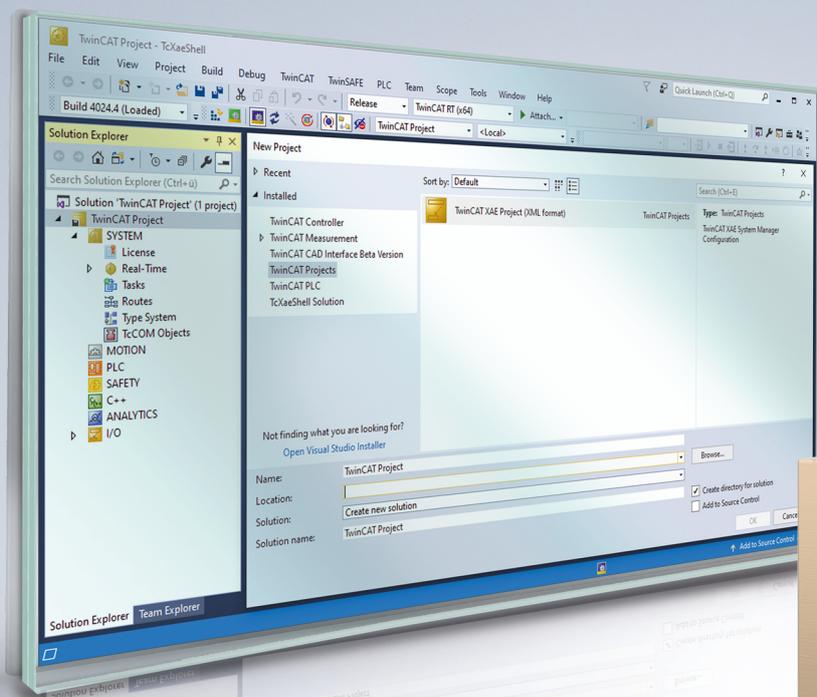


BECKHOFF New Automation Technology

Handbuch | DE

TF8550

TwinCAT 3 | Plastic HMI Framework



Inhaltsverzeichnis

1	Vorwort	7
1.1	Hinweise zur Dokumentation	7
1.2	Zu Ihrer Sicherheit	8
1.3	Hinweise zur Informationssicherheit	9
1.4	Ausgabestände der Dokumentation	9
2	Übersicht	10
3	Installation	11
3.1	Systemvoraussetzungen	11
3.2	Lizenzierung	11
3.3	Installation	13
4	Controls	14
4.1	ArrowMotionGraph	15
4.1.1	Attribute	16
4.1.2	Funktionen	23
4.1.3	Benutzerinteraktionen	25
4.2	BlowPressureChart	25
4.2.1	Attribute	26
4.3	CamControl	31
4.3.1	Attribute	31
4.3.2	Funktionen	36
4.3.3	Events	37
4.4	Configurator	37
4.4.1	Attribute	38
4.4.2	Unit_Config.json	45
4.5	CurveEditor	48
4.5.1	Attribute	49
4.5.2	Funktionen	69
4.5.3	Events	74
4.5.4	Benutzerinteraktionen	75
4.5.5	Schritt für Schritt	77
4.5.6	Verwendung der Rezeptverwaltung mit dem Kurveditor	79
4.6	FavoriteBarControl	80
4.6.1	Attribute	80
4.6.2	Benutzerinteraktionen	84
4.7	InputBox	85
4.7.1	Attribute	85
4.7.2	Funktionen	92
4.7.3	Events	93
4.7.4	Schritt für Schritt	93
4.8	ManualOperation	95
4.8.1	Attribute	96
4.8.2	ManualOperation-Control mit Hardware-Encoder	108
4.9	MeasurementUnitSelector	109

4.9.1	Attribute.....	110
4.9.2	Benutzerinteraktionen	113
4.10	Monitoring	113
4.10.1	Attribute.....	115
4.11	PfwSingleTempControl	121
4.11.1	Attribute.....	122
4.12	ProcessScheduler	126
4.12.1	Attribute.....	127
4.12.2	Funktionen	129
4.12.3	Events	131
4.12.4	Dialoge	132
4.13	StateIndicator	133
4.13.1	Attribute.....	133
4.14	Table	135
4.14.1	Attribute.....	136
4.14.2	Events	144
4.14.3	Schritt für Schritt.....	144
4.15	TimerControl	146
4.15.1	Attribute.....	147
4.16	UserLoggingViewer	151
4.16.1	Attribute.....	152
4.16.2	Dialoge	156
4.17	VariableViewControl.....	156
4.17.1	Attribute.....	157
4.17.2	Dialoge	165
4.18	Funktionen	166
4.18.1	UnitConversion.....	166
4.18.2	LogUserChanges	171
5	Functions	173
5.1	Axes	173
5.1.1	ChangeInputBoxUnit.....	173
5.1.2	OperateWtcWithHardware	173
5.1.3	UpdateAxesHomingUnits	174
5.2	CodeBehind	175
5.2.1	DisableContextMenu.....	175
5.2.2	SlidingFunction.....	175
5.3	Frame.....	175
5.3.1	HideSliderArea	175
5.3.2	ShowSliderArea	176
5.3.3	ToggleSliderArea	177
5.3.4	TakeScreenShot	177
5.3.5	UpdateSliderContentRegion	178
5.3.6	UpdateSliderContentRegionEx	179
5.4	General	180
5.4.1	CreateBinding	180
5.4.2	ModifyArray	181

5.4.3	SetSymbolArrayIdx	181
5.5	Parameters.....	182
5.5.1	FillParamTable	182
5.5.2	UpdateParametersSettingsControls.....	183
5.5.3	UpdateParametersSettingsControlsEx	186
5.5.4	UpdateTemperatureParametersTableNumber.....	187
5.5.5	UpdateTemperatureParametersTableNumberEx	188
5.6	Scope	188
5.6.1	UpdateScope	188
5.7	Trend.....	189
5.7.1	ResetTrendAxisColors	189
5.7.2	UpdateTrend	190
5.7.3	UpdateTrendAxisColors	190
6	Images	192
7	Localizations	197
8	RecipeHelper	198
8.1	Konzept.....	198
8.1.1	Erstellte Rezepttypen und Rezepte.....	198
8.1.2	Temporäre Rezepte	199
8.1.3	Automatisches Hinzufügen von Symbolen zu Rezepttypen.....	203
8.1.4	Handshake mit PLC	203
8.1.5	Fehlermeldungen auf dem Bildschirm.....	206
8.1.6	Rezepte sichern und wiederherstellen	207
8.2	Funktionen	207
8.2.1	ActivateBackupRecipe	207
8.2.2	SaveActiveRecipe	208
8.2.3	RcpAutoUpdate.....	209
8.2.4	ListActDataTypeRecipes.....	209
8.2.5	LoadProductRecipe.....	210
8.2.6	DeleteActRcp	211
8.2.7	ExportRecipe.....	211
8.2.8	ImportRecipe.....	212
8.3	Verwendung von RecipeHelper mit HMI-Controls	213
8.3.1	Auflistung von Rezepten in einem Datagrid.....	213
8.3.2	Aktives Rezept mit ungespeicherten Änderungen anzeigen.....	214
9	RFID.....	215
9.1	MiFare Classic.....	215
9.2	ConnectToComPort	216
9.3	GetConnectedComPortInfo	217
9.4	ReadMiFareData	218
9.5	WriteMiFareData	220
9.6	LoginWithMiFare	222
9.7	EnableAutoLogin	223
9.8	DisableAutoLogin	223
9.9	Spezifische Datentypen	224

9.9.1	RFIDConfig	224
10	Themes	225
11	Temperatur	226
11.1	PfwTempParameters	226
11.1.1	Attribute	227
11.2	TemperatureScheduler	228
11.2.1	Attribute	229
11.2.2	Funktionen	231
11.2.3	Ereignisse	232
11.2.4	Dialoge	233
11.3	ZoneConfiguration	234
11.3.1	Attribute	236
11.3.2	Benutzerinteraktionen	239
11.3.3	Funktionen	240
11.4	ZoneGrouping	241
11.4.1	Attribute	242
11.4.2	Funktionen	244
11.4.3	Benutzerinteraktionen	245
11.4.4	Dialoge	246
11.5	ZoneImageLayout	247
11.5.1	Attribute	249
11.5.2	Funktionen	253
11.5.3	Benutzerinteraktionen	255
11.5.4	Dialoge	260
11.5.5	Events	261
11.5.6	Schritt für Schritt	262

1 Vorwort

1.1 Hinweise zur Dokumentation

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs- und Automatisierungstechnik, das mit den geltenden nationalen Normen vertraut ist.

Zur Installation und Inbetriebnahme der Komponenten ist die Beachtung der Dokumentation und der nachfolgenden Hinweise und Erklärungen unbedingt notwendig.

Das Fachpersonal ist verpflichtet, stets die aktuell gültige Dokumentation zu verwenden.

Das Fachpersonal hat sicherzustellen, dass die Anwendung bzw. der Einsatz der beschriebenen Produkte alle Sicherheitsanforderungen, einschließlich sämtlicher anwendbaren Gesetze, Vorschriften, Bestimmungen und Normen erfüllt.

Disclaimer

Diese Dokumentation wurde sorgfältig erstellt. Die beschriebenen Produkte werden jedoch ständig weiterentwickelt.

Wir behalten uns das Recht vor, die Dokumentation jederzeit und ohne Ankündigung zu überarbeiten und zu ändern.

Aus den Angaben, Abbildungen und Beschreibungen in dieser Dokumentation können keine Ansprüche auf Änderung bereits gelieferter Produkte geltend gemacht werden.

Marken

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® und XPlanar® sind eingetragene und lizenzierte Marken der Beckhoff Automation GmbH.

Die Verwendung anderer in dieser Dokumentation enthaltenen Marken oder Kennzeichen durch Dritte kann zu einer Verletzung von Rechten der Inhaber der entsprechenden Bezeichnungen führen.

Patente

Die EtherCAT-Technologie ist patentrechtlich geschützt, insbesondere durch folgende Anmeldungen und Patente:

EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702

mit den entsprechenden Anmeldungen und Eintragungen in verschiedenen anderen Ländern.



EtherCAT® ist eine eingetragene Marke und patentierte Technologie lizenziert durch die Beckhoff Automation GmbH, Deutschland

Copyright

© Beckhoff Automation GmbH & Co. KG, Deutschland.

Weitergabe sowie Vervielfältigung dieses Dokuments, Verwertung und Mitteilung seines Inhalts sind verboten, soweit nicht ausdrücklich gestattet.

Zu widerhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patent-, Gebrauchsmuster- oder Geschmacksmustereintragung vorbehalten.

1.2 Zu Ihrer Sicherheit

Sicherheitsbestimmungen

Lesen Sie die folgenden Erklärungen zu Ihrer Sicherheit.
Beachten und befolgen Sie stets produktspezifische Sicherheitshinweise, die Sie gegebenenfalls an den entsprechenden Stellen in diesem Dokument vorfinden.

Haftungsausschluss

Die gesamten Komponenten werden je nach Anwendungsbestimmungen in bestimmten Hard- und Software-Konfigurationen ausgeliefert. Änderungen der Hard- oder Software-Konfiguration, die über die dokumentierten Möglichkeiten hinausgehen, sind unzulässig und bewirken den Haftungsausschluss der Beckhoff Automation GmbH & Co. KG.

Qualifikation des Personals

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs-, Automatisierungs- und Antriebstechnik, das mit den geltenden Normen vertraut ist.

Signalwörter

Im Folgenden werden die Signalwörter eingeordnet, die in der Dokumentation verwendet werden. Um Personen- und Sachschäden zu vermeiden, lesen und befolgen Sie die Sicherheits- und Warnhinweise.

Warnungen vor Personenschäden

GEFAHR

Es besteht eine Gefährdung mit hohem Risikograd, die den Tod oder eine schwere Verletzung zur Folge hat.

WARNUNG

Es besteht eine Gefährdung mit mittlerem Risikograd, die den Tod oder eine schwere Verletzung zur Folge haben kann.

VORSICHT

Es besteht eine Gefährdung mit geringem Risikograd, die eine mittelschwere oder leichte Verletzung zur Folge haben kann.

Warnung vor Umwelt- oder Sachschäden

HINWEIS

Es besteht eine mögliche Schädigung für Umwelt, Geräte oder Daten.

Information zum Umgang mit dem Produkt



Diese Information beinhaltet z. B.:
Handlungsempfehlungen, Hilfestellungen oder weiterführende Informationen zum Produkt.

1.3 Hinweise zur Informationssicherheit

Die Produkte der Beckhoff Automation GmbH & Co. KG (Beckhoff) sind, sofern sie online zu erreichen sind, mit Security-Funktionen ausgestattet, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen. Trotz der Security-Funktionen sind die Erstellung, Implementierung und ständige Aktualisierung eines ganzheitlichen Security-Konzepts für den Betrieb notwendig, um die jeweilige Anlage, das System, die Maschine und die Netzwerke gegen Cyber-Bedrohungen zu schützen. Die von Beckhoff verkauften Produkte bilden dabei nur einen Teil des gesamtheitlichen Security-Konzepts. Der Kunde ist dafür verantwortlich, dass unbefugte Zugriffe durch Dritte auf seine Anlagen, Systeme, Maschinen und Netzwerke verhindert werden. Letztere sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn entsprechende Schutzmaßnahmen eingerichtet wurden.

Zusätzlich sollten die Empfehlungen von Beckhoff zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Informationssicherheit und Industrial Security finden Sie in unserem <https://www.beckhoff.de/secguide>.

Die Produkte und Lösungen von Beckhoff werden ständig weiterentwickelt. Dies betrifft auch die Security-Funktionen. Aufgrund der stetigen Weiterentwicklung empfiehlt Beckhoff ausdrücklich, die Produkte ständig auf dem aktuellen Stand zu halten und nach Bereitstellung von Updates diese auf die Produkte aufzuspielen. Die Verwendung veralteter oder nicht mehr unterstützter Produktversionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Hinweise zur Informationssicherheit zu Produkten von Beckhoff informiert zu sein, abonnieren Sie den RSS Feed unter <https://www.beckhoff.de/secinfo>.

1.4 Ausgabestände der Dokumentation

Version	Änderung
1.3.x	Neu: RFID [► 215] Erweiterungen bei den Controls [► 14]

2 Übersicht

TF8550 Plastic TC HMI bietet zusammen mit den entsprechenden NuGet-Paketen eine Reihe neuer Controls, Funktionen, Code-Behind-Skripte, Bilder und ein spezielles Thema, die auf die spezifischen Anforderungen von HMI-Projekten für Kunststoffmaschinen zugeschnitten sind.

Diese Ergänzungen sollen den Prozess der Einrichtung eines HMI-Projekts für Kunststoffanwendungen vereinfachen und beschleunigen und es den Benutzern erleichtern, maßgeschneiderte und effektive Lösungen zu erstellen.

NuGet-Pakete des TF8550

Name des Pakets	Beschreibung
Beckhoff.TwinCAT.HMI.Plastic.Controls [► 14]	Dieses Paket bietet neue Controls für TwinCAT-HMI-Projekte.
Beckhoff.TwinCAT.HMI.Plastic.Functions [► 173]	Dieses Paket enthält Funktionen, die in TF85XX Plastic Application HMI-Projekten verwendet werden.
Beckhoff.TwinCAT.HMI.Plastic.Images [► 192]	Dieses Paket fügt Icons und Bilder hinzu, die für Kunststoff-HMI-Anwendungen verwendet werden können.
Beckhoff.TwinCAT.HMI.Plastic.Localizations [► 197]	Dieses Paket fügt lokalisierte Symbole hinzu, die in Kunststoff-HMI-Anwendungen verwendet werden können.
Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper [► 198]	Dieses Paket enthält Funktionen und Skripte, die eine vorkonfigurierte Rezeptstruktur hinzufügen und das automatische Speichern von Werten aus den Controls im Rezept ermöglichen.
Beckhoff.TwinCAT.HMI.Plastic.Temperature [► 14]	Dieses Paket bietet spezielle Controls und Funktionen für die temperaturbezogene Visualisierung in Plastic-HMI-Applications.
Beckhoff.TwinCAT.HMI.Plastic.Themes [► 225]	Dieses Paket fügt ein vorkonfiguriertes Plastic-Theme hinzu, um Standard-TwinCAT-HMI-Controls im Projekt zu gestalten.

3 Installation

3.1 Systemvoraussetzungen

Das Paket Beckhoff.TwinCAT.HMI.Plastic ist als NuGet-Paket verfügbar und kann mit TC3 TwinCAT HMI-Projekten verwendet werden, die mit TE2000 TC3 HMI Engineering erstellt wurden.

Die Mindestversion von TE2000 TC3 HMI Engineering ist 1.12.742.x

Weitere Informationen zu den [Mindestsystemanforderungen für TE2000 TC3 HMI Engineering](#) finden Sie hier.

3.2 Lizenzierung

Die TwinCAT 3 Function ist als Vollversion oder als 7-Tage-Testversion freischaltbar. Beide Lizenztypen sind über die TwinCAT-3-Entwicklungsumgebung (XAE) aktivierbar.

Lizenzierung der Vollversion einer TwinCAT 3 Function

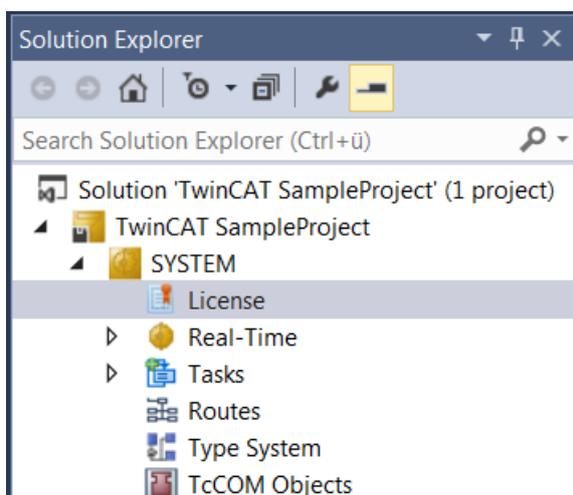
Die Beschreibung der Lizenzierung einer Vollversion finden Sie im Beckhoff Information System in der Dokumentation „[TwinCAT-3-Lizenzierung](#)“.

Lizenzierung der 7-Tage-Testversion einer TwinCAT 3 Function



Eine 7-Tage-Testversion kann nicht für einen [TwinCAT-3-Lizenz-Dongle](#) freigeschaltet werden.

1. Starten Sie die TwinCAT-3-Entwicklungsumgebung (XAE).
2. Öffnen Sie ein bestehendes TwinCAT-3-Projekt oder legen Sie ein neues Projekt an.
3. Wenn Sie die Lizenz für ein Remote-Gerät aktivieren wollen, stellen Sie das gewünschte Zielsystem ein. Wählen Sie dazu in der Symbolleiste in der Drop-down-Liste **Choose Target System** das Zielsystem aus.
 - ⇒ Die Lizenzierungseinstellungen beziehen sich immer auf das eingestellte Zielsystem. Mit der Aktivierung des Projekts auf dem Zielsystem werden automatisch auch die zugehörigen TwinCAT-3-Lizenzen auf dieses System kopiert.
4. Klicken Sie im **Solution Explorer** im Teilbaum **SYSTEM** doppelt auf **License**.



⇒ Der TwinCAT-3-Lizenzmanager öffnet sich.

5. Öffnen Sie die Registerkarte **Manage Licenses**. Aktivieren Sie in der Spalte **Add License** das Auswahlkästchen für die Lizenz, die Sie Ihrem Projekt hinzufügen möchten (z. B. „TF4100 TC3 Controller Toolbox“).

Order No	License	Add License
TF3601	TC3 Condition Monitoring Level 2	<input type="checkbox"/> cpu license
TF3650	TC3 Power Monitoring	<input type="checkbox"/> cpu license
TF3680	TC3 Filter	<input type="checkbox"/> cpu license
TF3800	TC3 Machine Learning Inference Engine	<input type="checkbox"/> cpu license
TF3810	TC3 Neural Network Inference Engine	<input type="checkbox"/> cpu license
TF3900	TC3 Solar-Position-Algorithm	<input type="checkbox"/> cpu license
TF4100	TC3 Controller Toolbox	<input checked="" type="checkbox"/> cpu license
TF4110	TC3 Temperature-Controller	<input type="checkbox"/> cpu license
TF4500	TC3 Speech	<input type="checkbox"/> cpu license

6. Öffnen Sie die Registerkarte **Order Information (Runtime)**.
 ⇒ In der tabellarischen Übersicht der Lizenzen wird die zuvor ausgewählte Lizenz mit dem Status „missing“ angezeigt.
7. Klicken Sie auf **7 Days Trial License...**, um die 7-Tage-Testlizenz zu aktivieren.

Order Information (Runtime)	Manage Licenses	Project Licenses	Online Licenses									
<input type="checkbox"/> Disable automatic detection of required licenses for project												
<table border="1"> <thead> <tr> <th>License Device</th> <th>Target (Hardware Id)</th> <th>Add...</th> </tr> </thead> <tbody> <tr> <td>System Id:</td> <td>Platform:</td> <td></td> </tr> <tr> <td>2DB25408-B4CD-81DF-5488-6A3D9B49EF19</td> <td>other (91)</td> <td></td> </tr> </tbody> </table>				License Device	Target (Hardware Id)	Add...	System Id:	Platform:		2DB25408-B4CD-81DF-5488-6A3D9B49EF19	other (91)	
License Device	Target (Hardware Id)	Add...										
System Id:	Platform:											
2DB25408-B4CD-81DF-5488-6A3D9B49EF19	other (91)											
<table border="1"> <thead> <tr> <th colspan="2">License Request</th> </tr> </thead> <tbody> <tr> <td>Provider:</td> <td>Beckhoff Automation</td> </tr> <tr> <td>License Id:</td> <td>Customer Id:</td> </tr> <tr> <td>Comment:</td> <td></td> </tr> </tbody> </table>				License Request		Provider:	Beckhoff Automation	License Id:	Customer Id:	Comment:		
License Request												
Provider:	Beckhoff Automation											
License Id:	Customer Id:											
Comment:												
<table border="1"> <thead> <tr> <th colspan="2">License Activation</th> </tr> </thead> <tbody> <tr> <td>7 Days Trial License...</td> <td>License Response File...</td> </tr> </tbody> </table>				License Activation		7 Days Trial License...	License Response File...					
License Activation												
7 Days Trial License...	License Response File...											

- ⇒ Es öffnet sich ein Dialog, der Sie auffordert, den im Dialog angezeigten Sicherheitscode einzugeben.

Enter Security Code	
Please type the following 5 characters:	OK
Kg8T4	
<input type="text"/>	Cancel

8. Geben Sie den Code genauso ein, wie er angezeigt wird, und bestätigen Sie ihn.
 9. Bestätigen Sie den nachfolgenden Dialog, der Sie auf die erfolgreiche Aktivierung hinweist.

⇒ In der tabellarischen Übersicht der Lizenzen gibt der Lizenzstatus nun das Ablaufdatum der Lizenz an.

10. Starten Sie das TwinCAT-System neu.

⇒ Die 7-Tage-Testversion ist freigeschaltet.

3.3 Installation

1. Das Beckhoff.TwinCAT.HMI.Plastic-Paket ist als NuGet-Paketdatei verfügbar.
2. Weitere Informationen über die NuGet Paketverwaltung in TwinCAT HMI-Projekten finden Sie hier.
3. Kopieren Sie die Datei Beckhoff.TwinCAT.HMI.Plastic.nuget in eines der konfigurierten NuGet-Repositories.
4. Dadurch wird das Paket für die Installation in jedem TwinCAT-HMI-Projekt verfügbar. Installieren Sie das Paket mit dem NuGet Package Manager.
5. Sobald das Paket installiert ist, sind alle Controls, Funktionen und anderen Ressourcen in Ihrem Projekt verfügbar.

4 Controls

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

Das NuGet-Paket **Beckhoff.TwinCAT.HMI.Plastic.Controls** kann über das Paketverwaltungssystem NuGet in das TwinCAT-HMI-Projekt installiert werden.

Name	Beschreibung
ArrowMotionGraph [► 15]	Mit dem ArrowMotionGraph-Control kann der Benutzer Segmente einer Achsenbewegung konfigurieren. Positionen und Geschwindigkeiten für verschiedene Segmente werden als Pfeile im Control visualisiert.
BlowPressureChart [► 25]	BlowPressureChart ist ein spezielles Chart Type Control, das den Blasprozess in einer Blasformmaschine visualisieren kann.
CamControl [► 31]	CamControl dient der Visualisierung und Konfiguration einer Reihe von Nockenschaltern. Jeder Nockenschalter ist durch Hysterese- und Schwellenwerte definiert.
Configurator [► 37]	Das Configurator-Control speichert die Benutzereinstellungen für das Standard-Input-Panel, das erscheint, wenn der Benutzer einen neuen Wert für die im Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket bereitgestellten Controls eingibt. Es speichert auch die Konfiguration der Maßeinheiten, bietet Funktionen für den Zugriff auf diese Konfiguration und führt Berechnungen zur Einheitenumrechnung durch.
CurveEditor [► 48]	Das CurveEditor-Control ermöglicht es dem Benutzer, eine Liste von Punkten auf seinem HMI-Bildschirm hinzuzufügen und zu bearbeiten. Dann kann das Control eine interpolierte Kurve berechnen, die durch jeden Punkt verläuft.
FavoriteBarControl [► 80]	Das FavoriteBarControl kann mit dem Control Responsive Navigation Bar in TwinCAT HMI verwendet werden. Der Zweck dieses Controls besteht darin, den Benutzern die Möglichkeit zu geben, beliebige Inhalte auf einfache Weise als Favoriten festzulegen, indem sie eine Verknüpfung erstellen, die immer auf dem FavoriteBarControl verfügbar ist.
InputBox [► 85]	InputBox ist ein Control, mit dem der Benutzer numerische oder boolesche Werte in das TwinCAT-HMI-Projekt eingeben kann.
ManualOperation [► 95]	Das ManualOperation-Control kann für Bildschirmtasten zum manuellen Bewegen einer Achse verwendet werden.
MeasurementUnitSelector [► 109]	Mit Hilfe des MeasurementUnitSelector-Controls kann der Benutzer auf die in der Datei Unit_Config.json konfigurierte Einheitenkonfiguration zugreifen und die Einstellungen für die Anzeigeeinheit auf dem Client ändern.
Monitoring [► 113]	Das Monitoring-Control wird verwendet, um mehrere FB_Monitoring-Objekte in der SPS von einem einzigen Framework-Control aus zu visualisieren und zu konfigurieren.

Name	Beschreibung
PfwSingleTempControl [► 121]	PfwSingleTempControl ermöglicht dem Anwender die einfache Anzeige und Steuerung einer Temperaturzone in TwinCAT HMI.
ProcessScheduler [► 126]	Das ProcessScheduler-Control ermöglicht dem Benutzer die Visualisierung und Bearbeitung von Wochenplänen, die von der SPS zur Ausführung beliebiger Prozesse verwendet werden können.
StateIndicator [► 133]	Das StateIndicator-Control zeigt verschiedene Zustände an und kann als LED-Anzeige verwendet werden.
Table [► 135]	Das Table-Control kann verwendet werden, um mehrere SPS-Variablen in einem paginierten Tabellenformat zu visualisieren.
TimerControl [► 146]	Das TimerControl kann verwendet werden, um ein FB_Timer-Symbol zu visualisieren und zu konfigurieren.
VariableViewControl [► 156]	VariableViewControl kann als Dashboard verwendet werden, um Live-Werte von mehreren Variablen auf dem Bildschirm zu visualisieren.
UserLoggingViewer [► 151]	UserLoggingViewer zeigt den Inhalt von Logdateien an, die von Plastic Controls und der Funktion LogUserChanges im Projekt erzeugt werden.

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Temperature

Hinweis: Verfügbar ab Version 12.6.0

Das NuGet-Paket **Beckhoff.TwinCAT.HMI.Plastic.Temperature** kann über das Paketverwaltungssystem NuGet in das TwinCAT-HMI-Projekt installiert werden.

Name	Beschreibung
ZoneConfiguration [► 234]	Das ZoneConfiguration-Control wird verwendet, um verschiedene Betriebsparameter der in der SPS konfigurierten Temperaturgruppen und -zonen zu visualisieren und dem Benutzer die Möglichkeit zu geben, diese zu ändern.
ZonelImageLayout [► 247]	Mit dem Framework-Control ZonelImageLayout können Benutzer Temperaturzonen über einem Bildhintergrund in einem Layout darstellen, das die physische Anordnung auf der Maschine nachbildet.
ZoneGrouping [► 241]	Das ZoneGrouping-Control wird verwendet, um die Konfiguration der Temperaturgruppen im Tc3 Plastic Application-Projekt zu visualisieren und dem Benutzer die Möglichkeit zu geben, diese zu ändern.
PfwTempParameters [► 226]	Das Control PfwTempParameters ist mittlerweile obsolet, es funktioniert mit der SPS-Temperaturschnittstelle aus der Bibliothek Tc3_PlasticaApplication Version 1.5.
TemperatureScheduler [► 228]	Mit dem TemperatureScheduler-Control kann der Benutzer wöchentliche Zeitpläne für die Aktivierung von Temperaturgruppen anzeigen und bearbeiten.

4.1 ArrowMotionGraph

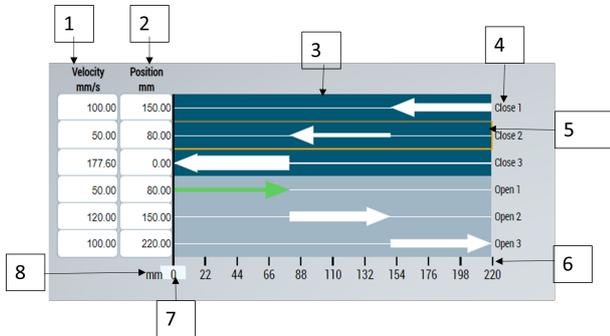
NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

Mit dem ArrowMotionGraph-Control können Benutzer auf Konfigurationen verschiedener Segmente von Achsenbewegungen zugreifen. Positionen und Geschwindigkeiten für die verschiedenen Segmente in jeder Bewegung werden als Pfeile auf das Control angezeigt; die Breite jedes Pfeils ist proportional zur Geschwindigkeit des jeweiligen Segments.

Aktion ActivateConfig

i Alle neuen Änderungen am Control müssen über die Aktion [ActivateConfig](#) [▶ 24] aktiviert werden, um neue Werte an die SPS zu senden.



1. Legt die Geschwindigkeit für die Achsensegmente fest
2. Legt die Position für die Achsensegmente fest
3. Achsbewegungen Segmente Visualisierung
4. Namen der Segmente
5. Ausgewähltes Segment hervorgehoben
6. Lineare Skala
7. Istposition der Achse
8. Einheit Skala

4.1.1 Attribute

Kategorie: Configuration

Name	Beschreibung
Achse [▶ 17]	Symbol für die Achse fb.
ColumnWidths [▶ 19]	Breite für jede der 4 Spalten.
RowHeight [▶ 22]	Zeilenhöhe der einzelnen angezeigten Zeilen.
SegmentNames [▶ 22]	Segmentbeschriftungstext für jedes Bewegungssegment.
Activation Required [▶ 17]	Wird auf <code>TRUE</code> gesetzt, wenn der Benutzer neue Änderungen vornimmt, die nicht aktiviert sind.
AxisUnitTypes [▶ 19]	Liste der Einheitentypen, die im Control verwendet werden können.
RowHeight Unit [▶ 22]	Legt die Einheit der RowHeight-Eigenschaft immer in Pixel fest.
InvertScale [▶ 21]	Bestimmt die Richtung der Skala auf dem Control.
CurrentValue [▶ 20]	Die Istposition der verknüpften Achse.
DigitAfterDecimal [▶ 20]	Dezimalgenauigkeit der auf dem Control angezeigten Werte.

Kategorie: Selection

Name	Beschreibung
SelectedRowSegFunction [▶ 23]	Eigenschaft der Segmentfunktion für das ausgewählte Segment.

Name	Beschreibung
DisplaySubPage [▶ 20]	Ermöglicht den Wechsel zwischen den Unterseiten Target , Dynamic oder Limiting .
SelectedMoveStartFunction [▶ 22]	Bewegt die Eigenschaft der Startfunktion der ausgewählten Bewegung.
SelectedMoveEndFunction [▶ 22]	Bewegt die Eigenschaft der Endfunktion der ausgewählten Bewegung.
SelectedMovIndex [▶ 23]	Der ausgewählte Move-Index.
SelectedSegIndex [▶ 23]	Der ausgewählte Segmentindex.

Recipe Management:

Name	Beschreibung
RecipeManagement [▶ 21]	Legen Sie fest, ob das Control Symbole, die ihm zugeordnet sind, zu einem Rezepttyp hinzufügt.
RecipeTypeName [▶ 21]	Mit diesem Attribut kann der Benutzer den Namen des Rezepttyps festlegen, zu dem das Control die ihm zugeordneten Symbole hinzufügt und alle Änderungen einlernt.

User Changes Logging:

Name	Beschreibung
LogChangesByUser [▶ 21]	Logging der vom Benutzer auf dem Client vorgenommenen Änderungen in einer Logdatei.
ChangeLogDescriptor [▶ 19]	Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

Sehen Sie dazu auch

- FirstMoveDirection (veraltet) [[▶ 20](#)]

4.1.1.1 Activation Required

Wird auf `TRUE` gesetzt, wenn der Benutzer neue Änderungen vornimmt, die nicht aktiviert sind.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getActivationRequired

Verfügbar: ab Version 12.6.0

4.1.1.2 Achse

Symbol von `fbAxis`, das mit diesem Control angezeigt und konfiguriert wird. Das an dieses Attribut gebundene Achsensymbol muss eine `variable` Struktur haben, wie nachstehend erläutert.

Die Struktur der obersten Ebene muss die unten aufgeführten Eigenschaften aufweisen:

Name	Typ	Beschreibung
ActPosition	Number	Aktuelle Position der Achse
ActiveMove	Integer	Index der aktuellen Bewegung
ActiveSegment	Integer	Index des Segmentes, das aktiv ist
MaxAcceleration	Number	Maximale Beschleunigung
MaxDeceleration	Number	Maximale Verzögerung
MaxPosition	Number	Maximalgrenze Position
MaxVelocity	Number	Maximalgrenze Geschwindigkeit
MinPosition	Number	Minimalgrenze Position

Name	Typ	Beschreibung
MinVelocity	Number	Minimalgrenze Geschwindigkeit
fbAxisMove	ARRAY [1..2] von 'Move Config'	Array von Bewegungen

<ul style="list-style-type: none"> PLC1.MAIN.fb_AMG <ul style="list-style-type: none"> ActPosition ActiveMove ActiveSegment MaxAcceleration MaxDeceleration MaxPosition MaxVelocity MinPosition MinVelocity SetPosition fbAxisMove 	<ul style="list-style-type: none"> PLC1.FB_AMGTest PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-SET-INT PLC1.PROPERTY-GET-SET-INT PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-SET-LREAL PLC1.FB_AxisMoveHmi 	<ul style="list-style-type: none"> PLC1::MAIN::fb_AMG ActPosition ActiveMove ActiveSegment MaxAcceleration MaxDeceleration MaxPosition MaxVelocity MinPosition MinVelocity SetPosition fbAxisMove
---	---	---

Jede Move-Konfigurationsstruktur oder jeder Funktionsbaustein muss die unten aufgeführten Eigenschaften aufweisen:

Name	Typ	Beschreibung
EndFunktion	Integer	Wert der Move-Endfunktion.
StartFunktion	Integer	Wert der Move-Startfunktion.
fbSegments	ARRAY [1..5] von 'Segment Config'	Array von Bewegungssegmenten.

<ul style="list-style-type: none"> fbAxisMove <ul style="list-style-type: none"> fbMoves <ul style="list-style-type: none"> [0] <ul style="list-style-type: none"> EndFunktion StartFunktion fbSegments <ul style="list-style-type: none"> [1] 	<ul style="list-style-type: none"> PLC1.FB_AxisMoveHmi PLC1.ARRAY_1..2_OF-FB_MoveHmiCfg PLC1.FB_MoveHmiCfg PLC1.PROPERTY-GET-SET-INT PLC1.PROPERTY-GET-SET-INT PLC1.ARRAY_1..5_OF-FB_SegHmiCfg PLC1.FB_MoveHmiCfg 	<ul style="list-style-type: none"> fbAxisMove fbMoves [0] EndFunktion StartFunktion fbSegments [1]
---	--	---

Jede Segment-Konfigurationsstruktur oder jeder Funktionsbaustein muss die unten aufgeführten Eigenschaften aufweisen:

Name	Typ	Beschreibung
Acceleration	Number	Beschleunigungs-Sollwert für das Segment
Deceleration	Integer	Verzögerungs-Sollwert für das Segment
InUse	BOOL	Diese Eigenschaft wird auf FALSE gesetzt, wenn ein Segment in der aktuellen Konfiguration nicht verwendet wird.
Position	Number	Positions-Sollwert für das Segment
SegFunction	Integer	Segmentfunktionswert für das Segment
Velocity	Number	Geschwindigkeits-Sollwert für das Segment

fbAxisMove	PLC1.FB_AxisMoveHmi	<input checked="" type="checkbox"/>	fbAxisMove
fbMoves	PLC1.ARRAY_1..2_OF-FB_MoveHmiCfg	<input checked="" type="checkbox"/>	fbMoves
[0]	PLC1.FB_MoveHmiCfg	<input checked="" type="checkbox"/>	[0]
EndFunction	PLC1.PROPERTY-GET-SET-INT	<input checked="" type="checkbox"/>	EndFunction
StartFunction	PLC1.PROPERTY-GET-SET-INT	<input checked="" type="checkbox"/>	StartFunction
fbSegments	PLC1.ARRAY_1..5_OF-FB_SegHmiCfg	<input checked="" type="checkbox"/>	fbSegments
[0]	PLC1.FB_SegHmiCfg	<input checked="" type="checkbox"/>	[0]
Acceleration	PLC1.PROPERTY-GET-SET-LREAL	<input checked="" type="checkbox"/>	Acceleration
Deceleration	PLC1.PROPERTY-GET-SET-LREAL	<input checked="" type="checkbox"/>	Deceleration
InUse	PLC1.PROPERTY-GET-SET-BOOL	<input checked="" type="checkbox"/>	InUse
Position	PLC1.PROPERTY-GET-SET-LREAL	<input checked="" type="checkbox"/>	Position
SegFunction	PLC1.PROPERTY-GET-SET-INT	<input checked="" type="checkbox"/>	SegFunction
Velocity	PLC1.PROPERTY-GET-SET-LREAL	<input checked="" type="checkbox"/>	Velocity
[1]	PLC1.FB_SegHmiCfg	<input checked="" type="checkbox"/>	[1]
[2]	PLC1.FB_SegHmiCfg	<input checked="" type="checkbox"/>	[2]
[3]	PLC1.FB_SegHmiCfg	<input checked="" type="checkbox"/>	[3]
[4]	PLC1.FB_SegHmiCfg	<input checked="" type="checkbox"/>	[4]

Schema: tchmi:framework#/definitions/Symbol

Attribut-Getter: getAxis

Attribut-Setter: setAxis

Verfügbar: ab Version 12.6.0

4.1.1.3 AxisUnitTypes

Liste der Einheitentypen, die im Control verwendet werden können.

Schema: tchmi:framework#/definitions/AMGAxisUnits

Attribut-Getter: getAxisUnitTypes

Attribut-Setter: setAxisUnitTypes

Verfügbar: ab Version 12.6.0

4.1.1.4 ChangeLogDescriptor

Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

Schema: "tchmi:general#/definitions/STRING"

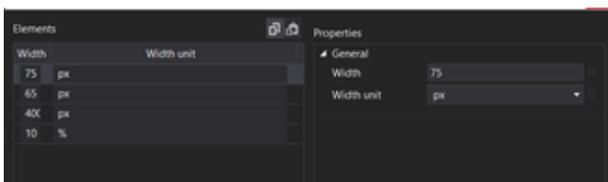
Attribut-Getter: getChangeLogDescriptor

Attribut-Setter: setChangeLogDescriptor

Verfügbar: ab Version 12.14.0

4.1.1.5 ColumnWidths

Breite für jede der 4 Spalten. Die Breiten können in Pixeln oder in Prozent angegeben werden.



Schema: tchmi: framework#/definitions/AMG_ColWidthList

Attribut-Getter: getColumnWidths

Attribut-Setter: setColumnWidths

Verfügbar: ab Version 12.6.0

4.1.1.6 CurrentValue

Aktuelle Position der Achse.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getCurrentValue

Attribut-Setter: setCurrentValue

Verfügbar: ab Version 12.8.0

4.1.1.7 DigitAfterDecimal

Dezimalgenauigkeit aller auf dem Control angezeigten numerischen Werte.

Schema: tchmi:general#/definitions/UINT

Attribut-Getter: getDigitsAfterDecimal

Attribut-Setter: setDigitsAfterDecimal

Verfügbar: ab Version 12.8.0

4.1.1.8 DisplaySubPage

Mit diesem Attribut kann der Benutzer zwischen der Unterseite **Target**, **Dynamic** oder **Limiting** des Controls wechseln.

- Auf der Unterseite **Target** werden Geschwindigkeits- und Positionswerte angezeigt.
- Die Unterseite **Dynamic** zeigt die Beschleunigungs- und Verzögerungswerte für jedes Segment.
- Auf der Unterseite **Limiting** kann der Benutzer Ruck- und Grenzwerte für jedes Segment sehen.

Schema: tchmi:general#/definitions/DisplaySubPage

Attribut-Getter: getDisplaySubPage

Attribut-Setter: setDisplaySubPage

Verfügbar: ab Version 12.6.0

4.1.1.9 FirstMoveDirection (veraltet)



Veraltetes Attribut

Dieses Attribut wird entfernt und kann nicht mehr verwendet werden.

Mit diesem Attribut wird die Richtung der ersten Bewegung auf dem Control festgelegt. Es gibt zwei Richtungen, die in diesem Attribut festgelegt werden können:

- In der Richtung **TowardsZero** beginnen die ersten Bewegungspfeile an der oberen Grenze der Achse.
- In der Richtung **AwayFromZero** beginnen die ersten Bewegungspfeile beim Minimalwert und zeigen auf die obere Grenze der Achse.

Schema: tchmi:framework#/definitions/AMGMoveDirection

Attribut-Getter: getFirstMoveDirection

Attribut-Setter: setFirstMoveDirection

Verfügbar: ab Version 12.6.0

Veraltet: ab Version 12.8.0

4.1.1.10 InvertScale

Die Skala am unteren Rand des Controls wird invertiert. Sie zeigt die maximale Position am linken Ende und die minimale Position am rechten Ende an.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getInvertScale

Attribut-Setter: setInvertScale

Verfügbar: ab Version 12.8.0

4.1.1.11 LogChangesByUser

Wenn dieses Attribut auf `TRUE` gesetzt ist, loggt das Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket jede Änderung durch den Benutzer auf dem HMI-Bildschirm in einer Logdatei.

Um dieses Attribut zu verwenden, muss [UserLoggerSymbol](#) [► 45] mit einem Binding an [FB_UserLoggingHmi](#) konfiguriert werden.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getLogUserChanges

Attribut-Setter: setLogUserChanges

Verfügbar: ab Version 12.12.0

4.1.1.12 RecipeManagement

Legen Sie fest, ob das Control die Symbole, die ihm zugeordnet sind, zu einem Rezepttyp hinzufügen soll.

Wenn das Projekt das [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper](#) [► 198]-Paket enthält, fügt das Control die Serversymbole, die darauf abgebildet sind, dem Rezepttyp hinzu, der durch das Attribut [RecipeTypeName](#) [► 21] definiert ist.

Nach jedem Benutzerwechsel lernt das Paket [RecipeHelper](#) das aktuell aktive Rezept ein.

Schema: "tchmi:framework#/definitions/TcHmi.Controls.Plastic.RecipeManagement"

Attribut-Getter: getRecipeManagement

Attribut-Setter: setRecipeManagement

Verfügbar: ab Version 12.14.0

4.1.1.13 RecipeTypeName

Mit diesem Attribut kann der Benutzer den Namen des Rezepttyps festlegen, zu dem das Control die ihm zugewiesenen Symbole hinzufügt und alle Änderungen speichert.

Damit das Control Symbole zum Rezepttyp hinzufügen kann, muss das Projekt das Paket [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper](#) [► 198] enthalten und das Attribut [RecipeManagement](#) [► 21] muss auf `IncludeToRecipe` gesetzt sein.



Wenn ein Symbol bereits Teil eines Rezepttyps ist, hat diese Eigenschaft keine Auswirkungen auf die Rezepte. Nur für das Hinzufügen von Symbolen, die nicht Teil eines Rezepttyps sind, wird das Attribut `RecipeTypeName` berücksichtigt.

Schema: "tchmi:general#/definitions/STRING"

Attribut-Getter: getRecipeTypeName

Attribut-Setter: setRecipeTypeName

Verfügbar: ab Version 12.14.0

4.1.1.14 RowHeight

Zeilenhöhe der einzelnen angezeigten Zeilen.

Schema: tchmi:general#/definitions/MeasurementValue

Attribut-Getter: getRowHeight

Attribut-Setter: setRowHeight

Verfügbar: ab Version 12.6.0

4.1.1.15 RowHeight Unit

Legt die Einheit der RowHeight-Eigenschaft immer in Pixel fest.

Schema: tchmi:framework#/definitions/PixelUnit

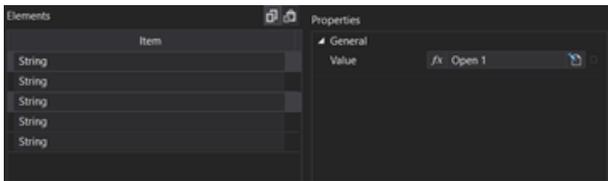
Attribut-Getter: getRowHeightUnit

Attribut-Setter: setRowHeightUnit

Verfügbar: ab Version 12.6.0

4.1.1.16 SegmentNames

Segmentbeschriftungstext für jedes Bewegungssegment. Das Array oder der String kann mit Lokalisierungsschlüsseln konfiguriert werden.



Schema: tchmi:framework#/definitions/SegNames

Attribut-Getter: getSegmentNames

Attribut-Setter: setSegmentNames

Verfügbar: ab Version 12.6.0

4.1.1.17 SelectedMoveEndFunction

Zeigt die Eigenschaft **EndFunction** für die ausgewählte Achsenbewegung an.

Wird ein Wert ungleich Null gesetzt, wird am Endpunkt der Bewegung ein Kreiselement hinzugefügt.



Schema: tchmi:general#/definitions/INT

Attribut-Getter: getSelectedMoveEndFunction

Attribut-Setter: setSelectedMoveEndFunction

Verfügbar: ab Version 12.6.0

4.1.1.18 SelectedMoveStartFunction

Zeigt die Eigenschaft **StartFunction** für die ausgewählte Achsenbewegung an.

Wenn ein Wert ungleich Null gesetzt wird, wird ein kreisförmiges Element am Anfang der Bewegung hinzugefügt.



Schema: tchmi:general#/definitions/INT
Attribut-Getter: getSelectedMoveStartFunction
Attribut-Setter: setSelectedMoveStartFunction
Verfügbar: ab Version 12.6.0

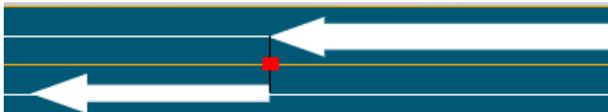
4.1.1.19 SelectedMovIndex

Der ausgewählte Move-Index.

Schema: tchmi:general#/definitions/UINT
Attribut-Getter: getSelectedMovIndex
Verfügbar: ab Version 12.6.0

4.1.1.20 SelectedRowSegFunction

Die Eigenschaft der Segmentfunktion für das ausgewählte Bewegungssegment. Der Höchstwert der Segmentfunktion ist 10. Ein Segment mit einem Segmentfunktionswert ungleich Null zeigt ein quadratisches Element an der Segmentposition.



Schema: tchmi:general#/definitions/INT
Attribut-Getter: getSelectedRowSegFunction
Attribut-Setter: setSelectedRowSegFunction
Verfügbar: ab Version 12.6.0

4.1.1.21 SelectedSegIndex

Der ausgewählte Segmentindex.

Schema: tchmi:general#/definitions/UINT
Attribut-Getter: getSelectedSegIndex
Verfügbar: ab Version 12.6.0

4.1.2 Funktionen

Name	Beschreibung
ActivateConfig [► 24]	Mit dieser Funktion werden die aktuellen Änderungen des Controls in die SPS geschrieben.
ResetConfigData [► 24]	Mit dieser Funktion werden alle Änderungen des Benutzers rückgängig gemacht, indem die zuletzt aktivierte Konfiguration erneut aus das Control geladen wird.
AddNewAxisSegment [► 24]	Diese Funktion fügt ein neues Achsenssegment in das Control ein.

Name	Beschreibung
RemoveAxisSegment [▶ 25]	Mit dieser Funktion wird das ausgewählte Achsensegment aus dem Control entfernt.

4.1.2.1 ActivateConfig

Durch den Aufruf dieser Funktion werden alle Benutzeränderungen aktiviert und in die SPS-Symbole geschrieben.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.1.2.2 ResetConfigData

Mit dieser Funktion werden alle Änderungen des Benutzers rückgängig gemacht, indem die zuletzt aktivierte Konfiguration erneut geladen wird.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: ab Version 12.6.0

4.1.2.3 AddNewAxisSegment

Diese Funktion fügt dem Control ein neues Achsensegment am gewählten MoveIndex und am gewählten SegmentIndex hinzu.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: ab Version 12.6.0

4.1.2.4 RemoveAxisSegment

Mit dieser Funktion wird das ausgewählte Achsenssegment aus dem Control entfernt. Wenn ein Segment, auf das weitere Segmente folgen, entfernt wird, werden die Segmente nach den entfernten Segmenten um eine Ebene nach oben verschoben, so dass keine Lücke in der Segmentliste entsteht.

Parameter

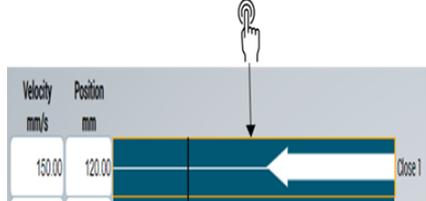
Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: ab Version 12.6.0

4.1.3 Benutzerinteraktionen

Name	Beschreibung	Ergebnis im Control
Einmaliges Antippen	Wählen Sie ein Segment aus, indem Sie es antippen/anklicken.	 <p>The screenshot shows a control interface with two columns: 'Velocity mm/s' and 'Position mm'. The values are 150.00 and 120.00 respectively. Below these is a blue bar with a white arrow pointing left. A hand icon is shown tapping the bar. A 'Close 1' button is visible on the right.</p>

Verfügbar: ab Version 12.6.0

4.2 BlowPressureChart

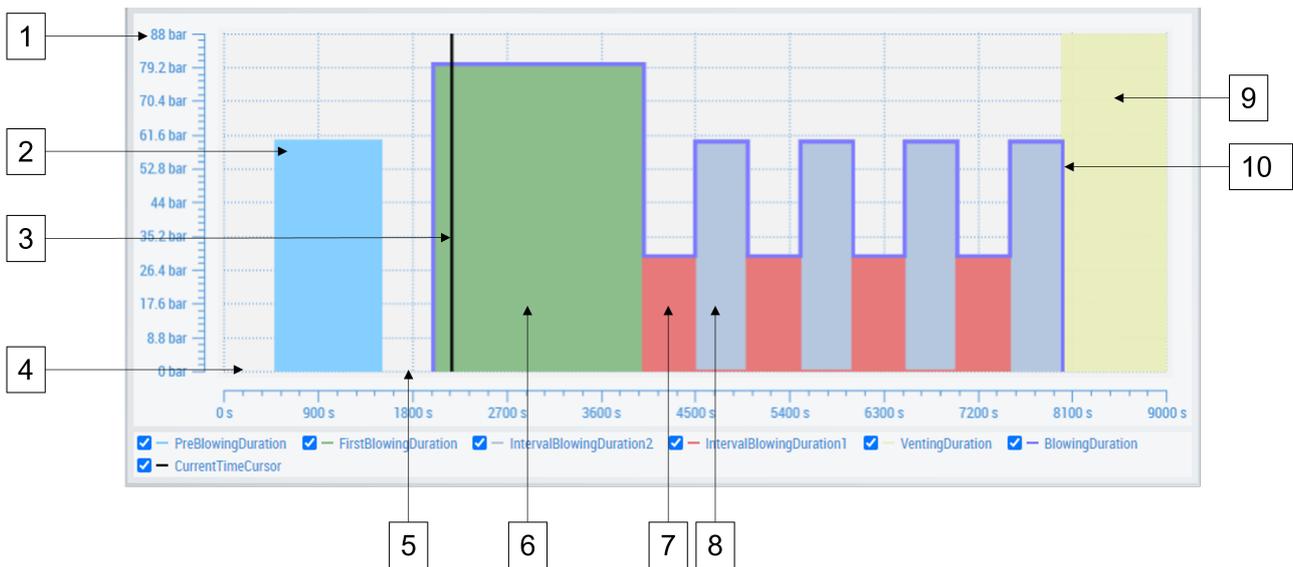
NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

BlowPressureChart ist ein spezielles Chart-Type-Framework-Control, das den Blasprozess in einer Blasformmaschine visualisieren kann.

Normalerweise ist der Blasprozess in verschiedene Abschnitte unterteilt, wobei jeder Abschnitt für eine bestimmte Dauer und mit einem bestimmten Druck Luft einbläst.

Das Druck- Zeit-Diagramm des BlowPressureChart ermöglicht es dem Benutzer, die Auswirkungen verschiedener Einstellungen auf dem Bildschirm zu sehen. Eine Cursorlinie zeigt die aktuelle Phase des Prozesses an, wenn sie aktiv ist.



1. Druck-Achse
2. Dauer und Druck des Vorblasens
3. Cursor für die aktuelle Zeit
4. Vorblasverzögerung
5. Blasverzögerung
6. Dauer und Druck des ersten Blasens
7. Intervall-Blasen Dauer 1 und Druck
8. Intervall-Blasen Dauer 2 und Druck
9. Dauer der Entlüftung
10. Gesamtdauer des Blasens.

4.2.1 Attribute

Kategorie: Colors

Name	Beschreibung
PreBlowingDurationColor [▶ 30]	Legt die Farbe für die Kurve der Vorblasdauer fest.
BlowingDurationColor [▶ 27]	Legt die Farbe für die Kurve der Blasdauer fest.
FirstBlowingDurationColor [▶ 29]	Legt die Farbe der Kurve für die erste Blasdauer fest.
IntervalBlowingDuration1Color [▶ 29]	Legt die Farbe der Kurve für die Dauer 1 des Intervall-Blasens fest.
IntervalBlowingDuration2Color [▶ 30]	Legt die Farbe der Kurve für die Dauer 2 des Intervall-Blasens fest.
CurrentTimeCursorlineColor [▶ 28]	Legt die Farbe der Cursorlinie der aktuellen Zeit fest.
VentingDurationColor [▶ 31]	Legt die Farbe für die Entlüftungsdauer fest.

Kategorie: Time

Name	Beschreibung
DelayPreBlowing [▶ 28]	Legt die Verzögerung für den Vorblaswert fest.
DurationPreBlowing [▶ 29]	Legt die Dauer für den Vorblaswert fest.
DelayBlowing [▶ 28]	Legt die Verzögerung für den Blaswert fest.
BlowingDuration [▶ 27]	Legt die Gesamtdauer des Blasens fest.
FirstBlowingDuration [▶ 29]	Legt den Wert für das erste Blasenintervall fest.
IntervalBlowingDuration1 [▶ 29]	Legt die Dauer des ersten Dauerwerts des Intervall-Blasens fest.

Name	Beschreibung
IntervalBlowingDuration2 [▶ 30]	Legt die Dauer des zweiten Dauerwerts des Intervall-Blasens fest.
VentingDuration [▶ 30]	Legt den Wert für die Entlüftungsdauer fest.
CurrentTimeCursor [▶ 28]	Cursor der aktueller Zeit im Blasdruckdiagramm.

Kategorie: Pressure

Name	Beschreibung
PreBlowPressure [▶ 30]	Legt den Wert für den Vorblasdruck fest.
BlowPressure [▶ 27]	Legt den Wert für den ersten Blasdruck fest.
BlowPressValue1 [▶ 27]	Legt den Blasdruckwert für das Intervall 1 Blasen fest.
BlowPressValue2 [▶ 28]	Legt den Blasdruckwert für das Intervall 2 Blasen fest.

Kategorie: Common

Name	Beschreibung
FillChart [▶ 29]	Wenn <code>TRUE</code> , wird der Bereich unterhalb des Diagramms hervorgehoben

4.2.1.1 BlowingDuration

Legt die Gesamtdauer des Blasens fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getBlowingDuration

Attribut-Setter: setBlowingDuration

Verfügbar: seit Version 12.6.0

4.2.1.2 BlowingDuration Color

Legt die Farbe für die Kurve der Blasdauer fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getBlowingDurationColor

Attribut-Setter: setBlowingDurationColor

Verfügbar: seit Version 12.6.0

4.2.1.3 BlowPressure

Legt den Wert für den ersten Blasdruck fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getBlowPressure

Attribut-Setter: setBlowPressure

Verfügbar: seit Version 12.6.0

4.2.1.4 BlowPressValue1

Legt den Blasdruckwert für das Intervall 1 Blasen fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: setBlowPressValue1

Attribut-Setter: getBlowPressValue1

Verfügbar: seit Version 12.6.0

4.2.1.5 **BlowPressValue2**

Legt den Blasdruckwert für das Intervall 2 Blasen fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getBlowPressValue2

Attribut-Setter: setBlowPressValue2

Verfügbar: seit Version 12.6.0

4.2.1.6 **CurrentTimeCursor**

Cursor der aktueller Zeit im Blasdruckdiagramm.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getCurrentTimeCursor

Attribut-Setter: setCurrentTimeCursor

Verfügbar: seit Version 12.6.0

4.2.1.7 **CurrentTimeCursorline Color**

Legt die Farbe der Cursorlinie der aktuellen Zeit fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getCurrentTimeCursorlineColor

Attribut-Setter: setCurrentTimeCursorlineColor

Verfügbar: seit Version 12.6.0

4.2.1.8 **DelayBlowing**

Legt den Wert für die Blasverzögerung fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getDelayBlowing

Attribut-Setter: setDelayBlowing

Verfügbar: seit Version 12.6.0

4.2.1.9 **DelayPreBlowing**

Legt die Verzögerung des Vorblaswertes fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getDelayPreBlowing

Attribut-Setter: setDelayPreBlowing

Verfügbar: seit Version 12.6.0

4.2.1.10 DurationPreBlowing

Legt die Dauer für den Vorblaswert fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getDurationPreBlowing

Attribut-Setter: setDurationPreBlowing

Verfügbar: seit Version 12.6.0

4.2.1.11 FillChart

Wenn `TRUE`, wird der Bereich unterhalb des Diagramms hervorgehoben.

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getFillChart

Attribut-Setter: setFillChart

Verfügbar: seit Version 12.6.0

4.2.1.12 FirstBlowingDuration

Legt den Wert für das erste Blasenintervall fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getFirstBlowingDuration

Attribut-Setter: setFirstBlowingDuration

Verfügbar: seit Version 12.6.0

4.2.1.13 FirstBlowingDuration Color

Legt die Farbe der Kurve für die erste Blasdauer fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getFirstBlowingDurationColor

Attribut-Setter: setFirstBlowingDurationColor

Verfügbar: seit Version 12.6.0

4.2.1.14 IntervalBlowingDuration1

Legt die Dauer des ersten Dauerwerts des Intervall-Blasens fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getIntervalBlowingDuration1

Attribut-Setter: setIntervalBlowingDuration1

Verfügbar: seit Version 12.6.0

4.2.1.15 IntervalBlowingDuration1 Color

Legt die Farbe der Kurve für die Dauer 1 des Intervall-Blasens fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getIntervalBlowingDuration1Color

Attribut-Setter: setIntervalBlowingDuration1Color

Verfügbar: seit Version 12.6.0

4.2.1.16 IntervalBlowingDuration2

Legt die Dauer des zweiten Dauerwerts des Intervall-Blasens fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getIntervalBlowingDuration2

Attribut-Setter: setIntervalBlowingDuration2

Verfügbar: seit Version 12.6.0

4.2.1.17 IntervalBlowingDuration2 Color

Legt die Farbe der Kurve für die Dauer 2 des Intervall-Blasens fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getIntervalBlowingDuration2Color

Attribut-Setter: setIntervalBlowingDuration2Color

Verfügbar: seit Version 12.6.0

4.2.1.18 PreBlowingDuration Color

Legt die Farbe für die Kurve der Vorblasdauer fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getPreBlowingDurationColor

Attribut-Setter: setPreBlowingDurationColor

Verfügbar: seit Version 12.6.0

4.2.1.19 PreBlowPressure

Legt den Wert für den Vorblasdruck fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getPreBlowPressure

Attribut-Setter: setPreBlowPressure

4.2.1.20 VentingDuration

Legt den Wert für die Entlüftungsdauer fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: setVentingDurationColor

Attribut-Setter: getVentingDurationColor

Verfügbar: seit Version 12.6.0

4.2.1.21 VentingDuration Color

Legt die Farbe für die Entlüftungsdauer fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getVentingDurationColor

Attribut-Setter: setVentingDurationColor

Verfügbar: seit Version 12.6.0

4.3 CamControl

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

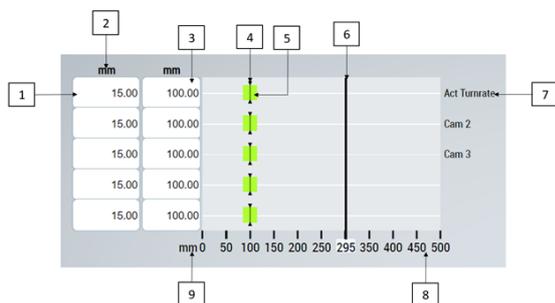
Das CamControl dient zur Visualisierung und Konfiguration von Nockenschaltern. Jeder Nockenschalter wird durch die Werte "Hysteresis" und "Threshold" definiert.

Threshold definiert die Position des Nockenschalters auf einer Achse ab Null.

Hysteresis ist die Länge, die zum Schwellenwert addiert und subtrahiert wird, um einen Bereich auf der Achse zu markieren. Liegt die Achsposition in diesem Bereich, sollte der zugehörige Nocken eingeschaltet werden.

● Aktion aktivieren

i Alle neuen Änderungen am Control müssen über die Aktion [ActivateConfig](#) [► 36] aktiviert werden, um neue Werte an die SPS zu senden.



1. Wert Hysteresis
2. Anzeigeeinheit der Werte
3. Wert Threshold
4. Visualisierung des Wertes Threshold
5. Visualisierung der Hysteresesezone
6. Aktuelle Position der Achse
7. Beschreibungstext für jeden Nockenschalter
8. Skala Achse
9. Einheit Skala

4.3.1 Attribute

Kategorie: Configuration

Name	Beschreibung
MinValue [► 35]	Mindestwert der Skala.
MaxValue [► 35]	Höchstwert der Skala.

Name	Beschreibung
CamConfig [► 33]	Liste der Nockenschalterkonfigurationen.
ColumnWidths [► 34]	Breite für jede der 4 Spalten.
RowHeight [► 36]	Höhe jeder Nockenschalterzeile in Pixel.
RowHeight Unit [► 22]	Legt die Einheit der Eigenschaft RowHeight immer in Pixel fest.
CamDescriptor [► 33]	Liste der Deskriptorstrings, die neben jeder Nockenschalterzeile angezeigt werden.
AxisUnitTypes [► 19]	Liste der Einheitentypen, die im Control verwendet werden können.
InvertScale [► 34]	Bestimmt die Richtung der Skala, die auf dem Control angezeigt wird.
DigitAfterDecimal [► 34]	Dezimalgenauigkeit aller auf dem Control angezeigten numerischen Werte.

Kategorie: State

Name	Beschreibung
CurrentValue [► 34]	Aktuelle Position der Achse.

Kategorie: Common

Name	Beschreibung
ActivationRequired [► 32]	Wird auf <code>TRUE</code> gesetzt, wenn der Benutzer neue Änderungen vornimmt, die nicht vom Benutzer aktiviert sind.

Recipe Management:

Name	Beschreibung
RecipeManagement [► 35]	Legen Sie fest, ob das Control Symbole, die ihm zugeordnet sind, zu einem Rezepttyp hinzufügt.
RecipeTypeName [► 35]	Mit diesem Attribut kann der Benutzer den Namen des Rezepttyps festlegen, zu dem das Control die ihm zugeordneten Symbole hinzufügt und alle Änderungen einlernt.

User Changes Logging:

Name	Beschreibung
LogChangesByUser [► 34]	Logging der vom Benutzer auf dem Client vorgenommenen Änderungen in einer Logdatei.
ChangeLogDescriptor [► 33]	Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

4.3.1.1 ActivationRequired

Dieses schreibgeschützte Attribut wird auf `TRUE` gesetzt, wenn ein Benutzer neue Änderungen vornimmt, die nicht aktiviert sind.

Dieser Wert kann von anderen Controls, wie z. B. einem `TchmiButton`, verwendet werden, um sein Aussehen zu ändern und den Benutzer darüber zu informieren, dass neue Änderungen aktiviert werden müssen.

Schema: `tchmi:general#/definitions/Boolean`

Attribut-Getter: `getActivationRequired`

Verfügbar: seit Version 12.6.0

4.3.1.2 AxisUnitTypes

Liste der Einheitentypen, die im Control verwendet werden können.

Schema: tchmi:framework#/definitions/AMGAxisUnits

Attribut-Getter: getAxisUnitTypes

Attribut-Setter: setAxisUnitTypes

Verfügbar: seit Version 12.6.0

4.3.1.3 CamConfig

Dieses Attribut sollte mit einem Serversymbol verknüpft werden, das den Array der Nockenkonfigurationsobjekte (Strukturen oder Funktionsbausteine) in der SPS darstellt. Das verknüpfte Symbol muss vom Typ Array mit einer Mindestlänge von 1 und einer Höchstlänge von 5 sein.

Jedes Element in diesem Array muss die folgenden Eigenschaften aufweisen.

Name	Typ	Beschreibung
CamActive	Boolean	Wird von der SPS für einen Nockenschalter auf <code>TRUE</code> gesetzt, wenn der Nockenschalter aktiv ist.
Hysteresis	Number	Hysteresese-Eigenschaft eines Nockenschalters.
Threshold	Number	Schwellenwerteeigenschaft eines Nockenschalters.

Path	Value	Symbol	Symbol Path
PLC1.MAIN.fbCamTest.fbCamHmiCfg			PLC1::MAIN::fbCamTest::fbCamHmiCfg
[0]			[0]
CamActive			CamActive
Hysteresis			Hysteresis
Threshold			Threshold
[1]			[1]
[2]			[2]
[3]			[3]
[4]			[4]

Schema: tchmi:framework#/definitions/Symbol

Attribut-Getter: getCamConfig

Attribut-Setter: setCamConfig

Verfügbar: seit Version 12.6.0

4.3.1.4 CamDescriptor

Liste der Deskriptorstrings, die neben jeder Nockenschalterzeile angezeigt werden.

Schema: tchmi:framework#/definitions/CamNameList

Attribut-Getter: getCamDescriptor

Attribut-Setter: setCamDescriptor

Verfügbar: seit Version 12.6.0

4.3.1.5 ChangeLogDescriptor

Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

Schema: "tchmi:general#/definitions/STRING"

Attribut-Getter: getChangeLogDescriptor

Attribut-Setter: setChangeLogDescriptor

Verfügbar: ab Version 12.14.0

4.3.1.6 ColumnWidths

Die Breite der einzelnen Spalten kann über dieses Attribut konfiguriert werden.

Schema: tchmi:framework#/definitions/AMG_ColWidthList

Attribut-Getter: getColumnWidths

Attribut-Setter: setColumnWidths

Verfügbar: seit Version 12.6.0

4.3.1.7 CurrentValue

Aktuelle Position der Achse.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getCurrentValue

Attribut-Setter: setCurrentValue

Verfügbar: seit Version 12.6.0

4.3.1.8 DigitAfterDecimal

Dezimalgenauigkeit aller auf dem Control angezeigten numerischen Werte.

Schema: tchmi:general#/definitions/UINT

Attribut-Getter: getDigitsAfterDecimal

Attribut-Setter: setDigitsAfterDecimal

Verfügbar: ab Version 12.8.0

4.3.1.9 InvertScale

Die Skala am unteren Rand des Controls wird invertiert. Sie zeigt die maximale Position am linken Ende und die minimale Position am rechten Ende an.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getInvertScale

Attribut-Setter: setInvertScale

Verfügbar: ab Version 12.8.0

4.3.1.10 LogChangesByUser

Wenn dieses Attribut auf `TRUE` gesetzt ist, loggt das Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket jede Änderung durch den Benutzer auf dem HMI-Bildschirm in einer Logdatei.

Um dieses Attribut zu verwenden, muss [UserLoggerSymbol \[► 45\]](#) mit einem Binding an [FB UserLoggingHmi](#) konfiguriert werden.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getLogUserChanges

Attribut-Setter: setLogUserChanges

Verfügbar: ab Version 12.12.0

4.3.1.11 MaxValue

Dies ist der Höchstwert der Skala in der Systemeinheit.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getMaxValue

Attribut-Setter: setMaxValue

Verfügbar: seit Version 12.6.0

4.3.1.12 MinValue

Dies ist der Mindestwert der Skala in der Systemeinheit.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getMinValue

Attribut-Setter: setMinValue

Verfügbar: seit Version 12.6.0

4.3.1.13 RecipeManagement

Legen Sie fest, ob das Control die Symbole, die ihm zugeordnet sind, zu einem Rezepttyp hinzufügen soll.

Wenn das Projekt das [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper \[▸ 198\]](#)-Paket enthält, fügt das Control die Serversymbole, die darauf abgebildet sind, dem Rezepttyp hinzu, der durch das Attribut [RecipeTypeName \[▸ 35\]](#) definiert ist.

Nach jedem Benutzerwechsel lernt das Paket RecipeHelper das aktuell aktive Rezept ein.

Schema: "tchmi:framework#/definitions/Tchmi.Controls.Plastic.RecipeManagement"

Attribut-Getter: getRecipeManagement

Attribut-Setter: setRecipeManagement

Verfügbar: ab Version 12.14.0

4.3.1.14 RecipeTypeName

Mit diesem Attribut kann der Benutzer den Namen des Rezepttyps festlegen, zu dem das Control die ihm zugewiesenen Symbole hinzufügt und alle Änderungen speichert.

Damit das Control Symbole zum Rezepttyp hinzufügen kann, muss das Projekt das Paket [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper \[▸ 198\]](#) enthalten und das Attribut [RecipeManagement \[▸ 35\]](#) muss auf IncludeToRecipe gesetzt sein.



Wenn ein Symbol bereits Teil eines Rezepttyps ist, hat diese Eigenschaft keine Auswirkungen auf die Rezepte. Nur für das Hinzufügen von Symbolen, die nicht Teil eines Rezepttyps sind, wird das Attribut RecipeTypeName berücksichtigt.

Schema: "tchmi:general#/definitions/STRING"

Attribut-Getter: getRecipeTypeName

Attribut-Setter: setRecipeTypeName

Verfügbar: ab Version 12.14.0

4.3.1.15 RowHeight

Höhe jeder Nockenschalterzeile in Pixel.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getRowHeight

Attribut-Setter: setRowHeight

Verfügbar: seit Version 12.6.0

4.3.1.16 RowHeightUnit

Legt die Einheit der RowHeight-Eigenschaft immer in Pixel fest.

Schema: tchmi:framework#/definitions/PixelUnit

Attribut-Getter: getRowHeightUnit

Attribut-Setter: setRowHeightUnit

Verfügbar: ab Version 12.6.0

4.3.2 Funktionen

Kategorie: Actions

Name	Beschreibung
ActivateConfig [▶ 36]	Mit dieser Funktion werden die aktuellen Änderungen des Benutzers in die SPS geschrieben.
ResetConfigdata [▶ 36]	Diese Funktion macht alle Änderungen des Benutzers rückgängig und lädt die zuletzt aktivierte Konfiguration aus der SPS neu.

4.3.2.1 ActivateConfig

Durch den Aufruf dieser Funktion werden alle Benutzeränderungen aktiviert und in die SPS-Symbole geschrieben.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.3.2.2 ResetConfigdata

Mit dieser Funktion werden alle Änderungen des Benutzers rückgängig gemacht, indem die zuletzt aktivierte Konfiguration erneut geladen wird.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.3.3 Events

Kategorie: Actions

Name	Beschreibung
onUserConfigActivated [► 37]	Dieses Ereignis wird generiert, wenn eine neue Konfiguration auf dem Control aktiviert wird.
onEditingCanceled [► 37]	Dieses Ereignis wird generiert, wenn Benutzeränderungen rückgängig gemacht werden und die alte Konfiguration wieder auf das Control geladen wird.

4.3.3.1 onUserConfigActivated

Dieses Ereignis wird generiert, wenn eine neue Konfiguration auf dem Control aktiviert wird.

Verfügbar: seit Version 12.6.0

4.3.3.2 onEditingCanceled

Dieses Ereignis wird generiert, wenn Benutzeränderungen rückgängig gemacht werden und die alte Konfiguration wieder auf das Control geladen wird.

Verfügbar: seit Version 12.6.0

4.4 Configurator

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

Das Configurator-Control speichert die Benutzereinstellungen für das Standard-InputPanel, das erscheint, wenn der Benutzer einen neuen Wert für die im Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket bereitgestellten Controls eingibt.

Es speichert auch die Konfiguration der Maßeinheiten, bietet Funktionen für den Zugriff auf diese Konfiguration und führt Berechnungen zur Einheitenumrechnung durch.

● Die Position des Controls

I Damit alle anderen Controls auf das Configurator-Control zugreifen können, ist es erforderlich, dieses Control an oberster Stelle in die Steuerhierarchie aufzunehmen. Es wird empfohlen, es als erstes Control in der Ansicht einzustellen.

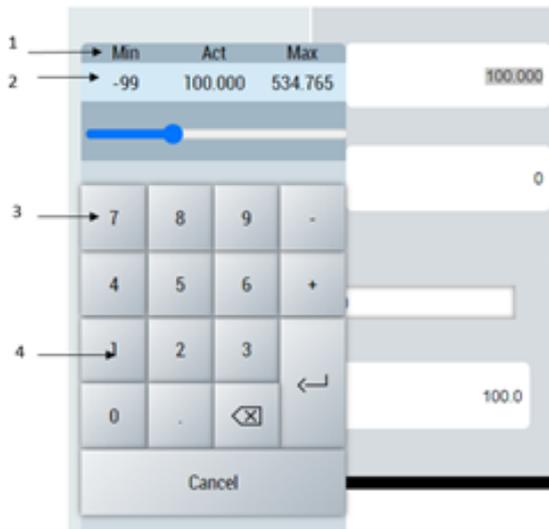


Input Panel

Colors	
Panel Background Colour	#FFA0B6C4
Text Background Color	#FFD3E9F6
Key Background Color	Theme
Key Label Color	#FF000000

Attribute zur Konfiguration des InputPanels:

1. Panel Background Color
2. Text Background Color
3. Key Background Color
4. Key Label Color



4.4.1 Attribute

Kategorie: Colors

Name	Beschreibung
PanelBackgroundColor [▶ 42]	Legt die Hintergrundfarbe für das Input Panel fest, das für die Bearbeitung der InputBox-Controls verfügbar ist.
TextBackgroundColor [▶ 44]	Legt die Hintergrundfarbe des auf dem Input Panel angezeigten Deskriptortextes fest.
KeyBackgroundColor [▶ 40]	Legt die Hintergrundfarbe der im Input Panel verwendeten Tastaturtasten fest.
KeyLabelColor [▶ 41]	Legt die Farbe der Beschriftung von Tastaturtasten fest, die im Input Panel verwendet werden.

Kategorie: InputPanel

Name	Beschreibung
KeyBoardLayout [▶ 40]	Legt das Layout der im Input Panel verwendeten Tastatur fest.
KeyBoardHeight [▶ 40]	Legt die Höhe der Tastatur in Pixeln fest.
KeyBoardWidth [▶ 41]	Legt die Breite der Tastatur in Pixeln fest.

Kategorie: Unit Conversion

Name	Beschreibung
UnitConfig [▶ 44]	Legt das Layout der im Input Panel verwendeten Tastatur fest.
DisplayUnitsChanged [▶ 40]	Wenn eine Eigenschaft der Anzeigeeinheit einer beliebigen Einheitengruppe geändert wird, ändert sich dieses Attribut.
SystemUnitsChanged [▶ 44]	Wird eine Eigenschaft der Systemeinheit einer beliebigen Einheitengruppe geändert, so ändert sich auch dieses Attribut.
ConfiguredPercentUnitID [▶ 40]	Gibt die ID der in der Datei Unit_Config.json konfigurierten <i>PercentUnit</i> zurück.
UnitGroups [▶ 44]	Gibt eine Liste der in der Datei Unit_Config.json konfigurierten Einheitengruppenobjekte zurück.
AllUnits [▶ 39]	Gibt eine Liste der in der Datei Unit_Config.json konfigurierten Einheiten zurück.

Kategorie: Navigation

Name	Beschreibung
NavigationConfig [▶ 41]	Mit diesem Attribut kann der Benutzer den Navigationsbaum konfigurieren und Informationen zu den Einstellungen hinzufügen, die in der Sliderregion angezeigt werden sollen, wenn der Zielinhalt in der Hauptregion der Anwendung angezeigt wird.
NavBarMenuData [▶ 41]	Dieses Attribut bietet eine abgeleitete Navigationskonfiguration, die an das MenuSrcData-Attribut des TcHmiNavigationBar-Controls gebunden werden kann.

Kategorie: PlcSymbols

Name	Beschreibung
RecipeHandshakeSymbol [▶ 43]	Symbol zur Instanz von FB_PlcStateToHmi.
TemperaturSymbol [▶ 44]	Symbol zur Instanz von FB_TempCtrlHmi.
UserLoggerSymbol [▶ 45]	Symbol zur Instanz von FB_UserLoggingHmi.

Kategorie: RecipeHelper

Name	Beschreibung
RecipeAutoSaveDelay [▶ 43]	Dieses Attribut definiert die Verzögerung in Sekunden, nach der RecipeHelper neue Änderungen im temporären Rezept speichert.
RecipeBackupDuration [▶ 43]	Mit diesem Attribut wird die Verzögerung in Minuten festgelegt, nach der die vom Benutzer vorgenommenen temporären Änderungen in einer Rezeptsicherungsdatei gespeichert werden.

4.4.1.1 AllUnits

Gibt eine Liste der in der Datei [Unit_Config.json](#) [[▶ 45](#)] konfigurierten Einheiten zurück.

Schema: tchmi:general#/definitions/Array

Attribut-Getter: getAllUnits

Verfügbar: ab Version 12.6.0

4.4.1.2 ConfiguredPercentUnitID

Gibt die in der JSON-Datei konfigurierte Unit ID von [PercentUnit](#) [► 45] zurück.

Schema: tchmi:general#/definitions/String

Attribut-Getter: getConfiguredPercentUnitID

Verfügbar: ab Version 12.6.0

4.4.1.3 DisplayUnitsChanged

Dieses Attribut ändert sich, wenn eine der Eigenschaften der Anzeigeeinheit einer beliebigen Einheitengruppe von einem Benutzer auf dem Client geändert wird.

Dieses Attribut kann als Ereignis für andere Controls verwendet werden, um einheitenabhängige Eigenschaften im Projekt neu zu berechnen.

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getDisplayUnitsChanged

Verfügbar: ab Version 12.6.0

4.4.1.4 KeyBackgroundColor

Legt die Hintergrundfarbe der im Input Panel verwendeten Tastaturtasten fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getKeyBackground

Attribut-Setter: setKeyBackground

Verfügbar: ab Version 12.6.0

4.4.1.5 KeyBoardHeight

Legt die Höhe der Tastatur in Pixeln fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getKeyBoardHeight

Attribut-Setter: setKeyBoardHeight

Verfügbar: ab Version 12.6.0

4.4.1.6 KeyBoardLayout

Legt das Layout der im Input Panel verwendeten Tastatur fest.

Das Standard-Layout ist ein vorkonfigurierter Ziffernblock mit Escape-Taste. Der Benutzer hat die Möglichkeit ein anderes Layout auszuwählen.

Schema: tchmi:framework#/definitions/TcHmi.Controls.Beckhoff.TcHmiKeyboard.ProjectKeyboardLayout

Attribut-Getter: getKeyBoardLayout

Attribut-Setter: setKeyBoardLayout

Verfügbar: ab Version 12.6.0

4.4.1.7 **KeyBoardWidth**

Legt die Breite der Tastatur in Pixeln fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getKeyBoardWidth

Attribut-Setter: setKeyBoardWidth

Verfügbar: ab Version 12.6.0

4.4.1.8 **KeyLabelColor**

Legt die Farbe der Beschriftung der im Input Panel verwendeten Tastaturtasten fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getKeyLabelColor

Attribut-Setter: setKeyLabelColor

Verfügbar: ab Version 12.6.0

4.4.1.9 **LogRecipeChangesByUser**

Mit diesem Attribut protokolliert der Recipe Helper die vom Benutzer vorgenommenen Rezeptänderungen im Log, indem er die Funktion [LogUserChanges \[▶ 171\]](#) aufruft.

Um dieses Attribut zu verwenden, muss [UserLoggerSymbol \[▶ 45\]](#) mit einem Binding an [FB_UserLoggingHmi](#) konfiguriert werden.

Schema: tchmi:general#/definitions/BOOL

Attribut-Setter: setLogRecipeChangesByUser

Attribut-Getter: getLogRecipeChangesByUser

Verfügbar: ab Version 12.14.0

4.4.1.10 **NavBarMenuData**

Die mit dem Attribut [NavigationConfig \[▶ 41\]](#) konfigurierte Navigationskonfiguration kann nicht direkt mit dem TcHmiNavigationBar-Control verwendet werden.

Das NavBarMenuData-Attribut bietet eine abgeleitete Navigationskonfiguration, die dem Schema des MenuSrcData-Attributs des TcHmiNavigationBar-Controls entspricht.

Dies ist ein schreibgeschütztes Attribut, das der Benutzer mit dem MenuSrcData-Attribut des TcHmiNavigationBar-Controls verbinden sollte.

Schema: tchmi:general#/definitions/Any

Attribut-Getter: getNavBarMenuData

Verfügbar: ab Version 12.8.0

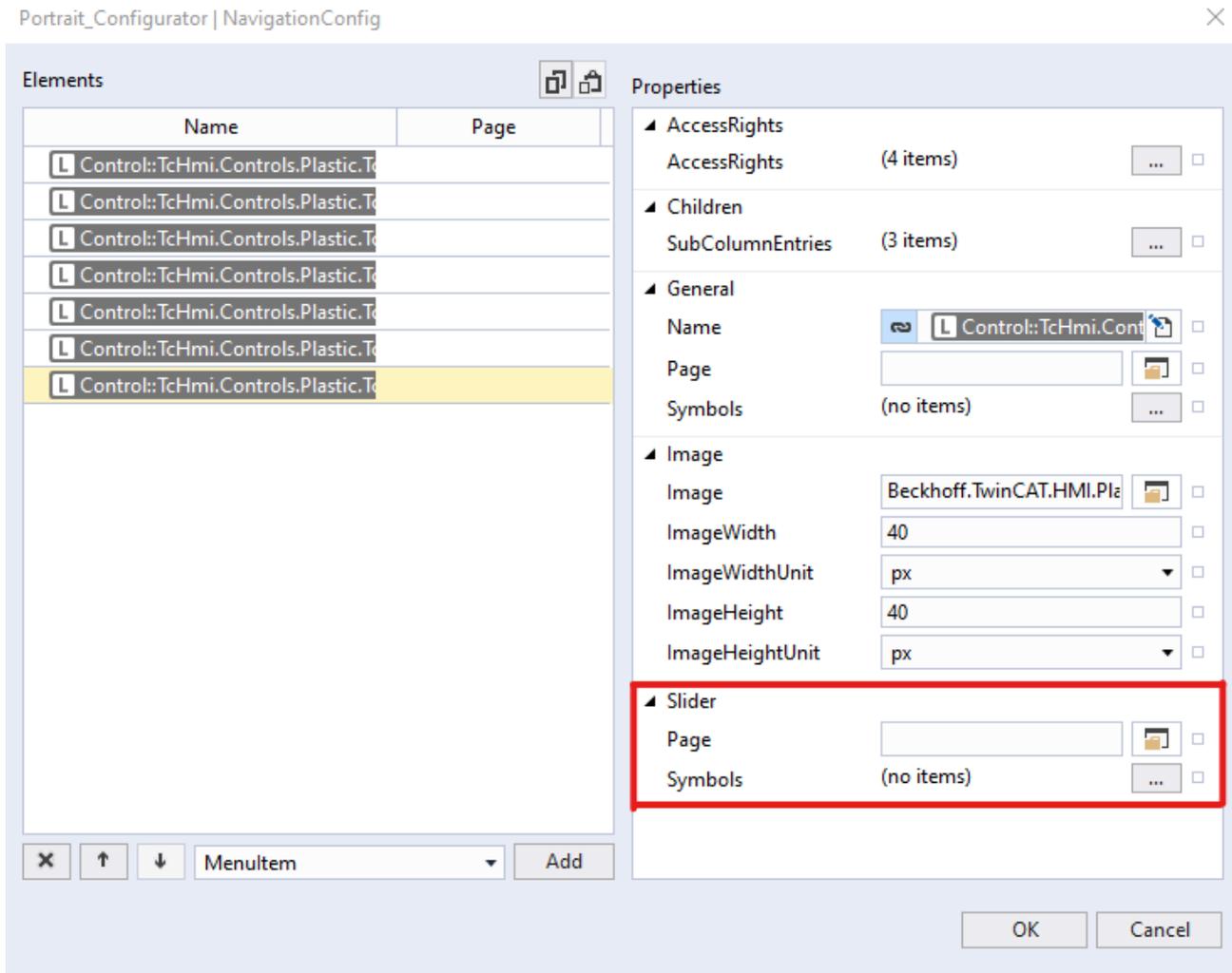
4.4.1.11 **NavigationConfig**

In Tc3_PlasticAppHmi-Projekten wird ein TcHmiNavigationBar Control aus dem Paket Beckhoff.TwinCAT.HMI.ResponsiveNavigation verwendet, um zu konfigurieren und Benutzern die Navigation zu verschiedenen Inhalten zu ermöglichen.

Für jeden in der Hauptregion angezeigten Inhalt können in der Sliderregion dieser Anwendung zugehörige Einstellungsinhalte angezeigt werden.

Mit diesem Attribut kann der Benutzer den gesamten Navigationsbaum konfigurieren und Informationen zu den Einstellungen hinzufügen, die in der Sliderregion angezeigt werden sollen, wenn der Zielinhalt in der Hauptregion der Anwendung angezeigt wird. Das Configurator-Control überwacht jede Änderung des Inhalts in der Hauptregion und lädt, wenn die Konfiguration dies erfordert, den Sliderinhalt in die Sliderregion.

Weitere Details zu Navigation Bar Control und dessen MenuSrcData finden Sie unter [diesem](#) Link.



Für jeden Inhalt, der mit diesem Attribut konfiguriert wird, sind zwei neue Eigenschaften im Vergleich zum `MenuSrcData` Attribut des Navigation Bar Controls verfügbar.

1. Sliderinhalt: Dies ist der Pfad des Inhalts, der in der Sliderregion angezeigt werden soll, wenn die Seite in der Hauptregion angezeigt wird.
2. Slider-Symbole: Wenn der im Sliderbereich angezeigte Inhalt 'Parameters_Settings.content' ist, können mit der Eigenschaft `symbols` die Parameter, die auf dem Inhalt angezeigt werden sollen, konfiguriert werden. Ein Benutzer kann ein Symbol des Typs 'fbParamHandle' mit dieser Eigenschaft verknüpfen.

Schema: tchmi:framework#/definitions/PlasticNavigation

Attribut-Getter: getNavigationConfig

Attribut-Setter: setNavigationConfig

Verfügbar: ab Version 12.8.0

4.4.1.12 PanelBackgroundColor

Legt die Hintergrundfarbe für das Input Panel fest, das für die Bearbeitung der InputBox-Controls verfügbar ist.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getPanelBackgroundColour

Attribut-Setter: setPanelBackgroundColour

Verfügbar: ab Version 12.6.0

4.4.1.13 RecipeAutoSaveDelay

Wenn das Projekt das Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper-Paket verwendet, kann es die Speicherung der Benutzeränderungen in einem temporären Rezept automatisch auslösen.

Dieses Attribut definiert die Verzögerung in Sekunden, nach der RecipeHelper vom Benutzer vorgenommene Änderungen der im Rezept gespeicherten Werte in einer temporären Rezeptdatei speichert.

Der Wert 0 stoppt die automatische Speicherung von Änderungen.

Schema: tchmi:framework#/definitions/PositiveNumber

Attribut-Getter: getRecipeAutoSaveDelay

Verfügbar: ab Version 12.12.0

4.4.1.14 RecipeBackupDuration

Wenn das Projekt das Paket Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper verwendet, kann es den Sicherungsvorgang jedes Mal auslösen, wenn Benutzeränderungen in einem temporären Rezept gespeichert werden.

Dieses Attribut legt die Verzögerung in Minuten fest, nach der vom Benutzer vorgenommene temporäre Änderungen in einer Rezeptsicherungsdatei gespeichert werden.

Wenn Sie den Wert auf 0 setzen, wird die Rezeptsicherung deaktiviert.

Weitere Informationen finden Sie hier: [Rezepte sichern und wiederherstellen \[► 207\]](#).

Schema: tchmi:framework#/definitions/PositiveNumber

Attribut-Getter: getRecipeBackupDuration

Attribut-Setter: setRecipeBackupDuration

Verfügbar: ab Version 12.14.0

4.4.1.15 RecipeHandshakeSymbol

Dieses Attribut wird benötigt, wenn das Projekt das [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper \[► 198\]](#)-Paket verwendet, das einen Handshake zwischen SPS und HMI bereitstellt, um sichere Rezeptoperationen zu ermöglichen.

Dieser Handshake arbeitet mit einer Instanz vom Typ [FB_PlcStateToHmi](#). Der Benutzer kann den gemappten Pfad dieser Instanz auf dieses Attribut setzen.

Schema: tchmi:framework#/definitions/Symbol

Attribut-Getter: getRecipeHandshakeSymbol

Attribut-Setter: setRecipeHandshakeSymbol

Verfügbar: ab Version 12.12.0

4.4.1.16 SystemUnitsChanged

Wird eine Eigenschaft der Systemeinheit einer beliebigen Einheitengruppe geändert, so ändert sich auch dieses Attribut.

Dieses Attribut kann als Ereignis für andere Controls verwendet werden, um einheitenabhängige Eigenschaften im Projekt neu zu berechnen.

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getSystemUnitsChanged

Verfügbar: ab Version 12.6.0

4.4.1.17 TemperaturSymbol

Wenn das Projekt das [Beckhoff.TwinCAT.HMI.Plastic.Temperature \[► 226\]](#)-Paket verwendet, benötigt es eine Instanz von [FB TemperatureHmi](#) in der SPS, um die Temperaturdaten zu erhalten.

Der Benutzer kann den gemappten Pfad dieses Symbols auf dieses Attribut setzen; dies ist notwendig, um die Kommunikation zwischen dem Temperaturpaket und der SPS zu initiieren.

Schema: tchmi:framework#/definitions/Symbol

Attribut Getter: getTemperatureSymbol

Verfügbar: ab Version 12.12.0

4.4.1.18 TextBackgroundColor

Legt die Hintergrundfarbe des auf das Input Panel angezeigten Descriptor-Textes fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getTextBackgroundColor

Attribut-Setter: setTextBackgroundColor

Verfügbar: ab Version 12.6.0

4.4.1.19 UnitConfig

Pfad zur Datei Unit_Config.json, die verschiedene Einheitengruppen, Einheiten und Umrechnungsfunktionen konfiguriert.

Die Standarddatei Unit_Config.json ist im Paket enthalten, und ihr Pfad wird als Standardattribut festgelegt.

So fügen Sie neue Einheitengruppen hinzu oder ändern bestehende Einheiten

1. Der Benutzer sollte eine Kopie der Standarddatei aus dem Pfad: `Beckhoff.TwinCAT.HMI.Plastic/InputGroup/Configuration/Unit_Config.json` im Projekt erstellen
2. Bearbeiten Sie die neue Datei
3. Pfad der neuen Datei als Attributwert festlegen

Schema: tchmi:framework#/definitions/Path

Attribut-Getter: getUnitConfig

Attribut-Setter: setUnitConfig

Verfügbar: ab Version 12.6.0

4.4.1.20 UnitGroups

Gibt eine Liste der in der Datei Unit_Config.json konfigurierten Einheitengruppenobjekte zurück.

Schema: tchmi:general#/definitions/Array

Attribut-Getter: getUnitGroups

Verfügbar: ab Version 12.6.0

4.4.1.21 UserLoggerSymbol

Die Controls in TF8550 Plastic HMI Framework unterstützen eine Funktion, die es ermöglicht, jede Benutzeränderung in einer CSV-Datei zu loggen.

Mit den Funktionen [LogUserChanges](#) [[▶ 171](#)] und [LogUserChangesSymbol](#) [[▶ 172](#)] können auch andere Aktionen in dieselbe Datei geloggt werden.

Diese Funktionalität erfordert eine Instanz von [FB UserLoggingHmi](#) im SPS-Projekt. Das Plastic-Framework kommuniziert mit diesem Funktionsbaustein, um jede Änderung zu loggen. Mit diesem Attribut kann der Benutzer den gemappten Pfad des Symbols auf eine Instanz von [FB UserLoggingHmi](#) setzen.

Schema: tchmi:framework#/definitions/Symbol

Attribut-Getter: getUserLoggerSymbol

Attribut-Setter: setUserLoggerSymbol

Verfügbar: ab Version 12.12.0

4.4.2 Unit_Config.json

Mit der Datei `Unit_Config.json` kann der Benutzer konfigurieren, welche Einheitengruppen, Einheiten-IDs und Einheitenumrechnungsfunktionen in jedem TcHmi-Projekt verwendet werden.

Eine Standard `Unit_Config.json` Datei ist im Paket enthalten und kann unter `<PackageInstallPath>/runtimes/native1.12-tchmi/InputGroup/Configuration/Unit_Config.json` gefunden werden.

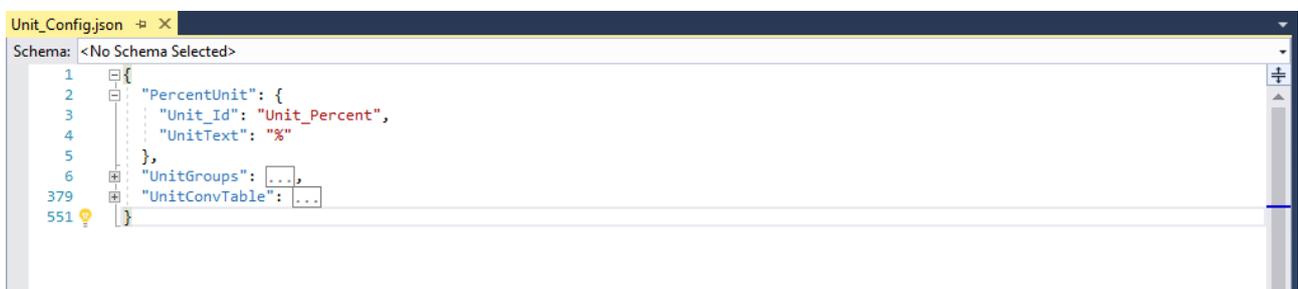
Diese Datei kann geändert oder eine Kopie davon dem Projekt hinzugefügt werden.

Um eine andere Datei als die Standarddatei `Unit_Config.json` festzulegen, verwenden Sie das Attribut [UnitConfig](#) [[▶ 44](#)] des Configurator-Controlss.

Die JSON-Struktur der Datei `Unit_Config.json` wird im Folgenden beschrieben.

Die obligatorischen Eigenschaften sind:

- [PercentUnit](#) [[▶ 45](#)]
- [UnitGroups](#) [[▶ 46](#)]
- [UnitConvTable](#) [[▶ 47](#)]



```
Unit_Config.json
Schema: <No Schema Selected>
1  {
2    "PercentUnit": {
3      "Unit_Id": "Unit_Percent",
4      "UnitText": "%"
5    },
6    "UnitGroups": [...],
379  "UnitConvTable": [...],
551 }
```

4.4.2.1 PercentUnit

Es kann nur eine `PercentUnit` in einer Konfiguration geben und sie kann von allen Einheitengruppen verwendet werden, wenn der Wert in/aus Prozent umgerechnet werden soll.

```
"PercentUnit": {
  "Unit_Id": "Unit_Percent",
  "UnitText": "%"
},
```

4.4.2.2 UnitGroups

Die Eigenschaft `UnitGroups` ist eine Liste von Einheitengruppenobjekten. Der Name jedes Objekts kann als `UnitGroup` [► 92] Attribut für das `InputBox`-Control verwendet werden.

Jedes Objekt der Einheitengruppe sollte die unten beschriebenen Eigenschaften aufweisen.

- **Obligatorische Eigenschaften:**
 - `Units`: Liste der Einheitenobjekte.
 - `SystemUnit`: Id der Einheit, die als Standard-Systemeinheit für die Einheitengruppe fungiert.
 - `DisplayUnit`: Id der Einheit, die als Standardanzeigeeinheit für die Einheitengruppe fungiert.
- **Optionale Eigenschaften:**
 - `LocalizedKey`: Der Benutzer kann der neu hinzugefügten Einheitengruppe lokalisierte Schlüssel hinzufügen, wobei der Schlüssel ein lokalisiertes Symbol im Projekt sein kann.
 - `LinkedDependentGroups`: Dies ist ein Array von Einheitsgruppen, die mit dieser Einheitsgruppe verknüpft sind und den Änderungen dieser Einheitsgruppe folgen. Jede in der Liste aufgeführte Einheitsgruppe muss diese Einheitsgruppenkennung als `LinkedSourceGroup` Eigenschaft aufweisen
 - `LinkedSourceGroup`: Dies ist der Gruppen-Identifizierer, der die Einheitengruppe kennzeichnet, die auf Änderungen überwacht wird, d.h. jede Änderung der Konfiguration von `LinkedSourceGroup` will be followed by this group.



Jede Einheitengruppe kann nur die Eigenschaft `LinkedDependentGroups` oder `LinkedSourceGroup` haben.

Im Folgenden werden beispielsweise die `UnitGroups` Länge und Geschwindigkeit dargestellt.

```
"UnitGroups": [
  {
    "Length": {
      "DisplayUnit": "Unit_Length_mm",
      "SystemUnit": "Unit_Length_mm",
      "LinkedDependentGroups": [
        "Velocity",
        "Acceleration",
        "Jerk"
      ],
    },
    "Units": [
      {
        "Unit_Id": "Unit_Length_mm",
        "UnitText": "mm"
      },
      {
        "Unit_Id": "Unit_Length_inch",
        "UnitText": "in"
      }
    ]
  },
  {
    "Velocity": {
      "DisplayUnit": "Unit_Velocity_mm_s",
      "LinkedSourceGroup": "Length",
      "SystemUnit": "Unit_Velocity_mm_s",
      "Units": [
        {
          "LinkedSourceUnit": "Unit_Length_mm",
          "Unit_Id": "Unit_Velocity_mm_s",
          "UnitText": "mm/s"
        },
        {
          "LinkedSourceUnit": "Unit_Length_inch",
          "Unit_Id": "Unit_Velocity_in_s",

```

```

    "UnitText": "in/s"
  }
]
},
}]

```

Jedes Einheitenobjekt in einer Einheitenliste sollte die unten aufgeführten Eigenschaften aufweisen:

- **Obligatorische Eigenschaften:**
 - `Unit_Id`: Eindeutige ID der Einheit.
 - `UnitText`: Text, der auf dem Bildschirm angezeigt wird, wenn diese Einheit als Anzeigeeinheit ihrer Gruppe festgelegt ist.
- **Optionale Eigenschaften:**
 - `LinkedSourceUnit`: Diese Eigenschaft gibt die Unit-ID der `LinkedSourceGroup` an, die mit dieser Unit verknüpft ist. Für eine detailliertere Verwendung der Verknüpfung von Einheiten und Einheitsgruppen lesen Sie bitte die [Verknüpfung von Einheiten](#) [▶ 47].

4.4.2.3 UnitConvTable

Die Eigenschaft `UnitConvTable` ist eine Liste von Einheitenumrechnungsobjekten.

Jedes Objekt zur Einheitenumrechnung muss drei Eigenschaften besitzen:

- `SourceUnit_Id`: Id der Quelleinheit für die Umrechnung.
- `TargetUnit_Id`: Id der Zieleinheit für die Umrechnung.
- `ConvFunction`: Umrechnungsfunktion, die auf einen beliebigen Wert angewendet wird, um ihn von der Ausgangs- in die Zieleinheit umzurechnen. Sie muss als Funktion im String-Format deklariert werden, wobei 'x' die Eingabe für die Funktion ist, und sie muss einen Wert zurückgeben.

```

"UnitConvTable": [
  {
    "SourceUnit_Id": "Unit_Length_mm",
    "TargetUnit_Id": "Unit_Length_inch",
    "ConvFunction": "function (x) {return x*0.03937;}"
  },
]

```

4.4.2.4 Verknüpfung von Einheitsgruppen

Wenn wir uns die Beispiel-Einheitsgruppen und -Einheiten ansehen, die im Kapitel [Einheitsgruppen](#) [▶ 46] konfiguriert sind, ist die Länge eine Basiseinheit und die Geschwindigkeit wird davon abgeleitet.

Idealerweise sollte das Verhalten so sein, dass, wenn der Benutzer die Anzeigeeinheit der Gruppe Länge von "mm" auf "inch" ändert, die verknüpfte Gruppe Geschwindigkeit ebenfalls ihre Anzeigeeinheit von "mm/s" auf "inch/s" ändern sollte.

Um dies zu erreichen, wird in der Konfiguration `Velocity` als eine der `LinkedDependentGroups` von `Length` markiert. Ebenso wird die Gruppe `Length` als `LinkedSourceGroup` in der Gruppe `Velocity` festgelegt.

Wenn der Benutzer bei dieser Konfiguration Änderungen in der Gruppe `Length` vornimmt, werden auch Änderungen in der Gruppe `Velocity` vorgenommen.

Um nun zu bestimmen, welche Einheit in der Gruppe `Velocity` als `DisplayUnit`-Einheit festgelegt werden soll, wird die Eigenschaft `LinkedSourceUnit` verwendet. In diesem Beispiel, wenn "inch" als Anzeigeeinheit für die Gruppe `Length` eingestellt ist, wird auch "inch/s" als Anzeigeeinheit für die Gruppe `Velocity` eingestellt. `Unit_Length_inch` aus der Gruppe `Length` wird als `LinkedSourceUnit` Eigenschaft für `Unit_Velocity_in_s` Einheit in der `Velocity` Gruppe gesetzt.

Alle Änderungen, die in einer `LinkedDependentGroups` wie z.B. `Velocity` vorgenommen werden, lösen keine Änderungen in ihrer `LinkedSourceGroup`, also der `Length` Gruppe, aus. Die Verlinkung ist nur unidirektional.

Die Funktion der Verknüpfung von Einheitsgruppen ist ab Version 12.8.0 verfügbar

4.5 CurveEditor

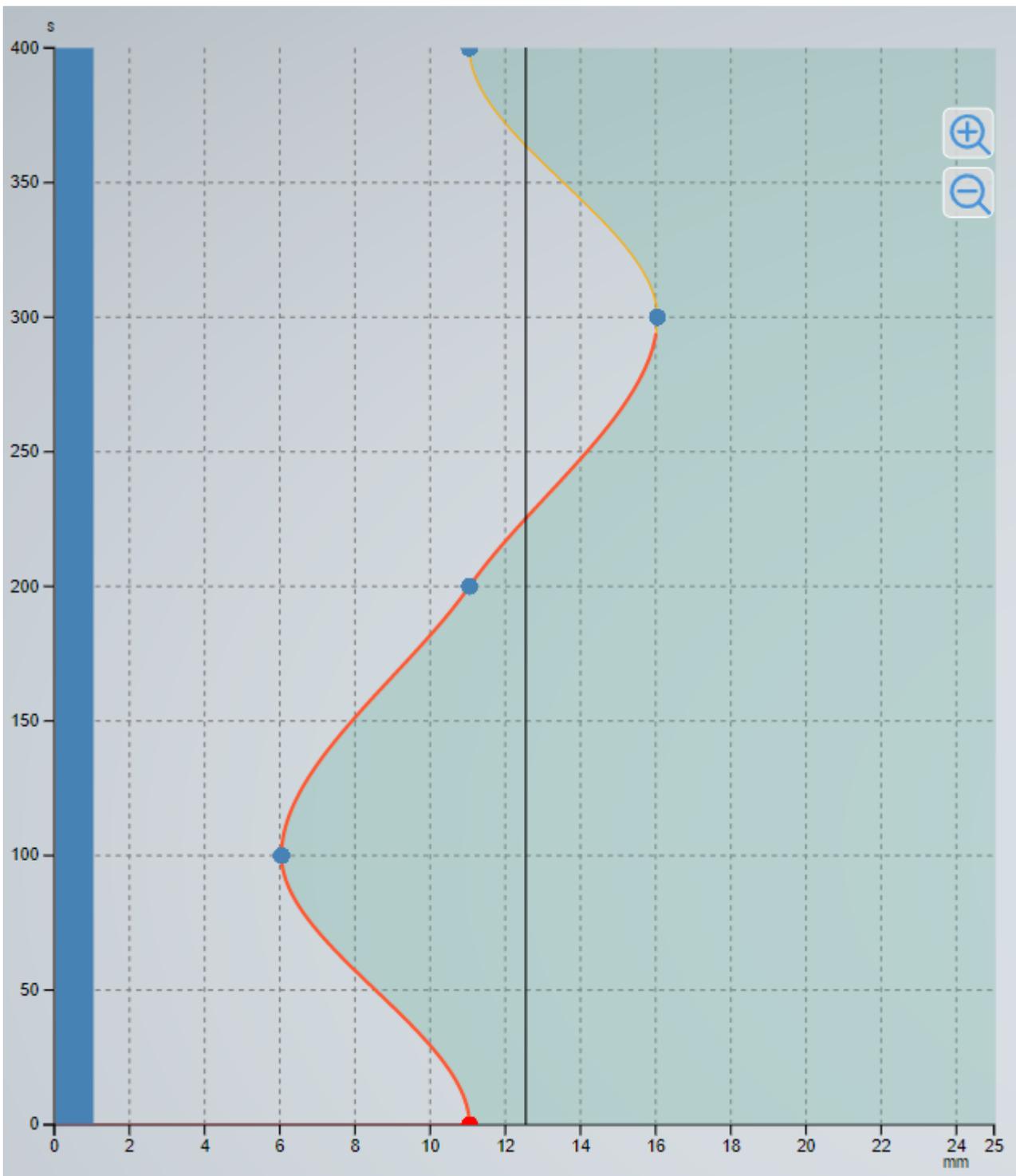
NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

Das CurveEditor-Control ermöglicht es dem Benutzer, eine Liste von Punkten auf seinem HMI-Bildschirm hinzuzufügen und zu bearbeiten. Auf diese Weise ist das Control in der Lage, eine interpolierte Kurve zu erzeugen, die durch jeden dieser Festpunkte verläuft.

Die Benutzer können eine Reihe von Berührungsgesten und -attributen verwenden, um die spezifischen Parameter des Interpolationsprozesses zu definieren, was ein hohes Maß an Anpassung und Kontrolle ermöglicht.

Das CurveEditor-Control eignet sich besonders gut für Anwendungen wie die Wanddickensteuerung der Vorformlinge in Blasformmaschinen, bei denen eine präzise und konsistente Kontrolle der Dicke des zu formenden Materials für die Qualität und Leistung des Endprodukts entscheidend ist.



4.5.1 Attribute

Kategorie: Curve Editor

Name	Beschreibung
EditorProfiles [▶ 57]	Liste der Konfigurationsprofile.
ProfilGeschichteGröße [▶ 62]	Legt die Anzahl der Profiländerungen fest, die im EditorProfiles-Array gespeichert werden.
ActiveProfileIndex [▶ 52]	Index des Konfigurationsprofils aus der Liste der Editor-Profile [▶ 57], das auf dem Control geladen ist.

Name	Beschreibung
ActualCurve [▶ 52]	Liste der Festpunkte, die die aktuellen Kurvenwerte des Ziels darstellen.
ActualCurveIndex [▶ 53]	Index des Arrays ActualCurve [▶ 52], bis zu dem das Control die Kurve zeichnet.
CurrentValue [▶ 55]	Der aktuelle Wert wird auf der Skala der Wertachse angezeigt.

Kategorie: Curve Data

Name	Beschreibung
BaseComponent [▶ 54]	Die Basiskomponente wird zu allen Sollwerten hinzugefügt.
CursorBase [▶ 56]	Basisposition des Cursors.
CursorValue [▶ 56]	Wertposition des Cursors.
Interpolation Modes [▶ 57]	Interpolationsmodus für Kurvenberechnungen.
TransitionModes [▶ 66]	Transition-Mode zwischen letztem und erstem Punkt der Kurve.
CalculateSlope [▶ 54]	Wenn <code>TRUE</code> , wird bei der Interpolation zwischen zwei Punkten die Steigung unter Berücksichtigung eines Punktes vor und eines Punktes nach diesen Punkten berechnet, was eine kontinuierliche, glatte Kurve ermöglicht.
AvoidNegativeVal [▶ 53]	Aktiviert/deaktiviert negative Werte in generierten Sollwerten.
DispCurrentProfile [▶ 56]	Wenn <code>TRUE</code> , wird die Kurve auf dem aktuellen Kurvenattribut angezeigt.
ShowCentreLine [▶ 65]	Eine Linie in der Mitte der Wertachse ein-/ausblenden.
StepSize [▶ 66]	Schrittweite für inkrementelle Änderungen bei anderen Attributen.
ActivationRequired [▶ 52]	Wird auf <code>TRUE</code> gesetzt, wenn der Benutzer neue Änderungen vornimmt, die nicht aktiviert sind.
ProfileComponent [▶ 62]	Höchster Wert in der aktuellen Festpunkt-Liste.
FixedPoints [▶ 57]	Legt die zu interpolierenden Festpunkte fest.
SetPoints [▶ 63]	Ergebnis der Interpolation der Festpunkte als Ausgangskurve.

Kategorie: Axes Data

Name	Beschreibung
BaseMax [▶ 54]	Basisachse maximal.
ValueMin [▶ 68]	Wertachse minimal.
ValueMax [▶ 68]	Wertachse maximal.
BaseAxisPosition [▶ 53]	Die Basisachse kann horizontal oder vertikal positioniert werden.
ValueAxisUnitGroup [▶ 68]	Legt die Einheitengruppe der Wertachse fest.
BaseAxisUnit [▶ 54]	Einheitentext, der auf der Basisachse angezeigt werden soll.
DisplayValueInPercent [▶ 56]	Wenn <code>TRUE</code> , werden alle Werte des Diagramms in Prozent angezeigt.

Kategorie: Marker

Name	Beschreibung
ActiveMarkers [▶ 52]	Wenn <code>TRUE</code> , werden die Marker aus der Markerliste aktiviert.
MarkerList [▶ 59]	Zeigt die Liste der ausgewählten Markerpunkte an.
MarkerWidth [▶ 59]	Legt die Breite der Marker fest, die sich auf die Basisachse bezieht.
MarkerLength [▶ 58]	Legt die Breite der Marker fest, die sich auf die Wertachse bezieht.

Kategorie: Shift Mode

Name	Beschreibung
ShiftMode [▶ 63]	Setzt das Control in den Shift-Mode, in dem ein oder mehrere Festpunkte in verschiedene Richtungen verschoben werden können.
AllowShiftVertically [▶ 53]	Wenn <code>TRUE</code> , können Punkte im Shift-Mode entlang der vertikalen Achse verschoben werden.
AllowShiftHorizontally [▶ 53]	Wenn <code>TRUE</code> , können Punkte im Shift-Mode entlang der horizontalen Achse verschoben werden.
ShiftSelectionMode [▶ 65]	Auswahlmodus zur Auswahl von Festpunkten für die Verschiebung.

Kategorie: Layout

Name	Beschreibung
ViewBoxWidth [▶ 68]	Legt die horizontale Auflösung des Diagramms in Pixeln fest.
ViewBoxHeight [▶ 68]	Legt die vertikale Auflösung des Diagramms in Pixeln fest.
Fixed Point Radius [▶ 57]	Legt den Radius der auf dem Control angezeigten Festpunkte in Pixeln fest.

Kategorie: Passive Curve

Name	Beschreibung
PassiveCurves [▶ 60]	Liste der Passive-Curve-Objekte.
ShowPassiveCurve [▶ 66]	Der Benutzer kann die passive Kurve auf dem Control mit dem Attribut <code>ShowPassiveCurve</code> anzeigen.
PassiveCurveIndex [▶ 61]	Index des PassiveCurves-Objekts, das vom Control angezeigt werden soll.

Kategorie: Recipe Management

Name	Beschreibung
RecipeManagement [▶ 62]	Fügt das <code>Symbol EditorProfiles</code> zum Rezepttyp hinzu, wenn das <code>RecipeHelper</code> -Paket verwendet wird und der Wert auf <code>"IncludeToRecipe"</code> gesetzt ist.
RecipeTypeName [▶ 63]	Mit diesem Attribut kann der Benutzer den Namen des Rezepttyps festlegen, zu dem das Control die ihm zugeordneten Symbole hinzufügt und alle Änderungen einlernt.

Kategorie: User Changes Logging

Name	Beschreibung
LogChangesByUser [▶ 58]	Logging der vom Benutzer auf dem Client vorgenommenen Änderungen in einer Logdatei.

Name	Beschreibung
ChangeLogDescriptor [► 55]	Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

4.5.1.1 ActivationRequired

Dieses schreibgeschützte Attribut wird auf `TRUE` gesetzt, wenn ein Benutzer neue Änderungen vornimmt, die nicht aktiviert sind.

Dieser Wert kann von anderen Controls, wie z. B. einem `TcHmiButton`, verwendet werden, um sein Aussehen zu ändern und den Benutzer darüber zu informieren, dass neue Änderungen aktiviert werden müssen.

Schema: `tchmi:general#/definitions/Boolean`

Attribut-Getter: `getActivationRequired`

Verfügbar: seit Version 12.6.0

4.5.1.2 ActiveMarkers

Wenn `TRUE`, werden Marker aus der [Marker List](#) [► 59] angezeigt und die generierten Sollwerte enthalten Marker-Bumps gemäß dem mit den Attributen [Marker Length](#) [► 58] und [Marker Width](#) [► 59] eingestellten Markerprofil.

Schema: `tchmi:general#/definitions/Boolean`

Attribut-Getter: `getActiveMarkers`

Attribut-Setter: `setActiveMarkers`

Verfügbar: seit Version 12.6.0

4.5.1.3 ActiveProfileIndex

Index des Konfigurationsprofils aus der Liste der Editorprofile, das derzeit auf dem Control angezeigt wird.

Durch Ändern dieses Indexes kann der Benutzer verschiedene Profile aus der Liste der Editorprofile auswählen, die auf dem Control angezeigt werden sollen. Um das angezeigte Profil zu aktivieren, muss die Funktion [ActivateConfig](#) [► 72] auf dem Control aufgerufen werden.

Bei der Aktivierung wird das ausgewählte Profil oben auf die Liste gesetzt, und das Indexattribut des aktiven Profils gibt 0 zurück.

Schema: `tchmi:general#/definitions/Number`

Attribut-Getter: `getActiveProfileIndex`

Attribut-Setter: `setActiveProfileIndex`

Verfügbar: seit Version 12.6.0

4.5.1.4 ActualCurve

Liste der Festpunkte, die die aktuellen Kurvenwerte des Ziels darstellen.

Der Benutzer kann die berechnete Kurve der Sollwerte mit den Istwerten des Ziels vergleichen, indem er die Istkurve anzeigt.

Es ist ratsam, ein Binding-Intervall von 50 Millisekunden zu verwenden, um die Übergangszeit der Animation für die reibungslose Anzeige einer sich ändernden Kurve auf dem Control anzupassen.

Schema: `tchmi:framework#/definitions/FixedPointList`

Attribut-Getter: getActCurve

Attribut-Setter: setActCurve

Verfügbar: seit Version 12.6.0

4.5.1.5 ActualCurveIndex

Letzter Index der Werte in [ActualCurve](#) [► 52], die auf dem Control angezeigt werden sollen.

Um nur einen Teil der Istkurve auf dem Control anzuzeigen, setzen Sie den Index des letzten gültigen Punktes auf dieses Attribut.

Um die gesamte Kurve anzuzeigen, muss die Gesamtlänge des Kurven-Arrays eingestellt werden.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getActCurveIndex

Attribut-Setter: setActCurveIndex

Verfügbar: seit Version 12.6.0

4.5.1.6 AllowShiftHorizontally

Wenn `TRUE`, können Festpunkte im Shift-Mode entlang der horizontalen Achse verschoben werden.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getAllowShiftInHorizontalDirection

Attribut-Setter: setAllowShiftInHorizontalDirection

Verfügbar: seit Version 12.6.0

4.5.1.7 AllowShiftVertically

Wenn `TRUE`, können Festpunkte im Shift-Mode entlang der vertikalen Achse verschoben werden.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getAllowShiftInVerticalDirection

Attribut-Setter: setAllowShiftInVerticalDirection

Verfügbar: seit Version 12.6.0

4.5.1.8 AvoidNegativeVal

Aktiviert/deaktiviert negative Werte in generierten Sollwerten. Wenn `TRUE`, werden die generierten Sollwerte auf ein Minimum von 0 gesetzt, und es gibt keine negative Ausgabe.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getAvoidNegativeValues

Attribut-Setter: setAvoidNegativeValues

Verfügbar: seit Version 12.6.0

4.5.1.9 BaseAxisPosition

Die Basisachse kann horizontal oder vertikal positioniert werden.

Schema: tchmi:framework#/definitions/BaseAxisPosition

Attribut-Getter: getBaseAxisPosition

Attribut-Setter: setBaseAxisPosition

Verfügbar: seit Version 12.6.0

4.5.1.10 BaseAxisUnit

Einheitentext, der auf der Basisachse angezeigt werden soll.

Schema: tchmi:general#/definitions/String

Attribut-Getter: getBaseAxisUnit

Attribut-Setter: setBaseAxisUnit

Verfügbar: seit Version 12.6.0

4.5.1.11 BaseComponent

Die Basiskomponente wird zu allen Sollwerten hinzugefügt. Sie kann geändert werden, indem dieses Attribut auf den gewünschten Wert gesetzt wird.

Es ist auch möglich, die Basiskomponente zu erhöhen oder zu verringern, indem man die Funktionen [IncreaseBaseComponent \[►_70\]](#) bzw. [DecreaseBaseComponent \[►_70\]](#) ausführt.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getBaseComponent

Attribut-Setter: setBaseComponent

Verfügbar: seit Version 12.6.0

4.5.1.12 BaseMax

Basisachse maximal.

Schema: tchmi:general#/definitions/Number

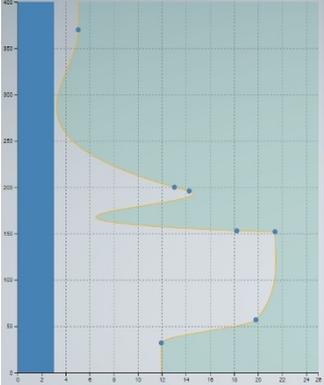
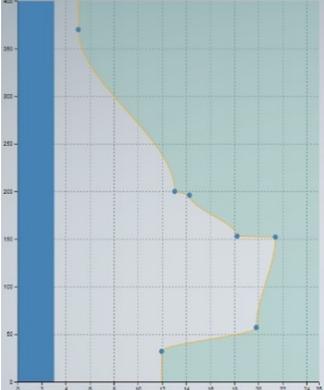
Attribut-Getter: getBaseMax

Attribut-Setter: setBaseMax

Verfügbar: seit Version 12.6.0

4.5.1.13 CalculateSlope

Wenn die **PolynomialDegreeof3**-Interpolation verwendet wird, kann die Kurveninterpolation mit dem Attribut Calculate Slope auf zwei verschiedene Arten berechnet werden:

Calculate Slope Eigenschaft	Beschreibung	Beispiel
TRUE	Die Steigung wird einer gemeinten linearen Verbindung zwischen der Basis vor und nach dem betrachteten Punkt berechnet und eine Gewichtung für die Steigung am betrachteten Punkt in Abhängigkeit vom Abstand zwischen dem vorherigen und dem nächsten Punkt vorgenommen.	
FALSE	Die Steigung an den Festpunkten ist 0. Durch die Interpolation wird ein horizontaler Verlauf durch die Festpunkte erreicht.	

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getCalculateSlopes

Attribut-Setter: setCalculateSlopes

Verfügbar: seit Version 12.6.0

4.5.1.14 ChangeLogDescriptor

Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

Schema: "tchmi:general#/definitions/STRING"

Attribut-Getter: getChangeLogDescriptor

Attribut-Setter: setChangeLogDescriptor

Verfügbar: ab Version 12.14.0

4.5.1.15 CurrentValue

Der aktuelle Wert der Kurve, der auf der Skala der Wertachse angezeigt werden soll.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: setCurrentValue

Attribut-Setter: getCurrentValue

Verfügbar: ab Version 12.12.0

4.5.1.16 CursorBase

Dies ist die Basiseigenschaft des mit dem Cursor ausgewählten aktuellen Sollwerts. Es kann in beide Richtungen verwendet werden.

1. Wenn der Benutzer die Cursorposition durch Interaktion mit der Kurve oder durch Verwendung der Funktionen [IncreaseCursorBase \[► 71\]](#) oder [DecreaseCursorBase \[► 71\]](#) ändert, kann die aktualisierte Cursorbasis aus diesem Attribut gelesen werden.
2. Wenn der Benutzer die gewünschte Cursor-Basis in dieses Attribut schreibt, bewegt sich der Cursor, um die gewünschte Position zu markieren.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getCursorBase

Attribut-Setter: setCursorBase

Verfügbar: seit Version 12.6.0

4.5.1.17 CursorValue

Dies ist die Value-Eigenschaft des mit dem Cursor ausgewählten Punktes.

1. Ändert der Benutzer die Cursorposition oder führt die Funktionen [IncreaseValueAtCursor \[► 72\]](#) oder [DecreaseValueAtCursor \[► 72\]](#) aus, kann die aktualisierte Value-Eigenschaft des aktuell mit dem Cursor ausgewählten Punktes aus diesem Attribut gelesen werden.
2. Oder wenn der Benutzer den gewünschten Wert in dieses Attribut schreibt, wird die Kurve einen neuen Festpunkt an der gewählten Basisposition erzeugen.
3. Wenn der Cursor bereits auf einen bestehenden Festpunkt zeigt, führt dies zu einer Änderung der Value-Eigenschaft des Festpunkts.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getCursorValue

Attribut-Setter: setCursorValue

Verfügbar: seit Version 12.6.0

4.5.1.18 DispCurrentProfile

Wenn `TRUE`, wird die Kurve auf dem [Actual Curve \[► 52\]](#)-Attribut angezeigt.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getDisplayCurrentProfile

Attribut-Setter: setDisplayCurrentProfile

Verfügbar: seit Version 12.6.0

4.5.1.19 DisplayValueInPercent

Wenn `TRUE`, werden alle Werte des Diagramms in Prozent angezeigt.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getDispValueInPercent

Attribut-Setter: setDispValueInPercent

Verfügbar: seit Version 12.6.0

4.5.1.20 EditorProfiles

Liste der Konfigurationsprofile.

Der Benutzer kann die Liste der Profile mit dem Attribut [ActiveProfileIndex \[▶ 52\]](#) durchgehen und jedes Profil aus der Liste aktivieren.

Bei der Aktivierung wird das neueste Profil an den Anfang der Profilliste gesetzt. Die maximale Anzahl der Profile, die in der Liste gespeichert werden können, wird durch das Attribut [ProfilGeschichteGröße \[▶ 62\]](#) festgelegt. Wenn Sie die Bearbeitung mit der Funktion [ResetConfigdata \[▶ 73\]](#) abbrechen, zeigt das Control das zuletzt aktivierte Profil an.

Schema: tchmi:framework#/definitions/CurveEditorConfigList

Attribut-Getter: getEditorProfiles

Attribut-Setter: setEditorProfiles

Verfügbar: ab Version 12.6.0

4.5.1.21 FixedPointRadius

Legt den Radius der auf dem Control angezeigten Festpunkte in Pixeln fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getFixedPointRadius

Attribut-Setter: setFixedPointRadius

Verfügbar: seit Version 12.6.0

4.5.1.22 FixedPoints

Legt die zu interpolierenden Festpunkte fest.

Schema: tchmi:framework#/definitions/FixedPointList

Attribut-Getter: getFixedPoints

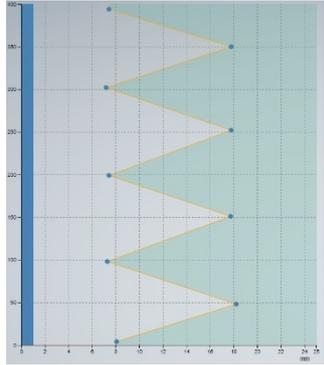
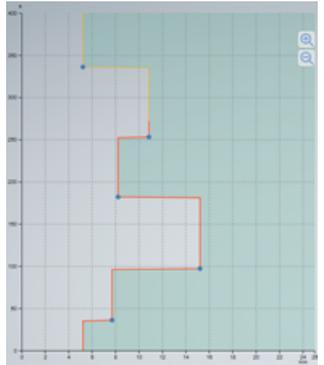
Attribut-Setter: setFixedPoints

Verfügbar: seit Version 12.6.0

4.5.1.23 InterpolationModes

Mit diesem Attribut kann der Interpolationsmodus für die Sollwertkurve eingestellt werden.

Interpolationsmodus	Beschreibung	Beispiel
PolynomialOfDegree3	Die verwendete Interpolation ist PolynomialOfDegree3; die Interpolation kann mit dem Attribut Calculate Slope [▶ 54] beeinflusst werden.	

Interpolationsmodus	Beschreibung	Beispiel
Linear	Für die Kurve wird eine Linearinterpolation zwischen zwei Festpunkten berechnet.	
StepAfter	Die Kurve ändert ihren Wert nur an Festpunkten und behält den letzten Festpunktwert zwischen zwei Punkten bei.	

Schema: tchmi:framework#/definitions/InterpolationMode

Attribut-Getter: getInterpolationMode

Attribut-Setter: setInterpolationMode

Verfügbar: seit Version 12.6.0

4.5.1.24 LogChangesByUser

Wenn dieses Attribut auf `TRUE` gesetzt ist, loggt das Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket jede Benutzeränderung auf dem HMI-Bildschirm in einer Logdatei.

Um dieses Attribut zu verwenden, muss [UserLoggerSymbol](#) [► 45] mit einem Binding an `FB_UserLoggingHmi` konfiguriert werden.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getLogUserChanges

Attribut-Setter: setLogUserChanges

Verfügbar: ab Version 12.14.0

4.5.1.25 MarkerLength

Legt die Länge jedes Marker-Bumps bezogen auf die Wertachse fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getMarkerLength

Attribut-Setter: setMarkerLength

Verfügbar: seit Version 12.6.0

4.5.1.26 MarkerList

Zeigt die Liste der ausgewählten Markerpunkte an.

Hinzufügen oder Entfernen eines Markers aus der Markerliste:

1. Platzieren Sie den Cursor an der Stelle, an der der Markerpunkt gesetzt/zurückgesetzt werden soll, indem Sie ihn einfach berühren oder das Attribut `Cursor Base` setzen.
2. Klicken Sie auf `AddNewMarker` oder `RemoveSelectedMarker`.

⇒ Ein neuer Markerpunkt wird gesetzt oder ein aktiver Markerpunkt wird gelöscht.

Schema: tchmi:framework#/definitions/FixedPointList

Attribut-Getter: getMarkerList

Attribut-Setter: setMarkerList

Verfügbar: seit Version 12.6.0

4.5.1.27 MarkerWidth

Legt die Breite des Marker-Bumps fest, die sich auf die Basisachse bezieht.

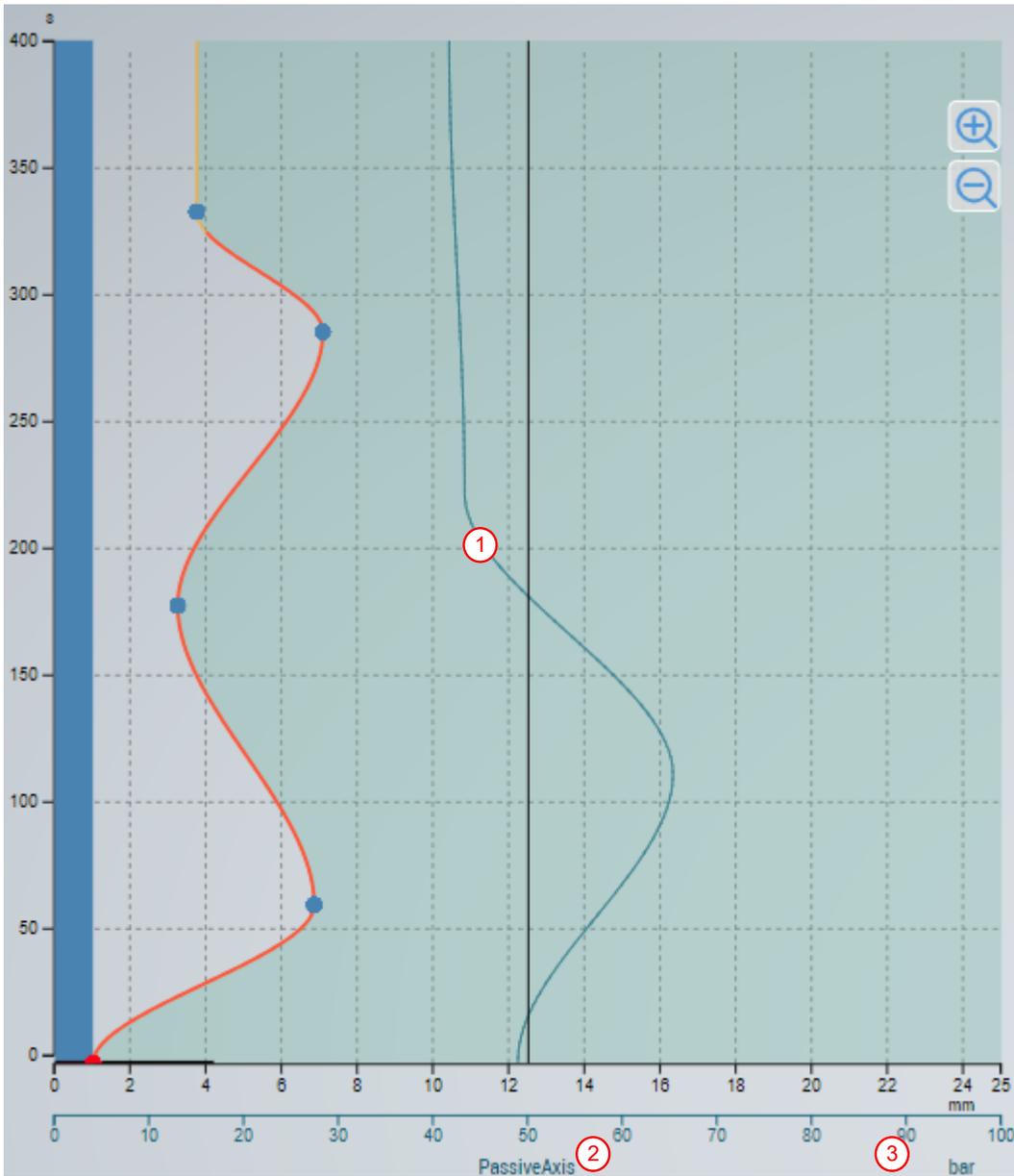
Schema: tchmi:general#/definitions/Number

Attribut-Getter: getMarkerWidth

Attribut-Setter: setMarkerWidth

Verfügbar: seit Version 12.6.0

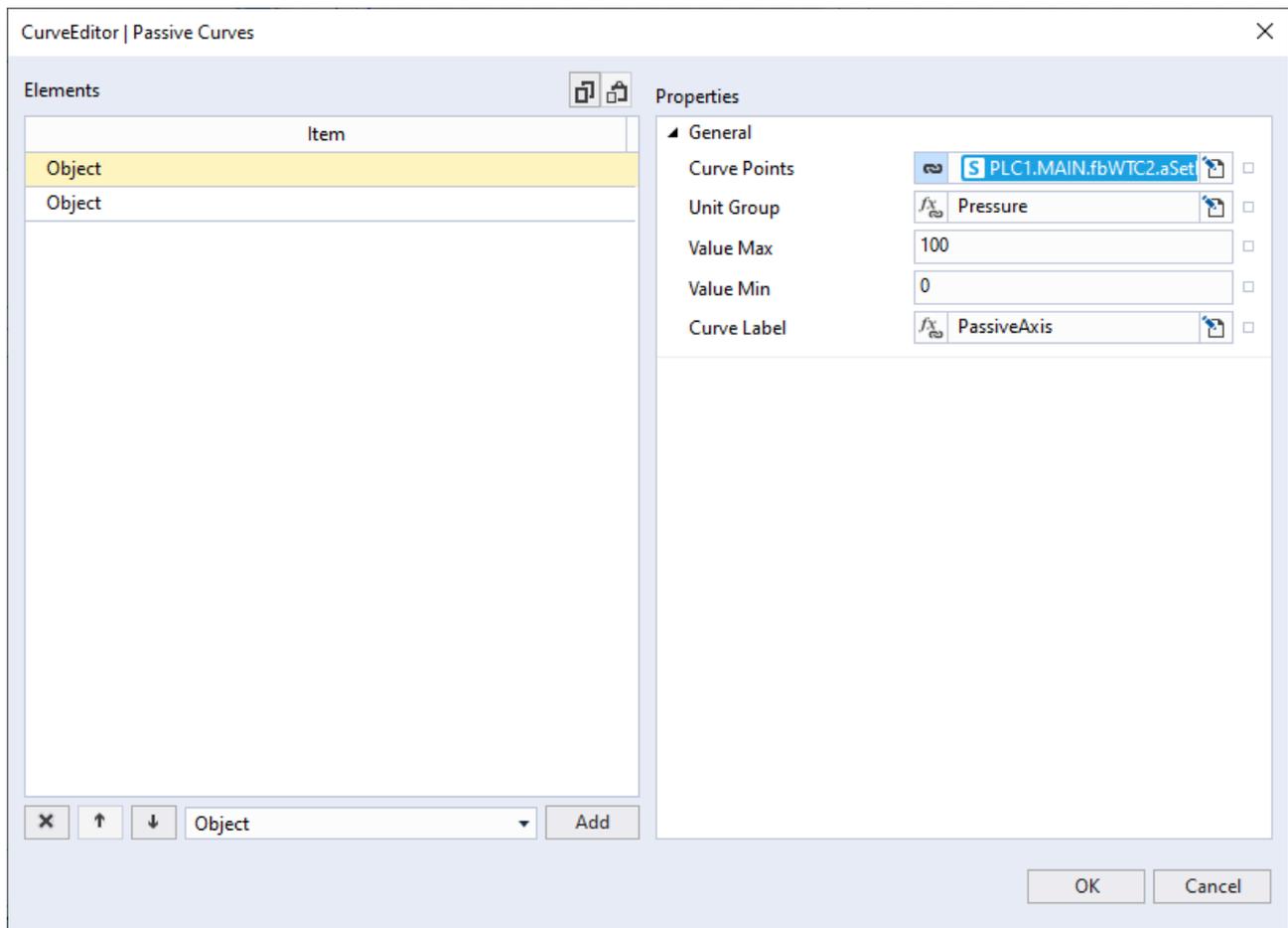
4.5.1.28 PassiveCurves



1. Passive Kurve, die auf dem Hintergrund des Kurveneditors angezeigt wird.
2. Achsenbeschriftung der passiven Kurve.
3. Achse der passiven Kurve.

Mit den Eigenschaften `PassiveCurves`, `ShowPassiveCurves` [► 66] und `PassiveCurveIndex` [► 61] kann der Benutzer eine zusätzliche Kurve auf dem Hintergrund des Kurveneditors anzeigen. Diese Kurve dient nur zur Anzeige und der Benutzer kann die Werte dieser Kurve nicht über das Control bearbeiten.

Das Attribut `Passive Curves` ist eine Liste von `Passive-Curve`-Objekten.



Jedes Passive-Curve-Objekt hat die folgenden Eigenschaften:

1. Curve Points: Liste der auf der Kurve angezeigten Punkte.
2. UnitGroup: Einheitengruppe, die für die Umrechnung der Einheiten der passiven Achse verwendet wird.
3. Value Max: Höchstwert der passiven Achse.
4. Value Min: Mindestwert der passiven Achse.
5. Curve Label: Beschriftung, die auf der passiven Achse angezeigt wird.

Der Benutzer kann in diesem Attribut verschiedene Passive-Curve-Objekte vorkonfigurieren und mit Hilfe des Attributs `PassiveCurveIndex` [► 61] durch die verschiedenen Kurven iterieren.

Schema: "tchmi:framework#/definitions/PassiveCurves"

Attribut-Getter: `getPassiveCurves`

Attribut-Setter: `setPassiveCurves`

Verfügbar: ab Version 12.14.0

4.5.1.29 **PassiveCurveIndex**

Der Benutzer kann den Index des `PassiveCurves`-Objekts [► 60] auswählen, das vom Control angezeigt werden soll.

Schema: `tchmi:general#/definitions/Number`

Attribut-Getter: `getPassiveCurveIndex`

Attribut-Setter: `setPassiveCurveIndex`

Verfügbar: ab Version 12.14.0

4.5.1.30 ProfileComponent

Höchster Wert in der generierten Sollwertliste.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getProfileComponent

Attribut-Setter: setProfileComponent

Verfügbar: seit Version 12.6.0

4.5.1.31 ProfilGeschichteGröße

Legt die Anzahl der Profiländerungen fest, die im [EditorProfiles](#) [► 57]-Array gespeichert werden sollen.

Schema: tchmi:framework#/definitions/CurveEditorControl.ProfileHistorySize

Attribut-Getter: getProfileHistorySize

Attribut-Setter: setProfileHistorySize

Verfügbar: ab Version 12.12.0

4.5.1.32 RecipeManagement

Wählen Sie aus, ob das Control das Serversymbol, das an das Attribut [EditorProfiles](#) [► 57] gebunden ist, zu einem Rezepttyp hinzufügen soll.

Wenn das Projekt das Paket [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper](#) [► 198] enthält, fügt das Control das darauf abgebildete Serversymbol zum Rezepttyp hinzu, der durch das Attribut [RecipeTypeName](#) [► 63] definiert ist.

Nach jedem Benutzerwechsel lernt das Paket RecipeHelper das aktuell aktive Rezept ein.

Schema: "tchmi:framework#/definitions/TcHmi.Controls.Plastic.RecipeManagement"

Attribut-Getter: getRecipeManagement

Attribut-Setter: setRecipeManagement

Verfügbar: ab Version 12.14.0

4.5.1.33 RecipeType (veraltet)



Veraltetes Attribut

Dieses Attribut wird entfernt und kann nicht mehr verwendet werden.

Wenn dieses Attribut auf 'R' gesetzt ist und das Projekt das [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper](#) [► 198]-Paket enthält, fügt das Control das Serversymbol, das an das [EditorProfiles](#) [► 57]-Attribut gebunden ist, dem Rezepttyp ActRcpDataType hinzu.

Bei jeder Benutzeraktivierung nach einem Kurvenwechsel lernt das RecipeHelper-Paket das aktuell aktive Rezept ein.

Schema: tchmi:framework#/definitions/RecipeType

Attribut-Getter: getRecipeType

Attribut-Setter: setRecipeType

Verfügbar: ab Version 12.12.0

Veraltet: ab Version 12.14.0

Ersetzt durch: [RecipeManagement](#) [► 62]

4.5.1.34 RecipeTypeName

Mit diesem Attribut kann der Benutzer den Namen des Rezepttyps festlegen, zu dem das Control die ihm zugewiesenen Symbole hinzufügt und alle Änderungen speichert.

Damit das Control Symbole zum Rezepttyp hinzufügen kann, muss das Projekt das Paket [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper](#) [► 198] enthalten und das Attribut [RecipeManagement](#) [► 62] muss auf IncludeToRecipe gesetzt sein.



Wenn ein Symbol bereits Teil eines Rezepttyps ist, hat diese Eigenschaft keine Auswirkungen auf die Rezepte. Nur für das Hinzufügen von Symbolen, die nicht Teil eines Rezepttyps sind, wird das Attribut RecipeTypeName berücksichtigt.

Schema: "tchmi:general#/definitions/STRING"

Attribut-Getter: getRecipeTypeName

Attribut-Setter: setRecipeTypeName

Verfügbar: ab Version 12.14.0

4.5.1.35 SetPoints

Ergebnis der Interpolation der Festpunkte als Ausgangskurve.

Schema: tchmi:framework#/definitions/FixedPointList

Attribut-Getter: getSetPoints

Attribut-Setter: setSetPoints

Verfügbar: seit Version 12.6.0

4.5.1.36 ShiftMode

Setzt das Control in den Shift-Mode, in dem ein oder mehrere Festpunkte in verschiedene Richtungen verschoben werden können.

Sobald der Shift-Mode aktiviert ist, erscheint ein roter Rahmen um das Control, um anzuzeigen, dass der Shift-Mode aktiv ist.



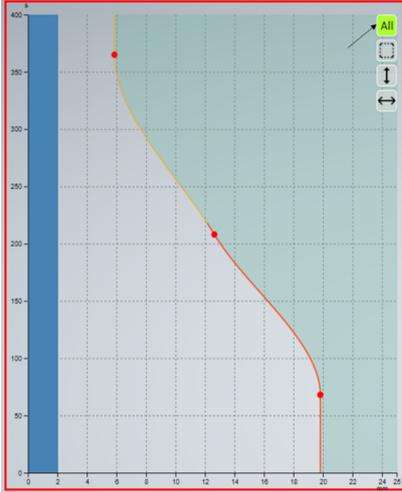
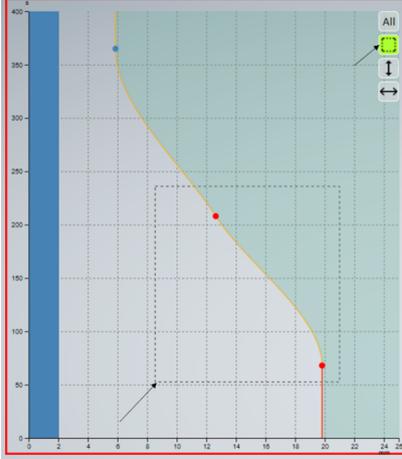
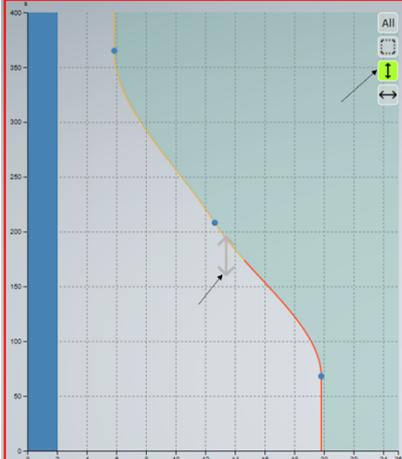
Keine Benutzerinteraktionen

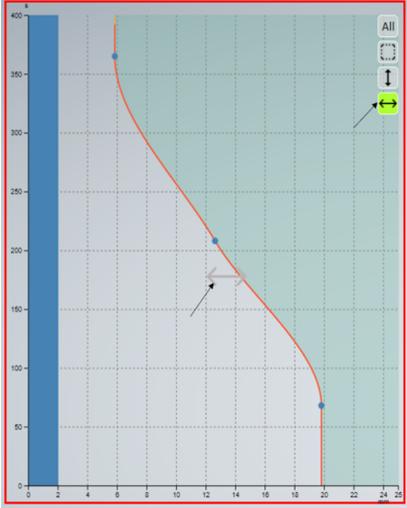
Im Shift-Mode sind die folgenden Benutzerinteraktionen deaktiviert: Zoomen mit zwei Fingern, Ziehen eines einzelnen Festpunkts, Doppeltippen zum Hinzufügen eines neuen Punkts und Ziehen von Basiskomponenten.

In diesem Modus können alle ausgewählten Festpunkte in drei Richtungen gezogen werden:

- Ziehen in Basisrichtung
- Ziehen in Richtung der Wertachse
- Ziehen Sie in beide Richtungen entlang des Diagramms

Mit den Schaltflächen in der oberen rechten Ecke kann der Benutzer im Shift-Mode Verschiebungen durchführen.

Aktion	Beschreibung	Beispiel
Alle Festpunkte auswählen	Wenn die erste Schaltfläche einmal angeklickt wird, werden alle im Diagramm verfügbaren Festpunkte ausgewählt.	
Festpunkte mit Auswahlrechteck auswählen	<p>Der Benutzer kann mit einem Finger ein Rechteck auf dem Bildschirm zeichnen.</p> <p>Alle Punkte, die von diesem Rechteck abgedeckt werden, und alle Punkte, die nicht ausgewählt sind, deren Basisindex aber zwischen dem ersten und dem letzten ausgewählten Punkt liegt, werden ausgewählt, sobald das Auswahlrechteck gezeichnet wurde.</p>	
Festpunkte in vertikaler Richtung ziehen	<p>Nach der Auswahl der gewünschten Festpunkte kann der Benutzer mit der in der nächsten Abbildung gezeigten Schaltfläche die Ziehrichtung auswählen.</p> <p>Wenn das Ziehen in vertikaler Richtung aktiviert ist, erscheint ein Ziehpunkt in der Mitte des Diagramms.</p> <p>Der Benutzer kann diesen Punkt entlang der Richtung der Basisachse ziehen und alle ausgewählten Festpunkte werden verschoben.</p>	

Aktion	Beschreibung	Beispiel
<p>Festpunkte in Richtung der horizontalen Achse ziehen</p>	<p>Nach der Auswahl der gewünschten Festpunkte kann der Benutzer mit der in der nächsten Abbildung gezeigten Schaltfläche die Ziehrichtung auswählen.</p> <p>Wenn das Ziehen in Richtung der horizontalen Achse aktiviert ist, wird in der Mitte des Diagramms ein Ziehpunkt angezeigt.</p> <p>Der Benutzer kann diesen Punkt entlang der Richtung der Basisachse ziehen, und alle ausgewählten Festpunkte werden in die Richtungen der Wertachsen verschoben.</p>	
<p>Festpunkte in alle Richtungen ziehen</p>	<p>Wenn die Richtungen der Basisachse und der Wertachse aktiviert sind, können die ausgewählten Festpunkte gleichzeitig in beide Richtungen verschoben werden.</p>	

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getShiftMode

Attribut-Setter: setShiftMode

Verfügbar: seit Version 12.6.0

4.5.1.37 ShiftSelectionMode

Auswahlmodus zur Auswahl von Festpunkten für die Verschiebung im Shift-Mode.

Schema: tchmi:framework#/definitions/ShiftSelectionMode

Attribut-Getter: getShiftSelectionMode

Attribut-Setter: setShiftSelectionMode

Verfügbar: seit Version 12.6.0

4.5.1.38 ShowCentreLine

Blendet eine Linie in der Mitte der Wertachse ein/aus.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getShowCentreLine

Attribut-Setter: setShowCentreLine

Verfügbar: seit Version 12.6.0

4.5.1.39 ShowPassiveCurve

Der Benutzer kann die passive Kurve auf dem Control anzeigen. Wenn diese Option auf true gesetzt ist, wird das Passive-Curve-Objekt aus der Liste [PassiveCurves](#) [▶ 60], das mit dem Attribut [PassiveCurveIndex](#) [▶ 61] ausgewählt wurde, auf dem Control angezeigt.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getShowPassiveCurve

Attribut-Setter: setShowPassiveCurve

Verfügbar: ab Version 12.14.0

4.5.1.40 StepSize

Schrittweite für inkrementelle Änderungen bei anderen Attributen. Dieses Zahlenattribut wirkt sich auf die Inkrement- oder Dekrementsschritte für die folgenden Funktionen aus:

- [IncreaseValueAtCursor](#) [▶ 72]
- [DecreaseValueAtCursor](#) [▶ 72]
- [IncreaseBaseComponent](#) [▶ 70]
- [DecreaseBaseComponent](#) [▶ 70]
- [IncreaseProfileComponent](#) [▶ 71]
- [DecreaseProfileComponent](#) [▶ 71]

Schema: tchmi:general#/definitions/Number

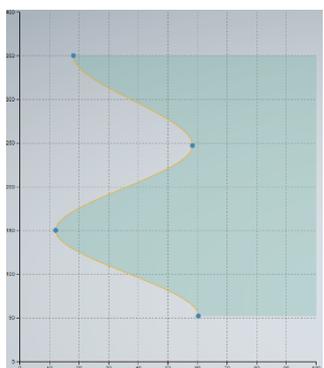
Attribut-Getter: getStepSize

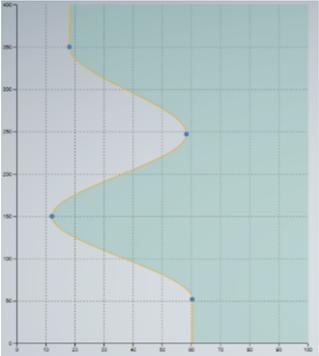
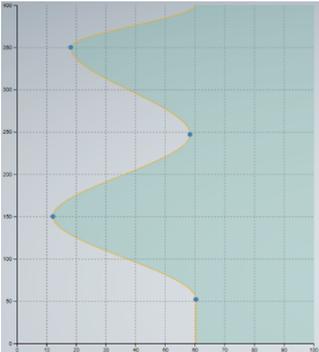
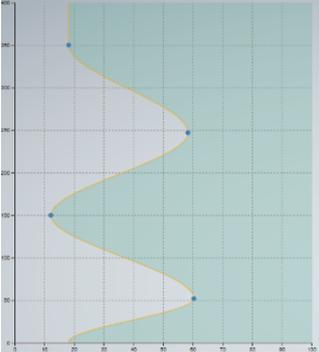
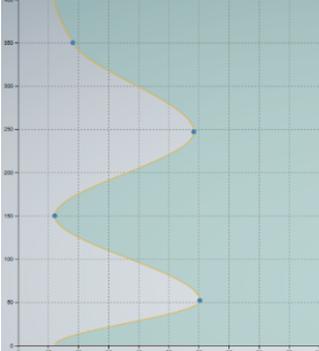
Attribut-Setter: setStepSize

Verfügbar: seit Version 12.6.0

4.5.1.41 TransitionModes

Übergang zwischen dem letzten und dem ersten Punkt der Kurve.

Transition Mode	Beschreibung	Beispiel
keiner	Es gibt keinen Übergang zwischen Start- und Endpunkt; die Kurve deckt nur die angegebenen Festpunkte ab.	

Transition Mode	Beschreibung	Beispiel
StartAndEndRepeat	Wenn auf der ersten und letzten Position der Kurve keine Festpunkte platziert sind, nehmen der erste und der letzte Sollwert den Wert des nächstgelegenen Festpunktes an.	
EndTransition	Wenn auf der ersten und letzten Position der Kurve keine Festpunkte platziert sind, nehmen der erste und der letzte Sollwert den Wert des Festpunktes mit der niedrigeren Basis an.	
StartTransition	Wenn auf der ersten und letzten Position der Kurve keine Festpunkte platziert sind, nehmen der erste und der letzte Profilpunkt den Wert des Festpunktes mit der höheren Basis an.	
StartAndEndThinnestPoint	Wenn auf der ersten und letzten Position der Kurve keine Festpunkte platziert sind, nehmen der erste und der letzte Profilpunkt den Wert der Festpunkte mit dem kleinsten Wert an.	

Schema: tchmi:framework#/definitions/TransitionMode

Attribut-Getter: getTransitionMode

Attribut-Setter: setTransitionMode

Verfügbar: seit Version 12.6.0

4.5.1.42 ValueAxisUnitGroup

Die Einheitengruppe der Wertachse. Diese Gruppe muss zuvor im [unit config file \[► 45\]](#) des [Configurator \[► 37\]](#)-Controls im Projekt konfiguriert werden. Die Achsen und Werte der Kurve werden automatisch geändert, wenn die Anzeigeeinheit der jeweiligen Gruppe geändert wird. Der Einheitentext der Anzeigeeinheitengruppe wird als Achsentext angezeigt.

Schema: tchmi:general#/definitions/String

Attribut-Getter: getValueAxisUnitGroup

Attribut-Setter: setValueAxisUnitGroup

Verfügbar: seit Version 12.6.0

4.5.1.43 ValueMax

Wertachse maximal.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getValueMax

Attribut-Setter: setValueMax

Verfügbar: seit Version 12.6.0

4.5.1.44 ValueMin

Minimum der Wertachse; es kann auf einen negativen Wert gesetzt werden, damit Kurvenwerte unter 0 angezeigt werden können.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getValueMin

Attribut-Setter: setValueMin

Verfügbar: seit Version 12.6.0

4.5.1.45 ViewBoxHeight

Legt die vertikale Auflösung des Diagramms in Pixeln fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getViewBoxHeight

Attribut-Setter: setViewBoxHeight

Verfügbar: seit Version 12.6.0

4.5.1.46 ViewBoxWidth

Legt die horizontale Auflösung des Diagramms in Pixeln fest.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getViewBoxWidth

Attribut-Setter: setViewBoxWidth

Verfügbar: seit Version 12.6.0

4.5.2 Funktionen

Kategorie: Actions

Name	Beschreibung
selectNextFixedPoint [► 69]	Mit dieser Funktion wird der nächste verfügbare Festpunkt auf der Kurve ausgewählt.
selectPreviousFixedPoint [► 70]	Mit dieser Funktion wird der zuvor vorhandene Festpunkt auf der Kurve ausgewählt.
IncreaseBaseComponent [► 70]	Diese Funktion erhöht den Wert der Basiskomponente um die Schrittweite.
DecreaseBaseComponent [► 70]	Mit dieser Funktion wird der Wert der Basiskomponente um die Schrittweite verringert.
IncreaseProfileComponent [► 71]	Diese Funktion erhöht die Profilkomponente um die Schrittweite.
DecreaseProfileComponent [► 71]	Diese Funktion verringert die Profilkomponente um die Schrittweite.
IncreaseCursorBase [► 71]	Diese Funktion erhöht die Cursorposition auf der Basisachse um 1. Wenn der Shift-Mode aktiv ist, erhöht sie die Basis aller ausgewählten Festpunkte um 1.
DecreaseCursorBase [► 71]	Diese Funktion verringert die Cursorposition auf der Basisachse um 1. Wenn der Shift-Mode aktiv ist, wird die Basis aller ausgewählten Festpunkte um 1 verringert.
IncreaseValueAtCursor [► 72]	Diese Funktion erhöht den Wert des aktuell mit dem Cursor ausgewählten Punktes um die Schrittweite im Normalmodus. Wenn der Shift-Mode aktiv ist, wird der Wert aller ausgewählten Festpunkte erhöht.
DecreaseValueAtCursor [► 72]	Mit dieser Funktion wird der Wert des aktuell mit dem Cursor ausgewählten Punktes um die Schrittweite im Normalmodus verringert. Wenn der Shift-Mode aktiv ist, wird der Wert aller ausgewählten Festpunkte verringert.
ActivateConfig [► 72]	Fügt die aktuelle Konfiguration zur Liste der Editorprofile hinzu und erzeugt einen Sollwert für die aktuellen Daten.
ResetConfigdata [► 73]	Diese Funktion löscht alle Änderungen des Benutzers und lädt die aktiven Konfigurationswerte neu.
AddNewFixPoint [► 73]	Diese Funktion fügt einen neuen Festpunkt an dem mit dem Cursor ausgewählten Punkt ein.
RemoveSelectedFixPoint [► 73]	Mit dieser Funktion wird der aktuell mit dem Cursor markierte Festpunkt entfernt; wenn kein Festpunkt markiert ist, ändert sich nichts.
AddNewMarker [► 74]	Diese Funktion fügt einen neuen Marker an der mit dem Cursor ausgewählten Basisposition ein.
RemoveSelectedMarker [► 74]	Diese Funktion entfernt den aktuell mit dem Cursor ausgewählten Marker; sie ändert nichts, wenn kein Marker ausgewählt ist.

4.5.2.1 selectNextFixedPoint

Mit dieser Funktion wird der nächste verfügbare Festpunkt auf der Kurve ausgewählt.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.2 selectPreviousFixedPoint

Mit dieser Funktion wird der zuvor vorhandene Festpunkt auf der Kurve ausgewählt.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.3 IncreaseBaseComponent

Diese Funktion erhöht den Wert der Basiskomponente um die Schrittweite [[▶ 66](#)].

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.4 DecreaseBaseComponent

Mit dieser Funktion wird der Wert der Basiskomponente um die Schrittweite verringert.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.5 IncreaseProfileComponent

Diese Funktion erhöht die Profilkomponente um die Schrittweite.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.6 DecreaseProfileComponent

Diese Funktion verringert die Profilkomponente um die Schrittweite.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.7 IncreaseCursorBase

Diese Funktion erhöht die Cursorposition auf der Basisachse um 1. Wenn der Shift-Mode aktiv ist, erhöht sie die Basis aller ausgewählten Festpunkte um 1.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.8 DecreaseCursorBase

Diese Funktion verringert die Cursorposition auf der Basisachse um 1. Wenn der Shift-Mode aktiv ist, wird die Basis aller ausgewählten Festpunkte um 1 verringert.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.9 IncreaseValueAtCursor

Diese Funktion erhöht den Wert des aktuell mit dem Cursor ausgewählten Punktes um die Schrittweite im Normalmodus. Wenn der Shift-Mode eingeschaltet ist, erhöht diese Funktion den Wert aller ausgewählten Festpunkte um einen Betrag, der der eingestellten Schrittweite entspricht.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.10 DecreaseValueAtCursor

Mit dieser Funktion wird der Wert des aktuell mit dem Cursor ausgewählten Punktes um die Schrittweite verringert.

Wenn der Shift-Mode eingeschaltet ist, verringert diese Funktion den Wert aller ausgewählten Festpunkte um einen Betrag, der der eingestellten Schrittweite entspricht.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.11 ActivateConfig

Fügt die aktuelle Konfiguration zur Liste der Editorprofile hinzu und erzeugt einen Sollwert für die aktuellen Daten.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.12 ResetConfigdata

Mit dieser Funktion werden alle Änderungen des Benutzers rückgängig gemacht, indem die zuletzt aktivierte Konfiguration erneut geladen wird.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.13 AddNewFixPoint

Diese Funktion fügt einen neuen Festpunkt an dem mit dem Cursor ausgewählten Punkt ein.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.14 RemoveSelectedFixPoint

Mit dieser Funktion wird der aktuell mit dem Cursor markierte Festpunkt entfernt; wenn kein Festpunkt markiert ist, ändert sich nichts.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.15 AddNewMarker

Diese Funktion fügt einen neuen Marker an der mit dem Cursor ausgewählten Basisposition ein.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.2.16 RemoveSelectedMarker

Mit dieser Funktion wird der mit dem Cursor ausgewählte Marker entfernt. Der Benutzer muss zunächst die Marker auswählen, indem er den Bildschirmbereich um sie herum berührt, und dann diese Funktion aufrufen.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.5.3 Events

Name	Beschreibung
onNewSetPointsGenerated [► 74]	Dieses Ereignis wird ausgelöst, wenn der Sollwertausgang geändert wird.
onUserConfigActivated [► 74]	Dieses Ereignis wird generiert, wenn eine neue Konfiguration auf dem Control aktiviert wird.
onEditingCancelled [► 75]	Dieses Ereignis wird generiert, wenn Benutzeränderungen rückgängig gemacht werden und die alte Konfiguration wieder aktiviert wird.

4.5.3.1 onNewSetPointsGenerated

Dieses Ereignis wird ausgelöst, wenn der Sollwertausgang geändert wird.

Verfügbar: seit Version 12.6.0

4.5.3.2 onUserConfigActivated

Dieses Ereignis wird generiert, wenn eine neue Konfiguration auf dem Control aktiviert wird.

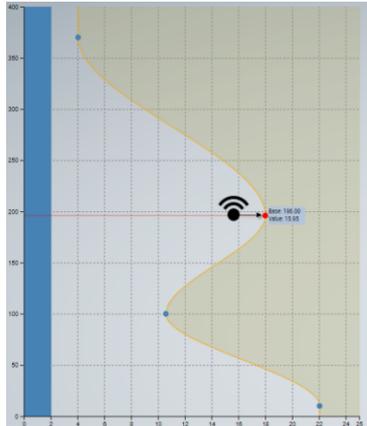
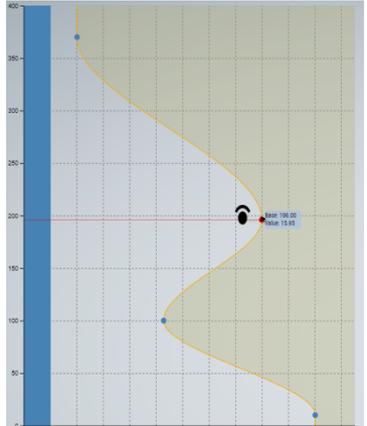
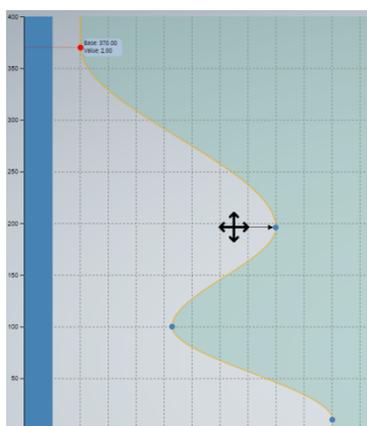
Verfügbar: seit Version 12.6.0

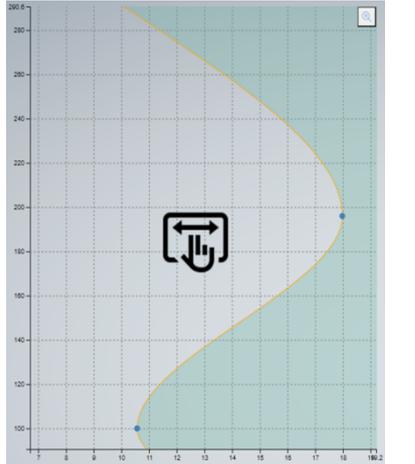
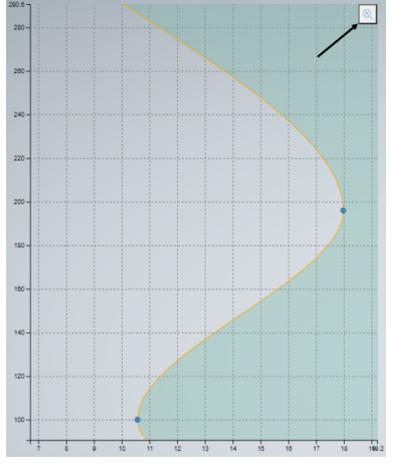
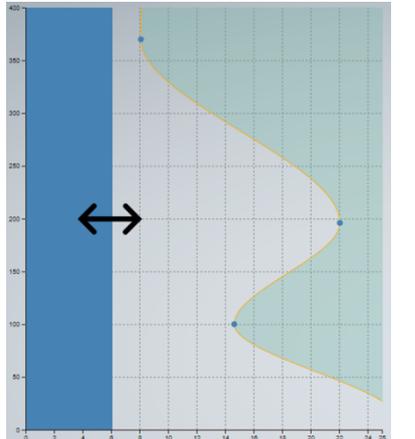
4.5.3.3 onEditingCancelled

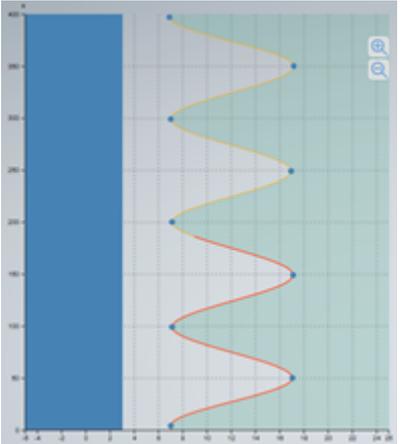
Dieses Ereignis wird generiert, wenn Benutzeränderungen rückgängig gemacht werden und die alte Konfiguration wieder aktiviert wird.

Verfügbar: seit Version 12.6.0

4.5.4 Benutzerinteraktionen

Name	Beschreibung	Ergebnis im Control
Doppeltippen	Durch Doppeltippen auf die Grafik wird ein neuer Festpunkt hinzugefügt, der in der Konfiguration durch Aktivieren oder Zurücksetzen der bearbeiteten Konfiguration übernommen werden kann.	
Einmaliges Antippen	Wenn Sie einmal auf einen Festpunkt tippen, wird der Cursorwert aktualisiert und der aktualisierte Cursorwert des Festpunkts wird in der Tooltip angezeigt.	
Festpunkt ziehen	Sobald ein einzelner Festpunkt ausgewählt ist, kann er innerhalb des Diagrammbereichs nach oben oder unten gezogen werden.	

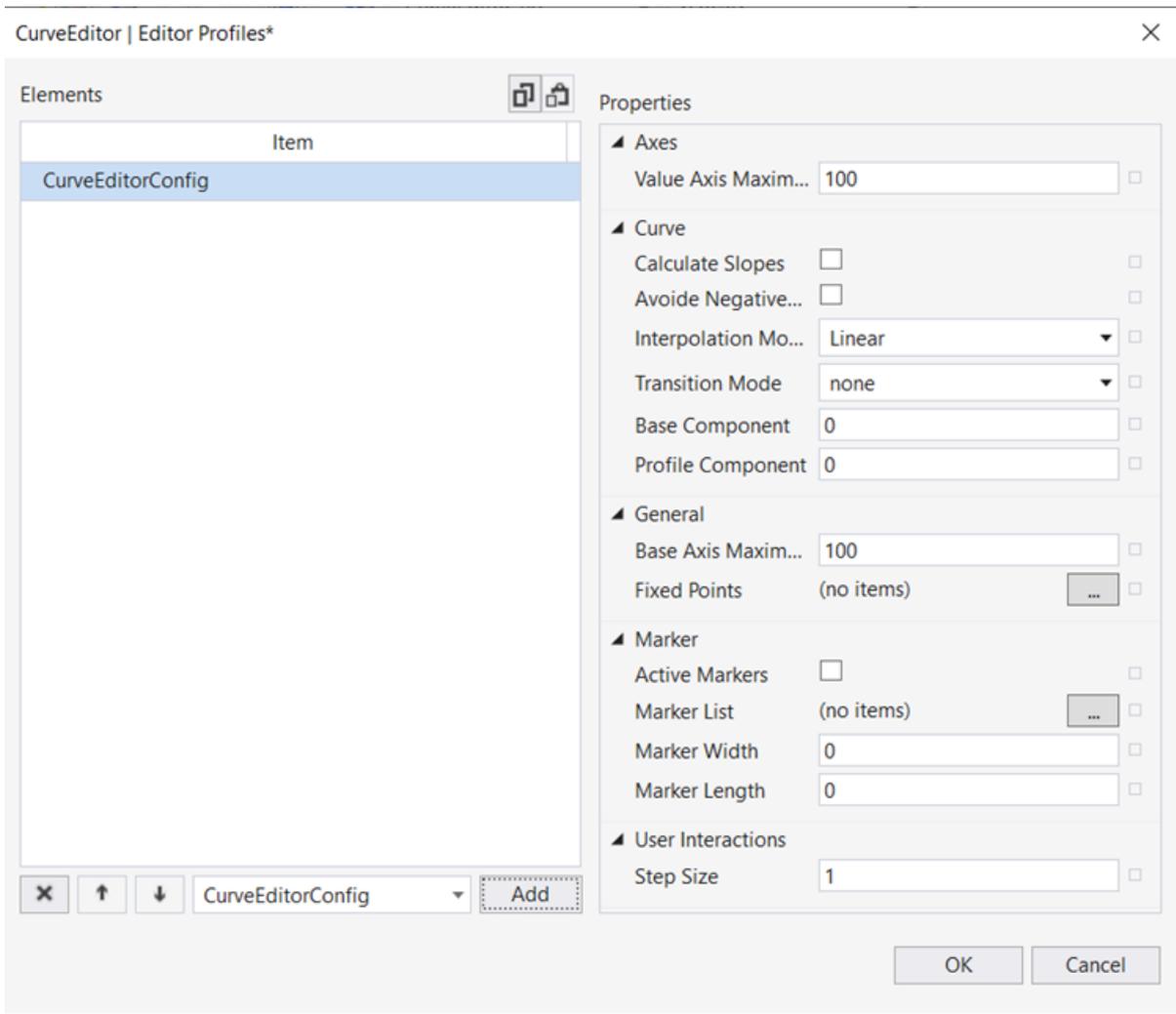
Name	Beschreibung	Ergebnis im Control
2-Finger-Zoom in der Grafik	<p>Sobald das Diagramm konfiguriert ist, kann es vergrößert werden, um die Einstellungen der Nebenkurven zu überprüfen. Der Zoom erfolgt mit zwei Fingern, um das Diagramm zu vergrößern. Das abgebildete Diagramm ist vollständig vergrößert.</p>	 <p>Das Diagramm zeigt eine Kurve auf einem Gitter. Die Y-Achse reicht von 100 bis 300, die X-Achse von 7 bis 19,2. Ein Hand-Symbol mit zwei Fingern ist über das Diagramm gelegt, was den Zoom-Effekt verdeutlicht.</p>
Grafik auszoomen	<p>Sobald das Diagramm in den Standardzustand gezoomt wurde, erscheint in der oberen rechten Ecke eine Schaltfläche zum Auszoomen.</p> <p>Das Diagramm wird auf die Standardposition ausgezoomt, wobei eine schöne Transaktionsanimation zu sehen ist.</p>	 <p>Das Diagramm ist in der Standardansicht dargestellt. In der oberen rechten Ecke ist eine kleine Schaltfläche (ein Quadrat mit einem Kreis) zu sehen, die durch einen Pfeil hervorgehoben ist.</p>
Basis-Komponentenleiste ziehen	<p>Die Basis-Komponentenleiste kann entlang der Wertachse gezogen werden. Die BasisKomponente ändert sich entsprechend und wird zu allen Festpunkten hinzugefügt.</p>	 <p>Das Diagramm zeigt eine Kurve auf einem Gitter. Die Y-Achse reicht von 0 bis 400, die X-Achse von 0 bis 25. Eine vertikale blaue Leiste ist an der Position x=5 platziert. Ein Doppelpfeil zeigt an, dass diese Leiste horizontal entlang der X-Achse verschoben werden kann.</p>

Name	Beschreibung	Ergebnis im Control
<p>Einzoomen / Auszoomen mit Tasten</p>	<p>Die beiden Schaltflächen in der oberen rechten Ecke des Diagramms dienen zum manuellen Vergrößern und Verkleinern des Diagramms.</p> <p>Durch Klicken auf die Schaltfläche Einzoomen wird das Diagramm um eine Stufe vergrößert. Der maximale 4-fach-Zoom ist in 8 Stufen unterteilt.</p> <p>Durch Klicken auf die Schaltfläche Auszoomen wird das Diagramm in einem Schritt verkleinert.</p>	

Verfügbar: seit Version 12.6.0

4.5.5 Schritt für Schritt

1. Ziehen Sie das Control für den Kurveneditor aus dem Fenster "Werkzeuge" auf Ihren Inhalt.
2. Richten Sie das Attribut Editorprofile ein.
 - Editorprofile haben von allen Attributen die höchste Priorität, sie werden bei der Initialisierung des Controls zuerst geladen.
 - Das aktive Profil überschreibt beim Laden andere darin eingestellte Attribute.
 - Editorprofile sind ein Array von Profilen; mindestens ein Profil sollte für die Verwendung des Controls konfiguriert sein.
 - Wenn Sie möchten, dass der Kurveneditor jedes Mal mit einem statischen Profil startet, können Sie das Objekt im TwinCAT-HMI-Objekteditorfenster programmieren.



3. Einstellung der Position der Basisachse.

- Das Attribut für die Position der Basisachse kann auf vertikal oder horizontal eingestellt werden. Damit wird festgelegt, wie das Control aussieht und sich verhält.
- Für die Einheitenumrechnung, die Erkennung von Anzeigeeinheiten und die Anzeige von Einheitentexten auf der Wertachse greift das Kurveneditor-Control auf Einheitenumrechnungsfunktionen zurück, die von den `Beckhoff.TwinCAT.HMI.Plastic.Controls`-Paketen und dem Configurator-Control bereitgestellt werden.

4. Datentyp der Wertachse einstellen.

- Sie können die Datei `unit_config.json` so konfigurieren, dass die Einheitengruppe in die Datei mit der zugehörigen Anzeige- und Systemeinheitenkonfiguration aufgenommen wird.
- Das Configurator-Control muss mit der Standard-ID auf oberster Ebene im Projekt instanziiert werden. (Erstes Control in `Desktop.view`)

i Es ist möglich, den Kurveneditor ohne die vom Plastic-Paket bereitgestellte Einheitenumrechnung zu verwenden.

Legen Sie eine Leerstring als Datentyp für die Wertachse fest. Auf diese Weise funktioniert das Control weiterhin ohne Einheitenumrechnung.

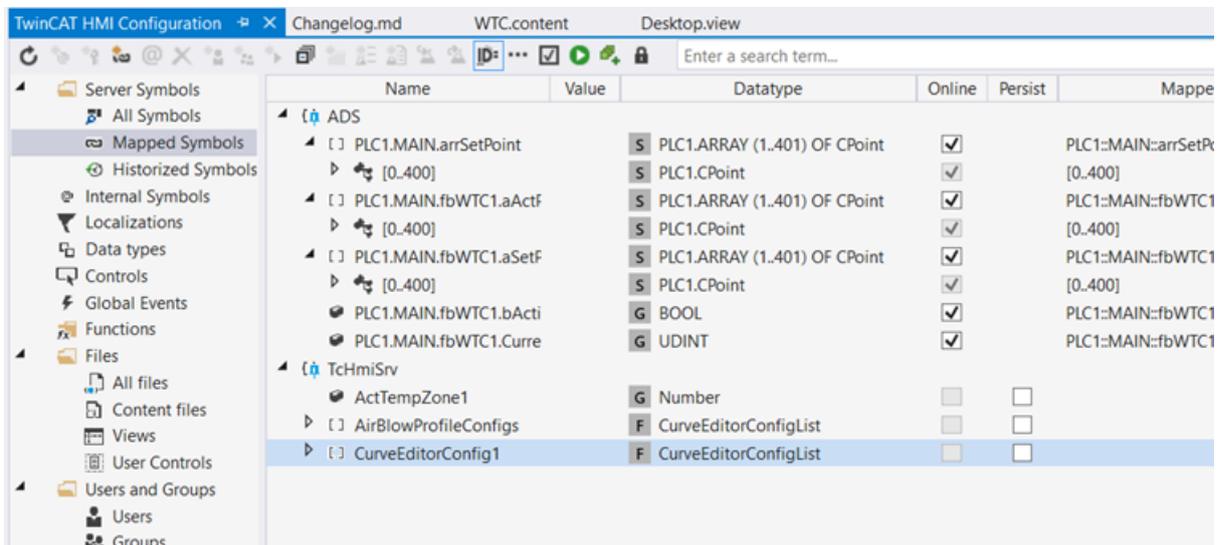
5. Setzen Sie alle anderen Attribute oder binden Sie andere Attribute an andere Controls oder Serversymbole.

4.5.6 Verwendung der Rezeptverwaltung mit dem Kurveneditor

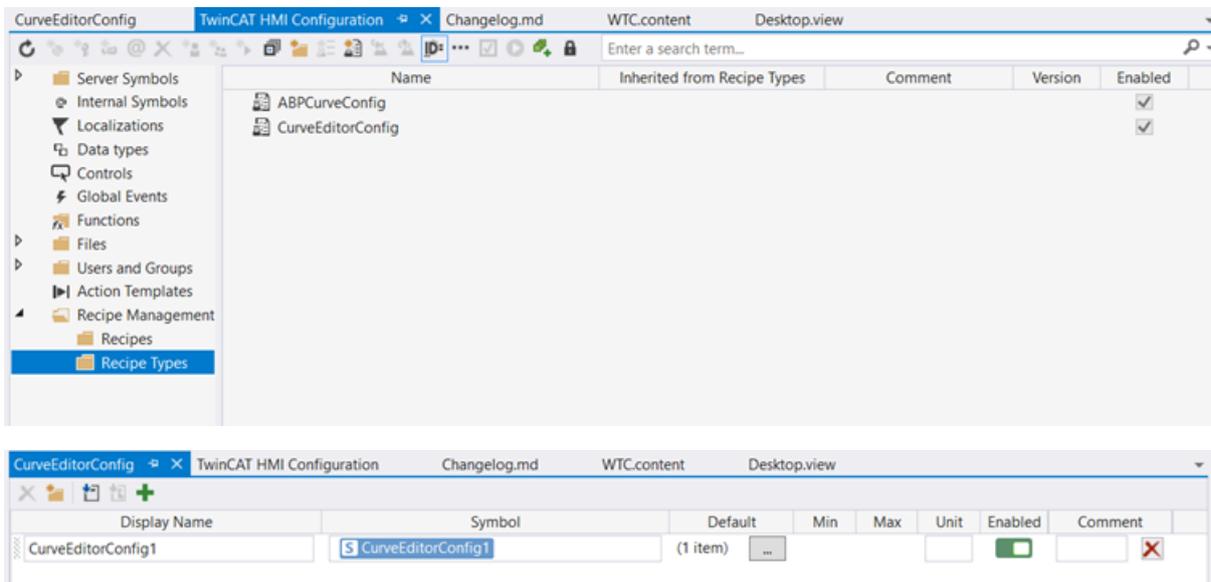
Mit dem Kurveneditor können mehrere Profile in einer Liste von Profilen gespeichert werden. Diese Profilliste kann in Rezepten gespeichert werden.

Um die TwinCAT Rezeptverwaltung mit dem Kurveneditor zu nutzen, sind folgende Schritte zu beachten.

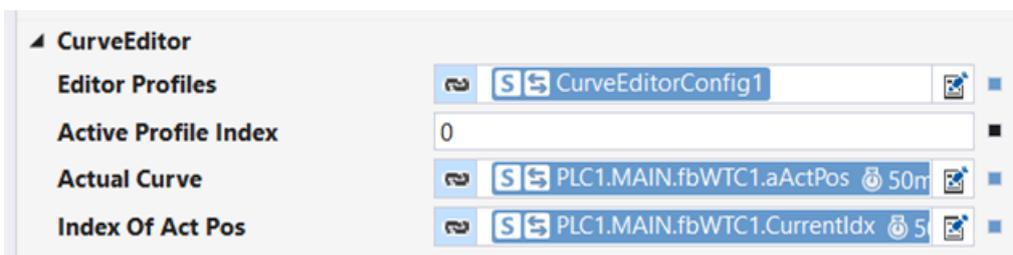
1. Erstellen Sie ein Serversymbol, das als Quelle für die Liste der Editorprofile für den Controller verwendet wird. Für jedes Kurveneditor-Control wird ein Symbol vom Typ CurveEditorConfigList benötigt.



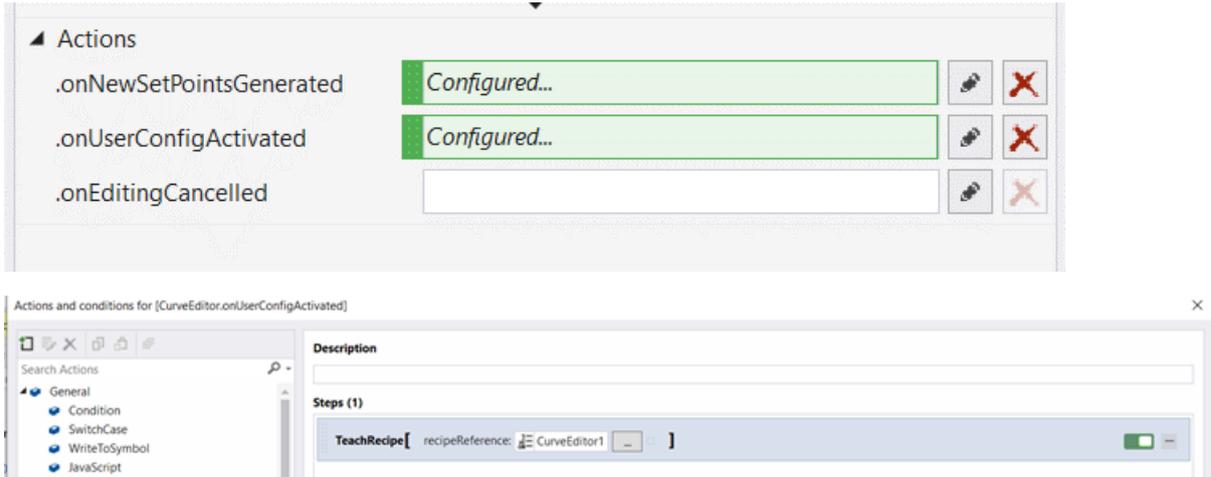
2. Fügen Sie dieses Server-Symbol zu einem Rezepttyp hinzu. Es kann eine einzelne Rezeptart sein oder mit anderen Symbolen in dem Produktrezepttyp gruppiert werden.



3. Verwenden Sie dieses Serversymbol als Konfigurationssymbol für die Eigenschaft Editor Profiles [▶ 57]. Eine bidirektionale Bindung ist der einfachste Weg, um das Symbol zu aktualisieren, wenn es eine Änderung in der Konfigurationsliste gibt.



- Bei der bidirektionalen Bindung wird das Serversymbol bei jeder Änderung aktualisiert. Und die Funktion zum Anlernen von Rezepten kann beim Ereignis **.onUserConfigActivated** aufgerufen werden.



4.6 FavoriteBarControl

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

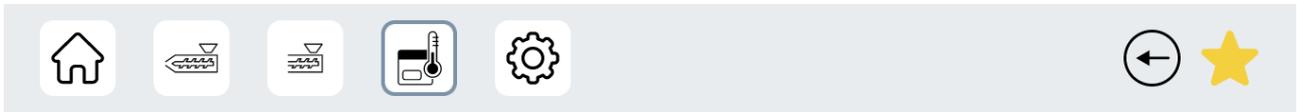
Verfügbar: seit Version 12.6.0

Das FavoriteBarControl kann mit dem Control Responsive Navigation Bar in TwinCAT HMI verwendet werden. Der Zweck dieses Controls besteht darin, den Benutzern die Möglichkeit zu geben, beliebige Inhalte auf einfache Weise als Favoriten festzulegen, indem sie eine Verknüpfung erstellen, die immer auf dem FavoriteBarControl verfügbar ist.

Das FavoriteBarControl ermöglicht es Nutzern, aktuelle Inhalte, die auf dem Zielhost angezeigt werden, als Favoriten auszuwählen, indem sie auf die Schaltfläche **Add to Favorites** klicken, die mit einem Sternsymbol versehen ist. Sobald der Inhalt hinzugefügt wurde, erscheint er als Verknüpfung auf dem FavoriteBarControl, so dass der Benutzer jederzeit schnell zu ihm navigieren kann.

Das FavoriteBarControl ist in hohem Maße anpassbar, mit Optionen zur Anzeige von Detailpfaden oder Symbolen für jeden ausgewählten Inhalt.

Es wird dazu beitragen, die Benutzerfreundlichkeit und Zugänglichkeit von TwinCAT HMI zu verbessern, so dass die Benutzer einfach auf ihre Favoriten-Inhalte zugreifen und ihre Arbeitsabläufe rationalisieren können.



4.6.1 Attribute

Kategorie: Common

Name	Beschreibung
FavContentList [► 82]	Standardliste der Inhalte, die auf dem Control angezeigt werden.
RegionControlSymbol [► 84]	Zielregion, in der ein favorisierter Inhalt angezeigt werden soll.
MenuDataSrc [► 84]	MenuDataSrc eines ResponsiveNavigation.TchmiNavigationBar-Controls, in dem weitere Informationen über das Favoriten-Control wie Symbol, Descriptor usw. verfügbar sind.

Name	Beschreibung
DisplayContentPath [► 81]	Der Benutzer kann auswählen, ob der vollständige Pfad jedes favorisierten Inhalts auf dem Control angezeigt werden soll.
FavButtonPosition [► 81]	Position der Schaltflächen Add to Favorite und Go Back .

Kategorie: Text

Name	Beschreibung
TextFontSize [► 84]	Die Schriftgröße des angezeigten Textes.
FontSizeUnit [► 83]	Legt die Einheit der Schriftgröße fest.

Kategorie: Image

Name	Beschreibung
ImageHeight [► 83]	Höhe eines beliebigen Bildes oder Symbols, das auf einer Favoriten-Schaltfläche angezeigt wird.
ImageHeightUnit [► 83]	Einheit für die Bildhöhe.
ImageWidth [► 83]	Breite eines beliebigen Bildes oder Symbols, das auf einer Favoriten-Schaltfläche angezeigt wird.
ImageWidthUnit [► 83]	Einheit für die Bildbreite

4.6.1.1 DisplayContentPath

Wenn ein Benutzer einen neuen Favoriten hinzufügt, kann das Control prüfen, ob ein Bild für diesen Inhalt in [MenuSrcData](#) [► 84] konfiguriert wurde. Wenn ein Bild verfügbar ist, zeigt das Control dieses Bild auf der Favoriten-Schaltfläche an. Wenn kein Bild konfiguriert wurde, zeigt das Control Initialen mit zwei Buchstaben an, um den Inhalt visuell darzustellen.

Darüber hinaus ist das `DisplayContentPath`-Attribut eine nützliche Option für Benutzer, die den vollständigen Pfad für ihren Inhalt in [MenuSrcData](#) [► 84] sehen möchten.

Wenn dieses Attribut auf `TRUE` gesetzt ist, wird der vollständige Pfad auf der Favoriten-Schaltfläche neben dem Symbol oder den Initialen angezeigt.

Favoriten-Schaltflächen mit `DisplayContentPath` auf `TRUE` gesetzt:



Schema: `tchmi:framework#/definitions/TcHmi.Controls.ResponsiveNavigation.TcHmiNavigationBar.MenuItemLitst`

Attribut-Setter: `setDisplayContentPath`

Attribut-Getter: `getDisplayContentPath`

Verfügbar: seit Version 12.6.0

4.6.1.2 FavButtonPosition

Der Benutzer kann auswählen, ob die Schaltflächen "Add To Favorite" und "Go Back" am linken oder rechten Ende des Controls positioniert werden.

Schema: `tchmi:framework#/definitions/FavButtonPosition`

Attribut-Setter: `setFavButtonPosition`

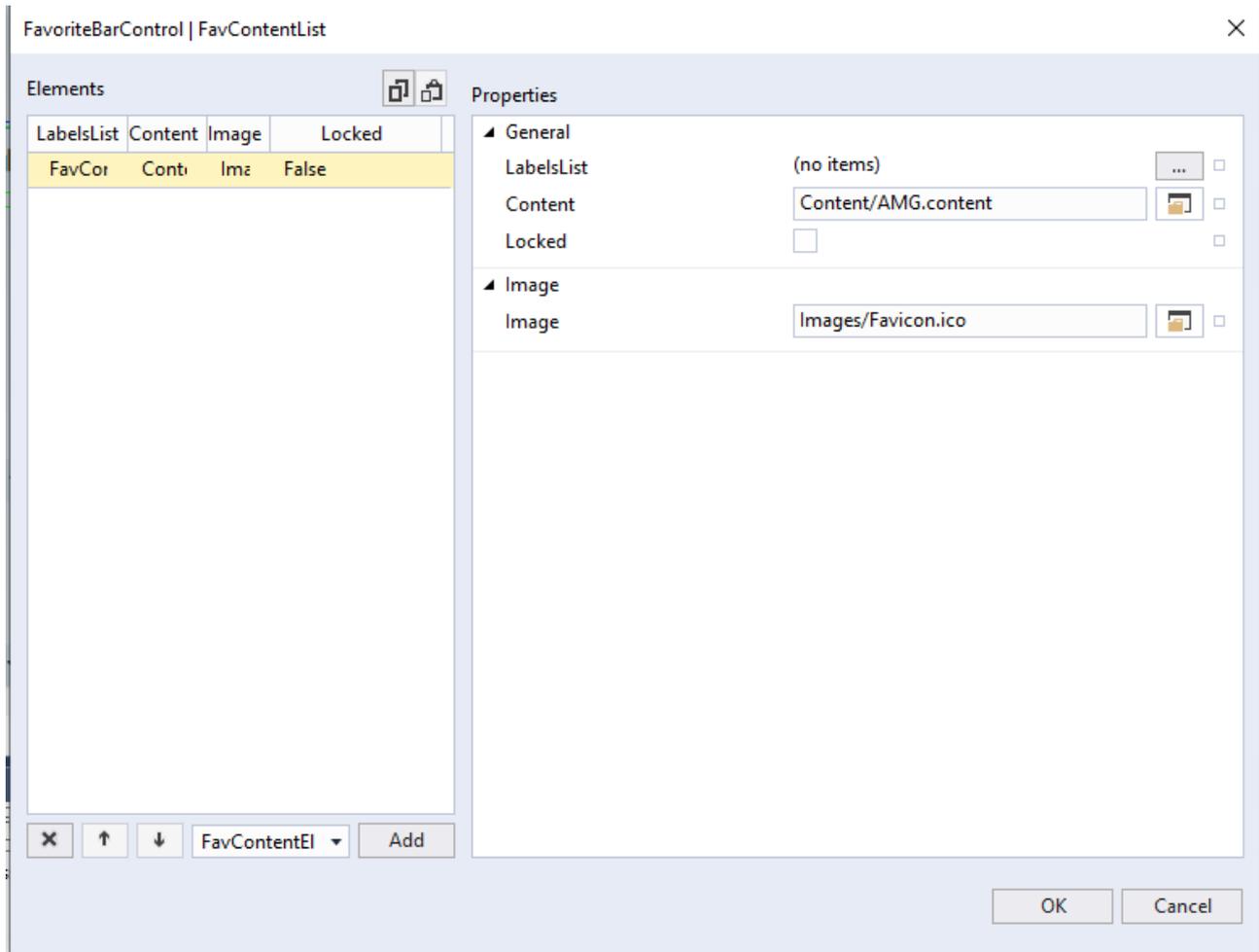
Attribut-Getter: getFavButtonPosition

Verfügbar: seit Version 12.6.0

4.6.1.3 FavContentList

Mit dem FavContentList-Attribut können Benutzer eine Standardliste von Inhalten konfigurieren, die auf dem FavoriteBarControl angezeigt werden sollen. Das bedeutet, dass der Benutzer beim Start der HMI eine vordefinierte Reihe von Favoriten-Verknüpfungen sieht, mit denen er schnell zu den am häufigsten aufgerufenen Inhalten navigieren kann.

In einem Dialogfenster können Sie die Standard-Favoritenliste konfigurieren:



Konfiguration von jedem Favoriten-Element:

Name	Beschreibung
LabelsList	Wenn ein Inhaltspfad auf der Favoriten-Schaltfläche angezeigt werden soll, kann er als Liste von Strings konfiguriert werden. Alle Strings werden mit dem Zeichen "/" verbunden, um den Inhaltspfad zu bilden.
Content	Der Pfad zur Inhaltsdatei
Locked	Wenn ein Favoriten-Element gesperrt ist, kann es nicht aus der Favoritenliste entfernt werden, indem der Benutzer auf die Schaltfläche "Add to Favorite" klickt.
Image	Pfad zu einem Bild, um es als Symbol für die Favoriten-Schaltfläche anzuzeigen.

Schema: tchmi:framework#/definitions/FavContentElementList

Attribut-Setter: setFavContentList

Attribut-Getter: getFavContentList

Verfügbar: seit Version 12.6.0

4.6.1.4 **FontSizeUnit**

Legt die Einheit der Schriftgröße fest.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getFontSizeUnit

Attribut-Setter: setFontSizeUnit

Verfügbar: seit Version 12.6.0

4.6.1.5 **ImageHeight**

Mit diesem Attribut wird die Höhe der Symbole, die auf den Favoriten-Schaltflächen angezeigt werden, festgelegt.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Setter: setImageHeight

Attribut-Getter: getImageHeight

Verfügbar: seit Version 12.6.0

4.6.1.6 **ImageHeightUnit**

Mit diesem Attribut wird die Einheit der Höhe der Bilder, die auf den Favoriten-Schaltflächen angezeigt werden, festgelegt.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Setter: setImageHeightUnit

Attribut-Getter: getImageHeightUnit

Verfügbar: seit Version 12.6.0

4.6.1.7 **ImageWidth**

Mit diesem Attribut wird die Breite der Bilder festgelegt, die auf den Favoriten-Schaltflächen angezeigt werden.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Setter: setImageWidth

Attribut-Getter: getImageWidth

Verfügbar: seit Version 12.6.0

4.6.1.8 **ImageWidthUnit**

Mit diesem Attribut wird die Einheit der Breite der Bilder, die auf den Favoriten-Schaltflächen angezeigt werden, festgelegt.

Schema: tchmi:framework#/definitions/MeasurementValueUnit

Attribut-Setter: setImageWidthUnit

Attribut-Getter: getImageWidthUnit

Verfügbar: seit Version 12.6.0

4.6.1.9 MenuDataSrc

Wenn ein Benutzer einen neuen Favoriten-Inhalt auf dem Bildschirm hinzufügt, benötigt die Favoritenleiste weitere Informationen über den Inhalt, z. B. Beschriftungen, Bildpfade usw.

Durch die Verknüpfung der Menüquelle der Navigationsleiste mit dem Control der Favoritenleiste können diese Informationen dem Control der Favoritenleiste zur Verfügung gestellt werden.

Der Vorteil dieses Ansatzes ist, dass der Benutzer alle seine Navigationsoptionen im TchmiNavigationBar-Control konfigurieren kann und das FavoriteBarController diese Konfiguration beim Hinzufügen eines neuen Favoriteninhalts lesen kann.

Schema: tchmi:framework#/definitions/
Tchmi.Controls.ResponsiveNavigation.TchmiNavigationBar.MenuItemList

Attribut-Setter: setMenuDataSrc

Attribut-Getter: getMenuDataSrc

Verfügbar: seit Version 12.6.0

4.6.1.10 RegionControlSymbol

Das Control-Symbol des Zielbereichs, in dem der ausgewählte Inhalt angezeigt werden soll.

Schema: tchmi:framework#/definitions/
Tchmi.Controls.ResponsiveNavigation.TchmiNavigationBar.TargetRegionControlSymbol

Attribut-Setter: setFavContentList

Attribut-Getter: getFavContentList

Verfügbar: seit Version 12.6.0

4.6.1.11 TextFontSize

Legt die Schriftgröße des angezeigten Textes fest.

Schema: tchmi:general#/definitions/UINT

Attribut-Getter: getTextFontSize

Attribut-Setter: setTextFontSize

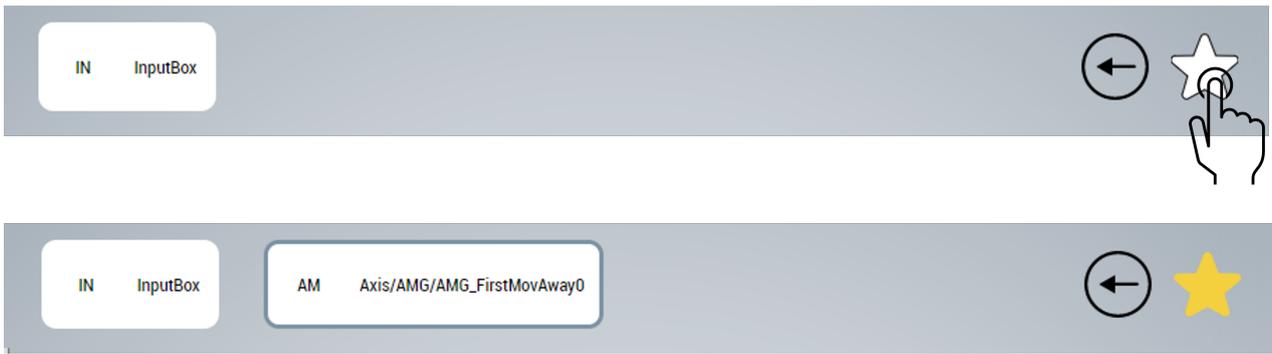
Verfügbar: ab Version 12.6.0

4.6.2 Benutzerinteraktionen

4.6.2.1 Hinzufügen oder Entfernen eines Inhalts als Favorit in einer Favoritenleiste

Sobald ein Benutzer zu einem Inhalt auf dem Control des Zielbereichs navigiert, kann er diesen Inhalt zur Favoritenleiste hinzufügen, indem er auf die Schaltfläche "Add to Favorite" klickt, die als Stern angezeigt wird.

Dadurch wird der aktuelle Inhalt zur Favoritenleiste hinzugefügt und das Aussehen der Schaltfläche **Add to Favorite** von einem weißen Stern zu einem gelben ausgefüllten Stern geändert.



Wenn ein Benutzer einen Inhalt als Favorit hinzufügt, wird jedes Mal, wenn derselbe Benutzer zu diesem Inhalt navigiert, die Schaltfläche **Add to Favorite** als gelber ausgefüllter Stern angezeigt. Dadurch weiß der Benutzer, dass der Inhalt als Favorit markiert ist.

Das Verfahren zum Entfernen eines Inhalts aus der Favoritenleiste ist ähnlich. Wenn der Benutzer zu einem Inhalt navigiert, der zuvor zur Favoritenleiste hinzugefügt wurde, und auf die Schaltfläche **Add to Favorite** klickt, wird dieser Inhalt aus der Favoritenleiste entfernt.

4.6.2.2 Zurück zum letzten Inhalt

Das FavoriteBarControl bietet den Benutzern auch die Möglichkeit, die zuletzt besuchten Inhalte neu zu laden. Um zum zuletzt navigierten Inhalt zurückzukehren, kann der Benutzer auf die Schaltfläche **Go Back** im Control klicken.

Der Benutzer kann zu bis zu 9 besuchten Inhalten zurück navigieren.

4.7 InputBox

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

InputBox ist ein Control, mit dem der Benutzer numerische oder boolesche Werte in das TwinCAT-HMI-Projekt eingeben kann.

Hauptmerkmale

- Bereichsprüfung
- Einheitenumrechnung von physikalischen Größen
- Visualisierung nach Datentyp (BOOL, INT, FLOAT)
- Eingabefilter, der nur numerische Eingaben zulässt

InputBox für FLOAT-Werte:



InputBox für BOOL-Werte:



4.7.1 Attribute

Kategorie: Colors

Name	Beschreibung
TextColor [▶ 91]	Die Schriftfarbe des im Normalzustand angezeigten Textes.

Kategorie: Inputbox

Name	Beschreibung
DataType [▶ 87]	Der Datentyp des Control-Werts.
IsReadOnly [▶ 88]	Wenn <code>TRUE</code> , ist das Control schreibgeschützt.
UnitGroup [▶ 92]	Die Einheitengruppe des Control-Wertes.
StepSize [▶ 91]	Die Schrittweite zum Erhöhen und Verringern des Wertes mit den Funktionen IncreaseValueByStepSize [▶ 92] und DecreaseValueByStepSize [▶ 93] mit den Tasten auf dem Ziffernblock.

Kategorie: Text

Name	Beschreibung
TextFontSize [▶ 91]	Die Schriftgröße des angezeigten Textes.
DigitsAfterDecimal [▶ 87]	Die angezeigte Anzahl der Nachkommastellen für <code>FloatDataType</code> .
DisplayUnitText [▶ 88]	Zeigt den konfigurierten Text der Anzeigeeinheit neben dem Wert an.
TextHorizontalAlignment [▶ 91]	Horizontale Ausrichtung für das Attribut TextVal [▶ 92].
ContentPadding [▶ 87]	Padding für das Attribut TextVal [▶ 92].

Kategorie: Common

Name	Beschreibung
MinVal [▶ 89]	Der Minimalwert des Attributs TextVal [▶ 92].
MaxVal [▶ 89]	Der Maximalwert des Attributs TextVal [▶ 92].
RefVal [▶ 90]	Referenzwert des Attributs TextVal [▶ 92], der für die Berechnung in Prozent verwendet wird.
TextVal [▶ 92]	Der Wert, der im Control angezeigt werden soll.
ReadBackOnWrite [▶ 89]	Dieses Attribut legt fest, ob das Control den mit dem Attribut TextVal [▶ 92] verknüpften Symbolwert jedes Mal lesen und aktualisieren soll, wenn der Benutzer einen neuen Wert eingibt.
UnitText [▶ 92]	Einheitentext oder Abkürzungs-String für die aktuelle Anzeigeeinheit.

Kategorie: Fixed-Units

Name	Beschreibung
DisplayUnit [▶ 88]	Legt eine feste Einheit als Anzeigeeinheit für das Control fest. Das Attribut UnitGroup [▶ 92] muss ein leerer String sein, um dies zu verwenden. Gilt nur, wenn das Attribut DataType [▶ 87] auf <code>FloatDataType</code> gesetzt ist.
SystemUnit [▶ 91]	Legt eine feste Einheit als Systemeinheit für das Control fest. Das Attribut UnitGroup [▶ 92] muss ein leerer String sein, um dies zu verwenden. Gilt nur, wenn das Attribut DataType [▶ 87] auf <code>FloatDataType</code> gesetzt ist.

Recipe Management:

Name	Beschreibung
RecipeManagement [► 89]	Wählen Sie aus, ob InputBox das Serversymbol, das an das Attribut TextVal [► 92] gebunden ist, zu einem Rezepttyp hinzufügen soll.
RecipeTypeName [► 90]	Mit diesem Attribut kann der Benutzer den Namen des Rezepttyps festlegen, zu dem das Control die ihm zugeordneten Symbole hinzufügt und alle Änderungen einlernt.

User Changes Logging:

Name	Beschreibung
LogChangesByUser [► 88]	Logging der vom Benutzer auf dem Client vorgenommenen Änderungen in einer Logdatei.
ChangeLogDescriptor [► 87]	Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

4.7.1.1 ChangeLogDescriptor

Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

Schema: "tchmi:general#/definitions/STRING"

Attribut-Getter: getChangeLogDescriptor

Attribut-Setter: setChangeLogDescriptor

Verfügbar: ab Version 12.14.0

4.7.1.2 ContentPadding

Dieses Attribut legt das Padding des Textwertes innerhalb der Inputbox fest. Dieser Abstand zwischen dem Text und dem Rand kann auf allen 4 Seiten des Textes individuell eingestellt werden.

Schema: tchmi:framework#/definitions/Padding

Attribut-Getter: getContentPadding

Attribut-Setter: setContentPadding

Verfügbar: seit Version 12.6.0

4.7.1.3 DataType

Der gewünschte Datentyp des Controls: `BoolDataType`, `IntDataType`, `FloatDataType` oder `TimeDataType`.

Schema: tchmi:framework#/definitions/FBValue_Type

Attribut-Getter: getDataType

Attribut-Setter: setDataType

Verfügbar: seit Version 12.6.0

4.7.1.4 DigitsAfterDecimal

Die angezeigte Anzahl der Nachkommastellen für `FloatDataType`.

Schema: tchmi:general#/definitions/UJINT

Attribut-Getter: getDigitsAfterDecimal

Attribut-Setter: setDigitsAfterDecimal

Verfügbar: seit Version 12.6.0

4.7.1.5 DisplayUnit

Legt eine feste Einheit als Anzeigeeinheit für das Control fest. Das Attribut [UnitGroup \[► 92\]](#) muss ein leerer String sein, um dies zu verwenden. Gilt nur, wenn das Attribut [DataType \[► 87\]](#) auf `FloatDataType` gesetzt ist.

Schema: tchmi:general#/definitions/String

Attribut-Getter: getDispUnit

Attribut-Setter: setDispUnit

Verfügbar: seit Version 12.6.0

4.7.1.6 DisplayUnitText

Wenn dieses Attribut auf `TRUE` gesetzt ist, wird der Einheitentext der aktuellen Variableneinheit neben der Inputbox angezeigt.

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getDisplayUnitText

Attribut-Setter: setDisplayUnitText

Verfügbar: seit Version 12.6.0

4.7.1.7 FontSizeUnit

Legt die Einheit der Schriftgröße fest.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getFontSizeUnit

Attribut-Setter: setFontSizeUnit

Verfügbar: seit Version 12.6.0

4.7.1.8 IsReadOnly

Wenn `TRUE`, lässt das Control keine Benutzerinteraktionen zu.

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getReadOnly

Attribut-Setter: setReadOnly

Verfügbar: seit Version 12.6.0

4.7.1.9 LogChangesByUser

Wenn dieses Attribut auf `TRUE` gesetzt ist, loggt das Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket jede Änderung durch den Benutzer auf dem HMI-Bildschirm in einer Logdatei.

Um dieses Attribut zu verwenden, muss [UserLoggerSymbol \[► 45\]](#) mit einem Binding an [FB_UserLoggingHmi](#) konfiguriert werden.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getLogUserChanges

Attribut-Setter: setLogUserChanges

Verfügbar: ab Version 12.12.0

4.7.1.10 MaxVal

Die Maximalgrenze des Textwerts, der eingegeben werden kann.

Schema: tchmi:framework#/definitions/ValueAttribute

Attribut-Getter: getMaxVal

Attribut-Setter: setMaxVal

Verfügbar: seit Version 12.6.0

4.7.1.11 MinVal

Die Minimalgrenze des Textwerts, der eingegeben werden kann.

Schema: tchmi:framework#/definitions/ValueAttribute

Attribut-Getter: getMinVal

Attribut-Setter: setMinVal

Verfügbar: seit Version 12.6.0

4.7.1.12 ReadBackOnWrite

Ist dieser Wert auf true gesetzt, liest und aktualisiert das Control den Symbolwert, der mit dem Attribut [TextVal \[► 92\]](#) verknüpft ist, jedes Mal, wenn der Benutzer einen neuen Wert eingibt.

Dieses Verhalten kann nützlich sein, wenn das Zielsymbol ignoriert, dass ein Wert von einem Benutzer eingegeben wird, und der Subscription-Modus des Symbols "Change" ist.

Weitere Informationen zum Subscription-Modus finden Sie [hier](#).

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getReadBackOnWrite

Attribut-Setter: setReadBackOnWrite

Verfügbar: ab Version 12.14.0

4.7.1.13 RecipeManagement

Legen Sie fest, ob die InputBox coll das Serversymbol, das an das Attribut [TextVal \[► 92\]](#) gebunden ist, zu einem Rezepttyp hinzufügen soll.

Wenn das Projekt das Paket [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper \[► 198\]](#) enthält, fügt das Control das darauf abgebildete Serversymbol zum Rezepttyp hinzu, der durch das Attribut [RecipeTypeName \[► 90\]](#) definiert ist.

Nach jedem Benutzerwechsel lernt das Paket RecipeHelper das aktuell aktive Rezept ein.

Schema: "tchmi:framework#/definitions/TcHmi.Controls.Plastic.RecipeManagement"

Attribut-Getter: getRecipeManagement

Attribut-Setter: setRecipeManagement

Verfügbar: ab Version 12.14.0

4.7.1.14 RecipeTypeName

Mit diesem Attribut kann der Benutzer den Namen des Rezepttyps festlegen, zu dem das Control die ihm zugewiesenen Symbole hinzufügt und alle Änderungen speichert.

Damit das Control Symbole zum Rezepttyp hinzufügen kann, muss das Projekt das Paket [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper \[► 198\]](#) enthalten und das Attribut [RecipeManagement \[► 89\]](#) muss auf `IncludeToRecipe` gesetzt sein.

i Wenn ein Symbol bereits Teil eines Rezepttyps ist, hat diese Eigenschaft keine Auswirkungen auf die Rezepte. Nur für das Hinzufügen von Symbolen, die nicht Teil eines Rezepttyps sind, wird das Attribut `RecipeTypeName` berücksichtigt.

Schema: "tchmi:general#/definitions/STRING"

Attribut-Getter: `getRecipeTypeName`

Attribut-Setter: `setRecipeTypeName`

Verfügbar: ab Version 12.14.0

4.7.1.15 RecipeType (veraltet)

i **Veraltetes Attribut**
Dieses Attribut wird entfernt und kann nicht mehr verwendet werden.

Wenn dieses Attribut auf 'R' gesetzt ist und das Projekt das Paket [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper \[► 198\]](#) enthält, fügt die `InputBox` das Serversymbol, das an das Attribut [TextVal \[► 92\]](#) gebunden ist, zum Rezepttyp `ActRcpDataType` hinzu.

Nach jedem Benutzerwechsel lernt das Paket `RecipeHelper` das aktuell aktive Rezept ein.

Schema: `tchmi:framework#/definitions/RecipeType`

Attribut-Getter: `getRecipeType`

Attribut-Setter: `setRecipeType`

Verfügbar: ab Version 12.6.0

Veraltet: ab Version 12.14.0

Ersetzt durch: [RecipeManagement \[► 89\]](#)

4.7.1.16 RefVal

Der Referenzwert ist erforderlich, wenn die Anzeigeeinheit Prozent [%] ist. Dieser Attributwert wird für die Berechnung des 100%-Bezugs verwendet.

Schema: `tchmi:framework#/definitions/ValueAttribute`

Attribut-Getter: `getRefVal`

Attribut-Setter: `setRefVal`

Verfügbar: seit Version 12.6.0

4.7.1.17 StepSize

Die Schrittweite zum Erhöhen und Verringern des Wertes mit den Funktionen [IncreaseValueByStepSize](#) [[▶ 92](#)] und [DecreaseValueByStepSize](#) [[▶ 93](#)] mit den Tasten auf dem Ziffernblock.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getStepSize

Attribut-Setter: setStepSize

Verfügbar: seit Version 12.6.0

4.7.1.18 SystemUnit

Legt eine feste Einheit als Systemeinheit für das Control fest. Das Attribut [UnitGroup](#) [[▶ 92](#)] muss ein leerer String sein, um dies zu verwenden. Gilt nur, wenn das Attribut [DataType](#) [[▶ 87](#)] auf `FloatDataType` gesetzt ist.

Schema: tchmi:general#/definitions/String

Attribut-Getter: getSysUnit

Attribut-Setter: setSysUnit

Verfügbar: seit Version 12.6.0

4.7.1.19 TextColor

Legt die Textfarbe im Normalzustand des Texttyp-Controls fest; wirkt sich nicht auf den Prüftyp des Controls aus.

Schema: tchmi:framework#/definitions/SolidColor

Attribut-Getter: getTextColor

Attribut-Setter: setTextColor

Verfügbar: seit Version 12.6.0

4.7.1.20 TextFontSize

Legt die Schriftgröße des Wertes fest.

Schema: tchmi:general#/definitions/UINT

Attribut-Getter: getTextFontSize

Attribut-Setter: setTextFontSize

Verfügbar: seit Version 12.6.0

4.7.1.21 TextHorizontalAlignment

Mit diesem Attribut wird die horizontale Ausrichtung des Wertetextes festgelegt. Der Text kann in der Mitte, links oder rechts in der Inputbox platziert werden.

Schema: tchmi:framework#/definitions/HorizontalAlignment

Attribut-Getter: getTextHorizontalAlignment

Attribut-Setter: setTextHorizontalAlignment

Verfügbar: seit Version 12.6.0

4.7.1.22 TextVal

Der Textwert eines Controls vom Typ Text und der Wert für den aktivierte Zustand bei einem Control vom Typ BOOL.

Schema: tchmi:framework#/definitions/ValueAttribute

Attribut-Getter: getTextVal

Attribut-Setter: setTextVal

Verfügbar: seit Version 12.6.0

4.7.1.23 UnitGroup

Die Einheitengruppe eines `FloatDataType`-Werts. Der String-Wert für die Einheitengruppe muss in der Datei `Unit_Config.json` [► 45] konfiguriert werden. Die Werte werden in der Anzeigeeinheit angezeigt und die Änderungen werden in der Systemeinheit in die SPS geschrieben.

Schema: tchmi:general#/definitions/String

Attribut-Getter: getUnitGroup

Attribut-Setter: setUnitGroup

Verfügbar: seit Version 12.6.0

4.7.1.24 UnitText

Dieses schreibgeschützte Attribut gibt einen Einheitentext oder eine Abkürzungs-String für die aktuelle Anzeigeeinheit des Controls zurück.

Schema: tchmi:general#/definitions/Any

Attribut-Getter: getUnitsText

Verfügbar: seit Version 12.6.0

4.7.2 Funktionen

Name	Beschreibung
IncreaseValueByStepSize [► 92]	Diese Funktion erhöht und aktualisiert den Textwert um die angegebene Schrittweite.
DecreaseValueByStepSize [► 93]	Diese Funktion verringert und aktualisiert den Textwert um die angegebene Schrittweite.

4.7.2.1 IncreaseValueByStepSize

Diese Funktion erhöht und aktualisiert den Textwert um die angegebene Schrittweite.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.7.2.2 DecreaseValueByStepSize

Diese Funktion verringert und aktualisiert den Textwert um die angegebene Schrittweite.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.7.3 Events

Kategorie: Control

Name	Beschreibung
onUserInteractionAction [▶ 93]	Dieses Ereignis wird ausgelöst, wenn der Text aufgrund einer Benutzerinteraktion geändert wird.

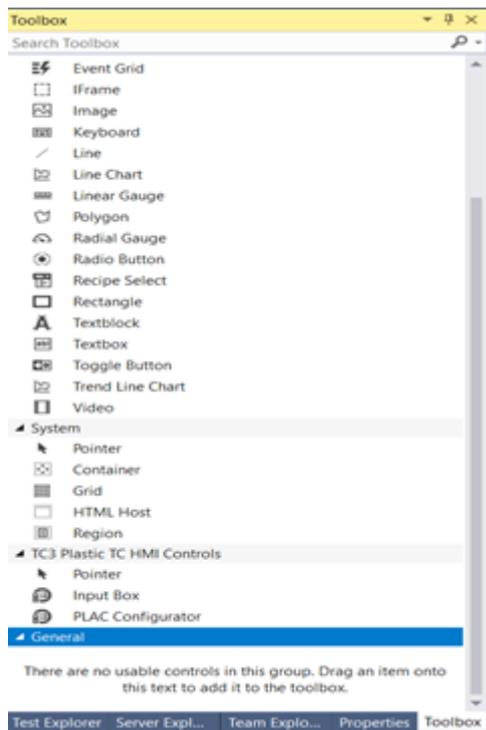
4.7.3.1 onUserInteractionAction

Dieses Ereignis wird ausgelöst, wenn das Attribut [TextVal](#) [[▶ 92](#)] durch eine Benutzerinteraktion geändert wird.

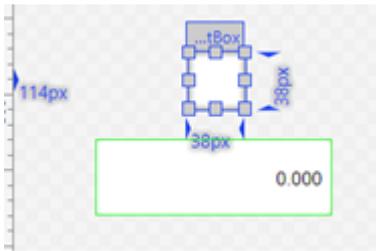
Verfügbar: seit Version 12.6.0

4.7.4 Schritt für Schritt

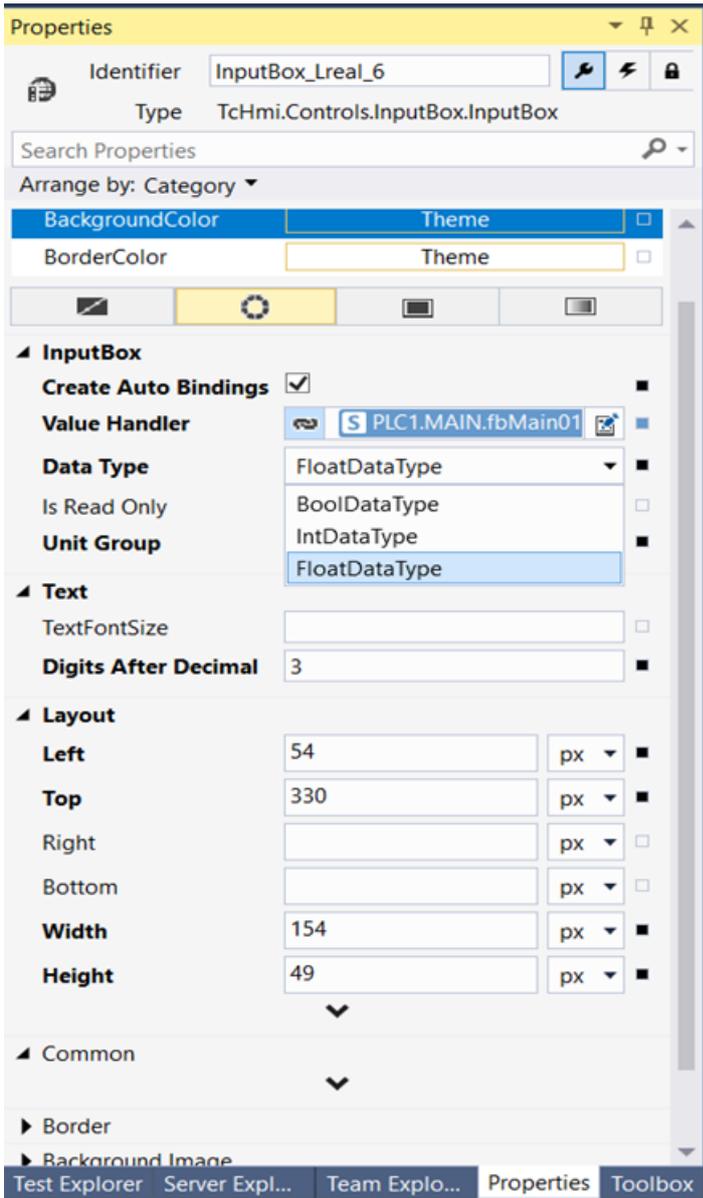
1. Ziehen Sie die Inputbox aus dem Werkzeug-Fenster auf die Inhalts- oder Ansichtsdatei.



2. Wenn das Attribut `DataType` [▶ 87] auf `BoolDataType` gesetzt ist, müssen seine Breite und Höhe für eine korrekte Anzeige gleich sein.



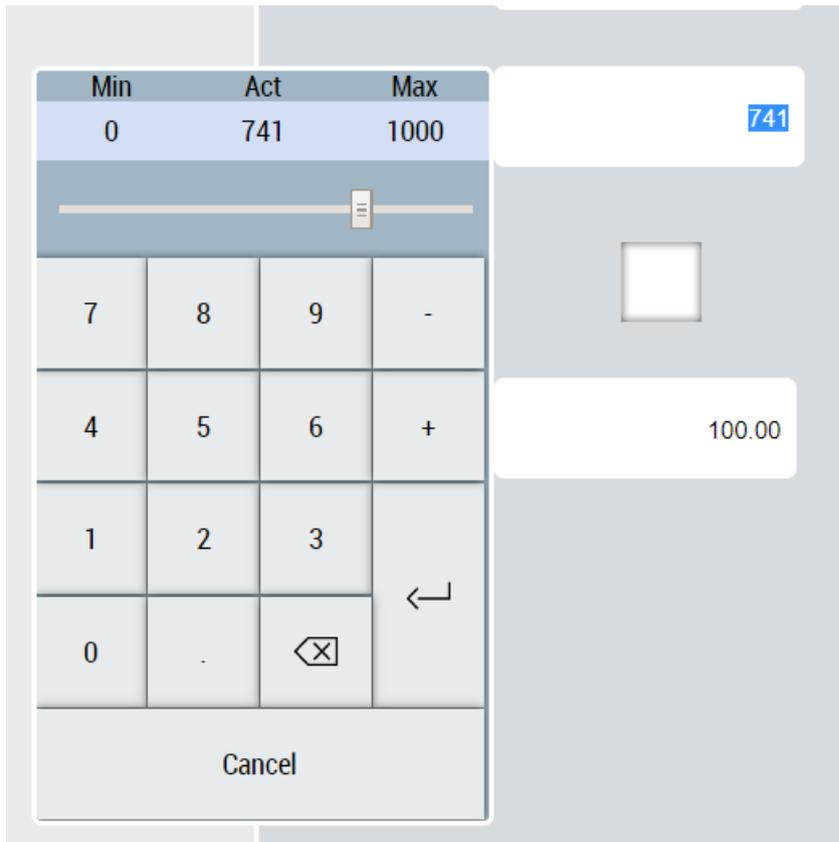
3. Das Attribut `DataType` [▶ 87] des Typs `FBValue_DataType` legt fest, wie das Control aussieht und funktioniert.



Wenn der Datentyp `BoolDataType` ist, bildet das Control eine Checkbox mit einem Häkchen, das dem booleschen Status des zugehörigen Attributs `TextVal` [▶ 92] entspricht.



Für `IntDataType`, `FloatDataType` oder `TimeDataType` wird das Control als Textfeld angezeigt, wobei die Zahlen aus dem Attribut `TextVal` [▶ 92] im Dezimalformat angezeigt werden. Das `InputPanel` wird angezeigt, wenn der Benutzer auf das Control klickt/tippt.

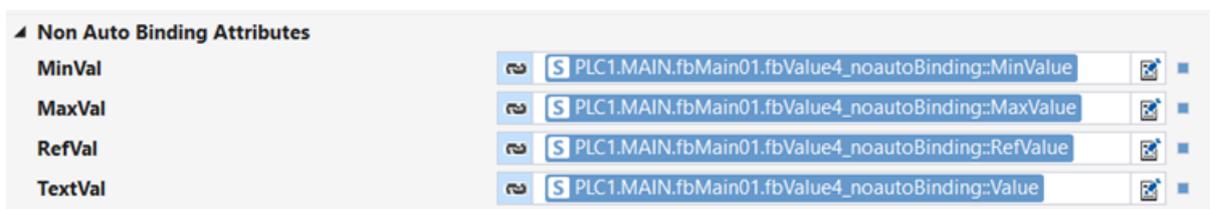


Wenn als Datentyp entweder `FloatDataType` oder `TimeDataType` eingestellt ist, werden Zahlen in der Anzeigeeinheit des angegebenen Datentyps angezeigt.

Der Einheitentext wird neben dem Control angezeigt, wenn das Attribut `DisplayUnitText` [▶ 88] auf `TRUE` gesetzt ist.



- Die Attribute `TextVal` [▶ 92], `RefVal` [▶ 90], `MinVal` [▶ 89] und `MaxVal` [▶ 89] können auf jede Zahl oder jedes boolesche Symbol im Projekt abgebildet werden.



4.8 ManualOperation

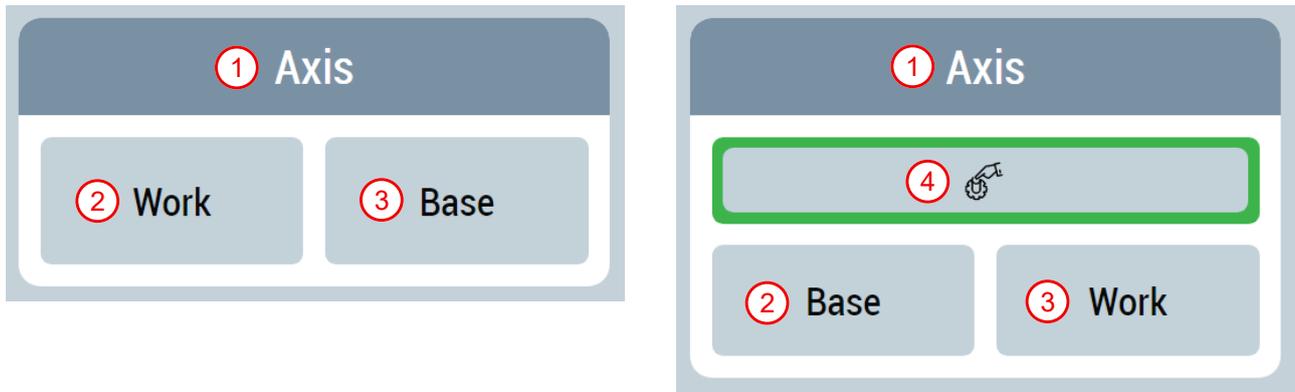
NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

Das Control `ManualOperation` kann als Bildschirmtasten zum manuellen Bewegen von Achsen verwendet werden. Die Achsen können in zwei Richtungen bewegt werden:

- In Richtung Grundposition
- In Richtung Arbeitsposition

Das Control verfügt über zwei Tastensätze und Zustandsanzeigen für jede Richtung.



1. Name der Achse, die konfiguriert werden kann
2. Taste zum Anfahren der Arbeitsposition mit Leuchtring (gelb: fahren, grün: Position erreicht)
3. Taste zum Anfahren der Grundposition mit Leuchtring (gelb: fahren, grün: Position erreicht)
4. Taste zur Aktivierung der [Hardwaresteuerung](#) [▶ 108] bei Verwendung mit CP2219-1025-0030 Panel mit Hardware-Encoder

4.8.1 Attribute

Kategorie: Parameters

Name	Beschreibung
ManualOperationSymbol [▶ 105]	Symbol vom Typ FB_ManualOperation.
AxisName [▶ 98]	Achsenname, der auf dem Control angezeigt wird.
BtnWorkText [▶ 101]	Texteigenschaft der Schaltfläche, die zum Senden eines Kommandos an die Arbeitsposition verwendet wird.
BtnBaseText [▶ 101]	Texteigenschaft der Schaltfläche, die zum Senden eines Kommandos an die Grundposition verwendet wird.
SingleMode [▶ 107]	Einstellung des Controls auf eine einzige Richtung.

Kategorie: Button Configuration

Name	Beschreibung
BorderRadius [▶ 101]	Legt den Rahmenradius für Schaltflächen auf dem Control fest.
ButtonOrientation [▶ 102]	Platziert die Schaltflächen horizontal oder vertikal nebeneinander.
InvertDirection [▶ 105]	Umgekehrte Reihenfolge der angeordneten Tasten.

Kategorie: Button Work Background Image

Name	Beschreibung
BackgroundImageWork [▶ 101]	Legt das Hintergrundbild für die Schaltfläche Arbeitsrichtung fest.
BackgroundImageWidthWork [▶ 100]	Legt die Bildbreite für die Schaltfläche Arbeitsrichtung fest.
BackgroundImageWidthWorkUnit [▶ 101]	Legt die Breitereinheit des Hintergrundbildes für die Arbeitsschaltfläche fest.

Name	Beschreibung
BackgroundImageHeightWork [▶ 98]	Legt die Bildhöhe für die Schaltfläche Arbeitsrichtung fest.
BackgroundImageHeightWorkUnit [▶ 99]	Legt die Höheneinheit des Hintergrundbildes für die Arbeitsschaltfläche fest.
BackgroundImageHorizontalAlignWork [▶ 99]	Legt die horizontale Ausrichtung des Bildes für die Schaltfläche Arbeitsrichtung fest.
BackgroundImageVerticalAlignWork [▶ 100]	Legt die vertikale Ausrichtung des Bildes für die Schaltfläche Arbeitsrichtung fest.
BackgroundImagePaddingWork [▶ 99]	Legt den Innenabstand des Bildes für die Schaltfläche Arbeitsrichtung fest.

Kategorie: Button Base Background Image

Name	Beschreibung
BackgroundImageBase [▶ 98]	Legt das Hintergrundbild für die Schaltfläche der Basisrichtung fest.
BackgroundImageWidthBase [▶ 100]	Legt die Bildbreite für die Schaltfläche der Basisrichtung fest.
BackgroundImageWidthBaseUnit [▶ 100]	Legt die Breitereinheit des Hintergrundbildes für die Basisschaltfläche fest.
BackgroundImageHeightBase [▶ 98]	Legt die Bildhöhe für die Schaltfläche der Basisrichtung fest.
BackgroundImageHeightBaseUnit [▶ 98]	Legt die Höheneinheit des Hintergrundbildes für die Basisschaltfläche fest.
BackgroundImageHorizontalAlignBase [▶ 99]	Legt die horizontale Ausrichtung des Bildes für die Schaltfläche Basisrichtung fest.
BackgroundImageVerticalAlignBase [▶ 100]	Legt die vertikale Ausrichtung des Bildes für die Schaltfläche Basisrichtung fest.
BackgroundImagePaddingBase [▶ 99]	Legt den Innenabstand des Bildes für die Schaltfläche Basisrichtung fest.

Kategorie: Size Configuration

Name	Beschreibung
Space [▶ 107]	Legt den Abstand zwischen den Elementen fest.
SpaceUnit [▶ 108]	Legt den Abstand zwischen den Elementen fest.
StateThickness [▶ 108]	Legt die Dicke des Statusanzeigebereichs um die Tasten fest.
StateThicknessUnit [▶ 108]	Legt die Einheit der Zustandsdicke fest.
HardwareButtonHeight [▶ 103]	Legt die Höhe der Schaltfläche für die Hardwareauswahl fest.
HardwareButtonHeightUnit [▶ 103]	Legt die Einheit für die Höhe der Hardware-Schaltflächen fest.

Kategorie: Colors

Name	Beschreibung
FaultColor [▶ 102]	Farbe des Leuchtrings, wenn der Status des FB_ManualOperation 'in fault' ist.
InPosColor [▶ 104]	Farbe des Leuchtrings, wenn der Status FB_ManualOperation 'in Position' ist.
MovingToPosColor [▶ 107]	Farbe des Leuchtrings, wenn der Status von FB_ManualOperation 'moving to position' ist.

Kategorie: Header Configuration

Name	Beschreibung
HeaderHeight [► 104]	Definition der Höhe der Kopfzeile.
HeaderHeightUnit [► 104]	Legt die Einheit für die Höhe der Kopfzeile fest.
HeaderFontSize [► 103]	Die Schriftgröße des Kopfzeilentextes.
HeaderFontSizeUnit [► 104]	Pixel oder Prozent für relative Größen.
HideHeaderText [► 104]	Blendet den Textbereich der Kopfzeile aus.

4.8.1.1 AxisName

Dieses Attribut legt den Achsenamen für das Control fest.

Schema: tchmi:general#/definitions/String

Attribut-Getter: getAxisName

Attribut-Setter: setAxisName

Verfügbar: seit Version 12.6.0

4.8.1.2 BackgroundImageBase

Pfad zu einem Bild, das als Hintergrundbild für eine Schaltfläche zum Bewegen der Achse in der Basisrichtung festgelegt werden soll.

Schema: tchmi:framework#/definitions/Path

Attribut-Getter: getBackgroundImageBase

Attribut-Setter: setBackgroundImageBase

Verfügbar: seit Version 12.6.0

4.8.1.3 BackgroundImageHeight Work

Definition der Höhe des Hintergrundbildes für eine Schaltfläche zum Verschieben der Achse in Arbeitsrichtung.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getBackgroundImageHeightWork

Attribut-Setter: setBackgroundImageHeightWork

Verfügbar: seit Version 12.6.0

4.8.1.4 BackgroundImageHeightBase

Definition der Höhe des Hintergrundbildes für eine Schaltfläche zum Verschieben der Achse in der Basisrichtung.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getBackgroundImageHeightBase

Attribut-Setter: setBackgroundImageHeightBase

Verfügbar: seit Version 12.6.0

4.8.1.5 BackgroundImageHeightBaseUnit

Legt die Höheneinheit des Hintergrundbildes für die Basisschaltfläche fest.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getBackgroundImageHeightBaseUnit

Attribut-Setter: setBackgroundImageHeightBaseUnit

Verfügbar: seit Version 12.6.0

4.8.1.6 BackgroundImageHeightWorkUnit

Legt die Höheneinheit des Hintergrundbildes für die Arbeitsschaltfläche fest.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getBackgroundImageHeightWorkUnit

Attribut-Setter: setBackgroundImageHeightWorkUnit

Verfügbar: seit Version 12.6.0

4.8.1.7 BackgroundImageHorizontalAlignBase

Definition der horizontalen Ausrichtung des Hintergrundbildes für eine Schaltfläche zum Verschieben der Achse in der Basisrichtung.

Schema: tchmi:framework#/definitions/HorizontalAlignment

Attribut-Getter: getBackgroundImageHorizontalAlgBase

Attribut-Setter: setBackgroundImageHorizontalAlgBase

Verfügbar: seit Version 12.6.0

4.8.1.8 BackgroundImageHorizontalAlignWork

Definition der horizontalen Ausrichtung des Hintergrundbildes für eine Schaltfläche zum Verschieben der Achse in Arbeitsrichtung.

Schema: tchmi:framework#/definitions/HorizontalAlignment

Attribut-Getter: getBackgroundImageHorizontalAlgWork

Attribut-Setter: setBackgroundImageHorizontalAlgWork

Verfügbar: seit Version 12.6.0

4.8.1.9 BackgroundImagePadding Base

Definition eines zusätzlichen Abstands zwischen Hintergrundbild und dem Rand der Schaltfläche, um die Achse für die vier Seiten in die Basisrichtung zu verschieben.

Schema: tchmi:framework#/definitions/Padding

Attribut-Getter: getBackgroundImagePaddingBase

Attribut-Setter: setBackgroundImagePaddingBase

Verfügbar: seit Version 12.6.0

4.8.1.10 BackgroundImagePadding Work

Definition eines zusätzlichen Abstands zwischen Hintergrundbild und dem Rand der Schaltfläche, um die Achse für die vier Seiten in die Arbeitsrichtung zu verschieben.

Schema: tchmi:framework#/definitions/Padding

Attribut-Getter: getBackgroundImagePaddingWork

Attribut-Setter: setBackgroundImagePaddingWork

Verfügbar: seit Version 12.6.0

4.8.1.11 BackgroundImageVerticalAlignBase

Definition der vertikalen Ausrichtung des Hintergrundbildes für eine Schaltfläche zum Verschieben der Achse in der Basisrichtung.

Schema: tchmi:framework#/definitions/VerticalAlignment

Attribut-Getter: getBackgroundImageVerticalAlgBase

Attribut-Setter: setBackgroundImageVerticalAlgBase

Verfügbar: seit Version 12.6.0

4.8.1.12 BackgroundImageVerticalAlignWork

Definition der vertikalen Ausrichtung des Hintergrundbildes für eine Schaltfläche zum Verschieben der Achse in Arbeitsrichtung.

Schema: tchmi:framework#/definitions/VerticalAlignment

Attribut-Getter: getBackgroundImageVerticalAlgWork

Attribut-Setter: setBackgroundImageVerticalAlgWork

Verfügbar: seit Version 12.6.0

4.8.1.13 BackgroundImageWidthBase

Definition der Breite des Hintergrundbildes für eine Schaltfläche zum Bewegen der Achse in der Basisrichtung.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getBackgroundImageWidthBase

Attribut-Setter: setBackgroundImageWidthBase

Verfügbar: seit Version 12.6.0

4.8.1.14 BackgroundImageWidthBaseUnit

Legt die Breitereinheit des Hintergrundbildes für die Basisschaltfläche fest.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getBackgroundImageWidthBaseUnit

Attribut-Setter: setBackgroundImageWidthBaseUnit

Verfügbar: seit Version 12.6.0

4.8.1.15 BackgroundImageWidthWork

Definition der Breite des Hintergrundbildes für eine Schaltfläche zum Bewegen der Achse in Arbeitsrichtung.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getBackgroundImageWidthWork

Attribut-Setter: setBackgroundImageWidthWork

Verfügbar: seit Version 12.6.0

4.8.1.16 BackgroundImageWidthWorkUnit

Legt die Breitereinheit des Hintergrundbildes für die Arbeitsschaltfläche fest.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getBackgroundImageWidthWorkUnit

Attribut-Setter: setBackgroundImageWidthWorkUnit

Verfügbar: seit Version 12.6.0

4.8.1.17 BackgroundImageWork

Pfad zu einem Bild, das als Hintergrundbild für eine Schaltfläche zum Bewegen der Achse in Arbeitsrichtung festgelegt werden soll.

Schema: tchmi:framework#/definitions/Path

Attribut-Getter: getBackgroundImageWork

Attribut-Setter: setBackgroundImageWork

Verfügbar: seit Version 12.6.0

4.8.1.18 BorderRadius

Dieses Attribut legt den Randradius der Schaltflächen auf dem Control fest.

Schema: tchmi:framework#/definitions/BorderRadius

Attribut-Getter: getBorderRadius

Attribut-Setter: setBorderRadius

Verfügbar: seit Version 12.6.0

4.8.1.19 BtnBaseText

Dadurch wird ein Text für die Schaltfläche zum Bewegen der Achse in der Basisrichtung festgelegt.

Schema: tchmi:general#/definitions/String

Attribut-Getter: getBtnBaseText

Attribut-Setter: setBtnBaseText

Verfügbar: seit Version 12.6.0

4.8.1.20 BtnWorkText

Dadurch wird ein Text für die Schaltfläche zum Bewegen der Achse in der Arbeitsrichtung festgelegt.

Schema: tchmi:general#/definitions/String

Attribut-Getter: getBtnWorkText

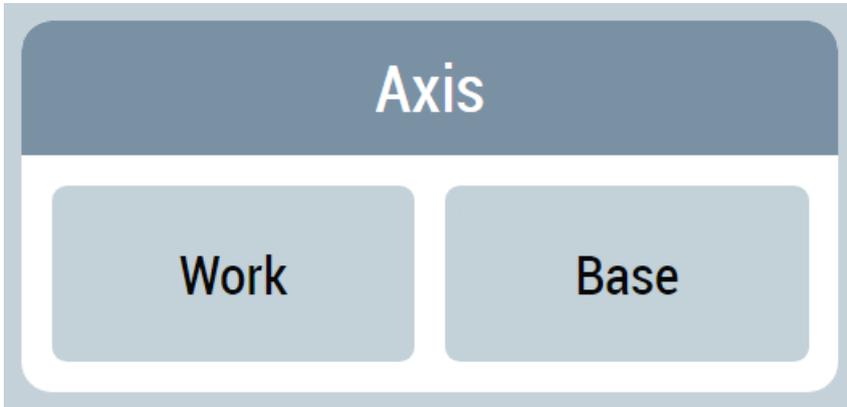
Attribut-Setter: setBtnWorkText

Verfügbar: seit Version 12.6.0

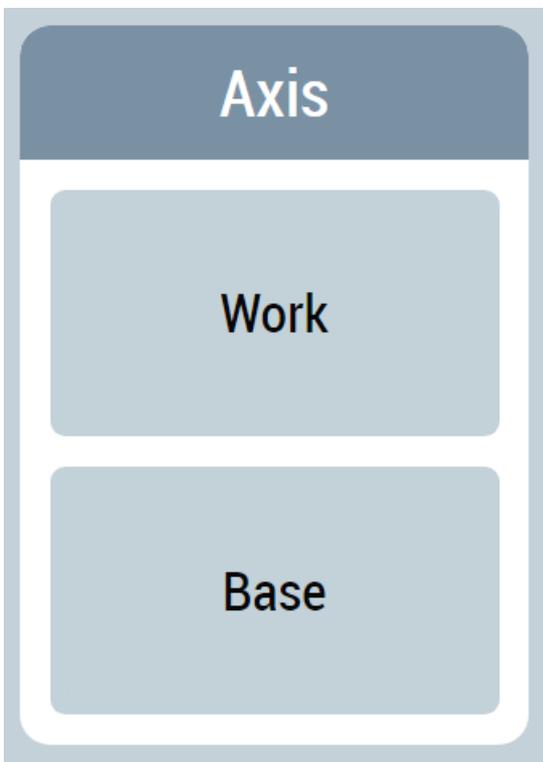
4.8.1.21 ButtonOrientation

Dieses Attribut legt die Ausrichtung der Schaltflächen fest, um sie nebeneinander (Vertical) oder übereinander (Horizontal) anzuzeigen.

Horizontale Ausrichtung der Schaltflächen:



Vertikale Ausrichtung der Schaltflächen:



Schema: tchmi:framework#/definitions/AxisBtnOrientation

Attribut-Getter: getAxisBtnOrientation

Attribut-Setter: setAxisBtnOrientation

Verfügbar: ab Version 12.6.0

4.8.1.22 FaultColor

Farbe des Leuchtrings um die Tasten, wenn der Status des konfigurierten Symbols FB_ManualOperation einen Fehler in der Achsenbewegung darstellt.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getFaultColor

Attribut-Setter: setFaultColor

Verfügbar: seit Version 12.6.0

4.8.1.23 HardwareButtonHeight

Legt die Höhe der Schaltfläche für die Hardwareauswahl fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHardwareButtonHeight

Attribut-Setter: setHardwareButtonHeight

Verfügbar: ab Version 12.12.0

4.8.1.24 HardwareButtonHeightUnit

Legt die Einheit für die Höhe der Hardwareschaltflächen fest.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHardwareButtonHeightUnit

Attribut-Setter: setHardwareButtonHeightUnit

Verfügbar: ab Version 12.12.0

4.8.1.25 HeaderBackgroundColor

Legt die Hintergrundfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderBackgroundColor

Attribut-Setter: setHeaderBackgroundColor

Verfügbar: ab Version 12.8.0

4.8.1.26 HeaderFontColor

Legt die Textfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderFontColor

Attribut-Setter: setHeaderFontColor

Verfügbar: ab Version 12.8.0

4.8.1.27 HeaderFontSize

Die Schriftgröße der Kopfzeile. Wenn Prozent als Einheit festgelegt ist, ist die Kopfzeilengröße relativ zur Schriftgröße des übergeordneten Elements.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderFontSize

Attribut-Setter: setHeaderFontSize

Verfügbar: ab Version 12.8.0

4.8.1.28 HeaderFontSizeUnit

Die Einheit der Schriftgröße der Kopfzeile (px, %).

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderFontSizeUnit

Attribut-Setter: setHeaderFontSizeUnit

Verfügbar: ab Version 12.8.0

4.8.1.29 HeaderHeight

Legt die Höhe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderHeight

Attribut-Setter: setHeaderHeight

Verfügbar: ab Version 12.8.0

4.8.1.30 HeaderHeightUnit

Legt die Einheit der Kopfzeilenhöhe fest:

- Pixel für eine absolute Größe
- Prozent für eine relative Größe

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderHeightUnit

Attribut-Setter: setHeaderHeightUnit

Verfügbar: ab Version 12.8.0

4.8.1.31 HideHeaderText

Bei der Einstellung `TRUE`, wird der Textbereich der Kopfzeile ausgeblendet.

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getHideHeaderText

Attribut-Setter: setHideHeaderText

Verfügbar: ab Version 12.12.0

4.8.1.32 InPosColor

Farbe des Leuchtrings um die Tasten, wenn der Status des konfigurierten Symbols `FB_ManualOperation` darstellt, dass die Achse die Bewegung in die gewünschte Richtung beendet hat.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getInPosColor

Attribut-Setter: setInPosColor

Verfügbar: seit Version 12.6.0

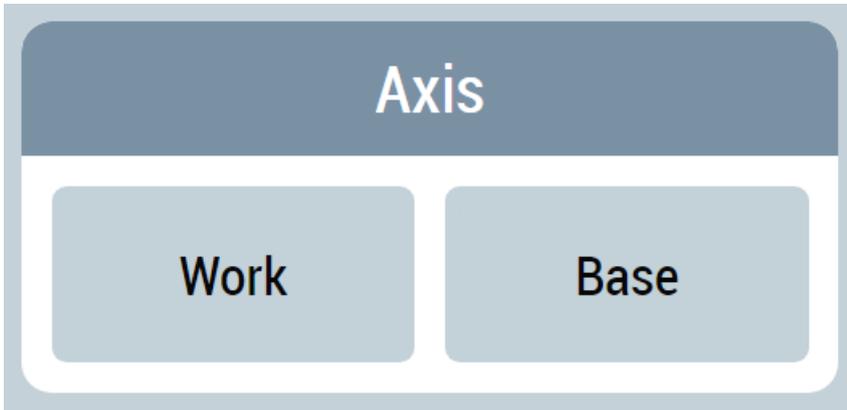
4.8.1.33 InvertDirection

Tauscht die Schaltfläche der Arbeits- und der Grundposition.

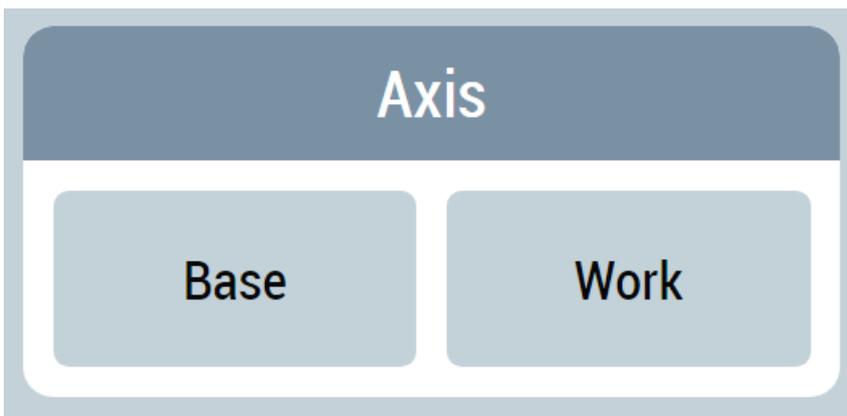
Standardmäßig befindet sich die Schaltfläche für den Richtungsbehehl "ToWork" je nach Ausrichtungsattribut links/oben.

Mit dem Attribut invert direction kann der Benutzer die Standardreihenfolge der organisierten Schaltflächen ändern.

Standardreihenfolge:



Reihenfolge, nachdem InvertDirection auf true gesetzt wurde:



Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getInvertDirection

Attribut-Setter: setInvertDirection

Verfügbar: ab Version 12.12.0

4.8.1.34 ManualOperationSymbol

Symbol von FB_ManualOperation, in das das Control die Kommandos schreiben und den Status lesen wird. Die Symbolbindung an diesem Attribut muss eine variable Struktur haben.

Die Struktur der obersten Ebene muss die nachfolgenden Eigenschaften aufweisen:

Name	Typ	Beschreibung
fbCmd	FB_CmdManualFunction	Funktionsbaustein oder Struktur, die Kommandovariablen vom Bediener speichert.
fbState	FB_StateManualFunction	Funktionsbaustein oder Struktur, die von dem Control gelesene Statusvariablen speichert.

Die Top-Level-Struktur kann die folgenden Eigenschaften haben, wenn ein Hardware-Rad in die manuelle Steuerung dieser Achse einbezogen werden soll:

Name	Typ	Beschreibung
DoSelect	BOOL	Das Control schreibt <code>TRUE</code> in diese Variable, wenn der Bediener die Taste zur Verwendung der Hardware-Steuerung drückt.
IsSelectable	BOOL	Wenn <code>TRUE</code> kann die Hardware-Steuerung über die Schaltfläche auf der HMI ausgewählt werden.
IsSelected	BOOL	Wenn <code>TRUE</code> wird dieses Control ausgewählt, um mit der Hardware-Steuerung bewegt zu werden.
SelectSupported	BOOL	Wenn <code>TRUE</code> unterstützt dieses Control eine Hardware-Steuerung und zeigt eine Schaltfläche zur Auswahl an.

Jeder `FB_CmdManualFunction`-Funktionsbaustein oder Strukturtyp muss die folgenden Eigenschaften haben:

Name	Typ	Beschreibung
ToBasePos	BOOL	Das Control schreibt <code>TRUE</code> in diese Variable, wenn der Bediener die Taste drückt, um sich in Basisrichtung zu bewegen. Und schreibt <code>FALSE</code> , sobald die Taste losgelassen wird.
ToWorkPos	BOOL	Das Control schreibt <code>TRUE</code> in diese Variable, wenn der Bediener die Taste drückt, um sich in Arbeitsrichtung zu bewegen. Und schreibt <code>FALSE</code> , sobald die Taste losgelassen wird.

Jeder `FB_StateManualFunction`-Funktionsbaustein oder Strukturtyp muss folgende Eigenschaften haben:

Name	Typ	Beschreibung
EnableBasePos	BOOL	Wenn <code>TRUE</code> , wird die Schaltfläche <code>ToBasePos</code> aktiviert, und der Bediener kann sie drücken.
EnableWorkPos	BOOL	Wenn <code>TRUE</code> , wird die Schaltfläche <code>ToWorkPos</code> aktiviert, und der Bediener kann sie drücken.
FaultBasePos	BOOL	Wenn <code>TRUE</code> , ist ein Fehler während der Bewegung in Basisrichtung aufgetreten.
FaultWorkPos	BOOL	Wenn <code>TRUE</code> , ist ein Fehler während der Bewegung in Arbeitsrichtung aufgetreten.
InBasePos	BOOL	Wenn <code>TRUE</code> , hat die Achse die Bewegung zur Basisposition beendet.
InWorkPos	BOOL	Wenn <code>TRUE</code> , hat die Achse die Bewegung zur Arbeitsposition beendet.

Name	Typ	Beschreibung
MovingToBasePos	BOOL	Wenn TRUE, bewegt sich die Achse jetzt in Richtung der Basisposition.
MovingToWorkPos	BOOL	Wenn TRUE, bewegt sich die Achse jetzt in Richtung der Arbeitsposition.

PLC1.MAIN.fbManuel	S	PLC1.FB_ManualFunctionHmi	<input checked="" type="checkbox"/>
fbCmd	S	PLC1.FB_CmdManualFunctionHmi	<input checked="" type="checkbox"/>
ToBasePos	S	PLC1.PROPERTY-GET-SET-BOOL	<input checked="" type="checkbox"/>
ToWorkPos	S	PLC1.PROPERTY-GET-SET-BOOL	<input checked="" type="checkbox"/>
fbState	S	PLC1.FB_StateManualFunctionHmi	<input checked="" type="checkbox"/>
EnableBasePos	S	PLC1.PROPERTY-GET-SET-BOOL	<input checked="" type="checkbox"/>
EnableWorkPos	S	PLC1.PROPERTY-GET-SET-BOOL	<input checked="" type="checkbox"/>
FaultBasePos	S	PLC1.PROPERTY-GET-SET-BOOL	<input checked="" type="checkbox"/>
FaultWorkPos	S	PLC1.PROPERTY-GET-SET-BOOL	<input checked="" type="checkbox"/>
InBasePos	S	PLC1.PROPERTY-GET-SET-BOOL	<input checked="" type="checkbox"/>
InWorkPos	S	PLC1.PROPERTY-GET-SET-BOOL	<input checked="" type="checkbox"/>
MovingToBasePos	S	PLC1.PROPERTY-GET-SET-BOOL	<input checked="" type="checkbox"/>
MovingToWorkPos	S	PLC1.PROPERTY-GET-SET-BOOL	<input checked="" type="checkbox"/>

Schema: tchmi:framework#/definitions/Symbol

Attribut-Getter: getFB_ManualOperation

Attribut-Setter: setFB_ManualOperation

Verfügbar: ab Version 12.6.0

4.8.1.35 MovingToPosColor

Farbe des Leuchtrings um die Tasten, wenn der Status des konfigurierten Symbols FB_ManualOperation anzeigt, dass sich die Achse in die gewünschte Richtung bewegt.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getMovingToPosColor

Attribut-Setter: setMovingToPosColor

Verfügbar: seit Version 12.6.0

4.8.1.36 SingleMode

Mit diesem Attribut kann das Control so konfiguriert werden, dass sie im Einzelmodus arbeitet. Im Einzelmodus steht dem Bediener nur die Taste in Arbeitsrichtung zur Verfügung. Dies kann zum Beispiel für ON/OFF-Operationen von der HMI aus genutzt werden.

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getSingleMode

Attribut-Setter: setSingleMode

Verfügbar: seit Version 12.6.0

4.8.1.37 Space

Legt den Abstand zwischen den Elementen fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getSpace

Attribut-Setter: setSpace

Verfügbar: ab Version 12.12.0

4.8.1.38 SpaceUnit

Legt die Einheit des Abstandes zwischen den Elementen fest.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getSpaceUnit

Attribut-Setter: setSpaceUnit

Verfügbar: ab Version 12.12.0

4.8.1.39 StateThickness

Legt die Dicke des Statusanzeigebereichs um die Schaltflächen fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getStateThickness

Attribut-Setter: setStateThickness

Verfügbar: ab Version 12.12.0

4.8.1.40 StateThicknessUnit

Legt die Einheit des Zustandsdickenwerts fest.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getSpaceUnit

Attribut-Setter: setSpaceUnit

Verfügbar: ab Version 12.12.0

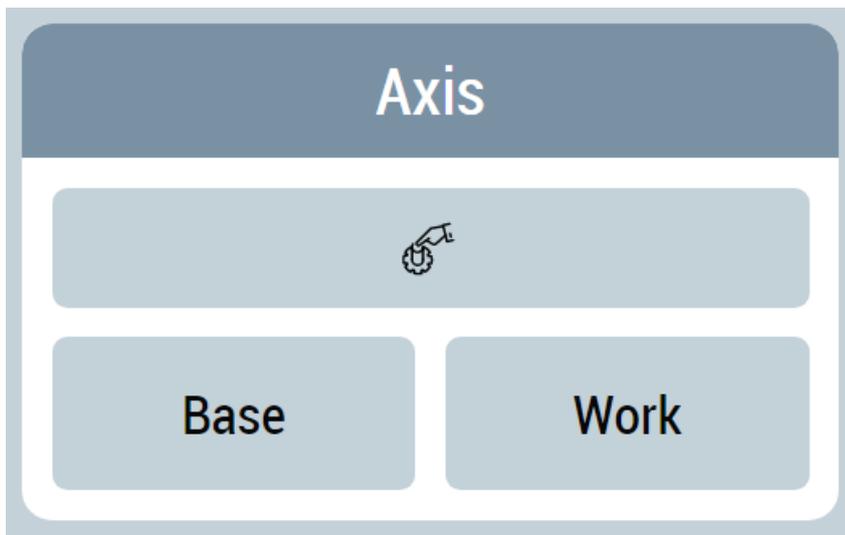
4.8.2 ManualOperation-Control mit Hardware-Encoder

Wenn das ManualOperation-Control mit einem Hardware-Setup verwendet wird, das einen einzelnen digitalen Encoder enthält, um mehrere Achsen manuell zu verfahren, hilft das Control bei der Verwaltung, welche Achse der Hardware-Encoder steuert.

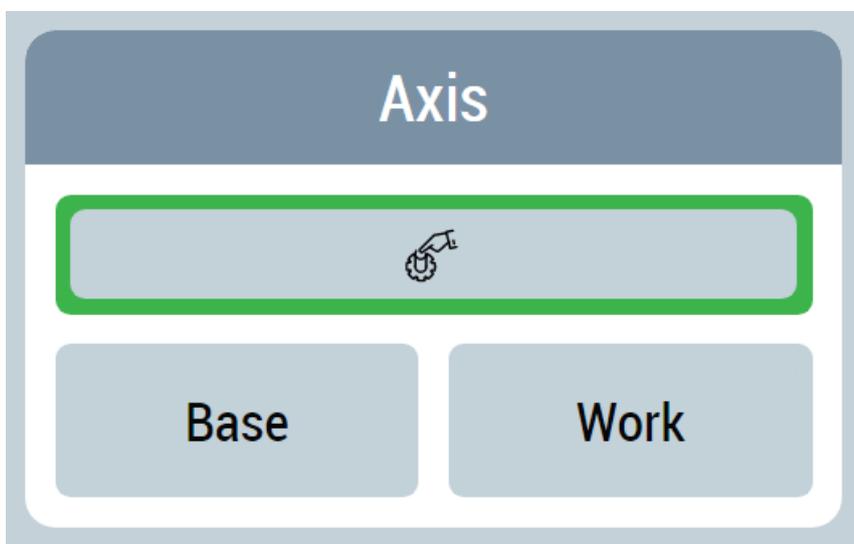
Wenn das ManualOperation-Control mit dem [FB_ManualFunctionHmi](#)-Funktionsbaustein verknüpft ist, stellt die SPS Eigenschaften zur Verfügung, die der Steuerung mitteilen, wann ein Hardware-Encoder für den Achsenbetrieb verfügbar ist.

Wenn ein Hardware-Encoder für den Tipbetrieb vorhanden ist, erscheint auf dem Control eine neue Schaltfläche "Hardware-Steuerung". Wenn der Benutzer auf die Schaltfläche für die Hardware-Steuerung klickt, kann er die ausgewählte Achse mithilfe des Hardware-Encoders bewegen.

Steuerung mit Hardware-Steuerschaltfläche:



Ausgewählte Achse nach einem Klick auf die Hardware-Schaltfläche:



Verfügbar: ab Version 12.12.0

4.9 MeasurementUnitSelector

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

Mit dem Control MeasurementUnitSelector kann der Benutzer auf die in der Datei [Unit_Config.json](#) [► 45] eingestellten Einheitenkonfiguration zugreifen und die Einstellungen der Anzeigeeinheit auf dem Client ändern.

GroupName	DisplayUnit	SystemUnit
Acceleration	<input checked="" type="checkbox"/> mm/s ²	mm/s ²
	<input type="checkbox"/> in/s ²	
	<input type="checkbox"/> %	
AngularAcceleration	<input checked="" type="checkbox"/> rad/s ²	rad/s ²
	<input type="checkbox"/> °/s ²	
	<input type="checkbox"/> %	
ElectricCurrent	<input checked="" type="checkbox"/> A	A
	<input type="checkbox"/> mA	
	<input type="checkbox"/> %	
ElectricPotential	<input checked="" type="checkbox"/> V	V
	<input type="checkbox"/> mV	
	<input type="checkbox"/> %	
ElectricResistance	<input checked="" type="checkbox"/> Ω	Ω
	<input type="checkbox"/> %	
	<input type="checkbox"/> %	
Force	<input checked="" type="checkbox"/> N	N
	<input type="checkbox"/> gf	
	<input type="checkbox"/> %	
Frequency	<input checked="" type="checkbox"/> Hz	Hz
	<input type="checkbox"/> %	
	<input type="checkbox"/> %	

- Sobald das Control aus der Toolbox gezogen und im Inhalt platziert wurde, werden alle im Konfigurator vorhandenen Einheitengruppen angezeigt
- Es sollte ein Control [Configurator](#) [► 37] zur Ansicht hinzugefügt werden, um dieses Control zu verwenden

4.9.1 Attribute

Kategorie: Common

Name	Beschreibung
GroupFilter [► 110]	Es ist ein Filter für alle Gruppennamen, der nur die Gruppennamen anzeigt, die in diesem Filter enthalten sind.
RowHeight [► 112]	Legt die Zeilenhöhe für das Control fest.
RowHeightUnit [► 112]	Legt die Einheit des Attributs Row Height [► 112] immer in Pixel fest.
TextFontSize [► 112]	Legt die Schriftgröße für das Control fest.
FontSizeUnit [► 110]	Legt die Schriftgrößeneinheit für das Control fest.

4.9.1.1 FontSizeUnit

Legt die Schriftgrößeneinheit für das Control fest.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getFontSizeUnit

Attribut-Setter: setFontSizeUnit

Verfügbar: seit Version 12.6.0

4.9.1.2 GroupFilter

Mit diesem Attribut können die auf dem Control aufgeführten Gruppen gefiltert werden.

Nach der Aktivierung werden nur noch die in dieser Liste aufgeführten Gruppen angezeigt. Wenn der Filter auf leer gesetzt oder kein Filter hinzugefügt wird, sind alle in der Datei [unit_config.json](#) [► 45] konfigurierten Einheitengruppen auf dem Control verfügbar.

Schema: tchmi:framework#/definitions/GroupFilterList

Attribut-Getter: getGroupFilter

Attribut-Setter: setGroupFilter

Verfügbar: seit Version 12.6.0

4.9.1.3 HeaderBackgroundColor

Legt die Hintergrundfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderBackgroundColor

Attribut-Setter: setHeaderBackgroundColor

Verfügbar: ab Version 12.8.0

4.9.1.4 HeaderFontColor

Legt die Textfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderFontColor

Attribut-Setter: setHeaderFontColor

Verfügbar: ab Version 12.8.0

4.9.1.5 HeaderHeight

Legt die Höhe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderHeight

Attribut-Setter: setHeaderHeight

Verfügbar: ab Version 12.8.0

4.9.1.6 HeaderHeightUnit

Legt die Einheit der Kopfzeilenhöhe fest:

- Pixel für eine absolute Größe
- Prozent für eine relative Größe

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderHeightUnit

Attribut-Setter: setHeaderHeightUnit

Verfügbar: ab Version 12.8.0

4.9.1.7 HeaderFontSize

Die Schriftgröße der Kopfzeile. Wenn Prozent als Einheit festgelegt ist, ist die Kopfzeilengröße relativ zur Schriftgröße des übergeordneten Elements.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderFontSize

Attribut-Setter: setHeaderFontSize

Verfügbar: ab Version 12.8.0

4.9.1.8 HeaderFontSizeUnit

Legt die Einheit der Kopfzeilenhöhe fest:

- Pixel für eine absolute Größe
- Prozent für eine relative Größe

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderHeightUnit

Attribut-Setter: setHeaderHeightUnit

Verfügbar: ab Version 12.8.0

4.9.1.9 RowHeight

Legt die Zeilenhöhe für das Control fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getRowHeight

Attribut-Setter: setRowHeight

Verfügbar: seit Version 12.6.0

4.9.1.10 RowHeightUnit

Legt die Einheit für das Attribut [Row Height \[►_112\]](#) immer in Pixel fest.

Schema: tchmi:framework#/definitions/PixelUnit

Attribut-Getter: getRowHeightUnit

Attribut-Setter: setRowHeightUnit

Verfügbar: seit Version 12.6.0

4.9.1.11 TextFontSize

Legt die Schriftgröße für das Control fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getFontSize

Attribut-Setter: setFontSize

Verfügbar: seit Version 12.6.0

4.9.2 Benutzerinteraktionen

Name	Beschreibung	Ergebnis im Control
Einmaliges Antippen	Mit einem Fingertipp kann der Benutzer eine Einheit innerhalb einer Einheitengruppe auswählen, die als Anzeigeeinheit für diese Gruppe dient.	

Verfügbar: seit Version 12.6.0

4.10 Monitoring

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

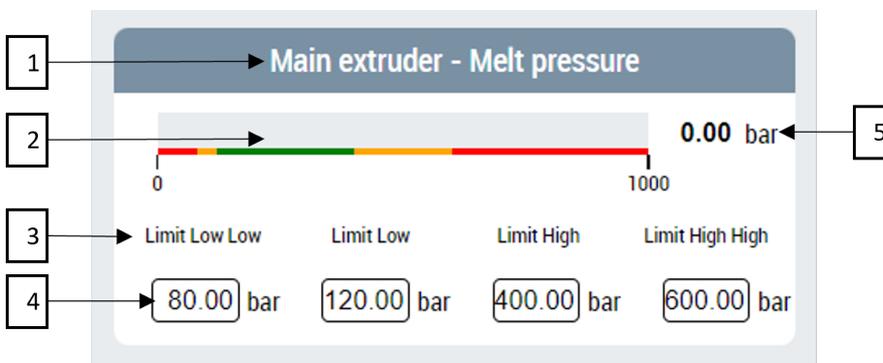
Verfügbar: ab Version 12.8.0

Das Monitoring-Control wird verwendet, um mehrere FB_Monitoring-Objekte in der SPS von einem einzigen Framework-Control aus zu visualisieren und zu konfigurieren.

Das Bild zeigt, wie sieben verschiedene Instanzen von FB_Monitoring auf diesem Control visualisiert werden.



Jedes einzelne Monitoring-Objekt wird mit einem eigenen Element dargestellt.



Komponenten der einzelnen Monitoring-Elemente:

1. Kopfzeilentext oder Beschriftung
2. Linearanzeige, die den Ist-Wert im Vergleich zu farbigen Bereichen anzeigt, die verschiedene Grenzwerte hervorheben.
3. Beschreibender Text für jedes Eingabefeld.

4. Felder, die es dem Benutzer ermöglichen, Limit Low Low, Limit Low, Limit High und Limit High High einzustellen.
5. Überwacher Wert in Ziffern.

Verfügbar: ab Version 12.8.0

4.10.1 Attribute

Kategorie: Monitoring

Name	Beschreibung
Konfiguration [▶ 116]	Mit diesem Attribut werden mehrere auf dem Control angezeigte Überwachungselemente konfiguriert.
Orientation [▶ 118]	Ausrichtung (vertikal/horizontal) der angeordneten Überwachungselemente.
Wrap [▶ 120]	Wählen Sie aus, ob neue Elemente in neue Zeilen/Spalten umgebrochen werden sollen.

Kategorie: Spacing Gap

Name	Beschreibung
RowGap [▶ 119]	Der Zeilenabstand ist der vertikale Abstand zwischen zwei Monitoring-Elementen.
RowGapUnit [▶ 120]	Einheit für das Attribut Zeilenabstand.
ColumnGap [▶ 115]	Der Spaltenabstand ist der horizontale Abstand zwischen zwei Monitoring-Elementen.
ColumnGapUnit [▶ 115]	Einheit für das Attribut Spaltenabstand.

Kategorie: User Change Logging

Name	Beschreibung
LogUserChanges [▶ 117]	Loggen jeder Änderung durch den Benutzer auf dem HMI-Bildschirm in einer Logdatei.

4.10.1.1 ColumnGap

Der Spaltenabstand ist der horizontale Abstand zwischen zwei Monitoring-Elementen.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getColumnGap

Attribut-Setter: setColumnGap

Verfügbar: ab Version 12.8.0

4.10.1.2 ColumnGapUnit

Einheit für das Attribut Spaltenabstand.

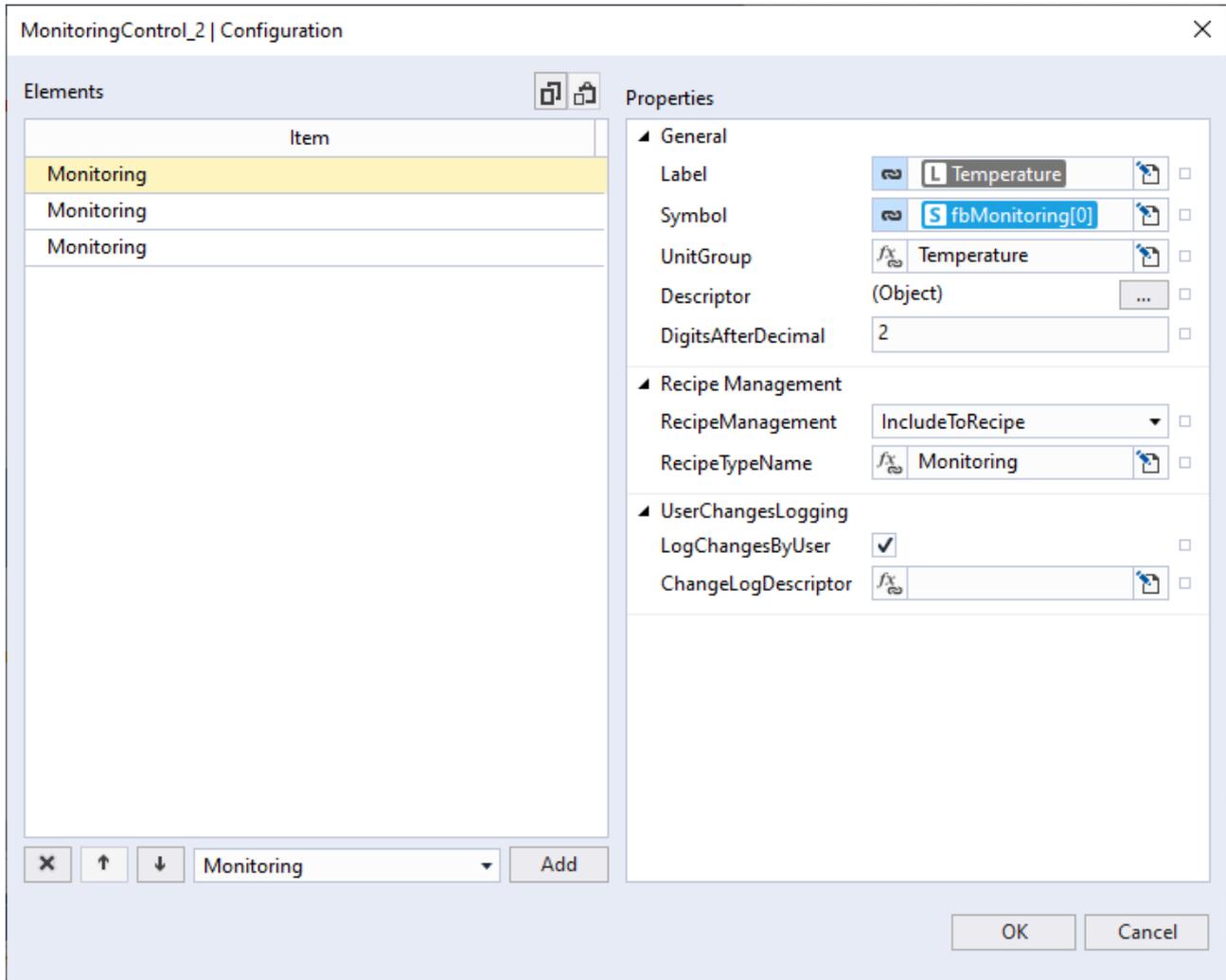
Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getColumnGapUnit

Attribut-Setter: setColumnGapUnit

Verfügbar: ab Version 12.8.0

4.10.1.3 Konfiguration



Das Attribut Konfiguration ist ein Array von Objekten, und jede Eigenschaft in diesen Objekten wird im Folgenden erläutert:

Name	Typ	Beschreibung
Label	string	Dies ist der Text, der im Kopfbereich des jeweiligen Monitoring-Elements angezeigt wird.
Symbol	tchmi:framework#/definitions/Symbol	Das Symbol der verknüpften FB_Monitoring-Funktionsbausteinvariable.
UnitGroup	string	Einheitengruppe, die für die Einheitenumrechnung aller numerischen Werte verwendet wird.
RecipeManagement	tchmi:framework#/definitions/RecipeType	Der Benutzer kann auswählen, ob Grenzwerte zu einem Rezepttyp durch das Monitoring-Control hinzugefügt werden sollen.
RecipeTypeName	string	Name des Rezepttyps, zu dem die Symbole von dem Monitoring-Control hinzugefügt werden sollen.

Name	Typ	Beschreibung
Descriptor	tchmi:framework#/definitions/Descriptor	Mit diesem Objekt kann der Benutzer Textwerte festlegen, die als Deskriptortext für verschiedene Grenzwerte im Control angezeigt werden sollen. Wenn dieses Objekt null ist, werden Standardwerte als Deskriptoren festgelegt.
DigitsAfterDecimal	number	Anzahl der angezeigten Nachkommastellen. Verfügbar: ab Version 12.14.0
LogUserChanges	tchmi:general#/definitions/Boolean	Jede Änderung der Grenzwerte wird in einer Logdatei gespeichert. Verfügbar: ab Version 12.12.0
ChangeLogDescriptor	string	Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird. Verfügbar: ab Version 12.14.0

Jedes FB Monitoring-Objekt in der SPS muss Eigenschaften haben, die im Folgenden erläutert werden:

Name	Typ	Beschreibung
ActualValue	Float	Überwacher Istwert.
LimitHH	Float	Obere kritische Grenze.
LimitH	Float	Obere Warngrenze.
LimitLL	Float	Untere kritische Grenze.
LimitL	Float	Untere Warngrenze.
ScalingValueMin	Float	Minimalwert, der als untere Grenze der Linearanzeige dient.
ScalingValueMax	Float	Maximalwert, der als Obergrenze der Linearanzeige dient.

Schema: tchmi:framework#/definitions/MonitoringConfig

Attribut-Getter: getConfiguration

Attribut-Setter: setConfiguration

Verfügbar: ab Version 12.8.0

4.10.1.4 LogUserChanges

Wenn dieses Attribut auf `TRUE` gesetzt ist, loggt das Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket jede Änderung durch den Benutzer auf dem HMI-Bildschirm in einer Logdatei.

Um dieses Attribut zu verwenden, muss UserLoggerSymbol [► 45] mit einem Binding an FB_UserLoggingHmi konfiguriert werden.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getLogUserChanges

Attribut-Setter: setLogUserChanges

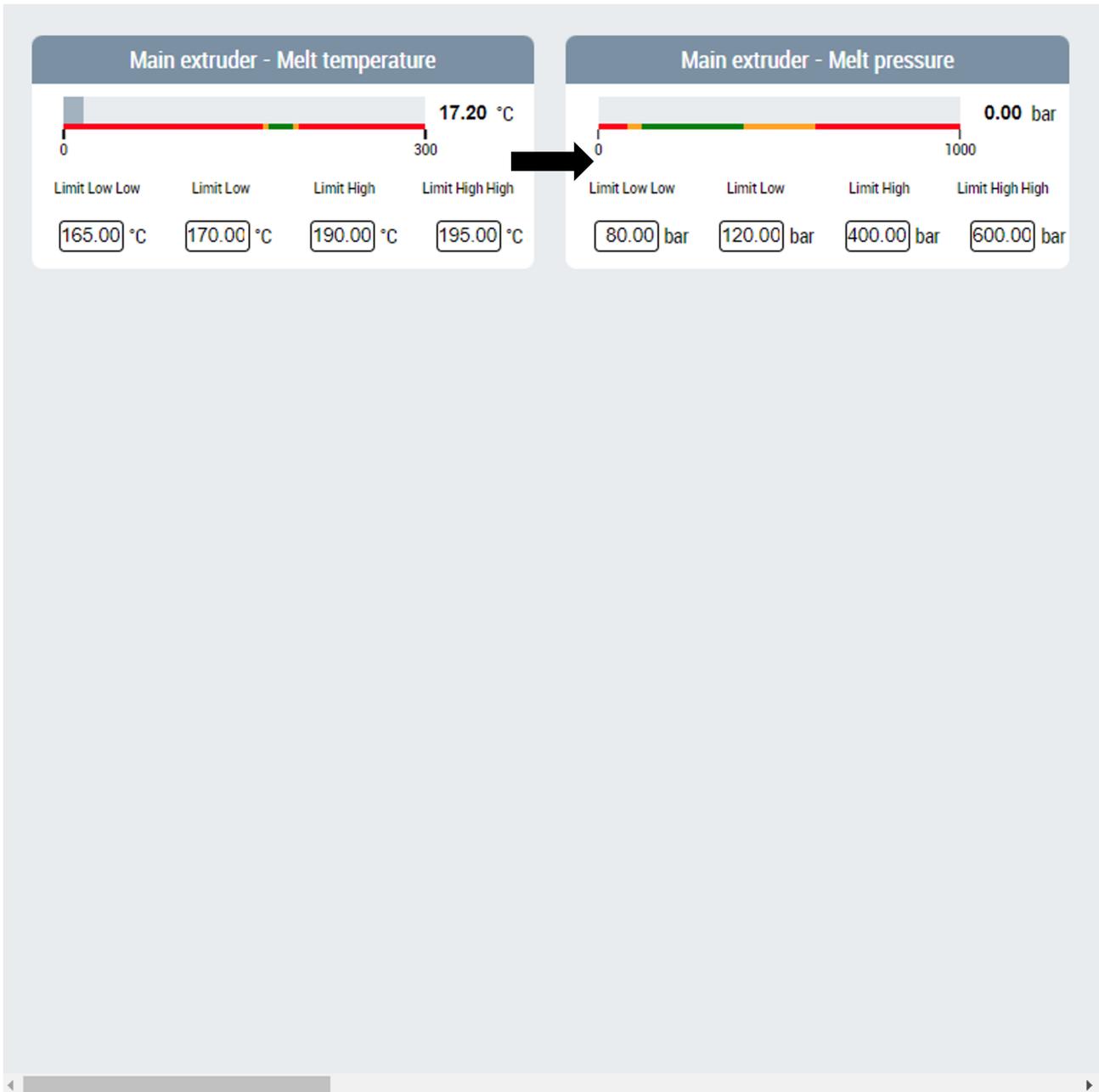
Verfügbar: ab Version 12.12.0

4.10.1.5 Orientation

Mit dem Monitoring-Control kann der Benutzer mehrere Monitoring-Objekte in einem Control visualisieren. Mit dem Attribut Orientation kann die Richtung ausgewählt werden, in der diese mehrfachen Monitoring-Elemente auf dem Bildschirm angeordnet werden sollen.

Zwei mögliche Werte für dieses Attribut sind:

Horizontal: Bei der horizontalen Ausrichtung werden alle Elemente horizontal nebeneinander angeordnet.



Vertikal: Bei vertikaler Ausrichtung werden die Elemente vertikal zueinander angeordnet.



Schema: tchmi:framework#/definitions/Orientation

Attribut-Getter: getOrientation

Attribut-Setter: setOrientation

Verfügbar: ab Version 12.8.0

4.10.1.6 RowGap

Der Zeilenabstand ist der vertikale Abstand zwischen zwei Monitoring-Elementen.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getRowGap

Attribut-Setter: setRowGap

Verfügbar: ab Version 12.8.0

4.10.1.7 RowGapUnit

Einheit für das Attribut Zeilenabstand.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getRowGapUnit

Attribut-Setter: setRowGapUnit

Verfügbar: ab Version 12.8.0

4.10.1.8 Wrap

Mit dem Wrap-Attribut kann der Benutzer wählen, ob die auf dem Control organisierten Elemente umgebrochen werden sollen.

Bei horizontaler Ausrichtung werden neue Elemente in die nächsten Zeilen umgebrochen.



Bei vertikaler Ausrichtung werden neue Elemente in der nächsten Spalte eingefügt.



Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getWrap

Attribut-Setter: setWrap

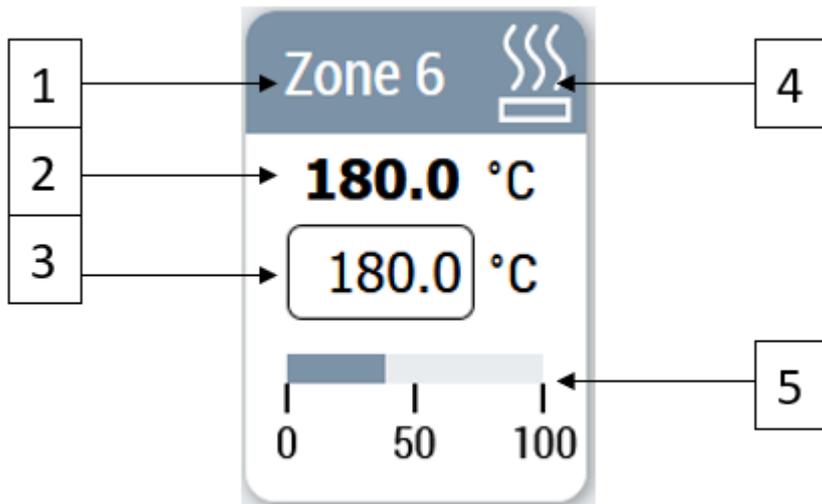
Verfügbar: ab Version 12.8.0

4.11 PfwSingleTempControl

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

PfwSingleTempControl ermöglicht die einfache Anzeige und Steuerung einer Temperaturzone in TwinCAT HMI.



- 1. Anzeige des Zonennamens
- 2. Ist-Temperaturwert der Zone
- 3. Soll-Temperaturwert der Zone
- 4. Statussymbol, das den Kühl-, Heiz- oder Fehlerstatus anzeigen kann
- 5. Leistungsstufe der Zone

4.11.1 Attribute

Kategorie: Colors

Name	Beschreibung
TopBarBackgroundColor [▶ 126]	Hintergrundfarbe der oberen Leiste.

Kategorie: Common

Name	Beschreibung
fbTempZone [▶ 123]	Symbol für FB_TempZone.
Minimode [▶ 125]	Wenn TRUE, wird das Control im Mini-Modus angezeigt.
ZoneName-FontSize [▶ 126]	Schriftgröße des Zonennamens, der in der oberen Leiste des Controls angezeigt wird.
ZoneName-FontSizeUnit [▶ 126]	Einheit der Schriftgröße für den Text des Zonennamens.
ZoneName [▶ 126]	Text, der als Zonenname angezeigt werden soll.
DigitsAfterDecimal [▶ 123]	Anzahl der Ziffern, die nach dem Dezimalpunkt für Float-Werte angezeigt werden.

Recipe Management:

Name	Beschreibung
RecipeManagement [▶ 125]	Legen Sie fest, ob das Control Symbole, die ihm zugeordnet sind, zu einem Rezepttyp hinzufügt.
RecipeTypeName [▶ 125]	Mit diesem Attribut kann der Benutzer den Namen des Rezepttyps festlegen, zu dem das Control die ihm zugeordneten Symbole hinzufügt und alle Änderungen einlernt.

User Changes Logging:

Name	Beschreibung
LogChangesByUser [► 125]	Logging der vom Benutzer auf dem Client vorgenommenen Änderungen in einer Logdatei.
ChangeLogDescriptor [► 123]	Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

4.11.1.1 ChangeLogDescriptor

Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

Schema: "tchmi:general#/definitions/STRING"

Attribut-Getter: getChangeLogDescriptor

Attribut-Setter: setChangeLogDescriptor

Verfügbar: ab Version 12.14.0

4.11.1.2 DigitsAfterDecimal

Mit diesem Attribut kann die Auflösung der einzelnen Zahlenwerte, die auf dem Control angezeigt werden, aktualisiert werden.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getDigitsAfterDecimal

Attribut-Setter: setDigitsAfterDecimal

Verfügbar: ab Version 12.8.0

4.11.1.3 fbTempZone

Symbole vom Typ FB_TempZone sollten mit diesem Attribut verknüpft werden. Das Control löst die einzelnen Variablenpfade innerhalb des Controls auf und holt die erforderlichen Daten.

Die Struktur von FB_TempZone:

Name	Typ	Beschreibung
ActualTemperature	Number	Ist-Temperaturwert.
SetpointTemperature	Number	Soll-Temperaturwert.
Error	Bool	Wenn die Zone fehlerhaft ist.
Heating	Bool	Wenn die Zone heizt.
Cooling	Bool	Wenn die Zone kühlt.
PowerLevel	Number	Ausgangsleistungspegel der Zone.

Schema: tchmi:framework#/definitions/Symbol

Attribut-Setter: setfbTempZone

Attribut-Getter: getfbTempZone

Verfügbar: seit Version 12.6.0

4.11.1.4 HeaderBackgroundColor

Legt die Hintergrundfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderBackgroundColor

Attribut-Setter: setBackgroundColor

Verfügbar: ab Version 12.8.0

4.11.1.5 HeaderFontColor

Legt die Textfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderFontColor

Attribut-Setter: setHeaderFontColor

Verfügbar: ab Version 12.8.0

4.11.1.6 HeaderFontSize

Die Schriftgröße der Kopfzeile. Wenn Prozent als Einheit festgelegt ist, ist die Kopfzeilengröße relativ zur Schriftgröße des übergeordneten Elements.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderFontSize

Attribut-Setter: setHeaderFontSize

Verfügbar: ab Version 12.8.0

4.11.1.7 HeaderFontSizeUnit

Legt die Einheit der Kopfzeilenhöhe fest:

- Pixel für eine absolute Größe
- Prozent für eine relative Größe

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderHeightUnit

Attribut-Setter: setHeaderHeightUnit

Verfügbar: ab Version 12.8.0

4.11.1.8 HeaderHeight

Legt die Höhe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderHeight

Attribut-Setter: setHeaderHeight

Verfügbar: ab Version 12.8.0

4.11.1.9 HeaderHeightUnit

Legt die Einheit der Kopfzeilenhöhe fest:

- Pixel für eine absolute Größe
- Prozent für eine relative Größe

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderHeightUnit

Attribut-Setter: setHeaderHeightUnit

Verfügbar: ab Version 12.8.0

4.11.1.10 LogUserChanges

Wenn dieses Attribut auf `TRUE` gesetzt ist, loggt das Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket jede Änderung durch den Benutzer auf dem HMI-Bildschirm in einer Logdatei.

Um dieses Attribut zu verwenden, muss [UserLoggerSymbol](#) [► 45] mit einem Binding an [FB_UserLoggingHmi](#) konfiguriert werden.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getLogUserChanges

Attribut-Setter: setLogUserChanges

Verfügbar: ab Version 12.12.0

4.11.1.11 MiniMode

Mit dem Attribut Mini Mode kann der Benutzer festlegen, ob das Control in einem minimierten Modus angezeigt werden soll, der für einen kleineren Bildschirm besser geeignet ist.

Im Mini Mode wird der Fortschrittsbalken für die Leistungsstufe nicht angezeigt.

Schema: tchmi:general#/definitions/Bool

Attribut-Setter: setMiniMode

Attribut-Getter: getMiniMode

Verfügbar: seit Version 12.6.0

4.11.1.12 RecipeManagement

Legen Sie fest, ob das Control die Symbole, die ihm zugeordnet sind, zu einem Rezepttyp hinzufügen soll.

Wenn das Projekt das [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper](#) [► 198]-Paket enthält, fügt das Control die Serversymbole, die darauf abgebildet sind, dem Rezepttyp hinzu, der durch das Attribut [RecipeTypeName](#) [► 125] definiert ist.

Nach jedem Benutzerwechsel lernt das Paket RecipeHelper das aktuell aktive Rezept ein.

Schema: "tchmi:framework#/definitions/TcHmi.Controls.Plastic.RecipeManagement"

Attribut-Getter: getRecipeManagement

Attribut-Setter: setRecipeManagement

Verfügbar: ab Version 12.14.0

4.11.1.13 RecipeTypeName

Mit diesem Attribut kann der Benutzer den Namen des Rezepttyps festlegen, zu dem das Control die Symbole hinzufügt, die ihm zugeordnet sind, und alle Änderungen einlernen.

Damit das Control Symbole zum Rezepttyp hinzufügen kann, muss das Projekt das Paket [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper](#) [► 198] enthalten und das Attribut [RecipeManagement](#) [► 125] muss auf `IncludeToRecipe` gesetzt sein.



Wenn ein Symbol bereits Teil eines Rezepttyps ist, hat diese Eigenschaft keine Auswirkungen auf die Rezepte. Nur für das Hinzufügen von Symbolen, die nicht Teil eines Rezepttyps sind, wird das Attribut `RecipeTypeName` berücksichtigt.

Schema: "tchmi:general#/definitions/STRING"

Attribut-Getter: `getRecipeTypeName`

Attribut-Setter: `setRecipeTypeName`

Verfügbar: ab Version 12.14.0

4.11.1.14 TopBarBackgroundColor

Hintergrundfarbe des oberen Balkenbereichs des Controls.

Schema: `tchmi:framework#/definitions/Color`

Attribut-Setter: `setTopBar_BackgroundColour`

Attribut-Getter: `getTopBar_BackgroundColour`

Verfügbar: seit Version 12.6.0

4.11.1.15 ZoneName

Text, der als Zonenname im Control angezeigt werden soll.

Schema: `tchmi:general#/definitions/string`

Attribut-Setter: `setZoneName`

Attribut-Getter: `getZoneName`

Verfügbar: seit Version 12.6.0

4.11.1.16 ZoneName-FontSize

Schriftgröße des Zonennamen-Textes, der im Control angezeigt wird.

Schema: `tchmi:framework#/definitions/MeasurementValue`

Attribut-Setter: `setZoneNameFontSize`

Attribut-Getter: `getZoneNameFontSize`

Verfügbar: seit Version 12.6.0

4.11.1.17 ZoneName-FontSizeUnit

Einheit der Schriftgröße, die über das Attribut `ZoneName-FontSize` [► 126] festgelegt wird. 'px' oder '%' sind möglich.

Schema: `tchmi:framework#/definitions/ MeasurementUnit`

Attribut-Setter: `setZoneNameFontSizeUnit`

Attribut-Getter: `getZoneNameFontSizeUnit`

Verfügbar: seit Version 12.6.0

4.12 ProcessScheduler

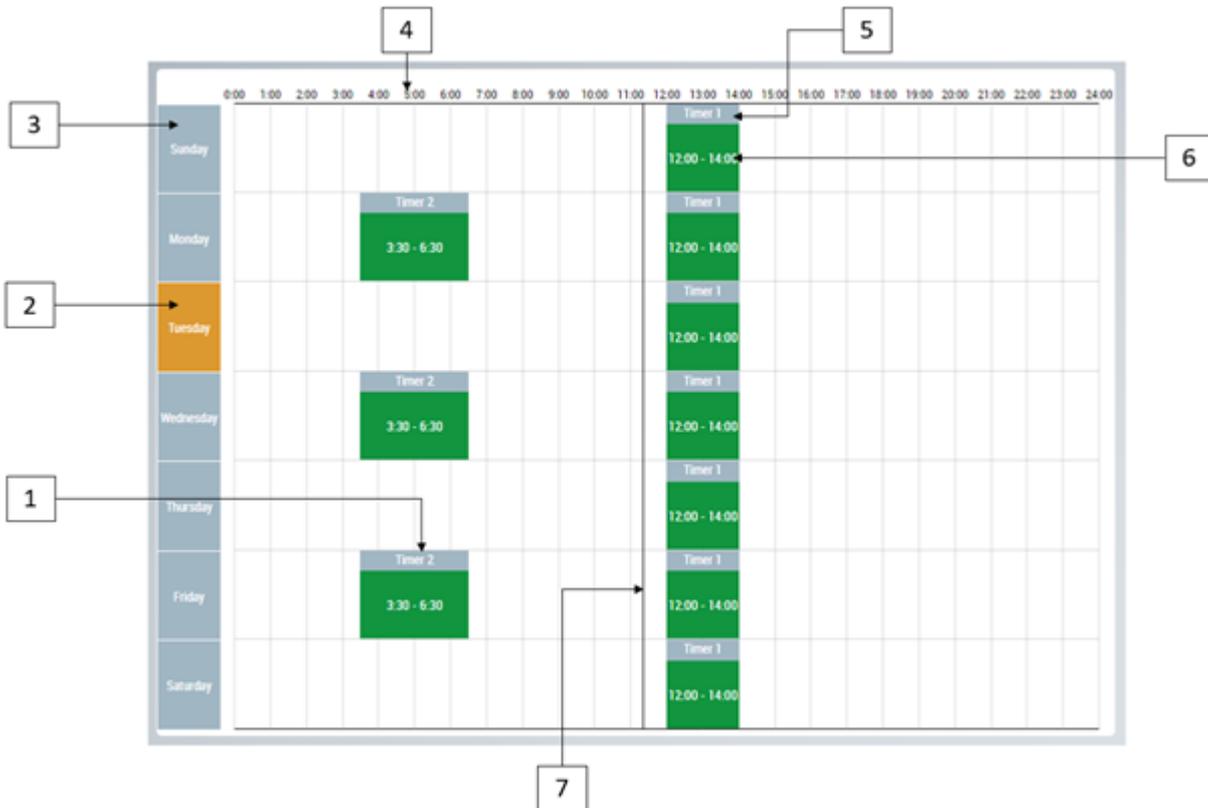
NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

Das Control ProcessScheduler ermöglicht dem Benutzer die Visualisierung und Bearbeitung von Wochenplänen, die von der SPS zur Ausführung beliebiger Prozesse verwendet werden können.

Aktion aktivieren

i Alle neuen Änderungen am Control müssen über die Aktion ActivateConfig [▶ 130] aktiviert werden, um neue Werte an die SPS zu senden.



- 1. Einzelnes Zeitplanobjekt
- 2. Aktueller Wochentag
- 3. Wochentage
- 4. Uhrzeit
- 5. Titel des Zeitplans
- 6. Zeitliche Dauer des Zeitplans
- 7. Aktuelle Zeit

4.12.1 Attribute

Kategorie: Scheduler

Name	Beschreibung
TimerList [▶ 129]	Symbol des Arrays mit dem Typ FB_TimerWeekdayHmi.
TimerLabels [▶ 129]	Array von Bezeichnungen für jedes Zeitplanobjekt.
SelectedTimerIndex [▶ 128]	Index des ausgewählten Zeitplans aus dem Array TimerList [▶ 129].

Kategorie: Common

Name	Beschreibung
ActivationRequired [► 128]	Wird auf <code>TRUE</code> gesetzt, wenn der Benutzer neue Änderungen vornimmt, die nicht aktiviert sind.

User Changes Logging:

Name	Beschreibung
LogChangesByUser [► 128]	Logging der vom Benutzer auf dem Client vorgenommenen Änderungen in einer Logdatei.
ChangeLogDescriptor [► 128]	Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird

4.12.1.1 ActivationRequired

Dieses schreibgeschützte Attribut wird auf `TRUE` gesetzt, wenn ein Benutzer neue Änderungen vornimmt, die nicht aktiviert sind.

Dieser Wert kann von anderen Controls, wie z. B. einem `TchmiButton`, verwendet werden, um sein Aussehen zu ändern und den Benutzer darüber zu informieren, dass neue Änderungen aktiviert werden müssen.

Schema: `tchmi:general#/definitions/Boolean`

Attribut-Getter: `getActivationRequired`

Verfügbar: seit Version 12.6.0

4.12.1.2 ChangeLogDescriptor

Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

Schema: `"tchmi:general#/definitions/STRING"`

Attribut-Getter: `getChangeLogDescriptor`

Attribut-Setter: `setChangeLogDescriptor`

Verfügbar: ab Version 12.14.0

4.12.1.3 LogUserChanges

Wenn dieses Attribut auf `TRUE` gesetzt ist, loggt das `Beckhoff.TwinCAT.HMI.Plastic.Controls`-Paket jede Änderung durch den Benutzer auf dem HMI-Bildschirm in einer Logdatei.

Um dieses Attribut zu verwenden, muss [UserLoggerSymbol](#) [► 45] mit einem Binding an [FB_UserLoggingHmi](#) konfiguriert werden.

Schema: `"tchmi:general#/definitions/BOOL"`

Attribut-Getter: `getLogUserChanges`

Attribut-Setter: `setLogUserChanges`

Verfügbar: ab Version 12.12.0

4.12.1.4 SelectedTimerIndex

Dieses Attribut gibt den Array-Index des Zeitplans wieder, den der Benutzer auf dem Bildschirm ausgewählt hat. Durch Auswahl eines Zeitplanobjekts kann der Benutzer diesen Zeitplan bearbeiten oder aus dem Array entfernen.

Schema: `tchmi:general#/definitions/Number`

Attribut-Getter: getSelectedTimerIndex

Attribut-Setter: setSelectedTimerIndex

Verfügbar: seit Version 12.6.0

4.12.1.5 TimerLabels

Satz von TimerLabels, der ein Array von Bezeichnungen ist. Der Benutzer kann einen einfachen Text eingeben oder Lokalisierungssymbole verwenden, um den Titel für die Zeitpläne festzulegen.

Schema: tchmi:framework#/definitions/TimerLabelList

Attribut-Getter: getTimerLabels

Attribut-Setter: setTimerLabels

Verfügbar: seit Version 12.6.0

4.12.1.6 TimerList

Dieses Attribut sollte mit einem Array-Symbol auf der SPS verbunden werden, wobei jedes Element die Konfiguration eines Zeitplans speichert.

Eigenschaften der einzelnen Elemente in diesem Array:

Name	Typ	Beschreibung
Startzeit	LREAL	Startzeit in Sekunden.
EndTimer	LREAL	Endzeit in Sekunden.
Weekday	BYTE	Ein BYTE, das zur Auswahl der Wochentage für den Zeitplan verwendet wird. Das 1. Bit (LSB) ist gesetzt, um anzuzeigen, dass der Zeitplan am Montag verwendet wird. In ähnlicher Weise ist das 7. Bit gesetzt, um anzuzeigen, dass der Zeitplan am Sonntag verwendet wird. Ein Zeitplan, der am Montag, Dienstag und Sonntag verwendet wird, hat zum Beispiel den Wert "2#0100 0011".

Schema: tchmi:general#/definitions/Symbol

Attribut-Getter: getTimerList

Attribut-Setter: setTimerList

Verfügbar: seit Version 12.6.0

4.12.2 Funktionen

Kategorie: Actions

Name	Beschreibung
ActivateConfig [▶ 130]	Dieses Ereignis wird generiert, wenn eine neue Konfiguration auf dem Control aktiviert wird.

Name	Beschreibung
ResetConfigdata [► 130]	Mit dieser Funktion werden alle Benutzeränderungen rückgängig gemacht und die zuletzt aktivierte Konfiguration aus der SPS neu geladen.
AddScheduleData [► 130]	Diese Funktion mit Zugriffsrecht zeigt ein Pop-up-Fenster auf dem Control an, das die Erstellung eines neuen Zeitplans ermöglicht.
EditScheduleData [► 131]	Diese Funktion mit Zugriffsrecht zeigt ein Pop-up-Fenster auf dem Control an, das die Bearbeitung des ausgewählten Zeitplans ermöglicht.

4.12.2.1 ActivateConfig

Durch den Aufruf dieser Funktion werden alle Benutzeränderungen aktiviert und in die SPS-Symbole geschrieben.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.12.2.2 ResetConfigdata

Mit dieser Funktion werden alle Änderungen des Benutzers rückgängig gemacht, indem die zuletzt aktivierte Konfiguration erneut geladen wird.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.12.2.3 AddScheduleData

Mit dieser Funktion wird ein Pop-up-Fenster auf dem Control angezeigt, das die Erstellung eines neuen Zeitplans ermöglicht.

Der Benutzer benötigt das Zugriffsrecht **Operate**, um diese Funktion aufzurufen.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.12.2.4 EditScheduleData

Mit dieser Funktion wird ein Pop-up-Fenster auf dem Control angezeigt, das die Bearbeitung des ausgewählten Zeitplans ermöglicht. Der Benutzer muss zunächst einen bestehenden Zeitplan auf dem Control auswählen und dann diese Funktion aufrufen.

Zur Ausführung dieser Funktion benötigt der Benutzer den Zugriff **Operate**.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: seit Version 12.6.0

4.12.3 Events**Kategorie: Actions**

Name	Beschreibung
onUserConfigActivated [► 131]	Dieses Ereignis wird generiert, wenn eine neue Konfiguration auf dem Control aktiviert wird.
onEditingCanceled [► 131]	Dieses Ereignis wird generiert, wenn Benutzeränderungen rückgängig gemacht werden und die alte Konfiguration wieder auf das Control geladen wird.

4.12.3.1 onUserConfigActivated

Dieses Ereignis wird generiert, wenn eine neue Konfiguration auf dem Control aktiviert wird.

Verfügbar: seit Version 12.6.0

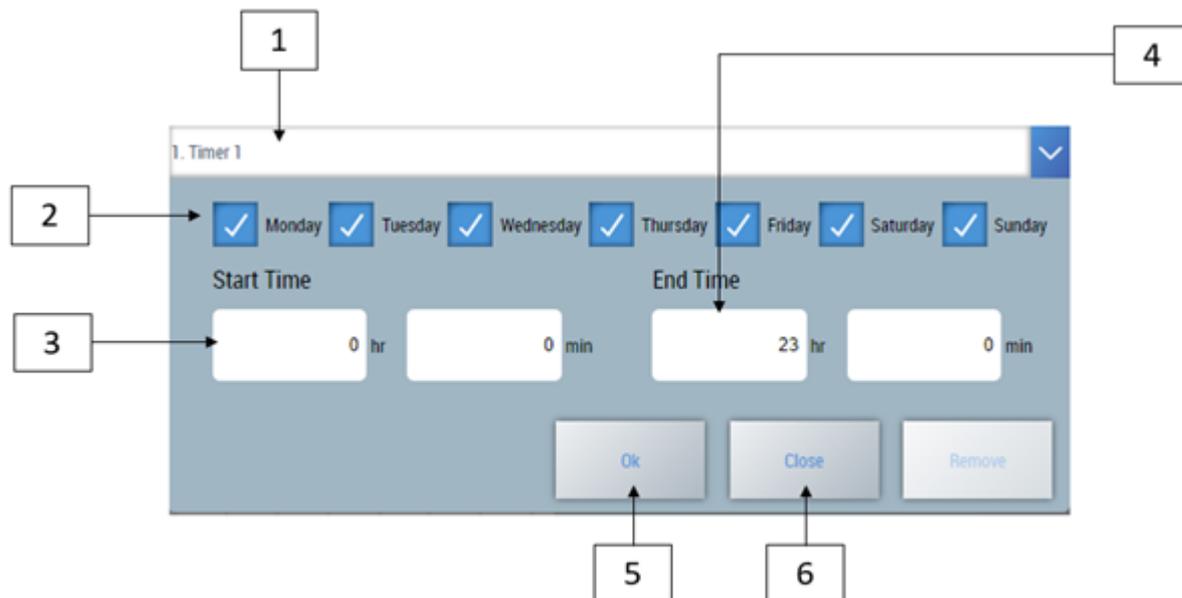
4.12.3.2 onEditingCanceled

Dieses Ereignis wird generiert, wenn Benutzeränderungen rückgängig gemacht werden und die alte Konfiguration wieder auf das Control geladen wird.

Verfügbar: seit Version 12.6.0

4.12.4 Dialoge

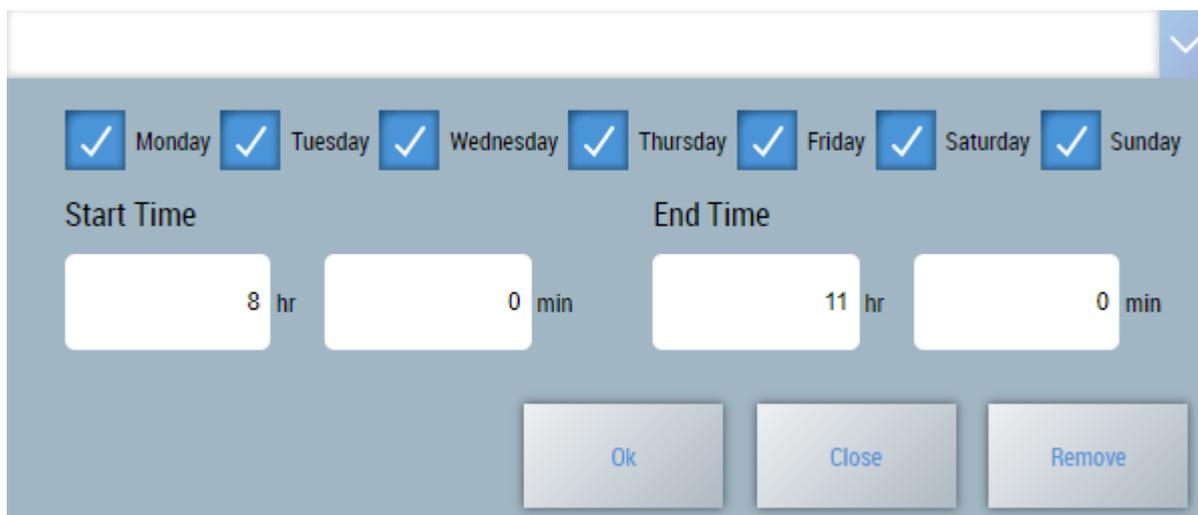
4.12.4.1 Neuen Zeitplan hinzufügen



1. Wählen Sie den Array-Index des hinzuzufügenden Zeitplans.
2. Je nach Checkbox wird der Zeitplan mit den ausgewählten Tagen konfiguriert.
3. Wählen Sie die Startzeit des Zeitplans.
4. Wählen Sie die Endzeit des Zeitplans.
5. Fügen Sie einen neuen Zeitplan hinzu und schließen Sie den Dialog.
6. Änderungen abrechnen und Dialog schließen.
7. Die neue Konfiguration muss durch Ausführen der Funktion [ActivateConfig](#) [► 130] aktiviert werden.

Verfügbar: seit Version 12.6.0

4.12.4.2 Bearbeiten oder Entfernen eines Zeitplans



1. Wählen Sie durch Antippen oder Anklicken einen zu bearbeitenden Zeitplan aus...
2. Rufen Sie die Funktion [EditScheduleData](#) [► 131] auf, um den Bearbeitungsdialog zu sehen

3. Ändern Sie die Parameter oder klicken Sie auf die Schaltfläche Remove, um einen Zeitplan aus der Liste zu entfernen.
4. Die neue Konfiguration muss durch Ausführen der Funktion [ActivateConfig \[► 130\]](#) aktiviert werden.

Verfügbar: seit Version 12.6.0

4.13 StateIndicator

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0



Das Control StateIndicator zeigt verschiedene Zustände an und kann als LED-Anzeige in TwinCAT-HMI-Projekten verwendet werden.

Seine Füllfarbe wechselt je nach aktuellem Zustand zwischen drei möglichen Farben.

Der Zustand kann einer der drei möglichen Zustände "Fehlerzustand", "Warnzustand" oder "Normalzustand" sein und kann durch Attribute geändert werden.

4.13.1 Attribute

Kategorie: Indicator State

Name	Beschreibung
IndicatorState [► 134]	Symbol für den Indikatorstatus des Datentyps <code>BYTE</code> . Die drei niederwertigsten Bits werden zur Berechnung des Status verwendet.
NormalState [► 134]	Wird dieses Attribut <code>TRUE</code> , wird der Zustand des Controls auf "Normal state" gesetzt.
WarningState [► 135]	Wird dieses Attribut <code>TRUE</code> , wird der Zustand des Controls auf "Warning state" gesetzt.
ErrorState [► 133]	Wird dieses Attribut <code>TRUE</code> , wird der Zustand des Controls auf "Error state" gesetzt.

Kategorie: Colors

Name	Beschreibung
NormalStateColor [► 134]	Legt die Farbe "Normal state" für das Control fest. Die Standardfarbe ist Grün.
WarningStateColor [► 135]	Legt die Farbe "Warning state" für das Control fest. Die Standardfarbe ist Orange.
ErrorStateColor [► 134]	Legt die Farbe "Error state" für das Control fest. Die Standardfarbe ist Rot.

4.13.1.1 ErrorState

Wird dieses Attribut `TRUE`, wird der Zustand des Controls auf "Error state" gesetzt, und das Control wird mit der Farbe "Error state" gefüllt.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: `getErrorState`

Attribut-Setter: `setErrorState`

Verfügbar: seit Version 12.6.0

4.13.1.2 ErrorStateColor

Legt die Farbe "Error state" für das Control fest. Die Standardfarbe ist Rot.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getErrorStateColor

Attribut-Setter: setErrorStateColor

Verfügbar: seit Version 12.6.0

4.13.1.3 IndicatorState

Symbol für den Indikatorstatus des Datentyps `BYTE`. Die drei niederwertigsten Bits werden zur Berechnung des Status verwendet. In der nachstehenden Tabelle werden die Bitpositionen und ihre Bedeutung für das Control erläutert.

Bit-Position	Name	Bedeutung
0	Normal state bit	Wird dieses Bit auf <code>TRUE</code> gesetzt, hat das Control den "Normal state" als aktiven Zustand.
1	Warning state bit	Wird dieses Bit auf <code>TRUE</code> gesetzt, hat das Control den "Warning state" als aktiven Zustand. Wenn <code>TRUE</code> , überschreibt dieses Bit den durch Bit 0 gesetzten Zustand.
2	Error state bit	Wird dieses Bit auf <code>TRUE</code> gesetzt, hat das Control den "Error state" als aktiven Zustand. Wenn <code>TRUE</code> , überschreibt dieses Bit die durch Bit 1 und Bit 0 gesetzten Zustände.
3..7	Not used	Diese Bits enthalten keine Informationen über den Zustand des Controls.

Schema: tchmi:general#/definitions/BYTE

Attribut-Getter: getIndicatorState

Attribut-Setter: setIndicatorState

Verfügbar: seit Version 12.6.0

4.13.1.4 NormalState

Wird dieses Attribut `TRUE`, wird der Zustand des Controls auf "Normal state" gesetzt, und das Control wird mit der Farbe "Normal state" gefüllt.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getNormalState

Attribut-Setter: setNormalState

Verfügbar: seit Version 12.6.0

4.13.1.5 NormalStateColor

Legt die Farbe "Normal state" für das Control fest. Die Standardfarbe ist Grün.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getNormalStateColor

Attribut-Setter: setNormalStateColor

Verfügbar: seit Version 12.6.0

4.13.1.6 WarningState

Wird dieses Attribut `TRUE`, wird der Zustand des Controls auf "Warning state" gesetzt, und das Control wird mit der Farbe "Warning state" gefüllt.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getWarningState

Attribut-Setter: setWarningState

Verfügbar: seit Version 12.6.0

4.13.1.7 WarningStateColor

Legt die Farbe "Warning state" für das Control fest. Die Standardfarbe ist Orange.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getWarningStateColor

Attribut-Setter: setWarningStateColor

Verfügbar: seit Version 12.6.0

4.14 Table

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

Das Table-Control kann verwendet werden, um mehrere SPS-Variablen in einem paginierten Tabellenformat zu visualisieren. Details zur Konfiguration finden Sie im Kapitel Konfiguration.

Row	Cutter 3	Unit
Actuator		
4 2	Feedback reversed	<input type="checkbox"/> ⓘ 8
3 5	Output reversed	<input type="checkbox"/> ⓘ
Actuator base position		
5	Feedback delay	<input type="text" value="0.00"/> s ⓘ
6	Feedback delay off	<input type="text" value="0.00"/> s ⓘ
7	Hold output signal	<input type="checkbox"/> ⓘ
8	Invert feedback	<input type="checkbox"/> ⓘ
9	Latch feedback	<input checked="" type="checkbox"/> ⓘ
10	Return time	<input type="text" value="0.00"/> s ⓘ
11	Enable return time	<input type="checkbox"/> ⓘ
12	Timeout	<input type="text" value="5.00"/> s ⓘ
13	Use impulse signal	<input type="checkbox"/> ⓘ
Actuator work position		
15	Feedback delay	<input type="text" value="0.00"/> s ⓘ
6 1 page 1 2		2 Actuator 1 2 7

1. Text, der auf dem Unterseiten-Selektor angezeigt wird
2. Text, der auf dem Tabellen-Selektor angezeigt wird
3. Titel der Tabelle
4. Zeilennummer
5. Zeilenbeschreibung
6. Schaltflächen zum Navigieren zwischen Unterseiten einer Tabelle
7. Schaltflächen zum Navigieren zwischen verschiedenen Tabellen
8. Schaltfläche zum Anzeigen von Zeileninformationen.

4.14.1 Attribute

Kategorie: Table Config

Name	Beschreibung
TableRowCount [► 142]	Zeilenanzahl.
SubTableText [► 142]	Text, der für den Tabellen-Selektor angezeigt werden soll.
FirstTable [► 137]	Die erste Tabellenkonfiguration.
CreateMultipleTables [► 137]	Ermöglicht die Erstellung mehrerer Tabellen im Control.
Tables [► 142]	Liste aller Tabellen.
RowHeight [► 142]	Legt die Zeilenhöhe fest.
RowHeightUnit [► 142]	Setzt die Row HeightUnit immer in Pixel.

Kategorie: Selection

Name	Beschreibung
ActiveSubPageIndex [► 137]	Index der angezeigten Unterseite.

Name	Beschreibung
ActiveTableIndex [► 137]	Index der angezeigten Tabelle.

Benutzeränderungen loggen:

Name	Beschreibung
LogChangesByUser [► 141]	Logging der vom Benutzer auf dem Client vorgenommenen Änderungen in einer Logdatei. Die Eigenschaft "LogChangesByUser" für jede Zeilenkonfiguration wird dieses Attribut ersetzen.

4.14.1.1 ActiveSubPageIndex

Legt den Index der aktiven Unterseite fest, die auf dem Control angezeigt wird.

Wenn der Index außerhalb des Bereichs liegt, wird er auf den Standardwert (0) zurückgesetzt.

Schema: tchmi:general#/definitions/INT

Attribut-Getter: getActSubPage

Attribut-Setter: setActSubPage

Verfügbar: seit Version 12.6.0

4.14.1.2 ActiveTableIndex

Legt den Index der Untertabelle fest, die auf dem Control angezeigt wird.

Wenn der Index außerhalb des Bereichs liegt, wird er auf den Standardwert (0) zurückgesetzt.

Schema: tchmi:general#/definitions/INT

Attribut-Getter: getActSubTable

Attribut-Setter: setActSubTable

Verfügbar: seit Version 12.6.0

4.14.1.3 CreateMutipleTables

Wenn `TRUE`, werden mehrere Tabellen gemäß dem Attribut [Tables](#) [► 142] erstellt.

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getCreateMultipleTables

Attribut-Setter: setCreateMultipleTables

Verfügbar: seit Version 12.6.0

4.14.1.4 FirstTable

Die Elemente der ersten Tabellenzeile werden mit diesem Attribut konfiguriert.

TplConfigTable_2 | FirstTable
✕

Elements

FBValueDataType	RowNumber	LocalizationKey
FloatDataType	1	L Row_Locale_1
BoolDataType	2	L Row_Locale_2
IntDataType	3	L Row_Locale_3
noneDataType	4	L Row_Locale_4
IntDataType_E	5	L Row_Locale_5
FloatDataType	6	L Row_Locale_6
BoolDataType	7	L Row_Locale_7
IntDataType	8	L Row_Locale_8
IntDataType	9	L Row_Locale_9
noneDataType	10	L Row_Locale_10
noneDataType	11	L Row_Locale_11
FloatDataType	12	L Row_Locale_12
FloatDataType	13	L Row_Locale_13
FloatDataType	14	L Row_Locale_12
FloatDataType	15	L Row_Locale_13
IntDataType_E	16	L Row_Locale_5
StringDataType	17	L Row_Locale_6

✕ ↑ ↓
TableRow
Add

Properties

- ▲ Access Rights
 - AccessRight (no items) ...
- ▲ ControlConfig
 - FBValue_DataType FloatDataType
 - UnitGroup f_x Length
 - MaxValue S fbConfigTable1::fbR
 - MinValue S fbConfigTable1::fbR
 - RefValue S fbConfigTable1::fbR
 - Value S ↑ fbConfigTable1::
 - ComboBoxSrcData (4 items) ...
 - isReadOnly
 - DigitsAfterDecimal
- ▲ Fixed-Units
 - DisplayUnit f_x
 - SystemUnit f_x
- ▲ General
 - isDescriptor
 - RowNumber 1
 - LocalizationKey L Row_Locale_1

OK Cancel

Erste Tabelle Array-Element-Eigenschaften:

Eigenschaft	Typ	Beschreibung
Access Right	tchmi:framework#/definitions/AccessConfig	Die Zugriffsrechte für jedes Zeilenelement können aufgelistet werden, was eine individuelle Konfiguration des Zugriffs für jede Zeile ermöglicht.
FBValue_DataType	tchmi:framework#/definitions/RowDataType	<ul style="list-style-type: none"> Wählen Sie den Datentyp aus der Aufzählung aus, um den Wert zu formatieren und anzuzeigen. Der Benutzer kann [noneDataType, BoolDataType, IntDataType, IntDataType_Enum, FloatDataType, oder TimeDataType, StringDataType] als Datentyp auswählen. Auf der Grundlage dieser Auswahl wird in der Zeile ein Control erstellt, das den Wert anzeigt und Benutzereingaben ermöglicht. Für BoolDataType wird eine Checkbox als Control zur Benutzerinteraktion angezeigt.

Eigenschaft	Typ	Beschreibung
		<ul style="list-style-type: none"> • Wenn <code>IntDataType_Enum</code> eingestellt ist, wird ein Kombinationsfeld angezeigt. Diese wird über die Eigenschaft <code>ComboBoxSrcData</code> konfiguriert. • Wenn der Datentyp <code>noneDataType</code> ausgewählt ist, wird keine Eingabe-Control angezeigt. • Für die übrigen Optionen erscheint eine <code>InputBox</code> mit den entsprechenden Konfigurationen.
UnitGroup	string	Legt die Einheitengruppe für das Control <code>InputBox</code> für diese Zeile fest.
MaxValue	tchmi:framework#/definitions/ValueAttribute	Legt den Maximalwert fest, den der Benutzer eingeben kann.
MinValue	tchmi:framework#/definitions/ValueAttribute	Legt den Minimalwert fest, den der Benutzer eingeben kann.
RefValue	tchmi:framework#/definitions/ValueAttribute	Legt den Referenzwert fest, der für die Berechnung des Prozentsatzes benötigt wird, wenn die Einheitengruppe Prozent ist.
Wert	tchmi:framework#/definitions/ValueAttribute	Wertesymbol-Pfad des Wertes, der bei einer Benutzereingabe angezeigt und aktualisiert wird.
ComboBox SrcData	tchmi:framework#/definitions/TcHmi.Controls.Beckhoff.TcHmiCombobox.ListItemList	Mit dieser Eigenschaft können Sie ein Kombinationsfeld konfigurieren.
isReadOnly	boolean	Auf <code>TRUE</code> setzen, wenn der in der Zeile angezeigte Wert vom Benutzer nicht bearbeitet werden soll.
DigitsAfterDecimal	number	Anzahl der angezeigten Nachkommastellen. Verfügbar: ab Version 12.14.0
DisplayUnit	string	Legt eine feste Einheit als Anzeigeeinheit für die Zeile fest. Der <code>UnitGroup</code> -Wert muss eine leere Zeichenkette sein, um dies zu verwenden. Verfügbar: ab Version 12.14.0
SystemUnit	string	Legt eine feste Einheit als Systemeinheit für die Zeile fest. Der <code>UnitGroup</code> -Wert muss eine leere Zeichenkette sein, um dies zu verwenden. Verfügbar: ab Version 12.14.0
RecipeManagement	tchmi:framework#/definitions/RecipeType	Der Benutzer kann auswählen, ob Werte von dem <code>Table</code> -Control zu einem Rezepttyp hinzugefügt werden sollen.
RecipeTypeName	string	Name des Rezepttyps, zu dem die Symbole von dem <code>Table</code> -Control hinzugefügt werden sollen.

Eigenschaft	Typ	Beschreibung
isDescriptor	boolean	Auf <code>TRUE</code> setzen, um diese Zeile nur als Deskriptor ohne Eingabemöglichkeit zu verwenden.
RowNumber	number	Die angezeigte Zeilennummer der Zeile.
LocalizationKey	string	Text, der als Zeilentitel für jede Zeile angezeigt wird.
RowInformation	string	Informationstext, der in einem Pop-up-Fenster angezeigt wird, wenn der Benutzer auf das Informationssymbol in der Zeile klickt. Verfügbar: ab Version 12.14.0
LogChangesByUser	boolean	Setzen Sie ihn auf <code>TRUE/FALSE</code> , um das Loggen von Benutzeränderungen für die Zeile zu aktivieren/deaktivieren. Wenn es nicht gesetzt wird, gilt der Wert des Attributs <code>LogUserChanges</code> [► 141]. Verfügbar: ab Version 12.12.0
ChangeLogDescriptor	String	Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird. Verfügbar: ab Version 12.14.0

Schema: tchmi:framework#/definitions/TableRowsList

Attribut-Getter: getReftable

Attribut-Setter: setReftable

Verfügbar: ab Version 12.6.0

4.14.1.5 HeaderBackgroundColor

Legt die Hintergrundfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderBackgroundColor

Attribut-Setter: setHeaderBackgroundColor

Verfügbar: ab Version 12.8.0

4.14.1.6 HeaderFontColor

Legt die Textfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderFontColor

Attribut-Setter: setHeaderFontColor

Verfügbar: ab Version 12.8.0

4.14.1.7 HeaderFontSize

Die Schriftgröße der Kopfzeile. Wenn Prozent als Einheit festgelegt ist, ist die Kopfzeilengröße relativ zur Schriftgröße des übergeordneten Elements.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderFontSize

Attribut-Setter: setHeaderFontSize

Verfügbar: ab Version 12.8.0

4.14.1.8 HeaderFontSizeUnit

Legt die Einheit der Kopfzeilenhöhe fest:

- Pixel für eine absolute Größe
- Prozent für eine relative Größe

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderHeightUnit

Attribut-Setter: setHeaderHeightUnit

Verfügbar: ab Version 12.8.0

4.14.1.9 HeaderHeight

Legt die Höhe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderHeight

Attribut-Setter: setHeaderHeight

Verfügbar: ab Version 12.8.0

4.14.1.10 HeaderHeightUnit

Legt die Einheit der Kopfzeilenhöhe fest:

- Pixel für eine absolute Größe
- Prozent für eine relative Größe

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderHeightUnit

Attribut-Setter: setHeaderHeightUnit

Verfügbar: ab Version 12.8.0

4.14.1.11 LogUserChanges

Wenn dieses Attribut auf `TRUE` gesetzt ist, loggt das Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket jede Änderung durch den Benutzer auf dem HMI-Bildschirm in einer Logdatei.

Die Eigenschaft "LogChangesByUser" für jede Zeilenkonfiguration wird dieses Attribut ersetzen.

Um dieses Attribut zu verwenden, muss [UserLoggerSymbol](#) [► 45] mit einem Binding an [FB_UserLoggingHmi](#) konfiguriert werden.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getLogUserChanges

Attribut-Setter: setLogUserChanges

Verfügbar: ab Version 12.12.0

4.14.1.12 RowHeight

Legt die Höhe für jede Zeile im Control fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getRowHeight

Attribut-Setter: setRowHeight

Verfügbar: seit Version 12.6.0

4.14.1.13 RowHeightUnit

Legt die Einheit der Zeilenhöhe immer in Pixel fest.

Schema: tchmi:framework#/definitions/PixelUnit

Attribut-Getter: getRowHeightUnit

Attribut-Setter: setRowHeightUnit

Verfügbar: seit Version 12.6.0

4.14.1.14 SubTableText

Der Text, der beim Tabellen-Selektor angezeigt wird.

Schema: tchmi:general#/definitions/String

Attribut-Getter: getSubTableText

Attribut-Setter: setSubTableText

Verfügbar: seit Version 12.6.0

4.14.1.15 TableRowCount

Anzahl der Zeilen, die auf einer Seite angezeigt werden. Die Gesamtzahl der Zeilen in jeder Tabelle wird durch dieses Attribut geteilt, um Unterseiten zu erstellen.

Schema: tchmi:general#/definitions/Number

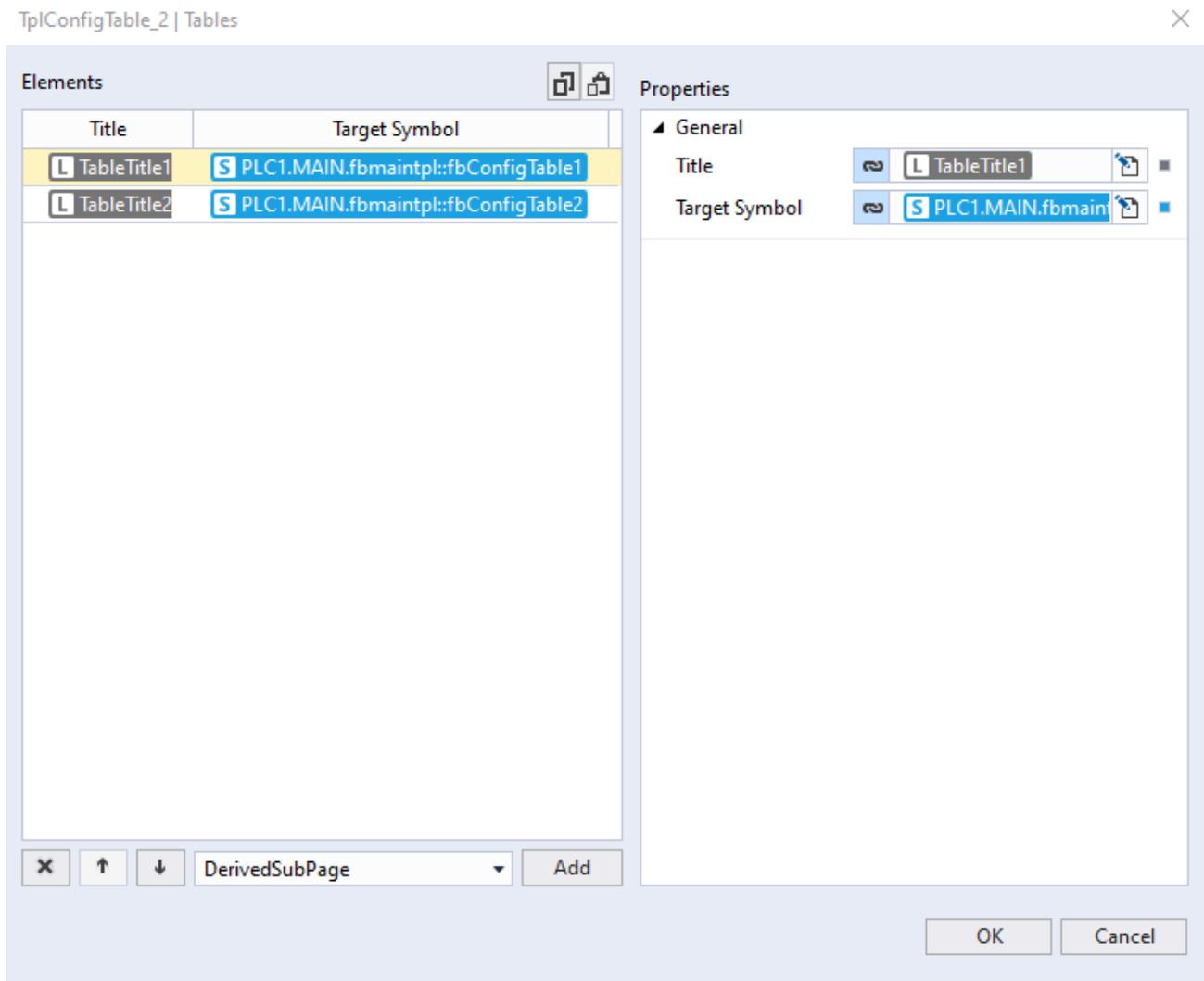
Attribut-Getter: getTableRowCount

Attribut-Setter: setTableRowCount

Verfügbar: seit Version 12.6.0

4.14.1.16 Tables

Nachdem Sie die erste Tabelle konfiguriert haben, können Sie dem Control weitere ähnliche Tabellen hinzufügen.



Eigenschaften der Tabellenelemente:

Name	Beschreibung
Table Title	Dies ist ein String, der in der obersten Zeile der Tabelle angezeigt wird.
Target Symbol	Dies ist ein übergeordnetes Symbol, das alle Untersymbole in der Zeile bedeckt.

Es muss mindestens ein Element in der Tabellenliste vorhanden sein, und das erste Element muss das Zielsymbol des FB haben, dessen direkte Kinder in allen ersten Tabellenzeilen konfiguriert sind.

Siehe Abschnitt [Schritt für Schritt \[▶ 144\]](#), um weitere Informationen über die Konfiguration des Attributs "Table" zur Anzeige mehrerer Untertabellen auf dem Control zu erhalten.

Schema: tchmi:framework#/definitions/DerivedSubPageList

Attribut-Getter: getTableList

Attribut-Setter: setTableList

Verfügbar: seit Version 12.6.0

4.14.2 Events

Name	Beschreibung
onTableDataChanged [▶ 144]	Dieses Ereignis wird ausgelöst, wenn ein Benutzer einen Wert ändert, indem er das Eingabe-Control einer beliebigen Zeile bearbeitet.

4.14.2.1 onTableDataChanged

Dieses Ereignis wird ausgelöst, wenn ein Benutzer einen Wert ändert, indem er das Eingabe-Control einer beliebigen Zeile bearbeitet.

Verfügbar: seit Version 12.6.0

4.14.3 Schritt für Schritt

4.14.3.1 Table-Control für die Anzeige einer einzelnen Tabelle konfigurieren

1. Ziehen Sie ein Table-Control aus der Toolbox auf den Inhalt.
2. Setzen Sie den Wert `FALSE` auf das Attribut [CreateMutipleTables \[\[▶ 137\]\(#\)\]](#).
3. Bearbeiten Sie das Attribut [First Table \[\[▶ 137\]\(#\)\]](#) mit dem Objekteditor-Dialog, der erscheint, wenn Sie auf die Schaltfläche Edit klicken.
4. Fügen Sie so viele Zeilen hinzu, wie Sie benötigen. Bearbeiten Sie jede Zeilenkonfiguration und stellen Sie die entsprechende Verbindung zu den SPS-Symbolen her.

4.14.3.2 Table-Control für die Anzeige mehrerer Tabellen konfigurieren

Table-Control kann für die Anzeige mehrerer Tabellen verwendet werden. Mehrere ähnliche Funktionsbausteine können als Untertabellen aufgelistet werden, und der Benutzer kann mit Hilfe von Navigationsschaltflächen auf dem Control durch jede Untertabelle navigieren.

In der Anleitung wird erläutert, wie ein Benutzer mehrere Funktionsbausteine desselben Typs visualisieren kann, und zwar anhand eines Beispiels zur Konfiguration mehrerer Temperaturzonen-Funktionsbausteine, indem die Attribute [First Table \[\[▶ 137\]\(#\)\]](#) und [Tables \[\[▶ 142\]\(#\)\]](#) konfiguriert werden.

Aufbau eines Temperaturzonen-Funktionsbausteins:

<ul style="list-style-type: none"> fbConfigTempZone <ul style="list-style-type: none"> AbsoluteHigh AbsoluteLow ActCurrent ActTempGain ActTempOffset ActualTemperature CJ_CompMode CJ_CompZone CoolerSwapIdx Cooling DisableAutoStep DisableTerminalCom Enable EnableErrorHeating ErrCurrent_Tolerance Error ErrorHeatingFactor ExtruderComp ExtruderCompEna ExtruderId HeaterSwapIdx Heating Hibernatel_Cool Hibernatel_Heat HighPrecision I_PartLimit 	<ul style="list-style-type: none"> PLC1.ARRAY_1..20_OF-FB_MdTempZoneHmi PLC1.FB_MdTempZoneHmi PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-LREAL PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-LREAL PLC1.PROPERTY-GET-SET-INT PLC1.PROPERTY-GET-SET-INT PLC1.PROPERTY-GET-SET-INT PLC1.PROPERTY-GET-SET-INT PLC1.PROPERTY-GET-BOOL PLC1.PROPERTY-GET-SET-BOOL PLC1.PROPERTY-GET-SET-BOOL PLC1.PROPERTY-GET-SET-BOOL PLC1.PROPERTY-GET-SET-BOOL PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-BOOL PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-SET-LREAL PLC1.PROPERTY-GET-SET-BOOL PLC1.PROPERTY-GET-SET-INT PLC1.PROPERTY-GET-SET-INT PLC1.PROPERTY-GET-BOOL PLC1.PROPERTY-GET-SET-BOOL PLC1.PROPERTY-GET-SET-BOOL PLC1.PROPERTY-GET-SET-BOOL PLC1.PROPERTY-GET-SET-LREAL
--	--

Jede Eigenschaft in diesem Funktionsbaustein sollte für den Benutzer auf dem Table-Control als Zeilenvariable zugänglich sein. Das Attribut [First Table \[► 137\]](#) kann zu diesem Zweck konfiguriert werden.

Es gibt mehrere Instanzen desselben Temperaturzonen-Funktionsbausteins im Projekt, und sie können mit dem Attribut **Tables** zum Table-Control hinzugefügt werden.

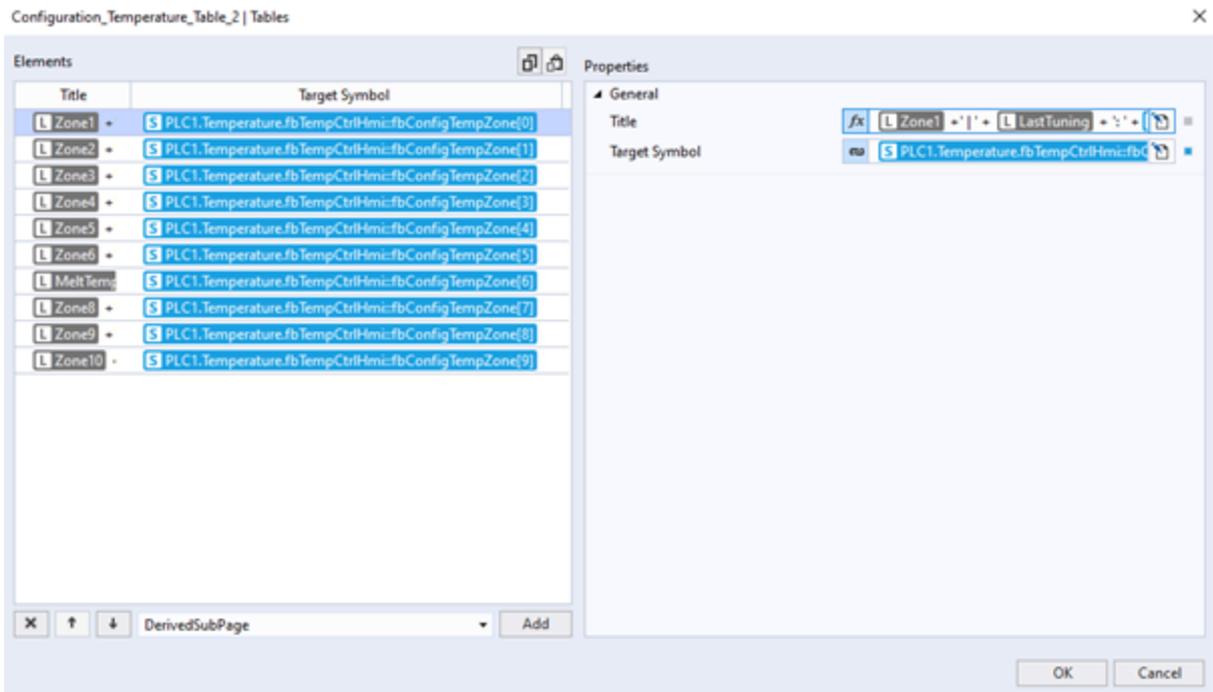
fbConfigTempZone	PLC1.ARRAY_1_20_OF-FB_MdTempZoneHmi
[0]	PLC1.FB_MdTempZoneHmi
[1]	PLC1.FB_MdTempZoneHmi
[2]	PLC1.FB_MdTempZoneHmi
[3]	PLC1.FB_MdTempZoneHmi
[4]	PLC1.FB_MdTempZoneHmi
[5]	PLC1.FB_MdTempZoneHmi
[6]	PLC1.FB_MdTempZoneHmi
[7]	PLC1.FB_MdTempZoneHmi
[8]	PLC1.FB_MdTempZoneHmi
[9]	PLC1.FB_MdTempZoneHmi
[10]	PLC1.FB_MdTempZoneHmi
[11]	PLC1.FB_MdTempZoneHmi
[12]	PLC1.FB_MdTempZoneHmi
[13]	PLC1.FB_MdTempZoneHmi
[14]	PLC1.FB_MdTempZoneHmi
[15]	PLC1.FB_MdTempZoneHmi

Gehen Sie folgendermaßen vor, um das Table-Control zu konfigurieren:

1. Ziehen Sie ein Table-Control aus der Toolbox auf den Inhalt.
2. Konfigurieren Sie das Attribut [First Table \[► 137\]](#), um die Eigenschaften des Symbols fbConfigTempZone[0] für jede Zeile aufzulisten.
 - Fügen Sie die gewünschte Anzahl von Zeilen im Objekteditor-Dialog hinzu, der beim Bearbeiten des Attributs erscheint.
 - Jede Zeile hat eine entsprechende Konfiguration bezüglich Datentyp, Einheitengruppe, Maximalwert, Minimalwert usw.
 - Jede Zeilenwerteigenschaft ist an eine Variable der ersten Ebene in fbConfigTempZone[0] gebunden, z. B. in diesem Fall AbsoluteHigh, AbsoluteLow usw.
 - Jetzt dient fbConfigTempZone[0] als Referenzsymbol, und jede Zeile ist an die Variable der ersten Ebene innerhalb von fbConfigTempZone[0] gebunden.
3. Setzen Sie das Attribut [CreateMultipleTables \[► 137\]](#) auf TRUE.
4. Konfigurieren Sie das Attribut **Tables**, um die Anzahl der Funktionsbausteininstanzen aufzulisten, die genau dieselbe Variablenstruktur wie das bei der Konfiguration des Attributs [FirstTable \[► 137\]](#) verwendete Referenzsymbol haben.
 - Fügen Sie mehrere Tabellenelemente in das Editor-Array ein.
 - Konfigurieren Sie das erste Element im Array. Die Eigenschaft **Title** sollte auf einen entsprechenden String gesetzt werden.
 - Die Eigenschaft **Target Symbol** des ersten Elements muss an das Referenzsymbol gebunden sein, dessen erste untergeordnete Variablen für jede Zeile des Attributs [FirstTable \[► 137\]](#) konfiguriert sind. In diesem Beispiel wird diese Eigenschaft an das Symbol fbConfigTempZone[0] gebunden.

Die Eigenschaft Target Symbol des ersten Elements im Array [Tables \[► 142\]](#) muss an dasselbe Symbol gebunden werden, dessen Untersymbole als Wert für jede Zeile im Array [First Table \[► 137\]](#) konfiguriert werden.

- Legen Sie für jede Tabelle den gewünschten Titel fest und setzen Sie das Zielsymbol auf das jeweilige Funktionsbausteinsymbol.



5. Dadurch sind alle 10 Temperaturzonen auf dem Control verfügbar und der Benutzer kann mit den Navigationstasten der Untertabelle zu der entsprechenden Zone navigieren.

Row		Unit
1	Save zone name	<input type="checkbox"/>
2	Module ID	<input type="text" value="0"/>
3	Zone ID	<input type="text" value="0"/>
4	Supply ID	<input type="text" value="0"/>
5	Extruder ID	<input type="text" value="0"/>
6	Select cooling output	NoSignal <input type="button" value="v"/>
7	Select heating output	NoSignal <input type="button" value="v"/>
8	Temperature sensor terminal	NoTerminal <input type="button" value="v"/>
9	Sensor type	NoSensor <input type="button" value="v"/>
10	Terminal channel	<input type="text" value="0"/>
11	In use	<input type="checkbox"/>
12	Enable	<input type="checkbox"/>
13	Use cooling	<input type="checkbox"/>
14	Tune cooling	<input type="checkbox"/>
15	Forced cooling enabled	<input type="checkbox"/>

Page 1|6 Zone 1|10

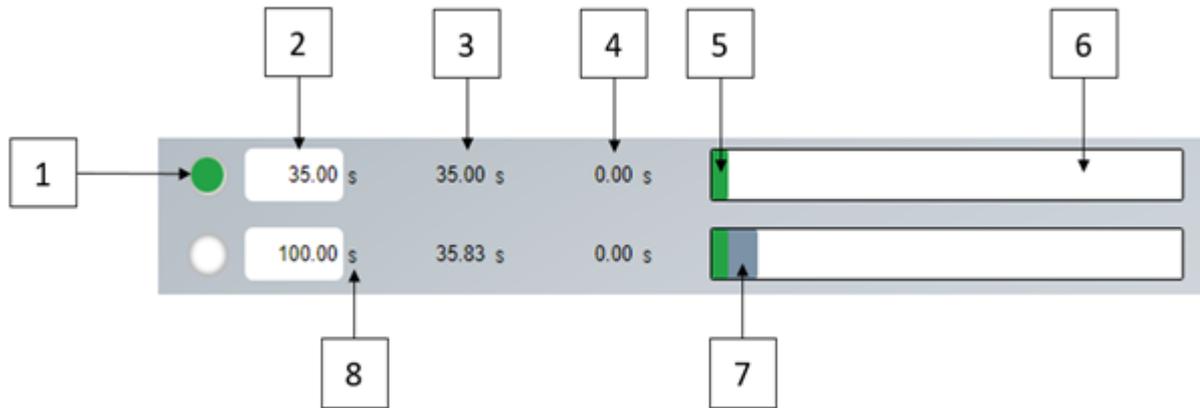
4.15 TimerControl

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

Das TimerControl kann verwendet werden, um ein FB_Timer-Symbol zu visualisieren und zu konfigurieren. Dieses Symbol ähnelt der Funktionsweise eines Timers in der SPS.

Auf dem Control sind verschiedene numerische und grafische Komponenten zu sehen, die weitere Informationen über den aktuellen Status liefern. Der Benutzer kann die SetValue-Eigenschaft des Timer-Funktionsbausteins über das Control eingeben.



- 1. Abgelaufene Anzeige
- 2. Threshold InputBox
- 3. Aktuelle Zeit (schreibgeschützt)
- 4. Gelacthte Zeit (schreibgeschützt)
- 5. Fortschritt Aktueller Wert
- 6. Maximale Dauer
- 7. Fortschritt Schwellenwert
- 8. Text der Einheit

4.15.1 Attribute

Kategorie: Colors

Name	Beschreibung
Elapsed Indicator Color [► 148]	Füllfarbe der Anzeige, wenn der Timer abläuft.

Kategorie: Common

Name	Beschreibung
FB Timer [► 148]	Symbol vom FB Timer-Typ.
Progress Bar Max [► 149]	Größter auf dem Fortschrittsbalken sichtbarer Wert.
Show Elapsed [► 150]	Anzeige der verstrichenen Zeit.
Show Latched [► 150]	Gelacthte Zeit anzeigen.
Show Progress [► 150]	Fortschrittsbalken anzeigen.
Threshold As Progress Max [► 151]	Wenn TRUE, wird der Schwellenwert als Maximalwert für die Visualisierung des Fortschrittsbalkens verwendet.
DigitsAfterDecimal [► 148]	Dezimalgenauigkeit aller auf dem Control angezeigten numerischen Werte.

Kategorie: Time Unit Group

Name	Beschreibung
Time Unit Group [► 151]	Legt die Einheitengruppe für Eingabefelder fest.

Recipe Management:

Name	Beschreibung
RecipeManagement [► 149]	Legen Sie fest, ob das Control Symbole, die ihm zugeordnet sind, zu einem Rezepttyp hinzufügt.
RecipeTypeName [► 150]	Mit diesem Attribut kann der Benutzer den Namen des Rezepttyps festlegen, zu dem das Control die ihm zugeordneten Symbole hinzufügt und alle Änderungen einlernt.

User Changes Logging:

Name	Beschreibung
LogChangesByUser [► 149]	Logging der vom Benutzer auf dem Client vorgenommenen Änderungen in einer Logdatei.
ChangeLogDescriptor [► 148]	Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

4.15.1.1 ChangeLogDescriptor

Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

Schema: "tchmi:general#/definitions/STRING"

Attribut-Getter: getChangeLogDescriptor

Attribut-Setter: setChangeLogDescriptor

Verfügbar: ab Version 12.14.0

4.15.1.2 DigitsAfterDecimal

Die Anzahl der Dezimalstellen aller numerischen Werte, die auf dem Control angezeigt werden.

Schema: tchmi:general#/definitions/UINT

Attribut-Getter: getDigitsAfterDecimal

Attribut-Setter: setDigitsAfterDecimal

Verfügbar: ab Version 12.12.0

4.15.1.3 ElapsedIndicatorColor

Füllfarbe der Anzeige, wenn der Timer abläuft.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getElapsedIndicatorColor

Attribut-Setter: setElapsedIndicatorColor

Verfügbar: seit Version 12.6.0

4.15.1.4 FBTimer

Dieses Attribut sollte mit einem Serversymbol (Funktionsbaustein oder Struktur) verknüpft werden, das über die unten beschriebenen Eigenschaften verfügt.

Name	Typ	Beschreibung
ActualValue	LREAL	Real verstrichene Zeit in LREAL.
LatchedValue	LREAL	Letzte gelatchte Zeit in LREAL.

Name	Typ	Beschreibung
Out	BOOL	Wird auf TRUE gesetzt, wenn der Timer abgelaufen ist.
SetValue	LREAL	Sollzeit vom Benutzer. Diese Eigenschaft sollte nicht schreibgeschützt sein.

fbHmi	PLC1.FB_TimerHmi	<input checked="" type="checkbox"/>	fbHmi
ActualValue	PLC1.PROPERTY-GET-LREAL	<input checked="" type="checkbox"/>	ActualValue
LatchedValue	PLC1.PROPERTY-GET-LREAL	<input checked="" type="checkbox"/>	LatchedValue
Out	PLC1.PROPERTY-GET-BOOL	<input checked="" type="checkbox"/>	Out
SetValue	PLC1.PROPERTY-GET-SET-LREAL	<input checked="" type="checkbox"/>	SetValue

Schema: tchmi:framework#/definitions/Symbol

Attribut-Getter: getTimer

Attribut-Setter: setTimer

Verfügbar: seit Version 12.6.0

4.15.1.5 LogUserChanges

Wenn dieses Attribut auf TRUE gesetzt ist, loggt das Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket jede Änderung durch den Benutzer auf dem HMI-Bildschirm in einer Logdatei.

Um dieses Attribut zu verwenden, muss [UserLoggerSymbol](#) [▶ 45] mit einem Binding an [FB_UserLoggingHmi](#) konfiguriert werden.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getLogUserChanges

Attribut-Setter: setLogUserChanges

Verfügbar: ab Version 12.12.0

4.15.1.6 ProgressBarMax

Damit wird der maximale Schwellenwert festgelegt, den ein Benutzer eingeben kann.

Dient auch als Höchstgrenze für den Fortschrittsbalken, wenn das Attribut [ThresholdAsProgressMax](#) [▶ 151] auf FALSE gesetzt ist.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getProgressBarMax

Attribut-Setter: setProgressBarMax

Verfügbar: seit Version 12.6.0

4.15.1.7 RecipeManagement

Legen Sie fest, ob das Control die Symbole, die ihm zugeordnet sind, zu einem Rezepttyp hinzufügen soll.

Wenn das Projekt das [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper](#) [▶ 198]-Paket enthält, fügt das Control die Serversymbole, die darauf abgebildet sind, dem Rezepttyp hinzu, der durch das Attribut [RecipeTypeName](#) [▶ 150] definiert ist.

Nach jedem Benutzerwechsel lernt das Paket [RecipeHelper](#) das aktuell aktive Rezept ein.

Schema: "tchmi:framework#/definitions/TcHmi.Controls.Plastic.RecipeManagement"

Attribut-Getter: getRecipeManagement

Attribut-Setter: setRecipeManagement

Verfügbar: ab Version 12.14.0

4.15.1.8 RecipeTypeName

Mit diesem Attribut kann der Benutzer den Namen des Rezepttyps festlegen, zu dem das Control die ihm zugewiesenen Symbole hinzufügt und alle Änderungen speichert.

Damit das Control Symbole zum Rezepttyp hinzufügen kann, muss das Projekt das Paket [Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper \[► 198\]](#) enthalten und das Attribut [RecipeManagement \[► 149\]](#) muss auf IncludeToRecipe gesetzt sein.



Wenn ein Symbol bereits Teil eines Rezepttyps ist, hat diese Eigenschaft keine Auswirkungen auf die Rezepte. Nur für das Hinzufügen von Symbolen, die nicht Teil eines Rezepttyps sind, wird das Attribut RecipeTypeName berücksichtigt.

Schema: "tchmi:general#/definitions/STRING"

Attribut-Getter: getRecipeTypeName

Attribut-Setter: setRecipeTypeName

Verfügbar: ab Version 12.14.0

4.15.1.9 ShowElapsed

Die Anzeige für die abgelaufene Zeit wird auf das Control angezeigt.

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getShowElapsed

Attribut-Setter: setShowElapsed

Verfügbar: seit Version 12.6.0

4.15.1.10 ShowLatched

Der gelatchte Zeitwert wird auf dem Control angezeigt.

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getShowLatched

Attribut-Setter: setShowLatched

Verfügbar: seit Version 12.6.0

4.15.1.11 ShowProgress

Es wird ein Fortschrittsbalken mit der aktuellen und der eingestellten Zeit angezeigt.

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getShowProgress

Attribut-Setter: setShowProgress

Verfügbar: seit Version 12.6.0

4.15.1.12 ThresholdAsProgressMax

Wenn `TRUE`, wird der Schwellenwert als Maximalwert für die Visualisierung des Fortschrittsbalkens verwendet:

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getThresholdAsProgressMax

Attribut-Setter: setThresholdAsProgressMax

Verfügbar: seit Version 12.6.0

4.15.1.13 TimeUnitGroup

Einheitengruppe für Eingabefelder.

Schema: tchmi:general#/definitions/String

Attribut-Getter: getTimeUnitGroup

Attribut-Setter: setTimeUnitGroup

Verfügbar: seit Version 12.6.0

4.16 UserLoggingViewer

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: ab Version 12.12.0

Dieses Control wird verwendet, um den Inhalt von Logdateien anzuzeigen, die von Plastic-Controls und [LogUserChanges \[► 171\]](#)-Funktionen im Projekt erzeugt werden.

Filename: 8 20240603_Beckhoff_UserLog.csv		Last modified: 6/3/2024, 11:13:24 AM		7	10	
	Timestamp ▾	Page ↕	Name ↕	Old	New	User ↕
1	2024-06-03-11:13:24.335	AMG_2	TheAMG::Move[1]::Segment[4]::Velocity	0	96.2	__SystemUser
2	2024-06-03-11:13:24.331	AMG_2	TheAMG::Move[1]::Segment[3]::Velocity	0	96.2	__SystemUser
3	2024-06-03-11:13:24.325	AMG_2	TheAMG::Move[1]::Segment[2]::Velocity	0	96.2	__SystemUser
4	2024-06-03-11:13:24.321	AMG_2	TheAMG::Move[1]::Segment[1]::Velocity	0	96.2	__SystemUser
5	2024-06-03-11:13:24.317	AMG_2	TheAMG::Move[1]::Segment[0]::Velocity	0	96.2	__SystemUser
6	2024-06-03-11:13:24.313	AMG_2	TheAMG::Move[0]::Segment[4]::Velocity	0	96.2	__SystemUser
7	2024-06-03-11:13:23.301	AMG_2	TheAMG::Move[0]::Segment[3]::Velocity	0	96.2	__SystemUser
8	2024-06-03-11:13:23.296	AMG_2	TheAMG::Move[0]::Segment[2]::Velocity	0	96.2	__SystemUser
9	2024-06-03-11:13:23.292	AMG_2	TheAMG::Move[0]::Segment[1]::Velocity	0	96.2	__SystemUser
10	2024-06-03-11:13:23.289	AMG_2	TheAMG::Move[0]::Segment[0]::Velocity	0	96.2	__SystemUser
11	2024-06-03-11:13:23.284	AMG_2	TheAMG::Move[1]::Segment[4]::InUse	false	true	__SystemUser
12	2024-06-03-11:13:23.281	AMG_2	TheAMG::Move[1]::Segment[4]::Position	0	220	__SystemUser
13	2024-06-03-11:13:22.273	AMG_2	TheAMG::Move[1]::Segment[3]::InUse	false	true	__SystemUser
14	2024-06-03-11:13:22.270	AMG_2	TheAMG::Move[1]::Segment[3]::Position	0	187	__SystemUser
15	2024-06-03-11:13:22.267	AMG_2	TheAMG::Move[1]::Segment[2]::InUse	false	true	__SystemUser
16	2024-06-03-11:13:22.263	AMG_2	TheAMG::Move[1]::Segment[2]::Position	0	151.8	__SystemUser
17	2024-06-03-11:13:22.257	AMG_2	TheAMG::Move[1]::Segment[1]::InUse	false	true	__SystemUser
18	2024-06-03-11:13:22.254	AMG_2	TheAMG::Move[1]::Segment[1]::Position	0	110	__SystemUser
19	2024-06-03-11:13:21.246	AMG_2	TheAMG::Move[0]::Segment[4]::InUse	false	true	__SystemUser
20	2024-06-03-11:13:21.240	AMG_2	TheAMG::Move[0]::Segment[4]::Position	0	11	__SystemUser
21	2024-06-03-11:13:21.236	AMG_2	TheAMG::Move[0]::Segment[3]::InUse	false	true	__SystemUser
22	2024-06-03-11:13:21.229	AMG_2	TheAMG::Move[0]::Segment[3]::Position	0	79.2	__SystemUser
23	2024-06-03-11:13:21.222	AMG_2	TheAMG::Move[0]::Segment[2]::InUse	false	true	__SystemUser

Start 1 2/1/2024
End 2 7/24/2024
3 2/11
4
5
6

1. Anzeige des Anfangsdatums und Anklicken dieses Elements öffnet den Datumsauswahldialog.
2. Anzeige des Enddatums und Klick auf dieses Element öffnet den Datumsauswahldialog.
3. Der aktuelle Dateiindex aus der Liste der Dateien.
4. Die Gesamtzahl der Dateien im Bereich des ausgewählten Start- und Enddatums.
5. Laden der vorherigen Datei in die Liste.
6. Laden der nächsten Datei in die Liste.
7. Das Datum der letzten Änderung der angezeigten Datei.
8. Der Dateiname der angezeigten Datei.
9. Logdaten des Benutzers der ausgewählten Datei.
10. Schaltfläche zum Öffnen des Dialogs zum Bearbeiten der angezeigten Spalten.

4.16.1 Attribute

Kategorie: Configuration

Name	Beschreibung
StartDate [► 155]	Legt das Startdatum für die Auflistung von Dateien fest, definiert als ISO-Zeitstring.
EndDate [► 153]	Legt das Enddatum für die Auflistung von Dateien fest, definiert als ISO-Zeitstring.
SelectedIndex [► 155]	Index der CSV-Datei, die auf dem Control angezeigt wird.

Kategorie: File Config

Name	Beschreibung
Path [▶ 155]	Pfad zum Ordner der Logdateien.
FileListingOrder [▶ 153]	Auflistung der Dateien nach neuestem oder ältestem Stand entsprechend der Änderungszeit.

Kategorie: Datagrid

Name	Beschreibung
ShowRowNumbers [▶ 155]	Zeilennummer im Raster anzeigen.
RowHeight [▶ 155]	Höhe der Rasterzeilen.
TextFontSize [▶ 155]	Schriftgröße der Rasterzeilen.
TextFontSizeUnit [▶ 156]	Einheit von TextFontSize.

Kategorie: Header

Name	Beschreibung
HeaderHeight [▶ 154]	Definition der Höhe der Kopfzeile.
HeaderHeightUnit [▶ 154]	Definition der Höhe der Kopfzeile.
HeaderFontSize [▶ 154]	Die Schriftgröße der Kopfzeile.
HeaderFontSizeUnit [▶ 154]	Einheit der HeaderFontSize.
HeaderBackgroundColor [▶ 153]	Definition der Hintergrundfarbe der Kopfzeile.
HeaderFontColor [▶ 154]	Definition der Textfarbe der Kopfzeile.

4.16.1.1 EndDate

Legt das Datum der Datei fest, die zuletzt aufgelistet werden soll (definiert als ISO-Zeitstring).

Schema: tchmi:general#/definitions/DATE

Attribut-Getter: getEndDate

Attribut-Setter: setEndDate

Verfügbar: ab Version 12.12.0

4.16.1.2 FileListingOrder

Listet die neuesten oder ältesten Dateien zuerst auf, entsprechend der Änderungszeit der Dateien.

Schema: tchmi:framework#/definitions/UserLoggingViewer.ListOrder

Attribut-Getter: getFileListingOrder

Attribut-Setter: setFileListingOrder

Verfügbar: ab Version 12.12.0

4.16.1.3 HeaderBackgroundColor

Legt die Hintergrundfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderBackgroundColor

Attribut-Setter: setHeaderBackgroundColor

Verfügbar: ab Version 12.12.0

4.16.1.4 HeaderFontColor

Legt die Textfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderFontColor

Attribut-Setter: setHeaderFontColor

Verfügbar: ab Version 12.12.0

4.16.1.5 HeaderFontSize

Die Schriftgröße der Kopfzeile. Wenn Prozent als Einheit festgelegt ist, ist die Kopfzeilengröße relativ zur Schriftgröße des übergeordneten Elements.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderFontSize

Attribut-Setter: setHeaderFontSize

Verfügbar: ab Version 12.12.0

4.16.1.6 HeaderFontSizeUnit

Die Einheit der Schriftgröße der Kopfzeile (px, %).

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderFontSizeUnit

Attribut-Setter: setHeaderFontSizeUnit

Verfügbar: ab Version 12.12.0

4.16.1.7 HeaderHeight

Legt die Höhe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderHeight

Attribut-Setter: setHeaderHeight

Verfügbar: ab Version 12.12.0

4.16.1.8 HeaderHeightUnit

Legt die Einheit der Kopfzeilenhöhe fest:

- Pixel für eine absolute Größe
- Prozent für eine relative Größe

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderHeightUnit

Attribut-Setter: setHeaderHeightUnit

Verfügbar: ab Version 12.12.0

4.16.1.9 Path

Pfad zum Ordner mit den Logdateien.

Schema: tchmi:general#/definitions/String

Attribut-Getter: getPath

Attribut-Setter: setPath

Verfügbar: ab Version 12.12.0

4.16.1.10 RowHeight

Die Höhe der Rasterzeilen.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getRowHeight

Attribut-Setter: setRowHeight

Verfügbar: ab Version 12.12.0

4.16.1.11 SelectedIndex

Der Index der CSV-Datei, die ausgewählt und auf dem Control angezeigt wird.

Schema: tchmi:general#/definitions/DATE

Attribut-Getter: getSelectedIndex

Attribut-Setter: setSelectedIndex

Verfügbar: ab Version 12.12.0

4.16.1.12 ShowRowNumbers

Mit der Einstellung `TRUE` werden Zeilennummern im Raster angezeigt.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getShowRowNumbers

Attribut-Setter: setShowRowNumbers

Verfügbar: ab Version 12.12.0

4.16.1.13 StartDate

Legt das Datum der Datei fest, die zuerst aufgelistet werden soll (definiert als ISO-Zeitstring).

Schema: tchmi:general#/definitions/DATE

Attribut-Getter: getStartDate

Attribut-Setter: setStartDate

Verfügbar: ab Version 12.12.0

4.16.1.14 TextFontSize

Die Schriftgröße der Rasterzeilen.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getTextFontSize

Attribut-Setter: setTextFontSize

Verfügbar: ab Version 12.12.0

4.16.1.15 TextFontSizeUnit

Die Einheit der Schriftgröße der Rasterzeile.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getTextFontSizeUnit

Attribut-Setter: setTextFontSizeUnit

Verfügbar: ab Version 12.12.0

4.16.2 Dialoge

4.16.2.1 Spalten bearbeiten

Filter columns					
	Name	Visible	Width		
1	Timestamp	<input checked="" type="checkbox"/>	15	%	▼
2	Page	<input checked="" type="checkbox"/>	15	%	▼
3	Name	<input checked="" type="checkbox"/>	300	px	▼
4	Old	<input checked="" type="checkbox"/>	12	%	▼
5	New	<input checked="" type="checkbox"/>	12	%	▼
6	User	<input checked="" type="checkbox"/>	12	%	▼



Der Benutzer kann in diesem Pop-up die anzuzeigenden Spalten auswählen und die Breite jeder Spalte einstellen.

4.17 VariableViewControl

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

Das VariableViewControl kann als Dashboard verwendet werden, um Live-Werte von mehreren Variablen auf dem Bildschirm zu visualisieren.

Der Benutzer kann die Konfiguration während der Live-Sitzung direkt auf dem Client-Bildschirm ändern und entscheiden, welche Variablen angezeigt werden sollen.

Es ermöglicht auch die individuelle Positionierung jedes Wertes, um die Variablen besser in Gruppen organisieren zu können.



4.17.1 Attribute

Kategorie: Colors

Name	Beschreibung
TileBackgroundColor [►_160]	Legt die Kachel-Hintergrundfarbe für jede Kachel fest.
LabelBackgroundColor [►_160]	Legt die Hintergrundfarbe der Beschriftung für jede Kachel fest.
LabelColor [►_160]	Legt die Farbe des Beschriftungstextes für jede Kachel fest.
BorderColor [►_158]	Farbe für den Kachelrand.

Kategorie: TileDesign

Name	Beschreibung
LabelFontSize [► 160]	Legt die Schriftgröße der Beschriftung für jede Kachel fest.
LabelFontSizeUnit [► 160]	Einheit des Attributs LabelFontSize [► 160].
ValueFontSize [► 162]	Legt die Schriftgröße des Wertes fest, der in jeder Kachel angezeigt wird.
ValueFontSizeUnit [► 162]	Wert Schriftgrößeneinheit für jede Kachel.
TileWidth [► 161]	Legt die Kachelbreite für jede Kachel im Control fest.
TileWidthUnit [► 161]	Einheit der Kachelbreite.
TileHeight [► 160]	Legt die Kachelhöhe für jede Kachel im Control fest.
TileHeightUnit [► 161]	Höheneinheit der Kacheln.
TileSpacing [► 161]	Legt den Abstand zwischen den Kacheln fest.
TileSpacingUnit [► 161]	Einheit des Kachelabstands.
BorderRadius [► 159]	Legt den Rahmenradius für jede Kachel im Control fest.
BorderWidth [► 159]	Breite des Rahmens für jede Kachel.
BorderStyle [► 159]	Legt den Stil des Rahmens für jede Kachel fest. Gültige Werte sind <code>Solid</code> , <code>Dashed</code> , <code>Dotted</code> und <code>None</code> .

Kategorie: ConfigPopupDesign

Name	Beschreibung
Config PopUp Font Size [► 159]	Legt die Schriftgröße für die Gruppen- und Variablenliste fest, die im Konfigurations-Popup-Fenster angezeigt wird.
Config PopUp Font SizeUnit [► 159]	Einheit des Attributs Config PopUp Font Size [► 159].

Kategorie: Common

Name	Beschreibung
Variable View Mode [► 164]	Einzeilige oder mehrzeilige Ansicht des Controls.
Variable Collection [► 162]	Liste aller Variablen, die vom Benutzer für die Anzeige auf dem Bildschirm ausgewählt werden können.
Active Config [► 158]	Liste der Variablen, die beim Start auf dem Control angezeigt werden.

4.17.1.1 ActiveConfig

Liste der Variablen, die beim Start auf dem Control angezeigt werden.

Schema: tchmi:framework#/definitions/VariableConfigList

Attribut-Getter: getActiveConfig

Attribut-Setter: setActiveConfig

Verfügbar: seit Version 12.6.0

4.17.1.2 BorderColor

Farbe für den Kachelrand.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getBorderColor

Attribut-Setter: setBorderColor

Verfügbar: seit Version 12.6.0

4.17.1.3 **BorderStyle**

Legt den Stil des Rahmens für jede Kachel fest. Gültige Werte sind `Solid`, `Dashed`, `Dotted` und `None`.

Schema: tchmi:framework#/definitions/BorderStyle

Attribut-Getter: getBorderStyle

Attribut-Setter: setBorderStyle

Verfügbar: seit Version 12.6.0

4.17.1.4 **BorderRadius**

Legt den Rahmenradius für jede Kachel im Control fest.

Schema: tchmi:framework#/definitions/BorderRadius

Attribut-Getter: getBorderRadius

Attribut-Setter: setBorderRadius

Verfügbar: seit Version 12.6.0

4.17.1.5 **BorderWidth**

Breite des Rahmens für jede Kachel.

Schema: tchmi:framework#/definitions/BorderWidth

Attribut-Getter: getBorderWidth

Attribut-Setter: setBorderWidth

Verfügbar: seit Version 12.6.0

4.17.1.6 **ConfigPopUpFontSize**

Legt die Schriftgröße für die Gruppen- und Variablenliste fest, die im Konfigurations-Popup-Fenster angezeigt wird.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getConfigPopUpFontSize

Attribut-Setter: setConfigPopUpFontSize

Verfügbar: seit Version 12.6.0

4.17.1.7 **ConfigPopUpFontSizeUnit**

Einheit der Schriftgröße für das Konfigurations-Popup-Fenster.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getConfigPopUpFontSizeUnit

Attribut-Setter: setConfigPopUpFontSizeUnit

Verfügbar: seit Version 12.6.0

4.17.1.8 LabelBackgroundColor

Legt die Hintergrundfarbe der Beschriftung für jede Kachel fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getLabelBackgroundColor

Attribut-Setter: setLabelBackgroundColor

Verfügbar: seit Version 12.6.0

4.17.1.9 LabelColor

Legt die Farbe des Beschriftungstextes für jede Kachel fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getLabelColor

Attribut-Setter: setLabelColor

Verfügbar: seit Version 12.6.0

4.17.1.10 LabelFontSize

Legt die Schriftgröße der Beschriftung für jede Kachel fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getLabelFontSize

Attribut-Setter: setLabelFontSize

Verfügbar: seit Version 12.6.0

4.17.1.11 LabelFontSizeUnit

Die Einheit für die Schriftgröße der Beschriftung.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getLabelFontSizeUnit

Attribut-Setter: setLabelFontSizeUnit

Verfügbar: seit Version 12.6.0

4.17.1.12 TileBackgroundColor

Legt die Kachel-Hintergrundfarbe für jede Kachel fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getTileBackgroundColor

Attribut-Setter: setTileBackgroundColor

Verfügbar: seit Version 12.6.0

4.17.1.13 TileHeight

Legt die Kachelhöhe für jede Kachel im Control fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getTileHeight

Attribut-Setter: setTileHeight

Verfügbar: seit Version 12.6.0

4.17.1.14 TileHeightUnit

Höheneinheit der Kacheln.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getTileHeightUnit

Attribut-Setter: setTileHeightUnit

Verfügbar: seit Version 12.6.0

4.17.1.15 TileSpacing

Legt den Abstand zwischen den Kacheln fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getTileSpacing

Attribut-Setter: setTileSpacing

Verfügbar: seit Version 12.6.0

4.17.1.16 TileSpacingUnit

Einheit des Kachelabstands.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getTileSpacingUnit

Attribut-Setter: setTileSpacingUnit

Verfügbar: seit Version 12.6.0

4.17.1.17 TileWidth

Legt die Kachelbreite für jede Kachel im Control fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getTileWidth

Attribut-Setter: setTileWidth

Verfügbar: seit Version 12.6.0

4.17.1.18 TileWidthUnit

Einheit der Kachelbreite.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getTileWidthUnit

Attribut-Setter: setTileWidthUnit

Verfügbar: seit Version 12.6.0

4.17.1.19 ValueFont Size

Legt die Schriftgröße des Wertes fest, der innerhalb jeder Kachel angezeigt wird.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getValueFontSize

Attribut-Setter: setValueFontSize

Verfügbar: seit Version 12.6.0

4.17.1.20 ValueFontSizeUnit

Wert Schriftgrößeneinheit für jede Kachel.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getValueFontSizeUnit

Attribut-Setter: setValueFontSizeUnit

Verfügbar: seit Version 12.6.0

4.17.1.21 VariableCollection

Wenn ein Benutzer eine neue Variable hinzufügen möchte, die auf dem VariableViewControl überwacht werden soll, benötigt das Control die folgenden Informationen für die neue Variable:

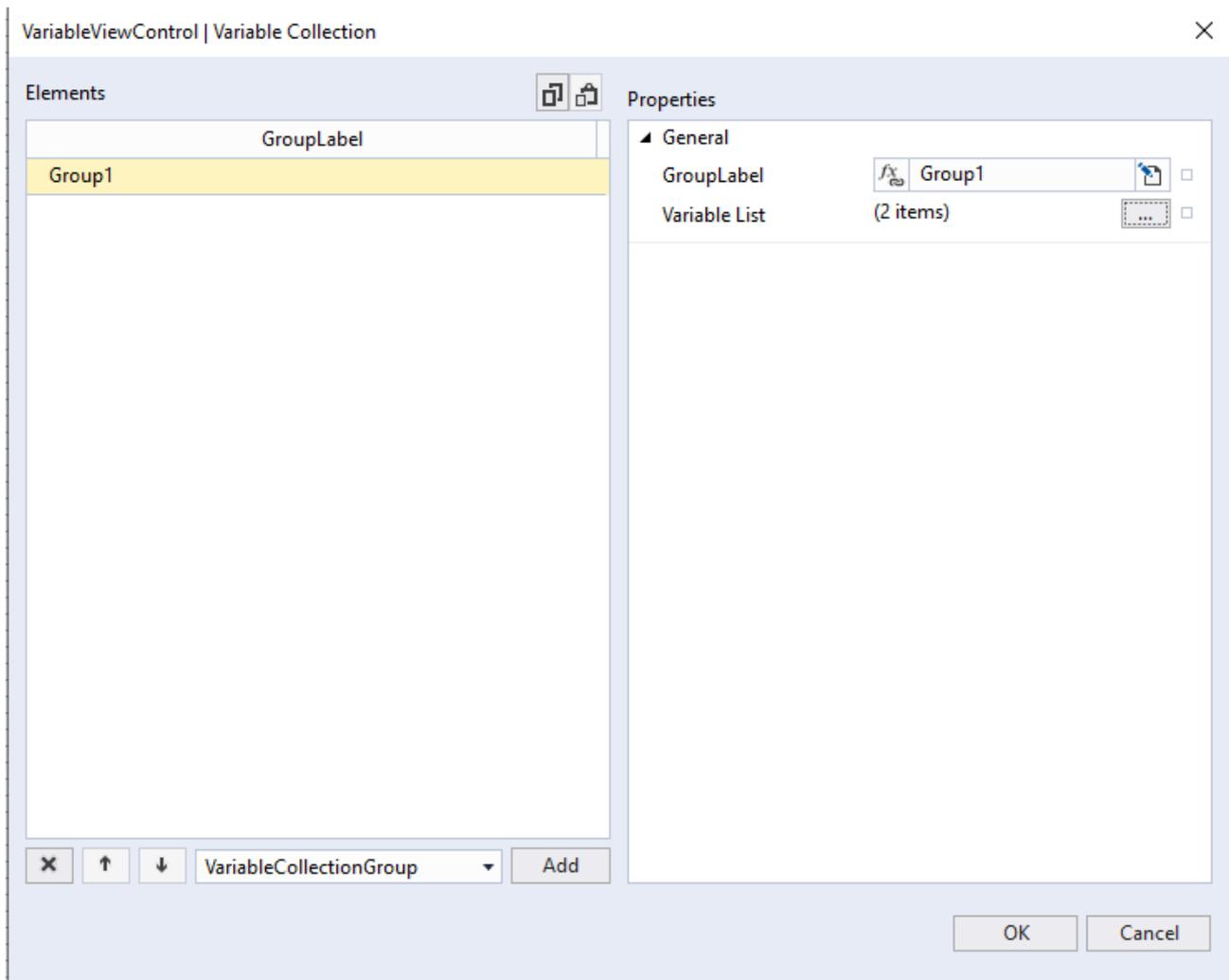
- Pfad des Symbols zum Lesen des Wertes.
- Beschriftungstext, der ein lokalisiertes Symbol oder ein Function Binding sein kann.
- Einheitengruppe dieser Variablen, um eine Einheitenumrechnung für die korrekte Anzeige der Werte zu ermöglichen.

Diese Informationen können mit Hilfe des Attributs Variable Collection aufbereitet werden.

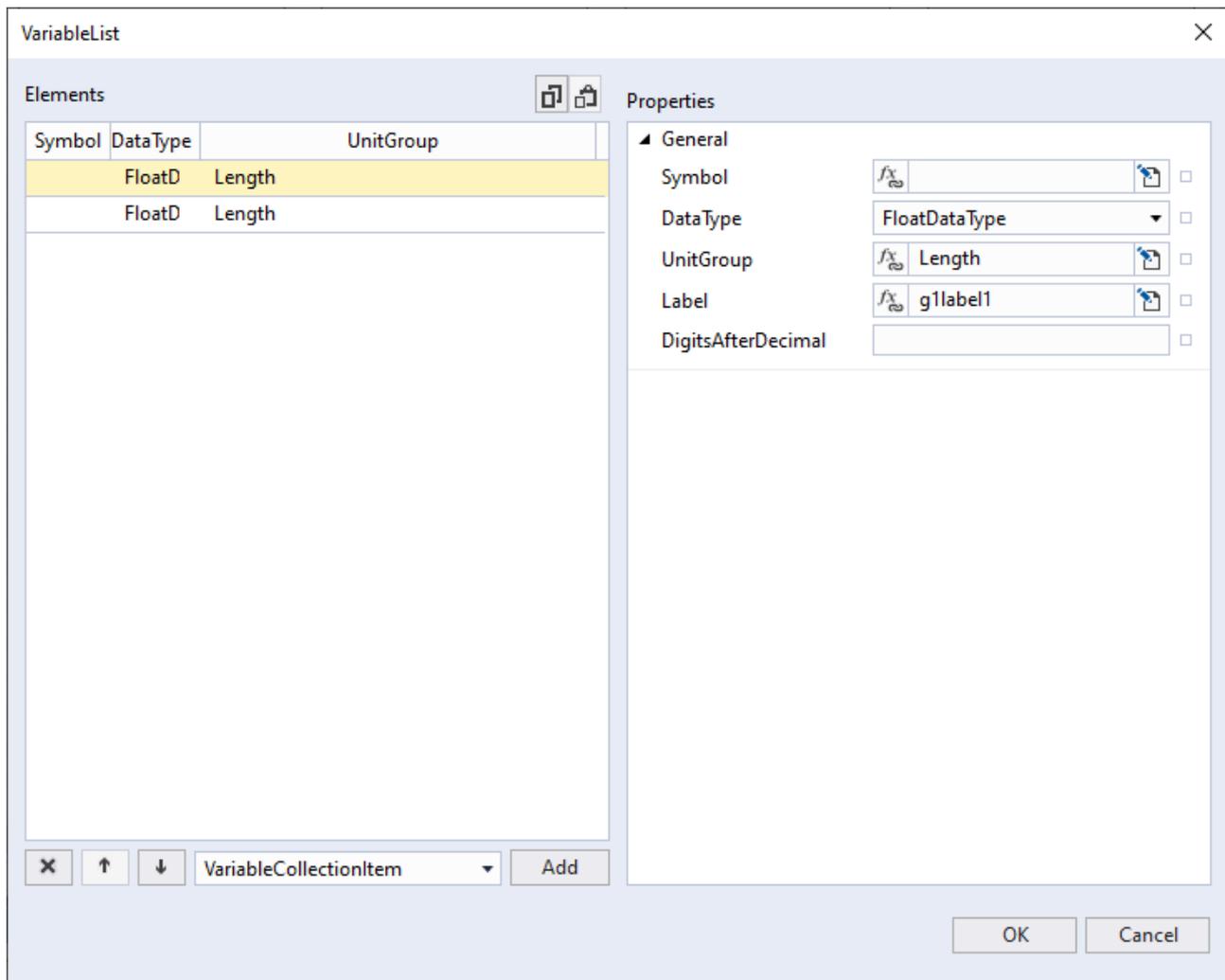
Das Attribut Variable Collection wird auch vom Konfigurationsdialog verwendet, der auf dem Control erscheint.

Um dem Benutzer die Auswahl einer geeigneten Variable zu erleichtern, sind alle Variablen in mehrere Gruppen unterteilt.

Der Name für jede Gruppe wird über das Attribut GroupLabel im Objekteditor-Dialog in der Engineering-Umgebung vom Benutzer konfiguriert.



Jede Gruppe von Variablen hat Group Label und Variable List.



Jede Variablenliste kann mehrere Variablen enthalten, wobei jede Variable die folgenden Eigenschaften hat:

- **Symbol:** Binding-Pfad zu dem Symbol, das auf dem Bildschirm angezeigt werden soll.
- **Data Type:** Zur besseren Darstellung kann der Benutzer den Datentyp als `FloatDataType`, `IntDataType` oder `BoolDataType` definieren.
- **Unit Group:** Einheitengruppe des angezeigten Symbols.
- **Label:** Text, der als Kennung der Kachel angezeigt wird.
- **DigitsAfterDecimal:** Die Anzahl der Dezimalstellen aller numerischen Werte, die auf dem Control angezeigt werden.

Liste aller Variablen, die vom Benutzer für die Anzeige auf dem Bildschirm ausgewählt werden können.

Schema: `tchmi:framework#/definitions/VariableCollection`

Attribut-Getter: `getVariableCollection`

Attribut-Setter: `setVariableCollection`

Verfügbar: seit Version 12.6.0

4.17.1.22 VariableViewMode

Einzeilige oder mehrzeilige Ansicht des Controls.

Schema: `tchmi:framework#/definitions/VariableViewMode`

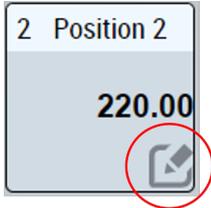
Attribut-Getter: `getVariableViewMode`

Attribut-Setter: `setVariableViewMode`

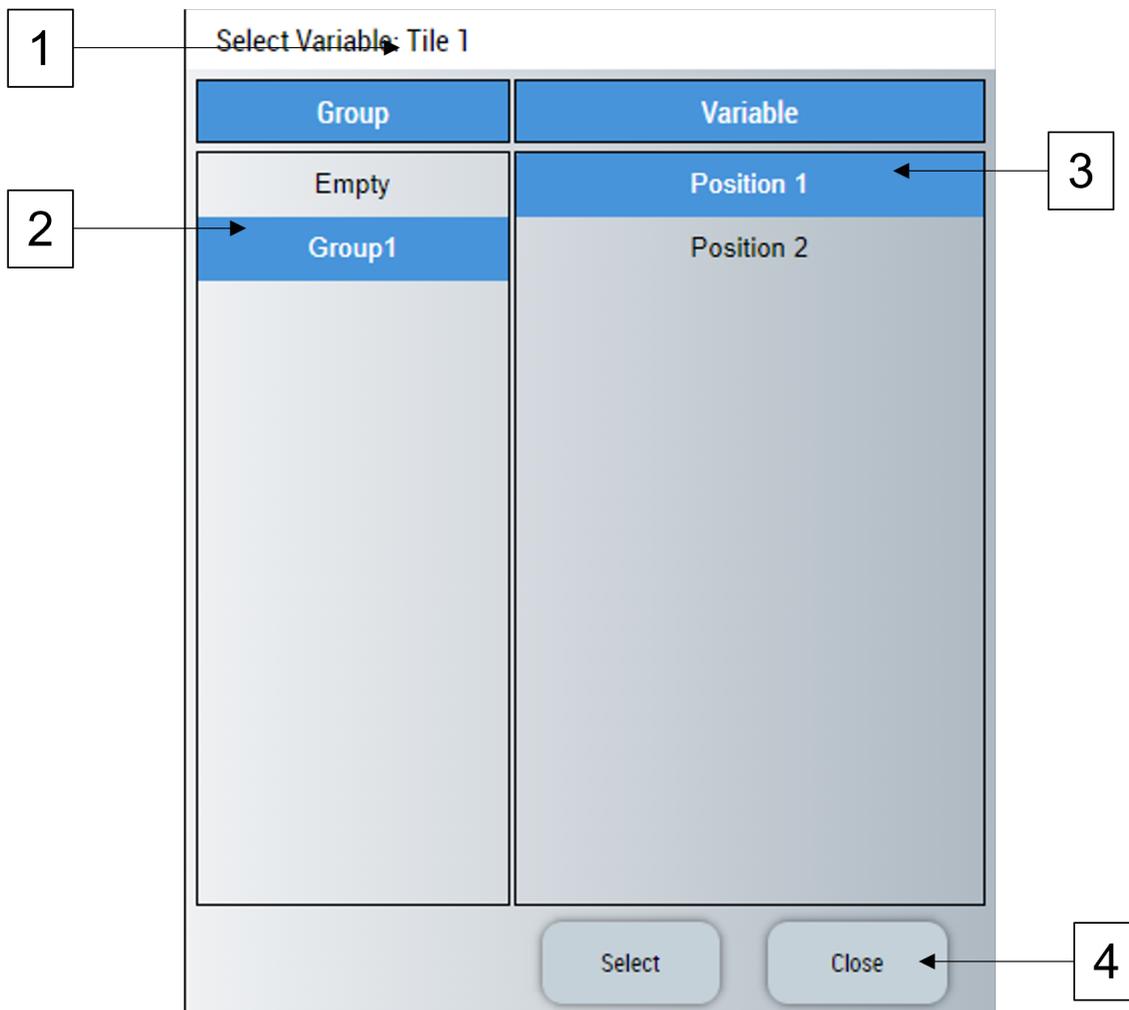
Verfügbar: seit Version 12.6.0

4.17.2 Dialoge

4.17.2.1 Angezeigte Variable einer Kachel ändern



Durch Klicken auf die Schaltfläche "Edit" auf einer der Kacheln des Controls kann der Benutzer ein Konfigurations-Popup öffnen. In diesem Popup-Fenster kann der Benutzer die Variable ändern, die auf der ausgewählten Kachel angezeigt werden soll.



1. Die Kachelnummer der konfigurierten Kachel.
2. Die Gruppe kann entsprechend der Konfiguration im Attribut Variable Collection [▶ 162] ausgewählt werden.
3. Das Symbol kann durch Auswahl des mit dem Symbol verbundenen Beschreibungstextes ausgewählt werden.
4. Der Benutzer kann auf die Schaltflächen Select oder Cancel klicken, um Änderungen zu bestätigen oder zu verwerfen.

4.18 Funktionen

4.18.1 UnitConversion

4.18.1.1 ConvToDisplayUnit

```
function ConvToDisplayUnit(GroupID: string, SrcVal: number, RefVal: number): number
```

Konvertiert den gegebenen SrcVal-Wert in der Systemeinheit in den entsprechenden Wert für die Anzeigeeinheit, wobei die angegebene Gruppen-ID und der Referenzwert verwendet werden.

Parameter

Name	Typ	Beschreibung
GroupID	string	Einheitengruppe des Quellwerts.
SrcVal	number	Zu konvertierender Wert.
RefVal	number	Referenzwert, der 100% ergibt, wenn die Anzeige- oder Systemeinheit Prozent ist.

Rückgabewert

Typ	Beschreibung
number	Ergebnis in der Anzeigeeinheit der gegebenen Einheitengruppe.

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

4.18.1.2 ConvToSystemUnit

```
function ConvToSystemUnit(GroupID: string, SrcVal: number, RefVal: number): number
```

Konvertiert einen gegebenen Wert in der Anzeigeeinheit in einen entsprechenden Wert in der Systemeinheit, wobei die Parameter GroupID und RefVal verwendet werden.

Parameter

Name	Typ	Beschreibung
GroupID	string	Einheitengruppe des Quellwerts.
SrcVal	number	Zu konvertierender Wert.
RefVal	number	Referenzwert, der 100% ergibt, wenn die Anzeige- oder Systemeinheit Prozent ist.

Rückgabewert

Typ	Beschreibung
number	Ergebnis in der Systemeinheit der gegebenen Einheitengruppe.

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

4.18.1.3 GetConfiguredPercentUnitID

```
GetConfiguredPercentUnitID(): string
```

Gibt die konfigurierte PercentUnit-ID zurück.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
string	Unit_Id-Eigenschaft der in unit_config.json [► 45] konfigurierten PercentUnit

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

4.18.1.4 GetDisplayUnitForUnitGroup

```
function GetDisplayUnitForUnitGroup(ctx: Context, GroupID: string)
```

Ermittelt die konfigurierte ID der Anzeigeeinheit für die angegebene Einheitengruppe und wartet darauf, dass das Configurator-Control die Einheitenkonfiguration aus der JSON-Datei lädt.

Parameter

Name	Typ	Beschreibung
ctx	TcHmi.Context	<ul style="list-style-type: none"> • Kontext-Objekt • Muss bei der Verwendung in einem Trigger oder einem Funktionsausdruck nicht vom Anwender angegeben werden. Die Übergabe und die Auswertung erfolgt in diesem Fall über den entsprechenden Parser. • Wird vom Aufrufer verwendet um dem Framework bei der Ausführung Erfolg oder Fehlschlag zu signalisieren. • Bei direkter Verwendung der Funktion in JavaScript/ TypeScript muss das Kontext-Objekt beim Aufruf übergeben und ggf. ausgewertet werden.
GroupID	string	ID der Einheitengruppe.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

4.18.1.5 GetDispUnitText

```
function GetDispUnitText(GroupID: string): string
```

Gibt die UnitText-Eigenschaft der Einheit zurück, die als Anzeigeeinheit für den angegebenen Einheitengruppen-Parameter konfiguriert ist.

Parameter

Name	Typ	Beschreibung
GroupID	string	ID der Einheitengruppe.

Rückgabewert

Typ	Beschreibung
string	UnitText-Eigenschaft der Einheit, die als Anzeigeeinheit für die gegebene Einheitengruppe festgelegt ist.

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

4.18.1.6 GetSystemUnitForUnitGroup

```
function GetSystemUnitForUnitGroup (ctx: Context, GroupID: string)
```

Ruft die konfigurierte Systemeinheit-ID für die angegebene Einheitengruppe ab und wartet darauf, dass das Configurator-Control die Einheitenkonfiguration aus der JSON-Datei lädt.

Parameter

Name	Typ	Beschreibung
ctx	TcHmi.Context	<ul style="list-style-type: none"> Kontext-Objekt Muss bei der Verwendung in einem Trigger oder einem Funktionsausdruck nicht vom Anwender angegeben werden. Die Übergabe und die Auswertung erfolgt in diesem Fall über den entsprechenden Parser. Wird vom Aufrufer verwendet um dem Framework bei der Ausführung Erfolg oder Fehlschlag zu signalisieren. Bei direkter Verwendung der Funktion in JavaScript/TypeScript muss das Kontext-Objekt beim Aufruf übergeben und ggf. ausgewertet werden.
GroupID	string	ID der Einheitengruppe.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

4.18.1.7 GetUnitConverted

```
function GetUnitConverted(SrcVal: number, SrcUnitID: string, TargetUnitID: string, RefVal: number): number
```

Konvertiert den Wert in der Quellmaßeinheit in die Zielmaßeinheit. Wenn keine gültige Umrechnungsformel in [unit_config.json](#) [► 45] konfiguriert ist, gibt die Funktion den Quellwert zurück, ohne ihn zu ändern.

Parameter

Name	Typ	Beschreibung
SrcVal	number	Zu konvertierender Wert.
SrcUnitID	string	ID der Einheitengruppe, in der der Quellwert berechnet wird.
TargetUnitID	string	Einheitengruppen-ID, in die das Ergebnis berechnet werden soll.
RefVal	number	Referenzwert, der 100% ergibt, wenn die Anzeige- oder Systemeinheit Prozent ist.

Rückgabewert

Typ	Beschreibung
number	Ergebnis der Einheitenumrechnung.

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

4.18.1.8 GetUnitTextForUnit

```
function GetUnitTextForUnit(ctx: Context, UnitID: string)
```

Ruft die Unit-Text-Eigenschaft der durch den UnitID-Parameter identifizierten Einheit ab und wartet darauf, dass das Configurator-Control die Einheitenkonfiguration aus der Datei [unit_config.json](#) [► 45] lädt.

Parameter

Name	Typ	Beschreibung
ctx	TcHmi.Context	<ul style="list-style-type: none"> • Kontext-Objekt • Muss bei der Verwendung in einem Trigger oder einem Funktionsausdruck nicht vom Anwender angegeben werden. Die Übergabe und die Auswertung erfolgt in diesem Fall über den entsprechenden Parser. • Wird vom Aufrufer verwendet um dem Framework bei der Ausführung Erfolg oder Fehlschlag zu signalisieren. • Bei direkter Verwendung der Funktion in JavaScript/ TypeScript muss das Kontext-Objekt beim Aufruf übergeben und ggf. ausgewertet werden.
UnitID	string	Unit-ID der Einheit, deren Texteingenschaft ausgelesen werden soll.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

4.18.1.9 SetDisplayUnitForUnitGroup

`SetDisplayUnitForUnitGroup(ctx: Context | null, GroupID: string, UnitID: string)`

Legt die durch UnitID identifizierte Einheit als Anzeigeeinheit für die durch GroupID identifizierte Einheitengruppe fest.

Parameter

Name	Typ	Beschreibung
ctx	<u>TcHmi.Context</u>	<ul style="list-style-type: none"> • Kontext-Objekt • Muss bei der Verwendung in einem Trigger oder einem Funktionsausdruck nicht vom Anwender angegeben werden. Die Übergabe und die Auswertung erfolgt in diesem Fall über den entsprechenden Parser. • Wird vom Aufrufer verwendet um dem Framework bei der Ausführung Erfolg oder Fehlschlag zu signalisieren. • Bei direkter Verwendung der Funktion in JavaScript/ TypeScript muss das Kontext-Objekt beim Aufruf übergeben und ggf. ausgewertet werden.
GroupID	string	ID der Einheitengruppe.
UnitID	string	Unit-ID, die als Anzeigeeinheit festgelegt werden soll.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

4.18.1.10 SetSystemUnitForUnitGroup

`SetSystemUnitForUnitGroup(ctx: Context | null, GroupID: string, UnitID: string)`

Setzt die durch den Parameter UnitID identifizierte Einheit als Systemeinheit für die durch den Parameter GroupID identifizierte Einheitengruppe.

Parameter

Name	Typ	Beschreibung
ctx	<u>TcHmi.Context</u>	<ul style="list-style-type: none"> • Kontext-Objekt

Name	Typ	Beschreibung
		<ul style="list-style-type: none"> Muss bei der Verwendung in einem Trigger oder einem Funktionsausdruck nicht vom Anwender angegeben werden. Die Übergabe und die Auswertung erfolgt in diesem Fall über den entsprechenden Parser. Wird vom Aufrufer verwendet um dem Framework bei der Ausführung Erfolg oder Fehlschlag zu signalisieren. Bei direkter Verwendung der Funktion in JavaScript/TypeScript muss das Kontext-Objekt beim Aufruf übergeben und ggf. ausgewertet werden.
GroupID	string	ID der Einheitengruppe.
UnitID	string	Unit-ID, die als Systemeinheit festgelegt werden soll.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Controls

Verfügbar: seit Version 12.6.0

4.18.2 LogUserChanges

Funktionen in dieser Kategorie führen das Loggen von Benutzeränderungen in eine CSV-Datei aus, indem sie die Methode 'Logg' von FB_UserLoggingHmi aufrufen.

Das Server-Symbol auf FB_UserLoggingHmi wird im Attribut UserLoggerSymbol [▶ 45] des Configurator-Controls konfiguriert.

4.18.2.1 LogUserChanges

```
function LogUserChanges(ParameterName: string, OldValue: string, NewValue: string, ContentID: string):void
```

Diese Funktion loggt die in ihren Parametern angegebenen Informationen in einer .CSV-Datei unter Verwendung des FB_UserLoggingHmi-Symbols von SPS, das im UserLoggerSymbol [▶ 45]-Attribut konfiguriert ist.

Parameter:

Name	Typ	Beschreibung
ParameterName	string	Name des vom Benutzer geänderten Parameters.
OldValue	string	Alter Wert des Parameters im String-Format.
NewValue	string	Neuer Wert des Parameters im String-Format.
ContentID	string	Id des Inhalts, an dem die Änderung vorgenommen wurde.

Rückgabewert:

Typ	Beschreibung
void	Kein Rückgabewert

4.18.2.2 LogUserChangesSymbol

```
function LogUserChangesSymbol (ParameterSymbol: any, OldValue: string, NewValue: string, ContentID: string): void
```

Diese Funktion loggt die in ihren Parametern angegebenen Informationen in einer .CSV-Datei unter Verwendung des FB_UserLoggingHmi-Symbols von SPS, das im [UserLoggerSymbol \[▶ 45\]](#)-Attribut konfiguriert ist.

Der Symbolname von ParameterSymbol wird in der Logdatei gespeichert.

Parameter:

Name	Typ	Beschreibung
ParameterSymbol	tchmi:general#/definitions/Symbol	Symbol für den vom Benutzer geänderten Parameter.
OldValue	string	Alter Wert des Parameters im String-Format.
NewValue	string	Neuer Wert des Parameters im String-Format.
ContentID	string	Id des Inhalts, an dem die Änderung vorgenommen wurde.

Rückgabewert:

Typ	Beschreibung
void	Kein Rückgabewert

5 Functions

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Das NuGet-Paket **Beckhoff.TwinCAT.HMI.Plastic.Functions** kann über das Paketverwaltungssystem NuGet im TwinCAT HMI Projekt installiert werden.

5.1 Axes

5.1.1 ChangelInputBoxUnit

```
function ChangeInputBoxUnit (InputBox: TcHmi.Controls.Plastic.InputBox): void
```

Die Funktion ChangelInputBoxUnit wechselt die Einheitengruppe eines InputBox-Controls in eine Winkel-Einheitengruppe um, wenn es sich bei dem Symbol im TextVal-Property des InputBox-Controls um einen Wert aus einem FB_Axis Baustein der Tc3_PlasticBaseApplication Bibliothek handelt und die Achse eine Transformationsachse ist.

Einheitengruppe	Winkel-Einheitengruppe
Length	Angular
Velocity	AngularVelocity
Acceleration	AngularAcceleration
Jerk	AngularJerk

Diese Funktion wird standardmäßig nur von der UpdateAxesHomingUnits [▶ 174] Funktion verwendet.

Parameter

Name	Typ	Beschreibung
InputBox	TcHmi.Controls.Plastic.InputBox [▶ 85]	Das Control, dessen <u>UnitGroup</u> verändert werden soll.

Rückgabewert

Typ	Beschreibung
void	kein Rückgabewert

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

5.1.2 OperateWtcWithHardware

```
function OperateWtcWithHardware (FB_CP22xx_1025Hmi: Symbol, CurveEditor: TcHmi.Controls.Plastic.CurveEditorControl): void
```

Die Funktion OperateWtcWithHardware ermöglicht das Einstellen der Fixpunkte des CurveEditor-Controls mithilfe eines Drehimpulsgebers. Dabei wechselt die Funktion durch ein Drücken auf den Drehimpulsgeber jeweils den Einstellmodus.

Modus	Verhalten beim Drehen des Drehimpulsgebers
Index	Selektion der bereits erzeugten Fixpunkte.
Value	Änderung des Value-Wertes des ausgewählten Fixpunktes.
Base	Änderung des Base-Wertes des ausgewählten Fixpunktes.

Parameter

Name	Typ	Beschreibung
FB_CP22xx_1025Hmi	TcHmi.Symbol	Das Symbol vom Datentyp FB_CP22xx_1025Hmi.
CurveEditor	TcHmi.Controls.Plastic.CurveEditorControl [► 48]	Das CurveEditor-Control das mithilfe des Drehimpulsgebers bearbeitet werden soll.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.12.0

Verwendung: TC3 Plastic Application HMI Projekte

5.1.3 UpdateAxesHomingUnits

```
function UpdateAxesHomingUnits(ContentId: string): void
```

Die Funktion UpdateAxesHomingUnits wechselt die Einheitengruppen von InputBox-Controls in eine Winkel-Einheitengruppe um, wenn es sich bei dem Symbol im TextVal-Property des InputBox-Controls um einen Wert aus einem FB_Axis Baustein der Tc3_PlasticBaseApplication Bibliothek handelt und die Achse eine Transformationsachse ist. Die Funktion ist für die Verwendung auf einer TwinCAT HMI Seite gedacht, auf der die Referenzfahrt-Parameter einer Achse mit InputBox-Controls verbunden sind. Die verwendeten Symbole sind im FB_Homing Baustein der Tc3_PlasticBaseApplication Bibliothek zu finden.

InputBox-Controls mit den folgenden IDs werden durch die Funktion bearbeitet, dabei ist ContentId der Eingabeparameter der Funktion:

- ContentId + _HomingBlock_Velocity_InputBox
- ContentId + _HomingBlock_Acceleration_InputBox
- ContentId + _HomingBlock_Deceleration_InputBox
- ContentId + _HomingBlock_Jerk_InputBox
- ContentId + _HomingBlock_DistanceLimit_InputBox
- ContentId + _HomingBlock_SetPosition_InputBox
- ContentId + _HomingBlockDetect_Velocity_InputBox
- ContentId + _HomingBlock_DetectionVelocityLimit_InputBox
- ContentId + _HomingBlockDetect_Acceleration_InputBox
- ContentId + _HomingBlockDetect_Deceleration_InputBox
- ContentId + _HomingBlockDetect_Jerk_InputBox
- ContentId + _HomingBlockDetect_DistanceLimit_InputBox
- ContentId + _HomingBlockDetect_RecordedPosition_InputBox
- ContentId + _HomingBlockDetect_DetectionVelocityLimit_InputBox
- ContentId + _HomingFinish_Distance_InputBox
- ContentId + _HomingFinish_Velocity_InputBox
- ContentId + _HomingFinish_Acceleration_InputBox
- ContentId + _HomingFinish_Deceleration_InputBox

Einheitengruppe	Winkel-Einheitengruppe
Length	Angular

Einheitengruppe	Winkel-Einheitengruppe
Velocity	AngularVelocity
Acceleration	AngularAcceleration
Jerk	AngularJerk

Parameter

Name	Typ	Beschreibung
ContentId	<u>string</u>	Der Text mit denen die IDs der InputBox-Controls beginnen.

Rückgabewert

Typ	Beschreibung
void	kein Rückgabewert

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

5.2 CodeBehind

5.2.1 DisableContextMenu

Die CodeBehind-Funktion DisableContextMenu deaktiviert das Kontextmenü, so dass kein Rechtsklick funktioniert.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

5.2.2 SlidingFunction

Die CodeBehind-Funktion SlidingFunction ermöglicht das Sliden von Contents, die sich in einer TcHmiRegion mit der ID Portrait_SliderContent_Region befinden. Dabei wird ein TcHmiGrid-Control auf diesem Content benötigt, bei dem sich die ID aus der ID des Contents und der Endung _Grid zusammensetzt. Die Funktion ändert je nach Sliding-Bewegung das Left-Property des TcHmiGrid-Controls auf 0, -1024 oder -2048. Somit kann eine Sliding-Funktion für bis zu drei Unterseiten mit je einer Breite von 1024px realisiert werden. Bei gewünschten drei Unterseiten muss das TcHmiGrid-Control dementsprechend eine Breite von 3052px haben.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.8.0

Verwendung: TC3 Plastic Application HMI Projekte

5.3 Frame

5.3.1 HideSliderArea

```
function HideSliderArea(SliderContentRegion: TcHmi.Controls.System.TcHmiRegion, MainGrid:
TcHmi.Controls.System.TcHmiGrid, SliderTabContainer: TcHmi.Controls.System.TcHmiContainerControl):
void
```

Die Funktion HideSliderArea schließt den Sliderbereich einer `.view`-Datei, wenn er zuvor angezeigt wurde.

Parameter

Name	Typ	Beschreibung
SliderContentRegion	<code>TcHmi.Controls.System.TcHmiRegion</code>	Das TcHmiRegion-Control, das als Sliderbereich dient.
MainGrid	<code>TcHmi.Controls.System.TcHmiGrid</code>	Das TcHmiGrid-Control, das die Hauptregion und die Sliderregion enthält.
SliderTabContainer	<code>TcHmi.Controls.System.TcHmiContainer</code>	Das TcHmiContainer-Control, das die Tabs des Sliderbereichs enthält, die zur Umschaltung des Sliderinhalts dienen.

Rückgabewert

Typ	Beschreibung
void	kein Rückgabewert

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

5.3.2 ShowSliderArea

```
function ShowSliderArea(SliderContentRegion: TcHmi.Controls.System.TcHmiRegion, MainGrid: TcHmi.Controls.System.TcHmiGrid, SliderTabContainer: TcHmi.Controls.System.TcHmiContainerControl): void
```

Die Funktion ShowSliderArea zeigt den Sliderbereich einer `.view`-Datei an, wenn er zuvor geschlossen war.

Parameter

Name	Typ	Beschreibung
SliderContentRegion	<code>TcHmi.Controls.System.TcHmiRegion</code>	Das TcHmiRegion-Control, das als Sliderbereich dient.
MainGrid	<code>TcHmi.Controls.System.TcHmiGrid</code>	Das TcHmiGrid-Control, das die Hauptregion und die Sliderregion enthält.
SliderTabContainer	<code>TcHmi.Controls.System.TcHmiContainer</code>	Das TcHmiContainer-Control, das die Tabs des Sliderbereichs enthält, die zur Umschaltung des Sliderinhalts dienen.

Rückgabewert

Typ	Beschreibung
void	kein Rückgabewert

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

5.3.3 ToggleSliderArea

```
function ToggleSliderArea(SliderContentRegion: TcHmi.Controls.System.TcHmiRegion, MainGrid:
TcHmi.Controls.System.TcHmiGrid, SliderTabContainer: TcHmi.Controls.System.TcHmiContainerControl):
void
```

Die Funktion ToggleSliderArea zeigt den Sliderbereich einer .view-Datei an, wenn er zuvor geschlossen war und schließt ihn, wenn er zuvor angezeigt wurde. Dazu werden die beiden Funktionen [HideSliderArea \[► 175\]](#) und [ShowSliderArea \[► 176\]](#) verwendet.

Parameter

Name	Typ	Beschreibung
SliderContentRegion	TcHmi.Controls.System.TcHmiRegion	Das TcHmiRegion-Control, das als Sliderbereich dient.
MainGrid	TcHmi.Controls.System.TcHmiGrid	Das TcHmiGrid-Control, das die Hauptregion und die Sliderregion enthält.
SliderTabContainer	TcHmi.Controls.System.TcHmiContainer	Das TcHmiContainer-Control, das die Tabs des Sliderbereichs enthält, die zur Umschaltung des Sliderinhalts dienen.

Rückgabewert

Typ	Beschreibung
void	kein Rückgabewert

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

5.3.4 TakeScreenshot

```
function TakeScreenshot(ContentName: string): void
```

Die Funktion TakeScreenShot wird zur Erstellung eines Screenshots benutzt. Der Übergabeparameter wird für die spezifische Benennung der Screenshot-Datei verwendet, ist jedoch für die Funktionsweise nicht zwingend erforderlich.

Parameter

Name	Typ	Beschreibung
ContentName	string	Der Pfad des aktuellen TcHmiContent, der dem Namen der Screenshot-Datei beigefügt werden soll. Ist nicht erforderlich.

Rückgabewert

Typ	Beschreibung
void	kein Rückgabewert

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

5.3.5 UpdateSliderContentRegion

```
function UpdateSliderContentRegion(MainContentRegion: TcHmi.Controls.System.TcHmiRegion,
SliderContentRegion: TcHmi.Controls.System.TcHmiRegion, MainContentSpecificToggleButton:
TcHmi.Controls.Beckhoff.TcHmiToggleButton, NavigationToggleButton:
TcHmi.Controls.Beckhoff.TcHmiToggleButton): void
```

Die Funktion UpdateSliderContentRegion wird verwendet, um je nach angezeigtem TcHmiContent in dem TcHmiRegion-Control für den Hauptcontent den passenden TcHmiContent in dem TcHmiRegion-Control für den Slidercontent reinzuladen, sofern das ToggleState-Property des zugehörigen TcHmiToggleButton-Controls (mit der ID MainContentSpecificToggleButton) auf Active steht. Dem TcHmiToggleButton-Control wird zusätzlich ein zugehöriges Icon als BackgroundImage-Property aus dem NuGet Paket Beckhoff.TwinCAT.HMI.Plastic.Images zugewiesen.

Hauptcontent Pfad	Slidercontent Pfad	Icon Pfad
Contents/Navigation/Axes/Blowpin/Blowpin.content	Contents/Slider/Axes/Blowpin/Blowpin_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Blowpin/Blowpin.svg
Contents/Navigation/Axes/Blowpin/Blowpin_Homing.content	Contents/Slider/Axes/Blowpin/Blowpin_Homing_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Blowpin/BlowpinHoming.svg
Contents/Navigation/Axes/Carriage/Carriage.content	Contents/Slider/Axes/Carriage/Carriage_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Carriage/Carriage.svg
Contents/Navigation/Axes/Carriage/Carriage_Homing.content	Contents/Slider/Axes/Carriage/Carriage_Homing_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Carriage/CarriageHoming.svg
Contents/Navigation/Axes/Clamp/Clamp.content	Contents/Slider/Axes/Clamp/Clamp_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Clamp/Clamp.svg
Contents/Navigation/Axes/Clamp/Clamp_Homing.content	Contents/Slider/Axes/Clamp/Clamp_Homing_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Clamp/ClampHoming.svg
Contents/Navigation/Extruder/Trend.content	Contents/Slider/Extruder/Extruder_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/Extruder.svg
Contents/Navigation/Parameters/Parameters_Blowpin.content	Contents/Slider/Parameters/Parameters_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersBlowpin.svg
Contents/Navigation/Parameters/Parameters_Carriage.content	Contents/Slider/Parameters/Parameters_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersCarriage.svg
Contents/Navigation/Parameters/Parameters_Clamp.content	Contents/Slider/Parameters/Parameters_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersClamp.svg
Contents/Navigation/Parameters/Parameters_CoExtruder.content	Contents/Slider/Parameters/Parameters_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersCoExtruder.svg
Contents/Navigation/Parameters/Parameters_MainExtruder.content	Contents/Slider/Parameters/Parameters_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersMainExtruder.svg
Contents/Navigation/Parameters/Parameters_Monitoring.content	Contents/Slider/Parameters/Parameters_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersMonitoring.svg
Contents/Navigation/Parameters/Parameters_Setpoints.content	Contents/Slider/Parameters/Parameters_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersSetpoints.svg
Contents/Navigation/Parameters/Parameters_Temperature.content	Contents/Slider/Parameters/Parameters_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersTemperature.svg
Contents/Navigation/Parameters/Parameters_Wtc.content	Contents/Slider/Parameters/Parameters_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersWtc.svg

Hauptcontent Pfad	Slidercontent Pfad	Icon Pfad
Contents/Navigation/System/RecipeManagement.content	Contents/Slider/System/RecipeManagement_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersRecipeManagement.svg
Contents/Navigation/Temperature/Temperature_Parameters.content	Contents/Slider/Temperature/Temperature_Parameters_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersTemperature.svg
Contents/Navigation/Temperature/Temperature_TimeScheduling.content	Contents/Slider/Temperature/Temperature_TimeScheduling_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/TemperatureTimeScheduling.svg
Contents/Navigation/Temperature/Temperature_Zones.content	Contents/Slider/Temperature/Temperature_Zones_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/TemperatureTimeScheduling.svg
Contents/Navigation/Wtc/Wtc.content	Contents/Slider/Wtc/Wtc_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Wtc/Wtc.svg
Contents/Navigation/Wtc/Wtc_Homing.content	Contents/Slider/Wtc/Wtc_Homing_Settings.content	Beckhoff.TwinCAT.HMI.Plastic.Images/Wtc/WtcHoming.svg

Parameter

Name	Typ	Beschreibung
MainContentRegion	<u>TcHmi.Controls.System.TcHmiRegion</u>	Das TcHmiRegion-Control, in dem der Hauptcontent angezeigt wird.
SliderContentRegion	<u>TcHmi.Controls.System.TcHmiRegion</u>	Das TcHmiRegion-Control, in dem der zugehörige Slidercontent des Hauptcontents angezeigt werden soll.
MainContentSpecificToggleButton	<u>TcHmi.Controls.Beckhoff.TcHmiToggleButton</u>	Das TcHmiToggleButton-Control, das für die Anzeige des Slidercontents im Sliderbereich dient.
NavigationToggleButton	<u>TcHmi.Controls.Beckhoff.TcHmiToggleButton</u>	Das TcHmiToggleButton-Control, das für die Anzeige der Navigationsseite im Sliderbereich dient.

Rückgabewert

Typ	Beschreibung
void	kein Rückgabewert

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

5.3.6 UpdateSliderContentRegionEx

```
function UpdateSliderContentRegionEx(MainContentRegion: TcHmi.Controls.System.TcHmiRegion,
SliderContentRegion: TcHmi.Controls.System.TcHmiRegion, MainContentSpecificToggleButton:
TcHmi.Controls.Beckhoff.TcHmiToggleButton, NavigationToggleButton:
TcHmi.Controls.Beckhoff.TcHmiToggleButton): void
```

Die Funktion UpdateSliderContentRegionEx wird verwendet, um mithilfe des eingestellten Schemas des [NavigationConfig](#) [▶ 41]-Properties des [Configurator](#) [▶ 37]-Controls den Slidercontent zu aktualisieren. Je nach angezeigtem TcHmiContent in dem TcHmiRegion-Control für den Hauptcontent wird der passende TcHmiContent in dem TcHmiRegion-Control für den Slidercontent reingeladen, sofern das [ToggleState](#)

Property des zugehörigen TcHmiToggleButton-Controls (mit der ID MainContentSpecificToggleButton) auf *Active* steht. Zusätzlich wird dem TcHmiToggleButton-Control ein Icon als *BackgroundImage*-Property zugeordnet, das ebenfalls über das *NavigationConfig [▶ 41]*-Property konfiguriert wird.

Parameter

Name	Typ	Beschreibung
MainContentRegion	TcHmi.Controls.System.TcHmiRegion	Das TcHmiRegion-Control, in dem der Hauptcontent angezeigt wird.
SliderContentRegion	TcHmi.Controls.System.TcHmiRegion	Das TcHmiRegion-Control, in dem der zugehörige Slidercontent des Hauptcontents angezeigt werden soll.
MainContentSpecificToggleButton	TcHmi.Controls.Beckhoff.TcHmiToggleButton	Das TcHmiToggleButton-Control, das für die Anzeige des Slidercontents im Sliderbereich dient.
NavigationToggleButton	TcHmi.Controls.Beckhoff.TcHmiToggleButton	Das TcHmiToggleButton-Control, das für die Anzeige der Navigationsseite im Sliderbereich dient.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.8.0

Verwendung: TC3 Plastic Application HMI Projekte

5.4 General

5.4.1 CreateBinding

```
function CreateBinding(expression: string, propertyName: string, control: TcHmi.Controls.System.TcHmiControl): void
```

Die Funktion CreateBinding erzeugt ein Binding zwischen einem Symbol und einem Property eines Controls.

Parameter

Name	Typ	Beschreibung
expression	string	Der Symbolausdruck mit dem ein Binding erzeugt werden soll.
propertyName	string	Der Name des Properties des Controls mit dem ein Binding erzeugt werden soll.
control	TcHmi.Controls.System.TcHmiControl	Das Control mit dem ein Binding erzeugt werden soll.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.8.0

Verwendung: TC3 Plastic Application HMI Projekte

5.4.2 ModifyArray

```
function ModifyArray(ArrayObject: any[], Operation: string, SelectedIndex: number, Input: any): any[]
```

Die Funktion ModifyArray ermöglicht das Modifizieren eines Arrays und gibt dies als Rückgabewert zurück. Die folgenden Operationen können am Array angewendet werden:

1. **Add:** Hinzufügen eines passenden Elementes an eine ausgewählte Stelle in das Array.
2. **AddEmpty:** Hinzufügen eines leeren Elementes an eine ausgewählte Stelle in das Array.
3. **RemoveSelected:** Entfernen eines ausgewählten Elementes aus dem Array.

Parameter

Name	Typ	Beschreibung
ArrayObject	<u>Array</u>	Das Symbol vom Typ Array, das modifiziert werden soll.
Operation	tchmi:framework#/definitions/ModifyArrayOperation	Die Art der Modifizierung, die mit dem Array gemacht werden soll. Kann aus der <u>Enumeration</u> ausgewählt werden: <ul style="list-style-type: none"> • Add: Hinzufügen des Elementes vom Parameter <i>Input</i> • AddEmpty: Fügt ein leeres Element hinzu. • RemoveSelected: Entfernt das Element.
SelectedIndex	<u>number</u>	Der Array-Index, bei dem das Array modifiziert werden soll. Wird nur optional benötigt.
Input	<u>any</u>	Ein nicht spezifischer Parameter, der für gewisse Änderungen benötigt wird. Wenn beim Parameter <i>Operation</i> „Add“ ausgewählt wird, dann benötigt dieser Parameter das Element, das in das Array eingesetzt werden soll. Wird nur optional benötigt.

Rückgabewert

Typ	Beschreibung
Array	Das modifizierte Array.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.12.0

Verwendung: TC3 Plastic Application HMI Projekte

5.4.3 SetSymbolArrayIdx

```
function SetSymbolArrayIdx(Symbol: TcHmi.Symbol, Idx: number, TargetArray: number): void
```

Die Funktion SetSymbolArrayIdx ändert in einem Symbol den adressierten Index eines darin befindlichen Arrays. Dabei kann zwischen mehreren Arrays in einem Symbol unterschieden werden. Voraussetzung ist die Übergabe der Arrays mit dem Index [0].

Beispielaufruf:

```
TcHmi.Functions.Plastic.SetSymbolArrayIdx(  TempSchedule::fbWeekTimers[0]::aMembers[0] , 4, 1)
```

Beispielresultat:

```
%%TempSchedule::fbWeekTimers[4]::aMembers[0]%/s%
```

Parameter

Name	Typ	Beschreibung
Symbol	TcHmi.Symbol	Symbol mit mindestens einem Array auf Index [0]
Idx	number	Der neue Index der anstatt [0] in das Array eingefügt werden soll
TargetArray	number	Das x-te Array, beginnend von links, das geändert werden soll.

Rückgabewert

Typ	Beschreibung
string	Ein Symbolausdruck mit dem ersetzten Index.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.8.0

Verwendung: TC3 Plastic Application HMI Projekte

5.5 Parameters

5.5.1 FillParamTable

```
function FillParamTable(TableControl, Symbol, Excluded): void
```

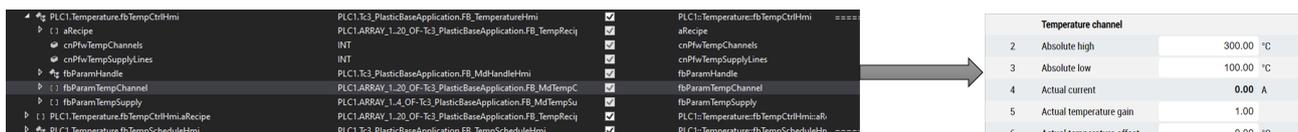
Die Funktion FillParamTable füllt eine Tabelle vom Typ [TcHmi.Controls.Plastic.Table](#) [135] mit den Maschinendaten eines PLC-Symbols. Dabei werden all die strukturierten Untersymbole berücksichtigt, dessen Instanz-Name mit „fbParams“ beginnt. Für jede gefundene Instanz eines fbParams wird eine Unterkategorie definiert und die verfügbaren Untersymbole eingefügt.

Zusätzlich zu den eingefügten Parametern, können in der [TcHmi.Controls.Plastic.Table](#) [135] weiterhin eigene Parameter unter dem Attribut FirstTable definiert werden. Diese werden hinter die generierten Tabelleneinträge eingefügt.

Beispielaufruf:

```
FillParamTable[TableControl: Parameters_Temperature_Table] , Symbol:  PLC1.Temperature.fbTempCtrlHmi , Excluded: (1 Item) ]
```

Beispielresultat:



Temperature channel		
2	Absolute high	300.00 °C
3	Absolute low	100.00 °C
4	Actual current	0.00 A
5	Actual temperature gain	1.00
6	Actual temperature offset	0.00 °C

Parameter

Name	Typ	Beschreibung
TableControl	TcHmi.Controls.Plastic.Table [135]	Die Tabelle die gefüllt werden soll.
Symbol	TcHmi.Symbol	Das Symbol das zum Füllen der Tabelle verwendet werden soll.

Name	Typ	Beschreibung
Excluded	TcHmi.Symbol[]	Untersymbole die beim Füllen der Tabelle ausgeschlossen werden sollen.

Rückgabewert

Typ	Beschreibung
void	kein Rückgabewert

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.8.0

Verwendung: TC3 Plastic Application HMI Projekte

5.5.2 UpdateParametersSettingsControls

```
function UpdateParametersSettingsControls(): void
```

Die Funktion UpdateParametersSettingsControls weist den folgenden Controls je nach TargetContent-Property des TcHmiRegion-Controls mit der ID Portrait_MainContent_Region unterschiedliche Symbole zu.

Control ID	Control Typ	Property
Parameters_Settings_Load_Button	TcHmi.Controls.Beckhoff.TcHmiButton	StateSymbol
Parameters_Settings_Save_Button	TcHmi.Controls.Beckhoff.TcHmiButton	StateSymbol
Parameters_Settings_CreationDateValue_Textblock	TcHmi.Controls.Beckhoff.TcHmiTextblock	Text
Parameters_Settings_StoreCountValue_Textblock	TcHmi.Controls.Beckhoff.TcHmiTextblock	Text
Parameters_Settings_StoredDateValue_Textblock	TcHmi.Controls.Beckhoff.TcHmiTextblock	Text
Parameters_Settings_Load_StateIndicator	TcHmi.Controls.Plastic.StateIndicator	IndicatorStatus
Parameters_Settings_Save_StateIndicator	TcHmi.Controls.Plastic.StateIndicator	IndicatorStatus

Contents/Navigation/Parameters/Parameters_Temperature.content

Control ID	PLC Symbol
Parameters_Settings_Load_Button	PLC1.Temperature.fbTempCtrlHmi::fbParamHandle::Load
Parameters_Settings_Save_Button	PLC1.Temperature.fbTempCtrlHmi::fbParamHandle::Save
Parameters_Settings_CreationDateValue_Textblock	PLC1.Temperature.fbTempCtrlHmi::fbParamHandle::FileCreationDate
Parameters_Settings_StoreCountValue_Textblock	PLC1.Temperature.fbTempCtrlHmi::fbParamHandle::FileStoreCount
Parameters_Settings_StoredDateValue_Textblock	PLC1.Temperature.fbTempCtrlHmi::fbParamHandle::FileStoreDate
Parameters_Settings_Load_StateIndicator	PLC1.Temperature.fbTempCtrlHmi::fbParamHandle::LoadingStatus
Parameters_Settings_Save_StateIndicator	PLC1.Temperature.fbTempCtrlHmi::fbParamHandle::SavingStatus

Contents/Navigation/Parameters/Parameters_MainExtruder.content

Control ID	PLC Symbol
Parameters_Settings_Load_Button	PLC1.Extruder.fbMainExtruderHmi::fbParamHandle::Load
Parameters_Settings_Save_Button	PLC1.Extruder.fbMainExtruderHmi::fbParamHandle::Save
Parameters_Settings_CreationDateValue_Textblock	PLC1.Extruder.fbMainExtruderHmi::fbParamHandle::FileCreationDate
Parameters_Settings_StoreCountValue_Textblock	PLC1.Extruder.fbMainExtruderHmi::fbParamHandle::FileStoreCount
Parameters_Settings_StoredDateValue_Textblock	PLC1.Extruder.fbMainExtruderHmi::fbParamHandle::FileStoreDate
Parameters_Settings_Load_StateIndicator	PLC1.Extruder.fbMainExtruderHmi::fbParamHandle::LoadingStatus
Parameters_Settings_Save_StateIndicator	PLC1.Extruder.fbMainExtruderHmi::fbParamHandle::SavingStatus

Contents/Navigation/Parameters/Parameters_CoExtruder.content

Control ID	PLC Symbol
Parameters_Settings_Load_Button	PLC1.Extruder.fbCoExtruderHmi::fbParamHandle::Load
Parameters_Settings_Save_Button	PLC1.Extruder.fbCoExtruderHmi::fbParamHandle::Save
Parameters_Settings_CreationDateValue_Textblock	PLC1.Extruder.fbCoExtruderHmi::fbParamHandle::FileCreationDate
Parameters_Settings_StoreCountValue_Textblock	PLC1.Extruder.fbCoExtruderHmi::fbParamHandle::FileStoreCount
Parameters_Settings_StoredDateValue_Textblock	PLC1.Extruder.fbCoExtruderHmi::fbParamHandle::FileStoreDate
Parameters_Settings_Load_StateIndicator	PLC1.Extruder.fbCoExtruderHmi::fbParamHandle::LoadingStatus
Parameters_Settings_Save_StateIndicator	PLC1.Extruder.fbCoExtruderHmi::fbParamHandle::SavingStatus

Contents/Navigation/Parameters/Parameters_Blowpin.content

Control ID	PLC Symbol
Parameters_Settings_Load_Button	PLC1.Blowpin.fbBlowpinHmi::fbParamHandle::Load
Parameters_Settings_Save_Button	PLC1.Blowpin.fbBlowpinHmi::fbParamHandle::Save
Parameters_Settings_CreationDateValue_Textblock	PLC1.Blowpin.fbBlowpinHmi::fbParamHandle::FileCreationDate
Parameters_Settings_StoreCountValue_Textblock	PLC1.Blowpin.fbBlowpinHmi::fbParamHandle::FileStoreCount
Parameters_Settings_StoredDateValue_Textblock	PLC1.Blowpin.fbBlowpinHmi::fbParamHandle::FileStoreDate
Parameters_Settings_Load_StateIndicator	PLC1.Blowpin.fbBlowpinHmi::fbParamHandle::LoadingStatus
Parameters_Settings_Save_StateIndicator	PLC1.Blowpin.fbBlowpinHmi::fbParamHandle::SavingStatus

Contents/Navigation/Parameters/Parameters_Carriage.content

Control ID	PLC Symbol
Parameters_Settings_Load_Button	PLC1.Carriage.fbCarriageHmi::fbParamHandle::Load

Control ID	PLC Symbol
Parameters_Settings_Save_Button	PLC1.Carriage.fbCarriageHmi::fbParamHandle::Save
Parameters_Settings_CreationDateValue_Textblock	PLC1.Carriage.fbCarriageHmi::fbParamHandle::FileCreationDate
Parameters_Settings_StoreCountValue_Textblock	PLC1.Carriage.fbCarriageHmi::fbParamHandle::FileStoreCount
Parameters_Settings_StoredDateValue_Textblock	PLC1.Carriage.fbCarriageHmi::fbParamHandle::FileStoreDate
Parameters_Settings_Load_StateIndicator	PLC1.Carriage.fbCarriageHmi::fbParamHandle::LoadingStatus
Parameters_Settings_Save_StateIndicator	PLC1.Carriage.fbCarriageHmi::fbParamHandle::SavingStatus

Contents/Navigation/Parameters/Parameters_Clamp.content

Control ID	PLC Symbol
Parameters_Settings_Load_Button	PLC1.Clamp.fbClampHmi::fbParamHandle::Load
Parameters_Settings_Save_Button	PLC1.Clamp.fbClampHmi::fbParamHandle::Save
Parameters_Settings_CreationDateValue_Textblock	PLC1.Clamp.fbClampHmi::fbParamHandle::FileCreationDate
Parameters_Settings_StoreCountValue_Textblock	PLC1.Clamp.fbClampHmi::fbParamHandle::FileStoreCount
Parameters_Settings_StoredDateValue_Textblock	PLC1.Clamp.fbClampHmi::fbParamHandle::FileStoreDate
Parameters_Settings_Load_StateIndicator	PLC1.Clamp.fbClampHmi::fbParamHandle::LoadingStatus
Parameters_Settings_Save_StateIndicator	PLC1.Clamp.fbClampHmi::fbParamHandle::SavingStatus

Contents/Navigation/Parameters/Parameters_Wtc.content

Control ID	PLC Symbol
Parameters_Settings_Load_Button	PLC1.Wtc.fbWtcHmi::fbParamHandle::Load
Parameters_Settings_Save_Button	PLC1.Wtc.fbWtcHmi::fbParamHandle::Save
Parameters_Settings_CreationDateValue_Textblock	PLC1.Wtc.fbWtcHmi::fbParamHandle::FileCreationDate
Parameters_Settings_StoreCountValue_Textblock	PLC1.Wtc.fbWtcHmi::fbParamHandle::FileStoreCount
Parameters_Settings_StoredDateValue_Textblock	PLC1.Wtc.fbWtcHmi::fbParamHandle::FileStoreDate
Parameters_Settings_Load_StateIndicator	PLC1.Wtc.fbWtcHmi::fbParamHandle::LoadingStatus
Parameters_Settings_Save_StateIndicator	PLC1.Wtc.fbWtcHmi::fbParamHandle::SavingStatus

Contents/Navigation/Parameters/Parameters_Monitoring.content

Control ID	PLC Symbol
Parameters_Settings_Load_Button	PLC1.Monitoring::fbParamHandle::Load
Parameters_Settings_Save_Button	PLC1.Monitoring::fbParamHandle::Save
Parameters_Settings_CreationDateValue_Textblock	PLC1.Monitoring::fbParamHandle::FileCreationDate
Parameters_Settings_StoreCountValue_Textblock	PLC1.Monitoring::fbParamHandle::FileStoreCount
Parameters_Settings_StoredDateValue_Textblock	PLC1.Monitoring::fbParamHandle::FileStoreDate
Parameters_Settings_Load_StateIndicator	PLC1.Monitoring::fbParamHandle::LoadingStatus
Parameters_Settings_Save_StateIndicator	PLC1.Monitoring::fbParamHandle::SavingStatus

Contents/Navigation/Parameters/Parameters_Setpoints.content

Control ID	PLC Symbol
Parameters_Settings_Load_Button	PLC1.Setpoints::fbParamHandle::Load
Parameters_Settings_Save_Button	PLC1.Setpoints::fbParamHandle::Save
Parameters_Settings_CreationDateValue_Textblock	PLC1.Setpoints::fbParamHandle::FileCreationDate
Parameters_Settings_StoreCountValue_Textblock	PLC1.Setpoints::fbParamHandle::FileStoreCount
Parameters_Settings_StoredDateValue_Textblock	PLC1.Setpoints::fbParamHandle::FileStoreDate
Parameters_Settings_Load_StateIndicator	PLC1.Setpoints::fbParamHandle::LoadingStatus
Parameters_Settings_Save_StateIndicator	PLC1.Setpoints::fbParamHandle::SavingStatus

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	kein Rückgabewert

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

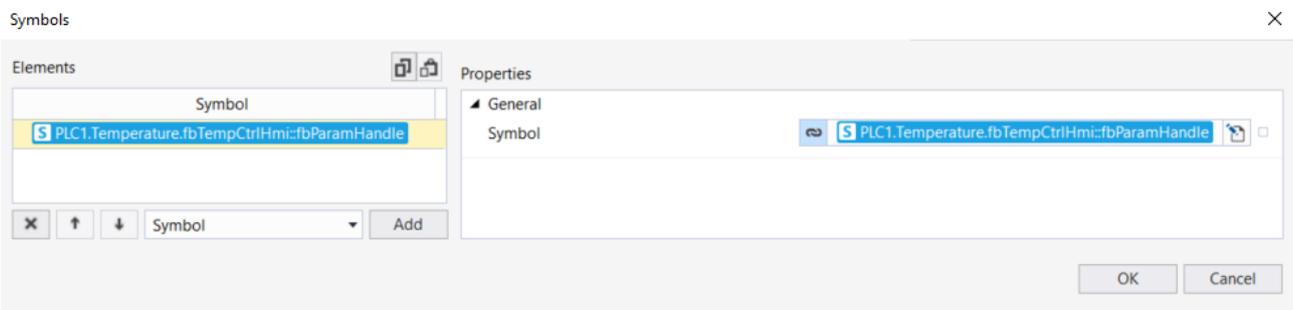
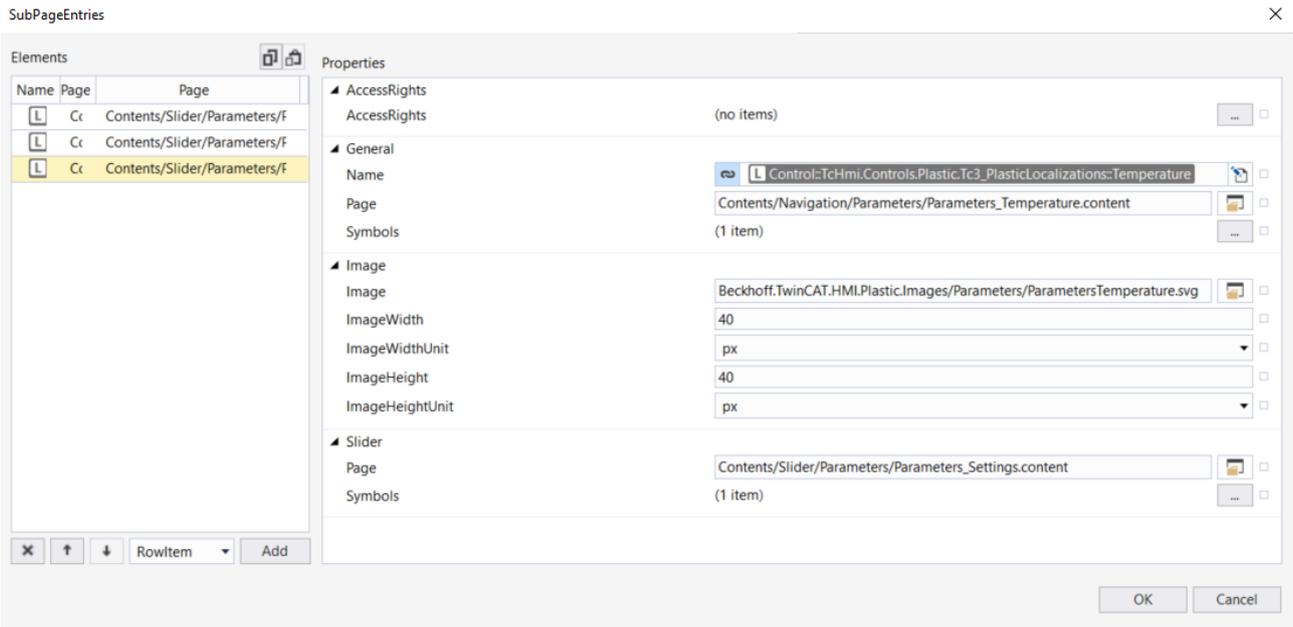
5.5.3 UpdateParametersSettingsControlsEx

```
function UpdateParametersSettingsControlsEx(): void
```

Die Funktion UpdateParametersSettingsControlsEx wird verwendet, um mithilfe des eingestellten Schemas des NavigationConfig-Properties des Configurator-Controls, je nach TargetContent-Property des TchmiRegion-Controls mit der ID Portrait_MainContent_Region den in der Tabelle aufgelisteten Controls die Symbole des eingestellten Symbols vom Typ `FB_ParamHandle` aus der Bibliothek `Tc3_PlasticBaseApplication` zuzuweisen.

Control ID	Control Typ	Property
Parameters_Settings_Load_Button	Tchmi.Controls.Beckhoff.TchmiButton	StateSymbol
Parameters_Settings_Save_Button	Tchmi.Controls.Beckhoff.TchmiButton	StateSymbol
Parameters_Settings_CreationDateValue_Textblock	Tchmi.Controls.Beckhoff.TchmiTextblock	Text
Parameters_Settings_StoreCountValue_Textblock	Tchmi.Controls.Beckhoff.TchmiTextblock	Text
Parameters_Settings_StoredDateValue_Textblock	Tchmi.Controls.Beckhoff.TchmiTextblock	Text
Parameters_Settings_Load_StateIndicator	Tchmi.Controls.Plastic.StateIndicator	IndicatorStatus
Parameters_Settings_Save_StateIndicator	Tchmi.Controls.Plastic.StateIndicator	IndicatorStatus

Beispieleinstellung des NavigationConfig-Properties:



Parameter

Name	Typ	Beschreibung
SymbolIndex	number	Der Index vom Slider-Symbol-Array des NavigationConfig [▶ 41]-Properties vom Configurator-Control. Das ausgewählte Symbol wird genutzt, um die Controls im Sliderbereich der Parameter-Seite upzudaten. Wird nur optional benötigt.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.8.0

Verwendung: TC3 Plastic Application HMI Projekte

5.5.4 UpdateTemperatureParametersTableNumber

```
function UpdateTemperatureParametersTableNumber () : void
```

Die Funktion `UpdateTemperatureParametersTableNumber` passt die Anzahl an Zonen in dem Table-Control mit der ID `Parameters_Temperature_Table` an, wenn sich das Control in dem TcHmiContent mit dem Pfad `Contents/Navigation/Parameters/Parameters_Temperature.content` befindet und dieser TcHmiContent in dem TcHmiRegion-Control mit der ID `Portrait_MainContent_Region` geladen ist. Die Anzahl an Zonen wird über das Symbol `PLC1.Temperature.fbTempCtrlHmi::CountPfwZones` bestimmt.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	kein Rückgabewert

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

5.5.5 UpdateTemperatureParametersTableNumberEx

```
function UpdateTemperatureParametersTableNumberEx(): void
```

Die Funktion `UpdateTemperatureParametersTableNumberEx` passt mithilfe des `NavigationConfig [▶ 41]`-Properties des Configurator-Controls die Anzahl an Temperaturkanälen in dem Table-Control mit der ID `Parameters_Temperature_Table` an, wenn sich das Control in dem TcHmiContent mit dem Pfad `Contents/Navigation/Parameters/Parameters_Temperature.content` befindet und dieser TcHmiContent in dem TcHmiRegion-Control mit der ID `Portrait_MainContent_Region` geladen ist. Dazu muss beim `NavigationConfig [▶ 41]`-Property des Configurator-Controls bei der entsprechenden Seite das Property `Symbols` der Kategorie `General` ein Symbol vom Typ `FB_TemperatureHmi` übergeben werden.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.8.0

Verwendung: TC3 Plastic Application HMI Projekte

5.6 Scope

5.6.1 UpdateScope

```
function UpdateScope(ToggleButtonControl: TcHmi.Controls.Beckhoff.TcHmiToggleButton, AxisPath: string): boolean
```

Die Funktion `UpdateScope` updated ein TcHmiTrendLineChart-Control mit der ID `Scope_TrendLineChart` und zeigt je nach Achstyp achsspezifische Werte an. Der Achstyp wird über das `AxisPath`-Property ermittelt. Das `AxisPath`-Symbol wird mit der Endung `::fbSpecific::fbNc::Valid` geprüft. Wenn dieser Wert auf `true` ist, dann ist damit festgelegt, dass die Achse vom Typ NC ist. Das TcHmiStatelImage-

Control mit der ID Scope_StateImage bekommt für das State-Property die ID des TcHmiToggleButton-Controls, das als Parameter in dieser Funktion übergeben wurde, übergeben. Dadurch kann pro anzuzeigende Achse im TcHmiStateImage-Control ein passendes Image hinterlegt werden, das dann passend umgeschaltet wird.

Parameter

Name	Typ	Beschreibung
ToggleButtonControl	TcHmi.Controls.Beckhoff.TcHmiToggleButton	Das TcHmiToggleButton-Control, das für die Umschaltung der Achsen genutzt wird.
AxisPath	string	Der Achs-Pfad der Achse, die angezeigt werden soll.

Rückgabewert

Typ	Beschreibung
boolean	Werte: <ul style="list-style-type: none"> • true: Die Achse ist vom Typ NC. • false: Die Achse ist nicht vom Typ NC.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

5.7 Trend

5.7.1 ResetTrendAxisColors

```
function ResetTrendAxisColors(): void
```

Die Funktion ResetTrendAxisColors resettet die in der Tabelle aufgelisteten Controls. Die TcHmiInput-Controls bekommen einen Hexadezimal-Farbcode zugewiesen und die zugehörigen TcHmiCheckbox-Controls die passende Hintergrundfarbe.

TcHmi.Controls.Beckhoff.TcHmi-Input Control ID	TcHmi.Controls.Beckhoff.TcHmi-Checkbox Control ID	Hexadezimal-Farbcode
Trend_Settings_1_Input	Trend_Settings_1_Rectangle	#E6194B
Trend_Settings_2_Input	Trend_Settings_2_Rectangle	#F58231
Trend_Settings_3_Input	Trend_Settings_3_Rectangle	#FFE119
Trend_Settings_4_Input	Trend_Settings_4_Rectangle	#BFEF45
Trend_Settings_5_Input	Trend_Settings_5_Rectangle	#3CB44B
Trend_Settings_6_Input	Trend_Settings_6_Rectangle	#469990
Trend_Settings_7_Input	Trend_Settings_7_Rectangle	#42D4F4
Trend_Settings_8_Input	Trend_Settings_8_Rectangle	#4363D8
Trend_Settings_9_Input	Trend_Settings_9_Rectangle	#911EB4
Trend_Settings_10_Input	Trend_Settings_10_Rectangle	#F032E6
Trend_Settings_11_Input	Trend_Settings_11_Rectangle	#469990
Trend_Settings_12_Input	Trend_Settings_12_Rectangle	#42D4F4

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	kein Rückgabewert

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

5.7.2 UpdateTrend

```
function UpdateTrend(FB_TrendHmi: TcHmi.Symbol): void
```

Die Funktion UpdateTrend updated ein TcHmiTrendLineChart-Control mit der ID Trend_TrendLineChart und zeigt Werte an. Über das TcHmiRegion-Control mit der ID Portrait_MainContent_Region wird überprüft, ob das TargetContent-Property den Wert Contents/Home.content hat oder nicht, da der Content, auf dem das TcHmiTrendLineChart-Control existiert auch auf dem Contents/Home.content durch ein weiteres TcHmiRegion-Control wiederverwendet werden kann. Durch die Funktion UpdateTrendAxisColors [► 190] können die Farben der Kurvenverläufe angepasst werden.

Parameter

Name	Typ	Beschreibung
FB_TrendHmi	TcHmi.Symbol	Das Symbol des FBs vom Typ FB_TrendHmi aus der Tc3_PlasticBaseApplication Bibliothek.

Rückgabewert

Typ	Beschreibung
void	kein Rückgabewert

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

5.7.3 UpdateTrendAxisColors

```
function UpdateTrendAxisColors(): void
```

Die Funktion UpdateTrendAxisColors liest das Text-Property des TcHmiInput-Controls aus. Wenn es sich bei dem ausgelesenem Text-Property um einen String zur Beschreibung von Farben im Hexadezimal-Farbcode (#RRGGBB) handelt, dann wird dieser Wert genutzt, um das BackgroundColor-Property des zugehörigen TcHmiCheckbox-Controls zur visuellen Darstellung der Farbe anzupassen. Außerdem wird der Farbwert in einem Array gespeichert, das als Rückgabewert dieser Funktion dient. Der Rückgabewert wird zum Beispiel in der Funktion UpdateTrend [► 190] verwendet, um die Farben der Kurvenverläufe anpassen zu können. Die folgenden Controls werden von der Funktion berücksichtigt:

TcHmi.Controls.Beckhoff.TcHmiInput Control ID	TcHmi.Controls.Beckhoff.TcHmiCheckbox Control ID
Trend_Settings_1_Input	Trend_Settings_1_Rectangle
Trend_Settings_2_Input	Trend_Settings_2_Rectangle
Trend_Settings_3_Input	Trend_Settings_3_Rectangle
Trend_Settings_4_Input	Trend_Settings_4_Rectangle
Trend_Settings_5_Input	Trend_Settings_5_Rectangle

TcHmi.Controls.Beckhoff.TcHmiInput Control ID	TcHmi.Controls.Beckhoff.TcHmiCheckbox Control ID
Trend_Settings_6_Input	Trend_Settings_6_Rectangle
Trend_Settings_7_Input	Trend_Settings_7_Rectangle
Trend_Settings_8_Input	Trend_Settings_8_Rectangle
Trend_Settings_9_Input	Trend_Settings_9_Rectangle
Trend_Settings_10_Input	Trend_Settings_10_Rectangle
Trend_Settings_11_Input	Trend_Settings_11_Rectangle
Trend_Settings_12_Input	Trend_Settings_12_Rectangle

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
string[]	Array mit den Farben in der Syntax, die von einem TcHmiTrendLineChart-Control genutzt werden kann.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Functions

Verfügbar: seit Version 12.6.0

Verwendung: TC3 Plastic Application HMI Projekte

6 Images

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Images

Verfügbar: seit Version 12.6.0

Das NuGet-Paket **Beckhoff.TwinCAT.HMI.Plastic.Images** kann über das Paketverwaltungssystem NuGet im TwinCAT HMI Projekt installiert werden.

Für die Verwendung der Icons aus dem Paket muss der Text nach dem folgenden Schema in das Property eines Controls (z. B. `BackgroundImage`) eingefügt werden:

```
Beckhoff.TwinCAT.HMI.Plastic.Images/Ordner/Icon.svg
```

In diesem Beispiel referenziert der String auf das Icon mit der Benennung `Icon.svg`, das sich innerhalb des Paketes im Ordner mit dem Namen „Ordner“ befindet.

Grafiken im Gallery Explorer

Die Icons werden mit der Installation des NuGet-Paketes in dem TwinCAT HMI Gallery Explorer eingebunden. Dort können sie angeschaut und nach Belieben auch exportiert werden. Außerdem hilft die Einbindung in den Gallery Explorer beim Zusammenstellen des Icon-Pfades.

Pfade der Grafiken

- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Axes.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Scope.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Blowpin/Blowpin.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Blowpin/BlowpinDown.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Blowpin/BlowpinHoming.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Blowpin/BlowpinUp.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Blowpin/CoolingBlowpin.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Blowpin/CoolingBlowpinDown.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Blowpin/CoolingBlowpinUp.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Blowpin/SideBlowpin.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Blowpin/SideBlowpinBlow.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Carriage/Carriage.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Carriage/CarriageHoming.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Carriage/CarriageLeft.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Carriage/CarriageRight.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Clamp/Clamp.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Clamp/ClampClose.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Clamp/ClampHoming.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Clamp/ClampLeft.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Clamp/ClampOpen.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Axes/Clamp/ClampRight.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/CoExtruder.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/CoExtruderVelocityDown.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/CoExtruderVelocityUp.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/Extruder.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/ExtruderDown.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/ExtruderHead.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/ExtruderHeadWhite.svg

- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/ExtruderUp.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/MainExtruder.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/MainExtruderVelocityDown.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/MainExtruderVelocityUp.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/Trend.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/ViewStripeExtruder.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/ViewStripeExtruderVelocityDown.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Extruder/ViewStripeExtruderVelocityUp.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/AdditionalPages.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Beckhoff_Logo.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Home.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Info.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Left.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/ManualFunctions.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Navigation.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Pause.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Right.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Screenshot.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/SliderDown.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/SliderUp.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Start.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Stop.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/ToBasePosition.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/User.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/br.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/can.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/cn.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/de.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/en_gb.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/en_us.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/eng.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/es.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/fr.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/in.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/is.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/it.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/jp.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/nd.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/pk.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/pr.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/ru.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/sa.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/tr.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/uk.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Frame/Flags/us.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/MachineRenderings/BlowMolding.png

- Beckhoff.TwinCAT.HMI.Plastic.Images/MachineRenderings/BlowMoldingLine.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/MachineRenderings/CoExtruder.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/MachineRenderings/ExtKopf.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/MachineRenderings/Extruder.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/MachineRenderings/ExtruderWith13Zones.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/MachineRenderings/ExtruderWith5Zones.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/MachineRenderings/Flansch.png
- Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/Parameters.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersAccumulator.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersActuators.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersBlowpin.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersCarriage.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersClamp.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersCoExtruder.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersMainExtruder.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersMonitoring.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersSetpoints.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersTemperature.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Parameters/ParametersWtc.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Process/Blowing.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Process/EnergyMeasurement.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Process/Monitoring.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Process/Process.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Process/Setpoints.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Process/Timer.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/Administration.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/AlarmColumns.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/AlarmHistory.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/Deselect.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/Events.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/EventsAlarms.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/FileManagement.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/Filter.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/Materials.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/Recipe.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/RecipeManagement.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/RFID.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/System.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/System/UserLogging.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/Cooling.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/Heating.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/Temperature.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/TemperatureControlCabinet.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/TemperatureOff.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/TemperatureOn.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/TemperatureParameters.svg

- Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/TemperatureStandby.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/TemperatureTimed.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/TemperatureTimeScheduling.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/TemperatureZones.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/TemperatureZonesLocalization.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Temperature/TemperatureZonesLocalizationReverse.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/Automatic.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/Cut.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/Drive.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/Info.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/Lamp.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/Plus.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/SemiAutomatic.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/CanisterHandleCutter.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/CanisterHandleCutterBackward.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/CanisterHandleCutterForward.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/Corepuller.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/CorepullerClose.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/CorepullerOpen.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/Cutter.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/CutterBackward.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/CutterForward.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/ExpandingMandrel.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/ExpandingMandrelIn.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/ExpandingMandrelOut.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/Gripper.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/GripperBackward.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/GripperClose.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/GripperForward.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/GripperOpen.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/PartGripper.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/PartGripperClose.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/PartGripperOpen.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/BlowMolding/SupportAir.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/Extruder/ExtruderAdministration.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/Extruder/ExtruderMachineOverview.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/Extruder/ExtruderOverviewWhite.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/Hydraulic/HydraulicPump.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/Hydraulic/HydraulicPumpTechnical.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/BarrelHeat.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/BarrelHeat2.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/Carriage.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/Carriage2.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/Clamp.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/ClampClose.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/ClampCooling.svg

- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/ClampLock.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/ClampOpen.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/Cloud.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/Configuration.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/Corepuller.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/DataLoadSave.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/Ejector.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/Injection.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/InjectionHoldingPressure.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/InjectionUnit.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/InjectionUnitConfiguration.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/InjectionUnitPressure.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/InjectionUnitTemperature.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/InjectionUnitTorque.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/Metering.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/Pressure.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/Pump.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/QualityTable.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/ShotScope.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/Temperature.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/TimingScope.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Various/InjectionMolding/UserManagement.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Wtc/Accumulator.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Wtc/AccumulatorHoming.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Wtc/EditorProfileBackward.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Wtc/EditorProfileForward.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Wtc/EditorProfileUndo.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Wtc/Wtc.svg
- Beckhoff.TwinCAT.HMI.Plastic.Images/Wtc/WtcHoming.svg

7 Localizations

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Localizations

Verfügbar: seit Version 12.6.0

Das NuGet-Paket **Beckhoff.TwinCAT.HMI.Plastic.Localizations** kann über das Paketverwaltungssystem NuGet im TwinCAT HMI Projekt installiert werden.

Es enthält zwei Controls, die jeweils eine englische (en-US) und eine deutsche (de-DE) Sprachdatei mit Sprachsymbolen die für Kunststoff-Bedienoberflächen relevant sein können beinhalten:

- **Tc3_PlasticLocalizations:** Allgemeine Sprachsymbole für die gesamte Bedienoberfläche.
- **Tc3_PlasticParameterInfo:** Texte zur Beschreibung der Parameter, die mithilfe der [FillParamTable](#) [ID_182](#) Funktion automatisch in das Table-Control eingebunden werden.

Für die Verwendung der Sprachsymbole aus dem Paket muss der folgende Text in das Property eines Controls (z. B. `Text`) eingefügt werden:

```
"%I%Control::TcHmi.Controls.Plastic.Tc3_PlasticLocalizations::Language%/I%"
```

In diesem Beispiel referenziert der String auf den Sprachschlüssel „Language“. Dabei ist die englische Übersetzung „Language“ und die deutsche „Sprache“.

Um die Liste aller zur Verfügung stehenden Sprachsymbole anzuzeigen bzw. die Übersetzungen der Sprachen englisch und deutsch zu ändern oder weitere Sprachen hinzuzufügen, können die Sprachdateien im TwinCAT HMI Projekt importiert werden. Dafür müssen die folgenden Schritte durchgeführt werden:

1. Klicken Sie mit einem Rechtsklick auf den Ordner **Localization** im TwinCAT HMI Projekt und fügen Sie ein neues Item hinzu.
2. Wählen Sie die Option **Import Localization** bei dem sich öffnenden Fenster aus und klicken Sie danach auf den Button **Add**.
3. Öffnen Sie im TwinCAT HMI Wizard Fenster den Reiter **Control**, wählen Sie entweder das Control mit dem Namen „Tc3_PlasticLocalizations“ oder „Tc3_PlasticParameterInfo“ über die Checkbox an und bestätigen Sie es mit einem Klick auf den Button **OK**. (Nach Belieben kann auch nur eine der beiden zur Verfügung stehenden Sprachen ausgewählt werden.)
4. Die angewählten Sprachdateien erscheinen im Ordner „Localizations“ und können mithilfe des Lokalisierungseditors bearbeitet werden.

8 RecipeHelper

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper

Verfügbar: seit Version 12.6.0

Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper ist ein NuGet-Paket, das in einem TwinCAT-HMI-Projekt zusätzlich zum Paket Beckhoff.TwinCAT.HMI.Plastic.Controls verwendet werden kann. Das RecipeHelper-Paket fügt der TwinCAT HMI Recipe Management Extension zusätzliche Funktionalitäten hinzu, wie z. B.

- Einfache Einrichtung der Rezeptstruktur für die Anwendung.
- Automatisches Hinzufügen von Sollwertsymbolen zu Rezepten, die mit Plastic-Controls verwendet werden.
- Vorübergehende Speicherung von Rezepten und die Möglichkeit, Änderungen am Originalrezept rückgängig zu machen.
- PLC Handshake für den sicheren Rezeptwechsel an einer Maschine.
- Handshake mit der SPS zur Reaktivierung des letzten Rezepts bei einem Neustart.

Das RecipeHelper-Paket bietet eine Reihe von JavaScript-Funktionen, die mit der bestehenden TwinCAT HMI Recipe Management Extension zusammenarbeiten. Weitere Informationen finden Sie unter: [TwinCAT HMI Rezeptverwaltung](#).

Zugang zur Rezeptkonfiguration

Wenn die Benutzerauthentifizierung aktiv ist, müssen Sie den Zugriff auf die Rezeptkonfiguration für die gewünschten Benutzergruppen explizit freigeben. Weitere Informationen über den Benutzerzugriff auf RecipeManagement.Config finden Sie unter dem [Link](#).



Ohne Lese- und Schreibzugriff auf das Symbol RecipeManagement.Config funktionieren die RecipeHelper-Funktionen nicht wie erwartet.

8.1 Konzept

8.1.1 Erstellte Rezepttypen und Rezepte

Wenn das NuGet-Paket RecipeHelper bei der ersten Initialisierung in ein TwinCAT-HMI-Projekt eingebunden wird, erstellt es automatisch Objekte wie Rezepttypen, Rezepte und Serversymbole, die im Projekt für seine Funktionen benötigt werden.

Die neu hinzugefügten Objekte sind unten aufgeführt.

Serversymbole:

1. `LastActivatedRecipe`: Hierbei handelt es sich um ein Serversymbol vom Typ String, in dem der Name des Rezepts gespeichert wird, das zuletzt auf dem Server aktiviert wurde.
2. `RecipeLastUpdatedTime`: Dieses Symbol speichert Datum und Uhrzeit in einem String-Format, wenn ein Rezept geändert wird. Dieses Symbol wird zu jedem Produktrezept hinzugefügt.
3. `UnsavedChangesInActRecipe`: Dieses boolesche Symbol ist TRUE, wenn es im aktiven temporären Rezept letzte Änderungen gibt, die noch nicht im Originalrezept gespeichert sind.
4. `PLC_HMI_Handshake_State`: Dieses Integer-Symbol wird aktualisiert, wenn die RecipeHelper-State-Machine die Handshake-Kommunikation mit der SPS verarbeitet. Der Status der Kommunikation ist eine wichtige Information, um zu verhindern, dass während eines ungültigen Status falsche Daten in das Rezept eingegeben werden.

Rezepttypen und Rezepte:

1. `ActRcpDataType`: Dies ist der Rezepttyp für Standardproduktrezepte. Bei der Erstellung dieses Rezepttyps gibt es nur eine Variable, nämlich `RecipeLastUpdatedTime`, als Member.

2. Der RecipeHelper erstellt auch eine Rezeptinstanz dieses Typs mit dem Namen `ActRcpData`. Diese fungiert als das Rezept, das immer als aktuelles Rezept funktioniert und alle Änderungen am ursprünglichen Rezept speichert.

Distributed-Recipe-Struktur:

Seit Version 12.14.0 unterstützt das NuGet-Paket RecipeHelper eine strukturierte Rezeptkonfiguration.

Jedes Framework-Control hat das Attribut `RecipeTypeName`, mit dem ein Benutzer einzelne Rezepttypen auswählen kann, um bestimmte Variablen zusammenzufassen.

Jeder dieser Rezepttypen wird erstellt und im Rezepttypenordner `ActRcpDataTypeFolder` gespeichert. Wenn ein Symbol nicht Teil eines anderen bestehenden Rezepttyps ist, fügt das Control es zum Rezepttyp hinzu, der durch das Attribut `RecipeTypeName` vorgegeben ist.

Jeder Unterrezepttyp innerhalb von `ActRcpDataTypeFolder` hat sein Instanzrezept für jedes Produkt unter einem Rezeptordner `{PRODUCTNAME}Folder`. Zum Beispiel werden die Unterrezepte von `ActRcpData` innerhalb von `ActRcpDataFolder` gespeichert.

Während der Aktivierung, des Exports und des Imports werden alle Unterrezepte durch das RecipeHelper NuGet-Paket verarbeitet.

Die Teach-Operation wird nur für einzelne Unterrezepte durchgeführt, die von Änderungen der Servervariablen betroffen sind.

8.1.2 Temporäre Rezepte

Das Konzept der Verwendung eines temporären Rezepts besteht darin, ein einziges Rezept zu haben, das immer auf dem Control aktiv ist. Die letzten Änderungen werden in diesem aktiven temporären Rezept gespeichert, bis der Benutzer explizit die Funktion `SaveActiveRecipe` aufruft, um das Rezept zu speichern. Dies gibt dem Benutzer die Möglichkeit, das Originalrezept erneut zu laden, um letzte Änderungen rückgängig zu machen.

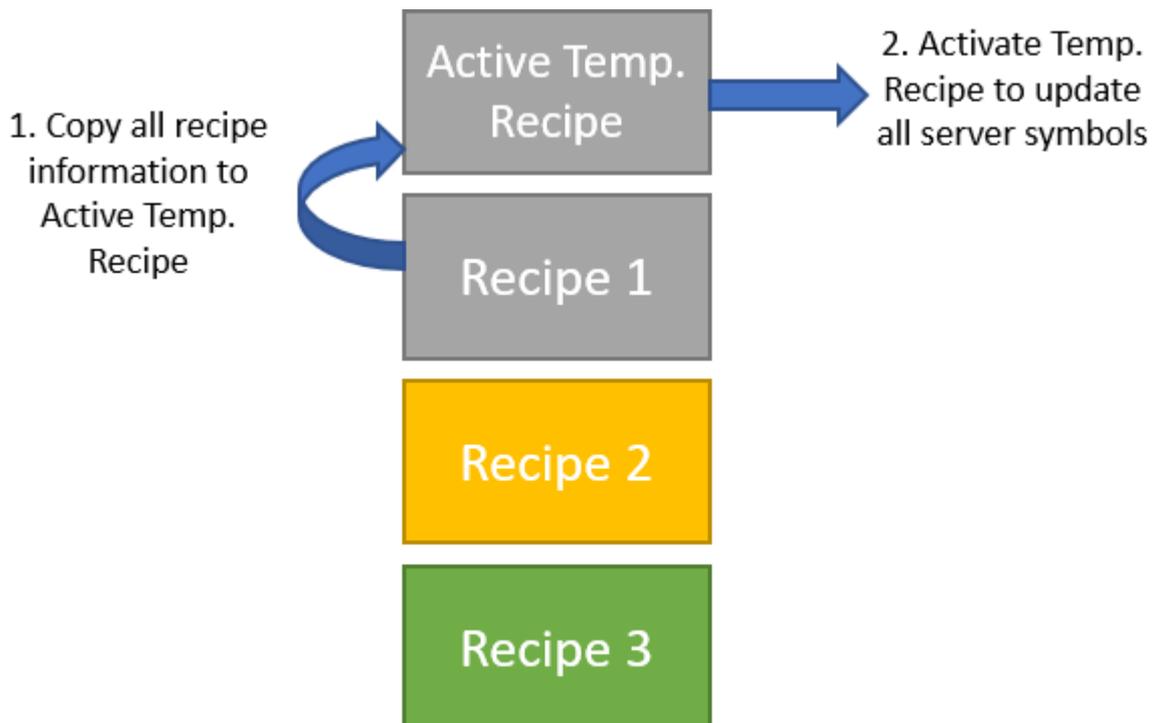
Da die aktiven temporären Änderungen in einem Rezept gespeichert werden, sind diese Änderungen auch nach einem Neustart auf der Steuerung verfügbar.

Im Folgenden werden die verschiedenen Rezeptoperationen, die in dieser Konfiguration durchgeführt werden, erläutert.

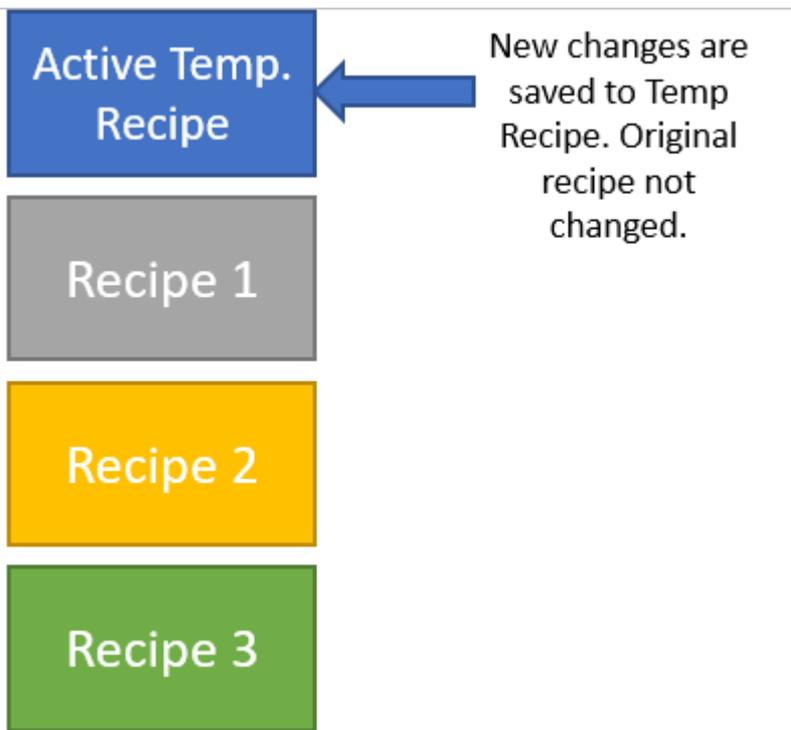
Rezeptstruktur: Ein automatisch erstelltes Rezept `ActRcpData` fungiert als aktives Rezept, und es kann mehrere von den Benutzern definierte Rezepte geben.



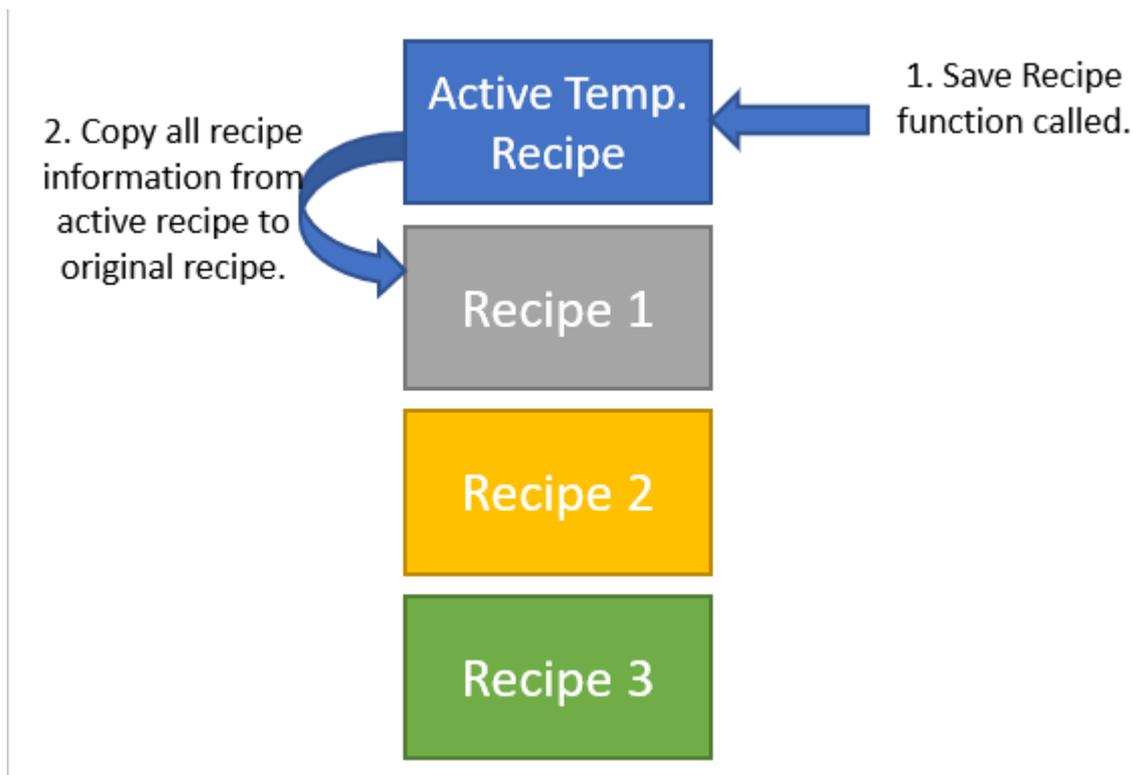
Aktivieren eines Rezepts: Die Funktion `LoadProductRecipe` muss aufgerufen werden, um ein Rezept zu laden und zu aktivieren. Das Rezept kopiert zunächst alle Rezeptinformationen in das temporäre Rezept `ActRcpData` und anschließend wird dieses temporäre Rezept aktiviert.



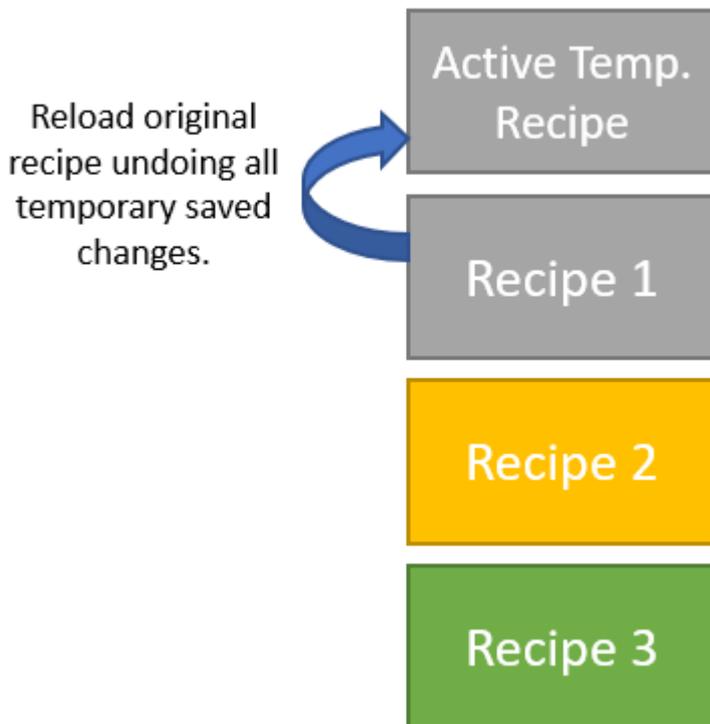
Neue Änderungen durch den Benutzer: Alle neuen Änderungen, die an den im Beckhoff.TwinCAT.HMI.Plastic-Paket enthaltenen Controls vorgenommen werden, werden automatisch im aktiven temporären Rezept gespeichert, ohne das ursprüngliche Rezept zu beeinflussen.



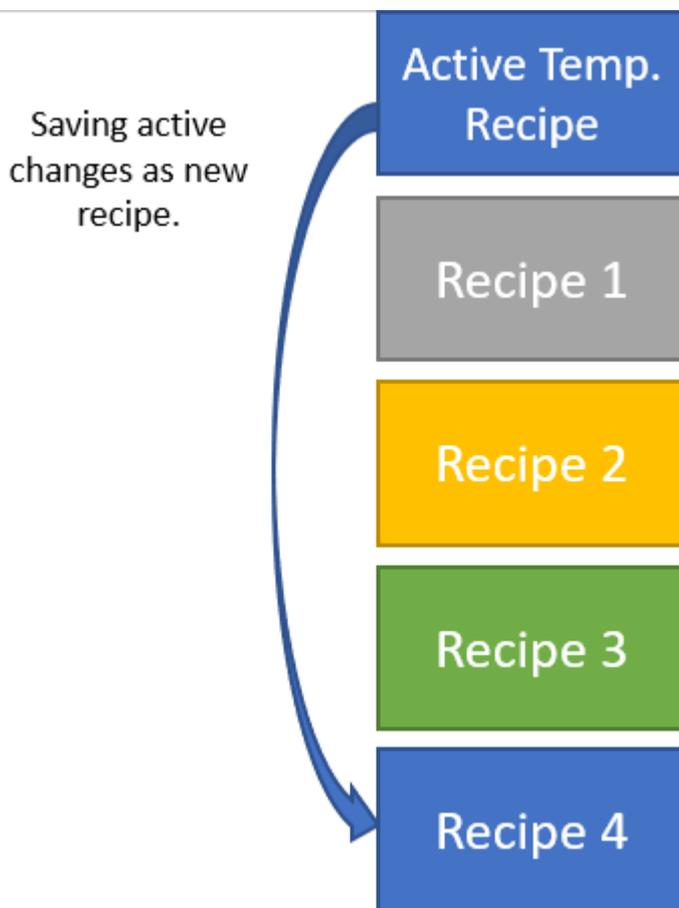
Speichern eines Rezepts: Durch Aufruf der Funktion `SaveActiveRecipe` mit dem Zielnamen, der mit dem zuletzt aktivierten Rezeptnamen übereinstimmt, kann der Benutzer den gesamten Inhalt des aktiven temporären Rezepts in seinem ursprünglichen Rezept speichern.



Rückgängigmachen aktueller Änderungen und erneutes Laden des ursprünglichen Rezepts: Durch den Aufruf von `LoadProductRecipe` mit dem Namen des zuletzt aktivierten Rezepts kann der Benutzer das aktive temporäre Rezept überschreiben und alle nicht gespeicherten Änderungen rückgängig machen.



Speichern der aktuellen Änderungen als neues Rezept: Wenn Sie `SaveActiveRecipe` mit einem neuen Zielrezeptnamen verwenden, werden alle aktiven temporären Rezepte als neue Rezepte gespeichert, ohne dass das ursprüngliche Rezept, das zuvor geladen wurde, davon betroffen ist.



8.1.3 Automatisches Hinzufügen von Symbolen zu Rezepttypen

Da der Rezepttyp `ActRcpDataType` vom RecipeHelper-Paket erstellt wird, hat er keine Membervariablen. Der Benutzer kann die Member des Rezepttyps hinzufügen. Einige Controls, die vom Beckhoff.TwinCAT.HMI.Plastic-Paket bereitgestellt werden, können automatisch Symbole zu diesem Rezepttyp und zu Rezepten, die auf diesem Typ basieren, mit Hilfe der JavaScript-API-Funktionen hinzufügen.

Wenn beispielsweise ein Symbol `FB_Axis` für PTP-Bewegungen an ein Attribut `FB_Axis` eines `ArrowMotionGraph`-Controls gebunden ist, fügt das Control bei der Initialisierung alle vom Benutzer konfigurierbaren Variablen unter dem gegebenen Symbol `FB_Axis` zum Typ `ActRcpDataType` hinzu. Alle Änderungen, die der Benutzer von einem Browser aus an dem Control vornimmt, werden automatisch aktualisiert und im aktiven temporären Rezept gespeichert.



Das Hinzufügen neuer Symbole zu Rezepttypen ist nur in der Engineering-Umgebung möglich, d.h. beim Bearbeiten von Inhalten und in der Live-Ansicht.



Wenn ein Inhalt in der Live-Ansicht nicht initialisiert wurde oder ein Membersymbol des Rezepts in der SPS geändert oder entfernt wurde, muss dies vor der Veröffentlichung des Projekts im Engineering überprüft werden.



Nach der Veröffentlichung des Projekts auf dem Zielgerät fügt das Paket dem Rezept keine neuen Symbole hinzu. Wenn eines der Rezeptsymbole auf dem Ziel nicht gefunden wird, führt dies zu einem Fehler und die Rezeptoperationen werden nicht ausgeführt.

8.1.4 Handshake mit PLC

Die Handshake-Kommunikation mit der SPS ist ein wesentliches Merkmal des RecipeHelper-Pakets. Für einen sicheren Betrieb führt der JavaScript-Code eine State-Machine mit Anfragesignalen an die SPS aus und wartet auf entsprechende Antwortsignale. Erst nach einem erfolgreichen Handshake führt der RecipeHelper die folgenden Operationen durch.

- Rezeptwechsel
- speichern neuer Daten aus der SPS im Rezept
- Aktivierung beim Starten

8.1.4.1 `FB_PlcStateToHmi`

SPS-Variablen für den Handshake sind im Funktionsbaustein `FB_PlcStateToHmi` verfügbar. Die folgende Abbildung zeigt die Eigenschaften/Variablen innerhalb dieses Funktionsbausteins.

ADS	object	
PLC1	object	
Constants	#/definit	
Global_Version	#/definit	
GVL	#/definit	
HmiCommunication	#/definit	
fbPlcStateToHmi	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi
AxesInitialized	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::AxesInitialized
BlockedAxisDataPlc	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::BlockedAxisDataPlc
BlockedTempDataPlc	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::BlockedTempDataPlc
ClientID	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::ClientID
DataReqFailed	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::DataReqFailed
DataRequestPlc	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::DataRequestPlc
DataRequestPlcActiveHmi	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::DataRequestPlcActiveHmi
DataRequestPlcQuitHmi	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::DataRequestPlcQuitHmi
DataValidPlc	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::DataValidPlc
LiveSignHMI	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::LiveSignHMI
ProductChangeConfirmPlc	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::ProductChangeConfirmPlc
ProductChangeEnable	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::ProductChangeEnable
ProductChangeRequestHmi	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::ProductChangeRequestHmi
Reset	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::Reset
SaveDataQuitPlc	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::SaveDataQuitPlc
SaveDataRequestPlc	#/definit	PLC1.HmiCommunication.fbPlcStateToHmi::SaveDataRequestPlc

Name	Typ	Beschreibung
AxesInitialized	Bool	Wird von der SPS verwendet.
BlockedAxisDataPlc	Bool	Wird von der SPS verwendet.
BlockedTempDataPlc	Bool	Wird von der SPS verwendet.
ClientID	UINT	Die Client-ID der Client-Instanz, die die Handshake-Kommunikation verwaltet.
DataReqFailed	Bool	Anzeige von der SPS, dass die Datenanforderung fehlgeschlagen ist. Die Handshake-State-Machine muss von der HMI zurückgesetzt werden.
DataRequestPlc	Bool	SPS, die Daten aus dem aktiven Rezept anfordert.
DataRequestPlcActiveHmi	Bool	Wird von der HMI auf TRUE gesetzt, um mitzuteilen, dass die HMI die Anforderung erhalten hat und den Datendownload zur SPS bearbeitet.
DataRequestPlcQuitHmi	Bool	Wird von der HMI auf TRUE gesetzt, wenn das Herunterladen aller Daten abgeschlossen ist.
DataValidPlc	Bool	Dies wird von der SPS auf TRUE gesetzt, um anzuzeigen, dass die SPS die Daten von der HMI empfangen und validiert hat und Benutzeroperationen sicher sind.
LiveSignHMI	UINT	Dies wird von dem Client, der den Handshake-Vorgang überwacht, aktualisiert; kein Live-Zeichen innerhalb einer bestimmten Zeitspanne bedeutet einen Verbindungsverlust.
ProductChangeEnable	Bool	Der Produktwechsel wird von der SPS erlaubt.
ProductChangeRequestHmi	Bool	Aufforderung zur Änderung eines Rezepts über die HMI.

Name	Typ	Beschreibung
ProductChangeConfirmPlc	Bool	Ermöglicht die Änderung eines Rezepts durch die SPS.
Reset	Bool	Reset-Anforderung von der HMI, wenn der Handshake fehlgeschlagen und das DataReqFailed-Flag gesetzt ist.
SaveDataRequestPlc	Bool	Die SPS fordert die HMI auf, Daten aus der SPS im Rezept zu speichern.
SaveDataQuitPlc	Bool	Die HMI setzt diese Variable, um anzuzeigen, dass die Anforderung zum Speichern von Daten empfangen und abgeschlossen wurde.

8.1.4.2 Arbeit mit mehreren Kunden

Wie beim Handshake zwischen der SPS und der HMI folgt der Client einer sequentiellen State-Machine. Es ist notwendig, dass nur ein einziger Kunde die Kommunikation initiiert und über alle Zustände hinweg überwacht.

Die ClientID-Variable der `FB_PlcStateToHmi` wird von dem ersten Client, der Zugriff auf diese Variable erhält, auf eine eindeutige ID gesetzt. Nur der durch diese ID identifizierte Client bearbeitet Kommunikationsanfragen, um die SPS über den Verbindungsstatus zu informieren, und er aktualisiert die Variable `LiveSignHMI` alle fünfhundert Millisekunden.

Wenn sich die Variable `LiveSignHMI` über einen Zeitraum von mehr als 5 Sekunden nicht ändert, sollte die SPS diesen Client entfernen, indem sie "Null" in die Variable `ClientID` schreibt.

Wenn noch andere Clients aktiv sind, erhält einer von ihnen den Zugang, indem er seine eigene ID in die Variable `ClientID` schreibt und seinen aktiven Status durch Aktualisierung der Variable `LiveSignHMI` anzeigt.

8.1.4.3 State-Machine starten

Die folgenden Schritte werden durchgeführt, um die Kommunikation abzuschließen und die letzten Rezeptdaten in die SPS-Variablen beim Start der Maschine herunterzuladen:

1. Beim Start verfügt die SPS über keine Rezeptdaten, da die HMI später eine Sitzung starten kann. Die SPS setzt den Flag `DataRequestPLC` auf TRUE und den Flag `DataValidPlc` auf FALSE.
2. Der HMI-Client liest bei der Initialisierung die Variable `DataRequestPLC` und setzt `DataRequestPLCActiveHmi` auf TRUE, um anzuzeigen, dass er die Anforderung erhalten hat.
3. Das Rezept, das als aktives temporäres Rezept gespeichert ist, wird von der HMI aus aktiviert und setzt die Werte für alle zugehörigen SPS-Variablen.
4. Der HMI-Client schreibt FALSE in `DataRequestPLCActiveHmi` und TRUE in `DataRequestPLCQuitHMI`, um die vollständige Aktivierung anzuzeigen.
5. Die SPS entfernt die Anfrage, indem sie `DataRequestPlc` auf FALSE zurücksetzt.

8.1.4.4 Ein Rezept speichern

Wenn es kürzlich Änderungen in den SPS-Variablen gab, die die SPS in das aktive Rezept speichern möchte, werden die folgenden Schritte durchgeführt, um den Prozess abzuschließen.

1. Die SPS löst eine Speichieranforderung aus, indem sie die Variable `SaveDataRequestPlc` auf TRUE setzt.
2. Der HMI-Client liest dies und speichert alle Änderungen in einem aktiven temporären Rezept.
3. Die HMI signalisiert das Ende des Speichervorgangs, indem sie den Wert TRUE in die Variable `SaveDataQuitPlc` schreibt.

4. Die SPS erkennt dies und löscht die Anfrage.

8.1.4.5 Ein Rezept laden

Diese Handshake-Zustandssequenz stellt sicher, dass die SPS eine sichere Änderung des Rezepts zulässt und wird aktualisiert, wenn neue Rezeptwerte heruntergeladen werden.

Die Schritte für diesen Handshake sind:

1. Die SPS muss eine Änderung des Rezepts zulassen, indem sie `ProductChangeEnable` auf TRUE setzt.
2. Der HMI-Client fordert einen Produktwechsel an, indem er `ProductChangeRequestHmi` auf TRUE setzt.
3. Wenn die Sicherheit gewährleistet ist, antwortet die SPS auf die Anfrage, indem sie `ProductChangeConfirmPlc` auf TRUE setzt.
4. Das HMI kopiert das neue Rezept in das temporäre Rezept `ActRcpData`, aktiviert es aber nicht.
5. Die HMI entfernt die Änderungsanforderung, indem sie `ProductChangeRequestHmi` auf FALSE zurücksetzt.
6. Die SPS fordert nun die Aktivierung des neuen Rezepts und den Download der Symbolwerte an, indem sie `DataRequestPLC` auf TRUE setzt.
7. Dies löst eine Handshake-Sequenz wie [Startup state-machine](#) [► 205] aus und die neuen Rezeptdaten werden in die SPS geladen.

8.1.4.6 DataReqFailed und Reset

1. Wenn während eines beliebigen Zustands der State-Machine der Zustand nicht innerhalb einer bestimmten Zeitspanne ausgeführt werden kann, kann die SPS die HMI über diesen Fehler informieren, indem sie die Variable `DataReqFailed` auf TRUE setzt.
2. Als Reaktion darauf wird die Variable `Reset` durch eine steigende Flanke von der HMI gesetzt und rückgesetzt.
3. Die SPS setzt ihre Zustände zurück und beginnt die Abfrage wieder beim ersten Schritt.

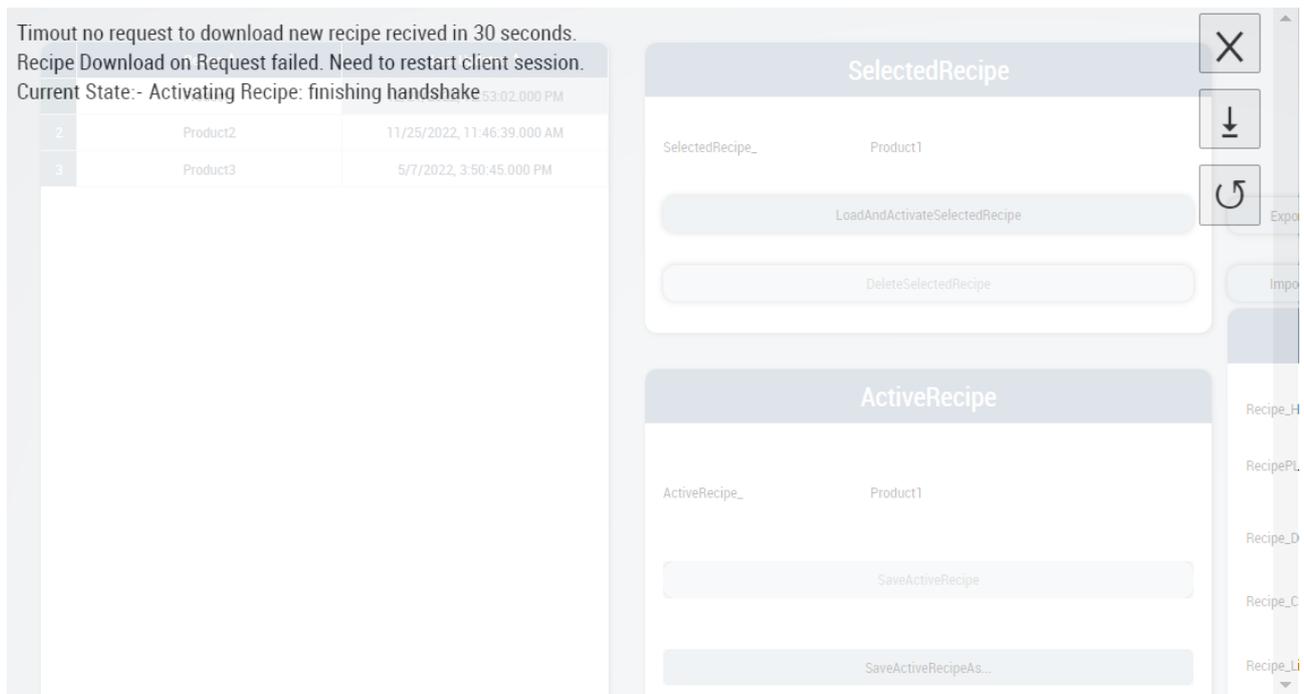
8.1.5 Fehlermeldungen auf dem Bildschirm

Alle Operationen und Funktionen des RecipeHelper-Pakets verwenden RecipeManagement-APIs für den Zugriff auf den Server.

Alle Fehler, die während der Ausführung von JavaScript auftreten, befinden sich in Client-Umgebungen und werden nicht im TwinCAT Event Logger protokolliert. Der Benutzer kann diese Fehler in der HMI nicht sehen und muss dazu das Browser-Protokollfenster öffnen.

Das RecipeHelper-Paket bietet ein Utility, das alle Fehler in einem speziellen Popup-Fenster auf der HMI anzeigt. Der Benutzer kann bei diesem Popup die folgenden Aktionen durchführen:

1. Ignorieren und schließen Sie das Popup-Fenster.
2. Laden Sie diese Fehlermeldung herunter, um sie zu protokollieren und den Support zu kontaktieren.
3. Setzen Sie das Kommunikations-Handshake zurück, um einen fehlgeschlagenen Vorgang erneut zu versuchen.



8.1.6 Rezepte sichern und wiederherstellen

Mit dem Attribut [RecipeBackupDuration \[▶ 43\]](#) kann der Benutzer entscheiden, ob er die von Beckhoff.TwinCAT.HMI.Plastic.RecipeHelper. angebotene Funktion zur Rezeptsicherung nutzen möchte. Diese Funktion ermöglicht es dem Paket, temporäre Rezeptänderungen, die vom Benutzer vorgenommen wurden, in Form von JSON-Dateien auf dem System zu sichern.

Wenn das Attribut [RecipeBackupDuration \[▶ 43\]](#) auf 10 gesetzt ist, erstellt die Rezeptsicherungsfunktion eine neue Datei mit einer Verzögerung von 10 Minuten seit der letzten Änderung durch den Benutzer. Der Dateiname enthält den Namen des zuletzt aktivierten Rezepts und den Zeitstempel der Sicherung.

Die Datensicherung erfolgt in einem Ringpufferformat. Alle Sicherungen, die älter als 72 Stunden sind, werden gelöscht.

Der Benutzer kann die Liste der Sicherungsdateien über ein [File-Explorer-Control](#) einsehen. Alle Sicherungsdateien werden in einem Ordner 'RecipeBackup' innerhalb des Projektverzeichnisses gespeichert. Um diese Dateien im File-Explorer-Control aufzulisten, muss '//RecipeBackup' als Root-Attributwert gesetzt werden.

Um Rezepte in einer beliebigen Sicherungsdatei zu aktivieren, kann der Benutzer die Funktion [ActivateBackupRecipe \[▶ 207\]](#) verwenden, die den Dateinamen der Sicherungsdatei als Eingabeparameter benötigt.

8.2 Funktionen

8.2.1 ActivateBackupRecipe

```
function ActivateBackupRecipe (ctx: TcHmi.Context, FileName: string)
```

Diese Funktion holt die mit dem Attribut FileName angegebene Sicherungsdatei der Rezepte und aktiviert die darin enthaltenen Rezepte.

Parameter

Name	Typ	Beschreibung
ctx	Required<TcHmi.Context>	Kontext-Objekt.

Name	Typ	Beschreibung
		<p>Muss bei der Verwendung in einem Trigger oder einem Funktionsausdruck nicht vom Anwender angegeben werden. Die Übergabe und die Auswertung erfolgt in diesem Fall über den entsprechenden Parser.</p> <p>Wird vom Aufrufer verwendet um dem Framework bei der Ausführung Erfolg oder Fehlschlag zu signalisieren.</p> <p>Bei direkter Verwendung der Funktion in JavaScript/TypeScript muss das Kontext-Objekt beim Aufruf übergeben und ggf. ausgewertet werden.</p>
FileName	String	<p>Dateiname der Rezeptsicherungsdatei, die sich am Speicherort der Rezeptsicherung befindet. Die in diesen Dateien enthaltenen Rezepte werden auf dem HMI aktiviert.</p>

8.2.2 SaveActiveRecipe

```
function SaveActiveRecipe(ctx: Context, TargetRecipe: string)
```

Die Funktion SaveActiveRecipe speichert das aktive temporäre Rezept `ActRcpData` unter dem durch den Parameter TargetRecipe festgelegten Rezeptnamen.

Parameter

Name	Typ	Beschreibung
ctx	Required<TcHmi.Context>	<p>Kontext-Objekt.</p> <p>Muss bei der Verwendung in einem Trigger oder einem Funktionsausdruck nicht vom Anwender angegeben werden. Die Übergabe und die Auswertung erfolgt in diesem Fall über den entsprechenden Parser.</p> <p>Wird vom Aufrufer verwendet um dem Framework bei der Ausführung Erfolg oder Fehlschlag zu signalisieren.</p> <p>Bei direkter Verwendung der Funktion in JavaScript/TypeScript muss das Kontext-Objekt beim Aufruf übergeben und ggf. ausgewertet werden.</p>
TargetRecipe	string	Target-Rezeptpfad.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

8.2.3 RcpAutoUpdate

```
function RcpAutoUpdate(symName: string, symNewValue: any, symRM: string, symDefValue: any)
```

Mit der Funktion RcpAutoUpdate wird das aktive temporäre Rezept ActRcpData eingelernt. Die Funktion prüft, ob die Variable DataValidPlc im Funktionsbaustein FB_PlcStateToHmi TRUE ist, um sicherzustellen, dass nur ein gültiger Zustand im Rezept gespeichert wird.

Parameter

Name	Typ	Beschreibung
symName	string	Symbolname des zu aktualisierenden Symbols im Rezept. Unabhängig vom Symbolnamen, der in diesem Parameter übergeben wird, wird das komplette Rezept ActRcpData aus den SPS-Werten aktualisiert.
symNewValue	any	Der in diesem Parameter übergebene Wert hat keine Auswirkungen, da das Rezept den Wert direkt aus der SPS liest.
symRM	string	Dies muss "R" sein, und es werden keine anderen Rezepttypen unterstützt.
symDefValue	any	Standardwert, wenn das Symbol zum Rezepttyp hinzugefügt wird. Dies wird nur in der Engineering-Umgebung verwendet und ist nach der Veröffentlichung des Projekts nicht mehr möglich.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

8.2.4 ListActDataTypeRecipes

```
function ListActDataTypeRecipes(ctx: TcHmi.Context, RecipeList?: TcHmi.Server.RecipeManagement.FolderRecipe)
```

Parameter

Name	Typ	Beschreibung
ctx	Required<TcHmi.Context>	Kontext-Objekt. Muss bei der Verwendung in einem Trigger oder einem Funktionsausdruck nicht vom Anwender angegeben werden. Die Übergabe und die Auswertung erfolgt in diesem Fall über den entsprechenden Parser. Wird vom Aufrufer verwendet um dem Framework bei der Ausführung Erfolg oder Fehlschlag zu signalisieren.

Name	Typ	Beschreibung
		Bei direkter Verwendung der Funktion in JavaScript/TypeScript muss das Kontext-Objekt beim Aufruf übergeben und ggf. ausgewertet werden.
recipeList	tchmi:server#/definitions/ folderRecipe	Die Referenz zu allen Rezepten. Die Liste aller Rezepte ist das Symbol <code>recipeList</code> in der Erweiterung <code>RecipeManagement</code> .

Gibt eine Liste aller Rezepte des Typs `ActRcpDataType` in einem Array von Objekten zurück, wobei jedes Objekt zwei Eigenschaften hat

- `RecipeName`: Name des aufgeführten Rezepts.
- `RecipeLastUpdatedTime`: Letzte Aktualisierungszeit für das Rezept in Form eines Strings.

Die Funktion wird auf das Attribut `SrcData` des Datagrid angewendet.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

8.2.5 LoadProductRecipe

```
function LoadProductRecipe(ctx: Tchmi.Context, NewActRecipeName: string)
```

Die Funktion `LoadProductRecipe` lädt ein Rezept und folgt der SPS-Handshake-Kommunikation, wie zuvor in diesem Dokument beschrieben.

Parameter

Name	Typ	Beschreibung
ctx	Required<Tchmi.Context>	Kontext-Objekt. Muss bei der Verwendung in einem Trigger oder einem Funktionsausdruck nicht vom Anwender angegeben werden. Die Übergabe und die Auswertung erfolgt in diesem Fall über den entsprechenden Parser. Wird vom Aufrufer verwendet um dem Framework bei der Ausführung Erfolg oder Fehlschlag zu signalisieren. Bei direkter Verwendung der Funktion in JavaScript/TypeScript muss das Kontext-Objekt beim Aufruf übergeben und ggf. ausgewertet werden.
recipeList	tchmi:server#/definitions/ folderRecipe	Die Referenz zu allen Rezepten. Die Liste aller Rezepte ist das Symbol <code>recipeList</code> in der Erweiterung <code>RecipeManagement</code> .

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

8.2.6 DeleteActRcp

```
function DeleteActRcp(ctx: Tchmi.Context, RecipeName: string)
```

Diese Funktion ermöglicht das Löschen eines Rezepts vom Server, wobei zur Sicherheit das aktuell aktivierte Rezept nicht vom Server gelöscht werden kann.

Parameter

Name	Typ	Beschreibung
ctx	Required<Tchmi.Context>	<p>Kontext-Objekt.</p> <p>Muss bei der Verwendung in einem Trigger oder einem Funktionsausdruck nicht vom Anwender angegeben werden. Die Übergabe und die Auswertung erfolgt in diesem Fall über den entsprechenden Parser.</p> <p>Wird vom Aufrufer verwendet um dem Framework bei der Ausführung Erfolg oder Fehlschlag zu signalisieren.</p> <p>Bei direkter Verwendung der Funktion in JavaScript/TypeScript muss das Kontext-Objekt beim Aufruf übergeben und ggf. ausgewertet werden.</p>
RecipeName	string	Name des zu löschenden Rezepts.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

8.2.7 ExportRecipe

```
function ExportRecipe(ctx: Context, recipeReference: string)
```

Diese Funktion exportiert ein Rezept in eine JSON-Datei und lädt es auf das Client-System herunter.

Parameter

Name	Typ	Beschreibung
ctx	Required<Tchmi.Context>	<p>Kontext-Objekt.</p> <p>Muss bei der Verwendung in einem Trigger oder einem Funktionsausdruck nicht vom Anwender angegeben werden. Die Übergabe und die Auswertung erfolgt in diesem Fall über den entsprechenden Parser.</p> <p>Wird vom Aufrufer verwendet um dem Framework bei der Ausführung Erfolg oder Fehlschlag zu signalisieren.</p> <p>Bei direkter Verwendung der Funktion in JavaScript/TypeScript muss das Kontext-Objekt beim Aufruf übergeben und ggf. ausgewertet werden.</p>

Name	Typ	Beschreibung
recipeReference	tchmi:framework#/definitions/RecipeReference	Die Referenz zu dem Rezept, welches heruntergeladen werden soll. Die Referenz besteht aus dem Rezeptpfad (mit Ordnerstruktur).

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

8.2.8 ImportRecipe

```
function ImportRecipe (ctx: Tchmi.Context)
```

Mit dieser Funktion können mehrere JSON-Dateien, die mit der Funktion `ExportRecipe` exportiert wurden, importiert und dem Server hinzugefügt werden.

Parameter

Name	Typ	Beschreibung
ctx	Required<Tchmi.Context>	Kontext-Objekt. Muss bei der Verwendung in einem Trigger oder einem Funktionsausdruck nicht vom Anwender angegeben werden. Die Übergabe und die Auswertung erfolgt in diesem Fall über den entsprechenden Parser. Wird vom Aufrufer verwendet um dem Framework bei der Ausführung Erfolg oder Fehlschlag zu signalisieren. Bei direkter Verwendung der Funktion in JavaScript/TypeScript muss das Kontext-Objekt beim Aufruf übergeben und ggf. ausgewertet werden.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

8.3 Verwendung von RecipeHelper mit HMI-Controls

8.3.1 Auflistung von Rezepten in einem Datagrid

	Recipe :	Last Updated :
1	Test1	6.7.2022, 13:45:14,000
2	testmould-backup21	30.6.2022, 16:04:54,000

1. Instanz eines TcHmiDataGrid-Controls erstellen.
2. Legen Sie das folgende JSON-Objekt als Attribut Columns fest.

```
[
{
  "name": "RecipeName",
  "label": "%1%Recipe%/1%",
  "control": "TextBlock",
  "width": 50,
  "widthUnit": "%",
  "resize": false,
  "sortable": true,
  "minWidth": 100,
  "minWidthUnit": "px",
  "cellBackground": null,
  "textColor": null,
  "editable": false,
  "horizontalAlignment": "Center",
  "verticalAlignment": "Center",
  "headerHorizontalAlignment": "Center",
  "headerVerticalAlignment": "Center"
},
{
  "name": "RecipeLastUpdatedTime",
  "label": "Last Updated",
  "control": "TextBlock",
  "width": 50,
  "widthUnit": "%",
  "resize": false,
  "sortable": true,
  "minWidth": 100,
  "minWidthUnit": "px",
  "cellBackground": null,
  "textColor": null,
  "editable": false,
  "horizontalAlignment": "Center",
  "verticalAlignment": "Center",
  "headerHorizontalAlignment": "Center",
  "headerVerticalAlignment": "Center",
  "format": {
```

```
"objectType": "Function",
"active": true,
"fn": "TcHmi.Functions.Beckhoff.ToDateTimeString",
"fnParams": [
{
"objectType": "StaticValue",
"valueType": "tchmi:general#/definitions/Locale",
"value": null
}
]
}
}
]
```

3. Legen Sie die Funktionsbindung wie folgt als Attribut `SrcData` fest.

```
"%f%TcHmi.Functions.Plastic.ListActDataTypeRecipes(%s%TcHmiRecipeManagement.Config::recipeList%/s)%/f%"
```

8.3.2 Aktives Rezept mit ungespeicherten Änderungen anzeigen

1. Binden Sie einen Textblock an das Serversymbol `LastActivatedRecipe`. Dadurch wird das zuletzt aktivierte Rezept angezeigt.
2. Das Symbol `UnsavedChangesInActRcpData` zeigt an, dass es ungespeicherte Änderungen gibt, wenn es `TRUE` ist, und kann verwendet werden, um ein Sternchen hinter dem aktiven Rezeptnamen hinzuzufügen.

Product2*

9 RFID

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.RFID

Verfügbar: seit Version 12.14.0

Das NuGet-Paket **Beckhoff.TwinCAT.HMI.Plastic.RFID** kann über das Paketverwaltungssystem NuGet im TwinCAT HMI Projekt installiert werden.

9.1 MiFare Classic

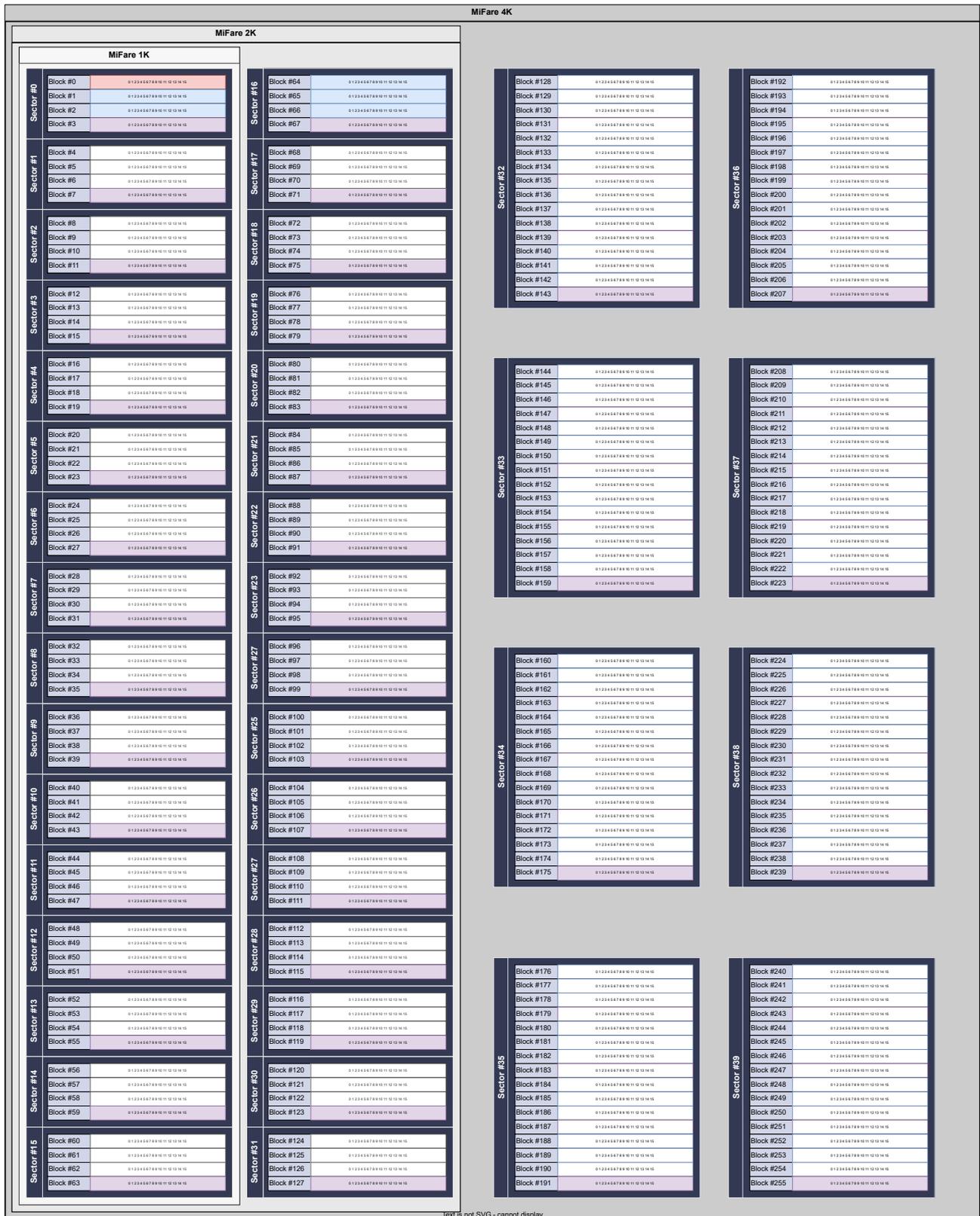
MiFare (ISO 14443A) ist eine kontaktlose RFID-Chipkartentechnik. Die Karten werden in verschiedene Speichergrößen unterteilt.

Kartentyp	Sektoren	Blöcke pro Sektor
MiFare Classic 1K	16	4
MiFare Classic 2K	32	4
MiFare Classic 4K	32 + 8	4 (Sektor 0-31) 16 (Sektor 32-39)

Ein Block besteht aus jeweils 16 Byte. Der letzte Block eines Sektors (*Sector Trailer*) enthält zwei Authentifizierungsschlüssel, um die Sektoren vor unzulässigen Zugriffen zu schützen. Die Sector Trailer dürfen nicht überschrieben werden, da die zugehörigen Sektoren unbrauchbar gemacht werden können. Eine besondere Rolle spielen auch die Sektoren 0 und 16. Diese Sektoren verwalten Speicher, die für Multi-Applikationen wichtig sind. In Sektor 0 sind darüber hinaus Herstellerdaten, inklusive der UID gespeichert.

HINWEIS

Beschreiben Sie nur die weiß markierten Bereiche.



9.2 ConnectToComPort

async function ConnectToComPort(ctx: any, vendorId?: number, productId?: number)

Die asynchrone Funktion ConnectToComPort wird verwendet, um eine Verbindung zwischen dem Browser und einer seriellen Schnittstelle herzustellen. Mithilfe der Hersteller- und Produkt-ID kann die Abfrage vorgefiltert werden. Wird die Funktion ausgeführt, öffnet sich ein Abfragefenster, in dem ein Port ausgewählt werden kann. Bei erfolgreicher Ausführung wird der verbundene Port zurückgegeben.



Sollte der Port bereits verbunden sein, wird dies im Popup durch den Hinweis „Paired“ angezeigt.

i **Browserkompatibilität**

Die Funktion ist nur mit Google Chrome und Microsoft Edge kompatibel.

Parameter

Name	Typ	Beschreibung
ctx	any, <u>Context</u>	Asynchrones Context Objekt
vendorId	<u>number</u>	Optional: Die Vendor-ID wird verwendet, um die Auswahl der Ports nach bestimmten Herstellern zu filtern.
productId	<u>number</u>	Optional: Die Product-ID wird verwendet, um die Auswahl der Ports nach bestimmten Produkten zu filtern.

Rückgabewert

Typ	Beschreibung
<u>Navigator.Serial.SerialPort</u>	Port Objekt aus der Serial Klasse.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.RFID

Verfügbar: seit Version 12.14.0

Verwendung: TC3 Plastic Application HMI Projekte

9.3 GetConnectedComPortInfo

```
async function GetConnectedComPortInfo(ctx: any): [{usbVendorId: number | undefined; usbProductId: number | undefined}]
```

Die asynchrone Funktion GetConnectedComPortInfo wird verwendet, um Informationen über die verbundenen seriellen Schnittstellen zu erhalten. Die Funktion gibt ein Array zurück, das die Hersteller- und Produkt-ID der verbundenen Ports enthält.

i **Browserkompatibilität**

Die Funktion ist nur mit Google Chrome und Microsoft Edge kompatibel.

Parameter

Name	Typ	Beschreibung
ctx	any, <u>Context</u>	Asynchrones Context Objekt

Rückgabewert

Typ	Beschreibung
<u>object</u> []	Objekt-Array, mit den Attributen „usbVendorId“ und „usbProductId“.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.RFID

Verfügbar: seit Version 12.14.0

Verwendung: TC3 Plastic Application HMI Projekte

9.4 ReadMiFareData

```
async function ReadMiFareData(ctx: any, startBlockAddress: number,
numberOfBlocks: number, usbConfig: any): {payload: string | null; serialNumber: string | null}
```

Die asynchrone Funktion ReadMiFareData liest auf einer MiFare Classic Standard-Karte gespeicherte Daten über die mit dem Browser verbundene serielle Schnittstelle aus. Die Daten werden über einen Speicherbereich ausgelesen, der mit der Adresse und der Anzahl der zu lesenden Blöcke vorgegeben wird. Ein Block entspricht hierbei 16 Byte, die Adresse gibt den Sektor an, in dem die Daten liegen.

Browserkompatibilität

Die Funktion ist nur mit Google Chrome und Microsoft Edge kompatibel.

Parameter

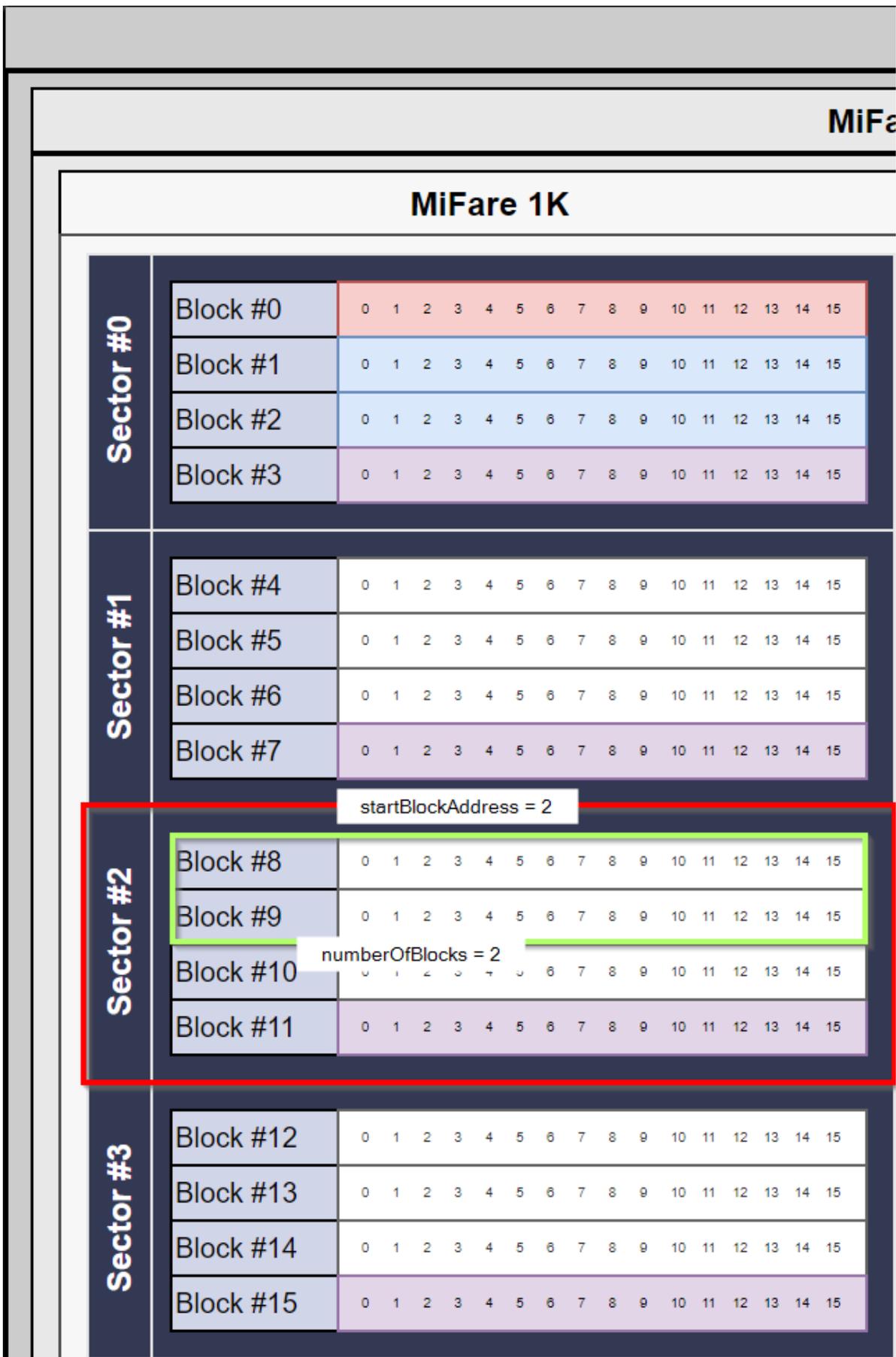
Name	Typ	Beschreibung
ctx	any, <u>Context</u>	Asynchrones Context Objekt
startBlockAddress	<u>number</u>	Sektor, in dem sich der Block mit dem zu lesenden Datensatz befindet.
numberOfBlocks	<u>number</u>	Anzahl der zu lesenden Blöcke.
usbConfig	<u>RFIDConfig</u> ▶ 224	Konfigurationsdaten der verbundenen seriellen Schnittstelle.

Rückgabewert

Typ	Beschreibung
<u>object</u>	{serialNumber: string; payload; string}

Beispiel:

Es soll ein Datensatz von 32 Byte aus Sektor 2 ausgelesen werden. Aus folgendem Bild ergeben sich die Parameter „startBlockAddress = 2“ und „numberOfBlocks = 2“. Der USB Port wurde bereits mit der Funktion ConnectToComPort verbunden und die Hersteller ID (4292) und Produkt ID (60000) über „GetConnectedComPortInfo“ ausgelesen. Die Baudrate muss aus dem Datenblatt der seriellen Schnittstelle ermittelt werden. Der keyType wird durch die benutzte MiFare Classic-Karte definiert - der Authentifizierungsschlüssel aus der Applikation (als Standardwert wird hier 281474976710655.0 gesetzt, was dem hexadezimalen Wert 0xFFFF FFFF FFFF entspricht).



```
const readMiFare = TcHmi.Functions.getFunction('TcHmi.Functions.Plastic.ReadMiFareData');
const data = await readMiFare(
  ctx,
```

```

2,
2,
{
  vendorId: 4292,
  productId: 60000,
  baudrate: 9600,
  keyType: 'A',
  authenticationKey = 281474976710655.0
}
);

console.log(data);

```

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.RFID

Verfügbar: seit Version 12.14.0

Verwendung: TC3 Plastic Application HMI Projekte

9.5 WriteMiFareData

```

async function WriteMiFareData(ctx: any, startBlockAddress: number, numberOfBlocks: number, dataToWrite: string | Uint8Array, usbConfig: any): {serialNumber: string; payload: string}

```

Die asynchrone Funktion WriteMiFareData schreibt Daten auf eine MiFare Classic Standard-Karte über die mit dem Browser verbundene serielle Schnittstelle. Die Daten werden über einen Speicherbereich gespeichert, der mit der Adresse und der Anzahl der zu schreibenden Blöcke vorgegeben wird. Ein Block entspricht hierbei 16 Byte, die Adresse gibt den Sektor an, in dem die Daten gespeichert werden.



Browserkompatibilität

Die Funktion ist nur mit Google Chrome und Microsoft Edge kompatibel.

Parameter

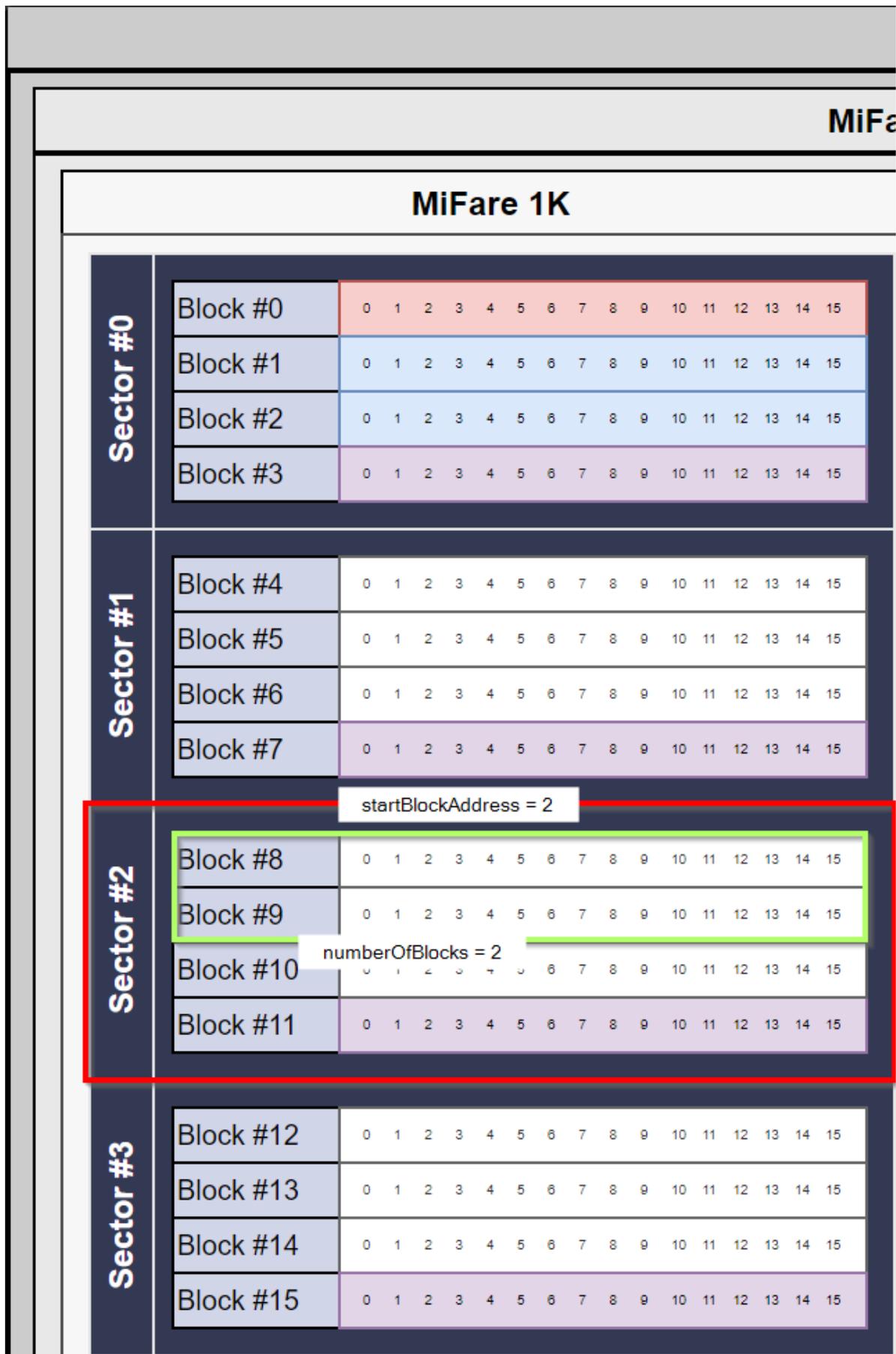
Name	Typ	Beschreibung
ctx	any, Context	Asynchrones Context Objekt
startBlockAddress	number	Sektor, in dem sich der Block mit dem zu lesenden Datensatz befindet.
numberOfBlocks	number	Anzahl der zu lesenden Blöcke.
dataToWrite	string	Datensatz, der auf der Karte gespeichert werden soll.
usbConfig	RFIDConfig ▶ 224	Konfigurationsdaten der verbundenen seriellen Schnittstelle.

Rückgabewert

Typ	Beschreibung
object	{serialNumber: string; payload; string}

Beispiel:

Der Datensatz (`dataToWrite = „Loremipsumdolorsitametconsectetu“`) der Größe 32 Byte soll auf Sektor 2 geschrieben werden. Aus folgendem Bild ergeben sich die Parameter „`startBlockAddress = 2`“ und „`numberOfBlocks = 2`“. Der USB-Port wurde bereits mit der Funktion `ConnectToComPort` verbunden und die Hersteller-ID (4292) und Produkt-ID (60000) über „`GetConnectedComPortInfo`“ ausgelesen. Die Baudrate muss aus dem Datenblatt der seriellen Schnittstelle ermittelt werden – in diesem Beispiel 9600. Der `keyType` wird durch die benutzte MiFare Classic-Karte definiert (hier: ‚A‘ - der Authentifizierungsschlüssel aus der Applikation (als Standardwert wird hier 281474976710655.0 gesetzt, was dem hexadezimalen Wert 0xFFFF FFFF FFFF entspricht).



```
const dataToWrite = 'Loremipsumdolorsitametconsectetu';
const writeMiFare = TcHmi.Functions.getFunction('TcHmi.Functions.Plastic.WriteMiFareData');

const data = await readMiFare(
```

```

    ctx,
    2,
    2,
    dataToWrite,
    {
        vendorId: 4292,
        productId: 60000,
        baudrate: 9600,
        keyType: 'A',
        authenticationKey = 281474976710655.0
    }
);

console.log(data);

```

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.RFID

Verfügbar: seit Version 12.14.0

Verwendung: TC3 Plastic Application HMI Projekte

9.6 LoginWithMiFare

```

async function LoginWithMiFare(ctx: any, usernameAddress: number, usernameBlockAmount: number, passwordAddress: number, passwordBlockAmount: number, reloadOnLogin: boolean, persistentLogin: boolean, usbConfig: any)

```

Die asynchrone Funktion `LoginWithMiFare` wird verwendet, um dem Benutzer das Einloggen mit einer MiFare Classic Standard-Karte zu ermöglichen. Hierfür muss ein entsprechender Benutzer im TwinCAT HMI Engineering angelegt sein ([mehr](#)). Das Passwort und der Benutzername müssen auf der Karte gespeichert und passend codiert sein. Zum Schreiben der Karte sollte somit die Funktion [WriteMiFareData](#) [► 220] genutzt werden.



Browserkompatibilität

Die Funktion ist nur mit Google Chrome und Microsoft Edge kompatibel.

Parameter

Name	Typ	Beschreibung
ctx	any, Context	Asynchrones Context Objekt
usernameAddress	number	Sektor, in dem sich der Block mit dem Benutzernamen befindet.
usernameBlockAmount	number	Anzahl der zu lesenden Blöcke.
passwordAddress	number	Sektor, in dem sich der Block mit dem Passwort befindet.
passwordBlockAmount	number	Anzahl der zu lesenden Blöcke.
reloadOnLogin	boolean	Ist der Wert <code>true</code> , wird die Seite nach dem Login neu geladen.
persistentLogin	boolean	Ist der Werte <code>true</code> , bleibt der Benutzer auch nach dem Schließen des Browsers angemeldet.
usbConfig	RFIDConfig [► 224]	Konfigurationsdaten der verbundenen seriellen Schnittstelle.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.RFID

Verfügbar: seit Version 12.14.0

Verwendung: TC3 Plastic Application HMI Projekte

9.7 EnableAutoLogin

```
function EnableAutoLogin(ctx: any, usernameAddress: number, usernameBlockAmount: number, passwordAddress: number, passwordBlockAmount: number, reloadOnLogin: boolean, persistentLogin: boolean, usbConfig: any)
```

Die Funktion `EnableAutoLogin` wird verwendet, um ein automatisches Einloggen per MiFare Classic Standard RFID-Karte zu ermöglichen. Zu diesem Zweck wird die [LoginWithMiFare \[▶ 222\]](#) Funktion zyklisch angetriggert. Somit muss der Login nicht über ein Event angestoßen werden.



Browserkompatibilität

Die Funktion ist nur mit Google Chrome und Microsoft Edge kompatibel.

Parameter

Name	Typ	Beschreibung
ctx	any, Context	Asynchrones Context Objekt
usernameAddress	number	Sektor, in dem sich der Block mit dem Benutzernamen befindet.
usernameBlockAmount	number	Anzahl der zu lesenden Blöcke.
passwordAddress	number	Sektor, in dem sich der Block mit dem Passwort befindet.
passwordBlockAmount	number	Anzahl der zu lesenden Blöcke.
reloadOnLogin	boolean	Ist der Wert <code>true</code> , wird die Seite nach dem Login neu geladen.
persistentLogin	boolean	Ist der Werte <code>true</code> , bleibt der Benutzer auch nach dem Schließen des Browsers angemeldet.
usbConfig	RFIDConfig [▶ 224]	Konfigurationsdaten der verbundenen seriellen Schnittstelle.

Rückgabewert

Typ	Beschreibung
number	ID des angemeldeten Intervalls.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.RFID

Verfügbar: seit Version 12.14.0

Verwendung: TC3 Plastic Application HMI Projekte

9.8 DisableAutoLogin

```
function DisableAutoLogin(IntervalId: number)
```

Meldet das mit [EnableAutoLogin \[▶ 223\]](#) erzeugte Intervall wieder ab und beendet somit die Auto Login Funktionalität.



Browserkompatibilität

Die Funktion ist nur mit Google Chrome und Microsoft Edge kompatibel.

Parameter

Name	Typ	Beschreibung
IntervallId	<u>number</u>	ID des angemeldeten Login Intervalls.

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.RFID

Verfügbar: seit Version 12.14.0

Verwendung: TC3 Plastic Application HMI Projekte

9.9 Spezifische Datentypen

9.9.1 RFIDConfig

[Object]

```
RFIDConfig {
  vendorId: number;
  productId: number;
  baudrate: number;
  keyType: 'A' | 'B';
  authenticationKey: number
}
```

Eigenschaften

Name	Typ	Beschreibung	Standardwert
vendorId	<u>number</u>	ID des Herstellers des RFID-Readers.	4292
productId	<u>number</u>	Produkt ID des RFID-Readers.	60000
baudrate	<u>number</u>	Baudrate des RFID-Readers.	9600
keyType	‚A‘ ‚B‘	Typ der RFID-Karte.	‚A‘
authenticationKey	<u>number</u>	Authentifizierungsschlüssel (Multiplikation) der RFID-Karte.	281474976710655.0 0xFFFF FFFF FFFF

Schema: tchmi:framework#/definitions/RFIDConfig

Verfügbar: seit Version 12.14.0

10 Themes

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Themes

Verfügbar: seit Version 12.6.0

Das NuGet-Paket Beckhoff.TwinCAT.HMI.Plastic.Themes kann über das Paketverwaltungssystem NuGet im TwinCAT HMI Projekt installiert werden.

Es beinhaltet Designanpassungen bestehender Controls und Klassen, die im TwinCAT HMI Projekt verwendet werden können.

Verwendung des Plastic-Designs

Um das Design aus dem NuGet-Paket in einem TwinCAT HMI Projekt anwenden zu können, muss zunächst ein Theme mit dem Namen „Plastic“ angelegt werden. Dafür führen Sie die folgenden Schritte durch:

1. Klicken Sie mit einem Rechtsklick auf den Ordner „Themes“ im TwinCAT HMI Projekt und fügen Sie ein neues Item hinzu.
 2. Wählen Sie die Option **Theme** bei dem sich öffnenden Fenster aus, vergeben Sie den Namen „Plastic“ und klicken Sie danach auf den Button **Add**.
- ⇒ Das Theme mit dem Namen „Plastic“ erscheint im Ordner „Themes“ und kann mithilfe des Theme-Editors bearbeitet werden.

Verwendung von Klassen-Themes

Zur Verwendung von Klassen-Themes aus dem NuGet-Paket müssen diese im Theme-Editor eingefügt werden. Folgende Schritte sind dafür erforderlich:

1. Öffnen Sie die Datei Plastic.theme im Theme-Editor.
 2. Klicken Sie per Rechtsklick auf den Reiter **Control Classes** und wählen Sie **Create new Control Class** aus.
 3. Fügen Sie nun die folgenden Control-Klassen hinzu, um sie im Projekt verwenden zu können: `inputbox_toggle_switch`; `SliderTab`; `Tiles`; `TilesHeadline`
- ⇒ Nun können die Control-Klassen den Controls über das Property **ClassNames** zugeordnet werden.



Die hinzugefügten Control-Klassen werden im Theme-Editor ausgegraut dargestellt, da sie keine Definition enthalten. Die Definition der zuvor hinzugefügten Control-Klassen über den Themes-Editor ist durch die Verwendung des Beckhoff.TwinCAT.HMI.Plastic.Themes NuGet-Paketes nicht nötig, da das Paket die Definition enthält.

11 Temperatur

Das Paket `Beckhoff.TwinCAT.HMI.Plastic.Temperature` bietet spezielle Controls für die temperaturbezogene Visualisierung in einem TwinCAT-HMI-Projekt.

Die für diese Temperatur-Controls erforderliche SPS-Schnittstelle wird [hier](#) erläutert. Das Symbol vom Typ `FB_TemperatureHmi` muss auf das Attribut `TemperaturSymbol` [[▶ 44](#)] eines Configurator-Controls gesetzt werden.

11.1 PfwTempParameters

NuGet Paket: Beckhoff.TwinCAT.HMI.Plastic.Temperature

Verfügbar: ab Version 12.6.0

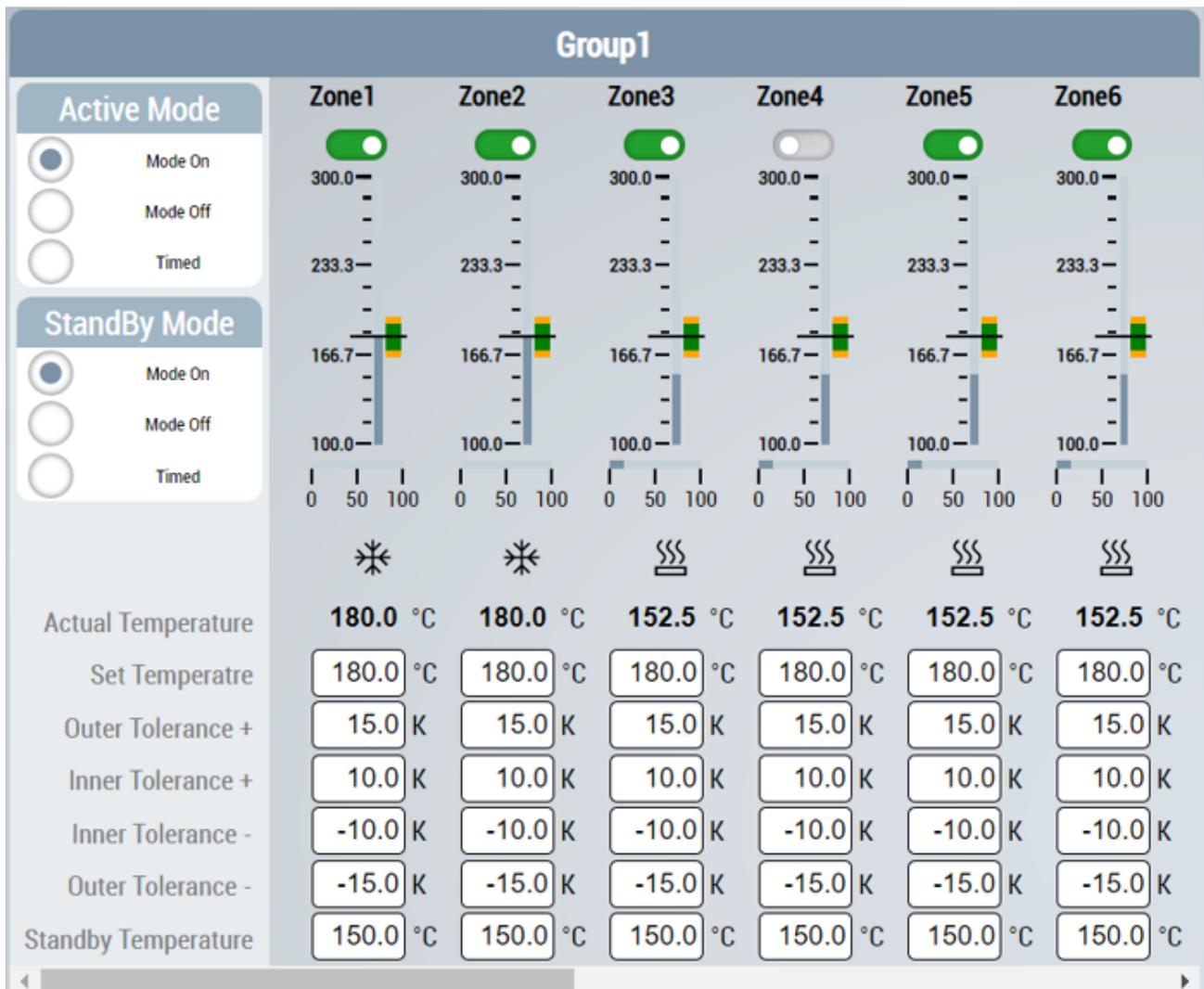
Das `PfwTempParameters`-Control wird verwendet, um verschiedene Betriebsparameter der in der SPS konfigurierten Temperaturgruppen und -zonen zu visualisieren und dem Benutzer die Möglichkeit zu geben, diese zu ändern.

Dieses Control arbeitet mit dem mittlerweile obsoleten `FB_TempCtrlHmi` in der SPS.

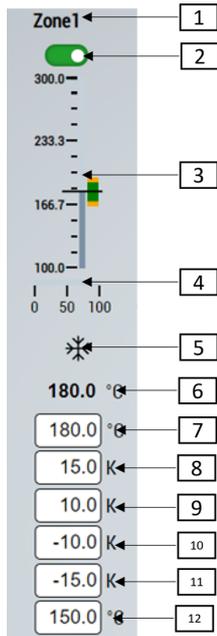
HINWEIS

Obsolet: Dieses Control und ihre SPS-Schnittstelle sind mittlerweile obsolet und werden für neue Projekte nicht empfohlen.

Bitte verwenden Sie stattdessen das [ZoneConfiguration](#) [[▶ 234](#)]-Control.



Jede Temperaturzone in der gewählten Temperaturzonengruppe wird mit verschiedenen Eingabe- und Grafik-Controls visualisiert.



1. Temperaturzonen Eigenschaft ZoneName.
2. Mit diesem Control kann der Benutzer eine Zone aktivieren/deaktivieren.
3. Eine Linearanzeige visualisiert die Ist-Temperatur, die Soll-Temperatur sowie die inneren und äußeren Toleranzbereiche auf einer Skala an.
4. Eine Anzeige visualisiert die Leistungseigenschaften der Zone.
5. Der Kühl-/Heizstatus der Temperaturzonen wird mit einem entsprechenden Symbol angezeigt.
6. Ist-Temperatur der Zone.
7. Der Benutzer kann die Solltemperatur für die jeweilige Zone einstellen.
8. Äußere Toleranz in positiver Richtung.
9. Innere Toleranz in positiver Richtung.
10. Innere Toleranz in negativer Richtung.
11. Äußere Toleranz in negativer Richtung.
12. Standby-Temperatur der Zone.

11.1.1 Attribute

Kategorie: Config

Name	Beschreibung
ActualZoneGroupId [► 228]	Index der auf dem Control angezeigten Zonengruppe.
DisplayOperationModes [► 228]	Mit diesem Attribut können die Bedienelemente hinzugefügt oder aus dem Control entfernt werden.
DigitsAfterDecimal [► 228]	Einzelne Werte zeigen die Displayauflösung an.

Benutzeränderungen loggen:

Name	Beschreibung
LogChangesByUser [► 228]	Logging der vom Benutzer auf dem Client vorgenommenen Änderungen in einer Logdatei.

11.1.1.1 ActualZoneGroupId

Mit diesem Attribut kann der Benutzer den Index der konfigurierten Temperaturzonengruppe ändern, der auf dem Control angezeigt wird. Durch Ändern dieses Wertes kann der Benutzer alle möglichen Temperaturzonengruppen im Projekt anzeigen und bearbeiten.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getActualZoneGroupId

Attribut-Setter: setActualZoneGroupId

Verfügbar: seit Version 12.6.0

11.1.1.2 DigitsAfterDecimal

Mit diesem Attribut kann die Auflösung der einzelnen Zahlenwerte, die auf dem Control angezeigt werden, aktualisiert werden.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getDigitsAfterDecimal

Attribut-Setter: setDigitsAfterDecimal

Verfügbar: seit Version 12.6.0

11.1.1.3 DisplayOperationModes

Standardmäßig werden die Bedienelemente, mit denen der aktive Modus und der Standby-Modus der ausgewählten Temperaturzonengruppe eingestellt werden, nicht auf dem Control angezeigt. Mit diesem Attribut können die Bedienelemente für den Benutzer zur Verfügung gestellt werden.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getDisplayOperationModes

Attribut-Setter: setDisplayOperationModes

Verfügbar: seit Version 12.6.0

11.1.1.4 LogUserChanges

Wenn dieses Attribut auf `TRUE` gesetzt ist, loggt das Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket jede Änderung durch den Benutzer auf dem HMI-Bildschirm in einer Logdatei.

Um dieses Attribut zu verwenden, muss [UserLoggerSymbol](#) [► 45] mit einem Binding an [FB_UserLoggingHmi](#) konfiguriert werden.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getLogUserChanges

Attribut-Setter: setLogUserChanges

Verfügbar: ab Version 12.12.0

11.2 TemperatureScheduler

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Temperature

Verfügbar: ab Version 12.14.0

Das Framework-Control TemperatureScheduler ist eine Erweiterung des Framework-Controls ProcessScheduler [► 126] mit zusätzlichen Funktionen, die sich besser für die Planung von Temperaturzonengruppen zu verschiedenen Zeiten der Woche eignen.

Der Benutzer kann jedem Zeitplanelement verschiedene Temperaturgruppen zuordnen und festlegen, ob die Gruppen während eines Zeitplans im Standby-Modus arbeiten sollen.



1. Einzelnes Zeitplanobjekt mit ausgewähltem Standby-Modus
2. Aktueller Wochentag
3. Wochentage
4. Stunden
5. Liste der mit dem Zeitplan verbundenen Temperaturgruppen.
6. Zeitliche Dauer des Zeitplans
7. Aktuelle Zeit

11.2.1 Attribute

11.2.1.1 ChangeLogDescriptor

Deskriptortext, der für den Eintrag im Änderungs-Log anstelle des Symbolpfads verwendet wird.

Schema: "tchmi:general#/definitions/STRING"

Attribut-Getter: getChangeLogDescriptor

Attribut-Setter: setChangeLogDescriptor

Verfügbar: ab Version 12.14.0

11.2.1.2 LogUserChanges

Wenn dieses Attribut auf `TRUE` gesetzt ist, loggt das Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket jede Benutzeränderung auf dem HMI-Bildschirm in einer Logdatei.

Um dieses Attribut zu verwenden, muss `UserLoggerSymbol` [► 45] mit einem Binding an `FB_UserLoggingHmi` konfiguriert werden.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: `getLogUserChanges`

Attribut-Setter: `setLogUserChanges`

Verfügbar: ab Version 12.14.0

11.2.1.3 SelectedTimerIndex

Dieses Attribut gibt den Array-Index des Zeitplans wieder, den der Benutzer auf dem Bildschirm ausgewählt hat. Durch Auswahl eines Zeitplanobjekts kann der Benutzer diesen Zeitplan bearbeiten oder aus dem Array entfernen.

Schema: `tchmi:general#/definitions/Number`

Attribut-Getter: `getSelectedTimerIndex`

Attribut-Setter: `setSelectedTimerIndex`

Verfügbar: ab Version 12.14.0

11.2.1.4 TimerList

Dieses Attribut sollte mit einem Array-Symbol auf der SPS verbunden werden, wobei jedes Element die Konfiguration eines Zeitplans speichert. Für das Projekt Plastic Application PLC sollte ein Symbol für das Array `FB_TimerTempHmi` an diese Eigenschaft gebunden werden.

Eigenschaften der einzelnen Elemente in diesem Array:

Name	Typ	Beschreibung
Startzeit	LREAL	Startzeit in Sekunden.
EndTimer	LREAL	Endzeit in Sekunden.
Weekday	BYTE	Ein BYTE, das zur Auswahl der Wochentage für den Zeitplan verwendet wird. Das 1. Bit (LSB) ist gesetzt, um anzuzeigen, dass der Zeitplan am Montag verwendet wird. In ähnlicher Weise ist das 7. Bit gesetzt, um anzuzeigen, dass der Zeitplan am Sonntag verwendet wird. Ein Zeitplan, der am Montag, Dienstag und Sonntag verwendet wird, hat zum Beispiel den Wert "2#0100 0011".
Standby	BOOL	Ob der Zeitplan seine Member-Temperaturgruppen in den Standby-Modus versetzen soll.
aMembers	Array of INT	Liste der Member-Gruppenindizes, die mit diesem Zeitplanobjekt geplant sind.

Schema: tchmi:general#/definitions/Symbol

Attribut-Getter: getTimerList

Attribut-Setter: setTimerList

Verfügbar: ab Version 12.14.0

11.2.2 Funktionen

11.2.2.1 ActivateConfig

Durch den Aufruf dieser Funktion werden alle Benutzeränderungen aktiviert und in die SPS-Symbole geschrieben.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: ab Version 12.14.0

11.2.2.2 ResetConfigdata

Mit dieser Funktion werden alle Änderungen des Benutzers rückgängig gemacht, indem die zuletzt aktivierte Konfiguration erneut geladen wird.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: ab Version 12.14.0

11.2.2.3 AddScheduleData

Mit dieser Funktion wird ein Pop-up-Fenster auf dem Control angezeigt, das die Erstellung eines neuen Zeitplans ermöglicht.

Der Benutzer benötigt das Zugriffsrecht **Operate**, um diese Funktion aufzurufen.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: ab Version 12.14.0

11.2.2.4 EditScheduleData

Mit dieser Funktion wird ein Pop-up-Fenster auf dem Control angezeigt, das die Bearbeitung des ausgewählten Zeitplans ermöglicht. Der Benutzer muss zunächst einen bestehenden Zeitplan auf dem Control auswählen und dann diese Funktion aufrufen.

Zur Ausführung dieser Funktion benötigt der Benutzer den Zugriff **Operate**.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: ab Version 12.14.0

11.2.3 Ereignisse**11.2.3.1 onUserConfigActivated**

Dieses Ereignis wird generiert, wenn eine neue Konfiguration auf dem Control aktiviert wird.

Verfügbar: ab Version 12.14.0

11.2.3.2 onEditingCanceled

Dieses Ereignis wird generiert, wenn Benutzeränderungen rückgängig gemacht werden und die alte Konfiguration wieder auf das Control geladen wird.

Verfügbar: ab Version 12.14.0

11.2.4 Dialoge

11.2.4.1 Neuen Zeitplan hinzufügen

1. [Dropdown menu with value 1]

2. Monday Tuesday Wednesday Thursday Friday Saturday Sunday

Start time: hr min

End time: hr min

5. Use standby temperature

Index	Group name	Select
1	Group 1	<input checked="" type="checkbox"/> (6)
2	Group 2	<input checked="" type="checkbox"/>
3	Group 3	<input type="checkbox"/>
4	Group 4	<input type="checkbox"/>
5	Group 5	<input type="checkbox"/>

7. Ok 8. Cancel Remove

1. Wählen Sie den Array-Index des hinzuzufügenden Zeitplans.
2. Je nach Checkbox wird der Zeitplan mit den ausgewählten Tagen konfiguriert.
3. Wählen Sie die Startzeit des Zeitplans.
4. Wählen Sie die Endzeit des Zeitplans.
5. Wählen Sie aus, ob der Zeitplan Membergruppen im Standby-Modus konfiguriert.
6. Wählen Sie die Temperaturgruppen, die Member des neuen Zeitplans sind.
7. Fügen Sie einen neuen Zeitplan hinzu und schließen Sie den Dialog.
8. Änderungen abrechnen und Dialog schließen.
9. Die neue Konfiguration muss durch Ausführen der Funktion [ActivateConfig](#) [► 231] aktiviert werden.

Verfügbar: ab Version 12.14.0

11.2.4.2 Bearbeiten oder Entfernen eines Zeitplans

The screenshot shows a configuration window for editing a schedule. At the top, a dropdown menu displays the number '2'. Below it, a row of checkboxes is checked for every day of the week: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday. The 'Start time' is set to 4 hours and 0 minutes, and the 'End time' is set to 6 hours and 30 minutes. A checkbox labeled 'Use standby temperature' is also checked. Below these settings is a table with the following data:

Index	Group name	Select
1	Group 1	<input checked="" type="checkbox"/>
2	Group 2	<input type="checkbox"/>
3	Group 3	<input type="checkbox"/>
4	Group 4	<input checked="" type="checkbox"/>
5	Group 5	<input checked="" type="checkbox"/>

At the bottom of the window are three buttons: 'Ok', 'Cancel', and 'Remove'.

1. Der zu bearbeitende Zeitplan kann durch Antippen oder Anklicken ausgewählt werden.
2. Bei der Ausführung der Funktion [EditScheduleData](#) [► 232] erscheint auf dem Bildschirm das Pop-up-Fenster zur Bearbeitung des ausgewählten Zeitplanobjekts.
3. Ändern Sie die Parameter oder klicken Sie auf die Schaltfläche Remove, um den ausgewählten Zeitplan aus der Liste zu entfernen.
4. Die neue Konfiguration muss durch Ausführen der Funktion [ActivateConfig](#) [► 231] aktiviert werden.

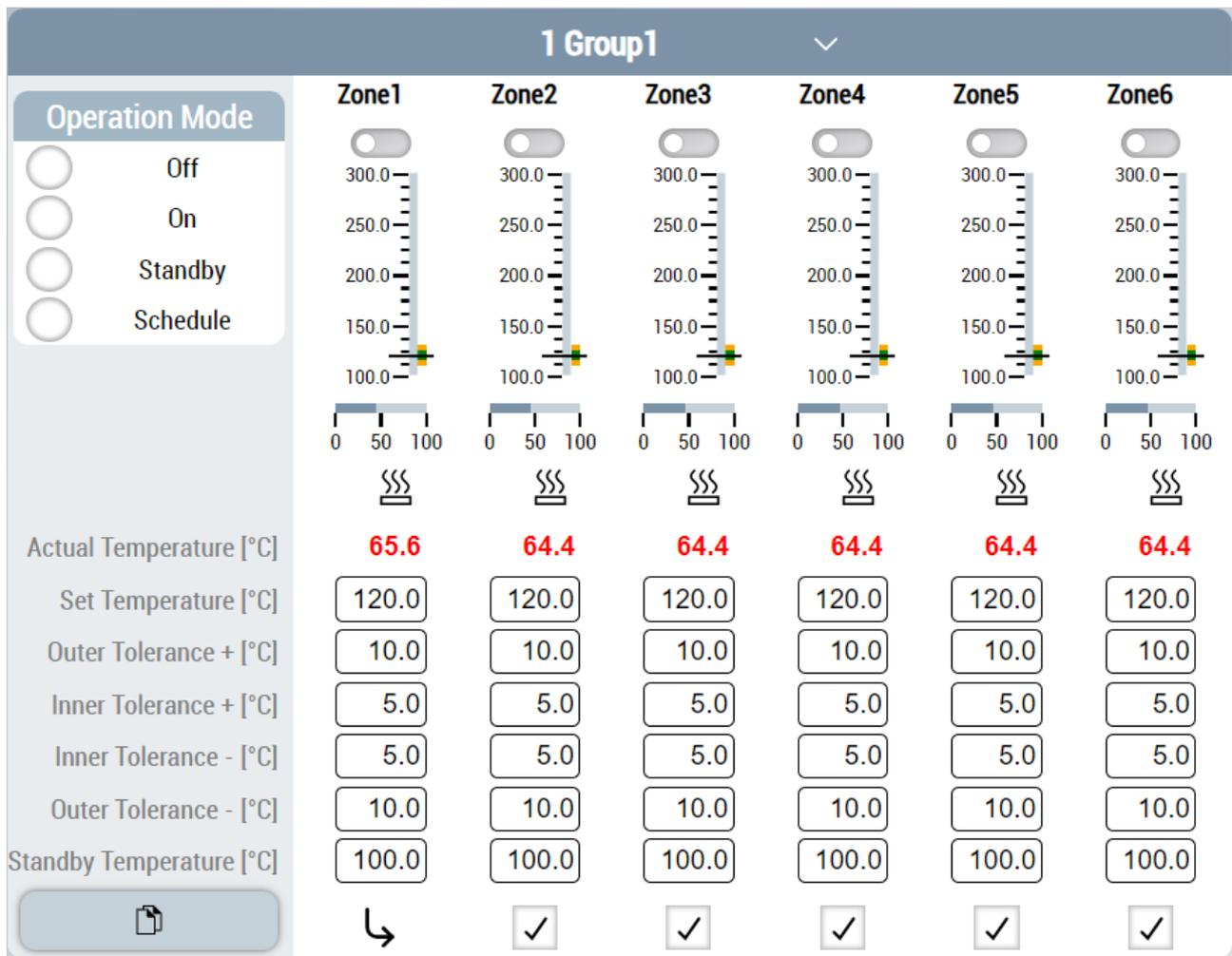
Verfügbar: ab Version 12.14.0

11.3 ZoneConfiguration

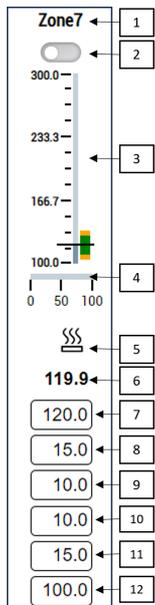
NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Temperature

Verfügbar: ab Version 12.8.0

Das ZoneConfiguration-Control wird verwendet, um verschiedene Betriebsparameter der in der SPS konfigurierten Temperaturgruppen und -zonen zu visualisieren und dem Benutzer die Möglichkeit zu geben, diese zu ändern.



Jede Temperaturzone in der gewählten Temperaturzonengruppe wird mit verschiedenen Eingabe- und Grafik-Controls visualisiert.



1. ZoneName des Kanals.
2. Mit diesem Control kann der Benutzer einen Kanal aktivieren/deaktivieren.
3. Eine Linearanzeige visualisiert die Ist-Temperatur, die Soll-Temperatur sowie die inneren und äußeren Toleranzbereiche auf einer Skala an.
4. Eine Anzeige visualisiert die PowerLevel-Eigenschaft des Kanals.

5. Der Kühl-/Heizstatus der Temperaturkanäle wird mit einem entsprechenden Symbol angezeigt.
6. ActualTemperature des Kanals.
7. SetpointTemperature-Eingang für den Kanal.
8. Äußere Toleranz in positiver Richtung.
9. Innere Toleranz in positiver Richtung.
10. Innere Toleranz in negativer Richtung.
11. Äußere Toleranz in negativer Richtung.
12. Standby-Temperatur der Zone.

11.3.1 Attribute

Kategorie: Colors

Name	Beschreibung
HeaderBackgroundColor [▶ 237]	Definition der Hintergrundfarbe der Kopfzeile.
HeaderFontColor [▶ 237]	Definition der Textfarbe der Kopfzeile.

Kategorie: Header Configuration

Name	Beschreibung
HeaderHeight [▶ 238]	Höhe der Kopfzeile des Controls.
HeaderHeightUnit [▶ 238]	Einheit des Attributs HeaderHeight.
HeaderFontSize [▶ 237]	Schriftgröße der Kopfzeile.
HeaderFontSizeUnit [▶ 238]	Einheit für die Schriftgröße der Kopfzeile.

Kategorie: Config

Name	Beschreibung
ActualZoneGroupId [▶ 236]	Gruppen-Id der auf dem Control angezeigten Temperaturgruppe.
DisplayOperationModes [▶ 237]	Mit diesem Attribut können die Bedienelemente auf dem Control angezeigt werden.
DigitsAfterDecimal [▶ 237]	Einzelne Werte zeigen die Displayauflösung an.

User Changes Logging:

Name	Beschreibung
LogChangesByUser [▶ 238]	Logging der vom Benutzer auf dem Client vorgenommenen Änderungen in einer Logdatei.
ZoneNameAsDescriptor [▶ 239]	Der Zonenname wird als Deskriptortext für den Eintrag im Änderungs-Log anstelle des vollständigen Symbolpfads verwendet.

11.3.1.1 ActualZoneGroupId

Mit diesem Attribut kann der Benutzer die GroupId der konfigurierten Temperaturzonengruppe ändern. Durch Ändern dieses Attributs kann der Benutzer die Temperaturgruppe auswählen, die angezeigt werden soll.

Schema: tchmi:general##/definitions/Number

Attribut-Getter: getActualZoneGroupId

Attribut-Setter: setActualZoneGroupId

Verfügbar: ab Version 12.8.0

11.3.1.2 DigitsAfterDecimal

Mit diesem Attribut kann die Auflösung der einzelnen Zahlenwerte, die auf dem Control angezeigt werden, aktualisiert werden.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getDigitsAfterDecimal

Attribut-Setter: setDigitsAfterDecimal

Verfügbar: ab Version 12.8.0

11.3.1.3 DisplayOperationModes

Standardmäßig werden die Bedienelemente, mit denen der aktive Modus der ausgewählten Temperaturzonengruppe eingestellt werden, nicht auf dem Control angezeigt. Mit diesem Attribut können die Controls für den Benutzer zur Verfügung gestellt werden.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getDisplayOperationModes

Attribut-Setter: setDisplayOperationModes

Verfügbar: ab Version 12.8.0

11.3.1.4 HeaderBackgroundColor

Legt die Hintergrundfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderBackgroundColor

Attribut-Setter: setHeaderBackgroundColor

Verfügbar: ab Version 12.8.0

11.3.1.5 HeaderFontColor

Legt die Textfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderFontColor

Attribut-Setter: setHeaderFontColor

Verfügbar: ab Version 12.8.0

11.3.1.6 HeaderFontSize

Die Schriftgröße der Kopfzeile. Wenn Prozent als Einheit festgelegt ist, ist die Kopfzeilengröße relativ zur Schriftgröße des übergeordneten Elements.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderFontSize

Attribut-Setter: setHeaderFontSize

Verfügbar: ab Version 12.8.0

11.3.1.7 HeaderFontSizeUnit

Legt die Einheit der Kopfzeilenhöhe fest:

- Pixel für eine absolute Größe
- Prozent für eine relative Größe

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderHeightUnit

Attribut-Setter: setHeaderHeightUnit

Verfügbar: ab Version 12.8.0

11.3.1.8 HeaderHeight

Legt die Höhe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderHeight

Attribut-Setter: setHeaderHeight

Verfügbar: ab Version 12.8.0

11.3.1.9 HeaderHeightUnit

Legt die Einheit der Kopfzeilenhöhe fest:

- Pixel für eine absolute Größe
- Prozent für eine relative Größe

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderHeightUnit

Attribut-Setter: setHeaderHeightUnit

Verfügbar: ab Version 12.8.0

11.3.1.10 LogUserChanges

Wenn dieses Attribut auf `TRUE` gesetzt ist, loggt das Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket jede Änderung durch den Benutzer auf dem HMI-Bildschirm in einer Logdatei.

Um dieses Attribut zu verwenden, muss [UserLoggerSymbol](#) [► 45] mit einem Binding an [FB_UserLoggingHmi](#) konfiguriert werden.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getLogUserChanges

Attribut-Setter: setLogUserChanges

Verfügbar: ab Version 12.12.0

11.3.1.11 ShowCopy

Blenden Sie die Kopierschaltfläche und die zugehörigen Controls ein oder aus.

Schema: tchmi:general#/definitions/BOOL

Attribut-Getter: getDigitsAfterDecimal

Attribut-Setter: setDigitsAfterDecimal

Verfügbar: ab Version 12.12.0

11.3.1.12 ZoneNameAsDescriptor

Der Zonenname wird als Deskriptortext für den Eintrag im Änderungs-Log anstelle des vollständigen Symbolpfads verwendet.

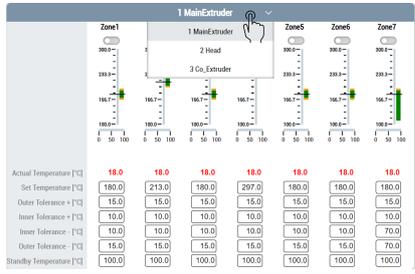
Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getZoneNameAsDescriptor

Attribut-Setter: setZoneNameAsDescriptor

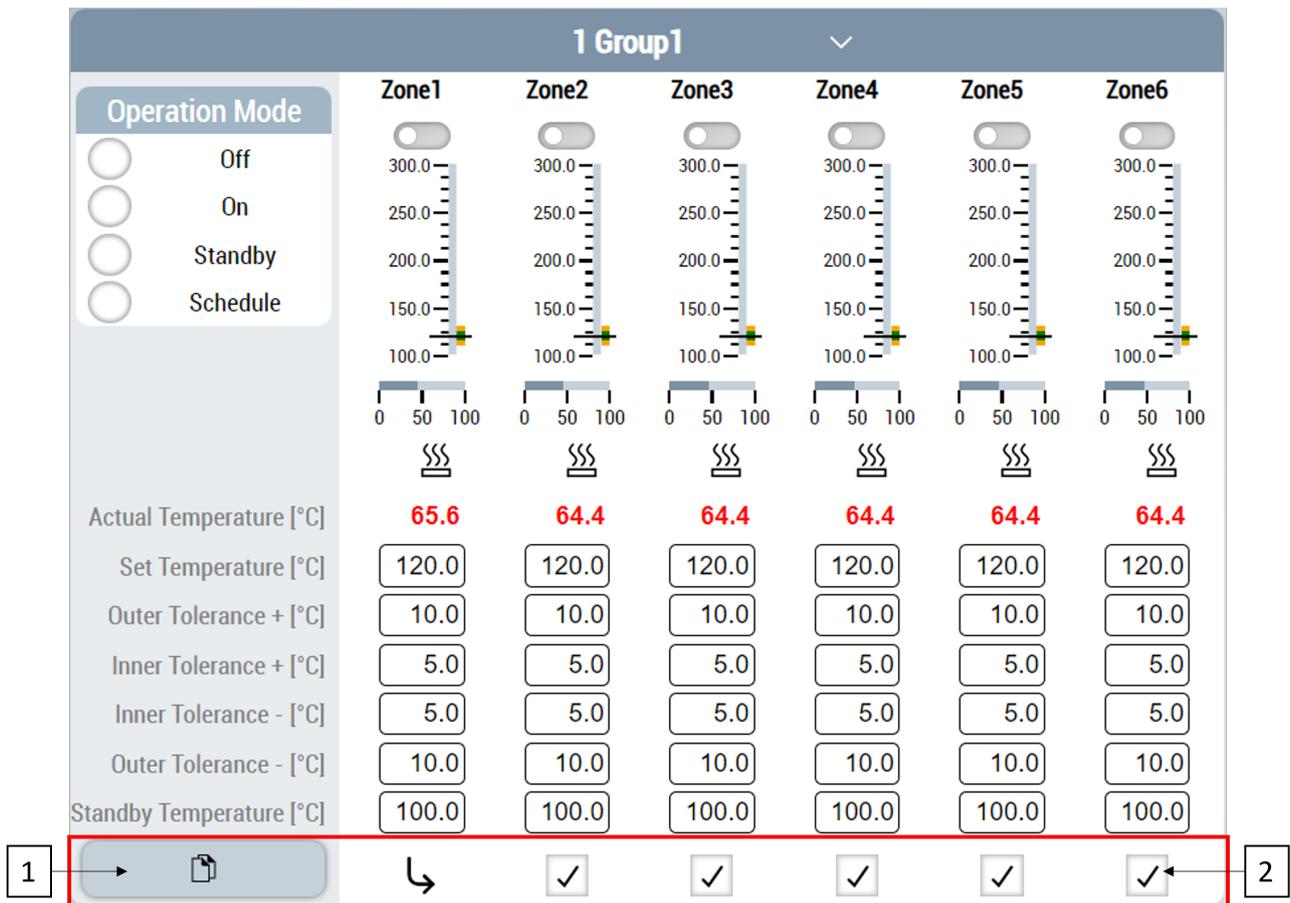
Verfügbar: ab Version 12.14.0

11.3.2 Benutzerinteraktionen

Name	Beschreibung	Ergebnis im Control																																																																
Einfaches Tippen auf die Kopfzeile des Controls.	Wählen Sie die Gruppe, die in dem Control angezeigt werden soll, aus dem Kombinationsfeld aus, das erscheint, wenn ein Benutzer auf den Kopfzeilenbereich des Controls tippt.	 <table border="1"> <thead> <tr> <th></th> <th>Zone1</th> <th>Zone2</th> <th>Zone3</th> <th>Zone4</th> <th>Zone5</th> <th>Zone6</th> <th>Zone7</th> </tr> </thead> <tbody> <tr> <td>Actual Temperature [°C]</td> <td>18.0</td> <td>18.0</td> <td>18.0</td> <td>18.0</td> <td>18.0</td> <td>18.0</td> <td>18.0</td> </tr> <tr> <td>Set Temperature [°C]</td> <td>100.0</td> <td>213.0</td> <td>180.0</td> <td>207.0</td> <td>160.0</td> <td>160.0</td> <td>180.0</td> </tr> <tr> <td>Outer Tolerance [°C]</td> <td>15.0</td> <td>15.0</td> <td>15.0</td> <td>15.0</td> <td>15.0</td> <td>15.0</td> <td>15.0</td> </tr> <tr> <td>Inner Tolerance [°C]</td> <td>10.0</td> <td>10.0</td> <td>10.0</td> <td>10.0</td> <td>10.0</td> <td>10.0</td> <td>10.0</td> </tr> <tr> <td>Inner Tolerance [°C]</td> <td>10.0</td> <td>10.0</td> <td>10.0</td> <td>10.0</td> <td>10.0</td> <td>10.0</td> <td>10.0</td> </tr> <tr> <td>Outer Tolerance [°C]</td> <td>15.0</td> <td>15.0</td> <td>15.0</td> <td>15.0</td> <td>15.0</td> <td>15.0</td> <td>10.0</td> </tr> <tr> <td>Standby Temperature [°C]</td> <td>100.0</td> <td>100.0</td> <td>100.0</td> <td>100.0</td> <td>100.0</td> <td>100.0</td> <td>100.0</td> </tr> </tbody> </table>		Zone1	Zone2	Zone3	Zone4	Zone5	Zone6	Zone7	Actual Temperature [°C]	18.0	18.0	18.0	18.0	18.0	18.0	18.0	Set Temperature [°C]	100.0	213.0	180.0	207.0	160.0	160.0	180.0	Outer Tolerance [°C]	15.0	15.0	15.0	15.0	15.0	15.0	15.0	Inner Tolerance [°C]	10.0	10.0	10.0	10.0	10.0	10.0	10.0	Inner Tolerance [°C]	10.0	10.0	10.0	10.0	10.0	10.0	10.0	Outer Tolerance [°C]	15.0	15.0	15.0	15.0	15.0	15.0	10.0	Standby Temperature [°C]	100.0	100.0	100.0	100.0	100.0	100.0	100.0
	Zone1	Zone2	Zone3	Zone4	Zone5	Zone6	Zone7																																																											
Actual Temperature [°C]	18.0	18.0	18.0	18.0	18.0	18.0	18.0																																																											
Set Temperature [°C]	100.0	213.0	180.0	207.0	160.0	160.0	180.0																																																											
Outer Tolerance [°C]	15.0	15.0	15.0	15.0	15.0	15.0	15.0																																																											
Inner Tolerance [°C]	10.0	10.0	10.0	10.0	10.0	10.0	10.0																																																											
Inner Tolerance [°C]	10.0	10.0	10.0	10.0	10.0	10.0	10.0																																																											
Outer Tolerance [°C]	15.0	15.0	15.0	15.0	15.0	15.0	10.0																																																											
Standby Temperature [°C]	100.0	100.0	100.0	100.0	100.0	100.0	100.0																																																											

11.3.2.1 CopyAction

Wenn das Attribut `ShowCopy` [► 238] auf true gesetzt ist, erscheint eine Kopierschaltfläche im linken unteren Bereich. Für jedes Zonenelement außer der ersten Zone von links erscheint eine Checkbox.



1. Die Kopierschaltfläche, um die Kopieraktion von der Benutzeroberfläche aus zu starten.
2. Die Checkbox zum Aktivieren der Kopie für jede Zone.

Durch Aktivieren oder Deaktivieren dieser Checkbox kann der Benutzer die Kopieraktion für die jeweilige Zone aktivieren oder deaktivieren.

Wenn der Benutzer eine Kopiertaste drückt, kopiert das Control die eingestellte Temperatur, die äußeren und inneren Toleranzen und die Standby-Temperaturwerte der ersten Zone von links auf alle anderen Zonen, die die Kopierfunktion aktiviert haben.

11.3.3 Funktionen

11.3.3.1 CopyFirstChannelToAll

Wenn diese Funktion ausgeführt wird, führt sie den Kopiervorgang durch. Er kopiert den Sollwert, die inneren und äußeren Toleranzen und die Standby-Temperaturwerte der ersten Zone von links in alle anderen Zonen der Gruppe.

Wenn das Attribut `ShowCopy` [► 238] auf `TRUE` gesetzt ist, werden durch diese Aktion nur die Zonen geändert, die zum Kopieren freigegeben sind.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

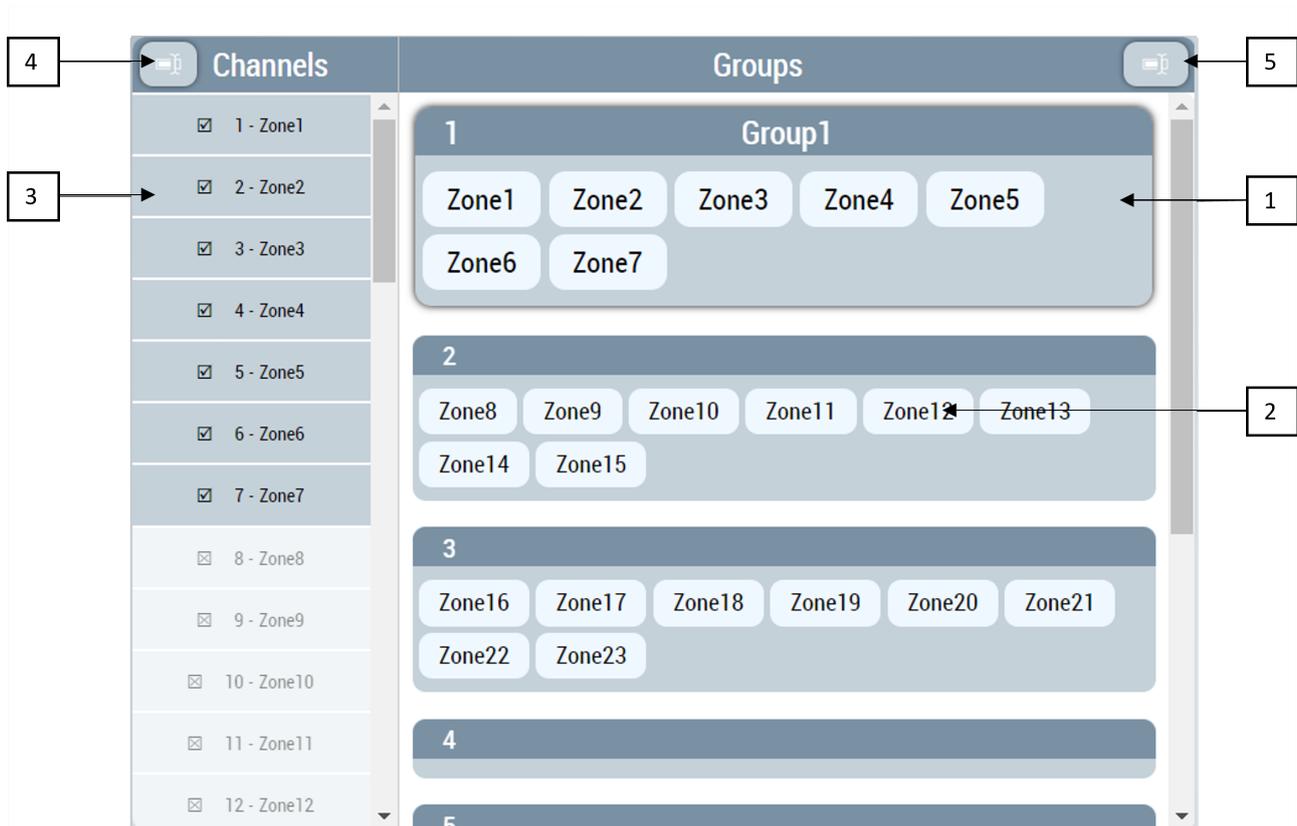
Verfügbar: ab Version 12.12.0

11.4 ZoneGrouping

NuGet-Paket: Beckhoff.TwinCAT.HMI.Plastic.Temperature

Verfügbar: ab Version 12.8.0

Das ZoneGrouping-Control wird verwendet, um die Konfiguration der Temperaturgruppen im Tc3 Plastic Application-Projekt zu visualisieren und dem Benutzer die Möglichkeit zu geben, diese zu ändern.



1. Liste der Temperaturgruppen.
2. Zu jeder Gruppe hinzugefügte Zonen werden innerhalb der Gruppe angezeigt.
3. Eine Liste aller verfügbaren Temperaturkanäle im Projekt.
4. Schaltfläche zum Starten des Dialogs zur Bearbeitung von Zonennamen.
5. Schaltfläche zum Starten des Dialogs zur Bearbeitung von Gruppennamen.

Der Benutzer kann alle Temperaturgruppen mit Informationen über die zu den einzelnen Gruppen gehörenden Zonen einsehen.

Benutzer können Zonen aus jeder Gruppe hinzufügen und entfernen.

Mit den Schaltflächen auf dem Control werden verschiedene Dialoge gestartet, die zum Ändern von Gruppennamen bzw. Zonennamen verwendet werden können.

In diesem Dokument werden die Begriffe "Kanal" und "Zone" zur Beschreibung derselben Objekte verwendet. Ein Kanal stellt einen Temperaturregelungskanal dar. Wenn ein Kanal Teil einer Temperaturgruppe ist, kann er als Zone bezeichnet werden.

Weitere Informationen über Temperaturkanäle und Temperaturzonen finden Sie unter diesem [Link](#).



Aktion aktivieren

Alle neuen Änderungen am Control müssen über die Aktion [ActivateConfig](#) [► 244] aktiviert werden, um neue Werte an die SPS zu senden.

11.4.1 Attribute

Kategorie: Colors

Name	Beschreibung
HeaderBackgroundColor [▶ 242]	Definition der Hintergrundfarbe der Kopfzeile.
HeaderFontColor [▶ 242]	Definition der Textfarbe der Kopfzeile.

Kategorie: Header Configuration

Name	Beschreibung
HeaderHeight [▶ 243]	Höhe der Kopfzeile des Controls.
HeaderHeightUnit [▶ 243]	Einheit des Attributs HeaderHeight.
HeaderFontSize [▶ 243]	Schriftgröße der Kopfzeile.
HeaderFontSizeUnit [▶ 243]	Einheit für die Schriftgröße der Kopfzeile.

User Changes Logging:

Name	Beschreibung
LogChangesByUser [▶ 243]	Logging der vom Benutzer auf dem Client vorgenommenen Änderungen in einer Logdatei.
ZoneNameAsDescriptor [▶ 244]	Der Zonenname wird als Deskriptortext für den Eintrag im Änderungs-Log anstelle des vollständigen Symbolpfads verwendet.

11.4.1.1 Activation Required

Wird auf TRUE gesetzt, wenn der Benutzer neue Änderungen vornimmt, die nicht aktiviert sind.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getActivationRequired

Verfügbar: ab Version 12.8.0

11.4.1.2 HeaderBackgroundColor

Legt die Hintergrundfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderBackgroundColor

Attribut-Setter: setHeaderBackgroundColor

Verfügbar: ab Version 12.8.0

11.4.1.3 HeaderFontColor

Legt die Textfarbe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/Color

Attribut-Getter: getHeaderFontColor

Attribut-Setter: setHeaderFontColor

Verfügbar: ab Version 12.8.0

11.4.1.4 HeaderHeight

Legt die Höhe der Kopfzeile fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderHeight

Attribut-Setter: setHeaderHeight

Verfügbar: ab Version 12.8.0

11.4.1.5 HeaderHeightUnit

Legt die Einheit der Kopfzeilenhöhe fest:

- Pixel für eine absolute Größe
- Prozent für eine relative Größe

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderHeightUnit

Attribut-Setter: setHeaderHeightUnit

Verfügbar: ab Version 12.8.0

11.4.1.6 HeaderFontSize

Die Schriftgröße der Kopfzeile. Wenn Prozent als Einheit festgelegt ist, ist die Kopfzeilengröße relativ zur Schriftgröße des übergeordneten Elements.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getHeaderFontSize

Attribut-Setter: setHeaderFontSize

Verfügbar: ab Version 12.8.0

11.4.1.7 HeaderFontSizeUnit

Die Einheit der Schriftgröße der Kopfzeile (px, %).

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getHeaderFontSizeUnit

Attribut-Setter: setHeaderFontSizeUnit

Verfügbar: ab Version 12.8.0

11.4.1.8 LogUserChanges

Wenn dieses Attribut auf `TRUE` gesetzt ist, loggt das Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket jede Änderung durch den Benutzer auf dem HMI-Bildschirm in einer Logdatei.

Um dieses Attribut zu verwenden, muss [UserLoggerSymbol](#) [► 45] mit einem Binding an [FB_UserLoggingHmi](#) konfiguriert werden.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getLogUserChanges

Attribut-Setter: setLogUserChanges

Verfügbar: ab Version 12.12.0

11.4.1.9 ZoneNameAsDescriptor

Der Zonenname wird als Deskriptortext für den Eintrag im Änderungs-Log anstelle des vollständigen Symbolpfads verwendet.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getZoneNameAsDescriptor

Attribut-Setter: setZoneNameAsDescriptor

Verfügbar: ab Version 12.14.0

11.4.2 Funktionen

11.4.2.1 ActivateConfig

Durch den Aufruf dieser Funktion werden alle Benutzeränderungen aktiviert und in die SPS-Symbole geschrieben.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: ab Version 12.8.0

11.4.2.2 ResetConfigData

Mit dieser Funktion werden alle Änderungen des Benutzers rückgängig gemacht, indem die zuletzt aktivierte Konfiguration erneut geladen wird.

Parameter

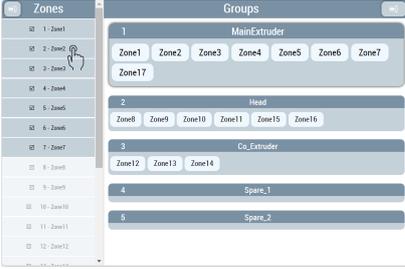
Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar ab Version 12.8.0

11.4.3 Benutzerinteraktionen

Name	Beschreibung	Ergebnisse im Control
<p>Eine Gruppe auswählen.</p>	<p>Wenn Sie auf eine beliebige Gruppenkachel tippen oder klicken, wird diese Gruppe ausgewählt.</p>	 <p>The screenshot shows the 'Groups' panel with five groups. Group 1, 'MainExtruder', is highlighted in blue. A mouse cursor is pointing at the 'Zone4' button within this group. The 'Zones' panel on the left shows a list of zones from 1 to 12.</p>
<p>Hinzufügen oder Entfernen von Zonen aus einer Gruppe.</p>	<p>Sobald eine Gruppe ausgewählt wurde, zeigt das Temperaturkanal-Panel die möglichen Optionen an.</p> <p>Kanäle, die Teil der Gruppe sind, werden durch Klicken/Tippen auf den Kanal entfernt.</p> <p>Zonen, die keiner Gruppe angehören, können durch Anklicken/Tippen zu einer ausgewählten Gruppe hinzugefügt werden.</p> <p>Jeder Kanal, der zu einer anderen Gruppe gehört, kann nicht zur ausgewählten Gruppe hinzugefügt werden und wird für jede Klickaktion deaktiviert.</p>	 <p>This screenshot is identical to the one above, but with a mouse cursor pointing at the 'Zone17' button in Group 1. This illustrates the state where a zone that is not part of the selected group is shown as a disabled option.</p>

11.4.4 Dialoge

11.4.4.1 Zonennamen bearbeiten

Edit Zone Names ✕

Index	Zone Name
1	Zone1
2	Zone2
3	Zone3
4	Zone4
5	Zone5
6	Zone6
7	Zone7
8	Zone8
9	Zone9
10	Zone10
11	Zone11
12	Zone12
13	Zone13

In diesem Dialogfeld kann der Benutzer den Wert des Zonennamens für jeden Kanal in der SPS bearbeiten. Nach der Eingabe der neuen Werte drücken Sie **Ok**, um die neuen Änderungen zu übernehmen, und **Abbrechen**, um das Dialogfeld zu schließen.

11.4.4.2 Gruppennamen bearbeiten

Edit Group Names ✕

Index	Group Name
1	Group1
2	Group 2
3	Hydraulic
4	
5	
6	
7	
8	
9	
10	

Ok Cancel

In diesem Dialogfeld werden die Gruppennamenwerte für jede Gruppe angezeigt. Der Benutzer kann die Namen ändern und **OK** drücken, um die neuen Änderungen zu übernehmen, oder **Abbrechen**, um das Dialogfeld zu schließen.

11.5 ZonedImageLayout

Mit dem Framework-Control ZonedImageLayout können Benutzer Temperaturzonen über einem Bildhintergrund in einem Layout darstellen, das die physische Anordnung auf der Maschine nachbildet. Alle Änderungen können direkt vom Kunden vorgenommen werden, so dass die Notwendigkeit entfällt, das Projekt über die Technik zu betreuen.

Basierend auf dem Standard-TcHMI-Control verwendet das ZonedImageLayout-Control die Standard-Hintergrundbild-Attribute wie `BackgroundImage`, um ein Bild im Hintergrund anzuzeigen. Verschiedene Attribute und Dialoge ermöglichen es dem Benutzer, das Hintergrundbild und damit verbundene Eigenschaften direkt vom Client aus zu ändern.

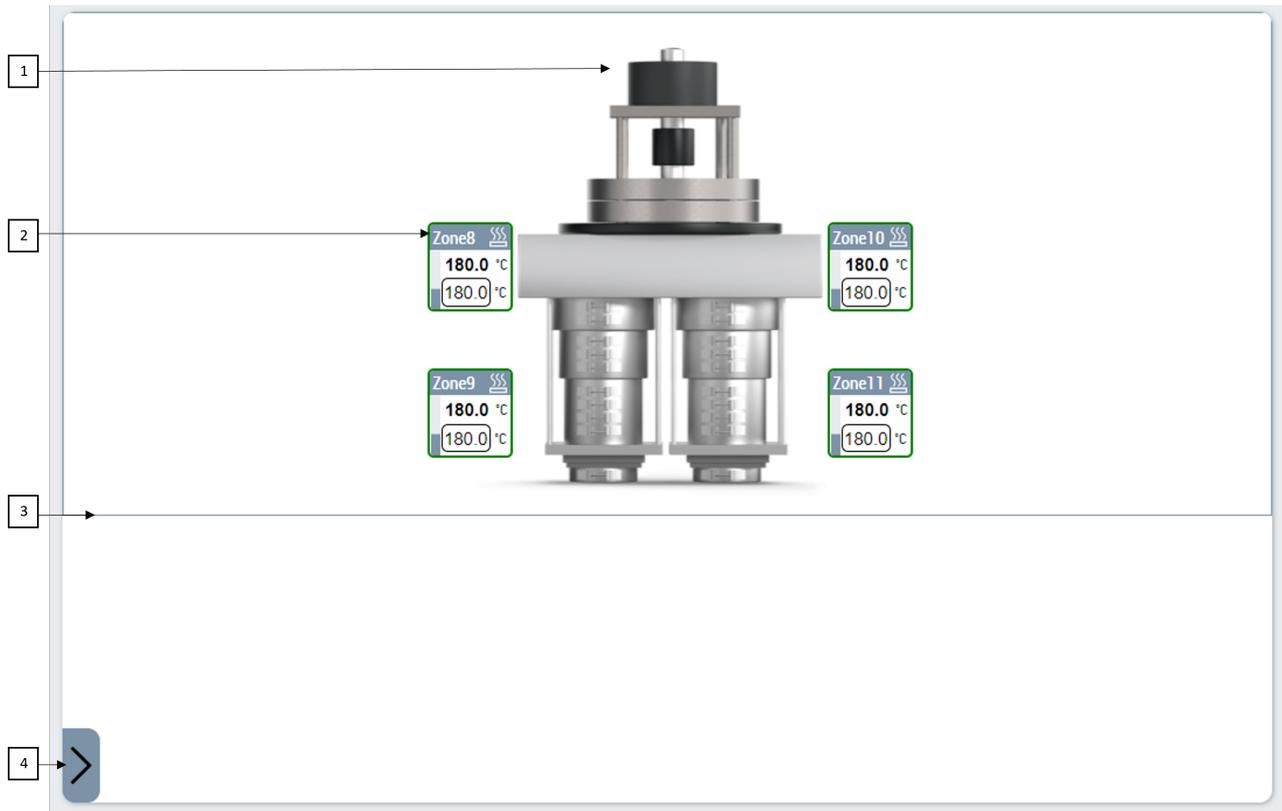
Auf dem Hintergrund des Controls werden kleine Kacheln dargestellt, wobei jede Kachel den aktuellen Status einer Temperaturzone anzeigt.

Der Benutzer kann verschiedene Attribute, Dialoge und Touch-/Klick-Interaktionen verwenden, um Temperaturzonen-Kacheln zu verschieben und in verschiedenen Layouts anzuordnen.



ActivateConfig action

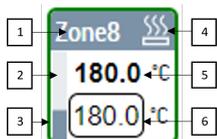
Alle neuen Änderungen am Control müssen über die Aktion [ActivateConfig](#) [► 253] aktiviert werden, um neue Werte an die SPS zu senden.



1. Hintergrundbild, das auf dem Control angezeigt wird.
2. Temperaturzonen
3. Layoutbereich zur Benutzerführung.
4. Schaltfläche zum Öffnen des Dialogs zur Zonenauswahl.

Kacheln zur Anzeige von Zonen

Jede Zone im Layout wird durch ein visuelles Element dargestellt, das weitere Informationen über die verknüpfte Temperaturzone in der Steuerung liefert.



1. ZoneName
2. PowerLevel
3. Rahmenfarbe gibt den Status der Zone an
4. Zone Heizung/Kühlung/Fehlerstatus
5. ActualTemperature der Zone
6. Eingang für den Benutzer zum Schreiben der Sollwerttemperatur der Zone

In der folgenden Tabelle werden alle möglichen Rahmenfarben und ihre Bedeutung erläutert.

Rahmenfarbe	Zone Status
Blau	Die Ist-Temperatur in der Zone ist sehr niedrig.

Rahmenfarbe	Zone Status
Grün	Die Ist-Temperatur der Zonen liegt innerhalb des inneren Toleranzbereichs.
Orange	Die Ist-Temperatur der Zonen liegt außerhalb der inneren Toleranzen, aber innerhalb der äußeren Toleranzbereiche.
Rot	Die Ist-Temperatur der Zonen liegt außerhalb des äußeren Toleranzbereichs.

11.5.1 Attribute

Konfiguration:

Name	Beschreibung
ZoneImageLayoutConfig [▶ 251]	Mit diesem Attribut kann der Benutzer die Standardkonfiguration festlegen, die auf dem Control angezeigt werden soll.
Mode [▶ 251]	Wechseln Sie zwischen Ansichts- und Konfigurationsmodus.

Common:

Name	Beschreibung
Activation Required [▶ 249]	Wird auf TRUE gesetzt, wenn der Benutzer neue Änderungen vornimmt, die nicht aktiviert sind.
ShiftMode [▶ 251]	Die Einstellung true ist notwendig, um das Layout der Temperaturzonen-Kacheln neu zu organisieren.
LayoutWidth [▶ 250]	Breite des rechteckigen Rahmens, der als visuelle Referenz dient.
LayoutWidthUnit [▶ 250]	Einheit der Layoutbreite.
LayoutHeight [▶ 250]	Höhe des rechteckigen Rahmens, der als visuelle Referenz dient.
LayoutHeightUnit [▶ 250]	Einheit der Layout-Höhe.
DigitsAfterDecimal [▶ 250]	Die Auflösung der einzelnen Zahlenwerte, die auf dem Control angezeigt werden.

User Changes Logging:

Name	Beschreibung
LogChangesByUser [▶ 251]	Logging der vom Benutzer auf dem Client vorgenommenen Änderungen in einer Logdatei.
ZoneNameAsDescriptor [▶ 253]	Der Zonenname wird als Deskriptortext für den Eintrag im Änderungs-Log anstelle des vollständigen Symbolpfads verwendet.

11.5.1.1 Activation Required

Wird auf TRUE gesetzt, wenn der Benutzer neue Änderungen vornimmt, die nicht aktiviert sind.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getActivationRequired

Verfügbar: ab Version 12.8.0

11.5.1.2 DigitsAfterDecimal

Mit diesem Attribut kann die Auflösung der einzelnen Zahlenwerte, die auf dem Control angezeigt werden, aktualisiert werden.

Schema: tchmi:general#/definitions/Number

Attribut-Getter: getDigitsAfterDecimal

Attribut-Setter: setDigitsAfterDecimal

Verfügbar: ab Version 12.14.0

11.5.1.3 LayoutWidth

Es kann zu Szenarien kommen, in denen eine einzige Layoutkonfiguration auf verschiedenen Controls mit unterschiedlichen Anzeigegrößen erscheint. Um den Benutzern zu helfen, ihre Layoutentwürfe auf geeignete Größen zu beschränken, ist auf dem Control ein bestimmter Layoutbereich eingezeichnet.

Dieser rechteckige Frame dient als visuelle Referenz und schränkt die Bewegung der Elemente außerhalb seiner Grenzen nicht ein.

Das Attribut LayoutWidth legt die Breite des auf dem Control gezeichneten Layoutrechtecks fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getLayoutWidth

Attribut-Setter: setLayoutWidth

Verfügbar: ab Version 12.8.0

11.5.1.4 LayoutWidthUnit

Einheit des Attributs LayoutWidth.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getLayoutWidthUnit

Attribut-Setter: setLayoutWidthUnit

Verfügbar: ab Version 12.8.0

11.5.1.5 LayoutHeight

Legt die Höhe des auf dem Control umrandeten Layoutrechtecks fest.

Schema: tchmi:framework#/definitions/MeasurementValue

Attribut-Getter: getLayoutHeight

Attribut-Setter: setLayoutHeight

Verfügbar: ab Version 12.8.0

11.5.1.6 LayoutHeightUnit

Einheit des Attributs LayoutHeight.

Schema: tchmi:framework#/definitions/MeasurementUnit

Attribut-Getter: getLayoutHeightUnit

Attribut-Setter: setLayoutHeightUnit

Verfügbar: ab Version 12.8.0

11.5.1.7 LogUserChanges

Wenn dieses Attribut auf `TRUE` gesetzt ist, loggt das Beckhoff.TwinCAT.HMI.Plastic.Controls-Paket jede Änderung durch den Benutzer auf dem HMI-Bildschirm in einer Logdatei.

Um dieses Attribut zu verwenden, muss [UserLoggerSymbol \[► 45\]](#) mit einem Binding an [FB_UserLoggingHmi](#) konfiguriert werden.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getLogUserChanges

Attribut-Setter: setLogUserChanges

Verfügbar: ab Version 12.12.0

11.5.1.8 Mode

- Im Modus View wird nur das konfigurierte Layout angezeigt. Es ist nicht möglich, Temperaturzonen hinzuzufügen oder zu entfernen, Zonen im Shift-Modus zu verschieben oder die Eigenschaften des Hintergrundbildes über Dialoge im View-Modus zu ändern.
- Im Config-Modus können Benutzer mit den Rechten `operate access` die Konfiguration über verschiedene Dialoge ändern.

Schema: tchmi:framework#/definitions/ZoneImageLayout.Mode

Attribut-Getter: getLayoutHeightUnit

Attribut-Setter: setLayoutHeightUnit

Verfügbar: ab Version 12.8.0

11.5.1.9 ShiftMode

Der ShiftMode ist ein binäres Attribut. Die Einstellung `true` ist notwendig, um das Layout der Temperaturzonen-Kacheln neu zu organisieren.

Sobald dieser Modus eingestellt ist, werden in den Kacheln nur noch der Zonenindex und der Name angezeigt. Die Benutzer können dann interaktiv einzelne Kacheln ziehen und nach Bedarf positionieren.

Es ist nicht möglich, neue Zonen zur Konfiguration hinzuzufügen oder zu entfernen, wenn ShiftMode auf `true` gesetzt ist.

Schema: tchmi:general#/definitions/Boolean

Attribut-Getter: getShiftMode

Attribut-Setter: setShiftMode

Verfügbar: ab Version 12.8.0

11.5.1.10 ZoneImageLayoutConfig

Mit dem Attribut `ZoneImageLayoutConfig` können Benutzer die Standardkonfiguration festlegen, die auf dem Control angezeigt werden soll.

Dieses Attribut ermöglicht auch die Bindung an ein Serversymbol, so dass Benutzer neue Änderungen von der Client-Seite aus erfassen und speichern können.

Editing object of type tchmi:framework#/definitions/ZonImageLayoutConfig

Properties

General

LayoutWidth px

LayoutHeight px

BackgroundPath 

BackgroundWidth px

BackgroundHeight px

Zones (no items)

BackgroundPadding
 ← 0 px → 0 px
 ↑ 0 px ↓ 0 px

BackgroundHorizontalAlignment

BackgroundVerticalAlignment

Show schema

Die Eigenschaften des Objekts werden in der nachstehenden Tabelle erläutert.

Name	Typ	Beschreibung
LayoutWidth	tchmi:framework#/definitions/MeasurementValue	Breite des Layout-Frames, der auf dem Control als Führungsbereich für den Benutzer gezeichnet wird.
LayoutWidthUnit	tchmi:framework#/definitions/MeasurementUnit	Einheit von LayoutWidth.
LayoutHeight	tchmi:framework#/definitions/MeasurementValue	Höhe des Layout-Frames.
LayoutHeightUnit	tchmi:framework#/definitions/MeasurementUnit	Einheit des Attributs LayoutHeight.
BackgroundPath	tchmi:framework#/definitions/Path	Pfad zu einem Bild, das im Hintergrund des Controls angezeigt werden soll.
BackgroundWidth	tchmi:framework#/definitions/MeasurementValue	Breite des Hintergrundbildes.
BackgroundWidthUnit	tchmi:framework#/definitions/MeasurementUnit	Einheit der Breite des Hintergrundbildes
BackgroundHeight	tchmi:framework#/definitions/MeasurementValue	Höhe des Hintergrundbildes.
BackgroundHeightUnit	tchmi:framework#/definitions/MeasurementUnit	Einheit der Höhe des Hintergrundbildes
Zonen	tchmi:framework#/definitions/ZonImageConfig	Liste der Temperaturzonen, die auf dem Control angezeigt werden. Jedes Zonenobjekt enthält 3 Eigenschaften:

Name	Typ	Beschreibung
		X: Position der Zonen-Kachel im Layout in X-Richtung von links oben. Y: Position der Zonen-Kachel im Layout in Y-Richtung von links oben. ZoneIndex: Index des Zonenobjekts im Array aChannels in der SPS.
BackgroundPadding	tchmi:framework#/definitions/BackgroundPadding	Innenabstand des Hintergrundbildes.
BackgroundHorizontalAlignment	tchmi:framework#/definitions/BackgroundHorizontalAlignment	Horizontale Ausrichtung des Hintergrundbildes.
BackgroundVerticalAlignment	tchmi:framework#/definitions/BackgroundVerticalAlignment	Vertikale Ausrichtung des Hintergrundbildes.

Schema: tchmi:framework#/definitions/ZoneImageLayoutConfig

Attribut-Getter: getZoneImageLayoutConfig

Attribut-Setter: setZoneImageLayoutConfig

Verfügbar: ab Version 12.8.0

11.5.1.11 ZoneNameAsDescriptor

Der Zonenname wird als Deskriptortext für den Eintrag im Änderungs-Log anstelle des vollständigen Symbolpfads verwendet.

Schema: "tchmi:general#/definitions/BOOL"

Attribut-Getter: getZoneNameAsDescriptor

Attribut-Setter: setZoneNameAsDescriptor

Verfügbar: ab Version 12.14.0

11.5.2 Funktionen

11.5.2.1 ActivateConfig

Durch den Aufruf dieser Funktion werden alle Benutzeränderungen aktiviert und in das Attribut ZoneImageLayoutConfig geschrieben.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar: ab Version 12.8.0

11.5.2.2 ResetConfigData

Mit dieser Funktion werden alle Änderungen des Benutzers rückgängig gemacht, indem die zuletzt aktivierte Konfiguration erneut geladen wird.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Verfügbar ab Version 12.8.0

11.5.2.3 ChangeBackgroundImage

Bei operativem Zugriff startet die Ausführung dieser Funktion den Dialog [Hintergrundbild ändern](#) [► 260]. Und erlaubt dem Benutzer, das Hintergrundbild zu ändern.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Das Attribut Mode **muss auf Config gesetzt werden, um diese Funktion zu nutzen. Für weitere Informationen besuchen Sie bitte** [Mode](#) [► 251]

Verfügbar: ab Version 12.8.0

11.5.2.4 EditBackgroundImage

Bei operativem Zugriff startet die Ausführung dieser Funktion den Dialog [Hintergrundbild bearbeiten](#) [► 260]. Und erlaubt dem Benutzer, die Eigenschaften des Hintergrundbilds zu ändern.

Parameter

Name	Typ	Beschreibung
-	-	-

Rückgabewert

Typ	Beschreibung
void	Kein Rückgabewert.

Das Attribut Mode **muss auf Config gesetzt werden, um diese Funktion zu nutzen. Für weitere Informationen besuchen Sie bitte** [Mode](#) [► 251]

Verfügbar: ab Version 12.8.0

11.5.3 Benutzerinteraktionen

11.5.3.1 Zonenliste öffnen

Um neue Zonen zum Layout hinzuzufügen oder eine bereits im Layout vorhandene Zone zu entfernen, muss der Benutzer auf die Liste der verfügbaren Temperaturzonen zugreifen.

Klicken Sie dazu auf die Schaltfläche, die die Zonenliste öffnet.

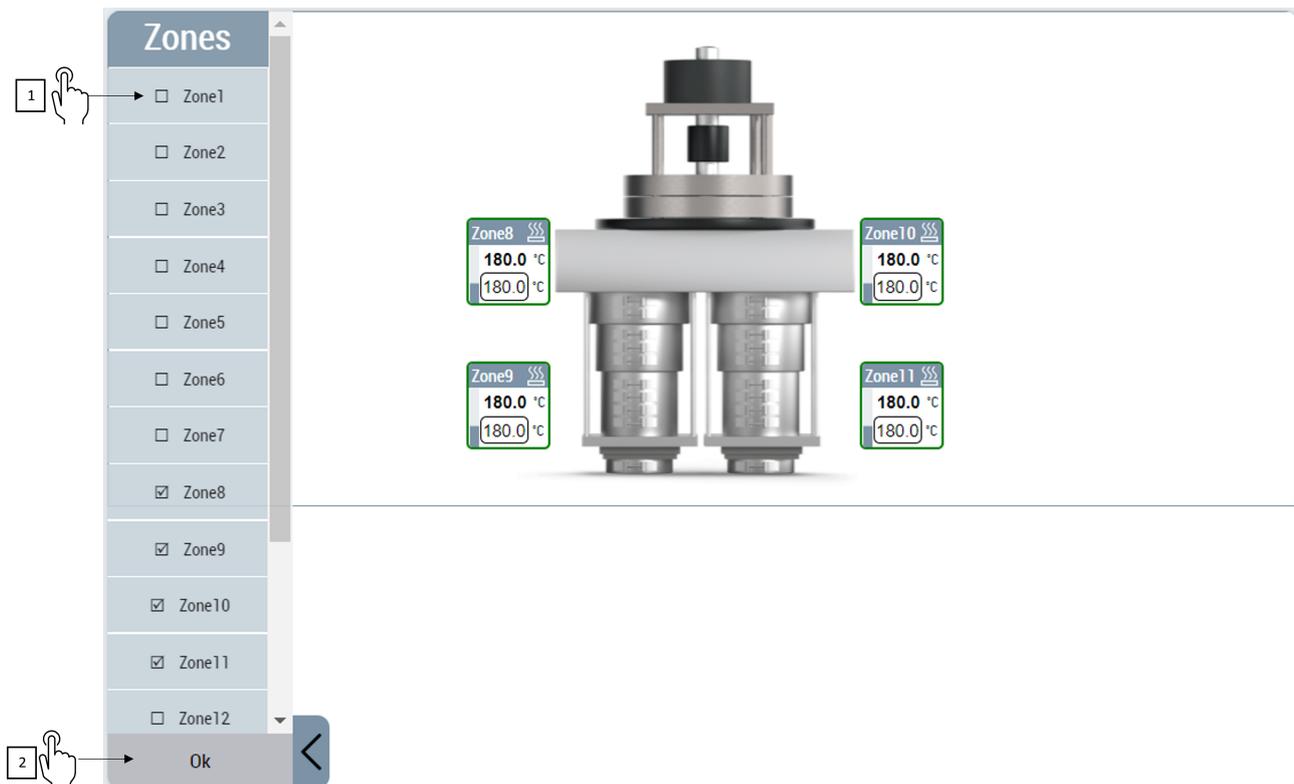


In der Zonenliste werden alle im Projekt verfügbaren Zonen aufgeführt. Jeder Temperaturkanal, der einer Temperaturgruppe zugeordnet ist, wird aufgelistet.

Das Attribut Mode **muss auf Config gesetzt werden, um diese Funktion zu nutzen. Für weitere Informationen besuchen Sie bitte [Mode \[► 251\]](#)**

11.5.3.2 Neue Temperaturzonen dem Layout hinzufügen

Um neue Temperaturzonen hinzuzufügen, die nicht bereits Teil des Layouts sind, öffnen Sie zunächst die Zonenliste.

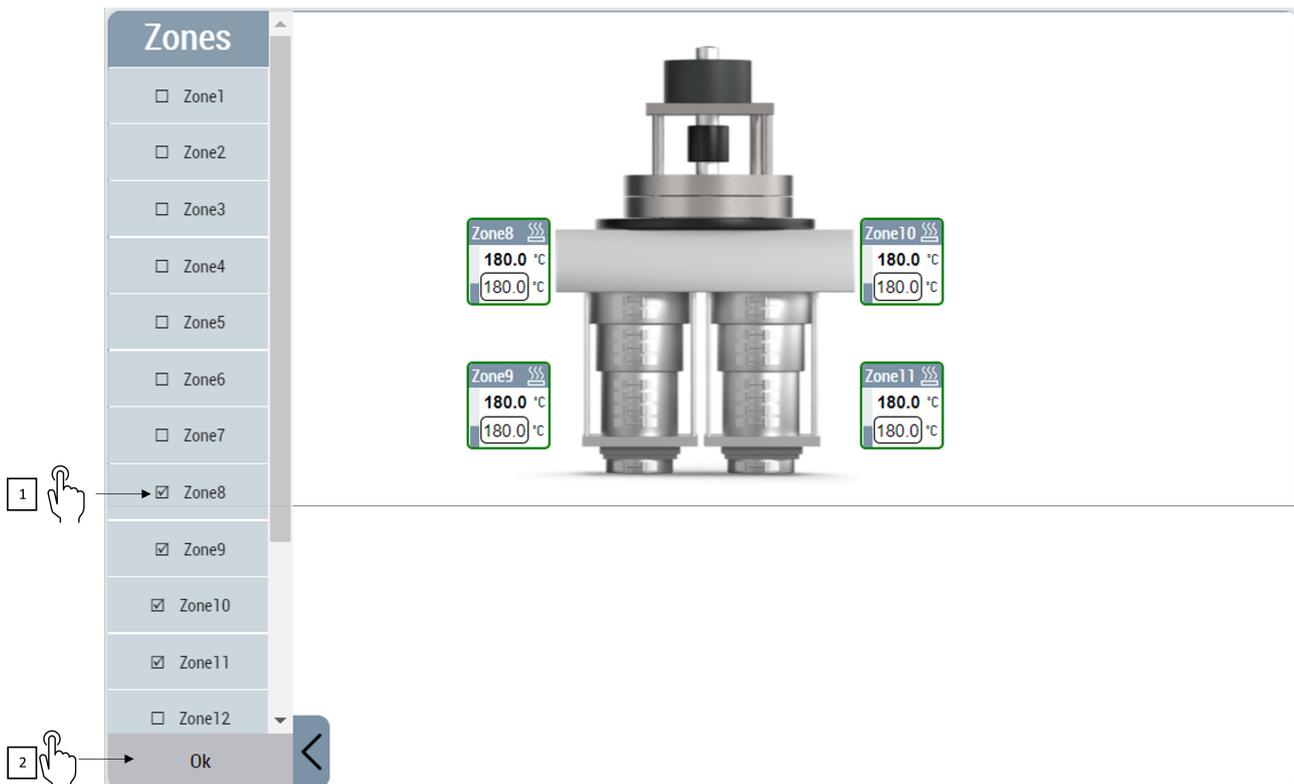


1. Klicken Sie auf die Temperaturzone, die hinzugefügt werden soll. Sie können mehrere Zonen auswählen und sie alle zusammen hinzufügen.
2. Klicken Sie auf die Schaltfläche **Ok** am unteren Ende der Zonenliste, um Ihre Änderungen zu übernehmen.
 - ⇒ Die neuen Zonen werden dem Layout hinzugefügt.

Das Attribut Mode **muss auf Config gesetzt werden, um diese Funktion zu nutzen. Für weitere Informationen besuchen Sie bitte [Mode \[► 251\]](#)**

11.5.3.3 Temperaturzonen aus dem Layout entfernen

Um eine Temperaturzone aus dem Layout zu entfernen:



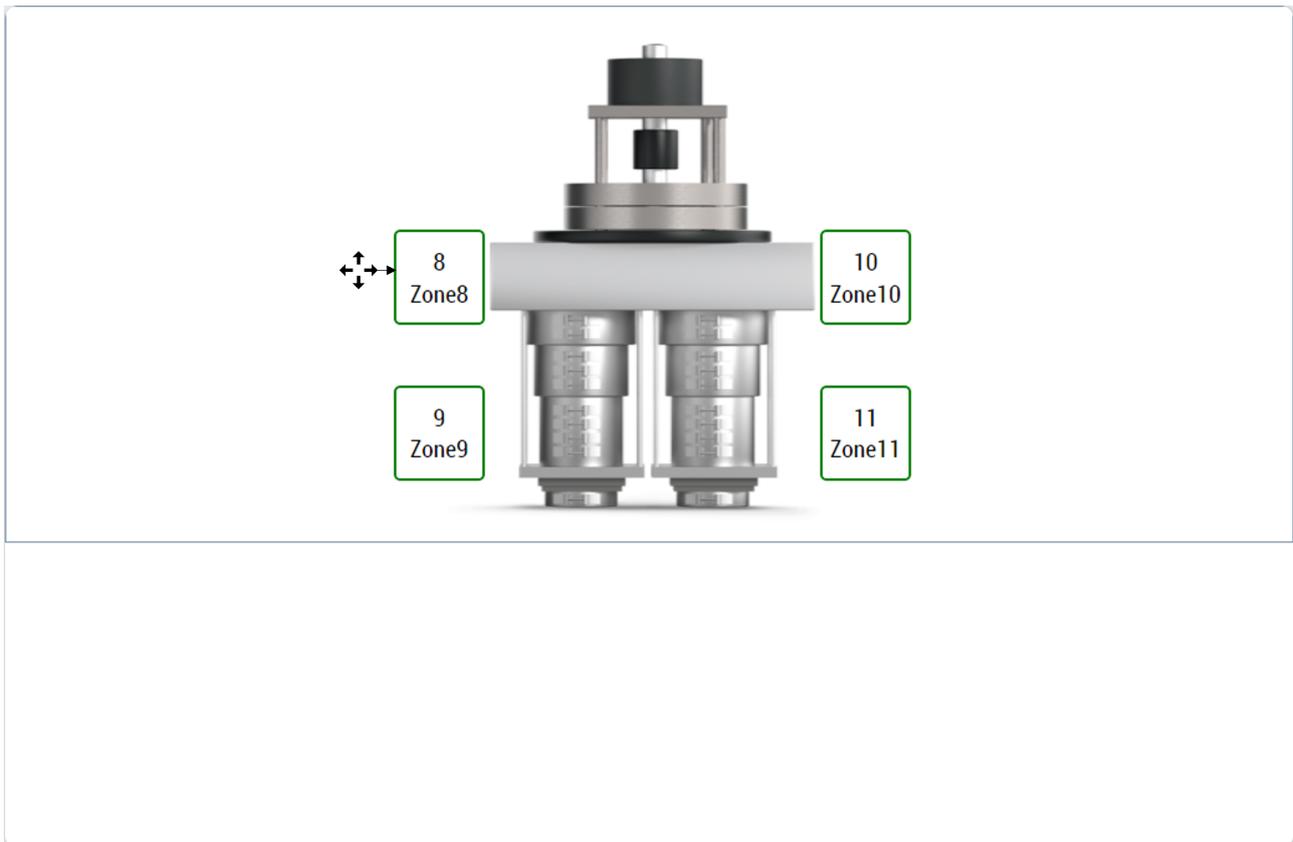
1. Nach dem Öffnen der Zonenliste klicken Sie auf die Temperaturzone, die entfernt werden soll. Sie können mehrere Zonen auswählen, die gleichzeitig hinzugefügt und entfernt werden sollen.
2. Klicken Sie auf die Schaltfläche **Ok** am unteren Ende der Zonenliste, um Ihre Änderungen zu übernehmen.
 - ⇒ Die ausgewählten Zonen werden aus dem Layout entfernt.

Das Attribut Mode **muss auf Config gesetzt werden, um diese Funktion zu nutzen. Für weitere Informationen besuchen Sie bitte** [Mode \[► 251\]](#)

11.5.3.4 Verschieben von Temperaturzonen im Shift-Modus

Um Temperaturzonen-Kacheln im Layoutbereich des Controls zu verschieben, muss das Attribut [ShiftMode \[► 251\]](#) auf TRUE gesetzt werden.

Sobald sich der Benutzer im Shift-Modus befindet, kann er einzelne Temperaturzonen durch Berühren und Ziehen in eine beliebige Richtung verschieben.



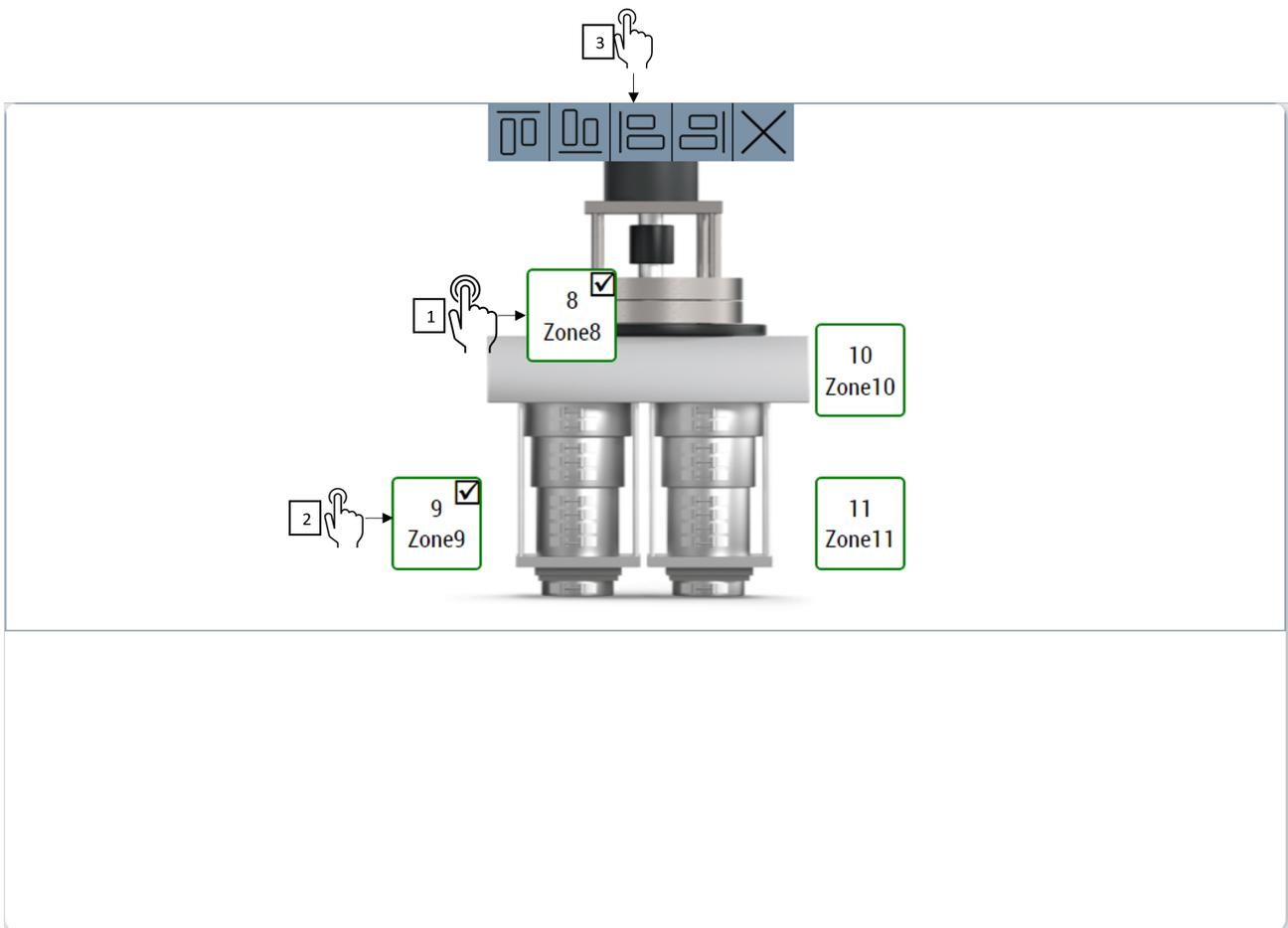
Das Attribut Mode **muss auf Config gesetzt werden, um diese Funktion zu nutzen. Für weitere Informationen besuchen Sie bitte** [Mode \[► 251\]](#)

11.5.3.4.1 Ausrichten mehrerer Zonen

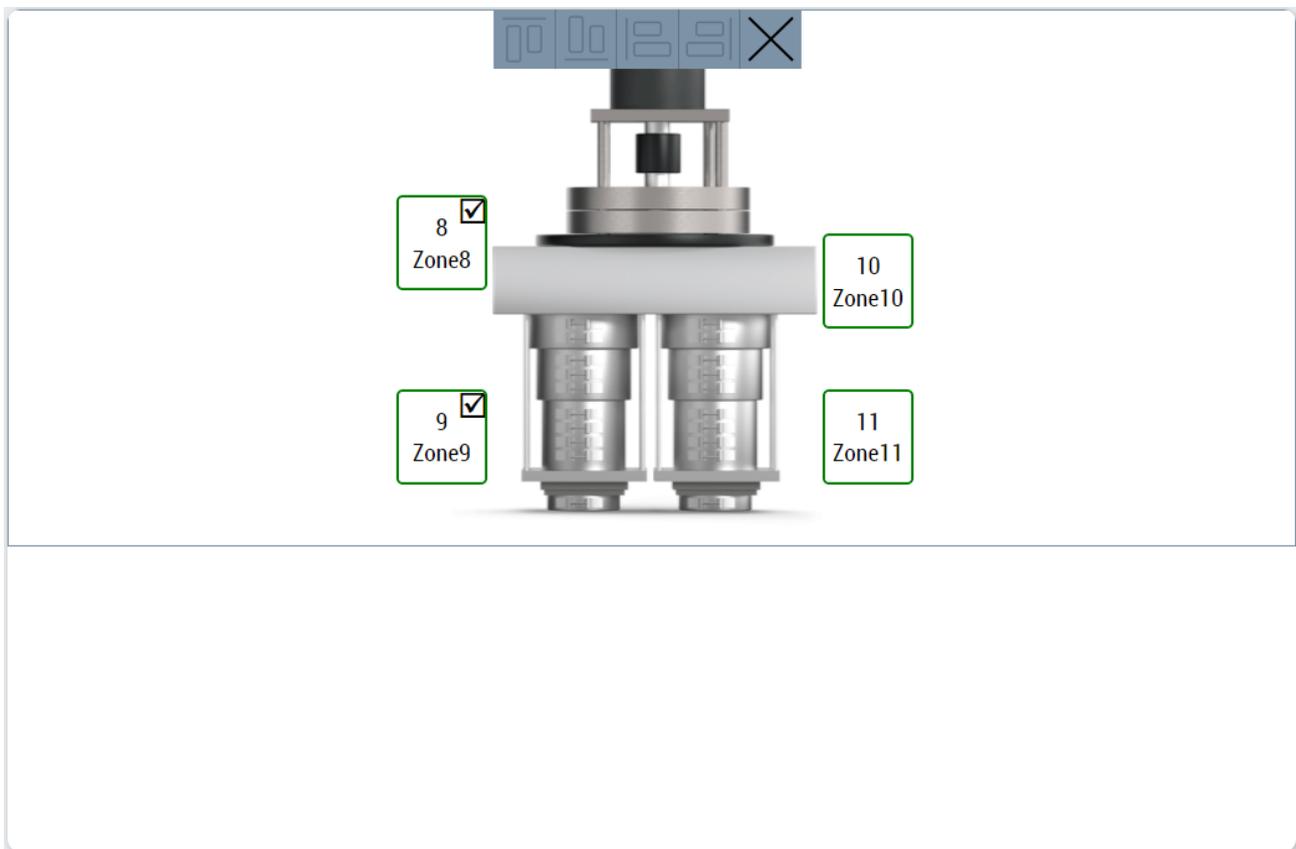
Zum Verschieben und Organisieren mehrerer Temperaturzonen-Kacheln auf dem Bildschirm.

1. Im Shift-Modus tippen Sie zweimal auf die erste Zone, um sie auszuwählen.
2. Dadurch wird der Auswahl- und Ausrichtungsmodus gestartet; am oberen Rand des Controls erscheint eine Ausrichtungsleiste mit den verfügbaren Ausrichtungsoptionen für die Auswahl.
3. Eine zweite Zone kann durch Antippen zur Auswahl hinzugefügt werden.
4. Ein Bereich, der zur Auswahl hinzugefügt wurde, kann durch erneutes Antippen wieder entfernt werden.
5. Sobald zwei oder mehr Zonen ausgewählt sind, kann der Benutzer auf eine der folgenden Ausrichtungsoptionen klicken.
 - Links ausrichten: Alle ausgewählten Zonen werden an der linken Position der Zone ausgerichtet, die am weitesten links liegt.
 - Rechts ausrichten: Alle ausgewählten Zonen werden an der rechten Position der Zone ausgerichtet, die sich am weitesten rechts befindet.
 - Oben ausrichten: Alle ausgewählten Zonen werden an der oberen Position der Zone ausgerichtet, die am weitesten oben liegt.
 - Unten ausrichten: Alle ausgewählten Zonen werden an der unteren Position der Zone ausgerichtet, die sich am weitesten unten befindet.

Zum Beispiel wurden die Temperaturzonen Zone8 und Zone9 ausgewählt und in der Abbildung unten links ausgerichtet.



Das Ergebnis des Ausrichtens sieht wie in der folgenden Abbildung aus.



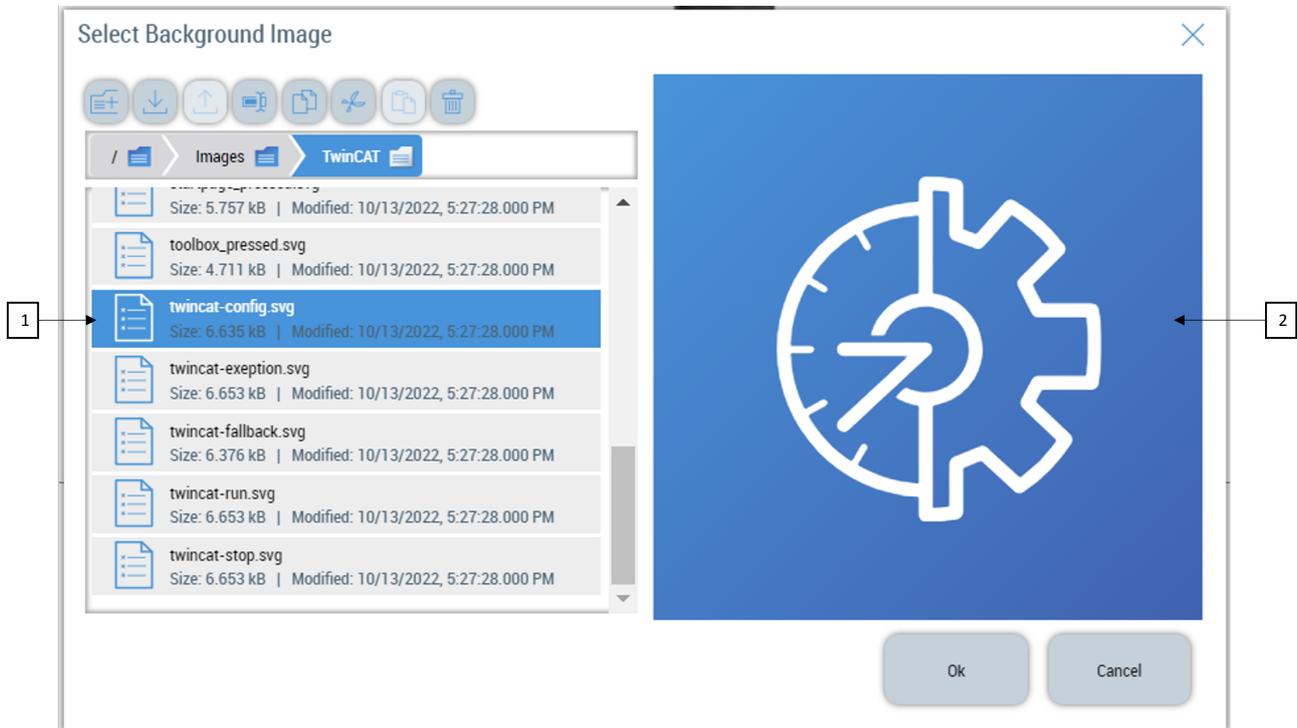
Um den Mehrfachauswahl- und Ausrichtungsmodus zu beenden, muss der Benutzer auf die Schaltfläche "Schließen" in der Ausrichtungsleiste oben auf dem Control klicken.

Das Attribut Mode **muss auf Config gesetzt werden, um diese Funktion zu nutzen. Für weitere Informationen besuchen Sie bitte [Mode.\[> 251\]](#)**

11.5.4 Dialoge

11.5.4.1 Hintergrundbild ändern

Der Benutzer kann das Hintergrundbild des ZonedImageLayout-Controls mithilfe dieses Dialogs ändern.



Das Pop-up-Fenster enthält:

1. Mit dem File Explorer Control kann der Benutzer zum richtigen Ziel der Bilddatei navigieren.
2. Image viewer zeigt das Vorschaubild der ausgewählten Bilddatei an.

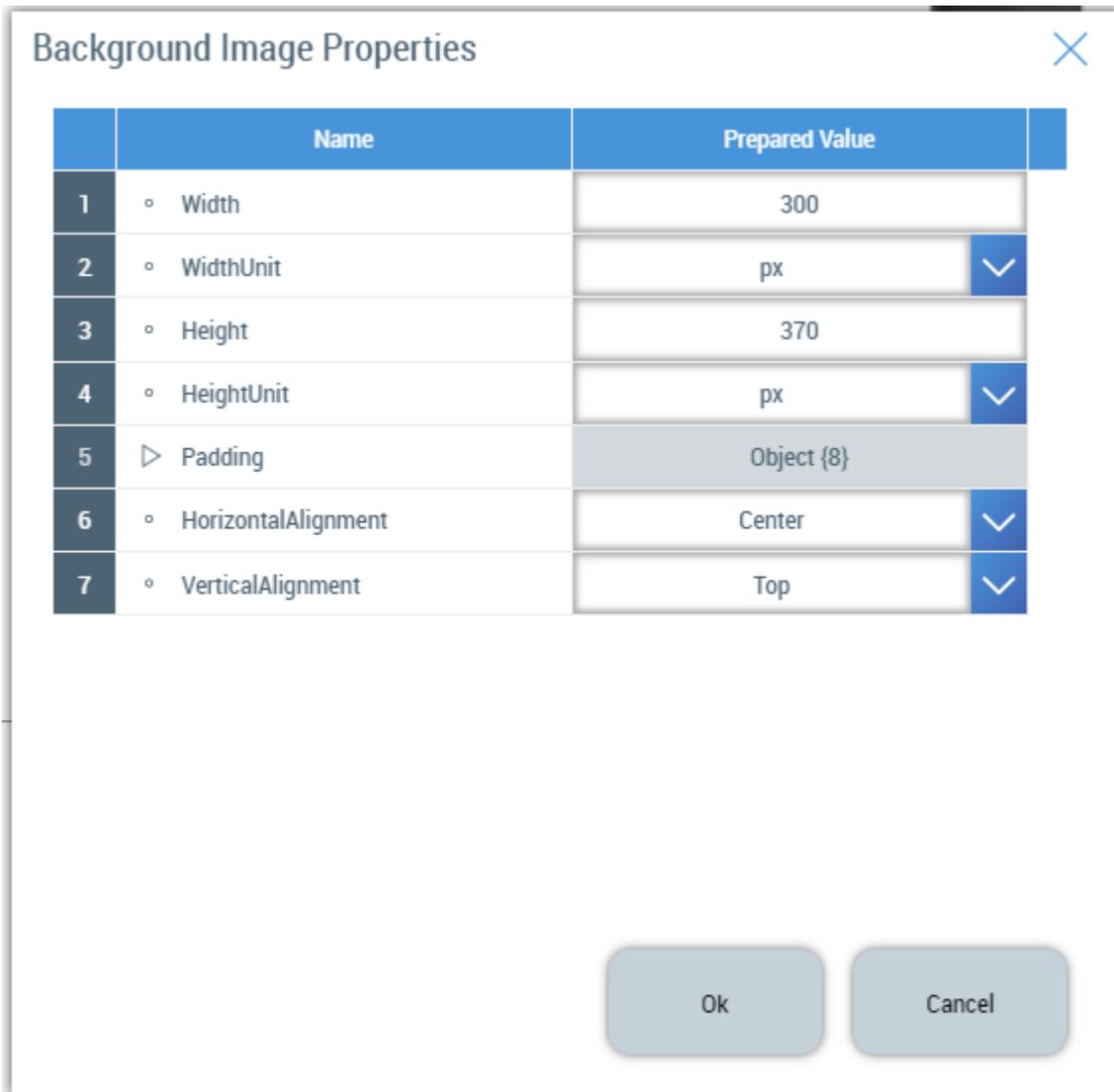
Sobald die richtige Datei in dem File-Explorer-Control ausgewählt ist, kann der Benutzer die Schaltfläche **OK** drücken, um die neue Änderung zu akzeptieren. Dadurch wird das ausgewählte Bild als Hintergrundbild für das Control verwendet.

Oder der Benutzer kann **Abbrechen** drücken, um den Wechsel des Hintergrundbildes zu vermeiden.

Das Attribut Mode **muss auf Config gesetzt werden, um diese Funktion zu nutzen. Für weitere Informationen besuchen Sie bitte [Mode.\[> 251\]](#)**

11.5.4.2 Hintergrundbild bearbeiten

In diesem Dialogfeld können alle Hintergrundbildeigenschaften des Controls bearbeitet werden.



Der Benutzer kann jede Eigenschaft bearbeiten und die Änderungen mit der Schaltfläche **Ok** bestätigen oder mit **Abbrechen** verwerfen.

Das Attribut Mode **muss auf** Config **gesetzt werden, um diese Funktion zu nutzen. Für weitere Informationen besuchen Sie bitte** [Mode \[▶ 251\]](#)

11.5.5 Events

Kategorie: Actions

Name	Beschreibung
onUserConfigActivated [▶ 261]	Dieses Ereignis wird generiert, wenn eine neue Konfiguration auf dem Control aktiviert wird.
onEditingCanceled [▶ 262]	Dieses Ereignis wird generiert, wenn Benutzeränderungen rückgängig gemacht werden und die alte Konfiguration wieder auf das Control geladen wird.

11.5.5.1 onUserConfigActivated

Dieses Ereignis wird generiert, wenn eine neue Konfiguration auf dem Control aktiviert wird.

Verfügbar: ab Version 12.8.0

11.5.5.2 onEditingCanceled

Dieses Ereignis wird generiert, wenn Benutzeränderungen rückgängig gemacht werden und die alte Konfiguration wieder auf das Control geladen wird.

Verfügbar: ab Version 12.8.0

11.5.6 Schritt für Schritt

11.5.6.1 Anzeige einer einzigen Layout-Konfiguration in verschiedenen Controls

Es ist möglich, das TcHMI-Projekt so zu organisieren, dass ein Benutzer mit speziellen Rechten nur das auf dem Control angezeigte Layout konfigurieren kann.

Dieses konfigurierte Layout sollte in anderen Inhalten mit verschiedenen Instanzen von ZoneImageLayout-Controls angezeigt werden, die im View-Modus arbeiten.

1. Erstellen Sie ein Serversymbol vom Typ `tchmi:framework#/definitions/ZoneImageLayoutConfig` und initialisieren Sie dessen Standardwerte.
2. Erstellen Sie eine Instanz des Konfigurations-Controls durch Drag-and-Drop in der Symbolleiste. Binden Sie das Attribut `ZoneImageLayoutConfig` dieses Controls an das im letzten Schritt erstellte Serversymbol. Behalten Sie die Bindung in beide Richtungen bei, damit Änderungen auf dem Serversymbol gespeichert werden.
3. Setzen Sie das Attribut `Mode` dieses Controls auf `Config`, da dieses Control zur Bearbeitung des Layouts verwendet wird.
4. Erstellen Sie eine neue Instanz des Controls, die nur für die Visualisierung des Layouts verwendet wird. Setzen Sie das Attribut `Mode` auf `View`.
5. Binden Sie das im 1. Schritt erstellte Serversymbol an sein Attribut `ZoneImageLayoutConfig`. Es ist nicht notwendig, einen Zwei-Wege-Bindungsmodus einzustellen, da dieses Control nur die Konfiguration anzeigt und sie nicht ändert.
6. Auf diese Weise kann dieselbe Konfiguration auf verschiedenen Controls in der HMI angezeigt werden.

Mehr Informationen:
www.beckhoff.com/tf8550

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

