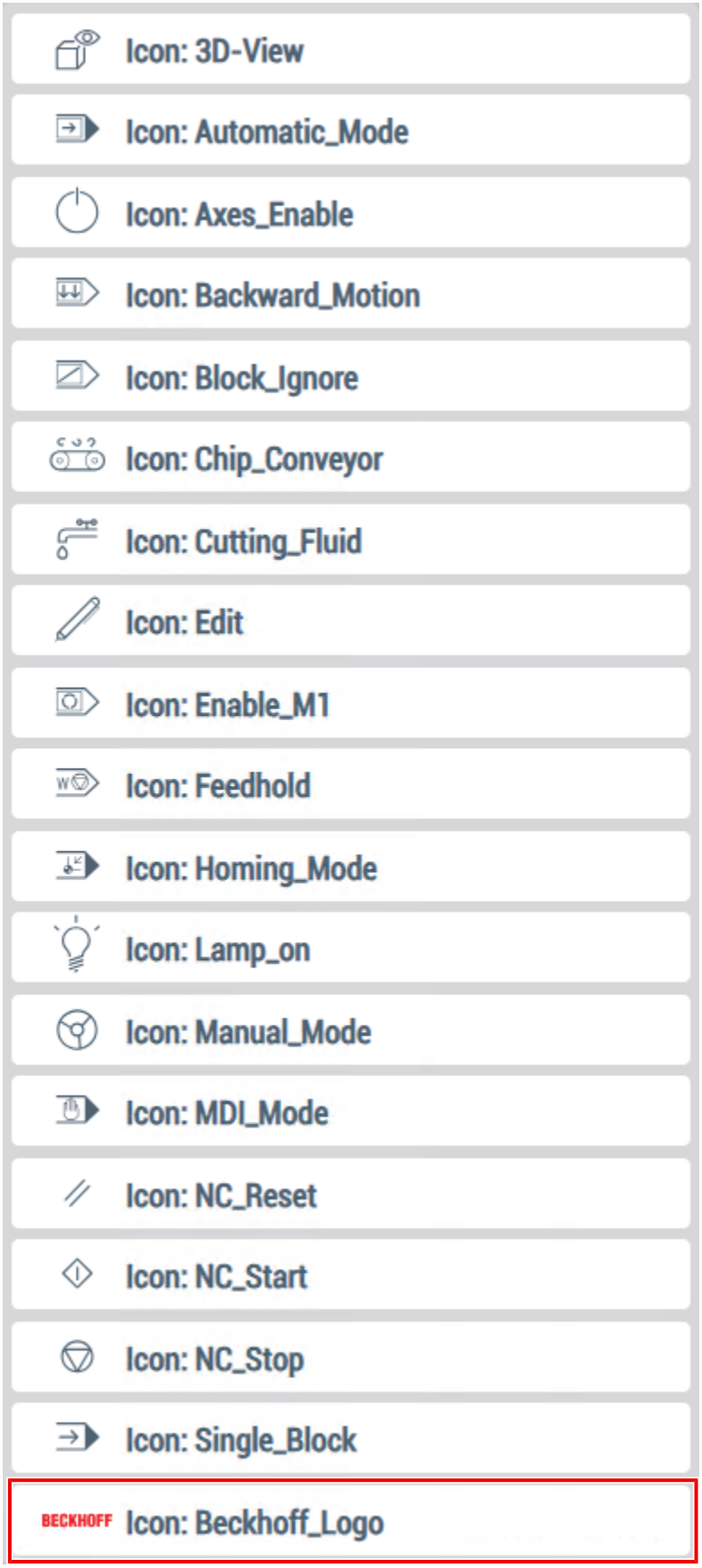
Configure the control `TcHmiCncLayoutManager` in TwinCAT XAE as follows. You do not need any special data binding with the TcHmiCnc extension.



✓ The control `TcHmiCncLayoutManager` is located on a content page and is selected.

1. Set the “Mobile View Width”.
2. Set the “Number of columns”.
3. Adjust the “Grid Gap.“
4. If necessary, set “Always Show Edit Button” to `true`. In the template, a separate button was used to make changes to the control `TcHmiCncLayoutManager`. In this case, you can set “Always Show Edit Button” to `false`.
5. Set “Show Edit Button” to `false`.
6. Set “Show username Button” to `false`.
7. Set “Show add chart Button” to `false`.
8. Set “Show add custom controls button” to `true`.
9. Click “Custom Controls List.”
   ⇨ The “CustomCtrlTypes” pop-up window appears.
10. Click on “Add”
11. Enter “MCP Button” as the “DisplayName”.
12. Specify `TcHmi.Controls.TcHmiCncBase.TcHmiCncConfigurableButton` as the “ControlType”.
13. Confirm the entry with **OK**.
14. Set “Show User Layouts Button” to `true`.
15. Set “Show Layout Import/Export Button” to `true`.

16. Optional: Specify the path to a folder containing SVG files that you want to use as custom icons for the generated buttons. The SVG files must be located at the top level of the hierarchy. Subfolders are not included. In the following example, the Beckhoff logo is displayed as a user-specific icon alongside the standard icons.



17. Optional: Specify a default layout.

⇨ In the HMI, you can add and configure the control TcHmiCncConfigurableButton [▶ 47] using the layout manager.

User settings via TcHmiCncSettings

The control `TcHmiCncLayoutManager` does not have any customizable settings in the control `TcHmiCncSettings`.

Keyboard shortcuts

You can use the following keyboard shortcuts in the control `TcHmiCncLayoutManager`:

Shortcut	Action
Ctrl + Q	Save and close the layout.

API functions

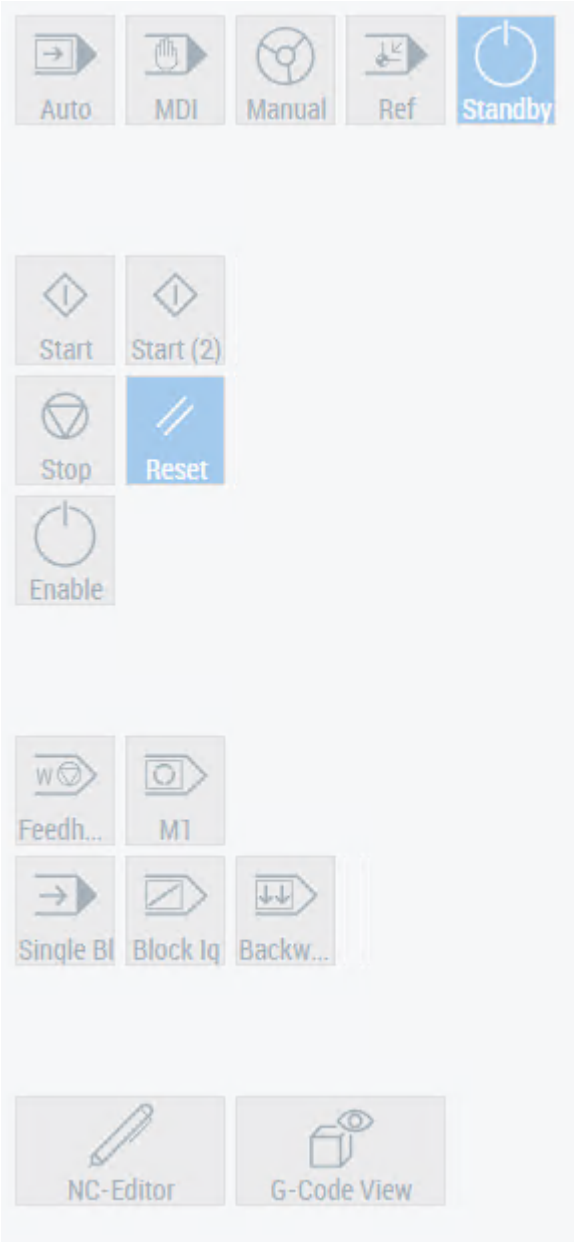
The control `TcHmiCncLayoutManager` has a public API that allows you to use the following functions:

Function	Description
setEditMode()	Set edit mode; this also shows or hides the scrollbars as needed.
storeAndExitLayoutEditor()	Save the layout and exit edit mode.
setIconLibrary() / getIconLibrary()	Set or read the symbol directory.

Impact of user management

The control `TcHmiCncLayoutManager` evaluates the “Operate” access right. For “Allow,” the control behaves as usual. When set to “Disallow,” the control is limited to the following:

- Display of the button. The button itself does not work.



The control `TcHmiCncLayoutManager` evaluates the “Editable” access right. For “Allow,” the control behaves as usual. If “Disallow” is selected, the button cannot be configured in the HMI.



With the “Editable” access right, you allow the user group to map to all PLC variables.

If you have not set up user management, “Operate” and “Editable” are considered “Allow,” and the control behaves as usual.

5.13 TcHmiCncLinearOverride

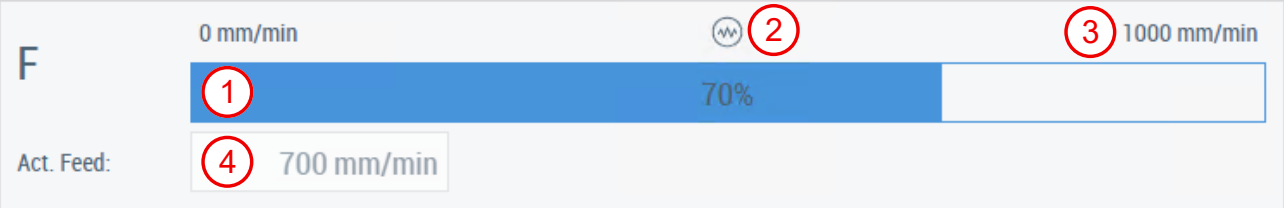
Component: `TcHmiCncLinearOverride`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

The control `TcHmiCncLinearOverride` is an override slider that you can use for both spindles and (axis) feed rate. The way the control works depends on the data bindings you select.



No.	Description
1	Display of the spindle name. If there are multiple spindles, you can switch between them using a combo box.
2	Slider for adjusting the override.
3	Progress bar
4	Configurable icon as an SVG file
5	Maximum rotation speed
6	A combo box for switching between the “rpm” and “Position” operating modes.
7	Actual speed or actual position
8	Target speed or target position
9	Buttons for starting and stopping the spindle.



No.	Description
1	Slider for adjusting the override.
2	Configurable icon as an SVG file
3	Maximum feed rate
4	Current feed rate

Operating spindles

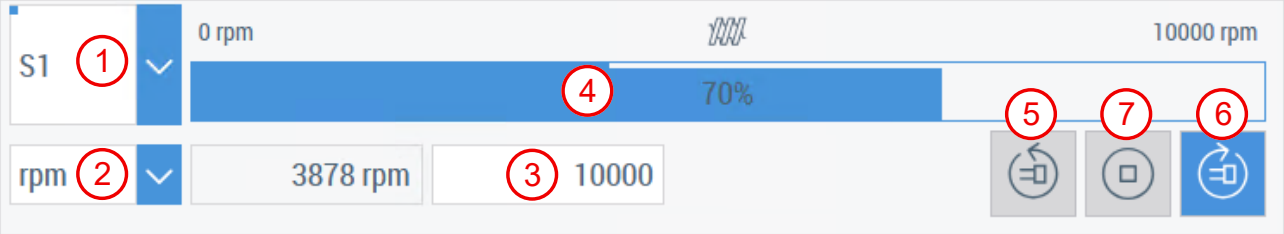
1. Use the combo box to select the spindle you want to control. The combo box appears only if there are multiple spindles in the channel.
2. Select the operating mode for the spindle. The following operating modes are available:

Operation mode	Description
rpm	Rotating the spindle at a constant speed
Position	Moving the spindle to a defined position

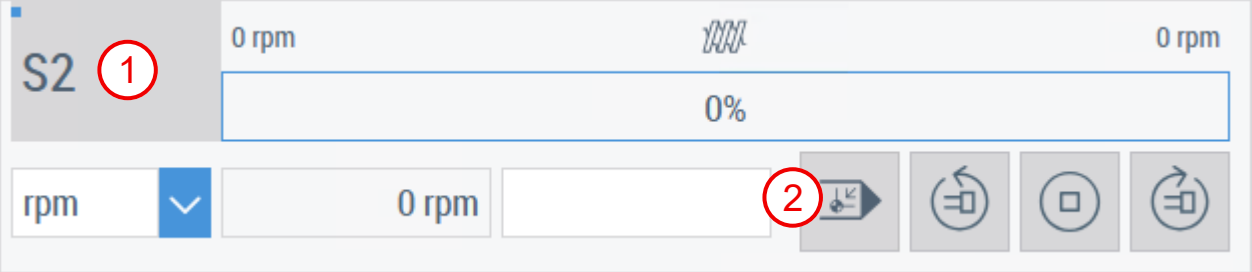
3. Enter a speed or a position in the input field.
4. Use the slider to set the desired override for the spindle.
5. Use the icon to start the spindle rotating counterclockwise (CCW) or

6. clockwise (CW).
7. Stop the rotation.

⇨ You moved the spindle.

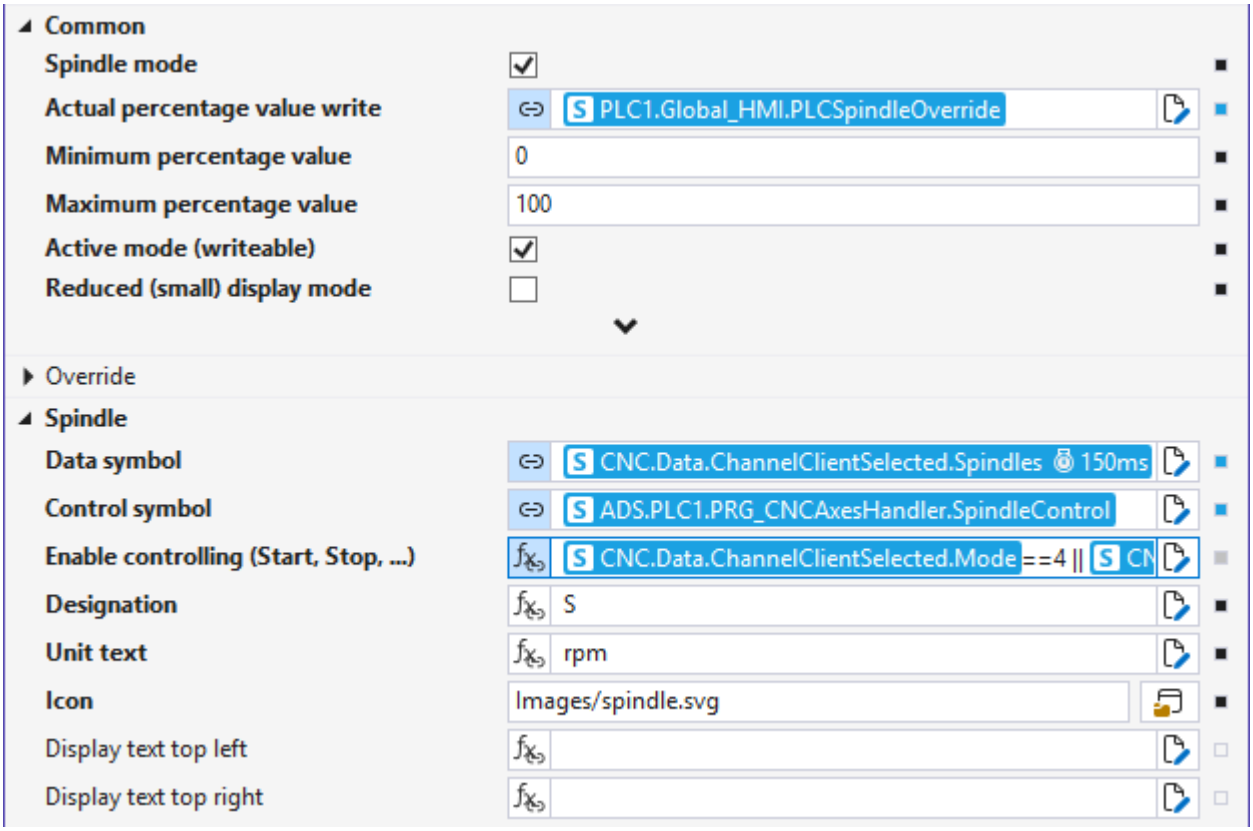


A gray background in the spindle name (1) indicates that the spindle has not yet been referenced. In “Manual” and “MDI” operation modes, you can trigger a **Homing**. An additional button (2) appears for the homing. Click (2) to start the homing.



Configuration for one spindle

Configure the spindle control `TcHmiCncLinearOverride` as follows:



Spindle mode

The spindle control functionality is enabled when you select the `true` checkbox.

Actual percentage value write

Use this setting to specify the variable in the PLC to which the override setpoint is transmitted.

Minimum percentage value

This setting allows you to specify the minimum value for the override slider.

Maximum percentage value

This setting allows you to specify the maximum value for the override slider.

Active mode (writable)

Use this setting to specify whether the slider can be used for the override.

Reduced (small) display mode

This setting reduces the control's display. The line indicating the spindle's movement is not displayed.

Data symbol

The control requires a data binding to the TcHmiCnc extension. Data bindings to a symbol of type "CNC-SpindleType" or an array of this data type are permitted. For example:

- CNC.Data.ChannelClientSelected.Spindles
- CNC.Data.ChannelClientSelected::MainSpindle
- CNC.Data.Channel.<CHANNEL>.Spindles
- CNC.Data.Spindles

Control symbol

The control requires a data binding to a symbol in the PLC in order to move the spindle. The PLC provides the following structure for sending commands to the PLC to control the spindle's movement:

```
TYPE ST_SpindleControl:
STRUCT
    logId: UINT; //Logical ID of Spindle
    cw     BOOL; // Start clock wise
    ccw    BOOL; // Start counter clock wise
    stop   BOOL; // Stop spindle
    mode   UINT := 0; (* 0 = Mode rpm, 1 = Mode Position, 2 = Mode Ref*)
    value: LREAL := 0; (* rpm or position, depending on mode *)
    busy:  BOOL; // Spindle is busy if true
END_STRUCT
END_TYPE
```

In the PLC, create a variable of type ST_SpindleControl and link it using a symbol. The HLI3_MoveSpindleControl function block is available for controlling the spindle via the PLC.

```
VAR
    SpindleControl : ST_SpindleControl;
    fbMoveSpindleControl : HLI3_MoveSpindleControl;
END_VAR
fbMoveSpindleControl(
    stControl := SpindleControl,
    bReset := ,
    bBusy => ,
    bError => ,
    nErrID => ,
    sErrText => );
```

Enable controlling (Start, Stop, …)

The buttons for moving a spindle become active when you set "Spindle enable controlling (Start, Stop, …)" to `true`. To do this, you can either check the box or use data binding. In the illustration, data binding to the `CNC.Data.ChannelClientSelected.Mode==4|| CNC.Data.ChannelClientSelected.Mode==3` icon was used to activate the buttons in "Manual" and "MDI" operation modes.

Designation

Use this setting to specify a label (e.g., "S") for the control.

Unit Text

Use this setting to define the unit displayed for the "Display text top left" and "Display text top right" settings.

Icon

This setting allows you to specify an icon as an SVG file that will be displayed by the control.

Display text top left

Use this setting to define the text displayed in the upper-left corner of the control. If the setting is left blank, the control displays the actual spindle speed.

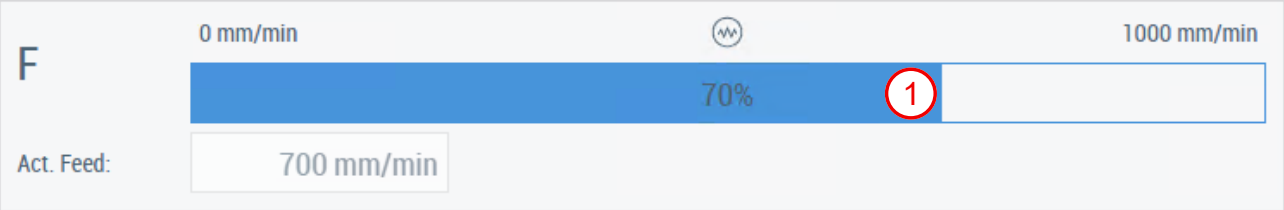
Display text top right

Use this setting to define the text displayed in the upper-right corner of the control. If the setting remains blank, the control displays the target spindle speed programmed via the S-word.

Operating the feed rate override

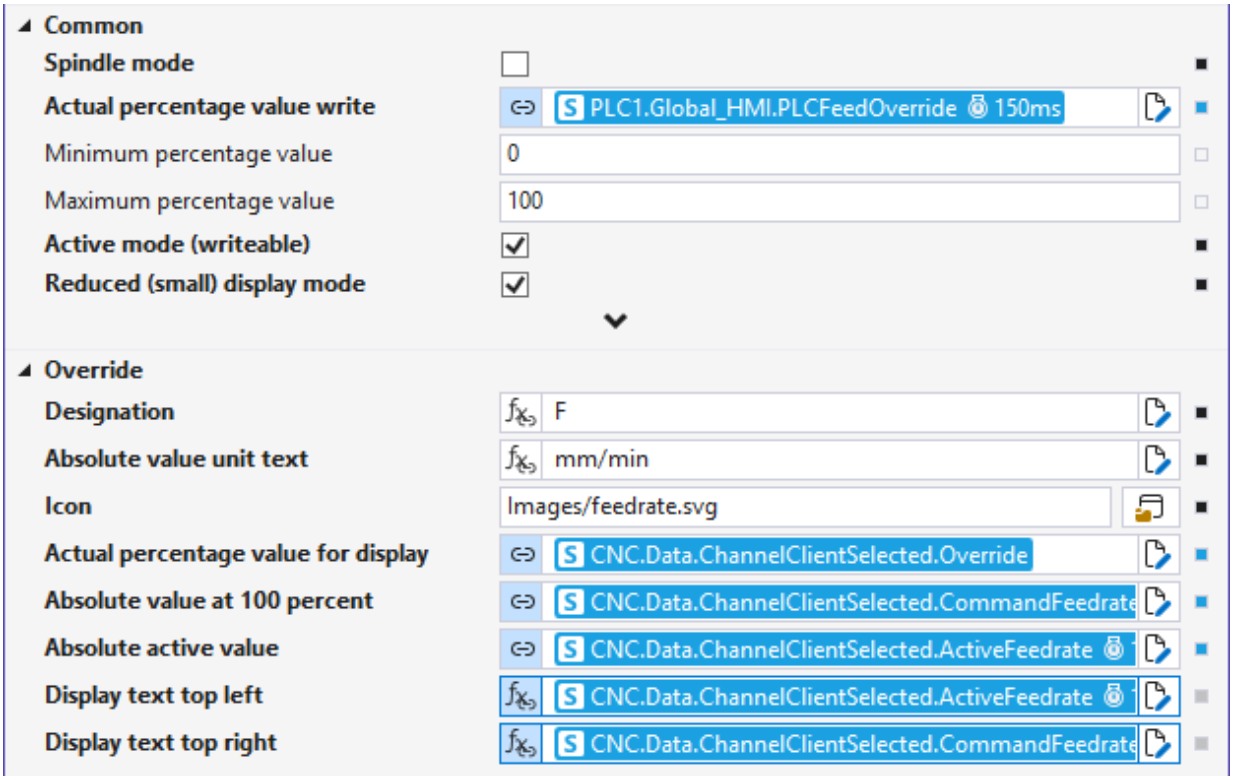
1. Adjust the override using the slider.

⇨ You have set the override.



Configuration for feed rates

Configure the control `TcHmiCncLinearOverride` for the (axis) feed rate as follows:



Spindle mode

The (axis) feed control functionality is enabled when you select `false` in this checkbox.

Actual percentage value write

Use this setting to specify the variable in the PLC to which the override setpoint is transmitted.

Minimum percentage value

This setting allows you to specify the minimum value for the override slider.

Maximum percentage value

This setting allows you to specify the maximum value for the override slider.

Active mode (writable)

Use this setting to specify whether the slider can be used for the override.

Reduced (small) display mode

This setting reduces the control's display. The row containing the actual feed rate is not displayed.

Designation

Use this setting to specify a label (e.g., "F") for the control.

Absolute value unit text

Use this setting to define the unit displayed for the "Display text top left" and "Display text top right" settings.

Icon

This setting allows you to specify an icon as an SVG file that will be displayed by the control.

Actual percentage value for display

With this setting, you define the variable in the PLC through which the control receives the override actual value.

Absolute value at 100 percent

Use this setting to specify the feed rate that corresponds to a 100% override.

Absolute active value

Use this setting to define the actual value used to display the progress bar.

Display text top left

Use this setting to define the text displayed in the upper-left corner of the control.

Display text top right

Use this setting to define the text displayed in the upper-right corner of the control.

User settings via TcHmiCncSettings

You can adjust the following settings via the control `TcHmiCncSettings`:

Section	Setting	Standard	Description
TcCncHmi	Decimal Places	3	Decimal places for displaying target and actual values

Keyboard shortcuts

The control `TcHmiCncLinearOverride` does not have any specific keyboard shortcuts.

API functions

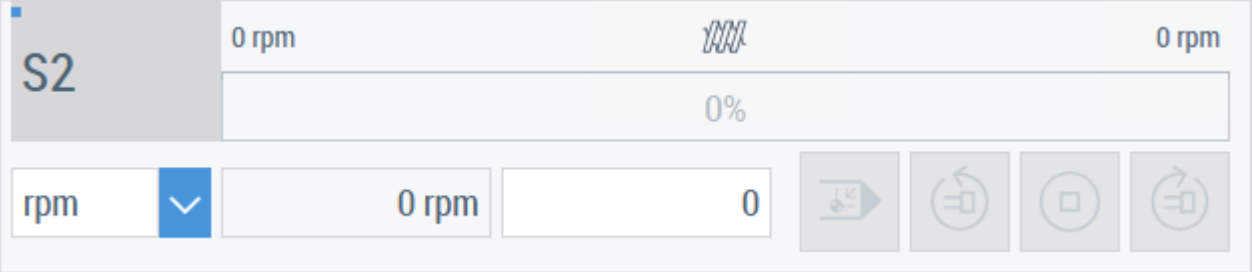
The control `TcHmiCncLinearOverride` has a public API that allows you to use the following functions.

Function	Description
setIsSpindle() / getIsSpindle()	Set or read the “Spindle mode” setting.
setSpindleDataSymbol() / getSpindleDataSymbol()	Set or read the “Data symbol” setting.
setSpindleControlling() / getSpindleControlling()	Set or read the “Enable controlling (Start, Stop, …)” setting.
setSpindleControlSymbol() / getSpindleControlSymbol()	Set or read the “Control symbol” setting.
setCurrValuePercent() / getCurrValuePercent()	Set or read the “Actual percentage value for display” setting.
setCurrValuePercentWriteSymbol() / getCurrValuePercentWriteSymbol()	Set or read the “Actual percentage value write” setting.
setMinValuePercent() / getMinValuePercent()	Set or read the “Minimum percentage value” setting.
setMaxValuePercent() / getMaxValuePercent()	Set or read the “Maximum percentage value” setting.
setValueAbsolute100Percent() / getValueAbsolute100Percent()	Set or read the “Absolute value at 100 percent” setting.
setValueAbsoluteActive() / getValueAbsoluteActive()	Set or read the “Absolute active value” setting.
setOverrideUnitText() / getOverrideUnitText()	Set or read the “Absolute value unit text” setting.
setSpindleUnitText() / getSpindleUnitText()	Set or read the “Unit text” setting.
setOverrideDesignation() / getOverrideDesignation()	Set or read the “Designation” setting.
setSpindleDesignation() / getSpindleDesignation()	Set or read the “Designation” setting
setActiveMode() / getActiveMode()	Set or read the “Active mode (writeable)” setting.
setReducedMode() / getReducedMode()	Set or view the “Reduced (small) display mode” setting.
setDisplayTextTopLeft() / getDisplayTextTopLeft()	Set or read the “Display text top left” setting.
setDisplayTextTopRight() / getDisplayTextTopRight()	Set or read the “Display text top right” setting.
setSpindleDisplayTextTopLeft() / getSpindleDisplayTextTopLeft()	Set or read the “Display text top left” setting.
setSpindleDisplayTextTopRight() / getSpindleDisplayTextTopRight()	Set or read the “Display text top right” setting.
setOverrideIconPath() / getOverrideIconPath()	Set or read the “Icon” setting.
setSpindleIconPath() / getSpindleIconPath()	Set or read the “Icon” setting.

Impact of user management

The control `TcHmiCncLinearOverride` evaluates the “Operate” access right. For “Allow,” the control behaves as usual. When “Disallow” is selected, the control is limited to the following:

- Display of target and actual values only
- Switching between spindles (when there are multiple spindles in the channel)



If you have not set up user management, “Operate” is considered “Allow,” and the control behaves as usual.

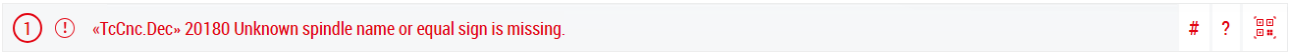
5.14 TcHmiCncMessageLine

Component: `TcHmiCncMessageLine`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

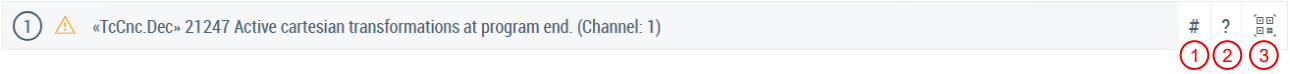
The control `TcHmiCncMessageLine` is a single-line event ticker that displays all unacknowledged (i.e., active) events from the TwinCAT event logger. For each event, the control displays the following:

- Active event number
- Icon indicating the severity of the event (“Critical,” “Error,” “Warning,” “Info,” “Message“)
- Optional: Source of the event (e.g., PLC, TcCnc.Dec)
- Display text
- Buttons for the event



Operation

The control `TcHmiCncMessageLine` always displays exactly one event. If there are multiple events, the display automatically switches between them. If there are no active events, the display area is automatically cleared.



No.	Description
1	The **#** button appears if the event is attributable to the executed NC program. Clicking the button takes you directly to the corresponding line in the NC program.
2	The **?** button appears if a “Description URL” has been specified for the event. The button typically takes you, via a pop-up window, to the online documentation for the event in the Beckhoff Information System (InfoSys). For events with the source “TcCnc,” offline documentation is also available; you can open it by clicking the **?** button.
3	This button is available only for events with the source “TcCnc”. When you click the button, a QR code appears in a pop-up window that links to the corresponding page in InfoSys.

Configuration

The control `TcHmiCncMessageLine` does not require any special data binding with the TcHmiCnc extension.

User settings via TcHmiCncSettings

You can adjust the following settings via the control `TcHmiCncSettings`:

Section	Setting	Standard	Description
TcCncHmi.MessageLine	Display interval in [ms]	1000 ms	Display interval in milliseconds between switching to the next event
TcCncHmi.MessageLine	Show source name of message	`false`	Display the source of the event
TcCncHmi.MessageLine	Only show max. severity	`false`	In `true`, only the events with the highest severity are displayed from the active events.

Keyboard shortcuts

The control `TcHmiCncMessageLine` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncMessageLine` does not have any specific API functions.

Impact of user management

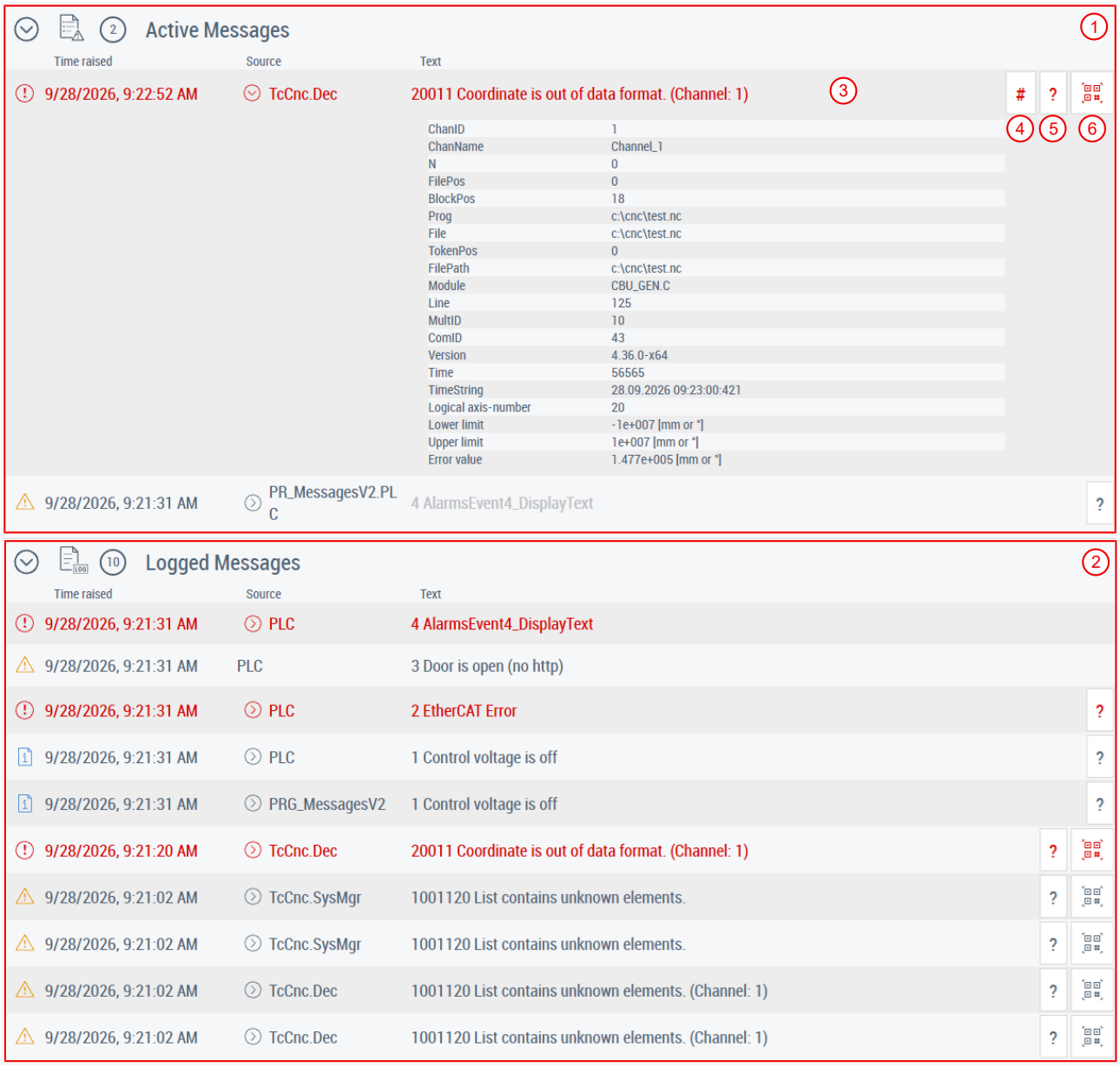
The control `TcHmiCncMessageLine` does not exhibit any specific change in behavior as a result of the “Operate” access right.

5.15 TcHmiCncMessageList

Component: `TcHmiCncMessageList`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

The control `TcHmiCncMessageList` lists all unacknowledged (i.e., active) events from the TwinCAT event logger. In addition, a configurable number of historical messages from the event logger will be displayed.



No.	Description
1	List of all unacknowledged (i.e., active) events from the TwinCAT event logger.
2	List containing a configurable number of historical entries from the event logger
3	Details about the event. Each row in the list shows you the following information about the event: • Icon indicating the severity of the event (“Critical,” “Error,” “Warning,” “Info,” “Message”) • Timestamp • Optional: Source of the event (e.g., PLC, TcCnc.Dec) • Display text • Buttons for the event
4	The **#** button appears if the event is caused by the NC program being executed. Clicking the button takes you directly to the corresponding line in the NC program.
5	The **?** button appears if a “Description URL” has been specified for the event. The button typically takes you, via a pop-up window, to the online documentation for the event in the Beckhoff Information System (InfoSys). For events with the source “TcCnc,” offline documentation is also available; you can open it by clicking the **?** button.
6	This button is available only for events with the source “TcCnc”. When you click the button, a QR code appears in a pop-up window that links to the corresponding page in InfoSys.

Operation

The operation of the `TcHmiCncMessageList` control is explained in the Description [▶ 95] section.

Configuration

The control `TcHmiCncMessageList` does not require any special data binding with the TcHmiCnc extension.

User settings via TcHmiCncSettings

You can adjust the following settings via the control `TcHmiCncSettings`:

Section	Setting	Standard	Description
TcCncHmi.MessageList	Number of logged messages displayed	10	Maximum number of historical records that can be retrieved and displayed.

Keyboard shortcuts

The control `TcHmiCncMessageList` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncMessageList` does not have any specific API functions.

Impact of user management

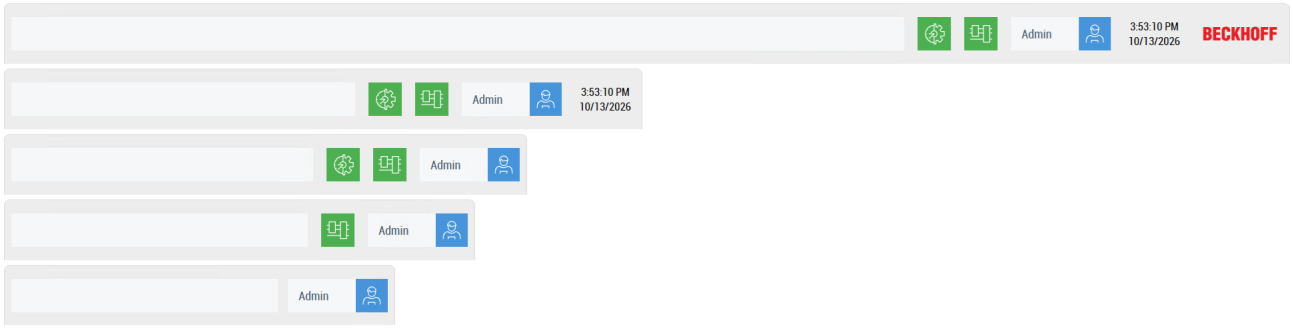
The control `TcHmiCncMessageList` does not exhibit any specific change in behavior as a result of the “Operate” access right.

5.16 TcHmiCncPriorityGrid

Component: TcHmiCncPriorityGrid
Namespace: TcHmi.Controls.TcHmiCncBase
Available since version: 1.0.0

Description

The control TcHmiCncPriorityGrid is a layout container that you can use to develop fluid and responsive designs. It extends the control TcHmiGrid by adding a priority order. Similar to the control TcHmiGrid, the control TcHmiCncPriorityGrid provides a grid consisting of a configurable number of columns and rows. You can assign an additional control to each cell and define how it behaves when the browser size changes.



The main enhancement compared to the control TcHmiGrid is that you can set a visibility priority for each column in the grid. When the browser width is reduced, the columns are hidden one after another based on their priority order so that the prioritized columns remain visible. As the browser width increases, the hidden columns are restored in reverse order.

Operation

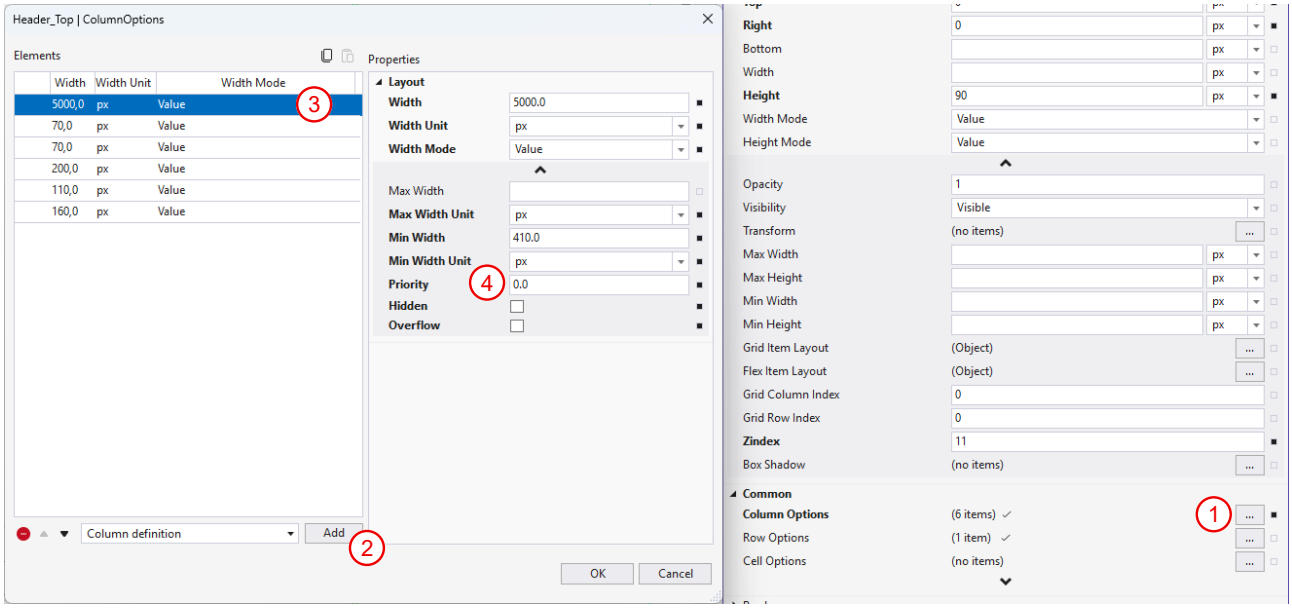
The control TcHmiCncPriorityGrid is a layout container without its own user interface. Columns are automatically adjusted based on the available width.

Configuration

The control TcHmiCncPriorityGrid does not require any special data binding with the TcHmiCnc extension. The configuration is similar to that of the control TcHmiGrid. To set the order of priority, follow these steps:

1. Click on the button “…”.
   ⇨ The “Header_Top | ColumnOptions” window opens.
2. If necessary, use the **Add** button to add additional columns.
3. Select one of the columns.
4. Set the priority for the column.

⇨ Columns with the lowest values have the highest priority and are hidden last. The value “0” means that the column will never be hidden automatically.



User settings via TcHmiCncSettings

The control `TcHmiCncPriorityGrid` does not have any customizable settings in the control `TcHmiCncSettings`.

Keyboard shortcuts

The control `TcHmiCncPriorityGrid` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncPriorityGrid` has a public API that allows you to use the following function.

Function	Description
setColumnOptions() / getColumnOptions()	Set or read column options. After setting this option, the visibility of the columns is recalculated immediately.

Impact of user management

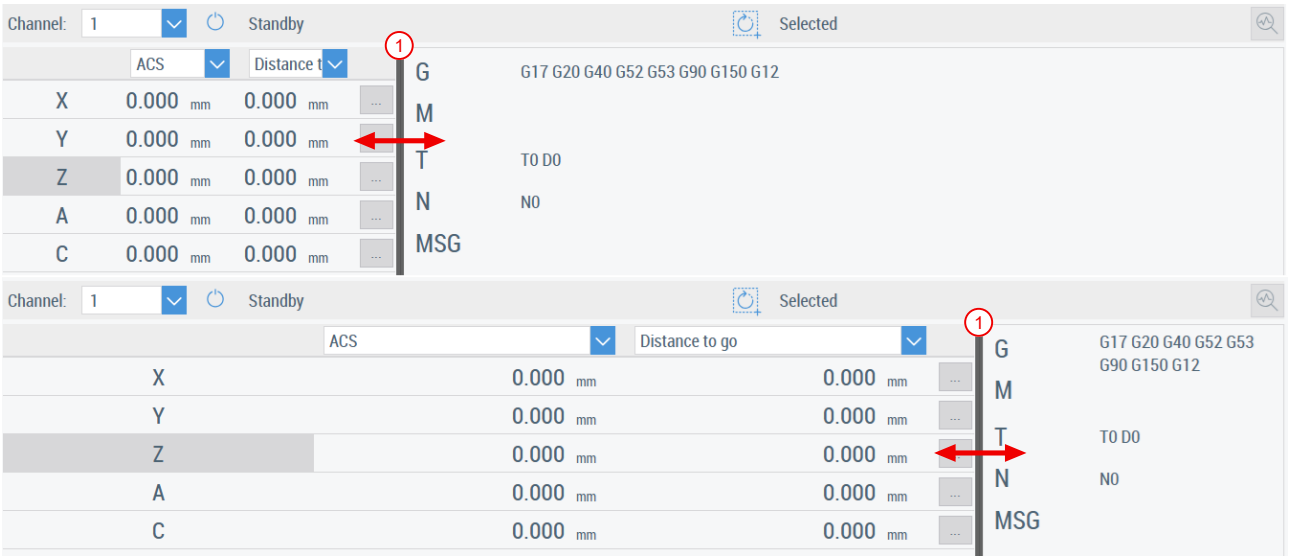
The control `TcHmiCncPriorityGrid` does not exhibit any specific change in behavior as a result of the “Operate” access right.

5.17 TcHmiCncScaleRegions

Component: TcHmiCncScaleRegions
Namespace: TcHmi.Controls.TcHmiCncBase
Available since version: 1.0.0

Description

The control TcHmiCncScaleRegions is a layout container for a grid consisting of a configurable number of columns and rows. For each cell, you can assign an additional control and define how the cell should behave when its size changes. There are sliding dividers (1) between adjacent cells. In the browser, you can use this to dynamically change the size of the adjacent areas.



The selected layout is saved locally in the browser, so each operating device retains its own layout.

Operation

The control TcHmiCncScaleRegions is a layout container without its own user interface.

Configuration

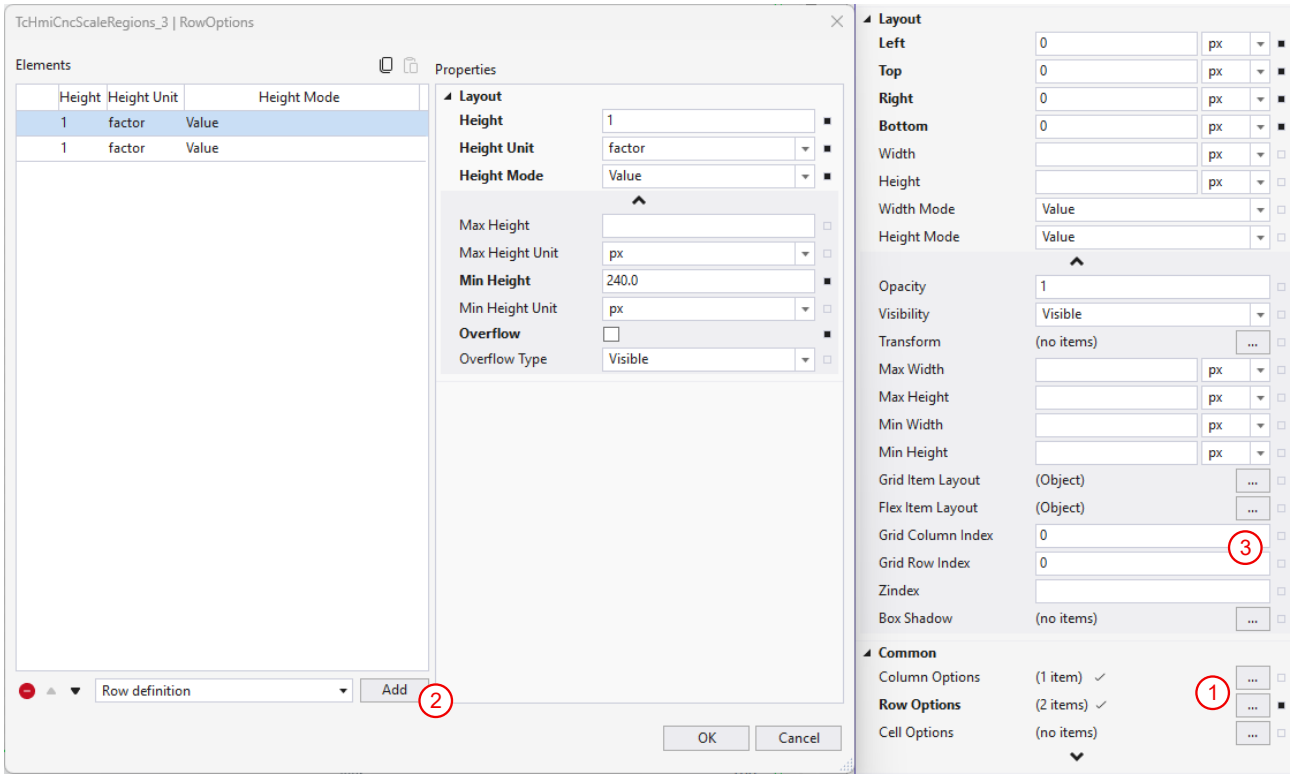
The control TcHmiCncScaleRegions does not require any special data binding with the TcHmiCnc extension. The configuration is similar to that of the TcHmiGrid control.
Observe the following instructions:

1. Set the number of columns and rows in the grid using “Column Options” and “Row Options”. Click on the button “…”.
   ⇨ A window will open where you can configure the rows and columns.
2. Use the **Add** button to add the appropriate elements for the columns or rows.
   ⇨ The control creates a grid with the number of columns and rows you specify. You can drag and drop your controls into the cells of the grid.

You can combine multiple TcHmiCncScaleRegions controls to divide a cell into additional sections. To do this, insert another TcHmiCncScaleRegions control into any cell in the grid.

3. Next, specify the column or row index for this cell.

⇨ The control is positioned in the corresponding cell in the parent grid using the column or row index.



User settings via TcHmiCncSettings

The control `TcHmiCncScaleRegions` does not have any customizable settings in the control `TcHmiCncSettings`.

Keyboard shortcuts

The control `TcHmiCncScaleRegions` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncScaleRegions` has a public API that allows you to use the following functions:

Function	Description
setCellWrap() / getCellWrap()	Turn cell wrapping on or off.
setStateDependingOn() / getStateDependingOn()	Set or read the status key. When a change is made, the control loads the factors saved for that state.

Impact of user management

The control `TcHmiCncScaleRegions` evaluates the “Operate” access right. For “Allow,” the control behaves as usual. When set to “Disallow,” the control is limited to the following:

- You can continue to use the controls in the cells (in accordance with your respective access rights).
- You can no longer adjust the size (by moving the divider).

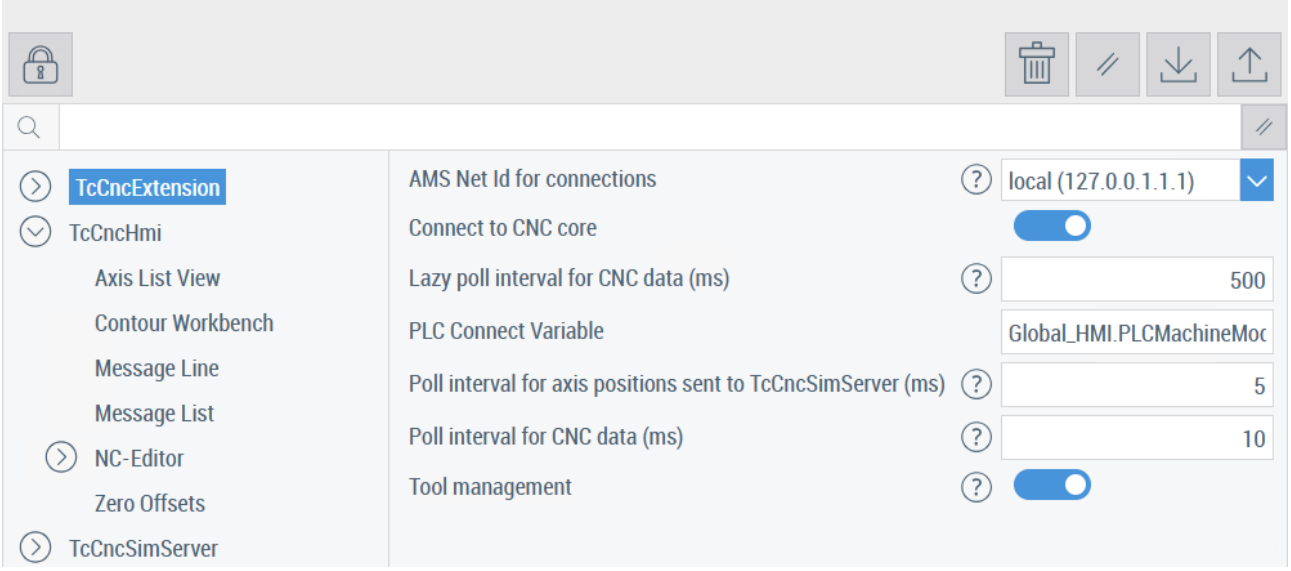
If you have not set up user management, “Operate” is considered “Allow,” and the control behaves as usual.

5.18 TcHmiCncSettings

Component: `TcHmiCncSettings`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

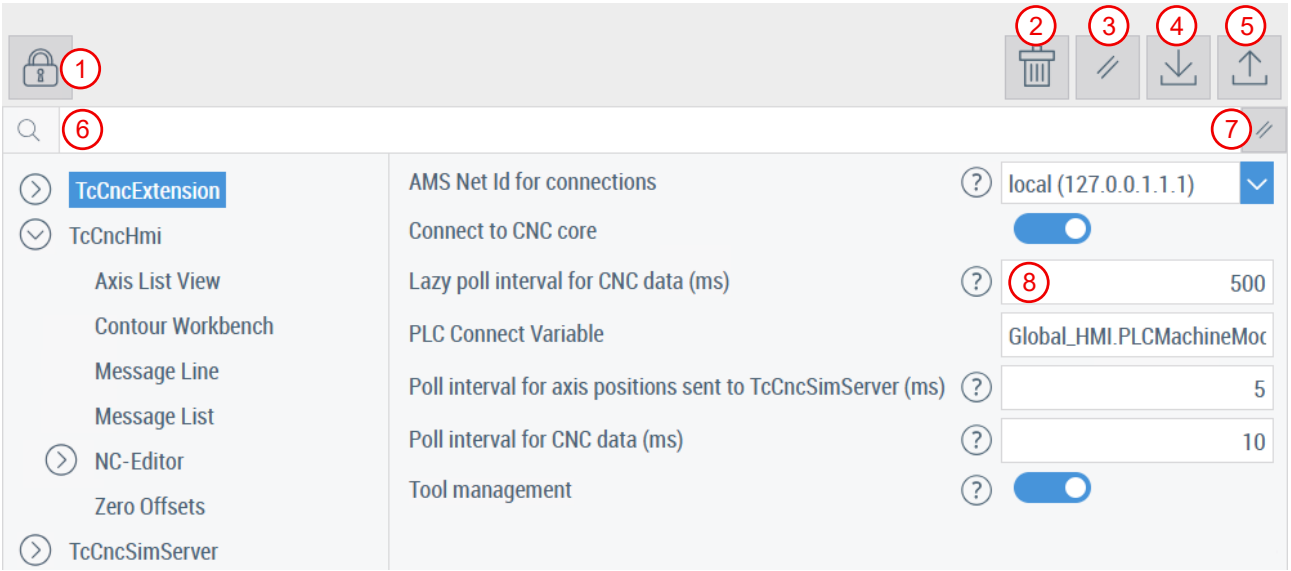
Description

The control `TcHmiCncSettings` allows you to adjust settings for the CNC HMI that take effect at the runtime of the HMI. The first time you use a control from the TF5310 product in the HMI, the corresponding settings are dynamically generated by the control `TcHmiCncSettings`.



Operation

The settings are displayed in a tree structure.



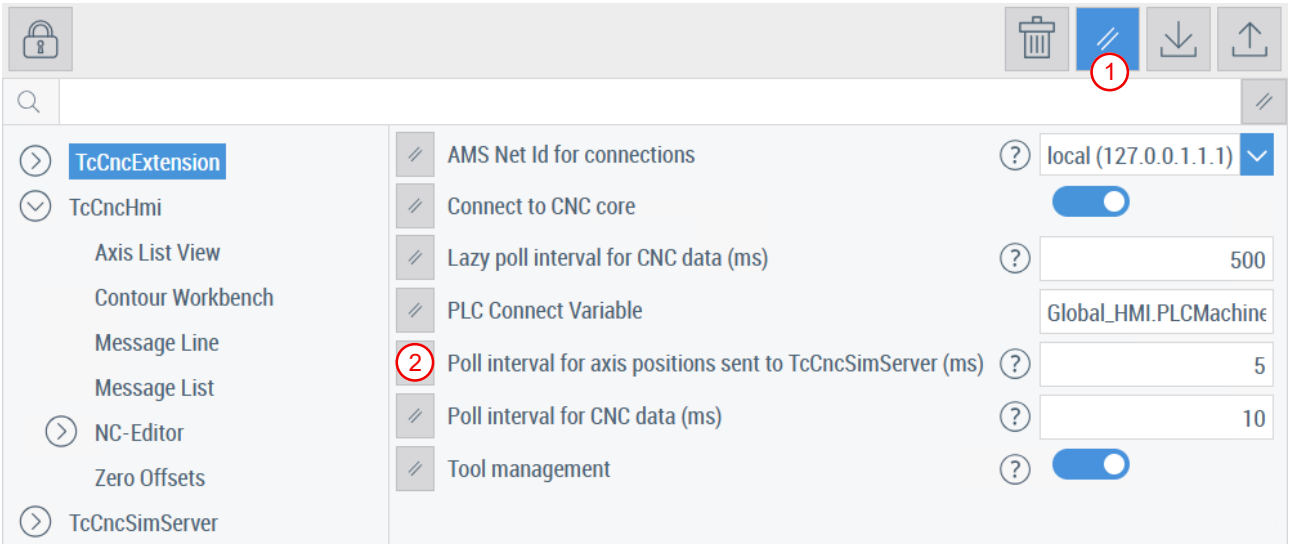
No.	Description
1	Changing access permissions
2	Deleting settings
3	Resetting settings to default values
4	Download all settings (export) as a .cncsettings file
5	Upload all settings (Import). During the import, the existing settings will be overwritten.
6	Search for settings
7	Reset text search
8	Input field for a setting. Changes will be saved as soon as you leave the input field.

Reset settings

You can reset each setting to its predefined default value.

1. Click the button to reset the settings.
   ⇨ An icon appears next to each setting to reset it.
2. Click the icon to reset the corresponding setting.

⇨ You have reset the setting. When a setting is reset, the configured access rights are retained.

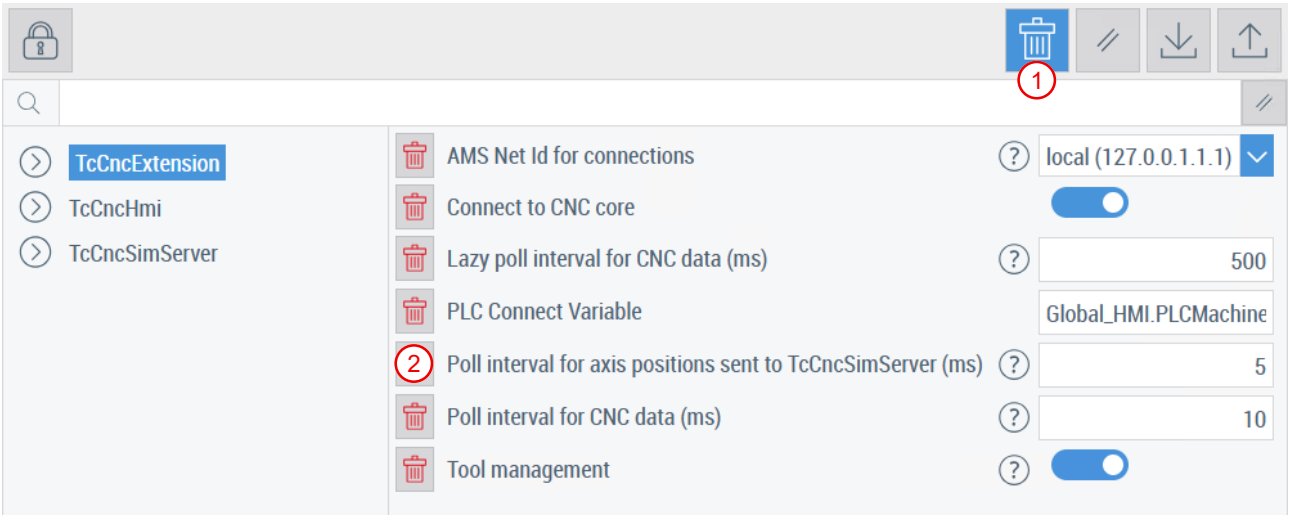


Delete settings

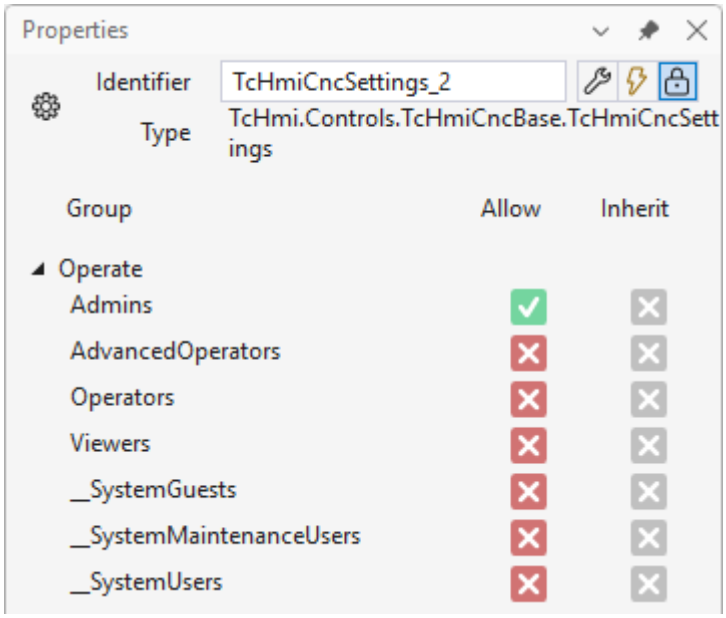
You can delete any setting individually.

1. Click the button to delete settings.
   ⇨ Next to each setting, there is an icon that allows you to delete that setting by double-clicking it.
2. Double-click the icon to delete the setting.

⇨ You have deleted the setting. When you delete a setting, the configured access rights are reset.



If the setting is recreated when a control is loaded, the default values are set as the access rights. By default, all user groups that have the “Operate” access right for the TcHmiCncSettings control can customize the settings.



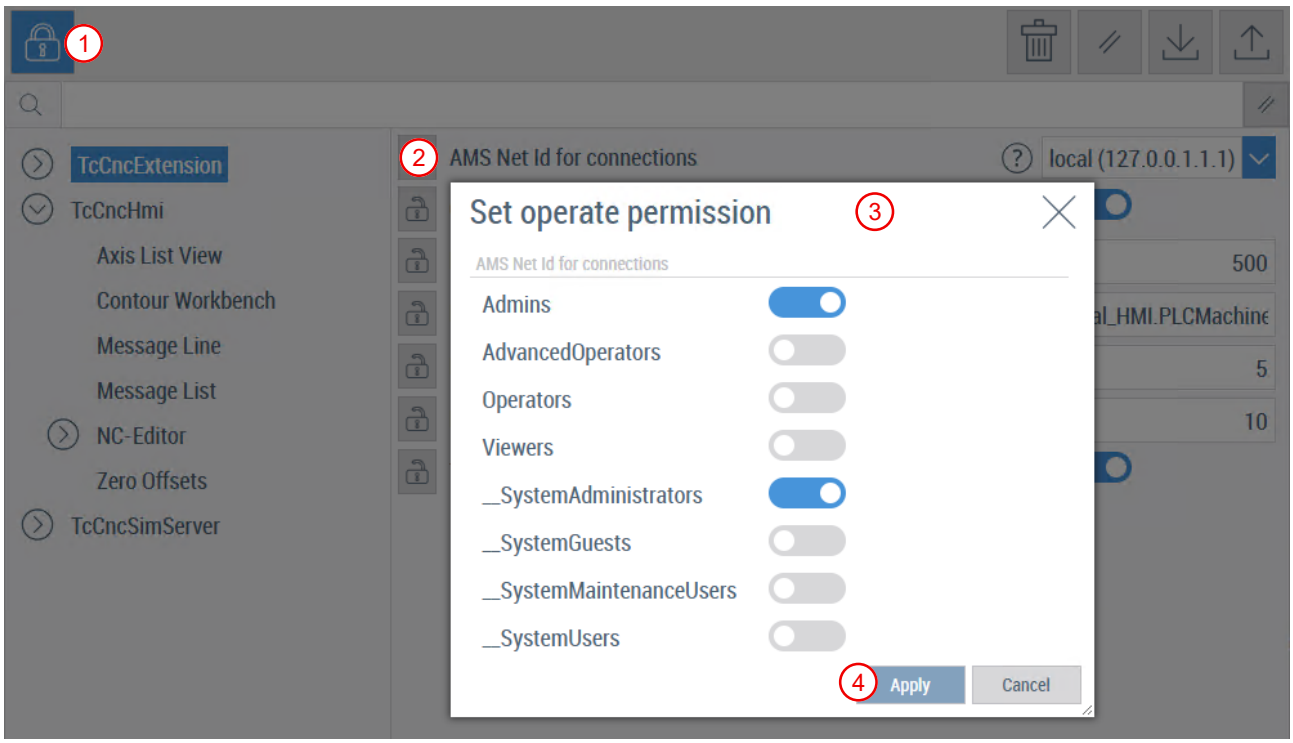
Adjust access rights

For each setting, you can specify which user group is allowed to change it.

1. Click the button to change access permissions.
   ⇨ An icon appears next to each setting to adjust access permissions.
2. Click one of the icons.
   ⇨ The “Set operate permission” pop-up window opens.
3. In the context menu, configure the access rights for the user groups.
4. Click “Apply” to save your changes.

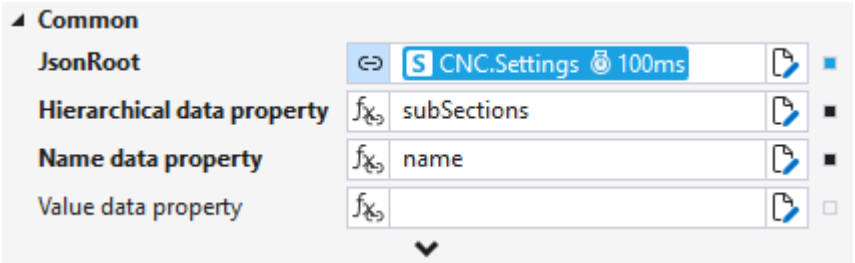
⇨ You have adjusted the access permissions.

When you delete a setting, the configured access rights are also reset.



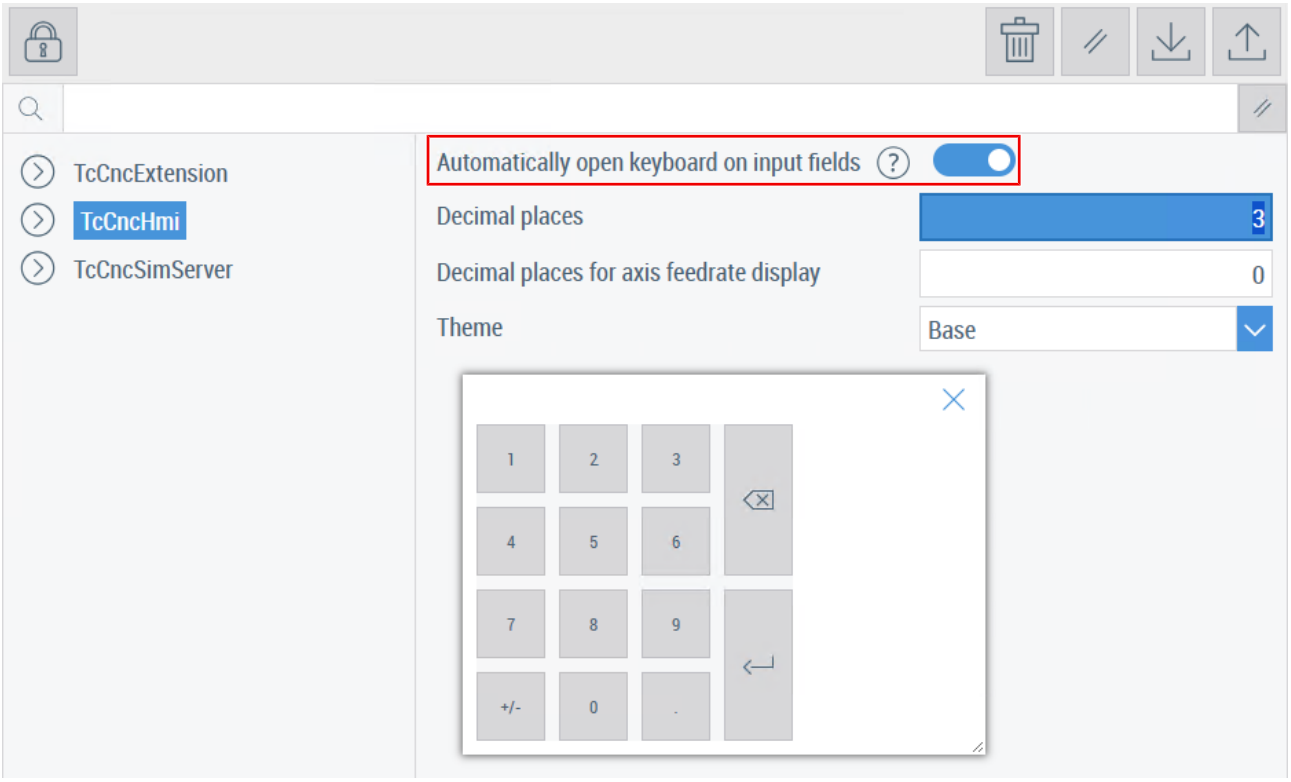
Configuration

The control `TcHmiCncSettings` requires a data binding to a symbol from the TcHmiCnc extension. Data bindings to the “CNC.Settings” symbol are permitted.

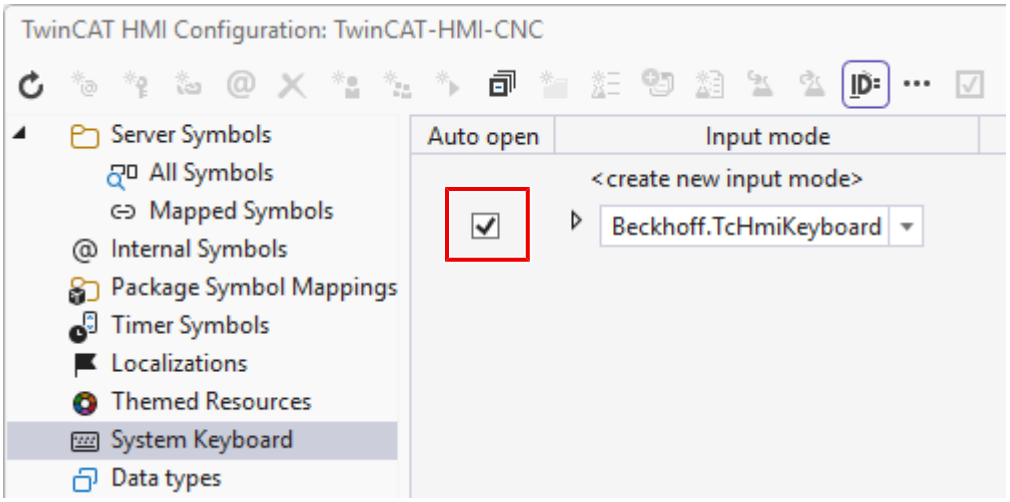


Virtual Keyboard

The HMI provides a virtual keyboard for data entry. In the control `TcHmiCncSettings`, you can specify whether a virtual keyboard appears when you click in an input field.



Make sure that “Auto open” is enabled for the “Beckhoff.TcHmiKeyboard” keyboard in the TwinCAT HMI Configuration. Unless this feature is enabled, the “Automatically open keyboard on input fields” setting has no effect.



User settings via TcHmiCncSettings

The control `TcHmiCncSettings` does not have any customizable settings in the control `TcHmiCncSettings`.

Keyboard shortcuts

The control `TcHmiCncSettings` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncSettings` has a public API that allows you to use the following functions.

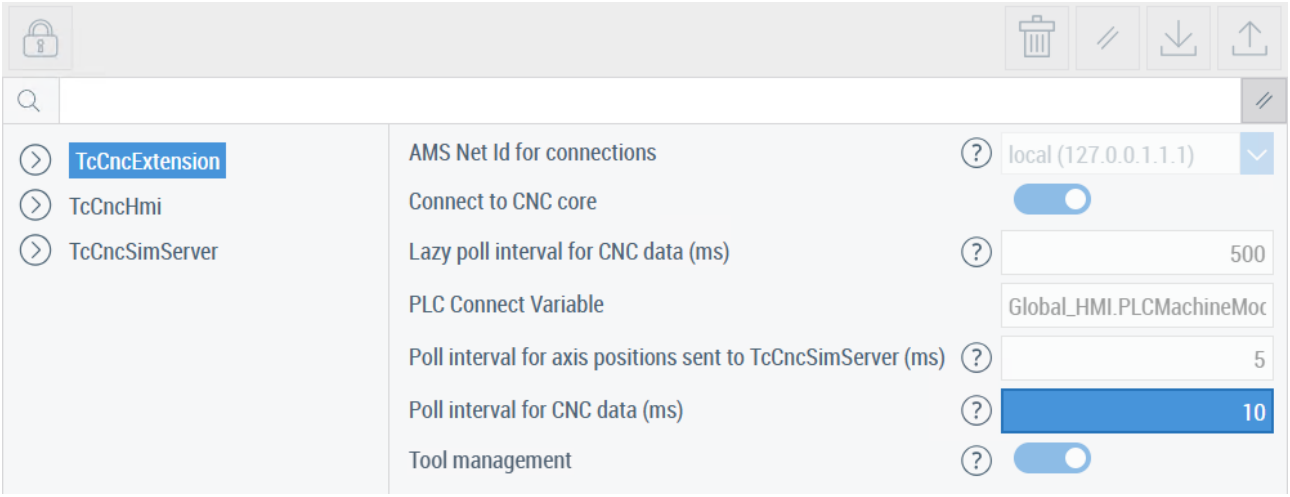
Function	Description
setJsonTree() / getJsonTree()	Set or read data binding for the TcHmiCnc extension. The settings are structured as JSON.
setHierarchical() / getHierarchical()	Set or read property names of the child node elements
setNameProp() / getNameProp()	Set or read the property names of an element's label
setValueProp() / getValueProp()	Set or read property names of the node value

Impact of user management

The control `TcHmiCncSettings` evaluates the "Operate" access right. For "Allow," the control behaves as usual. When set to "Disallow," the control is limited to the following:

- Navigating the tree structure and searching for settings
- Adjust settings if the user's user group has been granted access rights for that setting. The "Operate" access right is not required to adjust a setting.

In the example shown, the access permission for the user group is set to "Disallow". However, the user group was granted access rights for the "Poll interval for CNC data (ms)" setting. Users in this user group can therefore adjust the "Poll interval for CNC data (ms)" setting.



If you have not set up user management, "Operate" is considered "Allow," and the control behaves as usual.

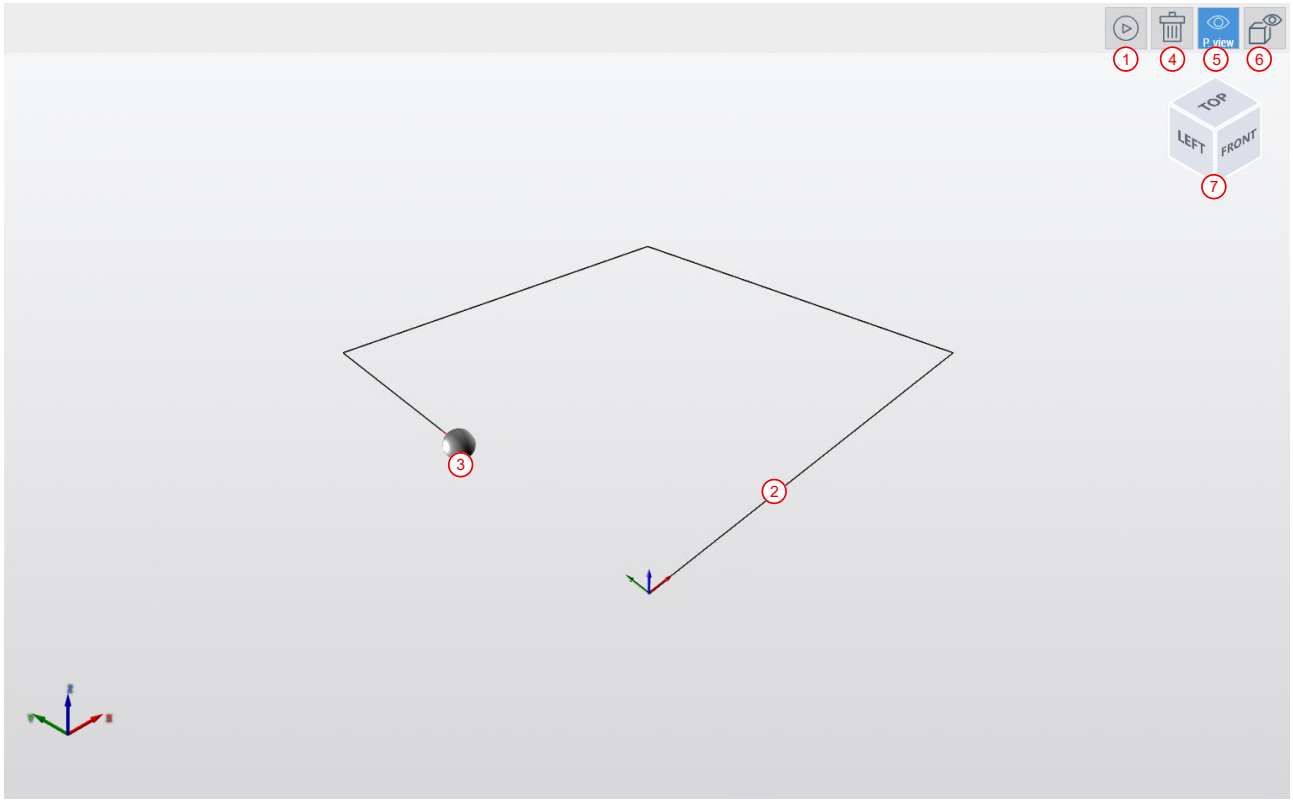
5.19 TcHmiCncSimulation

Component: `TcHmiCncSimulation`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

The control `TcHmiCncSimulation` displays a three-dimensional visualization of the tool path that the machine follows.

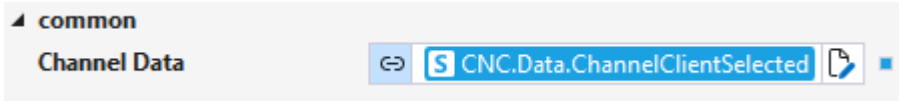
Operation



No.	Description
1	Button to start or stop recording the toolpath
2	Toolpath
3	Tool Center
4	Button to delete an existing toolpath
5	Button for switching between a perspective and an orthogonal view
6	Button to reset the camera to the default view
7	Navigation cube for changing the camera’s orientation

Configuration

The control `TcHmiCncSimulation` requires a data binding to a symbol from the TcHmiCnc extension. Data bindings to a symbol of type “CNC-ChannelType” are permitted.



Data Binding	Symbol
Channel Data	For example: • CNC.Data.ChannelClientSelected • CNC.Data.Channel.0

User settings via TcHmiCncSettings

Using the control TcHmiCncSettings, you can adjust the following settings:

Section	Setting	Standard	Description
TcCncHmi.TrackingView	ACS/PCS	PCS Position	Coordinate system of the recorded tool path
TcCncHmi.TrackingView	Enable tracking view	“`true`”	At `false`, the Start/Stop and Delete buttons are hidden.
TcCncHmi.TrackingView	Maximum number of vertices	15000	Maximum number of recorded support points for the toolpath. The oldest ones are discarded if the value has been exceeded.

Keyboard shortcuts

The control `TcHmiCncSimulation` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncSimulation` has a public API that allows you to use the following functions.

Function	Description
setChannelData / getChannelData	Set or read data binding for a symbol in the TcHmiCnc extension

Impact of user management

The control `TcHmiCncSimulation` does not exhibit any specific change in behavior as a result of the “Operate” access right.

5.20 TcHmiCncTechnoData

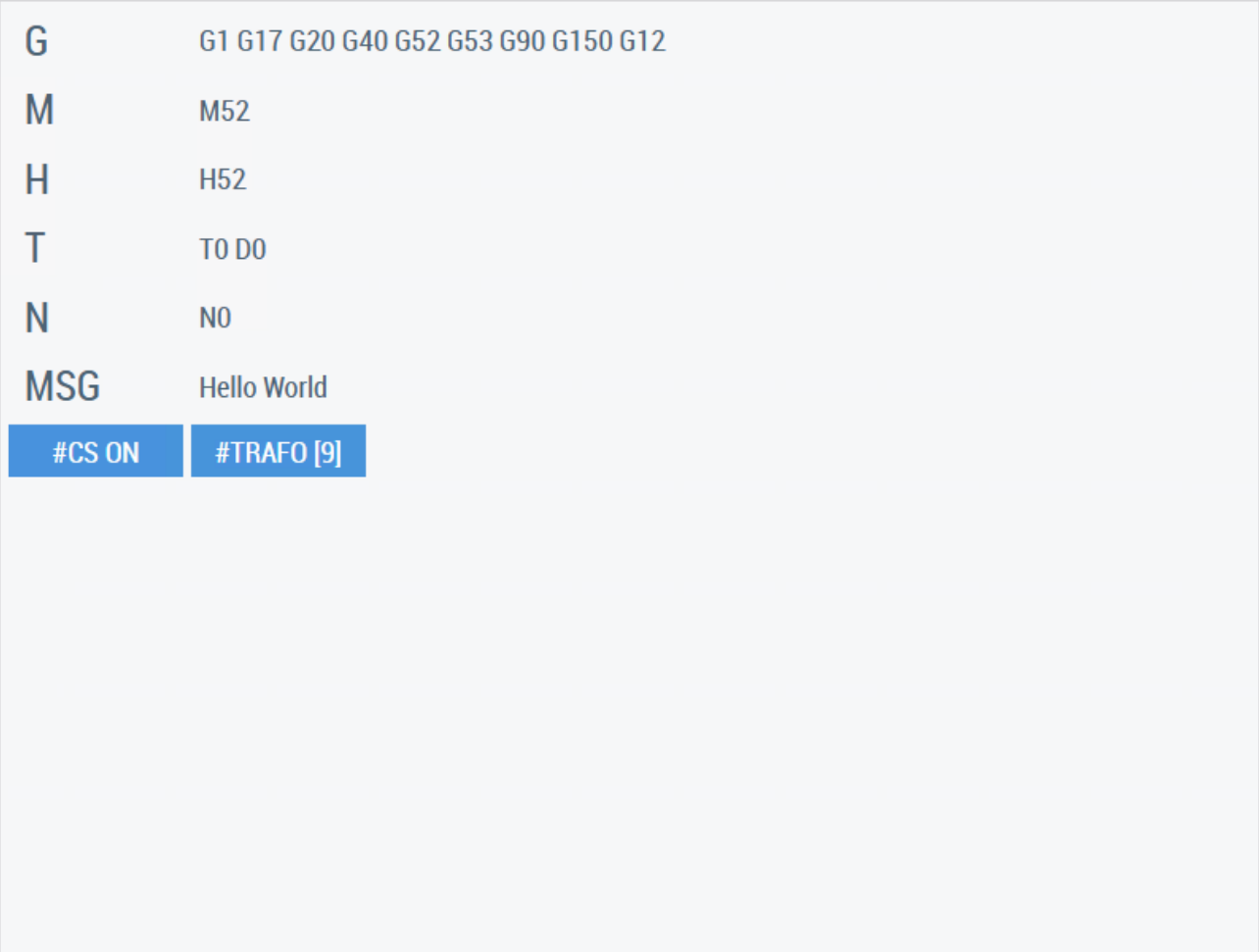
Component: `TcHmiCncTechnoData`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

The control `TcHmiCncTechnoData` displays the active technology functions (G functions, M-functions, H functions, T tool, N block number, and MSG message) for a CNC channel. You can show or hide each group of technology functions (G, M, H, T, N, MSG) individually. In addition, the system will indicate whether a kinematic transformation (#TRAFO[ID]) and a coordinate system (#CS) are active.

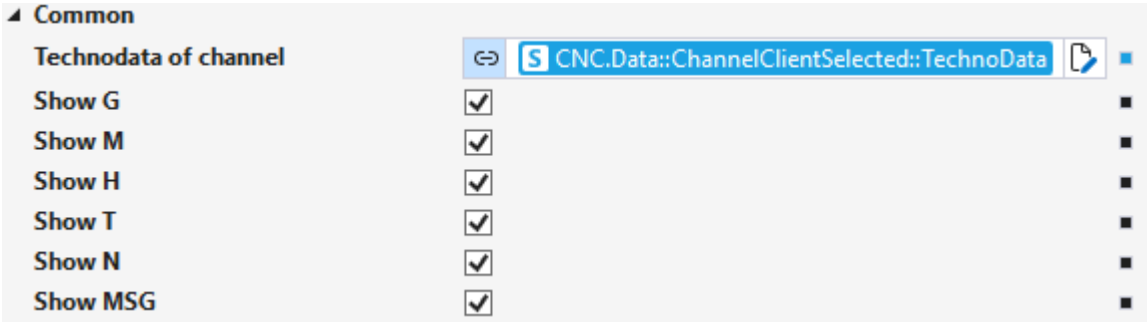
Operation

The control `TcHmiCncTechnoData` is read-only – there are no interactive controls.



Configuration

The control `TcHmiCncTechnoData` requires a data binding to a symbol from the TcHmiCnc extension. Data bindings to a symbol of type “CNC-TechnodataType” are permitted.



Data Binding	Symbol
Technodata of channel	Example • CNC.Data::ChannelClientSelected::TechnoData • CNC.Data.Channel.0.TechnoData

Use the checkboxes for each group of technology functions (G, M, H, T, N, MSG) to select whether they should be displayed.

User settings via TcHmiCncSettings

The control `TcHmiCncTechnoData` does not have any customizable settings in the control `TcHmiCncSettings`.

Key combinations

The control `TcHmiCncTechnoData` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncTechnoData` has a public API that allows you to use the following functions.

Function	Description
setTechnoData() / getTechnoData()	Set or read data binding for a symbol in the TcHmiCnc extension
setShowG() / getShowG()	Turn the G-function display on or off
setShowM() / getShowM()	Turn the M-function display on or off
setShowH() / getShowH()	Turn the H-correction display on or off
setShowT() / getShowT()	Turn the tool number display on or off
setShowN() / getShowN()	Turn the NC block number display on or off
setShowMSG() / getShowMSG()	Turn MSG output display on or off

Impact of user management

The control `TcHmiCncTechnoData` does not exhibit any specific change in behavior as a result of the “Operate” access right.

5.21 TcHmiCncTextEditor

Component: `TcHmiCncTextEditor`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

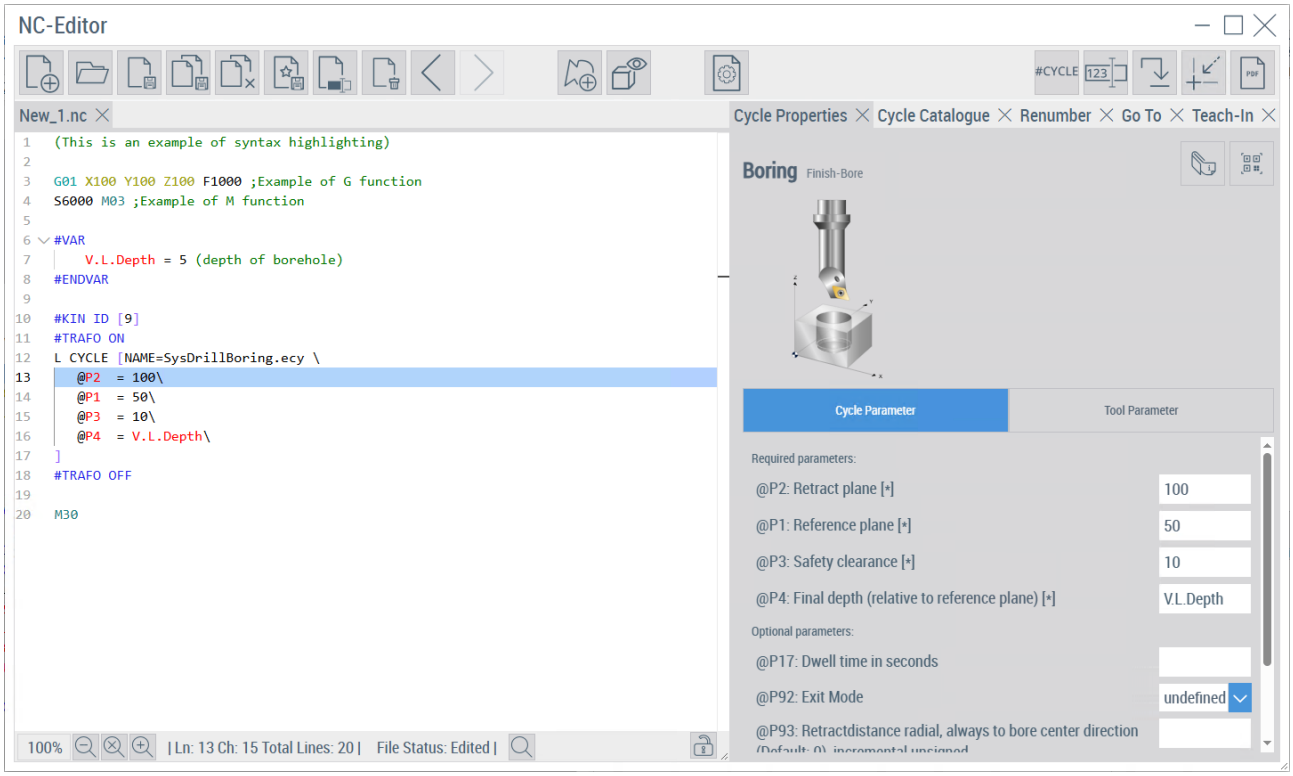
The control `TcHmiCncTextEditor` is a text editor for NC programs that offers the following functions:

- Text input
- Code completion and syntax highlighting
- Support:
  - Cycle programming
  - Numbering NC blocks
  - Search for selected programming commands
  - Teaching NC code
- File Operations (New NC Program, Save, Close, Open, etc.)
- Program selection
- G-Code viewer

You can use the editor in three different modes:

1. Full mode (Editor for creating NC programs – all functions available)
2. Automatic mode (Read-Only mode – Displays the line being executed)
3. MDI Mode (Text Input Only)

Full mode



In full mode, all of the editor's functions are available to you.

Automatic Mode

```
(This is an example of syntax highlighting)

G01 X100 Y100 Z1000 F1000 ;Example of G function
S6000 M03 ;Example of M function

#VAR
    V.L.Depth = 5 (depth of borehole)
#ENDVAR

#KIN ID [9]
#TRAFO ON
L CYCLE [NAME=SysDrillBoring.ecy \
   @P2  = 100\
   @P1  = 50\
   @P3  = 10\
   @P4  = V.L.Depth\
]
#TRAFO OFF

M30
```

In automatic mode, the selected NC program is displayed, with the line currently being executed highlighted in color. Double-clicking on the NC program opens it in the editor.

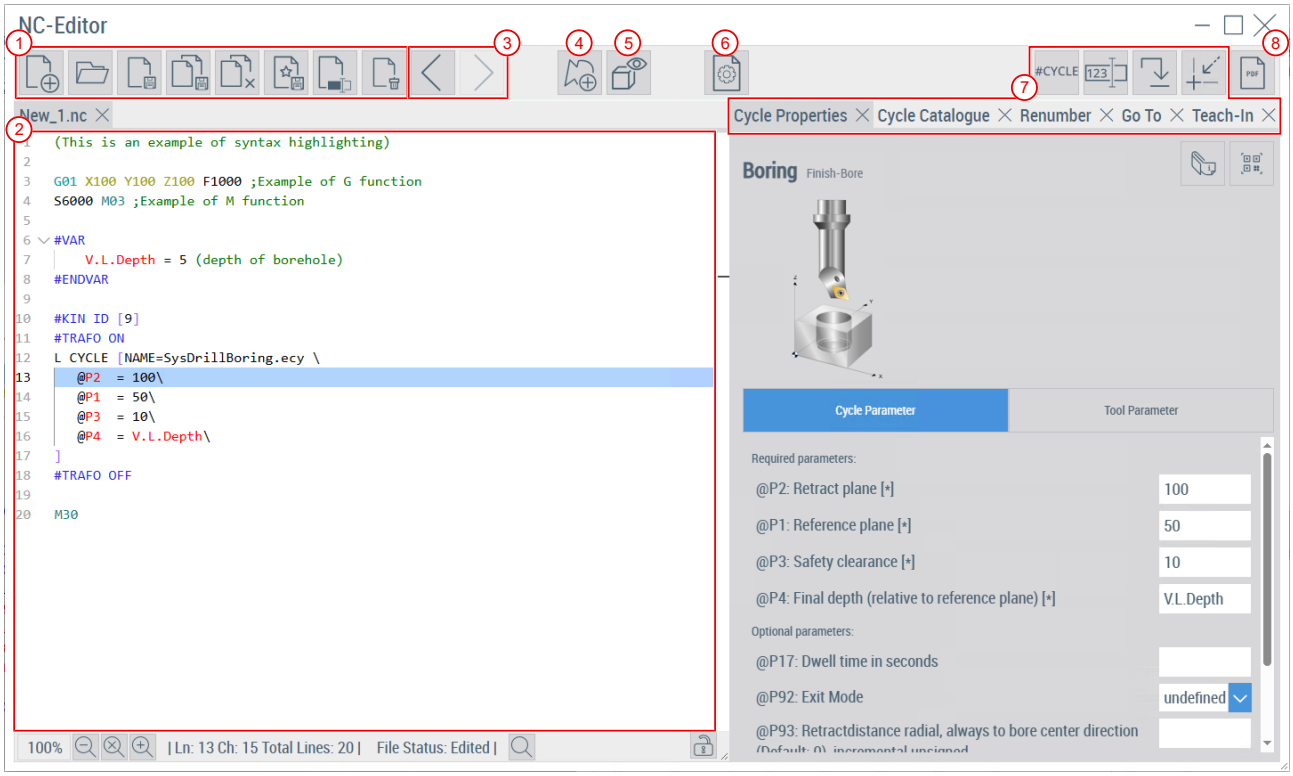
MDI Mode

```
(Only text input is supported)

G01 X100 Y100 Z100 F1000

M30
```

In MDI mode, only text entry is available.

Operation

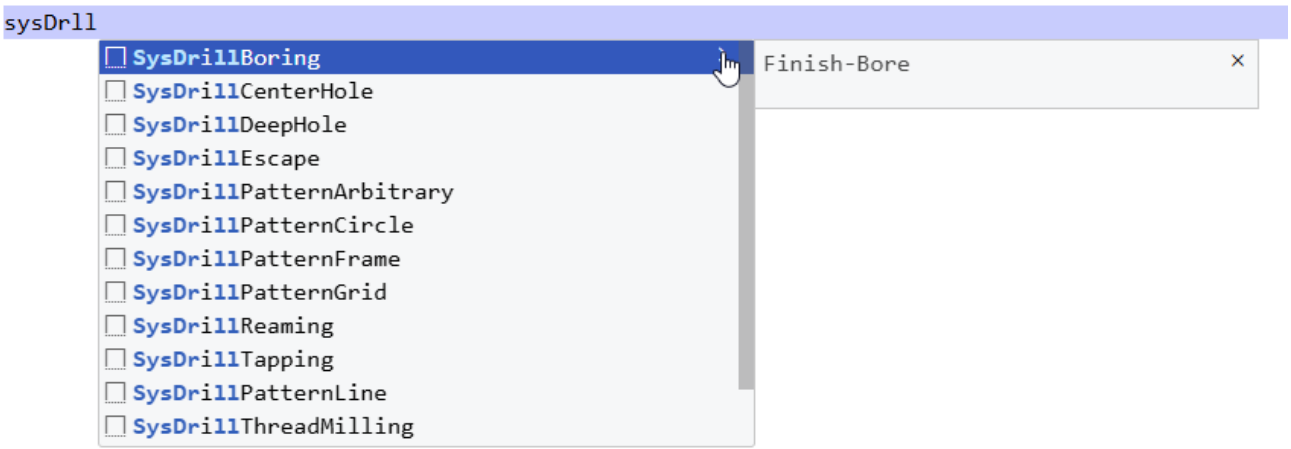
The controls for `TcHmiCncTextEditor` are organized into several sections, which will be introduced first.



No.	Description
1	Buttons for file operations (Create New NC Program, Save, Close, Open, etc.)
2	Area for editing the NC program
3	Undo or repeat changes to the NC program
4	Button to open the contour workbench via the control TcHmiCncContourWorkbench [▶ 60].
5	Button to open the G-Code viewer
6	Selecting the NC program
7	Buttons for opening the corresponding tabs in the sidebar. Each tab contains support functions for creating the NC program.
8	Opening a configurable PDF file (e.g., a programming guide)

Autocomplete and syntax highlighting

The editor provides code completion. As soon as you start typing, possible suggestions will appear. Click one of the suggestions to select that command. Alternatively, use the arrow keys to select the appropriate command and then press Tab. A description is also displayed for each command. A description will also appear when you hover over the command.



The editor provides syntax highlighting, which highlights keywords in color. You can control the individual colors using CSS variables. To do this, follow the instructions in the chapter Styling and themes [▶ 25]. The following CSS variables are available:

Variable name	“Base” theme	“Base-Dark” theme	Use	Example
--TcHmiCncTextEditor_SyntaxHighlighting_CNC-Comment	#007700	#007700	Comments	;Comment
--TcHmiCncTextEditor_SyntaxHighlighting_CNC-String	#FF0000	#ea8055	String literals	"HelloWorld"
--TcHmiCncTextEditor_SyntaxHighlighting_CNC-Sub	#A0A0A0	#D4D4D4	Subroutine references	%L Sub
--TcHmiCncTextEditor_SyntaxHighlighting_CNC-Main	#A0A0A0	#D4D4D4	Key program references	%MAIN
--TcHmiCncTextEditor_SyntaxHighlighting_CNC-Dollar	#AF00DB	#e69de0	$ System variables	$IF
--TcHmiCncTextEditor_SyntaxHighlighting_CNC-Techno	#1b8080	#1b8080	Technology commands, M commands	M30
--TcHmiCncTextEditor_SyntaxHighlighting_CNC-Hashtag	#2222ea	#6a71ff	Macro references (#)	#TRAFO ON
--TcHmiCncTextEditor_SyntaxHighlighting_CNC-Variable	#FF0000	#7fd0fc	Variables, parameters	V.L.
--TcHmiCncTextEditor_SyntaxHighlighting_CNC-GCommand	#00A0A0	#3fdddd	G-Code commands	G01
--TcHmiCncTextEditor_SyntaxHighlighting_CNC-Axis	#A0A000	#ebeb3b	Axis designators (X, Y, Z)	X100
--TcHmiCncTextEditor_SyntaxHighlighting_CNC-BlockNumber	#888888	#858585	Block numbers	N100

Cycle programming



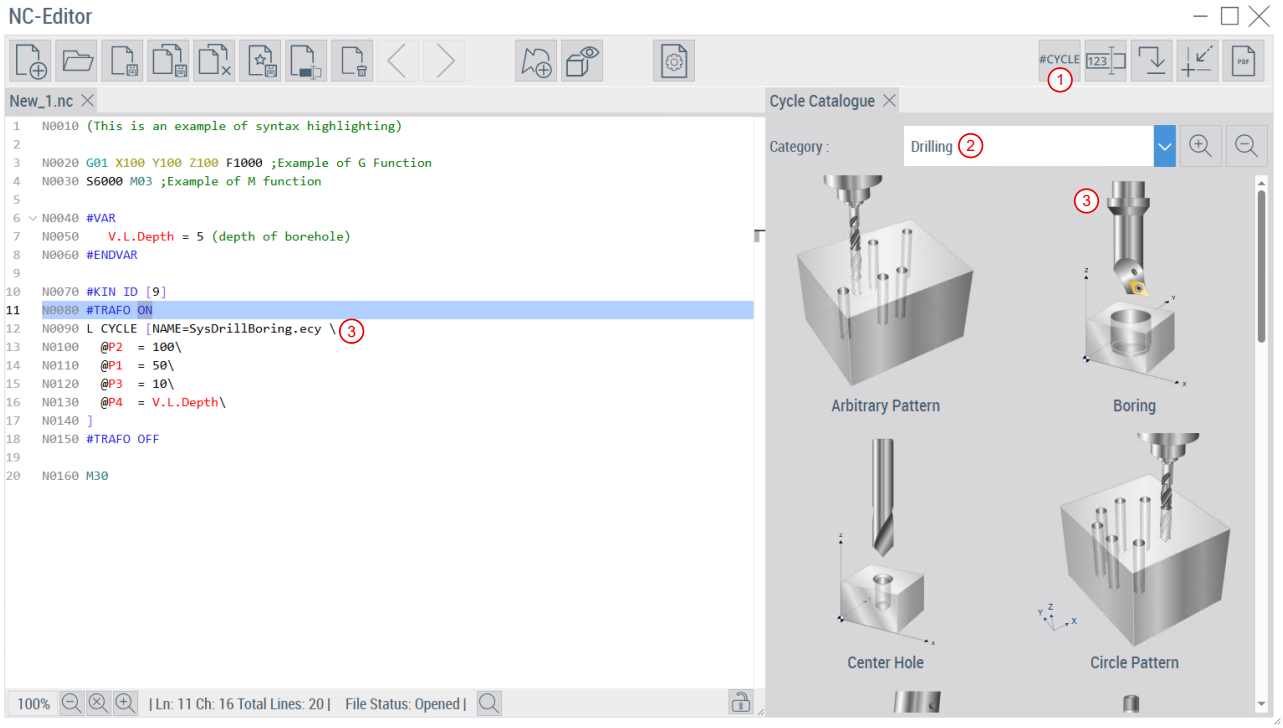
Cycles are an optional add-on that requires a license.

The editor helps you use cycles.

1. Click the “#Cycle” button.
   ⇨ The “Cycle Catalogue” tab will then appear in the sidebar.

2. Use the combo box to select a category for the cycles.
3. Use drag and drop to add the desired NC cycle to your NC program.

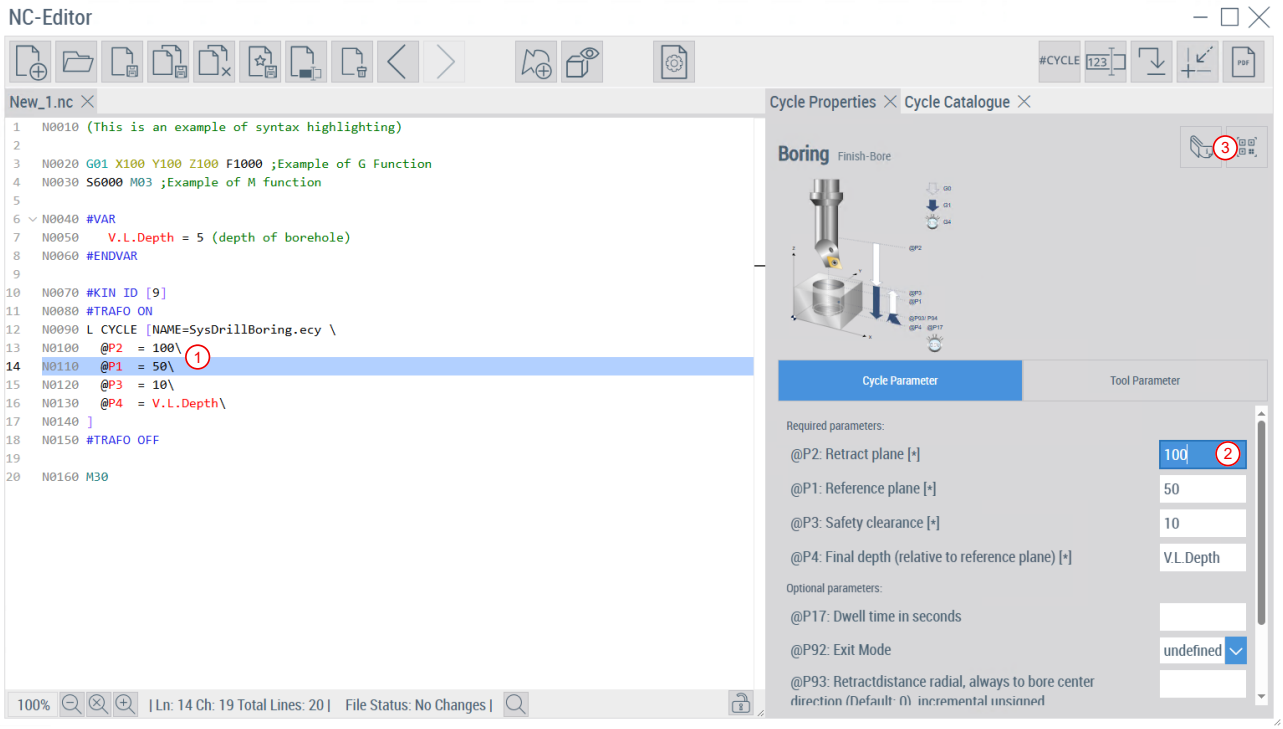
⇨ The cycle will be inserted into your NC program.



1. Click on a row in the cycle.

   ⇨ An additional tab titled “Cycle Properties” will open, allowing you to adjust the cycle parameters.
2. Adjust the cycle parameters. Changes to the parameters are applied as soon as you leave an input field.
3. Optional: Click one of the buttons to access the documentation for the cycle.

⇨ You have configured the cycle.

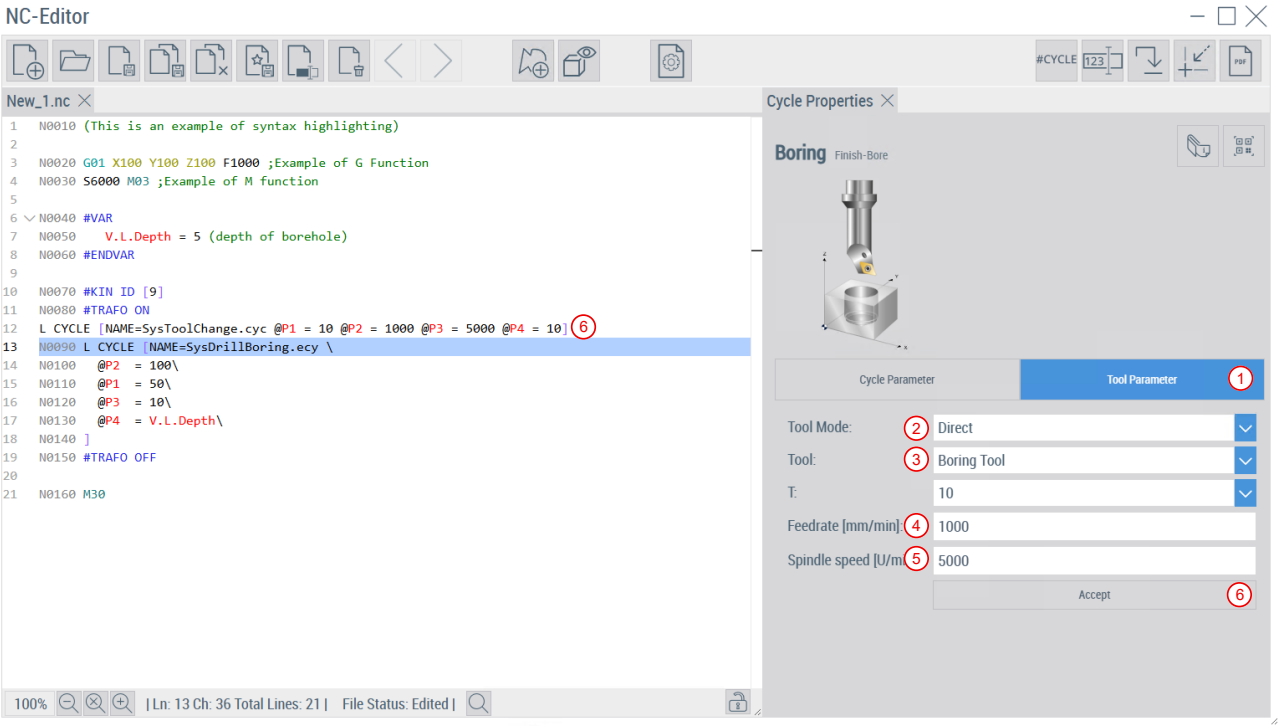


✓ You have configured the cycle.

1. Click the button to configure the tool.

2. For the tool, specify whether the feed rate (4) and spindle speed (5) are specified directly or indirectly based on the cutting speed and tooth feed.
3. Select a tool from the combo box.
4. Set the feed rate.
5. Set the spindle speed.
6. Click “Accept” to insert the SysToolChange.cyc cycle into the NC program. This cycle performs the tool change. The file name of the cycle can be set via the control `TcHmiCncSettings`.

⇨ You have configured a tool that is changed before the cycle begins.



SysToolChange.cyc is a cycle that you must program specifically for your machine and save in the NC program folder on your machine (e.g., C:\cnc). When doing so, be sure to perform the programming of cycles as global subroutines. The following parameters are passed to the cycle:

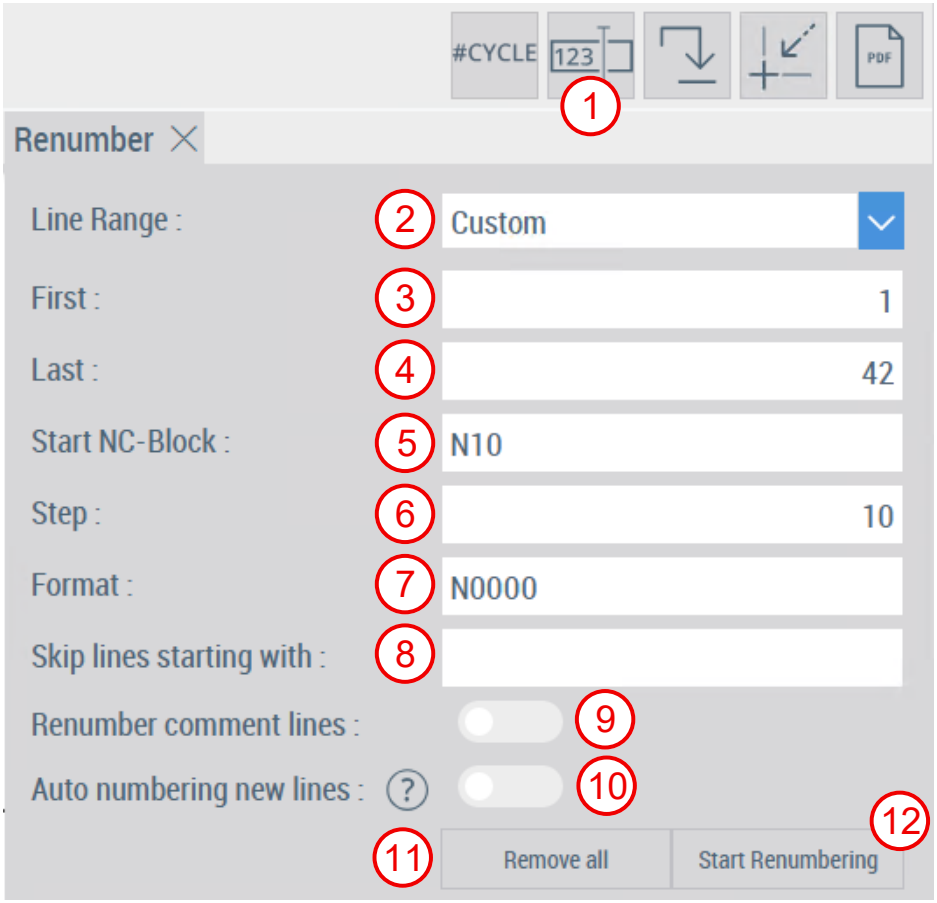
```
L CYCLE [NAME = SysToolChange.cy. @P1=<T-Word> @P2=<Feedrate> @P3=<Spindle speed> @P4=<D-Word>]
```

Numbering NC blocks

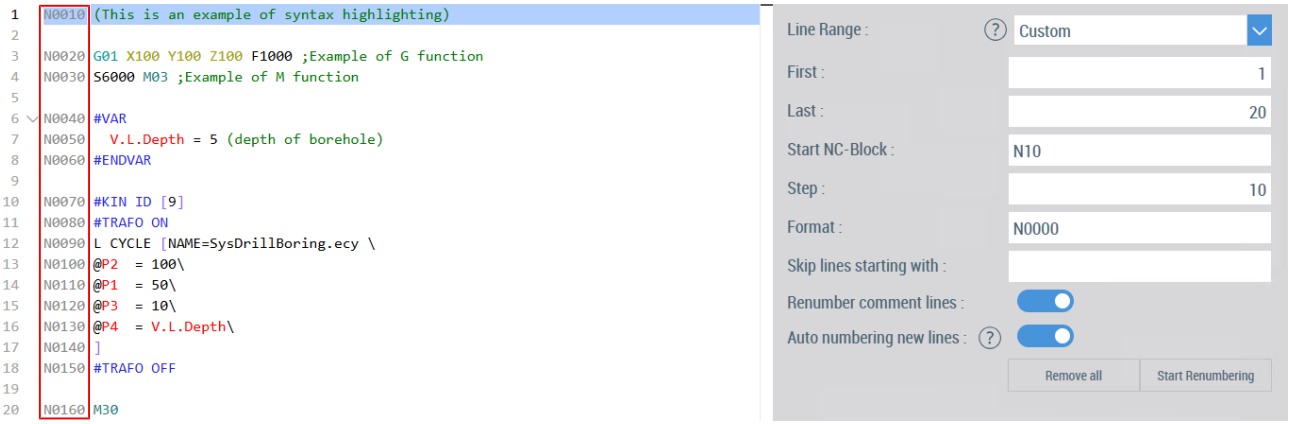
The editor allows you to automatically number NC blocks.

1. Click the “Renumber” button.
   ⇨ The “Renumber” tab will then appear in the sidebar.
2. Select “Custom” from the combo box.
3. Specify the first line where you want to insert numbering.
4. Specify the last line up to which you want to insert numbering.
5. Specify the starting value for the numbering in the first row.
6. Define the numbering steps.
7. Specify how many digits each numbering has. In the example shown, the numbering starts with “N0010”.
8. Optional: Specify a string. Lines that begin with this string are not numbered.
9. Specify whether comments should be numbered.
10. Specify whether new lines should receive automatic numbering.
11. Optional: Delete an existing numbering. Deletion always applies to all lines of the NC program.
12. Start the numbering

⇨ You have inserted a new numbering system.



As a result, the numbering is inserted at the beginning of the lines. Blank lines are not numbered.



Checking references

You will overwrite the existing numbering. Therefore, before running the NC program, check for any references to numbered NC blocks.

In addition to “Custom,” the combo box offers other options based on the custom function. The behavior of each selection option is described in the table below.

Combo box selection	Description
All Lines	All lines of the NC program will be renumbered.

Current to Last	The NC program is renumbered starting from the line where the cursor is located. The numbering starts with the line number where the cursor is located, multiplied by the numbering increment.
	Example:
	`3 G01 X100 Y100 Z100 F1000 ;Example of G function`
	`4 N0040 S6000 M03 ;Example of M function`
	`5`
	`6 N0050 #VAR`
	`7 N0060 V.L.Depth = 5 (depth of borehole)`
	`8 N0070 #ENDVAR`

First to current	The NC program is renumbered up to the line where the cursor is located. Numbering begins with the numbering increment.
	Example:
	`1 N0010 (This is an example of syntax highlighting)`
	`2`
	`3 N0020 G01 X100 Y100 Z100 F1000 ;Example of G function`
	`4 N0030 S6000 M03 ;Example of M function`
	`5`
	`6 #VAR`
	`7 V.L.Depth = 5 (depth of borehole)`
	`8 #ENDVAR`

Selection	The selected lines will be renumbered in the NC program. Numbering begins with the numbering increment.
	Example:
	`1 (This is an example of syntax highlighting)`
	`2`
	`3 N0010 G01 X100 Y100 Z100 F1000 ;Example of G function`
	`4 N0020 S6000 M03 ;Example of M function`
	`5`
	`6 N0030 #VAR`
	`7 V.L.Depth = 5 (depth of borehole)`
	`8 #ENDVAR`

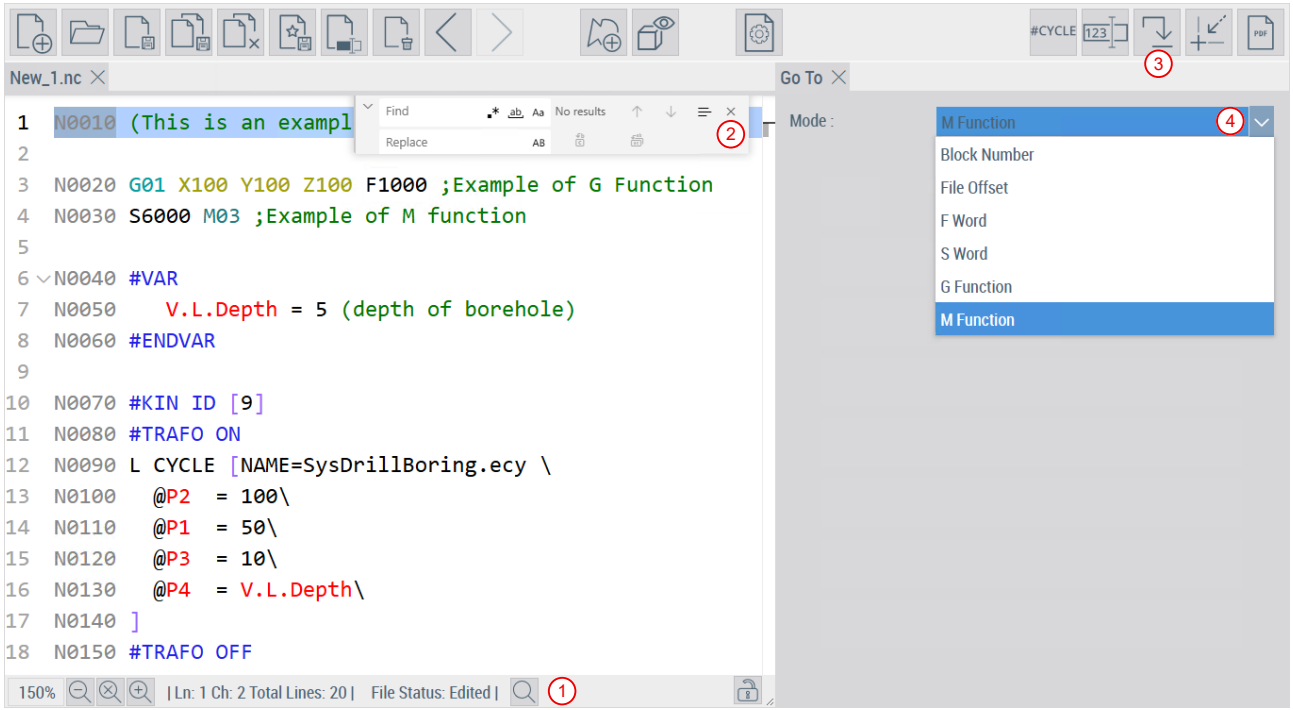
Custom	User-specific numbering.

Search for predefined programming commands (Go To)

The editor has a “Find and replace” function.

1. Click the search button.
   ⇨ A dialog opens.
2. Start your search using the dialog box. For complex search queries, you can use the syntax of the Regular Expression Language. In addition, a predefined search is available for selected programming commands.
3. Alternatively: Click on the “Go To” button.
   ⇨ The “Go To” tab will then appear in the sidebar.
4. Use the combo box to select the appropriate search.

⇨ You have selected a search function.



The following search functions are available:

Search function	Description
Block Number	Enter the block number you want to jump to.
File Offset	Enter the cursor position in the file where you want to jump to.
F Word	Allows you to jump between F-words (feed rates).
S Word	Allows you to jump between S-words (spindle speed).
G Function	Allows you to jump between G-functions in the NC program.
M Function	Allows you to jump between M-functions in the NC program.

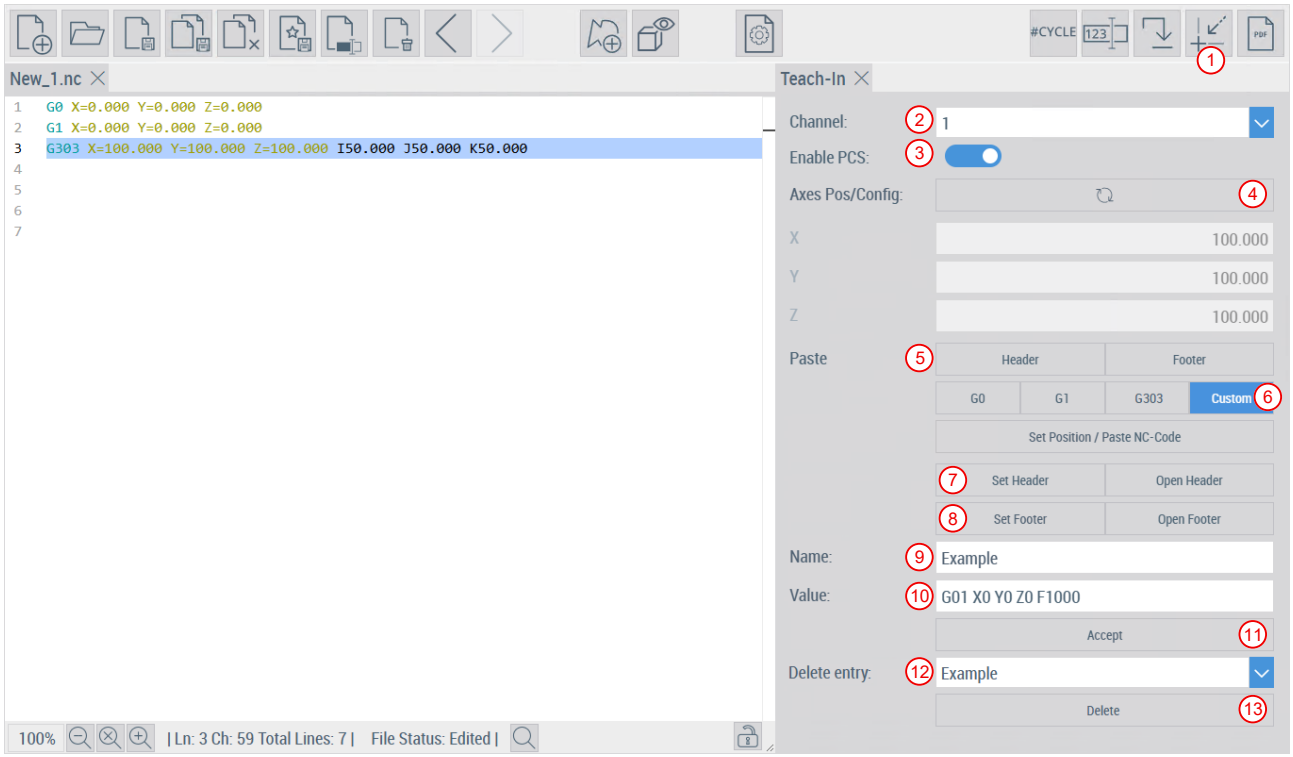
Teaching NC code

The Teach-In function allows you to incorporate manually moved positions of your machine axes into an NC program. You can add axis positions using linear movements (G0, G1) as well as a circular motion (G303). The axis positions are entered as absolute values (G90).

1. Click the “Teach-In” button.
   ⇨ The “Teach-In” tab will then appear in the sidebar.
2. Use the combo box to select the channel you want to program.
3. Select whether to use the axis positions in the axis coordinate system (ACS) or the part coordinate system (PCS).
4. Click the button to select the axes in the channel that you want to use for programming.
5. Optional: Insert a predefined NC program as a header or footer.
6. Click “Custom” to configure the header or footer, if necessary.
7. Optional: Define the header.
   ⇨ A pop-up window will then open, allowing you to select the NC program.
8. Optional: Define the footer.
   ⇨ A pop-up window will then open, allowing you to select the NC program.
9. You can create custom buttons for recurring commands. Please enter a name.
10. Define the NC code to be inserted using the user-defined button.
11. Click on “Accept”.
    ⇨ You can find the user-specific buttons in the “G0” and “G1” tabs.
12. Use the combo box to select a user-specific button that you want to delete.

13. Click “Delete” to delete the selected custom button.

⇨ You have configured the Teach-In function.

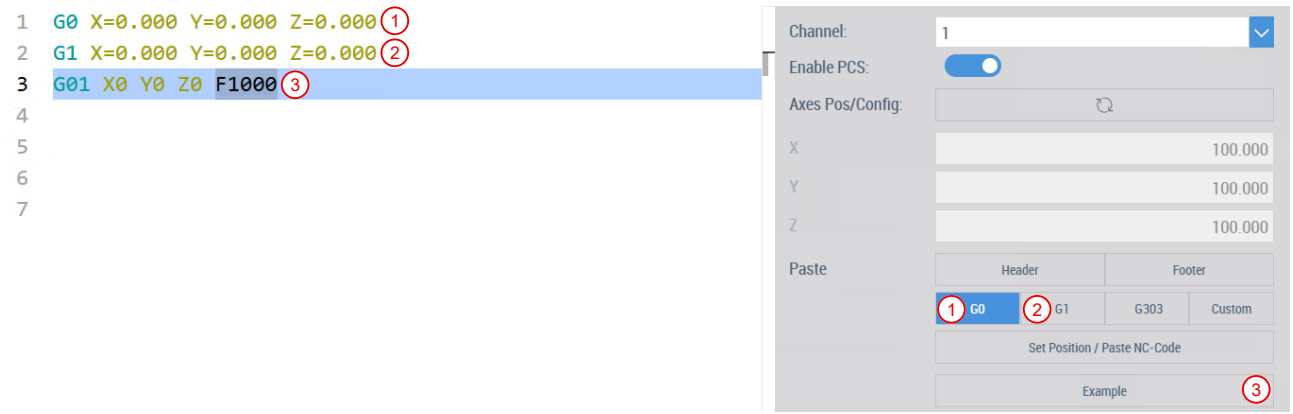


✓ You have manually moved the machine to the desired axis position (for example, in “MDI” or “Manual” operation modes).

1. Click the “G0” button to insert the current axis position into the NC program as a G0 command.
2. Click the “G1” button to insert the current axis position into the NC program as a G1 command.

   ⇨ The commands are inserted into the line where the cursor is located, and a line break is also inserted.
3. Click the “Example” button. This button is an example of a user-defined command.

⇨ You have taught the axis positions.



In addition to linear movements (G0, G1), you can also add a circular motion (G303). This requires that the first three axes in the channel be feed axes. The G303 command requires a starting point (Start), an interim point (Mid), and an end point (End). Set these three points one after another using the teach-in function.

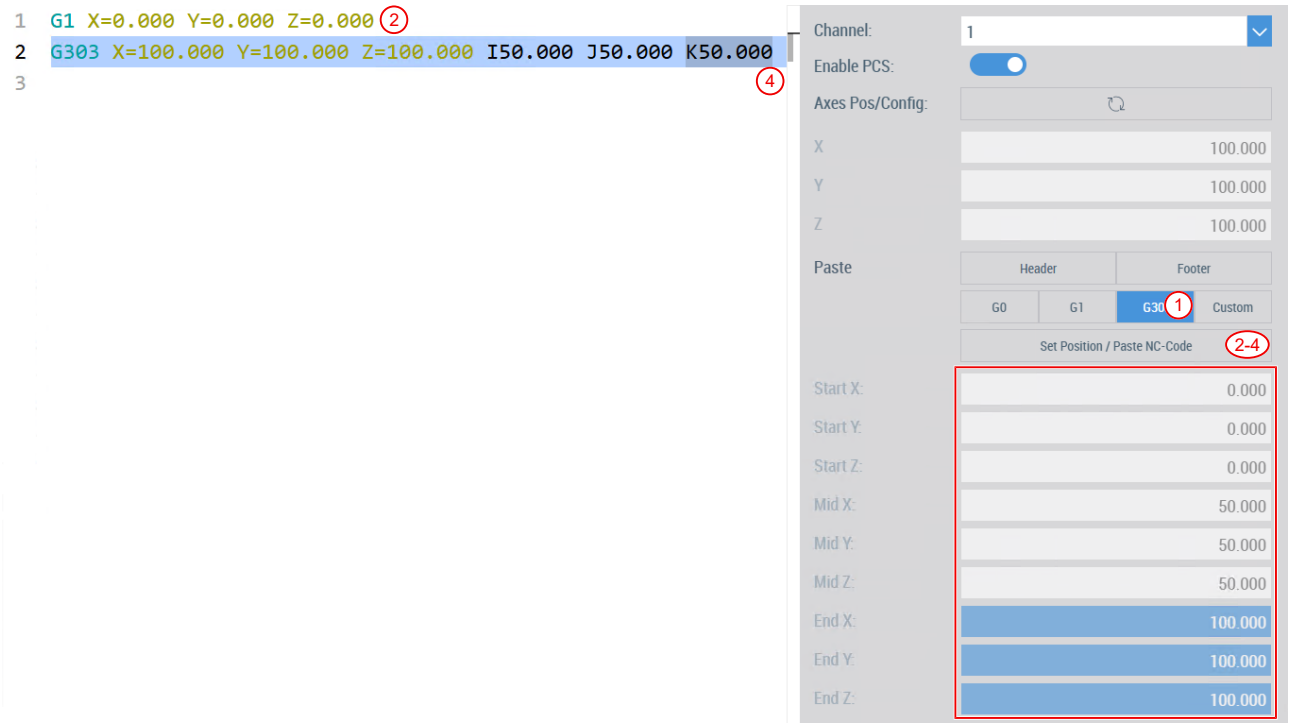
1. Click on “G303”

   ⇨ The coordinates for the start, interim, and end points are displayed in the highlighted area. The point you need to click next is highlighted in color.
2. Manually move the machine to the desired starting point (for example, in “MDI” or “Manual” operation mode). Then click “Set position / Paste NC Code”.

   ⇨ The starting position is set in the selected area and inserted into the NC program as a G1 command.

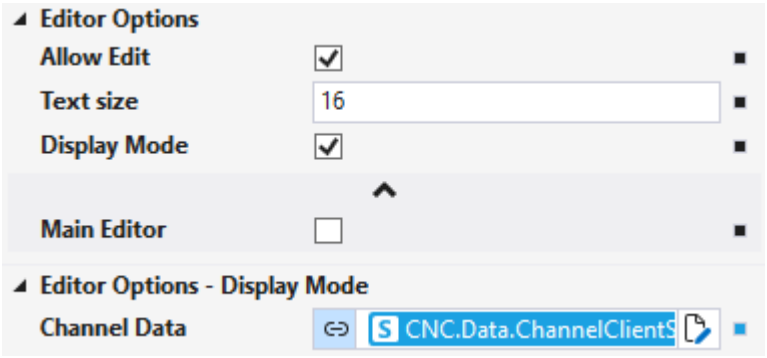
3. Manually move the machine to the desired interim point and click “Set Position / Paste NC Code” again.
   ⇨ The interim point is set in the selected area.
4. Manually move the machine to the desired end point and click “Set position / Paste NC Code” again.

⇨ The end position is set in the selected area. The interim and end positions are also included in the NC program as a G303 command.



Configuration

Configure the control `TcHmiCncTextEditor` as follows.



Allow Edit

Use this setting to specify whether the displayed NC program can be edited. Otherwise, the file may only be opened, renamed, or saved as a copy.

Text size

Use this setting to adjust the font size in the editor.

Display Mode

In Display mode, the editor is shown without a menu bar. Data binding for the display mode is used to determine the operating mode (Automatic or MDI), and the editor is displayed accordingly in either Automatic Mode or MDI Mode.

G-code generator

With this setting, you define a TypeScript class that you use as a custom G-Code generator (see TcHmiCncContourWorkbench [▶ 60])

Main Editor

In an HMI, you can use multiple controls of the `TcHmiCncTextEditor` type. In addition, other controls can call the editor (for example, when opening a file). Use this setting to specify which editor to open. The main editor is always launched first.

Channel Data

In display mode, the control requires a data binding to a symbol from the TcHmiCnc extension. Data bindings to a symbol of type “CNC-ChannelType” are permitted.

Data Binding	Symbol
Channel Data	Example • CNC.Data.ChannelClientSelected • CNC.Data.Channel.0

CSS Variables in the editor

In addition to the CSS variables for syntax highlighting, there are other CSS variables available that you can use to customize the editor’s appearance. To do this, follow the instructions in the chapter Styling and themes [▶ 25]. The following CSS variables are available:

Variable name	“Base” theme	“Base-Dark” theme	Use
--TcHmiCncTextEditor_Coloring_background-unlocked	#FFFFFF	--CNC-HMI-Input-Background	Background in editable state
--TcHmiCncTextEditor_Coloring_background-locked	#D0D0D0	#A4A4A4	Background in read-only state
--TcHmiCncTextEditor_Coloring_SelectedLine-Background	#0064FF4D	#0064FF4D	Background of the active row
--TcHmiCncTextEditor_Coloring_LineNumber	#A4A4A4	#D0D0D0	Inactive line numbers
--TcHmiCncTextEditor_Coloring_ActiveLineNumber	#000000	#FFFFFF	Active line number
--TcHmiCncTextEditor_Coloring_CursorColor	#000000	#FFFFFF	Cursor color
--TcHmiCncTextEditor_Coloring_NormalTextColor	#000000	#FFFFFF	Default text color
--TcHmiCncTextEditor_Coloring_BracketHighlight1	#9966FF	#FF8800	Highlighting pairs of parentheses, Level 1
--TcHmiCncTextEditor_Coloring_BracketHighlight2	#00FF00	#00FF00	Highlighting pairs of parentheses, Level 2
--TcHmiCncTextEditor_Coloring_BracketHighlight3	#FF8800	#96FAFF	Highlighting pairs of parentheses, Level 3
--TcHmiCncTextEditor_Coloring_BracketHighlight4	#FF00FF	#FF00FF	Highlighting pairs of parentheses, Level 4
--TcHmiCncTextEditor_Coloring_BracketHighlight5	#00CC66	#00CC66	Highlighting pairs of parentheses, Level 5
--TcHmiCncTextEditor_Coloring_BracketHighlight6	#FFFF00	#FFFF00	Highlighting pairs of parentheses, Level 6
--TcHmiCncTextEditor_Coloring_UnexpectedBracket	#FF0000	#FF0000	Unassigned bracket

User settings (Settings)

All settings are saved in the global CNC HMI settings memory (`CNC.Settings`) and are retained across sessions.

Section	Key	Standard	Description
TcCncHmi.NC-Editor	Auto-numbering	“true”	Automatic line numbering when inserting new lines
TcCncHmi.NC-Editor	Minimize Editor after program selection	“true”	In “true,” the editor that is replaced is minimized after the program is selected
TcCncHmi.NC-Editor	Prefer detached Editor	“true”	When “true” is selected, the main editor is automatically detached into a separate window when a file is opened
TcCncHmi.NC-Editor	PdfFilePath	“”	PDF file that opens when you click the PDF button
TcCncHmi.NC-Editor	ToolChangeCycleName	SysToolChange	Name of the tool change cycle
TcCncHmi.NC-Editor	Word-based suggestions	“true”	Word-based suggestions in the editor
TcCncHmi.NC-Editor.Teach-In	Auto-paste	“true”	Insert Headers and Footers Automatically
TcCncHmi.NC-Editor.Teach-In	Footer path	“”	NC file that is appended as a footer
TcCncHmi.NC-Editor.Teach-In	Header path	“”	NC file that is inserted as a header
TcCncHmi.NC-Editor.Teach-In	PLC trigger variable	“”	PLC variable used to trigger the teach-in function (rising edge)

Keyboard shortcuts

You can use the following keyboard shortcuts in the control `TcHmiCncTextEditor`.

Shortcut	Action
Alt+N	Create new file
Alt+O	Open file
Ctrl+S	Save active file
Ctrl+Shift+S	Save all files
Ctrl+Alt+W	Close all files
Ctrl+Alt+S	Save as
F2	Rename file
Alt+D	Delete file
Ctrl+Z	Undo
Ctrl+Y	Redo
Ctrl+F	Open Find/Replace
Tab	Accept AutoComplete
Ctrl+Alt+Enter	Replace all (in the find and replace function)

API functions

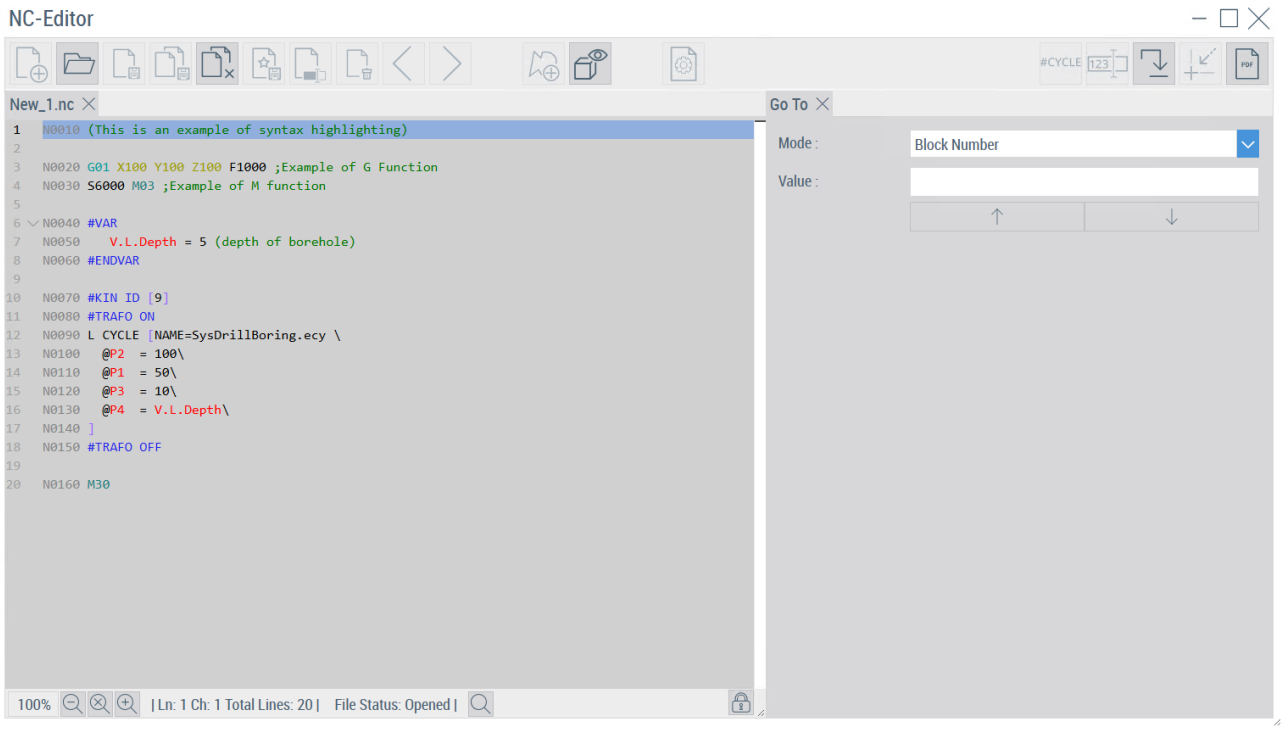
The control has a public API that allows you to use the following functions.

Function	Description
setAllowEdit() / getAllowEdit()	Set or read the “Allow Edit” setting
setMainEditor() / getMainEditor()	Set or read the “Main Editor” setting
setTextSize() / getTextSize()	Set or read the “Text size” setting
setDisplayMode() / getDisplayMode()	Set or read the “Display Mode” setting
setDataSymbol() / getDataSymbol()	Set or read the “Data Symbol” setting

Impact of user management

The control `TcHmiCncTextEditor` evaluates the “Operate” access right. For “Allow,” the control behaves as usual. When “Disallow” is selected, the control is limited to the following:

- Open NC program (read-only)
- G-Code view
- Go To Sidebar
- Open PDF
- Opening files in contour workbench (read-only)



If you have not set up user management, “Operate” is considered “Allow,” and the control behaves as usual.

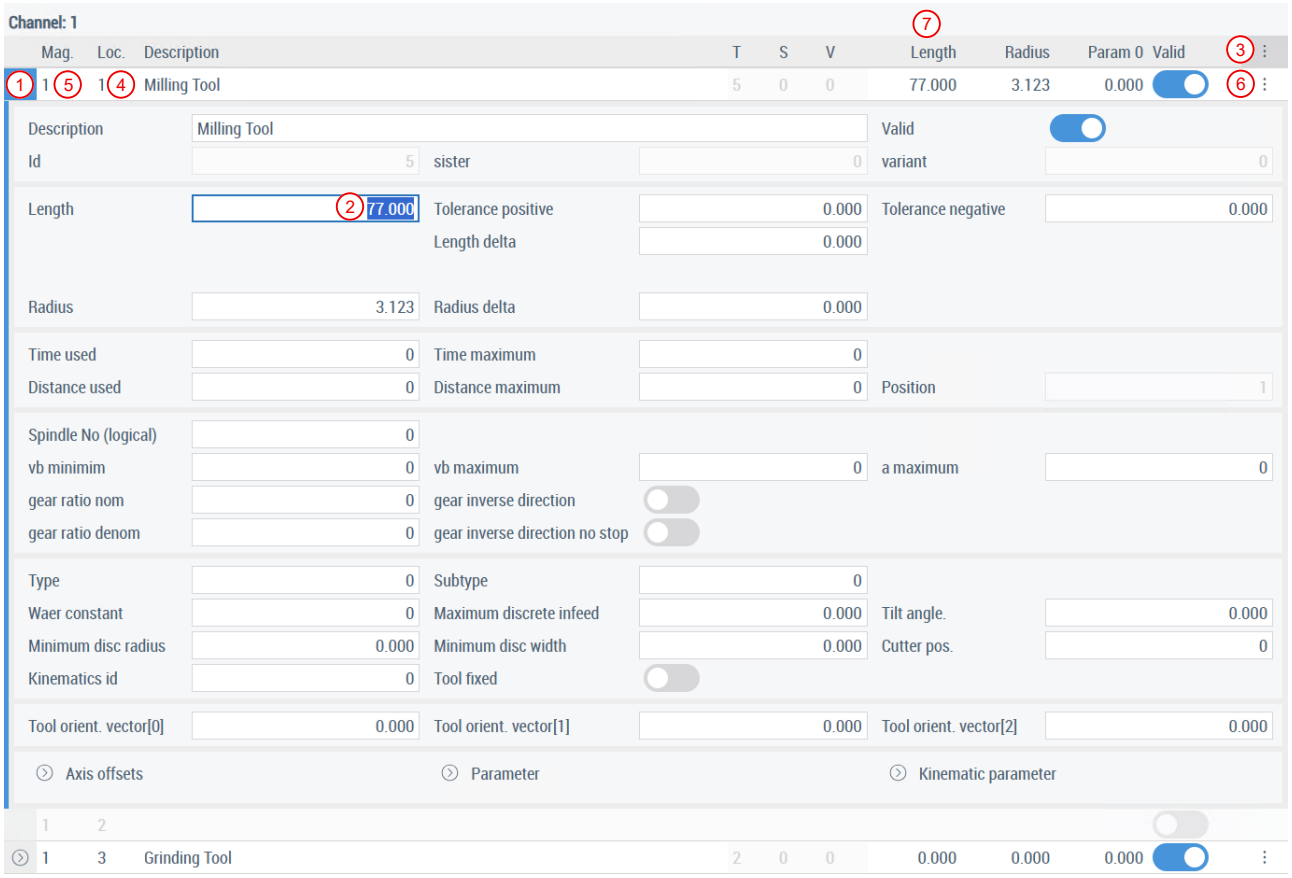
5.22 TcHmiCncToolList

Component: TcHmiCncToolList
Namespace: TcHmi.Controls.TcHmiCncBase
Available since version: 1.0.0

Description

The control TcHmiCncToolList displays the tool data for a CNC channel in a table. It also allows you to manage and customize existing tools, as well as create new ones.

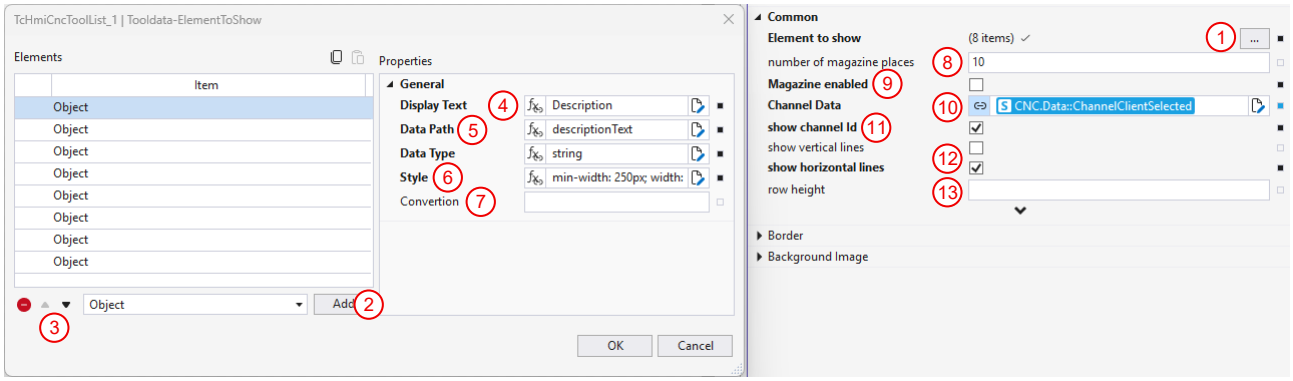
Operation



No.	Description
1	Button to view detailed tool data for a tool.
2	Input field for tool data. Changes are saved automatically as soon as you leave the input field.
3	Button to create a new tool and to download (export) or upload (import) the entire tool table. During import, the existing tool table will be overwritten.
4	Space in tool magazine. A new tool is not initially assigned to a location and is given the location number “-1”.
5	Tool magazine number. A new tool is not initially assigned to any magazine and is assigned the magazine number “-1”.
6	Button to change a tool’s ID or delete a tool. A tool’s ID consists of three components: Basic (T), Sister (S), and Variant (V).
7	Individually configurable column of the tool table

Configuration

You can customize the columns displayed in the table.



1. Click on the button “…”.
   ⇨ An additional window appears
2. If necessary, use “Add” to add additional objects. Each object corresponds to a column in the table.
3. Adjust the order of the objects.
4. Add a display name for an object.
5. Link the object to the corresponding tool data. Tool data from the ST_CNC_TOOL_EXT structure in the Tc3_CNC library is available to you. The “Data Path” and “Data Type” properties refer to this structure. Examples:

Data Path	Data Type
descriptionText	string
tool_desc.tool_id.basic	number
tool_desc.param.0	number
tool_desc.valid	boolean

6. Optional: Set a CSS style for the table column.
7. Optional: Specify a scaling factor for unit conversion for the object.
8. Specify the number of magazine slots per tool magazine.
9. Specify whether the column for the magazine number should be displayed.
10. The control `TcHmiCncToolList` requires a data binding to a symbol from the TcHmiCnc extension. Set up data binding. Data bindings to a symbol of type “CNC-ChannelType” are permitted.

Data Binding	Symbol
Channel Data	For example: • CNC.Data.ChannelClientSelected • CNC.Data.Channel.0

11. Specify whether the channel number is displayed.
12. Optional: Specify whether to display vertical or horizontal divider lines for the columns or rows in the table.
13. Set the row height of the table in px.

User settings via TcHmiCncSettings

You can adjust the following settings via the control `TcHmiCncSettings`:

Section	Setting	Standard	Description
TcCncHmi	Decimal places	3	Decimal places (after applying the scaling factor)

Key combinations

Shortcut	Action
Enter	Confirm and save your entry
Tab	Confirm the entry and move to the next field
Shift + Tab	Confirm entry and go to the previous field

API functions

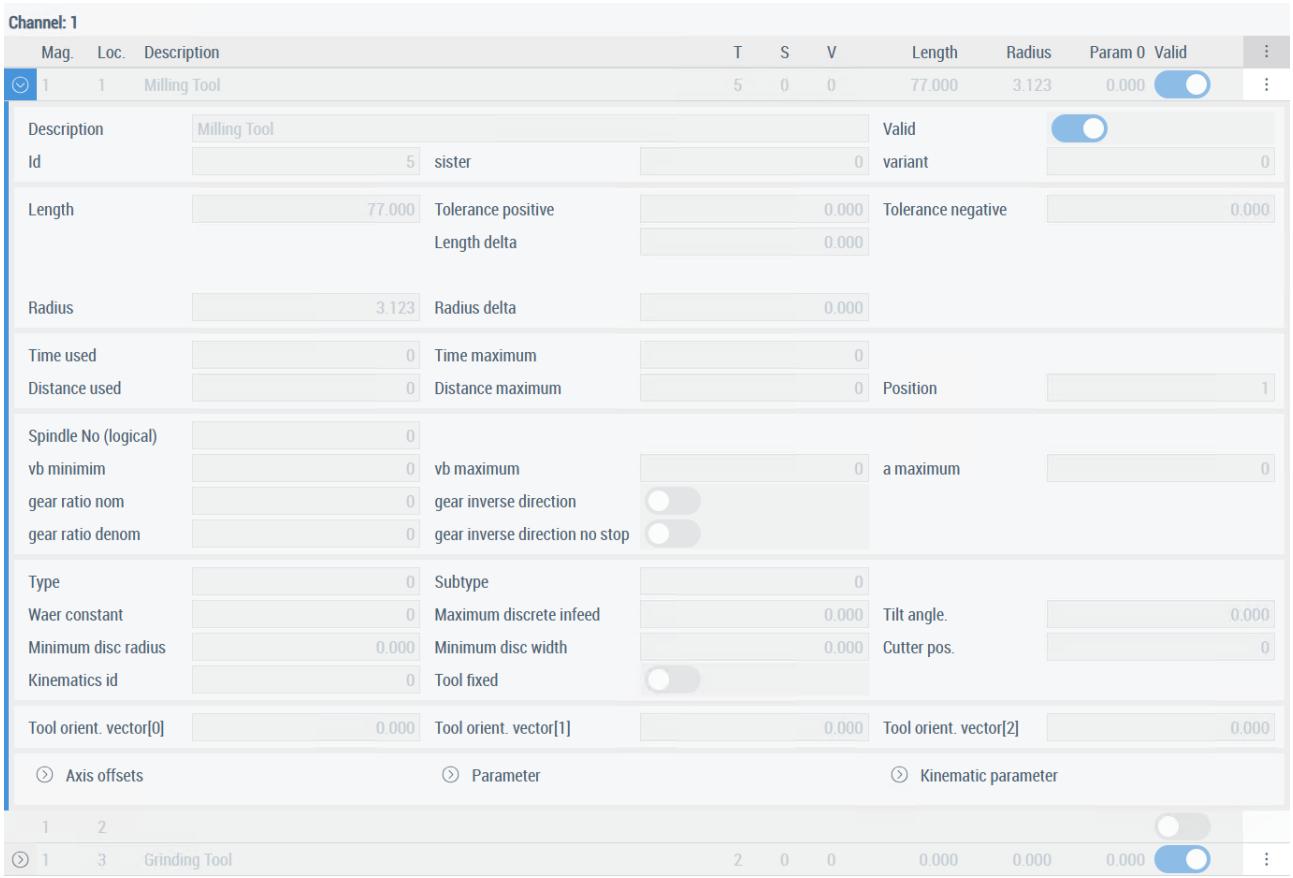
The control `TcHmiCncToolList` has a public API that allows you to use the following functions.

Function	Description
setDataSymbol() / getDataSymbol()	Set or read data binding for a symbol in the TcHmiCnc extension.
setElementToShow() / getElementToShow()	Set or read column description
setMagazineEnabled() / getMagazineEnabled()	Turn magazine management on or off
setPlaceCount() / getPlaceCount()	Set or read the number of magazine slots
setShowChannelID() / getShowChannelId()	Turn the channel number display on or off
setShowVerticalLines() / getShowVerticalLines()	Turn vertical dividing lines on or off
setShowHorizontalLines() / getShowHorizontalLines()	Turn horizontal dividing lines on or off
setRowHeight() / getRowHeight()	Specify the line height in pixels
setToolDataString() / getToolDataString()	Set or read the tool table as a JSON string

Impact of user management

The control `TcHmiCncToolList` evaluates the “Operate” access right. For “Allow,” the control behaves as usual. When “Disallow” is selected, the control is limited to the following:

- Display of tool data only



If you have not set up user management, “Operate” is considered “Allow,” and the control behaves as usual.

5.23 TcHmiCncTreeview

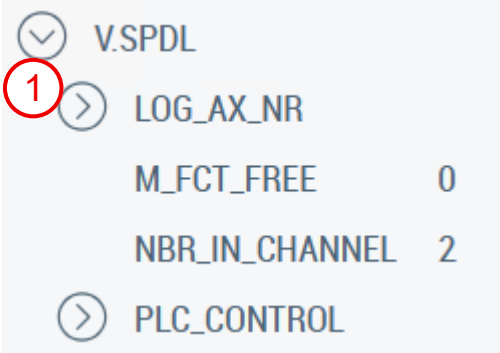
Component: TcHmiCncTreeview
Namespace: TcHmi.Controls.TcHmiCncBase
Available since version: 1.0.0

Description

The control TcHmiCncTreeview displays JSON data in a hierarchical, expandable tree view. It is used internally for the control TcHmiCncVariableView [▶ 132], among other things.

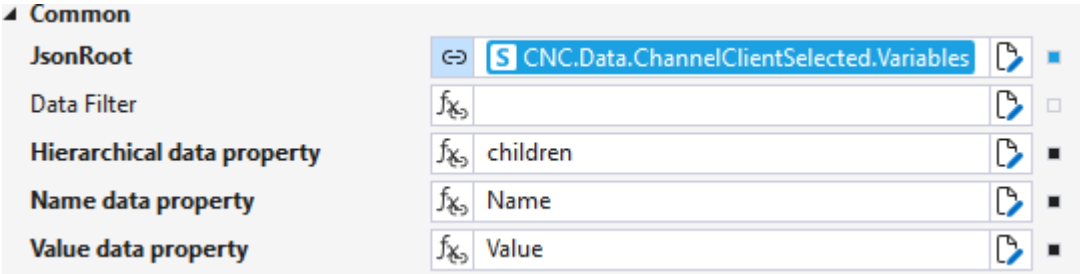
Operation

The operation is similar to that of the control TcHmiCncVariableView [▶ 132]. Click (1) to expand the next level of the tree structure.



Configuration

The control TcHmiCncTreeview requires a data binding to a symbol from the TcHmiCnc extension. Data bindings to a symbol of type “String” are allowed. The string must contain the JSON-structured data tree.



Data filter

Use this setting to define a search term, if needed, by which the JSON file will be filtered.

Hierarchical data property

Use this setting to define the property name of the child node elements.

Name data property

Use this setting to define the property name of an element’s label.

Value data property

Use this setting to define the property name of the node value.

```
[
    {
        "Name": "V.SPDL",
        "children": [
            {
                "Name": "LOG_AX_NR",
                "Value": "",
                "children": []
```

```
            },
            {
                "Name": "M_FCT_FREE",
                "Value": "0",
            },
            {
                "Name": "NBR_IN_CHANNEL",
                "Value": "2",
            },
            {
                "Name": "PLC_CONTROL",
                "Value": "",
                "children": []
            }]
    }
]
```

User settings via TcHmiCncSettings

The control `TcHmiCncTreeview` does not have any customizable settings in the control `TcHmiCncSettings`.

Keyboard shortcuts

The control `TcHmiCncTreeview` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncTreeview` has a public API that allows you to use the following functions.

Function	Description
setJsonTree() / getJsonTree()	Set or read a JSON-structured data tree as a JSON string
setJsonObject()	Pass the JSON-structured data tree as an object.
setDataFilter() / getDataFilter()	Set or read text filter
setHierarchical() / getHierarchical()	Set or read property names of the child node elements
setNameProp() / getNameProp()	Set or read the property names of an element’s label
setValueProp() / getValueProp()	Set or read property names of the node value
selectFirstNode()	Selects the first node and triggers the onNodeSelected event.

Impact of user management

The control `TcHmiCncTreeview` does not exhibit any specific change in behavior as a result of the “Operate” access right.

5.24 TcHmiCncVariableView

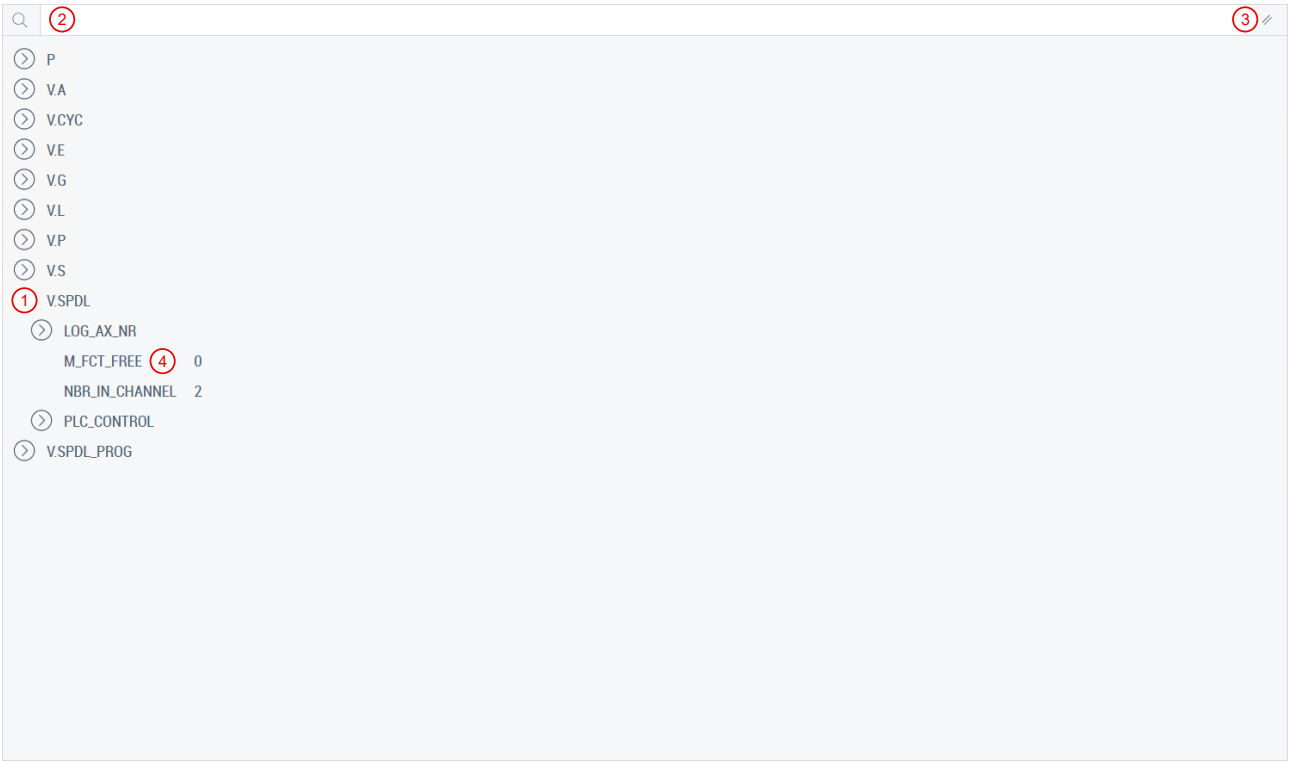
Component: `TcHmiCncVariableView`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

Description

The control `TcHmiCncVariableView` displays the online variable values of a CNC channel, grouped in a hierarchical tree structure. A text filter allows you to search for variable names. The following variable types are available:

- P
- V.A
- V.CYC
- V.E
- V.G
- V.L
- V.P
- V.S
- V.SPDL
- V.SPDL_PROG

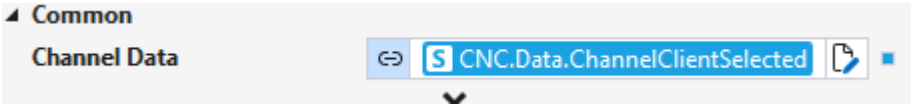
Operation



No.	Description
1	Button to expand the next level of the tree structure. Only the expanded variable types are read from the CNC kernel via ADS.
2	Search field for variables. The search results will only display variable names whose variable types were previously read via ADS. Therefore, expand the top-level hierarchy whose variables you want to search through.
3	Reset search function
4	Double-clicking a variable copies the corresponding variable path to the clipboard. For example, you can paste the variable path from the clipboard into an NC program.

Configuration

The control `TcHmiCncVariableView` requires a data binding to a symbol from the TcHmiCnc extension. Data bindings to a symbol of type “CNC-ChannelType” are permitted.



Data Binding	Symbol
Channel Data	For example: • CNC.Data.ChannelClientSelected • CNC.Data.Channel.0

User settings via TcHmiCncSettings

The control `TcHmiCncVariableView` does not have any customizable settings in the control `TcHmiCncSettings`.

Keyboard shortcuts

The control `TcHmiCncVariableView` does not have any specific keyboard shortcuts.

API functions

The control `TcHmiCncVariableView` has a public API that allows you to use the following function.

Function	Description
setChannelSymbol() / getChannelSymbol()	Set or read data binding for a symbol in the TcHmiCnc extension.

Impact of user management

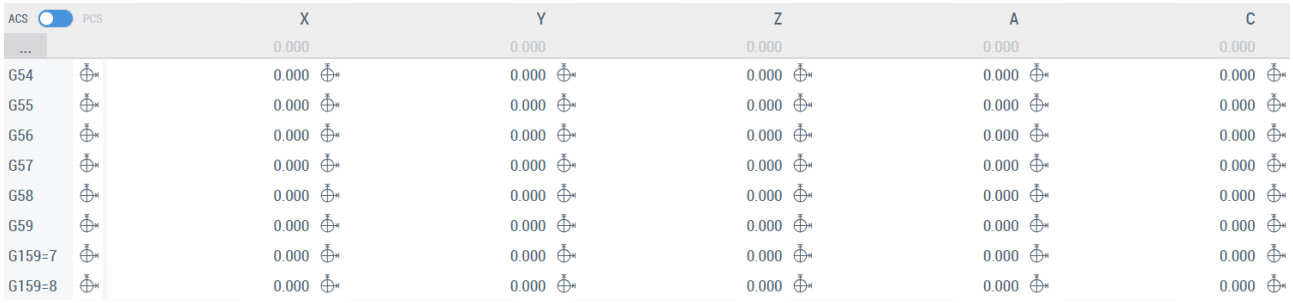
The control `TcHmiCncVariableView` does not exhibit any specific change in behavior as a result of the “Operate” access right.

5.25 TcHmiCncZeroOffsets

Component: `TcHmiCncZeroOffsets`
Namespace: `TcHmi.Controls.TcHmiCncBase`
Available since version: 1.0.0

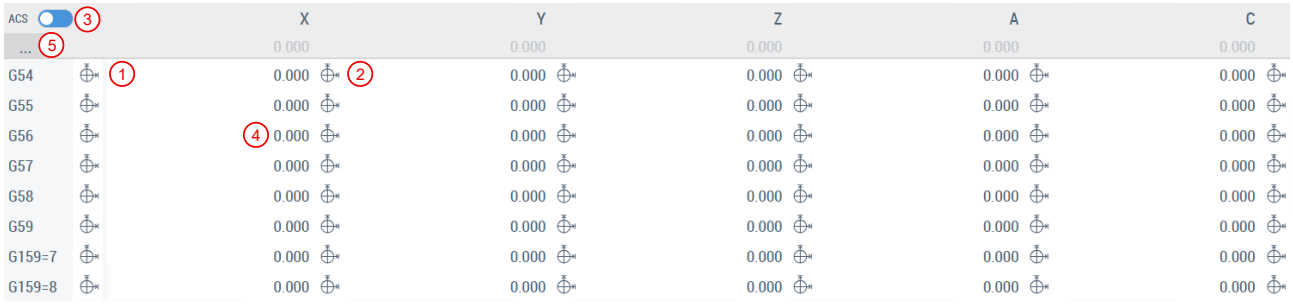
Description

The control `TcHmiCncZeroOffsets` displays the zero offset table for a CNC channel and allows you to edit it.



Each row represents a G-Code zero offset register (G54–G59, G159=n), and each column represents an axis in the channel.

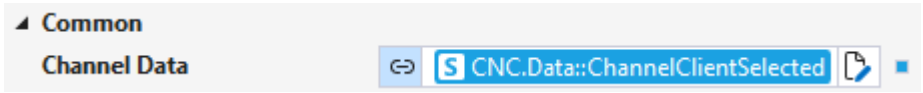
Operation



No.	Description
1	Set the zero offset for all axes
2	Set zero offset for one axis
3	Switch between the axis coordinate system (ACS) and the part coordinate system (PCS). In PCS, buttons (1) and (2) are only active if you have selected G53.
4	Input field for manually setting the zero offset. When entering data, you can also use mathematical expressions (+, -, *, /, pi, sin(), cos(), tan(), and cot()).
5	Download (export) or upload (import) the entire zero offset table. During import, the existing zero offset table will be overwritten

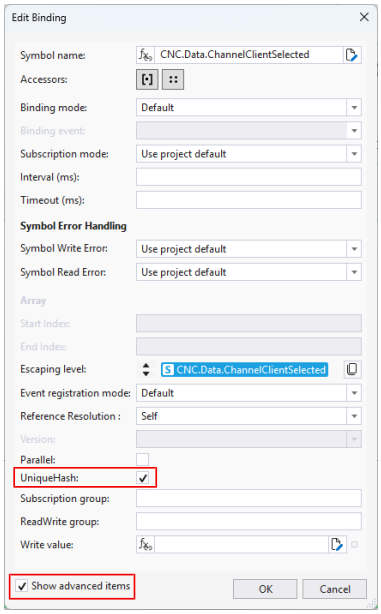
Configuration

The control `TcHmiCncZeroOffsets` requires a data binding to a symbol from the TcHmiCnc extension. Data bindings to a symbol of type “CNC-ChannelType” are permitted.



Data Binding	Symbol
Channel Data	For example: • CNC.Data.ChannelClientSelected • CNC.Data.Channel.0

When performing data binding, make sure that a unique hash is enabled. To do this, right-click the icon you created and open the “Edit Binding” context menu.



User settings via TcHmiCncSettings

You can adjust the following settings via the control `TcHmiCncSettings`:

Section	Setting	Standard	Description
TcCncHmi.ZeroOffsets	Number of G lines	8	Number of zero offset rows in the table (G54–G59, G159=n)
TcCncHmi	Decimal places	3	Decimal places for a zero offset

Keyboard shortcuts

You can use the following keyboard shortcuts in the control `TcHmiCncZeroOffsets`.

Abbreviation	Action
Enter	Confirm the entry and move to the next field in the same column
Tab	Confirm the entry and move to the next field on the same line

API functions

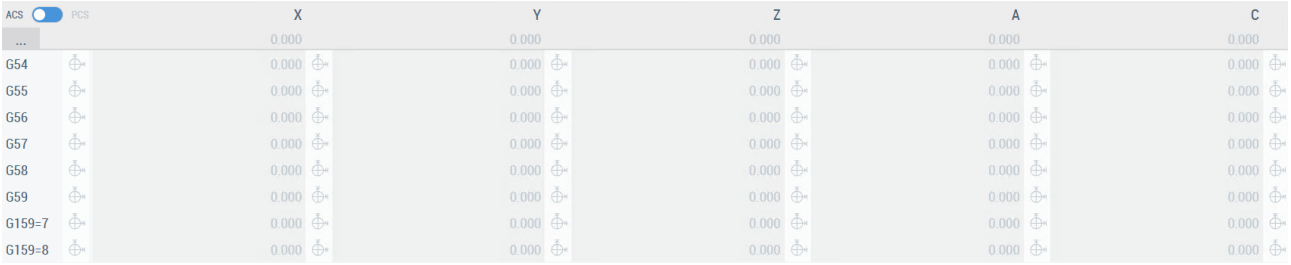
The control `TcHmiCncZeroOffsets` has a public API that allows you to use the following functions.

Function	Description
setDataSymbol() / getDataSymbol()	Set or read data binding for a symbol in the TcHmiCnc extension. The control then resubscribes to the zero offset table
setShowChannelName() / getShowChannelName()	Turn the channel name display on or off
setShowVerticalLines() / getShowVerticalLines()	Turn vertical dividing lines on or off
setShowHorizontalLines() / getShowHorizontalLines()	Turn horizontal dividing lines on or off
setRowHeight() / getRowHeight()	Specify the line height in pixels

Impact of user management

The control `TcHmiCncZeroOffsets` evaluates the “Operate” access right. For “Allow,” the control behaves as usual. When “Disallow” is selected, the control is limited to the following:

- Display of zero offset
- Switching Between ACS and PCS



If you have not set up user management, “Operate” is considered “Allow,” and the control behaves as usual.

6 Appendix

6.1 Third-party components

This software contains third-party components.
Please refer to the license file provided in the following folder for further information:

C:\ProgramData\Beckhoff\NuGetPackages\Beckhoff.TwinCAT.HMI.CNC.Base.<Version>.nupkg\Legal

C:\ProgramData\Beckhoff\NuGetPackages\Beckhoff.TwinCAT.HMI.CNC.Server.<Version>.nupkg\Legal

6.2 Support and Service

Beckhoff Support and Service provides assistance with questions regarding Beckhoff products and system solutions, as well as with their planning, commissioning, and operation.

Beckhoff subsidiaries and representative offices

Please contact your Beckhoff subsidiary or your representative office for local support and service for Beckhoff products. You can find the addresses of Beckhoff subsidiaries and representative offices at: www.beckhoff.com/worldwide.

For Beckhoff products and technologies, we offer Training programs, tutorials, and webinars.

Beckhoff Support

Beckhoff Support provides assistance with technical questions regarding the use, planning, programming, and commissioning of Beckhoff products.

☎ Hotline:	+49 5246 963-157
✉ Email:	support@beckhoff.com, Support form
Web page:	www.beckhoff.com/support

Beckhoff Service

Our service specialists are here to assist you with any questions you may have regarding after-sales service processes, such as replacement parts, repairs, etc.

☎ Hotline:	+49 5246 963-460
✉ Email:	service@beckhoff.com, Service form
Web page:	www.beckhoff.com/service

Download finder

The Download finder serves as a central source for downloadable documents and files related to Beckhoff products, such as application reports, technical documentation, drawings, and configuration files.

Beckhoff Information System

The Beckhoff Information System is the central online documentation resource for Beckhoff products and contains documentation on products and automation topics, as well as sample code.

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl, Germany

☎ Phone:	+49 5246 963-0
✉ Email:	info@beckhoff.com
Web page:	www.beckhoff.com

Trademark statements

Beckhoff®, ATRO®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, MX-System®, Safety over EtherCAT®, TC/BSD®, TwinCAT®, TwinCAT/BSD®, TwinSAFE®, XFC®, XPlanar® and XTS® are registered and licensed trademarks of Beckhoff Automation GmbH.

Third-party trademark statements

Excel, IntelliSense, Microsoft, Microsoft Azure, Microsoft Edge, PowerShell, Visual Studio, Windows and Xbox are trademarks of the Microsoft group of companies.

More Information:
www.beckhoff.com/TF5310

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Germany
Phone: +49 5246 9630
info@beckhoff.com
www.beckhoff.com