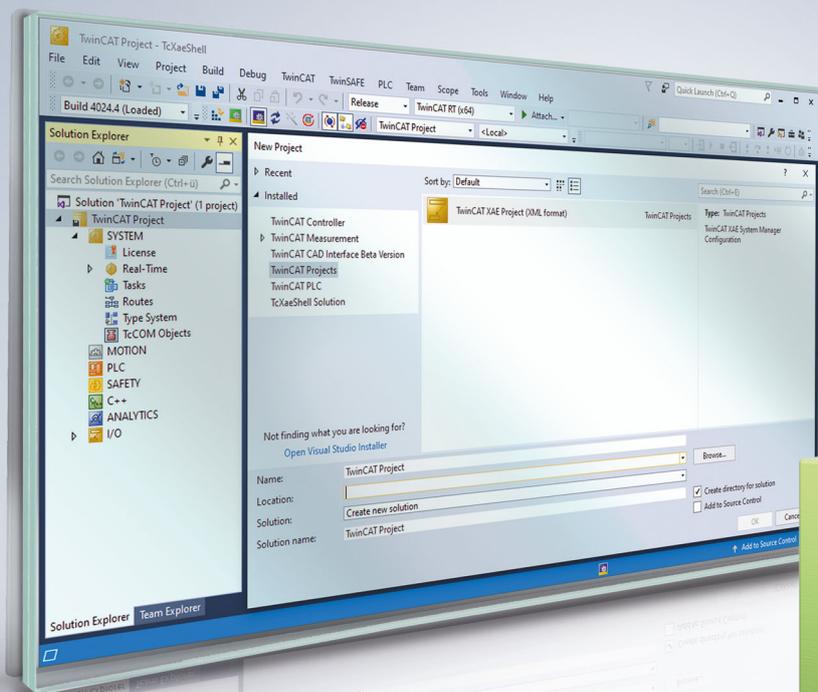


Handbuch | DE

# TE1000

TwinCAT 3 | Benutzeroberfläche





# Inhaltsverzeichnis

<b>1</b>	<b>Vorwort.....</b>	<b>13</b>
1.1	Hinweise zur Dokumentation .....	13
1.2	Zu Ihrer Sicherheit.....	14
1.3	Hinweise zur Informationssicherheit .....	15
<b>2</b>	<b>Komponenten der Benutzeroberfläche .....</b>	<b>16</b>
2.1	Standardmenüs.....	18
2.2	Standardsymbolleisten.....	18
2.3	Standardbefehle.....	19
2.4	Standardansichten und -fenster .....	27
2.5	Informationsleiste .....	28
2.6	Statussymbole.....	29
2.7	Systemmenü .....	29
2.7.1	Über TwinCAT.....	30
2.7.2	Echtzeit .....	31
2.7.3	Router .....	32
2.7.4	System .....	33
2.7.5	Tools .....	33
2.8	Benutzeroberfläche im Onlinebetrieb.....	34
<b>3</b>	<b>Benutzeroberfläche konfigurieren.....</b>	<b>35</b>
3.1	Optionen einstellen .....	35
3.1.1	Texteditoreinstellungen anpassen .....	35
3.1.2	Sprache auswählen.....	36
3.2	Benutzeroberfläche anpassen .....	36
3.2.1	Menüs anpassen.....	36
3.2.2	Symbolleisten anpassen .....	37
3.2.3	Tastaturkürzel anpassen.....	38
3.2.4	Menü- und Symbolleiste anordnen .....	39
3.2.5	Ansichten und Fenster anordnen .....	39
3.2.6	Zwischen Ansichten und Fenstern umschalten.....	40
3.2.7	Ansichten aus-/einblenden.....	40
3.2.8	Fenstergröße ändern .....	41
<b>4</b>	<b>TwinCAT Dokumentationen verwenden.....</b>	<b>42</b>
4.1	Aufruf der TwinCAT 3 Dokumentationen .....	42
4.2	Aktualisierung der TwinCAT 3 Dokumentationen .....	46
4.2.1	Aktualisierung in Visual Studio® .....	46
4.2.2	Aktualisierung in Visual Studio® 2010 .....	48
<b>5</b>	<b>Referenz Benutzeroberfläche .....</b>	<b>52</b>
5.1	Datei.....	52
5.1.1	Archivierungsmöglichkeiten .....	52
5.1.2	Befehl Projekt... (Neues TwinCAT-Projekt anlegen).....	58
5.1.3	Befehl Projekt/Projektmappe (Projekt/Projektmappe öffnen).....	60
5.1.4	Befehl Open Project from Target .....	61
5.1.5	Befehl Neues Projekt... (Neues TwinCAT-Projekt hinzufügen).....	61

5.1.6	Befehl Vorhandenes Projekt... (Vorhandenes TwinCAT-Projekt hinzufügen).....	61
5.1.7	Befehl Zuletzt geöffnete Projekte und Projektmappen.....	61
5.1.8	Befehl Alles speichern.....	62
5.1.9	Befehl Auswahl speichern/sichern .....	62
5.1.10	Befehl <Projektmappenname> speichern unter.....	62
5.1.11	Befehl Sichern <TwinCAT-Projektname> als.....	62
5.1.12	Befehl Sichern <SPS-Projektname> als .....	63
5.1.13	Befehl Disassemblierungsdatei erzeugen.....	63
5.1.14	Befehl Sende per E-Mail.../Send by E-Mail.....	63
5.1.15	Befehl Projekt/Projektmappe schließen .....	63
5.1.16	Befehl Schließen .....	63
5.1.17	Befehl Beenden.....	64
5.1.18	Befehl Seite einrichten.....	64
5.1.19	Befehl Drucken.....	64
5.2	Bearbeiten.....	64
5.2.1	Standardbefehle.....	64
5.2.2	Befehl Entfernen .....	65
5.2.3	Befehl Alles auswählen .....	65
5.2.4	Befehl Eingabehilfe .....	65
5.2.5	Befehl Variable deklarieren.....	67
5.2.6	Befehl Zur Überwachungsliste hinzufügen.....	71
5.2.7	Befehl Aufrufbaum ausgeben.....	72
5.2.8	Befehl Gehe zu .....	72
5.2.9	Befehl Gehe zur Definition .....	72
5.2.10	Befehl Gehe zur Instanz.....	73
5.2.11	Befehl Gehe zur Implementierung .....	73
5.2.12	Befehl Gehe zum Verweis.....	73
5.2.13	Befehl Verweise suchen.....	73
5.2.14	Befehl Navigiere zu .....	74
5.2.15	Befehl In Großbuchstaben umwandeln.....	74
5.2.16	Befehl In Kleinbuchstaben umwandeln .....	74
5.2.17	Befehl Leerstelle anzeigen.....	74
5.2.18	Befehl Auswahl auskommentieren.....	74
5.2.19	Befehl Auskommentierung der Auswahl aufheben .....	74
5.2.20	Befehl Schnellsuche (In Dateien suchen) .....	75
5.2.21	Befehl Schnellerersetzung (In Dateien ersetzen).....	76
5.2.22	Befehl Schreibmodus umschalten.....	78
5.2.23	Befehl Umbenennen .....	78
5.2.24	Befehl Objekt (offline) bearbeiten.....	78
5.2.25	Befehl Refactoring - <Variable> umbenennen .....	78
5.2.26	Befehl Refactoring - Variable hinzufügen.....	80
5.2.27	Befehl Refactoring - <Variable> entfernen.....	81
5.2.28	Befehl Refactoring - Variablen neu ordnen .....	82
5.3	Ansicht .....	82
5.3.1	Befehl Objekt öffnen.....	82
5.3.2	Befehl Textuelle Ansicht.....	83

5.3.3	Befehl Tabellarische Ansicht.....	83
5.3.4	Befehl Ganzer Bildschirm.....	84
5.3.5	Befehl Symbolleiste.....	84
5.3.6	Befehl Projektmappen-Explorer .....	84
5.3.7	Befehl Eigenschaftenfenster .....	85
5.3.8	Befehl Werkzeugkasten .....	86
5.3.9	Befehl Fehlerliste .....	87
5.3.10	Befehl Ausgabe.....	88
5.4	Projekt.....	89
5.4.1	Befehl Neues Element hinzufügen (Projekt) .....	89
5.4.2	Befehl Vorhandenes Element hinzufügen (Projekt) .....	91
5.4.3	Befehl Eigenschaften (Objekt) .....	94
5.4.4	Befehl Eigenschaften (SPS-Projekt) .....	99
5.4.5	SPS-Projekteinstellungen .....	120
5.5	Erstellen .....	124
5.5.1	Befehl Projektmappe erstellen .....	124
5.5.2	Befehl Projektmappe neu erstellen .....	125
5.5.3	Befehl Projektmappe bereinigen .....	125
5.5.4	Befehl Überprüfe alle Objekte .....	125
5.5.5	Befehl TwinCAT-Projekt erstellen .....	126
5.5.6	Befehl TwinCAT-Projekt neu erstellen .....	126
5.5.7	Befehl TwinCAT-Projekt bereinigen .....	126
5.5.8	Befehl SPS-Projekt erstellen.....	126
5.5.9	Befehl SPS-Projekt neu erstellen.....	127
5.5.10	Befehl SPS-Projekt bereinigen.....	127
5.6	Debuggen.....	127
5.6.1	Befehl Neuer Haltepunkt.....	127
5.6.2	Befehl Haltepunkt bearbeiten.....	131
5.6.3	Befehl Haltepunkt aktivieren .....	131
5.6.4	Befehl Haltepunkt deaktivieren .....	131
5.6.5	Befehl Haltepunkt umschalten .....	131
5.6.6	Befehl Prozedurschritt.....	132
5.6.7	Befehl Einzelschritt.....	132
5.6.8	Befehl Ausführen bis Rücksprung.....	132
5.6.9	Befehl Ausführen bis Cursor .....	133
5.6.10	Befehl Nächste Anweisung anzeigen.....	133
5.6.11	Befehl Nächste Anweisung festlegen.....	133
5.7	TwinCAT.....	134
5.7.1	Befehl Konfiguration aktivieren .....	134
5.7.2	Befehl Restart TwinCAT System.....	134
5.7.3	Befehl Restart TwinCAT (Config Mode).....	134
5.7.4	Befehl Reload Devices.....	134
5.7.5	Befehl Scan.....	135
5.7.6	Befehl Toggle Free Run State.....	135
5.7.7	Befehl Show Online Data .....	135
5.7.8	Befehl Choose Target System .....	135

5.7.9	Befehl Show Sub Items .....	135
5.7.10	Befehl Software Protection.....	136
5.7.11	Befehl Hide Disabled Items .....	136
5.8	PLC .....	136
5.8.1	Fenster .....	136
5.8.2	Core Dump.....	150
5.8.3	SPS Lesezeichen.....	152
5.8.4	Befehl Laden .....	154
5.8.5	Befehl Online-Change .....	154
5.8.6	Befehl Einloggen .....	156
5.8.7	Befehl Start .....	157
5.8.8	Befehl Stop.....	158
5.8.9	Befehl Ausloggen .....	158
5.8.10	Befehl Reset kalt.....	158
5.8.11	Befehl Reset Ursprung.....	159
5.8.12	Befehl Einzelzyklus .....	159
5.8.13	Befehl Ablaufkontrollmodus umschalten .....	159
5.8.14	Befehl Werte forcen .....	160
5.8.15	Befehl Forcen aufheben.....	161
5.8.16	Befehl Werte schreiben.....	162
5.8.17	Befehl Darstellung - Binär, Dezimal, Hexadezimal .....	163
5.8.18	Befehl Darstellung Vererbung – Einfach, Strukturiert .....	163
5.8.19	Befehl Lokalisierungsvorlage erzeugen .....	163
5.8.20	Befehl Lokalisierungen verwalten .....	164
5.8.21	Befehl Lokalisierung umschalten .....	164
5.8.22	Befehl Active PLC Project.....	165
5.8.23	Befehl Active PLC Instance.....	165
5.9	Extras .....	165
5.9.1	Befehl Optionen .....	165
5.9.2	Befehl Anpassen .....	196
5.10	Fenster .....	199
5.10.1	Befehl Verankerung aufheben .....	199
5.10.2	Befehl Andocken .....	199
5.10.3	Befehl Ausblenden .....	200
5.10.4	Befehl Alle automatisch ausblenden .....	200
5.10.5	Befehl Automatisch in den Hintergrund.....	200
5.10.6	Befehl Registerkarte anheften.....	201
5.10.7	Befehl Neue horizontale Registerkartengruppe .....	201
5.10.8	Befehl Neue vertikale Registerkartengruppe .....	201
5.10.9	Befehl Fenster-Layout zurücksetzen.....	201
5.10.10	Befehl Alle Dokumente schließen .....	202
5.10.11	Befehl Fenster.....	202
5.10.12	Befehle des Untermenüs Fenster .....	202
5.11	AS .....	202
5.11.1	Befehl Initialschritt.....	202
5.11.2	Befehl Schritt-Transition einfügen .....	203

5.11.3	Befehl Schritt-Transition danach einfügen .....	203
5.11.4	Befehl Parallel .....	203
5.11.5	Befehl Alternativ .....	204
5.11.6	Befehl Verzweigung einfügen .....	204
5.11.7	Befehl Verzweigung rechts einfügen.....	204
5.11.8	Befehl Aktionsassoziation einfügen .....	206
5.11.9	Befehl Aktionsassoziation danach einfügen.....	207
5.11.10	Befehl Sprung einfügen.....	207
5.11.11	Befehl Sprung danach einfügen.....	207
5.11.12	Befehl Makro einfügen .....	208
5.11.13	Befehl Makro danach einfügen .....	208
5.11.14	Befehl Makro anzeigen .....	208
5.11.15	Befehl Makro verlassen.....	209
5.11.16	Befehl Einfügen danach.....	209
5.11.17	Befehl Eingangsaktion hinzufügen.....	209
5.11.18	Befehl Ausgangsaktion hinzufügen.....	210
5.11.19	Befehl Change duplication - Setzen.....	211
5.11.20	Befehl Change duplication - Entfernen .....	211
5.11.21	Befehl Schritt einfügen .....	211
5.11.22	Befehl Schritt danach einfügen .....	212
5.11.23	Befehl Transition einfügen .....	212
5.11.24	Befehl Transition danach einfügen.....	212
5.12	CFC .....	213
5.12.1	Befehl Arbeitsblatt bearbeiten .....	213
5.12.2	Befehl Seitengröße bearbeiten .....	213
5.12.3	Befehl Negieren .....	214
5.12.4	Befehl EN/ENO .....	214
5.12.5	Befehl Kein.....	214
5.12.6	Befehl R (Reset).....	215
5.12.7	Befehl S (Set).....	215
5.12.8	Befehl REF= (Reference-Zuweisung) .....	215
5.12.9	Befehl Ausführungsreihenfolge anzeigen .....	216
5.12.10	Befehl Startpunkt der Rückkopplung setzen .....	216
5.12.11	Befehl An den Anfang .....	217
5.12.12	Befehl Ans Ende .....	217
5.12.13	Befehl Eins vor.....	218
5.12.14	Befehl Eins zurück .....	218
5.12.15	Befehl Ausführungsreihenfolge setzen .....	218
5.12.16	Befehl Nach Datenfluss anordnen .....	219
5.12.17	Befehl Topologisch anordnen .....	219
5.12.18	Befehl Selektierte Anschlüsse verbinden.....	220
5.12.19	Befehl Verbindung lösen.....	220
5.12.20	Befehl Nächste Kollision zeigen.....	220
5.12.21	Befehl Verbundene Anschlüsse selektieren.....	221
5.12.22	Befehl Attributierte Komponente als Eingang verwenden.....	221
5.12.23	Befehl Anschlüsse zurücksetzen .....	222

5.12.24	Befehl Nicht verbundene Anschlüsse entfernen .....	222
5.12.25	Befehl Eingangsanschluss hinzufügen .....	222
5.12.26	Befehl Ausgangsanschluss hinzufügen .....	222
5.12.27	Befehl Alle Verbindungen routen .....	223
5.12.28	Befehl Kontrollpunkt erzeugen .....	223
5.12.29	Befehl Kontrollpunkt entfernen .....	223
5.12.30	Befehl Verbindungsmarke .....	224
5.12.31	Befehl Gruppe erzeugen .....	224
5.12.32	Befehl Gruppierung aufheben .....	224
5.12.33	Befehl Parameter bearbeiten .....	225
5.12.34	Befehl FB-Eingang forcen .....	226
5.12.35	Befehl Vorbereitete Parameter im Projekt speichern .....	227
5.13	FUP/KOP/AWL .....	227
5.13.1	Befehl Kontakt einfügen (rechts) .....	227
5.13.2	Befehl Netzwerk einfügen .....	228
5.13.3	Befehl Netzwerk einfügen (unterhalb) .....	228
5.13.4	Befehl Kommentierung ein/aus .....	228
5.13.5	Befehl Zuweisung einfügen .....	228
5.13.6	Befehl Bausteinaufruf einfügen .....	229
5.13.7	Befehl Baustein mit EN/ENO einfügen .....	229
5.13.8	Befehl Leeren Baustein einfügen .....	229
5.13.9	Befehl Leeren Baustein mit EN/ENO einfügen .....	230
5.13.10	Befehl Sprung einfügen .....	230
5.13.11	Befehl Sprungmarke einfügen .....	230
5.13.12	Befehl Return einfügen .....	231
5.13.13	Befehl Bausteineingang einfügen .....	231
5.13.14	Befehl Baustein parallel einfügen (unterhalb) .....	231
5.13.15	Befehl Spule einfügen .....	231
5.13.16	Befehl Set-Spule einfügen .....	232
5.13.17	Befehl Reset-Spule einfügen .....	232
5.13.18	Befehl Kontakt einfügen .....	232
5.13.19	Befehl Kontakt parallel einfügen (unterhalb) .....	233
5.13.20	Befehl Kontakt parallel einfügen (oberhalb) .....	233
5.13.21	Befehl Negierten Kontakt einfügen .....	233
5.13.22	Befehl Negierten Kontakt parallel einfügen (unterhalb) .....	234
5.13.23	Befehl Kontakte einfügen: Darunter einfügen .....	234
5.13.24	Befehl Kontakte einfügen: Darüber einfügen .....	234
5.13.25	Befehl Kontakte einfügen: Rechts einfügen (danach) .....	234
5.13.26	Befehl AWL-Zeile danach einfügen .....	235
5.13.27	Befehl AWL-Zeile löschen .....	235
5.13.28	Befehl Negation .....	235
5.13.29	Befehl Flankenerkennung .....	235
5.13.30	Befehl Set/Reset .....	236
5.13.31	Befehl Weiterverschaltung festlegen .....	236
5.13.32	Befehl Leitungsverzweigung einfügen .....	236
5.13.33	Befehl Leitungsverzweigung oberhalb einfügen .....	237

5.13.34	Befehl Leitungsverzweigung unterhalb einfügen .....	237
5.13.35	Befehl Verzweigung Startpunkt setzen .....	237
5.13.36	Befehl Verzweigung Endpunkt setzen .....	237
5.13.37	Befehl Parallelen Modus wechseln .....	238
5.13.38	Befehl Parameter aktualisieren .....	238
5.13.39	Befehl Nicht verwendete FB-Aufruf-Parameter entfernen.....	238
5.13.40	Befehl Baustein reparieren.....	239
5.13.41	Befehl Als Funktionsbausteinsprache anzeigen .....	239
5.13.42	Befehl Als Kontaktplan anzeigen .....	239
5.13.43	Befehl Als Anweisungsliste anzeigen.....	240
5.13.44	Befehl Gehe zu .....	240
5.14	Ladder-Editor .....	240
5.14.1	Befehl Auskommentiert .....	240
5.14.2	Befehl Negieren .....	241
5.14.3	Befehl Parallele Verzweigung öffnen .....	241
5.14.4	Befehl Parallele Verzweigung schließen.....	241
5.14.5	Befehl Set/Reset - Set, Set/Reset - Reset .....	242
5.14.6	Befehl Flankenerkennung Steigende Flanke .....	242
5.14.7	Befehl Flankenerkennung: Fallende Flanke.....	242
5.14.8	Befehl EN/ENO: EN .....	242
5.14.9	Befehl EN/ENO: ENO .....	243
5.14.10	Befehl Netzwerk einfügen .....	243
5.14.11	Befehl Kontakt einfügen.....	243
5.14.12	Befehl Spule einfügen .....	244
5.14.13	Befehl Baustein einfügen .....	244
5.14.14	Befehl Sprung einfügen.....	245
5.14.15	Befehl Return einfügen .....	245
5.14.16	Befehl Eingang einfügen .....	245
5.14.17	Befehl Ausgang einfügen .....	246
5.14.18	Befehl In neuen Ladder konvertieren.....	246
5.15	Deklarationen .....	246
5.15.1	Befehl Einfügen.....	246
5.15.2	Befehl Deklarationskopf editieren .....	246
5.15.3	Befehl Nach unten verschieben .....	247
5.15.4	Befehl Nach oben verschieben .....	248
5.16	Textliste.....	248
5.16.1	Befehl Sprache einfügen.....	248
5.16.2	Befehl Sprache entfernen .....	248
5.16.3	Befehl Text einfügen .....	249
5.16.4	Befehl Import/Export Textlisten .....	249
5.16.5	Befehl Nicht verwendete Textlisteneinträge entfernen.....	250
5.16.6	Befehl Visualisierungstext-IDs prüfen .....	251
5.16.7	Befehl Visualisierungstext-IDs aktualisieren .....	251
5.16.8	Befehl Alles als Text exportieren.....	251
5.16.9	Befehl Alles als Unicode-Text exportieren .....	252
5.16.10	Befehl Textlistenunterstützung hinzufügen .....	252

5.16.11	Befehl Textlistenunterstützung entfernen.....	252
5.17	Rezepturen.....	253
5.17.1	Befehl Rezeptur einfügen.....	253
5.17.2	Befehl Rezeptur entfernen .....	253
5.17.3	Befehl Rezeptur laden.....	253
5.17.4	Befehl Rezeptur speichern.....	254
5.17.5	Befehl Rezeptur lesen.....	254
5.17.6	Befehl Rezeptur schreiben.....	255
5.17.7	Befehl Rezeptur laden und schreiben .....	255
5.17.8	Befehl Rezeptur lesen und speichern .....	255
5.17.9	Befehl Variable einfügen .....	256
5.17.10	Befehl Variablen entfernen.....	256
5.17.11	Befehl Strukturierte Variablen aktualisieren.....	256
5.17.12	Befehl Rezepturen vom Gerät laden.....	258
5.18	Bibliothek.....	258
5.19	Visualisierung .....	259
5.19.1	Befehl Schnittstellen-Editor .....	259
5.19.2	Befehl Tastaturkonfiguration .....	259
5.19.3	Befehl Elementliste .....	259
5.19.4	Befehl Links ausrichten .....	259
5.19.5	Befehl Oben ausrichten.....	259
5.19.6	Befehl Rechts ausrichten .....	260
5.19.7	Befehl Unten ausrichten.....	260
5.19.8	Befehl Vertikal zentrieren .....	260
5.19.9	Befehl Horizontal zentrieren.....	260
5.19.10	Befehl Horizontalen Abstand gleichmachen .....	260
5.19.11	Befehl Horizontalen Abstand vergrößern .....	260
5.19.12	Befehl Horizontalen Abstand verkleinern .....	261
5.19.13	Befehl Horizontalen Abstand entfernen .....	261
5.19.14	Befehl Vertikalen Abstand gleichmachen.....	261
5.19.15	Befehl Vertikalen Abstand vergrößern .....	261
5.19.16	Befehl Vertikalen Abstand verkleinern .....	262
5.19.17	Befehl Vertikalen Abstand entfernen.....	262
5.19.18	Befehl Breite gleichmachen .....	262
5.19.19	Befehl Höhe gleichmachen .....	262
5.19.20	Befehl Größe gleichmachen.....	262
5.19.21	Befehl Größe an Raster anpassen.....	263
5.19.22	Befehl Um Eins nach vorn legen.....	263
5.19.23	Befehl Nach vorn legen.....	263
5.19.24	Befehl Um Eins nach hinten legen .....	263
5.19.25	Befehl Nach hinten legen .....	263
5.19.26	Befehl Gruppieren .....	264
5.19.27	Befehl Gruppierung aufheben .....	264
5.19.28	Befehl Hintergrund .....	264
5.19.29	Befehl Alles auswählen .....	265
5.19.30	Befehl Alles deselektieren.....	265

5.19.31	Befehl Visualisierungselement vervielfachen.....	265
5.19.32	Befehl Tastaturbedienung aktivieren.....	266
5.20	Sonstiges .....	267
5.20.1	Befehl Schnittstellen implementieren .....	267
5.21	Kontextmenü Projektmappe .....	268
5.21.1	Befehl Projektmappe erstellen .....	268
5.21.2	Befehl Projektmappe neu erstellen .....	268
5.21.3	Befehl Projektmappe bereinigen .....	268
5.21.4	Befehl Save <Projektmappenname> as Archive.....	269
5.21.5	Befehl Sende per E-Mail.../Send by E-Mail.....	269
5.22	Kontextmenü TwinCAT Projekt .....	269
5.22.1	Befehl Sichern <TwinCAT-Projektname> als Archiv... ..	269
5.22.2	Befehl Sende per E-Mail.../Send by E-Mail.....	270
5.22.3	Befehl Sichere <TwinCAT-Projektnamen> automatisch auf dem Zielsystem.....	270
5.22.4	Befehl <TwinCAT-Projektname> mit dem Zielsystem vergleichen... ..	270
5.22.5	Befehl Projekt mit Zielsystem aktualisieren.....	270
5.22.6	Befehl Projekt mit TwinCAT 2.xx Version laden.....	270
5.22.7	Befehl Verborgene Konfigurationen anzeigen .....	271
5.22.8	Befehl Aus Projektmappe entfernen .....	271
5.22.9	Befehl Umbenennen .....	271
5.22.10	Befehl TwinCAT-Projekt erstellen .....	271
5.22.11	Befehl TwinCAT-Projekt neu erstellen .....	271
5.22.12	Befehl TwinCAT-Projekt bereinigen .....	272
5.22.13	Befehl Projekt entladen .....	272
5.22.14	Import AutomationML via AML DataExchange.....	272
5.22.15	Export AutomationML.....	272
5.23	Kontextmenü SPS-Projekt .....	273
5.23.1	Befehl Bootprojekt aktivieren .....	273
5.23.2	Befehl Autostart Bootprojekt .....	273
5.23.3	Befehl ADS-Port ändern.....	274
5.23.4	Befehl Projektbibliotheken installieren .....	274
5.23.5	Befehl Projektbibliotheken installieren (Unbekannte Versionen) .....	275
5.23.6	Befehl Projektbibliotheksordner aktualisieren .....	275
5.23.7	Befehl Projekt ändern.....	276
5.23.8	Befehl Neues Element hinzufügen (Instanz).....	276
5.23.9	Befehl Sichern <SPS-Projektname> als Archiv... ..	277
5.23.10	Befehl Sende per E-Mail.../Send by E-Mail.....	278
5.23.11	Befehl Projekt mit Zielsystem aktualisieren... ..	278
5.23.12	Befehl Unabhängige Projektdatei.....	278
5.23.13	Befehl Deaktivieren .....	278
5.24	Kontextmenü SPS-Projektobjekt (<SPS-Projektname> Project) .....	278
5.24.1	Befehl Einloggen .....	278
5.24.2	Befehl SPS-Projekt erstellen.....	280
5.24.3	Befehl SPS-Projekt neu erstellen.....	280
5.24.4	Befehl Überprüfe alle Objekte .....	280
5.24.5	Befehl SPS-Projekt bereinigen.....	280

5.24.6	Hinzufügen .....	281
5.24.7	Befehl Als ZIP exportieren .....	282
5.24.8	Befehl Aus ZIP importieren .....	282
5.24.9	Befehl PLCopenXML exportieren.....	282
5.24.10	Befehl PLCopenXML importieren.....	283
5.24.11	Befehl Als Bibliothek speichern.....	283
5.24.12	Befehl Als Bibliothek speichern und installieren.....	285
5.24.13	Befehl Als SPS-Projektvorlage speichern .....	286
5.24.14	Befehl Eigenschaften (SPS-Projekt) .....	288

# 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®, ATRO®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, MX-System®, Safety over EtherCAT®, TC/BSD®, TwinCAT®, TwinCAT/BSD®, TwinSAFE®, XFC®, XPlanar® und XTS® 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 Kennzeichnungen führen.



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.

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

### Fremdmarken

In dieser Dokumentation können Marken Dritter verwendet werden. Die zugehörigen Markenvermerke finden Sie unter: <https://www.beckhoff.com/trademarks>.

## 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.

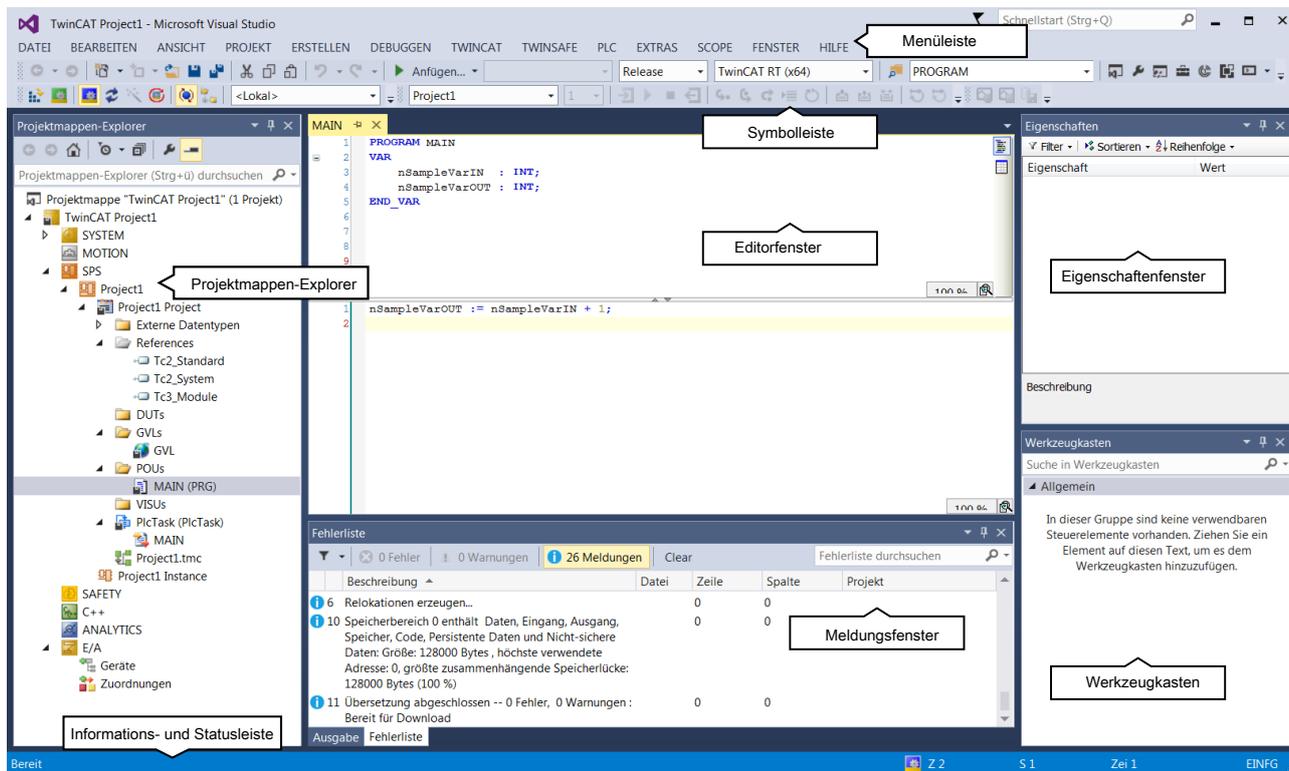
Zusätzlich sollten die Empfehlungen von Beckhoff zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Informationssicherheit und Industrial Security finden Sie in unserem <https://www.beckhoff.de/secguide>.

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>.

## 2 Komponenten der Benutzeroberfläche

Das TwinCAT-3-Engineering setzt sich aus verschiedenen Komponenten zusammen. Die Anordnung und Konfiguration der einzelnen Komponenten bestimmen das Aussehen der Benutzeroberfläche.



### Standardkomponenten

Menüleiste	Zeigt die Menüs gemäß den Einstellungen im Dialog <b>Anpassen</b> .
Symbolleiste	Zeigt die Befehle als mit Symbolen gekennzeichneten Schaltflächen gemäß den Einstellungen im Dialog <b>Anpassen</b> .
Werkzeugkasten	Zeigt die für den gerade aktiven Editor vorhandenen „Werkzeuge“ (z. B. grafische Programmiererelemente)
Projektmappen-Explorer	Zeigt das TwinCAT-3-Projekt mit den dazugehörigen Projektelementen in strukturierter Form.
Eigenschaftsfenster	Zeigt die Eigenschaften des gerade im Projektmappen-Explorer selektierten Elements.
Editorfenster	Dient dem Definieren und Bearbeiten von Objekten. Bei Spracheditoren (z. B. ST-Editor, CFC-Editor) enthält das Editorfenster üblicherweise im unteren Teil den Spracheditor (Implementierungsteil) und im oberen Teil den Deklarationseditor. Bei anderen Editoren kann das Editorfenster auch Dialoge enthalten (z. B. Task-Editor, Geräte-Editor).

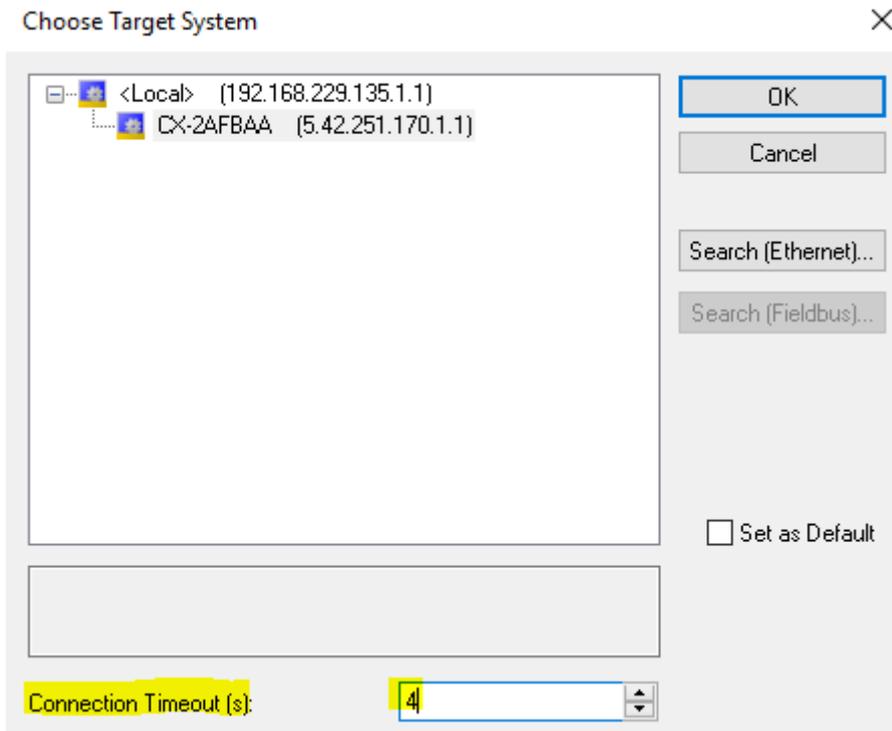
Information über die aktuellen Vorgänge im Projekt im Offline- oder Onlinebetrieb bieten die folgenden Komponenten:

Meldungsfenster	Zeigt aktuell vorliegende Fehler, Warnungen und Meldungen bezüglich Syntaxprüfung, Kompilervorgang etc.
Überwachungsfenster und Online-Ansichten von Editoren	Dient dem „Monitoring“ einer POU bzw. einer benutzerdefinierten Liste von Ausdrücken/Variablen.
Informations- und Statusleiste	Zeigt den Status der TwinCAT-3-Runtime. Wenn gerade ein Editorfenster aktiv ist, werden die aktuelle Position des Cursors und der eingestellte Editiermodus angezeigt. Im Onlinebetrieb sehen Sie den augenblicklichen Status des Programms.

Mit TwinCAT installieren Sie bestimmte Standardeinstellungen, die das Aussehen, die Struktur und das Verhalten der Standardkomponenten und somit der Benutzeroberfläche bestimmen. Im Abschnitt „[Benutzeroberfläche konfigurieren](#) [► 35]“ wird beschrieben, wie Sie die Standardeinstellungen bearbeiten und so die Benutzeroberfläche individuell anpassen können. Im Abschnitt „[Referenz Benutzeroberfläche](#) [► 52]“ erhalten Sie detaillierte Informationen zu den Befehlen.

### Angezeigter Status der TwinCAT-3-Runtime

Neben dem aktuellen Betriebsmodus der TwinCAT-3-Runtime (Run-Mode oder Config-Mode) wird auch die Qualität der Verbindung zur ausgewählten TwinCAT-3-Runtime dargestellt. Die Auswertung der Verbindungsqualität bezieht sich dabei immer auf das aktuell eingestellte Verbindungs-Timeout.



Das eingestellte Timeout wird dabei intern in Micro-Timeouts und Wiederholungen eingeteilt. Somit ergibt sich folgendes:

Eingestelltes Verbindungs-Timeout = Micro-Timeouts \* Anzahl der Wiederholungen.

Für die Verteilung zwischen Micro-Timeout und Wiederholungen gibt es eine Fallunterscheidung:

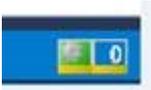
Wenn das eingestellte Verbindungs-Timeout  $\geq 30s$ ,  
dann Micro-Timeout = 5s Wiederholungen = eingestelltes Verbindungs-Timeout / 5

Wenn das eingestellte Verbindungs-Timeout  $\geq 10s$ ,  
dann Micro-Timeout = 2s Wiederholungen = eingestelltes Verbindungs-Timeout / 2

Sonst: Micro-Timeout = 500ms, Wiederholungen = eingestelltes Verbindungs-Timeout \* 2

Nach jedem Micro-Timeout wird erneut versucht, mit dem Target-System zu kommunizieren. Es wird keine Fehlermeldung ausgegeben. Diese Information wird u.a. dazu verwendet, um die Qualität der Verbindung mit dem Zielsystem auszuwerten.

Hierbei gibt es die folgenden 3 Stufen:

Gelber Balken: 	Verbindung ist von guter Qualität: weniger als ¼ der Micro-Timeouts sind abgelaufen.
Oranger Balken: 	1. Warnstufe: ¼ der Micro-Timeouts sind abgelaufen.
Roter Balken: 	2. Warnstufe: die Hälfte der Micro-Timeouts sind abgelaufen.
Blinkverhalten	Die Anzeige blinkt in den hier erklärten Farben und zeigt so die Micro-Timeouts an.

## 2.1 Standardmenüs

Die Menüleiste ist standardmäßig sichtbar und umfasst die in der Abbildung dargestellten Hauptmenüs.



Das Kontextmenü des TwinCAT Projektes kann standardmäßig aufgerufen werden.



## 2.2 Standardsymbolleisten

Die Symbolleisten ermöglichen Ihnen einen Schnellzugriff auf die Befehle. Die folgenden Symbolleisten sind standardmäßig eingestellt.

### Standard Symbolleistenoptionen



### TwinCAT SPS Symbolleistenoptionen



### TwinCAT XAE Base Symbolleistenoptionen



## 2.3 Standardbefehle

Die nachfolgend aufgelisteten Befehle sind standardmäßig in den Hauptmenüs enthalten. Editorspezifische Menüs und Befehle erscheinen nur, wenn der betreffende Editor geöffnet ist. Weitere Befehle sind verfügbar, die Sie in bereits vorhandenen oder neu definierten Menüs ergänzen können.

### Datei

Das Menü **Datei** enthält Befehle für Aktionen zur Projektdatei (Öffnen, Schließen, Speichern, Drucken, Seiteneinstellungen, ...)

Symbol	Befehl
	Befehl Projekt... (Neues TwinCAT-Projekt anlegen) [▶ 58]
	Befehl Projekt/Projektmappe (Projekt/Projektmappe öffnen) [▶ 53]
	Befehl Open Project from Target [▶ 61]
	Befehl Open Solution from Archive [▶ 53]
	Befehl Neues Projekt... (Neues TwinCAT-Projekt hinzufügen) [▶ 61]
	Befehl Vorhandenes Projekt... (Vorhandenes TwinCAT-Projekt hinzufügen) [▶ 61]
	Befehl Zuletzt geöffnete Projekte und Projektmappen [▶ 61]
	Befehl Alles speichern [▶ 62]
	Befehl Auswahl speichern/sichern [▶ 62]
	Befehl <Projektmappenname> speichern unter [▶ 62]
	Befehl Save <Projektmappenname> as Archive... [▶ 52]
	Befehl Sichern <TwinCAT-Projektname> als [▶ 62]
	Befehl Sichern <TwinCAT-Projektname> als Archiv... [▶ 269]
	Befehl Sichern <SPS-Projektname> als [▶ 63]
	Befehl Sichern <SPS-Projektname> als Archiv... [▶ 54]
	Befehl Sende per E-Mail.../Send by E-Mail... [▶ 63]
	Befehl Projekt/Projektmappe schließen [▶ 63]
	Befehl Schließen [▶ 63]
	Befehl Beenden [▶ 64]
	Befehl Seite einrichten... [▶ 64]
	Befehl Drucken [▶ 64]

## Bearbeiten

Das Menü **Bearbeiten** enthält Befehle zum Arbeiten in Texteditoren (Spracheditoren, Deklarationseditor...)

Symbol	Befehl
	Kopieren
	Ausschneiden
	Löschen
	Einfügen
	Wiederholen
	Rückgängig
	<a href="#">Befehl Entfernen</a> [▶ 65]
	<a href="#">Befehl Alles auswählen</a> [▶ 65]
	<a href="#">Befehl Eingabehilfe</a> [▶ 65]
	<a href="#">Befehl Variable deklarieren</a> [▶ 67]
	<a href="#">Befehl Gehe zu</a> [▶ 72]
	<a href="#">Befehl Gehe zur Definition</a> [▶ 72]
	<a href="#">Befehl Navigiere zu</a> [▶ 74]
	<a href="#">Befehl In Großbuchstaben umwandeln</a> [▶ 74]
	<a href="#">Befehl In Kleinbuchstaben umwandeln</a> [▶ 74]
	<a href="#">Befehl Leerstelle anzeigen</a> [▶ 74]
	<a href="#">Befehl Schnellersetzung (In Dateien ersetzen)</a> [▶ 76]
	<a href="#">Befehl Schnellsuche (In Dateien suchen)</a> [▶ 75]
	<a href="#">Befehl Schreibmodus umschalten</a> [▶ 78]
	<a href="#">Befehl Umbenennen</a> [▶ 78]
	<a href="#">Befehl Refactoring - &lt;Variable&gt; umbenennen</a> [▶ 78]
	<a href="#">Befehl Refactoring - &lt;Variable&gt; entfernen</a> [▶ 81]
	<a href="#">Befehl Refactoring - Variable hinzufügen</a> [▶ 80]
	<a href="#">Befehl Refactoring - Variablen neu ordnen</a> [▶ 82]

**Ansicht**

Das Menü **Ansicht** enthält Befehle zur Aktivierung der einzelnen Standardansichten in Fenstern der Benutzeroberfläche.

Symbol	Befehl
	Befehl Objekt öffnen [▶ 82]
	Befehl Tabellarische Ansicht [▶ 83]
	Befehl Textuelle Ansicht [▶ 83]
	Befehl Ganzer Bildschirm [▶ 84]
	Befehl Symbolleisten [▶ 84]
	Befehl Projektmappen-Explorer [▶ 84]
	Befehl Eigenschaftfenster [▶ 85]
	Befehl Werkzeugkasten [▶ 86]
	Befehl Fehlerliste [▶ 87]
	Befehl Ausgabe [▶ 88]

## Projekt

Das Menü **Projekt** enthält Befehle zur Handhabung der Projektobjekte und der Projekteigenschaften sowie für das Kopieren und Exportieren von Projekten bzw. Objekten.

Symbol	Befehl
	Befehl Neues Element hinzufügen (Projekt) [▶ 89]
	Befehl Vorhandenes Element hinzufügen (Projekt) [▶ 55]
	Befehl Neues Element hinzufügen (Objekt) [▶ 281]
	Befehl Als ZIP exportieren [▶ 282]
	Befehl Aus ZIP importieren [▶ 282]
	Befehl PLCopenXML exportieren [▶ 282]
	Befehl PLCopenXML importieren [▶ 283]
	Befehl Ordner hinzufügen [▶ 282]
	Befehl Eigenschaften (SPS-Projekt) [▶ 99]
	Befehl Eigenschaften (Objekt) [▶ 94]
	SPS-Projekteinstellungen [▶ 120]

**Erstellen**

Das Menü **Erstellen** enthält Befehle für das Erstellen durch Übersetzen des Projekts (ohne Code-Erzeugung), d. h. für das Ausführen eines Precompile-Laufs mit Syntaxprüfung und Befehle für das Löschen der letzten Übersetzungsinformationen (Bereinigen), was für einen Online-Change sowie die Offline-Code-Erzeugung von Bedeutung ist.

Symbol	Befehl
	Befehl SPS-Projekt erstellen [▶ 126]
	Befehl SPS-Projekt neu erstellen [▶ 127]
	Befehl Projektmappe erstellen [▶ 124]
	Befehl Projektmappe neu erstellen [▶ 125]
	Befehl SPS-Projekt bereinigen [▶ 127]
	Befehl Projektmappe bereinigen [▶ 125]
	Befehl Überprüfe alle Objekte [▶ 125]

**Debuggen**

Das Menü **Debuggen** enthält Befehle für das Kontrollieren des Programmlaufs auf der Steuerung (Start, Stopp) und für Debugging-Aktionen (Haltepunkte, schrittweise Abarbeitung, Werte schreiben und forcen).

Symbol	Befehl
	Befehl Neuer Haltepunkt [▶ 127]
	Befehl Haltepunkt bearbeiten [▶ 131]
	Befehl Haltepunkt aktivieren [▶ 131]
	Befehl Haltepunkt deaktivieren [▶ 131]
	Befehl Haltepunkt umschalten [▶ 131]
	Befehl Prozedurschritt [▶ 132]
	Befehl Einzelschritt [▶ 132]
	Befehl Ausführen bis Rücksprung [▶ 132]
	Befehl Nächste Anweisung anzeigen [▶ 133]
	Befehl Ausführen bis Cursor [▶ 133]

**TwinCAT**

Das Menü **TwinCAT** enthält Befehle zur Steuerung der TwinCAT-3-Runtime-Umgebung.

Symbol	Befehl
	Befehl Konfiguration aktivieren [▶ <a href="#">134</a> ]
	Befehl Restart TwinCAT System [▶ <a href="#">134</a> ]
	Befehl Restart TwinCAT (Config Mode) [▶ <a href="#">134</a> ]
	Befehl Reload Devices [▶ <a href="#">134</a> ]
	Befehl Scan [▶ <a href="#">135</a> ]
	Befehl Toggle Free Run State [▶ <a href="#">135</a> ]
	Befehl Show Online Data [▶ <a href="#">135</a> ]
	Befehl Show Sub Items [▶ <a href="#">135</a> ]
	Befehl Hide Disabled Items [▶ <a href="#">136</a> ]
	Befehl Choose Target System [▶ <a href="#">135</a> ]
	Befehl Software Protection [▶ <a href="#">136</a> ]
	Befehl Bootprojekt aktivieren [▶ <a href="#">273</a> ]

## PLC

Das Menü **PLC** enthält Befehle für das Öffnen von Werkzeugen, die dazu dienen, die Entwicklungsumgebung zum Arbeiten mit einem SPS-Projekt bereitzustellen.

Symbol	Befehl
	Befehl Haltepunkte [ <a href="#">▶ 143</a> ]
	Befehl Verweise suchen [ <a href="#">▶ 73</a> ]
	Befehl Querverweisliste [ <a href="#">▶ 141</a> ]
	Befehl Aufrufliste [ <a href="#">▶ 145</a> ]
	Befehl Aufrufbaum [ <a href="#">▶ 146</a> ]
	Befehl Aufrufbaum ausgeben [ <a href="#">▶ 72</a> ]
	Befehl Überwachungsliste <n> [ <a href="#">▶ 140</a> ]
	Befehl Alle Forces anzeigen [ <a href="#">▶ 140</a> ]
	Befehl Zur Überwachungsliste hinzufügen [ <a href="#">▶ 71</a> ]
	Befehl Einloggen [ <a href="#">▶ 156</a> ]
	Befehl Ausloggen [ <a href="#">▶ 158</a> ]
	Befehl Laden [ <a href="#">▶ 154</a> ]
	Befehl Reset kalt [ <a href="#">▶ 158</a> ]
	Befehl Reset Ursprung [ <a href="#">▶ 159</a> ]
	Befehl Online-Change [ <a href="#">▶ 154</a> ]
	Befehl Start [ <a href="#">▶ 157</a> ]
	Befehl Stop [ <a href="#">▶ 158</a> ]
	Befehl Einzelzyklus [ <a href="#">▶ 159</a> ]
	Befehl Werte forcen [ <a href="#">▶ 160</a> ]
	Befehl Werte schreiben [ <a href="#">▶ 162</a> ]
	Befehl Forcen aufheben [ <a href="#">▶ 161</a> ]
	Befehl Ablaufkontrollmodus umschalten [ <a href="#">▶ 159</a> ]
	Befehl Darstellung - Binär, Dezimal, Hexadezimal [ <a href="#">▶ 163</a> ]
	Befehl Lokalisierungsvorlage erzeugen [ <a href="#">▶ 163</a> ]
	Befehl Lokalisierungen verwalten [ <a href="#">▶ 164</a> ]
	Befehl Lokalisierung umschalten [ <a href="#">▶ 164</a> ]
	Befehl Active PLC Project [ <a href="#">▶ 165</a> ]
	Befehl Active PLC Instance [ <a href="#">▶ 165</a> ]
	Befehl Speicheransicht anzeigen [ <a href="#">▶ 136</a> ]

## Extras

Das Menü **Extras** enthält Befehle zum Öffnen von Werkzeugen (Tools), mit denen eine bestimmte Umgebung für das Arbeiten am Projekt geschaffen wird (Installation von Bibliotheken und Geräten, Anpassen der Benutzeroberfläche, Optionen für Editoren, Laden und Speichern etc.).

Symbol	Befehl
	<a href="#">Befehl Optionen [► 165]</a>
	<a href="#">Befehl Anpassen [► 196]</a>

## Fenster

Das Menü **Fenster** enthält Befehle zum Positionieren der Ansichten in der Benutzeroberfläche (Anordnung, Öffnen, Schließen etc.).

Symbol	Befehl
	<a href="#">Befehl Verankerung aufheben [► 199]</a>
	<a href="#">Befehl Andocken [► 199]</a>
	<a href="#">Befehl Ausblenden [► 200]</a>
	<a href="#">Befehl Alle automatisch ausblenden [► 200]</a>
	<a href="#">Befehl Automatisch in den Hintergrund [► 200]</a>
	<a href="#">Befehl Registerkarte anheften [► 201]</a>
	<a href="#">Befehl Neue horizontale Registerkartengruppe [► 201]</a>
	<a href="#">Befehl Neue vertikale Registerkartengruppe [► 201]</a>
	<a href="#">Befehl Fenster-Layout zurücksetzen [► 201]</a>
	<a href="#">Befehl Alle Dokumente schließen [► 202]</a>
	<a href="#">Befehl Fenster [► 202]</a>
	<a href="#">Befehle des Untermenüs Fenster [► 202]</a>

## Kontextmenü TwinCAT Projekt

Symbol	Befehl
	Befehl Sichern <TwinCAT –Projektname> als Archiv... [▶ 269]
	Befehl Alles speichern [▶ 62]
	Befehl Sende <TwinCAT-Projektname> per E-Mail... [▶ 63]
	Befehl Sichere <TwinCAT-Projektname> automatisch auf dem Zielsystem [▶ 270]
	Befehl <TwinCAT-Projektname> mit dem Zielsystem vergleichen... [▶ 270]
	Befehl Projekt mit Zielsystem aktualisieren... [▶ 270]
	Befehl Projekt mit TwinCAT 2.xx Version laden... [▶ 270]
	Befehl Verborgene Konfigurationen anzeigen [▶ 271]
	Befehl Aus Projektmappe entfernen [▶ 271]
	Befehl Umbenennen [▶ 271]
	Befehl <TwinCAT-Projektname> erstellen [▶ 126]
	Befehl <TwinCAT-Projektname> neu erstellen [▶ 127]
	Befehl Projekt entladen [▶ 272]
	Befehl AutomationML importieren... [▶ 272]
	Befehl AutomationML exportieren... [▶ 272]

## 2.4 Standardansichten und -fenster

In TwinCAT werden Fenster und Ansichten unterschieden.

### Ansichten

Als „Ansicht“ wird ein von einem SPS-Objekt unabhängiges Fenster innerhalb des Hauptfensters der Benutzeroberfläche bezeichnet. Ansichten sind durch folgende Eigenschaften gekennzeichnet:

- Sie können über das Menü **Ansicht** geöffnet werden.
- Sie enthalten meistens eine nicht-konfigurierbare Symbolleiste mit Schaltflächen zum Sortieren, Anzeigen und Suchen innerhalb der Ansicht.
- Sie können am Rahmen des Hauptfensters angedockt oder losgelöst davon an beliebiger Stelle auf dem Bildschirm positioniert werden.
- Sie können ausgeblendet und nur durch einen Reiter im Rahmen des Hauptfensters repräsentiert werden.

Folgende Ansichten sind standardmäßig geöffnet:

- Projektmappen-Explorer
- Fehlerliste
- Ausgabe
- Werkzeugkasten
- Eigenschaftenfenster

## Fenster

Als „Fenster“ wird ein von einem SPS-Objekt abhängiges Fenster bezeichnet. Fenster sind durch folgende Eigenschaften gekennzeichnet:

- Sie können über das Menü **Fenster** oder durch einen Doppelklick auf das Objekt geöffnet werden.
- Sie werden alle in einem bestimmten Bereich der Benutzeroberfläche als MDI-Fenster in Form von Registerblättern angeordnet. Die Anordnung der Fenster hängt von den benutzerspezifischen Einstellungen ab.
- Sie können nicht ausgeblendet werden.
- Sie sind standardmäßig nicht geöffnet.

Beispiele:

- Editorfenster der SPS-Objekte

## 2.5 Informationsleiste

Die Informationsleiste zeigt die aktuelle Cursor-Position und den Editiermodus sowie den Modus der TwinCAT-3-Runtime an, abhängig davon, welche Ansicht bzw. welches Fenster aktiv ist.

### Cursor-Position

Die aktuelle Cursor-Position wird angezeigt, wenn ein Editor-Fenster aktiv ist.

Aktuelle Cursor-Position vom linken bzw. oberen Rand des Editor-Fensters ausgehend:

Z (Ln)	Zeilennummer
S (Col)	Spaltennummer (eine Spalte enthält jeweils genau 1 Leerstelle bzw. Buchstaben oder Ziffer)
ZeI (Ch)	Anzahl der Zeichen (hier kann ein Zeichen, ein einzelner Buchstabe, eine einzelne Ziffer oder auch eine Tabulatorspanne, die z. B. 4 Spalten umfasst, sein)

Nach einem Doppelklick auf eines der Felder öffnet sich der Dialog **Gehe zu Zeile**. Hier können Sie eine Zeile angeben, in der der Cursor platziert werden soll.

### Editiermodus

Der aktuelle Editiermodus wird angezeigt, wenn ein Editor-Fenster aktiv ist.

Aktuell eingestellter Editiermodus:

EINF (INS)	Einfügemodus
ÜB	Überschreibmodus

Mit einem Doppelklick auf das Feld können Sie die Einstellung umschalten.

### Zustand TwinCAT-System

Sie erkennen den Zustand des TwinCAT-Systems anhand der Farbe des TwinCAT-System-Service-Symbols im Informationsbereich der Taskleiste. Folgende Zustände sind möglich:

	TwinCAT-3-Runtime läuft im Config-Modus
	TwinCAT-3-Runtime läuft im Run-Modus
	TwinCAT-3-Runtime befindet sich im Stop-Modus
	TwinCAT-3-Runtime befindet sich im Exception-Modus
	Blinkverhalten zeigt die <u>Micro-Timeouts</u> [ <a href="#">▶ 16</a> ] an.

## 2.6 Statussymbole

### Status der SPS-Module

SPS-Objekt	SPS-Instanz	Status
		TwinCAT im Config-Modus SPS ausgeloggt SPS gestoppt
		TwinCAT im Run-Modus SPS eingeloggt SPS gestoppt
		TwinCAT im Run-Modus SPS eingeloggt SPS gestartet
		TwinCAT im Run-Modus SPS ausgeloggt SPS gestoppt
		TwinCAT im Run-Modus SPS ausgeloggt SPS gestartet
		TwinCAT im Run-Modus Ein anderer Anwender ist auf der SPS eingeloggt. SPS gestartet

### Status von SPS-Objekten

	Das Objekt wurde seit dem letzten Speichern nicht verändert.
	Das Objekt wurde seit dem letzten Speichern verändert.
	Das Objekt ist signiert.
	Das Objekt ist verschlüsselt.

## 2.7 Systemmenü

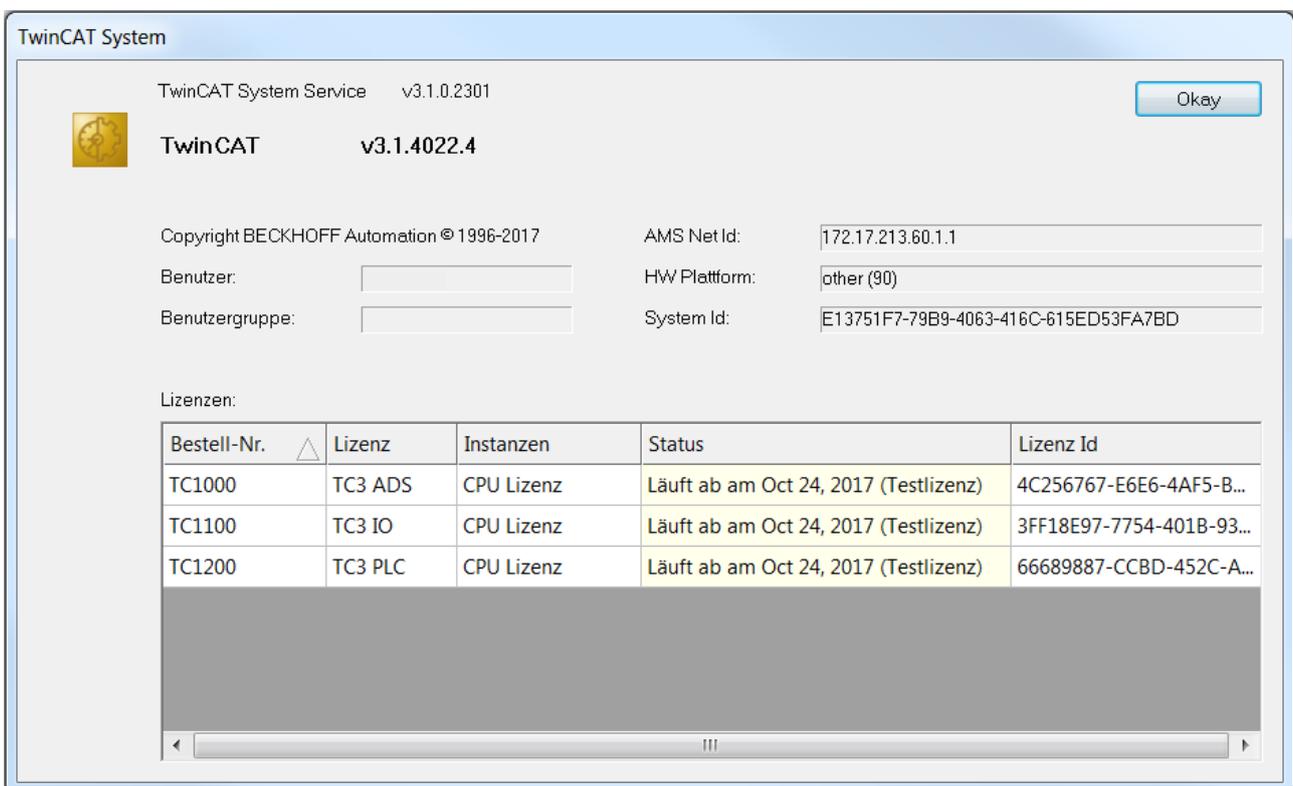
Das TwinCAT-Systemmenü stellt alle wichtigen Befehle für den Zugriff auf Funktionen des TwinCAT-Systems bereit. Neben wichtigen Runtime-Befehlen zum Starten und Stoppen der Runtime enthält das Menü Befehle für den Zugriff auf die Router-Einstellungen und die Kommunikation zwischen Engineering und Runtime. Die Router-Befehle sind vor allem dann wichtig, wenn sich Engineering und Runtime auf getrennten Rechnern befinden. Über das TwinCAT-Systemmenü kann außerdem die Engineering-Umgebung direkt gestartet werden.

Sie öffnen das Systemmenü mit einem Rechtsklick auf das TwinCAT-System-Service-Symbol im Informationsbereich der Taskleiste.

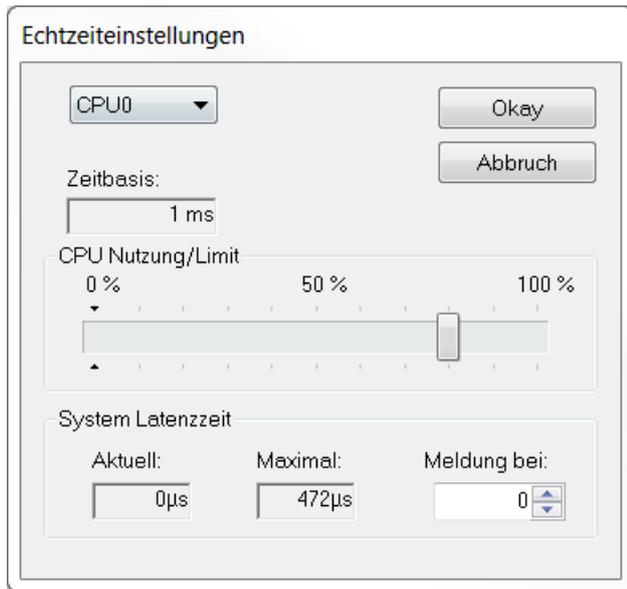


## 2.7.1 Über TwinCAT...

Im Dialog **TwinCAT System** sind alle wesentlichen Daten der installierten TwinCAT-Version aufgelistet. Diese können Sie bei Bedarf in die Zwischenablage kopieren oder als E-Mail an die TwinCAT-Support-Hotline senden.



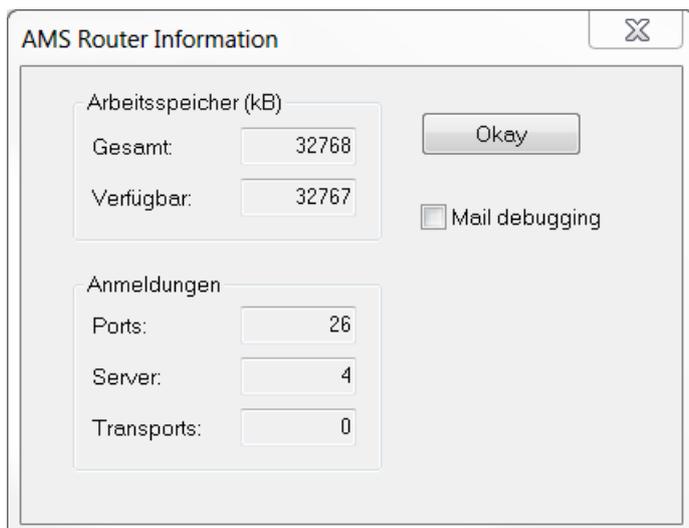
## 2.7.2 Echtzeit



Zeitbasis	Zeitbasis zur Berechnung der Prozeenteilung. Zurzeit ist eine Zeitbasis von einer Millisekunde festeingestellt.
CPU Nutzung/Limit	Über den Schieberegler kann dem TwinCAT-Echtzeitsystem Prozessorzeit zugeteilt werden. Aktuell ist hierfür eine Zeitbasis von einer Millisekunde festgelegt. Im gezeigten Beispiel bekommt TwinCAT maximal 80 % der Rechenzeit zugeteilt. Dies bedeutet bei einer Zeitbasis von 1 ms, dass TwinCAT jede Millisekunde maximal 800 µs zur Verfügung hat. Auf der anderen Seite heißt dies, dass für Windows mindestens 200 µs verbleiben. Wenn das TwinCAT-Echtzeitsystem in seine Idletask schaltet, wird der Prozessor wieder an Windows zurückgegeben. Der Balken im Schieberegler zeigt die aktuelle Auslastung des Echtzeitsystems an. Die Anzeige wird über 256 Zyklen (ms) gemittelt.
System Latenzzeit	Hier werden die aktuelle bzw. die maximale Latenzzeit im Echtzeitsystem angezeigt. Es wird die Zeit gemessen, um die der zentrale Systemtick zu spät kommt. Die maximale Zeit wird so lange gespeichert, bis der Schieberegler betätigt wird oder der Dialog verlassen wird. Die Latenzzeit wird natürlich auch gemessen, wenn der Dialog nicht geöffnet ist.  Bei Überschreiten der eingestellten Maximalzeit, wird einmalig eine Meldung im Fenster ausgegeben, und ein Logbucheintrag erzeugt. Durch Aufruf des Dialogs kann die Meldung wieder zurückgesetzt werden, sodass bei der nächsten Zeitüberschreitung wieder eine Meldung ausgegeben wird.

## 2.7.3 Router

### Info



### Arbeitsspeicher

Der angezeigte Arbeitsspeicher wird im TwinCAT-System für AMS-Nachrichten und für die Speicherverwaltung der TwinCAT-Echtzeitumgebung benötigt. Der gesamte Speicher wird beim Start vom TwinCAT-System bei Windows angefordert. Die Speichergröße kann im TwinCAT-System Knoten konfiguriert werden.

Gesamt	Arbeitsspeicher, der beim Systemstart von TwinCAT zur Verfügung stand.
Verfügbar	Arbeitsspeicher, der aktuell für TwinCAT zur Verfügung steht.

### Anmeldungen

Alle Teilnehmer des TwinCAT-Nachrichtensystems (AMS) müssen sich beim Router anmelden. TwinCAT Server haben feste Portnummern (z. B. TwinCAT PLC LZS1: 851, ...), TwinCAT Clients bekommen eine Portnummer vom Message Router zugeteilt.

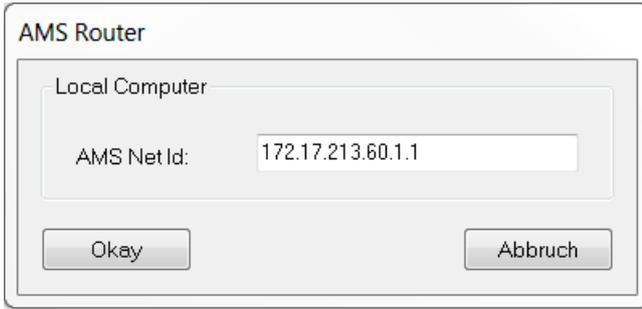
Ports	Anzahl der angemeldeten Ports
Server	Ports, die von TwinCAT Servern belegt werden
Transports	Anzahl der angemeldeten Teilnehmer (NetIds)

Mail debugging	In Verbindung mit dem TwinCAT ADS Monitor kann der gesamte TwinCAT-Nachrichtenverkehr aufgezeichnet werden. Zu beachten ist hierbei, dass der Nachrichtenverkehr durch die zusätzlichen Nachrichten verlangsamt wird. Der TwinCAT ADS Monitor gehört nicht zum Standardlieferungsumfang von TwinCAT.
----------------	--

### Cleanup

Mit der Cleanup-Funktion des Routers werden Ports im Router wieder freigegeben, die von nicht mehr funktionsfähigen Programmen stammen. Diese Funktion ist besonders in der Entwicklungsphase der SPS-Projekte sinnvoll.

**AMS NetId ändern**



**Local Computer**

AMS Net Id	Die AMS Net Id ist die Adresse des lokalen Rechners im TwinCAT-Netzwerk. Die AMS Net Id besteht aus 6 Byte und wird in einer Punktnotation dargestellt. Die „Net Ids“ müssen vom Projektierer vergeben werden und dürfen sich im TwinCAT-Netzwerk nicht wiederholen. Standardmäßig wird von der Installation eine AMS Net Id aus der IP-Adresse des Systems (falls vorhanden) + „1.1“ generiert. Wenn bei der Installation keine IP-Adresse ermittelt werden kann, so wird die AMS Net Id „1.1.1.1.1.1“ erzeugt.
------------	--

**Routes editieren**

Der Dialog **TwinCAT Static Routes** zeigt alle Routen-Informationen des lokalen TwinCAT-Systems an. Über die Schaltfläche **Add** können Sie neue Routen hinzufügen.

Weitere Informationen finden Sie in der Dokumentation Grundlagen im Kapitel Routes.

**2.7.4 System**

Wenn das TwinCAT-System nicht für einen automatischen Start konfiguriert ist, können Sie es auch manuell über das Systemmenü starten.

Start/Neustart	Das TwinCAT-System wird gestartet. Alle eingetragenen TwinCAT Server werden geladen und initialisiert. Das TwinCAT-I/O-Subsystem wird entsprechend der Konfiguration durch den TwinCAT-I/O-Manager parametrisiert. Alle eingetragenen Laufzeitsysteme des SPS-Subsystems werden initialisiert. Wenn für ein Laufzeitsystem ein Bootprojekt eingetragen ist, so wird dieses geladen und das SPS-Programm gestartet. Die remanenten (retain) Daten werden ebenfalls entsprechend der Konfiguration geladen.
Konfig	Das TwinCAT-System wird gestoppt. Alle eingetragenen TwinCAT Server werden heruntergefahren und entladen. Nach dem Stopp des TwinCAT-Systems befindet sich nur noch der TwinCAT Message Router im Speicher. Das TwinCAT System kann nun wieder über System > Start gestartet werden.

**2.7.5 Tools**

Tool		Beschreibung
Event Viewer		Öffnet die Windows-Ereignisanzeige, in welcher Systemfehler eingetragen werden
TwinCAT Project Compare		Öffnet ein Anwendungsprogramm zum Vergleichen oder Zusammenführen von TwinCAT-Projekten Siehe: Dokumentation Source-Control > Einleitung
TwinCAT Switch Runtime		Öffnet ein Anwendungsprogramm zum Umschalten der TwinCAT-Laufzeitumgebung zwischen TwinCAT 2 und TwinCAT 3

## 2.8 Benutzeroberfläche im Onlinebetrieb

Sobald Sie sich mit einem Projekt auf das Zielsystem einloggen, werden alle Objekte, die im Offlinebetrieb in Fenstern und Ansichten geöffnet waren, im Onlinebetrieb dargestellt.

Weiter Informationen zu den Online-Ansichten der einzelnen Editoren finden Sie in der jeweiligen Beschreibung des Editors.

**Siehe auch:**

- Dokumentation PLC: AS-Editor im Onlinebetrieb
- Dokumentation PLC: CFC-Editor im Onlinebetrieb
- Dokumentation PLC: FUP/KOP/AWL-Editor im Onlinebetrieb
- Dokumentation PLC: ST-Editor im Onlinebetrieb

### 3 Benutzeroberfläche konfigurieren

Sie können das Verhalten, das Aussehen, die Menüzusammensetzung und die Fensteranordnung von TwinCAT individuell konfigurieren. Im Menü **Extras** finden Sie Dialoge zum Anpassen der Oberfläche und für das Einstellen der TwinCAT-Optionen.

**Siehe auch:**

- [Befehl Optionen \[► 165\]](#)
- [Befehl Anpassen \[► 196\]](#)

#### 3.1 Optionen einstellen

Sie können das Verhalten und Aussehen von TwinCAT in den verschiedenen Registerkarten des Dialogs **Optionen** konfigurieren. Der Dialog öffnet nach Auswahl des Befehls **Optionen** im Menü **Extras**. Sie können hier die Einstellungen für verschiedene Editoren und Funktionalitäten vornehmen. Diese Einstellungen sind TwinCAT-weit gültig. Die Einstellungen werden in Ihrem aktuellen Benutzerprofil auf Ihrem lokalen System gespeichert.

**Siehe auch:**

- [Befehl Optionen \[► 165\]](#)
- [Benutzeroberfläche anpassen \[► 36\]](#)

##### 3.1.1 Texteditoreinstellungen anpassen

Um in unterschiedlichen Programmen ein einheitliches Schriftbild zu erreichen, sollten Sie identische Schrift- und Editoreinstellungen verwenden. Sie können diese Einstellungen in den TwinCAT-Optionen anpassen.

Bei der Integration von TwinCAT 3 ins Visual Studio werden u. a. Schriftart und Tabulatorgröße standardmäßig voreingestellt.

###### Anpassen der Schriftart und -farben

1. Wählen Sie im Menü **Extras** den Befehl **Optionen**.  
⇒ Der Dialog **Optionen** öffnet sich.
2. Wählen Sie die Kategorie **Umgebung > Schriftarten und Farben** aus.
3. Wählen Sie die gewünschten Schrifteinstellungen aus.
4. Schließen Sie den Dialog mit **OK**.  
⇒ Nach einem Programmneustart werden die Einstellungen übernommen.

###### Standardeinstellungen:

Element	Nur Text
Schriftart	Consolas
Schriftgrad	10
Elementvordergrund	Standard (schwarz)
Elementhintergrund	Standard (weiß)

###### Anpassen der Textgliederung

1. Wählen Sie im Menü **Extras** den Befehl **Optionen**.  
⇒ Der Dialog **Optionen** öffnet sich.
2. Wählen Sie die Kategorie **TwinCAT > SPS Programmierumgebung > Texteditor** aus.
3. Wählen Sie die gewünschten Gliederungseinstellungen aus.
4. Schließen Sie den Dialog mit **OK**.  
⇒ Nach einem Programmneustart werden die Einstellungen übernommen.

**Standardeinstellungen:**

Anzahl von Rückgängig Schritten	100
Gliederung	Einrückung
Wortumbruch	Keine Umbrüche
Tabulatorgröße	4
Tabs beibehalten	Ja
Einzugsgrößen	4
Automatisch einrücken	Intelligent mit Code-Komplettierung

**3.1.2 Sprache auswählen**

1. Wählen Sie im Menü **Extras** den Befehl **Optionen**.  
⇒ Der Dialog **Optionen** öffnet sich.
2. Wählen Sie die Kategorie **Umgebung > Internationale Einstellungen** aus.
3. Wählen Sie eine Sprache aus.  
⇒ Nach einem Programmneustart werden die Einstellungen übernommen.

**3.2 Benutzeroberfläche anpassen**

TwinCAT bietet Ihnen die Möglichkeit, die Benutzeroberfläche individuell an Ihre Bedürfnisse anzupassen. Dabei können Sie die Anordnung der Fenster sowie die Sichtbarkeit von Menüs und Befehlen frei gestalten.

**3.2.1 Menüs anpassen**

Sie können die Menübefehle der Benutzeroberfläche an Ihre Bedürfnisse anpassen. Über den Konfigurationsdialog **Anpassen** können Sie Menüs ausblenden oder hinzufügen.

---

**● Standardeinstellung wiederherstellen**

**I** Mit der Schaltfläche **Alles zurücksetzen** im Dialog **Extras > Anpassen > Befehle** können Sie die von Ihnen vorgenommenen Änderungen in den Anpassungseinstellungen zurücksetzen.

---

**● Spezielle Menüleisten**

**I** In einigen Ansichten wie z. B. im Projektmappen-Explorer gibt es eine spezielle Menüleiste mit Schaltflächen zum Sortieren, Anzeigen und Suchen im Fenster. Diese Menüleisten können nicht konfiguriert werden.

---

**Entfernen eines Menüs oder Befehls**

1. Wählen Sie im Menü **Extras** den Befehl **Anpassen**.  
⇒ Der Dialog **Anpassen** öffnet sich. Die Registerkarte **Symbolleisten** ist sichtbar.
2. Wählen Sie die Registerkarte **Befehle**.  
⇒ Das Feld **Menüleiste** ist standardmäßig aktiviert.
3. Wählen Sie in dem dazugehörigen Listenfeld das Menü aus, welches Sie bearbeiten möchten.  
⇒ Im Bereich **Befehle** werden die vorhandenen Untermenüs und Befehle angezeigt.
4. Wählen Sie ein Menü oder einen Befehl aus.
5. Klicken Sie auf **Löschen**.  
⇒ Das Menü bzw. der Befehl ist aus dem Menübaum gelöscht.
6. Klicken Sie auf **Schließen**.  
⇒ Der Dialog schließt und das Menü ist angepasst.

**Hinzufügen eines Menüs**

1. Wählen Sie im Menü **Extras** den Befehl **Anpassen**.

- ⇒ Der Dialog **Anpassen** öffnet sich. Die Registerkarte **Symbolleisten** ist sichtbar.
2. Wählen Sie die Registerkarte **Befehle**.
  - ⇒ Das Feld **Menüleiste** ist standardmäßig aktiviert.
3. Wählen Sie in dem dazugehörigen Listenfeld das Menü aus, welches sie bearbeiten möchten.
  - ⇒ Im Bereich **Steuerelemente** werden die vorhandenen Untermenüs und Befehle angezeigt.
4. Klicken Sie auf **Neues Menü hinzufügen**.
  - ⇒ Ein neues Menü wird ergänzt.
5. Klicken Sie auf das Kombinationsfeld **Auswahl ändern** und geben Sie einen Namen für das neue Menü ein.
6. Verändern Sie die Position in der Reihenfolge der Menüs mit den Schaltflächen **Nach oben** und **Nach unten**.
7. Klicken Sie auf **Schließen**.
  - ⇒ Der Dialog schließt. Das neue Menü wird erst vollständig angezeigt, wenn innerhalb des Menüs ein Befehl verfügbar ist.

### Hinzufügen eines Befehls

1. Wählen Sie im Menü **Extras** den Befehl **Anpassen**.
  - ⇒ Der Dialog **Anpassen** öffnet sich. Die Registerkarte **Symbolleisten** ist sichtbar.
2. Wählen Sie die Registerkarte **Befehle**.
  - ⇒ Das Feld **Menüleiste** ist standardmäßig aktiviert.
3. Wählen Sie in dem dazugehörigen Listenfeld das Menü aus, welches sie bearbeiten möchten.
  - ⇒ Im Bereich **Befehle** werden die vorhandenen Untermenüs und Befehle angezeigt.
4. Klicken Sie auf **Befehl hinzufügen...**.
  - ⇒ Der Dialog **Befehl hinzufügen** öffnet. In dem Dialog sind alle Befehle in Kategorien gruppiert aufgelistet.
5. Wählen Sie den Befehl, den Sie hinzufügen wollen und Klicken Sie auf **OK**.
  - ⇒ Der neue Befehl wird zum Menübaum hinzugefügt.
6. Verändern Sie die Position in der Reihenfolge der Menüs mit den Schaltflächen **Nach oben** und **Nach unten**.
7. Klicken Sie auf **Schließen**.
  - ⇒ Der Dialog schließt und der neue Befehl ist im Menü verfügbar.

### Siehe auch:

- [Befehl Anpassen](#) [► 196]

## 3.2.2 Symbolleisten anpassen

Sie können die Symbolleisten der TwinCAT-Benutzeroberfläche an Ihre Bedürfnisse anpassen. Über den Konfigurationsdialog **Anpassen** können Sie Symbolleisten ausblenden oder hinzufügen.

### Ein- und Ausblenden einer Symbolleiste

1. Wählen Sie im Menü **Extras** den Befehl **Anpassen**.
  - ⇒ Der Dialog **Anpassen** öffnet sich. Die Registerkarte **Symbolleisten** ist sichtbar.
2. Markieren Sie die Kontrollkästchen der Symbolleisten, welche Sie einblenden bzw. ausblenden möchten.
  - ⇒ Die Symbolleiste wird auf der Benutzeroberfläche ein- bzw. ausgeblendet
3. Klicken Sie auf **Schließen**.
  - ⇒ Der Dialog schließt.

### Hinzufügen und Entfernen einer Symbolleiste

1. Wählen Sie im Menü **Extras** den Befehl **Anpassen**.

- ⇒ Der Dialog **Anpassen** öffnet sich. Die Registerkarte **Symbolleisten** ist sichtbar.
2. Klicken Sie auf **Neu**.
  - ⇒ Der Dialog **Neue Symbolleiste** öffnet.
3. Geben Sie einen Namen ein und klicken Sie auf **OK**.
  - ⇒ Die neue noch leere Symbolleiste ist automatisch aktiviert und wird unterhalb der Menüleiste hinzugefügt.
4. Klicken Sie **Schließen**.
  - ⇒ Der Dialog schließt. Die neue Symbolleiste ist sichtbar. Vollständig sichtbar wird die neue Symbolleiste erst, wenn innerhalb der Symbolleiste ein Befehl verfügbar ist.

Mit dem Befehl **Löschen** in der Registerkarte **Anpassen > Symbolleisten** können Sie eine von Ihnen zusammengestellte Symbolleiste wieder entfernen.

### Hinzufügen und Entfernen eines Befehls

1. Wählen Sie im Menü **Extras** den Befehl **Anpassen**.
  - ⇒ Der Dialog **Anpassen** öffnet sich. Die Registerkarte **Symbolleisten** ist sichtbar.
2. Wählen Sie die Registerkarte **Befehle**.
  - ⇒ Das Feld **Menüleiste** ist standardmäßig aktiviert.
3. Aktivieren Sie das Feld **Symbolleiste** und wählen Sie in dem dazugehörigen Listenfeld das Menü aus, welches sie bearbeiten möchten.
4. Klicken Sie auf **Befehl hinzufügen....**
  - ⇒ Der Dialog **Befehl hinzufügen** öffnet sich. In dem Dialog sind alle Befehle in Kategorien gruppiert aufgelistet.
5. Wählen Sie den Befehl, den Sie hinzufügen wollen und klicken Sie auf **OK**.
  - ⇒ Der neue Befehl wird der Symbolleiste hinzugefügt.
6. Verändern Sie die Position in der Reihenfolge der Menüs mit den Schaltflächen **Nach oben** und **Nach unten**.
7. Klicken Sie auf **Schließen**.
  - ⇒ Der Dialog schließt. Der neue Befehl ist in der Symbolleiste verfügbar.

Mit dem Befehl **Löschen** in der Registerkarte **Anpassen > Befehle** können Sie einen Befehl wieder von der Symbolleiste entfernen.

### Siehe auch:

- [Befehl Anpassen \[► 196\]](#)

## 3.2.3 Tastaturkürzel anpassen

TwinCAT bietet Ihnen die Möglichkeit, Befehle direkt über ein Tastaturkürzel aufzurufen. Sie können die vordefinierten Tastaturkürzel anpassen oder erweitern.

1. Wählen Sie im Menü **Extras** den Befehl **Anpassen**.
  - ⇒ Der Dialog **Anpassen** öffnet sich. Die Registerkarte **Symbolleisten** ist sichtbar.
2. Wählen Sie die Registerkarte **Befehle**.
3. Klicken Sie auf **Tastatur...**
  - ⇒ Der Dialog **Optionen** öffnet.
4. Wählen Sie den Befehl im Listenfeld **Befehle mit folgendem Inhalt anzeigen:** aus.
5. Geben Sie im Textfeld **Tastenkombination drücken:** ein Tastaturkürzel ein.
6. Klicken Sie auf **Zuweisen**.
  - ⇒ Das Tastaturkürzel wird dem Befehl zugewiesen.
7. Klicken Sie auf **OK**.
  - ⇒ Sie können den Befehl nun über das Tastaturkürzel aufrufen.

**Siehe auch:**

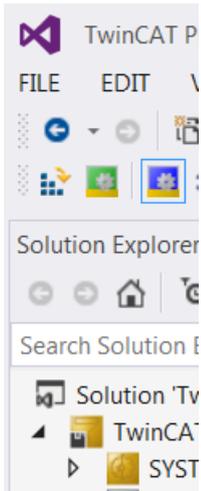
- [Befehl Anpassen \[► 196\]](#)

### 3.2.4 Menü- und Symbolleiste anordnen

Die Menüleiste befindet sich immer am oberen Rand des Hauptfensters der Benutzeroberfläche zwischen Titelleiste und Fensterbereich.

Eine Symbolleiste können Sie unterhalb der Menüleiste fest oder als unabhängiges Fenster irgendwo auf dem Bildschirm positionieren.

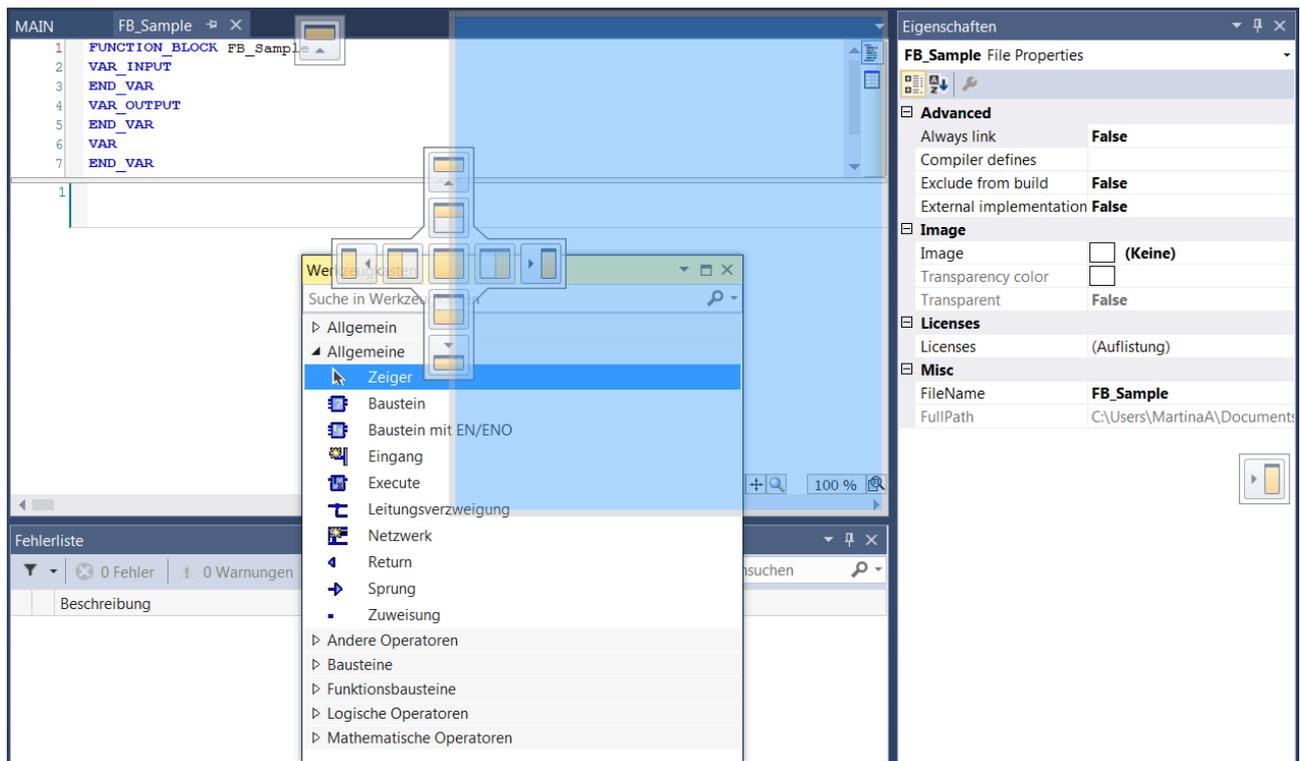
Um eine Leiste zu verschieben, klicken Sie mit dem Mauszeiger auf den gepunkteten Balken am linken Ende der Leiste, halten die Maustaste gedrückt und schieben die Leiste an die gewünschte Position.



### 3.2.5 Ansichten und Fenster anordnen

Sie können die Anordnung der verschiedenen Fenster in TwinCAT einfach an Ihre individuellen Bedürfnisse anpassen.

1. Ziehen Sie das Fenster an der Titelleiste oder an der Registerkarte.
  - ⇒ Pfeilsymbole werden eingeblendet, die die möglichen Positionen kennzeichnen.
2. Ziehen Sie das Fenster mit der Maus auf eines der Pfeilsymbole.
  - ⇒ Die Zielposition wird als blauschattierter Bereich angezeigt.
3. Lassen Sie die linke Maustaste los.
  - ⇒ Das Fenster wird im gewählten Zielbereich eingefügt.



Sie können das Fenster auch außerhalb der TwinCAT-Programmieroberfläche platzieren.

#### Siehe auch:

- [Zwischen Ansichten und Fenstern umschalten \[► 40\]](#)
- [Ansichten aus-/einblenden \[► 40\]](#)
- [Fenstergröße ändern \[► 41\]](#)

### 3.2.6 Zwischen Ansichten und Fenstern umschalten

Sie können direkt zwischen den gerade geöffneten Ansichten und Editorfenstern wechseln.

1. Drücken Sie gleichzeitig die Tasten **[Strg]** und **[Tab]**. Halten Sie die **[Strg]**-Taste weiter gedrückt.
  - ⇒ Eine Übersicht aller aktiven Ansichten und Editoren öffnet sich.
2. Halten Sie die **[Strg]**-Taste gedrückt, während Sie mit den Pfeiltasten das Fenster auswählen.
3. Lassen Sie die **[Strg]**-Taste los.
  - ⇒ Die gewählte Ansicht oder der gewählte Editor wird aktiv geschaltet.

### 3.2.7 Ansichten aus-/einblenden

#### Ausblenden von Ansichten

Wenn Sie eine Ansicht ausblenden, ist diese nur noch als Registerkarte im Rahmen der Benutzeroberfläche vorhanden. Wenn Sie auf die Registerkarte klicken, wird die Ansicht automatisch eingeblendet.

1. Klicken Sie in die Ansicht, die Sie ausblenden wollen.
2. Aktivieren Sie im Menü **Fenster** den Befehl **Automatisch im Hintergrund** oder klicken Sie auf die Schaltfläche  in der oberen rechten Ecke der Ansicht.
  - ⇒ Die Ansicht wird ausgeblendet und nur durch einen kleinen Reiter sichtbar, am seitlichen Rand des Hauptfensters dargestellt.
3. Klicken Sie auf die Registerkarte.
  - ⇒ Die Ansicht wird eingeblendet.

Der Befehl **Ausblenden** im Menü **Fenster** schließt die Ansicht.

### Einblenden von Ansichten

1. Klicken Sie auf den Reiter der ausgeblendeten Ansicht.
2. Wählen Sie im Menü **Fenster** den Befehl **Andocken** oder klicken Sie auf die Schaltfläche  in der oberen rechten Ecke der Ansicht.  
⇒ Die Ansicht wird dauerhaft eingeblendet.

### Siehe auch:

- [Befehl Automatisch in den Hintergrund \[► 200\]](#)
- [Befehl Ausblenden \[► 200\]](#)
- [Befehl Andocken \[► 199\]](#)

## 3.2.8 Fenstergröße ändern

1. Bewegen Sie die Maus über die Trennlinie zwischen zwei Fenstern oder Ansichten.  
⇒ Der Cursor wird zu einem zweiendigen Pfeil.
2. Ziehen Sie die Trennlinie mit der Maus in die gewünschte Position.  
⇒ Während das eine Fenster größer wird, verkleinert sich das andere Fenster.



Losgelöste Fenster können Sie durch Verschieben der Rahmenlinie in der Größe verändern.

---

## 4 TwinCAT Dokumentationen verwenden

### 4.1 Aufruf der TwinCAT 3 Dokumentationen

Ab Visual Studio® 2013 verwendet das TwinCAT 3 Hilfesystem das Beckhoff Information System, sofern noch kein lokales TwinCAT Information System installiert wurde. In diesem Fall verwendet die Kontexthilfe von TwinCAT 3 den Browser des Betriebssystems, um die Artikel online im Beckhoff Information System zu suchen und sie entsprechend der Suche im Webbrowser des Systems darzustellen.

Dafür ist eine Internetverbindung notwendig und die Adresse <https://infosys.beckhoff.com> muss erreichbar sein.

Mit der Installation des TwinCAT 3 Information System wird das Hilfesystem auf die Nutzung der lokalen Hilfe eingestellt.

Die Verwendung der lokalen Hilfe bzw. der Onlinehilfe kann im Hilfe Menü von TwinCAT (Visual Studio® 2013 und neuer) eingestellt werden.

Öffnen Sie im Hilfe Menü das Untermenü „Hilfeeinstellungen festlegen“.

- **Browser:** Sucht online im Beckhoff Information System nach Artikeln zur verwendeten Kontexthilfe.
- **Help Viewer:** Sucht in der installierten lokalen TwinCAT 3 Hilfe nach den entsprechenden Artikeln

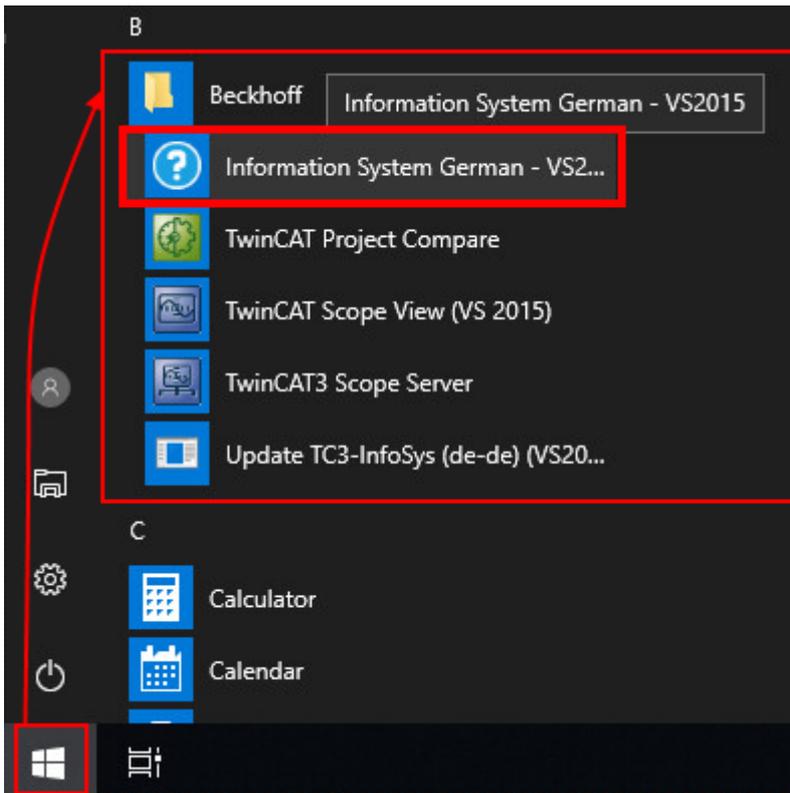
Sie können das lokale Hilfesystem über das Windows-Startmenü oder im TwinCAT 3 Engineering (XAE) über die Kontexthilfe (F1-Hilfe) oder das Menü **Hilfe** aufrufen.

- [TwinCAT 3 Dokumentationen über das Windows-Startmenü aufrufen \[► 44\]](#)
- [TwinCAT 3 Dokumentationen über das Menü Hilfe aufrufen \[► 43\]](#)
- [TwinCAT 3 Dokumentationen über die Kontexthilfe aufrufen \[► 44\]](#)

#### **TwinCAT 3 Dokumentationen über das Windows-Startmenü aufrufen**

- ✓ Sie haben das TwinCAT 3 Information System heruntergeladen und installiert.
- ✓ Während der Installation wurde im Windows-Startmenü eine Verknüpfung zum Help Viewer angelegt. Abhängig von der Systemkonfiguration und den (vor-)eingestellten Installationskomponenten können mehrere Verknüpfungen vorliegen (z. B. für die deutsche und englische Variante des TwinCAT 3 Information Systems).

1. Öffnen Sie im Windows-Startmenü den Ordner **Beckhoff**.

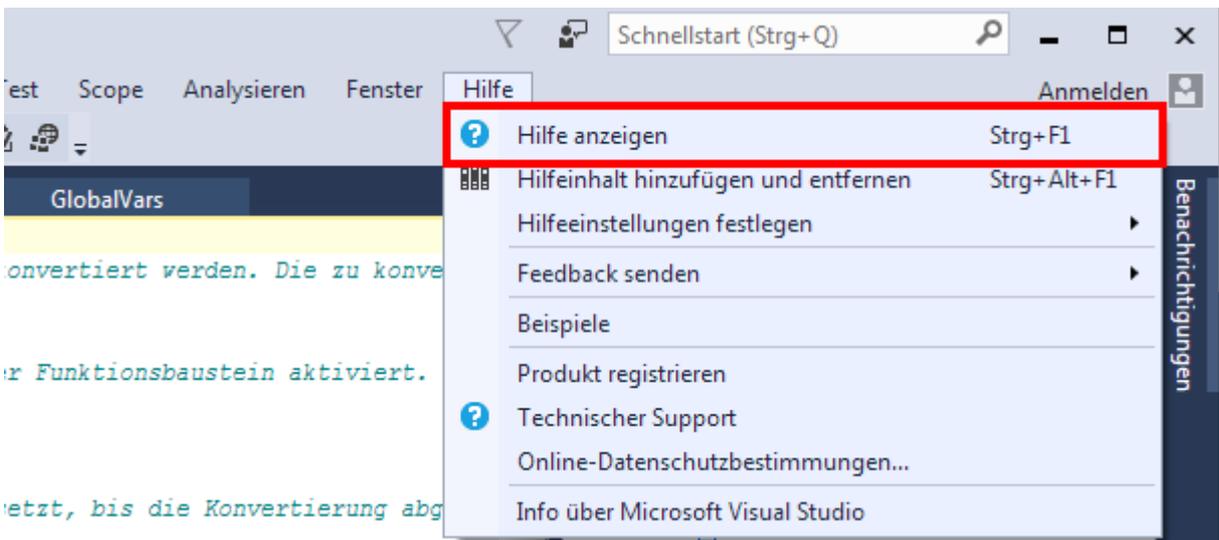


2. Klicken Sie auf **Information System <Sprache> - VS <Version>**.  
 ⇒ Das TwinCAT 3 Information System wird mit dem Help Viewer geöffnet.

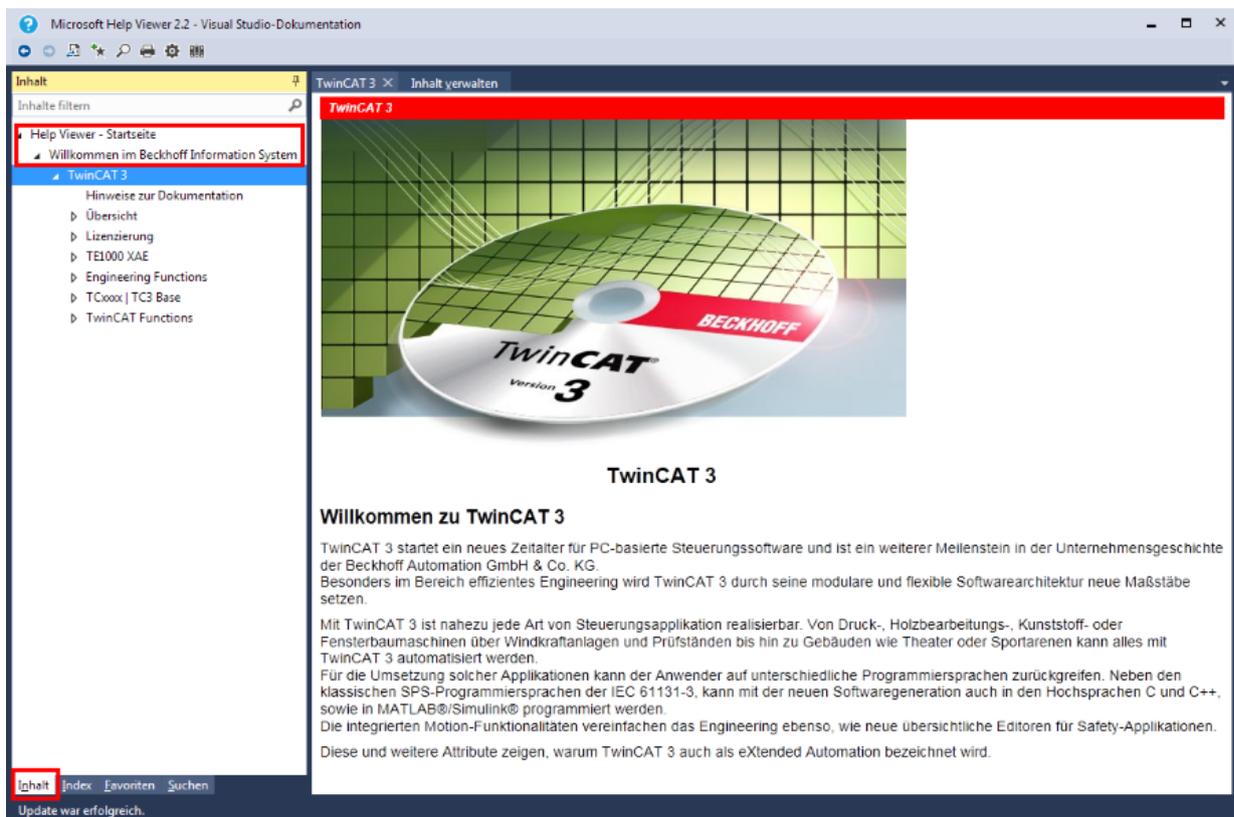
**TwinCAT 3 Dokumentationen über das Menü Hilfe aufrufen**

✓ Das TwinCAT 3 Engineering ist geöffnet.

1. Öffnen Sie das Menü **Hilfe** und wählen Sie den Befehl **Hilfe anzeigen**.



- ⇒ Das TwinCAT 3 Information System wird mit dem Help Viewer des Visual Studio® Hilfesystems geöffnet. Wählen Sie im geöffneten Help Viewer in der Baumansicht **Help Viewer - Startseite > Willkommen im Beckhoff Information System > TwinCAT 3**.



## TwinCAT 3 Dokumentationen über die Kontexthilfe aufrufen

Bei der Kontexthilfe wird die Position des Mauszeigers (bei graphischen Elementen) oder ein ausgewählter Textausdruck innerhalb des Editors vom Hilfesystem ausgewertet, um den passenden verknüpften Hilfetext anzuzeigen.

### ● **Eingeschränkte Hilfefunktion durch unterschiedliche Spracheinstellungen**

**i** Wenn Sie das TwinCAT 3 Information System in einem anderen Spracheinstellungen als das Visual Studio® installiert haben, können Sie die Hilfe nur eingeschränkt nutzen.

### ● **Verschiedene Darstellungen**

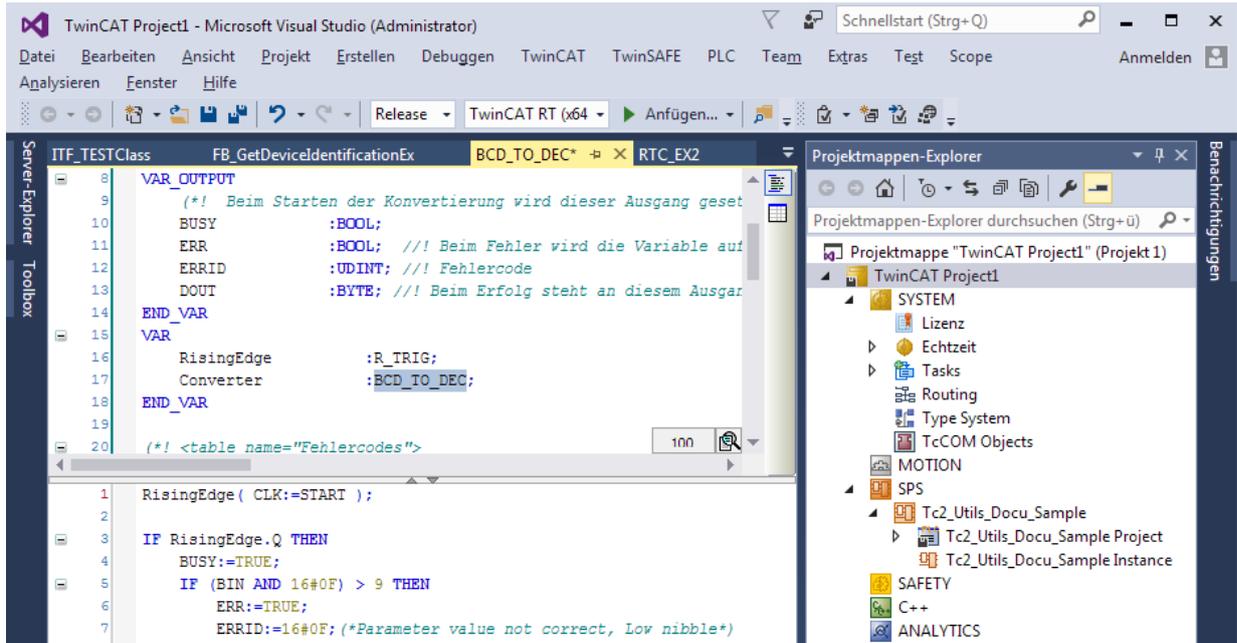
**i** Ab Visual Studio® 2013 ist die Nutzung der Kontext Hilfe mit dem Beckhoff Information System möglich.

Je nach [Einstellung des Hilfesystems \[► 42\]](#) wird das Ergebnis lokal im Help Viewer bzw. im Webbrowser des Systems dargestellt.

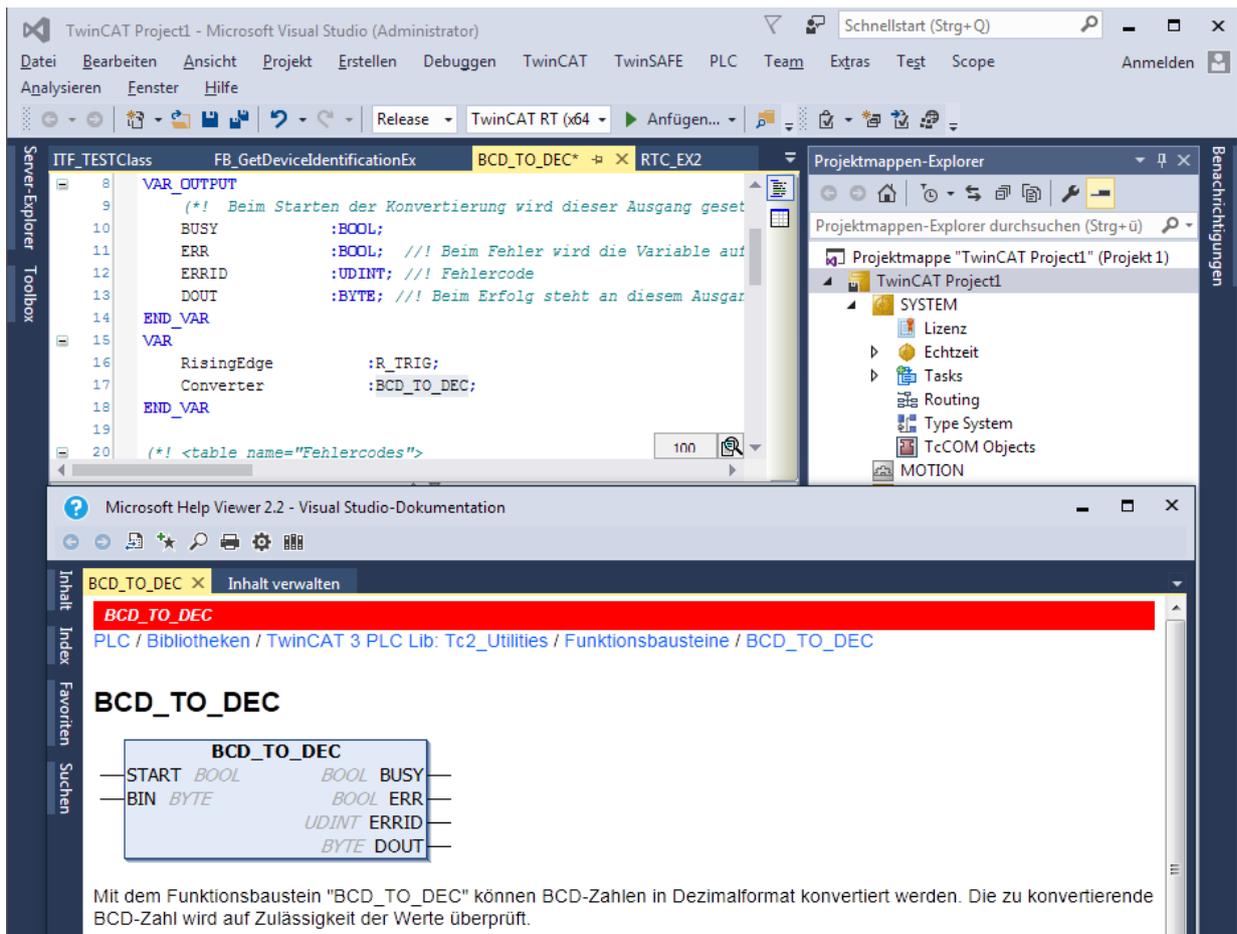
Die folgende Anleitung zeigt die Nutzung der lokalen TwinCAT 3 Hilfe. Die Nutzung der Onlinehilfe erfolgt analog. Lediglich die Darstellung der Artikel erfolgt im Webbrowser.

- ✓ Ein TwinCAT 3 Projekt ist geöffnet.
  - ✓ Im Menü **Hilfe > Hilfeinstellungen festlegen** ist der Befehl **In Help Viewer starten** aktiviert (ab Help Viewer 2.0).
1. Öffnen Sie einen Editor für die unterstützten textbasierten TwinCAT 3 Programmiersprachen, z. B. ST oder C++.

2. Markieren Sie den gewünschten Ausdruck und drücken Sie die Taste [F1].



⇒ Der entsprechende Help Viewer öffnet sich und der zum markierten Ausdruck verknüpfte Artikel aus dem TwinCAT 3 Information System wird angezeigt.



## 4.2 Aktualisierung der TwinCAT 3 Dokumentationen

Alle Sprachvarianten des TwinCAT 3 Information Systems werden wöchentlich aktualisiert. Ein lokal installiertes TwinCAT 3 Information System können Sie manuell über das Visual-Studio® Hilfesystem aktualisieren. Die Vorgehensweise ist dabei von der Visual Studio® Version bzw. dem Hilfesystem abhängig.

### Siehe auch:

- Installation der TwinCAT-3-Dokumentationen

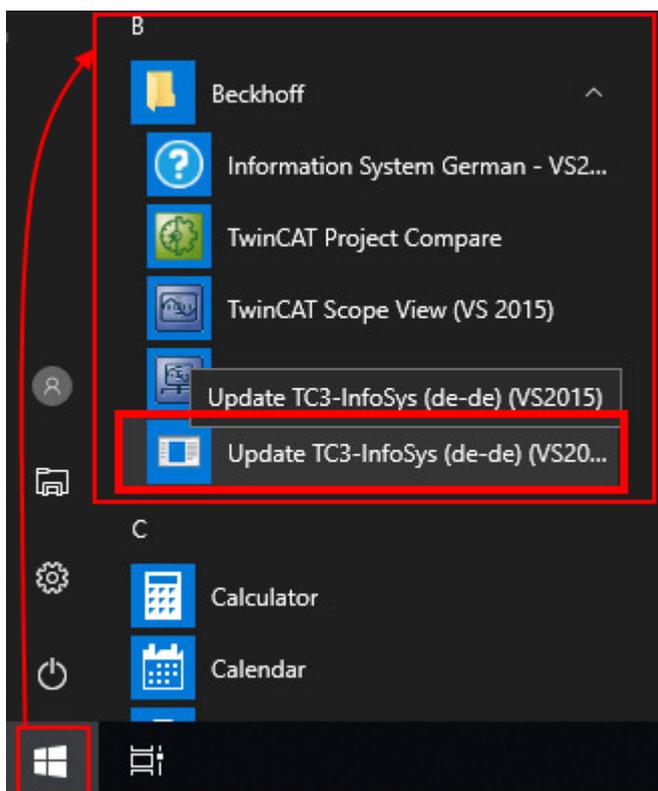
### 4.2.1 Aktualisierung in Visual Studio®



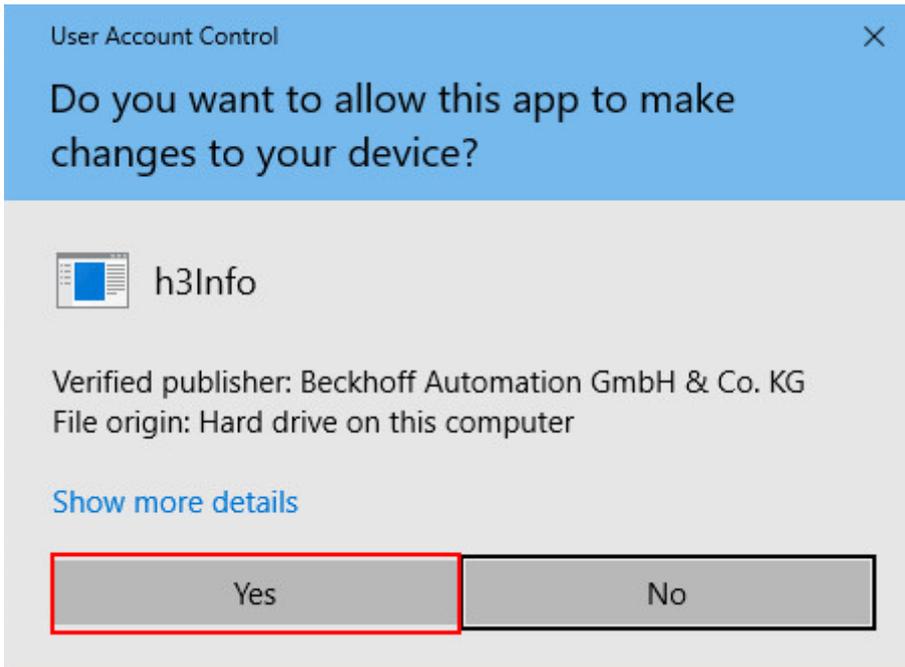
Das im Folgenden beschriebene Verfahren zur Aktualisierung des TwinCAT 3 Information System ist für Visual Studio® 2012 / 2013 / 2015 / 2017 / 2019 gültig.

Bei der Installation des TwinCAT 3 Information System wird im Windows-Startmenü ein Eintrag erstellt, über den das TwinCAT 3 Information System direkt aktualisiert werden kann.

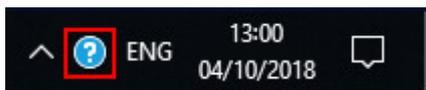
- ✓ Eine Internetverbindung ist vorhanden.
  - ✓ Sie haben Administratorrechte.
1. Öffnen Sie im Windows-Startmenü den Ordner **Beckhoff**.
  2. Klicken Sie auf **Update TC3-InfoSys (<Sprache>) (VS<Version>)**.



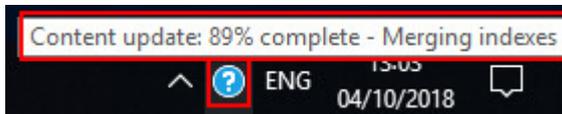
- Bestätigen Sie in dem sich öffnenden Dialog, dass Änderungen am System vorgenommen werden dürfen.



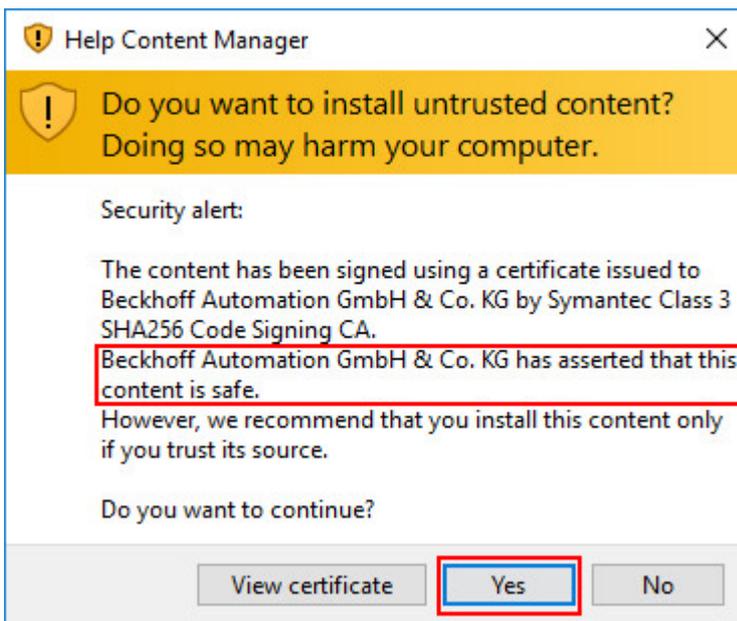
- ⇒ Das Hilfesystem prüft, ob eine Aktualisierung für das TwinCAT 3 Information System vorliegt. Dabei wird ein Symbol des Hilfesystems in der Menüleiste von Windows angezeigt.



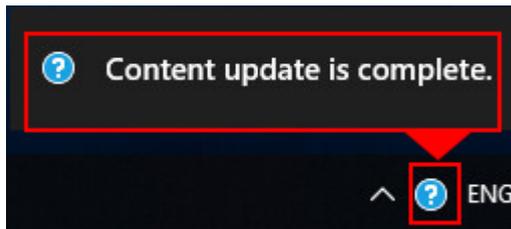
Wenn Sie Mauszeiger über das Symbol des Hilfesystems bewegen, wird der Aktualisierungsfortschritt angezeigt.



- Bestätigen Sie, dass das TwinCAT 3 Information System aktualisiert werden darf.



⇒ In der Menüleiste von Windows wird der Aktualisierungsstatus angezeigt:

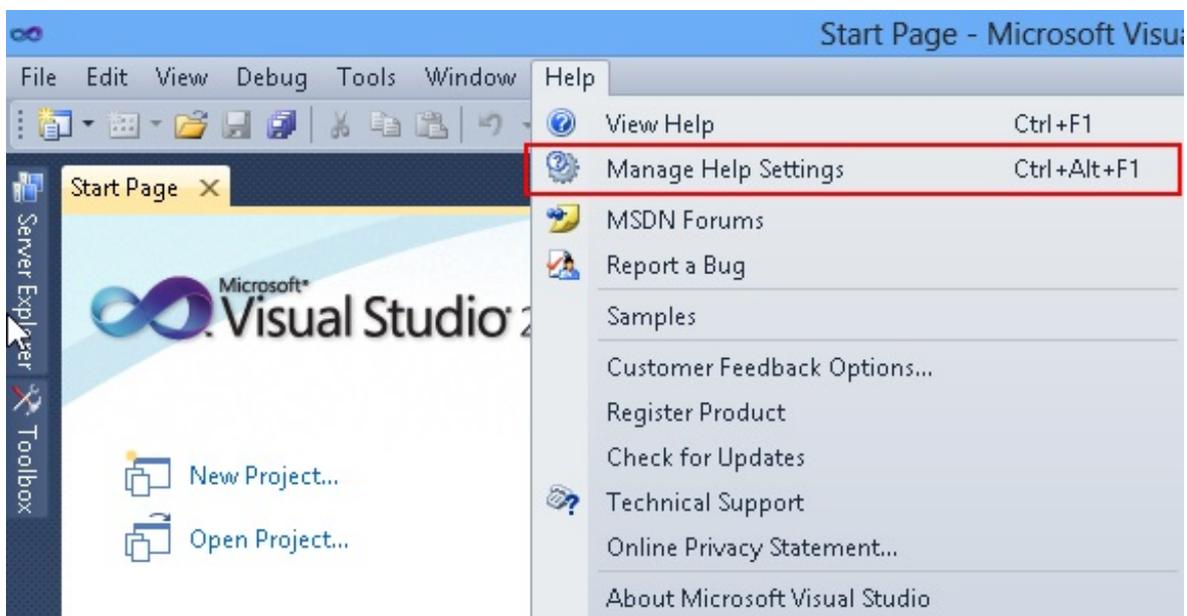


## 4.2.2 Aktualisierung in Visual Studio® 2010

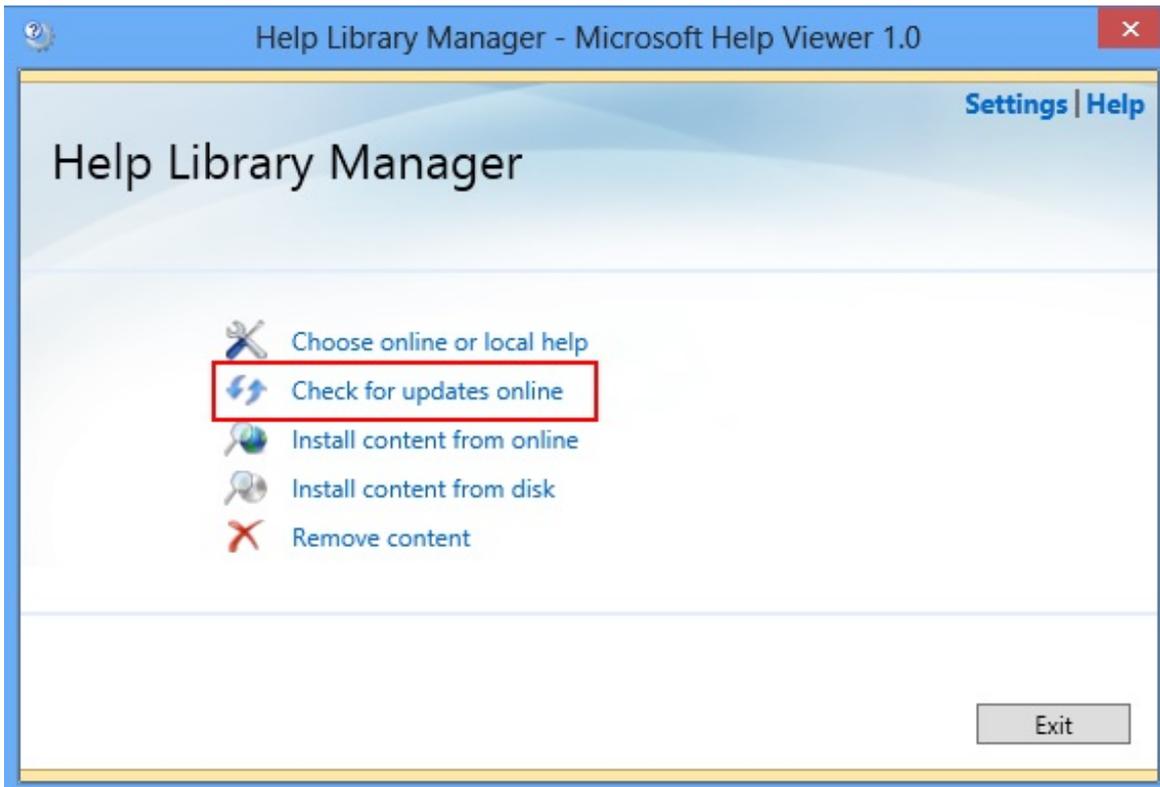
Wenn eine Internetverbindung vorhanden ist, kann das installierte TwinCAT 3 Information System direkt über das Visual Studio®-Hilfesystem im TwinCAT 3 Engineering aktualisiert werden.

✓ Eine Internetverbindung ist vorhanden.

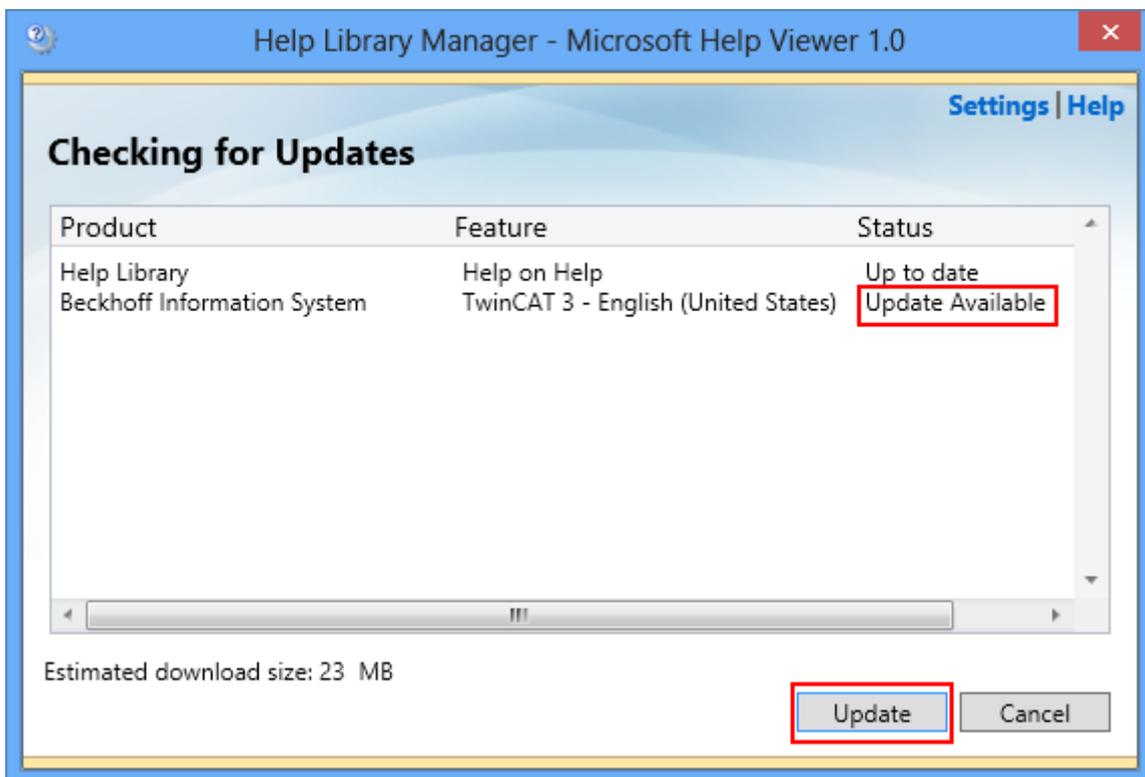
1. Starten Sie das TwinCAT 3 Engineering oder Visual Studio® 2010.
2. Wählen Sie im Menü **Help** den Befehl **Manage Help Setting**.



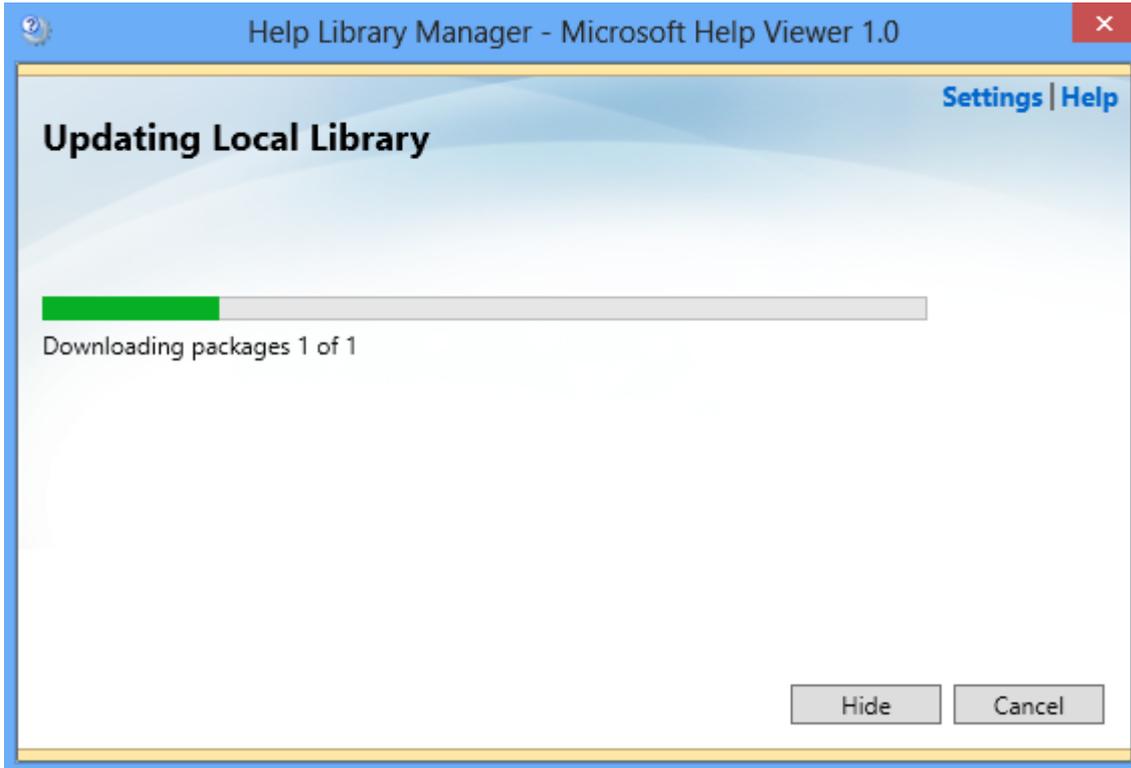
3. Klicken Sie in dem sich öffnenden Dialog des Hilfesystems auf **Check for updates online**.



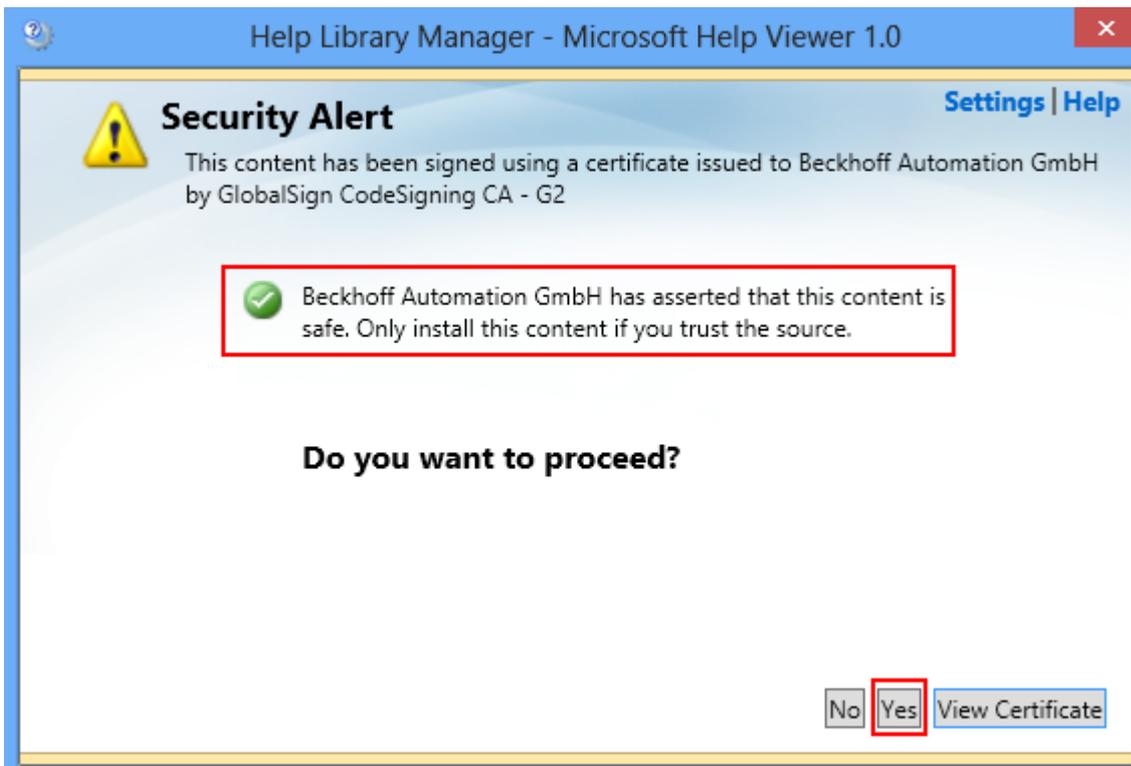
⇒ Das Hilfesystem prüft, ob eine Aktualisierung für das TwinCAT 3 Information System vorliegt. Verfügbare Aktualisierungen für installierte Dokumentationen werden angezeigt.



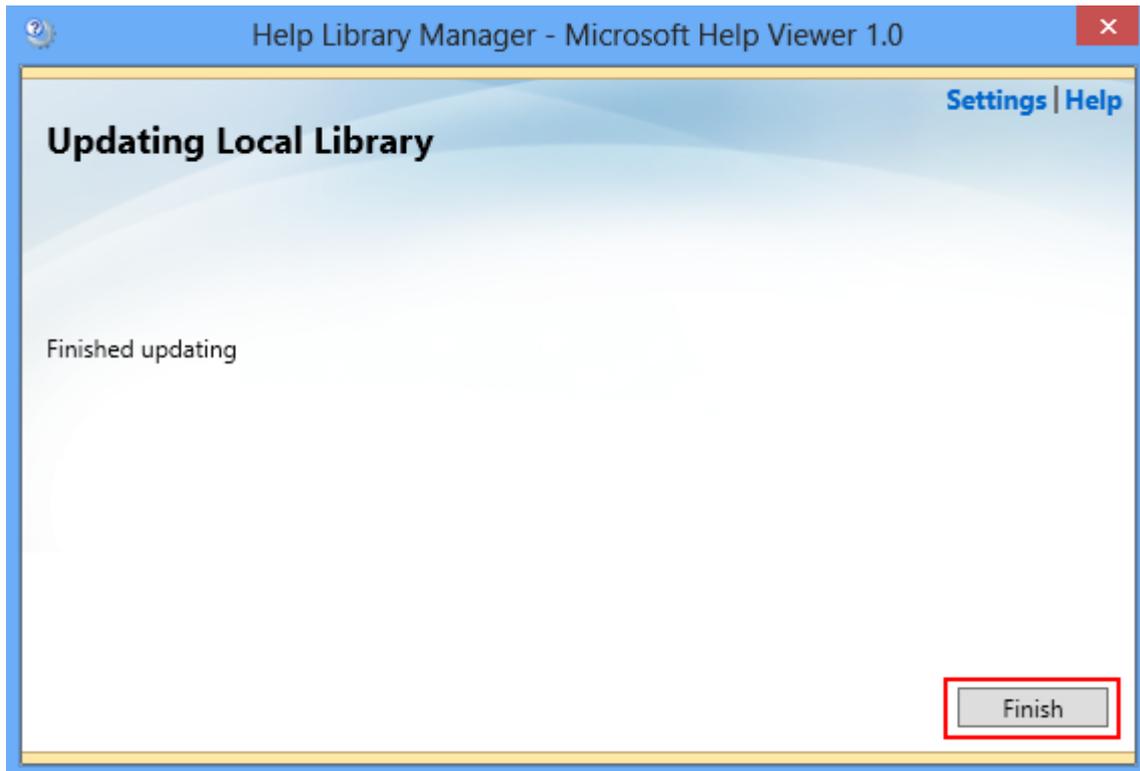
4. Klicken Sie auf **Update**, um die verfügbaren Aktualisierungen herunterzuladen.



5. Bestätigen Sie die Installation des Aktualisierungspaketes mit **Yes**.



⇒ Schließen Sie den Dialog nach erfolgreicher Aktualisierung mit **Finish**.



## 5 Referenz Benutzeroberfläche

Standardmäßig sind die wichtigsten Befehle in der Benutzeroberfläche von TwinCAT verfügbar. Wenn Sie die Menükonfiguration individuell anpassen möchten, wählen Sie den Befehl **Anpassen** im Menü **Extras**.

**Siehe auch:**

- [Benutzeroberfläche anpassen](#)

### 5.1 Datei

#### 5.1.1 Archivierungsmöglichkeiten

Um ein TwinCAT-Projekt z.B. für Archivierungszwecke oder für die Weitergabe an Kollegen zu speichern, bietet die TwinCAT-Entwicklungsumgebung drei verschiedene Archivierungsdateitypen an: tnzip, tszip und tpzip.

Welcher Archivierungsdateityp verwendet werden sollte, hängt davon ab, welche Projekte in dem Archivordner gespeichert werden sollen.

Archivierungsdateityp	Fokus	Inhalt	Befehle
tnzip	<a href="#">Projektmappe</a> [▶ 52]	Der Archivordner *.tnzip enthält alle in der Projektmappe enthaltenen TwinCAT-Projekttypen. Dies können TwinCAT-, TwinCAT HMI-, Scope- und Connectivity-Projekte sein.	<a href="#">Speichern</a> [▶ 52] <a href="#">Öffnen</a> [▶ 53]
tszip	<a href="#">TwinCAT-Projekt</a> [▶ 53]	Der Archivordner *.tszip enthält das TwinCAT-Projekt, welches archiviert wird.	<a href="#">Speichern</a> [▶ 52] <a href="#">Öffnen</a> [▶ 53]
tpzip	<a href="#">SPS-Projekt</a> [▶ 54]	Der Archivordner *.tpzip enthält das SPS-Projekt, welches archiviert wird.	<a href="#">Speichern</a> [▶ 52] <a href="#">Öffnen</a> [▶ 55]

**Sehen Sie dazu auch**

- ☰ [Befehl Sichern <TwinCAT-Projektname> als Archiv...](#) [▶ 269]
- ☰ [Befehl Sichern <SPS-Projektname> als Archiv...](#) [▶ 54]

##### 5.1.1.1 Projektmappe

- TwinCAT-Projektmappenarchiv \*.tnzip erstellen: [Befehl Save <Projektmappenname> as Archive...](#) [▶ 52]
- TwinCAT-Projektmappenarchiv \*.tnzip öffnen: [Befehl Open Solution from Archive](#) [▶ 53]

###### 5.1.1.1.1 Befehl Save <Projektmappenname> as Archive...

**Funktion:** Der Befehl öffnet den Standarddialog zum Speichern einer Datei als Archiv. Die Projektmappe kann unter dem gewünschten Speicherpfad als \*.tnzip-Archiv abgelegt werden.

**Aufruf:** Menü **Datei**, Kontextmenü

**Voraussetzung:** Die Projektmappe ist im **Projektmappen-Explorer** ausgewählt.

<b>Inhalt vom *.tnzip</b>	Der Archivordner *.tnzip enthält alle in der Projektmappe enthaltenen TwinCAT-Projekttypen. Dies können TwinCAT-, TwinCAT HMI-, Scope- und Connectivity-Projekte sein.
<b>Befehl zum Öffnen</b>	Ein tnzip-Archiv kann über den folgenden Befehl wieder geöffnet werden: Befehl <a href="#">Open Solution from Archive</a> [► 53].
<b>Hinweis zu SPS-Projekten</b>	Falls die Projektmappe ein oder mehrere SPS-Projekte enthält, sind die Dateien und Ordner, die bezüglich dieser SPS-Projekte in dem Archivordner gespeichert werden, abhängig von den SPS-Projekteinstellungen des jeweiligen SPS-Projekts. <a href="#">Registerkarte Settings</a> [► 122]

### 5.1.1.1.2 Befehl Open Solution from Archive

**Funktion:** Der Befehl extrahiert ein TwinCAT-Projektmappenarchiv \*.tnzip.

**Aufruf:** Menü **Datei > Öffnen**

Nachdem Sie den Befehl ausgeführt haben, öffnet sich der Dialog **Öffnen**. Wählen Sie aus dem Dateisystem die Archivdatei aus und bestätigen Sie den Dialog. Anschließend öffnet sich der Dialog **Wähle Verzeichnis für neue Arbeitsmappe**. Wählen Sie einen Ordner aus, in dem Sie die extrahierten Projektmappendateien speichern wollen.

<b>Inhalt vom *.tnzip</b>	Der Archivordner *.tnzip enthält alle in der Projektmappe enthaltenen TwinCAT-Projekttypen. Dies können TwinCAT-, TwinCAT HMI-, Scope- und Connectivity-Projekte sein.
<b>Befehl zum Erstellen</b>	Ein tnzip-Archiv kann über den folgenden Befehl erstellt werden: Befehl <a href="#">Save &lt;Projektmappenname&gt; as Archive...</a> [► 52]
<b>Hinweis zu SPS-Projekten</b>	Falls die Projektmappe ein oder mehrere SPS-Projekte enthält, sind die Dateien und Ordner, die bezüglich dieser SPS-Projekte in dem Archivordner gespeichert werden, abhängig von den SPS-Projekteinstellungen des jeweiligen SPS-Projekts. <a href="#">Registerkarte Settings</a> [► 122]

### 5.1.1.2 TwinCAT-Projekt

- TwinCAT-Projektarchiv \*.tszip erstellen: [Befehl Sichern <TwinCAT-Projektname> als Archiv...](#) [► 269]
- TwinCAT-Projektarchiv \*.tszip öffnen: [Befehl Projekt/Projektmappe \(Projekt/Projektmappe öffnen\)](#) [► 53]

#### 5.1.1.2.1 Befehl Projekt/Projektmappe (Projekt/Projektmappe öffnen)

Symbol: 

Tastaturkürzel: **[Strg] + [Umschalt] + [O]**

**Funktion:** Der Befehl öffnet den Standarddialog zum Öffnen einer Datei. Hier können Sie das Dateisystem nach einer TwinCAT-Projektdatei durchsuchen und im Entwicklungssystem öffnen.

**Aufruf:** Menü **Datei > Öffnen**

**Dialog Projekt öffnen**

<b>Dateityp</b>	Auswahlliste zum Filtern des Dateityps <ul style="list-style-type: none"> <li>• Dateien aller unterstützten Formate können geöffnet werden.</li> </ul>
<b>Optionen</b>	<ul style="list-style-type: none"> <li>• Hinzufügen (Verwenden Sie diese Option nur, um einer Projektmappe z. B. noch ein Measurement-Projekt hinzuzufügen. Verwenden Sie diese Option nicht, um einer Projektmappe mehrere TwinCAT-Projekte hinzuzufügen.)</li> <li>• Mappe schließen</li> </ul>
<b>Öffnen</b>	TwinCAT öffnet die gewählte Projektdatei. Wenn erforderlich, wird sie zuvor konvertiert.

**TwinCAT-Projektarchiv \*.tszip**

<b>Inhalt vom *.tszip</b>	Der Archivordner *.tszip enthält das TwinCAT-Projekt, welches archiviert wird.
<b>Befehl zum Erstellen</b>	Ein tszip-Archiv kann über den folgenden Befehl erstellt werden: Befehl Sichern <TwinCAT-Projektname> als Archiv... [ <a href="#">► 269</a> ]
<b>Hinweis zu SPS-Projekten</b>	Falls das TwinCAT-Projekt ein oder mehrere SPS-Projekte enthält, sind die Dateien und Ordner, die bezüglich dieser SPS-Projekte in dem Archivordner gespeichert werden, abhängig von den SPS-Projekteinstellungen des jeweiligen SPS-Projekts.  Registerkarte Settings [ <a href="#">► 122</a> ]

**Siehe auch:**

- Dokumentation PLC: Ihr erstes TwinCAT-3-SPS-Projekt
- Dokumentation PLC: Standardprojekt anlegen

**5.1.1.3 SPS-Projekt**

- SPS-Projektarchiv \*.tpzip erstellen: [Befehl Sichern <SPS-Projektname> als Archiv... \[\[► 54\]\(#\)\]](#)
- SPS-Projektarchiv \*.tpzip öffnen: [Befehl Vorhandenes Element hinzufügen \(Projekt\) \[\[► 55\]\(#\)\]](#)

**5.1.1.3.1 Befehl Sichern <SPS-Projektname> als Archiv...**

**Funktion:** Der Befehl öffnet den Standarddialog zum Speichern einer Datei als Archiv. Das SPS-Projekt kann unter dem gewünschten Speicherpfad als \*.tpzip-Archiv abgelegt werden.

**Aufruf:** Menü **Datei**, Kontextmenü

**Voraussetzung:** Das TwinCAT-SPS-Projekt (<SPS-Projektname>) ist im **Projektmappen-Explorer** ausgewählt.

Die Dateien und Ordner in dem Archivordner sind abhängig von den SPS-Projekteinstellungen.

<b>Inhalt vom *.tpzip</b>	Der Archivordner *.tpzip enthält das SPS-Projekt, welches archiviert wird.
<b>Befehl zum Öffnen</b>	Ein tpzip-Archiv kann über den folgenden Befehl wieder geöffnet werden: Befehl Vorhandenes Element hinzufügen (Projekt) [ <a href="#">► 55</a> ]
<b>Hinweis zu SPS-Projekten</b>	Die Dateien und Ordner, die bezüglich des SPS-Projekts in dem Archivordner gespeichert werden, sind abhängig von den SPS-Projekteinstellungen dieses SPS-Projekts.  Registerkarte Settings [ <a href="#">► 122</a> ]

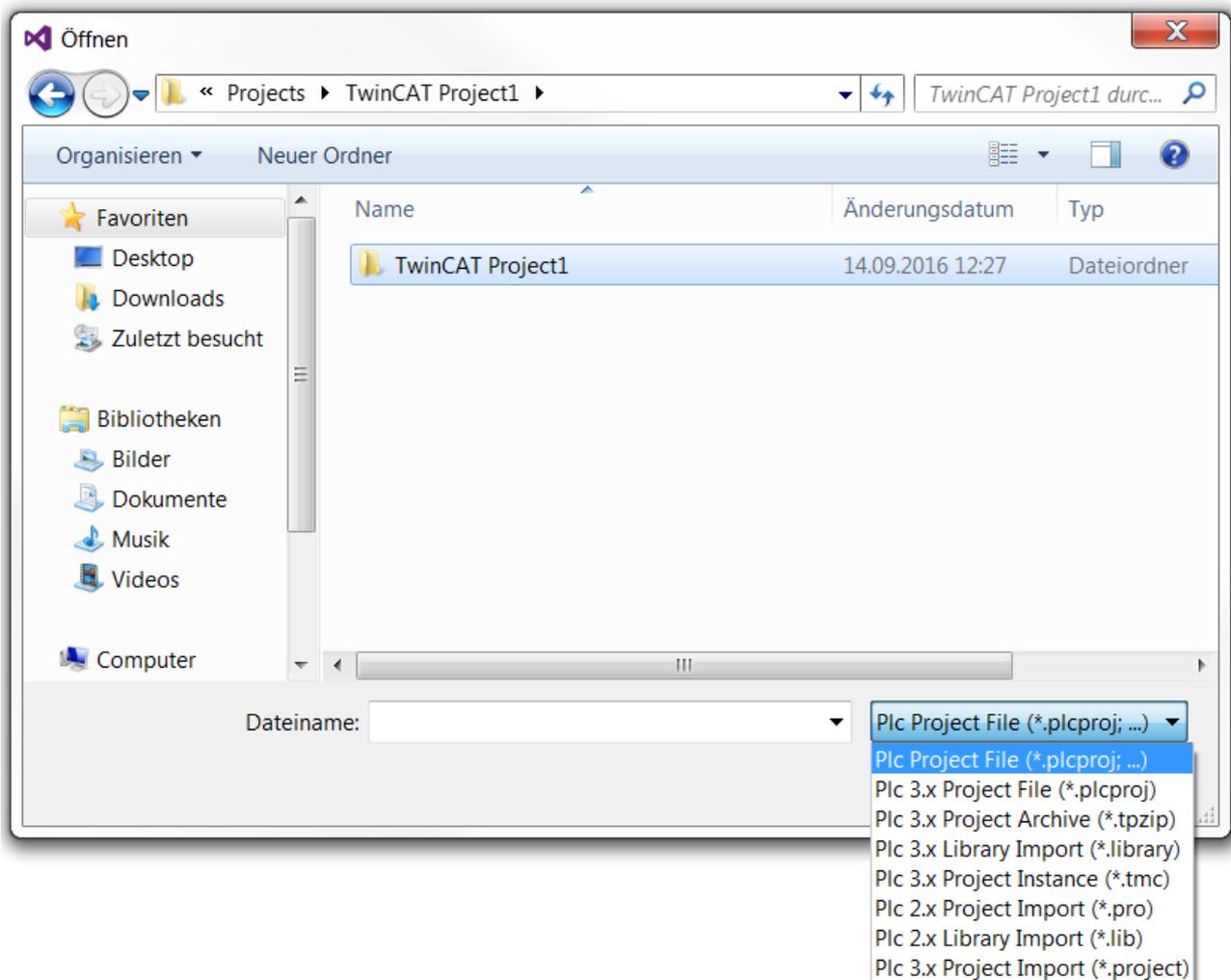
### 5.1.1.3.2 Befehl Vorhandenes Element hinzufügen (Projekt)

Symbol: 

**Funktion:** Der Befehl öffnet den Standard-Browserdialog, über den eine SPS-Projektdatei gesucht und im Programmiersystem geöffnet werden kann. Wenn ein entsprechender Konverter installiert ist, können auch Projekte in einem anderen Format geöffnet werden.

**Aufruf:** Menü **Projekt** oder Kontextmenü SPS-Objekt im **Projektmappen-Explorer**

**Voraussetzung:** Der SPS-Knoten ist im TwinCAT-Projektbaum selektiert.



<b>Dateityp</b>	<p>Standardmäßig können Sie den Filter auf einen der folgenden Dateitypen setzen:</p> <ul style="list-style-type: none"> <li>• PLC 3.x Projektdatei (*.PLCproject): TwinCAT-3-PLC-Projekte mit der Erweiterung „PLCproject“</li> <li>• PLC 3.x Projektarchiv (*.tpzip): TwinCAT-3-PLC-Projektarchive mit der Erweiterung „tpzip“             <ul style="list-style-type: none"> <li>◦ Siehe auch: <a href="#">Befehl Sichern &lt;SPS-Projektname&gt; als Archiv... [► 54]</a></li> </ul> </li> <li>• PLC 3.x Bibliotheksimport (*.library): TwinCAT-3-PLC-Bibliotheken mit der Erweiterung „library“,</li> <li>• PLC 2.x Projektdatei (*.pro): TwinCAT-2-PLC-Projekte mit der Erweiterung „pro“</li> <li>• PLC 2.x Bibliotheksimport (*.lib): TwinCAT-2-PLC-Bibliotheken mit der Erweiterung „lib“</li> <li>• PLC 3.x Projektimport (*.PLCproject): PLC-Projekte mit der Erweiterung „project“</li> </ul>
<b>Öffnen</b>	Die gewählte Projektdatei wird geöffnet bzw. konvertiert und dann geöffnet.

**SPS-Projektarchiv \*.tpzip**

<b>Inhalt vom *.tpzip</b>	Der Archivordner *.tpzip enthält das SPS-Projekt, welches archiviert wird.
<b>Befehl zum Erstellen</b>	Ein tpzip-Archiv kann über den folgenden Befehl erstellt werden: Befehl Sichern <SPS-Projektname> als Archiv... [► 54]
<b>Hinweis zu SPS-Projekten</b>	Die Dateien und Ordner, die bezüglich des SPS-Projekts in dem Archivordner gespeichert werden, sind abhängig von den SPS-Projekteinstellungen dieses SPS-Projekts. Registerkarte Settings [► 122]

**Mögliche Szenarien beim Öffnen eines SPS-Projekts**

Folgende Szenarien sind beim Öffnen eines Projekts möglich:

1. [Ein anderes Projekt ist noch geöffnet.](#) [► 56]
2. [Das Projekt wurde mit einer älteren TwinCAT-3-Version gespeichert.](#) [► 56]
3. [Das Projekt wurde nicht mit TwinCAT 3 gespeichert.](#) [► 56]
4. [Das Projekt wurde nicht regulär beendet und „Automatisch Speichern“ war aktiviert.](#) [► 58]
5. [Das Projekt ist schreibgeschützt.](#) [► 58]
6. [Es handelt sich um eine Bibliothek, die in einem Bibliotheks-Repository installiert ist und aus diesem aufgerufen wird.](#) [► 58]

**1. Ein anderes Projekt ist noch geöffnet.**

Sie werden gefragt, ob das andere Projekt gespeichert und geschlossen werden soll.

**2. Das Projekt wurde mit einer älteren TwinCAT-3-Version gespeichert.**

Wenn sich das Speicherformat unterscheidet, weil das geöffnete Projekt mit einer älteren Version von TwinCAT 3 gespeichert wurde, gibt es zwei Fälle:

- Wenn das Projekt nicht im Speicherformat des aktuell verwendeten Programmiersystems speicherbar ist, müssen Sie es aktualisieren, um weiter am Projekt arbeiten zu können. Der an dieser Stelle auftretende Ausdruck **Die durchgeführten Änderungen...** bezieht sich auf interne Aktionen verschiedener Komponenten während des Ladens des Projekts.
- Wenn das Projekt im bisherigen Speicherformat weiterhin speicherbar ist, können Sie entscheiden, ob das Speicherformat beibehalten oder aktualisiert werden soll. Wenn das Speicherformat beibehalten werden soll, muss mit möglichen Datenverlusten gerechnet werden. Wenn das Speicherformat aktualisiert werden soll, kann das Projekt nicht mehr mit der alten Version des Programmiersystems geöffnet werden.

Neben dem Speicherformat können sich auch die Versionen der explizit eingefügten Bibliotheken, des Visualisierungsprofils und die Compiler-Version des öffnenden Projekts von denen unterscheiden, die mit dem aktuellen Programmiersystem installiert wurden.

Wenn auf dem aktuellen Programmiersystem neuere Versionen installiert sind, öffnet sich automatisch der Dialog **Projektumgebung**, in dem Sie die Versionen aktualisieren können. Wenn an dieser Stelle noch keine Aktualisierung vorgenommen wird, kann dies jederzeit im Dialog **Optionen > Projektumgebung** nachgeholt werden.

---

**● Compiler-Version beachten**

**I** Wenn ein Projekt geöffnet wird, das mit einer älteren Version des Programmiersystems erstellt wurde und für das in den Projekteinstellungen die neueste Compiler-Version eingestellt ist, während im neuen Programmiersystem die Projektumgebungseinstellung für die Compiler-Version **Nicht aktualisieren** ist, dann wird die zuletzt im alten Projekt verwendete Compiler-Version weiter verwendet (nicht die „Aktuelle“ in der neuen Umgebung).

---

**3. Das Projekt wurde nicht mit TwinCAT 3 gespeichert.**

**Fall 1)**

Wenn Sie beim Auswählen des zu öffnenden Projekts den Dateifilter gezielt setzen und ein entsprechender Konverter verfügbar ist, wird der Konverter automatisch verwendet und das Projekt in das aktuelle Format gebracht. Die Konvertierung läuft konverterspezifisch ab. In der Regel werden Sie aufgefordert, die Behandlung von eingebundenen Bibliotheken oder Geräte-Referenzen zu definieren.

**● TwinCAT-3-Konverter**

**i** Die Anpassung eines TwinCAT-PLC-Control-Projekts an die TwinCAT-3-Syntax kann beim Import nur gelingen, wenn der Konverter das Projekt fehlerfrei übersetzen kann.

Wenn Sie beim Auswählen des zu öffnenden Projekts den Dateityp **Alle Dateien** eingestellt haben, ist kein Konverter aktiviert und es öffnet sich der Dialog **Projekt konvertieren**. In dem Dialog müssen Sie die Konvertierung des Projekts durch Auswählen einer der Optionen explizit anstoßen.

- **In das aktuelle Format konvertieren:** Wählen Sie aus der Auswahlliste den Konverter, der verwendet werden soll (Anwendung zum Konvertieren). Nach dem Konvertieren kann das Projekt nicht mehr in der alten Version geöffnet werden.
- **Ein neues Projekt erzeugen und ein spezielles Gerät hinzufügen:** (Noch nicht implementiert)

**● TwinCAT 2.x PLC-Control-Projektoptionen**

**i** Der in den TwinCAT 2.x PLC-Control-Projektoptionen eingestellte Projektverzeichnispfad sowie die Projektinformationen werden in den Dialog **Projektinformationen** übernommen.

**Fall 2)**

Wenn im Projekt Bibliotheken eingebunden sind, für die noch kein „Konvertierungs-Mapping“ in den Bibliotheks-Optionen gespeichert ist, erscheint der Dialog **Konvertierung einer Bibliotheksreferenz**, in dem Sie definieren, wie diese Referenz konvertiert werden sollen:

- **Die Bibliothek ebenfalls konvertieren und installieren:** Wenn Sie diese Option aktivieren, wird die eingebundene Bibliothek in das neue Format übergeführt und bleibt im Projekt referenziert. Sie wird automatisch im Bibliotheks-Repository in der Kategorie **Sonstige** installiert und weiterhin verwendet. Wenn die Bibliothek nicht die für eine Installation nötigen Projektinformationen (Titel, Version) mitbringt, werden Sie aufgefordert, diese im Dialog **Projektinformationen eingeben** nachzutragen.
- **Die folgende bereits installierte Bibliothek verwenden:** Wenn Sie die Optionen aktivieren, wird die eingebundene Bibliothek durch eine andere ersetzt, die bereits auf dem lokalen System installiert ist. Mit der Schaltfläche **Auswählen** öffnen Sie den Dialog **Auswählen...** Hier können Sie die gewünschte Version einer der installierten Bibliothek auswählen. Dies entspricht der Konfiguration des Versions-Handlings im Dialog **Bibliothekseigenschaften**. Ein Sternchen („\*“) bedeutet, dass immer die neueste Version der Bibliothek, die auf dem System verfügbar ist, im Projekt verwendet wird. Die Liste der verfügbaren Bibliotheken ist genauso strukturiert, wie im Dialog **Bibliotheks-Repository**. Sie können die Auflistung nach Firma und Kategorie sortieren.
- **Die Bibliothek ignorieren. Die Referenz wird im konvertierten Projekt nicht erscheinen:** Wenn Sie diese Option aktivieren, wird die Bibliotheksreferenz entfernt. Die Bibliothek ist dann nicht mehr im konvertierten Projekt eingebunden.
- **Dieses Mapping auch zukünftig verwenden, wenn diese Bibliothek auftritt:** Wenn Sie diese Option aktivieren, werden die hier im Dialog vorgenommen Einstellungen auch für künftige Projekt-Konvertierungen angewendet, sobald die betreffende Bibliothek referenziert ist.

Im konvertierten Projekt sind die Bibliotheksreferenzen im globalen Bibliotheksverwalter im Projektmappen-Explorer definiert. Nach Abschluss der Konvertierung der Bibliotheks-Referenzen wird die Projekt-Konvertierung wie oben beschrieben mit Dialog **Projekt öffnen** weitergeführt.

Generelle Informationen zur Bibliotheksverwaltung finden Sie in der PLC-Dokumentation im Abschnitt „Bibliotheken verwenden“.

**Fall 3)**

Wenn Sie ein TwinCAT 2.x PLC-Control-Projekt öffnen, das ein Gerät (Zielsystem) referenziert, für das noch kein „Konvertierungs-Mapping“ in den TwinCAT 2.x PLC-Control-Konverter-Optionen definiert ist, öffnet der Dialog **Gerätekonvertierung**, in dem Sie festlegen können, ob und wie die alten Geräte-Referenzen durch aktuellere ersetzt werden sollen. Das ursprünglich verwendete Gerät wird angezeigt. Wählen Sie eine der folgenden Optionen:

- **Das folgende bereits installierte Gerät verwenden:** Öffnen Sie mit der Schaltfläche **Auswählen** den Dialog **Zielsystem auswählen**, in dem Sie eines der aktuell auf dem System installierten Geräte auswählen können. Dieses Gerät wird dann anstelle des alten im **Projektmappen-Explorer** des konvertierten Projekts eingefügt. Aktivieren Sie Option **Wählen Sie ein Zielsystem aus...**, um eines der aufgelisteten Geräte auswählen zu können. Die Liste der verfügbaren Geräte ist genauso strukturiert wie im Dialog **Device-Repository**. Sie können die Auflistung nach Hersteller und Kategorie sortieren.
- **Das Gerät ignorieren. Alle applikationsspezifischen Objekte werden nicht verfügbar sein:** Wenn Sie diese Option aktivieren, wird im **Projektmappen-Explorer** des neuen Projekts kein Eintrag für das Gerät angelegt, d.h. das Gerät wird bei der Konvertierung ignoriert und auch applikationsspezifische Objekte wie z.B. die Taskkonfiguration werden nicht übernommen.
- **Diese Zuordnung für die Zukunft speichern:** Wenn Sie diese Option aktivieren, werden alle Einstellungen des Dialogs, d.h. das dargestellte „Konvertierungs-Mapping“ für das Gerät, in den TwinCAT 2.x PLC-Control-Konverter-Optionen gespeichert und für künftige Konvertierungen angewendet.

#### 4. Das Projekt wurde nicht regulär beendet und Automatisch Speichern war aktiviert.

Wenn die Funktion **Automatisch Speichern** in den Optionen **Laden und Speichern** aktiviert war und TwinCAT 3 PLC, nach der letzten Änderung des Projekts ohne Speichern, nicht regulär beendet worden war, öffnet der Dialog **Auto Save Backup** zur Handhabung der Sicherungskopie.

#### 5. Das Projekt ist schreibgeschützt.

Ist das Projekt, das geöffnet werden soll, schreibgeschützt, so werden Sie gefragt, ob Sie das Projekt in schreibgeschütztem Modus öffnen oder ob den Schreibschutz aufheben wollen.

#### 6. Es handelt sich um eine Bibliothek, die in einem Bibliotheks-Repository installiert ist und aus diesem aufgerufen wird.

Wenn Sie ein Bibliotheksprojekt öffnen möchten, das in einem Bibliotheks-Repository installiert ist, wird eine Fehlermeldung ausgegeben. Ein Bibliotheksprojekt können Sie über diesen Pfad nicht öffnen. Nach Schließen des Dialogs mit **OK** erscheint der Projektname in der Titelzeile der Benutzeroberfläche. Ein Sternchen („\*“) hinter dem Namen bedeutet, dass das Projekt seit dem letzten Speichern verändert wurde.

#### Siehe auch:

- Dokumentation PLC: TwinCAT-3-SPS-Projekt öffnen
- Dokumentation PLC: TwinCAT-2-SPS-Projekt öffnen

## 5.1.2 Befehl Projekt... (Neues TwinCAT-Projekt anlegen)

Symbol: 

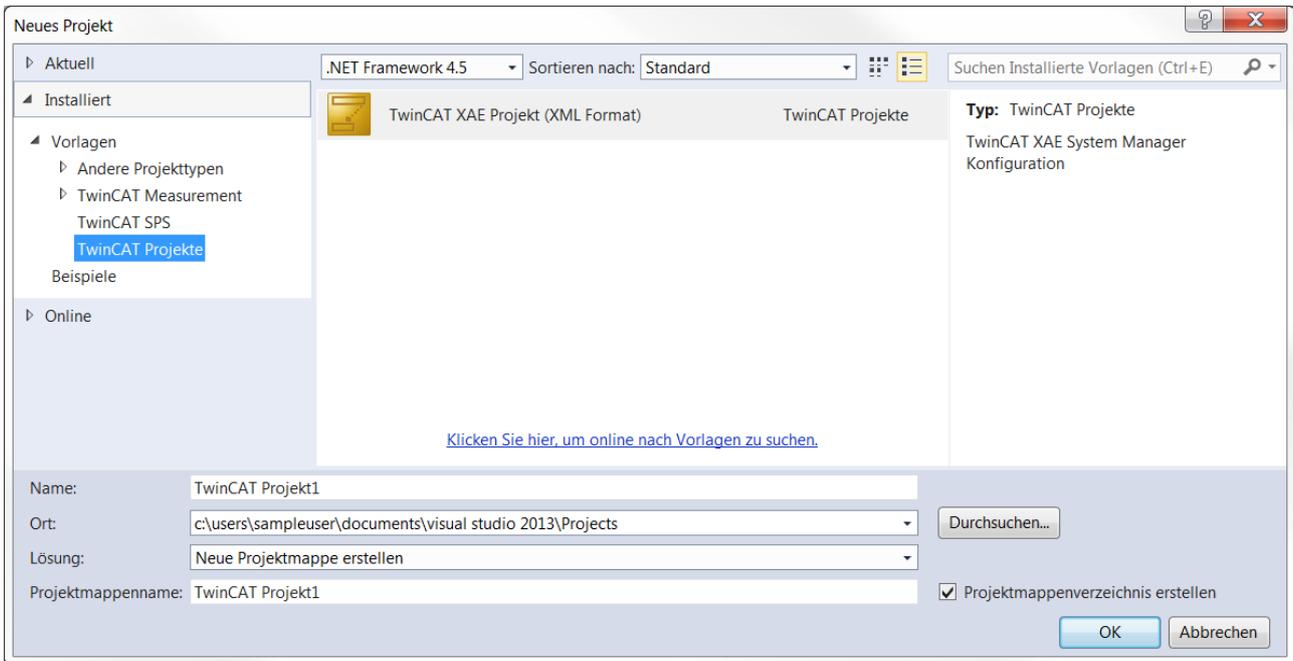
Tastaturkürzel: **[Strg] + [Umschalt] + [N]**

**Funktion:** Der Befehl öffnet den Dialog **Neues Projekt** zum Anlegen einer neuen TwinCAT-Projektdatei.

**Aufruf:** Menü **Datei > Neu**

#### Dialog Neues Projekt

Je nach gewählter Vorlage erhalten Sie ein Projekt, das automatisch mit einem bestimmten Umfang an Objekten ausgestattet ist.



**Kategorien**

Aktuell	Zeigt die zuletzt verwendete Projektvorlage an.
Installiert > Vorlagen	Zeigt die TwinCAT-Projektvorlagen an: <ul style="list-style-type: none"> <li>• Andere Projekttypen</li> <li>• TwinCAT Measurement</li> <li>• TwinCAT SPS</li> <li>• TwinCAT Projekte</li> </ul>
Online	Nicht relevant

**Vorlagen**

Kategorie Andere Projekttypen	
Visual Studio Projektmappen	Leere Visual Studio Projektmappe
Kategorie TwinCAT Measurement	
BodePlot	Bode Plot
Scope	Scope YT Project
	Scope YT NC Project
	Scope YT Project with Reporting
	Scope XY Project
	Scope XY Project with Reporting
Kategorie TwinCAT SPS	
	TwinCAT SPS Projekt
Kategorie TwinCAT Projekte	
	TwinCAT XAE Projekt

Name	Name des anzulegenden Projekts. Je nach Vorlage erscheint ein Standardname. Der numerische Zusatz stellt die Eindeutigkeit des Namens im Dateisystem sicher.  Sie können den Dateinamen unter Berücksichtigung der Dateipfadkonventionen des Betriebssystems ändern. Punkte im Namen sind nicht erlaubt.  TwinCAT fügt automatisch die der gewählten Vorlage entsprechende Dateierweiterung hinzu.
Ort	Speicherort für die neue Projektdatei.  Die Schaltfläche <b>Durchsuchen...</b> öffnet einen Dialog zum Durchsuchen des Dateisystems.  Das Kombinationsfeld zeigt die Historie früher eingegebener Pfade
Lösung	<ul style="list-style-type: none"> <li>• Neue Projektmappe erstellen</li> <li>• Hinzufügen</li> <li>• In neuer Instanz erstellen</li> </ul>
Projektmappenverzeichnis erstellen	<input checked="" type="checkbox"/> Projektmappenverzeichnis wird erstellt.
Projektmappenname	Name der Projektmappe. Standardmäßig wird der TwinCAT-Projektname automatisch übernommen.
OK	TwinCAT öffnet ein neues Projekt.

**Siehe auch:**

- Dokumentation PLC: Ihr erstes TwinCAT-3-SPS-Projekt
- Dokumentation PLC: Standardprojekt anlegen

### 5.1.3 Befehl Projekt/Projektmappe (Projekt/Projektmappe öffnen)

Symbol: 

Tastaturkürzel: **[Strg] + [Umschalt] + [O]**

**Funktion:** Der Befehl öffnet den Standarddialog zum Öffnen einer Datei. Hier können Sie das Dateisystem nach einer TwinCAT-Projektdatei durchsuchen und im Entwicklungssystem öffnen.

**Aufruf:** Menü **Datei > Öffnen**

**Dialog Projekt öffnen**

<b>Dateityp</b>	Auswahlliste zum Filtern des Dateityps <ul style="list-style-type: none"> <li>• Dateien aller unterstützten Formate können geöffnet werden.</li> </ul>
<b>Optionen</b>	<ul style="list-style-type: none"> <li>• Hinzufügen (Verwenden Sie diese Option nur, um einer Projektmappe z. B. noch ein Measurement-Projekt hinzuzufügen. Verwenden Sie diese Option nicht, um einer Projektmappe mehrere TwinCAT-Projekte hinzuzufügen.)</li> <li>• Mappe schließen</li> </ul>
<b>Öffnen</b>	TwinCAT öffnet die gewählte Projektdatei. Wenn erforderlich, wird sie zuvor konvertiert.

**TwinCAT-Projektarchiv \*.tszip**

<b>Inhalt vom *.tszip</b>	Der Archivordner *.tszip enthält das TwinCAT-Projekt, welches archiviert wird.
<b>Befehl zum Erstellen</b>	Ein tszip-Archiv kann über den folgenden Befehl erstellt werden: Befehl Sichern <TwinCAT-Projektname> als Archiv... [ <a href="#">▶ 269</a> ]
<b>Hinweis zu SPS-Projekten</b>	Falls das TwinCAT-Projekt ein oder mehrere SPS-Projekte enthält, sind die Dateien und Ordner, die bezüglich dieser SPS-Projekte in dem Archivordner gespeichert werden, abhängig von den SPS-Projekteinstellungen des jeweiligen SPS-Projekts.  Registerkarte Settings [ <a href="#">▶ 122</a> ]

**Siehe auch:**

- Dokumentation PLC: Ihr erstes TwinCAT-3-SPS-Projekt
- Dokumentation PLC: Standardprojekt anlegen

**5.1.4 Befehl Open Project from Target**

**Funktion:** Der Befehl lädt ein Projekt vom Zielsystem.

**Aufruf:** Menü **Datei > Öffnen**

**Voraussetzung:** Der Netzwerkpfad zum Zielsystem muss konfiguriert sein.

Nachdem Sie den Befehl ausgeführt haben, öffnet sich eine Übersicht mit allen Geräten im Netzwerk. Aus dieser Übersicht wählen Sie das Zielsystem aus. Danach öffnet sich der Dialog **Wähle Verzeichnis für neue Arbeitsmappe**.

**5.1.5 Befehl Neues Projekt... (Neues TwinCAT-Projekt hinzufügen)**

**Funktion:** Der Befehl öffnet den Dialog **Neues Projekt** zum Anlegen einer weiteren TwinCAT-Projektdatei in der Projektmappe.

**Aufruf:** Menü **Datei > Hinzufügen**

**Voraussetzung:** Ein TwinCAT-Projekt ist geöffnet.



Verwenden Sie diesen Befehl nur, um einer Projektmappe z. B. noch ein Measurement-Projekt hinzuzufügen. Verwenden Sie diesen Befehl nicht, um einer Projektmappe mehrere TwinCAT-Projekte hinzuzufügen. Diese Funktion wird von TwinCAT aktuell noch nicht unterstützt.

**5.1.6 Befehl Vorhandenes Projekt... (Vorhandenes TwinCAT-Projekt hinzufügen)**

**Funktion:** Der Befehl öffnet den Dialog **Vorhandenes Projekt hinzufügen** zum Hinzufügen einer TwinCAT-Projektdatei in die Projektmappe.

**Aufruf:** Menü **Datei > Hinzufügen**

**Voraussetzung:** Ein TwinCAT-Projekt ist geöffnet.



Verwenden Sie diesen Befehl nur, um einer Projektmappe z. B. noch ein Measurement-Projekt hinzuzufügen. Verwenden Sie diesen Befehl nicht, um einer Projektmappe mehrere TwinCAT-Projekte hinzuzufügen. Diese Funktion wird von TwinCAT aktuell noch nicht unterstützt.

**5.1.7 Befehl Zuletzt geöffnete Projekte und Projektmappen**

**Funktion:** Der Befehl öffnet die Liste der zuletzt verwendeten Projekte, aus der Sie ein Projekt zum Öffnen auswählen können.

**Aufruf:** Menü **Datei**

**Siehe auch:**

- Dokumentation PLC: Projekt anlegen und konfigurieren

### 5.1.8 Befehl Alles speichern

Symbol: 

**Funktion:** Der Befehl speichert alle Objekte des TwinCAT-Projekts.

**Aufruf:** Menü **Datei**, Standard Symbolleistenoptionen

### 5.1.9 Befehl Auswahl speichern/sichern

Symbol: 

Tastaturkürzel: **[Strg] + [S]**

**Funktion:** Der Befehl speichert die Projektmappe, das TwinCAT-Projekt, das TwinCAT-SPS-Projekt oder ein ausgewähltes SPS-Objekt (Main, GVL,...) unter dem aktuellen Namen.

**Aufruf:** Menü **Datei**, Standard Symbolleistenoptionen

**Voraussetzung:** Die Projektmappe, das TwinCAT-Projektobjekt, das SPS-Projektobjekt (<SPS-Projektname> Project) oder das zu speichernde SPS-Objekt ist im **Projektmappen-Explorer** ausgewählt.

#### Objekt speichern

Der Befehl speichert das Objekt unter dem aktuellen Namen. Wenn das Objekt seit dem letzten Speichern geändert wurde, ist das „Disketten“-Symbol am Objektsymbol rot und der Name in der Titelleiste des Editor des geöffneten Objekts mit einem Sternchen („\*“) versehen.

**Siehe auch:**

- [Befehl Sichern <TwinCAT-Projektname> als \[▶ 62\]](#)

### 5.1.10 Befehl <Projektmappenname> speichern unter

**Funktion:** Der Befehl öffnet den Standarddialog zum Speichern einer Datei. Die Projektmappe kann unter dem gewünschten Speicherpfad abgelegt werden. Standardmäßig ist der Dateityp UTF-8-Projektmappendatei (\*.sln) ausgewählt.

**Aufruf:** Menü **Datei**

**Voraussetzung:** Die Projektmappe ist im **Projektmappen-Explorer** ausgewählt.

### 5.1.11 Befehl Sichern <TwinCAT-Projektname> als

**Funktion:** Der Befehl öffnet den Standarddialog zum Speichern einer Datei. Das Projekt kann unter dem gewünschten Speicherpfad und Dateityp abgelegt werden. Standardmäßig ist der Dateityp TwinCAT XAE Project (\*.tsproj) ausgewählt.

**Aufruf:** Menü **Datei**

**Voraussetzung:** Das TwinCAT-Projektobjekt ist im **Projektmappen-Explorer** ausgewählt.

Beachten Sie, dass bei diesem Speichervorgang beispielsweise nur die \*.tsproj-Datei an einem anderen Speicherort erzeugt wird. Die referenzierten Projekte und die dort enthaltenen Objekte werden nicht an diesem neuen Speicherort gespeichert (z. B. ein SPS-Projekt, das in dem TwinCAT-3-Projekt eingebunden ist, und dessen Objekte).

**Siehe auch:**

- [Befehl Auswahl speichern/sichern \[► 62\]](#)
- Dokumentation PLC: Bibliothekserstellung

### 5.1.12 Befehl Sichern <SPS-Projektname> als

**Funktion:** Der Befehl öffnet einen Dialog, in dem ein Zielverzeichnis für die SPS-Projektdatei bestimmt werden kann. Die SPS-Projektobjekte und die .plcproj-Datei werden in dem ausgewählten Verzeichnis gespeichert.

**Aufruf:** Kontextmenü SPS-Projektobjekt

**Voraussetzung:** Das SPS-Projektobjekt (<SPS-Projektname>) ist im **Projektmappen-Explorer** ausgewählt.

### 5.1.13 Befehl Disassemblierungsdatei erzeugen

**Funktion:** Der Befehl erzeugt aus dem aktuellen Projekt eine Disassemblierungsdatei <project name>.asm und legt sie im Dateiverzeichnis im Projektordner ab.

**Aufruf:** Menü PLC > Disassemblierungsdatei erzeugen

### 5.1.14 Befehl Sende per E-Mail.../Send by E-Mail...

Symbol: 

**Funktion:** Der Befehl startet das aktuell im System eingestellte E-Mail-Programm und öffnet eine neue E-Mail. Diese enthält im Anhang die Archivdatei des ausgewählten Projekts.

**Aufruf:** Menü **Datei**, Kontextmenü

### 5.1.15 Befehl Projekt/Projektmappe schließen

Symbol: 

**Funktion:** Der Befehl schließt das gerade geöffnete Projekt. TwinCAT bleibt geöffnet.

**Aufruf:** Menü **Datei** oder implizit beim Öffnen eines neuen/anderen Projekts, während noch ein Projekt geöffnet ist.

Wenn das Projekt nicht-gespeicherte Änderungen enthält, erscheint eine Abfrage, ob das Projekt gespeichert werden soll.

Wenn Sie das Projekt noch nicht explizit gespeichert haben, erscheint eine Abfrage, ob Sie die Projektdateien löschen wollen.

**Siehe auch:**

- Dokumentation PLC: SPS-Projekt anlegen und konfigurieren

### 5.1.16 Befehl Schließen

**Funktion:** Der Befehl schließt den geöffneten Editor.

**Aufruf:** Menü **Datei**

**Voraussetzung:** Der zu schließende Editor ist aktiv bzw. das Objekt im SPS-Projektbaum ist selektiert.

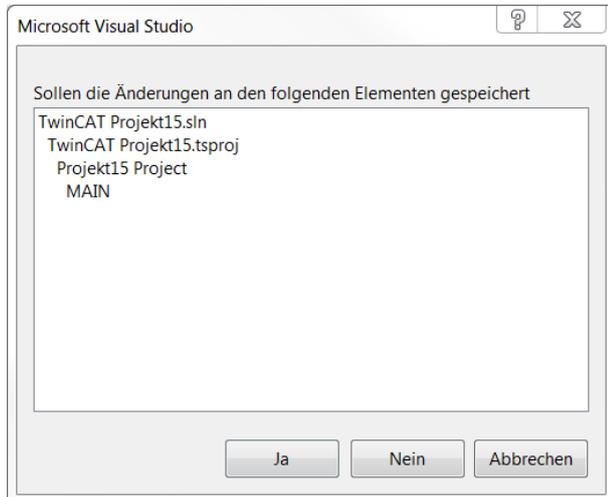
## 5.1.17 Befehl Beenden

Symbol: 

Tastaturkürzel: **[Alt] + [F4]**

**Funktion:** Der Befehl beendet das Programmiersystem. Wenn gerade ein Projekt geöffnet ist, das seit dem letzten Speichern verändert wurde, erscheint ein Dialog mit der Abfrage, ob das Projekt gespeichert werden soll.

**Aufruf:** Menü **Datei**



## 5.1.18 Befehl Seite einrichten...

Symbol: 

**Funktion:** Der Befehl öffnet den Dialog **Seiteneinstellungen** zur Konfiguration des Layouts für die Druckversion des Projektinhalts.

**Aufruf:** Menü **Datei**

**Voraussetzung:** Ein Editorfenster ist aktiv.

**Siehe auch:**

- [Befehl Drucken \[► 64\]](#)

## 5.1.19 Befehl Drucken

Symbol: 

**Funktion:** Der Befehl öffnet den Standarddialog von Windows zum Drucken von Dokumenten.

**Aufruf:** Menü **Datei**

**Voraussetzung:** Ein Editorfenster ist aktiv.

# 5.2 Bearbeiten

## 5.2.1 Standardbefehle

TwinCAT stellt Ihnen die folgenden Standardbefehle zur Verfügung:

- Rückgängig:



, Tastaturkürzel: **[Strg] + [Z]**

- Wiederholen:



, Tastaturkürzel: **[Strg] + [Y]**

- Ausschneiden:



, Tastaturkürzel: **[Strg] + [X]**

- Kopieren:



, Tastaturkürzel: **[Strg] + [C]**

- Einfügen:



, Tastaturkürzel: **[Strg] + [V]**

- Löschen:



, Tastaturkürzel: **[Entf]**

- Alles auswählen: Tastaturkürzel: **[Strg] + [A]**

**Aufruf:** Menü **Bearbeiten**, Kontextmenü SPS-Projektbaum, Kontextmenü Editorfenster

Der Befehl **Einfügen** wird nicht von allen Editoren unterstützt oder kann in manchen nur eingeschränkt verwendet werden. In grafischen Editoren wird der Befehl nur unterstützt, wenn durch das Einfügen ein korrektes Konstrukt entsteht.

Im SPS-Projektbaum bezieht sich der Befehl auf das gerade ausgewählte Objekt. Mehrfachauswahl ist möglich.

## 5.2.2 Befehl Entfernen

Tastaturkürzel: **[Entf]**

**Funktion:** Der Befehl entfernt das ausgewählte SPS-Objekt aus der Projektmappe. Im Projektverzeichnis bleibt das Objekt erhalten.

**Aufruf:** Kontextmenü SPS-Objekt

## 5.2.3 Befehl Alles auswählen

Tastaturkürzel: **[Strg+A]**

**Funktion:** Der Befehl wählt den gesamten Inhalt aus.

**Aufruf:** Menü **Bearbeiten**, Kontextmenü Editorfenster

## 5.2.4 Befehl Eingabehilfe

Symbol:

Tastaturkürzel: **[F2]**

**Funktion:** Der Befehl öffnet den Dialog **Eingabehilfe**, der Sie bei der Auswahl eines an der aktuellen Cursorposition möglichen Programmierelements unterstützt.

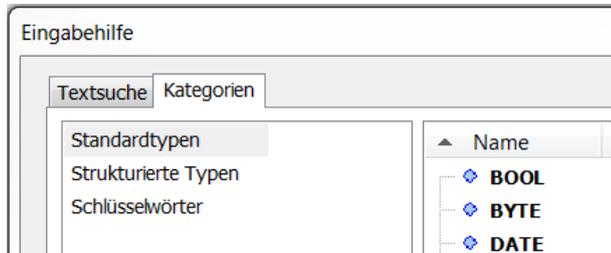
**Aufruf:** Menü **Bearbeiten**, Kontextmenü Editorfenster

**Voraussetzung:** Eine POU ist im Editor geöffnet und der Cursor steht in einer Programmzeile.

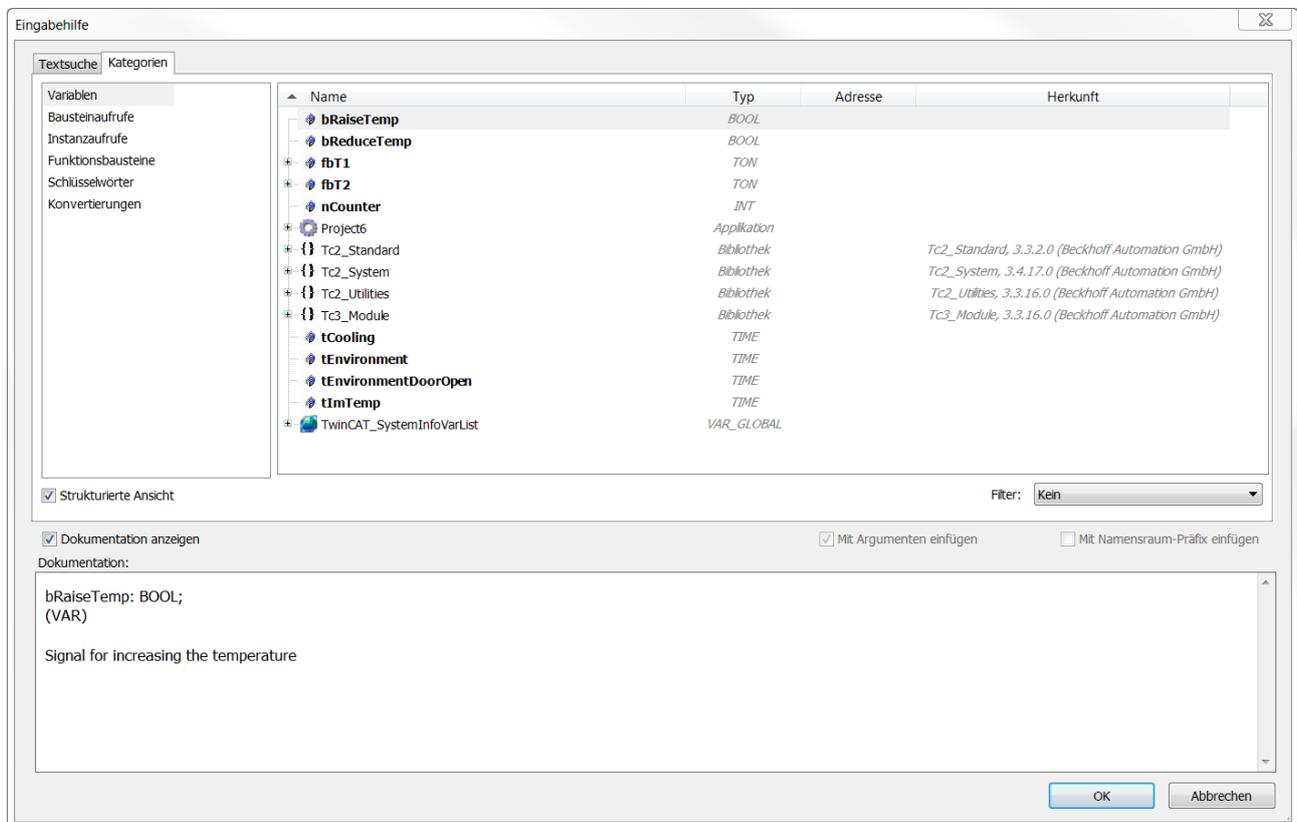
### Dialog Eingabehilfe - Registerkarte Kategorien

Der Dialog bietet Ihnen alle Programmierelemente an, die Sie im Editor an der aktuellen Cursorposition einfügen können. Die Elemente sind nach Kategorien sortiert. Sie können in der Kategorie **Variablen** zusätzlich einen Filter für den Gültigkeitsbereich setzen, wie beispielsweise Lokale Variablen, Globale Variablen oder Konstanten.

Ausschnitt aus dem Dialog **Eingabehilfe** im Deklarationsteil des Editors:



Dialog **Eingabehilfe** im Implementierungsteil des Editors:



Strukturierte Ansicht	<input checked="" type="checkbox"/> : Die Elemente werden in einem Strukturbaum dargestellt. Sie können die Spalten Typ, Adresse und Herkunft mit einem Rechtsklick in den Spaltentitel in einem Untermenü aus- oder einblenden.  <input type="checkbox"/> : Die Elemente werden in einer flachen Struktur dargestellt.
Filter	In dem Drop-down-Listefeld können Sie einen zusätzlichen Filter für den Variablentyp setzen.
Dokumentation anzeigen	<input checked="" type="checkbox"/> : Eine Beschreibung des ausgewählten Elements wird angezeigt.
Mit Argumenten einfügen	<input checked="" type="checkbox"/> : TwinCAT fügt Elemente, die Argumente besitzen, wie zum Beispiel Funktionen, mit diesen Argumenten an der Cursorposition ein.  Beispiel: Wenn Sie den Funktionsbaustein fb1, der eine Eingabevariable fb1_in und eine Ausgabevariable fb1_out enthält, "mit Argumenten" einfügen, sieht dies im Editor folgendermaßen aus: <code>fb1(fb1_in:= , fb1_out=&gt; )</code>
Mit Namensraum-Präfix einfügen	<input checked="" type="checkbox"/> : TwinCAT fügt das ausgewählte Element mit vorangestelltem Namensraum ein. Im Fall von Bibliotheksbausteinen können Sie das Auswahlkästchen nicht bedienen, wenn in den Eigenschaften der Bibliothek festgelegt ist, dass die Angabe des Namensraums zwingend ist.

**Dialog Eingabehilfe - Registerkarte Textsuche**

In der Registerkarte können Sie nach bestimmten Objekten suchen. Sobald Sie ein oder mehrere Zeichen in das Suchfeld eingeben, listet das Trefferfenster die Namen aller Objekte, deren Name diese Suchzeichenfolge enthält. Mit einem Doppelklick auf das gewünschte Objekt fügen Sie es an der aktuellen Cursorposition im Editor ein.

Filter	Einschränken der Suche auf eine bestimmte Variablenkategorie.
--------	---

**Siehe auch:**

- Dokumentation PLC: Eingabeunterstützung nutzen

**5.2.5 Befehl Variable deklarieren**

**Funktion:** Der Befehl öffnet den Dialog **Variable deklarieren**, der die Deklaration einer Variablen unterstützt.

**Aufruf:** Menü **Bearbeiten**, Kontextmenü Editorfenster

**Voraussetzung:** Eine POU ist im Editor geöffnet und der Cursor steht in einer Programmzeile.

Durch die Autodeklarationsfunktion erscheint der Dialog **Variable deklarieren** auch, wenn der Cursor im Implementierungsteil einer POU in einer Zeile steht, die den Namen einer nicht deklarierten Variablen enthält. Als Voraussetzung hierfür müssen Sie in den TwinCAT-Optionen die Option **Unbekannte Variablen automatisch deklarieren (AutoDeclare)** aktiviert haben (**Extras > Optionen > TwinCAT > SPS Programmierumgebung > Intelligentes Kodieren**).

Durch die Smart-Tag-Funktion erscheint der Befehl **Variable deklarieren** auch, wenn Sie im Implementierungsteil des ST-Editors den Cursor auf eine Variable setzen, die nicht deklariert wurde, und

anschließend auf  klicken (verfügbar ab Build 4026).

### Dialog Variable deklarieren

Variable deklarieren

Gültigkeitsbereich: VAR

Name: nVar1

Datentyp: INT

Objekt: MAIN [SamplePLCProject]

Initialisierungswert: ...

Adresse: ...

Flags:

- CONSTANT
- RETAIN
- PERSISTENT

Kommentar:

OK Abbrechen

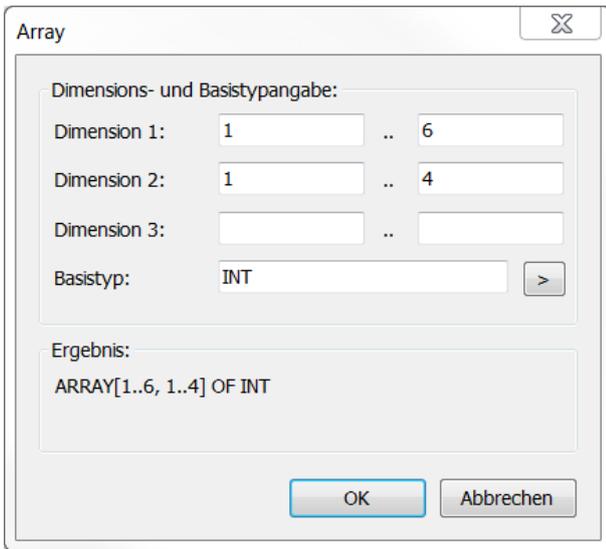
Gültigkeitsbereich	Gültigkeitsbereich der noch nicht deklarierten Variablen. Beispiel: VAR (Standardeinstellung bei lokaler Variable)
Name	Noch nicht deklariertes Variablenname Beispiel: bVar
Datentyp	<p>▼ : Listet die Standarddatentypen auf.</p> <p>☞ :</p> <ul style="list-style-type: none"> <li>• Eingabehilfe: Öffnet den Dialog <b>Eingabehilfe</b></li> <li>• Array-Assistent: Öffnet den Dialog <b>Array</b></li> </ul> <p>Beispiel: BOOL</p>
Objekt	<p>Objekt, in dem die neue Variable deklariert wird. Standardmäßig steht hier das Objekt, das Sie gerade bearbeiten.</p> <p>▼ : Listet die Objekte auf, in denen die Variable deklariert werden kann.</p> <p>Wenn für den ausgewählten Gültigkeitsbereich keine Objekte verfügbar sind, erscheint der Eintrag &lt;Objekt anlegen&gt;. Wenn Sie den Eintrag &lt;Objekt anlegen&gt; auswählen, öffnet sich der Dialog <b>Objekt hinzufügen</b> zur Erzeugung eines geeigneten Objekts.</p>
Initialisierungswert	<p>Wenn Sie keinen Initialisierungswert eingeben, wird die Variable automatisch initialisiert.</p> <p>☞ : Öffnet den Dialog <b>Initialisierungswert</b>. Diese Vorgehensweise ist für die Initialisierung von strukturierten Variablen hilfreich.</p> <p>Beispiel: FALSE</p>
Adresse	Speicheradresse (siehe Dokumentation PLC: Adressen) Beispiel: %IX1.0
Flags	<p>Attribut-Schlüsselwörter</p> <ul style="list-style-type: none"> <li>• CONSTANT: Schlüsselwort für eine Konstante</li> <li>• RETAIN: Schlüsselwort für eine remanente Variable</li> <li>• PERSISTENT: Schlüsselwort für eine persistente Variable (strenger als RETAIN)</li> </ul> <p>Das ausgewählte Attribut-Schlüsselwort wird der Variablendeklaration hinzugefügt.</p>
Kommentar	<p>Im tabellarischen Deklarationseditor erfolgt die Anzeige des eingegebenen Kommentars in der Spalte Kommentar, im textuellen Deklarationseditor oberhalb der Variablendeklaration.</p> <p>Beispiel: New variable</p>
Änderungen mit Hilfe von Refactoring anwenden	<p>Die Option erscheint bei folgenden Gültigkeitsbereichen:</p> <ul style="list-style-type: none"> <li>• Eingabevariable (VAR_INPUT)</li> <li>• Ausgabevariable (VAR_OUTPUT)</li> <li>• Ein- und Ausgabevariable (VAR_IN_OUT)</li> </ul> <p>Die Option ist automatisch aktiviert, wenn in den TwinCAT-Optionen die Autodeklaration-Optionen <b>Beim Umbenennen von Variablen</b> und <b>Für das Hinzufügen oder Entfernen von Variablen, oder für das Ändern des Namensraums</b> aktiviert sind (<b>Extras &gt; Optionen &gt; TwinCAT &gt; SPS Programmierumgebung &gt; Refactoring</b>) (siehe <a href="#">Dialog Optionen - Refactoring</a> [▶ 179]).</p> <p>Wenn die Option aktiviert ist, wird die Variable beim Beenden des Dialogs noch nicht deklariert, sondern es öffnet sich zunächst der Dialog <b>Refactoring</b>, in dem Sie die Änderungen weiter bearbeiten können.</p>

OK	<p>Die Variable wird deklariert und erscheint in der Deklaration.</p> <p><b>Beispiel:</b></p> <pre>VAR     // New variable     bVar: BOOL := FALSE; END_VAR</pre>
----	---

**Siehe auch:**

- Dokumentation PLC: Variablen deklarieren

**Dialog Array**

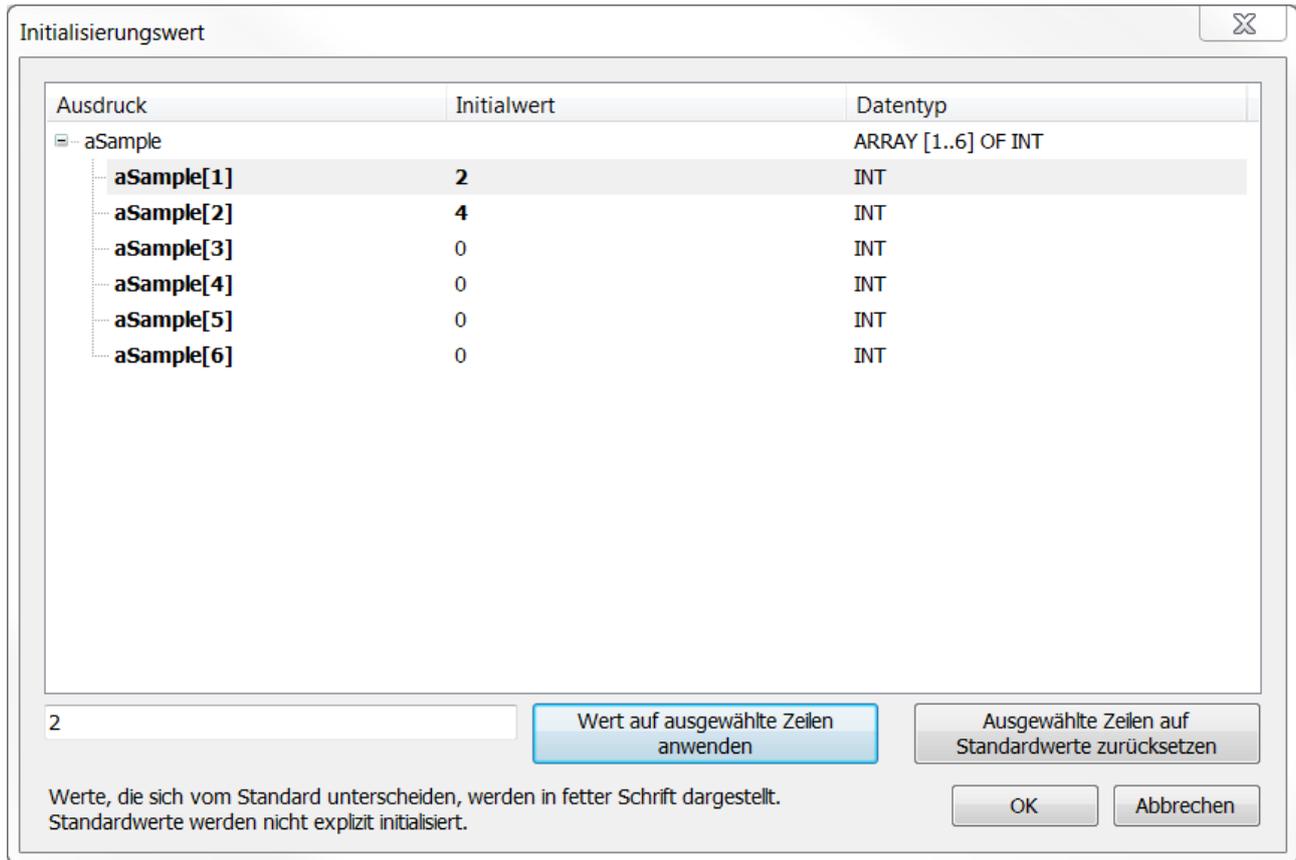


Dimensions- und Basistyp-Angabe	<p>Definition der Feldgrößen (Dimension) durch Eingeben der unteren und oberen Grenzen und des Basistyps des Arrays. Den Basistyp können Sie direkt eingeben oder mithilfe der Dialoge <b>Eingabehilfe</b> oder <b>Array</b>, wenn Sie die Schaltfläche  anklicken.</p>
Ergebnis	Anzeige des definierten Arrays.



TwinCAT initialisiert Variablen nur neu, wenn Sie die Initialisierungswerte der Variablen geändert haben.

Dialog Initialisierungswert



Auflistung der Variablen mit Namen (Ausdruck), Initialisierungswert und Datentyp. Geänderte Initialisierungswerte werden fett dargestellt.	
Eingabefeld unterhalb der Liste	Eingabe eines Initialisierungswerts für die selektierte Variable(n).
Wert auf ausgewählte Zeilen anwenden	Änderung des Initialisierungswerts der selektierten Zeile(n) entsprechend dem Wert des Eingabefelds.
Ausgewählte Zeilen auf Standardwerte zurücksetzen	Herstellung der Standard-Initialisierungswerte.
OK	TwinCAT übernimmt die Initialisierungswerte in den Dialog <b>Variable deklarieren</b> .

Falls die über diesen Dialog zu initialisierende Variable eine Funktionsbaustein-Instanz mit erweiterter FB\_Init-Methode ist, wird oberhalb der Tabelle Initialisierungswert eine weitere Tabelle angezeigt (siehe Dokumentation PLC: Methoden FB\_init, FB\_reinit und FB\_exit). In dieser Tabelle werden die zusätzlichen FB\_Init-Parameter aufgelistet. Die Bedeutung und Bedienung entspricht im Wesentlichen der unteren Tabelle mit folgenden Unterschieden:

- Es müssen alle Variablen mit Initialisierungswerten belegt sein. Ansonsten ist OK nicht anwählbar.
- Bei komplexen Datentypen (Strukturen, Arrays) werden keine darin enthaltenen Komponenten angezeigt (Typ kann nicht aufgeklappt werden). In diesem Fall muss der komplexe Typ mit einer entsprechenden Variable initialisiert werden.

Bei so konfigurierten FB\_Init-Parametern wird im Dialog **Variable deklarieren** ein entsprechendes Symbol hinter dem Initialisierungswert angezeigt.

Siehe auch:

- Dokumentation PLC: Eingabeunterstützung nutzen

5.2.6 Befehl Zur Überwachungsliste hinzufügen

Symbol:

**Funktion:** Der Befehl fügt die Variable, auf der gerade der Cursor steht, einer Überwachungsliste hinzu, die dem Online-Monitoring dient.

**Aufruf:** Kontextmenü

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb und der Cursor steht in einem Editor auf einer Variablen.

Der Befehl fügt die Variable in die gerade geöffnete Überwachungsliste ein. Wenn gerade keine Überwachungsliste geöffnet ist, fügt der Befehl die Variable in Überwachungsliste 1 ein und öffnet deren Ansicht.

**Siehe auch:**

- Dokumentation PLC: Überwachungslisten verwenden
- Dokumentation PLC: Monitoring von Werten

## 5.2.7 Befehl Aufrufbaum ausgeben

Symbol: 

**Funktion:** Der Befehl öffnet die Ansicht **Aufrufbaum**, die die Aufrufe des Bausteins sowie seine Aufrufer darstellt.

**Aufruf:** Kontextmenü Editorfenster

**Voraussetzung:** Ein Baustein ist im Editor geöffnet und der Cursor steht in einer Variablen.

## 5.2.8 Befehl Gehe zu

**Funktion:** Mit dem Befehl springt der Cursor zu einer definierten Zeile im Code.

**Aufruf:** Menü **Bearbeiten**

**Voraussetzung:** Ein Texteditor ist geöffnet und der Cursor steht in einer Programmzeile.

Der Befehl öffnet einen Dialog mit einem Eingabefeld **Zeilennummer**.

## 5.2.9 Befehl Gehe zur Definition

Symbol: 

Tastaturkürzel: **[F12]**

**Funktion:** Der Befehl zeigt die Definitionsstelle einer Variablen oder Funktion.

**SPS-Editor**

**Aufruf:** Kontextmenü Editorfenster

**Voraussetzung:** Eine POU ist im Editor geöffnet und der Cursor steht auf einer Variablen oder Funktion.

**SPS-Prozessabbild**

**Aufruf:** Kontextmenü Projektmappen-Explorer

**Voraussetzung:** Das Prozessabbild (Projektinstanz) ist ausgeklappt und der Cursor steht auf einer allokierten Variablen im Prozessabbild.

## 5.2.10 Befehl Gehe zur Instanz

Symbol: 

**Funktion:** Der Befehl öffnet die Instanz eines Funktionsbausteins in einem neuen Fenster.

**Aufruf:** Kontextmenü Editorfenster

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Eine POU ist im Editor geöffnet und der Cursor steht auf der Instanz eines Funktionsbausteins.

Der Befehl ist nicht verfügbar für temporäre Instanzen oder Instanzen aus übersetzten Bibliotheken.

## 5.2.11 Befehl Gehe zur Implementierung

Symbol: 

**Funktion:** Der Befehl öffnet die Online-Ansicht einer Methode in einem neuen Fenster.

**Aufruf:** Kontextmenü Editorfenster

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Eine POU ist im Editor geöffnet und der Cursor steht auf einem Methodenaufruf.



Verfügbar ab TwinCAT 3.1 Build 4026

---

## 5.2.12 Befehl Gehe zum Verweis

Symbol: 

**Funktion:** Der Befehl öffnet im Onlinebetrieb die Deklarationsstelle der Variablen, die von dem gerade fokussierten Pointer referenziert wird.

**Aufruf:** Kontextmenü Editorfenster

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Eine POU ist im Editor geöffnet und der Cursor steht auf einem Pointer. Die referenzierte Variable liegt in einem statischen Speicher.

Wenn der Pointer nicht exakt auf den Beginn der Variablen zeigt, wird beim Wechseln zur Variablendeklaration eine entsprechende Meldung ausgegeben.



Verfügbar ab TwinCAT 3.1 Build 4026

---

## 5.2.13 Befehl Verweise suchen

Tastaturkürzel: **[Umschalt+F12]**

**Funktion:** Der Befehl zeigt alle Verwendungsstellen einer Variablen in der Ansicht **Querverweisliste**.

**Aufruf:** Kontextmenü

**Voraussetzung:** Eine POU ist im Editor geöffnet und der Cursor steht in einer Variablen oder die Ansicht **Querverweisliste** ist geöffnet und eine Variable ist im Feld **Name** angegeben.

**Siehe auch:**

- Dokumentation PLC: Verwendungsstellen mit der Querverweisliste finden

### 5.2.14 Befehl Navigiere zu

**Funktion:** Der Befehl öffnet einen Dialog zur Auswahl eines Elements, das gezielt geöffnet werden soll.

**Aufruf:** Menü **Bearbeiten**

### 5.2.15 Befehl In Großbuchstaben umwandeln

**Funktion:** Der Befehl wandelt alle Kleinbuchstaben im selektierten Code in Großbuchstaben um.

**Aufruf:** Menü **Bearbeiten > Erweitert**

**Voraussetzung:** Eine POU ist im Editor geöffnet und Code ist selektiert.

### 5.2.16 Befehl In Kleinbuchstaben umwandeln

**Funktion:** Der Befehl wandelt alle Großbuchstaben im selektierten Code in Kleinbuchstaben um.

**Aufruf:** Menü **Bearbeiten > Erweitert**

**Voraussetzung:** Eine POU ist im Editor geöffnet und Code ist selektiert.

### 5.2.17 Befehl Leerstelle anzeigen

Symbol:

**Funktion:** Der Befehl bewirkt, dass Steuerzeichen für Leerzeichen und Tabulatoren angezeigt werden.

**Aufruf:** Menü **Bearbeiten > Erweitert**

**Voraussetzung:** Eine POU ist im Editor geöffnet.

TwinCAT visualisiert Leerzeichen durch einen Punkt und Tabulatoren durch einen Pfeil.

### 5.2.18 Befehl Auswahl auskommentieren

Tastaturkürzel: [Ctrl+K] + [Ctrl+C]

**Funktion:** Der Befehl kommentiert einen selektierten Codebereich aus. Der Codebereich wird von der Kompilierung ausgeschlossen und hat keinen Einfluss auf die Programmausführung.

**Aufruf:** Menü **Bearbeiten > Erweitert**

**Voraussetzung:** Ein Baustein ist im Editor geöffnet und Code ist selektiert.

Sie verwenden den Befehl, um Kommentare für die Dokumentation eines Programms oder Programmabschnitts zu erstellen oder um einen Codebereich kurzfristig von der Kompilierung auszuschließen. Über den [Befehl Auskommentierung der Auswahl aufheben](#) [► 74] können Sie einen Kommentar aufheben und einen auskommentierten Codebereich wieder in die Programmausführung einbinden.

### 5.2.19 Befehl Auskommentierung der Auswahl aufheben

Tastaturkürzel: [Ctrl+K] + [Ctrl+U]

**Funktion:** Der Befehl hebt einen Kommentar auf und fügt einen auskommentierten Codebereich wieder in die Programmausführung ein.

**Aufruf:** Menü **Bearbeiten > Erweitert**

**Voraussetzung:** Ein Baustein ist im Editor geöffnet und Code ist selektiert, welcher zuvor über den [Befehl Auswahl auskommentieren](#) [▶ 74] auskommentiert wurde.

## 5.2.20 Befehl Schnellsuche (In Dateien suchen)

Symbol: 

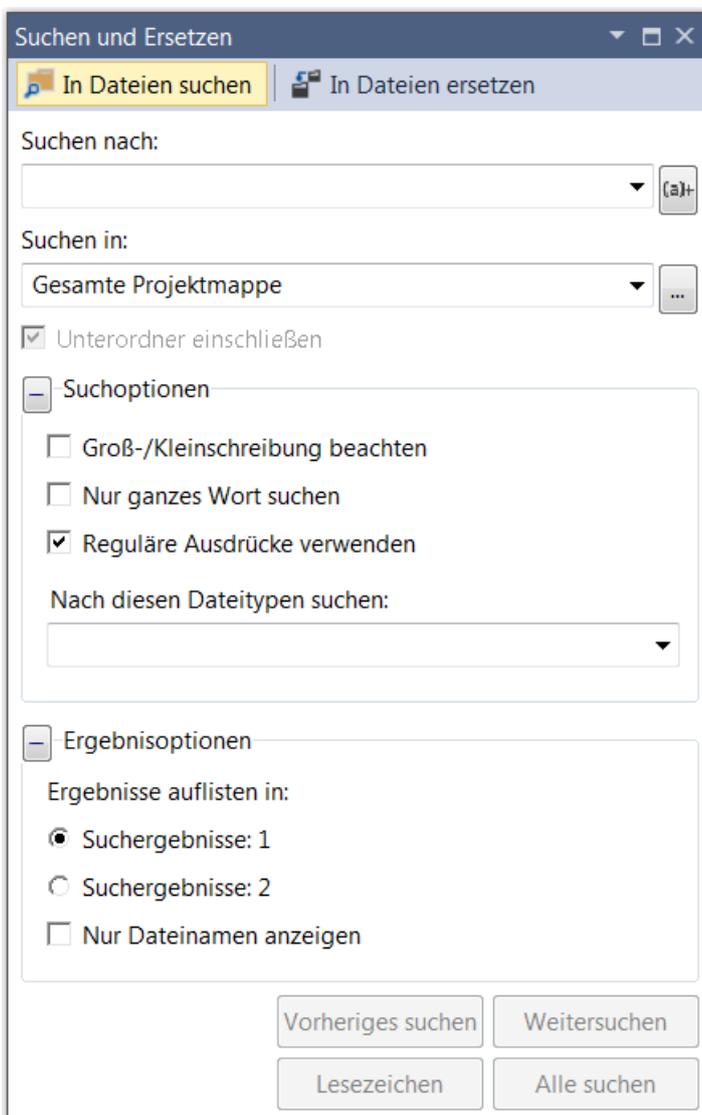
Tastaturkürzel: **[Strg] + [F]**

**Funktion:** Der Befehl durchsucht das Projekt oder Teile davon nach einer bestimmten Zeichenkette.

**Aufruf:** Menü **Bearbeiten > Suchen und Ersetzen**

Der Befehl öffnet den Dialog **Suchen und Ersetzen** (Schaltfläche **In Dateien suchen** ist aktiv), in dem die zu suchende Zeichenkette eingegeben und die Suchoptionen definiert werden.

### Dialog Suchen und Ersetzen



Suchen und Ersetzen

In Dateien suchen | In Dateien ersetzen

Suchen nach:

Suchen in:  
Gesamte Projektmappe

Unterordner einschließen

Suchoptionen

Groß-/Kleinschreibung beachten

Nur ganzes Wort suchen

Reguläre Ausdrücke verwenden

Nach diesen Dateitypen suchen:

Ergebnisoptionen

Ergebnisse auflisten in:

Suchergebnisse: 1

Suchergebnisse: 2

Nur Dateinamen anzeigen

Vorheriges suchen | Weitersuchen

Lesezeichen | Alle suchen

In Dateien ersetzen	Wechselt zum Dialog <b>Suchen und Ersetzen</b> (Schaltfläche <b>In Dateien ersetzen</b> ist aktiv)
Suchen nach	Zeichenkette, nach der gesucht wird.
Suchen in	 : Auswahlliste mit den Objekten, die durchsucht werden: Gesamte Projektmappe: Alle editierbaren Stellen in allen Objekten des Projekts werden durchsucht. Aktuelles Projekt: Alle geöffneten Dokumente: Alle Editoren, die gerade in einem Fenster geöffnet sind, werden durchsucht. Aktuelles Dokument: Nur der Editor, in dem gerade der Cursor steht, wird durchsucht.   : Öffnet einen Dialog, in dem die zu durchsuchenden Objekte genauer definiert werden können.
Groß-/Kleinschreibung	 : Die Suche beachtet Groß-/Kleinschreibung.
Nur ganzes Wort	 : Nur Zeichenketten, die genau die gesuchte Zeichenkette wiedergeben, werden gefunden.
Nach diesen Dateitypen suchen	Drop-down-Liste zur Auswahl eines Dateityps
Reguläre Ausdrücke verwenden	Aktiviert die Schaltfläche  , über die Sie Unterstützung bei der Eingabe von regulären Ausdrücken (Regular Expressions) erhalten. Diese Funktion wird für SPS-Editoren nicht unterstützt!
Nur Dateinamen anzeigen	 : Nur Dateinamen werden angezeigt.
Weitersuchen	Starten der Suche. Das nächste Suchergebnis wird an dessen Position im betreffenden Editor angezeigt.
Alle Suchen	Der Befehl gibt alle Treffer der Suche im Meldungsfenster aus. Dabei werden das Objekt und die genaue Position des Treffers angezeigt. <ul style="list-style-type: none"> <li>• (Dekl): Deklarationsteil des Objekts</li> <li>• (Impl): Implementierungsteil des Objekts</li> </ul> Durch einen Doppelklick auf den Listeneintrag wird der Treffer im Editor angezeigt.

**Siehe auch:**

- [Befehl Schnellersetzung \(In Dateien ersetzen\) \[► 76\]](#)
- Dokumentation PLC: Projektweites Suchen und Ersetzen

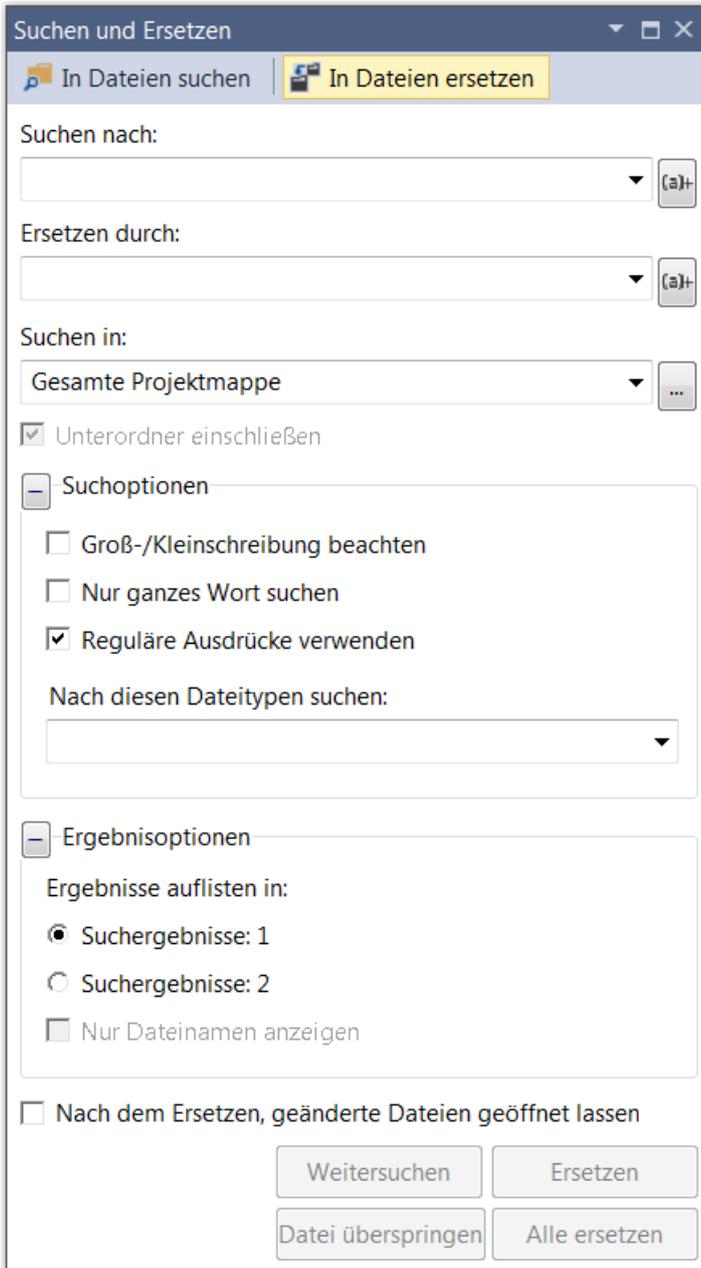
**5.2.21 Befehl Schnellersetzung (In Dateien ersetzen)**Symbol: Tastaturkürzel: **[Strg] + [H]**

**Funktion:** Die Befehle durchsuchen das Projekt oder Teile davon nach einer bestimmten Zeichenkette und ersetzen diese.

**Aufruf:** Menü **Bearbeiten > Suchen und Ersetzen**

Der Befehl öffnet den Dialog **Suchen und Ersetzen** (Schaltfläche **In Dateien ersetzen** ist aktiv), in dem die zu ersetzende Zeichenkette und die neue Zeichenkette eingegeben und die Suchoptionen definiert werden.

**Dialog Suchen und Ersetzen**



Zusätzlich zu den Optionen des Dialogs „Suchen“ sind noch folgende Einstellungen möglich:

Ersetzen durch	Eingabefeld für die neue Zeichenkette.
Ersetzen	Damit wird jeweils die nächste gefundene Zeichenkette im Editor hervorgehoben und ersetzt (schrittweises Ersetzen).
Alle Ersetzen	Alle gefundenen Zeichenketten werden auf einmal ersetzt, ohne dass sie in den Editoren angezeigt werden.
Nach dem Ersetzen, geänderte Dateien geöffnet lassen	Die Editoren der gefundenen Objekte bleiben geöffnet.

**Siehe auch:**

- [Befehl Schnellsuche \(In Dateien suchen\) \[► 75\]](#)
- [Dokumentation PLC: Projektweites Suchen und Ersetzen](#)

## 5.2.22 Befehl Schreibmodus umschalten

Tastaturkürzel: **[Einfg]**

**Funktion:** Dieser Befehl aktiviert den Überschreibmodus oder den Einfügemodus.

**Aufruf:** Doppelklick auf das Zeichen **[EINFG]** bzw. **[ÜB]** in der Status- und Informationsleiste

**Voraussetzung:** Ein Editorfenster ist aktiv.

Wenn der Überschreibmodus aktiviert ist, werden Zeichen vor dem Cursor bei der Eingabe neuer Zeichen überschrieben. Wenn der Einfügemodus aktiviert ist, werden Zeichen eingefügt und bestehende Zeichen vor dem Cursor bleiben erhalten.

## 5.2.23 Befehl Umbenennen

**Funktion:** Der Befehl ermöglicht das Umbenennen eines SPS-Objekts im **Projektmappen-Explorer**.

**Aufruf:** Kontextmenü SPS-Objekt

## 5.2.24 Befehl Objekt (offline) bearbeiten

**Funktion:** Der Befehl öffnet das Objekt offline in seinem Editor.

**Aufruf:** Menü **Projekt**, Kontextmenü

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Ein Objekt im SPS-Projektbaum ist selektiert.

Somit können Sie auch im Onlinebetrieb das Objekt editieren. Die Änderung übertragen Sie anschließend mit dem Befehl **Online-Change** oder **Laden** auf die Steuerung.

**Siehe auch:**

- [Befehl Online-Change \[► 154\]](#)
- [Befehl Laden \[► 154\]](#)

## 5.2.25 Befehl Refactoring - <Variable> umbenennen

**Funktion:** Der Befehl öffnet den Dialog **Umbenennen** zum projektweiten Umbenennen eines Objekts oder einer Variablen.

**Aufruf:** Kontextmenü SPS-Objekt, Kontextmenü Editorfenster > Refactoring

**Voraussetzung:** Im SPS-Projektbaum ist ein Objekt selektiert oder im Deklarationsteil eines Programmierobjekts ist der Cursor vor oder auf einem Variablenbezeichner positioniert.

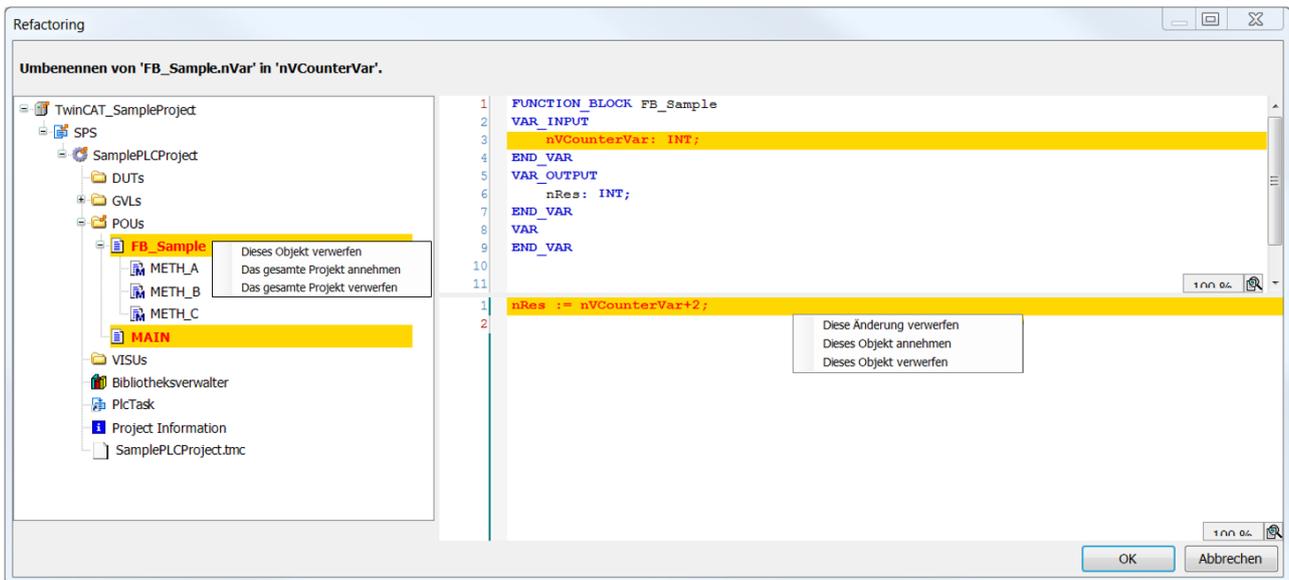
Sie können umbenennen:

- Variablen
- POUs
- GVLs
- Methoden
- Eigenschaften

**Dialog Umbenennen**

Aktueller Name	Name des Objekts oder der Variablen
Neuer Name	Eingabefeld für einen neuen Namen. Wenn der eingegebene Namen bereits existiert, meldet TwinCAT dies direkt unter diesem Eingabefeld.
OK	Aktivierbar, wenn Sie in <b>Neuer Name</b> einen gültigen Namen eingegeben haben. Öffnet den Dialog <b>Refactoring</b> . In beiden Fenstern sind die jeweiligen Objekte und Stellen farblich markiert. In beiden Fenstern können Sie für jede Verwendungsstelle festlegen, was gemacht werden soll. Dazu stehen Ihnen im Kontextmenü verschiedene Befehle zur Verfügung.

**Dialog Refactoring**



Der Dialog zeigt alle Verwendungsstellen innerhalb des Projekts. Die jeweiligen Objekte und Stellen sind farblich markiert.

Linker Dialogteil	Navigationsbaum des Projekts mit dem jeweiligen Objekt.
Rechter Dialogteil	Anzeige der jeweiligen Stelle innerhalb eines Objekts, wo der aktuelle Name vorkommt.

Sie können in beiden Fenstern für jede Verwendungsstelle festlegen, was gemacht werden soll. Dazu stehen Ihnen im Kontextmenü die nachfolgend beschriebenen Befehle zur Verfügung.

Diese Änderung verwerfen	Verwerfen der einzelnen Änderung im rechten Dialogteil.
Dieses Objekt annehmen	Annehmen aller Änderungen im betroffenen Objekt
Dieses Objekt verwerfen	Verwerfen aller Änderungen im betroffenen Objekt
Das gesamte Projekt annehmen	Annehmen aller Änderungen im Projekt
Das gesamte Projekt verwerfen	Verwerfen aller Änderungen im Projekt

TwinCAT stellt angenommene Änderungen mit gelbem Hintergrund und verworfene Änderungen mit grauem Hintergrund dar.

**Siehe auch:**

- Dokumentation PLC: Refactoring

## 5.2.26 Befehl Refactoring - Variable hinzufügen

Symbol: 

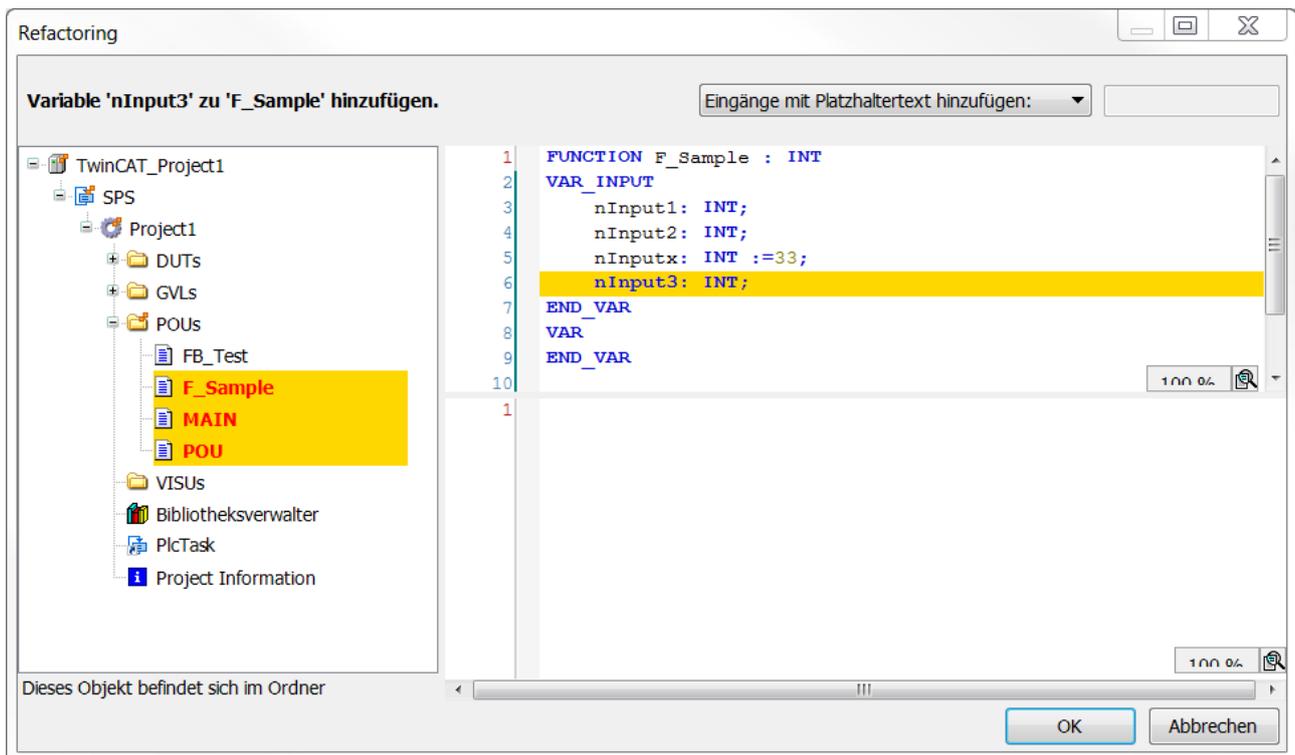
**Funktion:** Der Befehl ermöglicht das Deklarieren einer neuen Variablen in einer POU und die entsprechende automatische Aktualisierung an den Verwendungsstellen der POU.

**Aufruf:** Kontextmenü Editorfenster > Refactoring

**Voraussetzungen:** Der Fokus liegt im Deklarationsteil einer POU.

Der Befehl öffnet den Standarddialog zum Deklarieren der Variablen. Nach Schließen des Deklarationsdialogs mit **OK** erscheint der zweigeteilte Dialog **Refactoring**.

### Dialog Refactoring



<p>Linker Dialogteil</p>	<p>Navigationsbaum des Projekts. Farbliche Kennzeichnung der Bausteine, in denen die POU verwendet wird: rote Schrift und gelb hinterlegt. Nach einem Klick auf das POU-Objekt öffnet sich die Detailansicht im rechten Dialogteil.</p>
<p>Rechter Dialogteil</p>	<p>Deklarationsteil und Implementierung der POU, in deren Deklaration die Variable hinzugefügt wird. Farbliche Kennzeichnung der Änderungsstellen: Neu hinzugefügte Deklaration in blauer Schrift und gelb hinterlegt.</p>

Bevor Sie entscheiden, welche Änderungen Sie an welchen Stellen übernehmen wollen, wählen Sie die gewünschte Option aus der Auswahlliste rechts oben:

<p>Eingänge mit Platzhaltertext hinzufügen</p>	<p>Standardplatzhaltertext: <code>_REFACTOR_</code>; editierbar Der hier definierte Platzhaltertext erscheint an den Verwendungsstellen der neu hinzugefügten Variablen im Implementierungscode. Er dient der Suche nach den betroffenen Stellen.</p>
<p>Eingänge mit folgendem Wert hinzufügen</p>	<p>Initialisierungswert für die neue Variable</p>

Im Kontextmenü der Änderungsstellen, sowohl im linken als auch rechten Teil des Dialogs gibt es Befehle zum Annehmen oder Ablehnen der Änderung(en). Sehen Sie hierzu auch die Beschreibung zu [Befehl Refactoring - <Variable> umbenennen \[► 78\]](#).

### Beispiele:

1. Die Funktion F\_Sample erhält über Refactoring eine neue Eingangsvariable „nInput3“ mit Initialisierungswert „1“. Die Änderung wirkt sich folgendermaßen aus:

Vorher:

```
F_Sample(nVarA + nVarB, 3, TRUE);
F_Sample(nInput1:= nVarA + nVarB, nInput2 :=3 , nInputx := TRUE);
```

Nachher:

```
F_Sample(nVarA + nVarB, 3, TRUE, nInput3 := 1);
F_Sample(nInput1:= nVarA + nVarB, nInput2 :=3 , nInputx := TRUE, nInput3 := 1);
```

2. Die Funktion F\_Sample erhält über Refactoring eine neue Eingangsvariable „nInput3“ mit Platzhaltertext „\_REFACTOR\_“:

Vorher:

```
F_Sample(nInput1 := nVarA + nVarB, nInput2 := 3, nInputx := TRUE);
F_Sample(nVarA + nVarB, 3, TRUE);
```

Nachher:

```
F_Sample(nInput1 := nVarA + nVarB, nInput2 := 3, nInputx := TRUE, nInput3 := _REFACTOR_);
F_Sample(nVarA + nVarB, 3, TRUE, nInput3 := _REFACTOR_);
```

### Siehe auch:

- Dokumentation PLC: Refactoring
- Dokumentation PLC: [Dialog Variable deklarieren \[► 68\]](#)

## 5.2.27 Befehl Refactoring - <Variable> entfernen

Symbol: 

**Funktion:** Der Befehl entfernt eine Eingangs- oder Ausgangsvariable aus der POU und allen Verwendungsstellen der POU.

**Aufruf:** Kontextmenü Editorfenster > Refactoring

**Voraussetzungen:** Der Cursor steht im Bezeichner der zu entfernenden Variablen im Deklarationsteil der POU.

Der Befehl öffnet zunächst einen Dialog mit den Angaben zur gewünschten Entfernung. Nach dessen Bestätigung erscheint der Dialog **Refactoring**. Eine Beschreibung des Dialogs finden Sie im Abschnitt [„Befehl Refactoring - Variable hinzufügen \[► 80\]“](#).

Wenn Sie die Änderungen in Dialog **Refactoring** akzeptieren, werden an den Verwendungsstellen der betroffenen POU die entsprechenden Eingangs- oder Ausgangsparameter gelöscht.



Im CFC wird nur die Verbindung des entfernten Eingangs oder Ausgangs zum Baustein entfernt. Der Eingang oder Ausgang selbst bleibt im Chart enthalten.

### Beispiel in ST:

Sie entfernen in einer POU über **Refactoring** die Eingangsvariable „nInput4“. An den Verwendungsstellen erfolgt eine automatische Anpassung:

Vor der Entfernung:

```
F_Sample(nInput1 := nVarA + nVarB, nInput2 := 3, nInput4 := 1, nInput5 := TRUE);
F_Sample(nVarA + nVarB, 3, 1, TRUE);
```

Nach der Entfernung:

```
F_Sample(nInput1 := nVarA + nVarB, nInput2 := 3, nInput5 := TRUE);  
F_Sample(nVarA + nVarB, 3, TRUE);
```

**Siehe auch:**

- Dokumentation PLC: Refactoring

## 5.2.28 Befehl Refactoring - Variablen neu ordnen

Symbol: 

**Funktion:** Der Befehl ermöglicht im Deklarationseditor das Ändern der Reihenfolge der Variablen im gerade fokussierten Gültigkeitsbereich VAR\_INPUT, VAR\_OUTPUT oder VAR\_IN\_OUT.

**Aufruf:** Kontextmenü des gerade fokussierten Gültigkeitsbereichs im Deklarationseditor

**Voraussetzung:** Der Fokus steht in der Deklaration eines der oben genannten Gültigkeitsbereiche und es ist mehr als eine Variable darin deklariert.

Der Befehl öffnet den Dialog **Neu ordnen** mit einer Liste aller Deklarationen des gerade fokussierten Gültigkeitsbereichs. Durch Ziehen mit der Maus können Sie eine selektierte Deklaration nach oben oder unten an eine andere Position ziehen.

**Siehe auch:**

- Dokumentation PLC: Variablen in der Deklaration neu anordnen

## 5.3 Ansicht

### 5.3.1 Befehl Objekt öffnen

Symbol: 

**Funktion:** Der Befehl öffnet das Objekt in seinem Editor.

**Aufruf:** Menü **Ansicht**, Kontextmenü SPS-Objekt, Doppelklick auf das SPS-Objekt

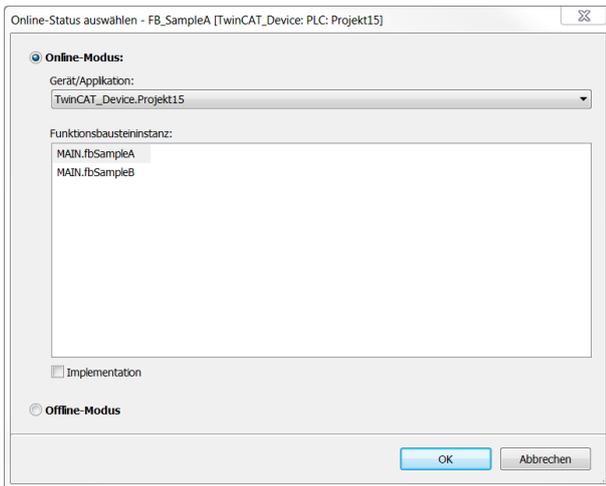
**Voraussetzung:** Ein Objekt im SPS-Projektbaum ist selektiert.

Im Onlinebetrieb öffnet der Dialog **Online Status auswählen**, in dem Sie auswählen können, in welcher Ansicht das Objekt geöffnet werden soll. Der Dialog öffnet nicht, wenn die die Auswahl des Objekts eindeutig ist. Dann wird das Objekt direkt im Online-Modus geöffnet.

#### Dialog Online Status auswählen

**Funktion:** Der Dialog legt fest, wie ein Objekt (Funktionsblock etc.) im Online-Modus geöffnet werden soll, das im Offline-Modus noch nicht geöffnet war. Sie können auswählen, ob eine Instanz oder die grundlegende Implementation des Objekts selbst geöffnet werden soll (und dies wahlweise im Online- oder Offline-Modus).

**Voraussetzung:** Im SPS-Projekt existieren mehrere Instanzen des ausgewählten Objekts.



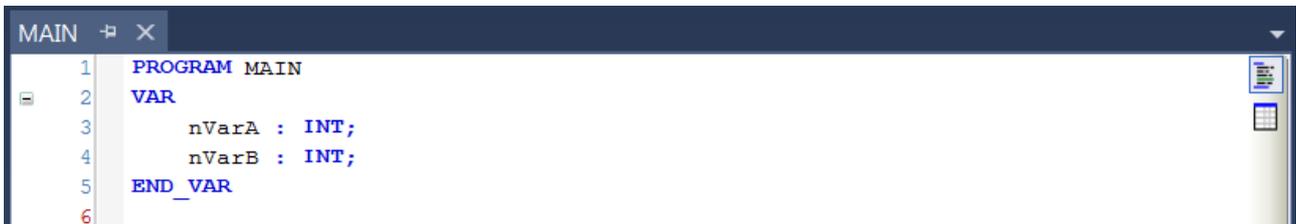
Online-Modus	Aktivieren Sie die Auswahlfläche, um eine Ansicht im Online-Modus zu erhalten.
Gerät/Applikation	Zeigt die Anwendung (das Projekt), denen das Objekt zugeordnet ist.
Funktionsbausteininstanz	Wenn das Objekt ein Funktionsblock ist, erscheint eine Liste aller momentan in der Anwendung verwendeten Instanzen.
Implementation	Wählen Sie diese Option, um die Basisimplementation des Funktionsblocks unabhängig von der gewählten Instanz anzuzeigen. Diese Option hat keine Funktion für nicht-instanzierte Objekte.
Offline-Modus	Aktivieren Sie die Auswahlfläche, um eine Ansicht im Offline-Modus zu erhalten.

### 5.3.2 Befehl Textuelle Ansicht

Symbol:

**Funktion:** Der Befehl öffnet den Deklarationseditor in textueller Ansicht.

**Aufruf:** Schaltfläche am rechten Rand des Editors



**Siehe auch:**

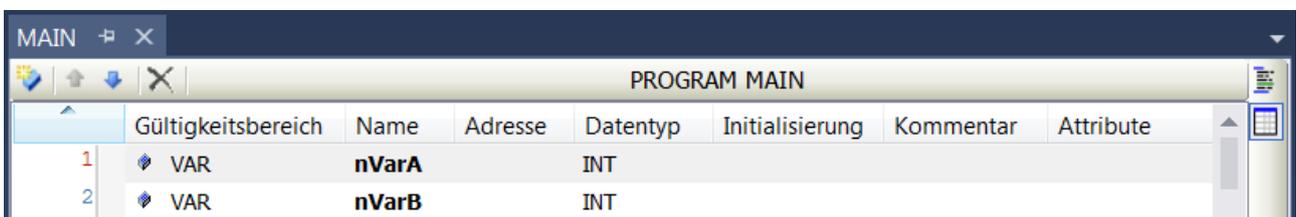
- Dokumentation PLC: Deklarationseditor verwenden

### 5.3.3 Befehl Tabellarische Ansicht

Symbol:

**Funktion:** Der Befehl öffnet den Deklarationseditor in tabellarischer Ansicht.

**Aufruf:** Schaltfläche am rechten Rand des Editors



**Siehe auch:**

- Dokumentation PLC: Deklarationseditor verwenden

### 5.3.4 Befehl Ganzer Bildschirm

Symbol: 

Tastaturkürzel: **[Strg] + [Umschalt] + [F12]**

**Funktion:** Der Befehl schaltet die Anzeige von TwinCAT in den Vollbild-Modus.

**Aufruf:** Menü **Ansicht**

Wenn Sie den Befehl aktivieren, wird das Hauptfenster der TwinCAT-Benutzeroberfläche im Vollbildmodus angezeigt. Sie können zur vorher eingestellten Größe zurückkehren, wenn Sie den Befehl wieder deaktivieren.

### 5.3.5 Befehl Symbolleisten

**Funktion:** Der Befehl öffnet ein Menü zur Auswahl der angezeigten Symbolleisten.

**Aufruf:** Menü **Ansicht**, Kontextmenü Symbolleistenbereich

Markieren Sie in dem sich öffnenden Menü die Symbolleisten, die Sie ein- bzw. ausblenden möchten. Der Befehl funktioniert als Option, d. h. bei einer eingeblendeten Symbolleiste erscheint diese mit einem davor gesetzten Haken im Menü.

**Siehe auch:**

- Dokumentation TC3 User Interface: [Symbolleisten anpassen](#) [▶ 37]

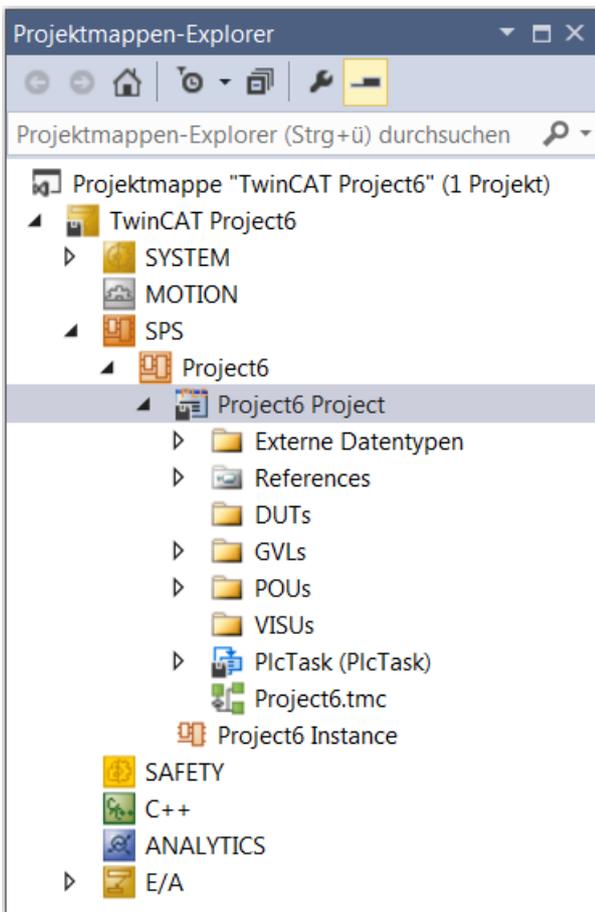
### 5.3.6 Befehl Projektmappen-Explorer

Symbol: 

**Funktion:** Der Befehl öffnet die Ansicht **Projektmappen-Explorer**.

**Ansicht Projektmappen-Explorer**

Die Ansicht **Projektmappen-Explorer** zeigt das TwinCAT-3-Projekt mit den dazugehörigen Projektelementen in strukturierter Form. Sie können in dieser Ansicht Objekte zum Bearbeiten und Konfigurieren öffnen.



### 5.3.7 Befehl Eigenschaftfenster

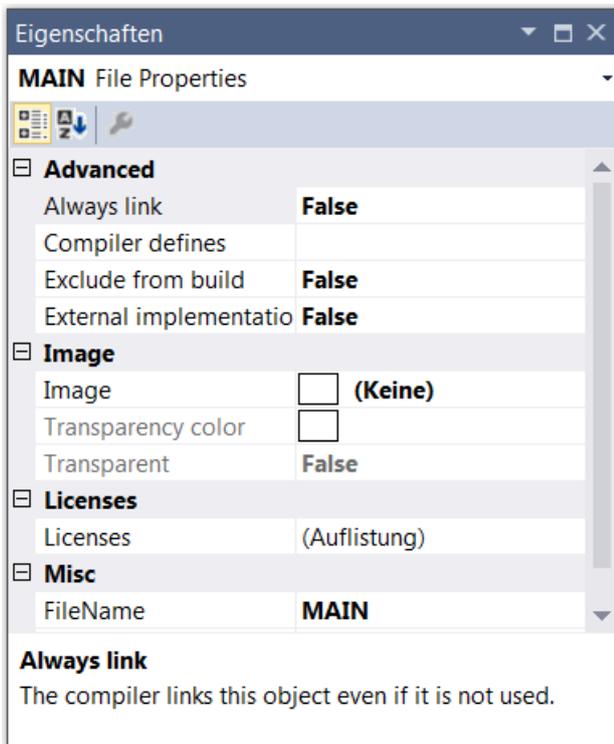
Symbol: 

**Funktion:** Der Befehl öffnet die Ansicht **Eigenschaften**.

**Aufruf:** Menü **Ansicht**

#### Ansicht Eigenschaften

Die Ansicht **Eigenschaften** zeigt die Eigenschaften für das gerade im Projektmappen-Explorer selektierte Objekt. Standardmäßig werden die Elementeigenschaften nach Kategorien sortiert in einer Tabelle dargestellt. Durch einen Mausklick auf die Plus- bzw. Minuszeichen vor der Kategorie können Sie die zugehörigen Parameter einblenden oder ausblenden. Durch einen Mausklick auf das Wertefeld eines Parameters gelangen Sie in den Eingabemodus und können den Wert bzw. die Eigenschaft bearbeiten. Sie können die Ansicht der Eigenschaften filtern oder sortieren.



Sie können das Eigenschaftfenster auch über das Kontextmenü eines Objekts im SPS-Projektbaum aufrufen. Eine Beschreibung des Befehls sowie verschiedener Objekteigenschaften finden Sie im Abschnitt [Befehl Eigenschaften \(Objekt\) \[► 94\]](#).

### 5.3.8 Befehl Werkzeugkasten

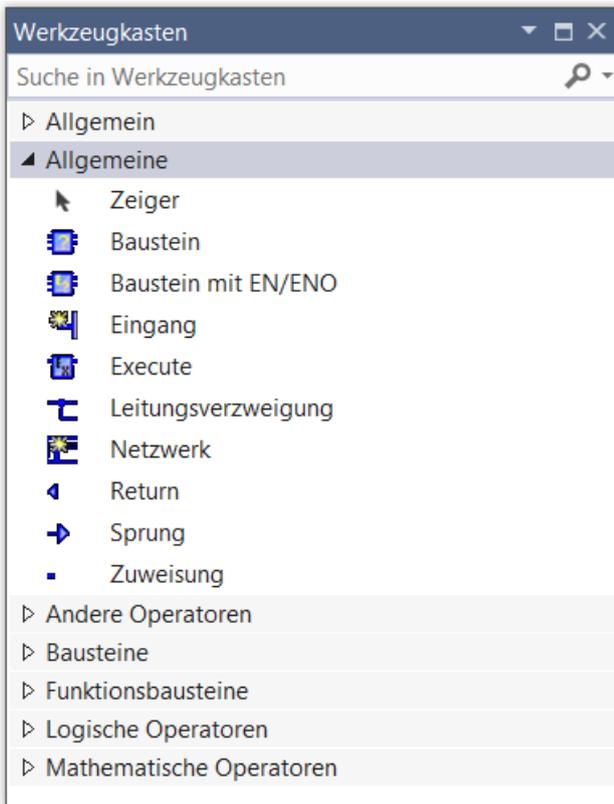
Symbol:

**Funktion:** Der Befehl öffnet die Ansicht **Werkzeugkasten**.

**Aufruf:** Menü **Ansicht**

#### Ansicht Werkzeugkasten

Die Ansicht **Werkzeugkasten** zeigt die vorhandenen „Werkzeuge“ für den gerade aktiven Editor. Diese Ansicht steht Ihnen standardmäßig bei einem grafischen Editor oder einer Visualisierung zur Verfügung. Sie enthält die grafischen Programmiererelemente, die Sie über drag-and-drop in das Editorfenster ziehen können.



### 5.3.9 Befehl Fehlerliste

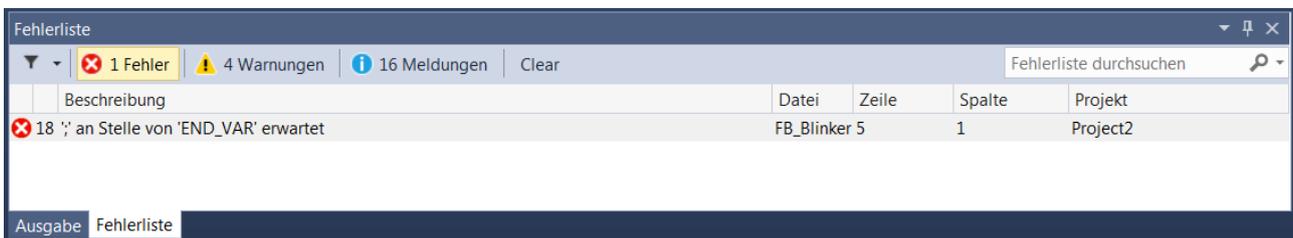
Symbol:

**Funktion:** Der Befehl öffnet die Ansicht **Fehlerliste**.

**Aufruf:** Menü **Ansicht**

#### Ansicht Fehlerliste

Die Ansicht **Fehlerliste** zeigt Fehler, Warnungen und Meldungen bezüglich Syntaxprüfung, Kompilervorgang (Übersetzungsfehler, Code-Größe), Importprozessen oder Bibliotheksverwalter. Die Meldungen werden in Tabellenform ausgegeben.



Filter	Drop-down-Liste zur Auswahl des zu verwendenden Satz von Codedateien: Geöffnete Dokumente Aktuelles Projekt Aktuelles Dokument
Meldungskategorien	Klicken Sie auf das Symbol der Meldungskategorie, um Meldungen ein- oder auszublenden. Neben jedem Symbol zeigt TwinCAT die Anzahl der aufgetretenen Meldungen an.
<ul style="list-style-type: none"> <li>•  : Fehler</li> <li>•  : Warnung</li> <li>•  : Information</li> </ul>	
Clear	Löscht die Meldungsanzeige
	Löscht die Filtereinstellungen der Fehlerliste
Schweregrad (Meldungskategorie)	Meldungstext mit dem verursachenden Objekt und der Position innerhalb des Objekts.
Code	Doppelklicken Sie auf einen Meldungseintrag in der Tabelle, um zur Quelltextposition zu gelangen.
Beschreibung	
Projekt	
Datei	
Zeile	

**Befehle im Kontextmenü**

Clear	Löscht die Meldungsanzeige
Spalten einblenden	Hinzufügen von weiteren Spalten, welche einen Fehler näher beschreiben können
 Kopieren	Kopieren der ausgewählten Fehlermeldung
Nächster Fehler	Selektiert die nächste Meldung. Die Quelltextposition des nächsten Fehlers wird angezeigt.
Vorheriger Fehler	Selektiert die vorherige Meldung. Die Quelltextposition des vorherigen Fehlers wird angezeigt.

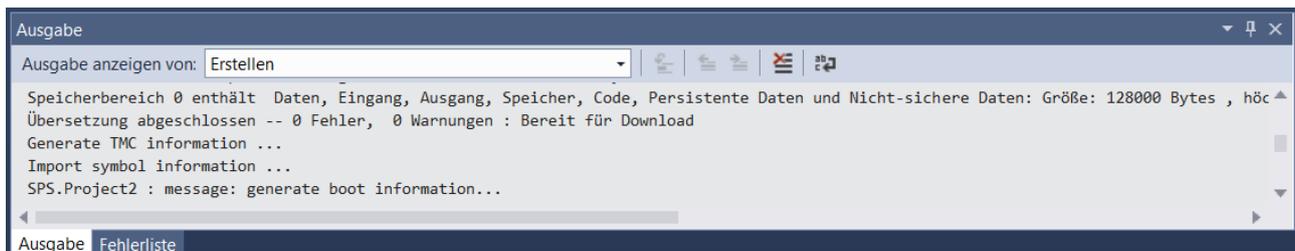
**5.3.10 Befehl Ausgabe**

Symbol: 

**Funktion:** Der Befehl öffnet die Ansicht **Ausgabe**.

**Aufruf:** Menü **Ansicht**

**Ansicht Ausgabe**



Meldungskategorie	Die Meldungen sind nach Komponente oder Funktionalität kategorisiert und in einem Auswahldialog verfügbar. Filtern Sie die Anzeige der Meldungen durch die Wahl einer Kategorie.
 Meldung im Code suchen	Die Quelltextposition der Meldung wird angezeigt. Voraussetzung: Eine Meldung ist markiert.
 Gehe zur vorherigen Meldung	Die vorherige Meldung wird ausgewählt.
 Gehe zur nächsten Meldung	Die nächste Meldung wird ausgewählt.
 Alle löschen	Löscht alle Meldungen
 Zeilenumbruch umschalten	Zeilenumbruch wird ein bzw. ausgeschalten

**Befehle im Kontextmenü**

 Kopieren	Meldungstext wird kopiert
 Alle Löschen	Löscht alle Meldungen
 Gehe zu Speicherort	Die Quelltextposition der Meldung wird angezeigt. Voraussetzung: Eine Meldung ist markiert.
 Gehe zu nächster Position	Die nächste Meldung wird ausgewählt.
 Gehe zu vorheriger Position	Die vorherige Meldung wird ausgewählt.

## 5.4 Projekt

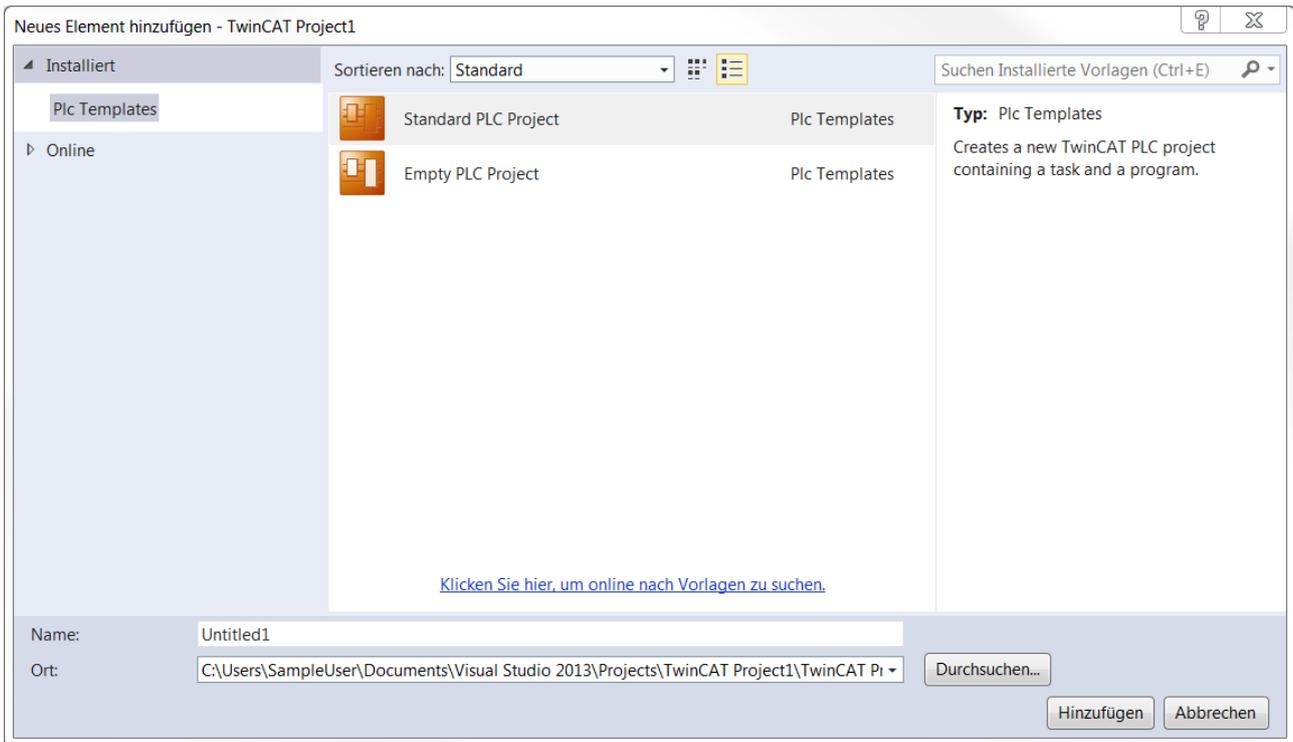
### 5.4.1 Befehl Neues Element hinzufügen (Projekt)

Symbol: 

**Funktion:** Der Befehl öffnet den Dialog **Neues Element hinzufügen**, über den eine neue SPS-Projektdatei angelegt werden kann. (Der Befehl steht nur bei ausgewähltem SPS-Knoten zur Verfügung.)

**Aufruf:** Menü **Projekt** oder Kontextmenü SPS-Objekt im **Projektmappen-Explorer**

**Voraussetzung:** Der SPS-Knoten ist im TwinCAT-Projektbaum selektiert.



<p>Plc Templates</p>	<p>Wählen Sie eine der aufgelisteten Vorlagen aus. Die Vorlage bestimmt die Basiskonfiguration einer SPS-Projektdatei. Folgende Vorlagen stehen standardmäßig zur Verfügung:</p> <ul style="list-style-type: none"> <li>• Standard PLC Project: Erstellt ein neues TwinCAT-SPS-Projekt (*.project). Das Projekt wird von einem Wizard unterstützt und enthält einen Bibliotheksverwalter, ein Programm POU „MAIN“ und eine referenzierte Task.</li> <li>• Empty PLC Project: Erstellt ein „leeres“ TwinCAT-SPS-Projekt für Bibliotheksprojekte (*.library). Das Projekt enthält keine Objekte oder Geräte.</li> </ul>
<p>Name</p>	<p>Definieren Sie hier den Namen des neuen Projekts. Der Default-Name ist von der gewählten Vorlage (üblicherweise „Unbenannt&lt;n&gt;“) abhängig und enthält eine fortlaufende Nummer, um die Eindeutigkeit des Projektnamens im Dateisystem zu gewährleisten. Den vorgegebenen Namen können Sie unter Berücksichtigung der Dateipfad-Konventionen des lokalen Betriebssystems ändern. Eine Dateierweiterung (z. B. .project) kann hinzugefügt werden. Standardmäßig wird durch die gewählte Vorlage automatisch die passende Erweiterung ergänzt.</p>
<p>Ort</p>	<p>Legen Sie hier den Speicherort für die neue Projektdatei fest. Der standardmäßig eingestellte Speicherpfad ist von der gewählten Vorlage abhängig. Sie können entweder über die Schaltfläche <b>Durchsuchen...</b> den Standard-Browser öffnen und einen Speicherpfad bestimmen oder in der zugehörigen Drop-down-Liste aus der Historie bereits eingegebener Speicherpfade einen auswählen.</p>
<p>Hinzufügen</p>	<p>Wenn Sie auf <b>Hinzufügen</b> klicken, wird ein neues Projekt entsprechend der vorgenommenen Einstellungen angelegt. Wenn der Cursor auf ein Fehlersymbol gesetzt wird, gibt ein Tooltip Hinweise, wie Sie weiter vorgehen müssen. Wenn bereits ein anders SPS-Projekt geöffnet ist, öffnet sich ein Dialog, indem Sie gefragt werden, ob das Projekt gespeichert und geschlossen werden soll, bevor das neue Projekt geöffnet wird. Der Name des neuen Projekts wird anschließend in der Titelleiste des TwinCAT-XAE-Rahmenfensters angezeigt. Ein Sternchen („*“) hinter dem Namen bedeutet, dass das Projekt seit dem letzten Speichern verändert wurde.</p>

**Siehe auch:**

- Dokumentation PLC: Ihr erstes TwinCAT-3-SPS-Projekt
- Dokumentation PLC: SPS-Projekt anlegen und konfigurieren

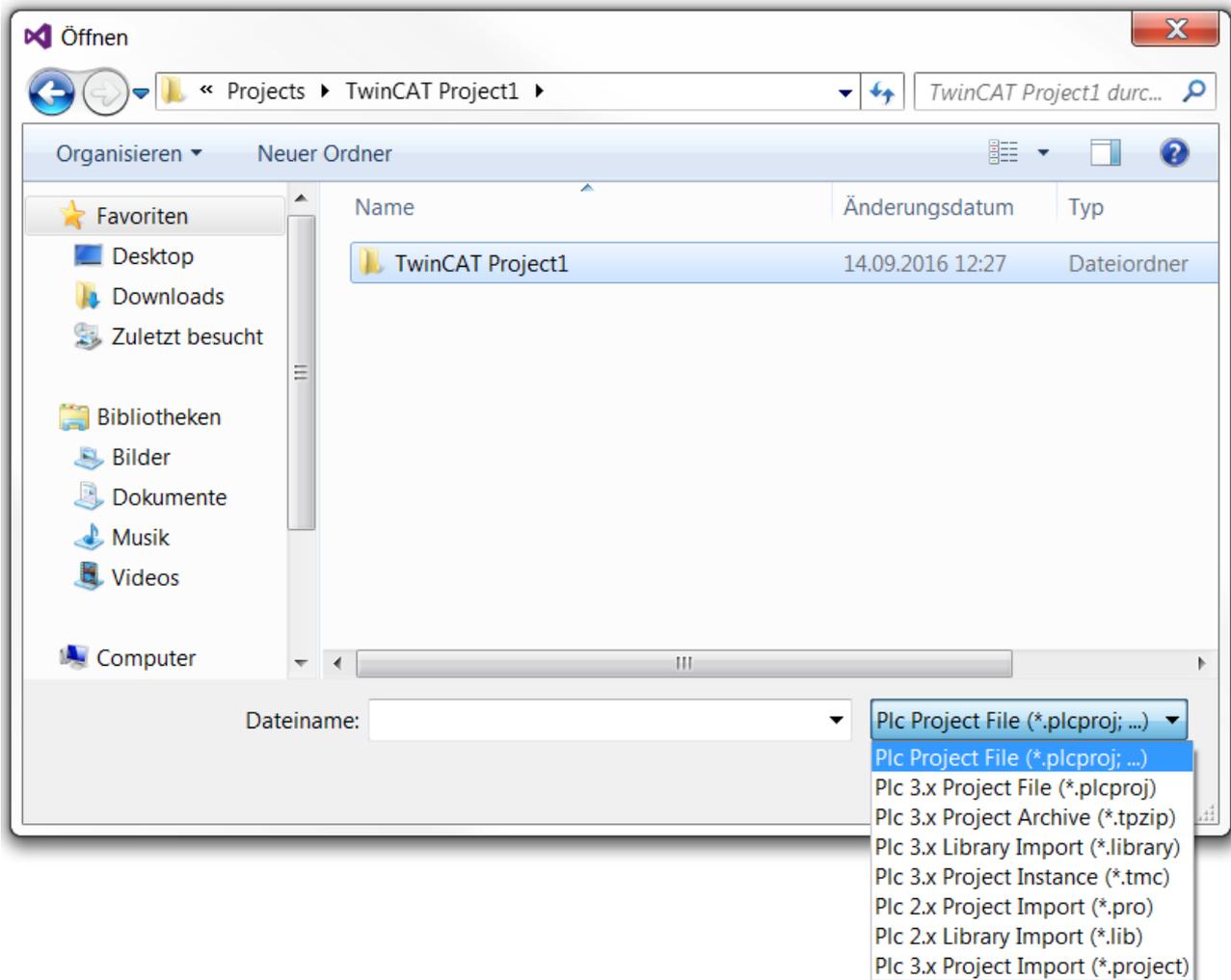
### 5.4.2 Befehl Vorhandenes Element hinzufügen (Projekt)

Symbol: 

**Funktion:** Der Befehl öffnet den Standard-Browserdialog, über den eine SPS-Projektdatei gesucht und im Programmiersystem geöffnet werden kann. Wenn ein entsprechender Konverter installiert ist, können auch Projekte in einem anderen Format geöffnet werden.

**Aufruf:** Menü **Projekt** oder Kontextmenü SPS-Objekt im **Projektmappen-Explorer**

**Voraussetzung:** Der SPS-Knoten ist im TwinCAT-Projektbaum selektiert.



<b>Dateityp</b>	<p>Standardmäßig können Sie den Filter auf einen der folgenden Dateitypen setzen:</p> <ul style="list-style-type: none"> <li>• PLC 3.x Projektdatei (*.PLCproject): TwinCAT-3-PLC-Projekte mit der Erweiterung „PLCproject“</li> <li>• PLC 3.x Projektarchiv (*.tpzip): TwinCAT-3-PLC-Projektarchive mit der Erweiterung „tpzip“             <ul style="list-style-type: none"> <li>◦ Siehe auch: <a href="#">Befehl Sichern &lt;SPS-Projektname&gt; als Archiv... [► 54]</a></li> </ul> </li> <li>• PLC 3.x Bibliotheksimport (*.library): TwinCAT-3-PLC-Bibliotheken mit der Erweiterung „library“,</li> <li>• PLC 2.x Projektdatei (*.pro): TwinCAT-2-PLC-Projekte mit der Erweiterung „pro“</li> <li>• PLC 2.x Bibliotheksimport (*.lib): TwinCAT-2-PLC-Bibliotheken mit der Erweiterung „lib“</li> <li>• PLC 3.x Projektimport (*.PLCproject): PLC-Projekte mit der Erweiterung „project“</li> </ul>
<b>Öffnen</b>	Die gewählte Projektdatei wird geöffnet bzw. konvertiert und dann geöffnet.

**SPS-Projektarchiv \*.tpzip**

<b>Inhalt vom *.tpzip</b>	Der Archivordner *.tpzip enthält das SPS-Projekt, welches archiviert wird.
<b>Befehl zum Erstellen</b>	Ein tpzip-Archiv kann über den folgenden Befehl erstellt werden: Befehl Sichern <SPS-Projektname> als Archiv... [▶ 54]
<b>Hinweis zu SPS-Projekten</b>	Die Dateien und Ordner, die bezüglich des SPS-Projekts in dem Archivordner gespeichert werden, sind abhängig von den SPS-Projekteinstellungen dieses SPS-Projekts. Registerkarte Settings [▶ 122]

**Mögliche Szenarien beim Öffnen eines SPS-Projekts**

Folgende Szenarien sind beim Öffnen eines Projekts möglich:

1. [Ein anderes Projekt ist noch geöffnet.](#) [▶ 92]
2. [Das Projekt wurde mit einer älteren TwinCAT-3-Version gespeichert.](#) [▶ 92]
3. [Das Projekt wurde nicht mit TwinCAT 3 gespeichert.](#) [▶ 92]
4. [Das Projekt wurde nicht regulär beendet und „Automatisch Speichern“ war aktiviert.](#) [▶ 94]
5. [Das Projekt ist schreibgeschützt.](#) [▶ 94]
6. [Es handelt sich um eine Bibliothek, die in einem Bibliotheks-Repository installiert ist und aus diesem aufgerufen wird.](#) [▶ 94]

**1. Ein anderes Projekt ist noch geöffnet.**

Sie werden gefragt, ob das andere Projekt gespeichert und geschlossen werden soll.

**2. Das Projekt wurde mit einer älteren TwinCAT-3-Version gespeichert.**

Wenn sich das Speicherformat unterscheidet, weil das geöffnete Projekt mit einer älteren Version von TwinCAT 3 gespeichert wurde, gibt es zwei Fälle:

- Wenn das Projekt nicht im Speicherformat des aktuell verwendeten Programmiersystems speicherbar ist, müssen Sie es aktualisieren, um weiter am Projekt arbeiten zu können. Der an dieser Stelle auftretende Ausdruck **Die durchgeführten Änderungen...** bezieht sich auf interne Aktionen verschiedener Komponenten während des Ladens des Projekts.
- Wenn das Projekt im bisherigen Speicherformat weiterhin speicherbar ist, können Sie entscheiden, ob das Speicherformat beibehalten oder aktualisiert werden soll. Wenn das Speicherformat beibehalten werden soll, muss mit möglichen Datenverlusten gerechnet werden. Wenn das Speicherformat aktualisiert werden soll, kann das Projekt nicht mehr mit der alten Version des Programmiersystems geöffnet werden.

Neben dem Speicherformat können sich auch die Versionen der explizit eingefügten Bibliotheken, des Visualisierungsprofils und die Compiler-Version des öffnenden Projekts von denen unterscheiden, die mit dem aktuellen Programmiersystem installierten wurden.

Wenn auf dem aktuellen Programmiersystem neuere Versionen installiert sind, öffnet sich automatisch der Dialog **Projektumgebung**, in dem Sie die Versionen aktualisieren können. Wenn an dieser Stelle noch keine Aktualisierung vorgenommen wird, kann dies jederzeit im Dialog **Optionen > Projektumgebung** nachgeholt werden.

---

**● Compiler-Version beachten**

**I** Wenn ein Projekt geöffnet wird, das mit einer älteren Version des Programmiersystems erstellt wurde und für das in den Projekteinstellungen die neueste Compiler-Version eingestellt ist, während im neuen Programmiersystem die Projektumgebungseinstellung für die Compiler-Version **Nicht aktualisieren** ist, dann wird die zuletzt im alten Projekt verwendete Compiler-Version weiter verwendet (nicht die „Aktuelle“ in der neuen Umgebung).

---

**3. Das Projekt wurde nicht mit TwinCAT 3 gespeichert.**

**Fall 1)**

Wenn Sie beim Auswählen des zu öffnenden Projekts den Dateifilter gezielt setzen und ein entsprechender Konverter verfügbar ist, wird der Konverter automatisch verwendet und das Projekt in das aktuelle Format gebracht. Die Konvertierung läuft konverterspezifisch ab. In der Regel werden Sie aufgefordert, die Behandlung von eingebundenen Bibliotheken oder Geräte-Referenzen zu definieren.

**● TwinCAT-3-Konverter**

**i** Die Anpassung eines TwinCAT-PLC-Control-Projekts an die TwinCAT-3-Syntax kann beim Import nur gelingen, wenn der Konverter das Projekt fehlerfrei übersetzen kann.

Wenn Sie beim Auswählen des zu öffnenden Projekts den Dateityp **Alle Dateien** eingestellt haben, ist kein Konverter aktiviert und es öffnet sich der Dialog **Projekt konvertieren**. In dem Dialog müssen Sie die Konvertierung des Projekts durch Auswählen einer der Optionen explizit anstoßen.

- **In das aktuelle Format konvertieren:** Wählen Sie aus der Auswahlliste den Konverter, der verwendet werden soll (Anwendung zum Konvertieren). Nach dem Konvertieren kann das Projekt nicht mehr in der alten Version geöffnet werden.
- **Ein neues Projekt erzeugen und ein spezielles Gerät hinzufügen:** (Noch nicht implementiert)

**● TwinCAT 2.x PLC-Control-Projektoptionen**

**i** Der in den TwinCAT 2.x PLC-Control-Projektoptionen eingestellte Projektverzeichnispfad sowie die Projektinformationen werden in den Dialog **Projektinformationen** übernommen.

**Fall 2)**

Wenn im Projekt Bibliotheken eingebunden sind, für die noch kein „Konvertierungs-Mapping“ in den Bibliotheks-Optionen gespeichert ist, erscheint der Dialog **Konvertierung einer Bibliotheksreferenz**, in dem Sie definieren, wie diese Referenz konvertiert werden sollen:

- **Die Bibliothek ebenfalls konvertieren und installieren:** Wenn Sie diese Option aktivieren, wird die eingebundene Bibliothek in das neue Format übergeführt und bleibt im Projekt referenziert. Sie wird automatisch im Bibliotheks-Repository in der Kategorie **Sonstige** installiert und weiterhin verwendet. Wenn die Bibliothek nicht die für eine Installation nötigen Projektinformationen (Titel, Version) mitbringt, werden Sie aufgefordert, diese im Dialog **Projektinformationen eingeben** nachzutragen.
- **Die folgende bereits installierte Bibliothek verwenden:** Wenn Sie die Optionen aktivieren, wird die eingebundene Bibliothek durch eine andere ersetzt, die bereits auf dem lokalen System installiert ist. Mit der Schaltfläche **Auswählen** öffnen Sie den Dialog **Auswählen...** Hier können Sie die gewünschte Version einer der installierten Bibliothek auswählen. Dies entspricht der Konfiguration des Versions-Handlings im Dialog **Bibliothekseigenschaften**. Ein Sternchen („\*“) bedeutet, dass immer die neueste Version der Bibliothek, die auf dem System verfügbar ist, im Projekt verwendet wird. Die Liste der verfügbaren Bibliotheken ist genauso strukturiert, wie im Dialog **Bibliotheks-Repository**. Sie können die Auflistung nach Firma und Kategorie sortieren.
- **Die Bibliothek ignorieren. Die Referenz wird im konvertierten Projekt nicht erscheinen:** Wenn Sie diese Option aktivieren, wird die Bibliotheksreferenz entfernt. Die Bibliothek ist dann nicht mehr im konvertierten Projekt eingebunden.
- **Dieses Mapping auch zukünftig verwenden, wenn diese Bibliothek auftritt:** Wenn Sie diese Option aktivieren, werden die hier im Dialog vorgenommen Einstellungen auch für künftige Projekt-Konvertierungen angewendet, sobald die betreffende Bibliothek referenziert ist.

Im konvertierten Projekt sind die Bibliotheksreferenzen im globalen Bibliotheksverwalter im Projektmappen-Explorer definiert. Nach Abschluss der Konvertierung der Bibliotheks-Referenzen wird die Projekt-Konvertierung wie oben beschrieben mit Dialog **Projekt öffnen** weitergeführt.

Generelle Informationen zur Bibliotheksverwaltung finden Sie in der PLC-Dokumentation im Abschnitt „Bibliotheken verwenden“.

**Fall 3)**

Wenn Sie ein TwinCAT 2.x PLC-Control-Projekt öffnen, das ein Gerät (Zielsystem) referenziert, für das noch kein „Konvertierungs-Mapping“ in den TwinCAT 2.x PLC-Control-Konverter-Optionen definiert ist, öffnet der Dialog **Gerätekonvertierung**, in dem Sie festlegen können, ob und wie die alten Geräte-Referenzen durch aktuellere ersetzt werden sollen. Das ursprünglich verwendete Gerät wird angezeigt. Wählen Sie eine der folgenden Optionen:

- **Das folgende bereits installierte Gerät verwenden:** Öffnen Sie mit der Schaltfläche **Auswählen** den Dialog **Zielsystem auswählen**, in dem Sie eines der aktuell auf dem System installierten Geräte auswählen können. Dieses Gerät wird dann anstelle des alten im **Projektmappen-Explorer** des konvertierten Projekts eingefügt. Aktivieren Sie Option **Wählen Sie ein Zielsystem aus...**, um eines der aufgelisteten Geräte auswählen zu können. Die Liste der verfügbaren Geräte ist genauso strukturiert wie im Dialog **Device-Repository**. Sie können die Auflistung nach Hersteller und Kategorie sortieren.
- **Das Gerät ignorieren. Alle applikationsspezifischen Objekte werden nicht verfügbar sein:** Wenn Sie diese Option aktivieren, wird im **Projektmappen-Explorer** des neuen Projekts kein Eintrag für das Gerät angelegt, d.h. das Gerät wird bei der Konvertierung ignoriert und auch applikationsspezifische Objekte wie z.B. die Taskkonfiguration werden nicht übernommen.
- **Diese Zuordnung für die Zukunft speichern:** Wenn Sie diese Option aktivieren, werden alle Einstellungen des Dialogs, d.h. das dargestellte „Konvertierungs-Mapping“ für das Gerät, in den TwinCAT 2.x PLC-Control-Konverter-Optionen gespeichert und für künftige Konvertierungen angewendet.

#### 4. Das Projekt wurde nicht regulär beendet und Automatisch Speichern war aktiviert.

Wenn die Funktion **Automatisch Speichern** in den Optionen **Laden und Speichern** aktiviert war und TwinCAT 3 PLC, nach der letzten Änderung des Projekts ohne Speichern, nicht regulär beendet worden war, öffnet der Dialog **Auto Save Backup** zur Handhabung der Sicherungskopie.

#### 5. Das Projekt ist schreibgeschützt.

Ist das Projekt, das geöffnet werden soll, schreibgeschützt, so werden Sie gefragt, ob Sie das Projekt in schreibgeschütztem Modus öffnen oder ob den Schreibschutz aufheben wollen.

#### 6. Es handelt sich um eine Bibliothek, die in einem Bibliotheks-Repository installiert ist und aus diesem aufgerufen wird.

Wenn Sie ein Bibliotheksprojekt öffnen möchten, das in einem Bibliotheks-Repository installiert ist, wird eine Fehlermeldung ausgegeben. Ein Bibliotheksprojekt können Sie über diesen Pfad nicht öffnen. Nach Schließen des Dialogs mit **OK** erscheint der Projektname in der Titelzeile der Benutzeroberfläche. Ein Sternchen („\*“) hinter dem Namen bedeutet, dass das Projekt seit dem letzten Speichern verändert wurde.

#### Siehe auch:

- Dokumentation PLC: TwinCAT-3-SPS-Projekt öffnen
- Dokumentation PLC: TwinCAT-2-SPS-Projekt öffnen

### 5.4.3 Befehl Eigenschaften (Objekt)

**Funktion:** Der Befehl aktiviert die Ansicht **Eigenschaften**, die allgemeine Informationen zum gerade selektierten Objekt anzeigt.

**Aufruf:** Kontextmenü SPS-Objekt

**Voraussetzung:** Ein Objekt im SPS-Projektbaum ist selektiert.

In Abhängigkeit vom gerade selektierten Objekt, werden folgende Eigenschaftsbereiche angezeigt:

- [Erweitert \[► 95\]](#) (Übersetzungseinstellungen)
- [Image \[► 95\]](#)
- [Licenses \[► 95\]](#)
- [Allgemein \[► 95\]](#) (Objektname, Objektpfad)
- [AS-Einstellungen \[► 98\]](#) (Flags für Ablaufsprache)
- [CFC-Einstellungen \[► 99\]](#) (Ausführungsreihenfolge-Modus)



Die speziellen Visualisierungseigenschaften sind unter Visualisierungsobjekt und die speziellen Bibliotheks- und Platzhaltereigenschaften sind unter Befehl Eigenschaften dokumentiert.

**Erweitert**

Dieser Bereich zeigt die Einstellungen bezüglich der Kompilation (Übersetzen) des Objekts.

<b>Advanced</b>	
Always link	<b>False</b>
Compiler defines	
Exclude from build	<b>False</b>
External implementation	<b>False</b>

Immer binden	True: Das Objekt ist beim Compiler markiert und damit immer in der Compile-Information enthalten. Es wird somit immer kompiliert und auf die SPS geladen. Diese Option wird dann relevant, wenn das Objekt unterhalb einer Applikation liegt oder über Bibliotheken referenziert wird, die ebenfalls unterhalb einer Applikation liegen. Die Compile-Information wird ebenfalls als Basis für die auswählbaren Variablen der Symbolkonfiguration verwendet.  Alternativ kann mittels des Pragmas {attribute 'linkalways'} der Compiler angewiesen werden, ein Objekt immer einzubinden.
Compilerdefinitionen	Die hier eingetragenen Compilerdefinitionen werden nicht ausgewertet. Wenn Sie Compilerdefinitionen verwenden möchten, tragen Sie diese in den SPS-Projekteigenschaften ein.  Siehe: <ul style="list-style-type: none"> <li>• Befehl Eigenschaften (SPS-Projekt) &gt; Kategorie Übersetzen  ▶ 103 </li> <li>• Dokumentation PLC: Referenz Programmierung &gt; Pragmas &gt; Bedingte Pragmas</li> </ul>
Vom Übersetzen ausschließen	True: Das Objekt wird beim nächsten Übersetzungslauf nicht berücksichtigt.
Externe Implementierung	(Spätes Binden im Laufzeitsystem)  Die Verwendung dieser Funktionalität ist lediglich in speziellen Konstellationen möglich. In aller Regel können Sie diese Option ignorieren.  True: Beim Übersetzen des Projekts wird für dieses Objekt kein Code generiert. Das Objekt wird erst gelinkt, wenn das Projekt auf das Zielsystem geladen wird, vorausgesetzt, es ist dort vorhanden (im SPS-Laufzeitsystem oder in einem anderen Echtzeitmodul).

**Image**

In diesem Bereich können Sie dem Objekt ein Bild zuweisen, mit dem es in der grafischen Ansicht des Bibliotheksverwalters und im Werkzeugkasten des FUP/KOP/IL-Editors dargestellt wird. Transparenz der Abbildung kann durch Auswahl einer Farbe erreicht werden, die dann transparent dargestellt wird. Wenn Sie die Option **Transparenzfarbe** aktivieren, können Sie über die rechteckige Schaltfläche rechts davon den Standarddialog zur Auswahl einer Farbe öffnen.

<b>Image</b>	
Image	<input type="checkbox"/> <b>(Keine)</b>
Transparency color	<input type="checkbox"/>
Transparent	<b>False</b>

**Licenses**

In diesem Bereich finden Sie eine Auflistung der Lizenzen zum Objekt.

<b>Licenses</b>	
Licenses	(Auflistung)

**Allgemein**

In diesem Bereich finden Sie allgemeine Informationen zum selektierten Objekt.

FileName	Datei-/Objektname
FullPath	Speicherpfad/-ort des Objekts (an dieser Stelle nicht editierbar)
Version	Dateiversion, Werte: <ul style="list-style-type: none"> <li>• 1.1.0.1: Diese Dateiversion wird verwendet, wenn das Objekt im Speicherformat XML gespeichert wird.</li> <li>• 1.2.0.0: Diese Dateiversion wird verwendet, wenn das Objekt im Speicherformat Base64 gespeichert wird. Bitte beachten Sie, dass Objekte mit der Dateiversion 1.2.0.0 (oder größer) nicht mit Engineering-Versionen &lt; TC3.1.4024 geladen werden können!</li> </ul> (an dieser Stelle nicht editierbar, indirekt konfigurierbar über das Speicherformat, siehe Eigenschaft <b>Format</b> )

### **i** Engineering-Inkompatibilität von Dateiversion 1.2.0.0 (oder größer) mit TwinCAT 3.1 < Build 4024

Beachten Sie, dass Objekte, die mit der Dateiversion 1.2.0.0 (oder größer) gespeichert werden, nicht mit Engineering-Versionen < TwinCAT 3.1.4024 geladen werden können!

Da ein Objekt bei Verwendung des optionalen Speicherformats „Base64“ automatisch mit der Dateiversion 1.2.0.0 gespeichert wird, können Objekte mit Base64-Speicherformat folglich nicht mit Engineering-Versionen < TwinCAT 3.1.4024 geladen werden.

Falls ein SPS-Projekt sowohl Objekte mit der Dateiversion 1.1.0.1 als auch Objekte mit der Dateiversion 1.2.0.0 enthält, werden die 1.1.0.1-Objekte mit einer Engineering-Version < TwinCAT 3.1.4024 weiterhin geladen. Lediglich die Objekte mit der Dateiversion 1.2.0.0 werden nicht geladen.

Die Dateiversion einer Datei, die mit der Dateiversion 1.2.0.0 gespeichert wurde, kann mit XAE-Versionen >= TwinCAT 3.1.4024 nachträglich wieder auf 1.1.0.1 gesetzt werden.

### Optionen

In diesem Bereich finden Sie einige Optionen, die für SPS-Objekte konfiguriert werden können.

<p>Format</p>	<p><b>Individuelle Einstellungsmöglichkeit des Speicherformats:</b></p> <p>Das Speicherformat eines Objekts kann für die unten genannten Objekttypen an dieser Stelle individuell konfiguriert werden.</p> <p>Speicherformat, Werte:</p> <ul style="list-style-type: none"> <li>• XML: Das Objekt wird im XML-Format gespeichert.             <ul style="list-style-type: none"> <li>◦ Objekte mit diesem Speicherformat werden in der Dateiversion 1.1.0.1 gespeichert, siehe Eigenschaft <b>Version</b>.</li> </ul> </li> <li>• Base64: Das Objekt wird im Base64-Format gespeichert.             <ul style="list-style-type: none"> <li>◦ Objekte mit diesem Speicherformat werden in der Dateiversion 1.2.0.0 gespeichert, siehe Eigenschaft <b>Version</b>. Bitte beachten Sie, dass Objekte mit der Dateiversion 1.2.0.0 (oder größer) nicht mit Engineering-Versionen &lt; TC3.1.4024 geladen werden können!</li> </ul> </li> </ul> <p><b>Vorteile von Base64 gegenüber XML:</b></p> <p>Im Vergleich zu XML ergibt sich mit Base64 eine komprimierte Speicherung. Als Folge dessen kann bei Dateizugriffen auf diese Objekte eine verbesserte Performance erreicht werden, welche beispielsweise beim Projektladen oder beim Verschieben/Kopieren der Objekte zum Tragen kommt.</p> <p><b>Verfügbarkeit von Base64:</b></p> <p>Das Speicherformat Base64 steht ab Build 4024 für die folgenden SPS-Objekte optional zur Verfügung:</p> <ul style="list-style-type: none"> <li>• POUs, bei denen der POU-Rumpf in einer grafischen Implementierungssprache programmiert ist             <ul style="list-style-type: none"> <li>◦ AS (Ablaufsprache)</li> <li>◦ FUP/KOP/AWL (Funktionsplan/Kontaktplan/Anweisungsliste)</li> <li>◦ CFC (Continuous Function Chart und Seitenorientierter CFC)</li> <li>◦ UML Klassendiagramm und Zustandsdiagramm</li> </ul> </li> <li>• POUs, die über ein Unterelement (z. B. Aktion, Methode) verfügen, das in einer grafischen Implementierungssprache programmiert ist (grafische Sprachen siehe erster Stichpunkt)</li> <li>• Visualisierungen</li> <li>• Visualisierungsmanager</li> <li>• Textlisten</li> <li>• Rezeptmanager</li> <li>• Bildersammlungen</li> </ul> <p><b>Einstellungsmöglichkeit des Standard-Speicherformats:</b></p> <p>Über die Einstellung „Objektinhalt schreiben als“ in den SPS-Projekteigenschaften (<a href="#">Kategorie Advanced [▶ 113]</a>) können Sie für ein SPS-Projekt definieren, welches das standardmäßige Speicherformat für die oben genannten Objekttypen ist.</p>
<p>Separate Linelds</p>	<p>Wert: True oder False</p> <ul style="list-style-type: none"> <li>• True: Die Line-IDs dieser POU werden in einer separaten Datei (LineIDs.dbg) gespeichert.</li> <li>• False: Die Line-IDs dieser POU werden in der POU selbst gespeichert.</li> </ul> <p>(an dieser Stelle nicht editierbar, konfigurierbar in den <a href="#">Write Options [▶ 194]</a>)</p>
<p>Sort</p>	<p>Wert: Name oder GUID</p> <p>Gibt die Art und Weise an, in welcher Reihenfolge die Unterobjekte (z.B. Methoden) in dem übergeordneten Objekt gespeichert werden: entweder sortiert nach dem Namen oder nach der GUID.</p> <p>(an dieser Stelle nicht editierbar, konfigurierbar in den <a href="#">Write Options [▶ 194]</a>)</p>

Write ProductVersion	Wert: True oder False (an dieser Stelle nicht editierbar, konfigurierbar über die Einstellung „Produktversion in Dateien schreiben“ in der <u>Kategorie Advanced</u> [▶ 113] der SPS-Projekteigenschaften)
----------------------	---

**AS-Einstellungen**

In diesem Bereich werden für das gerade ausgewählte AS-Objekt die aktuellen Einstellungen bezüglich des Übersetzens (Kompilieren) und der Handhabung der impliziten Variablen angezeigt.

<b>SFC</b>	
Use default SFC settings	True
<b>SFC Build</b>	
CalculateActiveTransitionOnly	False
<b>SFC Flags</b>	
SFCCurrentStep	Declare
SFCEnableLimit	Declare
SFCError	Declare
SFCErrorAnalyzation	Declare
SFCErrorAnalyzationTable	Declare
SFCErrorPOU	Declare
SFCErrorStep	Declare
SFCInit	Use
SFCPause	Declare
SFCQuitError	Declare
SFCReset	UseDeclare
SFCTip	Declare
SFCTipMode	Declare
SFCTrans	Declare

Voreingestellte AS-Einstellungen nutzen	True (default): Mit dieser Option können die in den SPS-Projekteigenschaften [▶ 107] definierten Standardwerte auf das gerade ausgewählte Objekt angewendet und in der Ansicht <b>Eigenschaften</b> des Objekts dargestellt werden. False: Mit dieser Option können speziell für dieses AS-Objekt gültige AS-Einstellungen konfiguriert werden.
Nur aktive Transitionen berechnen	True: TwinCAT erzeugt nur für momentan aktive Transitionen Code. False (default): TwinCAT erzeugt für alle Transitionen Code. Die Zustände aller Transitionen werden online farblich dargestellt (True in dunkelblau, False in hellblau).
SFC Flags	Implizit erzeugte Variablen (Flags) zur Kontrolle und zum Monitoring der Abarbeitung in einem AS-Diagramm. Use: Die entsprechende Variable wird verwendet. Declare: Die entsprechende Variable wird automatisch angelegt. UseDeclare: Die Variable wird angelegt und verwendet.

CFC-Einstellungen



Verfügbar ab TwinCAT 3.1 Build 4026

In diesem Bereich wird der Ausführungsreihenfolge-Modus für das ausgewählte CFC-Objekt festgelegt. Im CFC-Editor positionieren Sie die Elemente und damit die Netzwerke frei. Um zu vermeiden, dass die Ausführungsreihenfolge im CFC-Programmieraustein unbestimmt ist, stehen zwei Modi zur Verfügung.

CFC

Explicit Execution Order	False
--------------------------	-------

Explicit Execution Order	False (default): Automatischer Datenfluss-Modus True: Expliziter Ausführungsreihenfolge-Modus
--------------------------	--

**Automatischer Datenfluss-Modus**

In diesem Modus wird die Ausführungsreihenfolge automatisch nach Datenfluss und bei Mehrdeutigkeit nach Netzwerktopologie festgelegt. Die Programmierbausteine und die Ausgänge werden intern nummeriert. Dabei werden die oberen vor den unteren und die linken vor den rechten Netzwerken ausgeführt.

Vorteil: Die automatische festgelegte Ausführungsreihenfolge ist zeit- und zyklusoptimiert. Sie benötigen auch während des Entwicklungsprozesses keine Information über die intern verwaltete Ausführungsreihenfolge.

Die Elemente im CFC-Editor werden ohne Marken und ohne Nummerierung angezeigt. Ein manuelles Ändern der Ausführungsreihenfolge ist nicht möglich. Bei Netzwerken mit Rückkopplungen können Sie zusätzlich einen Startpunkt setzen.

Im Menü **CFC > Ausführungsreihenfolge** stehen in diesem Modus folgende Befehle zur Verfügung:

- [Befehl Ausführungsreihenfolge anzeigen \[► 216\]](#)
- [Befehl Startpunkt der Rückkopplung setzen \[► 216\]](#)

**Expliziter Ausführungsreihenfolge-Modus**

In diesem Modus können Sie die Ausführungsreihenfolge explizit festlegen. Dafür werden die Elemente im CFC-Editor mit Marken und Nummerierung angezeigt und Menübefehle bereitgestellt, mit denen Sie die Reihenfolge bestimmen können.

Im Menü **CFC > Ausführungsreihenfolge** stehen folgende Befehle zur Verfügung:

- [Befehl An den Anfang \[► 217\]](#)
- [Befehl Ans Ende \[► 217\]](#)
- [Befehl Eins vor \[► 218\]](#)
- [Befehl Eins zurück \[► 218\]](#)
- [Befehl Ausführungsreihenfolge setzen \[► 218\]](#)
- [Befehl Nach Datenfluss anordnen \[► 219\]](#)
- [Befehl Topologisch anordnen \[► 219\]](#)



Bis Build 4026 war dieses Verhalten das übliche Verhalten von CFC-Programmieraustein. Beachten Sie, dass Sie die Ausführungsreihenfolge in Eigenverantwortung anpassen und die Konsequenzen und Effekte selbst beurteilen müssen. Die Ausführungsreihenfolge wird dafür ständig eingeblendet.

**5.4.4 Befehl Eigenschaften (SPS-Projekt)**

Symbol:

**Funktion:** Dieser Befehl öffnet ein Editorfenster, in dem die Eigenschaften des Projekts und zusätzliche projektbezogene Informationen angezeigt und definiert werden können.

**Aufruf:** Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) oder Menü **Projekt**

**Voraussetzung:** Ein Projekt ist geöffnet.

TwinCAT speichert die SPS-Projekteigenschaften direkt im SPS-Projekt.

---

### ● Geltungsbereich von SPS-Projekteigenschaften



Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

---

**Siehe auch:**

- Dokumentation PLC: Projekt konfigurieren

#### 5.4.4.1 Kategorie Common

Die Kategorie **Common** enthält allgemeine Informationen und Metainformationen der Projektdatei. TwinCAT erstellt aus diesen Informationen Schlüssel in der Registerkarte **Eigenschaften**. Wenn zum Beispiel im Textfeld **Firma** der Name „Company\_A“ eingetragen ist, ist in der Registerkarte **Eigenschaften** der Schlüssel **Company** mit dem Wert „Company\_A“ vorhanden.

#### Projektinformation

Bei einem Bibliotheksprojekt muss eine Firma, ein Titel und eine Version eingetragen sein, um die Bibliothek installieren zu können.	
Firma	Name der Firma, in der dieses Projekt (Applikation oder Bibliothek) erstellt wurde. Er dient neben der Bibliothekskategorie zur Sortierung im Bibliotheks-Repository
Titel	Titel des Projekts
Version	Version des Projekts, zum Beispiel „0.0.0.1“
Freigegeben	<input checked="" type="checkbox"/> : Schutz gegen Änderung aktiviert. Folge: Wenn Sie nun das Projekt editieren, erscheint eine Eingabeaufforderung, ob das Projekt wirklich geändert werden soll. Wenn Sie diese Abfrage einmalig mit Ja beantworten, erscheinen bei weiteren Editieraktionen keine Abfrage mehr.
Bibliothekskategorien	Kategorien des Bibliotheksprojekts, nach denen Sie im Dialog Bibliotheks-Repository sortieren können. Wenn keine Kategorie angegeben ist, wird der Bibliothek die Kategorie „Sonstige“ zugewiesen. Soll sie einer anderen Kategorie angehören, muss eine solche definiert sein.  Die Definition von Bibliothekskategorien erfolgt in einer oder mehreren externen Beschreibungsdateien im XML-Format. Für die Zuordnung der Bibliothek kann entweder eine solche Datei aufgerufen werden oder aber eine andere Bibliotheksdatei, die bereits selbst die Informationen über die Kategorien aus einer Beschreibungsdatei aufgenommen hat.  Voraussetzung: Das Projekt ist ein Bibliotheksprojekt.
	Der Dialog <b>Bibliothekskategorien</b> öffnet sich, in dem Sie Bibliothekskategorien hinzufügen können.
Standard-Namensraum	Die Standardeinstellung für den Namensraum einer Bibliothek entspricht dem Bibliothekstitel. Für eine Bibliothek kann auch explizit ein davon abweichender Namensraum definiert werden: entweder allgemein für die Bibliothek bei der Bibliothekserstellung an dieser Stelle in den Projektinformationen oder für den lokalen Gebrauch der Bibliothek in einem Projekt im <b>Eigenschaften</b> -Dialog der Bibliotheksreferenz.  Der Namensraum der Bibliothek muss als Präfix des Bezeichners verwendet werden, damit ein eindeutiger Zugriff auf ein Modul möglich ist, das mehrfach im Projekt vorhanden ist, oder wenn der Gebrauch dieses Präfixes durch die Bibliothekseigenschaft LanguageModelAttribute „qualified-access-only“ („Eindeutiger Zugriff auf Bibliotheksmodule oder -variablen“) erzwungen wird.  Wenn Sie hier keinen Standardnamensraum definieren, gilt automatisch der Name der Bibliotheksdatei als Namensraum.
Platzhalter	An dieser Stelle kann ein Standardname des Platzhalters festgelegt werden, der diese Bibliothek repräsentiert bzw. referenziert. Falls ein dieser Stelle nicht explizit ein Platzhalter festgelegt wird, entspricht die Standardeinstellung für den Platzhalternamen einer Bibliothek dem Bibliothekstitel.
Autor	Autor des Projekts
Beschreibung	Kurze Beschreibung des Projekts (z. B. Inhalt, Funktionalitäten, allgemeine Hinweise wie nur für den internen Gebrauch etc.)

**Bibliothekseigenschaften**

Globale Versionsstruktur erzeugen	<input checked="" type="checkbox"/> : Im SPS-Projekt eine globale Variablenliste angelegt, welche die Versionsinformationen enthält. Die globale Variablenliste wird beim Speichern aktualisiert.
Automatisch Bibliotheksinformationen-POU erzeugen	<input checked="" type="checkbox"/> : POU-Objekte vom Typ „Funktion“ werden automatisch im Projektbaum angelegt, die verwendet werden können, um im Applikationsprogramm auf die Projekteigenschaften zuzugreifen. In diesem Fall werden spezielle Funktionen für die Eigenschaften <b>Firma</b> , <b>Title</b> und <b>Version</b> erzeugt (F_GetCompany, F_GetTitle, F_GetVersion). Die Funktionen werden beim Speichern aktualisiert.
Dokumentationsformat	Optionen: <ul style="list-style-type: none"> <li>• Bis TwinCAT 3.1 Build 4024: <b>reStructuredText</b></li> <li>• Ab TwinCAT 3.1 Build 4026: <b>TcDocGen</b></li> </ul> Bei der Bibliothekserstellung werden Kommentare, die einem bestimmten Format entsprechen, neustrukturiert und in dieser angepassten Darstellung in der Registerkarte <b>Dokumentation</b> im Bibliotheksverwalter angezeigt. Dadurch ergeben sich zusätzliche Möglichkeiten der Bibliotheksdokumentation.
Implizite Prüfungen für Compiled Libraries erlauben	Verfügbar ab TwinCAT 3.1 Build 4026 <input checked="" type="checkbox"/> : TwinCAT führt implizite Prüfungen auch für Bausteine aus geschützten Bibliotheken (*.compiled-libraries) aus. Voraussetzung: Die Compiler-Definition „checks_in_libs“ ist in den <b>SPS-Projekteigenschaften</b> im Feld <b>Compilerdefinitionen</b> eingetragen (Kategorie <b>Übersetzen</b> ). Siehe auch: Bausteine für implizite Prüfung verwenden
Qualified_only für Bibliothekszugriff erzwingen	Verfügbar ab TwinCAT 3.1 Build 4026 <input checked="" type="checkbox"/> : Objekte aus dieser Bibliothek dürfen nur mit der Angabe des Namensraums der Bibliothek verwendet werden. Siehe auch: Attribut 'qualified_only'
Referenzierung als Bibliothek erlauben	Verfügbar ab TwinCAT 3.1 Build 4026 <input checked="" type="checkbox"/> : Sie können das SPS-Projekt in einem anderen SPS-Projekt als Bibliothek referenzieren. Siehe auch: SPS-Projekt als referenzierte Bibliothek verwenden

**Dialog Bibliothekskategorien**



Liste von Kategorien	Liste der Kategorien, die dem Bibliotheksprojekt zugewiesen sind. Sie können aus mehrere Quellen stammen. Wenn Sie alle gewünschten Kategorien eingetragen haben, bestätigen Sie den Dialog mit <b>OK</b> .
Hinzufügen	Die Befehle <b>Aus Beschreibungsdatei...</b> und <b>Aus anderer Bibliothek...</b> erscheinen.
Entfernen	TwinCAT entfernt die selektierte Kategorie.
Aus Beschreibungsdatei...	Dialog <b>Beschreibungsdatei auswählen</b> erscheint, in dem Sie eine Beschreibungsdatei mit Erweiterung *.libcat.xml auswählen. Die Datei enthält Befehlskategorien. Wenn Sie den Dialog mit <b>Öffnen</b> beenden, übernimmt TwinCAT diese Kategorien.
Aus anderer Bibliothek...	Dialog <b>Bibliothek auswählen</b> erscheint, in dem Sie eine Bibliothek (*.library) auswählen, deren Befehlskategorien übernommen werden sollen. Wenn Sie den Dialog mit <b>Öffnen</b> beenden, übernimmt TwinCAT die Kategorien.
OK	TwinCAT stellt die Kategorien als Projektinformation zur Verfügung und zeigt sie im Feld <b>Bibliothekskategorien</b> an.
Abbrechen	Schließt den Dialog. Der Vorgang wird abgebrochen.

**Siehe auch:**

- Dokumentation PLC: Projekt konfigurieren
- Dokumentation PLC: Bibliotheken verwenden

**5.4.4.2 Kategorie Übersetzen**

### **i** Geltungsbereich von SPS-Projekteigenschaften

Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

Die Kategorie **Übersetzen** dient der Konfiguration der Compileroptionen.

**Einstellungen**

Compilerdefinitionen	<p>Hier können Compilerdefinitionen/„defines“ (siehe {define}-Anweisungen) und Bedingungen für die Kompilation der Applikation eingetragen werden (bedingte Kompilierung).</p> <p>Eine Beschreibung der verfügbaren bedingten Pragmas finden Sie im Abschnitt <b>Bedingte Pragmas</b>. Der Ausdruck <code>expr</code>, der in solchen Pragmas verwendet wird, kann auch hier eingegeben werden. Mehrere Einträge sind in Form einer komma-separierten Liste möglich.</p>
System-Compilerdefinitionen	<p>Verfügbar ab TwinCAT 3.1 Build 4024</p> <p>Hier werden automatisch die Compilerdefinitionen übernommen, die auf System Manager-Ebene in den SPS-Projekteinstellungen unter <a href="#">Compilerdefinitionen</a> [►_121] gesetzt worden sind.</p>
Download Applikationsinfo	<p>Verfügbar ab TwinCAT 3.1 Build 4024</p> <p>Situation: Sie sind dabei, ein SPS-Projekt auf die Steuerung zu laden, das sich von dem bereits dort liegenden unterscheidet. In diesem Fall erscheint ein Meldungsfenster, das einen <b>Details</b>-Button enthält. Über diesen Button können Sie das <b>Applikationsinformation</b>-Fenster öffnen, das eine Prüfung der Unterschiede zwischen dem aktuellen SPS-Projekt und dem SPS-Projekt auf der Steuerung ermöglicht. Dabei geht es um den Vergleich der Anzahl von Bausteinen, der Daten und der Speicherorte.</p> <p>Das <b>Applikationsinformation</b>-Fenster enthält eine grobe Beschreibung der Unterschiede, beispielsweise:</p> <ul style="list-style-type: none"> <li>• Deklaration von MAIN geändert</li> <li>• Variable fbMyNewInstance in MAIN eingefügt</li> <li>• Anzahl der Methoden/Aktionen von FB_Sample geändert</li> </ul> <p><input checked="" type="checkbox"/> (Voreinstellung): Wenn diese Einstellung aktiviert ist, wird die Information zum Inhalt des SPS-Projekts mit auf die SPS geladen. Dies ermöglicht eine erweiterte Prüfung der Unterschiede zwischen dem aktuellen SPS-Projekt und dem SPS-Projekt auf der Steuerung. Die erweiterte Prüfungsmöglichkeit besteht darin, dass das <b>Applikationsinformation</b>-Fenster den zusätzlichen Reiter <b>Onlinevergleich</b> enthält, welcher eine Baumvergleichsansicht zeigt. Anhand dieser können Sie erkennen, welche POUs geändert, gelöscht oder hinzugefügt wurden. Der zusätzliche Reiter erscheint, wenn Sie den blau unterstrichenen Befehl im unteren Bereich des <b>Applikationsinformation</b>-Fensters ausführen („Applikation nicht aktuell. Code jetzt generieren, um den Onlinevergleich anzuzeigen?“).</p>

<p>Generiere tpy-Datei</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4024</p> <p>Die tpy-Datei enthält u. a. Projekt-, Routing-, Compiler- und Zielsystem-Informationen und ist das Format zur Beschreibung eines TwinCAT 2 SPS-Projekts. Aus Kompatibilitätsgründen mit bestehenden Anwendungen kann diese Datei bei Bedarf für ein TwinCAT 3 SPS-Projekt erstellt werden.</p> <p><input type="checkbox"/> (Voreinstellung): Beim Erstellen des SPS-Projekts wird die zum Projekt gehörende tpy-Datei nicht erstellt.</p> <p><input checked="" type="checkbox"/> : Beim Erstellen des SPS-Projekts wird die zum Projekt gehörende tpy-Datei erstellt und im Projektordner abgelegt.</p> <p>Beachten Sie, dass der Wert und die Konfigurationsverfügbarkeit dieser Option davon abhängen, ob die TPY-Datei als Target-Datei konfiguriert ist oder nicht (siehe <a href="#">Registerkarte Settings</a> [▶ 122]).</p> <ul style="list-style-type: none"> <li>• Wenn die TPY-Datei als Target-Datei aktiviert wird, passiert folgendes:             <ul style="list-style-type: none"> <li>◦ TwinCAT merkt sich den aktuellen Status der Option „Generiere tpy-Datei“ (= „ursprünglicher Wert“, s.u.).</li> <li>◦ Falls dies nicht bereits der Fall ist, wird die Option „Generiere tpy-Datei“ beim nächsten Erstellen des Projekts automatisch aktiviert.</li> <li>◦ Außerdem wird die Option „Generiere tpy-Datei“ ausgegraut, sodass sie vom Anwender nicht deaktiviert werden kann, solange die TPY-Datei als Target-Datei konfiguriert ist.</li> </ul> </li> <li>• Wenn die TPY-Datei anschließend als Target-Datei wieder deaktiviert wird, passiert folgendes:             <ul style="list-style-type: none"> <li>◦ Beim nächsten Erstellen des Projekts erhält die Option „Generiere tpy-Datei“ ihren „ursprünglichen Wert“ (s.o.).</li> <li>◦ Außerdem ist die Option nicht mehr ausgegraut, sodass sie wieder für Konfigurationen durch den Anwender verfügbar ist.</li> </ul> </li> </ul>
<p>Initialisierungswerte von Deklarationen zur TMC-Datei hinzufügen</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4026.12</p> <p><input checked="" type="checkbox"/> (Voreinstellung): Initialisierungswerte von Deklarationen werden in der TMC-Datei gespeichert. Dies ist beispielsweise sinnvoll, wenn die Initialisierungswerte für andere Applikationen benötigt werden.</p> <p><input type="checkbox"/> : Initialisierungswerte von Deklarationen werden nicht mit in der TMC-Datei gespeichert. Dadurch werden die Größe der TMC-Datei und die Übersetzungsdauer verringert, insbesondere wenn die Deklarationen im SPS-Projekt sehr viele Initialisierungswerte enthalten.</p> <p>Beispiel:</p> <pre>VAR   aSample : ARRAY[1..10] OF INT := [2, 38, 5, 9, 74, 62, 87, 3, 16, 4]; END_VAR</pre>
<p>Kommentare über POU- und DUT-Deklarationen zur TMC-Datei hinzufügen</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p><input checked="" type="checkbox"/> (Voreinstellung): Kommentare zu POU- und DUTs, die oberhalb von deren Deklaration stehen, werden mit dem Datentyp in der TMC-Datei gespeichert. Dies ist beispielsweise sinnvoll, wenn die Kommentare für andere Applikationen benötigt werden.</p> <p><input type="checkbox"/> : Kommentare oberhalb der Deklaration von POU- und DUTs werden nicht mit in der TMC-Datei gespeichert. Dadurch werden die Größe der TMC-Datei und die Übersetzungsdauer verringert, insbesondere wenn die Kommentare ausführliche Beschreibungen zu den POU- und DUTs enthalten.</p> <p>Beispiel:</p> <pre>// This function block represents an axis FUNCTION_BLOCK FB_Axis ...</pre>

**Solution Options**

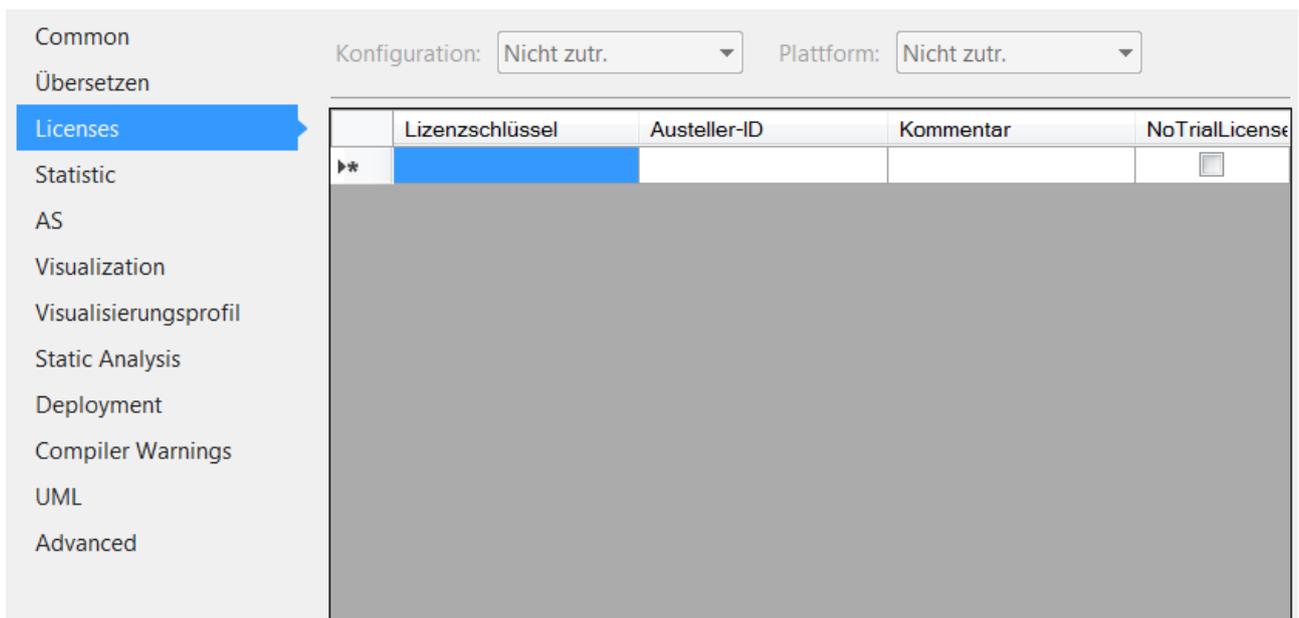
<p>Compilerversion</p>	<p>Verfügbar bis TwinCAT 3.1 Build 4024</p> <p>Definiert die Compilerversion, die TwinCAT beim Übersetzen und während des Ladens zum Übersetzen verwendet.</p> <p>Beachten Sie, dass diese Einstellung nicht den Remote Manager ersetzt. Für die Handhabung verschiedener Engineering-Versionen sollte, wenn es sich bei dem SPS-Projekt um ein Applikationsprojekt handelt, immer der Remote Manager verwendet werden.</p> <p>Die Compilerversion sollte in diesem Fall immer auf „neueste“ stehen.</p> <p>Die Einstellung der Compilerversion kommt nur dann zum Tragen, wenn es sich bei dem SPS-Projekt, welches versionsverwaltet werden soll, um ein Bibliotheksprojekt handelt.</p> <p>Es wird empfohlen, die Bibliothek mit der ältesten Version zu speichern, mit der sie letztlich verwendet werden soll. Dazu muss die Compilerversion auf die entsprechende fixe Version gesetzt werden (z. B. „3.1.4018.0“).</p>
<p>Maximale Anzahl an Warnungen</p>	<p>Bezieht sich auf die Warnungen, die TwinCAT in der Ansicht <b>Fehlerliste</b> maximal ausgibt.</p> <p>Die Auswahl der angezeigten Compilerwarnungen legen Sie im Dialog <b>Projekteinstellungen</b> in der Kategorie <b>Compilerwarnungen</b> fest.</p>
<p>Konstanten ersetzen</p>	<p><input checked="" type="checkbox"/> : Für jede Konstante skalaren Typs, also nicht für STRING, ARRAY oder Strukturen, lädt TwinCAT direkt den Wert. Im Onlinebetrieb kennzeichnet TwinCAT die Konstanten im Deklarationseditor oder Monitoring-Fenster durch ein dem Wert vorangestelltes Symbol. In diesem Fall ist ein Zugriff, zum Beispiel über einen ADR-Operator, Forcen und Schreiben, nicht möglich.</p> <p><input type="checkbox"/> (Voreinstellung): Der Zugriff auf Konstanten ist möglich. Die Rechenzeit verlängert sich minimal.</p>

**Siehe auch:**

- [Kategorie Compiler Warnings \[▶ 112\]](#)

**5.4.4.3 Kategorie Lizenzen**

Reservierung für zukünftige Nutzung!



Hinweis: Die Abfrage einer OEM-Lizenz für eine eigene Bibliothek muss immer im Code der Bibliothek durch den Anwender erfolgen. Siehe Abfrage einer OEM-Lizenz in der PLC Applikation.

### 5.4.4.4 Kategorie Statistic

Die Kategorie **Statistic** gibt eine statistische Auskunft, wie viele Objekte der einzelnen Typen im Projekt verwendet sind.

Konfiguration: Nicht zutr. Plattform: Nicht zutr.

Anzahl der Objekte: 12

Objekttyp	Anzahl
Aktion	2
Methode	2
Ordner	4
POU	3
Referenced Task	1

### 5.4.4.5 Kategorie AS

#### ● Geltungsbereich von SPS-Projekteigenschaften

**i** Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

Die Kategorie **AS** dient der Konfiguration der Einstellungen für AS-Objekte. Jedes neue AS-Objekt hat die konfigurierten Einstellungen automatisch in seinen Eigenschaften.

#### Registerkarte Variablen

Konfiguration: Nicht zutr. Plattform: Nicht zutr.

**Solution options**

Variablen Übersetzen

Aktiv	Variable	Deklarieren	Beschreibung
<input type="checkbox"/>	SFCInit	<input checked="" type="checkbox"/>	Alle Schritte und Aktionen werden zurückgesetzt. Der Initialschritt wird aktiviert. Keine Aktionen werden ausgeführt.
<input type="checkbox"/>	SFCReset	<input checked="" type="checkbox"/>	Alle Schritte und Aktionen werden zurückgesetzt. Der Initialschritt wird aktiviert und seine Aktionen werden ausgeführt.
<input type="checkbox"/>	SFCError	<input checked="" type="checkbox"/>	Wird 'TRUE', falls ein Zeitüberwachung fehlschlägt.
<input type="checkbox"/>	SFCEnableLimit	<input checked="" type="checkbox"/>	Aktiviert die Zeitüberwachung für Schritte.
<input type="checkbox"/>	SFCErrorStep	<input checked="" type="checkbox"/>	Enthält den Namen des Schritts, welcher SFCError auf 'TRUE' gesetzt hat. SFCError muss aktiviert sein.
<input type="checkbox"/>	SFCErrorPOU	<input checked="" type="checkbox"/>	Enthält den Namen der POU, welche SFCError auf 'TRUE' gesetzt hat. SFCError muss aktiviert sein.
<input type="checkbox"/>	SFCQuitError	<input checked="" type="checkbox"/>	Die Ausführung wird angehalten. SFCError wird zurückgesetzt. SFCError muss aktiviert sein.
<input type="checkbox"/>	SFCPause	<input checked="" type="checkbox"/>	Die Ausführung wird angehalten. SFCError wird zurückgesetzt.
<input type="checkbox"/>	SFCTrans	<input checked="" type="checkbox"/>	Wird 'TRUE', falls eine Transition schaltet.
<input type="checkbox"/>	SFCCurrentStep	<input checked="" type="checkbox"/>	Enthält den Namen des aktiven Schritts.
<input type="checkbox"/>	SFCtip	<input checked="" type="checkbox"/>	Schaltet mit die nächste Transition bei einer steigenden Flanke.
<input type="checkbox"/>	SFCtipMode	<input checked="" type="checkbox"/>	Falls 'TRUE', können Transitionen nur mittels SFCtip geschaltet werden.
<input type="checkbox"/>	SFCErrorAnalyzation	<input checked="" type="checkbox"/>	Enthält die möglichen Variablen, welche SFCError auf TRUE gesetzt haben, als Zeichenkette dargestellt. SFCError ist erforderlich
<input type="checkbox"/>	SFCErrorAnalyzationTable	<input checked="" type="checkbox"/>	Enthält die möglichen Variablen, welche SFCError auf TRUE gesetzt haben, in einer Tabelle. SFCError ist erforderlich

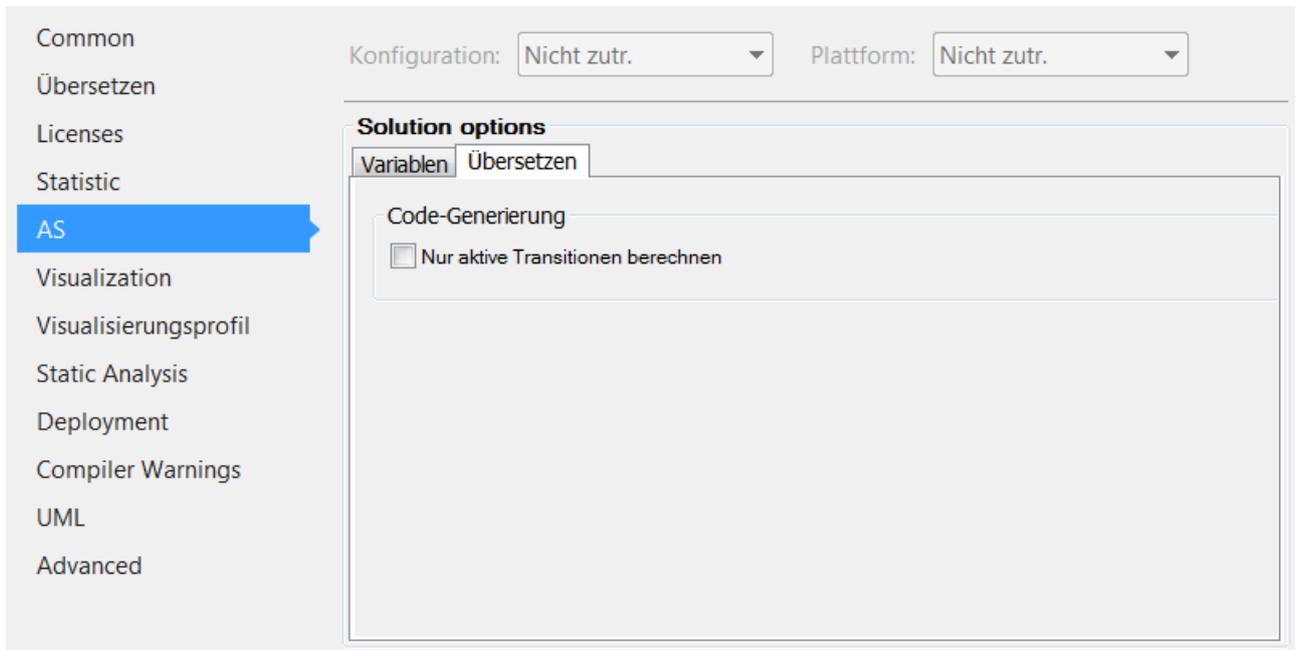
Implizit erzeugte Variablen (Flags) zur Kontrolle und zum Monitoring der Abarbeitung in einem AS-Diagramm.

Aktiv	 : Die entsprechende Variable wird verwendet.
Deklariert	 : Die entsprechende Variable wird automatisch angelegt. Ansonsten, falls die Verwendung beabsichtigt ist (Aktiv ist gesetzt), muss der Anwender sie selbst deklarieren.



Eine automatisch deklarierte Flag-Variable ist nur im Onlinebetrieb im Deklarationsteil des AS-Editors sichtbar.

## Registerkarte Übersetzen



### Codegenerierung

Nur aktive Transitionen berechnen	 : TwinCAT erzeugt nur für gerade aktive Transitionen Code.
-----------------------------------	--

#### Siehe auch:

- [AS-Flags](#)

### 5.4.4.6 Kategorie Visualization



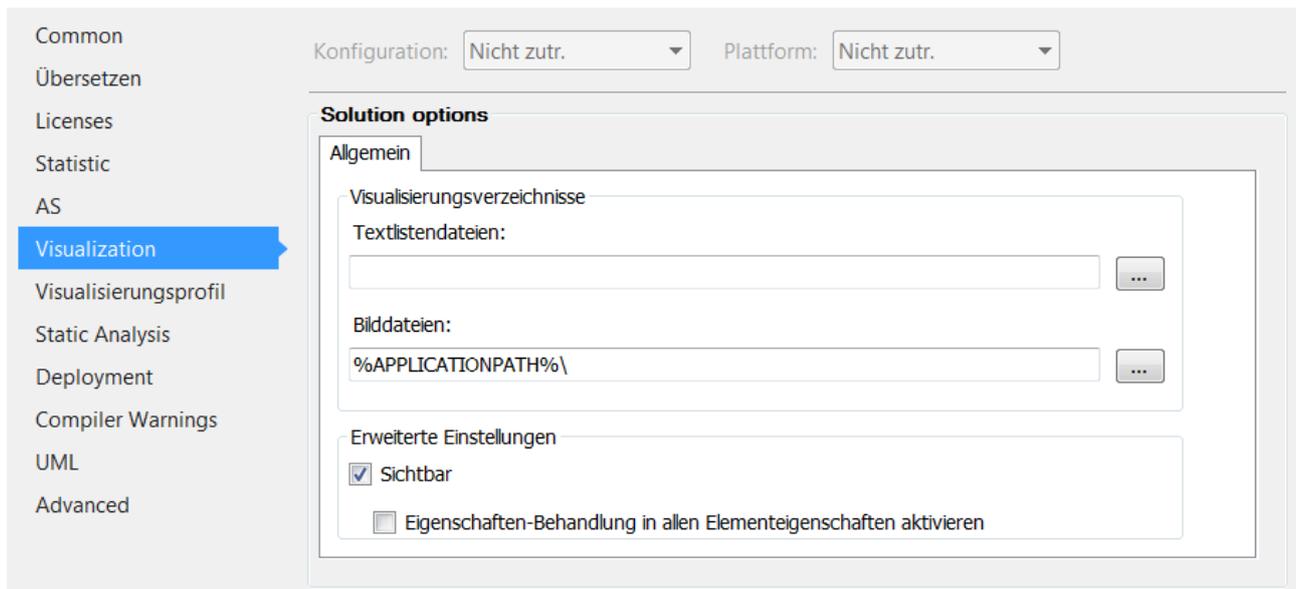
#### Geltungsbereich von SPS-Projekteigenschaften

Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

Die Kategorie **Visualization** dient der Konfiguration der projektweiten Einstellungen für Objekte des Typs Visualisierung.



Registerkarte Allgemein

Visualisierungsverzeichnisse

Textlistendateien	<p>Verzeichnis, das Textlisten enthält, die im Projekt zur Verfügung stehen, um Texte für verschiedene Sprachen zu konfigurieren. TwinCAT verwendet das Verzeichnis zum Beispiel beim Exportieren oder Importieren von Textlisten.</p> <p>Mit Klick auf  erscheint der Dialog <b>Ordner suchen</b>, der das Auswählen eines Verzeichnisses im Dateisystem ermöglicht.</p>
Bilddateien	<p>Verzeichnis, das Bilddateien enthält, die im Projekt zur Verfügung stehen. Mehrere Ordner sind mit einem Strichpunkt voneinander getrennt. TwinCAT verwendet das Verzeichnis zum Beispiel beim Exportieren oder Importieren von Bilddateien.</p> <p>Mit Klick auf  erscheint der Dialog <b>Ordner suchen</b>, der das Auswählen eines Verzeichnisses im Dateisystem ermöglicht.</p>

Erweiterte Einstellungen

Eigenschaften-Behandlung in allen Elementeigenschaften aktivieren	<p> : Sie können ein Visualisierungselement in denjenigen seiner Eigenschaften, in denen Sie eine IEC-Variable auswählen, auch mit einer Eigenschaft  konfigurieren. TwinCAT erzeugt dann beim Übersetzen einer Visualisierung zusätzlichen Code für die Eigenschaften-Behandlung.</p> <p>Voraussetzung: Ihr IEC-Code enthält mindestens ein Objekt des Typs Schnittstelleneigenschaft, also eine Eigenschaft  .</p> <p> <ul style="list-style-type: none"> <li>▲  MAIN (PRG)                             <ul style="list-style-type: none"> <li>▲  Property_A                                     <ul style="list-style-type: none"> <li> Get</li> <li> Set</li> </ul> </li> </ul> </li> </ul> </p> <p>Voraussetzung: <b>Sichtbar</b> ist aktiviert.</p>
---	---

### 5.4.4.7 Kategorie Visualisierungsprofil

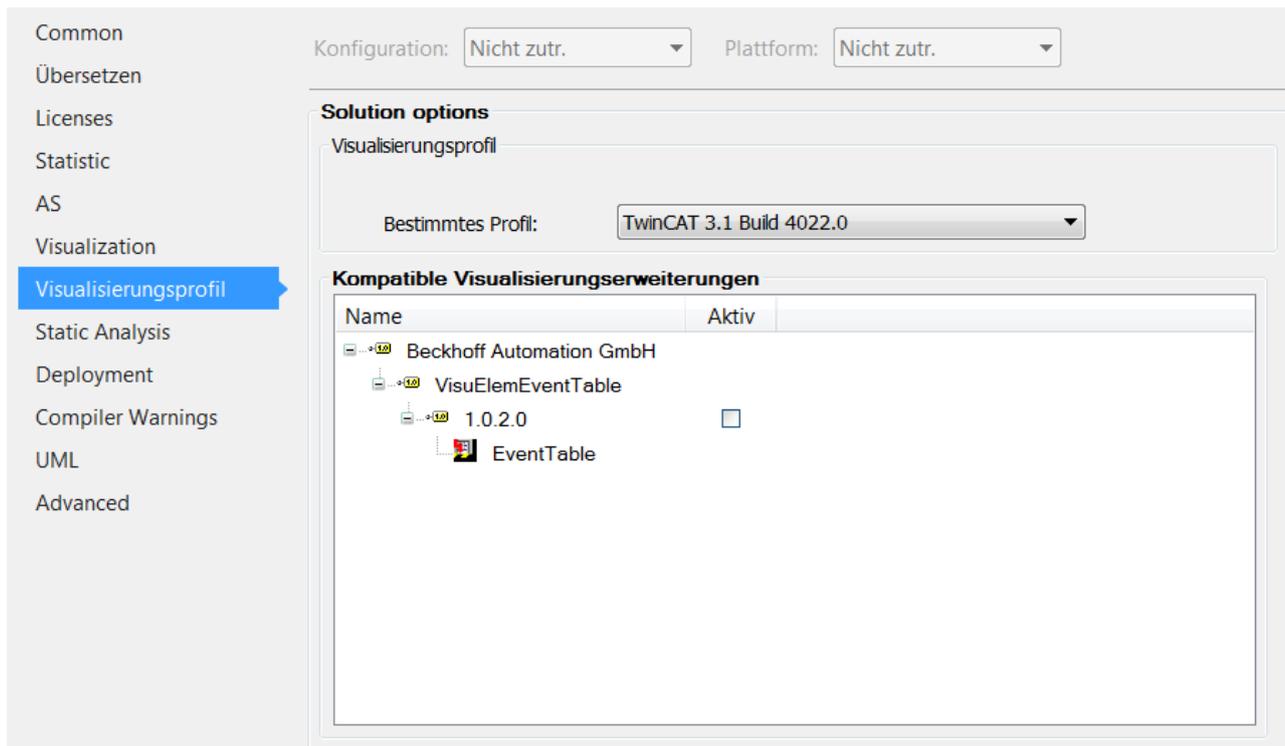
**i Geltungsbereich von SPS-Projekteigenschaften**

Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

Die Kategorie **Visualisierungsprofil** ermöglicht die Einstellung des Visualisierungsprofils.



#### Visualisierungsprofil

Bestimmtes Profil	Profil, das TwinCAT im Projekt verwendet und das die Visualisierungselemente bestimmt, die im Projekt zur Verfügung stehen. Die Auswahlliste enthält alle bisher installierten Profile.
-------------------	--

### 5.4.4.8 Kategorie Static Analysis

**i Geltungsbereich von SPS-Projekteigenschaften**

Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

Die Kategorie **Static Analysis** definiert die Prüfungen, die bei der statischen Codeanalyse berücksichtigt werden.

**Static Analysis Light:**

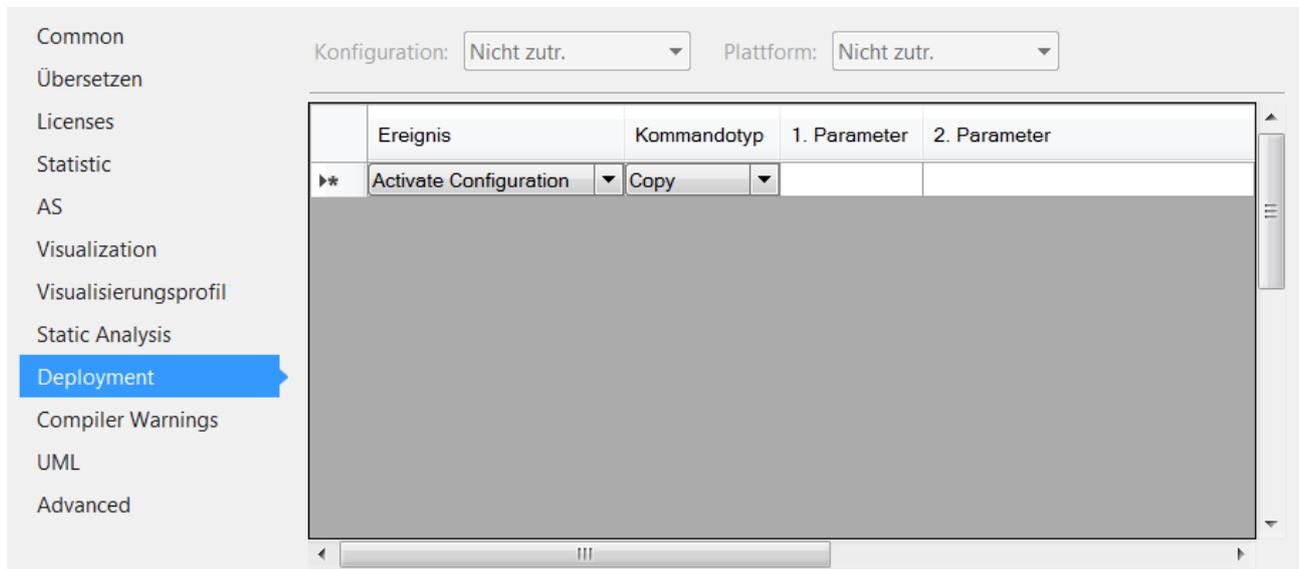
- Wenn Sie die zusätzliche Engineering-Lizenz TE1200 nicht aktiviert haben, können Sie die lizenzfreie Variante des Static Analysis (Static Analysis Light) nutzen, welche einige wenige Kodierregeln beinhaltet. Anhand der kostenfreien Light-Variante können Sie sich – auf Basis eines stark reduzierten Funktionsumfangs – beispielsweise mit dem prinzipiellen Handling des Produkts vertraut machen.
- Weitere Informationen zu Static Analysis Light finden Sie unter: Dokumentation PLC: SPS-Projekt programmieren > Syntax prüfen und Code analysieren > Codeanalyse (Static Analysis)

**Static Analysis Full:**

- Wenn Sie die zusätzliche Engineering-Lizenz TE1200 aktiviert haben, steht Ihnen der volle Funktionsumfang des Static Analysis zur Verfügung (Speichern und Laden von Einstellungen, mehr als 100 Kodierregeln, Namenskonventionen, Metriken, unzulässige Symbole).
- Weitere Informationen zu Static Analysis Full finden Sie unter: TE1200 Static Analysis.

**5.4.4.9 Kategorie Deployment**

Die Kategorie **Deployment** dient der Einstellung von Befehlen, welche während des Aufspielen und Startens einer Anwendung ausgeführt werden sollen.



Folgende Ereignisse stehen zur Verfügung, nach welchen die in der Liste aufgeführten Kommandos aufgerufen werden können:

Activate Configuration	Das gewünschte Kommando wird nach einem Aktivieren der Konfiguration aufgerufen.
Plc Download	Das gewünschte Kommando wird nach einem erfolgten Download der SPS-Applikation auf das Zielsystem aufgerufen.
Plc Online Change	Das gewünschte Kommando wird nach einem erfolgten Online Change aufgerufen.
Plc After Compile	Das gewünschte Kommando wird nach einem Kompilervorgang der SPS-Applikation aufgerufen.

Folgende Kommandos können ausgeführt werden:

Copy	Kopiert Dateien von Parameter 1 (Quellpfad) nach einem in Parameter 2 (Zielpfad) angegebenen Ort.
Execute	Führt die unter Parameter 1 aufgeführte Applikation bzw. das aufgeführte Skript aus.

Quell- und Zielpfade können virtuelle Umgebungsvariablen beinhalten, welche von TwinCAT entsprechend aufgelöst werden.

Die nachfolgenden Umgebungsvariablen werden u. a. unterstützt:

Virtuelle Umgebungsvariable	Registrierungswert	Defaultwert
%TC_INSTALLPATH%	InstallDir	C:\TwinCAT3.x \
%TC_TARGETPATH%	TargetDir	C:\TwinCAT3.x \Target\
%TC_BOOTPRJPATH%	BootDir	C:\TwinCAT3.x \Boot\
%TC_RESOURCEPATH%	ResourceDir	C:\TwinCAT3.x \Target\Resource\
%SOLUTIONPATH%	-	Speicherort der Solution-Datei

Die Registrierungswerte werden unter folgendem Schlüssel in der Registry abgelegt:  
**HKLM\Software\Beckhoff\TwinCAT3.**

**Beispiel:**

Im nachfolgenden Beispiel wird die Datei *SampleFile.xml* aus dem Projektunterordner Config der Solution in den Ordner *C:\plc\config* auf dem Zielsystem kopiert.

Ereignis	Kommandotyp	1. Parameter	2. Parameter
Activate Configuration	Copy	%SOLUTIONPATH%\Config\SampleFile.xml	C:\plc\Config\SampleFile.xml

### 5.4.4.10 Kategorie Compiler Warnings

Die Kategorie **Compiler Warnings** dient der Auswahl der Compilerwarnungen, die TwinCAT bei einem Kompilierungslauf im Meldungsfenster anzeigt.



Die maximale Anzahl aufgelisteter Warnungen legen Sie in der Kategorie **Compile** fest.

**Siehe auch:**

- [Befehl SPS-Projekt erstellen \[► 126\]](#)
- [Kategorie Übersetzen \[► 103\]](#)

### 5.4.4.11 Kategorie UML

#### ● Geltungsbereich von SPS-Projekteigenschaften

**i** Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

In der Kategorie **UML** können Sie die UML-Compiler-Version ändern. Diese Einstellung ist nur bei Verwendung des UML-Zustandsdiagramms relevant.

Weitere Informationen zu den Konfigurationsmöglichkeiten finden Sie im Abschnitt „UML Compiler-Version“ der Dokumentation TF1910 TC3 UML.

The screenshot shows the configuration interface for the UML category. On the left, a sidebar lists various categories, with 'UML' highlighted in blue. The main area shows configuration options for 'UML'. At the top, there are dropdown menus for 'Konfiguration:' (set to 'Nicht zutr.') and 'Plattform:' (set to 'Nicht zutr.'). Below this, the 'Solution options' section is expanded, displaying a table of settings:

Property	Value
UML compiler version in project	4.0.2.1
Recommended, newest version	4.0.2.1
Action	Do not update.

### 5.4.4.12 Kategorie Advanced

#### ● Geltungsbereich von SPS-Projekteigenschaften

**i** Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

Die Kategorie **Advanced** dient der Konfiguration von erweiterten Eigenschaften.

#### Write options



---

**Engineering-Inkompatibilität von Dateiversion 1.2.0.0 (oder größer) mit TwinCAT 3.1 < Build 4024**

Beachten Sie, dass Objekte, die mit der Dateiversion 1.2.0.0 (oder größer) gespeichert werden, nicht mit Engineering-Versionen < TwinCAT 3.1.4024 geladen werden können!

Da ein Objekt bei Verwendung des optionalen Speicherformats „Base64“ automatisch mit der Dateiversion 1.2.0.0 gespeichert wird, können Objekte mit Base64-Speicherformat folglich nicht mit Engineering-Versionen < TwinCAT 3.1.4024 geladen werden.

Falls ein SPS-Projekt sowohl Objekte mit der Dateiversion 1.1.0.1 als auch Objekte mit der Dateiversion 1.2.0.0 enthält, werden die 1.1.0.1-Objekte mit einer Engineering-Version < TwinCAT 3.1.4024 weiterhin geladen. Lediglich die Objekte mit der Dateiversion 1.2.0.0 werden nicht geladen.

Die Dateiversion einer Datei, die mit der Dateiversion 1.2.0.0 gespeichert wurde, kann mit XAE-Versionen >= TwinCAT 3.1.4024 nachträglich wieder auf 1.1.0.1 gesetzt werden.

---

Objektinhalt schreiben als  
(„Write object content as“)

Verfügbar ab TwinCAT 3.1 Build 4024

#### Hintergrundinformationen:

Ab Build 4024 wird mit **Base64** ein neues Speicherformat eingeführt, das für die folgenden SPS-Objekte optional zur Verfügung steht:

- POUs, bei denen der POU-Rumpf in einer grafischen Implementierungssprache programmiert ist
  - AS (Ablaufsprache)
  - FUP/KOP/AWL (Funktionsplan/Kontaktplan/Anweisungsliste)
  - CFC (Continuous Function Chart und Seitenorientierter CFC)
  - UML Klassendiagramm und Zustandsdiagramm
- POUs, die über ein Unterelement (z. B. Aktion, Methode) verfügen, das in einer grafischen Implementierungssprache programmiert ist (grafische Sprachen siehe erster Stichpunkt)
- Visualisierungen
- Visualisierungsmanager
- Textlisten
- Rezeptmanager
- Bildersammlungen

Bislang wurden diese Objekte standardmäßig als XML gespeichert.

Ob diese Objekttypen ab Build 4024 als XML oder als Base64 gespeichert werden sollen, ist konfigurierbar.

#### Vorteile von Base64 gegenüber XML:

Im Vergleich zu XML ergibt sich mit Base64 eine komprimierte Speicherung. Als Folge dessen kann bei Dateizugriffen auf diese Objekte eine verbesserte Performance erreicht werden, welche beispielsweise beim Projektladen oder beim Verschieben/Kopieren der Objekte zum Tragen kommt.

#### Einstellungsmöglichkeit des Standard-Speicherformats:

Über die Einstellung „Objektinhalt schreiben als“ in den SPS-Projekteigenschaften können Sie für ein SPS-Projekt definieren, welches das standardmäßige Speicherformat für die oben genannten Objekttypen ist.

Das ausgewählte Standard-Speicherformat wird nur bei neu hinzugefügten Objekten verwendet (Ausnahme: nicht bei neu hinzugefügten POU-Unterelementen. Beispiel: Eine POU ist als XML gespeichert und das Standard-Speicherformat ist als Base64 konfiguriert. Wenn zu der POU dann ein grafisches Unterelement hinzugefügt wird, bleibt das Speicherformat der POU und damit auch für das Unterelement als XML bestehen).

Das Speicherformat eines existierenden Objekts mit einem vom Standard abweichenden Speicherformat wird nicht automatisch geändert, wenn das Objekt geändert und gespeichert wird. Das Speicherformat eines existierenden Objekts kann entweder individuell über das Eigenschaften-Fenster geändert werden (siehe unten) oder alternativ besteht beim Umstellen des Standard-Speicherformats die Möglichkeit, das neu ausgewählte Speicherformat auch für alle bereits bestehenden Objekte zu übernehmen. Wenn Sie an dieser Stelle das Speicherformat umstellen, erscheint ein entsprechendes Abfragefenster.

Für die Einstellung „Objektinhalt schreiben als“ stehen die folgenden Auswahlmöglichkeiten zur Verfügung:

- **XML** (Voreinstellung): Die oben genannten SPS-Objekte werden standardmäßig im XML-Format gespeichert.
  - Objekte mit diesem Speicherformat werden in der Dateiversion 1.1.0.1 gespeichert.
- **Base64**: Die oben genannten SPS-Objekte werden standardmäßig im Base64-Format gespeichert.
  - Objekte mit diesem Speicherformat werden in der Dateiversion 1.2.0.0 gespeichert.  
Bitte beachten Sie, dass Objekte mit der Dateiversion 1.2.0.0 (oder größer) nicht mit Engineering-Versionen < TwinCAT 3.1.4024 geladen werden können!

#### Individuelle Einstellungsmöglichkeit des Speicherformats:

<p>Produktversion in Dateien schreiben („Write product version in files“)</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4024</p> <p>Die Produktversion repräsentiert, mit welcher Plugin-Version eine SPS-Datei (z. B. ein Funktionsbaustein) gespeichert wurde. Die Einstellung dieser Checkbox ist projektweit gültig und ist die Standard-Einstellung für alle geänderten oder neu hinzugefügten SPS-Objekte, die sich in diesem SPS-Projekt befinden.</p> <p><input type="checkbox"/> (Voreinstellung): Die Produkt- bzw. Plugin-Version wird nicht mit in die Datei geschrieben.</p> <ul style="list-style-type: none"> <li>• Wenn Sie die Einstellung von aktiviert nach deaktiviert ändern, erscheint ein Abfragefenster, in dem Sie auswählen können, ob die Produktversion aus allen bestehenden Dateien entfernt werden soll.</li> <li>• Anwendungsfall für die deaktivierte Option: Wenn die Produktversion nicht von Interesse ist, kann diese Einstellung verwendet werden. Dadurch werden Änderungen von Dateien hinsichtlich Sourcecode-Verwaltungssystemen minimiert.</li> </ul> <p><input checked="" type="checkbox"/> : Die Produkt- bzw. Plugin-Version wird mit in die Datei geschrieben (im XAE ist die Version nicht sichtbar, bei Analyse der Datei auf Dateiebene ist sie erkennbar).</p> <ul style="list-style-type: none"> <li>• Wenn Sie die Einstellung von deaktiviert nach aktiviert ändern, erscheint ein Abfragefenster, in dem Sie auswählen können, ob die Produktversion zu allen bestehenden Dateien hinzugefügt werden soll.</li> <li>• Anwendungsfall für die aktivierte Option: Diese Einstellung kann verwendet werden, um die Dateiversion beispielsweise zu Debug- oder Nachverfolgungszwecken mit in die Datei zu schreiben.</li> <li>• Beachten Sie folgendes: Wenn die Datei mit einer anderen Produktversion gespeichert wird, führt dies zu einer Änderung dieser Datei, welche bei Verwendung von Sourcecode-Verwaltungssystemen als Dateiu Unterschied erkenntlich ist.</li> </ul>
<p>Objekthalt mit Profil schreiben: (“Write object content with Profile”)</p>	<p>Das Profil definiert das Format, in dem Objekte abgespeichert werden. Mit dem Build 4024 wurden beispielsweise neue Funktionalitäten für das PLC HMI hinzugefügt. Aus diesem Grund können Visualisierungsdateien, die mit dem Build 4024 abgespeichert worden sind, nicht direkt mit älteren Builds geöffnet werden. Wenn Sie hier ein 4022 Profil einstellen, dann werden die Visualisierungsdateien in dem passenden Format abgespeichert und können mit dem Build 4022 geöffnet werden.</p> <p><b>Voraussetzung:</b> Damit zum Beispiel das 4022 Profil im Drop-Down-Menü verfügbar ist, muss entweder eine 4022 Remote Manager Installation durchgeführt worden sein oder die aktuelle 4024 XAE Installation über eine zuvor vorhandene 4022 XAE Installation installiert worden sein.</p>
<p>Write Bookmarks to File</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p><input type="checkbox"/> (Voreinstellung): Die Lesezeichen werden nur in der .suo-Datei vom Visual Studio gespeichert. Eine bereits erstellte .bookmarks-Datei wird aus dem Projektverzeichnis gelöscht.</p> <p><input checked="" type="checkbox"/> : Die abgelegten Lesezeichen werden aus der Benutzeroptionsdatei vom Visual Studio Projekt (.suo) zusätzlich in eine separate Datei geschrieben. Diese Datei endet mit .bookmarks und liegt im Projektverzeichnis. Damit ist sie dann ebenfalls Bestandteil der bekannten Archivmöglichkeiten.</p> <p>Die globale Standardeinstellung für neue SPS-Projekte, ob Lesezeichen in einer separaten Datei abgelegt werden sollen, finden Sie unter <u>Dialog Optionen - Write Options</u> [► 194]. Der Wert wird bei der Erstellung eines neuen SPS-Projekts einmalig in diese lokale Projekteinstellung übernommen.</p>

## Multouser options

<p>Multiuser nutzen („Use Multiuser“)</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4024</p> <p><input type="checkbox"/> (Voreinstellung): Die Multiuser-Funktionalität des SPS-Projekts ist nicht aktiviert.</p> <p><input checked="" type="checkbox"/> : Die Multiuser-Funktionalität des SPS-Projekts ist aktiviert.</p> <p>Bitte beachten Sie auch die weiterführenden Informationen hierzu in der Multiuser-Dokumentation.</p>
<p>Multiuser URL</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p>Beschreibung folgt.</p>
<p>Update Parent</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p>Beschreibung folgt.</p>

**Solution options**

<p>Sicherer Onlinebetrieb („Secure Online Mode“)</p>	<p><input type="checkbox"/> : (Voreinstellung): Aus Sicherheitsgründen wird der Benutzer beim Aufruf der folgenden Befehle grundsätzlich dazu aufgefordert, die Ausführung nochmals zu bestätigen.</p> <ul style="list-style-type: none"> <li>• Konfiguration aktivieren</li> <li>• Restart TwinCAT System in Config/Run Mode</li> <li>• Reset Kalt</li> <li>• Reset Ursprung</li> </ul> <p><input checked="" type="checkbox"/> : Zusätzlich zu den oben genannten Befehlen, bei denen standardmäßig eine Bestätigungsaufforderung erscheint, werden Sie bei den folgenden Befehlen ebenfalls zu einer Bestätigung aufgefordert.</p> <ul style="list-style-type: none"> <li>• Einloggen</li> <li>• Start</li> <li>• Stopp</li> <li>• Einzelzyklus</li> <li>• Werte schreiben</li> <li>• Werte forcen</li> <li>• Forcen aufheben</li> </ul>
<p>Autoupdate Visu Profile</p>	<p>Mit dieser Option können Sie das automatische Update-Verhalten des Visualisierungsprofils konfigurieren.</p> <p>Wenn Sie ein SPS-Projekt öffnen, in dem ein veraltetes Visualisierungsprofil verwendet wird, erscheint im Meldungsfenster eine entsprechende Warnung („New Version found for Visualization profile“).</p> <p><input type="checkbox"/> (Voreinstellung): Wenn die Option <b>Autoupdate Visu Profile</b> deaktiviert ist, wird die Visualisierungsprofil-Version nicht automatisch geändert. Per Doppelklick auf die Warnung „New Version found for Visualization profile“ können Sie den ProfileUpdate-Dialog öffnen, in welchem Sie die Visualisierungsprofil-Version manuell ändern können.</p> <p><input checked="" type="checkbox"/> : In einem solchen Fall wird die Visualisierungsprofil-Version automatisch auf die neueste Version gesetzt, falls die Option <b>Autoupdate Visu Profile</b> aktiviert ist. Bei einem solchen automatischen Update der Visualisierungsprofil-Version wird im Meldungsfenster eine entsprechende Warnung angezeigt (z. B. „Visualization profile set from ‚TwinCAT 3.1 Build 4020.10‘ to ‚TwinCAT 3.1 Build 4022.0‘“).</p>
<p>Autoupdate Uml Profile</p>	<p>Mit dieser Option können Sie das automatische Update-Verhalten der UML-Compiler-Version konfigurieren.</p> <p>Wenn Sie ein SPS-Projekt öffnen, in dem eine veraltete UML-Compiler-Version verwendet wird, erscheint im Meldungsfenster eine entsprechende Warnung („neue Version für UML gefunden“).</p> <p><input type="checkbox"/> (Voreinstellung): Wenn die Option <b>Autoupdate Uml Profile</b> deaktiviert ist, wird die UML-Compiler-Version nicht automatisch geändert. Per Doppelklick auf die Warnung „neue Version für UML gefunden“ können Sie den ProfileUpdate-Dialog öffnen, in welchem Sie die UML Compiler-Version manuell ändern können.</p> <p><input checked="" type="checkbox"/> : In einem solchen Fall wird die UML-Compiler-Version automatisch auf die neueste Version gesetzt, falls die Option <b>Autoupdate Uml Profile</b> aktiviert ist. Bei einem solchen automatischen Update der UML-Compiler-Version wird im Meldungsfenster eine entsprechende Warnung angezeigt (z. B. „UML set from ‚4.0.2.0‘ to ‚4.0.2.1‘“).</p> <p>Weitere Informationen hierzu finden Sie unter UML Compiler-Version.</p>

Write Line IDs	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p><input type="checkbox"/> (Voreinstellung): Es werden keine separaten Line-IDs erzeugt. Für die Zuordnung der Maschinencode-Anweisungen und das Breakpoint-Handling wird in diesem Fall die Zeilennummer genutzt. Bei Änderungen wie Leerzeichen oder Kommentaren ist daher ein Online-Change erforderlich.</p> <p><input checked="" type="checkbox"/> : Für die POU's des Projekts werden Line-IDs erzeugt und gespeichert (Standardverhalten bis TwinCAT 3.1 Build 4024). Über die Line-IDs können Code-Zeilen zu Maschinencode-Anweisungen zugeordnet werden, was u. a. für das Breakpoint-Handling benötigt wird.</p> <p>Die globale Standardeinstellung für neue SPS-Projekte bezüglich der Write Line IDs finden Sie unter <a href="#">Dialog Optionen - Write Options [► 194]</a>. Der Wert wird bei der Erstellung eines neuen SPS-Projekts einmalig in diese lokale Projekteinstellung übernommen.</p>
----------------	--

### Compatibility

Convert PLC Project to previous TwinCAT version	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p>In dem sich öffnenden Dialog können Sie eine TwinCAT-Version auswählen, in die das SPS Projekt konvertiert werden soll (TwinCAT 3.1 Build 4022 oder 4024). Nach Bestätigung der Konvertierung mit <b>Convert</b> wird das Projekt geschlossen und kompatibel zu der ausgewählten Version abgespeichert.</p> <p>Beachten Sie, dass die Projektdaten bei der Konvertierung verändert werden und Einstellungen und Eigenschaften späterer Versionen verloren gehen. Die Konvertierung ist daher nicht für einen mehrfachen Wechsel zwischen verschiedenen Versionen geeignet.</p>
---	--

### Write events to event log

Mit Hilfe der folgenden beiden Optionen können Sie konfigurieren, ob und welche Art von User-Events Sie loggen möchten. Als User-Events gelten dabei:

- Einloggen
- Ausloggen
- Download
- Online-Change
- SPS-Projekt starten
- SPS-Projekt stoppen
- Einzelzyklus
- Reset kalt
- Reset Ursprung
- Haltepunkt setzen
- Haltepunkt entfernen

Die Events werden über die Laufzeit an den Eventlogger gesendet. Zusätzlich werden die Events im Meldungsfenster (Error List) ausgegeben, falls die Eventlogger-Option **Output as Task Item** auf True gestellt ist.

Wenn Sie die im Folgenden beschriebenen Optionen ändern, ist beim nächsten Einloggen ein Online-Change erforderlich. Anschließend sind die Einstellungen aktiv.

Information	Verfügbar ab TwinCAT 3.1 Build 4026  <input type="checkbox"/> (Voreinstellung): Es werden keine Meldungen geloggt, wenn erfolgreiche User-Events stattfinden.  <input checked="" type="checkbox"/> : Wenn eins der o. g. User-Events stattfindet und erfolgreich ist, wird eine entsprechende Information über diesen Vorgang mitgeloggt.
Error	Verfügbar ab TwinCAT 3.1 Build 4026  <input type="checkbox"/> (Voreinstellung): Es werden keine Meldungen geloggt, wenn fehlgeschlagene User-Events stattfinden.  <input checked="" type="checkbox"/> : Wenn bei einem der o. g. User-Events ein Fehler auftritt, wird eine entsprechende Information über diesen fehlgeschlagenen Vorgang mitgeloggt. Dies ist z. B. der Fall, wenn ein Login aufgrund eines Fehlers in der Laufzeit fehlschlägt.

**General**

(Identische) Objekte mit der gleichen ID mehrfach in einer Solution erlauben  (bzw. „Allow (identical) objects with the same ID multiple times in one solution“)	<input type="checkbox"/> (Voreinstellung): Die SPS-Objekte (z. B. POU's) werden mit Hilfe ihrer GUID identifiziert, welche in der gesamten Solution jeweils einmalig vergeben ist. Für die üblichen Anwendungsfälle kann und sollte diese Option deaktiviert bleiben.  <input checked="" type="checkbox"/> : Die aktivierte Option ist ausschließlich für das Szenario relevant, bei dem ein Objekt innerhalb einer Solution mehrfach mit der gleichen ID/GUID existiert. Für diesen Fall muss sich die Identifikationsart der Objekte ändern, sodass die GUIDs der SPS-Objekte mit denen des SPS-Projekts verknüpft werden (mittels XOR). Dadurch werden Änderungen der GUIDs der SPS-Objekte vermieden, wenn diese mehrmals in verschiedenen Projekten innerhalb einer Solution verwendet werden.  <b>Informationen:</b> <ul style="list-style-type: none"> <li>• Wenn Sie den Zustand der Option ändern, muss das Projekt neugeladen werden. Zudem ist beim nächsten Einloggen ein <b>Download</b> erforderlich.</li> <li>• Wenn die Option <b>aktiviert</b> ist, ändert sich die Identifikationsart aller Objekte dieses SPS-Projekts.</li> <li>• Wenn die Option <b>aktiviert</b> ist, können die folgenden SPS-Funktionalitäten <b>nicht oder nicht vollumfänglich genutzt</b> werden:                         <ul style="list-style-type: none"> <li>◦ Visualisierung</li> <li>◦ UML SC</li> <li>◦ Security-Management</li> </ul> </li> </ul> Diese Option wurde zu TwinCAT 3.1 Build 4026.x umbenannt. Die vorherige Benennung lautete: „ID-Änderungen in TwinCAT-Dateien minimieren“ bzw. „Minimize ID changes in TwinCAT files“. Das Verhalten der Option bleibt unverändert.
--	---

**5.4.5 SPS-Projekteinstellungen**

**Funktion:** Dieser Befehl öffnet einen Editor, in dem Projekteinstellungen definiert werden können.

**Aufruf:** Doppelklick auf das SPS-Projekt im **Projektmappen-Explorer**

**Siehe auch:**

- Dokumentation PLC: Projekt konfigurieren

### 5.4.5.1 Registerkarte Projekt

Projekt
Settings

Project Name:  Id:

Project Path:

Project Type:  Port:

Project Guid:

Encryption:

Autostart Boot Project  
  Symbolic Mapping  
  Force Multi Instance

Kommentar:

Project Name	Name des SPS-Projekts
Id	ID des SPS-Projekts
Project Path	Pfad zum Ablageort des SPS-Projekts
Project Type	Projekttyp
Port	AMS-Portnummer des Laufzeitsystems
Project Guid	GUID des SPS-Projets
Encryption	Verschlüsselung des Bootprojekts <ul style="list-style-type: none"> <li>No boot project encryption (default)</li> <li>Encrypt boot project</li> </ul>
Autostart Boot Project	<input checked="" type="checkbox"/> Nach dem Start der TwinCAT-Laufzeitumgebung, wird das SPS-Bootprojekt automatisch geladen und gestartet. Die Einstellung wird direkt auf das gerade ausgewählte Zielsystem übertragen. Die Einstellung wird nicht im TwinCAT-Projekt gespeichert. Die Option entspricht dem Befehl <b>Autostart Boot Project</b> im Kontextmenü des SPS-Projektknotens im Projektmappen-Explorer.
Symbolic Mapping	<input checked="" type="checkbox"/> Symbolisches Mapping ist aktiviert.
Force Multi Instance	<input checked="" type="checkbox"/> Möglichkeit zum Einloggen von mehrfachen Instanzen des SPS-Projekts aktiviert.
Kommentar	Kommentarfeld
Compiler Defines (Verfügbar ab TC3.1 Build 4024)	
Manual	Hier können eigene Compilerdefinitionen auf System Manager-Ebene festgelegt werden, die an das SPS-Projekt weitergegeben werden. Die Definitionen sind in den SPS-Projekteigenschaften unter der Kategorie Übersetzen als <u>System-Compilerdefinitionen</u> [ <a href="#">▶_103</a> ] eingetragen.
Implicit	<input checked="" type="checkbox"/> Die Namen der ausgewählten Projektvariante, sowie aller Gruppen, zu denen die Projektvariante gehört, werden automatisch als Compilerdefinition gesetzt und an das SPS-Projekt weitergegeben. Die Definitionen sind in den SPS-Projekteigenschaften unter der Kategorie Übersetzen als <u>System-Compilerdefinitionen</u> [ <a href="#">▶_103</a> ] eingetragen.  <b>Hinweis:</b> Um diese Checkbox zu aktivieren, müssen die <u>Defines</u> für das Variantenmanagement freigegeben werden.

**Siehe auch:**

- Variantenmanagement: Konzept: [Integration ins SPS-Projekt](#)
- Befehl Eigenschaften (SPS-Projekt): [Kategorie Übersetzen: System-Compilerdefinitionen \[▶ 103\]](#)

**5.4.5.2 Registerkarte Settings**

**Target Archive**

Im Gruppenfeld **Target Archive** können Sie einstellen, welche Informationen beim Anlegen eines Bootprojekts zusammen mit anderen Daten auf das Zielsystem übertragen werden.

Login Information	COMPILEINFO-Datei, die die Compiler-Informationen des SPS-Projekts enthält.
Suppress Temporary	Wählt der Anwender beim Einloggen auf das Zielsystem zusätzlich die Option „Update Bootprojekt“ ab, so wird die COMPILEINFO-Datei temporär nicht mit auf das Zielsystem geladen. Dadurch wird der Download/ Online-Change beschleunigt.
Project Sources	Quellcode-Dateien des SPS-Projekts in lesbarer Quellcode-Form.
Compiled Libraries	Bibliotheken, die in kompilierter Form im SPS-Projekt verwendet werden.
Source Libraries	Bibliotheken, die in lesbarer Quellcode-Form im SPS-Projekt verwendet werden.

**File/E-Mail Archive**

Im Gruppenfeld **File/E-Mail Archive** können Sie einstellen, welche Informationen beim [Archivieren eines SPS-Projekts \[▶ 54\]](#), eines [TwinCAT-Projekts \[▶ 269\]](#) oder einer [Projektmappe \[▶ 52\]](#) gespeichert werden. Wenn Sie die entsprechende Checkbox aktivieren, werden die in der folgenden Tabelle beschriebenen Dateien im Projektarchiv gespeichert.

Login Information	COMPILEINFO-Datei, die die Compiler-Informationen des SPS-Projekts enthält.
Project Sources	Quellcode-Dateien des SPS-Projekts in lesbarer Quellcode-Form.
Compiled Libraries	Bibliotheken, die in kompilierter Form im SPS-Projekt verwendet werden.
Source Libraries	Bibliotheken, die in lesbarer Quellcode-Form im SPS-Projekt verwendet werden.
Core Dump	Core-Dump-Datei, die sich im Projektverzeichnis des SPS-Projekts befindet, sowie die Compile-Info-Dateien, die sich im „_CompileInfo“-Ordner des Projektverzeichnisses befinden.  Hinweis: Die Compile-Info-Dateien werden bei aktivierter „Core Dump [▶ 150]“-Einstellung ebenfalls im Archiv gespeichert, da diese Dateien benötigt werden, um den Core Dump nutzen zu können.

**• Weiteergabe von Quellcode**

**i** Wenn Sie bei den Einstellungen für das Target- oder das File/E-Mail-Archiv konfiguriert haben, dass in einem dieser Archive die Projekt-Sourcen und/oder Source-Bibliotheken enthalten sein sollen, beachten Sie bei der Weiteergabe/ Auslieferung des Zielsystems oder bei Weiteergabe des File-/ E-Mail-Archivs, dass die Projekt-Sourcen und/ oder die im Projekt verwendeten Source-Bibliotheken (\*.library) in lesbarer Quellcode-Form im ZIP-Archiv enthalten sind.

Beachten Sie dieses auch bei der Konfiguration der oben beschriebenen Einstellungen sowie beim Speichern und Referenzieren von Bibliotheken (\*.library vs. \*.compiled-library).

Weitere Informationen zum Thema Bibliotheksverwaltung finden Sie im Abschnitt Bibliotheken verwenden.

Informationen zu dem Thema Sourcecode-Verschlüsselung finden Sie in der Dokumentation [Security Management](#).

**Target Files**

Im Gruppenfeld Target Files können Sie einstellen, welche Informationen beim Anlegen eines Bootprojekts auf das Zielsystem in den Ordner `\Boot\Plc` übertragen werden.

TMC File	TMC-Datei (TwinCAT Module Class) eines SPS-Projekts
TPY File	TPY-Datei (enthält u. a. Projektinformationen, Routing-Informationen, Compiler-Informationen, Zielsystem-Informationen)

**Boot Project**

Verfügbar ab TwinCAT 3.1 Build 4026

Im Gruppenfeld **Boot Project** können Sie einstellen, ob das Bootprojekt auf dem Zielsystem bei Download oder Online-Change standardmäßig aktualisiert werden soll. Durch das Aktivieren oder Deaktivieren der Checkbox im Login-Dialog, der bei Änderungen am Projekt angezeigt wird (siehe [Befehl Einloggen \[▶ 156\]](#)), wird die hier festgelegte Standardeinstellung einmalig überschrieben.

Update Boot Project on Online Change	Bei einem Online-Change wird das Bootprojekt auf dem Zielsystem standardmäßig aktualisiert.
Update Boot Project on Download	Bei einem Download wird das Bootprojekt auf dem Zielsystem standardmäßig aktualisiert.

**Target Behavior**

Im Gruppenfeld **Target Behavior** können Sie einstellen, wie das Zielsystem mit persistenten Daten sowie mit Zeigern und Referenzen umgeht.

Clear Invalid Persistent Data	Das Backup der gespeicherten persistenten Daten wird ignoriert. So wird sichergestellt, dass möglicherweise ungültige Daten nicht übernommen und verworfen werden.  Siehe: <a href="#">Backup von persistenten Daten [► 124]</a>
Restore moved Pointers/References	Verfügbar ab TwinCAT 3.1 Build 4026.11  Bei einem Online-Change werden die Werte aller SPS-Zeiger automatisch aktualisiert, sodass der jeweilige Zeiger auf die gleiche Variable bzw. auf das gleiche Objekt verweist wie vor dem Online-Change. Dadurch behält ein Zeiger nach dem Online-Change seine Gültigkeit, auch wenn die Variable, auf die gezeigt wird, während des Online-Changes an eine andere Speicherposition verschoben wird.  Da das Aktualisieren der Zeiger Zeit benötigt, beschleunigt das Deaktivieren dieser Option den Online-Change. In diesem Fall ist die weitere Gültigkeit der Zeiger nicht gewährleistet. Sie sollten daher einen Mechanismus zur Aktualisierung der Zeiger manuell implementieren.  Siehe: <a href="#">Zeiger / POINTER</a>

## Backup von persistenten Daten

Persistente Daten werden beim TwinCAT-System-Stopp/Shutdown regulär in einer .bootdata-Datei *im* `TwinCAT\Boot-Ordner` gespeichert. Beim nächsten Systemstart (TwinCAT Run Mode) wird diese Datei eingelesen und die persistenten Variablen werden im Laufzeitsystem mit den Werten aus der Datei initialisiert. Die .bootdata-Datei wird vom System zu .bootdata-old umbenannt.

Diese Backup-Datei (.bootdata-old) der persistenten Daten wird beim Systemstart eingelesen, wenn die Datei (.bootdata) der persistenten Daten nicht vorhanden ist. Dieser Fall ist eine Ausnahme, kann aber z. B. dann eintreten, wenn ein IPC ohne USV einen Stromausfall erfährt und TwinCAT somit nicht ordnungsgemäß herunterfahren konnte.

- Wenn vorhersehbar ist, dass der Inhalt der Backup-Datei bei einem neuen Systemstart nicht nutzbar ist, können Sie die Option **Clear Invalid Persistent Data** aktivieren, um die Backup-Datei zu ignorieren. Dies kann z. B. der Fall sein, wenn Chargen-Informationen oder Werkzeugdaten in einer Produktion abgespeichert werden und aktuell sein müssen.
- Wenn die Struktur der persistenten Daten (deren Datentypen oder Symbolpfade im Programmcode) aufgrund von Online-Changes verändert wird, dann macht es keinen Sinn, später eine veraltete Datei der persistenten Daten zu laden. In diesem Fall sollten Sie die Option **Clear Invalid Persistent Data** vorausschauend aktivieren.

In beiden Fällen sollten Sie zusätzlich dafür sorgen, dass eine aktuelle Datei der persistenten Daten verfügbar ist. Hierfür gibt es einerseits Funktionsbausteine wie `FB_WritePersistentData` (PLC Lib `Tc2_Uilities`) und andererseits die USV-Absicherung für plötzliche Stromausfälle.

Grundsätzlich sollten bei der Verwendung von persistenten Daten die entsprechenden Flags (`BootDataLoaded` und `OldBootData`) aus der globalen Struktur `PlcAppSystemInfo` ausgewertet werden (siehe Dokumentation System > Globale Datentypen).

Wenn weder die reguläre Datei noch die Backup-Datei geladen werden können oder nicht vorhanden sind, werden die als `PERSISTENT` markierten Variablen ebenso wie andere „normale“ Variablen neu initialisiert, entweder mit ihren explizit vorgegebenen Initialwerten oder mit den Standardinitialisierungen.

### Siehe auch:

- Dokumentation PLC: Remanente Variablen - `RETAIN`, `PERSISTENT`

## 5.5 Erstellen

### 5.5.1 Befehl Projektmappe erstellen

Symbol: 

**Funktion:** Der Befehl startet den Übersetzungsprozess bzw. die Codegenerierung für alle in der Projektmappe enthaltenen Projekte.

**Aufruf:** Menü **Erstellen** oder Kontextmenü der Projektmappe

**Voraussetzung:** Die Projektmappe ist selektiert.

Alle in der Projektmappe enthaltenen Projekte werden der Reihe nach übersetzt. Dies betrifft auch die unterhalb eines TwinCAT-Projekts eingebundenen Projekte (SPS, C++, etc.). Die dabei für ein SPS-Projekt durchgeführten Schritte werden im Abschnitt [Befehl SPS-Projekt erstellen \[▶ 126\]](#) beschrieben.

## 5.5.2 Befehl Projektmappe neu erstellen

**Funktion:** Der Befehl startet den Übersetzungsprozess für alle in der Projektmappe enthaltenen Projekte, auch wenn sie zuletzt fehlerfrei übersetzt wurden.

**Aufruf:** Menü **Erstellen** oder Kontextmenü der Projektmappe

**Voraussetzung:** Die Projektmappe ist selektiert.

Bei einer Neuerstellung der Projektmappe wird die Projektmappe zunächst bereinigt (siehe auch: [Befehl Projektmappe bereinigen \[▶ 125\]](#)) und anschließend erstellt (siehe auch: [Befehl Projektmappe erstellen \[▶ 124\]](#)).

**Siehe auch:**

- [Befehl SPS-Projekt neu erstellen \[▶ 127\]](#)

## 5.5.3 Befehl Projektmappe bereinigen

**Funktion:** Der Befehl startet die Bereinigung für alle in der Projektmappe enthaltenen Projekte.

**Aufruf:** Menü **Erstellen** oder Kontextmenü der Projektmappe

**Voraussetzung:** Die Projektmappe ist selektiert.

Alle in der Projektmappe enthaltenen Projekte werden der Reihe nach bereinigt. Dies betrifft auch die unterhalb eines TwinCAT-Projekts eingebundenen Projekte (SPS, C++, etc.). Die dabei für ein SPS-Projekt durchgeführten Schritte werden im Abschnitt [Befehl SPS-Projekt bereinigen \[▶ 127\]](#) beschrieben.

## 5.5.4 Befehl Überprüfe alle Objekte

**Funktion:** Der Befehl veranlasst einen Übersetzungslauf, also eine Syntaxprüfung, für alle Objekte, die sich im Projektbaum des SPS-Projekts befinden. Dies ist in erster Linie bei der Erstellung von Bibliotheken bzw. bei der Bearbeitung von Bibliotheksprojekten nützlich.

**Aufruf:** Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) im **Projektmappen-Explorer**

Im Gegensatz zu dem [Befehl SPS-Projekt erstellen \[▶ 126\]](#), bei dem nur die verwendeten Objekte überprüft werden, werden bei der Ausführung dieses Befehls alle Objekte des SPS-Projekts syntaktisch überprüft.



Der Befehl führt nicht zur Codegenerierung. Es wird auch keine Datei mit Informationen zum Übersetzungslauf im Projektverzeichnis angelegt.

---

## 5.5.5 Befehl TwinCAT-Projekt erstellen

Symbol: 

**Funktion:** Der Befehl startet den Übersetzungsprozess bzw. die Codeerzeugung für das gerade aktive TwinCAT-Projekt.

**Aufruf:** Menü **Erstellen**, wenn aktuell ein TwinCAT-Projekt selektiert ist, oder Kontextmenü des TwinCAT-Projekts

**Voraussetzung:** Das TwinCAT-Projekt ist selektiert.

Alle in dem TwinCAT-Projekt enthaltenen Projekte (SPS, C++, etc.) werden der Reihe nach übersetzt. Die dabei für ein SPS-Projekt durchgeführten Schritte werden im Abschnitt [Befehl SPS-Projekt erstellen \[▶ 126\]](#) beschrieben.

**Siehe auch:**

- [Befehl TwinCAT-Projekt neu erstellen \[▶ 126\]](#)

## 5.5.6 Befehl TwinCAT-Projekt neu erstellen

**Funktion:** Der Befehl startet den Übersetzungsprozess bzw. die Codeerzeugung für das gerade aktive TwinCAT-Projekt, auch wenn es zuletzt fehlerfrei übersetzt wurde.

**Aufruf:** Menü **Erstellen**, wenn aktuell ein TwinCAT-Projekt selektiert ist, oder Kontextmenü des TwinCAT-Projekts

**Voraussetzung:** Das TwinCAT-Projekt ist selektiert.

Bei einer Neuerstellung des Projekts wird das TwinCAT-Projekt zunächst bereinigt (siehe auch: [Befehl TwinCAT-Projekt bereinigen \[▶ 126\]](#)) und anschließend erstellt (siehe auch: [Befehl TwinCAT-Projekt erstellen \[▶ 126\]](#)).

## 5.5.7 Befehl TwinCAT-Projekt bereinigen

**Funktion:** Der Befehl löscht die lokale Übersetzungsinformation für das gerade aktive SPS-Projekt und aktualisiert das Sprachmodell aller Objekte.

**Aufruf:** Menü **Erstellen**, wenn aktuell ein TwinCAT-Projekt selektiert ist, oder Kontextmenü des TwinCAT-Projekts

**Voraussetzung:** Das TwinCAT-Projekt ist selektiert.

Alle in dem TwinCAT-Projekt enthaltenen Projekte (SPS, C++, etc.) werden der Reihe nach bereinigt. Die dabei für ein SPS-Projekt durchgeführten Schritte werden im Abschnitt [Befehl SPS-Projekt bereinigen \[▶ 127\]](#) beschrieben.

**Siehe auch:**

- [Befehl TwinCAT-Projekt neu erstellen \[▶ 126\]](#)

## 5.5.8 Befehl SPS-Projekt erstellen

Symbol: 

**Funktion:** Der Befehl startet den Übersetzungsprozess bzw. die Codeerzeugung für das gerade aktive SPS-Projekt.

**Aufruf:** Menü **Erstellen**, wenn aktuell ein SPS-Projekt selektiert ist, oder Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist selektiert.

Bei der Übersetzung führt TwinCAT eine syntaktische Prüfung aller verwendeten Objekte des SPS-Projekts durch. Der Übersetzungsvorgang wird automatisch immer durchgeführt, wenn Sie das Projekt mit einem geänderten Programm einloggen möchten. Nach Abschluss der Überprüfung zeigt TwinCAT eventuelle Fehlermeldungen oder Warnungen in der Ansicht [Fehlerliste](#) [► 87] an.

Außerdem wird beim Erstellen des Projektes die Übersetzungsinformation des SPS-Projekts angelegt und in einer lokalen Datei (\*.compileinfo) im Projektordner gespeichert.

Wenn das Programm seit dem letzten fehlerfreien Übersetzungsprozess nicht mehr verändert wurde, wird es nicht neu übersetzt. Wenn die syntaktische Prüfung dennoch wiederholt werden soll, verwenden Sie den [Befehl SPS-Projekt neu erstellen](#) [► 127].

## 5.5.9 Befehl SPS-Projekt neu erstellen

**Funktion:** Der Befehl startet den Übersetzungsprozess bzw. die Codeerzeugung für das gerade aktive SPS-Projekt, auch wenn es zuletzt fehlerfrei übersetzt wurde.

**Aufruf:** Menü **Erstellen**, wenn aktuell ein SPS-Projekt selektiert ist, oder Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist selektiert.

Bei einer Neuerstellung des Projekts wird das Projekt zunächst bereinigt (siehe auch: [Befehl SPS-Projekt bereinigen](#) [► 127]) und anschließend erstellt (siehe auch: [Befehl SPS-Projekt erstellen](#) [► 126]).

## 5.5.10 Befehl SPS-Projekt bereinigen

**Funktion:** Der Befehl aktualisiert das Sprachmodell aller Objekte des gerade aktiven SPS-Projekts.

**Aufruf:** Menü **Erstellen**, wenn aktuell ein SPS-Projekt selektiert ist, oder Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist selektiert.

Wenn das SPS-Projekt bereinigt wird, wird lediglich das Sprachmodell aller Objekte im SPS-Projekt aktualisiert. Die Übersetzungsinformation auf dem Zielsystem bleibt bestehen.

**Siehe auch:**

- [Befehl SPS-Projekt neu erstellen](#) [► 127]

## 5.6 Debuggen

### 5.6.1 Befehl Neuer Haltepunkt

Symbol: 

**Funktion:** Der Befehl öffnet den Dialog **Eigenschaften Haltepunkt**.

**Aufruf:** Menü **Debuggen**, Schaltfläche  **New** in der Ansicht **Haltepunkt (PLC > Fenster > Haltepunkte)**.

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb.



Mit dem Befehl **Haltepunkt umschalten** können Sie im Onlinebetrieb einen neuen Haltepunkt direkt an der aktuellen Cursor-Position setzen

**Siehe auch:**

- [Befehl Haltepunkt umschalten \[► 131\]](#)
- [Menü PLC: Befehl Haltepunkte \[► 143\]](#)
- [Dokumentation PLC: Haltepunkte verwenden](#)

**Dialog Eigenschaften Haltepunkt**

Haltepunkt sofort aktivieren	<input checked="" type="checkbox"/> Der Haltepunkt ist aktiviert. <input type="checkbox"/> Der Haltepunkt ist nicht aktiviert. Zur späteren Aktivierung klicken Sie in der Ansicht <b>Haltepunkte</b> auf die Schaltfläche  .
------------------------------	---

**Registerkarte Bedingung**

Der Dialog legt fest, unter welchen Voraussetzungen die Programmabarbeitung am Haltepunkt stoppen soll.

**Tasks**

Nur anhalten, falls der Haltepunkt in einer der folgenden Tasks erreicht wird	<input checked="" type="checkbox"/> : TwinCAT wertet den Haltepunkt nur aus, wenn er von bestimmten Tasks erreicht wird. Die gewünschten Tasks müssen aktiviert werden. Sie können beispielsweise eine einzige „Debug Task“ definieren und damit verhindern, dass beim Debugging auch andere Tasks betroffen werden, die den Baustein ebenfalls verwenden.
---	---

**Trefferanzahl**

Trefferanzahl	<p>Immer anhalten: Das Programm stoppt immer an diesem Haltepunkt.</p> <p>Alternativ: Das Programm hält am Haltepunkt an, wenn der Haltepunkt so oft getroffen wurde, wie im Folgenden definiert ist (gewünschte Trefferzahl eintragen oder aus der Nummernliste auswählen):</p> <ul style="list-style-type: none"> <li>• Anhalten, wenn die Trefferanzahl gleich</li> <li>• Anhalten, wenn die Trefferanzahl ein Vielfaches von</li> <li>• Anhalten, wenn die Trefferanzahl größer oder gleich</li> </ul>
---------------	--

**Bedingung**

Halt, wenn TRUE	<p>Definition bedingter Haltepunkte. Die Bedingung können Sie nur im Onlinebetrieb eintragen.</p> <p> : TwinCAT wertet die angegebene Bedingung aus und hält das Programm an diesem Haltepunkt nur an, wenn das Ergebnis TRUE ist. Als Bedingung können Sie gültige boolesche Ausdrücke eintragen. Beispiele: <math>x &gt; 100</math>, <math>x[y]=z</math>, <math>a \text{ AND } b</math>, <math>\text{boolVar}</math>.</p>
-----------------	--



Die Verwendung bedingter Haltepunkte verlangsamt die Code-Ausführung, auch wenn die Bedingung nicht TRUE ist.

**Registerkarte Ort**

✕

Eigenschaften Haltepunkt

Bedinauna   **Ort**   Ausführungsapunkt   Einstellungen

Ort

POU: Simulation [TwinCAT\_Project6: SPS: Project6]

Position: Zeile 8, Spalte 9 (Impl)

Instanzen

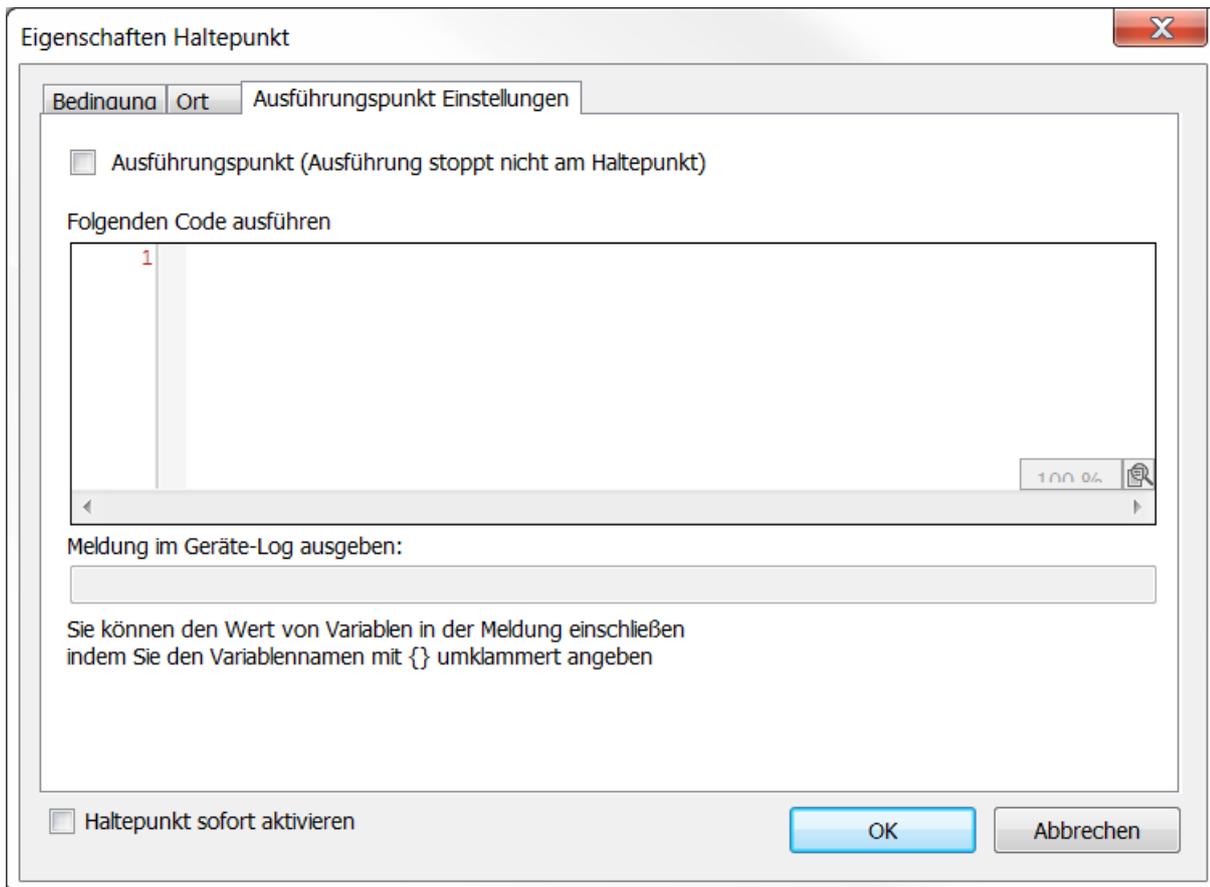
Ausgewählte Instanzen: 0

Haltepunkt sofort aktivieren
 OK
Abbrechen

POU	Baustein des aktiven SPS-Projekts, in dem der Haltepunkt positioniert sein soll.
Position	Position des Haltepunkts in der POU. Angabe in Form von Zeilen- und Spaltennummern (Texteditor) oder als Netzwerk- oder Elementnummern.
Instanzen	Bei Funktionsbausteinen müssen Sie festlegen, ob der Haltepunkt in der Implementierung oder in einer Instanz gesetzt werden soll  <input checked="" type="checkbox"/> TwinCAT setzt den Haltepunkt in der Instanz. Bei dieser Option wählen Sie den Instanzpfad aus.  <input type="checkbox"/> TwinCAT setzt den Haltepunkt in der Implementierung.

**Registerkarte Ausführungspunkt**

Hier kann ein bestehender Haltepunkt in einen Ausführungspunkt umgewandelt werden.



Ausführungspunkt (Ausführung stoppt nicht am Haltepunkt)	<input checked="" type="checkbox"/> : Der Haltepunkt wird zum Ausführungspunkt. Die Abarbeitung hält an diesem Punkt nicht an, es wird jedoch der angegebene Code ausgeführt.  aktiviert: ● , deaktiviert: ○
Folgenden Code ausführen	Code, der beim Erreichen des Ausführungspunkts ausgeführt wird. Schleifenkonstruktionen (For, While) und IF- oder CASE-Ausdrücke sind nicht möglich.
Meldung im Geräte-Log ausgeben	Diese Option ist nicht verfügbar.

**Siehe auch:**

- Dokumentation PLC: Haltepunkte verwenden

## 5.6.2 Befehl Haltepunkt bearbeiten

Symbol: 

**Funktion:** Der Befehl öffnet den Dialog **Eigenschaften Haltepunkte**.

**Aufruf:** Menü **Debuggen**, Schaltfläche  in der Ansicht **Haltepunkte (PLC > Fenster > Haltepunkte)**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Der Cursor steht auf einem Haltepunkt.

**Siehe auch:**

- Befehl Neuer Haltepunkt > [Dialog Eigenschaften Haltepunkt \[► 128\]](#)
- Dokumentation PLC: Haltepunkte verwenden

## 5.6.3 Befehl Haltepunkt aktivieren

Symbol: 

**Funktion:** Der Befehl aktiviert einen deaktivierten Haltepunkt.

**Aufruf:** Menü **Debuggen**, Schaltfläche  in der Ansicht **Haltepunkte (PLC > Fenster > Haltepunkte)**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Der Cursor steht auf einem deaktivierten Haltepunkt.

**Siehe auch:**

- Dokumentation PLC: Haltepunkte verwenden

## 5.6.4 Befehl Haltepunkt deaktivieren

Symbol: 

**Funktion:** Der Befehl deaktiviert einen aktivierten Haltepunkt.

**Aufruf:** Menü **Debuggen**, Schaltfläche  in der Ansicht **Haltepunkte (PLC > Fenster > Haltepunkte)**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Der Cursor steht auf einem aktivierten Haltepunkt.

**Siehe auch:**

- Dokumentation PLC: Haltepunkte verwenden

## 5.6.5 Befehl Haltepunkt umschalten

Tastaturkürzel: **[F9]**

**Funktion:** Der Befehl setzt einen Haltepunkt oder löscht einen bestehenden.

**Aufruf:** Menü **Debuggen**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Der Cursor steht auf einem Haltepunkt.

**Siehe auch:**

- Dokumentation PLC: Haltepunkte verwenden

## 5.6.6 Befehl Prozedurschritt

Symbol: 

Tastaturkürzel: **[F10]**

**Funktion:** Der Befehl führt die Anweisung, an der das Programm aktuell steht, aus und hält vor der nächsten Anweisung im Programmierbaustein an.

**Aufruf:** Menü **Debuggen**, TwinCAT SPS **Symboleistenoptionen**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Das Programm steht an einer Halteposition (Debugbetrieb).

Wenn die auszuführende Anweisung einen Aufruf enthält (von einem Programm, einer Funktionsbaustein-Instanz, einer Funktion, einer Methode oder einer Aktion), wird der untergeordnete Programmierbaustein vollständig in einem Schritt durchlaufen und zum Aufruf zurückgekehrt. Vor der nächsten Anweisung (in der nächsten Codezeile) wird angehalten.

Wählen Sie den Befehl **Einzelschritt**, um in einen untergeordneten Baustein zu springen und diesen schrittweise auszuführen.

**Siehe auch:**

- [Befehl Einzelschritt \[► 132\]](#)
- Dokumentation PLC: Schrittweises Abarbeiten des Programms (Stepping)

## 5.6.7 Befehl Einzelschritt

Symbol: 

Tastaturkürzel: **[F11]**

**Funktion:** Der Befehl führt die Anweisung, an der das Programm aktuell steht, aus und hält vor der nächsten Anweisung.

**Aufruf:** Menü **Debuggen**, TwinCAT SPS **Symboleistenoptionen**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Das Programm steht an einer Halteposition (Debugbetrieb).

Wenn die auszuführende Anweisung einen Aufruf enthält (von einem Programm, einer Funktionsbaustein-Instanz, einer Funktion, einer Methode oder einer Aktion), wird in diesen untergeordneten Programmierbaustein gesprungen. Dessen Code erscheint in einem eigenen Editor. Die erste Anweisung dort wird ausgeführt und vor der nächsten Anweisung wird angehalten. Die neue aktuelle Halteposition ist dann im aufgerufenen Programmierbaustein.

Wählen Sie den Befehl **Prozedurschritt**, um im aktuell aktiven Programmierbaustein zu bleiben und den Aufruf in einem Schritt zu durchlaufen.

**Siehe auch:**

- [Befehl Prozedurschritt \[► 132\]](#)
- Dokumentation PLC: Schrittweises Abarbeiten des Programms (Stepping)

## 5.6.8 Befehl Ausführen bis Rücksprung

Symbol: 

Tastaturkürzel: **[Umschalt] + [F11]**

**Funktion:** Der Befehl führt das Programm bis zum nächsten Rücksprung aus und hält danach an.

**Aufruf:** Menü **Debuggen**, **TwinCAT SPS Symbolleistenoptionen**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Das Programm steht an einer Halteposition (Debugbetrieb).

Wenn die aktuelle Halteposition in einem untergeordneten Programmierbaustein ist, wird dieser bis zum Ende durchlaufen. Dann wird zur Aufrufstelle im aufrufenden Programmierbaustein zurückgesprungen und dort angehalten (in der Zeile mit dem Aufruf).

Wenn die aktuelle Halteposition im Hauptprogramm ist, wird der Programmierbaustein bis zum Ende durchlaufen. Dann wird zurück an den Anfang (an den Programmstart an die erste Codezeile im Programmierbaustein) gesprungen und dort angehalten.

**Siehe auch:**

- Dokumentation PLC: Schrittweises Abarbeiten des Programms (Stepping)

## 5.6.9 Befehl Ausführen bis Cursor

Symbol: 

**Funktion:** Der Befehl führt ein Programm bis zu einer mit dem Cursor gekennzeichneten Position aus.

**Aufruf:** Kontextmenü

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Das Programm steht an einer Halteposition (Debugbetrieb). Sie haben mit dem Cursor eine beliebige Codezeile in einem beliebigen Programmierbaustein gekennzeichnet.

Die Anweisungen, die zwischen der aktuellen Halteposition und der Cursorposition liegen, werden in einem Schritt ausgeführt. Dann hält die Ausführung an der Cursorposition an, die damit zur nächsten Halteposition wird. Beachten Sie, dass die Codezeile, an der Sie den Cursor gesetzt haben, erreicht, aber nicht ausgeführt wird.

**Siehe auch:**

- Dokumentation PLC: Schrittweises Abarbeiten des Programms (Stepping)

## 5.6.10 Befehl Nächste Anweisung anzeigen

Symbol: 

**Funktion:** Der Befehl zeigt die Programmanweisung an, die im nächsten Schritt abgearbeitet wird.

**Aufruf:** Menü **Debuggen**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Das Programm steht an einer Halteposition. Die Halteposition ist in einer für Sie nicht sichtbaren Codezeile.

Der Befehl bewirkt, dass das Fenster mit der aktuellen Halteposition, die im Code gelb und mit dem Symbol  gekennzeichnet ist, aktiv und die Halteposition sichtbar wird. Das ist nützlich, wenn Sie viele Editoren geöffnet haben und die Halteposition sich verdeckt in einem nicht aktiven Editor befindet.

**Siehe auch:**

- Dokumentation PLC: Schrittweises Abarbeiten des Programms (Stepping)

## 5.6.11 Befehl Nächste Anweisung festlegen

Symbol: 

**Funktion:** Der Befehl legt fest, welche Anweisung als nächstes ausgeführt wird.

**Aufruf:** Kontextmenü

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb. Das Programm steht an einer Halteposition (Debugbetrieb). Sie haben mit dem Cursor eine beliebige Codezeile in einem beliebigen Programmierbaustein gekennzeichnet

Die mit Cursor gekennzeichnete Codezeile wird zur aktuellen Halteposition, ohne die Anweisungen dazwischen oder die angesprungene Anweisung auszuführen.

**Siehe auch:**

- Dokumentation PLC: Schrittweises Abarbeiten des Programms (Stepping)

## 5.7 TwinCAT

### 5.7.1 Befehl Konfiguration aktivieren

Symbol: 

**Funktion:** Der Befehl aktiviert eine neue Konfiguration. Die vorherige alte Konfiguration wird überschrieben.

**Aufruf:** Menü **TwinCAT**, **TwinCAT XAE Base Symbolleistenoptionen**

Innerhalb des Bestätigungsfensters, das nach Ausführung dieses Befehls erscheint, können Sie einstellen, ob für alle SPS-Projekte des TwinCAT-Projekts die Einstellung **Autostart Boot Projekt** aktiviert werden soll.

**Siehe auch:**

- [Befehl Bootprojekt aktivieren \[► 273\]](#)
- [Befehl Autostart Bootprojekt \[► 273\]](#)

### 5.7.2 Befehl Restart TwinCAT System

Symbol: 

**Funktion:** Der Befehl startet TwinCAT im Run-Modus.

**Aufruf:** Menü **TwinCAT**, **TwinCAT XAE Base Symbolleistenoptionen**

### 5.7.3 Befehl Restart TwinCAT (Config Mode)

Symbol: 

**Funktion:** Der Befehl startet TwinCAT im Konfigurationsmodus (Config-Modus).

**Aufruf:** Menü **TwinCAT**, **TwinCAT XAE Base Symbolleistenoptionen**

### 5.7.4 Befehl Reload Devices

Symbol: 

**Funktion:** Der Befehl lädt die angelegten I/O-Geräte.

**Aufruf:** Menü **TwinCAT**, **TwinCAT XAE Base Symbolleistenoptionen**

## 5.7.5 Befehl Scan

Symbol: 

**Funktion:** Der Befehl startet einen Gerätescan. Das System sucht nach verfügbaren I/O-Geräten, verbundenen „Boxen“ und gegebenenfalls Busmodulen und IP-Link-Erweiterungsmodulen.

**Aufruf:** Menü **TwinCAT, TwinCAT XAE Base Symbolleistenoptionen**

**Voraussetzung:** In der TwinCAT-Projektstruktur im **Projektmappen-Explorer** ist das Objekt „I/O“ markiert.

## 5.7.6 Befehl Toggle Free Run State

Symbol: 

**Funktion:** Der Befehl setzt gefundene I/O-Geräte in den Free-Run-Modus. Das bedeutet, dass z. B. I/O-Kanäle von Busklemmen auf einen bestimmten Status gesetzt (geschrieben) werden können, ohne dass ein SPS-Projekt oder eine sonstige auslösende Task aktiv ist.

**Aufruf:** Menü **TwinCAT, TwinCAT XAE Base Symbolleistenoptionen**

**Voraussetzung:** Das System befindet sich aktuell im Konfigurationsmodus.



Wenn sich das Zielsystem zuvor im Run-Modus befand, muss der Befehl **Reload Devices** einmal ausgeführt werden, bevor die I/O-Treiber für das Gerät in den Free-Run-Status gesetzt werden können.

## 5.7.7 Befehl Show Online Data

Symbol: 

**Funktion:** Der Befehl stellt eine Verbindung mit dem ausgewählten Zielsystem her und stellt die auf dem Zielsystem aktiven Parameterwerte und Einstellungen in den entsprechenden Ansichten dar.

**Aufruf:** Menü **TwinCAT, TwinCAT XAE Base Symbolleistenoptionen**

## 5.7.8 Befehl Choose Target System

**Funktion:** Drop-down-Liste zur Auswahl des Zielgeräts für die Steuerungsapplikation.

**Aufruf:** **TwinCAT XAE Base Symbolleistenoptionen**

Wenn der Steuerungscode direkt in Ihre lokale Laufzeit Ihres Programmiergeräts geladen werden soll, wählen Sie den Eintrag <Lokal>. Wenn Sie ein anderes Zielgerät auswählen wollen, wählen Sie in der Drop-down-Liste den Eintrag **Zielsystem wählen**.

## 5.7.9 Befehl Show Sub Items

Symbol: 

**Funktion:** Der Befehl zeigt in der Übersichtsansicht eines Geräts die Unterelemente eines Elements mit deren Eigenschaften und Werte an. Der Befehl kann aktiviert oder deaktiviert werden. Der Befehl bezieht sich nicht auf die Darstellung der Elemente im TwinCAT-Projektbaum.

**Aufruf:** Menü **TwinCAT, TwinCAT Base XAE Symbolleistenoptionen**

Name	Online	Type	Size	>Addr...	In/Out	User ID	Linked to
Status		Status_E2F...	2.0	26.0	Input	0	
Transmit accepted		BIT	0.1	26.0	Input	0	
Receive request		BIT	0.1	26.1	Input	0	
Init accepted		BIT	0.1	26.2	Input	0	
Buffer full		BIT	0.1	26.3	Input	0	
Parity error		BIT	0.1	26.4	Input	0	
Framing error		BIT	0.1	26.5	Input	0	
Overrun error		BIT	0.1	26.6	Input	0	
Input length		USINT	1.0	27.0	Input	0	
Data In 0		USINT	1.0	28.0	Input	0	
Data In 1		USINT	1.0	29.0	Input	0	
Data In 2		USINT	1.0	30.0	Input	0	
Data In 3		USINT	1.0	31.0	Input	0	
Data In 4		USINT	1.0	32.0	Input	0	
Data In 5		USINT	1.0	33.0	Input	0	
Data In 6		USINT	1.0	34.0	Input	0	
Data In 7		USINT	1.0	35.0	Input	0	
Data In 8		USINT	1.0	36.0	Input	0	
Data In 9		USINT	1.0	37.0	Input	0	

## 5.7.10 Befehl Software Protection

Symbol: 

**Funktion:** Der Befehl öffnet den Dialog **Software Protection**.

**Aufruf:** Menü **TwinCAT**

In dem Software-Protection-Dialog können Sie die Security- und Benutzer-Einstellungen eines TwinCAT-Projekts definieren.

Weitere Informationen zu den Security- und Benutzer-Einstellungen finden Sie in der Dokumentation **Software Protection**.

## 5.7.11 Befehl Hide Disabled Items

Symbol: 

**Funktion:** Der Befehl ermöglicht es, deaktivierte Objekte im gesamten Projektbaum unsichtbar und wieder sichtbar zu schalten. Auf diese Weise können ausschließlich aktive Objekte dargestellt und die Übersichtlichkeit innerhalb des Projektbaums erhöht werden.

**Aufruf:** Menü **TwinCAT**, **TwinCAT XAE Base Symboleistenoptionen**

## 5.8 PLC

### 5.8.1 Fenster

#### 5.8.1.1 Befehl Speicheransicht anzeigen

Symbol: 

**Funktion:** Der Befehl öffnet die Ansicht **Speicher**. Hier können Sie sich im Onlinebetrieb einen Speicherauszug aus der Steuerung anzeigen lassen. In der Ansicht konfigurieren Sie, für welche Applikation und für welchen Bereich der Speicher dargestellt werden soll.

**Aufruf:** Menü **PLC > Fenster**

**Voraussetzung:** Die Steuerung unterstützt die Funktionalität grundsätzlich. Mindestens eine Applikation ist geladen und im Onlinebetrieb.

**● Applikationsabsturz durch unbedachte Änderungen**

**I** Sie können in der Speicheransicht die angezeigten Bytes überschreiben und die Änderungen auf die Steuerung übertragen. TwinCAT prüft nicht, ob die Änderungen zulässig sind.

Unbedachte Änderungen können die Applikation zum Absturz bringen!

Sie können die Startadresse des Speicherauszugs absolut angeben, indem Sie zum Beispiel den Adresswert aus einer Zeigervariablen eintragen. Alternativ können Sie die Startadresse relativ zu einem Speicherbereich der Steuerung angeben, indem Sie den Bereich und einen Offset angeben. In der Speicherdarstellung können Sie in die angrenzenden Speichersegmente navigieren, sowie den Speicherauszug in einer Datei speichern. Sie können die Daten in Hexadezimaldarstellung überschreiben und die Änderungen auf die Steuerung übertragen. Dafür stehen Ihnen die Befehle der Symbolleiste zur Verfügung.

Wenn Sie, beispielsweise bei der Fehlersuche, die Applikation schrittweise ausführen, aktualisiert TwinCAT die Ansicht des Speicherauszugs fortlaufend.

Genereller Aufbau des Fensters:

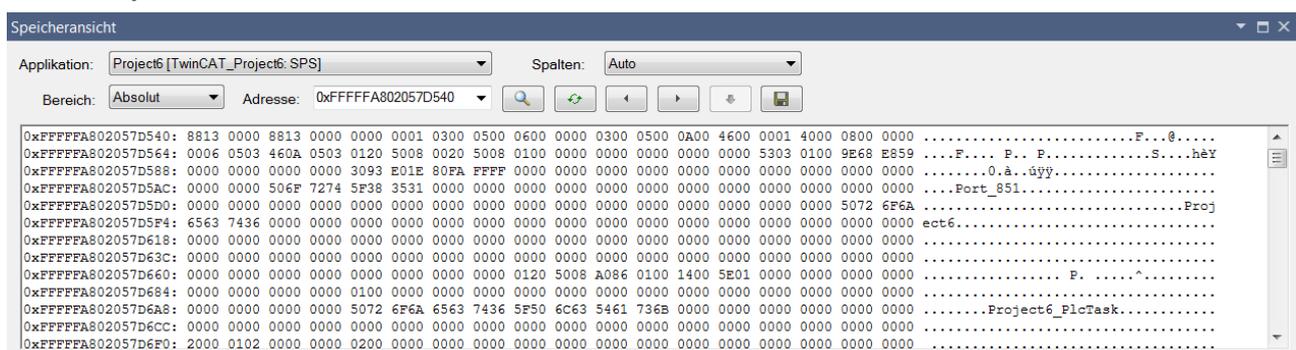
- Der Speicherauszug zeigt links die hexadezimalen Adressen des Speichers an.
- In der Mitte werden die Daten in Hexadezimaldarstellung angezeigt, wobei je Spalte je zwei Bytes dargestellt werden.
- Rechts werden die Daten als Text angezeigt. Daten, die kein darstellbares Zeichen repräsentieren, werden als „.“ (Punkt) angezeigt.

**Tipp:** Wenn Sie den Cursor in den hexadezimalen Daten platzieren, können Sie mit einem Rechtsklick die Adresse dieser Daten in die Zwischenablage kopieren (Befehl **Adresse kopieren**).

Im Folgenden wird zunächst die **Speicheransicht** erläutert. Anschließend werden die folgenden Anwendungsfälle beschrieben:

- Startadresse des Speicherauszugs über Variable festlegen
- Startadresse des Speicherauszugs über Angabe des Speicherbereichs (Area) festlegen
- Daten bearbeiten
- Speicherprüfungen für die aktive Applikation durchführen

**Ansicht Speicher**



Applikation	Auswahl des SPS-Projektes, für die die Speicheransicht dargestellt werden soll. Mit diesem Projekt müssen Sie auf der Steuerung eingeloggt sein. Es muss nicht das „aktive SPS-Projekt“ sein.
Bereich	<ul style="list-style-type: none"> <li>Absolut: Speicher wird direkt und vollständig adressiert. Die Adresse steht im Eingabefeld daneben.</li> <li>Area &lt;i&gt;: Speicherbereiche der Steuerung, beginnend mit Area 0. Speicherbereiche, die ausschließlich für Code reserviert sind, werden nicht angezeigt.</li> </ul>
Adresse	Absolute Startadresse des Core Dumps Voraussetzung: In Bereich ist <b>Absolut</b> ausgewählt.
Offset	Adressversatz zum gewählten Speicherbereich in Byte, zum Beispiel 0x0200, 16#0200 oder als Dezimalzahl 512 Voraussetzung: In Bereich ist ein Speicherbereich ausgewählt, zum Beispiel Area 0. TwinCAT bietet alle aktuell verwendeten Speicherbereiche zur Auswahl an. Speicherbereiche, die ausschließlich für Code reserviert sind, werden nicht angezeigt.
	Adresse für eine Variable herausfinden: Eingabeunterstützung zur Auswahl einer IEC-Variablen erscheint. Wenn Sie eine Variable ausgewählt haben, belegt TwinCAT die Startadresse mit der Variablenadresse vor.
	Speicheransicht laden/aktualisieren
	Vorheriges Speichersegment anzeigen: Navigieren zum vorherigen Speichersegment
	Nächstes Speichersegment anzeigen: Navigieren zum nächsten Speichersegment
	<b>Hinweis</b> TwinCAT prüft nicht, ob die Änderungen zulässig sind. Sie können die Applikation durch unbedachte Änderungen zum Absturz bringen Änderungen auf SPS laden: TwinCAT überträgt die neuen Daten auf die Steuerung. Voraussetzung: Sie haben ein oder mehrere Bytes in der Speicheransicht überschrieben.
	Speicherinhalt in Datei speichern: Dialog Speicherinhalt als Binärdatei erscheint. Wählen Sie einen Ablageort.
Spalten	Anzahl der Spalten der hexadezimalen Darstellung des Speicherauszugs, welche in der Mitte des Fensters zu finden ist. Bei Auto passt sich die Spaltenanzahl an die Fenstergröße an.

### Startadresse des Speicherauszugs über Variable festlegen

- ✓ Sie haben eine Applikation auf die Steuerung geladen und sind darauf eingeloggt. Als Startadresse des Speicherauszugs möchten Sie die Adresse einer bestimmten Variablen angeben.

1. Wählen Sie den Befehl **Ansicht** → **Speicheransicht** anzeigen.

⇒ Die Ansicht Speicher öffnet sich.

2. Geben Sie die Adresse der Variablen in **Bereich** ein. Tippen Sie die Adresse entweder direkt ein, beispielsweise abgelesen aus dem Wert einer Pointervariablen, oder wählen Sie die Variable mit Hilfe

der Eingabehilfe  aus.

⇒ Die Option **Absolut** wird automatisch eingestellt.

⇒ TwinCAT zeigt den Speicherauszug als absoluten Bereich um die Adresse der Variablen herum an.

### Startadresse des Speicherauszugs über Angabe des Speicherbereichs (Area) festlegen

- ✓ Sie haben eine Applikation auf die Steuerung geladen und sind darauf eingeloggt. Als Startadresse des Speicherauszugs möchten Sie die Startadresse eines bestimmten Speicherbereichs (Area) plus einen Versatz (Offset) angeben.

1. Wählen Sie den Befehl **Ansicht** → **Speicheransicht** anzeigen.
  - ⇒ Die Ansicht Speicher öffnet sich.
2. Wählen Sie in **Bereich** beispielsweise die Option **Area 0**. Die Liste an auswählbaren Bereichen ist geräteabhängig.
3. Geben Sie in **Offset** den gewünschten Versatz für die Startadresse an. Beispiel: „512„.
  - Information:** Das Feld muss ausgefüllt werden. Wenn kein Versatz gewünscht ist, tragen Sie 0 ein.
  - ⇒ TwinCAT zeigt den Speicherauszug ab dem Speicherbereich Area 0 + 512 an.

### Daten bearbeiten

**i** Sie können in der Speicheransicht die angezeigten Bytes überschreiben und die Änderungen auf die Steuerung übertragen. TwinCAT prüft nicht, ob die Änderungen zulässig sind. Sie können die Applikation durch unbedachte Änderungen zum Absturz bringen!

- ✓ Sie haben eine Applikation auf die Steuerung geladen und sind eingeloggt.
1. Führen Sie die Schritte 1-3 aus der oben beschriebenen Anleitung „Startadresse über Angabe des Speicherbereichs (Area) festlegen“ durch.
  2. Klicken Sie in der Speicheransicht in der hexadezimalen Darstellung der Daten auf ein Byte und geben Sie einen neuen Wert ein.
    - ⇒ Neue Werte, die einen bisherigen Wert überschreiben, werden rot angezeigt.

3. Klicken Sie auf die Schaltfläche  .

⇒ TwinCAT überträgt die Daten an die Steuerung.

### Speicherprüfungen für die aktive Applikation durchführen

Siehe: [Befehl Speicher für aktive Applikation prüfen \[► 139\]](#)

#### 5.8.1.2 Befehl Speicher für aktive Applikation prüfen

**Funktion:** Der Befehl startet eine Prüfung des Speicherbereichs der aktiven Applikation. Der Speicher wird dabei diversen Prüfungen unterzogen. Die Ergebnisse werden im Meldungsfenster ausgegeben.

**Aufruf:** Menü **PLC > Fenster**

**Voraussetzung:** Mindestens eine Applikation ist geladen und im Onlinebetrieb.

Der Speicherbereich der aktiven Applikation wird auf Folgendes überprüft:

- BOOL-Variablen haben definierte Werte.
- STRING- und WSTRING-Variablen sind terminiert.
- Enumerationsvariablen haben gültige Werte.
- Variablen eines Unterbereichstyps haben gültige Werte.
- Variablen eines konstanten Blocktyps (STRUCT, ARRAY etc.) sind gegenüber ihren Initialisierungswerten unverändert.
- Funktionspointeradressen sind intern konsistent.
- Funktionspointer im Projekt zeigen in einen von TwinCAT 3 PLC verwalteten Speicher. Funktionspointer auf Bibliotheksbausteine oder auf externe Bausteine sind ausgenommen, diese können in externem Speicher liegen.
- Pointer auf einen Typ zeigen auf Speicherstellen dieses Typs.
- Pointer zeigen auf Speicherstellen, deren Adresse auf die Typgranularität ausgerichtet ist, oder 0 oder -1 ist.
- Schnittstellenvariablen und Funktionstabellenpointer sind intern konsistent.
- Der Code von kompilierten Bausteinen ist unverändert.  
Hinweis: Kompilierte Bausteine mit Breakpoints werden von der Prüfung ausgenommen, weil für diese der Code zur Laufzeit manipuliert wird.

Die Ergebnisse der Prüfung werden im Meldungsfenster ausgegeben. Verstöße gegen die Prüfungskriterien erscheinen als Warnungen.

Bei einem Doppelklick auf eine Warnung erscheint die Ansicht **Speicher**, in der die betreffende Stelle orange markiert ist. Die Ansicht **Speicher** kann auch mit dem [Befehl Speicheransicht anzeigen](#) [► 136] geöffnet werden.

Mit Hilfe des Attributs '[memory\\_check](#)' kann die Prüfung eingeschränkt werden.

### 5.8.1.3 Befehl Überwachungsliste <n>

Symbol: 

**Funktion:** Der Befehl öffnet die Ansicht **Überwachungsliste <n>**. Eine Überwachungsliste können Sie mit Variablen aus Ihrem Projekt füllen, um im Onlinebetrieb für diese Variablen innerhalb einer einzigen Ansicht die Werte monitoren, forcen oder schreiben zu können. n kann 1,2,3,4 sein, das heißt, dass Sie bis zu vier Überwachungslisten konfigurieren können.

**Aufruf:** Menü **PLC > Fenster**

**Siehe auch:**

- Dokumentation PLC: Überwachungslisten verwenden

### 5.8.1.4 Befehl Alle Forces anzeigen

Symbol: 

**Funktion:** Der Befehl öffnet die Ansicht **Alle Forces anzeigen**, die eine spezielle Form einer Überwachungsliste ist.

**Aufruf:** Menü **PLC > Fenster**

**Voraussetzung:** Ein SPS-Projekt ist im Offline- oder Onlinebetrieb geöffnet.

Die Ansicht enthält alle derzeit zum Forcen vorbereiteten und alle geforcten Variablen des SPS-Projekts in einer Liste. In der Liste sind die Aktionen möglich, die auch in anderen Überwachungslisten möglich sind.

#### Alles Forces anzeigen

Tabellarische Anzeige aller geforcten und für das Forcen vorbereiteten Variablen der Applikation

Ausdruck	Variablenname
Datentype	Datentyp der Variablen
Wert	Aktuell geforcter Wert der Variablen
Vorbereiteter Wert	Für das Forcen vorbereiteter Wert
Überschriebener Wert am Anfang des Zyklus	Bei Eingängen wird der eigentliche Wert bereits vor dem Ausführen des Anwendercodes durch den Forcewert überschrieben. Somit ist dies der eigentliche Wert. Bei Ausgängen ist dies der geforcte Wert.
Überschriebener Wert am Ende des Zyklus	Bei Ausgängen ist dies der Wert, der im Zyklus berechnet wird. Dieser Wert wird jedoch am Ende des Zyklus durch den Forcewert überschrieben. Bei Eingängen ist dies der geforcte Wert.

Zusätzlich gibt es im Auswahlmenü **Force aufheben** folgende Befehle:

- **Forcen aufheben und alle ausgewählten Werte beibehalten:** Für alle selektierten Einträge der Liste werden die Variablen auf den geforcten Wert gesetzt und das Forcen aufgehoben.
- **Forcen aufheben und alle ausgewählten Werte wiederherstellen:** Für alle selektierten Einträge der Liste werden die Variablen auf den Wert zurückgesetzt, den sie vor dem Forcen hatten, und das Forcen wird aufgehoben.

**Siehe auch:**

- Dokumentation PLC: Forcen und Schreiben von Variablen
- Dokumentation PLC: Überwachungslisten verwenden

**5.8.1.5 Befehl Querverweisliste**

Symbol: 

**Funktion:** Der Befehl öffnet die Ansicht **Querverweisliste**.

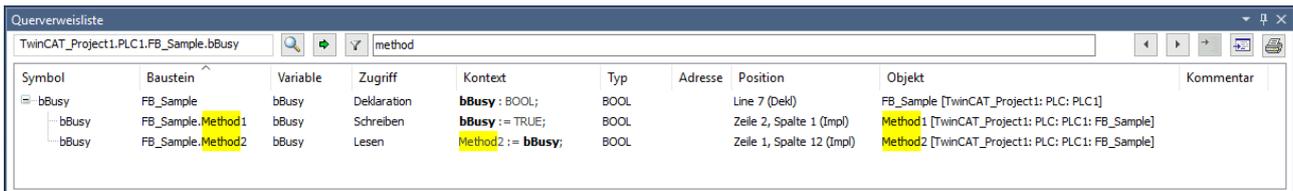
**Aufruf:** Menü **PLC > Fenster**

**Ansicht Querverweisliste**

Die Ansicht zeigt eine Liste der Querverweise im Projekt für ein Symbol. Das Symbol kann eine Variable, eine POU (Programm, Funktionsbaustein, Funktion) oder ein anwenderspezifischen Datentyp (DUT) sein.

Dabei bietet die Querverweisliste grundsätzlich zwei Sucharten:

- Textsuche: Durch die Eingabe eines Symbolnamens werden die Querverweise aller Symbole im Projekt mit diesem Namen angezeigt. Falls mehrere Symbole mit gleichem Namen gefunden werden, kann die Anzeige über das Kontextmenü auf einzelne Deklarationen eingeschränkt werden.
- Deklarationssuche: Das Symbol kann über die Eingabehilfe oder durch die Eingabe eines qualifizierten Pfads, beispielsweise MAIN.nVar ausgewählt werden. Danach werden nur die Verwendungsstellen dieses Symbols angezeigt, auch wenn noch andere Symbole mit dem gleichen Namen existieren.



**Werkzeuggeste**

Name (Eingabefeld)	<p>Symbolname (Variablenname, Bausteinname, DUT-Name) Eingabemöglichkeiten:</p> <ul style="list-style-type: none"> <li>• Auswahl eines deklarierten Symbols über die Eingabehilfe (Schaltfläche )</li> <li>• Händisches Eingeben des Symbolnamens.</li> </ul> <p>Auslösen der Suche über die Schaltfläche  oder die [Eingabetaste]. Sie können für die Textsuche die Platzhalter „*“ (beliebig viele Zeichen) oder „?“ (genau ein beliebiges Zeichen) in Kombination mit einer Teilzeichenkette eines Variablenbezeichners verwenden. Verwenden Sie „%“, wenn Sie nach IEC-Adressen suchen wollen. Beispiele: „%MW8“, „%M*“</p> <p>Zusätzliche Möglichkeiten von außerhalb der Ansicht <b>Querverweisliste</b>:</p> <ul style="list-style-type: none"> <li>• Verwenden des Kontextmenübefehls <b>Alle Verweise suchen</b>, wenn der Name eines deklarierten Symbols in einem Editor selektiert ist oder der Cursor im Namen steht.</li> <li>• Automatisch, wenn der Name eines deklarierten Symbols in einem Editor selektiert ist oder der Cursor im Namen steht. Eine automatische Suche ist auch möglich, wenn das Objekt im Projektbaum selektiert ist. Voraussetzung: Die Ansicht <b>Querverweisliste</b> ist geöffnet und die TwinCAT-Option <b>Querverweise automatisch bei Selektionsänderung aktualisieren</b>, Kategorie <b>Intelligentes Kodieren</b>, ist aktiviert.</li> </ul> <p>Folgende Eingaben sind gültig:</p> <ul style="list-style-type: none"> <li>• Variablenname, einfach oder qualifiziert: z. B. „nVar“, „MAIN.nVar“</li> <li>• Bausteinname: z. B. „MAIN“, „FB_MyFB“</li> <li>• DUT-Name: z. B. „ST_MySTRUCT“</li> <li>• Zeichenfolgen in Kombination mit Platzhalter „*“ (beliebige Zeichen) oder „?“ (genau ein beliebiges Zeichen): Beispiel: „nVar*“ betrifft nVar1, nVarGlob2, nVar45 usw... „nVar?“ betrifft nVar1, nVar2, nVarX usw., nicht aber nVarGlob2, nVar45 usw.</li> <li>• „%&lt;IEC-Adresse&gt;“: TwinCAT sucht nach Variablen, die dieser Adresse zugewiesen sind, und direkte Speicherzugriffe. Beispiel: „%QB0“, „%Q0 := 2“</li> </ul> <p>Groß/Kleinschreibung sowie Leerzeichen an Anfang und Ende der Eingabezeichenfolge werden nicht berücksichtigt.</p>
	Eingabehilfe öffnen zur Auswahl eines Symbols.
	Querverweise finden: Die Suche wird durchgeführt.
	Spalten definieren, in denen nach der Zeichenfolge gesucht wird
Filtern (Eingabefeld)	<p>Zeichenfolge, nach der in den angewählten Spalten gesucht wird Die Fundstellen werden gelb markiert. Querverweise ohne diese Zeichenfolge werden ausgeblendet.</p>
	Quellposition des vorherigen Querverweises anzeigen
	Quellposition des nächsten Querverweises anzeigen
	<p>Ergebnisse auf aktuelle Deklaration beschränken Verfügbar, wenn für ein Symbol mehrere Deklarationen gefunden wurden. Begrenzt die Anzeige auf die Deklaration, die Sie gerade in der Liste selektiert haben.</p>
	Quellposition des selektierten Querverweises anzeigen: Der Fokus springt zur Verwendungsstelle des Symbols.
	Querverweisliste drucken: Der Standarddialog zum Einrichten eines Druckauftrags erscheint.

**Tabelle der gefundenen Querverweise**

Symbol	Die Fundstellen für die Symbole (Variablen, POU, DUTs) werden nach ihrer Deklaration gruppiert. Die Deklarationsstelle bildet den Wurzelknoten, darunter eingerückt erscheinen die Verwendungsstellen im Projekt. Dabei wird genau der Ausdruck angezeigt, den das Symbol an der Verwendungsstelle hat.  Beispiel: Gibt es im Projekt eine globale Variable „nVar“ und in einem Baustein eine lokal deklarierte Variable „nVar“, dann erscheinen nach einer Textsuche nach den Querverweisen zwei Wurzelknoten-Einträge in der Liste und darunter jeweils die Verwendungsstellen der Variable „nVar“.
Baustein	Bausteinname, DUT-Name; auch beispielsweise Taskname im Falle eines Bausteinaufrufs in der Taskkonfiguration.
Variable	Reiner Variablenname. Beispiel: „nVar“.
Zugriff	Art des Zugriffs auf die Variable an der Verwendungsstelle: Deklaration / Lesen / Schreiben / Aufruf.  Sonderfall für Pointer: Eine Zuweisung der Art <code>pSample := ADR(nVar1)</code> wird bei Suche nach „nVar1“ als Schreiben   Adresse angezeigt. Grund: Eventuelle Schreibzugriffe auf „pSample“ werden bei Suche nach „nVar1“ nicht angezeigt. Über die Pointervariable sind auch Schreibzugriffe möglich.
Kontext	Kontext der Verwendung der Variablen. Beispiel: „nVar := 1“
Typ	Datentyp der Variablen.
Adresse	IEC-Adresse, falls der Variablen zugewiesen. Beispiel: „AT%QB0“
Position	Position der Verwendungsstelle innerhalb des Editors der betroffenen POU: beispielsweise Zeilennummer, Netzwerknummer, Deklarationsteil oder Implementierungsteil. Beispiel: „Zeile 1, Spalte 1 (Impl)“ oder „Zeile 9 (Dekl)“.
Objekt	POU-Name + in eckigen Klammern der komplette Pfad der Verwendungsstelle. Beispiel: „MAIN [TwinCAT_SampleProject: SPS: SamplePLCProject]“
Kommentar	Kommentar, falls in der Deklaration der Variable vorhanden.

Die Suche liefert alle Fundstellen im Projekt sowie in eingehängten, nicht kompilierten Bibliotheken.

**Befehle im Kontextmenü der Querverweisliste**

Quellcodeposition anzeigen	Öffnet den betroffenen Baustein und markiert die Verwendungsstelle: Für Root-Einträge die Deklaration, für darunter liegende Kind-Einträge die jeweilige Verwendungsstelle. Alternativ können Sie auf eine Zeile doppelklicken.
Ergebnisse auf aktuelle Deklaration beschränken	Grenzt im Fall von mehreren gefundenen Deklarationen die Anzeige der Ergebnisse auf die ausgewählte Symboldeklaration ein.
Alles erweitern	In der Liste erscheinen alle einzelnen Fundstellen.
Alles reduzieren	In der Liste erscheinen nur die Wurzelknoten aller Fundstellen.

**Siehe auch:**

- Befehl Ergebnisse auf aktuelle Deklaration beschränken
- Befehl Alles einklappen
- Befehl Alles ausklappen
- Dokumentation PLC: Verwendungsstellen mit der Querverweisliste finden

**5.8.1.6 Befehl Haltepunkte**

Symbol: 

**Funktion:** Der Befehl öffnet die Ansicht **Haltepunkte**.

**Aufruf:** Menü **PLC > Fenster****Ansicht Haltepunkte**

Die Ansicht zeigt Ihnen eine Übersicht aller definierten Haltepunkte einer Applikation. Innerhalb der Ansicht stehen Ihnen alle Befehle für Haltepunkte zur Verfügung.

POU	Position	Instanzpfad	Tasks	Bedingung	Trefferzahlbedingung	Aktuelle Trefferanzahl	Überwachte Werte zuletzt aktualisiert
MAIN	Zeile 1, Spalte 1 (Impl)	TwinCAT_Device.Project1.MAIN	(alle)	Immer anhalten	Immer anhalten	0	

**Tabelle der aktuellen Haltepunkte**

Applikation	Wählen Sie das gewünschte SPS-Projekt aus der Liste.
POU	Name des Bausteins, der den Haltepunkt enthält.
Position	Haltepunkt-Position innerhalb der POU <ul style="list-style-type: none"> <li>• Texteditor: Zeilen- plus Spaltennummer</li> <li>• Grafischer Editor: Netzwerk oder Elementnummer</li> </ul> „(Impl)“ im Fall von Funktionsbausteinen zeigt an, dass der Haltepunkt in der Implementierung des Funktionsbausteins sitzt, nicht in einer Instanz.
Instanzpfad	Vollständiger Objekt Pfad der Haltepunkt-Position.
Tasks	Namen der Tasks, bei deren Ausführung der Haltepunkt wirksam sein soll. Wenn keine Einschränkung gilt, steht hier „(alle)“.
Bedingung	<ul style="list-style-type: none"> <li>• Immer anhalten: Keine zusätzliche Aktivierungsbedingung definiert. Der Haltepunkt ist immer aktiv.</li> <li>• Boolescher Ausdruck. Der Ausdruck muss TRUE liefern, damit der Haltepunkt aktiv ist.</li> </ul>
Trefferzahlbedingung	Angabe, wann (in welcher Abhängigkeit von der Trefferanzahl) der Haltepunkt wirksam werden soll.
Aktuelle Trefferanzahl	Angabe, wie oft der Haltepunkt während der Ausführung bis jetzt bereits durchlaufen („getroffen“) wurde.

**Werkzeugleiste**

	Neuer Haltepunkt (entspricht dem <u>Befehl Neuer Haltepunkt</u> [ <a href="#">▶ 127</a> ] im Menü <b>Debuggen</b> )	Öffnet den Dialog <b>Eigenschaften Haltepunkt</b>
	Haltepunkt löschen	Entfernt den Haltepunkt. Verwechseln Sie den Befehl nicht mit dem Befehl Deaktivieren.
	Haltepunkt aktivieren/deaktivieren (entspricht dem <u>Befehl Haltepunkt aktivieren</u> [ <a href="#">▶ 131</a> ] und dem <u>Befehl Haltepunkt deaktivieren</u> [ <a href="#">▶ 131</a> ] im Menü <b>Debuggen</b> )	Schaltet den Haltepunkt oder Ausführungspunkt zwischen Status „aktiviert“ und „deaktiviert“ hin und her.  <ul style="list-style-type: none"> <li>• ● Haltepunkt aktiviert</li> <li>• ○ Haltepunkt deaktiviert</li> <li>• ● Ausführungspunkt aktiviert</li> <li>• ○ Ausführungspunkt deaktiviert</li> </ul> Im Gegensatz zum <b>Haltepunkt löschen</b> bleibt ein deaktivierter Haltepunkt in der Liste erhalten und kann wieder aktiviert werden.
	Eigenschaften	Öffnet den Dialog <b>Eigenschaften Haltepunkt</b> erscheint zur Bearbeitung der Haltepunkt-Parameter. Im Onlinebetrieb können Sie hier den Haltepunkt zum Ausführungspunkt umwandeln.
	Gehe zur Quellcodeposition	Öffnet die Online-Ansicht des betreffenden Bausteins. Der Cursor steht an der Haltepunkt-Position.
	Alle Haltepunkte löschen	Löscht alle Haltepunkte und Ausführungspunkte der Applikation. Die Liste wird geleert. Nicht zu verwechseln mit Deaktivieren!
	Alle Haltepunkte aktivieren	Aktiviert alle gerade deaktivierten Haltepunkte und Ausführungspunkte.
	Alle Haltepunkte deaktivieren	Deaktiviert alle gerade aktivierten Haltepunkte und Ausführungspunkte. Die Punkte bleiben in der Liste und können wieder aktiviert werden.

**Siehe auch:**

- Befehl Neuer Haltepunkt > [Dialog Eigenschaften Haltepunkt](#) [[▶ 128](#)]
- [Befehl Haltepunkt umschalten](#) [[▶ 131](#)]
- Dokumentation PLC: Haltepunkte verwenden

**5.8.1.7 Befehl Aufrufliste**

Symbol:

**Funktion:** Der Befehl öffnet die Ansicht **Aufrufliste**.

**Aufruf:** Menü **PLC > Fenster**

**Ansicht Aufrufliste**

Diese Ansicht ist nützlich, wenn Sie Programme schrittweise ausführen wollen. Sie zeigt die aktuell erreichte Position mit vollständigem Aufrufpfad.

POU	Position	Instanzpfad
FB_Blinker [TwinCAT_Device: PLC: Project2] Digital [TwinCAT_Device: PLC: Project2] MAIN [TwinCAT_Device: PLC: Project2]	Network 1 / Operand 'fbTimer1' (Impl) Network 1 / Operand 'fbBlinker1' (Impl) RETURN	Digital.fbBlinker1

Applikation	Name des aktiven SPS-Projekts, die den gerade erreichten Programmbaustein kontrolliert.
Task	Name der Task, die den gerade erreichten Programmbaustein kontrolliert.
POU	Name des Programmbausteins, in dem die Programmausführung steht. Die erste Zeile in der Liste beschreibt die aktuelle Ausführungsposition. Sie ist mit einem gelben Pfeil markiert. Wenn diese Position in einem Baustein liegt, der von einem anderen aufgerufen wird, wird die Position des Aufrufs in der zweiten Zeile beschrieben. Wenn der Aufrufer wiederum von einem anderen Baustein aufgerufen wird, wird diese Aufrufposition in der dritten Zeile beschrieben usw.
Position	Position innerhalb des Programmbausteins, an der die Programmausführung steht <ul style="list-style-type: none"> <li>• Zeilen- und Spaltennummer bei Texteditoren</li> <li>• Netzwerk- oder Elementnummer bei grafischen Editoren</li> </ul>
Instanzpfad	Instanz, in der die Programmausführung steht.

Die Aufrufliste ist auch im Offlinebetrieb verfügbar und im normalen Onlinebetrieb, wenn Sie gerade keine Debugging-Funktionen benutzen. In diesem Fall enthält sie die zuletzt während einer schrittweisen Ausführung angezeigte Position, allerdings in „gegrauter“ Schrift.



Die Ansicht **Aufrufbaum** liefert im Unterschied zur **Aufrufliste** jederzeit Aufrufinformationen zu einem Baustein.

**Siehe auch:**

- Dokumentation PLC: Verwenden von Haltepunkten

**5.8.1.8 Befehl Aufrufbaum**

Symbol:

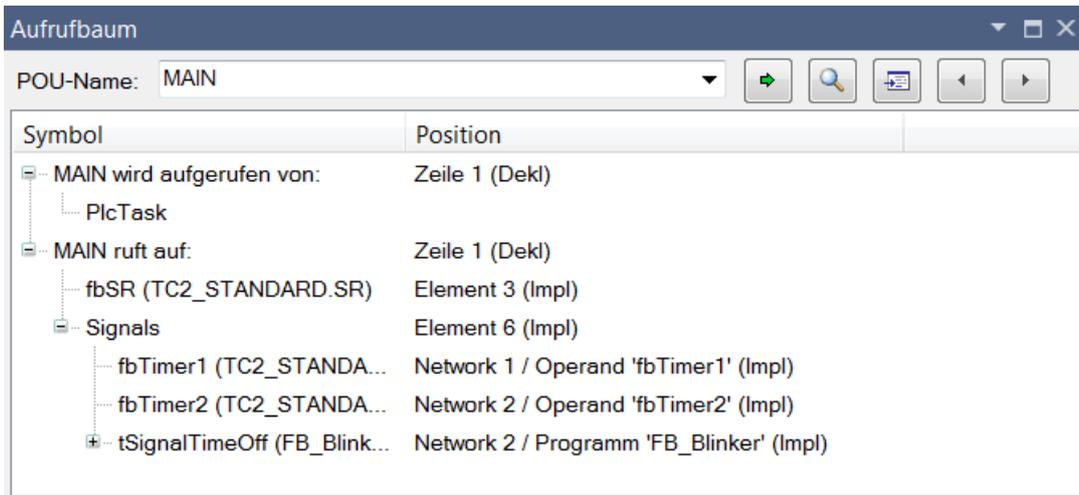
**Funktion:** Der Befehl öffnet die Ansicht **Aufrufbaum**.

**Aufruf:** Menü **PLC > Fenster**

**Ansicht Aufrufbaum**

Der Aufrufbaum steht jederzeit bereits vor dem Kompilieren (Übersetzen) der Applikation zur Verfügung. Er ist eine statische Darstellung der Aufrufer und der Aufrufe des Bausteins, den Sie explizit angeben. Somit enthält der Baum immer zwei Wurzelknoten, unter denen die jeweilige Aufrufabfolge als nacheinander eingerückte Einträge zu sehen ist. Rekursive Aufrufe sind in dieser Baumdarstellung schnell erkennbar.

Beispiel für einen Aufrufbaum (1) für Baustein (2) MAIN:



POU-Name	Name des Programmbausteins, kann manuell, oder durch Ziehen aus einer anderen Ansicht, oder mithilfe der Schaltfläche  eingegeben werden. Die Auswahlliste enthält die zuletzt eingegebenen Bausteinnamen.
----------	---

**Symbolleiste und Tastaturbedienung**

 Baustein finden	TwinCAT sucht nach dem in „Bausteinname“ angegebenen Baustein und stellt seine Aufrufer und seine Aufrufe dar.
 Baustein aus Eingabehilfe entnehmen	Der Dialog <b>Eingabehilfe</b> erscheint zur Auswahl eines Bausteinaufrufs oder Instanzaufrufs. Der Aufrufbaum wird nach der Auswahl automatisch aktualisiert.
 Quellcodeposition des selektierten Bausteins anzeigen	TwinCAT springt zur Verwendungsstelle des Bausteins im Quellcode Ihres Programms.
 Quellcodeposition des nächsten Bausteins anzeigen	Die Selektion im Aufrufbaum springt zum nächsten oder vorherigen Baustein in der Aufrufstruktur. Gleichzeitig wird die dazugehörige Quellcodeposition im jeweiligen Editor geöffnet.  Ein Doppelklick auf einen Eintrag im Aufrufbaum öffnet ebenfalls die zugehörige Quellcodeposition.
 Quellcodeposition des vorherigen Bausteins anzeigen	

**Darstellung des Aufrufbaums**

Position	Für die Wurzelknoten im Aufrufbaum: Zeilennummer der Deklaration ( „Dekl“ ) des Bausteins.  Für die Aufrufer oder Aufrufe unter den Wurzelknoten: Je nach Implementierungssprache Zeilennummer, Spaltennummer, Netzwerknummer ihrer Position.
----------	---

**Kontextmenü für den gerade im Baum selektierten Eintrag**

Alles reduzieren	Die ausgeklappten Einträge im Aufrufbaum werden bis auf die beiden Wurzelknoten zugeklappt.
Quellcodeposition anzeigen	TwinCAT springt zur Verwendungsstelle des Bausteins im Quellcode Ihres Programms.
Als neuen Wurzelknoten setzen	Der im Aufrufbaum selektierte Eintrag erscheint in „Bausteinname“. Der Baum wird automatisch für die neuen Wurzelknoten angepasst.



Im Unterschied zum statischen Aufrufbaum, der jederzeit Aufrufinformationen zu einem Baustein liefert, ist die Ansicht **Aufrufliste** für unmittelbare Information während der schrittweisen Abarbeitung eines Programms vorgesehen. Die Aufrufliste zeigt immer den kompletten Aufrufpfad der gerade erreichten Position.

### 5.8.1.9 Befehl Online Change Memory Reserve Settings

**Funktion:** Der Befehl öffnet die Ansicht **Online Change Memory Reserve**.

**Aufruf:** Menü **PLC > Fenster**.

In der Ansicht werden für Funktionsbausteine Speicherreserven für den Online-Change konfiguriert.

Applikation durchsuchen	<ul style="list-style-type: none"> <li>Durchsucht das ausgewählte SPS-Projekt nach Funktionsbausteinen und zeigt sie im Bereich <b>Funktionsbausteine</b> an</li> <li>Aktualisiert den Bereich <b>Funktionsbausteine</b>, nachdem das SPS-Projekt erneut übersetzt wurde</li> <li>Aktualisiert den Bereich <b>Funktionsbausteine</b> nach einem Online-Change</li> </ul>
Auswahlliste mit den SPS-Projekten des geöffneten TwinCAT-Projekts	Auswahl des SPS-Projekts, dessen Funktionsbausteine in dieser Ansicht angezeigt und/oder bearbeitet werden sollen

**Funktionsbausteine:**

Alle	Alle Funktionsbausteine des ausgewählten SPS-Projekts werden angezeigt.
Keine Speicherreserve	Alle Funktionsbausteine mit Speicherreserve 0 Bytes werden angezeigt.
<Speicherreserve> Bytes	Anzeige aller Funktionsbausteine mit der Anzahl Bytes, die in <b>Speicherreserve</b> definiert ist.
Informationen zu den Funktionsbausteinen Bei der Selektion eines Bausteins für die Konfiguration der Speicherreserve ist auch eine Mehrfachauswahl möglich.	
Funktionsbaustein	Name des Funktionsbausteins
Größe	Größe des Funktionsbausteins Größe einer Instanz des Funktionsbausteins Angabe in Bytes
Instanzzahl	Anzahl der Instanzen des Funktionsbausteins im Projekt
Speicherreserve	Anzeige der Speicherreserve pro Instanz des Funktionsbausteins
Zusätzlicher Speicher für alle Instanzen	Produkt aus <b>Instanzzahl</b> und <b>Speicherreserve</b>
Verbleibende Speicherreserve	Anzahl Bytes, die noch pro Funktionsbaustein-Instanz als Reserve zur Verfügung stehen

**Einstellungen:**

Speicherreserve (in Bytes)	Eingabefeld für die Speicherreserve für den selektierten Funktionsbaustein. Angabe in Bytes Voraussetzung: Das SPS-Projekt befindet sich noch nicht auf der Steuerung oder Sie haben durch eine Klick auf die Schaltfläche <b>Erlauben</b> im Bereich <b>Bearbeitung erlauben</b> die Änderung der Speicherreserve erlaubt.
Für Auswahl anwenden	Die <b>Speicherreserve (in Bytes)</b> wird dem Funktionsbaustein zugeordnet und die Tabellenspalte <b>Speicherreserve</b> wird aktualisiert. Bei Mehrfachauswahl wird der eingegebene Wert jedem Funktionsbaustein zugeordnet. Um die Spalten <b>Größe</b> , <b>Instanzzahl</b> , <b>Zusätzlicher Speicher für alle Instanzen</b> und <b>Verbleibende Größe der Speicherreserve</b> zu aktualisieren, wählen Sie zunächst den Befehl <b>Erstellen &gt; Erstellen</b> und klicken Sie anschließend auf die Schaltfläche <b>Applikation durchsuchen</b> .

**Bearbeitung erlauben:**

Erlauben	Das Eingabefeld <b>Speicherreserve (in Bytes)</b> wird editierbar. Diese Schaltfläche wird in <b>Bearbeitbar</b> geändert.
----------	---

**Information:**

Anzahl FBs	Gesamtanzahl der Funktionsbausteine in dem SPS-Projekt
Zusatzspeicher für alle Instanzen	Summe der Speicherreserven aller Funktionsbaustein-Instanzen des SPS-Projekts Angabe in Bytes

**Siehe auch:**

- Dokumentation PLC: SPS-Projekt programmieren > Speicherreserve für Online-Change konfigurieren

**5.8.1.10 Befehl SPS Lesezeichen**

Symbol: 

**Funktion:** Der Befehl öffnet die Ansicht **Lesezeichen**.

**Aufruf:** Menü **PLC > Fenster**

 Vorheriges Lesezeichen	Springt zum Lesezeichen, das in der Tabelle eine Zeile über der selektierten Zeile angezeigt wird, und öffnet die entsprechende POU im Editor.
 Nächstes Lesezeichen	Springt zum Lesezeichen, das in der Tabelle eine Zeile unter der selektierten Zeile angezeigt wird, und öffnet die entsprechende POU im Editor.
	Löscht das selektierte Lesezeichen aus der Tabelle und in der entsprechenden POU.

Auflistung der Lesezeichen des Projekts mit den Informationen Lesezeichen, Objekt und Position:

Lesezeichen	Von TwinCAT vergebene Bezeichnung der Lesezeichen in nummerierter, aufsteigender Reihenfolge: Lesezeichen_0, Lesezeichen_2 usw. Wenn das Lesezeichen selektiert ist und Sie in das Feld klicken, wird es editierbar und Sie können die Bezeichnung des Lesezeichens ändern.
Objekt	Name und Projektpfad der POU, in der das Lesezeichen gesetzt ist
Position	Position des Lesezeichens innerhalb der POU Beispiel: Zeile3, Spalte 1 (Impl) (Impl): im Implementierungsteil der POU (Decl): im Deklarationsteil der POU

Sie können Reihenfolge der Lesezeichen per Drag-and-drop verändern.

Wenn Sie eine Zeile doppelklicken, öffnet TwinCAT das entsprechende Objekt im Editor und springt zu diesem Lesezeichen.

#### Siehe auch

- [Befehl Vorheriges Lesezeichen \[► 153\]](#)
- [Befehl Nächstes Lesezeichen \[► 152\]](#)
- Lesezeichen setzen und verwenden (Dokumentation „PLC“)

## 5.8.2 Core Dump

### 5.8.2.1 Befehl Core Dump erzeugen

**Funktion:** Der Befehl bewirkt, dass TwinCAT zunächst prüft, ob bereits eine Core-Dump-Datei auf dem Zielsystem verfügbar ist.

- Wenn auf dem Zielsystem eine Core-Dump-Datei verfügbar ist, bietet TwinCAT Ihnen an, diese Datei ins Projektverzeichnis zu laden. Die Abfrage, ob die Core-Dump-Datei vom Zielsystem geladen werden soll, können Sie mit drei unterschiedlichen Möglichkeiten beantworten.
  - Ja: Falls die Core-Dump-Datei des Zielsystems zu dem aktuell eingeloggtten SPS-Projekt passt, lädt TwinCAT die Core-Dump-Datei vom Zielsystem ins Projektverzeichnis. Diese Datei können Sie öffnen, indem Sie das SPS-Projekt anschließend ausloggen und den [Befehl Core Dump laden \[► 151\]](#) verwenden.
  - Nein: Eine neue Core-Dump-Datei wird im Projektverzeichnis erzeugt. Die Voraussetzung hierfür ist, dass das SPS-Projekt gerade an einem Haltepunkt steht oder ein Ausnahmefehler aufgetreten ist.
  - Abbrechen: Die Erzeugung einer Core-Dump-Datei wird abgebrochen.
- Wenn auf dem Zielsystem keine Core-Dump-Datei verfügbar ist, veranlasst TwinCAT das Erzeugen einer neuen Datei mit den aktuellen SPS-Projektdateien im Projektverzeichnis. Die Voraussetzung hierfür ist, dass das SPS-Projekt gerade an einem Haltepunkt steht oder ein Ausnahmefehler aufgetreten ist.

Die erzeugte Core-Dump-Datei wird direkt im SPS-Projektverzeichnis abgelegt: <SPS-Projektnamen>.<SPS-Projekt-GUID>.core

**Aufruf:** Menü PLC > Core Dump

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb.

## ● Automatische Erzeugung eines Core Dumps auf dem Zielsystem

**i** Wenn das SPS-Projekt, das auf einem Zielsystem läuft, gerade nicht in einer Entwicklungsumgebung eingeloggt ist, erzeugt das Laufzeitsystem im Falle eines Ausnahmefehlers automatisch einen Core Dump auf dem Zielsystem. Diese Datei befindet sich standardmäßig im Boot-Ordner des Zielsystems (standardmäßig unter < TC3.1.4026.0: C:\TwinCAT\3.1\Boot\Plc; >=TC3.1.4026.0: C:\ProgramData\Beckhoff\TwinCAT\3.1\Boot\Plc). Wenn gewünscht, kann der Speicherpfad der automatischen Core-Dump-Erstellung auch angepasst werden, weitere Informationen finden Sie unter Fehleranalyse mit Core Dump.

Das automatische Laden dieser Core-Dump-Datei vom Zielsystem in das lokale Projektverzeichnis ist mit Hilfe des **Befehls Core Dump erzeugen** möglich. Das manuelle Kopieren der Core-Dump-Datei vom Zielsystem auf den Entwicklungsrechner ist ebenfalls möglich.

Das Anzeigen des Dumps mit Hilfe des **Befehls Core Dump laden** [► 151] kann für die (nachträgliche) Fehleranalyse verwendet werden.

## ● Core Dump nur nutzbar mit zugehöriger Compile-Info-Datei

**i** Wenn Sie eine Core-Dump-Datei archivieren oder abspeichern, beachten Sie, dass zum Laden eines Core Dumps das zugehörige Projekt und die zugehörige Compile-Info-Datei (\*.compileinfo-Datei, die z.B. beim Erstellen des Projekts im „\_CompileInfo“-Ordner abgelegt wird) vorliegen müssen. Falls dies nicht der Fall ist, kann TwinCAT den Dump später nicht mehr verwenden.

Beachten Sie hierzu auch die Einstellungsmöglichkeiten auf der **Registerkarte Settings** [► 122]. Mit Hilfe der Einstellung **Core Dump** können Sie konfigurieren, ob die Core-Dump-Datei, die sich möglicherweise im Projektverzeichnis befindet, zusammen mit den verfügbaren Compile-Info-Dateien in einem TwinCAT-Dateiarchiv gespeichert werden soll.

### Siehe auch:

- Dokumentation PLC: SPS-Projekt zur Laufzeit > Fehleranalyse mit Core Dump
- **Befehl Core Dump laden** [► 151]

## 5.8.2.2 Befehl Core Dump laden

**Funktion:** TwinCAT durchsucht das Projektverzeichnis nach Core Dump-Dateien.

- Wenn TwinCAT im Projektverzeichnis eine Core Dump-Datei findet, werden Sie gefragt, ob Sie diesen Core Dump laden oder nach einer Dump-Datei browsen möchten.
- Wenn TwinCAT im Projektverzeichnis keine Core Dump-Datei findet, können Sie nach einer anderen Dump-Datei browsen.

Das Laden ins Projekt bewirkt, dass eine Online-Ansicht des SPS-Projekts erscheint, mit dem Stand, den das SPS-Projekt zum Zeitpunkt des Erzeugens des Core Dumps hatte. Darin können Sie die Variablenwerte nachträglich ansehen. Zudem ist der Aufrufbaum verfügbar.

**Aufruf:** Menü PLC > Core Dump

**Voraussetzung:** Die Applikation ist im Offlinebetrieb.

**i** Sie können die Core Dump-Ansicht nur über den **Befehl Core Dump schließen** [► 152] wieder schließen. Der Befehl Ausloggen ist in dieser Ansicht nicht wirksam!

## ● Core Dump nur nutzbar mit zugehöriger Compile-Info-Datei

**i** Wenn Sie eine Core-Dump-Datei archivieren oder abspeichern, beachten Sie, dass zum Laden eines Core Dumps das zugehörige Projekt und die zugehörige Compile-Info-Datei (\*.compileinfo-Datei, die z.B. beim Erstellen des Projekts im „\_CompileInfo“-Ordner abgelegt wird) vorliegen müssen. Falls dies nicht der Fall ist, kann TwinCAT den Dump später nicht mehr verwenden.

Beachten Sie hierzu auch die Einstellungsmöglichkeiten auf der **Registerkarte Settings** [► 122]. Mit Hilfe der Einstellung **Core Dump** können Sie konfigurieren, ob die Core-Dump-Datei, die sich möglicherweise im Projektverzeichnis befindet, zusammen mit den verfügbaren Compile-Info-Dateien in einem TwinCAT-Dateiarchiv gespeichert werden soll.

**Siehe auch:**

- Dokumentation PLC: SPS-Projekt zur Laufzeit > Fehleranalyse mit Core Dump
- [Befehl Core Dump erzeugen \[►\\_150\]](#)
- [Befehl Core Dump schließen \[►\\_152\]](#)

### 5.8.2.3 Befehl Core Dump schließen

**Funktion:** Der Befehl schließt die gerade in der Entwicklungsumgebung geöffnete Core Dump-Ansicht des SPS-Projekts.

**Aufruf:** Menü PLC > Core Dump

**Voraussetzung:** Das SPS-Projekt ist im Offlinebetrieb und Sie haben eine Core Dump-Datei ins Projekt geladen.

**Siehe auch:**

- Dokumentation PLC: SPS-Projekt zur Laufzeit > Fehleranalyse mit Core Dump

## 5.8.3 SPS Lesezeichen

### 5.8.3.1 Befehl Lesezeichen ein-/ausschalten

Symbol: 

**Funktion:** Der Befehl setzt oder löscht ein Lesezeichen an der aktuellen Position.

**Aufruf:** Menü **SPS > SPS Lesezeichen**

**Voraussetzung:** Eine POU ist im Editor geöffnet und der Cursor steht in einer Programmzeile.

**Siehe auch**

- Lesezeichen setzen (Dokumentation „PLC“)

### 5.8.3.2 Befehl Nächstes Lesezeichen

Symbol: 

**Funktion:** Der Befehl springt in der Ansicht **Lesezeichen** und im Projekt zum nächsten Lesezeichen und öffnet die entsprechende POU. Die Reihenfolge, in der die Lesezeichen angesprungen werden, entspricht dabei der Reihenfolge der Lesezeichen in der Tabelle der Ansicht **Lesezeichen**.

**Aufruf:**

- Menü **SPS > SPS Lesezeichen**
- Schaltfläche  **Nächstes Lesezeichen** in der Ansicht **Lesezeichen**

**Voraussetzung:**

- Ein Projekt ist geöffnet
- Die Ansicht **Lesezeichen** ist geöffnet

**Siehe auch:**

- [Befehl SPS Lesezeichen \[►\\_149\]](#)
- [Befehl Nächstes Lesezeichen \(aktiver Editor\) \[►\\_153\]](#)
- Lesezeichen setzen (Dokumentation „PLC“)

### 5.8.3.3 Befehl Vorheriges Lesezeichen

Symbol: 

**Funktion:** Der Befehl springt in der Ansicht **Lesezeichen** und im Projekt zum vorherigen Lesezeichen und öffnet die entsprechende POU. Die Reihenfolge, in der die Lesezeichen angesprungen werden, entspricht dabei der Reihenfolge der Lesezeichen in der Tabelle der Ansicht **Lesezeichen**.

**Aufruf:**

- Menü **SPS > SPS Lesezeichen**
- Schaltfläche  Vorheriges Lesezeichen in der Ansicht Lesezeichen

**Voraussetzung:**

- Ein Projekt ist geöffnet
- Die Ansicht **Lesezeichen** ist geöffnet

**Siehe auch:**

- [Befehl SPS Lesezeichen \[► 149\]](#)
- [Befehl Vorheriges Lesezeichen \(aktiver Editor\) \[► 153\]](#)
- Lesezeichen setzen (Dokumentation „PLC“)

### 5.8.3.4 Befehl Alle Lesezeichen löschen

Symbol: 

**Funktion:** Der Befehl löscht alle Lesezeichen des geöffneten Projekts.

**Aufruf:** Menü **SPS > SPS Lesezeichen**

**Voraussetzung:** Eine POU ist im Editor geöffnet und der Cursor steht in der POU.

**Siehe auch:**

- [Befehl Alle Lesezeichen löschen \(aktiver Editor\) \[► 154\]](#)
- Lesezeichen setzen (Dokumentation „PLC“)

### 5.8.3.5 Befehl Nächstes Lesezeichen (aktiver Editor)

Symbol: 

**Funktion:** Der Befehl springt zum nächsten Lesezeichen im aktiven Editor.

**Aufruf:** Menü **SPS > SPS Lesezeichen**

**Voraussetzung:** Eine POU ist im Editor geöffnet und der Cursor steht in der POU

**Siehe auch:**

- [Befehl Nächstes Lesezeichen \[► 152\]](#)
- Lesezeichen setzen (Dokumentation „PLC“)

### 5.8.3.6 Befehl Vorheriges Lesezeichen (aktiver Editor)

Symbol: 

**Funktion:** Der Befehl springt zum vorherigen Lesezeichen im aktiven Editor.

**Aufruf:** Menü **SPS > SPS Lesezeichen**

**Voraussetzung:** Eine POU ist im Editor geöffnet und der Cursor steht in der POU

**Siehe auch:**

- [Befehl Vorheriges Lesezeichen \[► 153\]](#)
- Lesezeichen setzen (Dokumentation „PLC“)

### 5.8.3.7 Befehl Alle Lesezeichen löschen (aktiver Editor)

Symbol: 

**Funktion:** Der Befehl löscht alle Lesezeichen im aktiven Editor

**Aufruf:** Menü **SPS > SPS Lesezeichen**

**Voraussetzung:** Eine POU ist im Editor geöffnet und der Cursor steht in der POU.

**Siehe auch:**

- [Befehl Alle Lesezeichen löschen \[► 153\]](#)
- Lesezeichen setzen (Dokumentation „PLC“)

## 5.8.4 Befehl Laden

**Funktion:** Der Befehl bewirkt ein Übersetzen des aktiven SPS-Projekts mit anschließendem Download auf die Steuerung.

**Aufruf:** Menü **PLC**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb.

Mit dem Befehl führt TwinCAT eine Syntaxprüfung durch und erzeugt den Programmcode. Dieser Code wird auf die Steuerung geladen. Weiterhin erzeugt TwinCAT im Projektverzeichnis das Übersetzungsprotokoll <projectname>.<devicename>.<application ID>.compileinfo.



Beim Download werden alle Variablen mit Ausnahme von persistenten Variablen neu initialisiert.

---

Die Beschreibung des Befehls **Einloggen** erläutert die möglichen Situationen beim Einloggen und Laden.

Wenn Sie versuchen, ein SPS-Projekt zu laden, während die gleiche Version dieses Projektes bereits auf der Steuerung liegt, erscheint die Meldung: „Programm ist unverändert. Applikation wurde nicht geladen“. TwinCAT lädt das Projekt nicht auf die SPS.

Beim Laden erscheint in der Ansicht **Ausgabe** ein Protokoll der ablaufenden Aktionen (Erzeugen des Codes, Durchführen der Initialisierungen etc.). Weiterhin werden Informationen zu den Speicherbereichen, zur Größe des Codes, der globalen Daten und des allozierten Speichers ausgegeben. Zum Zweck der Übersichtlichkeit werden im Gegensatz zum Online-Change die geänderten Bausteine nicht mehr aufgelistet.

**Siehe auch:**

- [Befehl Einloggen \[► 156\]](#)

## 5.8.5 Befehl Online-Change

**Funktion:** Der Befehl dient dem Anstoßen eines Online-Change auf das gerade aktive SPS-Projekt. Dabei lädt TwinCAT nur die geänderten Teile eines bereits auf der Steuerung laufenden SPS-Projekts neu in die Steuerung.

**Aufruf:** Menü PLC

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb.

Ein Online-Change ist nicht möglich nach den Befehlen **Alles bereinigen** und **Bereinigen**. Der Bereinigen-Prozess löscht die Compileinformationen (Übersetzungsprotokoll), die automatisch bei jeder Codeerzeugung gespeichert werden und die die Basis für einen Online-Change sind.

### **WARNUNG**

#### **Sach- und Personenschäden durch unerwartetes Verhalten der Maschine oder Anlage**

Ein Online-Change verändert das laufende Anwendungsprogramm und bewirkt keinen Neustart. Abhängig von der gesteuerten Maschine können Schäden an Maschine und Werkstücken entstehen oder Gesundheit und Leben von Personen gefährdet werden.

- Stellen Sie sicher, dass der neue Programmcode das gewünschte Verhalten des gesteuerten Systems bewirkt.

#### **Projektspezifische Initialisierungen**

**i** Wenn ein Online-Change durchgeführt wird, werden die projektspezifischen Initialisierungen (Referenzfahrt etc.) nicht ausgeführt, weil die Maschine ihren Status beibehält. Aus diesem Grund hat der neue Programmcode möglicherweise nicht den gewünschten Effekt.

#### **Schwerwiegende Änderungen im Downloadcode**

**i** Wenn der Online-Change schwerwiegende Änderungen im Downloadcode bewirkt (z. B. Verschieben von Variablen nötig), informiert ein Dialog über die Effekte und ermöglicht, den Online-Change abubrechen.

#### **Schneller Online-Change**

**i** Für kleine Änderungen (z. B. kleine Änderung im Implementierungsbereich und Verschieben von Variablen nicht nötig) wird ein „schneller Online-Change“ durchgeführt. In diesem Fall wird nur der jeweils geänderte Baustein übersetzt und nachgeladen. Insbesondere wird in dem Fall kein Initialisierungscode erzeugt. Das bedeutet, dass auch kein Code zur Initialisierung von Variablen mit dem Attribut 'init\_on\_onlchange' erzeugt wird. In der Regel wird das keine Auswirkungen haben, da das Attribut meist dazu verwendet wird, um Variablen mit Adressen zu initialisieren. Beim schnellen Online-Change kommt es jedoch nicht dazu, dass eine Variable ihre Adresse ändert.

Um die Wirkung des Attributs init\_on\_onlchange auf den gesamten Applikationscode sicherzustellen, schalten Sie den schnellen Online-Change mithilfe der Compiler-Definition no\_fast\_online\_change generell für das SPS-Projekt aus. Fügen Sie die Definition zu diesem Zweck in den Eigenschaften des SPS-Projekts in der [Kategorie Übersetzen](#)  103 ein.

#### **Keine Wirkung des Attributs 'init\_on\_onlchange' bei einzelnen FB-Variablen**

**i** Das Attribut 'init\_on\_onlchange' wirkt nur bei globalen Variablen, Programmvariablen und lokalen statischen Variablen von Funktionsbausteinen.

Um einen Funktionsbaustein bei einem Online Change neu zu initialisieren muss die Funktionsbausteininstanz mit dem Attribut deklariert werden. Für eine einzelne Variable in einem Funktionsbaustein wird das Attribut nicht ausgewertet.

### **Zeigervariablen**

Zeiger behalten ihren Wert aus dem letzten Zyklus. Wenn ein Zeiger auf eine Variable zeigt, die durch den Online-Change ihre Größe verändert hat und dadurch im Speicher verschoben wurde, liefert die Zeigervariable nicht mehr die korrekte Position der Variablen. Stellen Sie sicher, dass Pointer in jedem Zyklus erneut zugewiesen werden.

Bei einem Online-Change mit möglicherweise unbeabsichtigten Konsequenzen listet TwinCAT in dem Details-Dialog die geänderten Schnittstellen, betroffenen Variablen und alle Bausteine, für die neuer Code generiert wurde. Wenn sich Speicherorte ändern, wird in einem Dialog auf mögliche Probleme in Zusammenhang mit Pointern hingewiesen.

### **Siehe auch:**

- Dokumentation PLC: SPS-Projekt programmieren

## Was verhindert einen Online-Change?

Es gibt Aktionen in TwinCAT, nach denen ein Online-Change auf einer Steuerung nicht mehr möglich ist. Danach ist immer ein Download des Projekts erforderlich. Ein typischer Fall sind die Aktionen **Bereinigen** und **Alles bereinigen**, die die beim letzten Download abgelegte Übersetzungsinformation löschen. Aber es gibt auch „normale“ Editieraktionen, die dazu führen, dass beim nächsten Einloggen ein Online-Change nicht mehr möglich ist. Folgende Aktionen können einen Online-Change verhindern:

Checkfunktionen	Aktivieren oder Entfernen einer Checkfunktion (CheckBounds, CheckRange, CheckDiv etc.). Änderung in der Schnittstelle einer Checkfunktion (auch das Einfügen und Löschen von lokalen Variablen).
Taskkonfiguration	Ändern in den Konfigurationseinstellungen.
Projekteinstellungen	<u>Kategorie Übersetzen</u> [▶ 103]: In der Sektion <b>Einstellungen</b> (Konstanten ersetzen), Änderung in den Compiler-Defines <u>Kategorie Advanced</u> [▶ 113]: Objekte mit der gleichen ID mehrfach in einer Solution erlauben
Bausteineigenschaften	Änderung der Option Externe Implementierung
Funktionsbaustein	Ändern des Basisbausteins eines Funktionsbausteins (EXTENDS FB_Base), auch das Einfügen oder Löschen eines solchen Basisbausteins. Änderung in der Schnittstellenliste (IMPLEMENTS I_Sample). Ausnahme: Hinzufügen einer neuen Schnittstelle am Ende der Liste.
Datentyp	Ändern des Datentyps einer Variable von einem benutzerdefinierten Datentyp zu einem anderen benutzerdefinierten Datentyp (beispielsweise von TON zu TOF). Ändern des Datentyps von einem benutzerdefinierten Datentyp zu einem Basisdatentyp (beispielsweise von TON zu TIME). Hinweis: Als Workaround ändern Sie gleichzeitig mit dem Datentyp immer auch den Namen der Variablen. Dann wird die Variable als neue Variable initialisiert und die alte entfernt. Ein Online-Change ist danach möglich.

### Siehe auch:

- Ausführen eines Online-Change
- Befehl Einloggen [▶ 156]

### Sehen Sie dazu auch

- Kategorie Common [▶ 100]

## 5.8.6 Befehl Einloggen

Symbol: 

**Funktion:** Der Befehl verbindet das Programmiersystem (das ausgewählte SPS-Projekt) mit dem Zielsystem (Steuerung) und stellt somit den Onlinebetrieb her. Eine Instanz des SPS-Projekts wird auf dem Zielsystem erzeugt und geladen.

**Aufruf:** Menü **PLC** oder **TwinCAT SPS Symbolleistenoptionen** oder Kontextmenü des SPS-Projektobjekts (<SPS-Projektname>Project) im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist fehlerfrei und das Zielsystem befindet sich im Run-Modus.

Mögliche Situationen beim Einloggen:

- Das SPS-Projekt existiert noch nicht auf der Steuerung: Sie werden aufgefordert, den Download zu bestätigen.
- Das SPS-Projekt liegt bereits auf der Steuerung und wurde seit dem letzten Download nicht verändert. Das Einloggen erfolgt ohne weitere Interaktion mit Ihnen.
- Das SPS-Projekt liegt bereits auf der Steuerung, wurde aber seit dem letzten Download verändert. Sie werden aufgefordert, eine der folgenden Optionen zu wählen:

- Mit Online-Change einloggen (Beachten Sie zum Online-Change die Hinweise im Abschnitt „Befehl Online-Change [► 154]“)
- Mit Download einloggen
- Ohne Änderung einloggen

An dieser Stelle erhalten Sie außerdem die Möglichkeit, das Bootprojekt auf der Steuerung zu aktualisieren.

- Eine unbekannte Version des SPS-Projekts liegt bereits auf der Steuerung. Sie werden gefragt, ob TwinCAT diese ersetzen soll.
- Eine Version des SPS-Projekts liegt bereits auf der Steuerung und läuft. Sie werden gefragt, ob TwinCAT trotzdem einloggen und das gerade laufende SPS-Programm überschreiben soll.
- Das SPS-Programm auf der Steuerung hält gerade an einem Haltepunkt. Sie haben ausgeloggt und das Programm geändert: TwinCAT warnt Sie, dass im Falle eines Online-Change oder Downloads die SPS komplett angehalten wird. Dies geschieht auch, wenn mehrere Tasks vorhanden sind und nur eine von dem Haltepunkt betroffen ist.

### Übersetzen des Projekts vor dem Einloggen

Wenn ein SPS-Projekt seit seiner letzten Änderung noch nicht kompiliert wurde, übersetzt TwinCAT das Projekt vor dem Einloggen. Dieser Vorgang entspricht dem Befehl **Übersetzen im ausgeloggten Zustand**.

Wenn während des Übersetzens Fehler auftreten, erscheint ein Meldungsdialog. Die Fehler werden in der Ansicht **Fehlerliste** ausgegeben. Sie können dann entscheiden, ob Sie einloggen wollen, ohne das Programm auf die Steuerung zu laden.

#### Siehe auch:

- [Befehl SPS-Projekt erstellen \[► 126\]](#)

### Fehler beim Login

Wenn während des Einloggens auf die Steuerung ein Fehler auftritt, bricht TwinCAT den Ladevorgang mit einer Fehlermeldung ab. Der Fehlerdialog bietet Ihnen die Möglichkeit, die Fehlerdetails anzuzeigen. Ist ein Ausnahmefehler aufgetreten und der Text \*SOURCEPOSITION\* in der Log-Meldung enthalten, können Sie mit dem Befehl **Im Editor anzeigen** die betroffene Funktion im Editor anzeigen. Dabei springt der Cursor an die fehlerverursachende Zeile.

### Ausgabe von Informationen zum Ladevorgang

Wenn TwinCAT das Projekt beim Einloggen auf die Steuerung lädt, werden folgende Informationen im Meldungsfenster ausgegeben:

- Generierte Codegröße
- Größe der globalen Daten
- Resultierender Speicherbedarf auf der Steuerung
- Eine Liste der betroffenen Bausteine (bei Online-Change)

---

**i** Im Onlinebetrieb können Sie die Einstellungen von Geräten oder Modulen nicht verändern. Um Geräteparameter zu ändern, müssen Sie das SPS-Projekt ausloggen. Abhängig vom Bussystem kann es jedoch einige spezielle Parameter geben, die Sie auch im Onlinebetrieb verändern können.

---

**i** Die Konfiguration der Ansicht speichert TwinCAT im Online- und Offlinebetrieb separat voneinander. Zusätzlich werden Ansichten, welche in einer Betriebsart nicht nutzbar sind, geschlossen. Aus diesem Grund kann sich die Ansicht beim Einloggen automatisch ändern.

---

## 5.8.7 Befehl Start

Symbol: 

Tastaturkürzel: **[F5]**

**Funktion:** Der Befehl startet die Ausführung des Programms.

**Aufruf:** Menü **PLC**, **TwinCAT SPS Symbolleistenoptionen**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb.

Wenn Sie den Befehl aus dem Menü **PLC** aufrufen, wirkt er auf das gerade aktive SPS-Projekt.

## 5.8.8 Befehl Stop

Symbol: 

Tastaturkürzel: **[Umschalt] + [F5]**

**Funktion:** Der Befehl stoppt die Ausführung des Programms.

**Aufruf:** Menü **PLC**, **TwinCAT SPS Symbolleistenoptionen**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb.

Wenn Sie den Befehl aus dem Menü **PLC** aufrufen, wirkt er auf das gerade aktive SPS-Projekt.

## 5.8.9 Befehl Ausloggen

Symbol: 

**Funktion:** Der Befehl beendet die Verbindung zwischen Entwicklungssystem und Zielsystem (Steuerung oder simuliertes Gerät), wodurch in den Offlinebetrieb gewechselt wird.

**Aufruf:** Menü **PLC**, **TwinCAT SPS Symbolleistenoptionen**

## 5.8.10 Befehl Reset kalt

Symbol: 

**Funktion:** Der Befehl setzt alle Variablen des aktiven SPS-Projekts, mit Ausnahme der PERSISTENT-Variablen und der RETAIN-Variablen, auf ihren Initialisierungswert zurück.

**Aufruf:** Menü **PLC**, **TwinCAT SPS Symbolleistenoptionen**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb.

Alle Variablen des aktiven SPS-Projekts, mit Ausnahme der PERSISTENT-Variablen und der RETAIN-Variablen, werden zurückgesetzt. Die Situation entspricht der beim Start eines Anwendungsprogramms, das gerade auf die Steuerung geladen wurde („Kaltstart“).

Haltepunkte des SPS-Programms, die vor dem Zurücksetzen des aktiven SPS-Projekts aktiviert waren, sind nach der Ausführung des Befehls weiterhin aktiviert. Haltepunkte, die vorher deaktiviert waren, sind nach der Ausführung des Befehls weiterhin deaktiviert.

Wenn Sie den Befehl auswählen, während der Programmablauf gerade auf einem Haltepunkt stoppt, werden Sie gefragt, ob der aktuelle Zyklus noch beendet werden soll. Alternativ führt TwinCAT den Reset sofort aus. Allerdings sind nicht alle Laufzeitsysteme in der Lage einen Reset ohne vorheriges Beenden des aktuellen Zyklus durchzuführen.

Nach dem Reset müssen Sie das SPS-Programm mit dem Befehl **Start** starten.

**Siehe auch:**

- Dokumentation PLC: Remanente Variablen - PERSISTENT, RETAIN
- Dokumentation PLC: Reset des SPS-Projekts durchführen
- [Befehl Reset Ursprung](#) [► [159](#)]

- [Befehl Start](#) [▶ 157]

## 5.8.11 Befehl Reset Ursprung

Symbol: 

**Funktion:** Der Befehl setzt alle Variablen des aktiven SPS-Projekts, einschließlich der remanenten Variablen (RETAIN-, PERSISTENT-Variablen), auf ihre Initialisierungswerte zurück und löscht das Anwendungsprogramm auf der Steuerung.

**Aufruf:** Menü **PLC, TwinCAT SPS Symbolleistenoptionen**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb.

Ein Reset deaktiviert die aktuell im Programm gesetzten Haltepunkte. Wenn Sie den Befehl auswählen, während der Programmablauf gerade auf einem Haltepunkt stoppt, werden Sie gefragt, ob der aktuelle Zyklus noch beendet werden soll. Alternativ führt TwinCAT den Reset sofort aus. Allerdings sind nicht alle Laufzeitsysteme in der Lage einen Reset ohne vorheriges Beenden des aktuellen Zyklus durchzuführen.

**Siehe auch:**

- Dokumentation PLC: Remanente Variablen - PERSISTENT, RETAIN
- Dokumentation PLC: Reset des SPS-Projekts durchführen
- [Befehl Reset kalt](#) [▶ 158]

## 5.8.12 Befehl Einzelzyklus

Symbol: 

**Funktion:** Der Befehl führt das aktive SPS-Programm für einen Zyklus aus. Ein begonnener Zyklus wird beendet. Haltepunkte werden ignoriert.

**Aufruf:** Menü **PLC, TwinCAT SPS Symbolleistenoptionen**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb.

## 5.8.13 Befehl Ablaufkontrollmodus umschalten

Symbol: 

**Funktion:** Der Befehl aktiviert oder deaktiviert die Ablaufkontrolle.

**Aufruf:** Menü **PLC, TwinCAT SPS Symbolleistenoptionen**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb.

In dem Dialog zur Aktivierung der Ablaufkontrolle kann in einer Auswahlliste die Task ausgewählt werden, auf die die Ablaufkontrolle angewendet wird. Standardmäßig ist die Option **Task automatisch auswählen** ausgewählt. Wenn der Ablaufkontrollmodus aktiviert ist, stehen im Projektmappen-Explorer im Kontextmenü aller Tasks die Befehle **Flow Control** → **Diese Task auswählen** und **Flow Control** → **Task automatisch auswählen** zur Verfügung, um eine andere Task für die Ablaufkontrolle auszuwählen.



Eine aktive Ablaufkontrolle verlängert die Laufzeit des SPS-Projekts!

---

**Siehe auch:**

- Dokumentation PLC: Ablaufkontrolle

## 5.8.14 Befehl Werte forcen

Symbol: 

**Funktion:** Der Befehl setzt den Wert einer Variablen auf der Steuerung dauerhaft auf einen vordefinierten Wert.

**Aufruf:** Menü **PLC, TwinCAT SPS Symbolleistenoptionen**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb.

### VORSICHT

#### Sach- und Personenschäden durch unerwartetes Verhalten der Maschine oder Anlage

Das außerordentliche Ändern von Variablenwerten in einem auf der Steuerung laufenden SPS-Programm kann zu einem unerwarteten Verhalten der gesteuerten Maschine führen. Abhängig von der gesteuerten Maschine können Schäden an Maschine und Werkstücken entstehen oder Gesundheit und Leben von Personen gefährdet werden.

- Evaluieren Sie vor einem Forcen von Variablenwerten mögliche Gefahren und treffen Sie entsprechende Sicherheitsvorkehrungen.

Mit dem Befehl setzt TwinCAT eine oder mehrere Variablen der aktiven Applikation auf der Steuerung permanent auf definierte Werte. Dieses Setzen erfolgt jeweils am Beginn und am Ende eines Abarbeitungszyklus. Abfolge der Abarbeitung: 1. Eingänge lesen, 2. Werte forcen 3. Code abarbeiten, 4. Werte forcen, 5. Ausgänge schreiben.

Sie können Werte vorbereiten durch

- Klicken in das Feld **vorbereiteter Wert** im Deklarationsteil und geben Sie den neuen Wert ein. Bei einer booleschen Variablen ändern Sie den Wert durch einfaches Klicken in das Feld.
- Klicken in das Inline-Monitoring-Feld im Implementierungsteil des FUP/KOP/AWL-Editors und geben Sie den neuen Wert ein.
- Klicken in das Feld **vorbereiteter Wert** im Überwachen-Fenster und geben Sie den neuen Wert ein.

Ein „geforceter“ Wert wird mit einem  Symbol gekennzeichnet.

Expression	Type	Value
 iCount	INT	45
 bSwitich	BOOL	 TRUE
 Axis1	AXIS_REF	

TwinCAT führt das Forcen durch, bis es explizit durch den Anwender aufgehoben wird durch

- den Befehl **Forcen aufheben**
- das Aufheben des Forces über den Dialog **Wert vorbereiten**
- Ausloggen aus der Applikation



Der Befehl **Werte forcen für alle Applikationen**, der alle SPS-Projekte im TwinCAT-Projekt betrifft, ist standardmäßig nicht in einem Menü enthalten.

**Siehe auch:**

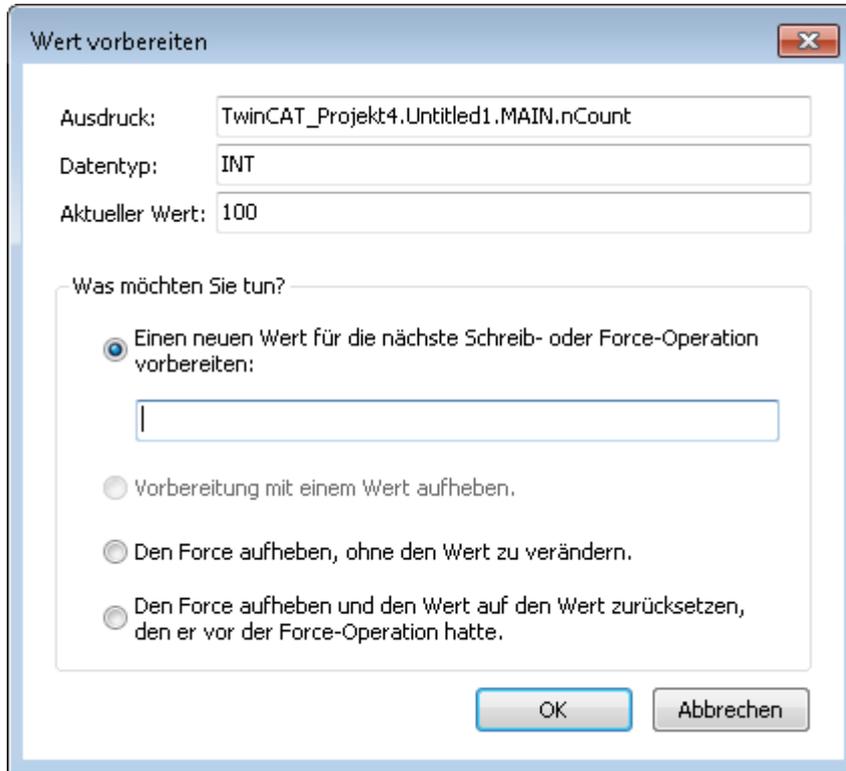
- [Befehl Forcen aufheben \[► 161\]](#)
- [Dialog Wert vorbereiten \[► 161\]](#)
- Dokumentation PLC: Forcen und Schreiben von Variablen

### 5.8.14.1 Dialog Wert vorbereiten

**Funktion:** Der Dialog dient zur Vorbereitung eines Werts für eine bereits geforcete Variable. TwinCAT führt die vorbereitete Aktion mit dem nächsten Forcen aus.

**Aufruf:** TwinCAT öffnet den Dialog in folgenden Situationen:

- wenn Sie im Deklarationsteil in das Feld **vorbereiteter Wert** einer geforcten Variable klicken
- wenn Sie im Implementierungsteil in das Inline-Monitoring-Feld einer geforcten Variablen klicken
- wenn Sie im Überwachen-Fenster in das Feld **vorbereiteter Wert** einer geforcten Variable klicken



Einen neuen Wert für die nächste Schreib- oder Force-Operation vorbereiten	Wert, den TwinCAT bei der nächsten Force-Operation auf die Variable schreibt
Vorbereitung mit einem Wert aufheben	TwinCAT löscht den vorbereiteten Wert.
Den Force aufheben, ohne den Wert zu verändern.	TwinCAT behält den geforcten Wert bei und beendet das Forcen. TwinCAT markiert die Variable mit <Unforce>.
Den Force aufheben und den Wert auf den Wert zurücksetzen, den er vor der Force-Operation hatte.	TwinCAT setzt den geforcten Wert zurück und beendet das Forcen. Dabei wird die Variable mit <Unforce and restore> markiert.

**Siehe auch:**

- [Befehl Werte forcen \[► 160\]](#)

### 5.8.15 Befehl Forcen aufheben

Symbol:

**Funktion:** Der Befehl setzt das Forcen aller Variablen zurück. Die Variablen erhalten dabei ihren aktuellen Wert aus der Steuerung.

**Aufruf:** Menü **PLC, TwinCAT SPS Symbolleistenoptionen**

**Voraussetzung:** Das SPS-Projekt befindet sich im Onlinebetrieb.

**⚠ VORSICHT****Sach- und Personenschäden durch unerwartetes Verhalten der Maschine oder Anlage**

Das außerordentliche Ändern von Variablenwerten in einem auf der Steuerung laufenden SPS-Programm kann zu einem unerwarteten Verhalten der gesteuerten Maschine führen. Abhängig von der gesteuerten Maschine können Schäden an Maschine und Werkstücken entstehen oder Gesundheit und Leben von Personen gefährdet werden.

- Evaluieren Sie vor einem Zurücksetzen der geforcten Variablenwerte mögliche Gefahren und treffen Sie entsprechende Sicherheitsvorkehrungen.



Der Befehl **Forcen aufheben für alle Applikationen**, der alle SPS-Projekte im TwinCAT-Projekt betrifft, ist standardmäßig nicht in einem Menü enthalten.

**Siehe auch:**

- [Befehl Werte forcen \[► 160\]](#)
- Dokumentation PLC: Forcen und Schreiben von Variablen

**5.8.16 Befehl Werte schreiben**

Symbol:

**Funktion:** Der Befehl setzt den Wert einer Variablen auf der Steuerung einmalig auf einen vordefinierten Wert.

**Aufruf:** Menü **PLC, TwinCAT SPS Symbolleistenoptionen**

**Voraussetzung:** Das SPS-Projekt befindet sich im Onlinebetrieb.

**⚠ VORSICHT****Sach- und Personenschäden durch unerwartetes Verhalten der Maschine oder Anlage**

Das außerordentliche Ändern von Variablenwerten in einem auf der Steuerung laufenden SPS-Programm kann zu einem unerwarteten Verhalten der gesteuerten Maschine führen. Abhängig von der gesteuerten Maschine können Schäden an Maschine und Werkstücken entstehen oder Gesundheit und Leben von Personen gefährdet werden.

- Evaluieren Sie vor einem Schreiben von Variablenwerten mögliche Gefahren und treffen Sie entsprechende Sicherheitsvorkehrungen.

Mit dem Befehl setzen Sie eine oder mehrere Variablen des aktiven SPS-Projekts auf der Steuerung einmalig auf definierte Werte. Das Schreiben erfolgt einmal zu Beginn des nächsten Zyklus.

Sie können Werte vorbereiten durch

- Klicken in das Feld **vorbereiteter Wert** im Deklarationsteil und geben Sie den neuen Wert ein. Bei einer booleschen Variablen ändern Sie den Wert durch einfaches Klicken in das Feld.
- Klicken in das Inline-Monitoring-Feld im Implementierungsteil des FUP/KOP/AWL-Editors und geben Sie den neuen Wert ein.
- Klicken in das Feld **vorbereiteter Wert** im Überwachen-Fenster und geben Sie den neuen Wert ein.



Der Befehl **Werte schreiben für alle Applikationen**, der alle SPS-Projekte im TwinCAT-Projekt betrifft, ist standardmäßig nicht in einem Menü enthalten.

**Siehe auch:**

- [Befehl Werte forcen \[► 160\]](#)
- Dokumentation PLC: Forcen und Schreiben von Variablen

### 5.8.17 Befehl Darstellung - Binär, Dezimal, Hexadezimal

**Funktion:** Die Befehle des Untermenüs **Darstellung** dienen der Einstellung des Formats für die Darstellung der Werte beim Monitoring im Onlinebetrieb.

**Aufruf:** Menü **PLC**, Kontextmenü

**Voraussetzung:** Das SPS-Projekt ist im Offline- oder Onlinebetrieb.



Die Darstellungsformate „Binär“ und „Hexadezimal“ sind vorzeichenlos, „Dezimal“ ist vorzeichenbehaftet.

**Siehe auch:**

- Dokumentation PLC: Monitoring von Werten
- Dokumentation PLC: Deklarationseditor

### 5.8.18 Befehl Darstellung Vererbung – Einfach, Strukturiert



Verfügbar ab TwinCAT 3.1 Build 4026

**Funktion:** Die Befehle des Untermenüs **Darstellung Vererbung** dienen der Einstellung des Formats für die Darstellung der Vererbungshierarchie von Funktionsbausteinen und Strukturen beim Monitoring im Onlinebetrieb.

**Aufruf:** Menü **PLC**, **Kontextmenü**

**Voraussetzung:** Das SPS-Projekt ist im Offline- oder Onlinebetrieb.

Strukturiert	Die Vererbungshierarchie von Funktionsbausteinen und Strukturen wird in einer Baumstruktur dargestellt. Die Variablen werden dabei als Kindknoten des Funktionsbausteins oder der Struktur dargestellt, in dem sie deklariert sind.
Einfach	Die Darstellung erfolgt als flache Liste.

**Siehe auch:**

- Dokumentation PLC: Monitoring von Werten
- Dokumentation PLC: Deklarationseditor

### 5.8.19 Befehl Lokalisierungsvorlage erzeugen

**Funktion:** Der Befehl öffnet den Dialog **Lokalisierungsvorlage erzeugen**. Hier definieren Sie, welche Textinformationen aus dem Projekt in eine Übersetzungsvorlage des Dateiformats pot exportiert werden sollen.

**Aufruf:** Menü **PLC > Project Localization**

**Voraussetzung:** Ein Projekt ist geöffnet.

#### Dialog Lokalisierungsvorlage erzeugen

Der Dialog dient der Auswahl der textuellen Informationen, die in die Lokalisierungsvorlage aufgenommen werden sollen.

**Folgende Informationen einschließen**

Namen	Texte wie beispielsweise Dialogtitel, Objektnamen im SPS-Projektbaum
Bezeichner	Variablenbezeichner, Beispiel: „nCounter“
Zeichenfolgen	Beispielsweise „count“ in der folgenden Deklaration: <code>sVar : STRING := 'count'</code>
Kommentare	Kommentartexte in den Programmierbausteinen
Positionsinformationen	<p>Auswahl, welche Positionen der oben ausgewählten Textkategorien im Projekt in die Übersetzungsdatei aufgenommen werden sollen. Die Positionsinformation steht jeweils in der/ den ersten Zeile(n) eines Abschnitts für eine Übersetzung.</p> <p>Beispiel:</p> <pre>#: D:\Proj1.project\Project_Settings:1 msgid 'Projekteinstellungen' msgstr ''</pre> <ul style="list-style-type: none"> <li>„Alle“: Alle gefundenen Positionen des Texts werden aufgeführt.</li> <li>„Erstes Auftreten“: In die Übersetzungsdatei wird die Position im Projekt aufgenommen, an der der zu übersetzende Text erstmalig auftritt.</li> <li>„Keine“</li> </ul>
Erzeugen	Die Schaltfläche öffnet den Dialog zum Speichern einer Datei. Die Übersetzungsvorlage wird in einer Textdatei vom Typ POT Translation Template (*.pot) angelegt. Jedes weitere Erzeugen erstellt wieder eine komplette neue Vorlagendatei.

## 5.8.20 Befehl Lokalisierungen verwalten

**Funktion:** Der Befehl öffnet den Dialog **Lokalisierung verwalten**. Im Dialog wählen Sie die gewünschte Lokalisierungssprache oder die Originalversion des Projekts aus. Weiterhin können Sie hier Lokalisierungsdateien \*.<Sprache>.po ins Projekt aufnehmen oder daraus entfernen.

**Aufruf:** Menü **PLC > Project Localization**

**Voraussetzung:** Ein Projekt ist geöffnet.

### Dialog Lokalisierung verwalten

Verfügbare Lokalisierungen	<p>Liste der im Projekt vorliegenden Lokalisierungsdateien.</p> <p>Beispiel:</p> <pre>proj1-de.po proj1-en.po &lt;Originalversion&gt;</pre> <p>Die Originalversion ist immer verfügbar. Nur in der Originalversion kann das Projekt editiert werden.</p>
Hinzufügen	Die Schaltfläche öffnet den Dialog zum Auswählen einer weiteren po-Datei aus dem Dateisystem.
Entfernen	Die Schaltfläche entfernt die links ausgewählte po-Datei aus dem Projekt.
Standardlokalisierung	 Die gerade selektierte Lokalisierung wird zur Standardlokalisierung. Der Eintrag wird fett dargestellt.
Lokalisierung wechseln	Mit der Schaltfläche wechseln Sie zur gerade selektierten Lokalisierung.
OK	Das Projekt wird in der Landessprache dargestellt, die durch die unter Dateien ausgewählte Datei geliefert wird. Wenn Sie <b>&lt;Originalversion&gt;</b> auswählen, erscheint das Projekt in der editierbaren, nicht lokalisierten Fassung.

## 5.8.21 Befehl Lokalisierung umschalten

**Symbol:** 

**Funktion:** Der Befehl schaltet zwischen der aktuell eingestellten Projektllokalisierung und der <Originalversion> um.

**Aufruf:**

- Menü **Project > Localization**
- Schaltfläche im Dialog **Lokalisierungen verwalten**
- Schaltfläche in der Werkzeugleiste

**Voraussetzung:** Ein Projekt ist geöffnet. Eine Standardlokalisierung für das Projekt ist im Dialog **Lokalisierungen verwalten** definiert.

**Siehe auch:**

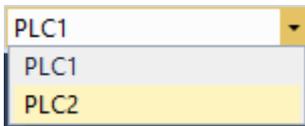
- [Befehl Lokalisierungen verwalten](#) [► 164]

## 5.8.22 Befehl Active PLC Project

**Funktion:** Drop-down-Liste zur Auswahl des aktiven SPS-Projekts. Das aktuell fokussierte Projekt wird automatisch als aktives SPS-Projekt gesetzt.

**Aufruf:** TwinCAT SPS Symbolleistenoptionen

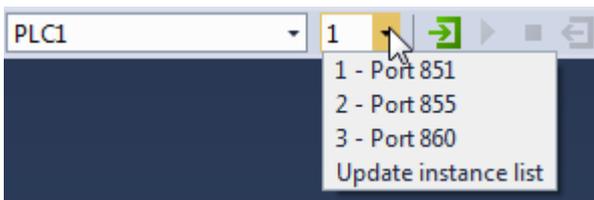
**Voraussetzung:** Das TwinCAT Projekt enthält mehrere SPS-Projekte.

**Siehe auch:**

- [Befehl Active PLC Instance](#) [► 165]
- [Befehl Einloggen](#) [► 156]

## 5.8.23 Befehl Active PLC Instance

**Funktion:** Drop-down-Liste zur Auswahl der aktiven SPS-Instanz des zugehörigen SPS-Projekts.

**Aufruf:** TwinCAT SPS Symbolleistenoptionen**Siehe auch:**

- [Befehl Active PLC Project](#) [► 165]
- [Befehl Einloggen](#) [► 156]

## 5.9 Extras

### 5.9.1 Befehl Optionen

**Funktion:** Der Befehl öffnet den Dialog **Optionen** zur Konfiguration der TwinCAT-Optionen. Diese Optionen definieren das Verhalten und Aussehen der TwinCAT-Benutzeroberfläche. TwinCAT speichert die aktuellen Einstellungen auf dem lokalen System als Standardeinstellungen.

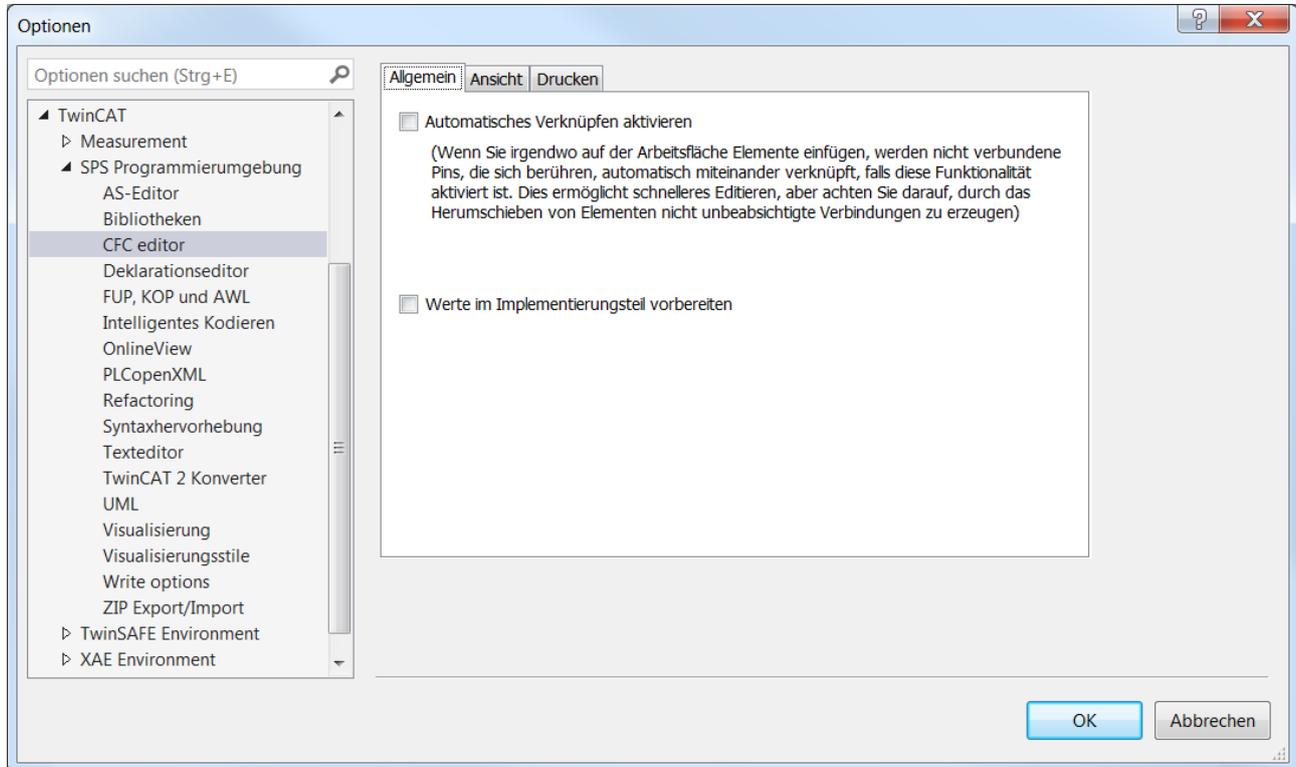
**Aufruf:** Menü **Extras > Optionen > TwinCAT > SPS Programmierumgebung**

### 5.9.1.1 Dialog Optionen - CFC-Editor

**Funktion:** Der Dialog dient der Konfiguration der Einstellungen für das Editieren und Drucken im CFC-Editor.

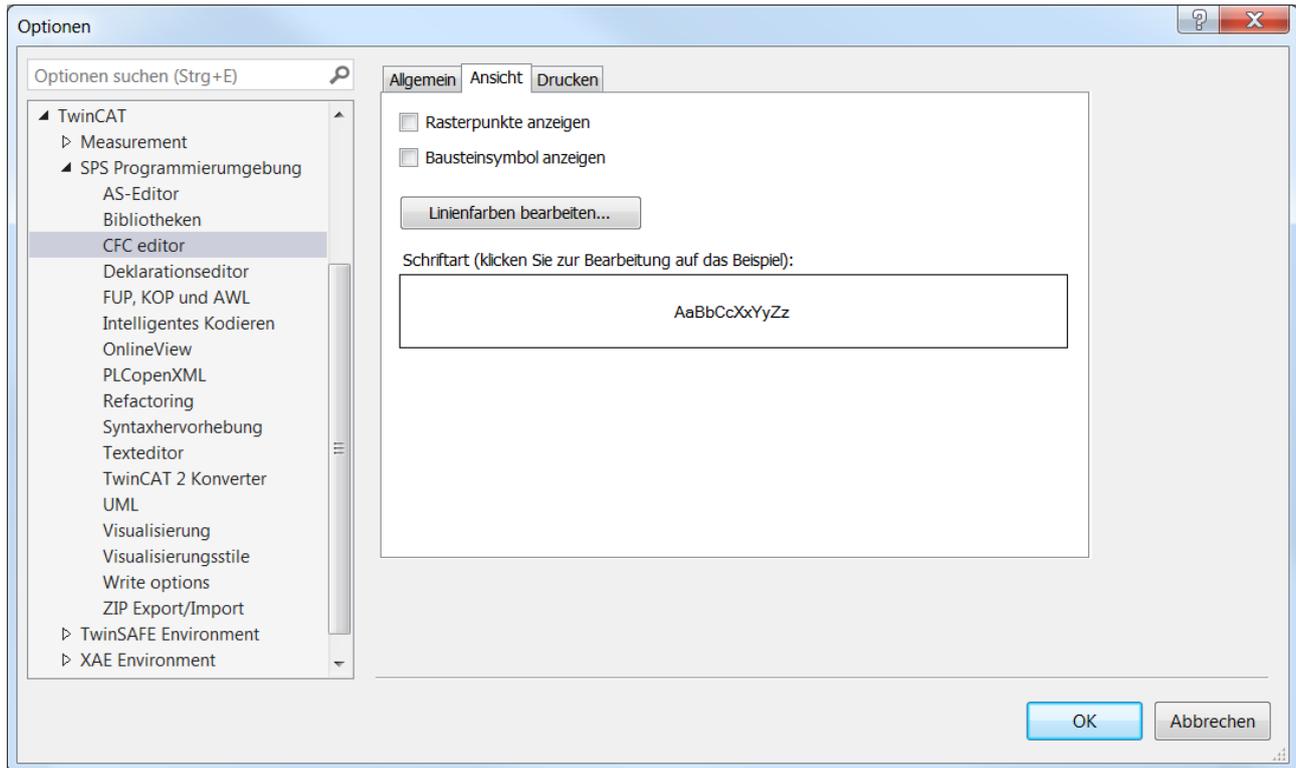
**Aufruf:** TwinCAT > SPS Programmierumgebung > CFC-Editor

#### Registerkarte Allgemein



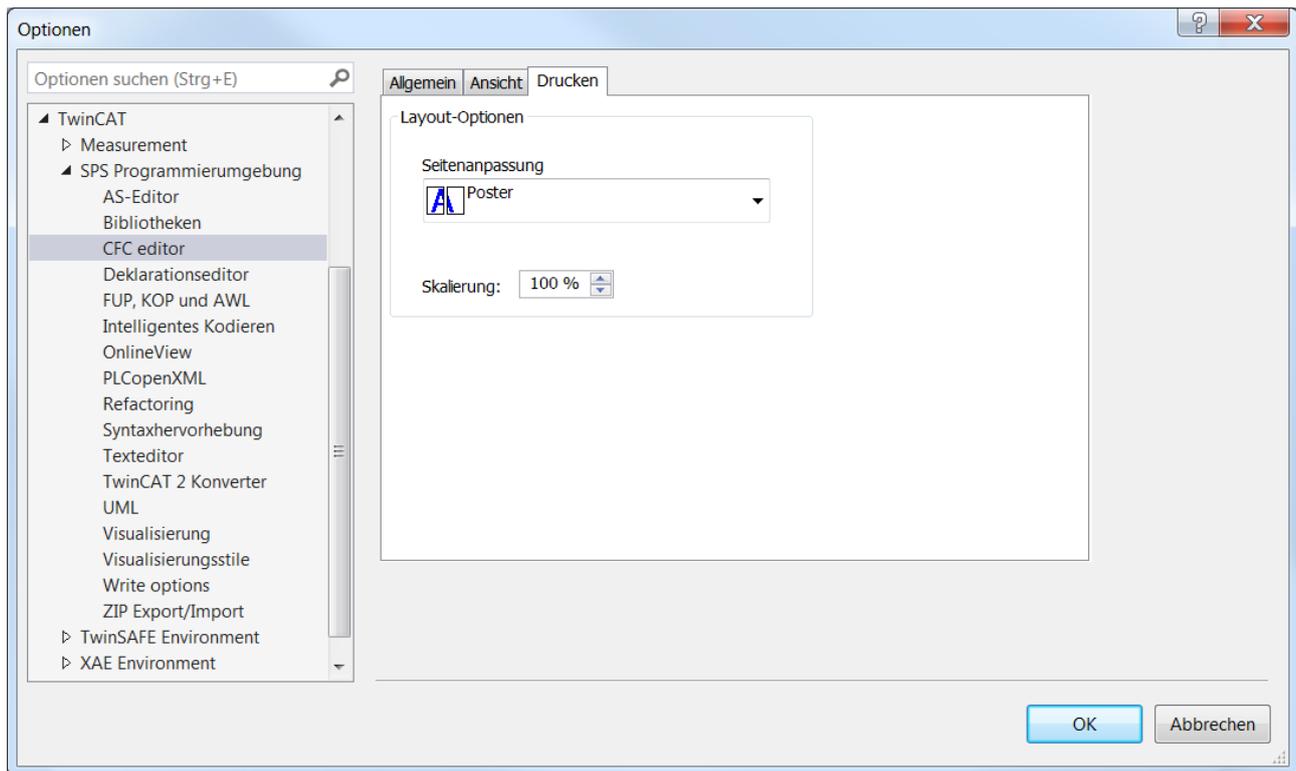
<p>Automatisches Verknüpfen aktivieren</p>	<p><input checked="" type="checkbox"/> : Wenn Sie ein CFC-Element auf die Arbeitsfläche des Editors ziehen und einfügen, verbindet TwinCAT automatisch unverknüpfte Pins, die sich „berühren“. Achten Sie darauf, dass Sie beim Verschieben von Elementen keine unerwünschten Verknüpfungen erzeugen!</p>
<p>Werte im Implementierungsteil vorbereiten</p>	<p><input checked="" type="checkbox"/> : Im Onlinebetrieb können Sie auch im Implementierungsteil des CFC-Bausteins Variablenwerte für das Schreiben und Forcen vorbereiten. Außerdem zeigt TwinCAT die gerade vorbereiteten Werte in der Inline-Monitoring-Box der Variablen in spitzen Klammern an.</p>

Registerkarte Ansicht



Rasterpunkte anzeigen	<input checked="" type="checkbox"/> : Im Editor sind Rasterpunkte Gültigkeitsbereich, an denen Sie die Elemente positionieren können.
Bausteinsymbol anzeigen	<input checked="" type="checkbox"/> : TwinCAT stellt im CFC-Editor vorhandene Funktionsbausteine, die mit einem Bitmap verknüpft sind, als Symbol dar. Voraussetzung: Sie haben für einen Funktionsbaustein oder eine Funktion die Verknüpfung entweder in den Objekteigenschaften erstellt oder über eine Bibliothek geladen.
Linienfarben bearbeiten	Öffnet den Dialog <b>Linienfarben bearbeiten</b> zur Definition der Farben für die Verbindungslinien in Abhängigkeit vom anliegenden Datentyp. Die Linien erscheinen im Offline- und Onlinebetrieb in diesen Farben, außer TwinCAT übermalt diese Farben durch die fette schwarze und blaue Liniendarstellung, die booleschen Datenfluss anzeigt. <ul style="list-style-type: none"> <li>• Typ hinzufügen: Fügt der Liste eine Datentyp hinzu.</li> <li>• Typ entfernen</li> </ul>
Schriftart	Anzeige der Schriftart und Schaltfläche zur Änderung der Schriftart.

**Registerkarte Drucken**



**Layout-Optionen**

Seitenanpassung	Seite oder Poster
Skalierung	Mögliche Werte: 20 % - 200 %

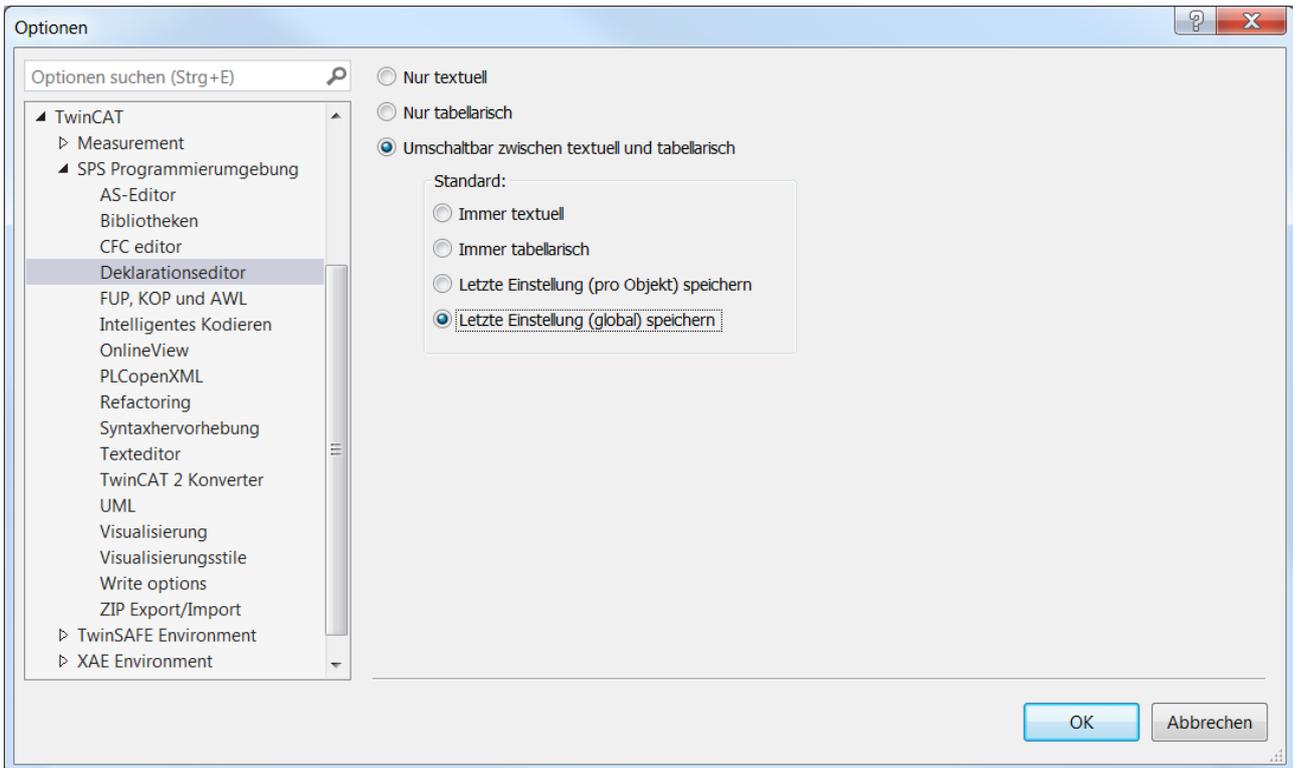
**Siehe auch:**

- Dokumentation PLC: Programmieren in CFC
- Dokumentation PLC: Programmiersprachen und ihre Editoren

**5.9.1.2 Dialog Optionen - Deklarationseditor**

**Funktion:** Der Dialog dient der Konfiguration der Anzeigeeinstellungen für den Deklarationseditor.

**Aufruf:** TwinCAT > SPS Programmierumgebung > Deklarationseditor



Nur textuell	Textuelle Ansicht des Deklarationseditors
Nur tabellarisch	Tabellarische Ansicht des Deklarationseditors
Umschaltbar zwischen textuell und tabellarisch	<p>Der Deklarationseditor bietet zwei Schaltflächen, um zwischen der textuellen und der tabellarischen Ansicht zu wechseln:</p> <p> : textuelle Ansicht</p> <p> : tabellarische Ansicht</p> <p>Die folgende Option definiert die Ansicht, die standardmäßig beim Öffnen eines Programmierobjekts erscheint:</p> <ul style="list-style-type: none"> <li>• Immer textuell</li> <li>• Immer tabellarisch</li> <li>• Letzte Einstellung (pro Objekt) speichern</li> <li>• Letzte Einstellung (global) speichern</li> </ul>

**Siehe auch:**

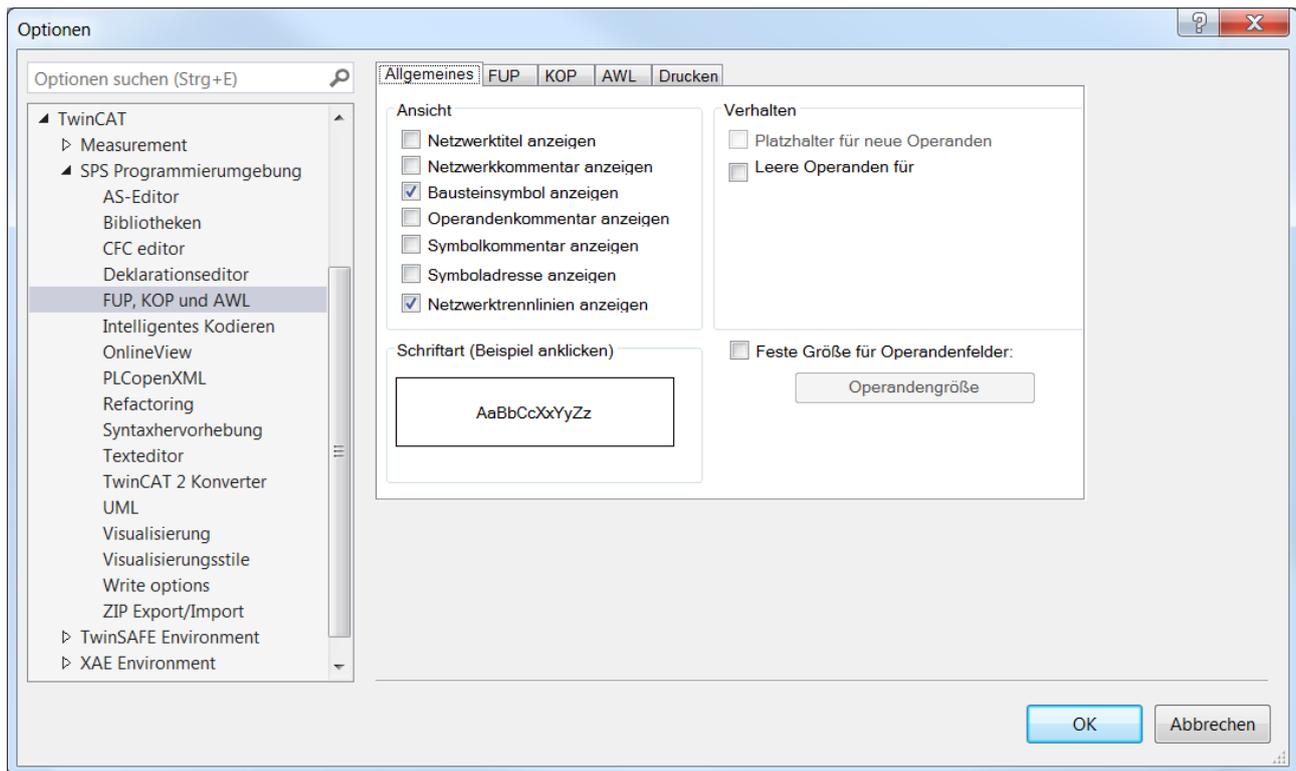
- Dokumentation PLC: Deklarationseditor verwenden

**5.9.1.3 Dialog Optionen - FUP, KOP und AWL**

**Funktion:** Der Dialog dient der Konfiguration der Darstellungsoptionen für den FUP/KOP/AWL-Editor.

**Aufruf:** TwinCAT > SPS Programmierumgebung > FUP, KOP und AWL

**Registerkarte Allgemeines**



**Ansicht**

Netzwerktitel anzeigen	Netzwerktitel wird in der oberen linken Ecke des Netzwerks angezeigt.
Netzwerkcommentar anzeigen	Netzwerkcommentar wird in der oberen linken Ecke des Netzwerks angezeigt. Wenn TwinCAT zusätzlich den Netzwerktitel darstellt, erscheint der Kommentar in der Zeile unterhalb.
Bausteinsymbol anzeigen	Bausteinsymbol wird im Bausteinelement im FUP- und KOP-Editor angezeigt. Auch die Standardoperatoren haben Symbole.
Operandencommentar anzeigen	TwinCAT zeigt den Kommentar an, den Sie einer Variablen im Implementierungsteil gegeben haben. Der Operandencommentar bezieht sich nur auf die lokale Verwendungsstelle der Variablen, im Gegensatz zum „Symbolcommentar“.
Symbolcommentar anzeigen	TwinCAT zeigt den Kommentar, den Sie einer Variablen oder einem Symbol bei der Deklaration gegeben haben, oberhalb des Variablennamens an. Zusätzlich oder anstelle des Symbolcommentars können Sie auch einen lokalen „Operandencommentar“ zuweisen.
Symboladresse anzeigen	Wenn einem Symbol (Variable) eine Adresse zugewiesen ist, wird diese Adresse oberhalb des Variablennamens angezeigt.
Netzwerkrennlinien anzeigen	Zwischen den einzelnen Netzwerken wird eine Trennlinie angezeigt.

**Verhalten**

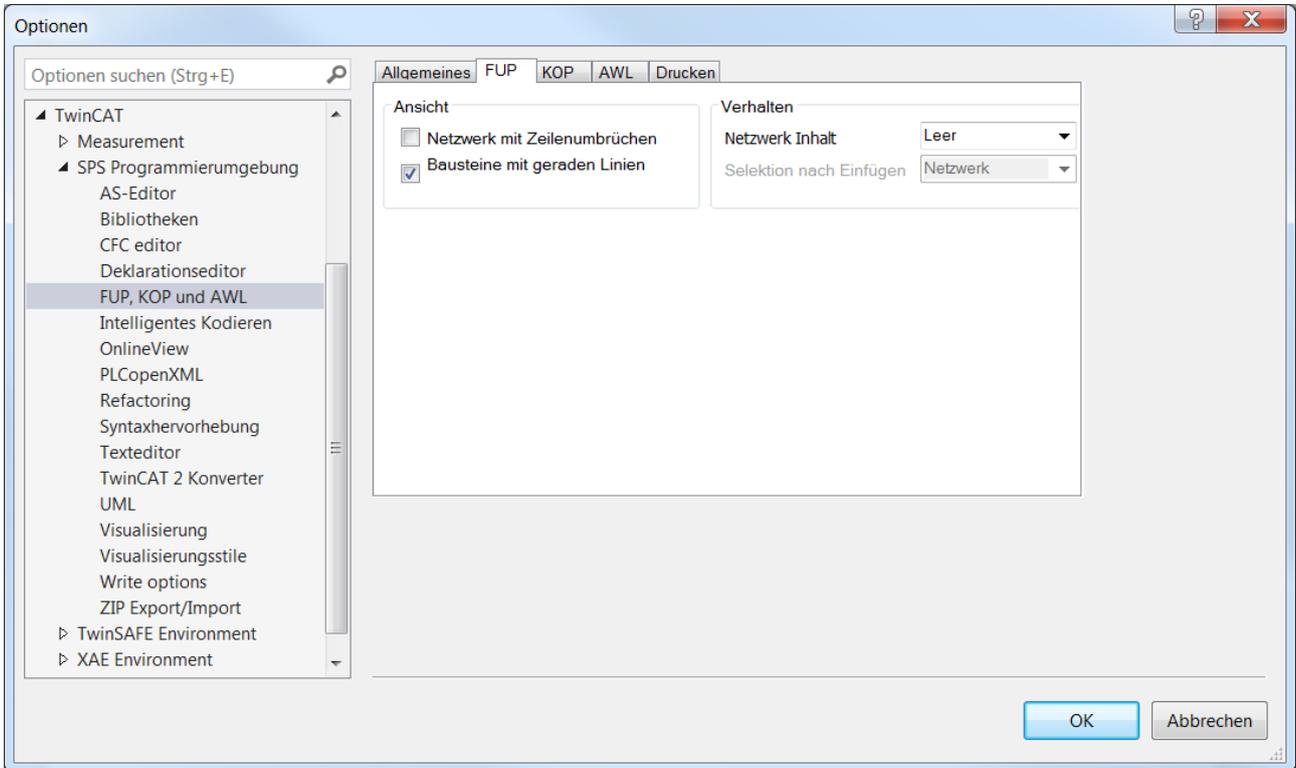
Platzhalter für neue Operanden	Operandenfeld der Pins der neuen Funktionsbausteine wird leer gelassen (anstelle von „???“).
Leere Operanden für Funktionsbaustein-Pins	Fügt anstelle von ??? leere Operanden ein.

**Schriftart**

Klick auf das Eingabefeld öffnet den Dialog **Schriftart**.

Feste Größe für Operandenfelder	<input checked="" type="checkbox"/> : Operandengröße bearbeiten ist aktivierbar.
Operandengröße bearbeiten	Dialog <b>Größe Operanden</b> erscheint zum Einstellen der Anzahl Zeichen und Zeilen.

**Registerkarte FUP**



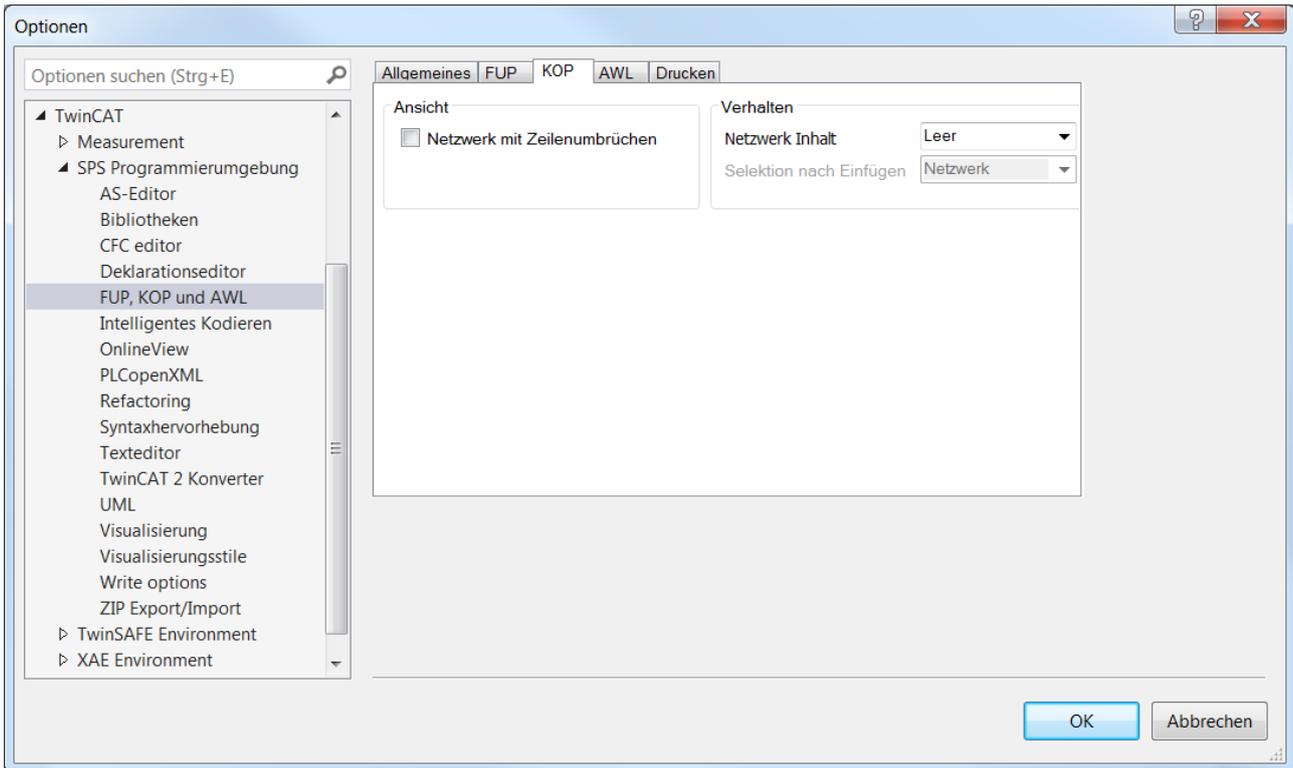
**Ansicht**

Netzwerk mit Zeilenumbrüchen	<input checked="" type="checkbox"/> : Darstellung der Netzwerke mit Zeilenumbrüchen, so dass TwinCAT möglichst viele Bausteine in der aktuellen Breite des Fensters darstellen kann.
Bausteine mit geraden Linien verbinden	<input checked="" type="checkbox"/> : Linien zwischen den Elementen erhalten eine feste, kurze Länge.

**Verhalten**

Netzwerk Inhalt	Auswahlliste: Inhalt eines neuen Netzwerks.
Selektion nach Einfügen	Auswahlliste: Element, das TwinCAT nach dem Einfügen eines neuen Netzwerks selektiert.

**Registerkarte KOP**



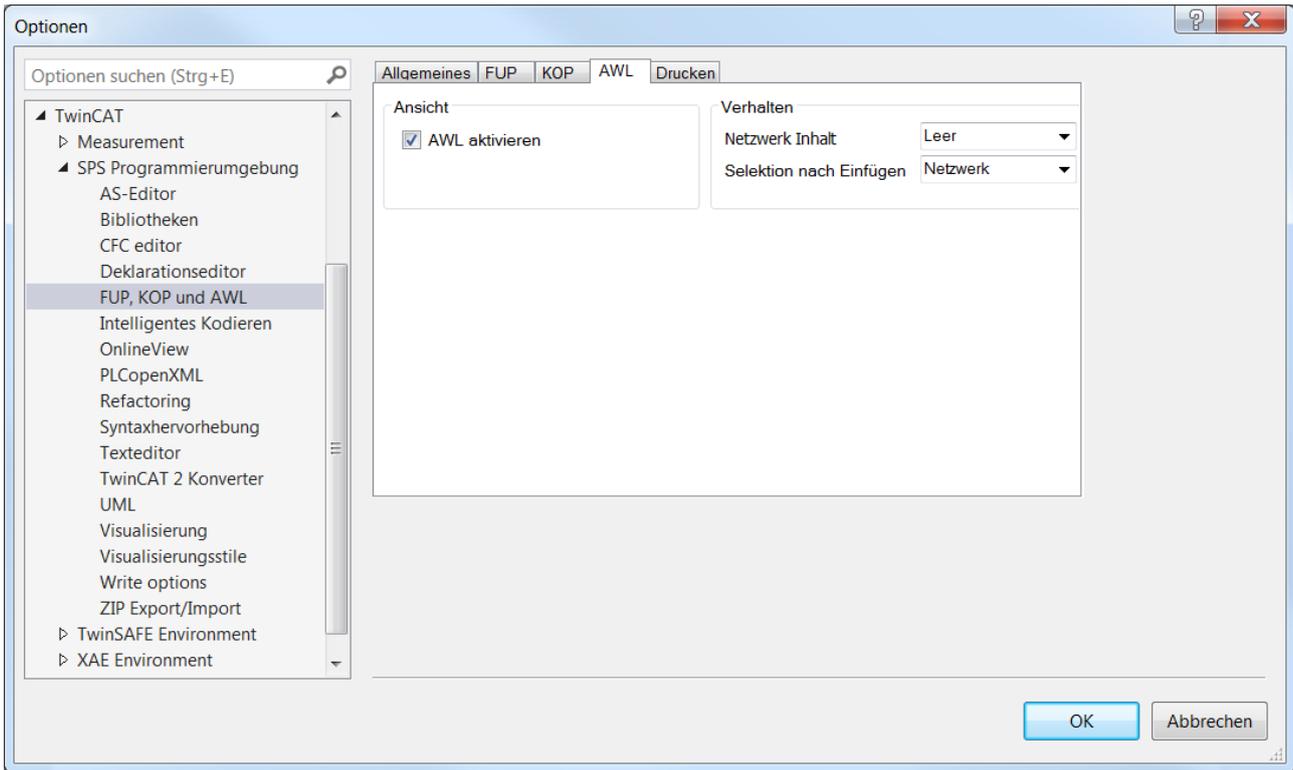
**Ansicht**

Netzwerk mit Zeilenumbrüchen	<input checked="" type="checkbox"/> : Darstellung der Netzwerke mit Zeilenumbrüchen, so dass TwinCAT möglichst viele Bausteine in der aktuellen Breite des Fensters darstellen kann.
------------------------------	--

**Verhalten**

Netzwerk Inhalt	Auswahlliste: Inhalt eines neuen Netzwerks.
Selektion nach Einfügen	Auswahlliste: Element, das TwinCAT nach dem Einfügen eines neuen Netzwerks selektiert.

Registerkarte AWL



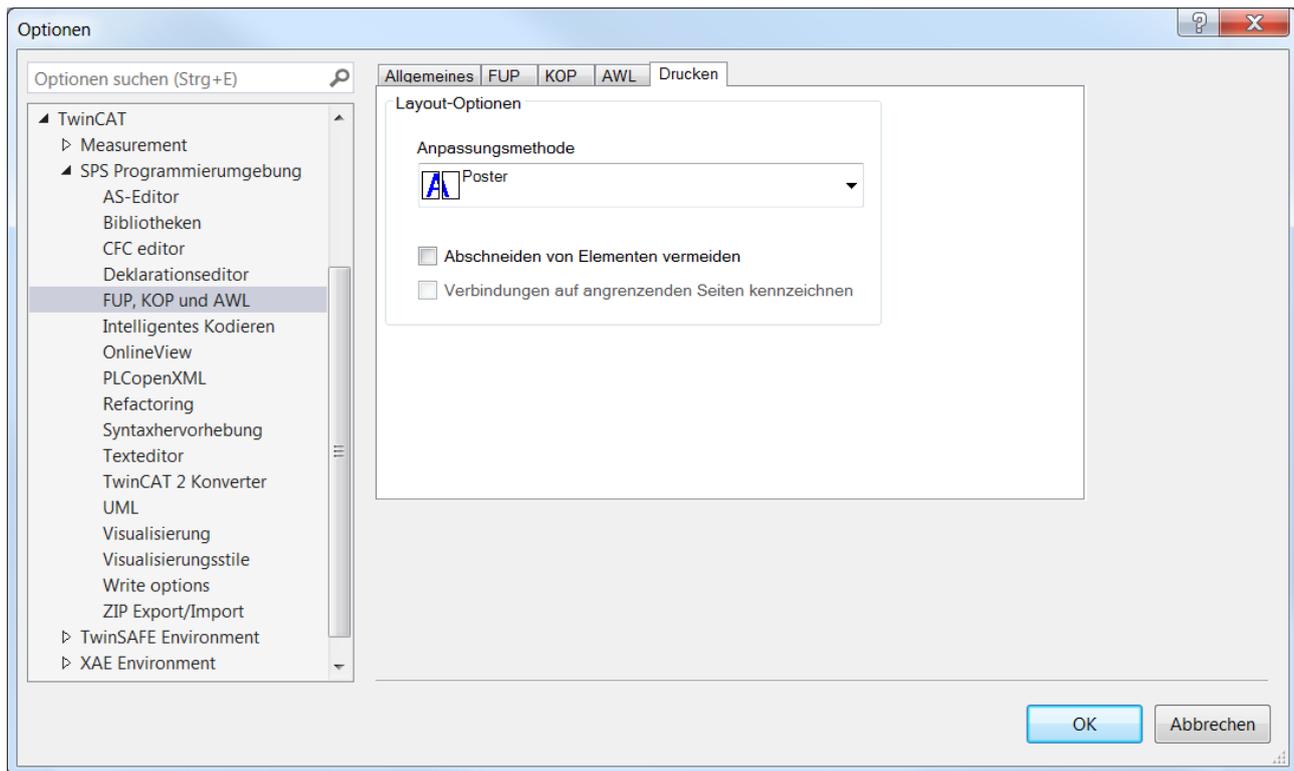
**Ansicht**

AWL aktivieren	Die Implementierungssprache AWL steht im Entwicklungssystem zur Verfügung.
----------------	--

**Verhalten**

Netzwerk Inhalt	Auswahlliste: Inhalt eines neuen Netzwerks.
Selektion nach Einfügen	Auswahlliste: Element, das TwinCAT nach dem Einfügen eines neuen Netzwerks selektiert.

## Registerkarte Drucken



### Layout-Optionen

Anpassungsmethode	Auswahlliste zur Größenanpassung.
Abschneiden von Elementen vermeiden	Elemente, die nicht auf die Seite passen, werden auf der nächsten Seite gedruckt.
Verbindungen auf angrenzenden Seiten kennzeichnen	Aktivierbar, wenn <b>Abschneiden von Elementen vermeiden</b> aktiviert ist.

### Siehe auch:

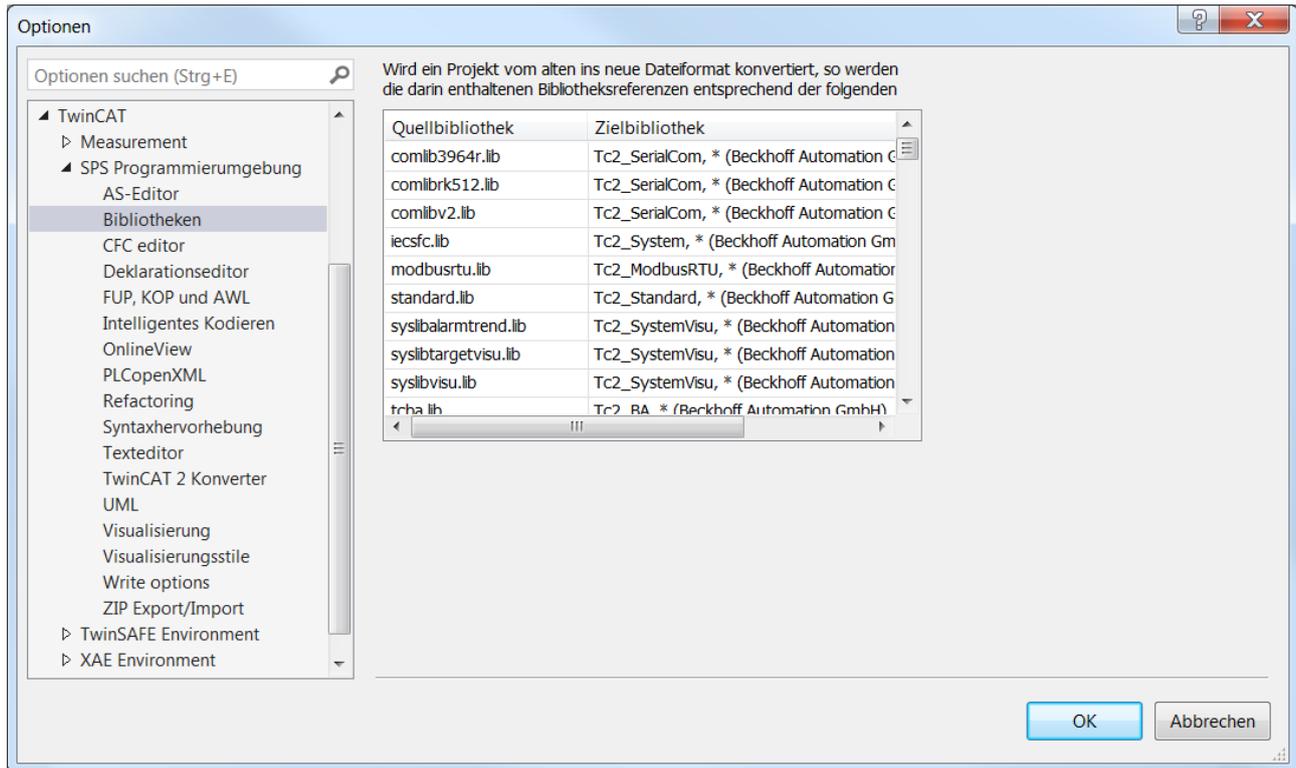
- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: Programmiersprachen und ihre Editoren

### 5.9.1.4 Dialog Optionen - Bibliotheken

**Funktion:** Der Dialog enthält Einstellungen für Bibliotheken.

**Aufruf:** TwinCAT > SPS Programmierumgebung > Bibliotheken

**Registerkarte Konvertierung**



Die Registerkarte dient dem Verwalten der Abbildungen von Bibliotheksreferenzen, die TwinCAT bei der Konvertierung eines alten Projekts anwendet. Wenn Sie für eine bestimmte Bibliothek noch keine Abbildung gespeichert haben, müssen Sie bei jedem Öffnen eines alten Projekts, in das diese Bibliothek eingebunden ist, die Abbildung neu definieren.

Eine Abbildung definiert, wie eine Bibliotheksreferenz nach der Konvertierung des Projekts ins aktuelle Format aussieht. Es gibt drei Möglichkeiten:

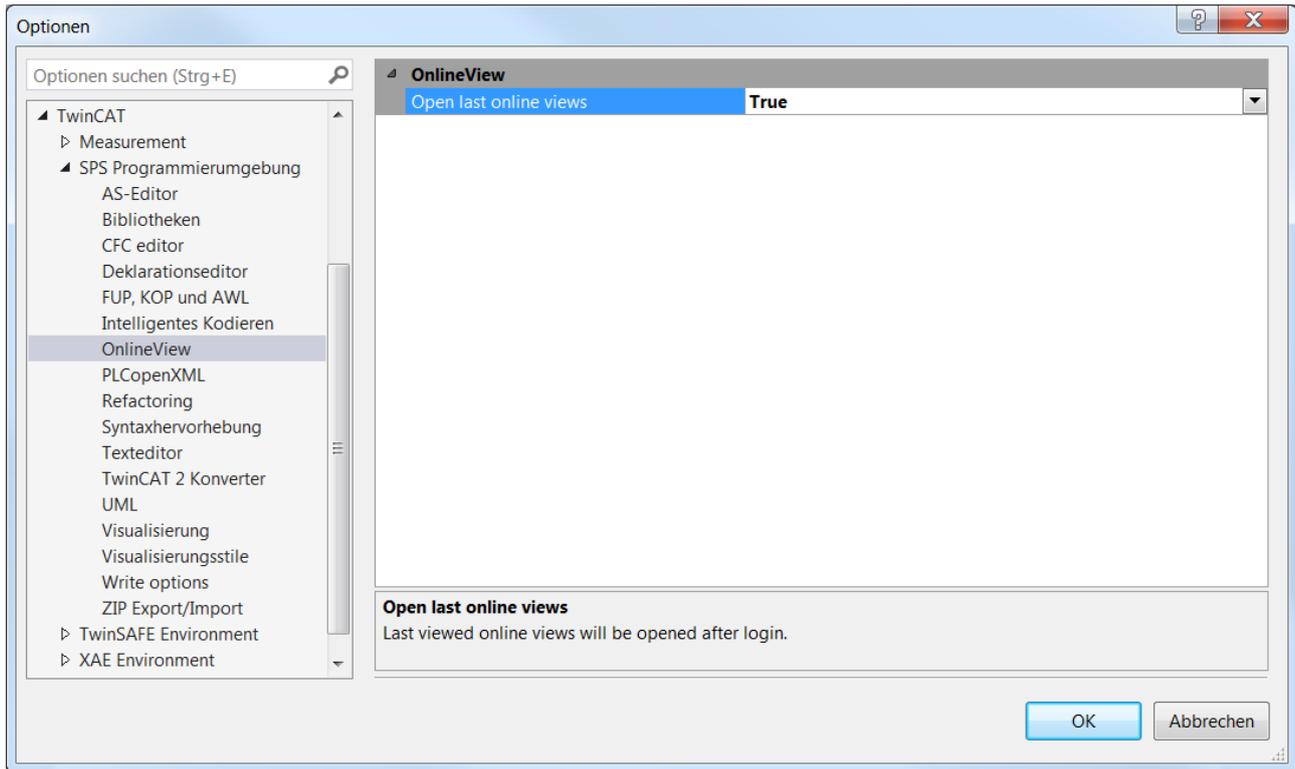
- Sie behalten die Referenz bei. Das bedeutet, dass TwinCAT die Bibliothek ebenfalls ins aktuelle Format (\*.library) konvertiert und im lokalen Bibliotheks-Repository installiert.
- Sie ersetzen eine Referenz durch eine andere Referenz. Das bedeutet, dass eine der installierten Bibliotheken die bisher eingebundene Bibliothek ersetzt.
- Sie löschen die Referenz. Das bedeutet, dass das konvertierte Projekt die Bibliothek nicht mehr einbindet.

TwinCAT wendet alle aufgelisteten Abbildungen bei der nächsten Konvertierung eines alten Projekts auf dessen Bibliotheksreferenzen an. Somit müssen Sie die Abbildungsdefinition wiederholen, wenn dieselbe Bibliothek wieder in einem zu konvertierenden Projekt eingebunden ist. In der letzten Zeile können Sie eine neue Abbildung eingeben.	
Quellbibliothek	Pfad der Bibliothek, die im Projekt vor der Konvertierung eingebunden ist. Ein Doppelklick auf einen Eintrag macht das Feld editierbar und die Schaltfläche für die Eingabeunterstützung erscheint.
Zielbibliothek	Name und Speicherort der Bibliothek, die nach der Konvertierung im Projekt eingebunden sein soll. Ein Doppelklick auf einen Eintrag öffnet den Dialog „Zielsystembibliothek setzen“.

**Siehe auch:**

- Dokumentation PLC: Bibliotheken verwenden

### 5.9.1.5 Dialog Optionen - Online View



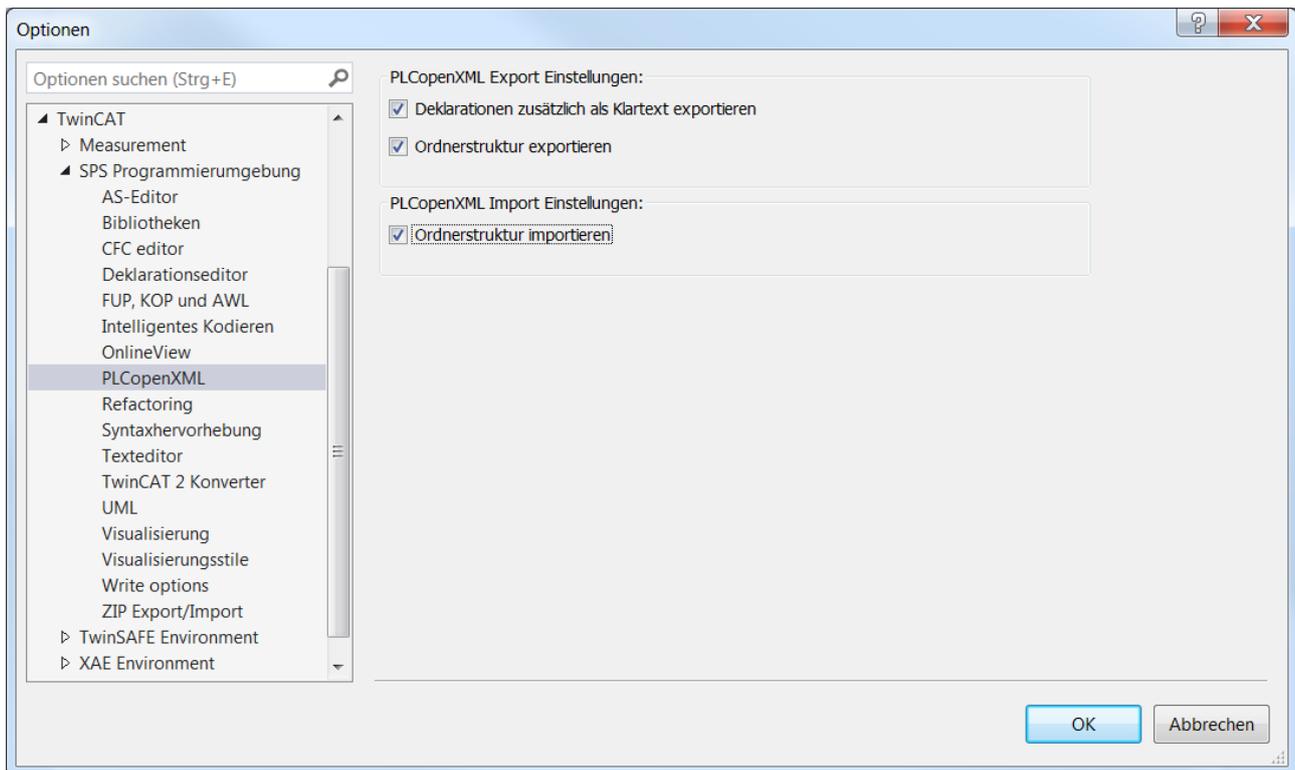
#### OnlineView

<p>Open last online views</p>	<ul style="list-style-type: none"> <li>• TRUE (Standardeinstellung): Beim Einloggen werden die Editorfenster der vorherigen Online-Sitzung geöffnet. Die aktuelle Offline-Ansicht bleibt geöffnet.</li> <li>• FALSE: Beim Einloggen bleibt die Offline-Ansicht geöffnet. Die Editorfenster der vorherigen Online-Sitzung werden verworfen und nicht wieder geöffnet.</li> </ul>
-------------------------------	---

### 5.9.1.6 Dialog Optionen - PLCopenXML

**Funktion:** Der Dialog enthält Einstellungen für das Verhalten von TwinCAT beim PLCopenXML-Export oder -Import.

**Aufruf:** TwinCAT > SPS Programmierumgebung > PLCopenXML



**PLCopenXML Export Einstellungen**

Deklarationen zusätzlich als Klartext exportieren	Standardmäßig splittet TwinCAT die Deklarationsteile gemäß dem PLCopenXML-Schema in einzelne Variablen auf und verliert somit die Formatierung und manche Kommentarinformationen.  <input checked="" type="checkbox"/> : Formatierung und Kommentare bleiben erhalten. TwinCAT schreibt den Klartext des exportierten Deklarationsteils zusätzlich in die PLCopenXML-Datei und erweitert somit das PLCopenXML-Schema.
Ordnerstruktur exportieren	<input checked="" type="checkbox"/> : TwinCAT exportiert auch die Ordner, wenn sie eines der ausgewählten Objekte enthalten. Das ist eine TwinCAT-spezifische Erweiterung zum PLCopenXML-Schema.

**PLCopenXML Import Einstellungen**

Ordnerstruktur importieren	<input checked="" type="checkbox"/> : Wenn die Importdatei-Informationen über die Ordnerstruktur der Objekte enthält, importiert TwinCAT diese Struktur mit.  <input type="checkbox"/> : TwinCAT importiert Objekte ohne Struktur.
----------------------------	--

**Siehe auch:**

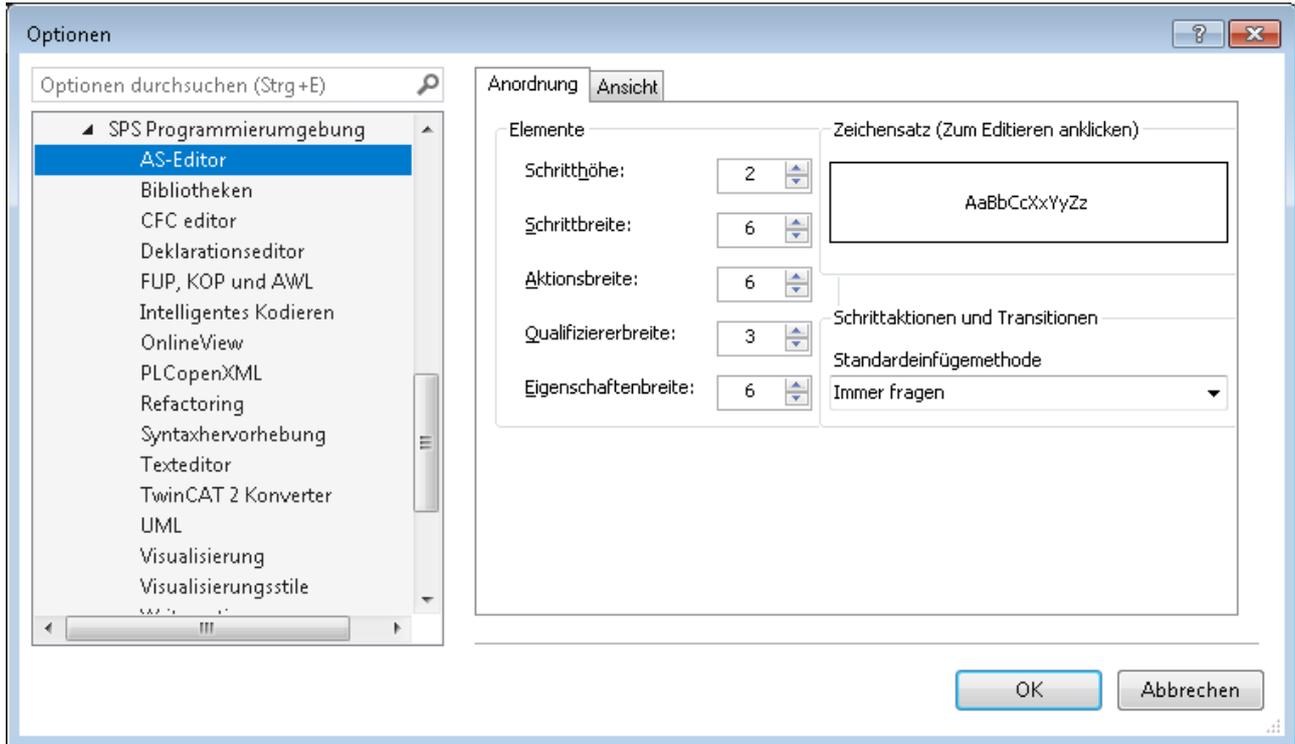
- [Befehl PLCopenXML exportieren \[► 282\]](#)
- [Befehl PLCopenXML importieren \[► 283\]](#)
- Dokumentation PLC: SPS-Projekt exportieren und importieren

**5.9.1.7 Dialog Optionen - AS-Editor**

**Funktion:** Der Dialog dient der Konfiguration der Einstellungen für den AS-Editor.

**Aufruf:** TwinCAT > SPS Programmierumgebung > AS-Editor

**Registerkarte Anordnung**



**Elemente**

Definiert die Größen der AS-Elemente Schritt, Aktion, Qualifizierer und Eigenschaft. Angabe der Werte in Rastereinheiten angegeben. 1 Rastereinheit = Fontgröße, die Sie aktuell in den Texteditor-Optionen gesetzt haben (Textbereich / Schriftart). Die Einstellungen werden immer sofort in allen gerade geöffneten AS-Editorfenstern wirksam.

Schritthöhe	Mögliche Werte: 1-100
Schrittbreite	Mögliche Werte: 2-100
Aktionsbreite	Mögliche Werte: 2-100
Qualifiziererbreite	Mögliche Werte: 2-100
Eigenschaftenbreite	Mögliche Werte: 2-100

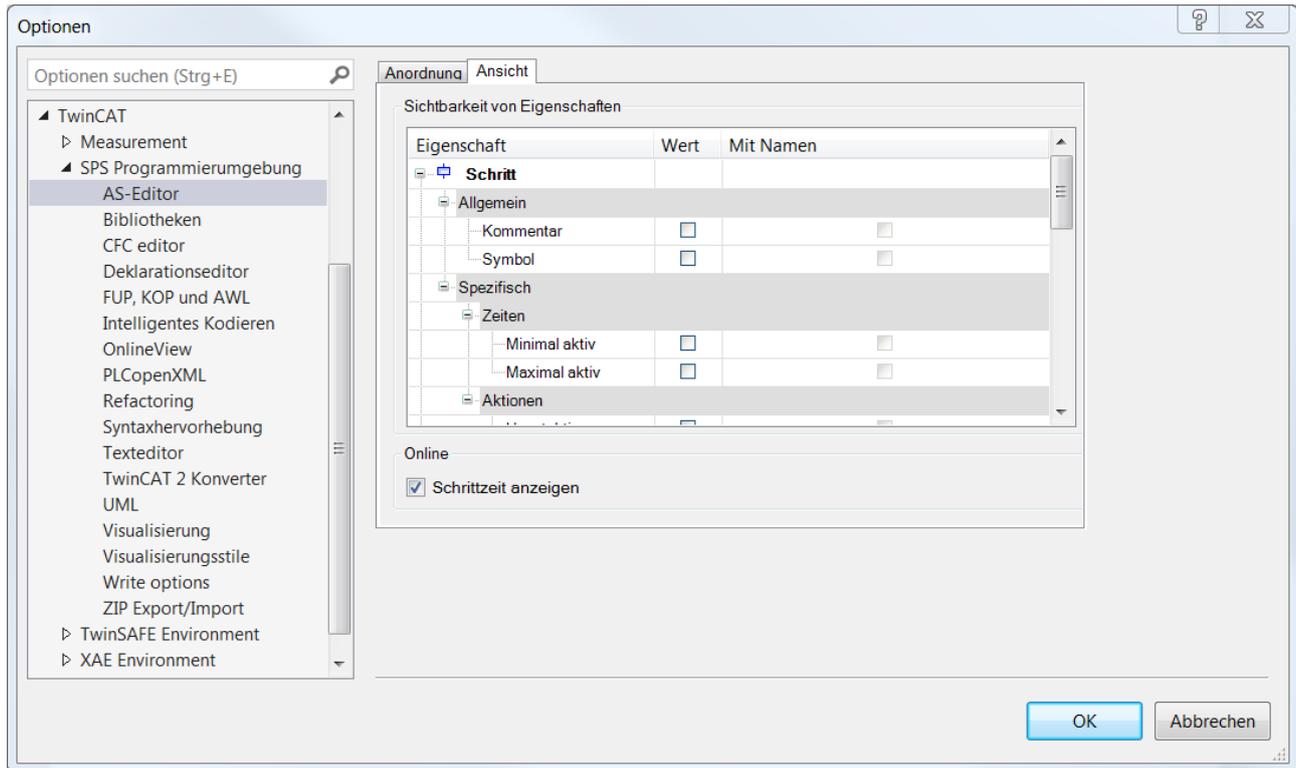
**Zeichensatz**

Der Beispielttext zeigt die aktuell eingestellte Schriftart. Klicken Sie darauf um die Schriftart zu verändern.

**Schrittaktionen**

Standardeingabemethode	<ul style="list-style-type: none"> <li>• Immer fragen</li> <li>• Implementierung duplizieren</li> <li>• Referenz kopieren</li> </ul>
------------------------	--

Registerkarte Ansicht



Sichtbarkeit von Eigenschaften

Auflistung der Elementeigenschaften der Kategorien Allgemein und Spezifisch und Definition der Anzeigeeoptionen.

Eigenschaft	Definiert die Elementeigenschaften, die neben dem Element im AS-Diagramm stehen.
Wert	<input checked="" type="checkbox"/> : Anzeige des Eigenschaftenwerts.
Mit Namen	<input checked="" type="checkbox"/> : Anzeige des Eigenschaftenwerts mit Namen.

Online

Schrittzeit anzeigen	<input checked="" type="checkbox"/> : TwinCAT zeigt im Onlinebetrieb rechts von den Schritten die Schrittzeit an.
----------------------	---

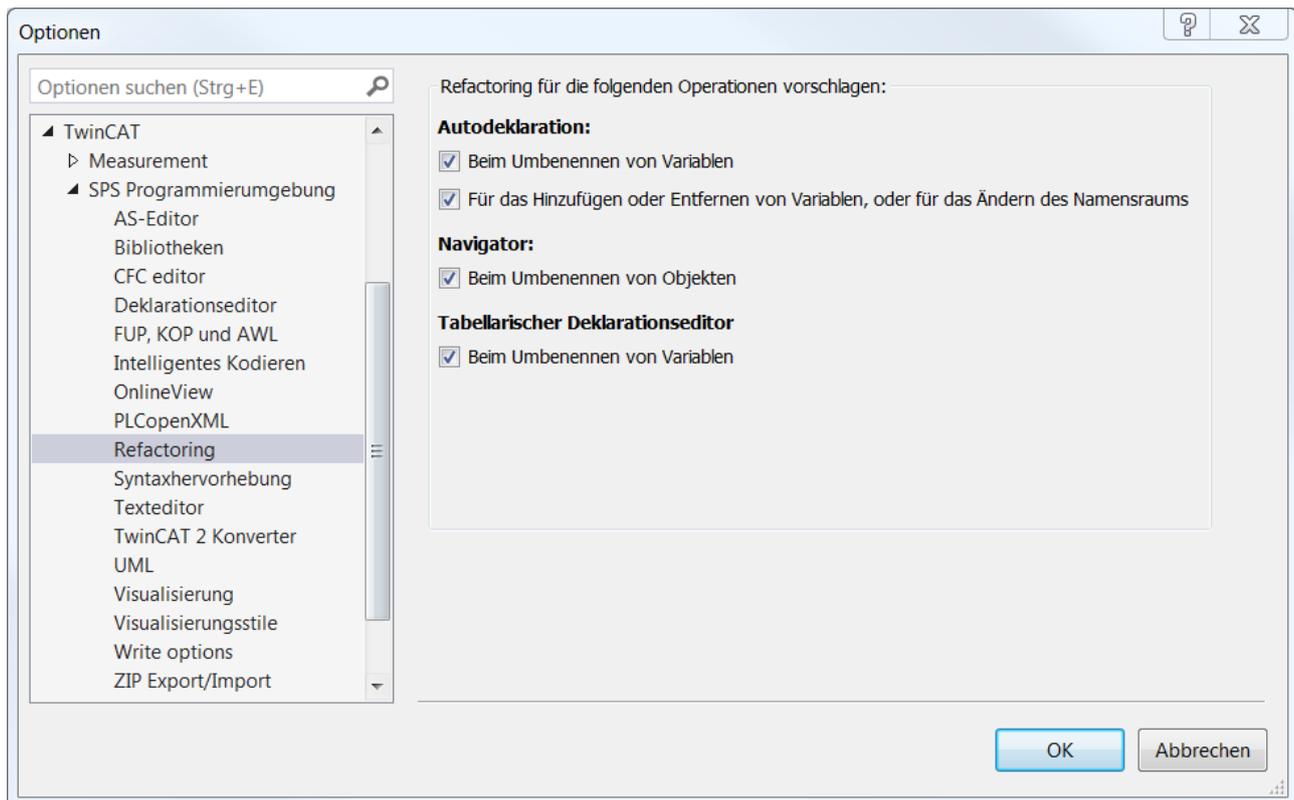
Siehe auch:

- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: Programmiersprachen und ihre Editoren

5.9.1.8 Dialog Optionen - Refactoring

**Funktion:** Der Dialog dient zur Festlegung der Operationen im Projekt, für die automatisch Refactoring vorgeschlagen wird. Die Refactoring-Funktionalität unterstützt Sie bei Ihren Verbesserungswünschen.

**Aufruf:** TwinCAT > SPS Programmierumgebung > Refactoring



#### Autodeklaration

Wenn Sie den Namen einer Variablen ändern oder eine Eingabe- bzw. Ausgabevariable ergänzen, indem Sie die Autodeklaration (Dialog **Variable deklarieren**) aufrufen, ist die Option **Änderungen mit Hilfe von Refactoring anwenden** automatisch aktiviert. Nach Bestätigung des Dialogs öffnet sich der Dialog **Refactoring** und Sie können die Variable projektweit ändern.

#### Beim Hinzufügen oder Entfernen von Variablen, oder für das Ändern des Gültigkeitsbereichs

: Sie fügen über die Autodeklaration (Dialog **Variable deklarieren**) eine neue Eingangs- oder Ausgangsvariable hinzu oder löschen in der Autodeklaration den Namen einer Variablen und beenden den Dialog mit **OK**. Daraufhin öffnet sich der Dialog **Refactoring**, um projektweit die Variable hinzuzufügen oder zu entfernen

#### Beim Umbenennen von Variablen

: Sie benennen in der Autodeklaration (Dialog **Variable deklarieren**) den Namen um und beenden den Dialog mit **OK**. Daraufhin öffnet sich der Dialog **Refactoring**, um projektweit die Variable umzubenennen.

#### Navigator

#### Beim Umbenennen von Objekten

: Wenn Sie im SPS-Projektbaum den Namen eines Objekts ändern, erscheint die Eingabeaufforderung, ob TwinCAT „Automatisches Refactoring“ durchführen soll.

#### Tabellarischer Deklarationseditor

#### Beim Umbenennen von Variablen

: Wenn Sie im tabellarischen Deklarationseditor den Namen einer Variablen ändern, erscheint die Eingabeaufforderung, ob TwinCAT „Automatisches Refactoring“ für das Umbenennen durchführen soll.

## UML Klassendiagramm

Optionen für die Unterstützung von Refactoring bei im Klassendiagramm-Editor vorgenommenen Änderungen.

**Beim Hinzufügen oder Entfernen von Variablen**

: Wenn Sie im Klassendiagramm Variablen in den Sektionen VAR\_INPUT, VAR\_OUTPUT und VAR\_IN\_OUT hinzufügen oder entfernen, wird Refactoring unterstützt.

**Beim Umbenennen eines Bausteins**

: Wenn Sie im Klassendiagramm einen Bausteinnamen ändern, wird Refactoring unterstützt.

**Beim Umbenennen von Variablen oder Eigenschaften**

: Wenn Sie im Klassendiagramm eine Variable oder eine Eigenschaft umbenennen, wird Refactoring unterstützt.

Wenn die Option **Refactoring-Vorschau überspringen** in den UML-Optionen aktiviert ist, wird das Refactoring je nach Fall möglicherweise ohne vorheriges Anzeigen im Dialog **Refactoring** an allen betroffenen Stellen im Projekt durchgeführt. (siehe [Dialog Optionen - UML](#) [▶ 190])

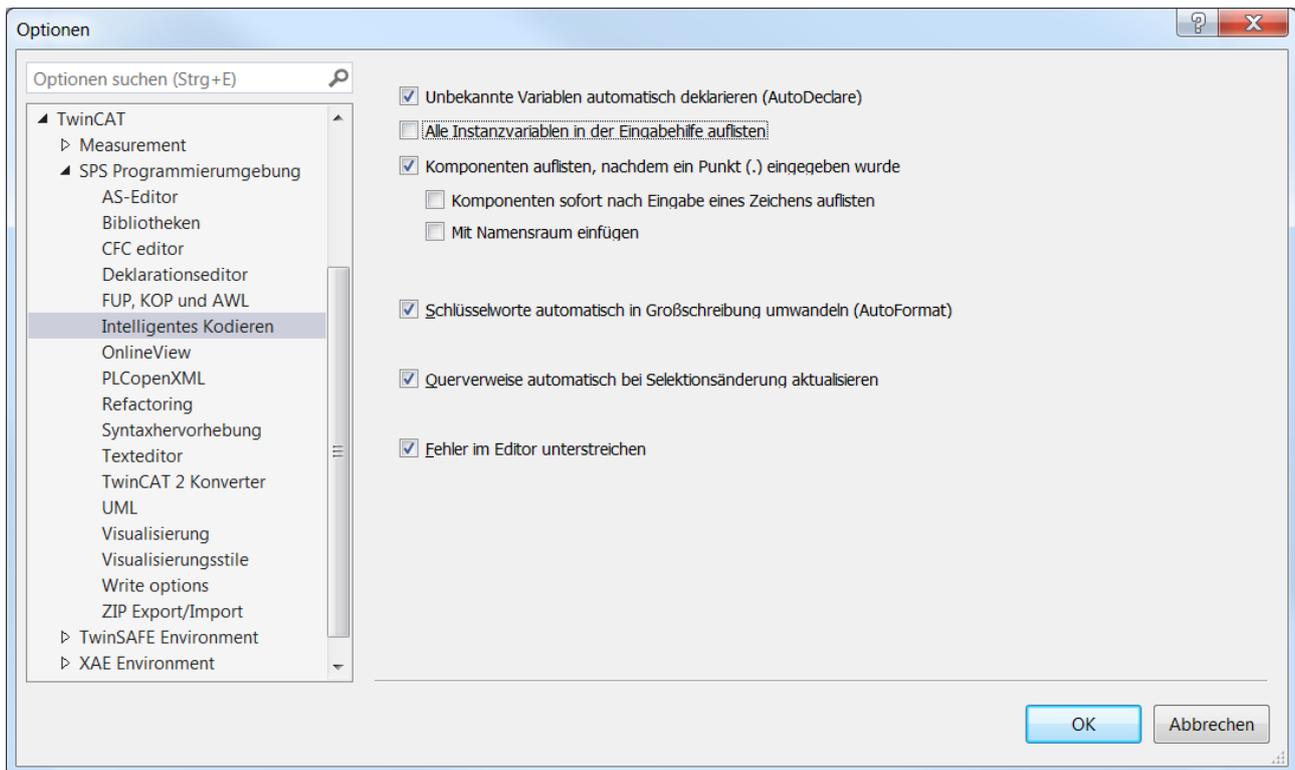
**Siehe auch:**

- [Befehl Variable deklarieren](#) [▶ 67]
- [Befehl Refactoring - <Variable> umbenennen](#) [▶ 78]
- [Befehl Refactoring - Variable hinzufügen](#) [▶ 80]
- [Befehl Refactoring - <Variable> entfernen](#) [▶ 81]
- Dokumentation PLC: Refactoring

**5.9.1.9 Dialog Optionen - Intelligentes Codieren**

**Funktion:** Der Dialog dient der Konfiguration der Einstellungen, die die Eingabe von Code erleichtern.

**Aufruf:** TwinCAT > SPS Programmierumgebung > Intelligentes Kodieren



<p>Unbekannte Variablen automatisch deklarieren (AutoDeclare)</p>	<p><input checked="" type="checkbox"/> : Der Dialog <b>Variable deklarieren</b> öffnet sich, sobald Sie einen noch nicht deklarierten Bezeichner in einem Programmiersprachen-Editor eingegeben und die Eingabezeile verlassen haben.</p> <p>Damit die AutoDeclare-Funktion auch im ST-Editor zur Verfügung steht, muss ab Build 4026 zusätzlich die Option <b>Für den ST-Editor aktivieren</b> aktiviert sein.</p>
<p>Für den ST-Editor aktivieren</p>	<p>Voraussetzung: Die Option <b>Unbekannte Variablen automatisch deklarieren (AutoDeclare)</b> ist aktiviert.</p> <p><input checked="" type="checkbox"/> : Die AutoDeclare-Funktion steht auch im ST-Editor zur Verfügung.</p> <p><input type="checkbox"/> : Im ST-Editor steht die AutoDeclare-Funktion nicht zur Verfügung. (Verfügbar ab Build 4026)</p>
<p>Alle Variablen einer Instanz in der Eingabehilfe anzeigen</p>	<p><input checked="" type="checkbox"/> : Die Funktion <b>Komponenten auflisten</b> bietet auch die lokalen Variablen einer Funktionsbausteininstanz zur Auswahl an.</p> <p><input type="checkbox"/> : Die Funktion <b>Komponenten auflisten</b> bietet nur die Eingangsvariablen und Ausgangsvariablen einer FB-Instanz zur Auswahl an.</p>
<p>Komponenten auflisten, nachdem ein Punkt (.) eingegeben wurde</p>	<p><input checked="" type="checkbox"/> : Aktiviert die Funktion <b>Komponenten auflisten</b>.</p> <p>Das bedeutet: Wenn Sie an einer Stelle, an der TwinCAT einen Bezeichner erwartet, einen Punkt eingeben, erscheint eine Auswahlliste mit Eingabemöglichkeiten.</p>
<p>Komponenten sofort nach Eingabe eines Zeichens auflisten</p>	<p>Voraussetzung: Option <b>Komponenten auflisten, nachdem ein Punkt (.) eingegeben wurde</b> ist aktiviert.</p> <p><input checked="" type="checkbox"/> : Nach Eingabe einer beliebigen Zeichenfolge erscheint eine Auswahlliste der verfügbaren Bezeichner und Operatoren</p>
<p>Mit Namensraum einfügen</p>	<p><input checked="" type="checkbox"/> : Vor dem Kennzeichner fügt TwinCAT automatisch den Namensraum mit ein.</p>
<p>Schlüsselworte automatisch in Großschreibung umwandeln (Autoformat)</p>	<p><input checked="" type="checkbox"/> : TwinCAT schreibt automatisch alle Schlüsselwörter in Großbuchstaben.</p>
<p>Querverweise automatisch bei Selektionsänderung aktualisieren</p>	<p><input checked="" type="checkbox"/> : Die Querverweisliste zeigt automatisch die Referenzen der Variablen/POUs/DUTs an, die Sie gerade selektieren oder in der der Cursor steht.</p>
<p>Fehler im Editor unterstreichen</p>	<p><input checked="" type="checkbox"/> : Fehlerhafter oder unbekannter Programmcode wird unterstrichen. (Verfügbar ab Build 4026)</p>
<p>Symbole hervorheben</p>	<p><input checked="" type="checkbox"/> : Alle Verwendungsstellen eines Symbols, auf dem der Cursor steht, werden innerhalb des Editors farblich markiert. So können Sie Querverweise innerhalb des Editors schnell erkennen. (Verfügbar ab Build 4026)</p>

**Siehe auch:**

- Dokumentation PLC: Programmiersprachen und ihre Editoren
- Dokumentation PLC: Eingabeunterstützung nutzen
- Dokumentation PLC: Verwendungsstellen mit der Querverweisliste finden

**5.9.1.10 Dialog Optionen - Ladder-Editor**

**Symbol:** 

**Funktion:** Der Dialog dient der Konfiguration der Darstellungsoptionen für den Kontaktplan-Editor.

**Aufruf: TwinCAT > SPS Programmierumgebung > Ladder-Editor****Registerkarte Allgemeines****Ansicht**

Netzwerktitel anzeigen	Der Netzwerktitel wird in der oberen linken Ecke des Netzwerks angezeigt.
Netzwerkkommentar anzeigen	Das Netzwerkkommentar wird in der oberen linken Ecke des Netzwerks angezeigt. Wenn TwinCAT zusätzlich den Netzwerktitel darstellt, erscheint der Kommentar in der Zeile unterhalb.
Bausteinsymbol anzeigen	Das Bausteinsymbol wird im Bausteinelement im Ladder-Editor angezeigt. Auch die Standardoperatoren haben Symbole.
Operandenkommentar anzeigen	TwinCAT zeigt den Kommentar an, den Sie einer Variablen im Implementierungsteil gegeben haben. Der Operandenkommentar bezieht sich im Gegensatz zum „Symbolkommentar“ nur auf die lokale Verwendungsstelle der Variablen.  Der Kommentar wird abhängig vom verfügbaren Platz automatisch umgebrochen.
Symbolkommentar anzeigen	TwinCAT zeigt den Kommentar, den Sie einer Variablen oder einem Symbol bei der Deklaration gegeben haben, oberhalb des Variablennamens an. Zusätzlich oder anstelle des Symbolkommentars können Sie auch einen lokalen „Operandenkommentar“ zuweisen.
Symboladresse anzeigen	Wenn einem Symbol (Variable) eine Adresse zugewiesen ist, wird diese Adresse oberhalb des Variablennamens angezeigt.

**Operandengröße**

Maximale Anzahl der angezeigten Zeilen	Maximale Anzahl von Zeilen des Operandennamens, die dargestellt werden.
Maximale Durchschnittszahl der Zeichen pro Zeile	Maximale Anzahl der Zeichen pro Zeile für die Darstellung des Operandennamens

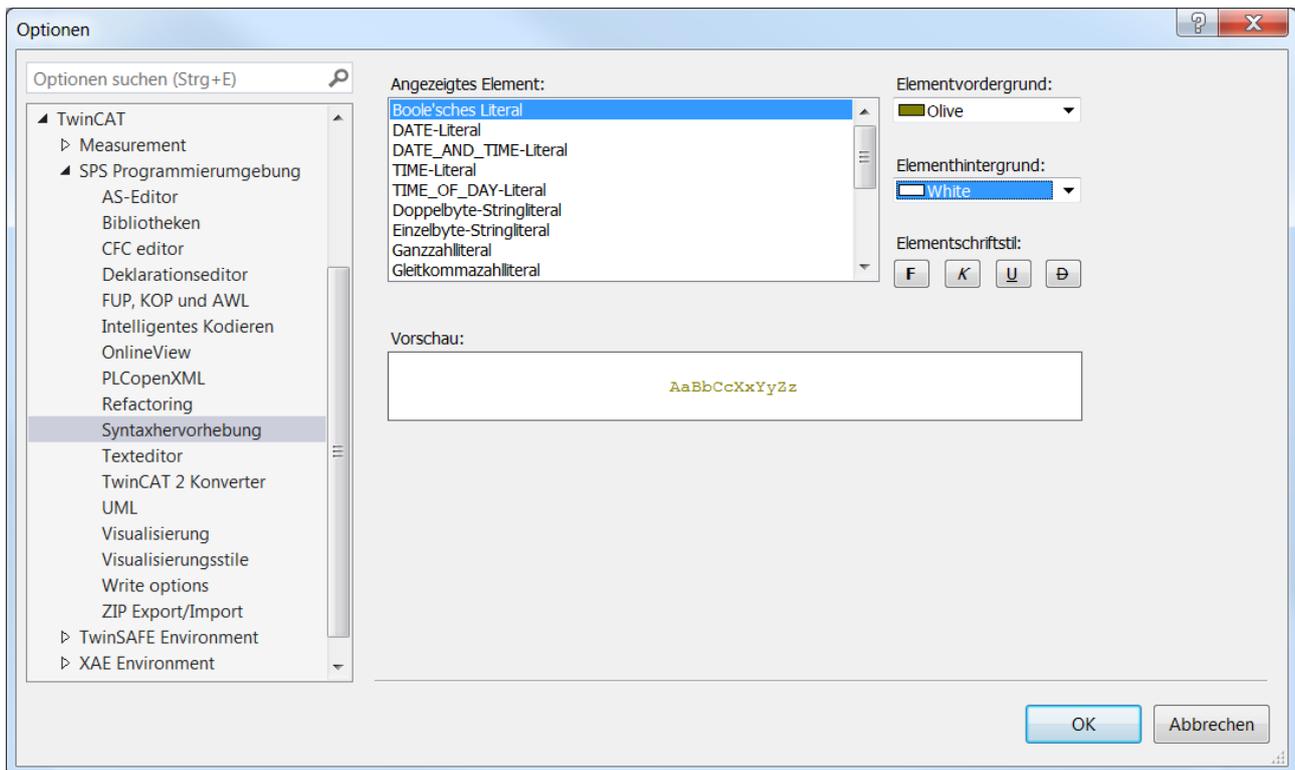
**Registerkarte Monitoring**

Angezeigte Ziffern für Gleichkommazahlen	Auswahlliste für die Anzahl der Ziffern Beispiel: 750.15 bei eingestellten 5 Ziffern wird zu 7.5e+02 bei eingestellten 2 Ziffern.
Angezeigte Länge für Zeichenfolgen	Auswahlliste für die Anzahl der Zeichen

**5.9.1.11 Dialog Optionen - Syntaxhervorhebung**

**Funktion:** Der Dialog dient der Konfiguration der Farb- und Schrifteinstellungen für die Textelemente eines Editors (zum Beispiel Operanden, Pragmas).

**Aufruf:** TwinCAT > SPS Programmierumgebung > Syntaxhervorhebung



Angezeigtes Element	Auswahlliste für Textelemente
Elementvordergrund	Vordergrundfarbe des Textelements
Elementhintergrund	Hintergrundfarbe des Textelements
Elementschriftstil	Schriftstil des Textelements (fett, kursiv, unterstrichen, durchgestrichen)
Vorschau	Der Beispieltext zeigt die aktuell eingestellten Einstellungen in einer Vorschau

### 5.9.1.12 Dialog Optionen - Texteditor

**Funktion:** Der Dialog enthält Einstellungen für die Darstellung und das Arbeiten in einem Texteditor.

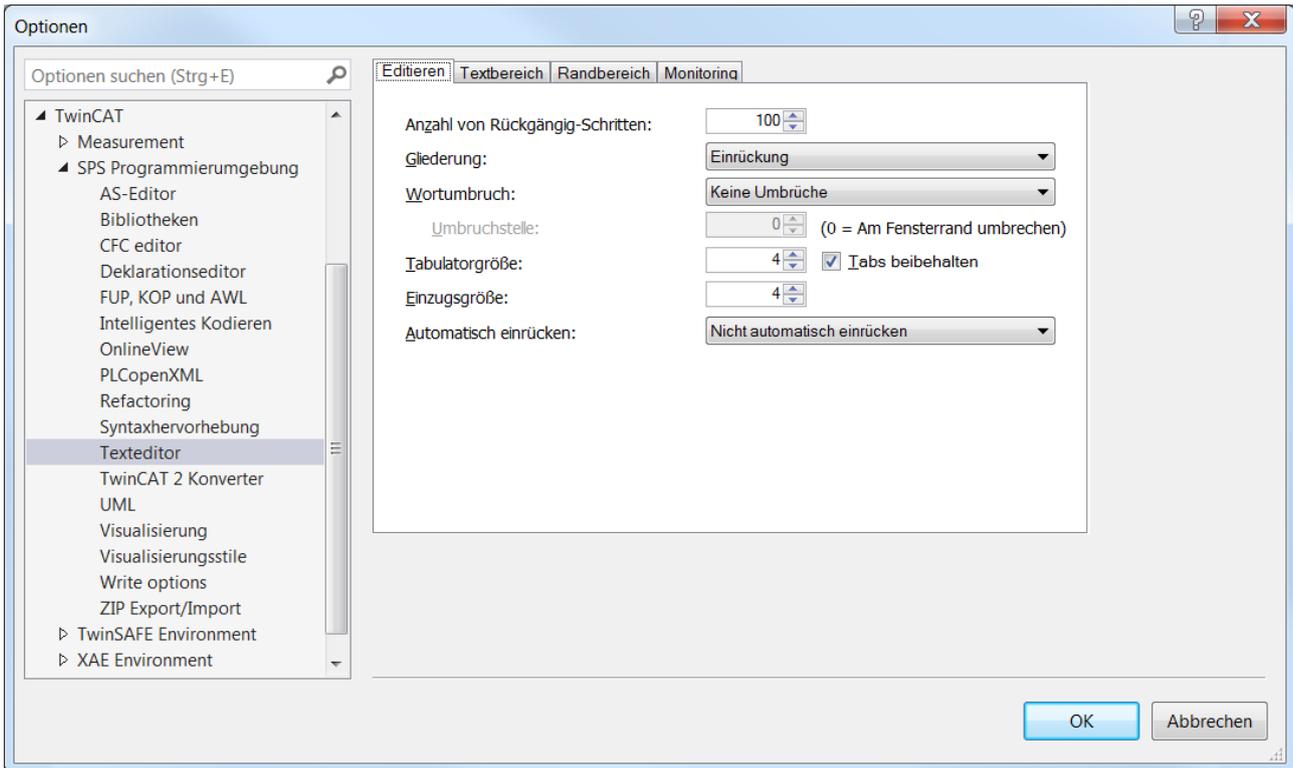
**Aufruf:** TwinCAT > SPS Programmierumgebung > Texteditor

#### Registerkarte Theme

In dieser Registerkarte stellen Sie das gewünschte Theme für die Oberflächengestaltung des ST-Editors ein. (Verfügbar ab Build 4026)

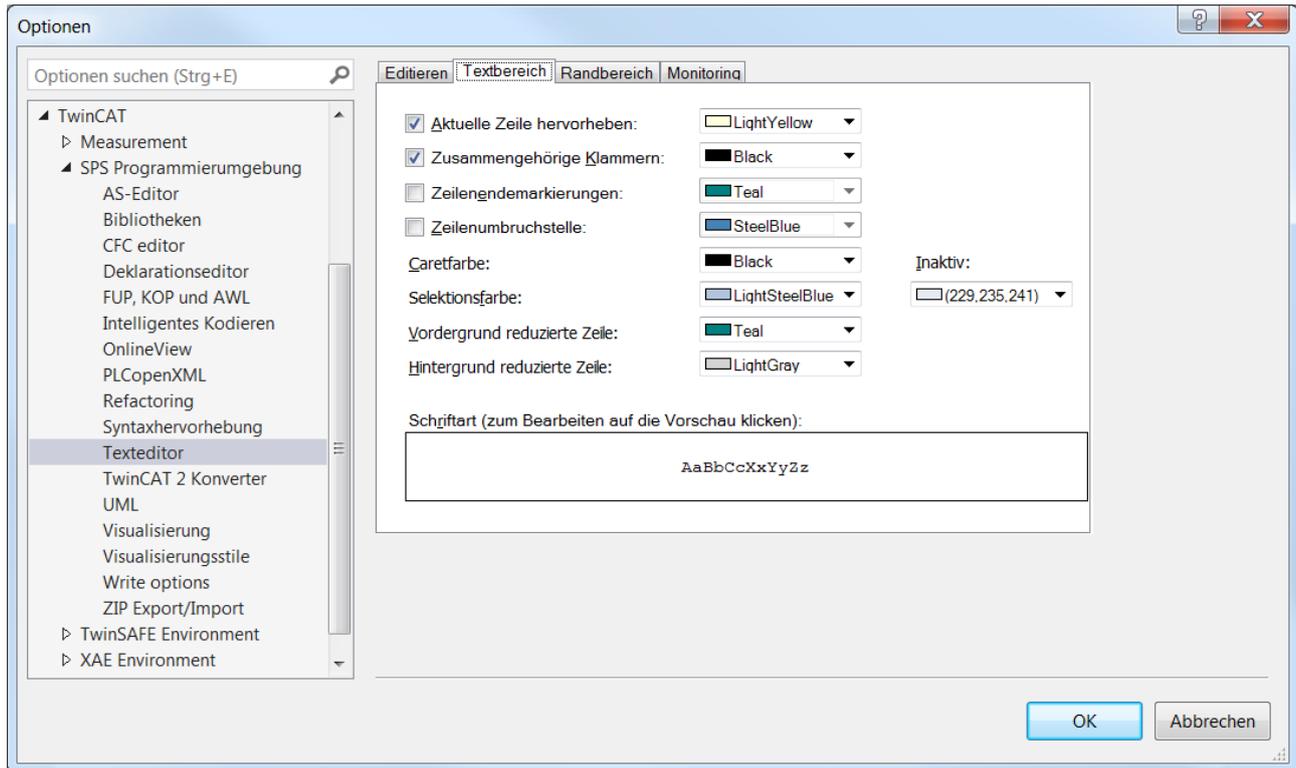
Theme	Farbschema für den Texteditor. Das gewählte Theme wird im Fenster Vorschau dargestellt.
-------	---

**Registerkarte Editieren**



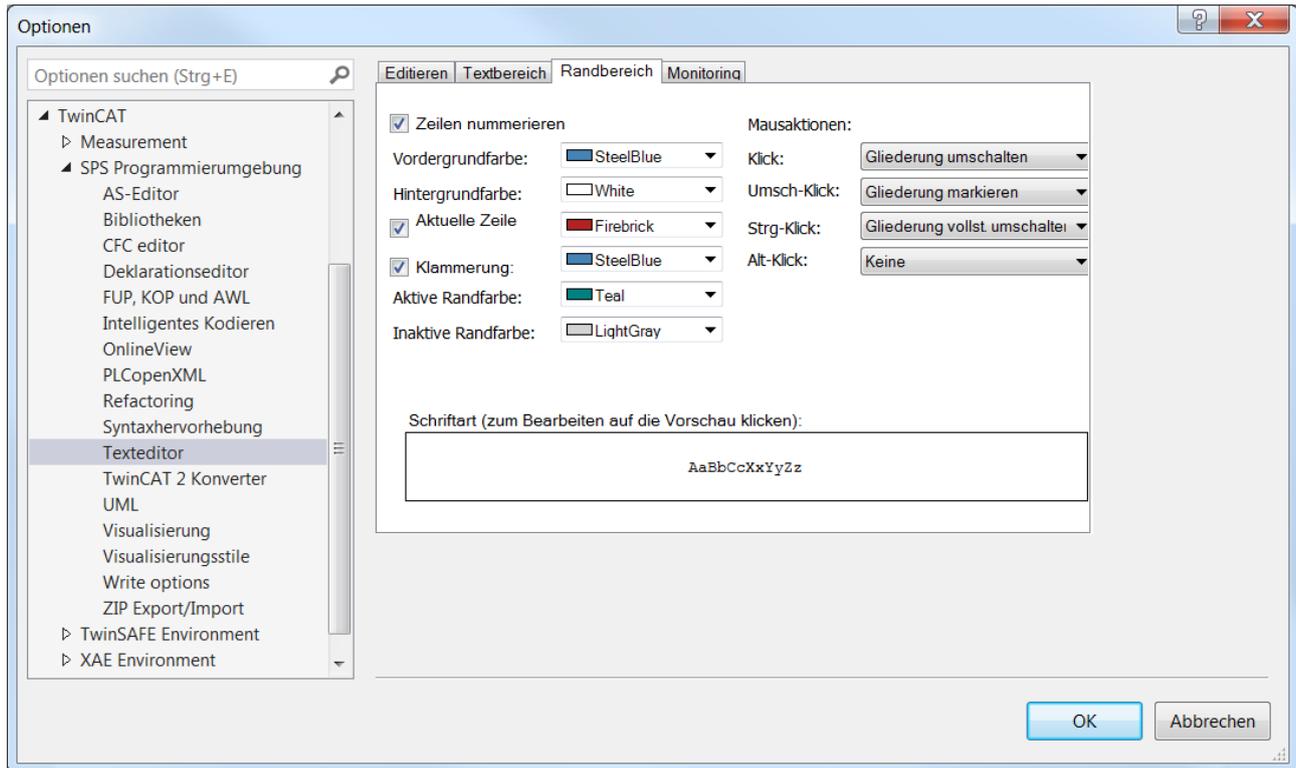
Anzahl von Rückgängig-Schritten	Maximale Anzahl der Bearbeitungsschritte, auf die Sie den Befehl <b>Bearbeiten &gt; Rückgängig</b> ausführen können.
Gliederung	<p>Definiert die Strukturierung des Codes durch Einrückungen.</p> <p>Wenn Sie eine Einrückung auswählen, können Sie den Einrückungsabschnitt mithilfe eines Plus- und Minuszeichens vor der ersten Zeile des jeweiligen Abschnitts auf- oder zuklappen.</p> <ul style="list-style-type: none"> <li>• <b>Einrückung:</b> TwinCAT fasst alle Zeilen, die gegenüber der vorausgehenden Zeile eingerückt sind, in einer Einrückungseinheit zusammen.</li> <li>• <b>Explizit:</b> Sie kennzeichnen explizit den Code-Abschnitt mit Kommentaren, der in einer Einrückungseinheit zusammengefasst werden soll: Vor dem Abschnitt muss ein Kommentar stehen, der 3 öffnende geschweifte Klammern „{{{,“ enthält, nach dem Abschnitt muss ein Kommentar folgen, der 3 schließende geschweifte Klammern „}}}\" enthält. Die Kommentare können zusätzlichen Text enthalten. Beispiel:</li> </ul> <pre> 1  IF nVar1=1 2  //comment {{{ 3      THEN 4      nVar2:=2; 5  ELSE nVar2:=10; 6  END_IF 7  //}}}} 8  nVar1:=nVar1+1;     </pre> <pre> 1  IF nVar1=1 2  //comment {{{ [5 lines] 8  nVar1:=nVar1+1;     </pre>
Wortumbruch	<ul style="list-style-type: none"> <li>• <b>Weich:</b> Der Zeilenumbruch erfolgt am Rand des Editorfensters, wenn bei <b>Umbruchsstelle</b> 0 eingetragen ist.</li> <li>• <b>Hart:</b> Der Zeilenumbruch erfolgt nach der bei Umbruchsstelle angegebenen Anzahl von Zeichen.</li> </ul>
Tabulatorgröße	Anzahl der Zeichen
Tabs beibehalten	<input checked="" type="checkbox"/> : Den Leerraum, den Sie mit der [ Tabulator ] -Taste eingefügt haben, löst TwinCAT hinterher nicht in einzelne Leerzeichen auf.
Einzugsgröße	Wenn Sie bei der Option <b>Automatisch Einrücken</b> Intelligent oder Intelligent mit Code-Komplettierung aktiviert haben, fügt TwinCAT die Anzahl Leerzeichen am Beginn der Zeile ein.
Automatisch einrücken	<ul style="list-style-type: none"> <li>• <b>Nicht automatisch einrücken</b></li> <li>• <b>Block:</b> Eine neue Zeile übernimmt automatisch die Einrückung der Vorgängerzeile.</li> <li>• <b>Intelligent:</b> Zeilen, die einer Zeile folgen, die ein Schlüsselwort enthält (zum Beispiel VAR), rücken automatisch um die angegebene Einzugsgröße ein.</li> <li>• <b>Intelligent mit Code-Komplettierung:</b> Einrückung wie bei der Option <b>Intelligent</b>, zusätzlich fügt TwinCAT das abschließende Schlüsselwort ein (zum Beispiel END_VAR).</li> </ul>

**Registerkarte Textbereich**



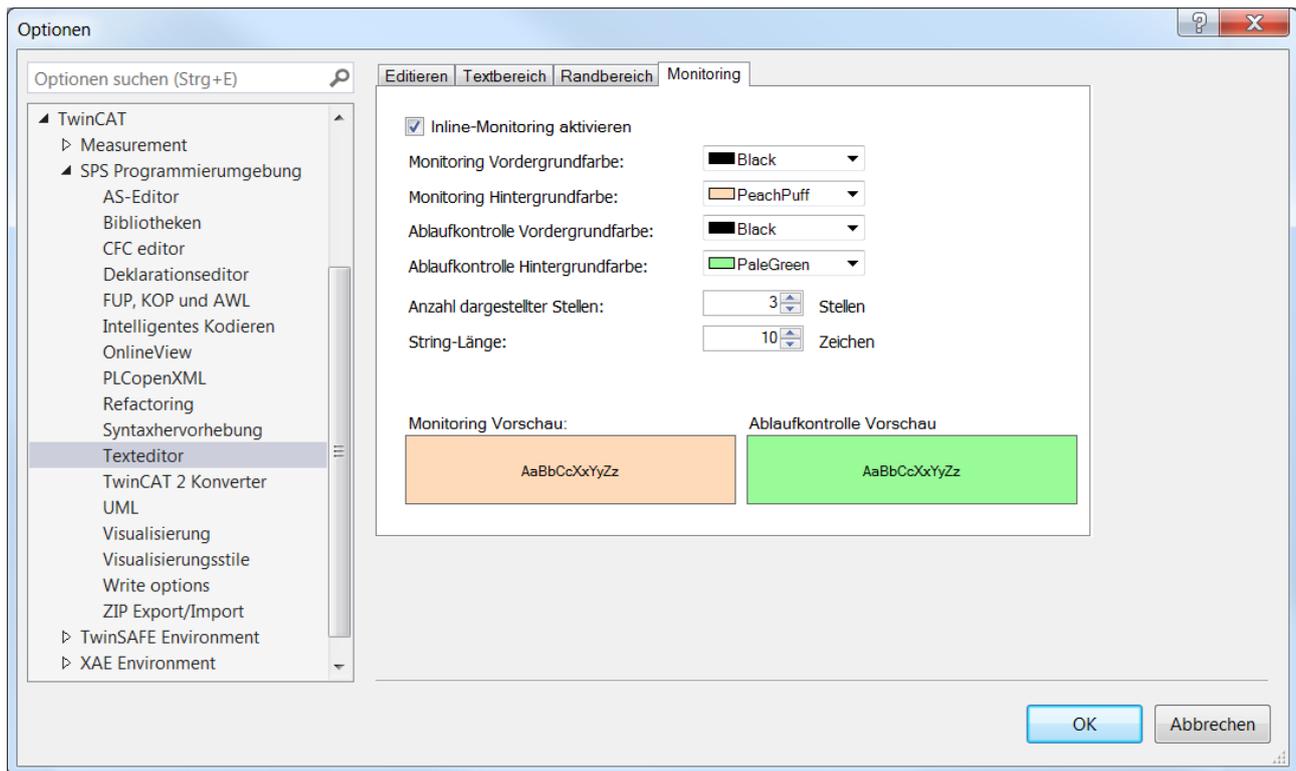
Aktuelle Zeile hervorheben	<input checked="" type="checkbox"/> : Die Zeile, in der der Cursor steht, wird farblich hinterlegt.
Zusammengehörige Klammern	<input checked="" type="checkbox"/> : Wenn der Cursor vor oder nach einer Klammer innerhalb einer Code-Zeile positioniert ist, markiert TwinCAT die zugehörige schließende oder öffnende Klammer durch einen Rahmen.
Zeilenmarkierungen	<input checked="" type="checkbox"/> : Das Ende jeder Editorzeile markiert TwinCAT durch einen kleinen Querstrich hinter dem letzten Zeichen (auch Leerzeichen) der Zeile.
Zeilenumbuchsstelle:	<input checked="" type="checkbox"/> : Wenn ein weicher oder harter Zeilenumbuch aktiviert ist, wird die definierte Zeilenumbuchsstelle durch eine senkrechte Linie angezeigt.
Caret-Farbe	Farbe des Cursorzeichens
Selektionsfarbe	Farbe des selektierten Textbereichs
Inaktiv	Farbe einer Selektierung, wenn das zugehörige Fenster nicht aktiv ist (Fokus liegt auf einem anderen Fenster).
Vordergrund reduzierte Zeile	Farbe der Kopfzeile eines geschlossenen, eingerückten Abschnitts im Code
Hintergrund reduzierte Zeile	Kopfzeile eines geschlossenen, eingerückten Abschnitts im Code wird in der Farbe hinterlegt.
Schriftart	Ein Klick auf das Feld öffnet den Standarddialog zum Konfigurieren der Schriftart.

Registerkarte Randbereich



Einstellungen für linken Randbereich des Texteditor-Fensters, der durch eine senkrechte Linie vom Eingabebereich abgetrennt ist:	
Zeilen nummerieren	<input checked="" type="checkbox"/> : Anzeige der Zeilennummern im Deklarations- und Implementierungsteil, jeweils mit 1 beginnend
Vordergrundfarbe	Farbe der Zeilennummern
Hintergrundfarbe	Farbe des Randbereichs
Aktuelle Zeile hervorheben	<input checked="" type="checkbox"/> : Die Zeilennummer der Zeile, in der der Cursor steht, wird farblich hervorgehoben.
Klammerung	<input checked="" type="checkbox"/> : Eine Klammerung umfasst die Zeilen zwischen den Schlüsselwörtern, die ein Konstrukt öffnen und abschließen, zum Beispiel IF und END_IF. Wenn die Option aktiviert ist und der Cursor vor, nach oder in einem der Schlüsselwörter eines Konstrukts steht, wird der Klammerungsbereich durch eine eckige Klammer im Randbereich angezeigt.
Aktive Randfarbe	Farbe der Trennlinie zwischen Rand- und Eingabebereich
Inaktive Randfarbe	Farbe der Trennlinie zwischen Rand- und Eingabebereich des gerade nicht aktiven Teils des Fensters
Mausaktionen	Eine der folgenden Aktionen können Sie jeder der angegebenen Mausaktionen oder Maus-Tastenkombinationen zuordnen. TwinCAT führt die Aktionen aus, wenn Sie die Mausaktion auf das Plus- oder Minuszeichen vor der Kopfzeile eines geklammerten Bereichs ausführen: <ul style="list-style-type: none"> <li>• <b>Keine:</b> Die Mausaktion löst keine Aktion aus.</li> <li>• <b>Gliederung markieren:</b> TwinCAT wählt alle Zeilen des geklammerten Bereichs aus.</li> <li>• <b>Gliederung umschalten:</b> TwinCAT öffnet oder schließt den geklammerten Bereich, oder wenn geschachtelte Klammerungen vorliegen, die erste Ebene des geklammerten Bereichs.</li> <li>• <b>Gliederung vollst. umschalten:</b> TwinCAT öffnet oder schließt alle Ebenen eines geschachtelt geklammerten Bereichs.</li> </ul>

**Registerkarte Monitoring**



Einstellungen für die Darstellung der Monitoring-Felder	
Inline-Monitoring aktivieren	<input checked="" type="checkbox"/> : Anzeige der Monitoring-Felder hinter den Variablen im Online-Modus
Monitoring Vordergrundfarbe	Darstellung des Werts im Monitoring-Feld
Monitoring Hintergrundfarbe	Darstellung des Hintergrunds im Monitoring-Feld
Ablaufkontrolle Vordergrundfarbe	Darstellung des Werts in den Monitoring-Feldern an den Ablaufkontroll-Positionen
Ablaufkontrolle Hintergrundfarbe	Darstellung des Hintergrunds der Monitoring-Felder an den Ablaufkontroll-Positionen
Anzahl dargestellter Stellen	Anzahl von Kommastellen im Monitoring-Feld
String-Länge	Maximale Länge von String-Variablenwerten im Monitoring-Feld

**Siehe auch:**

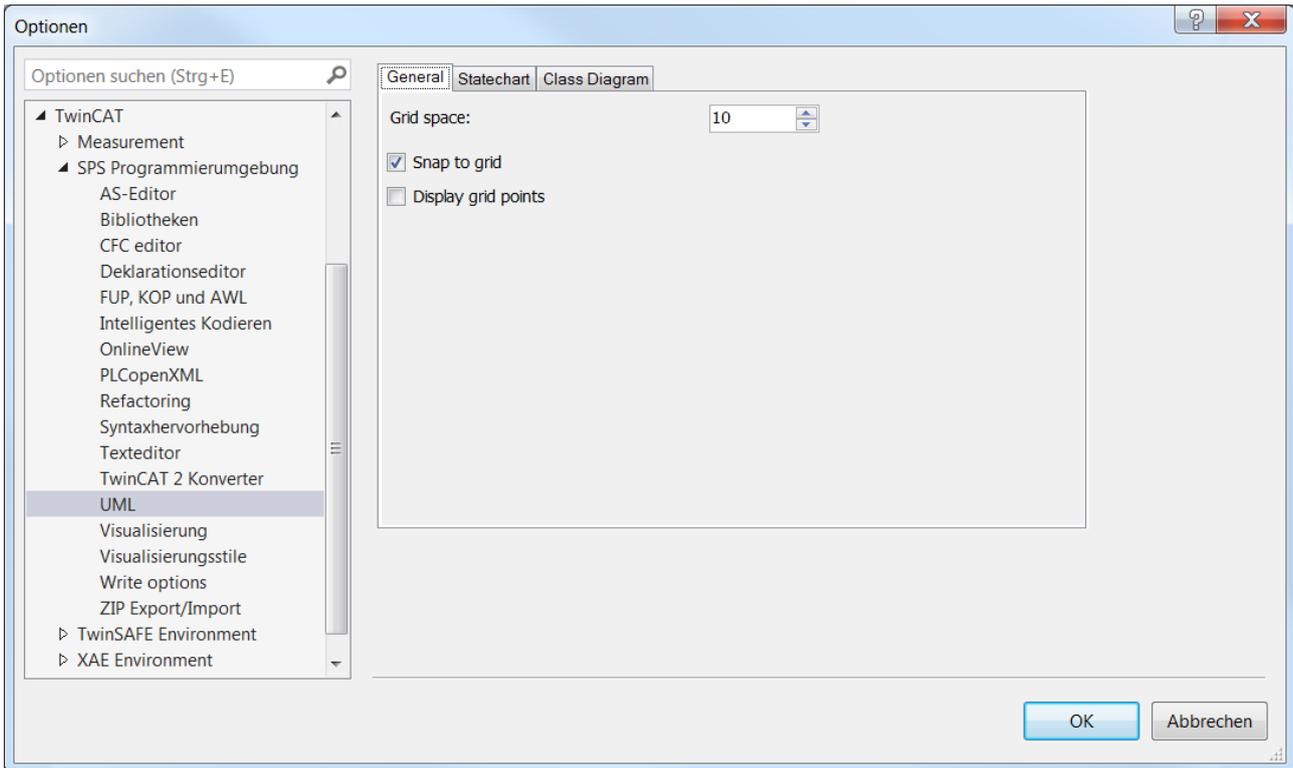
- Dokumentation PLC: Programmiersprachen und ihre Editoren

**5.9.1.13 Dialog Optionen - UML**

**Funktion:** Der Dialog dient der Konfiguration des UML-Editors.

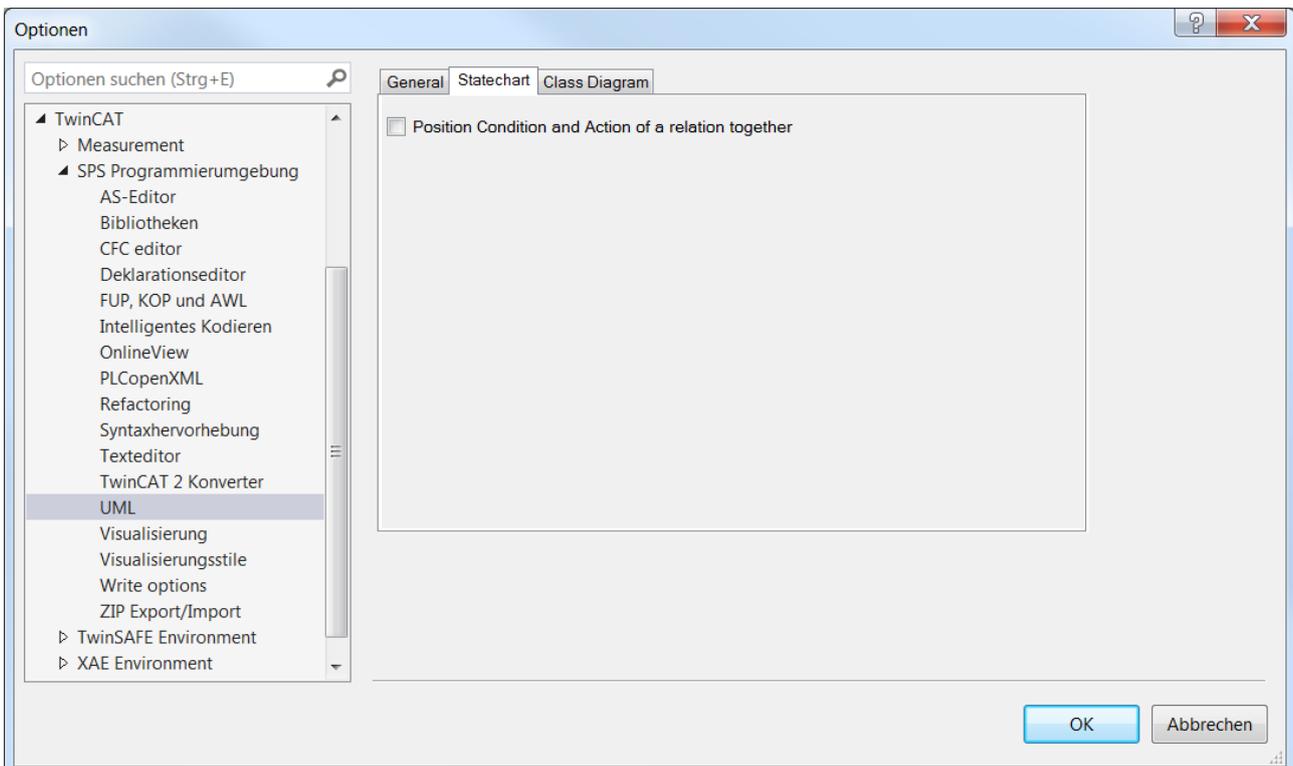
**Aufruf:** TwinCAT > SPS Programmierumgebung > UML

**Registerkarte Allgemein**



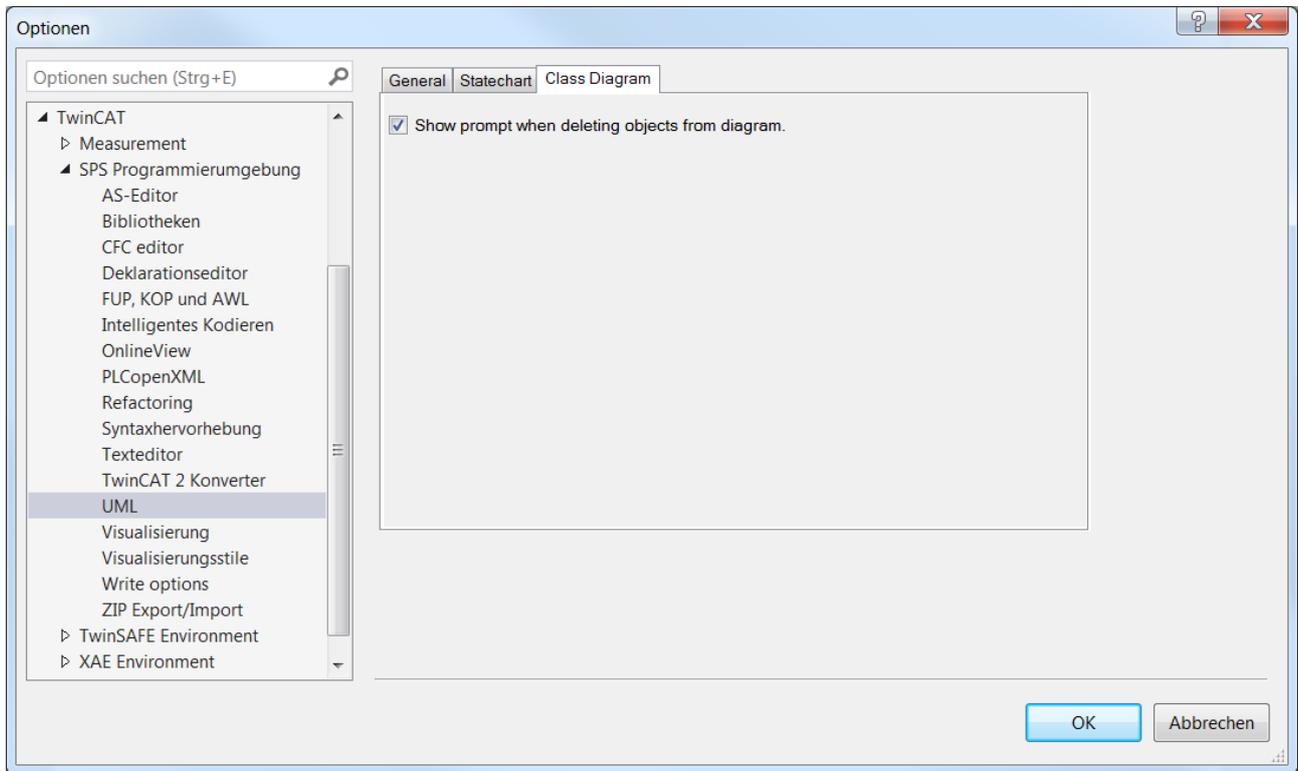
Rastermaß	Rasterlinienabstand in Pixel. Standardwert: 10
Am Raster einhängen	<input checked="" type="checkbox"/> : Alle Elemente in den UML-Editoren werden am Raster ausgerichtet.
Rasterpunkte anzeigen	<input checked="" type="checkbox"/> : Die Rasterpunkte werden in den UML-Editoren angezeigt.

**Registerkarte Zustandsdiagramm**



Bedingung und Aktion einer Beziehung gemeinsam positionieren	<input checked="" type="checkbox"/> : Im Zustandsdiagramm werden eine Wächterbedingung und eine Aktion, die zur selben Transition gehören, synchron verschoben.
--	---

**Registerkarte Klassendiagramm**



Eingabeaufforderung, wenn Objekte aus dem Diagramm gelöscht werden	Objekte können entweder nur aus dem Diagramm oder aus dem Diagramm und aus dem Projekt gelöscht werden.  <input type="checkbox"/> : Das Objekt wird standardmäßig nur aus dem Diagramm gelöscht.  <input checked="" type="checkbox"/> : Beim Löschen eines Objekts erscheint ein Auswahlfenster, um zu konfigurieren, ob das Objekt nur aus dem Diagramm oder auch aus dem Projekt gelöscht werden soll.
Refactoring-Vorschau überspringen	<input checked="" type="checkbox"/> : Wenn im Diagramm Refactoring angestoßen wird, wird die projektweite Änderung durchgeführt, ohne vorher den Dialog <b>Refactoring</b> mit einer Vorschau aller Änderungsstellen zu öffnen. Siehe auch: <ul style="list-style-type: none"> <li>• Dokumentation PLC: Refactoring</li> </ul>

**Siehe auch:**

- Dokumentation UML: Überblick

**5.9.1.14 Dialog Optionen - Visualisierung**

**Funktion:** Der Dialog dient der Konfiguration des Visualisierungseditors.

**Aufruf:** TwinCAT > SPS Programmierumgebung > Visualisierung

**Registerkarte Allgemein**



Diese Einstellungen werden ausschließlich bei der integrierten Visualisierung, nicht aber bei den Darstellungsvarianten TwinCAT PLC HMI (TargetVisu) und TwinCAT PLC HMI Web angewendet.

**Darstellungsoptionen**

Fest	<input type="radio"/> Die Visualisierung behält ihre Originalgröße.
Isotropisch	<input type="radio"/> Die Visualisierung behält ihre Proportionen.
Anisotropisch	<input type="radio"/> Die Visualisierung passt sich der Größe des Fensters im Entwicklungssystem an
Antialiased Zeichnen	<input checked="" type="checkbox"/> Die Visualisierung zeichnet sich mithilfe von Antialiasing-Methoden, sowohl während des Editierens, als auch als integrierte Visualisierung zur Laufzeit. Tipp: Wenn auf einer konkreten Visualisierungsplattform eine horizontale oder vertikale Linie unscharf gezeichnet wird, kann dies derzeit durch eine Verschiebung um 0.5px in Richtung der Liniendicke korrigiert werden; siehe Elementeigenschaft <b>Absolute Bewegung</b> , Option <b>REAL-Werte verwenden</b> . Voraussetzung: Die verwendete Plattform unterstützt die Verwendung von REAL-Koordinaten

**Bearbeitungsoptionen**

Mit Umschalten-/ Tastenvariable verknüpfen	<input checked="" type="checkbox"/> Der Platzhalter <Umschalten-/Tastenvariable> in den Visualisierungselementeigenschaften ist aktiviert. Wenn Sie ein Element, das über die Eigenschaft Farbvariablen, Farbumschlag verfügt, in die Visualisierungseditor ziehen, wird diese Eigenschaft mit dem Platzhalter <Umschalten-/Tastenvariable> konfiguriert sein. Folgende Elemente sind betroffen: Schaltfläche, Frame, Bild, Linie, Kreissektor, Polygon, Rechteck, Textfeld, Scrollbalken
--	---

**Registerkarte Raster**

**Raster**

Sichtbar	<input checked="" type="checkbox"/> Im Visualisierungseditor sind Rasterlinien im Abstand Größe sichtbar
Aktiv	<input checked="" type="checkbox"/> Im Visualisierungseditor sind Rasterlinien im Abstand Größe aktiv. Die Elemente sind am Raster ausgerichtet, ihre Positionswerte liegen jeweils auf einer Rasterlinie. Ein Element, das beim Aktivieren des Rasters bereits in einer Visualisierung ist, wird nicht automatisch ausgerichtet. Dazu müssen Sie es erst auf eine andere Position ziehen. Die Rasterlinien können aktiv und dabei unsichtbar sein
Größe	Abstand der Rasterlinien in Pixel

## Registerkarte Dateioptionen

Textdatei für textuelles "Komponenten auflisten"	<p>Dateiname und Speicherort einer Datei des Typs .csv. Sie enthält eine Tabelle mit Texten im Format einer Textliste.</p> <p>Die Einträge der Datei werden bereitgestellt, wenn die Funktion <b>Komponenten auflisten</b> als Eingabeunterstützung verwendet wird.</p> <p>Sie erzeugen diese Datei als Exportdatei der globalen Textliste mit dem Befehl Import/Export Textlisten.</p>
--	---

### Siehe auch:

- Dokumentation PLC: Visualisierung erstellen

### 5.9.1.15 Dialog Optionen - Visualisierungsstile

**Funktion:** Der Dialog dient der Konfiguration der Visualisierungsstile.

**Aufruf:** TwinCAT > SPS Programmierumgebung > Visualisierungsstile

#### Stilauswahl

Alle Versionen anzeigen (nur für Experten)	<p><input type="checkbox"/> Zur Auswahl stehen, neben dem aktuell ausgewählten Stil, alle weiteren Stile des Repositorys, aber nur in der aktuellsten Version. Wenn für den ausgewählten Stil neuere Versionen installiert sind, werden diese auch aufgelistet.</p> <p><input checked="" type="checkbox"/> Zur Auswahl stehen alle installierten Stile in allen installierten Versionen.</p>
--	--

#### Stil für neue Visualisierungsmanager

Zuletzt verwendet: <Stil, Version, Hersteller>	<p>Stil, der automatisch als ausgewählt eingestellt wird, wenn Sie eine neue Visualisierungsapplikation hinzufügen.</p> <p>Gerätebedingt ist es möglich, dass eine Darstellungsvariante trotz dieser Einstellung auf andere Art und Weise dargestellt wird.</p>
Voreinstellung: <Stil, Version, Hersteller>	Herstellerseitige Voreinstellung des Stils einstellen.
<Stil, Version, Hersteller>	Darstellungsvariante für Stil, Version und Hersteller einstellen.

### 5.9.1.16 Dialog Optionen - Write Options

#### Write Options

<p>Separate Line IDs</p>	<ul style="list-style-type: none"> <li>• TRUE: Die Line-IDs einer POU werden in einer separaten Datei (LineIDs.dbg) gespeichert, damit Änderungen in den Line-IDs nicht zu Änderungen in der POU führen, die dann im Source-Control-System als inhaltliche Änderung missinterpretiert werden würden. Voraussetzung dafür ist ab TC3.1 Build 4026, dass „Write Line IDs“ den Wert TRUE hat. (Standardeinstellung: FALSE) Bis TC3.1 Build 4024 werden Line-IDs u. a. für das Breakpoint-Handling benötigt und stellen sicher, dass eine Zuordnung der Code-Zeilen zu Maschinencode-Anweisungen möglich ist.</li> </ul>
<p>Sort by name</p>	<ul style="list-style-type: none"> <li>• TRUE (Standardeinstellung): Die Unterelemente von POU's (Aktionen, Methoden, Properties) werden nach Namen und nicht nach interner ID sortiert (siehe Beispiel [▶ 195]).</li> </ul>
<p>Write Line IDs</p>	<p>Verfügbar ab TC3.1 Build 4026</p> <ul style="list-style-type: none"> <li>• TRUE: In neuen Projekten werden für POU's Line-IDs erzeugt und gespeichert. (Standardeinstellung: FALSE)</li> </ul> <p>Bei dieser Einstellung handelt es sich um die globale Standardeinstellung. Bei der Erstellung eines neuen SPS-Projekts wird der Wert dieser Einstellung einmalig in die lokale Projekteinstellung übernommen. Diese ist in den SPS-Projekteigenschaften (Kategorie Advanced [▶ 113]) zu finden und kann dort projektbezogen angepasst werden.</p>
<p>Write PLC Bookmarks to File</p>	<p>Verfügbar ab TC3.1 Build 4026</p> <ul style="list-style-type: none"> <li>• TRUE: In neuen Projekten werden die Lesezeichen in einer separaten .bookmarks Datei im Projektverzeichnis ablegt. (Standardeinstellung: FALSE)</li> </ul> <p>Bei dieser Einstellung handelt es sich um die globale Standardeinstellung. Bei der Erstellung eines neuen SPS-Projekts wird der Wert dieser Einstellung einmalig in die lokale Projekteinstellung übernommen. Diese ist in den SPS-Projekteigenschaften (Kategorie Advanced [▶ 113]) zu finden und kann dort projektbezogen angepasst werden.</p>

**Beispiel**

Das Beispiel verdeutlicht die unterschiedliche Speicherreihenfolge der Methoden METH\_A, METH\_B und METH\_C, je nachdem, ob die Option **Sort by name** aktiviert oder deaktiviert ist. Wenn die Option deaktiviert (FALSE) ist, steht die Methode METH\_B nicht entsprechend ihres Namens an zweiter Stelle, sondern entsprechend ihrer internen ID an erster Stelle.

Sort by name = TRUE	Sort by name = FALSE

**5.9.1.17 Dialog Optionen - ZIP Export/Import**

**Funktion:** Der Dialog dient der Konfiguration der ZIP-Export- und Importeinstellungen.

**Aufruf:** TwinCAT > SPS Programmierumgebung > ZIP Export/Import

**Siehe auch:**

- Dokumentation PLC: SPS-Projekt exportieren und importieren

## 5.9.2 Befehl Anpassen

**Funktion:** Der Befehl öffnet den Dialog **Anpassen**. Der Dialog enthält Registerkarten für die Konfiguration der Benutzeroberfläche. Hier können Sie die Menüs, die Symbolleisten und die Tastaturbelegung Ihren individuellen Anforderungen anpassen.

**Aufruf:** Menü **Extras**

Sie können die TwinCAT-StandardEinstellungen jederzeit über die Schaltfläche **Zurücksetzen** wiederherstellen.

**Siehe auch:**

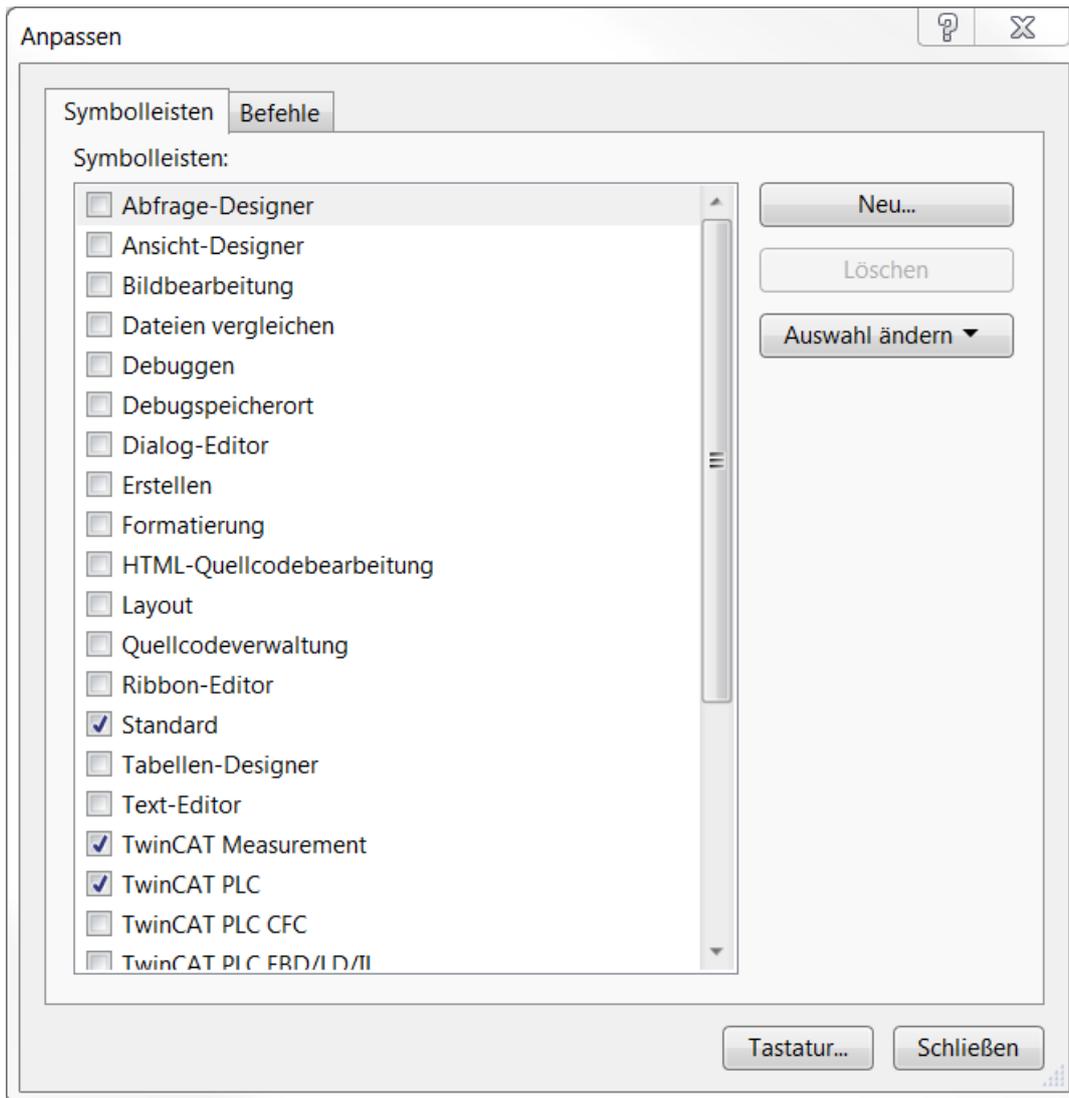
- Dokumentation TC3 User Interface > [Menüs anpassen \[▶ 36\]](#)
- Dokumentation TC3 User Interface > [Symbolleisten anpassen \[▶ 37\]](#)
- Dokumentation TC3 User Interface > [Tastaturkürzel anpassen \[▶ 38\]](#)

### 5.9.2.1 Dialog Anpassen – Registerkarte Symbolleisten

**Funktion:** Mit dem Dialog erzeugen Sie neue Symbolleisten, oder Sie passen bestehende Symbolleisten an.

**Aufruf:** Menü **Extras** > **Anpassen**

Wenn Sie den Dialog mit **Schließen** beenden, werden die Änderungen in der Menüleiste der TwinCAT-Benutzeroberfläche sichtbar.



Neu... (Symbolleiste hinzufügen)	TwinCAT fügt oberhalb der ausgewählten Symbolleiste eine Symbolleiste hinzu. Ein Dialog öffnet, indem ein Name eingeben werden kann.
Löschen (Symbolleiste entfernen)	TwinCAT entfernt die ausgewählte Symbolleiste. Sie können nur selbst erstellte Symbolleisten entfernen.
Auswahl ändern (Symbolleiste positionieren)	TwinCAT positioniert die ausgewählte Symbolleiste am oberen, unteren, linken oder rechten Rahmen des Hauptfensters
Tastatur...	Öffnet den Dialog <b>Optionen</b> , in dem Tastaturkürzel definiert werden können.

**Symbolleisten**

Darstellung der aktuell definierten Symbolleisten.	
<input type="checkbox"/> (Ausblenden)	Blendet die ausgewählte Symbolleiste auf der Benutzeroberfläche aus.
<input checked="" type="checkbox"/> (Einblenden)	Blendet die ausgewählte ausgeblendete Symbolleiste in der TwinCAT-Benutzeroberfläche ein.

**Siehe auch:**

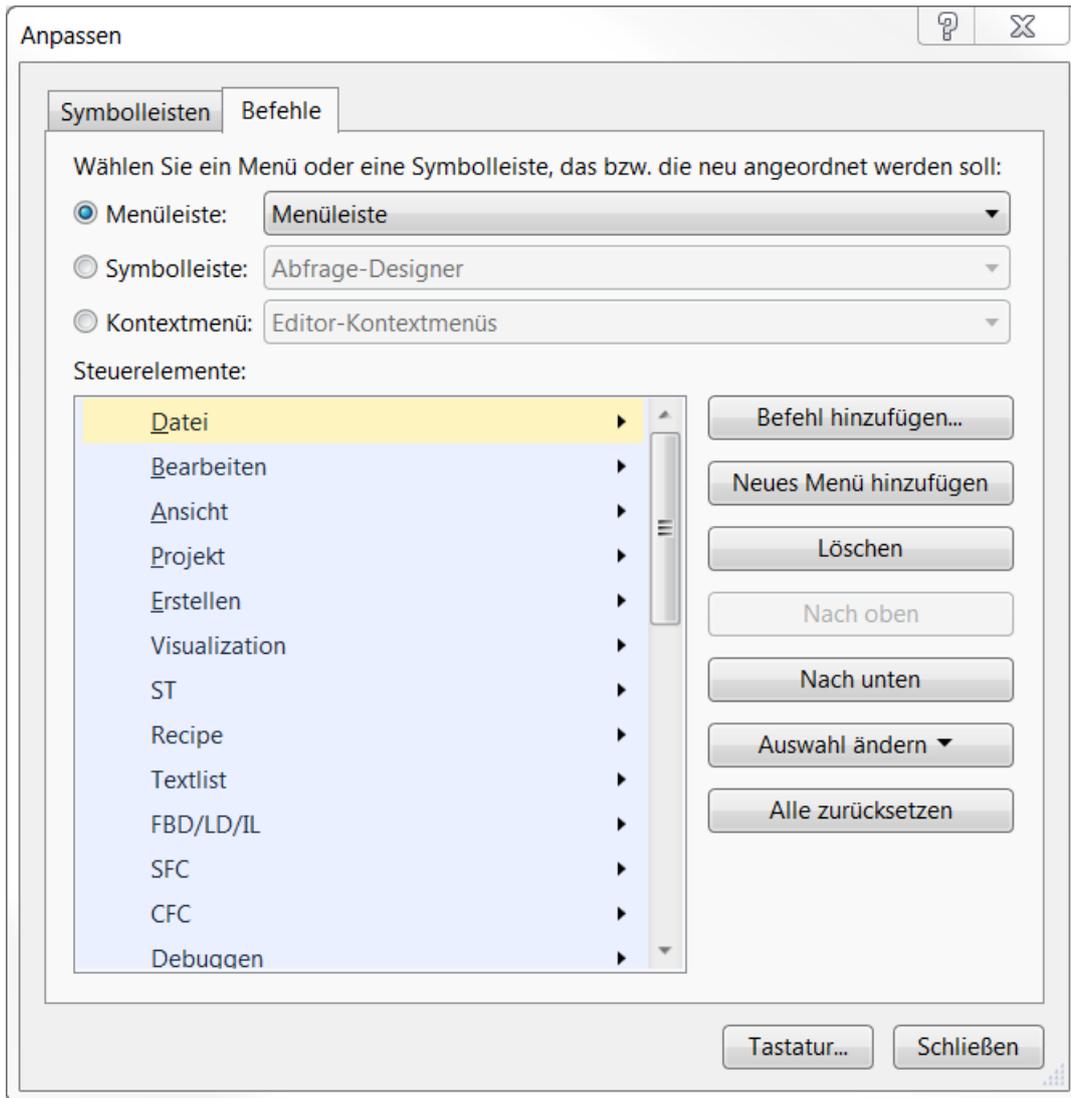
- Dokumentation TC3 User Interface > [Symbolleisten anpassen](#) [▶ 37]

**5.9.2.2 Dialog Anpassen – Registerkarte Befehle**

**Funktion:** Mit dem Dialog definieren Sie Befehle sowie die Struktur und den Inhalt der Menüs und Symbolleisten für die Benutzeroberfläche.

**Aufruf:** Menü **Extras > Anpassen**

Wenn Sie den Dialog mit **Schließen** beenden, werden die Änderungen in der Menüleiste der TwinCAT-Benutzeroberfläche sichtbar.



**Menüs und Symbolleisten**

Darstellung der aktuell definierten Symbolleisten, Menüs, Untermenüs und der enthaltenen Befehle.	
Menüleiste	Liste der Menüs und Untermenüs
Symbolleiste	Liste der Symbolleisten
Kontextmenü	Liste der Kontextmenüs

**Steuerelemente**

Steuerelemente	Auflistung der in dem ausgewählten Menü bzw. der Symbolleiste enthaltenen Befehle bzw. Untermenüs. Die Anordnung von oben nach unten entspricht der später im TwinCAT-Menü bzw. in der Symbolleiste dargestellten Anordnung.
Befehl hinzufügen	Öffnet den Dialog <b>Befehl hinzufügen</b> . Der Dialog <b>Befehl hinzufügen</b> dient zur Auswahl eines oder mehrerer Befehle. Linker Teil: Auflistung der Kategorien. Rechter Teil: Auflistung der Befehle der selektierten Kategorie. Fügt oberhalb des ausgewählten Befehls einen Befehl hinzu.
Neues Menü hinzufügen	Fügt oberhalb des ausgewählten Menüs ein neues Menü hinzu.
Auswahl ändern	Öffnet ein Menü, in dem der Name eines neu hinzugefügten Menüs bestimmt werden kann.
Löschen	Entfernt das ausgewählte Menü bzw. den ausgewählten Befehl.
Nach oben	Bewegt den ausgewählten Befehl bzw. das ausgewählte Menü in der Reihenfolge der Befehle bzw. der Menüs nach oben.
Nach unten	Bewegt den ausgewählten Befehl bzw. das ausgewählte Menü in der Reihenfolge der Befehle bzw. der Menüs nach unten.
Alle Zurücksetzen	Setzt das gesamte Menü auf die Standardeinstellungen zurück.
Tastatur	Öffnet den Dialog <b>Optionen</b> , in dem Tastaturkürzel definiert werden können.

## 5.10 Fenster

### 5.10.1 Befehl Verankerung aufheben

**Funktion:** Der Befehl löst eine Ansicht oder ein Fenster, die/das am Rahmen der Benutzeroberfläche angedockt (fixiert) ist, vom Rahmen und platziert es als unverankertes Fenster auf dem Bildschirm.

**Aufruf:** Menü **Fenster**, Kontextmenü oder Schaltfläche der Kopfleiste der Ansicht oder der Registerkarte (Fenster)

Die Ansicht kann dann auch außerhalb der Benutzeroberfläche platziert werden. Um eine unverankerte Ansicht wieder an den Rahmen der Benutzeroberfläche zu binden, verwenden Sie den Befehl **Andocken**.

**Siehe auch:**

- [Befehl Andocken \[► 199\]](#)
- Dokumentation TC3 User Interface: [Ansichten und Fenster anordnen \[► 39\]](#)

### 5.10.2 Befehl Andocken

**Funktion:** Der Befehl „dockt“ eine Ansicht, die vorher mit dem Befehl **Verankerung aufheben** gelöst wurde und nun als unverankerte Ansicht auf dem Bildschirm liegt, wieder an den Rahmen der Benutzeroberfläche „an“.

**Aufruf:** Menü **Fenster**, Kontextmenü oder Schaltfläche der Kopfleiste der Ansicht oder der Registerkarte (Fenster)

**Siehe auch:**

- [Befehl Verankerung aufheben \[► 199\]](#)
- Dokumentation TC3 User Interface: [Ansichten und Fenster anordnen \[► 39\]](#)

### 5.10.3 Befehl Ausblenden

Symbol: 

**Funktion:** Der Befehl blendet eine Ansicht aus.

**Aufruf:** Menü **Fenster**, Kontextmenü oder Schaltfläche der Kopfleiste der Ansicht

**Voraussetzung:** Eine Ansicht ist aktiviert.

Ausblenden bedeutet, dass die Ansicht geschlossen wird. Der Befehl entspricht so dem Schließen einer Ansicht über die Schaltfläche  in der Kopfleiste der Ansicht. Um eine ausgeblendete Ansicht wieder einzublenden bzw. zu öffnen, verwenden Sie die Befehle im Menü **Ansicht**.

**Siehe auch:**

- [Ansicht \[► 82\]](#)
- Dokumentation TC3 User Interface: [Ansichten aus-/einblenden \[► 40\]](#)

### 5.10.4 Befehl Alle automatisch ausblenden

**Funktion:** Der Befehl blendet alle Ansichten aus.

**Aufruf:** Menü **Fenster**, Kontextmenü oder Schaltfläche der Kopfleiste der Ansicht

Ausblenden bedeutet, dass TwinCAT alle Ansichten nur mehr als ein Reiter im Rahmen der Benutzeroberfläche anzeigt und es nur sichtbar werden, wenn Sie mit der Maus auf die Reiter klicken. Wenn Sie dann die Schaltfläche  der Kopfleiste der Ansicht anklicken oder den Befehl **Andocken** wählen, wird die Ansicht wieder auf der Benutzeroberfläche verankert.

**Siehe auch:**

- [Befehl Andocken \[► 199\]](#)
- [Ansicht \[► 82\]](#)
- Dokumentation TC3 User Interface: [Ansichten aus-/einblenden \[► 40\]](#)

### 5.10.5 Befehl Automatisch in den Hintergrund

Symbol: 

**Funktion:** Der Befehl setzt eine Ansicht in den Hintergrund.

**Aufruf:** Menü **Fenster**, Kontextmenü oder Schaltfläche der Kopfleiste der Ansicht

**Voraussetzung:** Eine Ansicht ist aktiviert.

In den Hintergrund setzen bedeutet, dass TwinCAT die Ansicht nur mehr als ein Reiter im Rahmen der Benutzeroberfläche anzeigt und sie nur sichtbar wird, wenn Sie mit der Maus auf den Reiter klicken. Wenn

Sie dann die Schaltfläche der Kopfleiste () erneut anklicken oder den Befehl **Andocken** wählen, wird die Ansicht wieder auf der Benutzeroberfläche verankert.

**Siehe auch:**

- [Befehl Andocken \[► 199\]](#)
- Dokumentation TC3 User Interface: [Ansichten aus-/einblenden \[► 40\]](#)

## 5.10.6 Befehl Registerkarte anheften

Symbol: 

**Funktion:** Der Befehl heftet die gerade aktive Registerkarte am linken Rand des Hauptfensters an.

**Aufruf:** Menü **Fenster**, Kontextmenü oder Schaltfläche der Kopfleiste der Registerkarte (Fenster)

**Voraussetzung:** Eine Registerkarte (Fenster) ist aktiviert.

**Siehe auch:**

- Dokumentation TC3 User Interface: [Ansichten aus-/einblenden \[► 40\]](#)

## 5.10.7 Befehl Neue horizontale Registerkartengruppe

Symbol: 

**Funktion:** Der Befehl verschiebt das gerade aktive Fenster in eine neue, separate Registerkartengruppe unterhalb der bereits existierenden.

**Aufruf:** Menü **Fenster**, Kontextmenü der Kopfleiste der Registerkarte (Fenster)

**Voraussetzung:** Mehrere Editorfenster sind als Registerblätter nebeneinander angeordnet.

Wenn Sie ein weiteres Objekt im Editor öffnen, wird dies automatisch in die Registerkartengruppe eingeordnet, in der der Fokus liegt.

**Siehe auch:**

- Dokumentation TC3 User Interface: [Ansichten und Fenster anordnen \[► 39\]](#)
- [Befehl Neue vertikale Registerkartengruppe \[► 201\]](#)

## 5.10.8 Befehl Neue vertikale Registerkartengruppe

Symbol: 

**Funktion:** Der Befehl verschiebt das gerade aktive Fenster in eine neue, separate Registerkarten-Gruppe rechts von der bereits existierenden.

**Aufruf:** Menü **Fenster**, Kontextmenü der Kopfleiste der Registerkarte (Fenster)

**Voraussetzung:** Mehrere Editorfenster sind als Registerblätter nebeneinander angeordnet.

Wenn Sie ein weiteres Objekt im Editor öffnen, wird dies automatisch in die Registerkartengruppe eingeordnet, in der der Fokus liegt.

**Siehe auch:**

- Dokumentation TC3 User Interface: [Ansichten und Fenster anordnen \[► 39\]](#)
- [Befehl Neue horizontale Registerkartengruppe \[► 201\]](#)

## 5.10.9 Befehl Fenster-Layout zurücksetzen

**Funktion:** Der Befehl setzt alle gerade geöffneten Fenster und Ansichten auf ihre Standardpositionen zurück. Sie müssen den Befehl vor der Ausführung bestätigen.

**Aufruf:** Menü **Fenster**

## 5.10.10 Befehl Alle Dokumente schließen

Symbol: 

**Funktion:** Der Befehl schließt alle gerade geöffneten Editorfenster.

**Aufruf:** Menü **Fenster**

**Voraussetzung:** Mindestens ein Editorfenster ist geöffnet.

**Siehe auch:**

- Dokumentation TC3 User Interface: [Ansichten aus-/einblenden \[► 40\]](#)

## 5.10.11 Befehl Fenster

**Funktion:** Der Befehl öffnet den Dialog **Fenster**, der alle geöffneten Objekte zeigt. Sie können darin Fenster aktivieren oder schließen.

**Aufruf:** Menü **Fenster**

## 5.10.12 Befehle des Untermenüs Fenster

**Funktion:** Der Befehl aktiviert das ausgewählte Fenster.

**Aufruf:** Menü **Fenster**

Für jedes geöffnete Editorfenster enthält das Menü **Fenster** einen Befehl **<n><Objektname>**, über den Sie das Fenster aktivieren, also den Fokus dorthin setzen. Im Offlinebetrieb ergänzt TwinCAT hinter dem Befehl die Erweiterung (Offline). Bei Funktionsbausteinen wird zur Unterscheidung von Implementierung und Instanz die Erweiterung (Impl) oder **<Instanzpfad>** hinzugefügt.

**Siehe auch:**

- [Befehl Fenster \[► 202\]](#)

# 5.11 AS

## 5.11.1 Befehl Initialschritt

Symbol: 

**Funktion:** Der Befehl wandelt den gerade selektierten Schritt in einen Initialschritt um.

**Aufruf:** Menü **AS**, Kontextmenü

Wenn Sie den Befehl ausführen, verändert sich der Rahmen des Schritt-Elements in eine Doppellinie. Der Schritt, der zuvor Initialschritt war, wird automatisch zu einem „normalen“ Schritt und wird mit einfachem Rahmen dargestellt.

Sie können die Eigenschaft **Initialschritt** auch in der Ansicht **Eigenschaften** eines Schritt-Elementes aktivieren oder deaktivieren, dabei passt TwinCAT jedoch die Einstellungen der anderen Schritte nicht automatisch an.

Der Befehl kann nützlich sein, wenn Sie das Diagramm umstellen wollen. Wenn Sie ein neues AS-Objekt anlegen, enthält es automatisch einen Initialschritt, gefolgt von einer Transition (TRUE) und einem Sprung zurück zum Initialschritt.



Beachten Sie die Möglichkeit, im Onlinebetrieb das Diagramm mithilfe der AS-Flags SFCInit bzw. SFCReset auf den Initialschritt zurückzusetzen.

**Siehe auch:**

- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor
- Dokumentation PLC: AS-Elementeigenschaften

## 5.11.2 Befehl Schritt-Transition einfügen

Symbol:

**Funktion:** Der Befehl fügt einen Schritt und eine Transition vor der gerade selektierten Position ein.

**Aufruf:** Menü **AS**, Kontextmenü

Wenn Sie einen Schritt selektiert haben, fügt TwinCAT eine neue Schritt-Transition-Kombination ein. Wenn Sie eine Transition selektiert haben, wird eine neue Transition-Schritt-Kombination eingefügt.

Der neue Schritt wird per Default Step<n> genannt. n ist eine fortlaufende Zahl, beginnend mit 0 für den ersten Schritt, der zusätzlich zum Initialschritt eingefügt wird. Die neue Transition wird entsprechend per Default Trans<n> genannt. Sie können die Default-Namen direkt durch Mausklick auf den Namen editieren.

**Siehe auch:**

- [Befehl Schritt-Transition danach einfügen \[► 203\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor
- Dokumentation PLC: AS-Elemente Schritt und Transition

## 5.11.3 Befehl Schritt-Transition danach einfügen

Symbol:

**Funktion:** Der Befehl fügt einen Schritt und eine Transition nach der gerade angewählten Position ein.

**Aufruf:** Menü **AS**, Kontextmenü

Wenn Sie einen Schritt selektiert haben, fügt TwinCAT eine neue Transition-Schritt-Kombination ein. Wenn Sie eine Transition selektiert haben, wird eine neue Schritt-Transition-Kombination eingefügt.

Der neue Schritt wird standardmäßig Step<n> genannt. n ist eine fortlaufende Zahl, beginnend mit 0 für den ersten Schritt, der zusätzlich zum Initialschritt eingefügt wird. Die neue Transition wird entsprechend per Default Trans<n> genannt. Sie können die Default-Namen direkt nach Mausklick auf den Namen editieren.

**Siehe auch:**

- [Befehl Schritt-Transition einfügen \[► 203\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor
- Dokumentation PLC: AS-Elemente 'Schritt' und 'Transition'

## 5.11.4 Befehl Parallel

Symbol:

**Funktion:** Der Befehl wandelt die gerade selektierte alternative Verzweigung in eine parallele Verzweigung um.

**Aufruf:** Menü **AS**, Kontextmenü

**Voraussetzung:** Die horizontale Verbindungslinie einer Verzweigung ist selektiert.

Beachten Sie, dass Sie nach der Umwandlung einer Verzweigung die Anordnung der Schritte und Transitionen vor und nach der Verzweigung prüfen und anpassen müssen.

**Siehe auch:**

- [Befehl Alternativ \[► 204\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor

### 5.11.5 Befehl Alternativ

Symbol: 

**Funktion:** Der Befehl wandelt die gerade selektierte parallele Verzweigung in eine alternative Verzweigung um.

**Aufruf:** Menü **AS**, Kontextmenü

**Voraussetzung:** Die horizontale Verbindungslinie einer Verzweigung ist selektiert.

Beachten Sie, dass Sie nach der Umwandlung einer Verzweigung die Anordnung der Schritte und Transitionen vor und nach der Verzweigung prüfen und anpassen müssen.

**Siehe auch:**

- [Befehl Parallel \[► 203\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor

### 5.11.6 Befehl Verzweigung einfügen

Symbol: 

**Funktion:** Der Befehl fügt eine Verzweigung links von der gerade angewählten Position ein.

**Aufruf:** Menü **AS**, Kontextmenü

Das Verhalten des Befehls entspricht dem Befehl **Verzweigung rechts einfügen**.

**Siehe auch:**

- [Befehl Verzweigung rechts einfügen \[► 204\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor
- Dokumentation PLC: AS-Element Verzweigung

### 5.11.7 Befehl Verzweigung rechts einfügen

Symbol: 

**Funktion:** Der Befehl fügt eine Verzweigung rechts von der gerade angewählten Position ein.

**Aufruf:** Menü **AS**, Kontextmenü

Der Typ der eingefügten Verzweigung hängt von dem ausgewählten Element ab:

- Wenn das oberste Element der gerade ausgewählten Elemente eine Transition oder eine alternative Verzweigung ist, fügt TwinCAT eine alternative Verzweigung ein.
- Wenn das oberste Element der gerade ausgewählten Elemente ein Schritt, ein Makro, ein Sprung oder eine parallele Verzweigung ist, fügt TwinCAT eine parallele Verzweigung mit Sprungmarke Branch<x> ein, wobei x eine fortlaufende Zahl ist. Sie können diesen Default-Name der Sprungmarke editieren. Sie können die Sprungmarke als Ziel eines Sprungs angeben.
- Wenn gerade ein gemeinsames Element einer bestehenden Verzweigung selektiert ist (horizontale Linie), fügt TwinCAT die neue Verzweigung ganz rechts als weiteren Zweig hinzu. Wenn gerade ein gesamter Zweig einer bestehenden Verzweigung selektiert ist, fügt TwinCAT die neue Verzweigung direkt rechts davon als neuen Zweig hinzu.



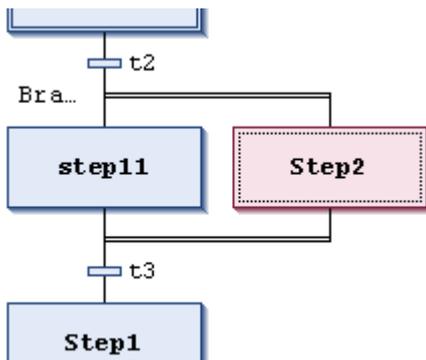
Beachten Sie, dass Sie eine Verzweigung mit den Befehlen **Alternativ** bzw. **Parallel** in den jeweils anderen Typ umwandeln können.

### Beispiel: Parallele Verzweigung

In der folgenden Abbildung sehen Sie eine neu eingefügte parallele Verzweigung, erzeugt mit Befehl Verzweigung rechts einfügen, während step11 ausgewählt war. TwinCAT fügt automatisch einen Schritt (Step2 im Beispiel) ein.

Abarbeitung im Onlinebetrieb: Wenn t2 TRUE liefert, führt TwinCAT Step2 sofort nach step11 aus, bevor t3 ausgewertet wird.

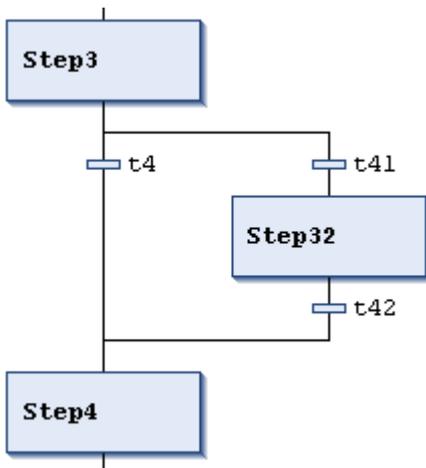
TwinCAT führt also im Gegensatz zu alternativen Verzweigungen beide Zweige aus.



### Beispiel: Alternative Verzweigung

In der folgenden Abbildung sehen Sie eine neu eingefügte alternative Verzweigung, erzeugt mit Befehl Verzweigung rechts einfügen während Transition t4 ausgewählt war. TwinCAT fügt automatisch einen Schritt (Step32 im Beispiel), eine vorausgehende und eine nachfolgende Transition (t41, t42) ein.

Abarbeitung im Onlinebetrieb: Wenn Step3 aktiv ist, wertet TwinCAT die nachfolgenden Transitionen (t4, t41) von links nach rechts aus. Der erste Zweig der Verzweigung, in dem die erste Transition TRUE liefert, wird ausgeführt. Somit wird im Gegensatz zu einer parallelen Verzweigung nur ein Zweig ausgeführt.

**Siehe auch:**

- [Befehl Alternativ \[► 204\]](#)
- [Befehl Parallel \[► 203\]](#)
- [Befehl Verzweigung einfügen \[► 204\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor
- Dokumentation PLC: AS-Element Verzweigung

## 5.11.8 Befehl Aktionsassoziation einfügen

Symbol:

**Funktion:** Der Befehl weist einem Schritt eine IEC-Aktion zu.

**Aufruf:** Menü **AS**, Kontextmenü

**Voraussetzung:** Ein Schritt ist selektiert.

TwinCAT fügt das Aktion-Element rechts neben dem gerade selektierten Schritt-Element ein.

Wenn Sie dem Schritt bereits eine oder mehrere Aktionen zugewiesen haben, werden diese in einer „Aktionsliste“ untereinander dargestellt. Die neue Aktion wird dann folgendermaßen platziert:

- Wenn Sie das Schritt-Element selektiert haben, als erste Aktion des Schrittes, d. h. an oberster Position der Aktionsliste.
- Wenn Sie eine der vorhandenen Aktionen in der Aktionsliste des Schrittes selektiert haben, direkt vor, d. h. oberhalb der Aktion.

Der linke Teil des Aktion-Elements enthält den Qualifizierer, standardmäßig N, im rechten Teil geben Sie den Aktionsnamen ein. Dazu klicken Sie in die Box, um einen Editerrahmen zu erhalten. Sie müssen diese Aktion vorher im Projekt als POU angelegt haben.

Sie können den Qualifizierer ebenfalls editieren. Eine Liste der gültigen Qualifizierer ist im Kapitel „Qualifizierer für Aktionen in AS“ beschrieben.

**Siehe auch:**

- [Befehl Aktionsassoziation danach einfügen \[► 207\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor
- Dokumentation PLC: Qualifizierer für Aktionen in AS

## 5.11.9 Befehl Aktionsassoziation danach einfügen

Symbol: 

**Funktion:** Der Befehl weist einem Schritt eine IEC-Aktion zu.

**Aufruf:** Menü **AS**, Kontextmenü

**Voraussetzung:** Ein Schritt ist selektiert.

Der Befehl entspricht der Beschreibung von Aktionsassoziation einfügen. Der Unterschied besteht darin, dass TwinCAT die neue Aktion nicht an der ersten, sondern an der letzten Position der Aktionenliste platziert. Wenn Sie in der Aktionsliste eine Aktion selektiert haben, platziert TwinCAT die neue Aktion nicht oberhalb, sondern unterhalb.

**Siehe auch:**

- [Befehl Aktionsassoziation einfügen \[► 206\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor
- Dokumentation PLC: Qualifizierer für Aktionen in AS

## 5.11.10 Befehl Sprung einfügen

Symbol: 

**Funktion:** Der Befehl fügt ein Sprung-Element vor dem gerade selektierten Element ein.

**Aufruf:** Menü **AS**, Kontextmenü

**Voraussetzung:** Ein Schritt ist selektiert.

TwinCAT fügt den Sprung automatisch mit Sprungziel Step ein. Sie müssen dieses Sprungziel anschließend noch durch ein reales Sprungziel ersetzen. Sie können das Ziel mit der Eingabehilfe auswählen.

**Siehe auch:**

- [Befehl Sprung danach einfügen \[► 207\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor
- Dokumentation PLC: AS-Element Sprung

## 5.11.11 Befehl Sprung danach einfügen

Symbol: 

**Funktion:** Der Befehl fügt ein Sprung-Element nach dem gerade selektierten Element ein.

**Aufruf:** Menü **AS**

TwinCAT fügt den Sprung automatisch mit Sprungziel Step ein. Sie müssen dieses Sprungziel anschließend noch durch ein reales Sprungziel ersetzen. Sie können das Ziel mit der Eingabehilfe auswählen.

**Siehe auch:**

- [Befehl Sprung einfügen \[► 207\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor
- Dokumentation PLC: AS-Element Sprung

## 5.11.12 Befehl Makro einfügen

Symbol: 

**Funktion:** Der Befehl fügt ein Makro-Element vor dem gerade selektierten Element ein.

**Aufruf:** Menü **AS**, Kontextmenü

Das neue Makro wird standardmäßig Macro<x> genannt. x ist eine fortlaufende Zahl, beginnend mit 0 für das erste Makro. Sie können den Default-Name direkt durch Mausklick auf den Namen editieren.

Um das Makro zu bearbeiten, öffnen Sie es mit dem Befehl **Makro anzeigen** im Makro-Editor.

**Siehe auch:**

- [Befehl Makro anzeigen \[► 208\]](#)
- [Befehl Makro danach einfügen \[► 208\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- AS-Editor

## 5.11.13 Befehl Makro danach einfügen

Symbol: 

**Funktion:** Der Befehl fügt ein Makro-Element nach dem gerade selektierten Element ein.

**Aufruf:** Menü **AS**, Kontextmenü

Der Befehl entspricht der Beschreibung von dem Befehl **Makro einfügen**.

**Siehe auch:**

- [Befehl Makro einfügen \[► 208\]](#)
- [Befehl Makro anzeigen \[► 208\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor

## 5.11.14 Befehl Makro anzeigen

Symbol: 

**Funktion:** Der Befehl öffnet ein Makro im Makro-Editor zur Bearbeitung.

**Aufruf:** Menü **AS**

**Voraussetzung:** Ein Makro ist selektiert.

Mit dem Befehl schließt TwinCAT die Hauptansicht des AS-Editors und öffnet stattdessen den Makro-Editor. Es handelt sich ebenfalls um einen AS-Editor, in dem Sie nun den Teil des AS-Diagramms bearbeiten können, der in der Hauptansicht als Makro-Box dargestellt wird.

Mit dem Befehl **Makro verlassen** kehren Sie zur Hauptansicht zurück.

**Siehe auch:**

- [Befehl Makro verlassen \[► 209\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor

### 5.11.15 Befehl Makro verlassen

Symbol: 

**Funktion:** Der Befehl schließt den Makro-Editor und kehrt zur Hauptansicht des AS-Editors zurück.

**Aufruf:** Menü **AS**

**Voraussetzung:** Ein Makro ist im Makro-Editor geöffnet.

**Siehe auch:**

- [Befehl Makro anzeigen \[► 208\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor

### 5.11.16 Befehl Einfügen danach

Symbol: 

**Funktion:** Der Befehl fügt die Elemente aus der Zwischenablage nach der gerade angewählten Position ein.

**Aufruf:** Menü **AS**

### 5.11.17 Befehl Eingangsaktion hinzufügen

Symbol: 

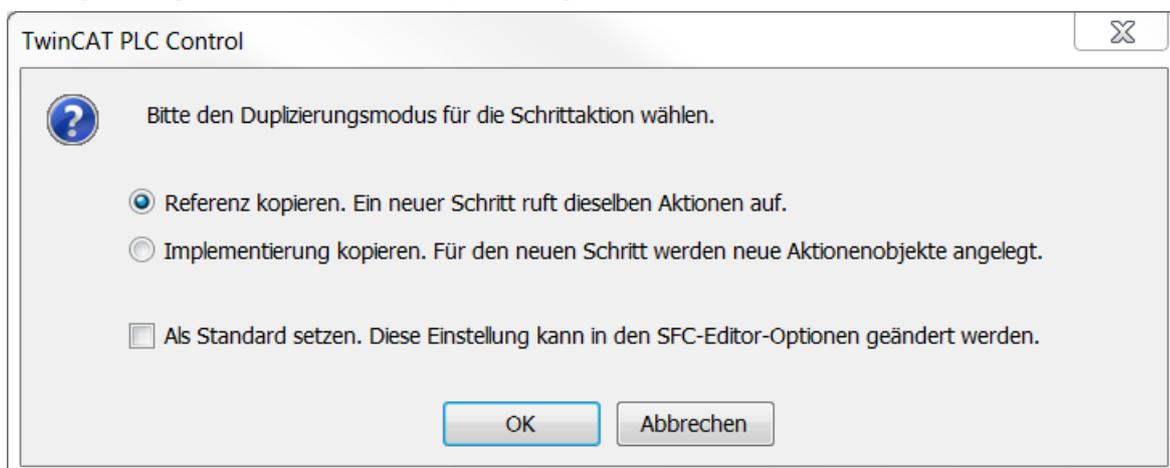
**Funktion:** Der Befehl führt zum Dialog **Eingangsaktion hinzufügen**, in dem Sie eine neue Schrittaktion des Typs „Eingangsaktion“ definieren. Abhängig von den AS-Optionen erscheint gegebenenfalls vorher noch eine Eingabeaufforderung zum Wählen des Duplizierungsmodus für die neue Schrittaktion.

**Aufruf:** Menü **AS**, Kontextmenü des ausgewählten Schrittelements

**Voraussetzung:** Ein Schrittelement ist ausgewählt.

Die Eingangsaktion wird automatisch im ST-Editor geöffnet. Das Schrittelement erhält ein „E“ in der linken unteren Ecke.

#### Abfragedialog zum Wählen des Duplizierungsmodus



Referenz kopieren. Ein neuer Schritt wird dieselben Aktionen aufrufen	Wenn der Schritt im AS kopiert wird, wird die Verknüpfung zu der/den Schritttaktionen mit kopiert. Die voneinander kopierten Schritte werden also alle dieselben Aktionen aufrufen.
Implementierung kopieren. Für den neuen Schritt werden neue Aktionsobjekte angelegt	Dies bedeutet eine „Einbettung“ der Schritttaktionen für die kopierten Schritte. Die neu erzeugten Aktionsobjekte erscheinen standardmäßig unter dem AS-Baustein im SPS-Projektbaum im Projektmappen-Explorer. Diese Objekte enthalten zunächst eine Kopie des ursprünglichen Implementierungscode der jeweiligen Aktion.
Als Standard setzen. Diese Einstellung kann in den SFC-Editor-Optionen geändert werden.	Die Einstellungen im Dialog werden als Standardeinstellung übernommen. Sie können die Standardeinstellung in den TwinCAT-Optionen in der Kategorie <b>AS-Editor</b> ändern. Wählen Sie dazu im Gruppenfeld <b>Schritttaktionen</b> in der Drop-down-Liste <b>Standardeinfügemethode</b> den Eintrag <b>Immer fragen, Referenz kopieren</b> oder <b>Implementierung duplizieren</b> .

**Siehe auch:**

- Befehl Optionen > [Dialog Optionen - AS-Editor \[► 177\]](#)
- Dokumentation PLC: [Befehl Ausgangsaktion hinzufügen \[► 210\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor
- Dokumentation PLC: Programmieren in Ablaufsprache (AS)
- Dokumentation PLC: AS-Element Aktion

**5.11.18 Befehl Ausgangsaktion hinzufügen**Symbol: 

**Funktion:** Der Befehl führt zum Dialog **Ausgangsaktion hinzufügen**, in dem Sie eine neue Schritttaktion des Typs Eingangsaktion definieren. Abhängig von den AS-Optionen erscheint gegebenenfalls vorher noch eine Eingabeaufforderung zum Wählen des Duplizierungsmodus für die neue Schritttaktion. Sehen Sie hierzu die Hilfeseite zum Befehl **Eingangsaktion hinzufügen**.

**Aufruf:** Menü **AS**, Kontextmenü des ausgewählten Schrittelements

**Voraussetzung:** Ein Schrittelement im AS ist ausgewählt.

**Siehe auch:**

- [Befehl Eingangsaktion hinzufügen \[► 209\]](#)
- Befehl Optionen > [Dialog Optionen - AS-Editor \[► 177\]](#)
- Dokumentation PLC: Ablaufsprache (AS)
- Dokumentation PLC: Programmieren in Ablaufsprache (AS)
- Dokumentation PLC: AS-Editor
- Dokumentation PLC: AS-Element Aktion

### 5.11.19 Befehl Change duplication - Setzen

**Funktion:** Der Befehl verknüpft jede Schrittaktion oder Transition, die von einem Schritt oder einer Transition im AS-Baustein aufgerufen wird, fest mit dem Aufrufer. Somit kann das Aktions- oder Transitionsobjekt nur noch von genau diesem einen Aufrufer aufgerufen werden (Pseudoeinbettung). Dies hat zur Folge, dass das Kopieren von Schritt- und Transitionselementen, die Aktionen oder Transitionen aufrufen, automatisch neue Aktions- oder Transitionsobjekte erzeugt. Der Implementierungscode wird jeweils kopiert.

**Aufruf:** Menü **AS**

Sehen Sie für Details zum Duplizierungsmodus die Hilfeseite zu den AS-Elementeigenschaften und die Anleitung zum Hinzufügen von Schrittaktionen.

**Siehe auch:**

- Dokumentation PLC: AS-Elementeigenschaften
- Dokumentation PLC: Programmieren in Ablaufsprache (AS)

### 5.11.20 Befehl Change duplication - Entfernen

**Funktion:** Der Befehl hebt die feste Verknüpfung von Aktions-, oder Transitionsobjekten mit dem Schritt oder der Transition, der/die sie aufruft, für den gesamten AS-Baustein auf. Damit wird die Pseudoeinbettung der Aktions- oder Transitionsobjekte aufgehoben. Wenn dann Schritt- und Transitionselemente, die Aktionen oder Transitionen aufrufen, kopiert werden, rufen die Kopien dieselben Aktionen und Transitionen auf wie die Quelle.

**Aufruf:** Menü **AS**

Sehen Sie für Details zum Duplizierungsmodus die Hilfeseite zu den AS-Elementeigenschaften und die Anleitung zum Hinzufügen von Schrittaktionen.

**Siehe auch:**

- Dokumentation PLC: AS-Elementeigenschaften
- Dokumentation PLC: Programmieren in Ablaufsprache (AS)

### 5.11.21 Befehl Schritt einfügen



Der Befehl ist standardmäßig nicht im Menü **AS** enthalten.

---

**Symbol:**

**Funktion:** Der Befehl fügt einen Schritt vor der gerade selektierten Position ein.

**Aufruf:** Menü **AS** , Kontextmenü im AS-Editor

Der neue Schritt wird standardmäßig Step<n> genannt. n ist eine fortlaufende Zahl, beginnend mit 0 für den ersten Schritt, der zusätzlich zum Initialschritt eingefügt wird. Der Name wird nach Mausklick auf den Namen editierbar.

Wenn Sie einen Schritt ohne Transition oder eine Transition ohne Schritt einfügen, führt dies beim Übersetzen zu einem Fehler.

**Siehe auch:**

- [Befehl Schritt-Transition danach einfügen \[► 203\]](#)
- [Befehl Initialschritt \[► 202\]](#)
- Dokumentation PLC: AS-Elemente Schritt und Transition

## 5.11.22 Befehl Schritt danach einfügen

---



Der Befehl ist standardmäßig nicht im Menü **AS** enthalten.

---

**Symbol:**

**Funktion:** Der Befehl fügt einen Schritt nach der gerade selektierten Position ein.

**Aufruf:** Menü **AS**, Kontextmenü im AS-Editor

Der neue Schritt wird standardmäßig Step<n> genannt. n ist eine fortlaufende Zahl, beginnend mit 0 für den ersten Schritt, der zusätzlich zum Initialschritt eingefügt wird. Der Name wird nach Mausklick auf den Namen editierbar.

Wenn Sie einen Schritt ohne Transition oder eine Transition ohne Schritt einfügen, führt dies beim Übersetzen zu einem Fehler.

**Siehe auch:**

- [Befehl Initialschritt \[► 202\]](#)
- [Befehl Schritt-Transition danach einfügen \[► 203\]](#)
- Dokumentation PLC: AS-Element Sprung

## 5.11.23 Befehl Transition einfügen

---



Der Befehl ist standardmäßig nicht im Menü **AS** enthalten.

---

**Symbol:**

**Funktion:** Der Befehl fügt eine Transition vor der gerade selektierten Position ein.

**Aufruf:** Menü **AS**, Kontextmenü im AS-Editor

Die neue Transition wird standardmäßig Trans<n> genannt. n ist eine fortlaufende Zahl, beginnend mit 0 für die erste Transition. Der Name wird nach Mausklick auf den Namen editierbar.

Wenn Sie einen Schritt ohne Transition oder eine Transition ohne Schritt einfügen, führt dies beim Übersetzen zu einem Fehler.

**Siehe auch:**

- [Befehl Schritt-Transition danach einfügen \[► 203\]](#)
- Dokumentation PLC: AS-Elemente Schritt und Transition

## 5.11.24 Befehl Transition danach einfügen

---



Der Befehl ist standardmäßig nicht im Menü **AS** enthalten.

---

**Symbol:**

**Funktion:** Der Befehl fügt eine Transition nach der gerade selektierten Position ein.

**Aufruf:** Menü **AS**, Kontextmenü im AS-Editor

Die neue Transition wird standardmäßig Trans<n> genannt. n ist eine fortlaufende Zahl, beginnend mit 0 für die erste Transition. Der Name wird nach Mausklick auf den Namen editierbar.

Wenn Sie einen Schritt ohne Transition oder eine Transition ohne Schritt einfügen, führt dies beim Übersetzen zu einem Fehler.

**Siehe auch:**

- [Befehl Schritt danach einfügen \[▶ 212\]](#)
- Dokumentation PLC: AS-Elemente Schritt und Transition

## 5.12 CFC

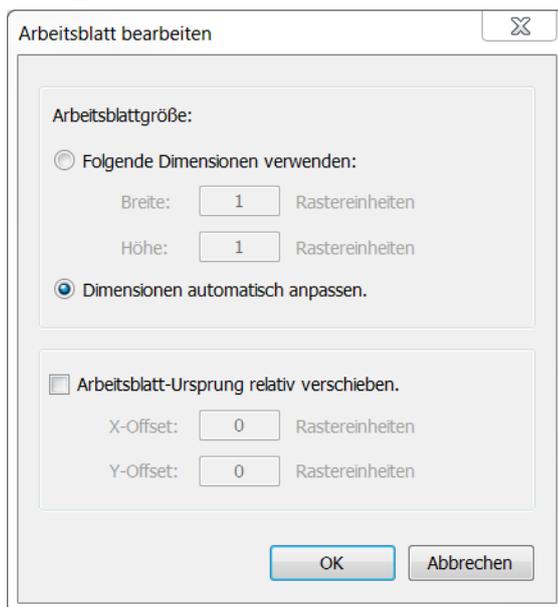
### 5.12.1 Befehl Arbeitsblatt bearbeiten

**Funktion:** Der Befehl öffnet den Dialog **Arbeitsblatt bearbeiten**, in dem Sie die Größe des Arbeitsblattes festlegen.

**Aufruf:** Menü **CFC**

**Voraussetzungen:** Ein CFC-Editor ist aktiv.

**Dialog Arbeitsblatt bearbeiten**



Folgende Dimensionen verwenden	Hier stellen Sie die Größe des Arbeitsblatts ein. Ihre Änderung wird nur übernommen, wenn die Größe für das bestehende Programm ausreichend ist.
Dimensionen automatisch anpassen	Passt die Größe des Arbeitsblatts automatisch an die Größe Ihres Programms an.
Arbeitsblatt Ursprung relativ verschieben	Verschiebt das Arbeitsblatt auf der x- oder y-Achse. Die Eingabe negativer Zahlen ist erlaubt.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

### 5.12.2 Befehl Seitengröße bearbeiten

**Funktion:** Der Befehl öffnet den Dialog **Seitengröße bearbeiten**. Darin verändern Sie die Größe des seitenorientierten CFC-Editors.

**Aufruf:** Menü **CFC**

**Voraussetzungen:** Ein seitenorientierter CFC-Editor ist aktiv.

### Dialog Seitengröße bearbeiten

Breite	Breite der Seite (Minimum 24, Maximum 1024). Elemente außerhalb des Arbeitsbereiches werden rot markiert.
Höhe	Höhe der Seite (Minimum 24, Maximum 1024). Elemente außerhalb des Arbeitsbereichs werden rot markiert.
Randbreite	Breite der Randspalte (Minimum 6, Maximum 25% oder Seitenbreite).
Als Standard für neue CFC-Objekte setzen	 : Die aktuellen Einstellungen werden als Standard für neue CFC-Objekte gewählt.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: CFC-Element Seite

## 5.12.3 Befehl Negieren

Symbol: 

**Funktion:** Der Befehl negiert den ausgewählten Bausteineingang oder Bausteinausgang.

**Aufruf:** Menü **CFC**, Kontextmenü

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Ein Bausteineingang oder Bausteinausgang ist selektiert.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

## 5.12.4 Befehl EN/ENO

Symbol: 

**Funktion:** Der Befehl fügt dem selektierten Baustein einen booleschen Eingang EN (Enable) und einen booleschen Ausgang ENO (Enable Out) hinzu.

**Aufruf:** Menü **CFC**, Kontextmenü

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Ein Baustein ist selektiert.

Der hinzugefügte Eingang „EN“ aktiviert den Baustein. Nur wenn er den Wert TRUE hat, wird der Baustein ausgeführt. Der Wert dieses Signals wird am Ausgang ENO ausgegeben.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

## 5.12.5 Befehl Kein

Symbol: 

**Funktion:** Der Befehl entfernt einen Reset (R), einen Set (S) oder ein REF vom Eingang des Elements „Ausgang“.

**Aufruf:** Menü **CFC > Set/Reset**, Kontextmenü > Set/Reset

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Der Eingang eines Elements **Ausgang** ist selektiert.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

## 5.12.6 Befehl R (Reset)

Symbol: 

**Funktion:** Der Befehl fügt dem Eingang eines booleschen Elements **Ausgang** einen Reset hinzu.

**Aufruf:** Menü **CFC > Set/Reset**, Kontextmenü > Set/Reset

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Der Eingang eines Elements „Ausgang“ ist selektiert.

Wenn ein Element **Ausgang** einen Reset-Eingang hat, wird der boolesche Ausgangswert auf FALSE gesetzt, sobald der Wert des Eingangs TRUE ist. Der Wert FALSE am Ausgang bleibt erhalten, auch wenn sich der Eingangswert wieder ändert.

**Siehe auch:**

- [Befehl S \(Set\) \[► 215\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

## 5.12.7 Befehl S (Set)

Symbol: 

**Funktion:** Der Befehl fügt dem Eingang eines booleschen Elements „Ausgang“ einen Set (S) hinzu.

**Aufruf:** Menü **CFC > Set/Reset**, Kontextmenü > Set/Reset

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Der Eingang eines Elements **Ausgang** ist selektiert.

Wenn ein Element **Ausgang** einen Set-Eingang hat, wird der boolesche Ausgangswert auf TRUE gesetzt, sobald der Wert des Eingangs TRUE ist. Der Wert TRUE am Ausgang bleibt erhalten, auch wenn sich der Eingangswert wieder ändert.

**Siehe auch:**

- [Befehl R \(Reset\) \[► 215\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

## 5.12.8 Befehl REF= (Reference-Zuweisung)

Symbol: 

**Funktion:** Der Befehl weist einem Element „Ausgang“ eine Referenz zu.

**Aufruf:** Menü **CFC > Set/Reset**, Kontextmenü > Set/Reset

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Der Eingang eines Elements **Ausgang** ist selektiert.

**Beispiel:**

Deklaration:

```
refInt : REFERENCE TO INT;
nVar1 : INT;
```

CFC:



Dies entspricht dem ST-Code

```
refInt REF= nVar1;
```

Weitere Informationen finden Sie in der Beschreibung zum Datentyps REFERENCE TO.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: Reference

## 5.12.9 Befehl Ausführungsreihenfolge anzeigen

Symbol:

**Funktion:** Der Befehl blendet für alle CFC-Elemente des Programmierobjekts kurzzeitig eine Marke mit Nummer ein.**Aufruf:**

- Menü **CFC > Ausführungsreihenfolge**
- Kontextmenü im CFC-Editor > **Ausführungsreihenfolge**

**Voraussetzungen:** Ein CFC-Editor ist aktiv und der Automatische Datenfluss-Modus ist aktiviert.

Die Nummern geben die automatisch ermittelte Ausführungsreihenfolge wieder. Die Ausführungsreihenfolge wird nach Datenfluss und bei mehreren Netzwerken nach deren topologischer Position im Editor ermittelt.

Sobald Sie in den CFC-Editor klicken, werden die Marken ausgeblendet.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: Automatische Ausführungsreihenfolge nach Datenfluss
- Dokumentation PLC: [CFC-Einstellungen](#) [► 99]

## 5.12.10 Befehl Startpunkt der Rückkopplung setzen

Symbol:

**Funktion:** Der Befehl definiert das selektierte Element als Startpunkt innerhalb einer Rückkopplung.**Aufruf:**

- Menü **CFC > Ausführungsreihenfolge**
- Kontextmenü im CFC-Editor > **Ausführungsreihenfolge**

**Voraussetzungen:** Ein CFC-Editor ist aktiv und der Automatische Datenfluss-Modus ist aktiviert. Ein Netzwerk des CFC-Programmierbausteins enthält eine Rückkopplung und ein Element innerhalb dieser Rückkopplung ist selektiert.

Im CFC-Editor ist der Startpunkt innerhalb von Rückkopplungen mit dem Symbol  dekoriert. Dieses Element hat die niedrigste Nummer der Ausführungsreihenfolge innerhalb der Rückkopplungen. Zur Laufzeit beginnt die Abarbeitung der Rückkopplung mit diesem Element.

**Siehe auch:**

- Dokumentation PLC: [Befehl Ausführungsreihenfolge anzeigen \[► 216\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: Automatische Ausführungsreihenfolge nach Datenfluss
- Dokumentation PLC: [CFC-Einstellungen \[► 99\]](#)

### 5.12.11 Befehl An den Anfang

Symbol: 

**Funktion:** Der Befehl nummeriert die Elemente so, dass die selektierten Elemente am Anfang der Ausführungsreihenfolge stehen.

**Aufruf:** Menü **CFC > Ausführungsreihenfolge**, Kontextmenü > Ausführungsreihenfolge

**Voraussetzungen:** Ein CFC-Editor ist aktiv und der Explizite Ausführungsreihenfolge-Modus ist aktiviert. Mindestens ein Element ist selektiert.

Die selektierten Elemente erhalten die niedrigsten Nummern beginnend mit 0 unter Beibehaltung der bisherigen Reihenfolge. Die restlichen Elemente werden so nummeriert, dass ihre Ausführungsreihenfolge bestehen bleibt. Die topologischen Positionen der Elemente werden nicht geändert.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: Automatische Ausführungsreihenfolge nach Datenfluss
- Dokumentation PLC: [CFC-Einstellungen \[► 99\]](#)

### 5.12.12 Befehl Ans Ende

Symbol: 

**Funktion:** Der Befehl nummeriert die Elemente so, dass die selektierten Elemente am Ende der Ausführungsreihenfolge stehen.

**Aufruf:** Menü **CFC > Ausführungsreihenfolge**, Kontextmenü > Ausführungsreihenfolge

**Voraussetzungen:** Ein CFC-Editor ist aktiv und der Explizite Ausführungsreihenfolge-Modus ist aktiviert. Mindestens ein Element ist selektiert.

Die selektierten Elemente erhalten die höchsten Nummern unter Beibehaltung der bisherigen Reihenfolge. Die restlichen Elemente werden so nummeriert, dass ihre Ausführungsreihenfolge bestehen bleibt. Die topologischen Positionen der Elemente werden nicht geändert.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: Automatische Ausführungsreihenfolge nach Datenfluss
- Dokumentation PLC: [CFC-Einstellungen \[► 99\]](#)

### 5.12.13 Befehl Eins vor

Symbol: 

**Funktion:** Der Befehl nummeriert die Elemente so, dass die selektierten Elemente um eins weiter vorne stehen.

**Aufruf:** Menü **CFC > Ausführungsreihenfolge**, Kontextmenü > Ausführungsreihenfolge

**Voraussetzungen:** Ein CFC-Editor ist aktiv und der Explizite Ausführungsreihenfolge-Modus ist aktiviert. Mindestens ein Element ist selektiert.

Die selektierten Elemente erhalten eine um eins niedrigere Nummerierung unter Beibehaltung der bisherigen Reihenfolge, sodass sie eine Ausführungsposition früher abgearbeitet werden. Die restlichen Elemente werden so nummeriert, dass ihre Ausführungsreihenfolge bestehen bleibt. Die topologischen Positionen der Elemente werden nicht geändert.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: Automatische Ausführungsreihenfolge nach Datenfluss
- Dokumentation PLC: [CFC-Einstellungen](#) [► 99]

### 5.12.14 Befehl Eins zurück

Symbol: 

**Funktion:** Der Befehl nummeriert die Elemente so, dass die selektierten Elemente um eins weiter hinten stehen.

**Aufruf:** Menü **CFC > Ausführungsreihenfolge**, Kontextmenü > Ausführungsreihenfolge

**Voraussetzungen:** Ein CFC-Editor ist aktiv und der Explizite Ausführungsreihenfolge-Modus ist aktiviert. Mindestens ein Element ist selektiert.

Die selektierten Elemente erhalten eine um eins höhere Nummerierung unter Beibehaltung der bisherigen Reihenfolge, sodass sie eine Ausführungsposition später abgearbeitet werden. Die restlichen Elemente werden so nummeriert, dass ihre Ausführungsreihenfolge bestehen bleibt. Die topologischen Positionen der Elemente werden nicht geändert.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: Automatische Ausführungsreihenfolge nach Datenfluss
- Dokumentation PLC: [CFC-Einstellungen](#) [► 99]

### 5.12.15 Befehl Ausführungsreihenfolge setzen

**Funktion:** Der Befehl öffnet einen Dialog zum Setzen der Nummer des selektierten Elementes auf einen beliebigen Wert.

**Aufruf:** Menü **CFC > Ausführungsreihenfolge**, Kontextmenü > Ausführungsreihenfolge

**Voraussetzungen:** Ein CFC-Editor ist aktiv und der Explizite Ausführungsreihenfolge-Modus ist aktiviert. Genau ein Element ist selektiert.

Das selektierte Element erhält die im Dialog angegebene Nummer. Die restlichen Elemente werden so nummeriert, dass ihre Ausführungsreihenfolge bestehen bleibt. Die topologischen Positionen der Elemente werden nicht geändert.

**Siehe auch:**

- Dokumentation PLC: [Befehl Nach Datenfluss anordnen \[► 219\]](#)
- Dokumentation PLC: [Befehl Topologisch anordnen \[► 219\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: Automatische Ausführungsreihenfolge nach Datenfluss
- Dokumentation PLC: [CFC-Einstellungen \[► 99\]](#)

## 5.12.16 Befehl Nach Datenfluss anordnen

**Funktion:** Der Befehl nummeriert die Elemente im Programm nach Datenfluss und bei mehreren Netzwerken nach ihrer topologischen Position im Editor.

**Aufruf:** Menü **CFC > Ausführungsreihenfolge**, Kontextmenü > Ausführungsreihenfolge

**Voraussetzungen:** Ein CFC-Editor ist aktiv und der Explizite Ausführungsreihenfolge-Modus ist aktiviert.

Die Ausführungsreihenfolge wird nach dem Datenfluss und (bei mehreren Netzwerken) nach der topologischen Position der Netzwerke angeordnet. Alle nummerierten Elemente des Programmierbausteins werden entsprechend gesetzt. Die Ausführungsreihenfolge ist danach identisch zu der im automatischen Datenfluss-Modus. Die topologischen Positionen der Elemente werden nicht geändert.

**Siehe auch:**

- Dokumentation PLC: [Befehl Topologisch anordnen \[► 219\]](#)
- Dokumentation PLC: [Befehl Ausführungsreihenfolge setzen \[► 218\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: Automatische Ausführungsreihenfolge nach Datenfluss
- Dokumentation PLC: [CFC-Einstellungen \[► 99\]](#)

## 5.12.17 Befehl Topologisch anordnen

**Funktion:** Der Befehl ordnet die Ausführungsreihenfolge der Elemente nach ihrer topologischen Position von rechts nach links und oben nach unten an.

**Aufruf:** Menü **CFC > Ausführungsreihenfolge**, Kontextmenü > Ausführungsreihenfolge

**Voraussetzungen:** Ein CFC-Editor ist aktiv und der Explizite Ausführungsreihenfolge-Modus ist aktiviert. Mindestens ein Element ist selektiert.

Der Befehl wirkt sich auf alle Elemente im Programm aus, auch wenn nicht alle Elemente beim Ausführen des Befehls selektiert sind. Die topologischen Positionen der Elemente werden nicht geändert.

**Siehe auch:**

- Dokumentation PLC: [Befehl Nach Datenfluss anordnen \[► 219\]](#)
- Dokumentation PLC: [Befehl Ausführungsreihenfolge setzen \[► 218\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: Automatische Ausführungsreihenfolge nach Datenfluss
- Dokumentation PLC: [CFC-Einstellungen \[► 99\]](#)

## 5.12.18 Befehl Selektierte Anschlüsse verbinden

Symbol: 

**Funktion:** Der Befehl stellt eine Verbindung zwischen den selektierten Anschlüssen her.

**Aufruf:** Menü **CFC**, Kontextmenü

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Genau ein Ausgang und mehrere Eingänge sind selektiert.

Zum Selektieren der Anschlüsse müssen Sie die **[STRG]**-Taste gedrückt halten, während Sie die Anschlüsse anklicken. Danach führen Sie den Befehl aus.

**Siehe auch:**

- [Befehl Verbundene Anschlüsse selektieren \[► 221\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

## 5.12.19 Befehl Verbindung lösen

Symbol: 

**Funktion:** Dieser Befehl löst eine gesperrte Verbindung.

**Aufruf:** Menü **CFC > Routing**, Kontextmenü > Routing

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Eine Verbindung oder eine Verbindungsmarke ist selektiert.

Eine gesperrte Verbindung erhalten Sie, wenn Sie die Verbindungen des automatischen Routings verändern. Wenn Sie erneut ein automatisches Routing durchführen wollen, müssen Sie vorher eine gesperrte Verbindung lösen.



Mit einem Mausklick auf das Icon einer gesperrten Verbindung können Sie diese Verbindung ebenfalls lösen.

---

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: CFC-Element Verbindungsmarke-Quelle/Ziel

## 5.12.20 Befehl Nächste Kollision zeigen

**Funktion:** Der Befehl zeigt die nächste Kollision im Editor auf und markiert die betroffene Stelle.

**Aufruf:** Schaltfläche  in der rechten oberen Ecke des Editors

**Voraussetzungen:** Ein CFC-Editor ist aktiv und es ist mindestens eine Verbindung mit Kollision vorhanden.

Diese Funktion ist sehr nützlich, wenn Sie mit großen Netzwerken arbeiten und nur einen Teilbereich sehen. Eine Kollision wird Ihnen zusätzlich durch das rot umrandete Symbol in der rechten oberen Ecke des Editors signalisiert.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

## 5.12.21 Befehl Verbundene Anschlüsse selektieren

Symbol: 

**Funktion:** Der Befehl selektiert alle Anschlüsse, die mit der gerade selektierten Leitung oder im seitenorientierten CFC mit der gerade selektierten Verbindungsmarke verbunden sind.

**Aufruf:** Kontextmenü

**Voraussetzungen:** Ein CFC-Editor oder ein seitenorientierter CFC-Editor ist aktiv. Eine Leitung und damit genau ein Anschluss oder genau eine Verbindungsmarke ist selektiert.

**Siehe auch:**

- [Befehl Selektierte Anschlüsse verbinden \[► 220\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

## 5.12.22 Befehl Attributierte Komponente als Eingang verwenden

Symbol: 

**Funktion:** Der Befehl ermöglicht das Verbinden einer Strukturkomponenten mit einem Eingang skalaren Typs.

**Aufruf:** Menü **CFC > Pins**, Kontextmenü > Pins

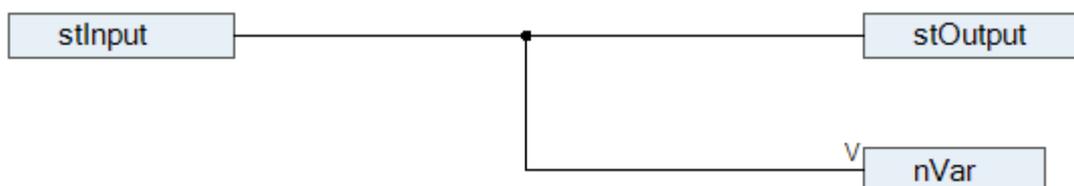
**Voraussetzungen:** Ein CFC-Editor ist aktiv und ein Bausteineingang ist selektiert.

Die Komponente der Struktur, die mit dem Eingang des Folgebausteins verbunden wird, muss mit dem Attribut {attribute 'ProcessValue'} versehen sein. Der Datentyp der Strukturkomponente muss zu dem Datentyp des nachfolgenden Eingangs kompatibel sein. So verbundene Eingänge werden mit dem Symbol V markiert.

**Beispiel**

```
TYPE ST_Sample :
STRUCT
  {attribute 'ProcessValue'}
  nVar1 : INT;
  nVar2 : INT;
END_STRUCT
END_TYPE

PROGRAM MAIN
VAR
  stInput   : ST_Sample;
  stOutput  : ST_Sample;
  nVar      : INT;
END_VAR
```



Wenn Sie bei dieser Verknüpfung den Befehl **Attributierte Komponente als Eingang verwenden** nicht ausführen, wird ein Compilerfehler erzeugt.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

### 5.12.23 Befehl Anschlüsse zurücksetzen

Symbol: 

**Funktion:** Der Befehl stellt gelöschte Anschlüsse eines Bausteins wieder her.

**Aufruf:** Menü **CFC > Pins**, Kontextmenü > Pins

**Voraussetzungen:** Ein CFC-Editor ist aktiv und ein Baustein ist selektiert.

Der Befehl stellt alle Ein- und Ausgänge des Bausteins wieder her, so wie sie in seiner Implementierung definiert sind.

**Siehe auch:**

- [Befehl Nicht verbundene Anschlüsse entfernen \[► 222\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

### 5.12.24 Befehl Nicht verbundene Anschlüsse entfernen

Symbol: 

**Funktion:** Der Befehl entfernt alle ungenutzten Anschlüsse des selektierten Elements.

**Aufruf:** Menü **CFC > Pins**, Kontextmenü > Pins

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Ein Element ist selektiert.

**Siehe auch:**

- [Befehl Anschlüsse zurücksetzen \[► 222\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

### 5.12.25 Befehl Eingangsanschluss hinzufügen

Symbol: 

**Funktion:** Der Befehl fügt dem selektierten Baustein einen weiteren Eingang hinzu.

**Aufruf:** Menü **CFC > Pins**, Kontextmenü > Pins

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Ein Baustein ist selektiert.

**Siehe auch:**

- [Befehl Ausgangsanschluss hinzufügen \[► 222\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

### 5.12.26 Befehl Ausgangsanschluss hinzufügen

Symbol: 

**Funktion:** Der Befehl fügt dem selektierten Baustein einen weiteren Ausgang hinzu.

**Aufruf:** Menü **CFC > Pins**, Kontextmenü > Pins

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Ein geeigneter Baustein ist selektiert.

**Siehe auch:**

- [Befehl Eingangsanschluss hinzufügen \[► 222\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

## 5.12.27 Befehl Alle Verbindungen routen

Symbol: 

**Funktion:** Der Befehl macht alle manuellen Änderungen an den Verbindungen im Programm rückgängig und stellt den ursprünglichen Zustand wieder her.

**Aufruf:** Menü **CFC > Routing**, Kontextmenü > Routing

**Voraussetzungen:** Ein CFC-Editor ist aktiv.

TwinCAT kann Verbindungen, die mit Kontrollpunkten fixiert sind, nicht automatisch routen. Sie müssen die Kontrollpunkte vor dem Ausführen des Befehls entfernen. Dazu verwenden Sie den Befehl Kontrollpunkt

entfernen. Weiterhin müssen Sie manuell veränderte Verbindungen, die mit dem Icon  gekennzeichnet sind, lösen. Dazu verwenden Sie den Befehl Verbindung lösen.

**Siehe auch:**

- [Befehl Kontrollpunkt entfernen \[► 223\]](#)
- [Befehl Verbindung lösen \[► 220\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

## 5.12.28 Befehl Kontrollpunkt erzeugen

Symbol: 

**Funktion:** Der Befehl erzeugt einen Kontrollpunkt an einem Verbinder.

**Aufruf:** Kontextmenü > Routing

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Der Cursor befindet sich über einer Verbindung.

Der Kontrollpunkt wird an der Stelle der Verbindung erstellt, an dem sich der Cursor bei Aufruf des Befehls befindet. Der Befehl entspricht dem Element **Kontrollpunkt** im Fenster **Werkzeugkasten**.

**Siehe auch:**

- [Befehl Kontrollpunkt entfernen \[► 223\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: CFC-Element Kontrollpunkt

## 5.12.29 Befehl Kontrollpunkt entfernen

**Funktion:** Der Befehl entfernt einen Kontrollpunkt.

**Aufruf:** Kontextmenü > Routing

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Sie haben eine Verbindungslinie selektiert.

Wenn Sie den Mauszeiger über eine selektierte Verbindungslinie bewegen, werden die vorhandenen Kontrollpunkte mit gelben Kreissymbolen angezeigt. Setzen Sie den Cursor auf den zu löschenden Kontrollpunkt und führen Sie den Befehl aus dem Kontextmenü aus.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: CFC-Element Kontrollpunkt
- [Befehl Kontrollpunkt erzeugen \[► 223\]](#)

### 5.12.30 Befehl Verbindungsmarke

Symbol: 

**Funktion:** Dieser Befehl schaltet die Darstellung der Verbindung zwischen zwei Elementen zwischen einer Verbindungslinie und der Verwendung von Verbindungsmarken hin und her.

**Aufruf:** Menü **CFC**, Kontextmenü

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Eine Verbindung oder eine Verbindungsmarke ist selektiert.

Wenn Sie eine Verbindungslinie selektiert haben, entfernt der Befehl diese Linie und fügt eine Verbindungsmarke-Quelle am Ausgang des einen Elements hinzu und eine Verbindungsmarke-Ziel am Eingang des anderen. Beide bekommen standardmäßig den gleichen Namen „C-<n>“, wobei n eine laufende Nummer ist.

Wenn Sie ein Verbindungsmarkenpaar selektieren, wandelt der Befehl diese Marken in eine Verbindungslinie um.

**Siehe auch:**

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor
- Dokumentation PLC: CFC-Element Verbindungsmarke-Quelle/Ziel

### 5.12.31 Befehl Gruppe erzeugen

Symbol: 

**Funktion:** Der Befehl gruppiert die selektierten Elemente.

**Aufruf:** Menü **CFC > Gruppieren**, Kontextmenü > Gruppieren

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Mehrere Elemente sind selektiert.

Gruppierte Elemente können nur noch zusammen bewegt werden. Die Position der Elemente wird durch die Gruppierung nicht beeinflusst.

**Siehe auch:**

- [Befehl Gruppierung aufheben \[► 224\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

### 5.12.32 Befehl Gruppierung aufheben

Symbol: 

**Funktion:** Der Befehl entfernt eine zuvor erstellte Gruppierung.

**Aufruf:** Menü **CFC > Gruppieren**, Kontextmenü > Gruppieren

**Voraussetzungen:** Ein CFC-Editor ist aktiviert. Eine Gruppierung ist selektiert.

**Siehe auch:**

- [Befehl Gruppe erzeugen \[► 224\]](#)
- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

### 5.12.33 Befehl Parameter bearbeiten

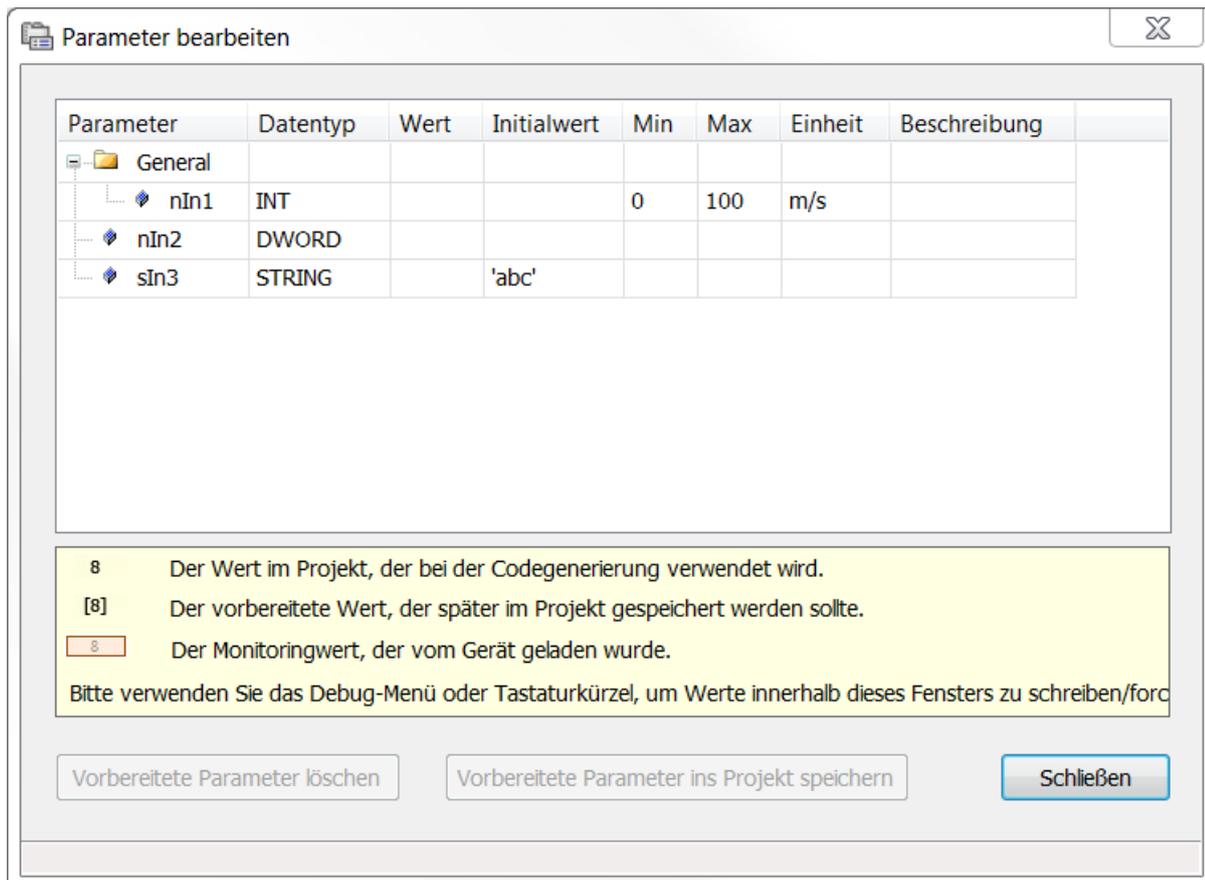
**Funktion:** Der Befehl öffnet den Dialog „Parameter bearbeiten“, in dem Sie die konstanten Eingangsparameter eines Funktionsbausteins verändern.

**Aufruf:** Menü **CFC > Parameter bearbeiten**, Kontextmenü > Parameter bearbeiten, Klick auf Funktionsbaustein **Parameter**.

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Es ist ein Funktionsbaustein instanziiert, der in seiner Deklaration VAR\_INPUT CONSTANT-Variablen hat.

Ein Baustein mit VAR\_INPUT CONSTANT-Variablen wird von TwinCAT durch das Wort „Parameter“ in der linken unteren Ecke des Bausteins angezeigt.

#### Dialog Parameter bearbeiten



Parameter	Name der Variablen
Datentyp	Datentyp der Variablen
Wert	Klicken Sie in das Feld um einen Wert einzugeben.
Initialwert	Initialisierungswert
Kategorie	Zusätzliche Informationen zu den Parametern. Diese Werte werden über Attribute definiert und können in diesem Dialog nicht verändert werden
Einheit	
Min	
Max	
Vorbereitete Parameter löschen	Der Befehl ist aktiv, wenn Sie einen vorbereiteten Wert geschrieben haben (Befehl Debug > Werte schreiben)

Nach Verlassen des Felds und Verlassen des Dialogs mit **OK** werden die Werteänderungen im Projekt angewendet.

### Beispiel eines Bausteins mit konstanten Eingängen:

```
FUNCTION_BLOCK FB_Sample
VAR_INPUT CONSTANT
    {attribute 'parameterCategory':='General'}
    {attribute 'parameterUnit':='m/s'}
    {attribute 'parameterMinValue':='0'}
    {attribute 'parameterMaxValue':='100'}
    nIn1 : INT;
    nIn2 : DWORD:=24354333;
    sIn3 : STRING:='abc';
END_VAR
```



Diese Funktionalität und die Deklaration von Variablen mit Schlüsselwort `VAR_INPUT CONSTANT` gelten nur für den CFC-Editor. Im FUP-Editor zeigt TwinCAT immer alle Eingangsparameter am Baustein an, egal ob sie als `VAR_INPUT` oder `VAR_INPUT CONSTANT` deklariert sind. Auch für Texteditoren unterscheidet TwinCAT diesbezüglich nicht.

### Siehe auch:

- [Befehl Vorbereitete Parameter im Projekt speichern \[► 227\]](#)
- Dokumentation PLC: Ändern konstanten Eingangsparametern von Funktionsbausteininstanzen

## 5.12.34 Befehl FB-Eingang forcen



Verfügbar ab TwinCAT 3.1 Build 4026



Diese Art des Forcens verwendet intern einen Haltepunkt und ist damit vom Forcen über den Befehl **Werte forcen** zu unterscheiden: Werte, die über den Befehl **FB-Eingang forcen** geforct wurden, reagieren nicht auf die Befehle **Alle Forces anzeigen** oder **Forcen für alle Werte aufheben**.

**Funktion:** Der Befehl öffnet einen Dialog **Wert forcen** zum Forcen des selektierten Eingangs eines Funktionsbausteins. Das Forcen kann mit dem gleichen Befehl und Dialog wieder aufgehoben werden.

**Aufruf:** Menü **CFC**, Kontextmenü

**Voraussetzungen:** Der CFC-Editor ist im Onlinebetrieb und der Eingang des Funktionsbausteins ist selektiert.

Im Dialog **Wert forcen** können Sie entweder einen Wert eingeben, mit dem der Eingang des Funktionsbausteins geforct werden soll, oder den aktuell geforcten Wert wieder entfernen.

Nach dem Forcen erscheint der Eingang grün hinterlegt. Boolesche Eingänge erhalten zudem ein kleines Monitoringfenster mit dem geforcten Wert. Der geforcte Wert wird in der Spalte **Wert** von Monitoringansichten angezeigt, also im Deklarationsteil des Programmbausteins oder in einer Überwachungsliste.

### Dialog Wert forcen

Ausdruck	Name des Funktionsbausteineingangs
Typ	Datentyp des Eingangs

Was wollen Sie tun?

Einen neuen Wert zum Forcen setzen	Im Eingabefeld können Sie den gewünschten neuen Wert eintragen. Das Format muss dem Datentyp entsprechen.
Wert entfernen	Das Forcen am Eingang wird aufgehoben.

#### Siehe auch:

- Dokumentation PLC: Continuous Function Chart (CFC)
- Dokumentation PLC: CFC-Editor

## 5.12.35 Befehl Vorbereitete Parameter im Projekt speichern

**Funktion:** Der Befehl übernimmt die vorbereiteten Parameterwerte in das Projekt.

**Aufruf:** Menü **CFC**

**Voraussetzungen:** Ein CFC-Editor ist aktiv. Parameterwerte von Funktionsbausteininstanzen wurden im Onlinebetrieb geändert. Die Anwendung ist im Offlinebetrieb.

Wenn sich die Werte von Konstanten auf der Steuerung von den Werten in der Applikation unterscheiden, wird dies durch einen roten Stern rechts neben dem Parameterfeld gekennzeichnet. Durch den Befehl **Vorbereitete Parameterwerte übernehmen** übernehmen Sie die Werte der Steuerung in Ihre Applikation.

#### Siehe auch:

- [Befehl Parameter bearbeiten \[► 225\]](#)
- Dokumentation PLC: Ändern konstanten Eingangsparametern von Funktionsbausteininstanzen

## 5.13 FUP/KOP/AWL

### 5.13.1 Befehl Kontakt einfügen (rechts)

Symbol: 

**Funktion:** Der Befehl fügt einen Kontakt rechts neben dem gewählten Element ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv. Eine Leitung, ein Kontakt oder ein Baustein ist selektiert.

#### Siehe auch:

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: KOP-Element Kontakt

## 5.13.2 Befehl Netzwerk einfügen

Symbol: 

**Funktion:** Der Befehl fügt ein weiteres Netzwerk im FUP/KOP/AWL-Editor ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP-, KOP- oder AWL-Editor ist aktiv. Kein Baustein ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: FUP/KOP/AWL-Element Netzwerk

## 5.13.3 Befehl Netzwerk einfügen (unterhalb)

Symbol: 

**Funktion:** Der Befehl fügt ein weiteres Netzwerk im FUP/KOP/AWL-Editor unterhalb des selektieren Netzwerks ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP-, KOP- oder AWL-Editor ist aktiv. Ein Netzwerk ist selektiert, aber kein Baustein ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: FUP/KOP/AWL-Element Netzwerk

## 5.13.4 Befehl Kommentierung ein/aus

Symbol: 

**Funktion:** Der Befehl kommentiert das selektierte Netzwerk ein oder aus.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP-, KOP-, oder AWL-Editor ist aktiv. Ein Netzwerk ist selektiert, aber kein Baustein ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

## 5.13.5 Befehl Zuweisung einfügen

Symbol: 

**Funktion:** Der Befehl fügt eine Zuweisung im FUP oder KOP Editor ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP-, KOP- oder AWL-Editor ist aktiv. Ein Netzwerk ist selektiert, aber kein Baustein ist selektiert.



In AWL wird eine Zuweisung über die Operatoren LD und ST programmiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: FUP/KOP/AWL-Element Zuweisung

### 5.13.6 Befehl Bausteinanruf einfügen

Symbol:

**Funktion:** Der Befehl fügt einen im Projekt verfügbaren Baustein am Ende des selektierten Netzwerks ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP-, KOP- oder AWL-Editor ist aktiv. Ein Netzwerk ist selektieren, aber kein Baustein ist selektiert.

Wenn Sie den Befehl auswählen, öffnet sich die Eingabehilfe. Dort können Sie den gewünschten Baustein auswählen.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: FUP/KOP/AWL-Element Baustein

### 5.13.7 Befehl Baustein mit EN/ENO einfügen

Symbol:

**Funktion:** Der Befehl fügt einen Baustein mit einem booleschen Eingang „Enable“ und einen booleschen Ausgang „Enable Out“ am Ende des selektierten Netzwerks ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP-, KOP- oder AWL-Editor ist aktiv. Ein Netzwerk ist selektiert, aber kein Baustein ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: FUP/KOP/AWL-Element Baustein mit EN/ENO

### 5.13.8 Befehl Leeren Baustein einfügen

Symbol:

**Funktion:** Der Befehl fügt am Ende des derzeit selektierten Netzwerks einen leeren Funktionsbaustein ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP-, KOP- oder AWL-Editor ist aktiv. Ein Netzwerk ist selektiert, aber kein Baustein ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL

- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: FUP/KOP/AWL-Element Baustein

### 5.13.9 Befehl Leeren Baustein mit EN/ENO einfügen

Symbol: 

**Funktion:** Der Befehl fügt einen leeren Baustein mit einem booleschen Eingang „Enable“ und einen booleschen Ausgang „Enable Out“ am Ende des selektierten Netzwerks ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Ein FUP-Editor, KOP-Editor oder ein AWL-Editor ist aktiv. Ein Netzwerk muss selektiert sein. Kein anderer Baustein darf selektiert sein.

Wenn zum Zeitpunkt des Bausteinaufrufs „Enable“ den Wert FALSE hat, werden die im Baustein definierten Operationen nicht ausgeführt. Andernfalls, also wenn „Enable“ den Wert TRUE hat, werden diese Operationen ausgeführt. Der ENO-Ausgang wirkt als Repeater des EN-Eingangs.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: FUP/KOP/AWL-Element Baustein mit EN/ENO

### 5.13.10 Befehl Sprung einfügen

Symbol: 

**Funktion:** Der Befehl fügt ein Sprungelement vor dem selektierten Element ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP-, KOP- oder AWL-Editor ist aktiv. Ein Verbinder ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: FUP/KOP/AWL-Element Sprung

### 5.13.11 Befehl Sprungmarke einfügen

Symbol: 

**Funktion:** Der Befehl fügt eine Sprungmarke in das aktuell selektierte Netzwerk ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP-, KOP- AWL-Editor ist aktiv. Ein Netzwerk ist selektiert. Keine Sprungmarke ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: FUP/KOP/AWL-Element Sprungmarke

### 5.13.12 Befehl Return einfügen

Symbol: 

**Funktion:** Der Befehl fügt ein Element „Return“ an der selektierten Stelle ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP-, KOP- oder AWL-Editor ist aktiv. Ein Bausteinausgang ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: FUP/KOP/AWL-Element Return

### 5.13.13 Befehl Bausteineingang einfügen

Symbol: 

**Funktion:** Der Befehl fügt einem erweiterbaren Baustein (ADD, OR, AND, MUL, SEL) einen weiteren Eingang oberhalb des ausgewählten Eingangs hinzu.

**Aufruf:** Menü **FUP/KOP/AWL**

**Voraussetzungen:** Der FUP-, KOP-Editor ist aktiv. Ein Eingang eines Bausteins ist selektiert.

Wenn ein Baustein selektiert ist, ist im Kontextmenü der Befehl **Namen Eingang anhängen** verfügbar. Der Eingang wird am unteren Ende des Bausteins eingefügt.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.14 Befehl Baustein parallel einfügen (unterhalb)

Symbol: 

**Funktion:** Der Befehl fügt einen leeren Baustein parallel unterhalb des selektierten Bausteins ein.

**Aufruf:** Menü **FBD/IL/LD**, Kontextmenü

**Voraussetzungen:** Im KOP-Editor ist ein Baustein selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.15 Befehl Spule einfügen

Symbol: 

**Funktion:** Der Befehl fügt eine Spule im Netzwerk ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv. Ein Netzwerk, eine Spule oder ein Verbinder ist selektiert, aber kein Baustein ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: KOP-Element Spule

### 5.13.16 Befehl Set-Spule einfügen

Symbol: 

**Funktion:** Der Befehl fügt eine Set-Spule im Netzwerk ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv. Ein Netzwerk, eine Spule oder eine Leitung ist selektiert, aber kein Baustein ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: KOP-Element Spule

### 5.13.17 Befehl Reset-Spule einfügen

Symbol: 

**Funktion:** Der Befehl fügt eine Reset-Spule im Netzwerk ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv. Ein Netzwerk, eine Spule oder ein Leitung ist selektiert, aber kein Baustein ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: KOP-Element Spule

### 5.13.18 Befehl Kontakt einfügen

Symbol: 

**Funktion:** Der Befehl fügt einen Kontakt links neben dem gewählten Element ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv. Eine Leitung oder ein Kontakt ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: KOP-Element Kontakt

### 5.13.19 Befehl Kontakt parallel einfügen (unterhalb)

Symbol: 

**Funktion:** Der Befehl fügt einen Kontakt mit Leitungen parallel unterhalb des gewählten Elements ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv. Eine Leitung oder ein Kontakt oder ein Baustein ist selektiert.

**i** Geschlossene parallele Verzweigungen in einem KOP-Netzwerk können Sie als „Short Circuit Evaluation“ (SCE) oder OR-Konstrukte programmieren. SCE-Zweige werden mit doppelten vertikalen Linien dargestellt, OR-Zweige mit einfachen. Sehen Sie hierzu die Hilfeseite zu „Geschlossenen Leitungsverzweigungen“.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: KOP-Element Kontakt

### 5.13.20 Befehl Kontakt parallel einfügen (oberhalb)

Symbol: 

**Funktion:** Der Befehl fügt einen Kontakt mit Leitungen parallel oberhalb des gewählten Elements ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv. Eine Leitung, in Kontakt oder ein Baustein ist selektiert.

**i** Geschlossene parallele Verzweigungen in einem KOP-Netzwerk können Sie als „Short Circuit Evaluation“ (SCE) oder OR-Konstrukte programmieren. SCE-Zweige werden mit doppelten vertikalen Linien dargestellt, OR-Zweige mit einfachen. Sehen Sie hierzu die Hilfeseite zu „Geschlossenen Leitungsverzweigungen“.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: KOP-Element Kontakt

### 5.13.21 Befehl Negierten Kontakt einfügen

Symbol: 

**Funktion:** Der Befehl fügt einen negierten Kontakt links neben dem gewählten Element ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv. Eine Leitung oder ein Kontakt ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: KOP-Element Kontakt

### 5.13.22 Befehl Negierten Kontakt parallel einfügen (unterhalb)

Symbol: 

**Funktion:** Der Befehl fügt einen negierten Kontakt mit Leitungen parallel unterhalb des gewählten Elements ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv. Eine Leitung, ein Kontakt oder ein Baustein ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: KOP-Element Kontakt

### 5.13.23 Befehl Kontakte einfügen: Darunter einfügen

**Funktion:** Der Befehl fügt einen zuvor kopierten Kontakt mit Leitungen unterhalb des selektierten Elements ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: KOP-Element Kontakt

### 5.13.24 Befehl Kontakte einfügen: Darüber einfügen

**Funktion:** Der Befehl fügt einen zuvor kopierten Kontakt mit Leitungen oberhalb des selektierten Elements ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv. Eine Leitung oder ein Kontakt ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: KOP-Element Kontakt

### 5.13.25 Befehl Kontakte einfügen: Rechts einfügen (danach)

**Funktion:** Der Befehl fügt einen zuvor kopierten Kontakt rechts neben dem selektierten Element ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv. Eine Leitung oder ein Kontakt ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: KOP-Element Kontakt

### 5.13.26 Befehl AWL-Zeile danach einfügen

Symbol: 

**Funktion:** Der Befehl fügt eine Anweisungszeile unterhalb der selektierten Zeile ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der AWL-Editor ist aktiv. Eine Zeile ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.27 Befehl AWL-Zeile löschen

Symbol: 

**Funktion:** Der Befehl löscht die selektierte Anweisungszeile.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der AWL-Editor ist aktiv. Eine Zeile ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.28 Befehl Negation

Symbol: 

**Funktion:** Der Befehl negiert folgende Elemente:

- Ein-/Ausgang eines Bausteins
- Sprung
- Return
- Spule

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP-, oder KOP-Editor ist aktiv. Das entsprechende Element ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.29 Befehl Flankenerkennung

Symbol FUP: 

Symbol KOP: 

**Funktion:** Der Befehl fügt eine Flankenerkennung vor dem selektierten Bausteineingang oder Bausteinausgang ein. Durch ein-, zwei- oder dreimaliges Ausführen des Befehls kann zwischen den folgenden Funktionalitäten unterschieden werden:

- Die Flankenerkennung, die beim einmaligen Ausführen des Befehls eingefügt wird, dient zum Erkennen einer steigenden Flanke. Dabei zeigt der Pfeil des Symbols nach rechts.
- Durch erneutes Ausführen des Befehls wird die Flankenerkennung umgedreht, sodass fallende Flanken erkannt werden. Hierbei zeigt der Pfeil des Symbols nach links.
- Wird der Befehl ein weiteres Mal ausgeführt, werden die Flankenerkennung und das Symbol entfernt.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP- oder KOP-Editor ist aktiv. Ein Bausteineingang oder Bausteinausgang ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.30 Befehl Set/Reset

Symbol: 

**Funktion:** Der Befehl wechselt bei einem Element mit booleschen Ausgang zwischen Reset, Set und keiner Auszeichnung.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP- oder KOP-Editor ist aktiv. Ein Element mit booleschem Ausgang ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.31 Befehl Weiterverschaltung festlegen

Symbol: 

**Funktion:** Der Befehl macht den selektierten Bausteinausgang zum weiterführenden Bausteinausgang.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP- oder KOP-Editor ist aktiv. Einer von mehreren Bausteinausgängen ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.32 Befehl Leitungsverzweigung einfügen

Symbol: 

**Funktion:** Der Befehl erstellt eine offene Leitungsverzweigung an der selektieren Leitung.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP- oder KOP-Editor ist aktiv. Ein Ein- oder Ausgang eines Bausteins ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL

- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: FUP/KOP/AWL-Element Leitungsverzweigung

### 5.13.33 Befehl Leitungsverzweigung oberhalb einfügen

Symbol: 

**Funktion:** Der Befehl erstellt eine Leitungsverzweigung oberhalb der selektierten offenen Leitungsverzweigung.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP- oder KOP-Editor ist aktiv. Eine offene Leitungsverzweigung ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: FUP/KOP/AWL-Element Leitungsverzweigung

### 5.13.34 Befehl Leitungsverzweigung unterhalb einfügen

Symbol: 

**Funktion:** Der Befehl erstellt eine Leitungsverzweigung unterhalb der selektierten offenen Leitungsverzweigung.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP- oder KOP-Editor ist aktiv. Eine offene Leitungsverzweigung ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: FUP/KOP/AWL-Element Leitungsverzweigung

### 5.13.35 Befehl Verzweigung Startpunkt setzen

Symbol: 

**Funktion:** Der Befehl setzt den Startpunkt einer Leitungsverzweigung an der selektieren Leitung.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv. Eine Leitung ist selektiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: Geschlossene Leitungsverzweigung

### 5.13.36 Befehl Verzweigung Endpunkt setzen

Symbol: 

**Funktion:** Der Befehl setzt den Endpunkt einer Leitungsverzweigung an der selektieren Leitung.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der KOP-Editor ist aktiv. Eine Leitung ist selektiert. Ein Startpunkt der Leitungsverzweigung wurde gesetzt.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: Geschlossene Leitungsverzweigung

### 5.13.37 Befehl Parallelen Modus wechseln

**Funktion:** Der Befehl wechselt eine parallele Leitungsverzweigung zwischen einem OR-Konstrukt und der Kurzschlussauswertung („Short Circuit Evaluation“ (SCE)).

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP/KOP/AWL-Editor ist aktiv. Eine senkrechte Linie einer parallelen Leitungsverzweigung ist selektiert.



Geschlossene parallele Verzweigungen in einem KOP-Netzwerk können Sie als „Short Circuit Evaluation“ (SCE) oder OR-Konstrukte programmieren. SCE-Zweige werden mit doppelten vertikalen Linien dargestellt, OR-Zweige mit einfachen. Sehen Sie hierzu die Hilfeseite zu „Geschlossenen Leitungsverzweigungen“.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor
- Dokumentation PLC: Geschlossene Leitungsverzweigung

### 5.13.38 Befehl Parameter aktualisieren

**Funktion:** Der Befehl pflegt Änderungen an der Deklaration des selektierten Elements in der Grafik ein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP-, KOP- oder CFC-Editor ist aktiv. Ein Baustein ist selektiert. Eine erweiternde Änderung an der Deklaration wurde vorgenommen.

Der Befehl prüft, ob ein Baustein und seine Deklaration im Deklarationseditor übereinstimmen. Die Änderung wird nur für den Baustein übernommen, wenn die Deklaration erweitert wurde. Löschungen und Überschreibungen werden nicht aktualisiert.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.39 Befehl Nicht verwendete FB-Aufruf-Parameter entfernen

Symbol: 

**Funktion:** Der Befehl löscht Eingänge und Ausgänge des selektierten Bausteins, denen keine Variable und kein Wert zugewiesen wurden. Die standardmäßigen Eingänge und Ausgänge des Bausteins bleiben aber immer erhalten.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP- oder KOP-Editor ist aktiv. Ein Baustein ist selektiert. Der Baustein verfügt über Schnittstellen, denen kein Wert zugewiesen ist.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.40 Befehl Baustein reparieren

Symbol: 

**Funktion:** Der Befehl repariert interne Inkonsistenzen am selektierten Baustein.

**Aufruf:** Menü **FUP/KOP/AWL**, Kontextmenü

**Voraussetzungen:** Der FUP- oder der KOP-Editor ist aktiv. Der fehlerhafte Baustein ist selektiert. Der Editor hat interne Inkonsistenzen im Programmierbaustein entdeckt, die eventuell automatisch aufgelöst werden können. Die Inkonsistenzen meldet TwinCAT in der Ansicht **Fehlerliste**.

Diese Situation ist denkbar beim Bearbeiten eines Projekts, das mit einer älteren Programmiersystemversion erstellt wurde, die die betreffende Inkonsistenz noch nicht als Fehler behandelte.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.41 Befehl Als Funktionsbausteinsprache anzeigen

**HINWEIS****Datenverlust**

Eine fehlerfreie Konvertierung setzt syntaktisch korrekten Code voraus. Anderenfalls können Teile der Implementierung verloren gehen.

**Funktion:** Der Befehl konvertiert die aktive Anweisungsliste oder den aktiven Kontaktplan in die Funktionsbausteinsprache.

**Aufruf:** Menü **FUP/KOP/AWL > Ansicht**

**Voraussetzungen:** Der KOP- oder der AWL-Editor ist aktiv.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.42 Befehl Als Kontaktplan anzeigen

**HINWEIS****Datenverlust**

Eine fehlerfreie Konvertierung setzt syntaktisch korrekten Code voraus. Anderenfalls können Teile der Implementierung verloren gehen.

**Funktion:** Der Befehl konvertiert das aktuelle Funktionsbausteincode oder die aktive Anweisungsliste in einen Kontaktplan.

**Aufruf:** Menü **FUP/KOP/AWL > Ansicht**

**Voraussetzungen:** Der FUP- oder AWL-Editor ist aktiv.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL

- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.43 Befehl Als Anweisungsliste anzeigen



AWL kann bei Bedarf über die TwinCAT-Optionen aktiviert werden.

#### HINWEIS

##### Datenverlust

Eine fehlerfreie Konvertierung setzt syntaktisch korrekten Code voraus. Anderenfalls können Teile der Implementierung verloren gehen.

**Funktion:** Der Befehl konvertiert den aktiven Funktionsbausteincode oder den aktiven Kontaktplan in eine Anweisungsliste.

**Aufruf:** Menü **FUP/KOP/AWL > Ansicht**

**Voraussetzungen:** Der KOP- oder FUP-Editor ist aktiv.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

### 5.13.44 Befehl Gehe zu

Symbol:

**Funktion:** Der Befehl lässt Sie zu einem beliebigen Netzwerk springen.

**Aufruf:** Menü **FUP/KOP/AWL**

**Voraussetzungen:** Der KOP-Editor, FUP-Editor oder der AWL-Editor ist aktiv. Ein Netzwerk ist selektiert.

Der Befehl öffnet einen Dialog mit einem Eingabefeld. Geben Sie die Nummer des gewünschten Netzwerks in das Eingabefeld ein.

**Siehe auch:**

- Dokumentation PLC: FUP/KOP/AWL
- Dokumentation PLC: FUP/KOP/AWL-Editor

## 5.14 Ladder-Editor

Die Befehle für das Arbeiten im Ladder-Editor sind im Menü **Ladder** sowie in den jeweiligen Kontextmenüs verfügbar.

### 5.14.1 Befehl Auskommentiert

Symbol:

**Funktion:** Der Befehl schaltet ein Netzwerk [► 243] zwischen den Zuständen auskommentiert und nicht auskommentiert um.

**Aufruf:** Menü **Ladder**, Kontextmenü

Der Netzwerkinhalt erscheint blass und die Texte in Kursivschrift. Das Netzwerk wird bei der Abarbeitung nicht berücksichtigt.

### 5.14.2 Befehl Negieren

**Symbol:**

**Funktion:** Der Befehl negiert folgende Elemente:

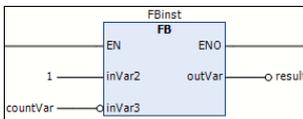
- Eingang/Ausgang eines Bausteins
- Sprung
- Return
- Spule
- Kontakt
- Variable

**Aufruf:** Menü **Ladder**, Kontextmenü

**Voraussetzungen:** Das Element, das negiert werden soll, ist selektiert.

**Beispiele:**

Negierter Eingang und negierter Ausgang an einem Funktionsbaustein:



Negierter Kontakt:



Negierte Spule:



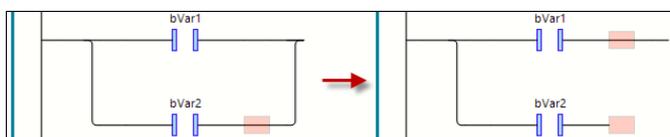
### 5.14.3 Befehl Parallele Verzweigung öffnen

**Symbol:**

**Funktion:** Der Befehl öffnet eine geschlossene parallele Leitungsverzweigung.

**Aufruf:** Menü **Ladder**, Kontextmenü

**Voraussetzungen:** Eine der beiden Leitungslinien der Verzweigung, die wieder geöffnet werden soll, muss selektiert sein.



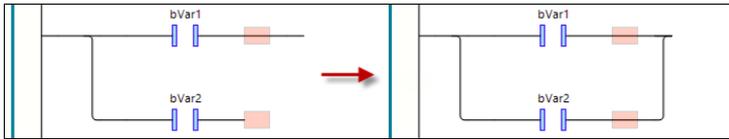
### 5.14.4 Befehl Parallele Verzweigung schließen

**Symbol:**

**Funktion:** Der Befehl schließt die offene parallele Leitungsverzweigung.

**Aufruf:** Menü **Ladder**, Kontextmenü

**Voraussetzungen:** Die beiden Leitungslinien der Verzweigung, die geschlossen werden soll, müssen selektiert sein:



Alternativ kann auch durch Ziehen der Selektionsmarkierung der einen Verzweigungslinie mit der Maus auf die Selektionsmarkierung der anderen Verzweigungslinie eine offene Verzweigung geschlossen werden.

### 5.14.5 Befehl Set/Reset - Set, Set/Reset - Reset

**Symbol:** Set , Reset 

**Funktion:** Der Befehl Set versieht das im Editor selektierte Element mit einem Set-Modifizierer. Der Befehl Reset versieht das Element mit einem Reset-Modifizierer.

Beispielsweise können Sie damit eine [Spule](#) [▶ 244] und einen nicht-primären Bausteinausgang zu einer "Set-Spule" machen. Der Befehl ist nur an den passenden Positionen verfügbar.

**Aufruf:** Menü **Ladder** - Untermenü **Set/Reset**, Kontextmenü

### 5.14.6 Befehl Flankenerkennung Steigende Flanke

**Symbol:** 

**Funktion:** Der Befehl fügt eine Erkennung für eine steigende Flanke vor dem selektierten Bausteineingang oder Kontakt ein.

**Aufruf:** Menü **Ladder**, Kontextmenü

**Voraussetzungen:** Ein Bausteineingang oder Kontakt ist selektiert.

### 5.14.7 Befehl Flankenerkennung: Fallende Flanke

**Symbol:** 

**Funktion:** Der Befehl fügt eine Erkennung für eine fallende Flanke vor dem selektierten Bausteineingang oder Kontakt ein.

**Aufruf:** Menü **Ladder**, Kontextmenü

**Voraussetzungen:** Ein Bausteineingang oder Kontakt ist selektiert.

### 5.14.8 Befehl EN/ENO: EN

**Symbol:** 

**Funktion:** Der Befehl dient dem Hinzufügen oder Entfernen eines booleschen Eingangs "Enable" am selektierten Baustein.

**Aufruf:** Menü **Ladder** - Untermenü **EN/ENO**, Kontextmenü

Mit dem booleschen Eingang EN wird die Ausführung des Bausteins gesteuert: Wenn der Eingang EN zum Zeitpunkt des Bausteinaufrufs den Wert FALSE hat, werden die im Baustein definierten Operationen nicht ausgeführt. Wenn EN den Wert TRUE hat, werden diese Operationen ausgeführt.

Wenn ein Baustein einen EN-Eingang hat, kann ihm auch ein [ENO-Ausgang](#) [▶ 243] hinzugefügt werden: ENO erhält denselben Wert wie EN.

## 5.14.9 Befehl EN/ENO: ENO

**Symbol:** 

**Funktion:** Der Befehl fügt einem Baustein mit EN-Eingang einen ENO-Ausgang hinzu.

**Aufruf:** Menü **Ladder** - Untermenü **EN/ENO**, Kontextmenü

**Voraussetzungen:** Ein Baustein, der einen EN-Eingang hat, ist selektiert.

Der ENO-Ausgang erhält denselben Wert wie der EN-Eingang.

## 5.14.10 Befehl Netzwerk einfügen

**Symbol:** 

**Funktion:** Der Befehl fügt ein weiteres Netzwerk im Ladder-Editor ein. Sie erkennen die möglichen Einfügepositionen am Plusymbol am Cursor , wenn Sie das Element mit der Maus über den Implementierungsteil ziehen.

**Aufruf:** Menü **Ladder**, Kontextmenü

**Voraussetzungen:** Kein Baustein ist selektiert.

### 5.14.10.1 Element: Netzwerk

Ein Netzwerk ist die Basiseinheit eines KOP-Programms. Im Ladder-Editor sind die Netzwerke in einer Liste untereinander angeordnet. Jedes Netzwerk ist an der linken Seite mit einer fortlaufenden Netzwerknummer versehen und kann Folgendes enthalten: Logische und arithmetische Ausdrücke, Programm-, Funktions- und Funktionsbausteinaufrufe, eine Sprung- oder Return-Anweisung.

Sie können jedes Netzwerk mit einem Titel, einem Kommentar oder einer Sprungmarke versehen:

Netzwerktitel: Doppelklicken Sie zur Eingabe auf die erste Zeile oben in einem Netzwerk.

Netzwerkcommentar: Doppelklicken Sie zur Eingabe auf die zweite Zeile oben in einem Netzwerk.

Sprungmarke: Doppelklicken Sie zur Eingabe auf die dritte Zeile oben in einem Netzwerk. Die Sprungmarke kann dann beim Element Sprung [\[► 245\]](#) als Ziel angegeben werden.

In den TwinCAT-Optionen, Kategorie Ladder [\[► 183\]](#) definieren Sie, ob Netzwerktitel, Netzwerkcommentar und Trennlinien zwischen den einzelnen Netzwerken im Editor angezeigt werden.

## 5.14.11 Befehl Kontakt einfügen

**Symbol:** , im Editor 

**Funktion:** Der Befehl fügt einen Kontakt im Netzwerk, links vom selektierten Element ein. Wenn Sie das Element aus der Werkzeugbox in den Implementierungsteil ziehen, erkennen Sie die möglichen

Einfügepositionen am Plusymbol am Cursor .

**Aufruf:** Menü **Ladder**, Kontextmenü

**Voraussetzungen:** Eine Leitung oder ein Kontakt ist selektiert.

### 5.14.11.1 Element: Kontakt

Ein Kontakt gibt das Signal TRUE (ON) oder FALSE (OFF) von links nach rechts weiter, bis das Signal eine Spule im rechten Teil des Netzwerks erreicht. Zu diesem Zweck wird dem Kontakt eine boolesche Variable zugewiesen, die das Signal enthält. Dazu ersetzen Sie den Platzhalter **???** oberhalb des Kontakts mit dem Namen einer booleschen Variablen.

Sie können mehrere Kontakte sowohl in Reihe als auch parallel anordnen. Bei zwei parallelen Kontakten muss nur einer den Wert TRUE erhalten, damit ON nach rechts weitergegeben wird. Wenn Kontakte hintereinandergeschaltet sind, müssen alle Kontakte den Wert TRUE erhalten, damit vom letzten Kontakt der Reihe ON nach rechts weitergegeben wird. Sie können also mit KOP elektrische Parallel- und Reihenschaltungen programmieren.

Ein negierter Kontakt  gibt das Signal TRUE weiter, wenn der Variablenwert FALSE ist. Sie können die Negation eines eingefügten Kontakts mit Hilfe des Befehls **Ladder** → **Negieren** aktivieren oder deaktivieren.

## 5.14.12 Befehl Spule einfügen

**Symbol:** 

**Symbol im Editor:** 

**Funktion:** Der Befehl fügt eine Spule im Netzwerk ein. Wenn Sie das Element aus der Werkzeugbox in den Implementierungsteil ziehen, erkennen Sie die möglichen Einfügepositionen am Plussymbol am Cursor .

**Aufruf:** Menü **Ladder**, Kontextmenü

**Voraussetzungen:** Ein Netzwerk, eine Spule oder ein Verbinder ist selektiert. Kein Baustein ist selektiert.

### 5.14.12.1 Element: Spule

Eine Spule nimmt den Wert auf, der von links geliefert wird und speichert ihn in der ihr zugewiesenen booleschen Variablen. Ihr Eingang kann den Wert TRUE (ON) oder FALSE (OFF) haben.

In einer negierten [[▶ 241](#)] Spule  wird der negierte Wert des eingehenden Signals in der booleschen Variablen gespeichert, die der Spule zugewiesen ist.

Mit den Befehlen des Untermenüs Set/Reset [[▶ 242](#)] können Sie eine "Set-Spule" oder "Reset-Spule" definieren:

**Set-Spule:** Wenn der Wert TRUE an einer Set-Spule ankommt, behält die Spule den Wert TRUE. Solange die Applikation läuft, kann der Wert an dieser Stelle nicht mehr überschrieben werden.

**Reset-Spule:** Wenn der Wert TRUE an einer Reset-Spule ankommt, behält die Spule den Wert FALSE. Solange die Applikation läuft, kann der Wert an dieser Stelle nicht mehr überschrieben werden.

## 5.14.13 Befehl Baustein einfügen

**Symbol:** 

**Funktion:** Der Befehl fügt einen Baustein am Ende des selektierten Netzwerks ein. Wenn Sie das Element aus der Werkzeugbox in den Implementierungsteil ziehen, erkennen Sie die möglichen Einfügepositionen am Plussymbol am Cursor .

**Aufruf:** Menü **Ladder**, Kontextmenü

Der Befehl fügt standardmäßig einen EN/ENO-Baustein ein. Mit den Befehlen des Menüs **Ladder** EN/ENO können die Modifizierer EN [[▶ 242](#)] und ENO [[▶ 243](#)] entfernt und hinzugefügt werden.

**AKTUELL NOCH NICHT IMPLEMENTIERT:** Wenn es sich um einen im Projekt definierten Baustein handelt, muss nach einer Änderung an diesem Baustein die Verwendung im Ladder aktualisiert werden.

### 5.14.13.1 Element: Baustein

Ein Bausteinelement kann zusätzliche Funktionen repräsentieren: Beispielsweise IEC-Funktionsbausteine, IEC-Funktionen, Bibliotheksbausteine, Operatoren.

Wenn der Baustein eine Bilddatei mitliefert, kann das Bausteinsymbol innerhalb des Bausteins angezeigt werden. Voraussetzung: In den TwinCAT-Optionen, Kategorie **Ladder** ist die Option **Bausteinsymbol anzeigen** aktiviert.

## 5.14.14 Befehl Sprung einfügen

**Symbol:** 

**Funktion:** Der Befehl fügt ein Sprungelement vor dem selektierten Element ein. Wenn Sie das Element aus der Werkzeugbox in den Implementierungsteil ziehen, erkennen Sie die möglichen Einfügepositionen am

Plussymbol am Cursor .

**Aufruf:** Menü **Ladder**, Kontextmenü

**Voraussetzungen:** Ein Verbinder ist selektiert.

### 5.14.14.1 Element: Sprung

In KOP wird ein Sprung abhängig von der aktuellen Cursorposition entweder direkt vor einem Eingang, direkt nach einem Ausgang oder am Ende des Netzwerks eingefügt.

Direkt hinter dem Sprung-Element geben Sie die Sprungmarke eines Netzwerks als Sprungziel ein.

## 5.14.15 Befehl Return einfügen

**Symbol:** 

**Funktion:** Der Befehl fügt ein Element **Return** an der selektierten Stelle ein. Wenn Sie das Element aus der Werkzeugbox in den Implementierungsteil ziehen, erkennen Sie die möglichen Einfügepositionen am

Plussymbol am Cursor .

**Aufruf:** Menü **Ladder**, Kontextmenü

### 5.14.15.1 Element: Return

Das Element dient dazu, die Abarbeitung des Ladder-Bausteins sofort abubrechen, wenn der Eingang des **Return**-Elements TRUE wird. Sie können die **Return**-Anweisung parallel zu oder anschließend an die vorausgehenden Elemente platzieren.

## 5.14.16 Befehl Eingang einfügen

**Symbol:** 

**Funktion:** Der Befehl fügt einen Eingang an der gewählten Position ein. Wenn Sie das Element aus der Werkzeugbox in den Implementierungsteil ziehen, erkennen Sie die möglichen Einfügepositionen am

Plussymbol am Cursor . Die zunächst angezeigten ??? können Sie mit einem Variablennamen oder einer Konstanten ersetzen.

**Aufruf:** Menü **Ladder**, Kontextmenü

### 5.14.16.1 Element: Eingang

Das Element dient der Zuweisung eines Variablenwerts zu einem anderen Element im Signalfluss. Es hat eine weiterführende Linie nach rechts.

## 5.14.17 Befehl Ausgang einfügen

**Symbol:** 

**Funktion:** Der Befehl fügt einen Ausgang an der gewählten Position ein. Wenn Sie das Element aus der Werkzeugbox in den Implementierungsteil ziehen, erkennen Sie die möglichen Einfügepositionen am

Plussymbol am Cursor . Die zunächst angezeigten ??? können Sie mit einem Variablennamen oder einer Konstanten ersetzen.

**Aufruf:** Menü **Ladder**, Kontextmenü

### 5.14.17.1 Element: Ausgang

Das Element dient der Zuweisung eines von links kommenden Signals zu der mit dem Element verknüpften Variablen. Es hat eine von links kommende Eingangslinie.

## 5.14.18 Befehl In neuen Ladder konvertieren

Verfügbar im FUP/KOP-Editor.

## 5.15 Deklarationen

### 5.15.1 Befehl Einfügen

**Symbol:** 

**Funktion:** Der Befehl fügt eine neue Zeile für eine Variablendeklaration im tabellarischen Deklarationseditor ein und das Eingabefeld für den Variablennamen öffnet sich.

**Aufruf:** Schaltfläche im Deklarationskopf des tabellarischen Deklarationseditors, Kontextmenü im tabellarischen Deklarationseditor

Um die anderen Felder der Deklarationszeile zu bearbeiten, doppelklicken Sie die Felder und wählen Sie die Angaben aus den Auswahllisten oder mithilfe der entsprechenden Dialoge.

**Siehe auch:**

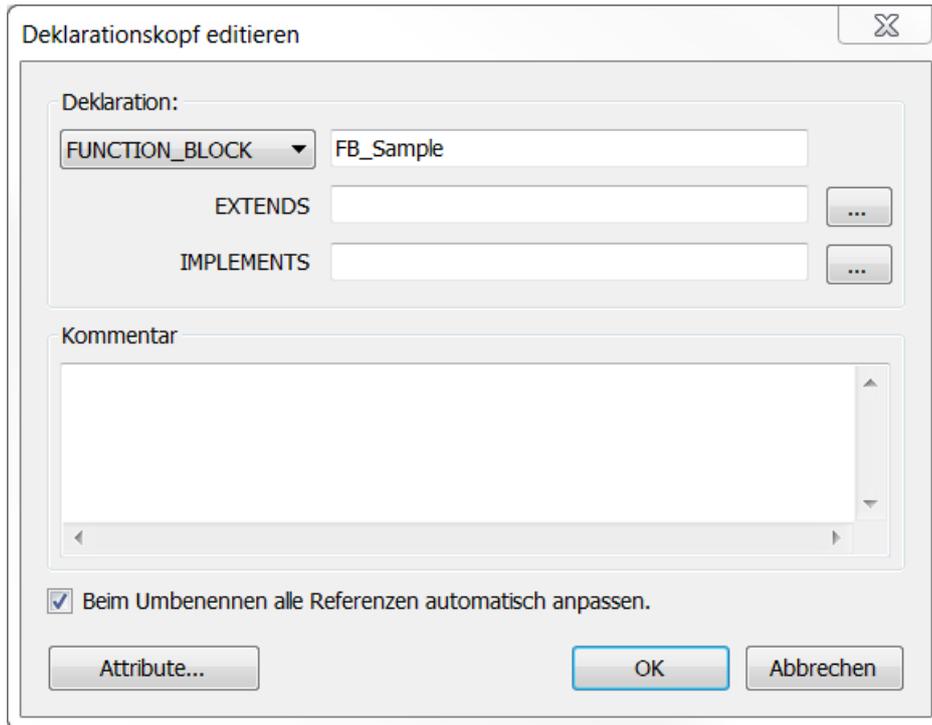
- Dokumentation PLC: Deklarationseditor verwenden

### 5.15.2 Befehl Deklarationskopf editieren

**Funktion:** Der Befehl öffnet den Dialog **Deklarationskopf editieren**, der im tabellarischen Deklarationseditor zur Konfiguration der Kopfzeile einer POU dient.

**Aufruf:** Mausklick auf die Kopfleiste des tabellarischen Deklarationseditors, Kontextmenü im tabellarischen Deklarationseditor

Dialog Deklarationskopf editieren



Deklaration	<p>Auswahlliste zur Änderung des POU-Typs</p> <ul style="list-style-type: none"> <li>• PROGRAM</li> <li>• FUNCTION_BLOCK             <ul style="list-style-type: none"> <li>◦ EXTENDS: Eingabefeld für einen Basis-Funktionsbaustein</li> <li>◦ IMPLEMENTS: Eingabefeld für eine Schnittstelle</li> </ul> </li> <li>• FUNCTION             <ul style="list-style-type: none"> <li>◦ Rückgabebetyp</li> </ul> </li> </ul> <p>Eingabefeld mit aktuellem POU-Namen: Sie können den Namen der POU ändern.</p>
Beim Umbenennen alle Referenzen automatisch anpassen	<p><input checked="" type="checkbox"/> : Dialog <b>Refactoring</b> öffnet sich.</p> <p><input type="checkbox"/> : Die Umbenennung wird nur im Deklarationskopf der POU wirksam.</p>
Attribute	Der Dialog <b>Attribute</b> öffnet sich zur Eingabe von Attributen und Pragmas.

Siehe auch:

- Dokumentation PLC: Deklarationseditor verwenden
- Dokumentation PLC: Pragmas
- Dokumentation PLC: Refactoring

### 5.15.3 Befehl Nach unten verschieben

Symbol: 

**Funktion:** Der Befehl verschiebt eine Variablendeklaration um eine Zeile nach unten.

**Aufruf:** Schaltfläche im Deklarationskopf des tabellarischen Deklarationseditors, Kontextmenü im tabellarischen Deklarationseditor

**Voraussetzung:** Im tabellarischen Deklarationseditor ist eine Zeile mit einer Variablendeklaration selektiert.

Siehe auch:

- Dokumentation PLC: Deklarationseditor verwenden

## 5.15.4 Befehl Nach oben verschieben

Symbol: 

**Funktion:** Der Befehl verschiebt eine Variablendeklaration um eine Zeile nach oben.

**Aufruf:** Schaltfläche im Deklarationskopf des tabellarischen Deklarationseditors, Kontextmenü im tabellarischen Deklarationseditor

**Voraussetzung:** Im tabellarischen Deklarationseditor ist eine Zeile mit einer Variablendeklaration selektiert.

**Siehe auch:**

- Dokumentation PLC: Deklarationseditor verwenden

## 5.16 Textliste

### 5.16.1 Befehl Sprache einfügen

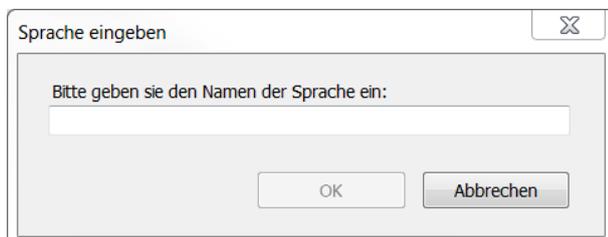
Symbol: 

**Funktion:** Der Befehl fügt eine weitere Sprachspalte der Textliste hinzu.

**Aufruf:** Menü **Textliste**, Kontextmenü

**Voraussetzung:** Eine Textliste oder eine globale Textliste ist geöffnet und aktiv.

Geben Sie im Dialog **Sprache eingeben** ein Kürzel für die neue Sprache ein, zum Beispiel „en-US“. TwinCAT fügt das Kürzel als Spaltenüberschrift ein.



**Siehe auch:**

- Dokumentation PLC: Text in einer Textliste verwalten

### 5.16.2 Befehl Sprache entfernen

Symbol: 

**Funktion:** Der Befehl entfernt die ausgewählte Sprachspalte aus der Textliste.

**Aufruf:** Menü **Textliste**, Kontextmenü

**Voraussetzung:** Eine Textliste oder eine globale Textliste ist geöffnet und aktiv. Ein Feld in der Spalte der Sprache, die Sie entfernen wollen, ist selektiert.

**Siehe auch:**

- Dokumentation PLC: Text in einer Textliste verwalten

### 5.16.3 Befehl Text einfügen

Symbol:

**Funktion:** Der Befehl fügt eine neue Zeile oberhalb der selektieren Zeile in die Textliste ein. Ein Eingabefeld unter Standard öffnet sich, in dem Sie den Ausgangstext eingeben.

**Aufruf:** Menü **Textliste**

**Voraussetzung:** Eine Textliste, keine GlobalTextList, ist geöffnet und aktiv. Ein Feld in der Tabelle ist selektiert.

**Siehe auch:**

- Dokumentation PLC: Text in einer Textliste verwalten

### 5.16.4 Befehl Import/Export Textlisten

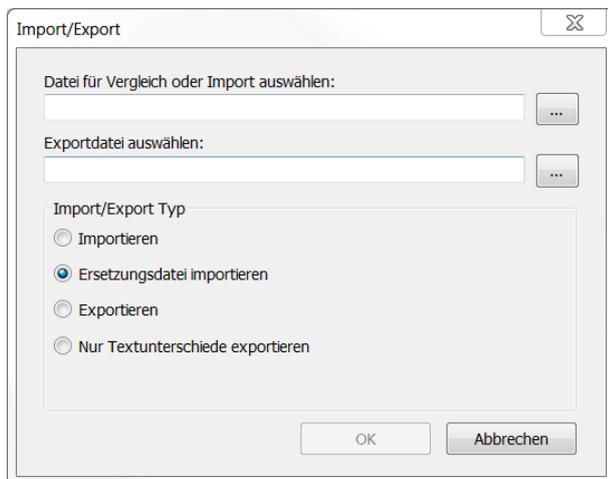
Symbol:

**Funktion:** Der Befehl exportiert eine aktive Textliste, importiert eine Datei oder gleicht eine Textliste mit einer Datei ab. Die Datei hat CSV-Format. Der Dialog **Import/Export** bietet dafür Optionen an.

**Aufruf:** Menü **Textliste**, Kontextmenü

**Voraussetzung:** Eine Textliste oder eine globale Textliste ist geöffnet und aktiv.

#### Dialog Import/Export



Datei für Vergleich oder Import auswählen	Datei, die TwinCAT ausliest. öffnet den Dialog <b>Textlistendatei auswählen</b> , in dem Sie eine Datei auswählen können.
Exportdatei auswählen	Datei, in die TwinCAT schreibt. öffnet den Dialog <b>Textlistendatei auswählen</b> , in dem Sie ein Datei und Verzeichnis auswählen können.

**Import/Export Typ:**

Importieren	<p>Voraussetzung: In <b>Datei für Vergleich oder Import auswählen</b> ist eine Datei ausgewählt.</p> <p>Die Datei kann Textlisteneinträge sowohl für die globale Textliste als auch für Textlisten enthalten.</p> <p>Globale Textliste:</p> <ul style="list-style-type: none"> <li>• TwinCAT liest die Datei, vergleicht für gleichen Ausgangstext die Textlisteneinträge und übernimmt Unterschiede bei den Übersetzungen. TwinCAT überschreibt gegebenenfalls die Übersetzungen im Projekt.</li> </ul> <p>Textlisten:</p> <ul style="list-style-type: none"> <li>• TwinCAT liest die Datei, vergleicht für gleiche IDs die Textlisteneinträge und übernimmt Unterschiede im Ausgangstext und den Übersetzungen in das Projekt. TwinCAT überschreibt gegebenenfalls die Textlisteneinträge im Projekt.</li> <li>• Wenn die Datei eine neue ID enthält, wird der Textlisteneintrag in die Textliste des Projekts importiert und die Textliste ergänzt.</li> </ul>
Ersetzungsdatei importieren	<p>Voraussetzung: In <b>Datei für Vergleich oder Import auswählen</b> ist eine Ersetzungsdatei ausgewählt.</p> <p>Die Ersetzungsdatei enthält Ersetzungen für die globale Textliste.</p> <p>TwinCAT arbeitet die Ersetzungsdatei zeilenweise ab und führt die spezifizierten Ersetzungen in der globalen Textliste durch. Der Aufbau der Ersetzungsdatei ist im Abschnitt Statischen Text in einer globalen Textliste verwalten beschrieben.</p>
Exportieren	<p>Voraussetzung: In „Exportdatei auswählen“ ist die Datei, in die TwinCAT schreibt, ausgewählt.</p> <p>TwinCAT exportiert alle Texte aus allen Textlisten des aktuellen Projekts. In der Exportdatei werden alle im Projekt vorhandenen Sprachen als Spalten eingefügt. Die Datei kann dazu verwendet werden die sprachenabhängigen Texte extern übersetzen zu lassen.</p>
Nur Textunterschiede exportieren	<p>Voraussetzung: In <b>Datei für Vergleich oder Import auswählen</b> ist eine Importdatei für den Vergleich ausgewählt. In <b>In Exportdatei auswählen</b> ist eine Exportdatei, in die TwinCAT schreibt, ausgewählt.</p> <p>TwinCAT liest die Importdatei, vergleicht die Zeilen der aktiven Textliste damit. Wenn Zeilen übereinstimmen, ignoriert TwinCAT diese. Wenn Zeilen sich unterscheiden, schreibt TwinCAT die Zeile in die Exportdatei und übernimmt dabei gegebenenfalls Übersetzungen aus der Textliste. TwinCAT übernimmt die Übersetzungen aus der Importdatei und überschreibt Sie gegebenenfalls.</p>

**Siehe auch:**

- Dokumentation PLC: Text in einer Textliste verwalten

## 5.16.5 Befehl Nicht verwendete Textlisteneinträge entfernen

Symbol: 

**Funktion:** Der Befehl prüft, ob ein Textlisteneintrag im Projekt als statischer Text verwendet wird. Wenn nicht, entfernt TwinCAT ihn aus der Textliste.

**Aufruf:** Menü **Textliste**, Kontextmenü

**Voraussetzung:** Die globale Textliste ist geöffnet und aktiv. Ein Feld in der Tabelle ist selektiert.

**Siehe auch:**

- Dokumentation PLC: Text in einer Textliste verwalten

## 5.16.6 Befehl Visualisierungstext-IDs prüfen

Symbol: 

**Funktion:** Der Befehl prüft, ob die ID eines Textlisteneintrags im Projekt korrekt ist und meldet das Ergebnis.

**Aufruf:** Menü **Textliste**, Kontextmenü

**Voraussetzung:** Die globale Textliste ist geöffnet und aktiv. Ein Feld in der Tabelle ist selektiert.

Wenn TwinCAT beim Prüfen feststellt, dass die globale Textliste und die statischen Texte der Visualisierungen nicht übereinstimmen, kann das daran liegen, dass die globale Textliste schreibgeschützt ist oder war. Voraussetzung dafür ist, dass Sie eine Benutzerverwaltung im Projekt eingerichtet haben.

**Siehe auch:**

- Dokumentation PLC: Text in einer Textliste verwalten

## 5.16.7 Befehl Visualisierungstext-IDs aktualisieren

Symbol: 

**Funktion:** Der Befehl aktualisiert alle inkonsistenten IDs in einer statischen Textliste.

**Aufruf:** Menü **Textliste**, Kontextmenü

**Voraussetzung:** Die globale Textliste ist geöffnet und aktiv. Ein Feld in der Tabelle ist selektiert. Das Objekt ist schreibgeschützt.

Wenn TwinCAT beim Prüfen feststellt, dass die globale Textliste und die statischen Texte der Visualisierungen nicht übereinstimmen, kann das daran liegen, dass die globale Textliste unter Schreibschutz steht oder stand. Voraussetzung dafür ist, dass Sie eine Benutzerverwaltung im Projekt eingerichtet haben.

**Siehe auch:**

- Dokumentation PLC: Text in einer Textliste verwalten

## 5.16.8 Befehl Alles als Text exportieren

Symbol: 

**Funktion:** Der Befehl exportiert alle Textlisten des Projekts.

**Aufruf:** Menü **Textliste**, Kontextmenü

**Voraussetzung:**

- Eine Textliste oder eine globale Textliste ist geöffnet und aktiv.
- Die Visualisierung codiert die Zeichen der Texte nicht in Unicode.

TwinCAT erstellt dabei für jede Textliste eine Datei als einfachen Text im Format .txt. Der Name der Textliste wird zum Namen der Datei. Die Datei wird in das Verzeichnis des TwinCAT Projekts gespeichert.

Eine Steuerung kann dieses Format lesen und verwenden. Sie können die Datei zum Beispiel auf eine Steuerung kopieren und über eine Einstellung im Visualisierungsmanager konfigurieren, dass beim Laden des SPS-Projekts die Textlisten nicht nochmals übertragen werden.

**Siehe auch:**

- Dokumentation PLC: Text in einer Textliste verwalten

## 5.16.9 Befehl Alles als Unicode-Text exportieren

Symbol: 

**Funktion:** Der Befehl exportiert alle Textlisten des Projekts.

**Aufruf:** Menü **Textliste**, Kontextmenü

**Voraussetzung:**

- Eine Textliste oder eine globale Textliste ist geöffnet und aktiv.
- Die Visualisierung codiert die Zeichen der Texte in Unicode.
  - Die Option **Unicode-Zeichenfolgen verwenden** im Visualisierungsmanager ist aktiviert.

TwinCAT erstellt dabei für jede Textliste eine Datei als einfachen Text im Format .txt. Der Name der Textliste wird zum Namen der Datei. Die Datei wird in das Verzeichnis des TwinCAT Projekts gespeichert.

Eine Steuerung kann dieses Format lesen und verwenden. Sie können die Datei zum Beispiel auf eine Steuerung kopieren und über eine Einstellung im Visualisierungsmanager konfigurieren, dass beim Laden des SPS-Projekts die Textlisten nicht nochmals übertragen werden.

**Siehe auch:**

- Dokumentation PLC: Text in einer Textliste verwalten

## 5.16.10 Befehl Textlistenunterstützung hinzufügen

Symbol: 

**Funktion:** Der Befehl fügt dem selektierten DUT-Objekt des Typs **Enumeration** eine Textlistenunterstützung hinzu.

**Aufruf:** Kontextmenü eines Standard-DUT-Objekts vom Typ **Enumeration** (  )

Die Textlistenunterstützung ermöglicht eine Lokalisierung der Enumerationskomponentenbezeichner und eine Darstellung des symbolischen Komponentenwerts in einer Textausgabe in der Visualisierung.

**Siehe auch:**

- Dokumentation PLC: Objekt DUT
- [Befehl Textlistenunterstützung entfernen \[► 252\]](#)

## 5.16.11 Befehl Textlistenunterstützung entfernen

Symbol: 

**Funktion:** Der Befehl entfernt die Textlistenunterstützung vom selektierten Enumerationsobjekt.

**Aufruf:** Kontextmenü eines Objekts einer textlistenunterstützten Enumeration (  ).

Die Textlistenunterstützung ermöglicht eine Lokalisierung der Enumerationskomponentenbezeichner und eine Darstellung des symbolischen Komponentenwerts in einer Textausgabe in der Visualisierung.

## 5.17 Rezepturen

### 5.17.1 Befehl Rezeptur einfügen

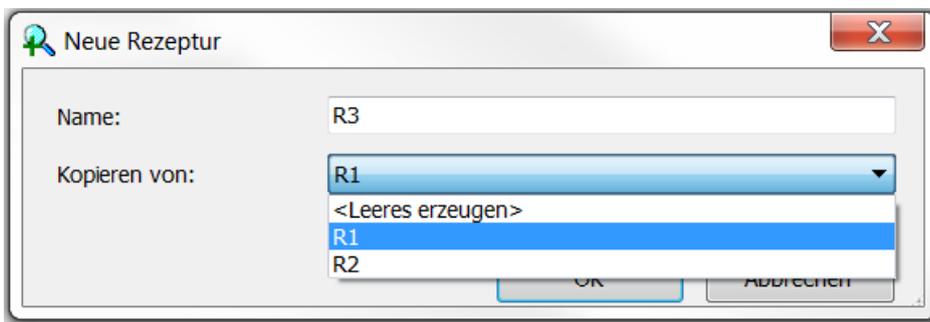
Symbol: 

**Funktion:** Der Befehl öffnet einen Dialog zum Einfügen einer neuen Rezeptur (einer neue Spalte) in die Rezepturdefinition.

**Aufruf:** Menü **Rezepturen**, Kontextmenü

**Voraussetzung:** Eine Rezepturdefinition ist im Editor geöffnet.

Wenn Sie den Befehl ausführen, öffnet sich ein Dialog, in dem Sie den Namen der neuen Rezeptur festlegen. Der Dialog bietet Ihnen auch die Möglichkeit, bestehende Rezepturen in die neue Rezeptur zu kopieren.



**Siehe auch:**

- Dokumentation PLC: Werte ändern mit Rezepturen

### 5.17.2 Befehl Rezeptur entfernen

Symbol: 

**Funktion:** Der Befehl löscht eine Rezeptur aus der gerade geöffneten Rezepturdefinition.

**Aufruf:** Menü **Rezepturen**, Kontextmenü

**Voraussetzung:** Ein Feld in der Rezepturspalte einer Rezepturdefinition ist selektiert.

**Siehe auch:**

- Dokumentation PLC: Werte ändern mit Rezepturen

### 5.17.3 Befehl Rezeptur laden

Symbol: 

**Funktion:** Der Befehl lädt eine Rezeptur aus einer Datei.

**Aufruf:** Menü **Rezepturen**, Kontextmenü

**Voraussetzung:** Ein Feld in der Rezepturspalte einer Rezepturdefinition ist selektiert.

Wenn Sie den Befehl ausführen, öffnet sich der Standarddialog zur Auswahl einer Datei. Der Filter ist automatisch auf die Dateierweiterung \*.txtrecipe eingestellt. Nach dem Laden werden die Werte der selektierten Rezeptur der Rezepturdefinition überschrieben und die Anzeige aktualisiert.

Wenn Sie nur einzelne Variablen der Rezeptur mit neuen Werten überschreiben möchten, entfernen Sie vor dem Laden in der Rezepturdatei die Werte für die restlichen Variablen. Einträge ohne Wertangabe werden nicht eingelesen und somit bleiben diese Variablen auf der Steuerung und im Projekt durch die Aktualisierung unberührt. Im Folgenden sehen Sie das Beispiel für die Einträge einer Rezepturdatei, bei deren Laden nur MAIN.nVar mit einem neuen Wert (6) geschrieben wird:

```
MAIN.nVar1:=
MAIN.nVar2:=6
MAIN.nVar3:=
MAIN.sVar4:=
MAIN.wsVar5:=
```

Bei Werten vom Datentyp REAL/LREAL wird in manchen Fällen auch der Hexadezimalwert in die Rezepturdatei geschrieben. Dies ist notwendig, damit bei der Rückkonvertierung der exakt identische Wert wiederhergestellt wird. In diesem Fall ändern Sie den Dezimalwert und löschen Sie den Hexadezimalwert.

#### Siehe auch:

- Dokumentation PLC: Werte ändern mit Rezepturen

## 5.17.4 Befehl Rezeptur speichern

Symbol: 

**Funktion:** Der Befehl speichert die Werte der Variablen einer Rezeptur in eine Datei.

**Aufruf:** Menü **Rezepturen**, Kontextmenü

**Voraussetzung:** Ein Wert der Rezeptur in der Rezepturdefinition ist selektiert.

Beim Ausführen des Befehls speichert TwinCAT die Werte der selektierten Rezeptur in eine Datei mit der Erweiterung \*.txtrecipe, deren Name definiert werden muss. Dazu öffnet der Standarddialog zum Speichern einer Datei. Das Format ergibt sich aus den Einstellungen des Rezepturverwalters in der Registerkarte **Speicherung**.

### ● Überschreiben der impliziten Rezepturdatei

**i** Die implizit verwendeten Rezepturdateien, die als Zwischenablage beim Lesen und Schreiben von Rezepturen nötig sind, dürfen nicht überschrieben werden. D. h. der Name für die Datei muss anders lauten als <Rezepturname>.<Rezepturdefinitionsname>.txtrecipe

#### Siehe auch:

- Dokumentation PLC: Werte ändern mit Rezepturen

## 5.17.5 Befehl Rezeptur lesen

Symbol: 

**Funktion:** Der Befehl liest die Werte der Variablen einer Rezeptur aus der Steuerung.

**Aufruf:** Menü **Rezepturen**, Kontextmenü

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb und ein Wert der Rezeptur in der Rezepturdefinition ist selektiert.

Beim Ausführen des Befehls überschreibt TwinCAT die Werte der selektierten Rezeptur mit den gelesenen Werten aus der Steuerung. Dabei werden die Werte implizit gespeichert (in einer Datei auf der Steuerung) und gleichzeitig in der Tabelle der Rezepturdefinition angezeigt.

#### Siehe auch:

- Dokumentation PLC: Werte ändern mit Rezepturen

## 5.17.6 Befehl Rezeptur schreiben

Symbol: 

**Funktion:** Der Befehl schreibt die Werte einer Rezeptur in die Variablen in der Steuerung.

**Aufruf:** Menü **Rezepturen**, Kontextmenü

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb und ein Wert der Rezeptur in der Rezepturdefinition ist selektiert.

Beim Ausführen des Befehls überschreibt TwinCAT die Werte in der Steuerung mit den Werten der selektierten Rezeptur.

**Siehe auch:**

- Dokumentation PLC: Werte ändern mit Rezepturen

## 5.17.7 Befehl Rezeptur laden und schreiben

Symbol: 

**Funktion:** Der Befehl lädt eine Rezeptur aus einer Datei und schreibt die Werte auf die Variablen in der Steuerung.

**Aufruf:** Menü **Rezepturen**, Kontextmenü

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb und ein Wert der Rezeptur in der Rezepturdefinition ist selektiert.

Nach dem Ausführen des Befehls werden Sie gefragt, ob die Werte aus der Datei auch in die Rezeptur im Projekt, oder nur zur SPS geschrieben werden sollen. Ein Aktualisieren der Werte in der Rezeptur kann bei erneutem Einloggen einen Online Change notwendig machen.

Wenn Sie den Befehl ausführen, überschreibt TwinCAT die Werte der selektierten Rezeptur der Rezepturdefinition. Weiterhin werden die Werte der Variablen in der Steuerung mit diesen Rezepturwerten überschrieben.

Wenn Sie nur einzelne Variablen der Rezeptur mit neuen Werten überschreiben möchten, entfernen Sie vor dem Laden in der Rezepturdatei die Werte für die restlichen Variablen. Einträge ohne Wertangabe werden nicht eingelesen und somit bleiben diese Variablen auf der Steuerung und im Projekt durch die Aktualisierung unberührt. Im Folgenden sehen Sie das Beispiel für die Einträge einer Rezepturdatei, bei deren Laden nur MAIN.nVar1 mit einem neuen Wert (6) geschrieben wird.

```
MAIN.nVar1:=  
MAIN.nVar2:=6  
MAIN.nVar3:=  
MAIN.sVar4:=  
MAIN.wstVar5:=
```

Bei Werten vom Datentyp REAL/LREAL wird in manchen Fällen auch der Hexadezimalwert in die Rezepturdatei geschrieben. Dies ist notwendig, damit bei der Rückkonvertierung der exakt identische Wert wiederhergestellt wird. In diesem Fall ändern Sie den Dezimalwert und löschen Sie den Hexadezimalwert.

**Siehe auch:**

- Dokumentation PLC: Werte ändern mit Rezepturen

## 5.17.8 Befehl Rezeptur lesen und speichern

Symbol: 

**Funktion:** Der Befehl liest die Werte der Variablen einer Rezeptur aus der Steuerung und speichert sie in eine Datei.

**Aufruf:** Menü **Rezepturen**, Kontextmenü

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb und ein Wert der Rezeptur in der Rezepturdefinition ist selektiert.

Nach dem Ausführen des Befehls werden Sie gefragt, ob die Variablenwerte in die Rezeptur im Projekt eingelesen oder nur gespeichert werden sollen. Ein Aktualisieren der Werte in der Rezeptur könnte bei erneutem Einloggen einen Online-Change notwendig machen.

Die Werte werden gemäß den Einstellungen des Rezepturverwalters (Registerkarte **Speicherung**) mit dem Standardnamen für Rezepturdateien abgespeichert.

**Siehe auch:**

- Dokumentation PLC: Werte ändern mit Rezepturen

### 5.17.9 Befehl Variable einfügen

Symbol: 

**Funktion:** Der Befehl fügt eine Variable in die gerade geöffnete Rezepturdefinition vor der ausgewählten Position ein.

**Aufruf:** Menü **Rezepturen**, Kontextmenü

**Voraussetzung:** Eine Rezepturdefinition ist im Editor geöffnet ist und die einfache Ansicht gewählt.

Dabei fügt TwinCAT in der Spalte **Variable** den Standardtext „NeueVariable“ ein. Diesen Namen müssen Sie durch den entsprechenden gültigen Variablennamen ersetzen werden. Dazu öffnen Sie die Eingabehilfe über Schaltfläche  oder tragen Sie den Variablennamen direkt in das Tabellenfeld ein.

**Siehe auch:**

- Dokumentation PLC: Werte ändern mit Rezepturen

### 5.17.10 Befehl Variablen entfernen

Symbol: 

**Funktion:** Der Befehl entfernt in einer Rezepturdefinition die selektierte(n) Variablen.

**Aufruf:** Taste **[Entf]**, Kontextmenü

**Voraussetzung:** Sie haben eine Variable selektiert.

**Siehe auch:**

- Dokumentation PLC: Werte ändern mit Rezepturen

### 5.17.11 Befehl Strukturierte Variablen aktualisieren

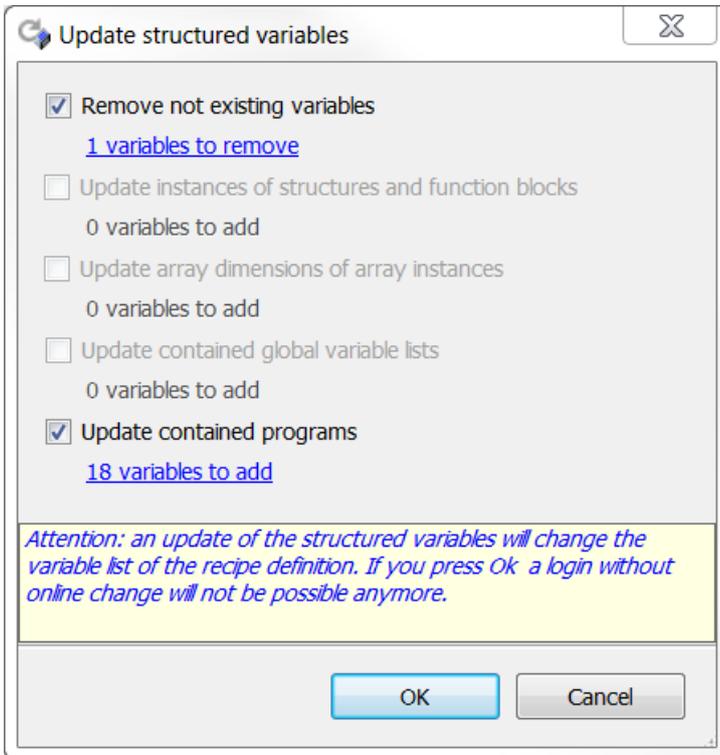
Symbol: 

**Funktion:** Der Befehl öffnet den Dialog **Strukturierte Variablen aktualisieren**.

**Aufruf:** Menü **Rezepturen**

In dem Dialog können Sie Rezepturdefinitionen aktualisieren, wenn sich die Deklaration einer strukturierten Variablen oder eines Bausteins geändert hat. Wenn beispielsweise die Dimension eines Arrays verändert wurde, können Sie die Einträge in der Rezepturdefinition automatisch entsprechend entfernen oder hinzufügen.

**Dialog Strukturierte Variablen aktualisieren**



Nicht existierende Variablen entfernen	: Variablen, die es auf Grund der Änderung eines strukturierten Elements im Projekt nicht länger gibt, werden aus der Rezepturdefinition entfernt.
Instanzen von Strukturen und Funktionsbausteinen aktualisieren	: Wenn die Deklaration einer Struktur oder eines Funktionsbausteins erweitert wird, die/der mit einer Instanz in der Rezepturdefinition vertreten ist, werden der Rezepturdefinition die entsprechenden Variablen hinzugefügt.
Array-Dimensionen von Array-Instanzen aktualisieren	: Wenn die Dimension eines Arrays erweitert wird, das mit einer Instanz in der Rezepturdefinition vertreten ist, werden der Rezepturdefinition die entsprechenden Variablen hinzugefügt.
Enthaltene globale Variablenlisten aktualisieren	: Wenn die Deklaration einer globalen Variablenliste erweitert wird, die mit einer Instanz in der Rezepturdefinition vertreten sind, werden der Rezepturdefinition die entsprechenden Variablen hinzugefügt.
Enthaltene Programme aktualisieren	: Wenn die Deklaration eines Programms erweitert wird, das in der Rezepturdefinition instanziiert wurde, werden der Rezepturdefinition die entsprechenden Variablen hinzugefügt.

**Siehe auch:**

- Dokumentation PLC: Werte ändern mit Rezepturen

## 5.17.12 Befehl Rezepturen vom Gerät laden

Symbol: 

**Funktion:** Der Befehl stößt die Synchronisierung der Rezepturen der gerade geöffneten Rezepturdefinition im Projekt mit den auf dem Gerät in Form von Rezepturdateien liegenden Rezepturen an.

**Aufruf:** Menü **Rezepturen**

**Voraussetzung:** Das SPS-Projekt ist im Onlinebetrieb und Sie haben eine Rezepturdefinition im Editor geöffnet. Im Detail bedeutet die Synchronisierung Folgendes:

- Die aktuellen Werte für die im Projekt vorliegenden Rezepturvariablen werden mit den Werten aus den Rezepturen auf der Steuerung überschrieben. Dadurch wird beim nächsten Einloggen möglicherweise ein Online-Change ausgelöst.
- Wenn in den Rezepturdateien auf der Steuerung Rezepturvariablen definiert sind, die in der Rezepturdefinition im Projekt fehlen, werden diese Variablen beim Laden ignoriert. Zuvor erscheint pro Rezepturdatei eine Meldung mit den betroffenen Variablen.
- Wenn in den Rezepturdateien auf der Steuerung Rezepturvariablen fehlen, die in der Rezepturdefinition im Projekt enthalten sind, erscheint pro Rezepturdatei eine Meldung mit den betroffenen Variablen.
- Wenn auf der Steuerung zusätzliche Rezepturen für diese Variablen angelegt wurden, werden diese der Rezepturdefinition im Projekt hinzugefügt.

## 5.18 Bibliothek

Befehl	Weiterführende Informationen in der PLC-Dokumentation
<b>Bibliothekserstellung</b>	
Befehl Als Bibliothek speichern...	<a href="#">Befehl Als Bibliothek speichern [► 283]</a>
Befehl Als Bibliothek speichern und installieren...	<a href="#">Befehl Als Bibliothek speichern und installieren [► 285]</a>
<b>Bibliotheksinstallation</b>	
Befehl Bibliotheksrepository	Bibliotheksrepository
<b>Bibliotheksverwaltung</b>	
Dialog Bibliotheksverwalter	Bibliotheksverwalter
<b>Weitere Befehle und Dialoge</b>	
Befehl Bibliothek hinzufügen	Befehl Bibliothek hinzufügen
Befehl Bibliothek ohne Auflösung hinzufügen	Befehl Bibliothek ohne Platzhalterauflösung hinzufügen
Befehl Bibliothek erneut laden	Befehl Bibliothek erneut laden
Befehl Bibliothek entfernen	Befehl Bibliothek entfernen
Befehl Details	Befehl Details
Befehl Abhängigkeiten	Befehl Abhängigkeiten
Befehl Eigenschaften	Befehl Eigenschaften
Befehl Platzhalter	Platzhalter
Befehl Effektive Version verwenden	Befehl Immer effektive Version verwenden
Befehl Immer neueste Version verwenden	Befehl Immer neueste Version verwenden
Befehl In Projektreferenz umwandeln	Befehl In Projektreferenz umwandeln
Befehl In Bibliothek umwandeln	Befehl In Bibliothek umwandeln
Befehl In Bibliothek umwandeln und installieren	Befehl In Bibliothek umwandeln und installieren

**Siehe auch:**

- Dokumentation PLC: [Bibliotheken verwenden > Weitere Befehle und Dialoge](#)

## 5.19 Visualisierung

Die Visualisierungsbefehle werden vom Plug-in **Visual Editor** für die Menükategorie Visualisierungskommandos zur Verfügung gestellt, die im Dialog **Extras > Anpassen** zu finden sind. Sie ermöglichen es, ein Visualisierungsobjekt im Visualisierungseditor zu editieren.

Die meisten sind standardmäßig Teil des Menüs **Visualisierung** und deswegen auch im Kontextmenü des Visualisierungseditors vorhanden. Wenn nötig, öffnen Sie den Dialog **Extras > Anpassen**, um die Menükonfiguration der Kategorie **Visualisierung** anzusehen oder zu ändern.

### 5.19.1 Befehl Schnittstellen-Editor

Symbol: 

**Funktion:** Dieser Befehl öffnet den Schnittstelleneditor, um Frameparameter in der Visualisierung zu definieren, die dazu bestimmt sind, im Element **Frame** einer anderen Visualisierung referenziert zu werden. Er wird in einer Registerblattansicht im oberen Teil des Visualisierungseditors angezeigt.

**Aufruf:** Menü **Visualisierung**

### 5.19.2 Befehl Tastaturkonfiguration

Symbol: 

**Funktion:** Dieser Befehl öffnet den Konfigurationseditor für die Tastaturbedienung für die aktuelle Visualisierung. Er wird in einer Registerblattansicht im oberen Teil des Visualisierungseditors angezeigt.

**Aufruf:** Menü **Visualisierung**

### 5.19.3 Befehl Elementliste

Symbol: 

**Funktion:** Dieser Befehl öffnet den Elementlisteneditor für die aktuelle Visualisierung. Er wird in einer Registerblattansicht im oberen Teil des Visualisierungseditors angezeigt.

**Aufruf:** Menü **Visualisierung**

### 5.19.4 Befehl Links ausrichten

Symbol: 

**Funktion:** Dieser Befehl bewirkt, dass alle ausgewählten Visualisierungselemente am linken Rand ihres am weitesten links positionierten Elements ausgerichtet werden.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

### 5.19.5 Befehl Oben ausrichten

Symbol: 

**Funktion:** Dieser Befehl bewirkt, dass alle ausgewählten Visualisierungselemente am oberen Rand ihres am weitesten oben positionierten Elements ausgerichtet werden.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

## 5.19.6 Befehl Rechts ausrichten

Symbol: 

**Funktion:** Dieser Befehl bewirkt, dass alle ausgewählten Visualisierungselemente am rechten Rand ihres am weitesten rechts positionierten Elements ausgerichtet werden.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

## 5.19.7 Befehl Unten ausrichten

Symbol: 

**Funktion:** Dieser Befehl bewirkt, dass alle ausgewählten Visualisierungselemente am unteren Rand ihres am weitesten unten positionierten Elements ausgerichtet werden.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

## 5.19.8 Befehl Vertikal zentrieren

Symbol: 

**Funktion:** Dieser Befehl bewirkt, dass alle ausgewählten Visualisierungselemente an ihrer gemeinsamen vertikalen Mitte ausgerichtet werden.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

## 5.19.9 Befehl Horizontal zentrieren

Symbol: 

**Funktion:** Dieser Befehl bewirkt, dass alle ausgewählten Visualisierungselemente an ihrer gemeinsamen horizontalen Mitte ausgerichtet werden.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

## 5.19.10 Befehl Horizontalen Abstand gleichmachen

Symbol: 

Der Befehl wird aktiv, wenn drei oder mehr Elemente selektiert sind.

1. Selektieren Sie alle Elemente, die horizontal mit gleichem Abstand positioniert werden sollen.  
⇒ Das erste Element ist blau markiert, die weiteren Elemente sind grau markiert.
2. Führen Sie den Befehl **Horizontalen Abstand gleichmachen** aus.  
⇒ Die Elemente werden so positioniert, dass das linkeste Element und das rechteste Element seine Position behalten und die Elemente dazwischen horizontal mit gleichem Abstand ausgerichtet werden.

## 5.19.11 Befehl Horizontalen Abstand vergrößern

Symbol: 

Der Befehl wird aktiv, wenn zwei oder mehr Elemente selektiert sind.

1. Selektieren Sie alle Elemente, die horizontal mit mehr Abstand positioniert werden sollen.  
⇒ Das erste Element ist blau markiert, die weiteren Elemente sind grau markiert.

2. Führen Sie den Befehl **Horizontalen Abstand vergrößern** aus.
  - ⇒ Die Elemente werden so positioniert, dass das blaue Element seine Position behält und die weiteren Elemente horizontal mit mehr Abstand zwischen den Elementen ausgerichtet werden. Der Abstand vergrößert sich um 1 Pixel.

### 5.19.12 Befehl Horizontalen Abstand verkleinern

Symbol: 

Der Befehl wird aktiv, wenn zwei oder mehr Elemente selektiert sind.

1. Selektieren Sie alle Elemente, die horizontal mit weniger Abstand positioniert werden sollen.
2. Führen Sie den Befehl **Horizontalen Abstand verkleinern** aus.
  - ⇒ Die Elemente werden so positioniert, dass das blaue Element seine Position behält und die weiteren Elemente horizontal mit weniger Abstand zwischen den Elementen ausgerichtet werden. Der Abstand verkleinert sich um 1 Pixel.

### 5.19.13 Befehl Horizontalen Abstand entfernen

Symbol: 

Der Befehl wird aktiv, wenn zwei oder mehr Elemente selektiert sind.

1. Selektieren Sie alle Elemente, die horizontal ohne Abstand zueinander positioniert werden sollen.
  - ⇒ Das erste Element ist blau markiert, die weiteren Elemente sind grau markiert.
2. Führen Sie den Befehl **Horizontalen Abstand entfernen** aus.
  - ⇒ Die Elemente werden so positioniert, dass das blaue Element seine Position behält und die weiteren Elemente horizontal ohne Abstand zueinander ausgerichtet werden.

### 5.19.14 Befehl Vertikalen Abstand gleichmachen

Symbol: 

Der Befehl wird aktiv, wenn zwei oder mehr Elemente selektiert sind.

1. Selektieren Sie alle Elemente, die vertikal mit gleichem Abstand positioniert werden sollen.
  - ⇒ Das erste Element ist blau markiert, die weiteren Elemente sind grau markiert.
2. Führen Sie den Befehl **Vertikalen Abstand gleichmachen** aus.
  - ⇒ Die Elemente werden so positioniert, dass das oberste Element und das unterste Element seine Position behalten und die Elemente dazwischen vertikal mit gleichem Abstand ausgerichtet werden.

### 5.19.15 Befehl Vertikalen Abstand vergrößern

Symbol: 

Der Befehl wird aktiv, wenn zwei oder mehr Elemente selektiert sind.

1. Selektieren Sie alle Elemente, die vertikal mit mehr Abstand positioniert werden sollen.
  - ⇒ Das erste Element ist blau markiert, die weiteren Elemente sind grau markiert.
2. Führen Sie den Befehl **Vertikalen Abstand vergrößern** aus.
  - ⇒ Die Elemente werden so positioniert, dass das blaue Element seine Position behält und die weiteren Elemente vertikal mit mehr Abstand zwischen den Elementen ausgerichtet werden. Der Abstand vergrößert sich um 1 Pixel.

### 5.19.16 Befehl Vertikalen Abstand verkleinern

Symbol: 

Der Befehl ist aktiv, wenn zwei oder mehr Elemente selektiert sind.

1. Selektieren Sie alle Elemente, die vertikal mit weniger Abstand positioniert werden sollen.  
⇒ Das erste Element ist blau markiert, die weiteren Elemente sind grau markiert.
2. Führen Sie den Befehl **Vertikalen Abstand verkleinern** aus.  
⇒ Die Elemente werden so positioniert, dass das blaue Element seine Position behält und die weiteren Elemente vertikal mit weniger Abstand zwischen den Elementen ausgerichtet werden. Der Abstand verkleinert sich um 1 Pixel.

### 5.19.17 Befehl Vertikalen Abstand entfernen

Symbol: 

Der Befehl wird aktiv, wenn zwei oder mehr Elemente selektiert sind.

1. Selektieren Sie alle Elemente, die vertikal ohne Abstand zueinander positioniert werden sollen.  
⇒ Das erste Element ist blau markiert, die weiteren Elemente sind grau markiert.
2. Führen Sie den Befehl **Vertikalen Abstand entfernen** aus.  
⇒ Die Elemente werden so positioniert, dass das blaue Element seine Position behält und die weiteren Elemente vertikal ohne Abstand zueinander ausgerichtet werden.

### 5.19.18 Befehl Breite gleichmachen

Symbol: 

Der Befehl wird aktiv, wenn mehr als ein Element selektiert ist, außer ein Linien- oder Polygonelement ist selektiert.

1. Selektieren Sie alle Elemente, die die gleiche Breite bekommen sollen.  
⇒ Das erste Element ist blau markiert, die weiteren Elemente sind grau markiert.
2. Führen Sie den Befehl **Breite gleichmachen** aus.  
⇒ Alle Elemente bekommen die Breite des blau markierten Elements.

### 5.19.19 Befehl Höhe gleichmachen

Symbol: 

Der Befehl wird aktiv, wenn mehr als ein Element selektiert ist, außer ein Linien- oder Polygonelement ist selektiert.

1. Selektieren Sie alle Elemente, die die gleiche Höhe bekommen sollen.  
⇒ Das erste Element ist blau markiert, die weiteren Elemente sind grau markiert.
2. Führen Sie den Befehl **Höhe gleichmachen** aus.  
⇒ Alle Elemente bekommen die Höhe des blau markierten Elements.

### 5.19.20 Befehl Größe gleichmachen

Symbol: 

Der Befehl ist aktiv, wenn mehr als ein Element selektiert ist, außer ein Linien- oder Polygonelement ist selektiert.

1. Selektieren Sie alle Elemente, die die gleiche Größe bekommen sollen.  
⇒ Das erste Element ist blau markiert, die weiteren Elemente sind grau markiert.
2. Führen Sie den Befehl **Größe gleichmachen** aus.  
⇒ Alle Elemente bekommen die Größe des blau markierten Elements.

### 5.19.21 Befehl Größe an Raster anpassen

Symbol: 

Der Befehl wird aktiv, wenn mehr als ein Element selektiert ist, außer ein Linien- oder Polygonelement ist selektiert.

1. Selektieren Sie alle Elemente, die in Position und Größe am Raster ausgerichtet werden sollen.  
⇒ Das erste Element ist blau markiert, die weiteren Elemente sind grau markiert.
2. Führen Sie den Befehl **Größe an Raster** anpassen aus.  
⇒ Alle Elemente werden in Größe und Position am Raster ausgerichtet.

### 5.19.22 Befehl Um Eins nach vorn legen

Symbol: 

**Funktion:** Dieser Befehl positioniert das ausgewählte Element eine Ebene höher, d. h. weiter in den Vordergrund der Visualisierung. Elemente auf tieferen Ebenen werden von denen auf höheren verdeckt.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

### 5.19.23 Befehl Nach vorn legen

Symbol: 

**Funktion:** Dieser Befehl positioniert das ausgewählte Element im Vordergrund der Visualisierung, d. h. auf höchster Ebene. Elemente auf tieferen Ebenen werden von denen auf höheren verdeckt.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

### 5.19.24 Befehl Um Eins nach hinten legen

Symbol: 

**Funktion:** Dieser Befehl positioniert das ausgewählte Element eine Ebene tiefer, d. h. weiter in den Hintergrund der Visualisierung. Elemente auf tieferen Ebenen werden von denen auf höheren verdeckt.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

### 5.19.25 Befehl Nach hinten legen

Symbol: 

**Funktion:** Dieser Befehl positioniert das ausgewählte Element im Hintergrund der Visualisierung, d. h. auf tiefster Ebene. Elemente auf tieferen Ebenen werden von denen auf höheren verdeckt.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

## 5.19.26 Befehl Gruppieren

Symbol: 

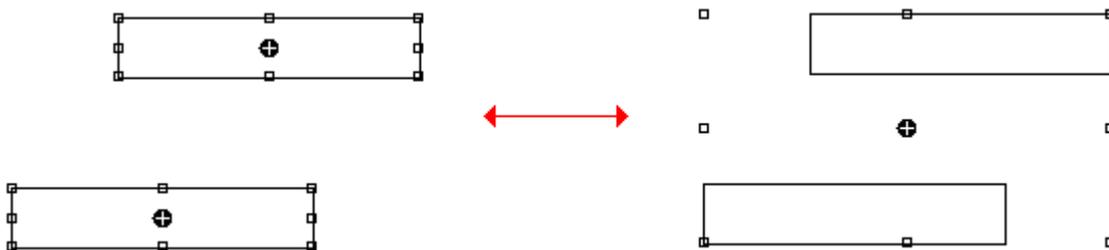
**Funktion:** Mit diesem Befehl werden die gerade ausgewählten Visualisierungselemente gruppiert und die Gruppe als Einzelobjekt selektiert dargestellt.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

**Voraussetzung:** Mehrere Visualisierungselemente sind ausgewählt. Für die Mehrfachauswahl halten Sie die [**Umschalt**]-Taste gedrückt, während Sie die gewünschten Elemente anklicken. Alternativ können Sie außerhalb eines Elements ins Editorfenster klicken und bei gedrückter Maustaste ein Rechteck um die gewünschten Elemente ziehen.

Zum Auflösen der Gruppe verwenden Sie Befehl **Gruppierung aufheben**.

Die folgende Abbildung zeigt die Gruppierung (von links nach rechts) bzw. das Aufheben der Gruppierung (von rechts nach links) zweier Rechteck-Elemente:



- [Befehl Gruppierung aufheben \[► 264\]](#)

## 5.19.27 Befehl Gruppierung aufheben

Symbol: 

**Funktion:** Mit diesem Befehl wird eine Gruppe von Visualisierungselementen wieder aufgelöst. Die einzelnen Elemente werden jeweils wieder einzeln selektiert dargestellt.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

**Siehe auch:**

- [Gruppieren \[► 264\]](#)

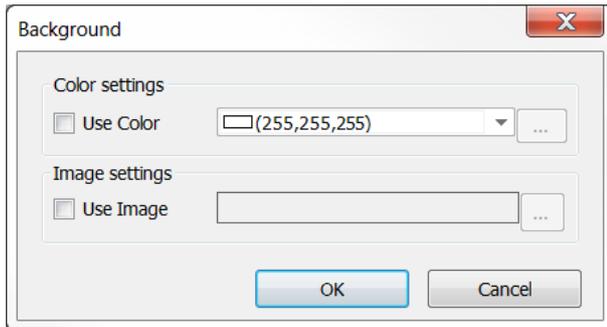
## 5.19.28 Befehl Hintergrund

Symbol: 

**Funktion:** Dieser Befehl öffnet den Dialog **Konfiguration der Framevisualisierung**, in dem eine Farbe und/oder eine Abbildung für den Visualisierungshintergrund ausgewählt werden kann.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

## Dialog Hintergrund



Aktivieren Sie die gewünschte(n) Option(en):

- **Bitmap:** Um ein Hintergrundbild zu definieren, muss hier der Pfad einer Bilddatei, die in einer Bildersammlung im Projekt verfügbar ist, eingetragen werden. Geben Sie dazu den Namen der Bildersammlung und die ID der Bilddatei - getrennt durch einen Punkt „.“ - ein: <bildersammlung>.<ID> (z. B. „Images\_1.drive\_icon“, „Images\_1.43“).
- **Farbe:** Um die Hintergrundfarbe der Visualisierung zu definieren, wählen Sie aus der Farbauswahlliste die gewünschte Farbe aus.

### 5.19.29 Befehl Alles auswählen

Symbol:  / 

**Funktion:** Dieser Befehl wählt alle Elemente der gerade im Editor geöffneten Visualisierung aus.

**Aufruf:** Menü **Visualisierung**, Kontextmenü

**Siehe auch:**

- [Alles deselektieren \[► 265\]](#)

### 5.19.30 Befehl Alles deselektieren

Symbol: 

**Funktion:** Dieser Befehl hebt die aktuelle Selektion von Visualisierungselementen auf.

**Aufruf:** Menü **Visualisierung**

**Siehe auch:**

- [Befehl Alles auswählen \[► 265\]](#)

### 5.19.31 Befehl Visualisierungselement vervielfachen

Symbol: 

**Funktion:** Der Befehl öffnet den Dialog **Visualisierungselement vervielfachen**, mit dem Sie das Vervielfachen konfigurieren können.

**Aufruf:** Menü **Visualisierung** , Kontextmenü

**Voraussetzung:** Die Visualisierung ist aktiv und ein Vorlagenelement ist selektiert.

#### Dialog Visualisierungselement vervielfachen

TwinCAT fügt beim Vervielfachen weitere Elemente ein, die dem Vorlagenelement gleichen. Sie können in diesem Dialog die Anzahl und die Anordnung, sowie die Ersetzung der Indizes konfigurieren.

**Registerkarte Grundeinstellung****Gesamtzahl Elemente:**

TwinCAT fügt die neuen Elemente als Tabelle ein. Die Zeilenanzahl ist in <b>Horizontal</b> eingestellt, die Spaltenanzahl in <b>Vertikal</b> . Das Produkt der beiden bestimmt die tatsächliche Gesamtzahl der einzufügenden Elemente.	
Horizontal	Anzahl an Elementen pro Zeile Voreinstellung: Entspricht der Anzahl der Komponenten in \$FIRSTDIM\$
Vertikal	Anzahl an Elementen pro Spalte Voreinstellung: Entspricht der Anzahl der Komponenten in \$SECONDDIM\$

**Offset zwischen den Elementen:**

TwinCAT ordnet die Elemente in der Visualisierung als Tabelle an. Wenn Sie einen Offset angeben, wird zwischen den Elementen ein Abstand eingefügt.	
<ul style="list-style-type: none"> <li>• 0 : Die Rahmen der Elemente überlappen um ein Pixel</li> <li>• 1 : Die Elemente stoßen aneinander</li> <li>• &lt;n&gt; : Zwischen den Elementen ist ein sichtbarer Abstand von n-1 Pixel</li> </ul>	
Horizontal	Zeilenabstand der Elemente in Pixel
Vertikal	Spaltenabstand der Elemente in Pixel

**Registerkarte Erweiterte Einstellung****Erste Dimension:**

Startindex	Startindex für \$FIRSTDIM\$ Voreinstellung: 1, bedeutet, das der konkrete Index für \$FIRSTDIM\$ mit 1 beginnt. Beispiel Array[1, <\$SECONDDIM\$>]
Hochzählen	Inkrement, um das der Index hochgezählt wird Voreinstellung: 1

**Zweite Dimension:**

Startindex	Startindex für \$SECONDDIM\$ Voreinstellung: 1, bedeutet, das der konkrete Index für \$SECONDDIM\$ mit 1 beginnt . Beispiel Array[<\$FIRSTDIM\$>, 1] relevant sind
Hochzählen	Inkrement, um das der Index hochgezählt wird Voreinstellung: 1

**5.19.32 Befehl Tastaturbedienung aktivieren**Symbol: 

**Funktion:** Dieser Befehl ist in der Menüleiste für eine integrierte Visualisierung (Diagnosevisualisierung) verfügbar. Er aktiviert bzw. deaktiviert die Tastaturbedienung im Onlinebetrieb einer Visualisierung.

**Aufruf:** Menü **Visualisierung**

Wenn die Tastaturbedienung aktiviert ist, können Eingaben auf Elemente und die Auswahl der Elemente über bestimmte Tastenkombinationen vorgenommen werden. In diesem Fall werden andere über Tastatur gegebene Befehle nicht ausgeführt, solange der Visualisierungseditor aktiv und im Onlinebetrieb ist.

## 5.20 Sonstiges

### 5.20.1 Befehl Schnittstellen implementieren

**Funktion:** Der Befehl aktualisiert für einen Funktionsbaustein die implementierten Schnittstellen, indem die Schnittstellenelemente hinzugefügt werden, die der Funktionsbaustein aktuell nicht enthält.

**Aufruf:** Kontextmenü, wenn der Funktionsbaustein im SPS-Projektbaum selektiert ist.

**Voraussetzung:** Der Funktionsbaustein implementiert eine Schnittstelle, die Sie verändert haben. Beispielsweise haben Sie der Schnittstelle eine weitere Methode hinzugefügt.

#### Anwendungsfälle

Bei Ausführung dieses Befehls werden die automatisch angelegten Methoden oder Eigenschaften mit einem Pragmaattribut versehen, welches Übersetzungsfehler oder -warnungen provoziert. Dadurch werden Sie dahingehend unterstützt, dass automatisch angelegte Elemente nicht unbeabsichtigt leer bleiben. Ob ein error- oder warning-Attribut verwendet wird, hängt vom Anwendungsfall ab.

#### Fall 1:

**Situation:** Der Funktionsbaustein, für den der Befehl **Schnittstellen implementieren** ausgeführt wird, ist **nicht** von einem anderen Funktionsbaustein abgeleitet.

**Konsequenz:** Bei Ausführung des Befehls werden die Schnittstellenelemente in dem Funktionsbaustein ohne Implementierung angelegt („Stubs“) und mit einem warning-Attribut versehen (in der ersten Zeile der Methoden-/Eigenschaftendeklaration). Durch die Warnungen, die beim Kompilieren erzeugt werden, werden Sie darauf aufmerksam gemacht, dass diese Elemente automatisch erzeugt wurden und der gewünschte Implementierungscode hinzugefügt werden muss.

```
{warning 'add method/property implementation'}
```

**Vorgehen:** Fügen Sie dem jeweiligen Schnittstellenelement (Methode oder Eigenschaft) den gewünschten Implementierungscode hinzu. Entfernen Sie anschließend das warning-Attribut aus der Methoden- bzw. Eigenschaftendeklaration.

#### Fall 2:

**Situation:** Der Funktionsbaustein, für den der Befehl Schnittstellen implementieren ausgeführt wird, ist von einem anderen Funktionsbaustein abgeleitet. Das Element (Methode oder Eigenschaft), das bei Ausführung des Befehls im abgeleiteten Funktionsbaustein angelegt wird, wird **nicht** über Vererbung von dem Basis-Funktionsbaustein zur Verfügung gestellt (d. h. das Element ist nicht unterhalb des Basis-Funktionsbausteins oder einer höheren Elternklasse vorhanden).

**Konsequenz/Vorgehen:** s. Fall 1.

#### Fall 3:

**Situation:** Der Funktionsbaustein, für den der Befehl **Schnittstellen implementieren** ausgeführt wird, ist von einem anderen Funktionsbaustein abgeleitet. Das Element (Methode oder Eigenschaft), das bei Ausführung des Befehls im abgeleiteten Funktionsbaustein angelegt wird, wird bereits über Vererbung von dem Basis-Funktionsbaustein zur Verfügung gestellt (d. h. das Element ist unterhalb des Basis-Funktionsbausteins oder einer höheren Elternklasse vorhanden).

**Konsequenz:** Bei Ausführung des Befehls wird das Schnittstellenelement in dem abgeleiteten Funktionsbaustein ohne Implementierung angelegt („Stub“) und mit einem error-Attribut versehen (in der ersten Zeile der Methoden-/Eigenschaftendeklaration). Durch den Fehler, der beim Kompilieren erzeugt wird, werden Sie darauf aufmerksam gemacht, dass dieses Schnittstellenelement automatisch erzeugt wurde und dass diese Methode oder Eigenschaft das entsprechende Element des Basis-Funktionsbausteins überschreibt.

```
{error 'add method/property implementation or delete method/property to use base implementation'}
```

**Vorgehen:** Falls Sie die Methode oder Eigenschaft des Basis-Funktionsbausteins überschreiben oder erweitern möchten, fügen Sie dem Element unterhalb des abgeleiteten Bausteins den gewünschten Implementierungscode hinzu. Entfernen Sie anschließend das error-Attribut aus der Methoden- bzw. Eigenschaftendeklaration. Falls Sie die Methode oder Eigenschaft des Basis-Funktionsbausteins hingegen **nicht** überschreiben möchten, löschen Sie die Methode oder Eigenschaft unterhalb des abgeleiteten Funktionsbausteins. Dadurch wird die Methoden- bzw. Eigenschaftimplementierung des Basis-Funktionsbausteins verwendet.

## 5.21 Kontextmenü Projektmappe

### 5.21.1 Befehl Projektmappe erstellen

Symbol: 

**Funktion:** Der Befehl startet den Übersetzungsprozess bzw. die Codegenerierung für alle in der Projektmappe enthaltenen Projekte.

**Aufruf:** Menü **Erstellen** oder Kontextmenü der Projektmappe

**Voraussetzung:** Die Projektmappe ist selektiert.

Alle in der Projektmappe enthaltenen Projekte werden der Reihe nach übersetzt. Dies betrifft auch die unterhalb eines TwinCAT-Projekts eingebundenen Projekte (SPS, C++, etc.). Die dabei für ein SPS-Projekt durchgeführten Schritte werden im Abschnitt [Befehl SPS-Projekt erstellen \[▶ 280\]](#) beschrieben.

### 5.21.2 Befehl Projektmappe neu erstellen

**Funktion:** Der Befehl startet den Übersetzungsprozess für alle in der Projektmappe enthaltenen Projekte, auch wenn sie zuletzt fehlerfrei übersetzt wurden.

**Aufruf:** Menü **Erstellen** oder Kontextmenü der Projektmappe

**Voraussetzung:** Die Projektmappe ist selektiert.

Bei einer Neuerstellung der Projektmappe wird die Projektmappe zunächst bereinigt (siehe auch: [Befehl Projektmappe bereinigen \[▶ 268\]](#)) und anschließend erstellt (siehe auch: [Befehl Projektmappe erstellen \[▶ 268\]](#)).

**Siehe auch:**

- [Befehl SPS-Projekt neu erstellen \[▶ 280\]](#)

### 5.21.3 Befehl Projektmappe bereinigen

**Funktion:** Der Befehl startet die Bereinigung für alle in der Projektmappe enthaltenen Projekte.

**Aufruf:** Menü **Erstellen** oder Kontextmenü der Projektmappe

**Voraussetzung:** Die Projektmappe ist selektiert.

Alle in der Projektmappe enthaltenen Projekte werden der Reihe nach bereinigt. Dies betrifft auch die unterhalb eines TwinCAT-Projekts eingebundenen Projekte (SPS, C++, etc.). Die dabei für ein SPS-Projekt durchgeführten Schritte werden im Abschnitt [Befehl SPS-Projekt bereinigen \[▶ 280\]](#) beschrieben.

### 5.21.4 Befehl Save <Projektmappenname> as Archive...

**Funktion:** Der Befehl öffnet den Standarddialog zum Speichern einer Datei als Archiv. Die Projektmappe kann unter dem gewünschten Speicherpfad als \*.tzip-Archiv abgelegt werden.

**Aufruf:** Menü **Datei**, Kontextmenü

**Voraussetzung:** Die Projektmappe ist im **Projektmappen-Explorer** ausgewählt.

<b>Inhalt vom *.tzip</b>	Der Archivordner *.tzip enthält alle in der Projektmappe enthaltenen TwinCAT-Projekttypen. Dies können TwinCAT-, TwinCAT HMI-, Scope- und Connectivity-Projekte sein.
<b>Befehl zum Öffnen</b>	Ein tzip-Archiv kann über den folgenden Befehl wieder geöffnet werden: Befehl <a href="#">Open Solution from Archive [► 53]</a> .
<b>Hinweis zu SPS-Projekten</b>	Falls die Projektmappe ein oder mehrere SPS-Projekte enthält, sind die Dateien und Ordner, die bezüglich dieser SPS-Projekte in dem Archivordner gespeichert werden, abhängig von den SPS-Projekteinstellungen des jeweiligen SPS-Projekts. <a href="#">Registerkarte Settings [► 122]</a>

### 5.21.5 Befehl Sende per E-Mail.../Send by E-Mail...

Symbol: 

**Funktion:** Der Befehl startet das aktuell im System eingestellte E-Mail-Programm und öffnet eine neue E-Mail. Diese enthält im Anhang die Archivdatei des ausgewählten Projekts.

**Aufruf:** Menü **Datei**, Kontextmenü

## 5.22 Kontextmenü TwinCAT Projekt

### 5.22.1 Befehl Sichern <TwinCAT-Projektname> als Archiv...

Symbol: 

**Funktion:** Der Befehl öffnet den Standarddialog zum Speichern einer Datei als Archiv. Das Projekt kann unter dem gewünschten Speicherpfad als \*.tzip-Archiv abgelegt werden.

**Aufruf:** Menü **Datei**, Kontextmenü

**Voraussetzung:** Das TwinCAT-Projekt ist **Projektmappen-Explorer** ausgewählt.

<b>Inhalt vom *.tzip</b>	Der Archivordner *.tzip enthält das TwinCAT-Projekt, welches archiviert wird.
<b>Befehl zum Öffnen</b>	Ein tzip-Archiv kann über den folgenden Befehl wieder geöffnet werden: Befehl <a href="#">Projekt/Projektmappe (Projekt/Projektmappe öffnen) [► 53]</a>
<b>Hinweis zu SPS-Projekten</b>	Falls das TwinCAT-Projekt ein oder mehrere SPS-Projekte enthält, sind die Dateien und Ordner, die bezüglich dieser SPS-Projekte in dem Archivordner gespeichert werden, abhängig von den SPS-Projekteinstellungen des jeweiligen SPS-Projekts. <a href="#">Registerkarte Settings [► 122]</a>

## 5.22.2 Befehl **Sende per E-Mail.../Send by E-Mail...**

Symbol: 

**Funktion:** Der Befehl startet das aktuell im System eingestellte E-Mail-Programm und öffnet eine neue E-Mail. Diese enthält im Anhang die Archivdatei des ausgewählten Projekts.

**Aufruf:** Menü **Datei**, Kontextmenü

## 5.22.3 Befehl **Sichere <TwinCAT-Projektnamen> automatisch auf dem Zielsystem**

**Funktion:** Über diesen Befehl können Sie aktivieren oder deaktivieren, dass das TwinCAT Projekt beim Aktivieren automatisch als .tszip auf dem Zielsystem abgelegt wird. Dies ist notwendig, wenn Sie das Projekt zu einem späteren Zeitpunkt mit dem [Befehl Open Project from Target \[▶ 61\]](#) vom Zielsystem laden und öffnen wollen.

**Aufruf:** Kontextmenü des TwinCAT Projektes

**Voraussetzung:** Das TwinCAT Projekt ist im Projektmappen-Explorer selektiert.

## 5.22.4 Befehl **<TwinCAT-Projektname> mit dem Zielsystem vergleichen...**

**Funktion:** Dieser Befehl ermöglicht, das selektierte TwinCAT Projekt mit dem Projektstand auf dem Zielsystem mit des TwinCAT Project Compare Tools zu vergleichen.

**Aufruf:** Kontextmenü des TwinCAT Projektes

**Voraussetzung:** Das TwinCAT Projekt ist im Projektmappen-Explorer selektiert und ein Zielsystem ist ausgewählt, auf welches bereits ein TwinCAT Projekt heruntergeladen worden ist.

**Siehe auch:**

- [Befehl Projekt mit Zielsystem aktualisieren... \[▶ 270\]](#)

## 5.22.5 Befehl **Projekt mit Zielsystem aktualisieren...**

**Funktion:** Dieser Befehl ermöglicht, das selektierte TwinCAT Projekt auf den Projektstand des verbundenen Zielsystems zu aktualisieren. Dafür wird ein Popup-Dialog mit einer Liste der geänderten Dateien geöffnet, der bestätigt werden muss, um die Aktion erfolgreich durchzuführen. Ein detaillierter Vergleich ist hier nicht vorgesehen.

**Aufruf:** Kontextmenü des TwinCAT Projektes

**Voraussetzung:** Das TwinCAT Projekt ist im Projektmappen-Explorer selektiert.

**Siehe auch:**

- [Befehl <TwinCAT-Projektname> mit dem Zielsystem vergleichen... \[▶ 270\]](#)

## 5.22.6 Befehl **Projekt mit TwinCAT 2.xx Version laden...**

**Funktion:** Dieser Befehl öffnet den Standard-Browserdialog, über den eine TwinCAT 2 System Manager-Datei ausgewählt und importiert werden kann. Auf diese Weise können Sie ein bestehendes TwinCAT 2 Projekt inklusive der System Manager-Einstellungen und des SPS-Projektes nach TwinCAT 3 konvertieren.

**Aufruf:** Kontextmenü des TwinCAT Projektes

**Voraussetzung:** Das TwinCAT Projekt ist ohne Änderungen neu angelegt worden und im Projektmappen-Explorer selektiert.

**Siehe auch:**

- TwinCAT-2-SPS-Projekt öffnen

### 5.22.7 Befehl Verborgene Konfigurationen anzeigen

**Funktion:** Dieser Befehl öffnet ein Detailmenü, in dem die verborgenen Konfigurationen aufgelistet werden und wieder angezeigt werden können.

**Aufruf:** Kontextmenü des TwinCAT Projektes

**Voraussetzung:** Das TwinCAT Projekt ist im Projektmappen-Explorer selektiert.

### 5.22.8 Befehl Aus Projektmappe entfernen

Symbol: 

**Funktion:** Der Befehl ermöglicht es, das TwinCAT Projekt aus der Projektmappe zu löschen.

**Aufruf:** Kontextmenü des TwinCAT Projektes

**Voraussetzung:** Das TwinCAT Projekt ist im Projektmappen-Explorer selektiert.

### 5.22.9 Befehl Umbenennen

Symbol: 

**Funktion:** Der Befehl ermöglicht das Umbenennen des TwinCAT Projektes im **Projektmappen-Explorer**.

**Aufruf:** Kontextmenü des TwinCAT Projektes

**Voraussetzung:** Das TwinCAT Projekt ist im Projektmappen-Explorer selektiert.

### 5.22.10 Befehl TwinCAT-Projekt erstellen

Symbol: 

**Funktion:** Der Befehl startet den Übersetzungsprozess bzw. die Codeerzeugung für das gerade aktive TwinCAT-Projekt.

**Aufruf:** Menü **Erstellen**, wenn aktuell ein TwinCAT-Projekt selektiert ist, oder Kontextmenü des TwinCAT-Projekts

**Voraussetzung:** Das TwinCAT-Projekt ist selektiert.

Alle in dem TwinCAT-Projekt enthaltenen Projekte (SPS, C++, etc.) werden der Reihe nach übersetzt. Die dabei für ein SPS-Projekt durchgeführten Schritte werden im Abschnitt [Befehl SPS-Projekt erstellen \[► 280\]](#) beschrieben.

**Siehe auch:**

- [Befehl TwinCAT-Projekt neu erstellen \[► 271\]](#)

### 5.22.11 Befehl TwinCAT-Projekt neu erstellen

**Funktion:** Der Befehl startet den Übersetzungsprozess bzw. die Codeerzeugung für das gerade aktive TwinCAT-Projekt, auch wenn es zuletzt fehlerfrei übersetzt wurde.

**Aufruf:** Menü **Erstellen**, wenn aktuell ein TwinCAT-Projekt selektiert ist, oder Kontextmenü des TwinCAT-Projekts

**Voraussetzung:** Das TwinCAT-Projekt ist selektiert.

Bei einer Neuerstellung des Projekts wird das TwinCAT-Projekt zunächst bereinigt (siehe auch: [Befehl TwinCAT-Projekt bereinigen \[► 272\]](#)) und anschließend erstellt (siehe auch: [Befehl TwinCAT-Projekt erstellen \[► 271\]](#)).

### 5.22.12 Befehl TwinCAT-Projekt bereinigen

**Funktion:** Der Befehl löscht die lokale Übersetzungsinformation für das gerade aktive SPS-Projekt und aktualisiert das Sprachmodell aller Objekte.

**Aufruf:** Menü **Erstellen**, wenn aktuell ein TwinCAT-Projekt selektiert ist, oder Kontextmenü des TwinCAT-Projekts

**Voraussetzung:** Das TwinCAT-Projekt ist selektiert.

Alle in dem TwinCAT-Projekt enthaltenen Projekte (SPS, C++, etc.) werden der Reihe nach bereinigt. Die dabei für ein SPS-Projekt durchgeführten Schritte werden im Abschnitt [Befehl SPS-Projekt bereinigen \[► 280\]](#) beschrieben.

**Siehe auch:**

- [Befehl TwinCAT-Projekt neu erstellen \[► 271\]](#)

### 5.22.13 Befehl Projekt entladen

**Funktion:** Dieser Befehl entlädt das TwinCAT Projekt, sodass alle Dateien dieses TwinCAT Projekts freigegeben sind.

**Aufruf:** Menü Projekt oder Kontextmenü des TwinCAT Projektes

**Voraussetzung:** Das TwinCAT Projekt ist im Projektmappen-Explorer selektiert.

### 5.22.14 Import AutomationML via AML DataExchange...

**Funktion:** Der Befehl öffnet den Standard-Browse-Dialog, über den eine Datei im AutomationML-Format gesucht und importiert werden kann.

**Aufruf:** Der Befehl kann über das Kontextmenü des TwinCAT Projekts unter **Import AutomationML** oder über den **TwinCAT** Eintrag in der Menüleiste unter **AutomationML** und **Import AutomationML** aufgerufen werden.

**Voraussetzung:** Das TwinCAT Projekt ist im Projektmappen-Explorer selektiert.

**Siehe auch**

- Befehl: Open AML DataExchange Log (local)

### 5.22.15 Export AutomationML...

**Funktion:** Der Befehl öffnet den Standard-Browserdialog zum Speichern einer Datei im AutomationML-Format, um die unter dem Knoten E/A vorhandene Topologie zu exportieren.

**Aufruf:** Der Befehl kann über das Kontextmenü des TwinCAT Projekts oder über den **TwinCAT** Eintrag in der Menüleiste unter **AutomationML** aufgerufen werden.

**Voraussetzung:** Das TwinCAT Projekt ist im Projektmappen-Explorer selektiert.

## 5.23 Kontextmenü SPS-Projekt

### 5.23.1 Befehl Bootprojekt aktivieren

Symbol: 

**Funktion:** Der Befehl erzeugt bzw. aktualisiert das Bootprojekt des Zielsystems.

**Aufruf:** Kontextmenü des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist ausgewählt.

Ein Bootprojekt ist das Projekt, das automatisch gestartet wird, wenn die Steuerung eingeschaltet oder gestartet wird. TwinCAT speichert das Bootprojekt im Boot-Ordner des Zielsystems unter dem Dateinamen <SPS-ADS-Port>.app.

Um das Bootprojekt automatisch zu laden und zu starten, aktivieren Sie die Option **Autostart Boot Projekt**. Dies ist über zwei verschiedene Wege möglich:

- [Befehl Autostart Bootprojekt \[► 273\]](#)
- Checkbox innerhalb der SPS-Projekteinstellungen, [Registerkarte Settings \[► 122\]](#)

Aktualisieren Sie das Bootprojekt des Zielsystems anschließend manuell mit dem Befehl **Boot Projekt aktivieren**.

Alternativ können Sie innerhalb des Bestätigungsfensters, das nach Ausführung des [Befehls Konfiguration aktivieren \[► 134\]](#) erscheint, einstellen, ob für alle SPS-Projekte des TwinCAT-Projekts die Einstellung **Autostart Boot Projekt** aktiviert werden soll. Eine manuelle Ausführung des Befehls **Boot Projekt aktivieren** ist in diesem Fall nicht notwendig, da dies implizit bei Ausführung des [Befehls Konfiguration aktivieren \[► 134\]](#) ausgeführt wird.

### 5.23.2 Befehl Autostart Bootprojekt

**Funktion:** Der Befehl ändert die Option **Autostart Boot Projekt** des SPS-Projekts. Falls die Option vorher deaktiviert war, ist sie nach Ausführung des Befehls aktiviert. Falls die Option vorher aktiviert war, ist sie nach Ausführung des Befehls deaktiviert.

**Aufruf:** Kontextmenü des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist ausgewählt.

Ein Bootprojekt ist das Projekt, das automatisch gestartet wird, wenn die Steuerung eingeschaltet oder gestartet wird. TwinCAT speichert das Bootprojekt im Boot-Ordner des Zielsystems unter dem Dateinamen <SPS-ADS-Port>.app.

Um das Bootprojekt automatisch zu laden und zu starten, aktivieren Sie die Option **Autostart Boot Projekt**. Dies ist über zwei verschiedene Wege möglich:

- [Befehl Autostart Bootprojekt \[► 273\]](#)
- Checkbox innerhalb der SPS-Projekteinstellungen, [Registerkarte Settings \[► 122\]](#)

Aktualisieren Sie das Bootprojekt des Zielsystems anschließend manuell mit dem Befehl **Boot Projekt aktivieren**.

Alternativ können Sie innerhalb des Bestätigungsfensters, das nach Ausführung des [Befehls Konfiguration aktivieren \[► 134\]](#) erscheint, einstellen, ob für alle SPS-Projekte des TwinCAT-Projekts die Einstellung **Autostart Boot Projekt** aktiviert werden soll. Eine manuelle Ausführung des Befehls **Boot Projekt aktivieren** ist in diesem Fall nicht notwendig, da dies implizit bei Ausführung des [Befehls Konfiguration aktivieren \[► 134\]](#) ausgeführt wird.

### 5.23.3 Befehl ADS-Port ändern

**Funktion:** Über einen sich öffnenden Dialog können Sie die ADS-Portnummer des SPS-Laufzeitsystems ändern. Die Standardeinstellung des ADS-Ports für das erste SPS-Laufzeitsystem ist 851, für das Zweite 852 etc.

**Aufruf:** Kontextmenü des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist ausgewählt.

Falls das SPS-Projekt mehrere Instanzen besitzt (siehe [Befehl Neues Element hinzufügen \(Instanz\) \[▶ 276\]](#)), erscheint der Dialog zum Ändern des ADS-Ports so oft hintereinander, wie es der Anzahl der Projektinstanzen entspricht. Über den ersten Dialog stellen Sie den ADS-Port der ersten Projektinstanz ein, über den zweiten Dialog den ADS-Port der zweiten Projektinstanz, etc.

### 5.23.4 Befehl Projektbibliotheken installieren

**Funktion:** Die Bibliotheken, die in dem `_Libraries` Ordner auf Dateiebene des SPS-Projekts enthalten sind, werden in dem lokalen Bibliotheksrepository installiert. Anschließend sind alle dort enthaltenen Bibliotheken bekannt und können in Projekten verwendet werden. Falls eine Bibliothek bereits vor Ausführung des Befehls im lokalen Repository bekannt war, wird die Bibliothek erneut installiert. Im Normalfall genügt daher die Ausführung des Befehls [Projektbibliotheken installieren \(Unbekannte Versionen\) \[▶ 275\]](#).

**Aufruf:** Kontextmenü des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist ausgewählt.

#### Hintergrundinformationen:

Der `_Libraries` Ordner enthält die Bibliotheken, die in dem SPS-Projekt direkt oder indirekt referenziert werden. Eine direkte Referenzierung meint eine Bibliothek, die auf oberster Ebene innerhalb des Bibliotheksverwalters eingebunden ist. Eine indirekte Referenzierung meint eine Bibliothek, die innerhalb einer anderen Bibliothek referenziert wird.

Der Ordner dient beispielsweise dazu, die in einem Projekt verwendeten Bibliotheken im Rahmen eines Projektarchivs zu archivieren (ob dies der Fall sein soll, können Sie in den Projekteinstellungen konfigurieren, siehe [Registerkarte Settings \[▶ 122\]](#)). Dadurch können Sie das Projektarchiv z.B. an einen Kollegen weitergeben und falls in dem Projekt Bibliotheken verwendet werden, die in dem lokalen Bibliotheksrepository des Kollegen nicht vorhanden sind, können die fehlenden Bibliotheken mit Hilfe des [Befehls Projektbibliotheken installieren \[▶ 274\]](#) oder [Projektbibliotheken installieren \(Unbekannte Versionen\) \[▶ 275\]](#) installiert werden.

Zu den folgenden Zeitpunkten werden neuerdings verwendete Bibliotheken zum `_Libraries` Ordner hinzugefügt:

- Beim [Erstellen eines Datei-/E-Mailarchivs \[▶ 52\]](#), falls für das Datei-/E-Mailarchiv konfiguriert ist, dass Bibliotheken enthalten sein sollen (siehe [Registerkarte Settings \[▶ 122\]](#)).
- Beim [Aktivieren der Konfiguration \[▶ 134\]](#) oder beim [Aktivieren des Bootprojekts \[▶ 273\]](#), falls für das Target-Archiv konfiguriert ist, dass Bibliotheken enthalten sein sollen (siehe [Registerkarte Settings \[▶ 122\]](#)).
- Bei Ausführung des [Befehls Projektbibliotheksordner aktualisieren \[▶ 275\]](#).

Zu den folgenden Zeitpunkten werden nicht mehr verwendete Bibliotheken aus dem `_Libraries` Ordner entfernt:

- Bei Ausführung des [Befehls Projektbibliotheksordner aktualisieren \[▶ 275\]](#).

#### Siehe auch:

- [Befehl Projektbibliotheken installieren \(Unbekannte Versionen\) \[▶ 275\]](#)
- [Befehl Projektbibliotheksordner aktualisieren \[▶ 275\]](#)
- Doku PLC: Bibliotheken verwenden

### 5.23.5 Befehl Projektbibliotheken installieren (Unbekannte Versionen)

**Funktion:** Die Bibliotheken, die in dem `_Libraries` Ordner auf Dateiebene des SPS-Projekts enthalten sind und bislang nicht Teil des lokalen Bibliotheksrepositorys sind, werden in dem lokalen Bibliotheksrepository installiert. Anschließend sind alle dort enthaltenen Bibliotheken bekannt und können in Projekten verwendet werden. Falls eine Bibliothek bereits vor Ausführung des Befehls im lokalen Repository bekannt war, wird diese nicht erneut installiert.

**Aufruf:** Kontextmenü des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist ausgewählt.

#### Hintergrundinformationen:

Der `_Libraries` Ordner enthält die Bibliotheken, die in dem SPS-Projekt direkt oder indirekt referenziert werden. Eine direkte Referenzierung meint eine Bibliothek, die auf oberster Ebene innerhalb des Bibliotheksverwalters eingebunden ist. Eine indirekte Referenzierung meint eine Bibliothek, die innerhalb einer anderen Bibliothek referenziert wird.

Der Ordner dient beispielsweise dazu, die in einem Projekt verwendeten Bibliotheken im Rahmen eines Projektarchivs zu archivieren (ob dies der Fall sein soll, können Sie in den Projekteinstellungen konfigurieren, siehe [Registerkarte Settings \[► 122\]](#)). Dadurch können Sie das Projektarchiv z.B. an einen Kollegen weitergeben und falls in dem Projekt Bibliotheken verwendet werden, die in dem lokalen Bibliotheksrepository des Kollegen nicht vorhanden sind, können die fehlenden Bibliotheken mit Hilfe des [Befehls Projektbibliotheken installieren \[► 274\]](#) oder [Projektbibliotheken installieren \(Unbekannte Versionen\) \[► 275\]](#) installiert werden.

Zu den folgenden Zeitpunkten werden neuerdings verwendete Bibliotheken zum `_Libraries` Ordner hinzugefügt:

- Beim [Erstellen eines Datei-/E-Mailarchivs \[► 52\]](#), falls für das Datei-/E-Mailarchiv konfiguriert ist, dass Bibliotheken enthalten sein sollen (siehe [Registerkarte Settings \[► 122\]](#)).
- Beim [Aktivieren der Konfiguration \[► 134\]](#) oder beim [Aktivieren des Bootprojekts \[► 273\]](#), falls für das Target-Archiv konfiguriert ist, dass Bibliotheken enthalten sein sollen (siehe [Registerkarte Settings \[► 122\]](#)).
- Bei Ausführung des [Befehls Projektbibliotheksordner aktualisieren \[► 275\]](#).

Zu den folgenden Zeitpunkten werden nicht mehr verwendete Bibliotheken aus dem `_Libraries` Ordner entfernt:

- Bei Ausführung des [Befehls Projektbibliotheksordner aktualisieren \[► 275\]](#).

#### Siehe auch:

- [Befehl Projektbibliotheken installieren \[► 274\]](#)
- [Befehl Projektbibliotheksordner aktualisieren \[► 275\]](#)
- Doku PLC: Bibliotheken verwenden

### 5.23.6 Befehl Projektbibliotheksordner aktualisieren

**Funktion:** Der `_Libraries` Ordner auf Dateiebene des SPS-Projekts wird aktualisiert. Anschließend enthält der Ordner alle Bibliotheken, die in dem SPS-Projekt verwendet werden. Bibliotheken, die zuvor in dem `_Libraries` Ordner enthalten waren, aber inzwischen nicht mehr verwendet werden, werden aus dem Ordner entfernt.

**Aufruf:** Kontextmenü des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist ausgewählt.

#### Hintergrundinformationen:

Der `_Libraries` Ordner enthält die Bibliotheken, die in dem SPS-Projekt direkt oder indirekt referenziert werden. Eine direkte Referenzierung meint eine Bibliothek, die auf oberster Ebene innerhalb des Bibliotheksverwalters eingebunden ist. Eine indirekte Referenzierung meint eine Bibliothek, die innerhalb einer anderen Bibliothek referenziert wird.

Der Ordner dient beispielsweise dazu, die in einem Projekt verwendeten Bibliotheken im Rahmen eines Projektarchivs zu archivieren (ob dies der Fall sein soll, können Sie in den Projekteinstellungen konfigurieren, siehe [Registerkarte Settings \[► 122\]](#)). Dadurch können Sie das Projektarchiv z.B. an einen Kollegen weitergeben und falls in dem Projekt Bibliotheken verwendet werden, die in dem lokalen Bibliotheksrepository des Kollegen nicht vorhanden sind, können die fehlenden Bibliotheken mit Hilfe des [Befehls Projektbibliotheken installieren \[► 274\]](#) oder [Projektbibliotheken installieren \(Unbekannte Versionen\) \[► 275\]](#) installiert werden.

Zu den folgenden Zeitpunkten werden neuerdings verwendete Bibliotheken zum `_Libraries` Ordner hinzugefügt:

- Beim [Erstellen eines Datei-/E-Mailarchivs \[► 52\]](#), falls für das Datei-/E-Mailarchiv konfiguriert ist, dass Bibliotheken enthalten sein sollen (siehe [Registerkarte Settings \[► 122\]](#)).
- Beim [Aktivieren der Konfiguration \[► 134\]](#) oder beim [Aktivieren des Bootprojekts \[► 273\]](#), falls für das Target-Archiv konfiguriert ist, dass Bibliotheken enthalten sein sollen (siehe [Registerkarte Settings \[► 122\]](#)).
- Bei Ausführung des [Befehls Projektbibliotheksordner aktualisieren \[► 275\]](#).

Zu den folgenden Zeitpunkten werden nicht mehr verwendete Bibliotheken aus dem `_Libraries` Ordner entfernt:

- Bei Ausführung des [Befehls Projektbibliotheksordner aktualisieren \[► 275\]](#).

#### Siehe auch:

- [Befehl Projektbibliotheken installieren \[► 274\]](#)
- [Befehl Projektbibliotheken installieren \(Unbekannte Versionen\) \[► 275\]](#)
- Doku PLC: Bibliotheken verwenden

## 5.23.7 Befehl Projekt ändern

**Funktion:** Der Befehl öffnet den Standard-Browserdialog, über den eine SPS-Projektdatei gesucht und ausgewählt werden kann. Das aktuell selektierte SPS-Projekt wird anschließend durch das neu ausgewählte SPS-Projekt ersetzt.

**Aufruf:** Kontextmenü des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist ausgewählt.

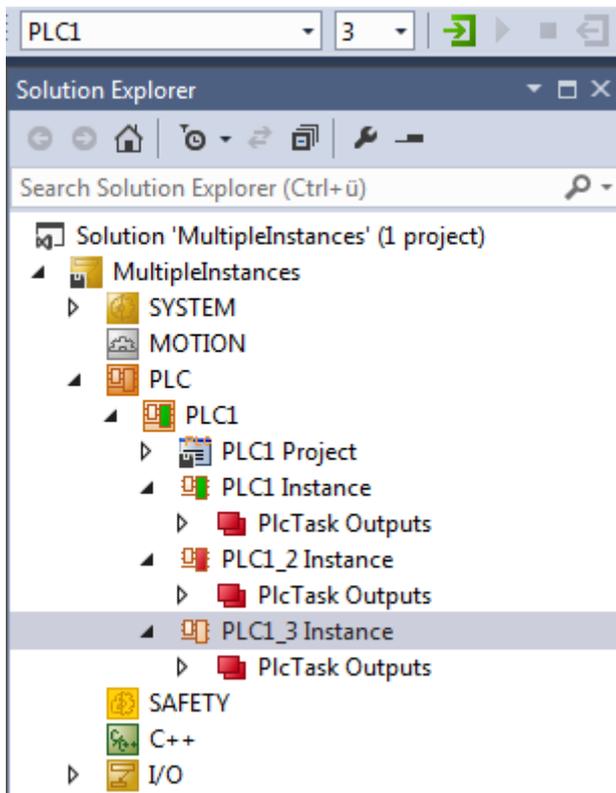
## 5.23.8 Befehl Neues Element hinzufügen (Instanz)

**Funktion:** Mit diesem Befehl kann zu dem ausgewählten SPS-Projekt eine weitere Instanz (Prozessabbild) hinzugefügt werden („Multiple PLC Instances“).

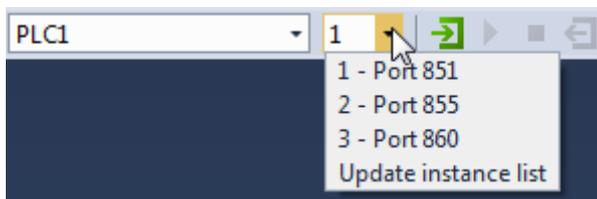
**Aufruf:** Kontextmenü des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist ausgewählt.

Über diesen Mechanismus kann ein SPS-Projekt instanziiert werden, sodass sich pro Projektinstanz ein SPS-Laufzeitsystem ergibt. Für jede Projektinstanz kann eine eigene ADS-Portnummer vergeben werden (siehe [Befehl ADS-Port ändern \[► 274\]](#)). Des Weiteren verfügt jede Projektinstanz über eigene Verknüpfungen innerhalb ihres Prozessabbilds.



Jede Projektinstanz besitzt ihren eigenen Onlinebetrieb, d.h. jede Projektinstanz kann beispielsweise individuell eingeloggt und gestartet werden.



### 5.23.9 Befehl Sichern <SPS-Projektname> als Archiv...

**Funktion:** Der Befehl öffnet den Standarddialog zum Speichern einer Datei als Archiv. Das SPS-Projekt kann unter dem gewünschten Speicherpfad als \*.tzip-Archiv abgelegt werden.

**Aufruf:** Menü **Datei**, Kontextmenü

**Voraussetzung:** Das TwinCAT-SPS-Projekt (<SPS-Projektname>) ist im **Projektmappen-Explorer** ausgewählt.

Die Dateien und Ordner in dem Archivordner sind abhängig von den SPS-Projekteinstellungen.

<b>Inhalt vom *.tzip</b>	Der Archivordner *.tzip enthält das SPS-Projekt, welches archiviert wird.
<b>Befehl zum Öffnen</b>	Ein tzip-Archiv kann über den folgenden Befehl wieder geöffnet werden: <a href="#">Befehl Vorhandenes Element hinzufügen (Projekt) [► 91]</a>
<b>Hinweis zu SPS-Projekten</b>	Die Dateien und Ordner, die bezüglich des SPS-Projekts in dem Archivordner gespeichert werden, sind abhängig von den SPS-Projekteinstellungen dieses SPS-Projekts. <a href="#">Registerkarte Settings [► 122]</a>

### 5.23.10 Befehl Sende per E-Mail.../Send by E-Mail...

Symbol: 

**Funktion:** Der Befehl startet das aktuell im System eingestellte E-Mail-Programm und öffnet eine neue E-Mail. Diese enthält im Anhang die Archivdatei des ausgewählten Projekts.

**Aufruf:** Menü **Datei**, Kontextmenü

### 5.23.11 Befehl Projekt mit Zielsystem aktualisieren...

**Funktion:** Dieser Befehl ermöglicht, das selektierte SPS-Projekt auf den Projektstand des verbundenen Zielsystems zu aktualisieren.

**Aufruf:** Kontextmenü des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist im Projektmappen-Explorer ausgewählt und auf dem verbundenen Zielsystem befinden sich die Sourcen eines SPS-Projekts.

### 5.23.12 Befehl Unabhängige Projektdatei

**Funktion:** Das SPS-Projekt wird explizit in einer separaten (\*.XTI) Datei gespeichert. So werden beispielsweise auch die Verknüpfungen des SPS-Projekts nicht mehr in der TwinCAT-Projektdatei gespeichert, sondern in der XTI-Datei des SPS-Projekts.

**Aufruf:** Kontextmenü des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist ausgewählt.

### 5.23.13 Befehl Deaktivieren

**Funktion:** Das SPS-Projekt wird deaktiviert und wird beispielsweise beim Erstellen des TwinCAT-Projekts nicht berücksichtigt.

**Aufruf:** Kontextmenü des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist ausgewählt.

## 5.24 Kontextmenü SPS-Projektobjekt (<SPS-Projektname> Project)

### 5.24.1 Befehl Einloggen

Symbol: 

**Funktion:** Der Befehl verbindet das Programmiersystem (das ausgewählte SPS-Projekt) mit dem Zielsystem (Steuerung) und stellt somit den Onlinebetrieb her. Eine Instanz des SPS-Projekts wird auf dem Zielsystem erzeugt und geladen.

**Aufruf:** Menü **PLC** oder **TwinCAT SPS Symbolleistenoptionen** oder Kontextmenü des SPS-Projektobjekts (<SPS-Projektname>Project) im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist fehlerfrei und das Zielsystem befindet sich im Run-Modus.

Mögliche Situationen beim Einloggen:

- Das SPS-Projekt existiert noch nicht auf der Steuerung: Sie werden aufgefordert, den Download zu bestätigen.

- Das SPS-Projekt liegt bereits auf der Steuerung und wurde seit dem letzten Download nicht verändert. Das Einloggen erfolgt ohne weitere Interaktion mit Ihnen.
- Das SPS-Projekt liegt bereits auf der Steuerung, wurde aber seit dem letzten Download verändert. Sie werden aufgefordert, eine der folgenden Optionen zu wählen:
  - Mit Online-Change einloggen (Beachten Sie zum Online-Change die Hinweise im Abschnitt „Befehl Online-Change [► 154]“)
  - Mit Download einloggen
  - Ohne Änderung einloggen

An dieser Stelle erhalten Sie außerdem die Möglichkeit, das Bootprojekt auf der Steuerung zu aktualisieren.

- Eine unbekannt Version des SPS-Projekts liegt bereits auf der Steuerung. Sie werden gefragt, ob TwinCAT diese ersetzen soll.
- Eine Version des SPS-Projekts liegt bereits auf der Steuerung und läuft. Sie werden gefragt, ob TwinCAT trotzdem einloggen und das gerade laufende SPS-Programm überschreiben soll.
- Das SPS-Programm auf der Steuerung hält gerade an einem Haltepunkt. Sie haben ausgeloggt und das Programm geändert: TwinCAT warnt Sie, dass im Falle eines Online-Change oder Downloads die SPS komplett angehalten wird. Dies geschieht auch, wenn mehrere Tasks vorhanden sind und nur eine von dem Haltepunkt betroffen ist.

### Übersetzen des Projekts vor dem Einloggen

Wenn ein SPS-Projekt seit seiner letzten Änderung noch nicht kompiliert wurde, übersetzt TwinCAT das Projekt vor dem Einloggen. Dieser Vorgang entspricht dem Befehl **Übersetzen im ausgeloggten Zustand**.

Wenn während des Übersetzens Fehler auftreten, erscheint ein Meldungsdialog. Die Fehler werden in der Ansicht **Fehlerliste** ausgegeben. Sie können dann entscheiden, ob Sie einloggen wollen, ohne das Programm auf die Steuerung zu laden.

#### Siehe auch:

- [Befehl SPS-Projekt erstellen \[► 280\]](#)

### Fehler beim Login

Wenn während des Einloggens auf die Steuerung ein Fehler auftritt, bricht TwinCAT den Ladevorgang mit einer Fehlermeldung ab. Der Fehlerdialog bietet Ihnen die Möglichkeit, die Fehlerdetails anzuzeigen. Ist ein Ausnahmefehler aufgetreten und der Text \*SOURCEPOSITION\* in der Log-Meldung enthalten, können Sie mit dem Befehl **Im Editor anzeigen** die betroffene Funktion im Editor anzeigen. Dabei springt der Cursor an die fehlerverursachende Zeile.

### Ausgabe von Informationen zum Ladevorgang

Wenn TwinCAT das Projekt beim Einloggen auf die Steuerung lädt, werden folgende Informationen im Meldungsfenster ausgegeben:

- Generierte Codegröße
- Größe der globalen Daten
- Resultierender Speicherbedarf auf der Steuerung
- Eine Liste der betroffenen Bausteine (bei Online-Change)

---

**i** Im Onlinebetrieb können Sie die Einstellungen von Geräten oder Modulen nicht verändern. Um Geräteparameter zu ändern, müssen Sie das SPS-Projekt ausloggen. Abhängig vom Bussystem kann es jedoch einige spezielle Parameter geben, die Sie auch im Onlinebetrieb verändern können.

---

**i** Die Konfiguration der Ansicht speichert TwinCAT im Online- und Offlinebetrieb separat voneinander. Zusätzlich werden Ansichten, welche in einer Betriebsart nicht nutzbar sind, geschlossen. Aus diesem Grund kann sich die Ansicht beim Einloggen automatisch ändern.

---

## 5.24.2 Befehl SPS-Projekt erstellen

Symbol: 

**Funktion:** Der Befehl startet den Übersetzungsprozess bzw. die Codeerzeugung für das gerade aktive SPS-Projekt.

**Aufruf:** Menü **Erstellen**, wenn aktuell ein SPS-Projekt selektiert ist, oder Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist selektiert.

Bei der Übersetzung führt TwinCAT eine syntaktische Prüfung aller verwendeten Objekte des SPS-Projekts durch. Der Übersetzungsvorgang wird automatisch immer durchgeführt, wenn Sie das Projekt mit einem geänderten Programm einloggen möchten. Nach Abschluss der Überprüfung zeigt TwinCAT eventuelle Fehlermeldungen oder Warnungen in der Ansicht [Fehlerliste](#) [► 87] an.

Außerdem wird beim Erstellen des Projektes die Übersetzungsinformation des SPS-Projekts angelegt und in einer lokalen Datei (\*.compileinfo) im Projektordner gespeichert.

Wenn das Programm seit dem letzten fehlerfreien Übersetzungsprozess nicht mehr verändert wurde, wird es nicht neu übersetzt. Wenn die syntaktische Prüfung dennoch wiederholt werden soll, verwenden Sie den [Befehl SPS-Projekt neu erstellen](#) [► 280].

## 5.24.3 Befehl SPS-Projekt neu erstellen

**Funktion:** Der Befehl startet den Übersetzungsprozess bzw. die Codeerzeugung für das gerade aktive SPS-Projekt, auch wenn es zuletzt fehlerfrei übersetzt wurde.

**Aufruf:** Menü **Erstellen**, wenn aktuell ein SPS-Projekt selektiert ist, oder Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist selektiert.

Bei einer Neuerstellung des Projekts wird das Projekt zunächst bereinigt (siehe auch: [Befehl SPS-Projekt bereinigen](#) [► 280]) und anschließend erstellt (siehe auch: [Befehl SPS-Projekt erstellen](#) [► 280]).

## 5.24.4 Befehl Überprüfe alle Objekte

**Funktion:** Der Befehl veranlasst einen Übersetzungslauf, also eine Syntaxprüfung, für alle Objekte, die sich im Projektbaum des SPS-Projekts befinden. Dies ist in erster Linie bei der Erstellung von Bibliotheken bzw. bei der Bearbeitung von Bibliotheksprojekten nützlich.

**Aufruf:** Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) im **Projektmappen-Explorer**

Im Gegensatz zu dem [Befehl SPS-Projekt erstellen](#) [► 280], bei dem nur die verwendeten Objekte überprüft werden, werden bei der Ausführung dieses Befehls alle Objekte des SPS-Projekts syntaktisch überprüft.



Der Befehl führt nicht zur Codegenerierung. Es wird auch keine Datei mit Informationen zum Übersetzungslauf im Projektverzeichnis angelegt.

---

## 5.24.5 Befehl SPS-Projekt bereinigen

**Funktion:** Der Befehl aktualisiert das Sprachmodell aller Objekte des gerade aktiven SPS-Projekts.

**Aufruf:** Menü **Erstellen**, wenn aktuell ein SPS-Projekt selektiert ist, oder Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projekt ist selektiert.

Wenn das SPS-Projekt bereinigt wird, wird lediglich das Sprachmodell aller Objekte im SPS-Projekt aktualisiert. Die Übersetzungsinformation auf dem Zielsystem bleibt bestehen.

**Siehe auch:**

- [Befehl SPS-Projekt neu erstellen \[► 280\]](#)

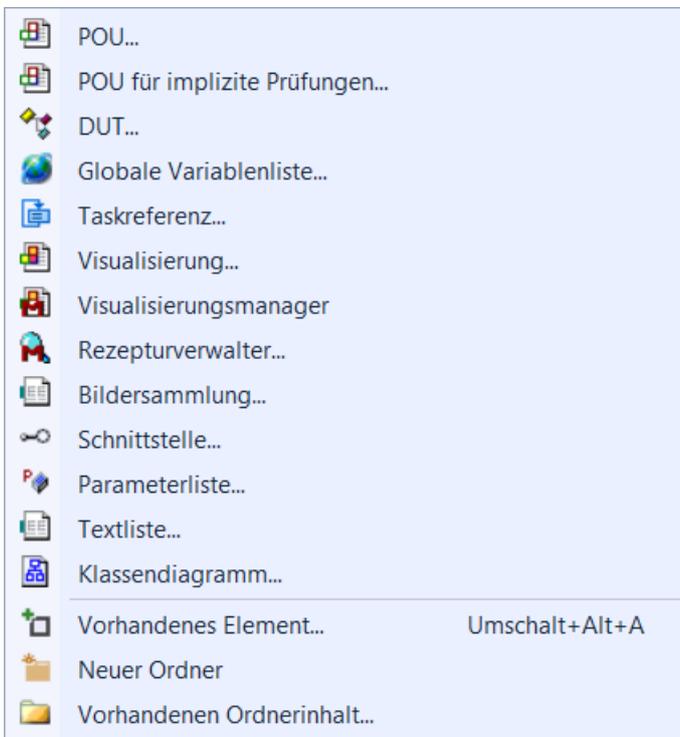
## 5.24.6 Hinzufügen

### 5.24.6.1 Befehl Neues Element hinzufügen (Objekt)

**Funktion:** Der Befehl öffnet ein Untermenü mit Objekten, das abhängig von der aktuellen Position im SPS-Projektbaum alle Objekte enthält, die eingefügt werden können.

**Aufruf:** Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) oder Kontextmenü eines bereits vorhandenen Objekts oder Ordners innerhalb des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Wenn TwinCAT das Objekt in den SPS-Projektbaum einfügen soll, selektieren Sie ein bereits vorhandenes Objekt oder einen Ordner, unterhalb dessen das neue eingerückt angelegt soll.



**Siehe auch:**

- Dokumentation PLC: Objekte hinzufügen
- Dokumentation PLC: Projekt anlegen und konfigurieren

### 5.24.6.2 Befehl Vorhandenes Element hinzufügen (Objekt)

Symbol:

**Funktion:** Der Befehl öffnet den Standard-Browserdialog, über den eine Datei gesucht und zu dem selektierten SPS-Projekt hinzugefügt werden kann.

**Aufruf:** Kontextmenü SPS-Projektobjekt (<SPS-Projektname> Project) oder Kontextmenü eines bestehenden SPS-Ordners im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projektobjekt oder der Ordner innerhalb eines SPS-Projekts ist im TwinCAT-Projektbaum selektiert.

### 5.24.6.3 Befehl Ordner hinzufügen

Symbol: 

**Funktion:** Der Befehl fügt im SPS-Projekt einen neuen Ordner ein.

**Aufruf:** Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) oder Kontextmenü eines bereits vorhandenen Objekts oder Ordners innerhalb des SPS-Projekts im **Projektmappen-Explorer** > Hinzufügen

**Voraussetzung:** Das SPS-Projektobjekt oder der Ordner innerhalb eines SPS-Projekts ist im TwinCAT-Projektbaum selektiert.

Der Befehl fügt den Ordner unterhalb des gerade selektierten Objekts im Projektbaum ein.

**Siehe auch:**

- Dokumentation PLC: Objekte hinzufügen

### 5.24.6.4 Befehl Bestehenden Ordnerinhalt hinzufügen

**Funktion:** Der Befehl öffnet den Standard-Browserdialog, über den ein Ordner auf Dateiebene gesucht und ausgewählt werden kann. Der Inhalt des Ordners wird zu dem selektierten SPS-Projekt hinzugefügt.

**Aufruf:** Kontextmenü SPS-Projektobjekt (<SPS-Projektname> Project) oder Kontextmenü eines bestehenden SPS-Ordners im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projektobjekt oder der Ordner innerhalb eines SPS-Projekts ist im TwinCAT-Projektbaum selektiert.

### 5.24.7 Befehl Als ZIP exportieren

**Funktion:** Der Befehl öffnet den Standarddialog zum Speichern der ausgewählten Objekte im ZIP-Format.

**Aufruf:** Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) oder Kontextmenü eines bereits vorhandenen Ordners innerhalb des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projektobjekt (<SPS-Projektname>Project) oder die SPS-Objekte sind ausgewählt.

**Siehe auch:**

- [Befehl Aus ZIP importieren \[► 282\]](#)
- Dokumentation PLC: SPS-Projekt exportieren und importieren

### 5.24.8 Befehl Aus ZIP importieren

**Funktion:** Der Befehl öffnet den Standarddialog zum Importieren von Objekten aus einer ZIP-Datei.

**Aufruf:** Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) oder Kontextmenü eines bereits vorhandenen Ordners innerhalb des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projektobjekt oder der Ordner innerhalb eines SPS-Projekts ist im TwinCAT-Projektbaum selektiert.

**Siehe auch:**

- [Befehl Als ZIP exportieren \[► 282\]](#)
- Dokumentation PLC: SPS-Projekt exportieren und importieren

### 5.24.9 Befehl PLCopenXML exportieren

Symbol: 

**Funktion:** Der Befehl öffnet einen Dialog zum Exportieren von Objekten eines Projekts in eine XML-Datei im PLCopen-Format.

**Aufruf:** Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) oder Kontextmenü eines bereits vorhandenen Ordners innerhalb des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projektobjekt (<SPS-Projektname>Project) oder die SPS-Objekte sind ausgewählt.

**Siehe auch:**

- [Befehl PLCopenXML importieren \[► 283\]](#)
- Dokumentation PLC: SPS-Projekt exportieren und importieren

### 5.24.10 Befehl PLCopenXML importieren

Symbol: 

**Funktion:** Der Befehl öffnet einen Dialog zum Importieren von Objekten aus einer XML-Datei im PLCopen-Format.

**Aufruf:** Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) oder Kontextmenü eines bereits vorhandenen Ordners innerhalb des SPS-Projekts im **Projektmappen-Explorer**

**Voraussetzung:** Das SPS-Projektobjekt oder der Ordner innerhalb eines SPS-Projekts ist im TwinCAT-Projektbaum selektiert.

#### Dialog PLCopenXML importieren

Der Dialog listet alle Objekte aus der PLCopen-Exportdatei auf, die TwinCAT an dieser Stelle importieren kann.

Aktuell ausgewähltes Zielobjekt	Objekt, das im Projektbaum selektiert ist.
Einfügbare Einträge	Zeigt alle Objekte der Exportdatei, die TwinCAT unterhalb des selektierten Objekts einfügen kann.
Konfliktlösung	Wenn Objekte importiert werden, die den gleichen Namen wie bestehende Objekte haben, kann der Konflikt für jedes Objekt folgendermaßen gelöst werden: <ul style="list-style-type: none"> <li>• Bestehendes Objekt ersetzen: Das im Projekt bestehende Objekt wird mit dem importierten Objekt überschrieben.</li> <li>• Neues Objekt umbenennen: Das neue Objekt wird mit geändertem Namen importiert. Dem Namen wird ein _&lt;nr&gt; angehängt.</li> <li>• Neues Objekt überspringen: Das Objekt wird nicht importiert.</li> </ul>
Auswählen	Öffnet eine Auswahlliste zum Auswählen von Objekttypen.
Auswahl aufheben	Öffnet eine Auswahlliste zur Abwahl von Objekttypen.

**Siehe auch:**

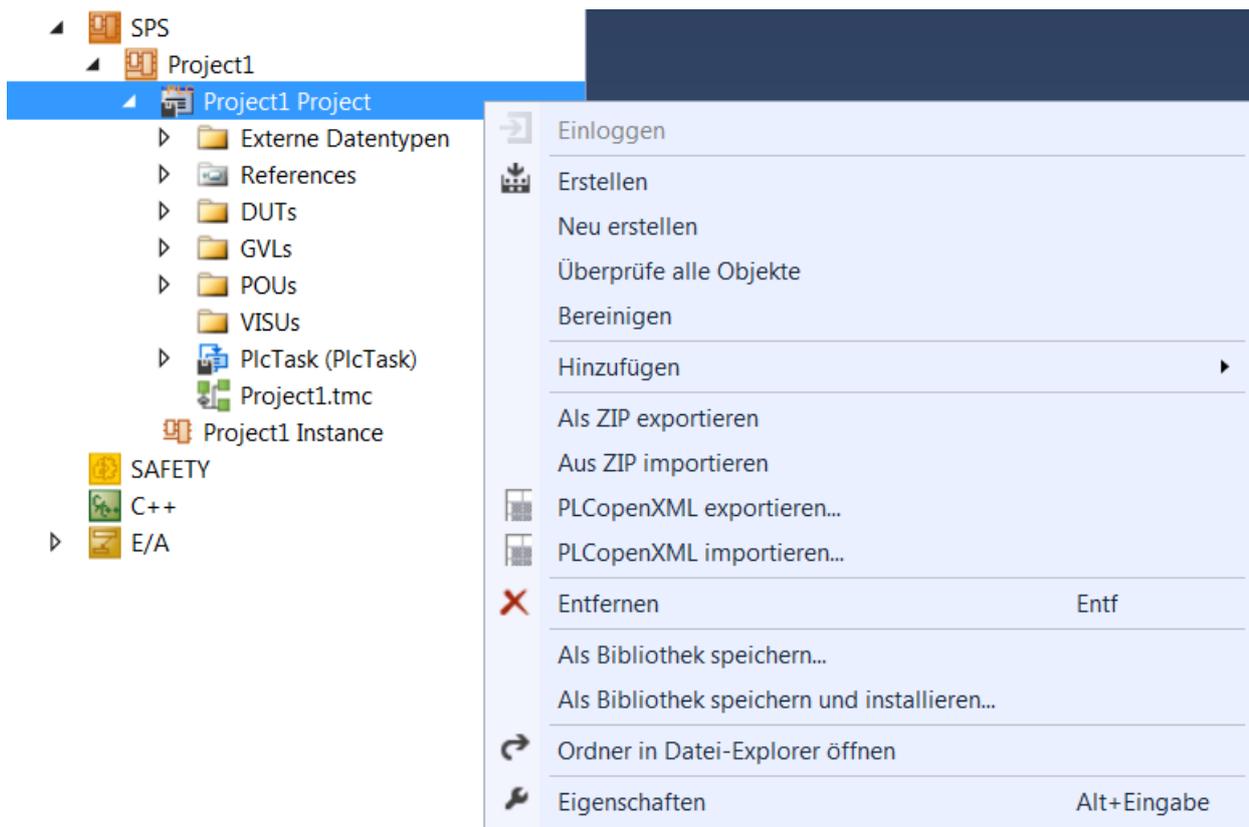
- [Befehl PLCopenXML exportieren \[► 282\]](#)
- SPS-Projekt exportieren und importieren

### 5.24.11 Befehl Als Bibliothek speichern

**Funktion:** Der Befehl öffnet den Standarddialog zum Speichern eines SPS-Projekts als SPS-Bibliothek.

**Aufruf:** Kontextmenü des SPS-Projektobjekts (<SPS-Projektname>Project) im Projektmappen-Explorer

Ein SPS-Projekt kann als SPS-Bibliothek gespeichert werden, um Quellcode für andere Applikationen als Bibliothek und damit über eine definierte Schnittstelle zur Verfügung zu stellen. Der Befehl zum Speichern einer Bibliothek ist im Kontextmenü des SPS-Projekts verfügbar.

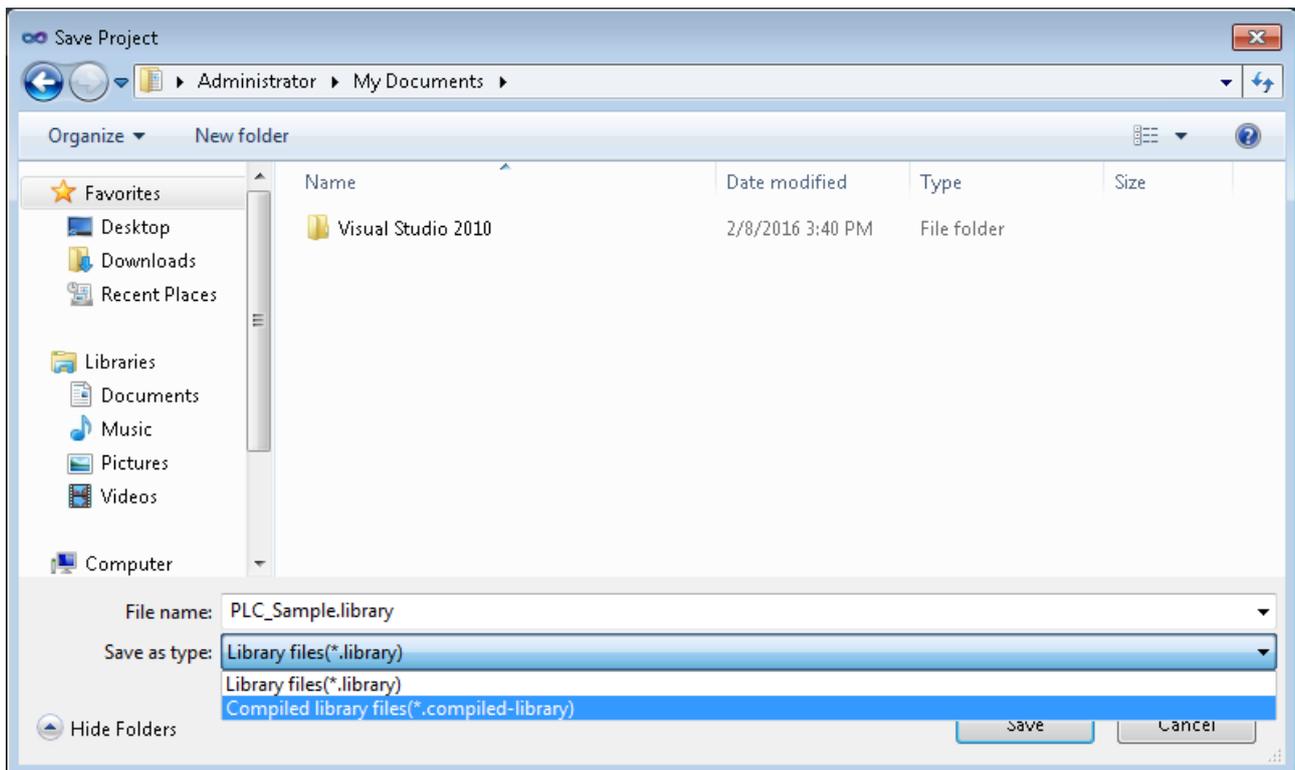


Der Befehl öffnet den Standarddialog zum Speichern einer Datei im Dateisystem. Automatisch wird der bisherige Projektname angeboten – dieser kann bei Bedarf auch verändert werden. Beim Speichern eines Projekts als Bibliothek kann zwischen zwei Bibliotheksdateiformaten gewählt werden:

- \*.library (Source-Bibliothek)
  - Eine Source-Bibliothek können Sie mithilfe des Befehls **Bestehendes Element hinzufügen**, der auf dem SPS-Knoten innerhalb des Projektbaums verfügbar ist, öffnen (zur Einsicht und/oder zur Bearbeitung).
  - Sie können mithilfe der üblichen Debug-Funktionalitäten in eine Source-Bibliothek „hineinstepsen“.
- \*.compiled-library (übersetzte Bibliothek)
  - Mit dieser Dateierweiterung kann ein Bibliotheksprojekt in kompiliertem Format gespeichert werden. Dabei wird ein verschlüsseltes Abbild des Precompile-Kontexts der Bibliothek abgelegt, was bedeutet, dass die Implementierungen der Bibliotheksbausteine nicht mehr zugänglich oder sichtbar sind.
  - Sie können eine übersetzte Bibliothek daher weder öffnen noch debuggen.
  - In der weiteren Handhabung verhalten sich \*.compiled-library-Dateien genauso wie \*.library-Dateien. Sie können sie also auf die gleiche Art und Weise installieren und referenzieren.
  - Durch Verwendung einer übersetzten Bibliothek kann zum einen der Quellcode einer Bibliothek geschützt werden und zum anderen ergibt sich der Vorteil kürzerer Ladezeiten und kleinerer Bibliotheksdateien.

### **i** Schrittweises Durchlaufen des Codes nicht möglich

Die üblichen Debug-Funktionalitäten können auf eine kompilierte Bibliothek (\*.compiled-library) nicht angewendet werden. Es ist somit nicht möglich, in einen Bibliotheksbaustein einer \*.compiled-library per Debugging hineinzuspringen.

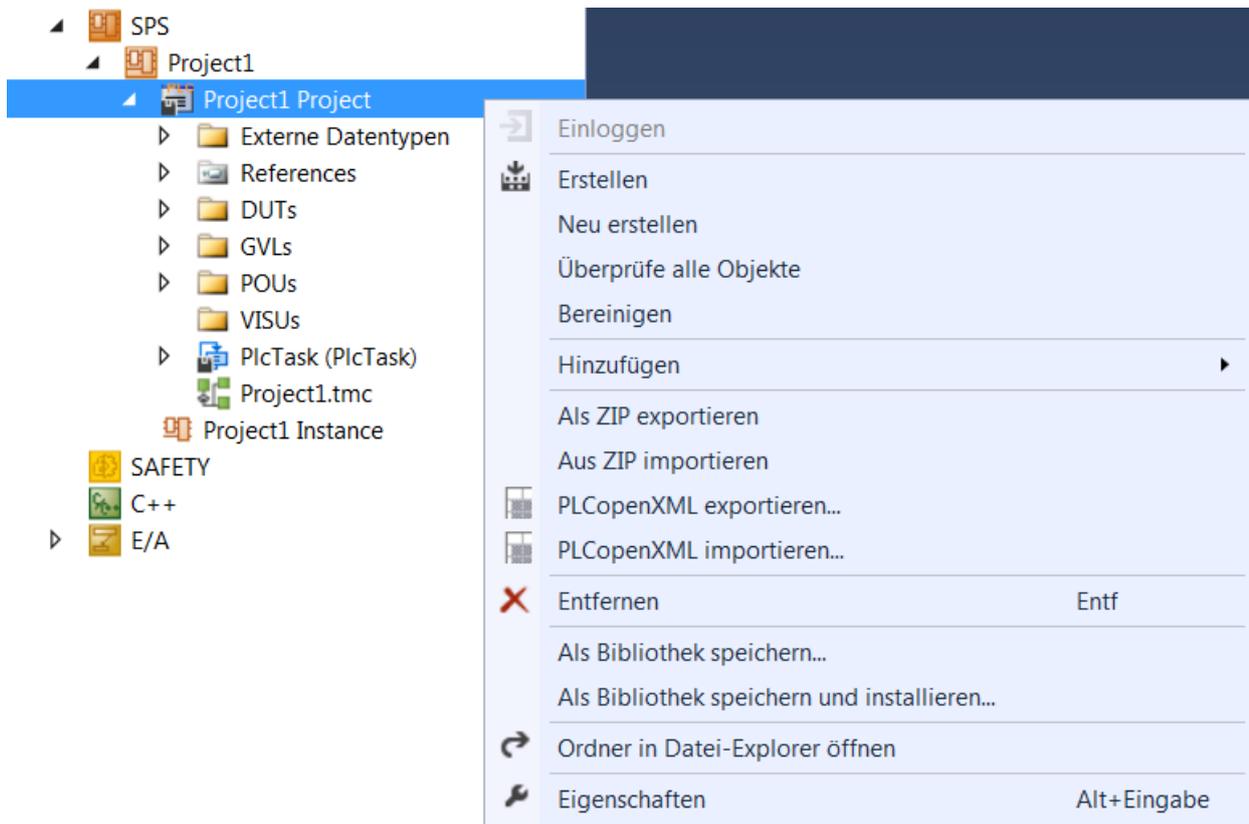


### 5.24.12 Befehl Als Bibliothek speichern und installieren

**Funktion:** Der Befehl öffnet den Standarddialog zum Speichern eines SPS-Projekts als SPS-Bibliothek. Zusätzlich installiert der Befehl die gespeicherte Bibliothek in das Bibliotheksrepository. Die Bibliothek kann damit direkt über den Bibliotheksverwalter in ein Projekt eingefügt werden.

**Aufruf:** Kontextmenü des SPS-Projektobjekts (<SPS-Projektname>Project) im Projektmappen-Explorer

Dieser Befehl speichert das SPS-Projekt als SPS-Bibliothek und installiert sie ins Bibliotheksrepository. Der Befehl zum Speichern und Installieren einer Bibliothek ist im Kontextmenü des SPS-Projekts verfügbar.



Die zusätzlich zur Speicherung durchgeführte Installation der Bibliothek ist eine Erweiterung zu dem [Befehl Als Bibliothek speichern \[► 283\]](#), da die Bibliothek gleichzeitig auf dem lokalen System installiert wird. Dadurch steht die Bibliothek via Bibliotheksverwalter unmittelbar zum Einfügen in einem Projekt zur Verfügung.

### 5.24.13 Befehl Als SPS-Projektvorlage speichern



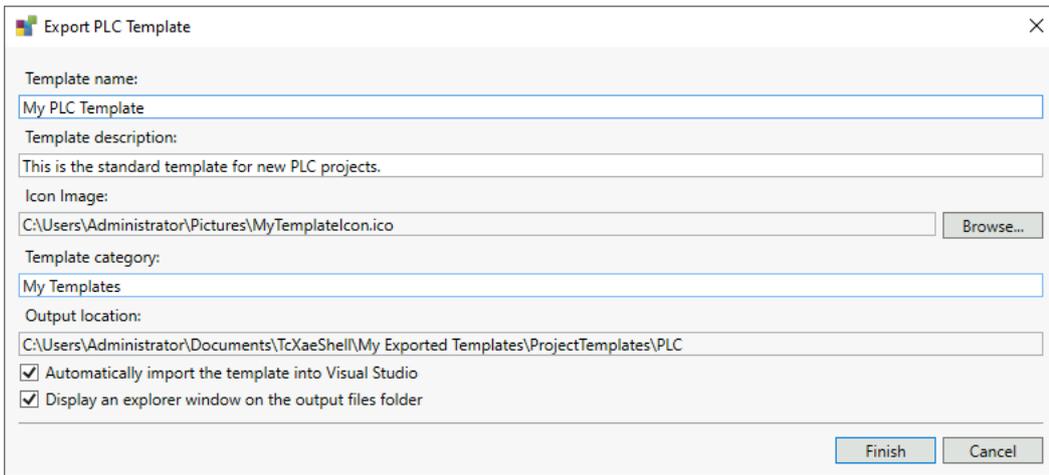
Verfügbar ab TwinCAT 3.1 Build 4026

**Funktion:** Der Befehl öffnet den Dialog zum Speichern eines SPS-Projekts als SPS-Projektvorlage.

**Aufruf:** Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Projekt) im Projektmappen-Explorer

SPS-Projektvorlagen können beim Hinzufügen eines SPS-Projekts im Dialog **Neues Element hinzufügen** ausgewählt werden und bilden den Ausgangspunkt für neue Projekte. In der SPS-Projektvorlage kann die Struktur eines Projekts festgelegt und wiederverwendbarer Code kann eingefügt werden. Außerdem können Sie die SPS-Projekteigenschaften für alle aus der Vorlage erstellten Projekte festlegen.

Der Befehl **Als SPS-Projektvorlage speichern** öffnet den Dialog zum Exportieren einer SPS-Projektvorlage:

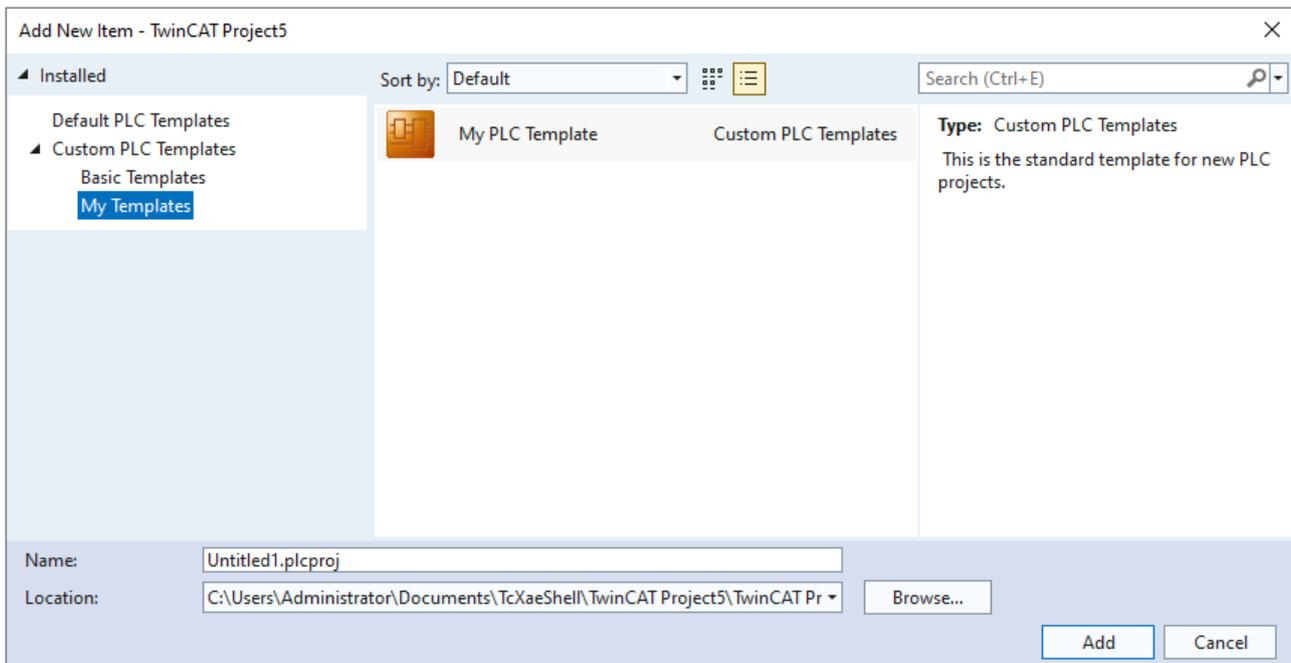


Template name	Name der Projektvorlage
Template description	Beschreibung der Projektvorlage (optional)
Icon image	Symbolbild für die Projektvorlage (optional)
Template category	Kategorie, unter der die Projektvorlage im <b>Add New Item</b> Dialog angezeigt wird (optional).
Output Location	Pfad, unter dem die Projektvorlagen gespeichert werden (nicht veränderbar).
Automatically import the template into Visual Studio	<input checked="" type="checkbox"/> : Die Projektvorlage wird in Visual Studio importiert und im <b>Add New Item</b> Dialog angezeigt.
Display an explorer window on the output files folder	<input checked="" type="checkbox"/> : Ein Explorer-Fenster der Output Location mit der erstellten Projektvorlage öffnet sich.

Mit dem Beenden des Export-Dialogs wird die Projektvorlage als ZIP-Datei unter dem unter „Output Location“ angegebenen Pfad gespeichert (Ordner „My Exported Templates“ im Visual-Studio-Standardprojektverzeichnis). Die ZIP-Dateien können von dort kopiert und auf andere Systeme übertragen werden.

In Visual Studio importierte Projektvorlagen sind im Ordner „Templates“ im Visual-Studio-Standardprojektverzeichnis zu finden. Um eine importierte Projektvorlage zu entfernen, müssen Sie die Vorlage aus dem Ordner „Templates“ löschen und Visual Studio neu starten.

Beim Hinzufügen eines neuen SPS-Projekts werden importierte Projektvorlagen unterhalb der beim Exportieren angegebenen Projektvorlagenkategorie angezeigt:



Das neue Projekt wird auf Basis der gewählten Projektvorlage erstellt.

## 5.24.14 Befehl Eigenschaften (SPS-Projekt)

Symbol:

**Funktion:** Dieser Befehl öffnet ein Editorfenster, in dem die Eigenschaften des Projekts und zusätzliche projektbezogene Informationen angezeigt und definiert werden können.

**Aufruf:** Kontextmenü des SPS-Projektobjekts (<SPS-Projektname> Project) oder Menü **Projekt**

**Voraussetzung:** Ein Projekt ist geöffnet.

TwinCAT speichert die SPS-Projekteigenschaften direkt im SPS-Projekt.

### **i** Geltungsbereich von SPS-Projekteigenschaften

Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

**Siehe auch:**

- Dokumentation PLC: Projekt konfigurieren

### 5.24.14.1 Kategorie Common

Die Kategorie **Common** enthält allgemeine Informationen und Metainformationen der Projektdatei. TwinCAT erstellt aus diesen Informationen Schlüssel in der Registerkarte **Eigenschaften**. Wenn zum Beispiel im Textfeld **Firma** der Name „Company\_A“ eingetragen ist, ist in der Registerkarte **Eigenschaften** der Schlüssel **Company** mit dem Wert „Company\_A“ vorhanden.

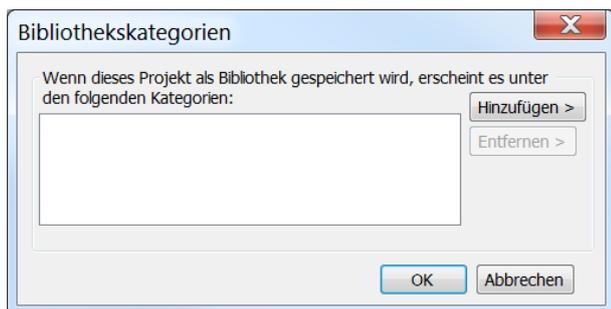
**Projektinformation**

Bei einem Bibliotheksprojekt muss eine Firma, ein Titel und eine Version eingetragen sein, um die Bibliothek installieren zu können.	
Firma	Name der Firma, in der dieses Projekt (Applikation oder Bibliothek) erstellt wurde. Er dient neben der Bibliothekskategorie zur Sortierung im Bibliotheks-Repository
Titel	Titel des Projekts
Version	Version des Projekts, zum Beispiel „0.0.0.1“
Freigegeben	<input checked="" type="checkbox"/> : Schutz gegen Änderung aktiviert. Folge: Wenn Sie nun das Projekt editieren, erscheint eine Eingabeaufforderung, ob das Projekt wirklich geändert werden soll. Wenn Sie diese Abfrage einmalig mit Ja beantworten, erscheinen bei weiteren Editieraktionen keine Abfrage mehr.
Bibliothekskategorien	Kategorien des Bibliotheksprojekts, nach denen Sie im Dialog Bibliotheks-Repository sortieren können. Wenn keine Kategorie angegeben ist, wird der Bibliothek die Kategorie „Sonstige“ zugewiesen. Soll sie einer anderen Kategorie angehören, muss eine solche definiert sein.  Die Definition von Bibliothekskategorien erfolgt in einer oder mehreren externen Beschreibungsdateien im XML-Format. Für die Zuordnung der Bibliothek kann entweder eine solche Datei aufgerufen werden oder aber eine andere Bibliotheksdatei, die bereits selbst die Informationen über die Kategorien aus einer Beschreibungsdatei aufgenommen hat.  Voraussetzung: Das Projekt ist ein Bibliotheksprojekt.
	Der Dialog <b>Bibliothekskategorien</b> öffnet sich, in dem Sie Bibliothekskategorien hinzufügen können.
Standard-Namensraum	Die Standardeinstellung für den Namensraum einer Bibliothek entspricht dem Bibliothekstitel. Für eine Bibliothek kann auch explizit ein davon abweichender Namensraum definiert werden: entweder allgemein für die Bibliothek bei der Bibliothekserstellung an dieser Stelle in den Projektinformationen oder für den lokalen Gebrauch der Bibliothek in einem Projekt im <b>Eigenschaften</b> -Dialog der Bibliotheksreferenz.  Der Namensraum der Bibliothek muss als Präfix des Bezeichners verwendet werden, damit ein eindeutiger Zugriff auf ein Modul möglich ist, das mehrfach im Projekt vorhanden ist, oder wenn der Gebrauch dieses Präfixes durch die Bibliothekseigenschaft LanguageModelAttribute „qualified-access-only“ („Eindeutiger Zugriff auf Bibliotheksmodule oder -variablen“) erzwungen wird.  Wenn Sie hier keinen Standardnamensraum definieren, gilt automatisch der Name der Bibliotheksdatei als Namensraum.
Platzhalter	An dieser Stelle kann ein Standardname des Platzhalters festgelegt werden, der diese Bibliothek repräsentiert bzw. referenziert. Falls ein dieser Stelle nicht explizit ein Platzhalter festgelegt wird, entspricht die Standardeinstellung für den Platzhalternamen einer Bibliothek dem Bibliothekstitel.
Autor	Autor des Projekts
Beschreibung	Kurze Beschreibung des Projekts (z. B. Inhalt, Funktionalitäten, allgemeine Hinweise wie nur für den internen Gebrauch etc.)

**Bibliothekseigenschaften**

Globale Versionsstruktur erzeugen	<input checked="" type="checkbox"/> : Im SPS-Projekt eine globale Variablenliste angelegt, welche die Versionsinformationen enthält. Die globale Variablenliste wird beim Speichern aktualisiert.
Automatisch Bibliotheksinformationen-POU erzeugen	<input checked="" type="checkbox"/> : POU-Objekte vom Typ „Funktion“ werden automatisch im Projektbaum angelegt, die verwendet werden können, um im Applikationsprogramm auf die Projekteigenschaften zuzugreifen. In diesem Fall werden spezielle Funktionen für die Eigenschaften <b>Firma</b> , <b>Title</b> und <b>Version</b> erzeugt (F_GetCompany, F_GetTitle, F_GetVersion). Die Funktionen werden beim Speichern aktualisiert.
Dokumentationsformat	Optionen: <ul style="list-style-type: none"> <li>• Bis TwinCAT 3.1 Build 4024: <b>reStructuredText</b></li> <li>• Ab TwinCAT 3.1 Build 4026: <b>TcDocGen</b></li> </ul> Bei der Bibliothekserstellung werden Kommentare, die einem bestimmten Format entsprechen, neustrukturiert und in dieser angepassten Darstellung in der Registerkarte <b>Dokumentation</b> im Bibliotheksverwalter angezeigt. Dadurch ergeben sich zusätzliche Möglichkeiten der Bibliotheksdokumentation.
Implizite Prüfungen für Compiled Libraries erlauben	Verfügbar ab TwinCAT 3.1 Build 4026 <input checked="" type="checkbox"/> : TwinCAT führt implizite Prüfungen auch für Bausteine aus geschützten Bibliotheken (*.compiled-libraries) aus. Voraussetzung: Die Compiler-Definition „checks_in_libs“ ist in den <b>SPS-Projekteigenschaften</b> im Feld <b>Compilerdefinitionen</b> eingetragen (Kategorie <b>Übersetzen</b> ). Siehe auch: Bausteine für implizite Prüfung verwenden
Qualified_only für Bibliothekszugriff erzwingen	Verfügbar ab TwinCAT 3.1 Build 4026 <input checked="" type="checkbox"/> : Objekte aus dieser Bibliothek dürfen nur mit der Angabe des Namensraums der Bibliothek verwendet werden. Siehe auch: Attribut 'qualified_only'
Referenzierung als Bibliothek erlauben	Verfügbar ab TwinCAT 3.1 Build 4026 <input checked="" type="checkbox"/> : Sie können das SPS-Projekt in einem anderen SPS-Projekt als Bibliothek referenzieren. Siehe auch: SPS-Projekt als referenzierte Bibliothek verwenden

**Dialog Bibliothekskategorien**



Liste von Kategorien	Liste der Kategorien, die dem Bibliotheksprojekt zugewiesen sind. Sie können aus mehrere Quellen stammen. Wenn Sie alle gewünschten Kategorien eingetragen haben, bestätigen Sie den Dialog mit <b>OK</b> .
Hinzufügen	Die Befehle <b>Aus Beschreibungsdatei...</b> und <b>Aus anderer Bibliothek...</b> erscheinen.
Entfernen	TwinCAT entfernt die selektierte Kategorie.
Aus Beschreibungsdatei...	Dialog <b>Beschreibungsdatei auswählen</b> erscheint, in dem Sie eine Beschreibungsdatei mit Erweiterung *.libcat.xml auswählen. Die Datei enthält Befehlskategorien. Wenn Sie den Dialog mit <b>Öffnen</b> beenden, übernimmt TwinCAT diese Kategorien.
Aus anderer Bibliothek...	Dialog <b>Bibliothek auswählen</b> erscheint, in dem Sie eine Bibliothek (*.library) auswählen, deren Befehlskategorien übernommen werden sollen. Wenn Sie den Dialog mit <b>Öffnen</b> beenden, übernimmt TwinCAT die Kategorien.
OK	TwinCAT stellt die Kategorien als Projektinformation zur Verfügung und zeigt sie im Feld <b>Bibliothekskategorien</b> an.
Abbrechen	Schließt den Dialog. Der Vorgang wird abgebrochen.

**Siehe auch:**

- Dokumentation PLC: Projekt konfigurieren
- Dokumentation PLC: Bibliotheken verwenden

**5.24.14.2 Kategorie Übersetzen**

### **i** Geltungsbereich von SPS-Projekteigenschaften

Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

Die Kategorie **Übersetzen** dient der Konfiguration der Compileroptionen.

**Einstellungen**

Compilerdefinitionen	<p>Hier können Compilerdefinitionen/„defines“ (siehe {define}-Anweisungen) und Bedingungen für die Kompilation der Applikation eingetragen werden (bedingte Kompilierung).</p> <p>Eine Beschreibung der verfügbaren bedingten Pragmas finden Sie im Abschnitt <b>Bedingte Pragmas</b>. Der Ausdruck <code>expr</code>, der in solchen Pragmas verwendet wird, kann auch hier eingegeben werden. Mehrere Einträge sind in Form einer komma-separierten Liste möglich.</p>
System-Compilerdefinitionen	<p>Verfügbar ab TwinCAT 3.1 Build 4024</p> <p>Hier werden automatisch die Compilerdefinitionen übernommen, die auf System Manager-Ebene in den SPS-Projekteinstellungen unter <a href="#">Compilerdefinitionen</a> [►_121] gesetzt worden sind.</p>
Download Applikationsinfo	<p>Verfügbar ab TwinCAT 3.1 Build 4024</p> <p>Situation: Sie sind dabei, ein SPS-Projekt auf die Steuerung zu laden, das sich von dem bereits dort liegenden unterscheidet. In diesem Fall erscheint ein Meldungsfenster, das einen <b>Details</b>-Button enthält. Über diesen Button können Sie das <b>Applikationsinformation</b>-Fenster öffnen, das eine Prüfung der Unterschiede zwischen dem aktuellen SPS-Projekt und dem SPS-Projekt auf der Steuerung ermöglicht. Dabei geht es um den Vergleich der Anzahl von Bausteinen, der Daten und der Speicherorte.</p> <p>Das <b>Applikationsinformation</b>-Fenster enthält eine grobe Beschreibung der Unterschiede, beispielsweise:</p> <ul style="list-style-type: none"> <li>• Deklaration von MAIN geändert</li> <li>• Variable fbMyNewInstance in MAIN eingefügt</li> <li>• Anzahl der Methoden/Aktionen von FB_Sample geändert</li> </ul> <p><input checked="" type="checkbox"/> (Voreinstellung): Wenn diese Einstellung aktiviert ist, wird die Information zum Inhalt des SPS-Projekts mit auf die SPS geladen. Dies ermöglicht eine erweiterte Prüfung der Unterschiede zwischen dem aktuellen SPS-Projekt und dem SPS-Projekt auf der Steuerung. Die erweiterte Prüfungsmöglichkeit besteht darin, dass das <b>Applikationsinformation</b>-Fenster den zusätzlichen Reiter <b>Onlinevergleich</b> enthält, welcher eine Baumvergleichsansicht zeigt. Anhand dieser können Sie erkennen, welche POUs geändert, gelöscht oder hinzugefügt wurden. Der zusätzliche Reiter erscheint, wenn Sie den blau unterstrichenen Befehl im unteren Bereich des <b>Applikationsinformation</b>-Fensters ausführen („Applikation nicht aktuell. Code jetzt generieren, um den Onlinevergleich anzuzeigen?“).</p>

<p>Generiere tpy-Datei</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4024</p> <p>Die tpy-Datei enthält u. a. Projekt-, Routing-, Compiler- und Zielsystem-Informationen und ist das Format zur Beschreibung eines TwinCAT 2 SPS-Projekts. Aus Kompatibilitätsgründen mit bestehenden Anwendungen kann diese Datei bei Bedarf für ein TwinCAT 3 SPS-Projekt erstellt werden.</p> <p><input type="checkbox"/> (Voreinstellung): Beim Erstellen des SPS-Projekts wird die zum Projekt gehörende tpy-Datei nicht erstellt.</p> <p><input checked="" type="checkbox"/> : Beim Erstellen des SPS-Projekts wird die zum Projekt gehörende tpy-Datei erstellt und im Projektordner abgelegt.</p> <p>Beachten Sie, dass der Wert und die Konfigurationsverfügbarkeit dieser Option davon abhängen, ob die TPY-Datei als Target-Datei konfiguriert ist oder nicht (siehe <a href="#">Registerkarte Settings</a> [▶ 122]).</p> <ul style="list-style-type: none"> <li>• Wenn die TPY-Datei als Target-Datei aktiviert wird, passiert folgendes:             <ul style="list-style-type: none"> <li>◦ TwinCAT merkt sich den aktuellen Status der Option „Generiere tpy-Datei“ (= „ursprünglicher Wert“, s.u.).</li> <li>◦ Falls dies nicht bereits der Fall ist, wird die Option „Generiere tpy-Datei“ beim nächsten Erstellen des Projekts automatisch aktiviert.</li> <li>◦ Außerdem wird die Option „Generiere tpy-Datei“ ausgegraut, sodass sie vom Anwender nicht deaktiviert werden kann, solange die TPY-Datei als Target-Datei konfiguriert ist.</li> </ul> </li> <li>• Wenn die TPY-Datei anschließend als Target-Datei wieder deaktiviert wird, passiert folgendes:             <ul style="list-style-type: none"> <li>◦ Beim nächsten Erstellen des Projekts erhält die Option „Generiere tpy-Datei“ ihren „ursprünglichen Wert“ (s.o.).</li> <li>◦ Außerdem ist die Option nicht mehr ausgegraut, sodass sie wieder für Konfigurationen durch den Anwender verfügbar ist.</li> </ul> </li> </ul>
<p>Initialisierungswerte von Deklarationen zur TMC-Datei hinzufügen</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4026.12</p> <p><input checked="" type="checkbox"/> (Voreinstellung): Initialisierungswerte von Deklarationen werden in der TMC-Datei gespeichert. Dies ist beispielsweise sinnvoll, wenn die Initialisierungswerte für andere Applikationen benötigt werden.</p> <p><input type="checkbox"/> : Initialisierungswerte von Deklarationen werden nicht mit in der TMC-Datei gespeichert. Dadurch werden die Größe der TMC-Datei und die Übersetzungsdauer verringert, insbesondere wenn die Deklarationen im SPS-Projekt sehr viele Initialisierungswerte enthalten.</p> <p>Beispiel:</p> <pre>VAR   aSample : ARRAY[1..10] OF INT := [2, 38, 5, 9, 74, 62, 87, 3, 16, 4]; END_VAR</pre>
<p>Kommentare über POU- und DUT-Deklarationen zur TMC-Datei hinzufügen</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p><input checked="" type="checkbox"/> (Voreinstellung): Kommentare zu POU- und DUTs, die oberhalb von deren Deklaration stehen, werden mit dem Datentyp in der TMC-Datei gespeichert. Dies ist beispielsweise sinnvoll, wenn die Kommentare für andere Applikationen benötigt werden.</p> <p><input type="checkbox"/> : Kommentare oberhalb der Deklaration von POU- und DUTs werden nicht mit in der TMC-Datei gespeichert. Dadurch werden die Größe der TMC-Datei und die Übersetzungsdauer verringert, insbesondere wenn die Kommentare ausführliche Beschreibungen zu den POU- und DUTs enthalten.</p> <p>Beispiel:</p> <pre>// This function block represents an axis FUNCTION_BLOCK FB_Axis ...</pre>

**Solution Options**

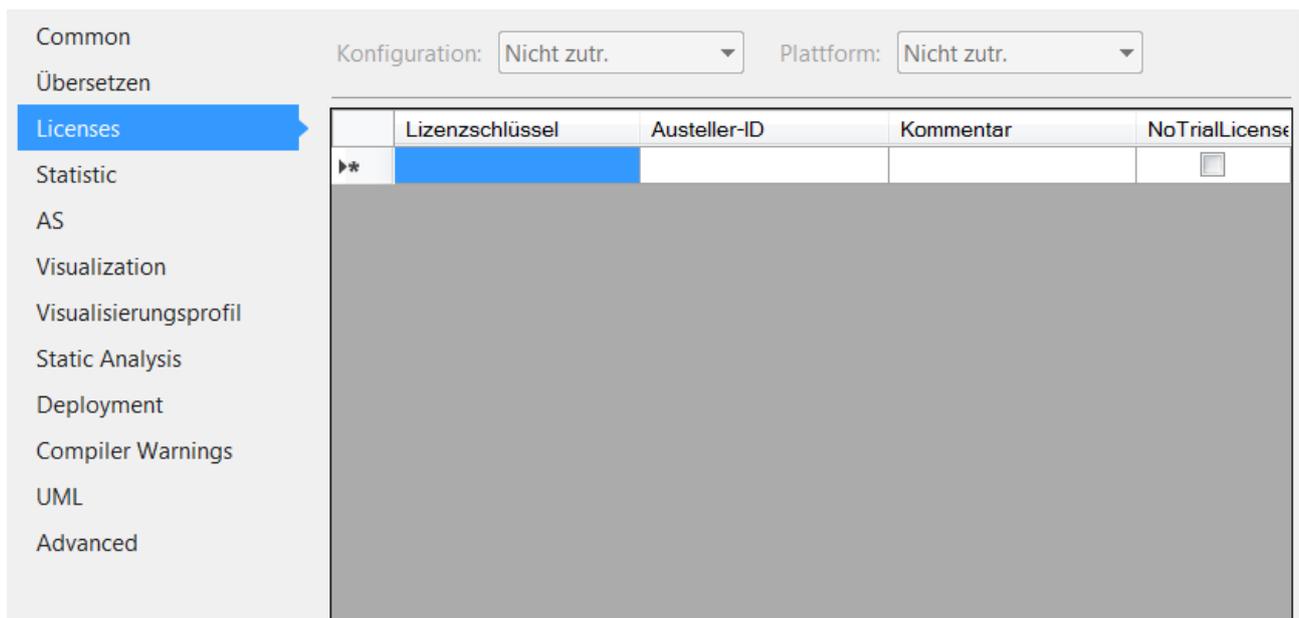
<p>Compilerversion</p>	<p>Verfügbar bis TwinCAT 3.1 Build 4024</p> <p>Definiert die Compilerversion, die TwinCAT beim Übersetzen und während des Ladens zum Übersetzen verwendet.</p> <p>Beachten Sie, dass diese Einstellung nicht den Remote Manager ersetzt. Für die Handhabung verschiedener Engineering-Versionen sollte, wenn es sich bei dem SPS-Projekt um ein Applikationsprojekt handelt, immer der Remote Manager verwendet werden.</p> <p>Die Compilerversion sollte in diesem Fall immer auf „neueste“ stehen.</p> <p>Die Einstellung der Compilerversion kommt nur dann zum Tragen, wenn es sich bei dem SPS-Projekt, welches versionsverwaltet werden soll, um ein Bibliotheksprojekt handelt.</p> <p>Es wird empfohlen, die Bibliothek mit der ältesten Version zu speichern, mit der sie letztlich verwendet werden soll. Dazu muss die Compilerversion auf die entsprechende fixe Version gesetzt werden (z. B. „3.1.4018.0“).</p>
<p>Maximale Anzahl an Warnungen</p>	<p>Bezieht sich auf die Warnungen, die TwinCAT in der Ansicht <b>Fehlerliste</b> maximal ausgibt.</p> <p>Die Auswahl der angezeigten Compilerwarnungen legen Sie im Dialog <b>Projekteinstellungen</b> in der Kategorie <b>Compilerwarnungen</b> fest.</p>
<p>Konstanten ersetzen</p>	<p><input checked="" type="checkbox"/> : Für jede Konstante skalaren Typs, also nicht für STRING, ARRAY oder Strukturen, lädt TwinCAT direkt den Wert. Im Onlinebetrieb kennzeichnet TwinCAT die Konstanten im Deklarationseditor oder Monitoring-Fenster durch ein dem Wert vorangestelltes Symbol. In diesem Fall ist ein Zugriff, zum Beispiel über einen ADR-Operator, Forcen und Schreiben, nicht möglich.</p> <p><input type="checkbox"/> (Voreinstellung): Der Zugriff auf Konstanten ist möglich. Die Rechenzeit verlängert sich minimal.</p>

**Siehe auch:**

- [Kategorie Compiler Warnings \[▶ 300\]](#)

**5.24.14.3 Kategorie Lizenzen**

Reservierung für zukünftige Nutzung!



Hinweis: Die Abfrage einer OEM-Lizenz für eine eigene Bibliothek muss immer im Code der Bibliothek durch den Anwender erfolgen. Siehe Abfrage einer OEM-Lizenz in der PLC Applikation.

### 5.24.14.4 Kategorie Statistic

Die Kategorie **Statistic** gibt eine statistische Auskunft, wie viele Objekte der einzelnen Typen im Projekt verwendet sind.

Konfiguration: Nicht zutr. Plattform: Nicht zutr.

Anzahl der Objekte: 12

Objekttyp	Anzahl
Aktion	2
Methode	2
Ordner	4
POU	3
Referenced Task	1

### 5.24.14.5 Kategorie AS

#### ● Geltungsbereich von SPS-Projekteigenschaften

**I** Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

Die Kategorie **AS** dient der Konfiguration der Einstellungen für AS-Objekte. Jedes neue AS-Objekt hat die konfigurierten Einstellungen automatisch in seinen Eigenschaften.

#### Registerkarte Variablen

Konfiguration: Nicht zutr. Plattform: Nicht zutr.

**Solution options**

Variablen Übersetzen

Aktiv	Variable	Deklarieren	Beschreibung
<input type="checkbox"/>	SFCInit	<input checked="" type="checkbox"/>	Alle Schritte und Aktionen werden zurückgesetzt. Der Initialschritt wird aktiviert. Keine Aktionen werden ausgeführt.
<input type="checkbox"/>	SFCReset	<input checked="" type="checkbox"/>	Alle Schritte und Aktionen werden zurückgesetzt. Der Initialschritt wird aktiviert und seine Aktionen werden ausgeführt.
<input type="checkbox"/>	SFCError	<input checked="" type="checkbox"/>	Wird 'TRUE', falls ein Zeitüberwachung fehlschlägt.
<input type="checkbox"/>	SFCEnableLimit	<input checked="" type="checkbox"/>	Aktiviert die Zeitüberwachung für Schritte.
<input type="checkbox"/>	SFCErrorStep	<input checked="" type="checkbox"/>	Enthält den Namen des Schritts, welcher SFCError auf 'TRUE' gesetzt hat. SFCError muss aktiviert sein.
<input type="checkbox"/>	SFCErrorPOU	<input checked="" type="checkbox"/>	Enthält den Namen der POU, welche SFCError auf 'TRUE' gesetzt hat. SFCError muss aktiviert sein.
<input type="checkbox"/>	SFCQuitError	<input checked="" type="checkbox"/>	Die Ausführung wird angehalten. SFCError wird zurückgesetzt. SFCError muss aktiviert sein.
<input type="checkbox"/>	SFCPause	<input checked="" type="checkbox"/>	Die Ausführung wird angehalten. SFCError wird zurückgesetzt.
<input type="checkbox"/>	SFCTrans	<input checked="" type="checkbox"/>	Wird 'TRUE', falls eine Transition schaltet.
<input type="checkbox"/>	SFCCurrentStep	<input checked="" type="checkbox"/>	Enthält den Namen des aktiven Schritts.
<input type="checkbox"/>	SFCtip	<input checked="" type="checkbox"/>	Schaltet mit die nächste Transition bei einer steigenden Flanke.
<input type="checkbox"/>	SFCtipMode	<input checked="" type="checkbox"/>	Falls 'TRUE', können Transitionen nur mittels SFCtip geschaltet werden.
<input type="checkbox"/>	SFCErrorAnalyzation	<input checked="" type="checkbox"/>	Enthält die möglichen Variablen, welche SFCError auf TRUE gesetzt haben, als Zeichenkette dargestellt. SFCError ist erforderlich
<input type="checkbox"/>	SFCErrorAnalyzationTable	<input checked="" type="checkbox"/>	Enthält die möglichen Variablen, welche SFCError auf TRUE gesetzt haben, in einer Tabelle. SFCError ist erforderlich

Implizit erzeugte Variablen (Flags) zur Kontrolle und zum Monitoring der Abarbeitung in einem AS-Diagramm.	
Aktiv	 : Die entsprechende Variable wird verwendet.
Deklarierten	 : Die entsprechende Variable wird automatisch angelegt. Ansonsten, falls die Verwendung beabsichtigt ist (Aktiv ist gesetzt), muss der Anwender sie selbst deklarieren.



Eine automatisch deklarierte Flag-Variable ist nur im Onlinebetrieb im Deklarationsteil des AS-Editors sichtbar.

**Registerkarte Übersetzen**

**Codegenerierung**

Nur aktive Transitionen berechnen	 : TwinCAT erzeugt nur für gerade aktive Transitionen Code.
-----------------------------------	--

**Siehe auch:**

- [AS-Flags](#)

**5.24.14.6 Kategorie Visualization**



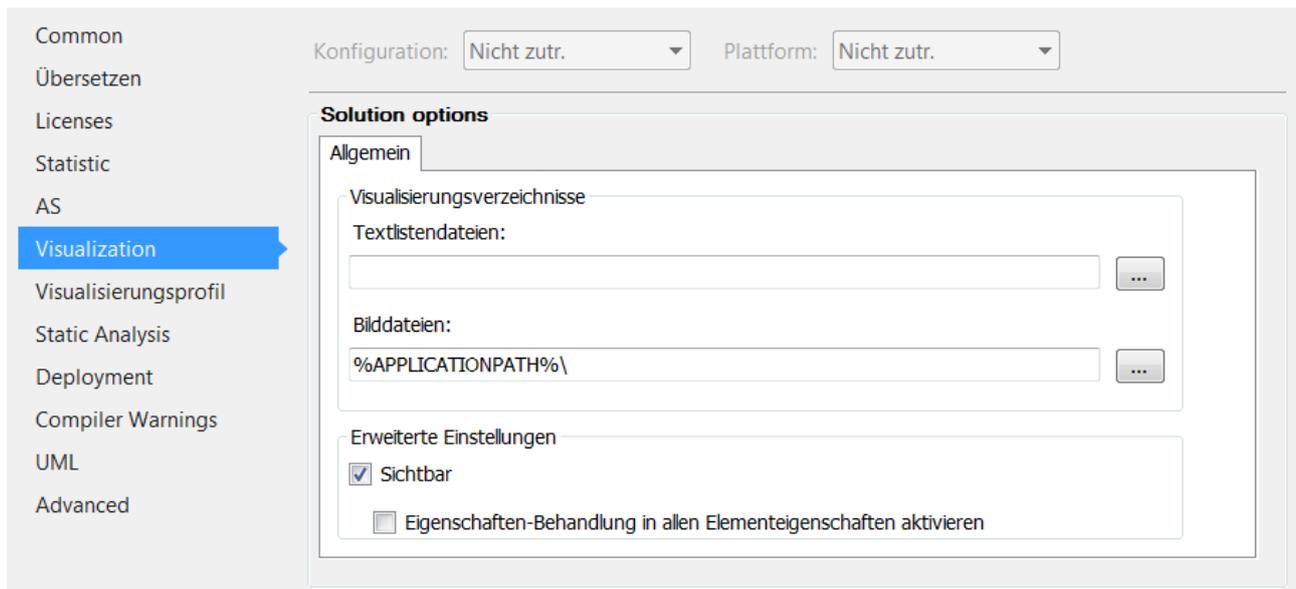
**Geltungsbereich von SPS-Projekteigenschaften**

Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

Die Kategorie **Visualization** dient der Konfiguration der projektweiten Einstellungen für Objekte des Typs Visualisierung.



**Registerkarte Allgemein**

**Visualisierungsverzeichnisse**

Textlistendateien	<p>Verzeichnis, das Textlisten enthält, die im Projekt zur Verfügung stehen, um Texte für verschiedene Sprachen zu konfigurieren. TwinCAT verwendet das Verzeichnis zum Beispiel beim Exportieren oder Importieren von Textlisten.</p> <p>Mit Klick auf  erscheint der Dialog <b>Ordner suchen</b>, der das Auswählen eines Verzeichnisses im Dateisystem ermöglicht.</p>
Bilddateien	<p>Verzeichnis, das Bilddateien enthält, die im Projekt zur Verfügung stehen. Mehrere Ordner sind mit einem Strichpunkt voneinander getrennt. TwinCAT verwendet das Verzeichnis zum Beispiel beim Exportieren oder Importieren von Bilddateien.</p> <p>Mit Klick auf  erscheint der Dialog <b>Ordner suchen</b>, der das Auswählen eines Verzeichnisses im Dateisystem ermöglicht.</p>

**Erweiterte Einstellungen**

Eigenschaften-Behandlung in allen Elementeigenschaften aktivieren	<p> : Sie können ein Visualisierungselement in denjenigen seiner Eigenschaften, in denen Sie eine IEC-Variable auswählen, auch mit einer Eigenschaft  konfigurieren. TwinCAT erzeugt dann beim Übersetzen einer Visualisierung zusätzlichen Code für die Eigenschaften-Behandlung.</p> <p>Voraussetzung: Ihr IEC-Code enthält mindestens ein Objekt des Typs Schnittstelleneigenschaft, also eine Eigenschaft  .</p> <p> MAIN (PRG)   Property_A   Get   Set</p> <p>Voraussetzung: <b>Sichtbar</b> ist aktiviert.</p>
---	--

### 5.24.14.7 Kategorie Visualisierungsprofil

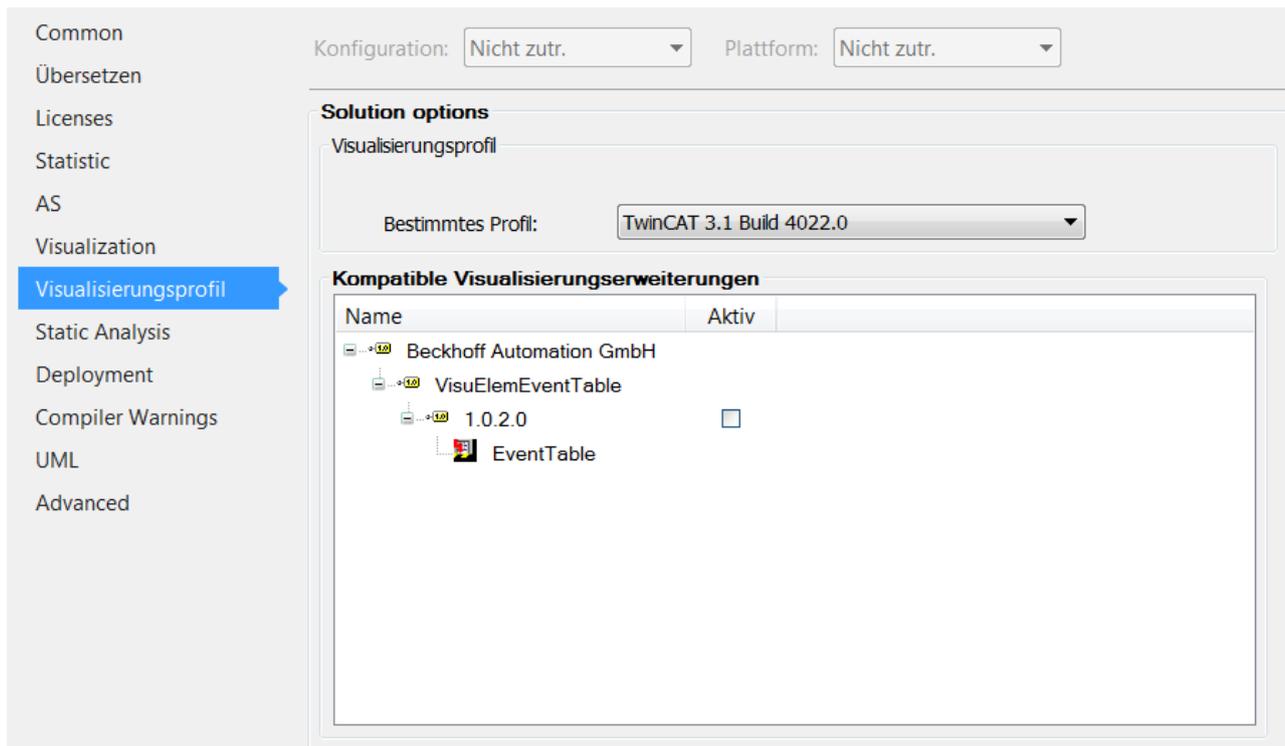
**i Geltungsbereich von SPS-Projekteigenschaften**

Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

Die Kategorie **Visualisierungsprofil** ermöglicht die Einstellung des Visualisierungsprofils.



#### Visualisierungsprofil

Bestimmtes Profil	Profil, das TwinCAT im Projekt verwendet und das die Visualisierungselemente bestimmt, die im Projekt zur Verfügung stehen. Die Auswahlliste enthält alle bisher installierten Profile.
-------------------	--

### 5.24.14.8 Kategorie Static Analysis

**i Geltungsbereich von SPS-Projekteigenschaften**

Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

Die Kategorie **Static Analysis** definiert die Prüfungen, die bei der statischen Codeanalyse berücksichtigt werden.

**Static Analysis Light:**

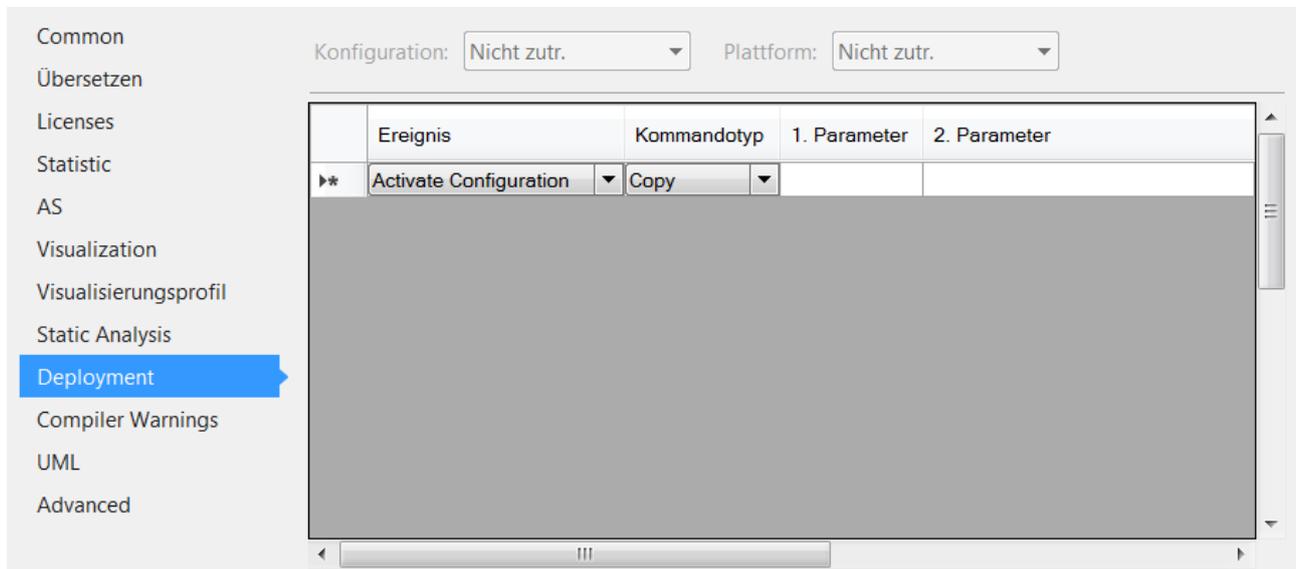
- Wenn Sie die zusätzliche Engineering-Lizenz TE1200 nicht aktiviert haben, können Sie die lizenzfreie Variante des Static Analysis (Static Analysis Light) nutzen, welche einige wenige Kodierregeln beinhaltet. Anhand der kostenfreien Light-Variante können Sie sich – auf Basis eines stark reduzierten Funktionsumfangs – beispielsweise mit dem prinzipiellen Handling des Produkts vertraut machen.
- Weitere Informationen zu Static Analysis Light finden Sie unter: Dokumentation PLC: SPS-Projekt programmieren > Syntax prüfen und Code analysieren > Codeanalyse (Static Analysis)

**Static Analysis Full:**

- Wenn Sie die zusätzliche Engineering-Lizenz TE1200 aktiviert haben, steht Ihnen der volle Funktionsumfang des Static Analysis zur Verfügung (Speichern und Laden von Einstellungen, mehr als 100 Kodierregeln, Namenskonventionen, Metriken, unzulässige Symbole).
- Weitere Informationen zu Static Analysis Full finden Sie unter: TE1200 Static Analysis.

**5.24.14.9 Kategorie Deployment**

Die Kategorie **Deployment** dient der Einstellung von Befehlen, welche während des Aufspielen und Startens einer Anwendung ausgeführt werden sollen.



Folgende Ereignisse stehen zur Verfügung, nach welchen die in der Liste aufgeführten Kommandos aufgerufen werden können:

Activate Configuration	Das gewünschte Kommando wird nach einem Aktivieren der Konfiguration aufgerufen.
Plc Download	Das gewünschte Kommando wird nach einem erfolgten Download der SPS-Applikation auf das Zielsystem aufgerufen.
Plc Online Change	Das gewünschte Kommando wird nach einem erfolgten Online Change aufgerufen.
Plc After Compile	Das gewünschte Kommando wird nach einem Kompilervorgang der SPS-Applikation aufgerufen.

Folgende Kommandos können ausgeführt werden:

Copy	Kopiert Dateien von Parameter 1 (Quellpfad) nach einem in Parameter 2 (Zielpfad) angegebenen Ort.
Execute	Führt die unter Parameter 1 aufgeführte Applikation bzw. das aufgeführte Skript aus.

Quell- und Zielpfade können virtuelle Umgebungsvariablen beinhalten, welche von TwinCAT entsprechend aufgelöst werden.

Die nachfolgenden Umgebungsvariablen werden u. a. unterstützt:

Virtuelle Umgebungsvariable	Registrierungswert	Defaultwert
%TC_INSTALLPATH%	InstallDir	C:\TwinCAT3.x \
%TC_TARGETPATH%	TargetDir	C:\TwinCAT3.x \Target\
%TC_BOOTPRJPATH%	BootDir	C:\TwinCAT3.x \Boot\
%TC_RESOURCEPATH%	ResourceDir	C:\TwinCAT3.x \Target\Resource\
%SOLUTIONPATH%	-	Speicherort der Solution-Datei

Die Registrierungswerte werden unter folgendem Schlüssel in der Registry abgelegt:  
**HKLM\Software\Beckhoff\TwinCAT3.**

**Beispiel:**

Im nachfolgenden Beispiel wird die Datei *SampleFile.xml* aus dem Projektunterordner Config der Solution in den Ordner *C:\plc\config* auf dem Zielsystem kopiert.

Ereignis	Kommandotyp	1. Parameter	2. Parameter
Activate Configuration	Copy	%SOLUTIONPATH%\Config\SampleFile.xml	C:\plc\Config\SampleFile.xml

### 5.24.14.10 Kategorie Compiler Warnings

Die Kategorie **Compiler Warnings** dient der Auswahl der Compilerwarnungen, die TwinCAT bei einem Kompilierungslauf im Meldungsfenster anzeigt.



Die maximale Anzahl aufgelisteter Warnungen legen Sie in der Kategorie **Compile** fest.

The screenshot shows the 'Compiler Warnings' configuration window. On the left, a sidebar lists various settings categories, with 'Compiler Warnings' highlighted. The main area displays a list of warning messages, each with a checkbox for selection. The list includes messages such as 'C0353. Typ ... ist möglicherweise nicht in Typ ... konvertierbar', 'C0100. Bibliothek ... wurde nicht zum Bibliothekverwalter hinzugefügt', and 'C0220. Das Gerät ist nicht installiert'. The 'Compile' category is currently selected, and its settings are visible in the right-hand pane.

**Siehe auch:**

- [Befehl SPS-Projekt erstellen \[► 280\]](#)
- [Kategorie Übersetzen \[► 291\]](#)

### 5.24.14.11 Kategorie UML

#### **i** Geltungsbereich von SPS-Projekteigenschaften

Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

In der Kategorie **UML** können Sie die UML-Compiler-Version ändern. Diese Einstellung ist nur bei Verwendung des UML-Zustandsdiagramms relevant.

Weitere Informationen zu den Konfigurationsmöglichkeiten finden Sie im Abschnitt „UML Compiler-Version“ der Dokumentation TF1910 TC3 UML.

The screenshot shows the configuration interface for the UML category. On the left, a sidebar lists various categories, with 'UML' highlighted in blue. The main area shows configuration options for 'UML'. At the top, there are dropdown menus for 'Konfiguration:' (set to 'Nicht zutr.') and 'Plattform:' (set to 'Nicht zutr.'). Below this, the 'Solution options' section is expanded, displaying a table of settings:

Property	Value
UML compiler version in project	4.0.2.1
Recommended, newest version	4.0.2.1
Action	Do not update.

### 5.24.14.12 Kategorie Advanced

#### **i** Geltungsbereich von SPS-Projekteigenschaften

Beachten Sie, dass sich der Geltungsbereich zwischen verschiedenen Projekteigenschaften unterscheidet!

Einige Eigenschaften betreffen nur das SPS-Projekt, dessen Eigenschaften Sie aktuell konfigurieren.

Andere Eigenschaften wirken sich hingegen auf alle SPS-Projekte aus, die sich in der Entwicklungsumgebung befinden. Solche Eigenschaften, die Sie zwar in den Projekteigenschaften eines SPS-Projekts ändern, die aber auch alle anderen SPS-Projekte betreffen, sind mit der Überschrift **Solution options** betitelt.

Die Kategorie **Advanced** dient der Konfiguration von erweiterten Eigenschaften.

#### **Write options**



---

**Engineering-Inkompatibilität von Dateiversion 1.2.0.0 (oder größer) mit TwinCAT 3.1 < Build 4024**

Beachten Sie, dass Objekte, die mit der Dateiversion 1.2.0.0 (oder größer) gespeichert werden, nicht mit Engineering-Versionen < TwinCAT 3.1.4024 geladen werden können!

Da ein Objekt bei Verwendung des optionalen Speicherformats „Base64“ automatisch mit der Dateiversion 1.2.0.0 gespeichert wird, können Objekte mit Base64-Speicherformat folglich nicht mit Engineering-Versionen < TwinCAT 3.1.4024 geladen werden.

Falls ein SPS-Projekt sowohl Objekte mit der Dateiversion 1.1.0.1 als auch Objekte mit der Dateiversion 1.2.0.0 enthält, werden die 1.1.0.1-Objekte mit einer Engineering-Version < TwinCAT 3.1.4024 weiterhin geladen. Lediglich die Objekte mit der Dateiversion 1.2.0.0 werden nicht geladen.

Die Dateiversion einer Datei, die mit der Dateiversion 1.2.0.0 gespeichert wurde, kann mit XAE-Versionen >= TwinCAT 3.1.4024 nachträglich wieder auf 1.1.0.1 gesetzt werden.

---

Objektinhalt schreiben als  
(„Write object content as“)

Verfügbar ab TwinCAT 3.1 Build 4024

#### Hintergrundinformationen:

Ab Build 4024 wird mit **Base64** ein neues Speicherformat eingeführt, das für die folgenden SPS-Objekte optional zur Verfügung steht:

- POUs, bei denen der POU-Rumpf in einer grafischen Implementierungssprache programmiert ist
  - AS (Ablaufsprache)
  - FUP/KOP/AWL (Funktionsplan/Kontaktplan/Anweisungsliste)
  - CFC (Continuous Function Chart und Seitenorientierter CFC)
  - UML Klassendiagramm und Zustandsdiagramm
- POUs, die über ein Unterelement (z. B. Aktion, Methode) verfügen, das in einer grafischen Implementierungssprache programmiert ist (grafische Sprachen siehe erster Stichpunkt)
- Visualisierungen
- Visualisierungsmanager
- Textlisten
- Rezeptmanager
- Bildersammlungen

Bislang wurden diese Objekte standardmäßig als XML gespeichert.

Ob diese Objekttypen ab Build 4024 als XML oder als Base64 gespeichert werden sollen, ist konfigurierbar.

#### Vorteile von Base64 gegenüber XML:

Im Vergleich zu XML ergibt sich mit Base64 eine komprimierte Speicherung. Als Folge dessen kann bei Dateizugriffen auf diese Objekte eine verbesserte Performance erreicht werden, welche beispielsweise beim Projektladen oder beim Verschieben/Kopieren der Objekte zum Tragen kommt.

#### Einstellungsmöglichkeit des Standard-Speicherformats:

Über die Einstellung „Objektinhalt schreiben als“ in den SPS-Projekteigenschaften können Sie für ein SPS-Projekt definieren, welches das standardmäßige Speicherformat für die oben genannten Objekttypen ist.

Das ausgewählte Standard-Speicherformat wird nur bei neu hinzugefügten Objekten verwendet (Ausnahme: nicht bei neu hinzugefügten POU-Unterelementen. Beispiel: Eine POU ist als XML gespeichert und das Standard-Speicherformat ist als Base64 konfiguriert. Wenn zu der POU dann ein grafisches Unterelement hinzugefügt wird, bleibt das Speicherformat der POU und damit auch für das Unterelement als XML bestehen).

Das Speicherformat eines existierenden Objekts mit einem vom Standard abweichenden Speicherformat wird nicht automatisch geändert, wenn das Objekt geändert und gespeichert wird. Das Speicherformat eines existierenden Objekts kann entweder individuell über das Eigenschaften-Fenster geändert werden (siehe unten) oder alternativ besteht beim Umstellen des Standard-Speicherformats die Möglichkeit, das neu ausgewählte Speicherformat auch für alle bereits bestehenden Objekte zu übernehmen. Wenn Sie an dieser Stelle das Speicherformat umstellen, erscheint ein entsprechendes Abfragefenster.

Für die Einstellung „Objektinhalt schreiben als“ stehen die folgenden Auswahlmöglichkeiten zur Verfügung:

- **XML** (Voreinstellung): Die oben genannten SPS-Objekte werden standardmäßig im XML-Format gespeichert.
  - Objekte mit diesem Speicherformat werden in der Dateiversion 1.1.0.1 gespeichert.
- **Base64**: Die oben genannten SPS-Objekte werden standardmäßig im Base64-Format gespeichert.
  - Objekte mit diesem Speicherformat werden in der Dateiversion 1.2.0.0 gespeichert.  
Bitte beachten Sie, dass Objekte mit der Dateiversion 1.2.0.0 (oder größer) nicht mit Engineering-Versionen < TwinCAT 3.1.4024 geladen werden können!

#### Individuelle Einstellungsmöglichkeit des Speicherformats:

<p>Produktversion in Dateien schreiben („Write product version in files“)</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4024</p> <p>Die Produktversion repräsentiert, mit welcher Plugin-Version eine SPS-Datei (z. B. ein Funktionsbaustein) gespeichert wurde. Die Einstellung dieser Checkbox ist projektweit gültig und ist die Standard-Einstellung für alle geänderten oder neu hinzugefügten SPS-Objekte, die sich in diesem SPS-Projekt befinden.</p> <p><input type="checkbox"/> (Voreinstellung): Die Produkt- bzw. Plugin-Version wird nicht mit in die Datei geschrieben.</p> <ul style="list-style-type: none"> <li>• Wenn Sie die Einstellung von aktiviert nach deaktiviert ändern, erscheint ein Abfragefenster, in dem Sie auswählen können, ob die Produktversion aus allen bestehenden Dateien entfernt werden soll.</li> <li>• Anwendungsfall für die deaktivierte Option: Wenn die Produktversion nicht von Interesse ist, kann diese Einstellung verwendet werden. Dadurch werden Änderungen von Dateien hinsichtlich Sourcecode-Verwaltungssystemen minimiert.</li> </ul> <p><input checked="" type="checkbox"/> : Die Produkt- bzw. Plugin-Version wird mit in die Datei geschrieben (im XAE ist die Version nicht sichtbar, bei Analyse der Datei auf Dateiebene ist sie erkennbar).</p> <ul style="list-style-type: none"> <li>• Wenn Sie die Einstellung von deaktiviert nach aktiviert ändern, erscheint ein Abfragefenster, in dem Sie auswählen können, ob die Produktversion zu allen bestehenden Dateien hinzugefügt werden soll.</li> <li>• Anwendungsfall für die aktivierte Option: Diese Einstellung kann verwendet werden, um die Dateiversion beispielsweise zu Debug- oder Nachverfolgungszwecken mit in die Datei zu schreiben.</li> <li>• Beachten Sie folgendes: Wenn die Datei mit einer anderen Produktversion gespeichert wird, führt dies zu einer Änderung dieser Datei, welche bei Verwendung von Sourcecode-Verwaltungssystemen als Dateiu Unterschied erkenntlich ist.</li> </ul>
<p>Objekthalt mit Profil schreiben: (“Write object content with Profile”)</p>	<p>Das Profil definiert das Format, in dem Objekte abgespeichert werden. Mit dem Build 4024 wurden beispielsweise neue Funktionalitäten für das PLC HMI hinzugefügt. Aus diesem Grund können Visualisierungsdateien, die mit dem Build 4024 abgespeichert worden sind, nicht direkt mit älteren Builds geöffnet werden. Wenn Sie hier ein 4022 Profil einstellen, dann werden die Visualisierungsdateien in dem passenden Format abgespeichert und können mit dem Build 4022 geöffnet werden.</p> <p><b>Voraussetzung:</b> Damit zum Beispiel das 4022 Profil im Drop-Down-Menü verfügbar ist, muss entweder eine 4022 Remote Manager Installation durchgeführt worden sein oder die aktuelle 4024 XAE Installation über eine zuvor vorhandene 4022 XAE Installation installiert worden sein.</p>
<p>Write Bookmarks to File</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p><input type="checkbox"/> (Voreinstellung): Die Lesezeichen werden nur in der .suo-Datei vom Visual Studio gespeichert. Eine bereits erstellte .bookmarks-Datei wird aus dem Projektverzeichnis gelöscht.</p> <p><input checked="" type="checkbox"/> : Die abgelegten Lesezeichen werden aus der Benutzeroptionsdatei vom Visual Studio Projekt (.suo) zusätzlich in eine separate Datei geschrieben. Diese Datei endet mit .bookmarks und liegt im Projektverzeichnis. Damit ist sie dann ebenfalls Bestandteil der bekannten Archivmöglichkeiten.</p> <p>Die globale Standardeinstellung für neue SPS-Projekte, ob Lesezeichen in einer separaten Datei abgelegt werden sollen, finden Sie unter <a href="#">Dialog Optionen - Write Options [► 194]</a>. Der Wert wird bei der Erstellung eines neuen SPS-Projekts einmalig in diese lokale Projekteinstellung übernommen.</p>

## Multuser options

<p>Multiuser nutzen („Use Multiuser“)</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4024</p> <p><input type="checkbox"/> (Voreinstellung): Die Multiuser-Funktionalität des SPS-Projekts ist nicht aktiviert.</p> <p><input checked="" type="checkbox"/> : Die Multiuser-Funktionalität des SPS-Projekts ist aktiviert.</p> <p>Bitte beachten Sie auch die weiterführenden Informationen hierzu in der Multiuser-Dokumentation.</p>
<p>Multiuser URL</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p>Beschreibung folgt.</p>
<p>Update Parent</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p>Beschreibung folgt.</p>

**Solution options**

<p>Sicherer Onlinebetrieb („Secure Online Mode“)</p>	<p><input type="checkbox"/> : (Voreinstellung): Aus Sicherheitsgründen wird der Benutzer beim Aufruf der folgenden Befehle grundsätzlich dazu aufgefordert, die Ausführung nochmals zu bestätigen.</p> <ul style="list-style-type: none"> <li>• Konfiguration aktivieren</li> <li>• Restart TwinCAT System in Config/Run Mode</li> <li>• Reset Kalt</li> <li>• Reset Ursprung</li> </ul> <p><input checked="" type="checkbox"/> : Zusätzlich zu den oben genannten Befehlen, bei denen standardmäßig eine Bestätigungsaufforderung erscheint, werden Sie bei den folgenden Befehlen ebenfalls zu einer Bestätigung aufgefordert.</p> <ul style="list-style-type: none"> <li>• Einloggen</li> <li>• Start</li> <li>• Stopp</li> <li>• Einzelzyklus</li> <li>• Werte schreiben</li> <li>• Werte forcen</li> <li>• Forcen aufheben</li> </ul>
<p>Autoupdate Visu Profile</p>	<p>Mit dieser Option können Sie das automatische Update-Verhalten des Visualisierungsprofils konfigurieren.</p> <p>Wenn Sie ein SPS-Projekt öffnen, in dem ein veraltetes Visualisierungsprofil verwendet wird, erscheint im Meldungsfenster eine entsprechende Warnung („New Version found for Visualization profile“).</p> <p><input type="checkbox"/> (Voreinstellung): Wenn die Option <b>Autoupdate Visu Profile</b> deaktiviert ist, wird die Visualisierungsprofil-Version nicht automatisch geändert. Per Doppelklick auf die Warnung „New Version found for Visualization profile“ können Sie den ProfileUpdate-Dialog öffnen, in welchem Sie die Visualisierungsprofil-Version manuell ändern können.</p> <p><input checked="" type="checkbox"/> : In einem solchen Fall wird die Visualisierungsprofil-Version automatisch auf die neueste Version gesetzt, falls die Option <b>Autoupdate Visu Profile</b> aktiviert ist. Bei einem solchen automatischen Update der Visualisierungsprofil-Version wird im Meldungsfenster eine entsprechende Warnung angezeigt (z. B. „Visualization profile set from ‚TwinCAT 3.1 Build 4020.10‘ to ‚TwinCAT 3.1 Build 4022.0‘“).</p>
<p>Autoupdate Uml Profile</p>	<p>Mit dieser Option können Sie das automatische Update-Verhalten der UML-Compiler-Version konfigurieren.</p> <p>Wenn Sie ein SPS-Projekt öffnen, in dem eine veraltete UML-Compiler-Version verwendet wird, erscheint im Meldungsfenster eine entsprechende Warnung („neue Version für UML gefunden“).</p> <p><input type="checkbox"/> (Voreinstellung): Wenn die Option <b>Autoupdate Uml Profile</b> deaktiviert ist, wird die UML-Compiler-Version nicht automatisch geändert. Per Doppelklick auf die Warnung „neue Version für UML gefunden“ können Sie den ProfileUpdate-Dialog öffnen, in welchem Sie die UML Compiler-Version manuell ändern können.</p> <p><input checked="" type="checkbox"/> : In einem solchen Fall wird die UML-Compiler-Version automatisch auf die neueste Version gesetzt, falls die Option <b>Autoupdate Uml Profile</b> aktiviert ist. Bei einem solchen automatischen Update der UML-Compiler-Version wird im Meldungsfenster eine entsprechende Warnung angezeigt (z. B. „UML set from ‚4.0.2.0‘ to ‚4.0.2.1‘“).</p> <p>Weitere Informationen hierzu finden Sie unter UML Compiler-Version.</p>

Write Line IDs	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p><input type="checkbox"/> (Voreinstellung): Es werden keine separaten Line-IDs erzeugt. Für die Zuordnung der Maschinencode-Anweisungen und das Breakpoint-Handling wird in diesem Fall die Zeilennummer genutzt. Bei Änderungen wie Leerzeichen oder Kommentaren ist daher ein Online-Change erforderlich.</p> <p><input checked="" type="checkbox"/> : Für die POU's des Projekts werden Line-IDs erzeugt und gespeichert (Standardverhalten bis TwinCAT 3.1 Build 4024). Über die Line-IDs können Code-Zeilen zu Maschinencode-Anweisungen zugeordnet werden, was u. a. für das Breakpoint-Handling benötigt wird.</p> <p>Die globale Standardeinstellung für neue SPS-Projekte bezüglich der Write Line IDs finden Sie unter <a href="#">Dialog Optionen - Write Options [► 194]</a>. Der Wert wird bei der Erstellung eines neuen SPS-Projekts einmalig in diese lokale Projekteinstellung übernommen.</p>
----------------	--

**Compatibility**

Convert PLC Project to previous TwinCAT version	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p>In dem sich öffnenden Dialog können Sie eine TwinCAT-Version auswählen, in die das SPS Projekt konvertiert werden soll (TwinCAT 3.1 Build 4022 oder 4024). Nach Bestätigung der Konvertierung mit <b>Convert</b> wird das Projekt geschlossen und kompatibel zu der ausgewählten Version abgespeichert.</p> <p>Beachten Sie, dass die Projektdaten bei der Konvertierung verändert werden und Einstellungen und Eigenschaften späterer Versionen verloren gehen. Die Konvertierung ist daher nicht für einen mehrfachen Wechsel zwischen verschiedenen Versionen geeignet.</p>
---	--

**Write events to event log**

Mit Hilfe der folgenden beiden Optionen können Sie konfigurieren, ob und welche Art von User-Events Sie loggen möchten. Als User-Events gelten dabei:

- Einloggen
- Ausloggen
- Download
- Online-Change
- SPS-Projekt starten
- SPS-Projekt stoppen
- Einzelzyklus
- Reset kalt
- Reset Ursprung
- Haltepunkt setzen
- Haltepunkt entfernen

Die Events werden über die Laufzeit an den Eventlogger gesendet. Zusätzlich werden die Events im Meldungsfenster (Error List) ausgegeben, falls die Eventlogger-Option **Output as Task Item** auf True gestellt ist.

Wenn Sie die im Folgenden beschriebenen Optionen ändern, ist beim nächsten Einloggen ein Online-Change erforderlich. Anschließend sind die Einstellungen aktiv.

<p>Information</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p><input type="checkbox"/> (Voreinstellung): Es werden keine Meldungen geloggt, wenn erfolgreiche User-Events stattfinden.</p> <p><input checked="" type="checkbox"/> : Wenn eins der o. g. User-Events stattfindet und erfolgreich ist, wird eine entsprechende Information über diesen Vorgang mitgeloggt.</p>
<p>Error</p>	<p>Verfügbar ab TwinCAT 3.1 Build 4026</p> <p><input type="checkbox"/> (Voreinstellung): Es werden keine Meldungen geloggt, wenn fehlgeschlagene User-Events stattfinden.</p> <p><input checked="" type="checkbox"/> : Wenn bei einem der o. g. User-Events ein Fehler auftritt, wird eine entsprechende Information über diesen fehlgeschlagenen Vorgang mitgeloggt. Dies ist z. B. der Fall, wenn ein Login aufgrund eines Fehlers in der Laufzeit fehlschlägt.</p>

**General**

<p>(Identische) Objekte mit der gleichen ID mehrfach in einer Solution erlauben</p> <p>(bzw. „Allow (identical) objects with the same ID multiple times in one solution“)</p>	<p><input type="checkbox"/> (Voreinstellung): Die SPS-Objekte (z. B. POUs) werden mit Hilfe ihrer GUID identifiziert, welche in der gesamten Solution jeweils einmalig vergeben ist. Für die üblichen Anwendungsfälle kann und sollte diese Option deaktiviert bleiben.</p> <p><input checked="" type="checkbox"/> : Die aktivierte Option ist ausschließlich für das Szenario relevant, bei dem ein Objekt innerhalb einer Solution mehrfach mit der gleichen ID/GUID existiert. Für diesen Fall muss sich die Identifikationsart der Objekte ändern, sodass die GUIDs der SPS-Objekte mit denen des SPS-Projekts verknüpft werden (mittels XOR). Dadurch werden Änderungen der GUIDs der SPS-Objekte vermieden, wenn diese mehrmals in verschiedenen Projekten innerhalb einer Solution verwendet werden.</p> <p><b>Informationen:</b></p> <ul style="list-style-type: none"> <li>• Wenn Sie den Zustand der Option ändern, muss das Projekt neugeladen werden. Zudem ist beim nächsten Einloggen ein <b>Download</b> erforderlich.</li> <li>• Wenn die Option <b>aktiviert</b> ist, ändert sich die Identifikationsart aller Objekte dieses SPS-Projekts.</li> <li>• Wenn die Option <b>aktiviert</b> ist, können die folgenden SPS-Funktionalitäten <b>nicht oder nicht vollumfänglich genutzt</b> werden:             <ul style="list-style-type: none"> <li>◦ Visualisierung</li> <li>◦ UML SC</li> <li>◦ Security-Management</li> </ul> </li> </ul> <p>Diese Option wurde zu TwinCAT 3.1 Build 4026.x umbenannt. Die vorherige Benennung lautete: „ID-Änderungen in TwinCAT-Dateien minimieren“ bzw. „Minimize ID changes in TwinCAT files“. Das Verhalten der Option bleibt unverändert.</p>
---	--

## **Trademark statements**

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

## **Third-party trademark statements**

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

Mehr Informationen:  
**[www.beckhoff.com/te1000](http://www.beckhoff.com/te1000)**

Beckhoff Automation GmbH & Co. KG  
Hülshorstweg 20  
33415 Verl  
Deutschland  
Telefon: +49 5246 9630  
[info@beckhoff.com](mailto:info@beckhoff.com)  
[www.beckhoff.com](http://www.beckhoff.com)

