

# SoMachine Basic

## Betriebshandbuch

12/2017

EIO0000001356.10

[www.schneider-electric.com](http://www.schneider-electric.com)

**Schneider**  
 **Electric**

---

Die Informationen in der vorliegenden Dokumentation enthalten allgemeine Beschreibungen und/oder technische Leistungsmerkmale der hier erwähnten Produkte. Diese Dokumentation dient keinesfalls als Ersatz für die Ermittlung der Eignung oder Verlässlichkeit dieser Produkte für bestimmte Verwendungsbereiche des Benutzers und darf nicht zu diesem Zweck verwendet werden. Jeder Benutzer oder Integrator ist verpflichtet, angemessene und vollständige Risikoanalysen, Bewertungen und Tests der Produkte im Hinblick auf deren jeweils spezifischen Verwendungszweck vorzunehmen. Weder Schneider Electric noch deren Tochtergesellschaften oder verbundene Unternehmen sind für einen Missbrauch der Informationen in der vorliegenden Dokumentation verantwortlich oder können diesbezüglich haftbar gemacht werden. Verbesserungs- und Änderungsvorschläge sowie Hinweise auf angetroffene Fehler werden jederzeit gern entgegengenommen.

Sie erklären, dass Sie ohne schriftliche Genehmigung von Schneider Electric dieses Dokument weder ganz noch teilweise auf beliebigen Medien reproduzieren werden, ausgenommen zur Verwendung für persönliche nichtkommerzielle Zwecke. Darüber hinaus erklären Sie, dass Sie keine Hypertext-Links zu diesem Dokument oder seinem Inhalt einrichten werden. Schneider Electric gewährt keine Berechtigung oder Lizenz für die persönliche und nichtkommerzielle Verwendung dieses Dokument oder seines Inhalts, ausgenommen die nichtexklusive Lizenz zur Nutzung als Referenz. Das Handbuch wird hierfür „wie besehen“ bereitgestellt, die Nutzung erfolgt auf eigene Gefahr. Alle weiteren Rechte sind vorbehalten.

Bei der Montage und Verwendung dieses Produkts sind alle zutreffenden staatlichen, landesspezifischen, regionalen und lokalen Sicherheitsbestimmungen zu beachten. Aus Sicherheitsgründen und um die Übereinstimmung mit dokumentierten Systemdaten besser zu gewährleisten, sollten Reparaturen an Komponenten nur vom Hersteller vorgenommen werden.

Beim Einsatz von Geräten für Anwendungen mit technischen Sicherheitsanforderungen sind die relevanten Anweisungen zu beachten.

Die Verwendung anderer Software als der Schneider Electric-eigenen bzw. einer von Schneider Electric genehmigten Software in Verbindung mit den Hardwareprodukten von Schneider Electric kann Körperverletzung, Schäden oder einen fehlerhaften Betrieb zur Folge haben.

Die Nichtbeachtung dieser Informationen kann Verletzungen oder Materialschäden zur Folge haben!

© 2017 Schneider Electric. Alle Rechte vorbehalten.



|                  |   |           |
|------------------|---|-----------|
|                  | <b>Sicherheitshinweise</b> .....                        | <b>9</b>  |
|                  | <b>Über dieses Buch</b> .....                           | <b>11</b> |
| <b>Teil I</b>    | <b>Erste Schritte mit SoMachine Basic</b> .....         | <b>19</b> |
| <b>Kapitel 1</b> | <b>Einführung in SoMachine Basic</b> .....              | <b>21</b> |
| 1.1              | Systemanforderungen und unterstützte Geräte .....       | <b>22</b> |
|                  | Systemanforderungen .....                               | <b>23</b> |
|                  | Unterstützte Geräte .....                               | <b>24</b> |
|                  | Unterstützte Programmiersprachen .....                  | <b>26</b> |
| 1.2              | Grundlagen der SoMachine Basic-Benutzeroberfläche ..... | <b>27</b> |
|                  | Erstellen von Projekten mit SoMachine Basic .....       | <b>28</b> |
|                  | Entwickeln von Programmen mit SoMachine Basic .....     | <b>29</b> |
|                  | Navigation in SoMachine Basic .....                     | <b>30</b> |
|                  | Betriebsmodi .....                                      | <b>31</b> |
| <b>Kapitel 2</b> | <b>Arbeiten mit SoMachine Basic</b> .....               | <b>33</b> |
| 2.1              | Fenster „Startseite“ .....                              | <b>34</b> |
|                  | Einführung in die Startseite .....                      | <b>35</b> |
|                  | Registrierung der SoMachine Basic-Software .....        | <b>36</b> |
|                  | Fenster „Projekte“ .....                                | <b>37</b> |
|                  | Fenster „Verbinden“ .....                               | <b>41</b> |
|                  | Direktes Herunterladen einer Anwendung .....            | <b>46</b> |
|                  | Speicherverwaltung .....                                | <b>47</b> |
|                  | Fenster „Projektvorlagen“ .....                         | <b>48</b> |
|                  | Hilfefenster .....                                      | <b>49</b> |
| <b>Teil II</b>   | <b>Entwicklung von SoMachine Basic-Anwendungen</b> ..   | <b>51</b> |
| <b>Kapitel 3</b> | <b>Fenster von SoMachine Basic</b> .....                | <b>53</b> |
| 3.1              | Übersicht über das SoMachine Basic-Fenster .....        | <b>54</b> |
|                  | Schaltflächen der Symbolleiste .....                    | <b>55</b> |
|                  | Statusbereich .....                                     | <b>57</b> |
|                  | Systemeinstellungen .....                               | <b>60</b> |
|                  | Drucken von Berichten .....                             | <b>62</b> |
| <b>Kapitel 4</b> | <b>Eigenschaften</b> .....                              | <b>65</b> |
| 4.1              | Übersicht über das Fenster „Eigenschaften“ .....        | <b>66</b> |
|                  | Fenster „Eigenschaften“ .....                           | <b>67</b> |
|                  | Projekteigenschaften .....                              | <b>68</b> |

|                  |   |           |
|------------------|---|-----------|
| <b>Kapitel 5</b> | <b>Konfiguration</b> .....  | <b>71</b> |
| 5.1              | Übersicht über das Fenster „Konfiguration“ .....                  | 72        |
|                  | Übersicht des Konfigurationsfensters .....                        | 73        |
|                  | Erstellen einer Konfiguration .....                               | 74        |
| <b>Kapitel 6</b> | <b>Programmierung</b> .....                                       | <b>75</b> |
| 6.1              | Übersicht über den Arbeitsbereich der Programmierung .....        | 76        |
|                  | Überblick des Arbeitsbereichs ‚Programmierung‘ .....              | 76        |
| 6.2              | Sonderfunktionen .....  | 78        |
|                  | Objekte .....   | 79        |
|                  | Symbolische Adressierung .....                                    | 80        |
|                  | Speicherzuweisung .....   | 83        |
|                  | Umkehrbarkeit Kontaktplan/Anweisungsliste .....                   | 84        |
| 6.3              | Konfigurieren des Programmverhaltens und der Tasks .....          | 90        |
|                  | Verhalten der Anwendung .....                                     | 91        |
|                  | Tasks und Abfragemodi .....                                       | 95        |
| 6.4              | Verwalten von POUs .....  | 98        |
|                  | POUs .....  | 99        |
|                  | Verwalten von POUs mit Tasks .....                                | 100       |
|                  | Verwalten von Programmbausteinen .....                            | 103       |
|                  | Verwalten von Grafset (SFC)-POUs .....                            | 107       |
|                  | Freie POUs .....  | 109       |
| 6.5              | Benutzerdefinierte Funktionen .....                               | 113       |
|                  | Erstellen einer benutzerdefinierten Funktion .....                | 114       |
|                  | Definieren einer benutzerdefinierten Funktion .....               | 115       |
|                  | Verwalten von benutzerdefinierten Funktionen .....                | 120       |
| 6.6              | Benutzerdefinierte Funktionsbausteine .....                       | 123       |
|                  | Erstellen eines benutzerdefinierten Funktionsbausteins .....      | 124       |
|                  | Definieren eines benutzerdefinierten Funktionsbausteins .....     | 125       |
|                  | Verwalten von benutzerdefinierten Funktionsbausteinen .....       | 128       |
| 6.7              | Master-Task .....   | 131       |
|                  | Beschreibung der Master-Task .....                                | 132       |
|                  | Konfigurieren des Master-Tasks .....                              | 133       |
| 6.8              | Zeichenketten .....   | 135       |
|                  | Konfigurieren von Zeichenketten in Konstanten im Wortformat ..... | 137       |
|                  | Zuweisen von Zeichenfolgen in Speicherwörtern .....               | 138       |
|                  | Verwalten von Zeichenfolgen .....                                 | 139       |

|      |  |     |
|------|--|-----|
| 6.9  | Periodische Task                                     | 142 |
|      | Erstellen periodischer Tasks                         | 143 |
|      | Konfigurieren der Abfragedauer des periodischen Task | 145 |
| 6.10 | Ereignis-Task  | 146 |
|      | Übersicht über die Ereignistasks                     | 147 |
|      | Ereignisquellen                                      | 148 |
|      | Ereignisprioritäten                                  | 149 |
|      | Betrachten von Ereignistasks                         | 150 |
| 6.11 | Verwenden von Tools                                  | 153 |
|      | Meldungen  | 154 |
|      | Animationstabellen                                   | 157 |
|      | Speicherobjekte                                      | 163 |
|      | Systemobjekte  | 168 |
|      | E/A-Objekte  | 169 |
|      | Netzwerkobjekte                                      | 170 |
|      | Software-Objekte                                     | 171 |
|      | PTO-Objekte  | 172 |
|      | Antriebsobjekte                                      | 173 |
|      | Kommunikationsobjekte                                | 174 |
|      | Suchen und Ersetzen                                  | 175 |
|      | Querverweis  | 177 |
|      | Symbolliste  | 178 |
|      | Ansicht der Speichernutzung                          | 183 |
| 6.12 | Programmieren in Kontaktplan                         | 185 |
|      | Einführung in Kontaktplan                            | 186 |
|      | Programmiergrundlagen für Kontaktplan                | 188 |
|      | Farbkodierungen von Programmbausteinen               | 190 |
|      | Grafikelemente in Kontaktplan                        | 192 |
|      | Vergleichsbausteine                                  | 199 |
|      | Operationsbausteine                                  | 200 |
|      | Hinzufügen von Kommentaren                           | 204 |
|      | Best Practices für die Programmierung                | 205 |
| 6.13 | Programmieren in Anweisungsliste                     | 209 |
|      | Übersicht über Programme in Anweisungsliste          | 210 |
|      | Funktionsweise der Anweisungsliste                   | 213 |
|      | AWL-Anweisungen                                      | 214 |
|      | Verwenden von Klammern                               | 218 |

|                  |   |            |
|------------------|---|------------|
| 6.14             | Programmierung mit Grafcet (Liste) . . . . .  | 221        |
|                  | Beschreibung der Programmierung mit Grafcet (Liste) . . . . .                             | 222        |
|                  | Grafcet-Programmstruktur (Liste) . . . . .  | 224        |
|                  | Verwendung von Grafcet-Anweisungen (Liste) in einem SoMachine<br>Basic-Programm . . . . . | 228        |
| 6.15             | Grafcet (SFC)Programmierung . . . . .   | 230        |
|                  | Einführung in die Grafcet (SFC)-Programmierung . . . . .                                  | 231        |
|                  | Verwenden des Grafischen Grafcet (SFC)-Editors . . . . .                                  | 234        |
|                  | Verzweigung . . . . .   | 239        |
|                  | Beste Verfahren beim Programmieren (Best Practices) . . . . .                             | 243        |
| 6.16             | Fehlerbehebung im Online-Modus . . . . .  | 245        |
|                  | Trace-Fenster . . . . .   | 246        |
|                  | Ändern von Werten . . . . .   | 249        |
|                  | Forcieren von Werten . . . . .  | 250        |
|                  | Änderungen im Online-Modus . . . . .  | 251        |
| <b>Kapitel 7</b> | <b>Inbetriebnahme . . . . .</b>   | <b>259</b> |
| 7.1              | Übersicht über das Fenster „Inbetriebnahme“ . . . . .                                     | 260        |
|                  | Überblick des Fensters ‚Inbetriebnahme‘ . . . . .   | 260        |
| 7.2              | <b>Aufbauen einer Verbindung zu einer Steuerung . . . . .</b>                             | <b>261</b> |
|                  | Verbinden mit einer Logiksteuerung . . . . .  | 262        |
|                  | Herunter- und Hochladen von Anwendungen . . . . .   | 268        |
| 7.3              | <b>Steuerungs-Update . . . . .</b>  | <b>272</b> |
|                  | Firmware-Updates der Steuerung . . . . .  | 272        |
| 7.4              | Speicherverwaltung . . . . .  | 274        |
|                  | Verwaltung des Steuerungsspeichers . . . . .  | 274        |
| 7.5              | <b>SPS-Infos . . . . .</b>  | <b>280</b> |
|                  | Informationen zur Steuerung . . . . .   | 280        |
| 7.6              | <b>RTC-Verwaltung . . . . .</b>   | <b>283</b> |
|                  | Verwalten der RTC . . . . .   | 283        |
| <b>Kapitel 8</b> | <b>Simulator . . . . .</b>  | <b>285</b> |
|                  | Überblick über den SoMachine Basic-Simulator . . . . .                                    | 286        |
|                  | E/A-Manager-Fenster des SoMachine Basic-Simulators . . . . .                              | 288        |
|                  | SoMachine Basic-Simulator – Fenster „ <b>Zeitverwaltung</b> “ . . . . .                   | 290        |
|                  | Ändern von Werten mit dem SoMachine Basic-Simulator . . . . .                             | 293        |
|                  | Verwendung des SoMachine Basic-Simulators . . . . .                                       | 299        |
|                  | Starten der Simulation in Vijeo-Designer . . . . .  | 300        |

---

|                  |  |            |
|------------------|--|------------|
| <b>Kapitel 9</b> | <b>Speichern von Projekten und Schließen von SoMachine Basic</b> | <b>301</b> |
|                  | Speichern eines Projekts   | 302        |
|                  | Speichern der Projektdatei als Vorlage                           | 303        |
|                  | Schließen von SoMachine Basic                                    | 304        |
| <b>Anhang</b>    |  | <b>305</b> |
| <b>Anhang A</b>  | <b>Konvertieren von Twido-Projekten zu SoMachine Basic</b>       | <b>307</b> |
|                  | Konvertieren von Twido-Projekten zu SoMachine Basic              | 307        |
| <b>Anhang B</b>  | <b>SoMachine Basic-Tastaturkombinationen</b>                     | <b>319</b> |
|                  | SoMachine Basic Tastaturkombinationen                            | 319        |
| <b>Glossar</b>   |  | <b>327</b> |
| <b>Index</b>     |  | <b>331</b> |





## Wichtige Informationen

### HINWEISE

Lesen Sie sich diese Anweisungen sorgfältig durch und machen Sie sich vor Installation, Betrieb, Bedienung und Wartung mit dem Gerät vertraut. Die nachstehend aufgeführten Warnhinweise sind in der gesamten Dokumentation sowie auf dem Gerät selbst zu finden und weisen auf potenzielle Risiken und Gefahren oder bestimmte Informationen hin, die eine Vorgehensweise verdeutlichen oder vereinfachen.



Wird dieses Symbol zusätzlich zu einem Sicherheitshinweis des Typs „Gefahr“ oder „Warnung“ angezeigt, bedeutet das, dass die Gefahr eines elektrischen Schlags besteht und die Nichtbeachtung der Anweisungen unweigerlich Verletzung zur Folge hat.



Dies ist ein allgemeines Warnsymbol. Es macht Sie auf mögliche Verletzungsgefahren aufmerksam. Beachten Sie alle unter diesem Symbol aufgeführten Hinweise, um Verletzungen oder Unfälle mit Todesfälle zu vermeiden.

## **GEFAHR**

**GEFAHR** macht auf eine gefährliche Situation aufmerksam, die, wenn sie nicht vermieden wird, Tod oder schwere Verletzungen **zur Folge hat**.

## **WARNUNG**

**WARNUNG** macht auf eine gefährliche Situation aufmerksam, die, wenn sie nicht vermieden wird, Tod oder schwere Verletzungen **zur Folge haben kann**.

## **VORSICHT**

**VORSICHT** macht auf eine gefährliche Situation aufmerksam, die, wenn sie nicht vermieden wird, leichte Verletzungen **zur Folge haben kann**.

## **HINWEIS**

**HINWEIS** gibt Auskunft über Vorgehensweisen, bei denen keine Verletzungen drohen.

---

## **BITTE BEACHTEN**

Elektrische Geräte dürfen nur von Fachpersonal installiert, betrieben, bedient und gewartet werden. Schneider Electric haftet nicht für Schäden, die durch die Verwendung dieses Materials entstehen.

Als qualifiziertes Fachpersonal gelten Mitarbeiter, die über Fähigkeiten und Kenntnisse hinsichtlich der Konstruktion und des Betriebs elektrischer Geräte und deren Installation verfügen und eine Schulung zur Erkennung und Vermeidung möglicher Gefahren absolviert haben.

---

# Über dieses Buch

---



## Auf einen Blick

### Ziel dieses Dokuments

In diesem Handbuch wird die Verwendung der Software SoMachine Basic zur Konfiguration, Programmierung und Inbetriebnahme von Anwendungen für unterstützte Steuerungen beschrieben.

### Gültigkeitsbereich

Die Informationen in diesem Handbuch beziehen sich **ausschließlich** auf SoMachine Basic-Produkte.

Dieses Dokument wurde für die SoMachine Basic-Version V1.6 aktualisiert.

Die technischen Merkmale der hier beschriebenen Geräte sind auch online abrufbar. So greifen Sie auf diese Informationen online zu:

| Schritt | Aktion  |
|---------|---|
| 1       | Gehen Sie zur Homepage von Schneider Electric <a href="http://www.schneider-electric.com">www.schneider-electric.com</a> .  |
| 2       | Geben Sie im Feld <b>Search</b> die Referenz eines Produkts oder den Namen einer Produktreihe ein. <ul style="list-style-type: none"><li>Die Referenz bzw. der Name der Produktreihe darf keine Leerstellen enthalten.</li><li>Wenn Sie nach Informationen zu verschiedenen vergleichbaren Modulen suchen, können Sie Sternchen (*) verwenden.</li></ul>  |
| 3       | Wenn Sie eine Referenz eingegeben haben, gehen Sie zu den Suchergebnissen für technische Produktdatenblätter ( <b>Product Datasheets</b> ) und klicken Sie auf die Referenz, über die Sie mehr erfahren möchten.<br>Wenn Sie den Namen einer Produktreihe eingegeben haben, gehen Sie zu den Suchergebnissen <b>Product Ranges</b> und klicken Sie auf die Reihe, über die Sie mehr erfahren möchten. |
| 4       | Wenn mehrere Referenzen in den Suchergebnissen unter <b>Products</b> angezeigt werden, klicken Sie auf die gewünschte Referenz.   |
| 5       | Je nach der Größe der Anzeige müssen Sie ggf. durch die technischen Daten scrollen, um sie vollständig einzusehen.  |
| 6       | Um ein Datenblatt als PDF-Datei zu speichern oder zu drucken, klicken Sie auf <b>Download XXX product datasheet</b> .   |

Die in diesem Handbuch vorgestellten Merkmale sollten denen entsprechen, die online angezeigt werden. Im Rahmen unserer Bemühungen um eine ständige Verbesserung werden Inhalte im Laufe der Zeit möglicherweise überarbeitet, um deren Verständlichkeit und Genauigkeit zu verbessern. Sollten Sie einen Unterschied zwischen den Informationen im Handbuch und denen online feststellen, nutzen Sie die Online-Informationen als Referenz.

## Weiterführende Dokumentation

| Titel der Dokumentation   | Referenz-Nummer  |
|---|--|
| SoMachine Basic Allgemeine Funktionen – Bibliothekshandbuch               | <a href="#">EIO0000001474 (ENG)</a><br><a href="#">EIO0000001475 (FRE)</a><br><a href="#">EIO0000001476 (GER)</a><br><a href="#">EIO0000001477 (SPA)</a><br><a href="#">EIO0000001478 (ITA)</a><br><a href="#">EIO0000001479 (CHS)</a><br><a href="#">EIO0000001480 (POR)</a><br><a href="#">EIO0000001481 (TUR)</a> |
| Modicon M221 Logic Controller Erweiterte Funktionen – Bibliothekshandbuch | <a href="#">EIO0000002007 (ENG)</a><br><a href="#">EIO0000002008 (FRE)</a><br><a href="#">EIO0000002009 (GER)</a><br><a href="#">EIO0000002010 (SPA)</a><br><a href="#">EIO0000002011 (ITA)</a><br><a href="#">EIO0000002012 (CHS)</a><br><a href="#">EIO0000002013 (POR)</a><br><a href="#">EIO0000002014 (TUR)</a> |
| Modicon M221 Logic Controller – Programmierhandbuch                       | <a href="#">EIO0000001360 (ENG)</a><br><a href="#">EIO0000001361 (FRE)</a><br><a href="#">EIO0000001362 (GER)</a><br><a href="#">EIO0000001363 (SPA)</a><br><a href="#">EIO0000001364 (ITA)</a><br><a href="#">EIO0000001365 (CHS)</a><br><a href="#">EIO0000001368 (POR)</a><br><a href="#">EIO0000001369 (TUR)</a> |
| Modicon M221 Logic Controller – Hardwarehandbuch                          | <a href="#">EIO0000001384 (ENG)</a><br><a href="#">EIO0000001385 (FRE)</a><br><a href="#">EIO0000001386 (GER)</a><br><a href="#">EIO0000001387 (SPA)</a><br><a href="#">EIO0000001388 (ITA)</a><br><a href="#">EIO0000001389 (CHS)</a><br><a href="#">EIO0000001370 (POR)</a><br><a href="#">EIO0000001371 (TUR)</a> |
| Modicon TMC2-Steckmodule – Programmierhandbuch                            | <a href="#">EIO0000001782 (ENG)</a><br><a href="#">EIO0000001783 (FRE)</a><br><a href="#">EIO0000001784 (GER)</a><br><a href="#">EIO0000001785 (SPA)</a><br><a href="#">EIO0000001786 (ITA)</a><br><a href="#">EIO0000001787 (CHS)</a><br><a href="#">EIO0000001788 (POR)</a><br><a href="#">EIO0000001789 (TUR)</a> |

| Titel der Dokumentation   | Referenz-Nummer  |
|---|--|
| Modicon TMC2 Steckmodule – Hardwarehandbuch                             | <a href="#"><u>EIO0000001768 (ENG)</u></a><br><a href="#"><u>EIO0000001769 (FRE)</u></a><br><a href="#"><u>EIO0000001770 (GER)</u></a><br><a href="#"><u>EIO0000001771 (SPA)</u></a><br><a href="#"><u>EIO0000001772 (ITA)</u></a><br><a href="#"><u>EIO0000001773 (CHS)</u></a><br><a href="#"><u>EIO0000001774 (POR)</u></a><br><a href="#"><u>EIO0000001775 (TUR)</u></a> |
| Modicon TM3 Konfiguration von Erweiterungsmodulen – Programmierhandbuch | <a href="#"><u>EIO0000001396 (ENG)</u></a><br><a href="#"><u>EIO0000001397 (FRE)</u></a><br><a href="#"><u>EIO0000001398 (GER)</u></a><br><a href="#"><u>EIO0000001399 (SPA)</u></a><br><a href="#"><u>EIO0000001400 (ITA)</u></a><br><a href="#"><u>EIO0000001401 (CHS)</u></a><br><a href="#"><u>EIO0000001374 (POR)</u></a><br><a href="#"><u>EIO0000001375 (TUR)</u></a> |
| Modicon TM3 Digitale E/A-Module – Hardwarehandbuch                      | <a href="#"><u>EIO0000001408 (ENG)</u></a><br><a href="#"><u>EIO0000001409 (FRE)</u></a><br><a href="#"><u>EIO0000001410 (GER)</u></a><br><a href="#"><u>EIO0000001411 (SPA)</u></a><br><a href="#"><u>EIO0000001412 (ITA)</u></a><br><a href="#"><u>EIO0000001413 (CHS)</u></a><br><a href="#"><u>EIO0000001376 (POR)</u></a><br><a href="#"><u>EIO0000001377 (TUR)</u></a> |
| Modicon TM3 Analoge E/A-Module – Hardwarehandbuch                       | <a href="#"><u>EIO0000001414 (ENG)</u></a><br><a href="#"><u>EIO0000001415 (FRE)</u></a><br><a href="#"><u>EIO0000001416 (GER)</u></a><br><a href="#"><u>EIO0000001417 (SPA)</u></a><br><a href="#"><u>EIO0000001418 (ITA)</u></a><br><a href="#"><u>EIO0000001419 (CHS)</u></a><br><a href="#"><u>EIO0000001378 (POR)</u></a><br><a href="#"><u>EIO0000001379 (TUR)</u></a> |
| Modicon TM3 Expertenmodule – Hardwarehandbuch                           | <a href="#"><u>EIO0000001420 (ENG)</u></a><br><a href="#"><u>EIO0000001421 (FRE)</u></a><br><a href="#"><u>EIO0000001422 (GER)</u></a><br><a href="#"><u>EIO0000001423 (SPA)</u></a><br><a href="#"><u>EIO0000001424 (ITA)</u></a><br><a href="#"><u>EIO0000001425 (CHS)</u></a><br><a href="#"><u>EIO0000001380 (POR)</u></a><br><a href="#"><u>EIO0000001381 (TUR)</u></a> |

| Titel der Dokumentation   | Referenz-Nummer  |
|---|--|
| Modicon TM3 Sicherheitsmodule – Hardwarehandbuch                        | <a href="#"><i>EIO0000001831 (ENG)</i></a><br><a href="#"><i>EIO0000001832 (FRE)</i></a><br><a href="#"><i>EIO0000001833 (GER)</i></a><br><a href="#"><i>EIO0000001834 (SPA)</i></a><br><a href="#"><i>EIO0000001835 (ITA)</i></a><br><a href="#"><i>EIO0000001836 (CHS)</i></a><br><a href="#"><i>EIO0000001837 (POR)</i></a><br><a href="#"><i>EIO0000001838 (TUR)</i></a> |
| Modicon TM3 Sender- und Empfängermodule – Hardwarehandbuch              | <a href="#"><i>EIO0000001426 (ENG)</i></a><br><a href="#"><i>EIO0000001427 (FRE)</i></a><br><a href="#"><i>EIO0000001428 (GER)</i></a><br><a href="#"><i>EIO0000001429 (SPA)</i></a><br><a href="#"><i>EIO0000001430 (ITA)</i></a><br><a href="#"><i>EIO0000001431 (CHS)</i></a><br><a href="#"><i>EIO0000001382 (POR)</i></a><br><a href="#"><i>EIO0000001383 (TUR)</i></a> |
| Modicon TM2 Konfiguration von Erweiterungsmodulen – Programmierhandbuch | <a href="#"><i>EIO0000000396 (ENG)</i></a><br><a href="#"><i>EIO0000000397 (FRE)</i></a><br><a href="#"><i>EIO0000000398 (GER)</i></a><br><a href="#"><i>EIO0000000399 (SPA)</i></a><br><a href="#"><i>EIO0000000400 (ITA)</i></a><br><a href="#"><i>EIO0000000401 (CHS)</i></a>   |
| Modicon TM2 Digitale E/A-Module – Hardwarehandbuch                      | <a href="#"><i>EIO0000000028 (ENG)</i></a><br><a href="#"><i>EIO0000000029 (FRE)</i></a><br><a href="#"><i>EIO0000000030 (GER)</i></a><br><a href="#"><i>EIO0000000031 (SPA)</i></a><br><a href="#"><i>EIO0000000032 (ITA)</i></a><br><a href="#"><i>EIO0000000033 (CHS)</i></a>   |
| Modicon TM2 Analoge E/A-Module – Hardwarehandbuch                       | <a href="#"><i>EIO0000000034 (ENG)</i></a><br><a href="#"><i>EIO0000000035 (FRE)</i></a><br><a href="#"><i>EIO0000000036 (GER)</i></a><br><a href="#"><i>EIO0000000037 (SPA)</i></a><br><a href="#"><i>EIO0000000038 (ITA)</i></a><br><a href="#"><i>EIO0000000039 (CHS)</i></a>   |
| SR2MOD02 and SR2MOD03 Wireless Modem - User Guide                       | <a href="#"><i>EIO0000001575 (ENG)</i></a>   |

Diese technischen Veröffentlichungen sowie andere technische Informationen stehen auf unserer Website <http://www.schneider-electric.com/en/download> zum Download bereit.

## **WARNUNG**

### **STEUERUNGS AUSFALL**

- Bei der Konzeption von Steuerungsstrategien müssen mögliche Störungen auf den Steuerpfaden berücksichtigt werden, und bei bestimmten kritischen Steuerungsfunktionen ist dafür zu sorgen, dass während und nach einem Pfadfehler ein sicherer Zustand erreicht wird. Beispiele kritischer Steuerungsfunktionen sind die Notabschaltung (Not-Aus) und der Nachlauf-Stopp, Stromausfall und Neustart.
- Für kritische Steuerungsfunktionen müssen separate oder redundante Steuerpfade bereitgestellt werden.
- Systemsteuerungspfade können Kommunikationsverbindungen umfassen. Dabei müssen die Auswirkungen unerwarteter Sendeverzögerungen und Verbindungsstörungen berücksichtigt werden.
- Sämtliche Unfallverhütungsvorschriften und lokale Sicherheitsrichtlinien sind zu beachten.<sup>1</sup>
- Jede Implementierung des Geräts muss individuell und sorgfältig auf einen einwandfreien Betrieb geprüft werden, bevor das Gerät an Ort und Stelle in Betrieb gesetzt wird.

**Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.**

<sup>1</sup> Weitere Informationen finden Sie in den aktuellen Versionen von NEMA ICS 1.1 „Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control“ sowie von NEMA ICS 7.1, „Safety Standards for Construction and Guide for Selection, Installation, and Operation of Adjustable-Speed Drive Systems“ oder den entsprechenden, vor Ort geltenden Vorschriften.

## **WARNUNG**

### **UNBEABSICHTIGTER GERÄTEBETRIEB**

- Verwenden Sie mit diesem Gerät nur von Schneider Electric genehmigte Software.
- Aktualisieren Sie Ihr Anwendungsprogramm jedes Mal, wenn Sie die physische Hardwarekonfiguration ändern.

**Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.**

## Terminologie gemäß den geltenden Standards

Die technischen Begriffe, Terminologien, Symbole und zugehörigen Beschreibungen, die in diesem Handbuch oder auf dem Produkt selbst verwendet werden, werden im Allgemeinen von den Begriffen oder Definitionen internationaler Standards abgeleitet.

Im Bereich der funktionalen Sicherheitssysteme, Antriebe und allgemeinen Automatisierungssysteme betrifft das unter anderem Begriffe wie *Sicherheit*, *Sicherheitsfunktion*, *Sicherer Zustand*, *Fehler*, *Fehlerreset/Zurücksetzen bei Fehler*, *Ausfall*, *Störung*, *Warnung/Warntmeldung*, *Fehlermeldung*, *gefährlich/gefahrbringend* usw.

Nachstehend einige der geltenden Standards:

| Norm                           | Beschreibung  |
|--------------------------------|---|
| EN 61131-2:2007                | Speicherprogrammierbare Steuerungen, Teil 2: Betriebsmittelanforderungen und Prüfungen.   |
| ISO 13849-1:2008               | Sicherheit von Maschinen: Sicherheitsbezogene Teile von Steuerungen. Allgemeine Gestaltungsleitsätze  |
| EN 61496-1:2013                | Sicherheit von Maschinen: Berührungslos wirkende Schutzeinrichtungen. Teil 1: Allgemeine Anforderungen und Prüfungen  |
| ISO 12100:2010                 | Sicherheit von Maschinen – Allgemeine Gestaltungsleitsätze – Risikobeurteilung und Risikominderung  |
| EN 60204-1:2006                | Sicherheit von Maschinen – Elektrische Ausrüstungen von Maschinen – Teil 1: Allgemeine Anforderungen  |
| EN 1088:2008<br>ISO 14119:2013 | Sicherheit von Maschinen – Verriegelungseinrichtungen in Verbindung mit trennenden Schutzeinrichtungen – Leitsätze für Gestaltung und Auswahl   |
| ISO 13850:2006                 | Sicherheit von Maschinen – Not-Halt – Gestaltungsleitsätze  |
| EN/IEC 62061:2005              | Sicherheit von Maschinen – Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbar elektronischer Steuerungssysteme  |
| IEC 61508-1:2010               | Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme: Allgemeine Anforderungen   |
| IEC 61508-2:2010               | Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme: Anforderungen an sicherheitsbezogene elektrische/elektronische/programmierbare elektronische Systeme |
| IEC 61508-3:2010               | Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme: Anforderungen an Software  |
| IEC 61784-3:2008               | Industrielle Kommunikationsnetze – Profile – Teil 3: Funktional sichere Übertragung bei Feldbussen  |
| 2006/42/EC                     | Maschinenrichtlinie   |
| 2014/30/EU                     | EMV-Richtlinie (Elektromagnetische Verträglichkeit)   |
| 2014/35/EU                     | Niederspannungsrichtlinie   |

---

Darüber hinaus wurden einige der in diesem Dokument verwendeten Begriffe unter Umständen auch anderen Normen entnommen, u. a.:

| Norm                  | Beschreibung  |
|-----------------------|---|
| Normenreihe IEC 60034 | Rotierende elektrische Geräte   |
| Normenreihe IEC 61800 | Elektrische Leistungsantriebssysteme mit einstellbarer Drehzahl               |
| Normenreihe IEC 61158 | Industrielle Kommunikationsnetze – Feldbus für industrielle Steuerungssysteme |

Bei einer Verwendung des Begriffs *Betriebsumgebung/Betriebsbereich* in Verbindung mit der Beschreibung bestimmter Gefahren und Risiken entspricht der Begriff der Definition von *Gefahrenbereich* oder *Gefahrenzone* in der *Maschinenrichtlinie (2006/42/EC)* der Norm *ISO 12100:2010*.

**HINWEIS:** Die vorherig erwähnten Standards können auf die spezifischen Produkte in der vorliegenden Dokumentation zutreffen oder nicht. Für weitere Informationen hinsichtlich individueller Standards, die auf hier beschriebene Produkte zutreffen, siehe die Eigenschaftstabellen der hier erwähnten Produkte.



---

# Teil I

## Erste Schritte mit SoMachine Basic

---

### Inhalt dieses Teils

Dieser Teil enthält die folgenden Kapitel:

| Kapitel | Kapitelname                   | Seite |
|---------|-------------------------------|-------|
| 1       | Einführung in SoMachine Basic | 21    |
| 2       | Arbeiten mit SoMachine Basic  | 33    |



---

# Kapitel 1

## Einführung in SoMachine Basic

---

### Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Abschnitte:

| Abschnitt | Thema   | Seite |
|-----------|---|-------|
| 1.1       | Systemanforderungen und unterstützte Geräte       | 22    |
| 1.2       | Grundlagen der SoMachine Basic-Benutzeroberfläche | 27    |

# Abschnitt 1.1

## Systemanforderungen und unterstützte Geräte

---

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                            | Seite |
|----------------------------------|-------|
| Systemanforderungen              | 23    |
| Unterstützte Geräte              | 24    |
| Unterstützte Programmiersprachen | 26    |

## Systemanforderungen

### Überblick

Für den PC, auf dem die Software SoMachine Basic installiert ist, gelten folgende systemspezifischen Mindestanforderungen:

- Intel Core 2 Duo Prozessor oder höher
- 1 GB RAM
- Anzeigeauflösung 1280 x 768 Pixel oder höher
- Eines der folgenden Betriebssysteme in 32- oder 64-Bit-Version:
  - Microsoft Windows 7
  - Microsoft Windows 8
  - Microsoft Windows 8.1
  - Microsoft Windows 10

## Unterstützte Geräte

### M221-Steuerungen

Detaillierte Informationen zur Konfiguration der M221-Steuerung finden Sie in folgenden Programmier- und Hardwarehandbüchern:

| Typ der Steuerung     | Hardwarehandbuch                                 | Programmierhandbuch                                 |
|-----------------------|--|---|
| M221 Logic Controller | Modicon M221 Logic Controller – Hardwarehandbuch | Modicon M221 Logic Controller – Programmierhandbuch |

### TM3 Erweiterungsmodule

Weitere Informationen zur Modulkonfiguration finden Sie in den folgenden Programmier- und Hardwarehandbüchern der jeweiligen Erweiterungsmodultypen:

| Erweiterungsmodultyp                | Hardwarehandbuch                                       | Programmierhandbuch                          |
|-------------------------------------|--|--|
| TM3 Digitale E/A-Erweiterungsmodule | TM3 Digitale E/A-Erweiterungsmodule – Hardwarehandbuch | TM3-Erweiterungsmodule – Programmierhandbuch |
| TM3 Analoge E/A-Erweiterungsmodule  | TM3 Analoge E/A-Module – Hardwarehandbuch              |  |
| TM3-E/A-Experten-Erweiterungsmodule | TM3 E/A-Expertenmodule – Hardwarehandbuch              |  |
| TM3-Sicherheitsmodule               | TM3 Sicherheitsmodule – Hardwarehandbuch               |  |
| TM3 Sender- und Empfängermodule     | TM3 Sender- und Empfängermodule – Hardwarehandbuch     |  |

### TM2-Erweiterungsmodule

Weitere Informationen zur Modulkonfiguration finden Sie in den Programmier- und Hardwarehandbüchern der jeweiligen Erweiterungsmodultypen:

| Erweiterungsmodultyp    | Hardwarehandbuch                           | Programmierhandbuch                          |
|-------------------------|--|--|
| TM2 Digitale E/A-Module | TM2 Digitale E/A-Module – Hardwarehandbuch | TM2-Erweiterungsmodule – Programmierhandbuch |
| TM2 Analoge E/A-Module  | TM2 Analoge E/A-Module – Hardwarehandbuch  |  |

### TMC2-Steckmodule

Detaillierte Informationen zur Steckmodulkonfiguration finden Sie in folgenden Programmier- und Hardwarehandbüchern:

| Steckmodultyp    | Hardwarehandbuch                    | Programmierhandbuch                    |
|------------------|-------------------------------------|--|
| TMC2-Steckmodule | TMC2 Steckmodule – Hardwarehandbuch | TMC2-Steckmodule – Programmierhandbuch |

### TMH2GDB Dezentrale Grafikanzeige: Remote Graphic Display

Informationen zu Installation, Kompatibilität, Konfiguration und Betrieb des Dezentrale Grafikanzeige: Remote Graphic Display finden Sie in folgendem Handbuch:

| Anzeigetyp                                       | Benutzerhandbuch  |
|--|---|
| Dezentrale Grafikanzeige: Remote Graphic Display | TMH2GDB Dezentrale Grafikanzeige: Remote Graphic Display – Benutzerhandbuch |

## Unterstützte Programmiersprachen

### Übersicht

Eine speicherprogrammierbare Steuerung (Logiksteuerung) liest Eingänge, schreibt Ausgänge und arbeitet die Logik entsprechend eines Steuerungsprogramms ab. Zur Erstellung eines Steuerungsprogramms für eine Steuerung wird eine Reihe von Anweisungen in einer der unterstützten Programmiersprachen geschrieben.

SoMachine Basic unterstützt folgende Programmiersprachen gemäß IEC-61131-3:

- Kontaktplan (KOP oder LD: Ladder)
- Anweisungsliste (AWL oder IL: Instruction List)
- Grafcet (Liste)
- Grafcet (SFC)

---

# Abschnitt 1.2

## Grundlagen der SoMachine Basic-Benutzeroberfläche

---

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema   | Seite |
|---|-------|
| Erstellen von Projekten mit SoMachine Basic   | 28    |
| Entwickeln von Programmen mit SoMachine Basic | 29    |
| Navigation in SoMachine Basic                 | 30    |
| Betriebsmodi                                  | 31    |

## Erstellen von Projekten mit SoMachine Basic

### Übersicht

SoMachine Basic ist ein grafisches Programmierwerkzeug für die Konfiguration, Entwicklung und Inbetriebnahme von Programmen für Steuerungen.

### Einige Grundbegriffe

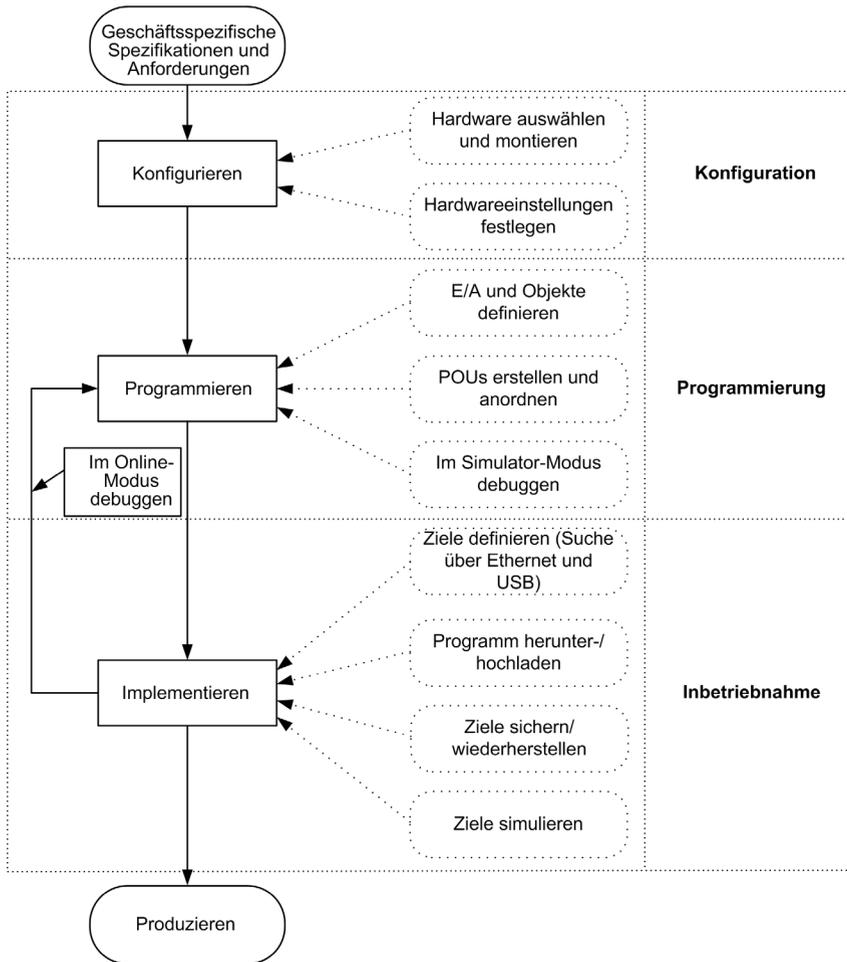
In SoMachine Basic werden folgende Begriffe verwendet:

- **Projekt:** Ein SoMachine Basic-Projekt enthält Informationen über den Entwickler und den Zweck des Projekts, die Konfiguration der vom Projekt zu kontrollierenden Logiksteuerung und zugehörigen Erweiterungsmodule, den Quellcode eines Programms, Symbole, Kommentare, Dokumentation und sonstige zugehörige Informationen.
- **Anwendung:** Enthält alle Teile des in die Steuerung heruntergeladenen Projekts. Dazu gehören das kompilierte Programm, Informationen zur Hardwarekonfiguration sowie andere Daten als Programmdateien (Keine Programmdateien), d. h. Projekteigenschaften, Symbole und Kommentare.
- **Programm:** Der kompilierte Quellcode, der in der Steuerung ausgeführt wird.
- **POU (Program Organization Unit):** Das wiederverwendbare Objekt, das eine Variablendeklaration und einen Satz Anweisungen enthält, die in einem Programm verwendet werden.

## Entwickeln von Programmen mit SoMachine Basic

### Einführung

Das nachstehende Diagramm zeigt die typischen Phasen der Projektentwicklung in SoMachine Basic (Registerkarten **Konfiguration**, **Programmierung** und **Inbetriebnahme**):



## Navigation in SoMachine Basic

### Startseite

Das Fenster **Startseite** wird beim Start von SoMachine Basic angezeigt. Über dieses Fenster können Sie Ihre SoMachine Basic-Software registrieren, die Verbindung zur Steuerung verwalten und ein Projekt erstellen bzw. zur Bearbeitung auswählen.

### Modulbereiche

Sobald Sie ein Projekt ausgewählt haben, zeigt SoMachine Basic das Hauptfenster an.

Die Symbolleiste (*siehe Seite 55*) am oberen Fensterrand enthält eine Reihe von Symbolen, die die Durchführung allgemeiner Aufgaben ermöglichen, einschließlich der Rückkehr zur **Startseite**.

Neben der Symbolleiste befindet sich der Statusbereich (*siehe Seite 57*), in dem Informationsmeldungen zum aktuellen Status der Verbindung mit der Steuerung angezeigt werden.

Darunter ist das Hauptfenster in eine Reihe von *Modulen* untergliedert. Jedes Modul bezieht sich auf eine andere Phase des Entwicklungszyklus und kann durch Klicken auf eine Registerkarte am oberen Rand des Modulbereichs aufgerufen werden. Zur Entwicklung einer Anwendung werden die Module von links nach rechts bearbeitet.

- **Eigenschaften** (*siehe Seite 65*)  
Konfigurieren Sie die Projekteigenschaften.
- **Konfiguration** (*siehe Seite 71*)  
Definieren Sie die Hardwarekonfiguration der Logiksteuerung und der zugehörigen Erweiterungsmodule.
- **Programmierung** (*siehe Seite 75*)  
Entwickeln Sie Ihr Programm in einer der unterstützten Programmiersprachen.
- **Anzeige** (*siehe Modicon M221, Logic Controller, Programmierhandbuch*)  
Erstellen Sie eine Bedienerschnittstelle für das Modul TMH2GDB Dezentrale Grafikanzeige: Remote Graphic Display.
- **Inbetriebnahme** (*siehe Seite 259*)  
Verwalten Sie die Verbindung zwischen SoMachine Basic und der Steuerung, laden Sie Anwendungen hoch/herunter und nehmen Sie eine Anwendung in Betrieb.

## Betriebsmodi

### Einführung

Die Betriebsarten bieten Steuerungsmöglichkeiten zum Entwickeln, Debuggen, Überwachen und Ändern der Anwendung, wenn die Steuerung mit SoMachine Basic verbunden oder nicht verbunden ist.

SoMachine Basic kann in den folgenden Modi betrieben werden:

- Offline-Modus
- Online-Modus
- Simulator-Modus

### Offline-Modus

SoMachine Basic läuft im Offline-Modus, wenn keine physische Verbindung zu einer Steuerung hergestellt wurde.

Im Offline-Modus können Sie SoMachine Basic konfigurieren und an die zu kontrollierenden Hardwarekomponenten anpassen und dann Ihre Anwendung entwickeln.

### Online-Modus

SoMachine Basic arbeitet im Online-Modus, wenn eine Steuerung physisch an den PC angeschlossen ist.

Im Online-Modus können Sie die Anwendung in die Logiksteuerung herunterladen (Das Herunter- und Hochladen der Anwendung ist im Simulator-Modus nicht möglich, da die Anwendung direkt in der simulierten Logiksteuerung gespeichert wird). SoMachine Basic synchronisiert dann die Anwendung im PC-Speicher mit der in der Logiksteuerung gespeicherten Version, sodass Sie die Anwendung debuggen, überwachen und ändern können.

Sie können bestimmte Elemente eines Programms im Online-Modus bearbeiten. Beispielsweise können Sie Programmbausteine hinzufügen oder löschen oder die Werte bestimmter Funktionsbausteinparameter ändern.

**HINWEIS:** Online-Programmänderungen unterliegen der vordefinierten Konfiguration. Siehe Speicherverwaltung (*siehe Seite 47*) Weitere Informationen finden Sie unter Debugging im Online-Modus (*siehe Seite 245*)

### Simulator-Modus

SoMachine Basic wird im Online-Modus betrieben, wenn eine physische Verbindung zu einer Logiksteuerung hergestellt wurde. Im Simulator-Modus wird keine physische Verbindung zu einer Logiksteuerung hergestellt; stattdessen simuliert SoMachine Basic eine Verbindung zu einer Logiksteuerung und den Erweiterungsmodulen, um das Programm auszuführen und zu testen.

Weitere Informationen finden Sie unter SoMachine Basic-Simulator (*siehe Seite 286*).



---

# Kapitel 2

## Arbeiten mit SoMachine Basic

---

# Abschnitt 2.1

## Fenster „Startseite“

---

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                                      | Seite |
|--|-------|
| Einführung in die Startseite               | 35    |
| Registrierung der SoMachine Basic-Software | 36    |
| Fenster „Projekte“                         | 37    |
| Fenster „Verbinden“                        | 41    |
| Direktes Herunterladen einer Anwendung     | 46    |
| Speicherverwaltung                         | 47    |
| Fenster „Projektvorlagen“                  | 48    |
| Hilfefenster                               | 49    |

## Einführung in die Startseite

### Übersicht

Das Fenster „Startseite“ ist das erste Fenster, das beim Start von SoMachine Basic geöffnet wird.

Das Fenster „Startseite“ verfügt über folgende Fenster:

- **Registrieren** (*siehe Seite 36*)  
Zur Registrierung der SoMachine Basic-Software und Anzeige der Lizenzdetails.
- **Projekte** (*siehe Seite 37*)  
Zur Erstellung eines neuen Projekts oder zum Öffnen eines vorhandenen Projekts.
- **Verbinden** (*siehe Seite 41*)  
Zur Herstellung einer Verbindung mit einer Logiksteuerung, Download/Upload der Anwendung an die bzw. von der Steuerung, Sicherung/Wiederherstellung des Steuerungsspeichers sowie zum Blinken der LEDs der verbundenen Steuerung.
- **Vorlagen** (*siehe Seite 48*)  
Zur Erstellung eines neuen Projekts unter Verwendung eines Beispielprojekts als Vorlage.
- **Hilfe** (*siehe Seite 49*)  
Zum Anzeigen der Online-Hilfe, zugehöriger Dokumenten, Schulungsmaterialien und Anleitungen.
- **Info über**  
Zur Anzeige von Informationen über SoMachine Basic.
- **Beenden**  
Zum Beenden von SoMachine Basic.

## Registrierung der SoMachine Basic-Software

### Überblick

Sie können die SoMachine Basic-Software 30 Tage nutzen, bevor Sie die Software registrieren müssen. Zum Zeitpunkt der Registrierung erhalten Sie einen Genehmigungscode zur Verwendung der Software.

Durch die Registrierung der SoMachine Basic-Software erhalten Sie Anspruch auf technischen Support sowie Software-Updates.

### Registrieren

Gehen Sie zur Registrierung Ihrer SoMachine Basic-Software vor wie folgt:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Schaltfläche <b>Jetzt registrieren</b> am oberen Rand der <b>Startseite</b> .  |
| 2       | Halten Sie sich an die Anweisungen im Registrierungsassistenten. Klicken Sie auf die Schaltfläche <b>Hilfe</b> , um detaillierte Informationen anzuzeigen. |

Um detaillierte Informationen zu dem auf Ihrem PC installierten Lizenzschlüssel anzuzeigen, klicken Sie auf **Info über** auf der **Startseite**.

## Fenster „Projekte“

### Übersicht

Im Fenster **Projekte** können Sie ein neues SoMachine Basic-Projekt erstellen oder ein vorhandenes SoMachine Basic-, TwidoSoft- oder TwidoSuite-Projekt zur Bearbeitung öffnen.

Am rechten Rand des Fensters **Projekte** sind Links zu weiteren hilfreichen Informationen verfügbar.

### Öffnen einer SoMachine Basic-Projektdatei

Gehen Sie wie folgt vor, um eine Projektdatei zu öffnen:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie im Fenster <b>Startseite</b> auf <b>Projekte</b> .  |
| 2       | Sie haben dann verschiedene Möglichkeiten: <ul style="list-style-type: none"><li>● Klicken Sie auf ein kürzlich erstelltes Projekt in der Liste <b>Zuletzt verwendete Projekte</b>.</li><li>● Klicken Sie auf <b>Neues Projekt anlegen</b>.</li><li>● Klicken Sie auf <b>Vorhandenes Projekt öffnen</b> und wählen Sie eine vorhandene SoMachine Basic-Projektdatei (*.smbp) oder eine Beispielprojektdatei (*.smbe) aus.</li></ul> |

| Schritt | Aktion  |
|---------|---|
| 3       | <p><b>Fall 1</b><br/> Wenn ein Fenster mit der Aufforderung zur Passworteingabe angezeigt wird, dann ist das Projekt passwortgeschützt:</p> <ol style="list-style-type: none"> <li>1. Geben Sie das Verschlüsselungspasswort ein.</li> <li>2. Klicken Sie auf <b>Übernehmen</b></li> <li>3. Gehen Sie wie folgt vor, um das Projekt zu ändern: <ol style="list-style-type: none"> <li>a. Klicken Sie auf  auf der Registerkarte <b>Eigenschaften</b>.</li> </ol> <p><b>Ergebnis:</b> Ein Fenster mit der Aufforderung zur Passworteingabe wird angezeigt.</p> <ol style="list-style-type: none"> <li>b. Geben Sie das Änderungspasswort ein.</li> <li>c. Klicken Sie auf <b>Übernehmen</b>.</li> </ol> </li> </ol> <p><b>Ergebnis:</b> Die Projektdatei wird geöffnet und die Registerkarte <b>Konfiguration</b> wird angezeigt.</p> <p><b>Fall 2</b><br/> Wenn auf der Registerkarte <b>Eigenschaften</b> ein Fehlersymbol angezeigt wird, bedeutet das, dass das Projekt, das Sie öffnen möchten, in einer früheren Version von SoMachine Basic mit einem Passwort versehen wurde, wobei <b>Anzeigen und herunterladen</b> ausgewählt war:</p> <ol style="list-style-type: none"> <li>1. Klicken Sie auf die <b>Registerkarte „Eigenschaften“</b> → <b>Projektschutz</b>.</li> <li>2. Klicken Sie auf  auf der Registerkarte <b>Eigenschaften</b>.</li> <li>3. Geben Sie ein Passwort ein, um das Projekt zu verschlüsseln.<br/> Um das Projekt speichern zu können, muss es verschlüsselt werden.</li> <li>4. Klicken Sie auf <b>Übernehmen</b>.</li> </ol> <p><b>Fall 3</b><br/> Wenn das Fenster <b>Fehler</b> ein Fehlersymbol angezeigt wird, bedeutet das, dass das Projekt, das Sie öffnen möchten, in einer früheren Version von SoMachine Basic mit einem Passwort versehen wurde, wobei <b>Nur herunterladen</b> ausgewählt war:</p> <ol style="list-style-type: none"> <li>1. Klicken Sie auf <b>OK</b><br/> <b>Ergebnis:</b> Die Registerkarte <b>Eigenschaften</b> wird angezeigt.</li> <li>2. Klicken Sie auf <b>Projektschutz</b>.</li> <li>3. Klicken Sie auf  und geben Sie das Projekt-Passwort ein.</li> <li>4. Wenn Sie den Projektschutz entfernen möchten, wählen Sie <b>Inaktiv</b> aus und klicken Sie auf <b>Übernehmen</b>.<br/> Wenn Sie den Projektschutz beibehalten möchten, geben Sie das Verschlüsselungspasswort ein, wählen Sie <b>Anzeigen und herunterladen</b> aus und klicken Sie auf <b>Übernehmen</b>.</li> </ol> |

## Öffnen einer TwidoSuite- oder TwidoSoft-Projektdatei

Mit SoMachine Basic können Sie für Twido-SPS erstellte Anwendungen öffnen und diese zu SoMachine Basic-Projektdateien konvertieren.

Gehen Sie wie folgt vor, um eine TwidoSuite- oder TwidoSoft-Projektdatei zu öffnen:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie im Fenster <b>Startseite</b> auf <b>Projekte</b> .   |
| 2       | <p>Klicken Sie auf <b>Vorhandenes Projekt öffnen</b> und wählen Sie in der Liste <b>Dateityp</b> einen der folgenden Dateitypen aus. Navigieren Sie dann zu einem vorhandenen Projekt mit der entsprechenden Dateierweiterung und wählen Sie es aus:</p> <ul style="list-style-type: none"> <li>● TwidoSuite-Projektdateien (*.xpr)</li> <li>● Twido Archiv-Projektdateien (*.xar)</li> <li>● TwidoSoft-Projektdateien (*.twd)</li> </ul> <p>Wenn die ausgewählte Twido-Projektdatei in TwidoSoft geöffnet ist, dann sperrt TwidoSoft die Projektdatei und sie kann nicht in SoMachine Basic geöffnet werden. Schließen Sie das Projekt in TwidoSoft bevor Sie es in SoMachine Basic öffnen.</p> <p><b>Ergebnis:</b> Die ausgewählte Projektdatei wird geöffnet und die Registerkarte <b>Konfiguration</b> wird angezeigt.</p> |
| 3       | Ein Fenster mit einem Konvertierungsbericht erscheint. Prüfen Sie die Konvertierungsergebnisse sorgfältig, um eventuelle Anomalien festzustellen, die bei der Konvertierung von einer Steuerungsplattform zu einer anderen entstanden sind. Für Hilfe bei der Behebung dergleicher Anomalien, siehe Konvertieren von Twido-Projekten zu SoMachine Basic ( <i>siehe Seite 307</i> ).  |

**HINWEIS:** TwidoSuite verwendet %I0.0.1 (oder %I0.0.7) als Impulseingang auf dem Funktionsbaustein Very Fast Counter (%VFC). In SoMachine Basic verwendet der entsprechende High Speed Counter (%HSC) -Funktionsbaustein %I0.0 (oder %I0.6) Nehmen Sie nach der Konvertierung angemessene Änderungen an Ihren Anwendungen vor.

Im Allgemeinen ist die Konvertierung von anderen Steuerungsplattformen zur M221 Logic Controller- und SoMachine Basic-Plattform insofern eingeschränkt, als Unterschiede zwischen diesen Plattformen bestehen. Daher ist es notwendig, diese Unterschiede wie im Hinweis oben beschrieben, manuell zu beheben.

## **WARNUNG**

### **UNBEABSICHTIGTER GERÄTEBETRIEB**

- Vergewissern Sie sich immer, dass Ihr Anwendungsprogramm weiterhin so funktioniert wie vor der Konvertierung, mit den richtigen Konfigurationen, Parametern, Parameterwerten, Funktionen und Funktionsbausteinen, die erforderlich sind.
- Ändern Sie die Anwendung, so dass es dem vorherigen Betrieb entspricht.
- Testen Sie Ihre neu kompilierte Version sorgfältig und validieren Sie sie, bevor Sie die Anwendung in Betrieb nehmen.

**Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.**

## Fenster „Verbinden“

### Verbundene Geräte

Im Fenster **Verbinden** werden zwei Gerätelisten angezeigt:

#### 1. Lokale Geräte

Zeigt alle Geräte an, die mit dem PC verbunden sind und den Zugriff auf Logiksteuerungen ermöglichen:

- über die physischen COM-Ports des PCs (z. B. COM1)
- über USB-Kabel
- über die virtualisierten COM-Ports (über USB-zu-Seriell-Konverter oder Bluetooth-Dongles)
- über Modem(s) und zugehörige Telefonnummern, die Sie der Liste manuell hinzufügen

**HINWEIS:** Wenn ein COM-Port ausgewählt ist und das Kontrollkästchen **Modbus-Treiberparameter beibehalten** aktiviert ist, wird die Kommunikation mit den im Modbus-Treiber definierten Parameter hergestellt.

#### 2. Ethernet-Geräte

In dieser Liste sind alle Steuerungen enthalten, die sich im selben Ethernet-Teilnetzwerk befinden wie der PC, auf dem SoMachine Basic ausgeführt wird. Geräte hinter einem Router oder einem anderen Gerät, das UDP-Broadcast-Sendungen blockiert, werden nicht aufgeführt. Die Liste enthält Steuerungen, die automatisch von SoMachine Basic erkannt werden, sowie sämtliche Steuerungen, die Sie manuell hinzufügen.

Sie können die Schaltfläche **Blinken der LEDs starten** nur für Logiksteuerungen verwenden, die automatisch hinzugefügt werden (bei aktiver Option **Automatisches Erkennungsprotokoll aktiviert**).

### Manuelles Hinzufügen von Steuerungen

Gehen Sie wie folgt vor, um der Liste **Ethernet-Geräte** eine Logiksteuerung hinzuzufügen:

| Schritt | Aktion   |
|---------|--|
| 1       | Geben Sie im Feld <b>Dezentraler Lookup</b> die IP-Adresse der hinzuzufügenden Steuerung ein, z. B. „12.123.134.21“. |
| 2       | Klicken Sie auf <b>Hinzufügen</b> , um das Gerät in der Liste <b>Ethernet-Geräte</b> hinzuzufügen.                   |

### Hinzufügen von Modemverbindungen

Gehen Sie zum Hinzufügen einer Modemverbindung in der Liste **Lokale Geräte** vor wie folgt:

| Schritt | Aktion  |
|---------|---|
| 1       | <p>Klicken Sie auf die Schaltfläche  <b>Modemkonfiguration hinzufügen</b>.</p> <p><b>Ergebnis:</b> Das Fenster <b>Modemkonfiguration</b> wird angezeigt.</p>   |
| 2       | <p>Wählen Sie aus der Dropdown-Liste den <b>COM-Port</b> des Modems aus:</p> <div data-bbox="322 422 861 1117" style="border: 1px solid gray; padding: 10px;"> <p style="text-align: right; color: green;">Konfiguration des Modems <span style="float: right; color: red;">✕</span></p> <hr style="border: 5px solid green;"/> <p style="color: green; margin-top: 10px;">Einstellungen</p> <p>COM-Port <span style="float: right;">COM2 ▾</span></p> <p>Gerät <span style="float: right;">ELTIMA Virtual Serial Port (COM2)</span></p> <p>Telefonnummer <span style="float: right;">0512345678</span></p> <p>Modeminitialisierungsbefehl <span style="float: right;">AT&amp;D0</span></p> <p>Escape-Zeichen <span style="float: right;">+</span></p> <p>Baudrate <span style="float: right;">19200 ▾</span></p> <p>Parität <span style="float: right;">Gerade ▾</span></p> <p>Datenbits <span style="float: right;">8 ▾</span></p> <p>Stoppbits <span style="float: right;">1 ▾</span></p> <p>Timeout (ms) <span style="float: right;">15000</span></p> <p>Inter-Char Timeout (ms) <span style="float: right;">10</span> <input type="checkbox"/> Automatisch</p> <p style="text-align: right; margin-top: 10px;"> <input type="button" value="Übernehmen"/> <input type="button" value="Abbrechen"/> </p> </div> |
| 3       | <p>Konfigurieren der Kommunikationsparameter.</p> <p>Weitere Informationen über die Parameter zur Modemkonfiguration finden Sie in der nachstehenden Tabelle.</p>   |
| 4       | <p>Klicken Sie auf <b>Übernehmen</b>.</p> <p><b>HINWEIS:</b> Diese Schaltfläche ist nur aktiviert, wenn alle Einstellungen konfiguriert sind.</p> <p><b>Ergebnis:</b> Die Modemverbindung wird der Liste <b>Lokale Geräte</b> hinzugefügt (z. B. <b>COM2@0612345678,GenericModem</b>).</p>  |
| 5       | <p>Im Fenster <b>Inbetriebnahme</b> (<i>siehe Seite 260</i>), geben Sie die <b>Unit-ID</b> ein, um die im Slave-Gerät konfigurierte Adresse (<i>siehe Modicon M221, Logic Controller, Programmierhandbuch</i>) abzustimmen.</p>   |

| Schritt | Aktion   |
|---------|--|
| 6       | Falls erforderlich, können Sie die <b>Modemkonfiguration</b> ändern, indem Sie das zu bearbeitende Modem in der Liste <b>Lokale Geräte</b> auswählen und auf die Schaltfläche  <b>Modemkonfiguration ändern</b> über der Liste klicken. |

### Parameter für die Modemkonfiguration

In dieser Tabelle werden die Parameter zur Modemkonfiguration beschrieben:

| Parameter                                  | Wert  | Standardwert | Beschreibung  |
|--|---|--------------|---|
| <b>COM-Port</b>                            | <b>COMx</b>   | -            | Ermöglicht die Auswahl des <b>COM-Ports</b> des Modems aus der Dropdown-Liste.  |
| <b>Gerät</b>                               | -   | -            | Enthält den Namen des Modems.   |
| <b>Telefonnummer</b>                       | -   | -            | Ermöglicht das Eingeben der Telefonnummer des Modems, das mit der Logiksteuerung verbunden ist.<br>In dieses Textfeld können alle Zeichen eingegeben werden, insgesamt bis zu 32 Zeichen. Es muss mindestens ein Zeichen eingegeben werden, damit die Konfiguration angewendet werden kann. |
| <b>Befehl für die Modeminitialisierung</b> | -   | AT&D0        | Ermöglicht das Bearbeiten des AT-Initialisierungsbefehls des Modems.<br>Der AT-Initialisierungsbefehl ist optional (falls das Feld leer ist, wird die Zeichenfolge <code>AT</code> gesendet).   |
| <b>Escape-Zeichen</b>                      | -   | +            | Ermöglicht das Bearbeiten des Escape-Zeichens für den Auflegvorgang.  |
| <b>Baudrate</b>                            | 1200<br>2400<br>4800<br>9600<br>19200<br>38400<br>57600<br>115200 | 19200        | Ermöglicht das Auswählen der Datenübertragungsrate des Modems.  |
| <b>Parität</b>                             | Ohne<br>Gerade<br>Ungerade  | Gerade       | Ermöglicht die Auswahl der Parität der übertragenen Daten für die Fehlererkennung.  |
| <b>Datenbits</b>                           | 7<br>8  | 8            | Ermöglicht die Auswahl der Anzahl der Datenbits.  |
| <b>Stoppbits</b>                           | 1<br>2  | 1            | Ermöglicht die Auswahl der Anzahl der Stopp-Bits.   |

| Parameter               | Wert        | Standardwert | Beschreibung   |
|-------------------------|-------------|--------------|--|
| Timeout (ms)            | 0 bis 60000 | 15000        | Ermöglicht das Festlegen des Übertragungszeitlimits (in ms).   |
| Eingabeverzögerung (ms) | 0 bis 10000 | 10           | Ermöglicht das Festlegen der Eingabeverzögerung (in ms).<br>Wenn das Kontrollkästchen <b>Automatisch</b> aktiviert ist, wird der Wert automatisch berechnet. |

### Aufbauen einer Verbindung zu einer Steuerung

Gehen Sie wie folgt vor, um eine Steuerung mit SoMachine Basic zu verbinden:

| Schritt | Aktion   |
|---------|--|
| 1       |  <p>Klicken Sie auf  (Schaltfläche <b>Geräte aktualisieren</b>), um die Liste verbundener Geräte zu aktualisieren.</p>   |
| 2       | <p>Wählen Sie eine der Logiksteuerungen in der Liste <b>Lokale Geräte</b> oder <b>Ethernet-Geräte</b> aus. Wenn eine Steuerung über Ethernet auf demselben Netzkabel wie Ihr PC verbunden ist, wird die IP-Adresse der Steuerung in der Liste angezeigt. Durch die Auswahl der IP-Adresse in der Liste wird  (Schaltfläche für <b>IP-Adressenkonfiguration</b>) verfügbar. Klicken Sie auf diese Schaltfläche, um die IP-Adresse der Steuerung zu ändern.</p> <p><b>HINWEIS:</b> Wenn Sie das Kontrollkästchen <b>Post-Konfigurationsdatei schreiben</b> aktivieren, werden die Ethernet-Parameter in der Post-Konfigurationsdatei geändert und auch nach dem Ein- und Ausschalten beibehalten.</p> |
| 3       |  <p>Falls erforderlich, klicken Sie auf  (Schaltfläche <b>Blinken der LEDs starten</b>), um die LEDs der ausgewählten Steuerung blinken zu lassen und dadurch die Steuerung physisch zu identifizieren. Klicken Sie erneut auf diese Schaltfläche, um das Blinken der LEDs einzustellen.</p> <p><b>HINWEIS:</b> Sie können die Schaltfläche <b>Blinken der LEDs starten</b> nur für Logiksteuerungen verwenden, die automatisch hinzugefügt werden (bei aktiver Option <b>Automatisches Erkennungsprotokoll aktiviert</b>).</p>  |
| 4       | <p>Klicken Sie auf <b>Login</b>, um sich bei der ausgewählten Steuerung anzumelden. Wenn die Logiksteuerung über Passwortschutz verfügt, werden Sie zur Eingabe des Passworts aufgefordert. Geben Sie das Passwort ein und klicken Sie auf <b>OK</b>, um die Verbindung herzustellen.</p> <p><b>Ergebnis:</b> Es wird eine Statusleiste angezeigt, anhand der Sie den Status des Verbindungsaufbaus mitverfolgen können.</p>   |

| Schritt | Aktion  |
|---------|---|
| 5       | <p>Sobald die Verbindung erfolgreich hergestellt wurde, werden Detailinformationen zur Steuerung im Fensterbereich <b>Ausgewählte Steuerung</b> angezeigt und folgende Schaltflächen stehen zur Verfügung:</p> <ul style="list-style-type: none"><li>● <b>Download application to controller:</b> Gehen Sie wie folgt vor, um eine Anwendung in die Logiksteuerung zu laden, ohne sie in SoMachine Basic zu öffnen: Siehe Direktes Herunterladen einer Anwendung (<i>siehe Seite 46</i>).</li><li>● <b>Speicherverwaltung:</b> Zur Sicherung (<i>siehe Seite 274</i>) oder Wiederherstellung (<i>siehe Seite 275</i>) des Speichers der Logiksteuerung auf einem PC bzw. von einem PC. Siehe Speicherverwaltung (<i>siehe Seite 47</i>).</li><li>● <b>Upload application from controller:</b> Zur Erstellung einer neuen SoMachine Basic-Projektdatei durch Hochladen einer Anwendung aus der verbundenen Logiksteuerung. Siehe Hochladen einer Anwendung (<i>siehe Seite 270</i>).</li></ul> |
| 6       | <p>Klicken Sie auf die Schaltfläche <b>Abmelden</b>, um sich bei der ausgewählten Steuerung abzumelden.</p>   |

## Direktes Herunterladen einer Anwendung

### Übersicht

Sie können die in einer Projektdatei enthaltene Anwendung in eine Steuerung herunterladen, ohne das Projekt in SoMachine Basic öffnen zu müssen. Das kann sich als nützlich erweisen, wenn das Projekt verschlüsselt ist, d. h. die Benutzer können das Projekt nur öffnen, wenn Sie über das entsprechende Passwort verfügen.

In diesem Fall ist ausschließlich der Download möglich. Informationen zum Upload einer Anwendung aus einer Steuerung in SoMachine Basic finden Sie unter Upload einer Anwendung (*siehe Seite 270*).

### Direktes Herunterladen einer Anwendung

Gehen Sie zum direkten Herunterladen einer Anwendung in eine Steuerung vor wie folgt:

| Schritt | Aktion  |
|---------|---|
| 1       | Stellen Sie über ein serielles, ein USB- oder ein Ethernet-Kabel eine physische Verbindung zwischen dem PC, auf dem SoMachine Basic ausgeführt wird, und der Steuerung her.   |
| 2       | Wählen Sie die Registerkarte <b>Verbinden</b> im Fenster „Startseite“ aus.  |
| 3       | Wählen Sie die Steuerung in der Liste <b>Lokale Geräte</b> oder <b>Ethernet-Geräte</b> aus und klicken Sie anschließend auf <b>Anmelden</b> .<br><b>Ergebnis:</b> SoMachine Basic stellt eine Verbindung zur Steuerung her.   |
| 4       | Klicken Sie auf <b>Download application to controller</b> .   |
| 5       | Klicken Sie im Feld <b>Projektdatei</b> auf die Schaltfläche „Durchsuchen“, wählen Sie die herunterzuladende SoMachine Basic-Projektdatei (*.smbp) aus und klicken Sie auf <b>Öffnen</b> . Im Fensterbereich <b>Informationen</b> werden daraufhin Informationen zur ausgewählten Projektdatei angezeigt: <ul style="list-style-type: none"> <li>• Informationen dazu, ob die Projektdatei verschlüsselt und mit einem Passwort versehen ist.</li> <li>• Informationen zu der in der Projektdatei enthaltenen Konfiguration, z. B. ob die erkannte Konfiguration des Steuerungssystems mit der im ausgewählten Projekt enthaltenen Konfiguration kompatibel ist.</li> </ul> |
| 6       | SoMachine Basic kompiliert die Anwendung in der ausgewählten Projektdatei. Alle Fehler, die bei der Kompilierung festgestellt werden, werden unter <b>Kompilierungsfehler</b> angegeben. Wenn Kompilierungsfehler festgestellt wurden, lässt SoMachine Basic keinen Anwendungsdownload zu. Öffnen Sie das Projekt in SoMachine Basic, beheben Sie die Fehler und versuchen Sie es dann erneut.  |
| 7       | Vor dem Download können Sie auf die folgenden Schaltflächen klicken, um den aktuellen Status der Steuerung zu kontrollieren: <ul style="list-style-type: none"> <li>• <b>Steuerung stoppen</b></li> <li>• <b>Steuerung starten</b></li> <li>• <b>Steuerung initialisieren</b></li> </ul>  |
| 8       | Klicken Sie auf <b>PC zu SPS (Download)</b> .<br><b>Ergebnis:</b> SoMachine Basic lädt die Anwendung in die verbundene Steuerung.   |

## Speicherverwaltung

### Übersicht

Klicken Sie auf die Schaltfläche **Speicherverwaltung** im Fenster **Verbinden**, um den Speicher der Steuerung zu sichern bzw. wiederherzustellen.

Wählen Sie den auszuführenden Vorgang aus:

- Sichern in einem PC (*siehe Seite 274*)
- Wiederherstellen aus einem PC (*siehe Seite 275*)

## Fenster „Projektvorlagen“

### Übersicht

Sie können Beispielprojekte als Ausgangspunkt für neue SoMachine Basic-Projekte heranziehen.

### Öffnen einer Projektvorlage

Gehen Sie wie folgt vor, um ein neues Projekt anhand einer Projektvorlage zu erstellen:

| Schritt | Aktion  |
|---------|---|
| 1       | Wählen Sie die Registerkarte <b>Vorlagen</b> im Fenster <b>Startseite</b> aus.  |
| 2       | <p>Verwenden Sie das Textfeld <b>In Vorlagen suchen</b> im oberen rechten Bereich des Fensters, um nach Projekten zu suchen. Während Sie Ihre Suche eingeben, sucht SoMachine Basic im Projektnamen, in der Beschreibung des im unteren Teil des Fensters verfügbaren Projekts und in den Projekteigenschaften. Während Sie Ihre Suche eingeben, erscheinen übereinstimmende Projekte.</p> <p>Wählen Sie eine Projektvorlagendatei (*.smbe) in der Liste <b>Projekte</b> aus und klicken Sie auf <b>Vorlage öffnen</b>.</p> <p><b>Ergebnis:</b> Es wird eine Kopie der ausgewählten Vorlage als neues Projekt erstellt.</p> <p>Wenn bei Projekten eine Hilfedatei mit der Projektvorlage verknüpft ist, dann können Sie die Schaltfläche <b>Zugehörige Hilfe öffnen</b> klicken, damit eine <b>Zugehörige Hilfe</b> geöffnet wird. Diese Option wird, wenn verfügbar, unter der Liste <b>Projekte</b> hervorgehoben.</p> <p><b>HINWEIS:</b> SoMachine Basic stellt zudem für einige Beispielprojekte eine Vijeo-Designer-Anwendungsdatei und ein Systembenutzerhandbuch bereit. Lesen Sie im Bereich <b>Beschreibung</b> die Beschreibung für das ausgewählte Projekt, um zu erfahren, ob diese Dateien für ihr Projekt bereitgestellt werden oder nicht. Falls diese Dateien zur Verfügung gestellt wurden, klicken Sie auf <b>Open associated folder</b> (zugehörigen Ordner öffnen), um die Projektvorlagendateien (*.smbe) und Vijeo-Designer-Anwendungsdateien (*.vdz) im Windows Explorer zu durchsuchen.</p> |

## Hilfefenster

### Überblick

Dieses Fenster enthält Links zu zusätzlichen SoMachine Basic-Ressourcen:

- Das Online-Hilfesystem von SoMachine Basic
- Zugehörige PDF-Dokumente, wie Systembenutzerhandbücher (System User Guides - SUGs), Schulungsmaterialien, Anweisungsblätter und Beschreibungen von Beispielapplikationen
- Schulungsmaterialien für E-Learning
- Lernprogramme
- Informationen zur Konvertierung von Twido-Applikationen zur Nutzung mit SoMachine Basic.



---

# Teil II

## Entwicklung von SoMachine Basic-Anwendungen

---

### Inhalt dieses Teils

Dieser Teil enthält die folgenden Kapitel:

| Kapitel | Kapitelname   | Seite |
|---------|---|-------|
| 3       | Fenster von SoMachine Basic                               | 53    |
| 4       | Eigenschaften   | 65    |
| 5       | Konfiguration   | 71    |
| 6       | Programmierung  | 75    |
| 7       | Inbetriebnahme  | 259   |
| 8       | Simulator   | 285   |
| 9       | Speichern von Projekten und Schließen von SoMachine Basic | 301   |



---

# Kapitel 3

## Fenster von SoMachine Basic

---

# Abschnitt 3.1

## Übersicht über das SoMachine Basic-Fenster

---

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                          | Seite |
|--------------------------------|-------|
| Schaltflächen der Symbolleiste | 55    |
| Statusbereich                  | 57    |
| Systemeinstellungen            | 60    |
| Drucken von Berichten          | 62    |

## Schaltflächen der Symbolleiste

### Einführung

Die Symbolleiste wird am oberen Rand des SoMachine Basic-Fensters angezeigt und ermöglicht den Zugriff auf häufig verwendete Funktionen.

### Symbolleiste

Die Symbolleiste umfasst die folgenden Schaltflächen:

| Symbol  | Beschreibung  |
|---|---|
|    | Neues Projekt erstellen (Strg+N)  |
|    | Ein vorhandenes Projekt öffnen (Strg+O)   |
|    | Das aktuelle Projekt speichern (Strg+S). Klicken Sie auf den Nach-unten-Pfeil, um ein Menü mit weiteren Speicheroptionen anzuzeigen.  |
|    | Einen Bericht drucken (Strg+P). Klicken Sie auf den Nach-unten-Pfeil, um den zu druckenden ( <i>siehe Seite 62</i> ) Bereich auszuwählen oder Berichtinhalt und -format ( <i>siehe Seite 63</i> ) zu konfigurieren.   |
|    | Ausschneiden (Strg+X)   |
|    | Kopieren (Strg+C)   |
|   | Einfügen (Strg+V)   |
|  | Rückgängig machen (Strg+Z). Klicken Sie auf die zuletzt durchgeführte Aktion im Programm-Editor.<br>Klicken Sie auf den Pfeil nach unten, und wählen Sie aus der Liste eine Aktion aus, um alle Aktionen bis einschließlich zur ausgewählten Aktion rückgängig zu machen. Sie können bis zu zehn Aktionen rückgängig machen.                  |
|  | Wiederherstellen (Strg+Y). Klicken Sie einmal auf diese Option, um das letzte Rückgängigmachen rückgängig zu machen.<br>Klicken Sie auf den Pfeil nach unten, und wählen Sie aus der Liste eine Aktion aus, um alle Aktionen bis einschließlich zur ausgewählten Aktion wiederherzustellen. Sie können bis zu zehn Aktionen wiederherstellen. |
|  | Fenster Systemeinstellungen ( <i>siehe Seite 60</i> ) öffnen  |

| Symbol  | Beschreibung  |
|---|---|
|  | Klicken Sie auf den Pfeil nach unten und wählen Sie Aktionen aus der Liste aus. Zeigen Sie die Online-Hilfe oder die Kontexthilfe an, betrachten Sie Vorlagen, Versionshinweise, Einführungen und e-Learning-Dokumente oder kontaktieren Sie den technischen Kundendienst von Schneider Electric. |
|  | Logiksteuerung starten (Strg+M). Nur im Online-Modus verfügbar und wenn sich die Steuerung nicht bereits im <code>RUNNING</code> -Modus befindet.   |
|  | Logiksteuerung beenden (Strg+L). Nur im Online-Modus verfügbar und wenn sich die Steuerung im <code>RUNNING</code> -Modus befindet.   |
|  | Die Steuerung initialisieren. Nur im Online-Modus verfügbar.  |
|  | Das Programm kompilieren.   |
|  | An der ausgewählten Steuerung anmelden (Strg+G) oder von ihr abmelden (Strg+H).<br><b>HINWEIS:</b> Links neben dieser Schaltfläche wird die ausgewählte Steuerung angezeigt.  |
|  | Den SoMachine Basic Simulator ( <i>siehe Seite 286</i> ) starten (Strg+B) oder beenden (Strg+W).  |

## Statusbereich

### Übersicht

Im Statusbereich am oberen Rand des Hauptfensters werden Informationsmeldungen zum aktuellen Status des Systems angezeigt:



- 1 Programmstatus:**  
Gibt an, ob das Programm Fehler erkannt hat.
- 2 Verbindungsstatus:**  
Gibt den Verbindungsstatus zwischen SoMachine Basic und der Logiksteuerung oder der simulierten Logiksteuerung an.
- 3 SPS-Status:**  
Gibt den aktuellen Status der Logiksteuerung an (RUNNING, STOPPED, HALTED usw.).
- 4 Zykluszeit:**  
Gibt die letzte Abfragezeit an.
- 5 Letzter Fehler der Steuerung:**  
Gibt den zuletzt erkannten Fehler an. Die Information werden aus den Systembits und Systemwörtern extrahiert, wenn sich die Logiksteuerung im Status STOPPED oder HALTED befindet.

### Meldungen im Statusbereich

Im Statusbereich können folgende Meldungen erscheinen:

| Meldungstyp       | Mögliche Meldung                      | Beschreibung   |
|-------------------|---------------------------------------|--|
| Programmstatus    | <b>[Keine Fehler]</b>                 | Im Programm wurden keine Fehler erkannt.                   |
|                   | <b>[Programm-Advisory(s) erkannt]</b> | Programm ist unvollständig.                                |
|                   | <b>[Programmfehler erkannt]</b>       | Kein Programm oder im Programm liegen erkannte Fehler vor. |
| Verbindungsstatus | <b>[Nicht verbunden]</b>              | SoMachine Basic wird im Offline-Modus ausgeführt.          |
|                   | <b>[Online]</b>                       | SoMachine Basic wird im Online-Modus ausgeführt.           |

| Meldungstyp                          | Mögliche Meldung        | Beschreibung   |
|--------------------------------------|-------------------------|--|
| SPS-Status<br>(nur im Online-Modus)  | [Nicht verbunden]       | Steuerung ist nicht mit SoMachine Basic verbunden.   |
|                                      | [Angehalten]            | Steuerung befindet sich im Zustand HALTED.<br>Steuerung wurde aufgrund eines Applikationsfehler angehalten.  |
|                                      | [Stop]                  | Steuerung befindet sich im Zustand STOPPED.<br>Die Steuerung verfügt über eine gültige Anwendung, die gestoppt wurde.                                  |
|                                      | [Run]                   | Steuerung befindet sich im Zustand RUNNING.<br>Steuerung führt die Anwendung aus.  |
|                                      | [Spannungslos]          | Steuerung befindet sich im Zustand POWERLESS.<br>Steuerung wird nur vom USB-Kabel versorgt und ist bereit, die Firmware via USB hoch-/herunterzuladen. |
|                                      | [Download der Firmware] | Die Steuerung lädt die Firmware herunter.  |
|                                      | [Firmware-Fehler]       | Firmware-Fehler erkannt. Die Version der Firmware, die in die Steuerung heruntergeladen wird, ist älter als die aktuelle Firmware-Version.             |
|                                      | [Keine Anwendung]       | Steuerung verfügt über keine Anwendung.  |
|                                      | [Einschalten]           | Steuerung wird gestartet (BOOTING).  |
| Abfragezeit<br>(nur im Online-Modus) | [Abfragezeit 0 µs]      | Die letzte Abfragezeit in Mikrosekunden.   |

| Meldungstyp  | Mögliche Meldung   | Beschreibung   |
|--|--|--|
| Letzter erkannter Fehler der Steuerung (nur im Online-Modus) | <b>[Kein(e) Fehler erkannt]</b>                                      | Im Programm wurden keine Systemfehler erkannt.   |
|  | <b>[Die Steuerung konnte nicht in den RUNNING-Zustand wechseln.]</b> | Steuerung ist nicht betriebsfähig.   |
|  | <b>[Batteriestand niedrig]</b>                                       | Der Batteriestand der Steuerung ist niedrig.   |
|  | <b>[Run/Stop-Eingang]</b>  | Steuerung wurde gestoppt, da ein Stoppbefehl erteilt wurde.  |
|  | <b>[Stoppbefehl]</b>   | Steuerung wurde gestoppt, da ein Stoppbefehl erteilt wurde.  |
|  | <b>[Softwarefehler erkannt (Überschreitung der SPS-Abfrage)]</b>     | Die Steuerung wurde aufgrund eines erkannten Fehlers in der Anwendungssoftware angehalten. Zykluszeitüberschreitung der Steuerung. Die Zykluszeit der Steuerung ist höher als der in der Konfiguration vom Benutzerprogramm definierte Zeitraum. |
|  | <b>[Wegen Hardware-Fehler angehalten]</b>                            | Die Steuerung wurde aufgrund eines erkannten Fehlers in der Hardware gestoppt.   |
|  | <b>[Spannungsausfall]</b>  | Steuerung wurde aufgrund von Spannungsausfall gestoppt.  |
|  | <b>[Steuerung ist im Modus 'Start in STOP' konfiguriert]</b>         | Steuerung startet aufgrund der Konfiguration des Startverhaltens im automatischen Anwendungsausführmodus.  |
|  | <b>[Init-Befehl]</b>   | Init im Kaltstart.   |
| <b>[Unbekannter Stopp-Grund: {0}]</b>                        | Nicht identifizierter Grund.   |  |

Eine vollständige Liste der Systembits und -wörter finden Sie im Programmierhandbuch der Logiksteuerung.

## Systemeinstellungen

### Übersicht

In diesem Fenster können Sie die Sprache der SoMachine Basic-Software einstellen, den Kontaktplan-Editor anpassen und die Standardsteuerung auswählen, die auf der Registerkarte **Konfiguration** bei der Erstellung eines neuen Projekts angezeigt werden soll.

### Ändern der Sprache für die Benutzeroberfläche

Gehen Sie wie folgt vor, um die Sprache für die Benutzeroberfläche zu ändern:

| Schritt | Aktion  |
|---------|---|
| 1       | Wählen Sie <b>Systemeinstellungen</b> → <b>Allgemein</b> im Fenster <b>Systemeinstellungen</b> .            |
| 2       | Wählen Sie die zu verwendende Sprache in der Liste <b>Sprache</b> aus.<br>Die Standardsprache ist Englisch. |
| 3       | Klicken Sie auf <b>Übernehmen</b> und schließen Sie das Fenster <b>Systemeinstellungen</b> .                |
| 4       | Schließen und starten Sie SoMachine Basic neu, um die Benutzeroberfläche in der neuen Sprache anzuzeigen.   |

### Ändern der Tastaturkürzel für „Hilfe“

Gehen Sie wie folgt vor, um das Tastaturkürzel für den Zugriff auf die Kontext- oder allgemeine Hilfe zu ändern:

| Schritt | Aktion  |
|---------|---|
| 1       | Wählen Sie <b>Systemeinstellungen</b> → <b>Allgemein</b> im Fenster <b>Systemeinstellungen</b> .  |
| 2       | Wählen Sie <b>F1</b> oder <b>Umschalt + F1</b> , um die Kontexthilfe aufzurufen.<br>Das Tastaturkürzel für die <b>Allgemeine Hilfe</b> wird automatisch aktualisiert. |

### Anpassen des Kontaktplan-Editors

Gehen Sie wie folgt vor, um den Kontaktplan-Editor anzupassen:

| Schritt | Aktion   |
|---------|--|
| 1       | Wählen Sie <b>Systemeinstellungen</b> → <b>Kontaktplan-Editor</b> im Fenster <b>Systemeinstellungen</b> .  |
| 2       | Wählen Sie den <b>Stil der Rasterlinien</b> für den Kontaktplan-Editor aus. <ul style="list-style-type: none"> <li>● <b>Punkte</b> (Standard)</li> <li>● <b>Gestrichelte Linien</b></li> <li>● <b>Linien</b></li> </ul>                                  |
| 3       | Legen Sie die <b>Anzahl Spalten</b> (11 bis 30) für die Zellen im Kontaktplan-Editor fest.<br>Der Standardwert für die Anzahl Zellen ist 11.<br>Weitere Informationen finden Sie unter Programmiergrundlagen für Kontaktplan ( <i>siehe Seite 188</i> ). |

| Schritt | Aktion  |
|---------|---|
| 4       | <p>Wählen Sie unter <b>Tool-Auswahl beibehalten</b> eine der folgenden Optionen aus:</p> <ul style="list-style-type: none"> <li>● <b>Ausgewähltes Tool beibehalten</b> (Standard): Nach der Auswahl und Positionierung eines Grafikelements in einem Programmbaustein wird das zuletzt ausgewählte Grafikelement beibehalten. Dadurch kann dasselbe Elemente erneut in einem Programmbaustein abgelegt werden, ohne dass es wieder ausgewählt werden muss. Drücken Sie die ESC-Taste oder klicken Sie mit der rechten Maustaste in eine leere Zelle im Programmbaustein, um das Zeigertool  auszuwählen.</li> <li>● <b>Zu Zeiger zurückkehren</b>: Nach der Auswahl und Positionierung eines Kontakts oder einer Spule in einem Programmbaustein wird automatisch das Zeigertool  wieder ausgewählt. Wenn dasselbe Kontakt- bzw. Spulenelement erneut eingefügt werden soll, müssen Sie es in der Symbolleiste wieder auswählen.</li> </ul> |
| 5       | <p>Wählen Sie die Einstellung <b>Tastaturkürzel und Stil der Symbolleiste</b> für den Kontaktplan-Editor aus:</p> <ul style="list-style-type: none"> <li>● <b>SoMachine Basic set</b> (Standard)</li> <li>● <b>Asiatische Definition 1</b></li> <li>● <b>Asiatische Definition 2</b></li> <li>● <b>Europäische Definition</b></li> <li>● <b>Amerikanische Definition</b></li> </ul> <p>Für den ausgewählten Stil wird in der Tabelle eine Liste der Tastaturkürzel für alle in der Symbolleiste verfügbaren Schaltflächen angezeigt.</p>  |
| 6       | <p>Klicken Sie auf <b>Übernehmen</b> und schließen Sie das Fenster <b>Systemeinstellungen</b>, um die Änderungen im Kontaktplan-Editor anzuzeigen.</p>  |

### Auswählen einer Standardsteuerung

Gehen Sie wie folgt vor, um eine Standard-Logiksteuerung auszuwählen:

| Schritt | Aktion  |
|---------|---|
| 1       | Wählen Sie <b>Systemeinstellungen</b> → <b>Konfiguration</b> im Fenster <b>Systemeinstellungen</b> .  |
| 2       | Klicken Sie auf <b>Bevorzugte Steuerung</b> und wählen Sie eine Standardsteuerung in der Liste aus.   |
| 3       | Klicken Sie auf <b>Übernehmen</b> und schließen Sie das Fenster <b>Systemeinstellungen</b> .  |
| 4       | Schließen Sie SoMachine Basic und starten Sie es neu, damit bei Erstellung eines neuen Projekts die neue Standard-Logiksteuerung auf der Registerkarte <b>Konfiguration</b> angezeigt wird. |

## Drucken von Berichten

### Beschreibung

Sie können bedarfsgerecht angepasste Berichte erstellen und diese dann drucken oder im PDF-Format auf dem PC speichern.

Über die Schaltfläche **Drucken** stehen folgende Optionen zur Auswahl:

- **Drucken**, um einen benutzerdefinierten Bericht zu drucken, der nach Bedarf eine Liste der Hardwarekomponenten, die Anwendungsarchitektur und den Inhalt des Projekts, des Programms und der Anwendung enthalten kann.
- **Materialliste drucken**, um die Liste der in der Projektkonfiguration verwendeten Hardwarekomponenten zu drucken.
- **Einstellungen**, um den Projektbericht bedarfsgerecht anzupassen, d. h. Sie können zum einen die Elemente auswählen, die in den Bericht aufgenommen werden sollen, und zum anderen das Seitenlayout konfigurieren.

### Drucken des Projektberichts

Gehen Sie vor wie folgt, um den Projektbericht zu drucken:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf den Nach-unten-Pfeil rechts neben der Schaltfläche <b>Drucken</b>  in der Symbolleiste und wählen Sie den Menübefehl <b>Drucken</b> aus. Sie können ebenfalls die Tastenkombination Strg+P drücken. Das Fenster <b>Druckvorschau</b> wird angezeigt.   |
| 2       | <ul style="list-style-type: none"> <li>• Klicken Sie auf  in der Symbolleiste des Fensters <b>Druckvorschau</b>, um den Projektbericht zu drucken.</li> <li>• Klicken Sie auf  in der Symbolleiste des Fensters <b>Druckvorschau</b>, um den Projektbericht als PDF-Datei auf dem PC zu speichern.</li> </ul> |

## Drucken der Materialliste

Gehen Sie vor wie folgt, um die **Materialliste** zu drucken:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf den Nach-unten-Pfeil rechts neben der Schaltfläche <b>Drucken</b>  in der Symbolleiste und wählen Sie den Menübefehl <b>Materialliste drucken</b> aus. Das Fenster <b>Druckvorschau</b> wird angezeigt.  |
| 2       | <ul style="list-style-type: none"> <li>● Klicken Sie auf  in der Symbolleiste des Fensters <b>Druckvorschau</b>, um die <b>Materialliste</b> zu drucken.</li> <li>● Klicken Sie auf  in der Symbolleiste des Fensters <b>Druckvorschau</b>, um die <b>Materialliste</b> als PDF-Datei auf dem PC zu speichern.</li> </ul> |

## Anpassen des Projektberichts

Gehen Sie vor wie folgt, um die in den Projektbericht aufzunehmenden Elemente auszuwählen und das Berichtlayout zu konfigurieren:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf den Nach-unten-Pfeil rechts neben der Schaltfläche <b>Drucken</b>  in der Symbolleiste und wählen Sie den Menübefehl <b>Einstellungen</b> aus. Das Fenster <b>Einstellungen</b> wird angezeigt. |
| 2       | Klicken Sie auf den Knoten <b>Bericht</b> , um die Formateinstellungen für den Bericht zu konfigurieren (Papiergröße, Seitenränder und Ausrichtung).   |

| Schritt | Aktion  |
|---------|---|
| 3       | <p>Wählen Sie die Elemente aus, die in den Projektbericht aufgenommen werden sollen:</p> <ul style="list-style-type: none"> <li>● <b>Beschreibung:</b> Die Projektbeschreibung im Fenster <b>Projektinformationen</b>.</li> <li>● <b>Materialliste:</b> Die Liste der in der Projektkonfiguration verwendeten Hardwarekomponenten.</li> <li>● <b>Hardwarekonfiguration:</b> Die Liste der in der Konfiguration verwendeten Hardwaregeräte: <ul style="list-style-type: none"> <li>○ <b>E/A-Bus:</b> Die Liste der verwendeten E/A-Erweiterungsmodule.</li> <li>○ <b>Steckmodule:</b> Die Liste der verwendeten Steckmodule.</li> </ul> </li> <li>● <b>Softwarekonfiguration:</b> Ermöglicht die Aufnahme bzw. den Ausschluss folgender Elemente: <ul style="list-style-type: none"> <li>○ <b>Konstanten im Wortformat:</b> Die Liste der im Projekt verwendeten Wortkonstantobjekte (%KW).</li> <li>○ <b>Netzwerkobjekte</b> ist eine Liste an Objekten, die zur Kommunikation mit EtherNet/IP- oder Modbus TCP-Geräten verwendet werden.</li> <li>○ <b>Softwareobjekte:</b> Die Liste der im Programm verwendeten Softwareobjekte, z. B. Timer und Zähler.</li> <li>○ <b>PTO-Objekte:</b> Die Liste der im Programm verwendeten PTO-Funktionsbausteine.</li> <li>○ <b>Kommunikationsobjekte:</b> Die Liste der im Programm verwendeten Kommunikationsobjekte.</li> </ul> </li> <li>● <b>Programm:</b> Ermöglicht die Aufnahme bzw. den Ausschluss folgender Elemente: <ul style="list-style-type: none"> <li>○ <b>Verhalten:</b> Die im Fenster <b>Verhalten</b> konfigurierten Einstellungen.</li> <li>○ <b>Speichernutzung:</b> Der Umfang des von der Anwendung, vom Programm und von den zugehörigen Benutzerdaten beanspruchten Steuerungsspeichers.</li> <li>○ <b>Anwendungsarchitektur:</b> Die in den Fenstern <b>Master-Task</b> und <b>Periodische Task</b> konfigurierten Einstellungen.</li> <li>○ <b>POU</b> ist eine Liste aller im Programm verwendeten POU's.</li> </ul> </li> <li>● <b>Anzeige:</b> Eine Bericht-Section mit Informationen zum Dezentrale Grafikanzeige: Remote Graphic Display: <ul style="list-style-type: none"> <li>○ <b>Allgemeine Eigenschaften:</b> Die allgemeinen Parameter, die auf der Registerkarte <b>Anzeige</b> angezeigt werden. Ihnen steht eine Option zum Druck des Passworts in Ihrem Bericht zur Auswahl.</li> <li>○ <b>Alarm-Ansicht:</b> Die Liste der ausgelösten Alarme.</li> <li>○ <b>Seiten:</b> Die Liste der auf der Registerkarte <b>Anzeige</b> erstellten Bedienerschnittstellenseiten.</li> </ul> </li> <li>● <b>Symbole:</b> Die Liste aller Symbole bzw. der im Projekt verwendeten Symbole.</li> <li>● <b>Querverweis:</b> Eine Tabelle mit den verwendeten Adressen, Objekten und Programmbausteinen sowie den Codezeilen, in denen sie verwendet werden.</li> <li>● <b>Animationstabelle:</b> Eine Tabelle mit den den Animationstabellen im Projekt hinzugefügten Objekten.</li> </ul> |
| 4       | Schließen Sie das Fenster.  |

---

# Kapitel 4

## Eigenschaften

---

# Abschnitt 4.1

## Übersicht über das Fenster „Eigenschaften“

---

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                   | Seite |
|-------------------------|-------|
| Fenster „Eigenschaften“ | 67    |
| Projekteigenschaften    | 68    |

## Fenster „Eigenschaften“

### Übersicht

Auf der Registerkarte **Eigenschaften** können Sie Informationen zum Projekt angeben und einen Passwortschutz festlegen:

- Details zum Entwickler und der das Projekt entwickelnden Firma
- Informationen zum Projekt selbst.
- Projekt mit Passwortschutz: Das zum Öffnen des Projekts in SoMachine Basic einzugebende Passwort
- In der Steuerung gespeicherte Anwendung mit Passwortschutz: Das zum Hochladen der Anwendung in ein SoMachine Basic-Projekt einzugebende Passwort

The screenshot shows the 'Eigenschaften' window in SoMachine Basic. The window title is 'Schneider Electric SoMachine Basic' and the project name is 'Neues Projekt'. The 'Eigenschaften' tab is active, showing a list of properties on the left and input fields for each on the right. The properties listed are: Frontseite, Firma, Projektinformationen, Projektschutz, and Anwendungsschutz. The input fields are: Nachname, Vorname, Telefon, Zellennummer, E-Mail, Straße, Ort, Postleitzahl (with '0' entered), Status, and Land. At the bottom right are 'Übernehmen' and 'Abbrechen' buttons. Two blue arrows point to the left and right panes, labeled '1' and '2' respectively.

- 1 Im linken Fensterbereich wird eine Liste der verfügbaren Eigenschaften angezeigt..
- 2 Der rechte Fensterbereich enthält die Eigenschaften des jeweils im linken Bereich ausgewählten Elements..

## Projekteigenschaften

### Übersicht

Im Fenster **Eigenschaften** finden Sie detaillierte Informationen zum Benutzer von SoMachine Basic, zur Firma, die die Anwendung entwickelt hat, sowie zum Projekt. In diesem Fenster können Sie außerdem einen Passwortschutz für die Projektdatei und die Anwendung definieren, wenn diese in der Logiksteuerung gespeichert werden.

### Eingeben von Informationen zum Anwendungsentwickler

Gehen Sie zur Eingabe von Informationen zum Anwendungsentwickler vor wie folgt:

| Schritt | Aktion  |
|---------|---|
| 1       | Öffnen Sie die Registerkarte <b>Eigenschaften</b> und klicken Sie auf <b>Projekteigenschaften</b> → <b>Frontseite</b> . |
| 2       | Geben Sie die zutreffenden Informationen ein.   |
| 3       | Klicken Sie auf <b>Übernehmen</b> .   |

**HINWEIS:** Diese Informationen werden nach einem Rechtsklick auf eine SoMachine Basic-Projektdatei im Eigenschaftsfenster von Windows Explorer angezeigt.

### Eingeben von Informationen zur Firma

Gehen Sie zur Eingabe von Informationen zur Firma vor wie folgt:

| Schritt | Aktion   |
|---------|--|
| 1       | Öffnen Sie die Registerkarte <b>Eigenschaften</b> und klicken Sie auf <b>Projekteigenschaften</b> → <b>Firma</b> .   |
| 2       | Geben Sie die zutreffenden Informationen ein.<br>Um das Bild für das Firmelogo hochzuladen, klicken Sie auf <b>Ändern</b> , und navigieren Sie anschließend zu der zu ladenden Datei. Klicken Sie auf <b>Entfernt</b> , um das aktuelle Bild zu löschen. |
| 3       | Klicken Sie auf <b>Übernehmen</b> .  |

## Eingeben von Informationen zum Projekt

Gehen Sie zur Eingabe von Informationen zum Projekt vor wie folgt:

| Schritt | Aktion  |
|---------|---|
| 1       | Öffnen Sie die Registerkarte <b>Eigenschaften</b> und klicken Sie auf <b>Projekteigenschaften → Projektinformationen..</b>  |
| 2       | Geben Sie die zutreffenden Informationen ein.<br>Wenn Sie ein Bild, z. B. ein Foto oder ein CAD-Bild der instrumentierten Maschine, hochladen möchten, klicken Sie auf <b>Ändern</b> und navigieren Sie zu der Datei, die hochgeladen werden soll. Klicken Sie auf <b>Entfernt</b> , um das aktuelle Bild zu löschen. |
| 3       | Klicken Sie auf <b>Übernehmen</b> .   |

## Passwortschutz für ein Projekt

Eine Projektdatei kann verschlüsselt und mit einem Passwortschutz versehen werden.

Wenn ein Projekt verschlüsselt ist, dann werden Sie bei jedem Versuch, das Projekt zu öffnen, aufgefordert, das Verschlüsselungspasswort einzugeben.

Wenn das Projekt gegen Änderungen geschützt ist, dann kann es standardmäßig nur angezeigt werden. Um Änderungen am Projekt vorzunehmen, müssen Sie das Änderungspasswort eingeben.

Gehen Sie wie folgt vor, um eine Projektdatei zu verschlüsseln und mit einem Passwortschutz zu versehen:

| Schritt | Aktion   |
|---------|--|
| 1       | Öffnen Sie die Registerkarte <b>Eigenschaften</b> und klicken Sie auf <b>Projekteigenschaften → Projektschutz</b> .          |
| 2       | Wählen Sie die Option <b>Aktiv</b> aus. Erforderliche Informationen sind mit einem Sternchen (*) gekennzeichnet.             |
| 3       | Geben Sie zunächst das Passwort ein und wiederholen Sie dann die Eingabe, um die Verschlüsselung des Projekts zu bestätigen. |
| 4       | Geben Sie optional ein Passwort ein und bestätigen Sie dieses, um das Projekt gegen Änderungen zu schützen.                  |
| 5       | Klicken Sie auf <b>Übernehmen</b> .  |

Wenn ein Programm ausschließlich im schreibgeschützten Modus ausgeführt werden soll, dann erstellen Sie ein Steuerungsabbild und stellen Sie es auf der Steuerung wieder her (*siehe Seite 275*).

### Entfernen des Passwortschutzes von einem Projekt

Gehen Sie wie folgt vor, um den Passwortschutz von einem Projekt zu entfernen:

| Schritt | Aktion   |
|---------|--|
| 1       | Öffnen Sie die Registerkarte <b>Eigenschaften</b> und klicken Sie auf <b>Projekteigenschaften</b> → <b>Projektschutz</b> .   |
| 2       | Wählen Sie die Option <b>Inaktiv</b> aus.  |
| 3       | Klicken Sie auf <b>Übernehmen</b> .<br><b>HINWEIS:</b> Wenn Sie aufgefordert werden, das Änderungspasswort einzugeben, dann geben Sie dieses ein und klicken Sie auf <b>Übernehmen</b> . |

### Passwortschutz von Anwendungen

Mit SoMachine Basic kann eine in der Steuerung gespeicherte Anwendung durch ein Passwort geschützt werden. Anhand dieses Passworts lässt sich der Upload der Anwendung von der Steuerung in ein SoMachine Basic-Projekt kontrollieren.

Gehen Sie wie folgt vor, um eine Anwendung mit Passwortschutz zu versehen:

| Schritt | Aktion   |
|---------|--|
| 1       | Öffnen Sie die Registerkarte <b>Eigenschaften</b> und klicken Sie auf <b>Projekteigenschaften</b> → <b>Anwendungsschutz</b> .  |
| 2       | Wählen Sie den Grad des Anwendungsschutzes aus: <ul style="list-style-type: none"> <li>• Wählen Sie <b>Aktiv</b> aus und lassen Sie das Feld <b>Passwort</b> leer, um Anwendungs-Uploads von der Logiksteuerung an den PC zu unterbinden.</li> <li>• Wählen Sie <b>Aktiv</b> aus und geben Sie dasselbe Passwort in die Felder <b>Passwort</b> und <b>Bestätigen</b> ein, um die Anwendung mit einem Passwortschutz zu versehen. Sie müssen dann dieses Passwort eingeben, wenn Sie dazu aufgefordert werden, bevor Sie die Anwendung aus der Logiksteuerung in einen PC hochladen.</li> </ul> |
| 3       | Klicken Sie auf <b>Übernehmen</b> .  |

### Entfernen des Passwortschutzes von einer Anwendung

Gehen Sie wie folgt vor, um den Passwortschutz von einer Anwendung zu entfernen:

| Schritt | Aktion  |
|---------|---|
| 1       | Öffnen Sie die Registerkarte <b>Eigenschaften</b> und klicken Sie auf <b>Projekteigenschaften</b> → <b>Anwendungsschutz</b> .   |
| 2       | Wählen Sie die Option <b>Inaktiv</b> aus.   |
| 3       | Klicken Sie auf <b>Übernehmen</b> .<br><b>HINWEIS:</b> Wenn Sie zur Angabe des aktuellen Passworts aufgefordert werden, bevor die Option <b>Inaktiv</b> effektiv geworden ist, geben Sie das Passwort ein und klicken Sie auf <b>Übernehmen</b> . |

---

# Kapitel 5

## Konfiguration

---

# Abschnitt 5.1

## Übersicht über das Fenster „Konfiguration“

---

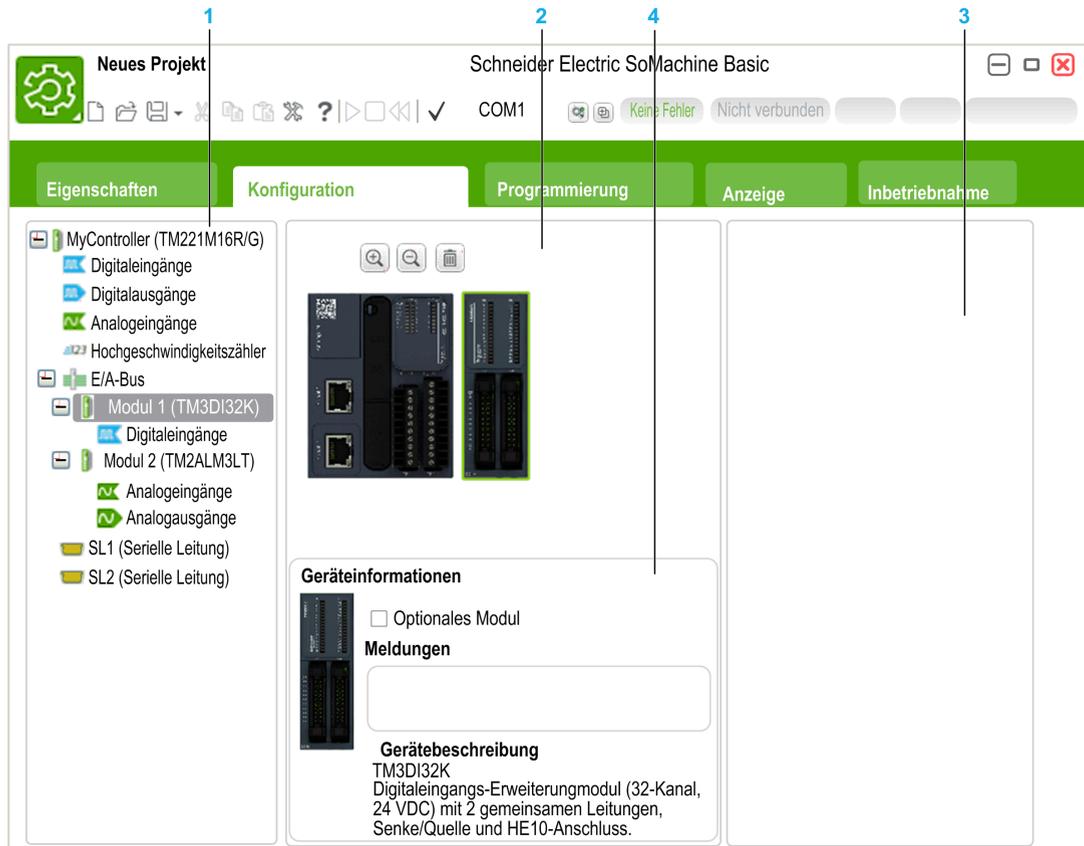
### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                                | Seite |
|--------------------------------------|-------|
| Übersicht des Konfigurationsfensters | 73    |
| Erstellen einer Konfiguration        | 74    |

## Übersicht des Konfigurationsfensters

Im Fenster **Konfiguration** können Sie die Hardwarekonfiguration der vom Programm zu kontrollierenden Steuerung und der Erweiterungsmodule anpassen.



- 1 Die Hardwareübersicht entspricht einer strukturierten Ansicht der Hardwarekonfiguration.
- 2 Die Konfiguration umfasst eine Steuerung und Erweiterungsmodule.
- 3 Katalogreferenzen der unterstützten Hardwarekomponenten für Steuerungen und Erweiterungsmodule. Um eine Komponente in der Hardwarekonfiguration hinzuzufügen, ziehen Sie sie einfach in die Konfiguration und legen Sie sie dort ab.
- 4 Die Eigenschaften der in der Konfiguration ausgewählten Komponente bzw. des in der Hardwareübersicht ausgewählten Elements.

## Erstellen einer Konfiguration

### Ersetzen der Standardsteuerung

Bei der Erstellung eines neuen SoMachine Basic-Projekts wird im zentralen Bereich des Fensters **Konfiguration** eine Steuerungsreferenz angezeigt.

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Konfiguration</b> .  |
| 2       | Erweitern Sie die Kategorie der Logiksteuerungen im Katalogbereich auf der rechten Seite, sofern diese noch nicht angezeigt wird.                             |
| 3       | Wählen Sie eine Steuerungsreferenz aus. Im Bereich <b>Gerätebeschreibung</b> wird eine Kurzbeschreibung der physischen Eigenschaften der Steuerung angezeigt. |
| 4       | Ziehen Sie die Steuerungseferenz auf das Abbild der vorhandenen Steuerung im zentralen Fensterbereich und legen Sie sie dort ab.                              |
| 5       | Klicken Sie auf <b>Ja</b> , wenn Sie aufgefordert werden, das Ersetzen der Steuerungsreferenz zu bestätigen.  |

**HINWEIS:** Die Standard-Steuerungsreferenz wird im Fenster **Systemeinstellungen** angegeben (*siehe Seite 60*).

### Konfigurieren der Logiksteuerung

Die Konfiguration der Steuerung erfolgt im Fenster **Konfiguration**.

Detaillierte Informationen hierzu finden Sie im *Programmierhandbuch* der in der Konfiguration verwendeten Steuerung.

### Konfigurieren der Erweiterungsmodule

Verwenden Sie das Fenster **Konfiguration**, um Erweiterungsmodule hinzuzufügen und zu konfigurieren.

Detaillierte Informationen hierzu finden Sie im *Programmierhandbuch* der in der Konfiguration verwendeten Erweiterungsmodule.

---

# Kapitel 6

## Programmierung

---

### Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Abschnitte:

| Abschnitt | Thema  | Seite |
|-----------|--|-------|
| 6.1       | Übersicht über den Arbeitsbereich der Programmierung | 76    |
| 6.2       | Sonderfunktionen                                     | 78    |
| 6.3       | Konfigurieren des Programmverhaltens und der Tasks   | 90    |
| 6.4       | Verwalten von POUs                                   | 98    |
| 6.5       | Benutzerdefinierte Funktionen                        | 113   |
| 6.6       | Benutzerdefinierte Funktionsbausteine                | 123   |
| 6.7       | Master-Task  | 131   |
| 6.8       | Zeichenketten  | 135   |
| 6.9       | Periodische Task                                     | 142   |
| 6.10      | Ereignis-Task  | 146   |
| 6.11      | Verwenden von Tools                                  | 153   |
| 6.12      | Programmieren in Kontaktplan                         | 185   |
| 6.13      | Programmieren in Anweisungsliste                     | 209   |
| 6.14      | Programmierung mit Grafcet (Liste)                   | 221   |
| 6.15      | Grafcet (SFC)Programmierung                          | 230   |
| 6.16      | Fehlerbehebung im Online-Modus                       | 245   |

# Abschnitt 6.1

## Übersicht über den Arbeitsbereich der Programmierung

### Überblick des Arbeitsbereichs ‚Programmierung‘

#### Übersicht

Die Registerkarte **Programmierung** ist in 3 Hauptbereiche untergliedert:

The screenshot shows the Schneider Electric SoMachine Basic software interface. The top bar includes the project name 'Neues Projekt', the title 'Schneider Electric SoMachine Basic', and various status indicators like 'COM1', 'Keine Fehler', and 'Nicht verbunden'. Below the top bar are five tabs: 'Eigenschaften', 'Konfiguration', 'Programmierung' (active), 'Anzeige', and 'Inbetriebnahme'. The 'Programmierung' tab is divided into three main sections:

- 1. Tasks/Tools (Left):** A sidebar containing a list of tasks and tools such as 'Meldungen', 'Animationstabellen', 'Speicherobjekte', 'Systemobjekte', 'E/A-Objekte', 'Netzwerkobjekte', 'Softwareobjekte', 'Timers', 'Zähler', 'Meldungen', 'LIFO/FIFO-Register', 'Drums', 'Shift Bit Register', 'Schrittzähler', and 'Zeitplan-Bausteine'. A red number '1' points to this sidebar.
- 2. Main Programming Workspace (Center):** The central area for creating and editing logic. It shows a ladder logic diagram with a 'Rung0' (rung) containing a normally open contact labeled '%S6' connected to a coil labeled 'REGISTER\_1 %SBR0'. The workspace includes a toolbar at the top with various editing tools and a 'Sicherheit' (Safety) button. A red number '2' points to this workspace.
- 3. Properties Table (Bottom):** A table titled 'Eigenschaften von Shift Bit Register' (Properties of Shift Bit Register). It shows the current configuration for the selected register:
 

| Verwendet                           | Adresse | Symbol     | Kommentar |
|-------------------------------------|---------|------------|-----------|
| <input checked="" type="checkbox"/> | %SBR0   | REGISTER_1 |           |
| <input type="checkbox"/>            | %SBR1   |            |           |
| <input type="checkbox"/>            | %SBR2   |            |           |

 The table also shows 'Zuweisung: Automatisch', 'Anzahl Objekte: 0', and 'Zugewiesen: 1, Max.: 8'. A red number '3' points to this table.

- 1 Die Programmierübersicht ermöglicht Ihnen die Auswahl der Eigenschaften des Programms und der zugehörigen Objekte, der Funktionen sowie einiger Tools, die zur Überwachung und zum Debugging des Programms verwendet werden können.
- 2 Der obere mittlere Bereich ist der Arbeitsbereich der Programmierung, in dem Sie den Quellcode Ihres Programms eingeben..
- 3 Der untere mittlere Bereich ermöglicht die Anzeige und Konfiguration der Eigenschaften des im Arbeitsbereich der Programmierung oder in der Programmierübersicht ausgewählten Elements.

## Abschnitt 6.2

### Sonderfunktionen

---

#### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                                     | Seite |
|---|-------|
| Objekte                                   | 79    |
| Symbolische Adressierung                  | 80    |
| Speicherzuweisung                         | 83    |
| Umkehrbarkeit Kontaktplan/Anweisungsliste | 84    |

## Objekte

### Überblick

In SoMachine Basic wird der Begriff *Objekt* verwendet, um einen Speicherbereich der logischen Steuerung darzustellen, der der Nutzung durch eine Anwendung vorbehalten ist. Objekte können Folgendes sein:

- Einfache Softwarevariablen, wie z. B. Speicherbits und -wörter
- Die Adressen der digitalen oder analogen Ein- und Ausgänge
- Steuerungsinterne Variablen, wie z. B. Systemwörter und -bits
- Vordefinierte Systemfunktionen oder Funktionsbausteine, wie z. B. Timer und Zähler

Der Steuerungsspeicher wurde entweder vorab bestimmten Objekttypen zugeordnet oder die Zuordnung erfolgt automatisch, wenn die Anwendung in die Steuerung heruntergeladen wird.

Die Objekte können erst dann von einem Programm adressiert werden, wenn der Speicher zugeordnet wurde. Die Adressierung der Objekte erfolgt über das Präfix `%`. Beispiel: `%MW12` ist die Adresse eines Speicherworts, `%Q0.3` die Adresse eines integrierten Digitalausgangs und `%TM0` die Adresse eines `Timer`-Funktionsbausteins.

## Symbolische Adressierung

### Einführung

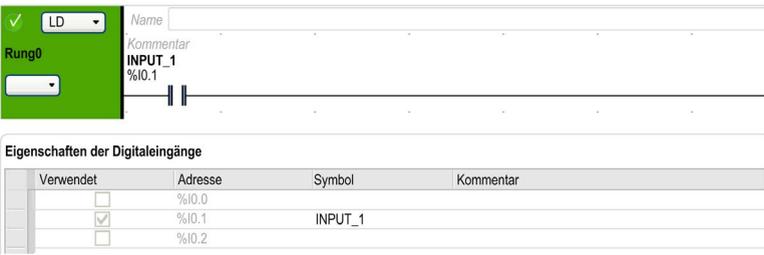
SoMachine Basic unterstützt die symbolische Adressierung von Sprachobjekten, d. h. die indirekte Adressierung von Objekten nach Namen. Die Verwendung von Symbolen ermöglicht eine schnellere Prüfung und Analyse der Programmlogik. Entwicklung und Test einer Anwendung gestalten sich dadurch wesentlich einfacher.

### Beispiel

Beispielsweise ist `WASH_END` ein Symbol, das zum Erkennen einer Instanz eines `Timer`-Funktionsbausteins verwendet werden kann, welcher das Ende eines Waschzyklus darstellt. Es ist einfacher, sich anhand eines Namens an den Zweck zu erinnern als zu versuchen, sich die Funktion einer Programmadresse wie `%TM3` einzuprägen.

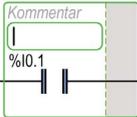
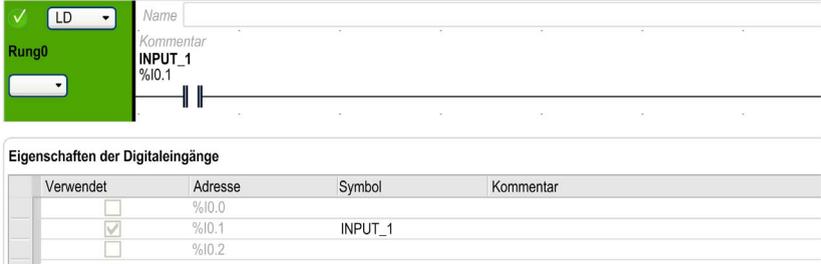
### Definieren eines Symbols im Fenster „Eigenschaften“

Gehen Sie zur Definition eines Symbols im Eigenschaftsfenster vor wie folgt:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tools</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Wählen Sie den Typ des Objekts aus, für das Sie ein Symbol definieren möchten, z. B. <b>E/A-Objekte → Digitaleingänge</b> , um die Eigenschaften der Digitaleingänge anzuzeigen. Das Eigenschaftsfenster des Objekttyps wird im unteren mittleren Bereich des Fensters <b>Programmierung</b> angezeigt.   |
| 3       | Doppelklicken Sie in die Spalte <b>Symbol</b> der Eigenschaftstabelle und geben Sie das für ein bestimmtes Element zu definierende Symbol ein, z. B. <code>Input_1</code> für den Eingang <code>%I0.2</code> .<br><br> |
| 4       | Klicken Sie auf <b>Übernehmen</b> .   |

## Definieren eines Symbols im Kontaktplan-Editor

Gehen Sie zur Definition eines Symbols im Kontaktplan-Editor vor wie folgt:

| Schritt | Aktion  |
|---------|---|
| 1       | <p>Klicken Sie im Kontaktplan-Editor auf die Zeile <b>Symbol</b> eines Grafikelements, z. B. eines Statusspeicherungsingangs oder Funktionsbausteins. Daraufhin wird ein Cursor angezeigt:</p>   |
| 2       | <p>Geben Sie das zu verwendende Symbol ein, beispielsweise <code>Input_1</code>, und drücken Sie dann die <b>Eingabetaste</b>. Für Symbole gelten folgende Regeln:</p> <ul style="list-style-type: none"> <li>• Maximal 32 Zeichen.</li> <li>• Buchstaben (A-Z), Ziffern (0 -9) und Unterstreichungen (_).</li> <li>• Das erste Zeichen muss ein Buchstabe sein. Das Prozentzeichen (%) darf nicht verwendet werden.</li> <li>• Bei Symbolen wird nicht zwischen Groß- und Kleinschreibung unterschieden. Beispiel: <code>Pump1</code> und <code>PUMP1</code> entsprechen demselben Symbol und können nur einmal für ein bestimmtes Objekt verwendet werden, d. h. Sie können dasselbe Symbol nicht verschiedenen Objekten zuweisen.</li> </ul> |
| 3       | <p>Wenn dem Grafikelement noch kein Objekt zugeordnet wurde, wird das Fenster <b>Anmerkung</b> angezeigt. Wählen Sie das dem Symbol zuzuordnende Objekt aus und klicken Sie auf <b>OK</b>. Andernfalls klicken Sie auf <b>Ja</b>, wenn Sie zur Bestätigung der Zuordnung des Symbols zu dem Objekt aufgefordert werden.</p>   |
| 4       | <p>Doppelklicken Sie entweder auf das Symbol oder auf das Objekt des Grafikelements, um das Symbol in der Spalte <b>Symbol</b> im Eigenschaftsfenster anzuzeigen.</p>    |

## Anzeigen aller definierten Symbole

Wählen Sie **Tools** → **Symbolliste** aus, um eine Liste aller definierten Symbole anzuzeigen (*siehe Seite 178*).

### Speichern von Symbolen

Symbole sind keine Programmdateien. Symbole werden in der Steuerung als Teil einer SoMachine Basic-Anwendung gespeichert.

## Speicherzuweisung

### Einführung

SoMachine Basic ermöglicht Ihnen die vorzeitige Zuweisung (Reservierung) von Speicherblöcken in der Steuerung zur Verwendung durch bestimmte, in einem Programm verwendete Objekttypen, einschließlich einfacher Objekte (Speicherwörter, Konstantwörter) und Softwareobjekte (Funktionsbausteine).

### Zuweisungsmodi

Im Offline-Modus können Sie den Speicherzuweisungsmodus für jeden Objekttyp festlegen. Bei der Konfiguration dieser Objekte (**Programmierung** → **Tools**) wird dann über der Liste der konfigurierbaren Objekte folgendes Fenster angezeigt:

Zuweisung  Anzahl Objekte  Zugewiesen: 1, Verfügbar: 1024

Wählen Sie den zu verwendenden Speicherzuweisungsmodus aus:

- Automatisch:** Alle Objekte ab dem Offset 0 bis zur höchsten im Programm verwendeten Speicheradresse bzw. alle einem Symbol zugeordneten Objekte werden automatisch im SPS-Speicher zugewiesen. Beispiel: Wenn das Speicherwort  $\%MW20$  im Programm verwendet wird, werden alle Objekte von  $\%MW0$  bis einschließlich  $\%MW20$  (21 Objekte) automatisch im Speicher zugewiesen.  
 Wenn Sie später in den Online-Modus umschalten, können Sie keine neuen Speicherobjekte zuweisen, deren Adresse die höchste, vor dem Wechsel in den Online-Betrieb verwendete Adresse überschreitet.
- Manuell:** Geben Sie im Feld **Anzahl Objekte** eine Anzahl an Objekten an, die im Speicher zugewiesen werden sollen. Wenn Sie dann in den Online-Modus umschalten, können Sie neue Kontakte, Spulen oder Gleichungen in Ihrem Programm hinzufügen (bis zur Kapazitätsgrenze des zugewiesenen Speichers), ohne sich von der Steuerung abmelden, das Programm ändern, sich wieder anmelden und die Anwendung erneut herunterladen zu müssen.  
 SoMachine Basic zeigt die Anzahl der von Ihnen angegebenen Objekte an.

SoMachine Basic zeigt die Gesamtanzahl der **Zugewiesenen** Speicherobjekte sowie die Anzahl der in der Steuerung **Verfügbaren** Speicherobjekte an.

Wenn Sie die Anzahl der Objekte festgelegt haben, dann werden nur diese Objekte in der Tabelle angezeigt.

Für die Verwendung der Anweisungen mit mehreren Operanden sind 20  $\%MW$  erforderlich sowie zusätzliche 20  $\%MW$  bei Verwendung der periodischen.

## Umkehrbarkeit Kontaktplan/Anweisungsliste

### Einführung

SoMachine Basic unterstützt die Konvertierung von Programmbausteine von Kontaktplan in Anweisungsliste sowie zurück von Anweisungsliste in Kontaktplan. Dies wird als *Programmumkehrbarkeit* bezeichnet.

In SoMachine Basic können Sie die Programmbausteine jederzeit nach Bedarf zwischen den Programmiersprachen umschalten. Sie können somit ein Programm mit ein paar Programmbausteinen in Kontaktplan und ein paar anderen Programmbausteinen in Anweisungsliste anzeigen.

**HINWEIS:** Sie können Kontaktplan- oder Anweisungslistenprogramme nicht zu Grafcet (SFC) konvertieren, oder Grafcet (SFC)-Programme zu Kontaktplan- oder Anweisungslisten konvertieren, oder Grafcet (IL) zu Grafcet (SFC) konvertieren.

### Grundlagen der Umkehrbarkeit

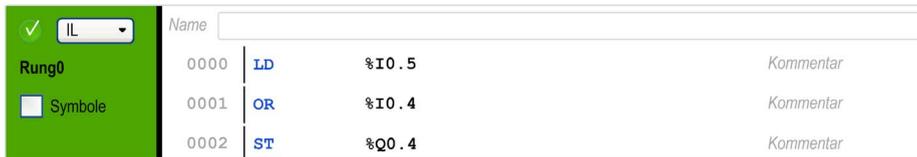
Wesentlich für das Verständnis der Umkehrbarkeit von Programmen ist das Verständnis der Beziehung zwischen einem Programmbaustein in Kontaktplan und dem entsprechenden Programmbaustein in Anweisungsliste:

- **Programmbaustein in Kontaktplan:** Folge von KOP-Anweisungen, die einen logischen Ausdruck ergeben.
- **Anweisungsfolge:** Folge von AWL-Programmieranweisungen, die den KOP-Anweisungen entsprechen und denselben logischen Ausdruck darstellen.

Die folgende Abbildung zeigt einen herkömmlichen Programmbaustein in Kontaktplan und die entsprechende Programmlogik in einer Folge von Anweisungen in Anweisungsliste.



Entsprechende AWL- (IL-)Anweisung:



Ein Programm wird intern stets in Form von AWL-Anweisungen gespeichert, ungeachtet dessen, ob es ursprünglich in Kontaktplan oder Anweisungsliste geschrieben wurde. SoMachine Basic macht sich die Ähnlichkeit der Programmstruktur zwischen den 2 Sprachen zu Nutze und verwendet dieses interne AWL-Abbild des Programms zur Programmanzeige entweder als Anweisungsfolge oder in grafischer Form als Kontaktplan-Diagramm.

### Für die Umkehrbarkeit erforderliche Anweisungen

Die Struktur eines umkehrbaren Funktionsbausteins in Anweisungsliste erfordert die Verwendung folgender Anweisungen:

- **BLK** markiert den Beginn des Blocks und definiert den Beginn des Programmblocks und den Beginn des Eingangsteils des Block.
- **OUT\_BLK** markiert den Beginn des Ausgangsteils des Blocks.
- **END\_BLK** markiert das Ende des Blocks und des Programmblocks.

Die Verwendung dieser umkehrbaren Funktionsbausteinanweisungen ist für ein ordnungsgemäß funktionierendes AWL-Programm nicht obligatorisch.

### Programmierungsprobleme und IL/Ladder-Umkehrbarkeit

In der folgenden Tabelle werden Programmierungsprobleme in den Sprachen Ladder und IL aufgeführt, die, falls sie nicht behoben werden, zu Hinweis- oder Fehlermeldungen führen können. Auch die Umkehrbarkeit kann verloren gehen.

| Situation  | AWL   | Ladder | Umkehrbar (Programmbaustein) |
|--|-------|--------|------------------------------|
| Zu einer nicht definierten Marke springen                      | Error | Error  | Ja                           |
| Nicht definierte Subroutine aufrufen                           | Error | Error  | Ja                           |
| Nicht definierten Grafset-Schritt aktivieren oder deaktivieren | Error | Error  | Ja                           |
| Sprunganweisung in Klammern                                    | Error | -      | Keine                        |
| Marke in Klammern  | Error | -      | Keine                        |
| Subroutine in Klammern   | Error | -      | Keine                        |
| Über 32 verschachtelte Klammern                                | Error | -      | Keine                        |
| Öffnende Klammer ohne schließende Klammer                      | Error | -      | Keine                        |
| Reserviert   | -     | -      | -                            |

| Situation  | AWL   | Ladder | Umkehrbar<br>(Programmbaustein) |
|--|-------|--------|---------------------------------|
| Nicht symmetrische Klammern                                  | Error | -      | Keine                           |
| BLK ohne END_BLK   | Error | -      | Keine                           |
| OUT_BLK oder END_BLK ohne BLK                                | Error | -      | Keine                           |
| Markendefinition ohne LD oder BLK                            | Error | -      | Keine                           |
| Subroutinendefinition ohne LD oder BLK                       | Error | -      | Keine                           |
| Reserviert   | -     | -      | -                               |
| Über 11 verschachtelte MPS                                   | Error | -      | Keine                           |
| MRD ohne MPS   | Error | -      | Keine                           |
| MPP ohne MPS   | Error | -      | Keine                           |
| Grafcet-Anweisung in POST verwenden                          | Error | Error  | Ja                              |
| Grafcet-Definition ohne BLK oder LD                          | Error | -      | Keine                           |
| Nicht symmetrische Stapelvorgänge                            | Error | -      | Keine                           |
| Reserviert   | -     | -      | -                               |
| Doppelte Marke   | Error | Error  | Nur LD->IL                      |
| Doppelte Subroutine  | Error | Error  | Nur LD->IL                      |
| Doppelter Grafcet-Schritt                                    | Error | Error  | Nur LD->IL                      |
| Reserviert   | -     | -      | -                               |
| Doppeltes POST   | Error | Error  | Nur LD->IL                      |
| Verschachtelter FB   | Error | -      | Keine                           |
| OUT_BLK zwischen BLK und END_BLK                             | Error | -      | Keine                           |
| BLK nicht gefolgt von LD                                     | Error | -      | Keine                           |
| LD von FB-Ausgang nicht in OUT_BLK                           | Error | -      | Keine                           |
| FB-Ausgänge werden außerhalb ihrer FB-Struktur verwendet     | Error | -      | Keine                           |
| FB-Ausgänge werden nicht in richtiger Reihenfolge wiederholt | Error | -      | Keine                           |
| FB-Eingänge nicht in BLK vor OUT_BLK                         | Error | -      | Keine                           |
| FB-Eingänge werden außerhalb ihrer FB-Struktur verwendet     | Error | -      | Keine                           |
| FB-Eingänge werden nicht in richtiger Reihenfolge wiederholt | Error | -      | Keine                           |
| Marke in BLK deklariert                                      | Error | -      | Keine                           |
| Subroutinen in BLK deklariert                                | Error | -      | Keine                           |

| Situation   | AWL     | Ladder | Umkehrbar<br>(Programmbaustein) |
|---|---------|--------|---------------------------------|
| Grafcet-Schritte in <code>BLK</code> deklariert   | Error   | -      | Keine                           |
| <code>LD</code> eines Nicht-FB-Ausgangs in <code>OUT_BLK</code> (Versuch)   | Error   | -      | Keine                           |
| FB-Ausgang verwendet zwischen <code>BLK</code> und <code>END_BLK</code>   | Error   | -      | Keine                           |
| Verschachtelte Subroutinen  | Error   | Error  | Keine                           |
| Subroutinen-Aufruf zwischen <code>MPS</code> und <code>MPP</code>   | Error   | Error  | Keine                           |
| Subroutinen-Aufruf in Klammern  | Error   | -      | Keine                           |
| Reserviert  | -       | -      |                                 |
| Erste Anweisung des Programms kein Programmbaustein-Trennzeichen  | Error   | -      | Keine                           |
| Sprunganweisung zwischen <code>MPS</code> und <code>MPP</code>  | Error   | Error  | Keine                           |
| Programmbaustein enthält Syntaxfehler   | Error   | -      | Keine                           |
| Reserviert  | -       | -      | -                               |
| Reserviert  | -       | -      | -                               |
| Programmanweisungen gemäß unbedingter Anweisung ( <code>JMP</code> oder <code>END</code> )                                | Error   | -      | Keine                           |
| Programmbaustein beginnt mit <code>LD</code> -Anweisung und endet nicht mit bedingter Aktionsanweisung                    | Hinweis | -      | Keine                           |
| Aktionsanweisung in Klammern  | Error   | -      | Keine                           |
| Stapelanweisung in Klammern   | Error   | -      | Keine                           |
| Direktzugriffsanweisungen für FB (z. B. " <code>CU %C0</code> ")  | Hinweis | -      | Keine                           |
| Aktionsanweisungen im Eingangsbereich eines FB  | Error   | -      | Keine                           |
| Anweisungen nach <code>END_BLK</code>   | Error   | -      | Keine                           |
| FB-Ausgänge verwendet zwischen <code>AND</code> - und <code>OR</code> -Anweisungen  | Hinweis | -      | Keine                           |
| <code>OR</code> -Anweisung in einem FB-Ausgang nicht in Klammern  | Hinweis | -      | Keine                           |
| Anweisung vor <code>MRD</code> oder <code>MPP</code> ist keine bedingte Aktion oder nicht mit Stapelanweisungen verbunden | Hinweis | -      | Keine                           |
| Unverschachteltes <code>OR</code> zwischen <code>MPS</code> und <code>MPP</code>  | Hinweis | -      | Keine                           |
| <code>OR</code> nach einer Aktionsanweisung   | Hinweis | -      | Keine                           |
| <code>OR</code> nach <code>MPS</code> , <code>MRD</code> oder <code>MPP</code>  | Hinweis | -      | Keine                           |
| Reserviert  | -       | -      |                                 |
| Subroutinen-Aufruf oder <code>JMPC</code> nicht die letzte Aktionsanweisung des Programmbausteins                         | Hinweis | Error  | Keine                           |

| Situation   | AWL     | Ladder | Umkehrbar<br>(Programmbaustein) |
|---|---------|--------|---------------------------------|
| Kanonischer Programmbaustein überschreitet 7x11 Zellen in Twido, 256 x 30 Zellen in SoMachine Basic                 | Hinweis | -      | Keine                           |
| Unbedingte Aktionsanweisung zwischen BLK und END_BLK  | Error   | -      | Keine                           |
| OUT_BLK nicht gefolgt von LD eines gültigen FB-Ausgangs oder END_BLK  | Error   | -      | Keine                           |
| FB kann nicht die erste Zelle einnehmen   | -       | -      | Ja                              |
| FB ganz oben im Programmbaustein. Ersetzt die Elemente in den Zellen  | -       | -      | Ja                              |
| Keine Logik über oder unter einem FB  | -       | Error  | Keine                           |
| XOR in erster Spalte  | -       | Error  | Keine                           |
| Kontakte und horizontale Verbindungen in letzter Spalte   | -       | Error  | Keine                           |
| Abwärtsverbindungen in letzter Zeile oder Spalte  | -       | Error  | Keine                           |
| Nur Subroutinen 0 bis 63 zulassen   | -       | Error  | Keine                           |
| Nur gültige Marken 0 bis 63 zulassen  | -       | Error  | Keine                           |
| Ungültige Operatorenausdrücke in Operatorenbaustein   | -       | Error  | Keine                           |
| Ungültige Vergleichsausdrücke in Vergleichsbaustein   | -       | Error  | Keine                           |
| Ungültige Adresse oder ungültiges Symbol in Kontakt und Spule   | -       | Error  | Keine                           |
| Ungültiger Operand oder Ausdruck in Ladder-Anweisung  | -       | Error  | Keine                           |
| Programmbaustein ohne Ausgangsaktionselement  | -       | Error  | Keine                           |
| Unterbrechung zwischen linker und rechten Stromleiste   | -       | Error  | Keine                           |
| Ladder-Programmbaustein unvollständig   | -       | Error  | Keine                           |
| Ladder-Programmbaustein enthält kurzgeschlossene Elemente   | -       | Error  | Keine                           |
| Alle Verzweigungen, die nur boolesche Logikelemente enthalten, müssen in umgekehrter Reihenfolge konvergiert werden | -       | Error  | Keine                           |
| FB hat keinen verbundenen Eingang   | -       | Error  | Keine                           |
| FB-Ausgangspole können nicht miteinander verbunden werden   | -       | Error  | Keine                           |
| XOR mit Stromleiste verbunden   | -       | Error  | Keine                           |

| Situation  | AWL     | Ladder | Umkehrbar<br>(Programmbaustein) |
|--|---------|--------|---------------------------------|
| Subroutinen-Aufruf und -Sprung nicht das letzte Ausgangsaktionselement   | Hinweis | Error  | Keine                           |
| Kanonischer Programmbaustein, der einen FB enthält, wobei ein Teil des FB in der letzten Spalte gespeichert ist  | -       | -      | Keine                           |
| Kanonischer Programmbaustein überschreitet 7x11 Zellen in Twido, 256 x 30 Zellen in SoMachine Basic              | Hinweis | Error  | Keine                           |
| OPEN und SHORT mit dem linken Knoten des Subnetzwerks verbunden  | -       | Error  | Keine                           |
| XOR mit dem linken Knoten des Subnetzwerks verbunden   | -       | Error  | Keine                           |
| Es ist nicht mindestens ein vorhandener LIST-Satz vorhanden, der den Ladder-Programmbaustein repräsentieren kann | -       | Error  | Keine                           |

## Abschnitt 6.3

### Konfigurieren des Programmverhaltens und der Tasks

---

#### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                   | Seite |
|-------------------------|-------|
| Verhalten der Anwendung | 91    |
| Tasks und Abfragemodi   | 95    |

## Verhalten der Anwendung

### Übersicht

Sie können folgende Aspekte der Interaktion zwischen Anwendung und Steuerung definieren:

- **Start** (siehe Seite 91)
- **Watchdog** (siehe Seite 93)
- **Fehlerabweichverhalten** (siehe Seite 93)
- **Funktionsebenen** (siehe Seite 94)

### Konfigurieren des Verhaltens der Anwendung

Gehen Sie wie folgt vor, um das Verhalten der Anwendung zu konfigurieren:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Wählen Sie das Element <b>Verhalten</b> aus.<br><b>Ergebnis:</b> Im unteren mittleren Bereich des Fensters <b>Programmierung</b> werden daraufhin die Eigenschaften in Verbindung mit dem <b>Verhalten</b> angezeigt. |
| 3       | Nehmen Sie an den Eigenschaften die gewünschten Änderungen vor.   |
| 4       | Klicken Sie auf <b>Übernehmen</b> , um die Änderungen zu speichern.   |

### Start

Geben Sie an, wie sich das Programm im Anschluss an einen Neustart der Steuerung zu verhalten hat:

- **In vorherigem Zustand starten:** Die Steuerung startet in dem Zustand, in dem sich vor dem Stopp befunden hat.
- **Start in STOP:** Die Steuerung startet die Anwendungsausführung nicht automatisch.
- **Start in RUN (Standard):** Die Steuerung startet die Anwendungsausführung automatisch, sobald bestimmte Kriterien erfüllt sind, wie z. B. eine vorhandene und geladene Batterie.
- **Unbedingter Start in RUN:** Die Steuerung startet die Anwendungsausführung automatisch selbst dann, wenn keine SPS-Batterie vorhanden oder diese nicht geladen ist.

Bei Verwendung der Funktion „Start in RUN“ beginnt die Steuerung mit der Ausführung der Programmlogik, sobald dem Gerät Strom zugeführt wird. Es ist unbedingt erforderlich, im Voraus zu wissen, wie die automatische Reaktivierung der Ausgänge den gesteuerten Prozess bzw. die gesteuerte Maschine beeinflussen wird. Konfigurieren Sie den Run/Stop-Eingang, um die Funktion „Start in RUN“ besser kontrollieren zu können. Darüber hinaus ermöglicht der Run/Stop-Eingang eine lokale Kontrolle über Remote-RUN-Befehle. Wenn ein Remote-RUN-Befehl nach dem lokalen Stopp der Steuerung über SoMachine unbeabsichtigte Folgen mit sich bringen würde, müssen Sie den Run/Stop-Eingang konfigurieren und verdrahten, um diese Situation kontrollieren zu können.

## **WARNUNG**

### **UNBEABSICHTIGTER MASCHINENSTART**

- Stellen Sie sicher, dass das automatische Wiedereinschalten der Ausgänge keine unbeabsichtigten Folgen hat, bevor Sie die Einstellung „Start in RUN“ verwenden.
- Verwenden Sie den Run/Stop-Eingang, um die Funktion „Start in RUN“ besser kontrollieren zu können und einen unbeabsichtigten Systemstart von einem dezentralen Ort aus zu verhindern.
- Überprüfen Sie den Sicherheitszustand der Maschinen- oder Prozessumgebung, bevor Sie den Run/Stop-Eingang unter Spannung setzen oder einen Run-Befehl von einem dezentralen Ort aus erteilen.

**Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.**

## **WARNUNG**

### **UNBEABSICHTIGTER MASCHINEN- ODER PROZESSSTART**

- Überprüfen Sie den Sicherheitsstatus Ihrer Maschinen- bzw. Prozessumgebung, bevor Sie den Run/Stop-Eingang unter Spannung setzen.
- Verwenden Sie den Run/Stop-Eingang, um den unbeabsichtigten Start von einem entfernten Standort aus zu verhindern.

**Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.**

Bei Verwendung der Funktion „Unbedingter Start in RUN“ versucht die Steuerung, die Ausführung der Programmlogik zu starten, sobald das Gerät unter Spannung gesetzt wird, ungeachtet der Ursache für den vorhergehenden Stopp der Steuerung. Das ist selbst dann der Fall, wenn die Batterie nicht geladen ist oder fehlt. Aus diesem Grund startet die Steuerung mit auf Null neu initialisierten Speicherwerten bzw. mit anderen vordefinierten Standardwerten. Es ist durchaus vorstellbar, dass beim Neustart der Steuerung im Anschluss an eine kurze Spannungsunterbrechung beispielsweise die Werte, die zum Zeitpunkt des Spannungsausfalls im Speicher enthalten waren, verloren gegangen sind und der Neustart der Maschine u. U. unerwartete Folgen mit sich bringt, da keine Batterie zur Sicherung der Speicherwerte vorhanden war. Es muss unbedingt im Voraus bekannt sein, wie sich ein unbedingter Start auf den gesteuerten Prozess bzw. die gesteuerte Maschine auswirkt. Konfigurieren Sie den Run/Stop-Eingang, um die Funktion „Unbedingter Start in RUN“ besser kontrollieren zu können.

## ⚠️ WARNUNG

### UNBEABSICHTIGTER MASCHINENBETRIEB

- Führen Sie eine umfassende Risikoanalyse durch, um die Auswirkungen einer Konfiguration der Steuerung mit der Funktion „Unbedingter Start in RUN“ unter allen möglichen Bedingungen zu erfassen.
- Verwenden Sie den Run/Stop-Eingang, um einen unerwünschten unbedingten Neustart zu vermeiden.

**Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.**

### Watchdog

Ein Watchdog ist ein spezieller Zeitgeber (Timer), der gewährleistet, dass Programme nicht die ihnen zugewiesene Zykluszeit überschreiten.

Der Watchdog-Timer hat einen Standardwert von 250 ms. Geben Sie die Dauer der Abfrage-Task des Watchdogs ein. Gültiger Wertebereich: 10 bis 500 ms.

### Fehlerausweichverhalten

Geben Sie das Fehlerausweichverhalten an, das verwendet werden soll, wenn die Steuerung aus einem beliebigen Grund in den Zustand `STOPPED` (Angehalten) oder in einen Ausnahmestatus übergeht.

Zwei Fehlerausweichverhalten stehen zur Auswahl:

- Wählen Sie die **Fehlerausweichwerte** aus, so dass die Ausgänge auf die Fehlerausweichwerte eingestellt werden, die in den Konfigurationseigenschaften der integrierten Logiksteuerung und Erweiterungsmodulausgängen definiert sind. Dies ist die Standardeinstellung. Informationen zur Konfiguration der Fehlerausweichwerte für Ausgänge finden Sie im *Programmierhandbuch* der Steuerung und der Erweiterungsmodule. Für konfigurierte Statusalarm-, PTO- und FREQGEN-Ausgänge können keine individuellen Fehlerausweichwerte definiert werden. Der Fehlerausweichwert für diese Objekte ist 0 und kann nicht geändert werden.
- Wählen Sie **Werte beibehalten** aus, wenn jeder Ausgang den Status beibehalten soll, der aktuell war, wenn die Steuerung in den Zustand `STOPPED` (Angehalten) oder in einen Ausnahmestatus überging. In diesem Modus werden die für die Logiksteuerungs- und Erweiterungsmodulausgänge konfigurierten Fehlerausweichwerte ignoriert, und stattdessen werden die Werte auf den letzten Wert des Ausganges eingestellt. Das Verhalten **Werte beibehalten** wird nicht auf schnelle Ausgänge (HSC-Reflexausgänge, PLS, PWM, PTO und FREQGEN) angewandt. Der Fehlerausweichwert für diese Objekte ist 0.

## Funktionsebenen

Ihr System umfasst möglicherweise Logiksteuerungen mit verschiedenen Firmware-Versionen und somit mit unterschiedlichen Kapazitätsniveaus. SoMachine Basic unterstützt die Verwaltung von Funktionsebenen, sodass Sie die Funktionsebene Ihrer Anwendung steuern können.

Wenn SoMachine Basic eine Verbindung zur Steuerung herstellt, liest die Software die Funktionsebene von:

- Der Firmware der Logiksteuerung, um den Download der SoMachine Basic-Anwendung in die Steuerung zu autorisieren. Die für die Anwendung ausgewählte Funktionsebene darf höchstens der maximalen von der Steuerung unterstützten Funktionsebene entsprechen. Ist dies nicht der Fall, wird eine Meldung angezeigt, dass Sie entweder die Firmware aktualisieren oder die Funktionsebene der Steuerung manuell herabsetzen müssen (durch Auswahl einer Ebene aus der Liste „Funktionsebenen“, siehe unten).
- Der in der Steuerung integrierten Anwendung, um festzulegen, ob ein Upload der Anwendung in der Steuerung auf den PC, auf dem SoMachine Basic ausgeführt wird, autorisiert werden soll. Um den Upload der Anwendung zu autorisieren, darf die Funktionsebene der Anwendung in der Steuerung höchstens der maximalen von der installierten Version von SoMachine Basic unterstützten Funktionsebene entsprechen. Ist dies nicht der Fall, muss SoMachine Basic vor dem Upload auf die neueste Version aktualisiert werden.

Das Fenster **Inbetriebnahme** zeigt die Funktionsebenen der SoMachine Basic-Anwendung sowie die in die verbundene Steuerung integrierte Anwendung an.

Wählen Sie in der Liste **Funktionsebenen** eine Ebene aus:

- **Ebene 6.0:** Enthält Modbus TCP IOScanner, benutzerdefinierte Funktionen, benutzerdefinierte Funktionsbausteine, Datenprotokollierung auf SD-Karte, Verwaltung von Zeichenfolgen, Strukturelemente von KOP-Bausteinen, Funktionen bei steigender und fallender Flanke.
- **Ebene 5.1:** Enthält die Änderung der Sicherheitsstrategie.
- **Ebene 5.0Modbus:** Enthält Serial-E/A-Scanner, Antriebs- und RTC-Funktionsbausteine, Anweisungen mit mehrfachen Operanden.
- **Ebene 4.1:** Enthält Online-Moduserweiterungen, Unterstützung für ein Modem auf SL2.
- **Ebene 4.0:** Enthält die Unterstützung für Transistorausgangssteuerungen (Senke), Grafcet (SFC), Frequenzgenerator, „Remanenten“ Timer, Speicherverwaltung, dezentrale grafische Anzeigenentwicklung.
- **Ebene 3.3:** Enthält Erweiterungen (PTO-Motion-Task, HSC-Entwicklung)
- **Ebene 3.2:** Enthält Erweiterungen zur Unterstützung der Funktion **Optionales Modul**, des EtherNet/IP adapters und des Funktionsbausteins %SEND\_RECV\_SMS.
- **Ebene 3.1:** Beinhaltet Erweiterungen (Funktion **Unbedingter Start in RUN**).
- **Ebene 3.0:** Beinhaltet Erweiterungen (Kommunikation, Modem, Dezentrale Grafikanzeige: Remote Graphic Display) in Bezug auf die vorhergehende Ebene der Software und Hardware.
- **Ebene 2.0:** Beinhaltet alle Erweiterungen/Verbesserungen und Korrekturen an der vorherigen Ebene der Software und Firmware. Beispielsweise müsste zur Unterstützung von Impulswellen- ausgang (Pulse Train Output, PTO) diese oder eine höhere Funktionsebene ausgewählt werden.
- **Ebene 1.0:** Erstes Release der Kombination der SoMachine Basic-Software und der kompatiblen Firmware-Version(en).

## Tasks und Abfragemodi

### Überblick

SoMachine Basic hat die folgenden Abfragemodi für die Master-Task:

- **Normaler Modus**

Wenn sich eine Logiksteuerung im zyklischen Abfragemodus (freilaufenden Modus) befindet, startet ein neuer Abfragezyklus, sobald die vorhergehende Abfrage abgeschlossen ist.

- **Periodischer Modus**

Wenn sich eine Logiksteuerung im periodischen zyklischen Abfragemodus befindet, wird eine neue Abfrage erst dann gestartet, wenn die konfigurierte Abfragezeit der vorhergehenden Abfrage abgelaufen ist. Daher ist jede Abfrage von gleicher Dauer.

SoMachine Basic bietet die folgenden Tasktypen:

- **Master-Task:** Haupt-Task der Anwendung.

Die Master-Task wird entweder durch eine kontinuierliche zyklische Abfrage (im normalen Abfragemodus) oder durch eine spezifizierte Abfrageperiode von 1 bis 150 ms (standardmäßig 100 ms) im periodischen Abfragemodus gesteuert.

- **Periodischer Task:** Ein in regelmäßigen Abständen verarbeitetes Unterprogramm von kurzer Dauer.

Periodische Tasks werden durch die Spezifizierung der Abfrageperiode zwischen 1 und 255 ms (standardmäßig 255 ms) definiert.

- **Ereignistask:** Ein Unterprogramm von sehr kurzer Dauer zur Reduzierung der Antwortzeit der Anwendung.

Ereignistasks werden über physische Eingänge oder HSC-Funktionsbausteine ausgelöst. Die Ereignisse werden integrierten Digitaleingängen (%I0.2 bis %I0.5) (steigende, fallende oder beide Flanken) oder den Hochgeschwindigkeitszählern (%HSC0 und %HSC1) zugeordnet (wenn der Zähler den Schwellenwert für den Hochgeschwindigkeitszähler erreicht). Sie können für jeden HSC-Funktionsbaustein 2 Ereignisse konfigurieren.

### Task-Prioritäten

Die folgende Tabelle bietet einen Überblick über die Tasktypen und ihre jeweiligen Prioritäten:

| Tasktyp    | Abfragemodus | Auslösebedingung | Konfigurierbarer Bereich  | Maximale Anzahl an Tasks | Priorität   |
|------------|--------------|------------------|---------------------------|--------------------------|---|
| Master     | Normal       | Normal           | Nicht zutreffend          | 1                        | Niedrigste  |
|            | Periodisch   | Software-Timer   | 1 bis 150 ms <sup>1</sup> |                          |   |
| Periodisch | Periodisch   | Software-Timer   | 1 bis 255 ms <sup>1</sup> | 1                        | Höher als die Master-Task und niedriger als Ereignistasks |

<sup>1</sup> Die Anwendung muss mit einer Funktionsebene (*siehe Seite 94*) von mindestens Ebene 5.0 konfiguriert werden, damit ein Minimalwert von 1 ms konfiguriert werden kann. Andernfalls ist der Minimalwert 2 ms.

| Tasktyp  | Abfragemodus | Auslösebedingung        | Konfigurierbarer Bereich     | Maximale Anzahl an Tasks | Priorität |
|----------|--------------|-------------------------|------------------------------|--------------------------|-----------|
| Ereignis | Periodisch   | Physische Eingänge      | %I0.2...%I0.5                | 4                        | Höchste   |
|          |              | %HSC-Funktionsbausteine | 2 Ereignisse pro %HSC-Objekt |                          |           |

<sup>1</sup> Die Anwendung muss mit einer Funktionsebene (*siehe Seite 94*) von mindestens Ebene 5.0 konfiguriert werden, damit ein Minimalwert von 1 ms konfiguriert werden kann. Andernfalls ist der Minimalwert 2 ms.

### Ereignisprioritäten

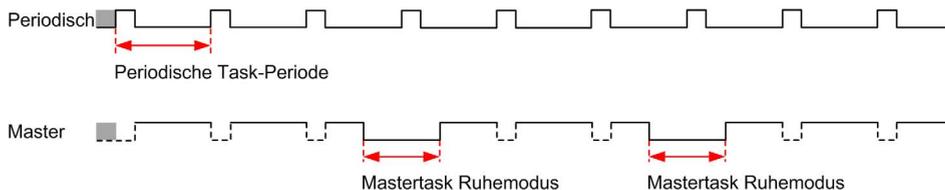
Siehe Ereignisprioritäten und -warteschlangen (*siehe Seite 149*).

### Master-Task im normalen Abfragemodus

Die Abbildung zeigt die Beziehung zwischen Master-Tasks und periodischen Tasks, wenn der Master-Task im normalen Abfragemodus konfiguriert ist:

Periodische Task: Periodischer Modus

Master-Task: Normaler Modus



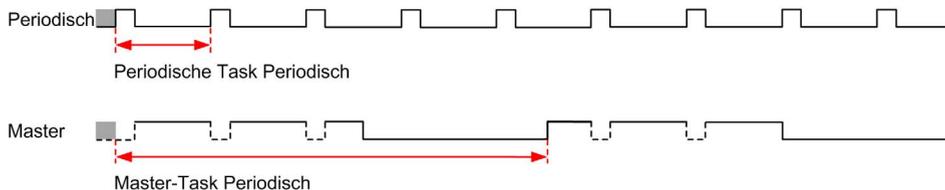
**HINWEIS:** Die Ruhezeit der Master-Task entspricht mindestens 30% der totalen Zykluszeit mit einem Minimum von einer (1) Millisekunde.

### Master-Task im periodischen Abfragemodus

Die Abbildung zeigt die Beziehung zwischen Master-Tasks und periodischen Tasks, wenn der Master-Task im periodischen Abfragemodus konfiguriert ist:

Periodische Task: Periodischer Modus

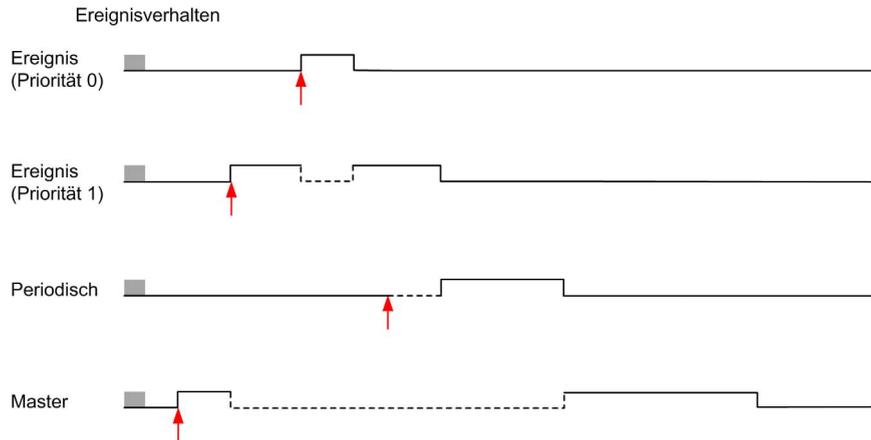
Master-Task: Periodischer Modus



### Priorität von Ereignissen vor Master- und periodischen Tasks

Die Ereignisprioritäten steuern die Beziehung zwischen den Ereignistasks, den Master-Tasks und den periodischen Tasks. Die Ereignistask unterbricht die Ausführung der Master-Task und der periodischen Task.

Diese Abbildung zeigt die Beziehung zwischen den Ereignistasks, den Master-Tasks und den periodischen Tasks im periodischen Modus:



Die Ereignistasks werden über ein Hardware-Interrupt ausgelöst, das ein Task-Ereignis an die Ereignistask sendet.

# Abschnitt 6.4

## Verwalten von POUs

---

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                            | Seite |
|----------------------------------|-------|
| POUs                             | 99    |
| Verwalten von POUs mit Tasks     | 100   |
| Verwalten von Programmbausteinen | 103   |
| Verwalten von Grafset (SFC)-POUs | 107   |
| Freie POUs                       | 109   |

## POUs

### Überblick

Eine Programmorganisationseinheit (POU: Program Organization Unit) ist ein in einem Programm verwendetes, wiederverwendbares Objekt. Jede POU besteht aus einer Variablendeklaration und einer Reihe von Anweisungen im Quellcode in einer unterstützten Programmiersprache.

Es ist immer eine POU vorhanden, die mit der Master-Task des Programms verknüpft ist. Diese POU wird dann bei jedem Programmstart automatisch aufgerufen.

Sie können weitere POU's mit anderen Objekten, z. B. Funktionen oder Funktionsbausteinen, erstellen.

Bei der ersten Erstellung einer POU sind folgende Möglichkeiten gegeben:

- Die POU kann einem Task zugeordnet (*siehe Seite 100*) werden. Oder
- Es kann sich um eine freie POU (*siehe Seite 109*) handeln. Die Free POU ist keiner spezifischen Task bzw. keinem Ereignis zugeordnet. Eine freie POU kann beispielsweise Bibliotheksfunktionen enthalten, die unabhängig vom Hauptprogramm verwaltet werden. Freie POU's werden von innerhalb eines Programms als Unterprogramme oder Sprünge aufgerufen. Eine periodische Task (*siehe Seite 143*) ist ein Unterprogramm, das als freie POU implementiert ist.

## Verwalten von POU's mit Tasks

### Hinzufügen einer neuen, einem Task zugeordneten POU

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | <p>Fügen Sie mithilfe einer der folgenden Methoden eine neue POU hinzu:</p> <ul style="list-style-type: none"> <li>• Klicken Sie mit der rechten Maustaste auf die <b>Master-Task</b> und wählen Sie im Kontextmenü die Option <b>POU hinzufügen</b> aus.</li> <li>• Wählen Sie die <b>Master-Task</b> aus und klicken Sie auf  (<b>POU hinzufügen</b>) in der Symbolleiste am oberen Rand der Registerkarte <b>Tasks</b>.</li> </ul> <p><b>Ergebnis:</b> Daraufhin wird in der Programmstruktur direkt unter der Standard- bzw. letzten POU in der <b>Master-Task</b> eine neue POU hinzugefügt. Der Standardname lautet <math>n</math> - <b>Neue POU</b>, wobei <math>n</math> einer Ganzzahl entspricht, die bei jeder Erstellung einer weiteren POU inkrementiert wird.</p> |
| 3       | Wenn eine POU in der <b>Master-Task</b> an eine andere Position verschoben werden soll, wählen Sie die jeweilige POU aus und klicken Sie in der Symbolleiste am oberen Rand des Fensters <b>Tasks</b> auf die AUFWÄRTS- oder ABWÄRTS-Pfeilschaltfläche, um die POU in der Programmstruktur nach oben oder unten zu verschieben.  |

### Einfügen einer neuen POU

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Wählen Sie eine existierende POU aus, über welcher die neue POU eingefügt werden soll.  |
| 3       | Klicken Sie mit der rechten Maustaste auf die gewählte POU und wählen Sie im daraufhin angezeigten Kontextmenü <b>POU einfügen</b> aus.   |
| 4       | Wenn eine POU in der <b>Master-Task</b> an eine andere Position verschoben werden soll, wählen Sie die jeweilige POU aus und klicken Sie in der Symbolleiste am oberen Rand des Fensters <b>Tasks</b> auf die AUFWÄRTS- oder ABWÄRTS-Pfeilschaltfläche, um die POU in der Programmstruktur nach oben oder unten zu verschieben. |

## Kopieren und Einfügen vorhandener POUs, die einer Task zugeordnet sind

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Wählen Sie eine oder mehrere POUs aus: <ul style="list-style-type: none"> <li>• Wählen Sie ein bestehende POU in der <b>Master-Task</b> aus.</li> <li>• Drücken und halten Sie die STRG-Taste gedrückt, um jede POU in der <b>Master-Task</b> auszuwählen.</li> </ul>  |
| 3       | Klicken Sie mit der rechten Maustaste auf eine der ausgewählten POUs in der <b>Master-Task</b> und wählen Sie im Kontextmenü die Option <b>POU kopieren</b> aus.   |
| 4       | Klicken Sie mit der rechten Maustaste auf die <b>Master-Task</b> und wählen Sie im Kontextmenü <b>POU einfügen</b> aus.<br><b>Ergebnis:</b> Daraufhin wird in der Programmstruktur direkt unter der ausgewählten POU in der <b>Master-Task</b> eine oder mehrere POUs mit dem gleichen Namen wie die kopierte POU hinzugefügt. |

## Exportieren von POUs oder Freien POUs

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Wählen Sie eine oder mehrere der bestehenden POUs oder Freien POUs in der <b>Master-Task</b> aus.  |
| 3       | Klicken Sie mit der rechten Maustaste auf eine ausgewählte POU oder Freie POU in der <b>Master-Task</b> und wählen Sie im erscheinenden Kontextmenü die Option <b>POU exportieren</b> aus. |
| 4       | Speichern Sie die POUs-Dateien (*.smbf) im <b>Export-Ordner</b> , der erscheint.   |

## Importieren von POUs oder Freien POUs

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Wählen Sie eine oder mehrere der bestehenden POUs oder Freien POUs in der <b>Master-Task</b> aus.  |
| 3       | Klicken Sie mit der rechten Maustaste auf eine ausgewählte POU oder Freie POU in der <b>Master-Task</b> und wählen Sie im erscheinenden Kontextmenü die Option <b>POU importieren</b> aus.   |
| 4       | Wählen Sie die POUs-Dateien (*.smbf) im Ordner, der erscheint, aus.<br><b>HINWEIS:</b> Wenn die maximale Anzahl Freier POUs erreicht wurde oder die Datei beschädigt ist (ungültiges Format), dann erscheint eine Fehlermeldung und die Freien POUs werden nicht importiert. |

## Umbenennen einer POU

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Bearbeiten Sie den POU-Namen mit einer der folgenden Methoden: <ul style="list-style-type: none"> <li>● Klicken Sie mit der rechten Maustaste auf eine POU und wählen Sie im daraufhin angezeigten Kontextmenü <b>POU umbenennen</b> aus.</li> <li>● Doppelklicken Sie auf eine POU.</li> <li>● Wählen Sie eine POU aus und doppelklicken Sie im Arbeitsbereich der Programmierung auf den POU-Namen.</li> <li>● Wählen Sie eine POU aus und drücken Sie die F2-Taste.</li> </ul> |
| 3       | Geben Sie einen neuen Namen für die POU ein und drücken Sie die Eingabetaste (ENTER).   |

## Entfernen von POUs

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Wählen Sie eine oder mehrere POUs aus: <ul style="list-style-type: none"> <li>● Wählen Sie ein bestehende POU in der <b>Master-Task</b> aus.</li> <li>● Drücken und halten Sie die STRG-Taste gedrückt, um jede POU in der <b>Master-Task</b> auszuwählen.</li> </ul>               |
| 3       | Löschen der ausgewählten POUs: <ul style="list-style-type: none"> <li>● Klicken Sie mit der rechten Maustaste auf eine ausgewählte POU in der <b>Master-Task</b> und wählen Sie im Kontextmenü die Option <b>POU löschen</b> aus.</li> <li>● Drücken Sie die Taste Entf.</li> </ul> |

## Verwalten von Programmbausteinen

### Erstellen eines Programmbausteins

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | <p>Fügen Sie mithilfe einer der folgenden Methoden einen Programmbaustein in einer POU hinzu:</p> <ul style="list-style-type: none"> <li>• Klicken Sie mit der rechten Maustaste auf eine POU und wählen Sie im daraufhin angezeigten Kontextmenü <b>Programmbaustein hinzufügen</b> aus.</li> <li>• Wählen Sie eine POU aus und klicken Sie auf  (Schaltfläche <b>Programmbaustein hinzufügen</b>) in der Symbolleiste am oberen Rand der Registerkarte <b>Tasks</b>.</li> <li>• Wählen Sie eine POU aus und klicken Sie auf  (Schaltfläche <b>Neuen Programmbaustein erstellen</b>) in der Symbolleiste am oberen Rand des Arbeitsbereichs für die Programmierung.</li> </ul> <p><b>Ergebnis:</b> In der Programmstruktur wird direkt unter dem letzten Programmbaustein ein neuer Programmbaustein hinzugefügt.</p>                       |
| 3       | Um einen Programmbaustein in einer POU an eine andere Position zu verschieben, wählen Sie den jeweiligen Programmbaustein aus und klicken Sie in der Symbolleiste am oberen Rand der Registerkarte <b>Tasks</b> auf die AUFWÄRTS- oder ABWÄRTS-Pfeilschaltfläche, um den Programmbaustein in der Programmstruktur nach oben oder unten zu verschieben.   |
| 4       | Dem Programmbaustein wird eine Sequenz-ID zugewiesen, wie z. B. <code>Rung0</code> . Sie können zusätzlich einen Programmbaustein-Kommentar hinzufügen, um den Programmbaustein durch Klicken auf die Kopfzeile des Programmbausteins zu identifizieren.   |
| 5       | Die Standard-Programmiersprache ist <b>LD</b> (Ladder; dt.: Kontaktplan bzw. KOP). Wenn Sie für diesen Programmbaustein eine andere Programmiersprache auswählen möchten, klicken Sie auf <b>KOP</b> und wählen Sie eine andere Programmiersprache aus.  |
| 6       | <p>Wenn dieser Programmbaustein mittels einer <b>JUMP</b>-Anweisung aufgerufen werden soll, weisen Sie dem Programmbaustein eine Sprungmarke zu, indem Sie auf die Dropdown-Schaltfläche unter der Sequenz-ID des Programmbausteins <code>Rung<math>x</math></code> klicken, wobei <math>x</math> der Programmbausteinnummer in einer POU entspricht. Wählen Sie dann <b>%L</b> aus der Liste aus.</p> <p><b>Ergebnis:</b> Der Programmbaustein erhält die Sprungmarke <b>%L<math>y</math></b>, wobei <math>y</math> der Sprungmarkennummer entspricht. <b>%L</b> wird auf der Schaltfläche angezeigt und die Sprungmarkennummer <math>y</math> erscheint im Suffix aus der Schaltfläche.</p> <p><b>HINWEIS:</b> Wenn Sie die nächste Sprungmarke definieren, wird die Sprungmarkennummer um 1 inkrementiert.</p> <p>Um die Sprungmarkennummer zu ändern, doppelklicken Sie auf die Sprungmarkennummer in einem Programmbaustein, geben Sie die neue Nummer ein, und drücken Sie die Eingabetaste (ENTER).</p> |

## Einfügen eines Programmbausteins über einem bestehenden Programmbaustein

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Wählen Sie im Arbeitsbereich <b>Programmierung</b> einen bereits vorhandenen Programmbaustein aus.   |
| 3       | <p>Klicken Sie auf  (Schaltfläche <b>Neuen Programmbaustein einfügen</b>) in der Symbolleiste am oberen Rand des Arbeitsbereichs für die Programmierung.</p> <p><b>Ergebnis:</b> Über dem ausgewählten Programmbaustein wird ein neuer Programmbaustein angezeigt.</p>  |
| 4       | Dem Programmbaustein wird eine Sequenz-ID zugewiesen, wie z. B. <b>Rung0</b> . Sie können zusätzlich einen Programmbaustein-Kommentar hinzufügen, um den Programmbaustein durch Klicken auf die Kopfzeile des Programmbausteins zu identifizieren.   |
| 5       | Die Standard-Programmiersprache ist <b>LD</b> (Ladder; dt.: Kontaktplan bzw. KOP). Wenn Sie für diesen Programmbaustein eine andere Programmiersprache auswählen möchten, klicken Sie auf <b>LD</b> und wählen Sie ein andere Sprache aus.   |
| 6       | <p>Wenn dieser Programmbaustein mittels einer <b>JUMP</b>-Anweisung aufgerufen werden soll, weisen Sie dem Programmbaustein eine Sprungmarke zu, indem Sie auf die Dropdown-Schaltfläche unter der Sequenz-ID des Programmbausteins <b>Rung<math>x</math></b> klicken, wobei <math>x</math> der Programmbausteinnummer in einer POU entspricht. Wählen Sie dann <b>%L</b> aus der Liste aus.</p> <p><b>Ergebnis:</b> Der Programmbaustein erhält die Sprungmarke <b>%L<math>y</math></b>, wobei <math>y</math> der Sprungmarkennummer entspricht. <b>%L</b> wird auf der Schaltfläche angezeigt und die Sprungmarkennummer <math>y</math> erscheint im Suffix aus der Schaltfläche.</p> <p><b>HINWEIS:</b> Wenn Sie die nächste Sprungmarke definieren, wird die Sprungmarkennummer um 1 inkrementiert.</p> <p>Um die Sprungmarkennummer zu ändern, doppelklicken Sie auf die Sprungmarkennummer in einem Programmbaustein, geben Sie die neue Nummer ein, und drücken Sie die Eingabetaste (ENTER).</p> |

## Kopieren von Programmbausteinen

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Wählen Sie eine oder mehrere Programmbausteine: <ul style="list-style-type: none"> <li>● Wählen Sie einen bestehenden Programmbaustein.</li> <li>● Drücken und halten Sie die STRG-Taste gedrückt, um jeden Programmbaustein auszuwählen.</li> </ul>   |
| 3       | Klicken Sie mit der rechten Maustaste auf einen der ausgewählten Programmbausteine, um diesen zu kopieren und wenden Sie eine der folgenden Methoden an: <ul style="list-style-type: none"> <li>● Wählen Sie im angezeigten Kontextmenü die Option <b>Ausgewählten Programmbaustein kopieren</b> aus.</li> <li>● Drücken Sie Strg + C.</li> </ul>  |
| 4       | Klicken Sie mit der rechten Maustaste auf einen Programmbaustein und wenden Sie eine der folgenden Methoden an: <ul style="list-style-type: none"> <li>● Wählen Sie im angezeigten Kontextmenü die Option <b>Programmbaustein einfügen</b> aus.</li> <li>● Drücken Sie Strg + V.</li> </ul> <p><b>Ergebnis:</b> Eine Kopie des Programmbausteins wird mit derselben Sprungmarke wie der des Original-Programmbaustein eingefügt. Bearbeiten Sie die Sprungmarke nach Bedarf.</p> |

**HINWEIS:** Sie können Programmbausteine auch im Fenster **Programmierung** kopieren und einfügen:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie mit der rechten Maustaste auf den Programmbaustein und wählen Sie <b>Ausgewählten Programmbaustein kopieren</b> aus.               |
| 2       | Klicken Sie mit der rechten Maustaste in den Arbeitsbereich der Programmierung und wählen Sie die Option <b>Programmbaustein einfügen</b> aus. |

## Umbenennen von Programmbausteinen

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Bearbeiten Sie den Programmbausteinennamen mit einer der folgenden Methoden: <ul style="list-style-type: none"> <li>● Klicken Sie mit der rechten Maustaste auf einen Programmbaustein und wählen Sie im daraufhin angezeigten Kontextmenü <b>Programmbaustein umbenennen</b> aus.</li> <li>● Doppelklicken Sie auf einen Programmbaustein.</li> <li>● Wählen Sie einen Programmbaustein aus und doppelklicken Sie auf den Programmbausteinennamen oder den Text <b>Name</b> im Arbeitsbereich der Programmierung.</li> <li>● Wählen Sie einen Programmbaustein und drücken Sie die F2-Taste.</li> </ul> |
| 3       | Geben Sie einen neuen Namen für den Programmbaustein ein und drücken Sie die Eingabetaste (ENTER).   |

## Löschen von Programmbausteinen

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | <p>Löschen Sie einen Programmbaustein mithilfe einer der folgenden Methoden:</p> <ul style="list-style-type: none"> <li>● Klicken Sie mit der rechten Maustaste auf einen Programmbaustein und wählen Sie im daraufhin angezeigten Kontextmenü <b>Programmbaustein löschen</b> aus.</li> <li>● Wählen Sie einen Programmbaustein aus und klicken Sie auf  (Schaltfläche <b>Programmbaustein löschen</b>) in der Symbolleiste am oberen Rand der Registerkarte <b>Tasks</b>.</li> <li>● Wählen Sie einen Programmbaustein aus und klicken Sie auf  (Schaltfläche <b>Programmbaustein löschen</b>) in der Symbolleiste am oberen Rand des Arbeitsbereichs der Programmierung.</li> <li>● Klicken Sie mit der rechten Maustaste auf den Programmbaustein im Arbeitsbereich der Programmierung und wählen Sie im daraufhin angezeigten Kontextmenü <b>Ausgewählten Programmbaustein löschen</b> aus.</li> <li>● Wählen Sie einen Programmbaustein und drücken Sie die ENTF-Taste.</li> </ul> |
| 3       | Wenn der Programmbaustein nicht leer ist, werden Sie aufgefordert, den Löschvorgang zu bestätigen.   |

## Verwalten von Grafcet (SFC)-POUs

### Erstellen einer Grafcet-POU

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | <p>Fügen Sie mithilfe einer der folgenden Methoden eine neue Grafcet-POU hinzu:</p> <ul style="list-style-type: none"> <li>• Klicken Sie mit der rechten Maustaste auf die <b>Master-Task</b> und wählen Sie im Kontextmenü die Option <b>Grafcet-POU hinzufügen</b> aus.</li> <li>• Klicken Sie auf die Schaltfläche  (<b>Grafcet-POU hinzufügen</b>) in der Symbolleiste am oberen Rand der Registerkarte <b>Tasks</b>.</li> </ul> <p><b>Ergebnis:</b> Ein <math>n</math> - <b>Grafcet</b>-Knoten erscheint unter dem Knoten <b>Master-Task</b>, wobei <math>n</math> einer Ganzzahl entspricht, die bei jeder Erstellung einer Grafcet-POU erhöht wird.</p> |

### Einfügen eines Neuen Grafcet-POU

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Wählen Sie eine existierende Grafcet-POU aus, über welcher die neue Grafcet-POU eingefügt werden soll.   |
| 3       | Klicken Sie mit der rechten Maustaste auf die gewählte POU und wählen Sie im daraufhin angezeigten Kontextmenü <b>Grafcet-POU einfügen</b> aus.  |
| 4       | Wenn eine Grafcet-POU in der <b>Master-Task</b> an eine andere Position verschoben werden soll, wählen Sie die Grafcet-POU aus und klicken Sie in der Symbolleiste am oberen Rand des Fensters <b>Tasks</b> auf die AUFWÄRTS- oder ABWÄRTS-Schaltfläche, um die Grafcet-POU in der Programmstruktur nach oben oder unten zu verschieben. |

### Kopieren und Einfügen von Grafcet-POU

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | <p>Wählen Sie eine oder mehrere Grafcet-POUs:</p> <ul style="list-style-type: none"> <li>• Wählen Sie ein bestehende Grafcet-POU in der <b>Master-Task</b>.</li> <li>• Drücken und halten Sie die Strg-Taste gedrückt, um jede Grafcet-POU in der <b>Master-Task</b> auszuwählen.</li> </ul> |
| 3       | Klicken Sie mit der rechten Maustaste auf eine der ausgewählten Grafcet-POUs in der <b>Master-Task</b> und wählen Sie im Kontextmenü die Option <b>POU kopieren</b> aus.   |

| Schritt | Aktion  |
|---------|---|
| 4       | <p>Klicken Sie mit der rechten Maustaste auf die <b>Master-Task</b> und wählen Sie im Kontextmenü <b>POU einfügen</b> aus.</p> <p><b>Ergebnis:</b> Daraufhin wird in der Programmstruktur direkt unter der ausgewählten Grafcet-POU in der <b>Master-Task</b> eine oder mehrere Grafcet-POUs mit dem gleichen Namen wie die kopierte Grafcet-POU hinzugefügt.</p> |

### Umbenennen einer Grafcet-POU

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | <p>Bearbeiten Sie den Namen von Grafcet-POU mit einer der folgenden Methoden:</p> <ul style="list-style-type: none"> <li>● Klicken Sie mit der rechten Maustaste auf eine Grafcet-POU und wählen Sie im daraufhin angezeigten Kontextmenü <b>POU umbenennen</b> aus.</li> <li>● Doppelklicken Sie auf eine Grafcet-POU.</li> <li>● Wählen Sie ein Grafcet-POU und drücken Sie die F2-Taste.</li> </ul> |
| 3       | Geben Sie einen neuen Namen für den Grafcet-POU-Knoten ein und drücken Sie die Eingabetaste (ENTER).   |

### Entfernen von Grafcet-POUs

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | <p>Wählen Sie eine oder mehrere Grafcet-POUs:</p> <ul style="list-style-type: none"> <li>● Wählen Sie ein bestehende Grafcet-POU in der <b>Master-Task</b>.</li> <li>● Drücken und halten Sie die Strg-Taste gedrückt, um jede Grafcet-POU in der <b>Master-Task</b> auszuwählen.</li> </ul>          |
| 3       | <p>Ausgewählte Grafcet-POUs löschen:</p> <ul style="list-style-type: none"> <li>● Klicken Sie mit der rechten Maustaste auf eine ausgewählte Grafcet-POU in der <b>Master-Task</b> und wählen Sie im Kontextmenü die Option <b>POU löschen</b> aus.</li> <li>● Drücken Sie die Taste Entf.</li> </ul> |

## Freie POUs

### Einführung

Eine freie POU in SoMachine Basic ist ein besonderer Typ von POU, der keine Task explizit zugeordnet ist:

- [-] Freie POUs
  - [-] Freie POU\_0 (SR2)
    - Rung0
    - Rung1
  - [-] Freie POU\_1 (SR3)
    - Rung0
  - [-] Freie POU\_2 (SR4)
    - Rung0
    - Rung1
    - Rung2

Jede Freie POU wird als Unterprogramm implementiert und besteht aus einem (1) oder mehreren Programmbausteinen, die in der Programmiersprache des Kontaktplans (Ladder) oder der Anweisungsliste (IL) geschrieben sind.

**HINWEIS:** Grafcet-POUs können keine Freien POUs sein.

Freie POUs werden in folgenden Fällen verbraucht:

- Sie werden über einen Unterprogrammaufruf (SRi) von innerhalb eines Programmbausteins aufgerufen.
- Sie werden als periodische Task konfiguriert.
- Sie werden als Ereignistask konfiguriert, z. B. das Unterprogramm für Schwellenwert 0 eines Hochgeschwindigkeitszähler-Funktionsbausteins (%HSCi.TH0).

Bei einem Verbrauch als periodische Task oder Ereignistask wird das freie POU-Unterprogramm automatisch vom Bereich **Freie POUs** im Fenster **Tasks** in den Fensterbereich **Periodische Task** bzw. **Ereignisse** verschoben.

Sobald das Unterprogramm nicht mehr als periodische Task oder Ereignistask verbraucht wird, wird es in den Bereich **Freie POUs** zurückverschoben und steht für einen neuen Verbrauch durch andere Tasks oder Ereignisse zur Verfügung.

### Erstellen einer neuen freien POU

Gehen Sie zur Erstellung einer neuen freien POU wie folgt vor:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Klicken Sie mit der rechten Maustaste auf <b>Freie POU</b> s und wählen Sie im Kontextmenü <b>Freie POU hinzufügen</b> aus.<br><b>Ergebnis:</b> Daraufhin wird unter dem Zweig Freie POU <sub>SR0</sub> eine neue POU mit dem Standardnamen „ <b>Freie POU_0</b> “ und einer Standard-Unterprogrammnummer „ <b>0</b> “ sowie ein neuer Programmbaustein im Arbeitsbereich <b>Programmierung</b> angezeigt. |
| 3       | Sie können optional mit der rechten Maustaste auf die neue POU klicken und <b>POU umbenennen</b> auswählen, dann einen neuen Namen für die POU eingeben und die Eingabetaste drücken. Der Name der freien POU wird ebenfalls in dem im Arbeitsbereich <b>Programmierung</b> angezeigten Programmbaustein aktualisiert.   |
| 4       | Sie können nach Bedarf auch einen Kommentar ( <i>siehe Seite 204</i> ) für die freie POU eingeben.   |
| 5       | Wählen Sie <b>Unterprogramm Nr.</b> rechts neben dem Kommentarfeld und dann eine Unterprogrammnummer in der Liste aus.<br><b>Ergebnis:</b> Die Beschreibung der POU in der Liste <b>Freie POU</b> s wird mit der ausgewählten Unterprogrammnummer aktualisiert, z. B. „SR11“.  |
| 6       | Erstellen der Programmbausteine/Schritte und Quellcodes für freie POU <sub>s</sub> /freie Grafnet POU <sub>s</sub> in der Ladder/IL-Programmiersprache.  |

### Kopieren und Einfügen bestehender freier POU<sub>s</sub>

Gehen Sie vor wie folgt, um vorhandene, einer Task zugeordnete POU<sub>s</sub> zu kopieren und einzufügen, um eine freie POU zu erstellen:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Wählen Sie eine oder mehrere freie POU <sub>s</sub> aus: <ul style="list-style-type: none"> <li>● Wählen Sie eine bestehende freie POU aus.</li> <li>● Drücken und halten Sie die STRG-Taste gedrückt, um jede freie POU auszuwählen.</li> </ul>   |
| 3       | Klicken Sie mit der rechten Maustaste auf eine der ausgewählten freien POU <sub>s</sub> und wählen Sie im daraufhin angezeigten Kontextmenü <b>POU kopieren</b> aus.   |
| 4       | Klicken Sie mit der rechten Maustaste und wählen Sie im daraufhin angezeigten Kontextmenü <b>POU einfügen</b> aus.<br><b>Ergebnis:</b> Unter <b>Freie POU</b> s erscheint eine oder mehrere neue freie POU <sub>s</sub> mit dem Namen <b>Freie POU_x</b> , wobei x der nächst verfügbaren freien POU-Nummer entspricht, und eine Standard-Unterprogrammnummer <b>SRx</b> , wobei x der nächst verfügbaren Unterprogrammnummer entspricht. Alle Programmbausteine der POU werden der neuen freien POU-Unterprogrammnummer automatisch zugewiesen. |

## Kopieren und Einfügen vorhandener POUs, die einer Task zugeordnet sind

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Wählen Sie eine oder mehrere POUs aus: <ul style="list-style-type: none"> <li>• Wählen Sie ein bestehende POU in der <b>Master-Task</b> aus.</li> <li>• Drücken und halten Sie die STRG-Taste gedrückt, um jede POU in der <b>Master-Task</b> auszuwählen.</li> </ul>  |
| 3       | Klicken Sie mit der rechten Maustaste auf eine der ausgewählten POUs in der <b>Master-Task</b> und wählen Sie im Kontextmenü die Option <b>POU kopieren</b> aus.   |
| 4       | Klicken Sie mit der rechten Maustaste auf die <b>Master-Task</b> und wählen Sie im Kontextmenü <b>POU einfügen</b> aus.<br><b>Ergebnis:</b> Daraufhin wird in der Programmstruktur direkt unter der ausgewählten POU in der <b>Master-Task</b> eine oder mehrere POUs mit dem gleichen Namen wie die kopierte POU hinzugefügt. |

## Exportieren Freier POUs

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken oberen Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Wählen Sie eine oder mehrere der bestehenden Freien POUs in der <b>Master-Task</b> aus.   |
| 3       | Klicken Sie mit der rechten Maustaste auf eine ausgewählte Freie POU in der <b>Master-Task</b> und wählen Sie im erscheinenden Kontextmenü die Option <b>POU exportieren</b> aus. |
| 4       | Speichern Sie die exportieren Freien POUs-Dateien (*.smbf) im Export-Ordner, der erscheint.   |

## Importieren Freier POUs

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken oberen Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Wählen Sie eine oder mehrere der bestehenden Freien POUs in der <b>Master-Task</b> aus.  |
| 3       | Klicken Sie mit der rechten Maustaste auf eine ausgewählte Freie POU in der <b>Master-Task</b> und wählen Sie im erscheinenden Kontextmenü die Option <b>POU importieren</b> aus.  |
| 4       | Wählen Sie die Freien POUs-Dateien (*.smbf) aus dem Ordner aus, der erscheint und klicken Sie auf <b>Öffnen</b> .<br><b>HINWEIS:</b> Wenn die maximale Anzahl Freier POUs erreicht wurde oder die Datei beschädigt ist (ungültiges Format), dann erscheint eine Fehlermeldung und die Freien POUs werden nicht importiert. |

### Entfernen freier POU's

Gehen Sie wie folgt vor, um freie POU's zu entfernen:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Wählen Sie eine oder mehrere freie POU's aus: <ul style="list-style-type: none"> <li>● Wählen Sie eine bestehende freie POU aus.</li> <li>● Drücken und halten Sie die STRG-Taste gedrückt, um jede freie POU auszuwählen.</li> </ul>  |
| 3       | Löschen der ausgewählten freien POU's: <ul style="list-style-type: none"> <li>● Klicken Sie mit der rechten Maustaste auf eine ausgewählte freie POU und wählen Sie im daraufhin angezeigten Kontextmenü <b>POU löschen</b> aus.</li> <li>● Drücken Sie die Taste Entf.</li> </ul> |

**HINWEIS:** Führen sie ein Aufheben der Zuweisung (*siehe Seite 144*) aus, um eine Zuweisung zwischen einer freien POU und einer Task aufzuheben, bevor Sie diese entfernen.

### Zuweisen freier POU's zu Ereignissen oder periodischen Tasks

Standardmäßig werden freie POU's und Unterprogramme keine Ereignissen oder Tasks zugewiesen.

Unter Erstellen periodischer Tasks (*siehe Seite 143*) finden Sie Informationen dazu, wie eine freie POU mit einem periodischen Task verknüpft wird.

Unter Erstellen von Ereignis-Tasks (*siehe Seite 150*) finden Sie Informationen dazu, wie eine freie POU mit einem Ereignis verknüpft wird.

# Abschnitt 6.5

## Benutzerdefinierte Funktionen

### Überblick

Eine benutzerdefinierte Funktion ermöglicht das Erstellen von neuen Funktionen mit Eingangsparametern, lokalen Variablen und Rückgabewert. Benutzerdefinierte Funktionen werden als Teil des SoMachine Basic-Projekts gespeichert.

Benutzerdefinierte Funktionen können aufgerufen werden in:

- Der Master-Task
- Periodischen Tasks
- Ereignisse
- Freie POU's

**HINWEIS:** Die Anwendung muss mit einer Funktionsebene (*siehe Seite 94*) von mindestens **Ebene 6.0** konfiguriert werden, damit benutzerdefinierte Funktionen unterstützt werden.

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema   | Seite |
|---|-------|
| Erstellen einer benutzerdefinierten Funktion  | 114   |
| Definieren einer benutzerdefinierten Funktion | 115   |
| Verwalten von benutzerdefinierten Funktionen  | 120   |

## Erstellen einer benutzerdefinierten Funktion

### Eine neue benutzerdefinierte Funktion hinzufügen

Ein Projekt kann bis zu 64 benutzerdefinierte Funktionen enthalten.

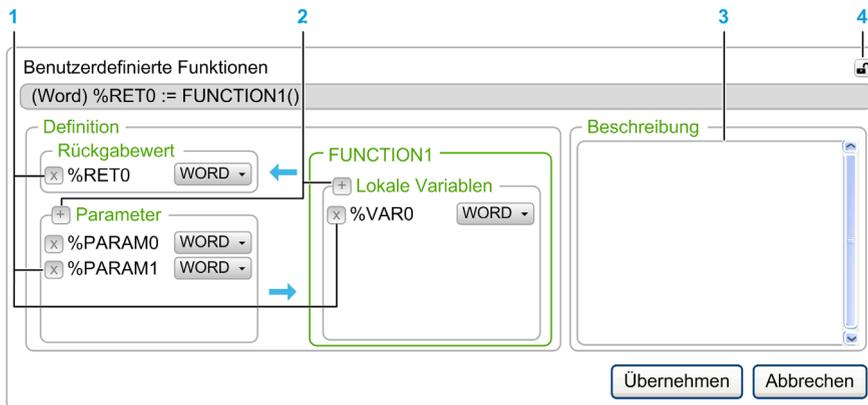
| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | <p>Benutzerdefinierte Funktionen können mithilfe der folgenden Methoden erstellt werden:</p> <ul style="list-style-type: none"> <li>• Klicken Sie mit der rechten Maustaste auf <b>Benutzerdefinierte Funktionen</b> und wählen Sie <b>Benutzerdefinierte Funktion hinzufügen</b> aus dem angezeigten Kontextmenü aus.</li> </ul>  <ul style="list-style-type: none"> <li>• Wählen Sie die <b>Benutzerdefinierte Funktionen</b> aus und klicken Sie auf <b>(Benutzerdefinierte Funktion hinzufügen)</b> in der Symbolleiste am oberen Rand der Registerkarte <b>Tasks</b>.</li> <li>• Wählen Sie eine vorhandene benutzerdefinierte Funktion aus, klicken Sie mit der rechten Maustaste und wählen Sie <b>Benutzerdefinierte Funktion einfügen</b> aus.</li> </ul> <p><b>Ergebnis:</b> Eine neue benutzerdefinierte Funktion wird der Programmstruktur am Ende der Liste hinzugefügt. Wenn Sie eine benutzerdefinierte Funktion eingefügt haben, befindet sich die neue benutzerdefinierte Funktion oberhalb der ausgewählten. Der Standardname lautet <b>FUNCTIONn</b>, wobei n einer Ganzzahl entspricht, die bei jeder Erstellung einer weiteren benutzerdefinierten Funktion inkrementiert wird.</p> |
| 3       | Sie können die benutzerdefinierte Funktion optional umbenennen. Siehe Umbenennen einer benutzerdefinierten Funktion ( <i>siehe Seite 121</i> ).  |
| 4       | Definieren Sie die benutzerdefinierte Funktion. Siehe Definieren einer benutzerdefinierten Funktion ( <i>siehe Seite 115</i> ).  |

Programmbausteine werden in einer benutzerdefinierten Funktion auf dieselbe Weise erstellt und verwaltet, wie Programmbausteine in einer POU. Siehe Verwalten von Programmbausteinen (*siehe Seite 103*).

## Definieren einer benutzerdefinierten Funktion

### Beschreibung

Die folgende Abbildung zeigt die in der Ansicht **Eigenschaften** der benutzerdefinierten Funktion verfügbaren Aktionen:

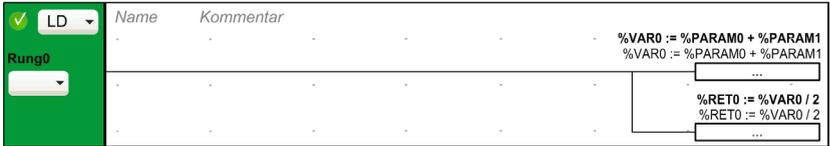


- 1 Löschen des **Rückgabewertes**, eines **Eingangsparameters** oder einer **Lokalen Variablen**
- 2 Hinzufügen eines **Rückgabewertes**, eines **Eingangsparameters** oder einer **Lokalen Variablen**
- 3 Optional können Sie eine Beschreibung des Zwecks der benutzerdefinierten Funktion hinzufügen. Diese Beschreibung wird in einer Quickinfo angezeigt, wenn Sie die benutzerdefinierte Funktion in einem **Operationsbaustein** verwenden.
- 4 Eigenschaftenansicht lösen

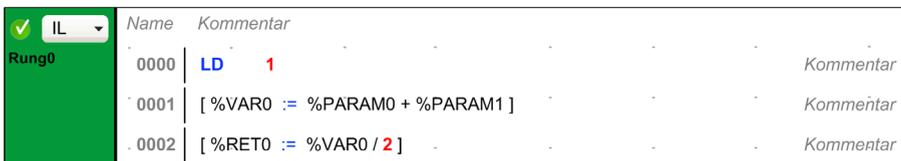
### Programmieren einer benutzerdefinierten Funktion

Gehen Sie wie folgt vor, um eine benutzerdefinierte Funktion zu programmieren:

| Schritt | Aktion   |
|---------|--|
| 1       | Eine neue benutzerdefinierte Funktion hinzufügen. Siehe Hinzufügen einer benutzerdefinierten Funktion ( <i>siehe Seite 114</i> ).  |
| 2       | Definieren Sie die Schnittstelle einer benutzerdefinierten Funktion, indem Sie den <b>Rückgabewert</b> , die <b>Eingangsparameter</b> und die <b>Lokalen Variablen</b> definieren. Siehe Definieren der Schnittstelle einer benutzerdefinierten Funktion ( <i>siehe Seite 117</i> ). |
| 3       | Klicken Sie auf <b>Übernehmen</b> .  |

| Schritt | Aktion  |
|---------|---|
| 4       | <p>Gehen Sie wie folgt vor, um die Funktionsweise der benutzerdefinierten Funktion in einem oder mehreren Kontaktplan-/IL-Programmbausteinen (<i>siehe Seite 103</i>) zu definieren:</p> <ol style="list-style-type: none"> <li>Fügen Sie ein Kontaktplan-Strukturelement ein.</li> <li>Programmieren Sie die benutzerdefinierte Funktion.</li> </ol> <p>Beispiel:</p>  |

Die benutzerdefinierte Funktion kann auch direkt im AWL-Editor programmiert werden:

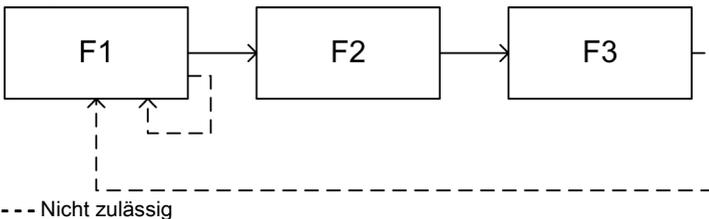


| Name                              | Kommentar |
|-----------------------------------|-----------|
| 0000 LD 1                         | Kommentar |
| 0001 [%VAR0 := %PARAM0 + %PARAM1] | Kommentar |
| 0002 [%RET0 := %VAR0 / 2]         | Kommentar |

Sie können andere benutzerdefinierte Funktionen in den Programmbausteinen aufrufen, die eine benutzerdefinierte Funktion implementieren.

**HINWEIS:** Benutzerdefinierte Funktionen können nicht rekursiv sein: eine benutzerdefinierte Funktion kann sich selbst nicht direkt oder indirekt aufrufen.

Beispiel:



Eine benutzerdefinierte Funktion kann kein Unterprogramm aufrufen, ein Unterprogramm kann jedoch eine benutzerdefinierte Funktion aufrufen.

## Variablen und globale Variablen

Die drei folgenden Variablen können nur in den Programmbausteinen verwendet werden, welche die benutzerdefinierte Funktion implementieren:

- %RET0
- %PARAMn
- %VARn

Globale Variablen sind die anderen Variablen, die in einem SoMachine Basic-Programm verwendet werden können, einschließlich der Programmbausteine einer benutzerdefinierten Funktion.

## Definieren der Schnittstelle einer benutzerdefinierten Funktion

Um eine benutzerdefinierte Funktion verwenden zu können, müssen die Objekte und ihre Datentypen definiert werden.

| Objekt  | Datentyp  | Beschreibung   |
|---|---|--|
| <b>Rückgabewert</b><br>%RET0  | <b>Wort</b><br><b>Doppel</b><br><b>Gleitkomma</b> | Der von der benutzerdefinierten Funktion zurückgegebene Wert.<br>Kann nur in einem Programmbaustein einer benutzerdefinierten Funktion verwendet werden.   |
| <b>Parameter</b><br>%PARAMn <sup>(1)</sup>  |   | Parameter einer benutzerdefinierten Funktion.<br>Kann nur in einem Programmbaustein einer benutzerdefinierten Funktion verwendet werden.<br>Die Standard-Parameteradresse kann nicht geändert werden.<br>In Animationstabellen können keine Parameter hinzugefügt werden.<br>Im Online-Modus werden die aktuellen Werte der Parameter nicht im IL/Ladder-Editor angezeigt.   |
| <b>Lokale Variablen</b><br>%VARn <sup>(1)</sup>   |   | Variablen zum Speichern von Datenwerten innerhalb der benutzerdefinierten Funktion.<br>Können nur in einem Programmbaustein einer benutzerdefinierten Funktion verwendet werden.<br>Die Standardadresse der lokalen Variablen kann nicht geändert werden.<br>In Animationstabellen können keine lokalen Variablen hinzugefügt werden.<br>Im Online-Modus werden die aktuellen Werte von lokalen Variablen nicht im IL/Ladder-Editor angezeigt. |
| <sup>(1)</sup> n ist eine Ganzzahl, die bei jedem Erstellen eines Parameters oder einer lokalen Variablen inkrementiert wird. |   |  |

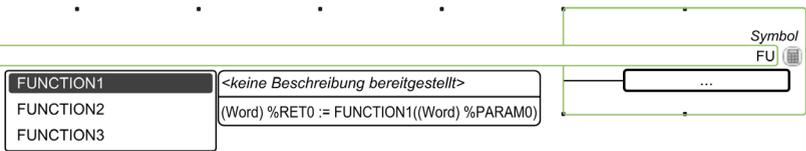
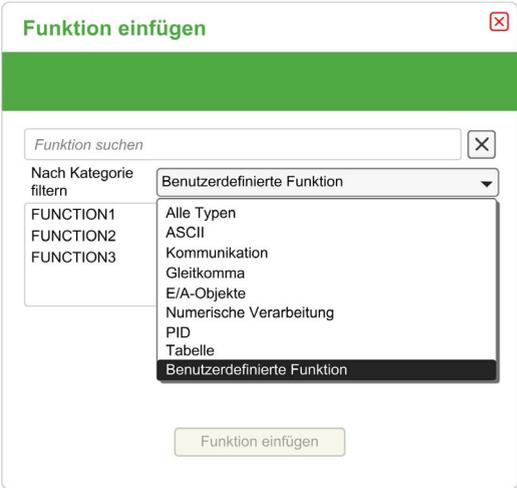
Diese Objekte sind optional.

### Verwenden von benutzerdefinierten Funktionen

Sobald benutzerdefinierte Funktionen definiert wurden, können sie überall im Programm verwendet werden, indem ein **Operationsbaustein** auf dieselbe Weise wie jede andere Funktion verwendet wird.

Im Kontaktplan-Editor:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Schaltfläche <b>Operationsbaustein</b>  in der Symbolleiste. |
| 2       | Klicken Sie in den Aktionsbereich ( <i>siehe Seite 188</i> ) des Programmbausteins, um den <b>Operationsbaustein</b> einzufügen.                                  |
| 3       | Klicken Sie auf die Schaltfläche <b>Auswahlmodus</b>  in der Symbolleiste.       |

| Schritt | Aktion  |
|---------|---|
| 4       | <p>Doppelklicken Sie auf die Zeile <b>Operationsausdruck</b>. Folgende Möglichkeiten stehen Ihnen zur Verfügung:</p> <ul style="list-style-type: none"> <li>• Den Namen der benutzerdefinierten Funktion eingeben. Geben Sie beispielsweise für den Namen „FUNKTION1“ „FU“ ein, dann werden die Namen aller benutzerdefinierten Funktionen angezeigt, die mit „FU“ beginnen:</li> </ul>  <ul style="list-style-type: none"> <li>• Die Intelligente Code-Bearbeitung (<i>siehe Seite 201</i>) verwenden:             <ol style="list-style-type: none"> <li>a. Klicken Sie auf die Schaltfläche Intelligente Code-Bearbeitung .</li> <li>b. Wählen Sie <b>Nach Kategorie filtern</b> und anschließend <b>Benutzerdefinierte Funktion</b> aus.</li> <li>c. Wählen Sie die benutzerdefinierte Funktion aus.</li> </ol> </li> </ul>  |
| 5       | Klicken Sie auf <b>Funktion einfügen</b> .  |
| 6       | Vervollständigen Sie die Definition der benutzerdefinierten Funktion, indem Sie den Rückgabewert und die Parameter eingeben, wie sie in Definieren der Schnittstelle einer benutzerdefinierten Funktion ( <i>siehe Seite 117</i> ) definiert sind.  |

## Verwalten von benutzerdefinierten Funktionen

### Benutzerdefinierte Funktionen im Offline- und Online-Modus

Benutzerdefinierte Funktionen können im Offline-Modus verwaltet werden.

Im Online-Modus haben Sie folgende Möglichkeiten:

- Hinzufügen eines Programmbausteins zu einer vorhandenen benutzerdefinierten Funktion
- Kopieren/Einfügen einer benutzerdefinierten Funktion
- Importieren/Exportieren einer benutzerdefinierten Funktion
- Im Status `STOPPED` kann ein Programmbausteins durch Aufrufen einer benutzerdefinierten Funktion geändert werden

### Kopieren/Ausschneiden und Einfügen von vorhandenen benutzerdefinierten Funktionen

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Auswählen einer oder mehrerer benutzerdefinierter Funktionen: <ul style="list-style-type: none"> <li>● Klicken Sie zum Auswählen auf eine benutzerdefinierte Funktion.</li> <li>● Wählen Sie mehrere benutzerdefinierte Funktionen aus, indem Sie die Strg-Taste gedrückt halten.</li> </ul>  |
| 3       | Klicken Sie mit der rechten Maustaste und wählen Sie <b>Benutzerdefinierte Funktionen kopieren</b> oder <b>Benutzerdefinierte Funktionen ausschneiden</b> aus dem angezeigten Kontextmenü aus.  |
| 4       | Klicken Sie mit der rechten Maustaste auf <b>Benutzerdefinierte Funktionen</b> und wählen Sie <b>Benutzerdefinierte Funktionen einfügen</b> aus dem angezeigten Kontextmenü aus.<br><b>Ergebnis:</b> Eine oder mehrere benutzerdefinierte Funktionen werden am Ende der Programmstruktur unter <b>Benutzerdefinierte Funktionen</b> hinzugefügt. SoMachine Basic weist der kopierten benutzerdefinierten Funktion automatisch einen neuen Namen zu. |

### Exportieren von benutzerdefinierten Funktionen

Benutzerdefinierte Funktionen werden als Teil des Projekts gespeichert. Wenn Sie eine benutzerdefinierte Funktion in einem anderen Projekt verwenden möchten, müssen Sie diese exportieren und anschließend in das andere Projekt importieren.

Das Kopieren/Einfügen zwischen den SoMachine Basic-Instanzen ist möglich.

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Auswählen einer oder mehrerer benutzerdefinierter Funktionen: <ul style="list-style-type: none"> <li>● Klicken Sie zum Auswählen auf eine benutzerdefinierte Funktion.</li> <li>● Wählen Sie mehrere benutzerdefinierte Funktionen aus, indem Sie die Strg-Taste gedrückt halten.</li> </ul> |

| Schritt | Aktion  |
|---------|---|
| 3       | Klicken Sie mit der rechten Maustaste auf die ausgewählten benutzerdefinierten Funktionen unter <b>Benutzerdefinierte Funktionen</b> und wählen Sie <b>Benutzerdefinierte Funktion exportieren</b> aus dem angezeigten Kontextmenü aus. |
| 4       | Speichern Sie die Datei der benutzerdefinierten Funktion (*.smbf) im <b>Export</b> -Ordner, der erscheint.  |

### Importieren einer benutzerdefinierten Funktion

Benutzerdefinierte Funktionen werden als Teil des Projekts gespeichert. Wenn Sie eine benutzerdefinierte Funktion in einem anderen Projekt verwenden möchten, müssen Sie diese exportieren und anschließend in das andere Projekt importieren.

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Wählen Sie unter <b>Benutzerdefinierte Funktionen</b> eine benutzerdefinierte Funktion aus.   |
| 3       | Klicken Sie mit der rechten Maustaste auf die ausgewählte benutzerdefinierte Funktion unter <b>Benutzerdefinierte Funktionen</b> und wählen Sie <b>Benutzerdefinierte Funktion importieren</b> aus dem angezeigten Kontextmenü aus. |
| 4       | Navigieren Sie zu dem Ordner, der die Datei der benutzerdefinierte Funktion (*.smbf) enthält und wählen Sie die benutzerdefinierte Funktion aus.  |
| 5       | Bestätigen Sie mit <b>OK</b> .  |

### Umbenennen einer benutzerdefinierten Funktion

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Umbenennen mithilfe einer der folgenden Methoden: <ul style="list-style-type: none"> <li>• Klicken Sie mit der rechten Maustaste auf eine benutzerdefinierte Funktion und wählen Sie <b>Benutzerdefinierte Funktion umbenennen</b> aus dem angezeigten Kontextmenü aus.</li> <li>• Doppelklicken Sie im Arbeitsbereich der Programmierung auf die benutzerdefinierte Funktion.</li> <li>• Wählen Sie eine benutzerdefinierte Funktion aus und drücken Sie die F2-Taste.</li> </ul> |
| 3       | Geben Sie einen neuen Namen für die benutzerdefinierte Funktion ein und drücken Sie die Eingabetaste (ENTER).<br>Die folgenden Zeichen sind zulässig: A...Z, 0...9, _ . Der Name muss eindeutig sein. Andernfalls bleibt der Name unverändert.   |

## Löschen von benutzerdefinierten Funktionen

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Wählen Sie eine oder mehrere benutzerdefinierte Funktionen aus, indem Sie die Strg-Taste gedrückt halten.  |
| 3       | <p>Die ausgewählten benutzerdefinierten Funktionen können mithilfe der folgenden Methoden gelöscht werden:</p> <ul style="list-style-type: none"> <li>● Klicken Sie mit der rechten Maustaste auf eine ausgewählte benutzerdefinierte Funktion unter <b>Benutzerdefinierte Funktionen</b> und wählen Sie <b>Benutzerdefinierte Funktion löschen</b> aus dem angezeigten Kontextmenü aus.</li> <li>● Drücken Sie die Taste Entf.</li> </ul> <p>● Klicken Sie auf  in der Symbolleiste am oberen Rand der Registerkarte <b>Tasks</b>.</p> |

# Abschnitt 6.6

## Benutzerdefinierte Funktionsbausteine

### Überblick

Ein benutzerdefinierter Funktionsbaustein ermöglicht das Erstellen von neuen Funktionsbausteinen mit einem oder mehreren Eingangs- und Ausgangsparametern und lokalen Variablen. Benutzerdefinierte Funktionsbausteine werden als Teil des SoMachine Basic-Projekts gespeichert.

Benutzerdefinierte Funktionsbausteine können aufgerufen werden in:

- Der Master-Task
- Periodischen Tasks
- Ereignisse
- Freie POUs

**HINWEIS:** Die Anwendung muss mit einer Funktionsebene (*siehe Seite 94*) von mindestens **Ebene 6.0** konfiguriert werden, damit benutzerdefinierte Funktionsbausteine unterstützt werden.

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema   | Seite |
|---|-------|
| Erstellen eines benutzerdefinierten Funktionsbausteins  | 124   |
| Definieren eines benutzerdefinierten Funktionsbausteins | 125   |
| Verwalten von benutzerdefinierten Funktionsbausteinen   | 128   |

## Erstellen eines benutzerdefinierten Funktionsbausteins

### Einen neuen benutzerdefinierten Funktionsbaustein hinzufügen

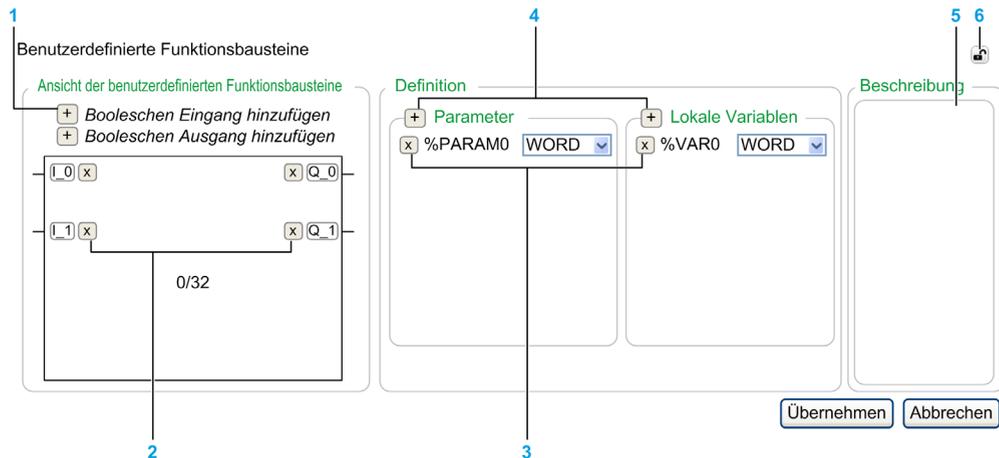
| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | <p>Benutzerdefinierte Funktionsbausteine können mithilfe der folgenden Methoden erstellt werden:</p> <ul style="list-style-type: none"> <li>• Klicken Sie mit der rechten Maustaste auf <b>Benutzerdefinierte Funktionsbausteine</b> und wählen Sie <b>Benutzerdefinierten Funktionsbausteine hinzufügen</b> aus dem angezeigten Kontextmenü aus.</li> </ul> <p style="text-align: right;"></p> <ul style="list-style-type: none"> <li>• Wählen Sie <b>Benutzerdefinierte Funktionsbausteine</b> aus und klicken Sie auf <b>(Benutzerdefinierten Funktionsbaustein hinzufügen)</b> in der Symbolleiste am oberen Rand der Registerkarte <b>Tasks</b>.</li> <li>• Wählen Sie einen vorhandenen benutzerdefinierten Funktionsbaustein aus, klicken Sie mit der rechten Maustaste und wählen Sie <b>Benutzerdefinierten Funktionsbaustein einfügen</b> aus.</li> </ul> <p><b>Ergebnis:</b> Ein neuer benutzerdefinierter Funktionsbaustein wird der Programmstruktur direkt am Ende der Liste hinzugefügt. Wenn Sie einen benutzerdefinierten Funktionsbaustein eingefügt haben, befindet sich der neue benutzerdefinierte Funktionsbaustein oberhalb des ausgewählten. Der Standardname lautet <b>UDFBn</b>, wobei n einer Ganzzahl entspricht, die bei jeder Erstellung eines weiteren benutzerdefinierten Funktionsbausteins inkrementiert wird.</p> |
| 3       | Sie können den benutzerdefinierten Funktionsbaustein optional umbenennen. Siehe Umbenennen eines benutzerdefinierten Funktionsbausteins ( <i>siehe Seite 129</i> ).   |
| 4       | Definieren Sie den benutzerdefinierten Funktionsbaustein. Siehe Definieren eines benutzerdefinierten Funktionsbausteins ( <i>siehe Seite 125</i> ).   |

Ein Programmbaustein wird in einem benutzerdefinierten Funktionsbaustein auf dieselbe Weise erstellt und verwaltet, wie ein Programmbaustein in einer POU. Siehe Verwalten von Programmbausteinen (*siehe Seite 103*).

## Definieren eines benutzerdefinierten Funktionsbausteins

### Beschreibung

Die folgende Abbildung zeigt die in der Ansicht **Eigenschaften** der benutzerdefinierten Funktion verfügbaren Aktionen:



- 1 Hinzufügen eines **Eingangs** oder **Ausgangs**
- 2 Löschen eines **Eingangs** oder **Ausgangs**
- 3 Löschen des **Parameters** oder der **lokalen Variable**
- 4 Hinzufügen eines **Parameters** oder einer **lokalen Variable**
- 5 Optional können Sie eine Beschreibung des Zwecks des benutzerdefinierten Funktionsbausteins hinzufügen. Diese Beschreibung wird in einer Quickinfo angezeigt, wenn Sie den benutzerdefinierten Funktionsbaustein in einem **Operationsbaustein** verwenden.
- 6 Eigenschaftenansicht lösen

### Programmieren eines benutzerdefinierten Funktionsbausteins

Gehen Sie wie folgt vor, um einen benutzerdefinierten Funktionsbaustein zu programmieren:

| Schritt | Aktion   |
|---------|--|
| 1       | Einen neuen benutzerdefinierten Funktionsbaustein hinzufügen. Siehe Hinzufügen eines benutzerdefinierten Funktionsbausteins ( <i>siehe Seite 124</i> ).  |
| 2       | Definieren Sie die Schnittstelle des benutzerdefinierten Funktionsbausteins, indem Sie die <b>Eingangsparameter</b> und die <b>Lokalen Variablen</b> definieren. Siehe Definieren der Schnittstelle eines benutzerdefinierten Funktionsbausteins ( <i>siehe Seite 126</i> ). |
| 3       | Klicken Sie auf <b>Übernehmen</b> .  |

| Schritt | Aktion   |
|---------|--|
| 4       | <p>Gehen Sie wie folgt vor, um die Funktionsweise des benutzerdefinierten Funktionsbausteins in einem oder mehreren Kontaktplan-/IL-Programmbausteinen (<i>siehe Seite 103</i>) festzulegen:</p> <ol style="list-style-type: none"> <li>1. Klicken Sie auf die Schaltfläche <b>Funktionsbausteine</b>  in der Symbolleiste.</li> <li>2. Wählen Sie  → den benutzerdefinierten Funktionsbaustein, den Sie einfügen möchten.</li> <li>3. Klicken Sie in den Aktionsbereich (<i>siehe Seite 188</i>) des Programmbausteins.</li> <li>4. Programmieren Sie den benutzerdefinierten Funktionsbaustein.</li> </ol> |

Im AWL-Editor kann kein benutzerdefinierter Funktionsbaustein programmiert werden.

### Lokale und globale Variablen

Die lokalen Variablen sind Variablen, die nur in den Programmbausteinen verwendet werden können, welche den benutzerdefinierten Funktionsbaustein implementieren:

- %PARAMn
- %VARn

Globale Variablen sind alle anderen Variablen, die in einem SoMachine Basic-Programm verwendet werden können, einschließlich der Programmbausteine eines benutzerdefinierten Funktionsbausteins.

### Definieren der Schnittstelle eines benutzerdefinierten Funktionsbausteins

Um einen benutzerdefinierten Funktionsbaustein verwenden zu können, müssen die Eingänge, Ausgänge, Objekttypen und ihre Datentypen definiert werden.

| Objekttyp  | Datentyp  | Beschreibung  |
|--|---|---|
| <b>Parameter</b><br>%PARAMn <sup>(1)</sup>   | <b>Wort</b><br><b>Doppel</b><br><b>Gleitkomma</b> | Können nur in einem Programmbaustein einer benutzerdefinierten Funktion und eines benutzerdefinierten Funktionsbausteins verwendet werden.<br>Die Standard-Parameteradresse kann nicht geändert werden.<br>In Animationstabellen können keine Parameter hinzugefügt werden.                     |
| <b>Lokale Variablen</b><br>%VARn <sup>(1)</sup>  |   | Können nur in einem Programmbaustein einer benutzerdefinierten Funktion und eines benutzerdefinierten Funktionsbausteins verwendet werden.<br>Die Standardadresse der lokalen Variablen kann nicht geändert werden.<br>In Animationstabellen können keine lokalen Variablen hinzugefügt werden. |
| (1) n ist eine Ganzzahl, die bei jedem Erstellen eines Parameters oder einer lokalen Variablen inkrementiert wird. |   |   |

## Verwenden eines benutzerdefinierten Funktionsbausteins

Gehen Sie wie folgt vor, um einen benutzerdefinierten Funktionsbaustein in einen Programm-  
baustein einzufügen:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Schaltfläche <b>Funktionsbausteine</b>  in der Symbolleiste.  |
| 2       | Wählen Sie  → den benutzerdefinierten Funktionsbaustein, den Sie einfügen möchten.  |
| 3       | Klicken Sie in den Aktionsbereich ( <i>siehe Seite 188</i> ) des Programmbausteins.  |
| 4       | <p>Verwenden Sie optional Operationsbausteine zum Lesen oder Schreiben von Funktionsbausteinparametern.<br/>Die Syntax lautet <code>%&lt;UDFB name&gt;&lt;instance number&gt;.PARAMn</code>, wobei n eine Ganzzahl ist, die der Parameternummer entspricht.</p> <p>Beispiel:</p> <ul style="list-style-type: none"> <li>• Sie haben einen benutzerdefinierten Funktionsbaustein mit der Bezeichnung <code>MY_FB</code> mit einem Parameter <code>%PARAM0</code> definiert.</li> <li>• Eine Instanz dieses benutzerdefinierten Funktionsbausteins wird in der Master-Task platziert und ihr wird die Instanznummer 0 zugewiesen.</li> </ul> <p><b>Ergebnis:</b> Das Objekt <code>%MY_FB0.PARAM0</code> ist in jeder Task verfügbar.</p> |

## Verwalten von benutzerdefinierten Funktionsbausteinen

### Benutzerdefinierte Funktionsbausteine im Offline- und Online-Modus

Die benutzerdefinierten Funktionsbausteine können im Offline-Modus verwaltet werden.

Im Online-Modus haben Sie folgende Möglichkeiten:

- Hinzufügen eines Programmbausteins zu einem vorhandenen benutzerdefinierten Funktionsbaustein
- Kopieren/Einfügen eines benutzerdefinierten Funktionsbausteins
- Importieren/Exportieren eines benutzerdefinierten Funktionsbausteins
- Im Status `STOPPED` kann ein Programmbaustein durch Aufrufen eines benutzerdefinierten Funktionsbausteins geändert werden

### Kopieren/Ausschneiden und Einfügen von vorhandenen benutzerdefinierten Funktionsbausteinen

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Auswählen eines oder mehrerer benutzerdefinierter Funktionsbausteine: <ul style="list-style-type: none"> <li>● Klicken Sie zum Auswählen auf einen benutzerdefinierten Funktionsbaustein.</li> <li>● Wählen Sie mehrere benutzerdefinierte Funktionsbausteine aus, indem Sie die Strg-Taste gedrückt halten.</li> </ul>   |
| 3       | Klicken Sie mit der rechten Maustaste und wählen Sie <b>Benutzerdefinierte Funktionsbausteine kopieren</b> oder <b>Benutzerdefinierte Funktionsbausteine ausschneiden</b> aus dem angezeigten Kontextmenü aus.  |
| 4       | Klicken Sie mit der rechten Maustaste auf <b>Benutzerdefinierte Funktionsbausteine</b> und wählen Sie <b>Benutzerdefinierte Funktionsbausteine einfügen</b> aus dem angezeigten Kontextmenü aus.<br><b>Ergebnis:</b> Einer oder mehrere benutzerdefinierte Funktionsbausteine werden am Ende der Programmstruktur unter <b>Benutzerdefinierte Funktionsbausteine</b> hinzugefügt. SoMachine Basic weist dem kopierten benutzerdefinierten Funktionsbaustein automatisch einen neuen Namen zu. |

### Exportieren von benutzerdefinierten Funktionsbausteinen

Benutzerdefinierte Funktionsbausteine werden als Teil des Projekts gespeichert. Wenn Sie einen benutzerdefinierten Funktionsbaustein in einem anderen Projekt verwenden möchten, müssen Sie diesen exportieren und anschließend in das andere Projekt importieren.

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Auswählen eines oder mehrerer benutzerdefinierter Funktionsbausteine: <ul style="list-style-type: none"> <li>● Klicken Sie zum Auswählen auf einen benutzerdefinierten Funktionsbaustein.</li> <li>● Wählen Sie mehrere benutzerdefinierte Funktionsbausteine aus, indem Sie die Strg-Taste gedrückt halten.</li> </ul> |

| Schritt | Aktion  |
|---------|---|
| 3       | Klicken Sie mit der rechten Maustaste auf die ausgewählten benutzerdefinierten Funktionsbausteine unter <b>Benutzerdefinierte Funktionsbausteine</b> und wählen Sie <b>Benutzerdefinierten Funktionsbaustein exportieren</b> aus dem angezeigten Kontextmenü aus. |
| 4       | Speichern Sie die Datei des benutzerdefinierten Funktionsbausteins (*.smbf) im <b>Export</b> -Ordner, der erscheint.  |

### Importieren eines benutzerdefinierten Funktionsbausteins

Benutzerdefinierte Funktionsbausteine werden als Teil des Projekts gespeichert. Wenn Sie einen benutzerdefinierten Funktionsbaustein in einem anderen Projekt verwenden möchten, müssen Sie diesen exportieren und anschließend in das andere Projekt importieren.

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Wählen Sie unter <b>Benutzerdefinierte Funktionsbausteine</b> eine benutzerdefinierte Funktion aus.   |
| 3       | Klicken Sie mit der rechten Maustaste auf die ausgewählte benutzerdefinierte Funktion unter <b>Benutzerdefinierte Funktionsbausteine</b> und wählen Sie <b>Benutzerdefinierten Funktionsbaustein importieren</b> aus dem angezeigten Kontextmenü aus. |
| 4       | Navigieren Sie zu dem Ordner, der die Datei des benutzerdefinierten Funktionsbausteins (*.smbf) enthält und wählen Sie den benutzerdefinierten Funktionsbaustein aus.   |
| 5       | Bestätigen Sie mit <b>OK</b> .  |

### Umbenennen eines benutzerdefinierten Funktionsbausteins

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Umbenennen mithilfe einer der folgenden Methoden: <ul style="list-style-type: none"> <li>● Klicken Sie mit der rechten Maustaste auf einen benutzerdefinierten Funktionsbaustein und wählen Sie <b>Benutzerdefinierten Funktionsbaustein umbenennen</b> aus dem angezeigten Kontextmenü aus.</li> <li>● Doppelklicken Sie im Arbeitsbereich der Programmierung auf den Namen des benutzerdefinierten Funktionsbausteine.</li> <li>● Wählen Sie einen benutzerdefinierten Funktionsbaustein aus und drücken Sie die F2-Taste.</li> </ul> |
| 3       | Geben Sie einen neuen Namen für den benutzerdefinierten Funktionsbaustein ein und drücken Sie die Eingabetaste (ENTER).<br>Die folgenden Zeichen sind zulässig: A...Z, 0...9, _, -. Der Name muss eindeutig sein. Andernfalls bleibt der Name unverändert.  |

## Löschen von benutzerdefinierten Funktionsbausteinen

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Wählen Sie einen oder mehrere benutzerdefinierte Funktionsbausteine aus, indem Sie die Strg-Taste gedrückt halten.   |
| 3       | <p>Die ausgewählten benutzerdefinierten Funktionsbausteine können mithilfe der folgenden Methoden gelöscht werden:</p> <ul style="list-style-type: none"> <li>● Klicken Sie mit der rechten Maustaste auf einen ausgewählten benutzerdefinierten Funktionsbaustein unter <b>Benutzerdefinierte Funktionsbausteine</b> und wählen Sie <b>Benutzerdefinierten Funktionsbaustein löschen</b> aus dem angezeigten Kontextmenü aus.</li> <li>● Drücken Sie die Taste Entf.</li> <li>● Klicken Sie auf  in der Symbolleiste am oberen Rand der Registerkarte <b>Tasks</b>.</li> </ul> |

---

## Abschnitt 6.7

### Master-Task

---

#### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                          | Seite |
|--------------------------------|-------|
| Beschreibung der Master-Task   | 132   |
| Konfigurieren des Master-Tasks | 133   |

## Beschreibung der Master-Task

### Übersicht

Die Master-Task fungiert als Haupttask in einem Anwendungsprogramm. Sie ist obligatorisch und wird standardmäßig erstellt. Die Master-Task besteht aus Sections und Unterprogrammen, die innerhalb von Programmorganisationseinheiten (POUs) dargestellt werden. Die jeweiligen POU's der Master-Task können in jeder der unterstützten Programmiersprachen programmiert werden.

### Vorgehensweise

| Verwendung                               | Siehe   |
|--|---|
| Erstellen einer neuen POU im Master-Task | Erstellen einer neuen, einem Task zugeordneten POU<br>(siehe Seite 100) |
| Umbenennen einer POU im Master-Task      | Umbenennen einer POU (siehe Seite 102)                                  |
| Entfernen einer POU aus dem Master-Task  | Entfernen einer POU (siehe Seite 102)                                   |

## Konfigurieren des Master-Tasks

### Vorgehensweise

Gehen Sie wie folgt vor, um den Master-Task zu konfigurieren:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Wählen Sie das Element <b>Master-Task</b> aus.<br><b>Ergebnis:</b> Im unteren mittleren Bereich des SoMachine Basic-Fensters wird daraufhin das Eigenschaftsfenster der <b>Master-Task</b> angezeigt. |
| 3       | Nehmen Sie an den Eigenschaften die gewünschten Änderungen vor.   |
| 4       | Klicken Sie auf <b>Übernehmen</b> , um die Änderungen zu speichern.   |

### Eigenschaften des Master-Tasks

#### Abfragemodus

Wählen Sie den für das Programm zu verwendenden Abfragemodus aus:

- **Normal** Wenn sich eine Logiksteuerung im normalen (freilaufenden) Abfragemodus befindet, startet ein neuer Abfragezyklus, sobald die vorhergehende Abfrage abgeschlossen ist.
- **Periodisch:** Im periodischen Abfragemodus wartet die Logiksteuerung, bis die konfigurierte Abfragezeit abgelaufen ist, bevor eine neue Abfrage gestartet wird. Daher ist jede Abfrage von gleicher Dauer.

Geben Sie für den periodischen Abfragemodus die **Periode** (Dauer) des Abfragezyklus zwischen 2 und 150 ms ein.

### Systembits und -wörter zur Steuerung der Master-Task

Die Master-Task kann über Systembits (%S) und Systemwörter (%SW) gesteuert werden.

In der nachstehenden Tabelle werden die Systembits aufgeführt:

| Systembits | Beschreibung  |
|------------|---|
| %S11       | Watchdog-Überlauf                                       |
| %S19       | Überlauf der Abfrageperiode (periodischer Abfragemodus) |

In der nachstehenden Tabelle werden die Systemwörter aufgeführt:

| Systemwörter | Beschreibung   |
|--------------|--|
| %SW0         | Abfrageperiode der Steuerung (periodischer Abfragemodus)   |
| %SW30, %SW70 | <p>Letzte Zykluszeit. Gibt die Ausführungszeit des letzten Abfragezyklus der Steuerung an, d. h. Zeit zwischen dem Start (Erfassung der Eingänge) und dem Ende (Aktualisierung der Ausgänge) eines Abfragezyklus der Master-Task. %SW30 gibt Millisekunden wieder und %SW70 gibt Mikrosekunden wieder.</p> <p>Beispielsweise entspricht bei der Abfragezeit 2,250 ms, %SW30 = 2 und %SW70 = 250.</p> |
| %SW31, %SW71 | <p>Maximale Zykluszeit. Gibt die Ausführungszeit des längsten Abfragezyklus der Steuerung seit dem letzten Kaltstart an. %SW31 gibt Millisekunden wieder und %SW71 gibt Mikrosekunden wieder.</p> <p>Beispielsweise entspricht bei der Abfragezeit 2,250 ms, %SW31 = 2 und %SW71 = 250.</p>  |
| %SW32, %SW72 | <p>Minimale Zykluszeit. Gibt die Ausführungszeit des kürzesten Abfragezyklus der Steuerung seit dem letzten Kaltstart an. %SW32 gibt Millisekunden wieder und %SW72 gibt Mikrosekunden wieder.</p> <p>Beispielsweise entspricht bei der Abfragezeit 2,250 ms, %SW32 = 2 und %SW72 = 250.</p>   |

Eine vollständige Liste der Systembits und -wörter sowie deren Bedeutung finden Sie im *Programmierhandbuch* Ihrer Hardwareplattform.

# Abschnitt 6.8

## Zeichenketten

### Überblick

Zeichenketten sind eine Byte-Sequenz, die ASCII-Zeichen enthält, die in den folgenden Speicherwörtern gespeichert werden können:

- Speicherwörter %MW
- Konstante Wörter %KW

In einem Wort sind zwei Byte enthalten.

Syntax zum Programmieren einer Zeichenkette:

`%MWx : L`

**x** Index des Speicherobjekts

**L** Die Anzahl der von der Zeichenkette verwendeten Wörter muss zwischen 1...255 liegen.

Die unterstützten Steuerungen verfügen über eine Little-Endian-Architektur; die Byte werden beginnend mit dem niederwertigsten Byte hin zum höchstwertigen Byte gespeichert.

Die folgende Tabelle zeigt ein Beispiel für die Speicherung der Byte für die Zeichenkette *Basis*:

| Speicherobjekte   | Hexadezimal | ASCII              |
|---|-------------|--------------------|
| %MW0 oder %KW0  | 6142        | aB                 |
| %MW1 oder %KW1  | 6973        | gleich             |
| %MW2 oder %KW2  | 0D63        | \rc <sup>(1)</sup> |
| <sup>(1)</sup> \r ist die Markierung, die das Endzeichen der Zeichenkette darstellt. Diese Markierung wird bei der Verarbeitung der Zeichenketten berücksichtigt. |             |                    |

Sie können bis zu 509 Zeichen eingeben.

**HINWEIS:** Die Speicherobjekte werden als Variable oder für eine Zeichenkette verwendet. Wenn Sie ein Speicherobjekt für eine Zeichenkette konfiguriert haben, dann konfigurieren Sie nicht die als Variable enthaltenen Speicherwörter.

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema   | Seite |
|---|-------|
| Konfigurieren von Zeichenketten in Konstanten im Wortformat | 137   |
| Zuweisen von Zeichenfolgen in Speicherwörtern               | 138   |
| Verwalten von Zeichenfolgen                                 | 139   |

## Konfigurieren von Zeichenketten in Konstanten im Wortformat

### Eingeben einer Zeichenkette

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie im Fenster <b>Programmierung</b> auf <b>Tools</b> → <b>Speicherobjekte</b> → <b>Konstanten im Wortformat</b> .  |
| 2       | Klicken Sie in den <b>Eigenschaften der Konstantwörter</b> auf <b>%KW</b> .   |
| 3       | Klicken Sie für die Konstante im Wortformat, die Sie konfigurieren möchten, auf die Schaltfläche  in der Spalte <b>Konfiguration</b> .<br>Wurde die Konstante im Wortformat bereits konfiguriert, wird das Fenster <b>Bestätigung</b> angezeigt. Klicken Sie auf <b>Ok</b> , um den Wert zu überschreiben. Klicken Sie andernfalls auf <b>Abbrechen</b> .<br><b>Ergebnis:</b> Das Fenster <b>Assistent für Konstantzeichenfolgen</b> wird angezeigt. |
| 4       | Geben Sie die Zeichenkette ein.<br><b>Ergebnis: Erforderlicher Konstantenbereich</b> definiert die für die Zeichenkette verwendeten Konstanten im Wortformat.   |
| 5       | Klicken Sie auf <b>Übernehmen</b> .   |

**Ergebnis:** Die eingegebenen Zeichen werden auf die entsprechenden und erforderlichen konstanten Variablen angewandt. Die Zeichen werden invertiert. Siehe Übersicht dieses Abschnitts (*siehe Seite 135*).

## Zuweisen von Zeichenfolgen in Speicherwörtern

### Syntax

Im Folgenden wird die Syntax in Anweisungsliste (AWL) beschrieben. Über ein **Operationsbaustein**-Grafikelement können Sie AWL-Operationen und -Zuweisungsanweisungen (*siehe Seite 200*) in Kontaktplan-Programmbausteinen einfügen.

Verwenden Sie zum Zuweisen einer Zeichenfolge in einem Speicherwort die folgende Syntax: Op1 := "Your string".

Beispiel:

```
%MW10:20 := "This is a SoMachine Basic string."
```

Geben Sie %MWx:? := "Your string" ein, wenn Sie möchten, dass der erforderliche Speicherplatz von der Software berechnet wird.

### Verwendungsregeln

Beim Zuweisen einer Zeichenfolge:

- Stellen Sie sicher, dass es zu keiner Überlappung kommt. Sie können eine Zeichenfolge durch eine andere Zeichenfolge löschen.
- Verwenden Sie nicht das Zeichen “.

## Verwalten von Zeichenfolgen

### Einführung

Die folgenden Funktionen bieten folgende Möglichkeiten:

- Kopieren einer Zeichenfolge.
- Abrufen der Länge einer Zeichenfolge.
- Verkettung von zwei Zeichenfolgen.
- Vergleichen von zwei Zeichenfolgen.

### Syntax

Im Folgenden wird die Syntax in Anweisungsliste (AWL) beschrieben. Über ein **Operationsbaustein**-Grafikelement können Sie AWL-Operationen und -Zuweisungsanweisungen (*siehe Seite 200*) in Kontaktplan-Programmbausteinen einfügen.

### Kopieren einer Zeichenfolge

Verwenden Sie zum Kopieren einer Zeichenfolge die folgende Syntax:  $Op1 := Op2$ .

Die folgende Tabelle zeigt die für Op1 und Op2 autorisierten Speicherobjekte:

| Parameter   | Beschreibung           |
|---|------------------------|
| Op1   | %MWx : L               |
| Op2   | %MWy : L oder %KWy : L |
| <b>x, y</b> Indizes des Speicherobjekts<br><b>L</b> müssen für Op1 und Op2 dieselben sein |                        |

Direkte Zeichenfolgen werden nicht akzeptiert.

### Abrufen der Länge einer Zeichenfolge

Verwenden Sie zum Abrufen der Länge einer Zeichenfolge die folgende Syntax:  $Op1 := LENGTH(Op2)$ .

Die folgende Tabelle zeigt die für Op1 und Op2 autorisierten Speicherobjekte:

| Parameter                               | Beschreibung           |
|---|------------------------|
| Op1                                     | %MWx                   |
| Op2                                     | %MWy : L oder %KWy : L |
| <b>x, y</b> Indizes des Speicherobjekts |                        |

Direkte Zeichenfolgen werden nicht akzeptiert.

### Verketten von zwei Zeichenfolgen

Verwenden Sie zum Verketten von zwei Zeichenfolgen die folgende Syntax: `Op1 := CONCAT(Op2, Op3)`.

Die folgende Tabelle zeigt die für Op1, Op2 und Op3 autorisierten Speicherobjekte:

| Parameter  | Beschreibung           |
|--|------------------------|
| Op1  | %MWx : L               |
| Op2  | %MWy : A oder %KWy : A |
| Op3  | %MWz : B oder %KWz : B |
| <b>x, y, z</b> Indizes des Speicherobjekts<br><br>SoMachine Basic validiert nicht, dass L eine ausreichende Größe für die Verkettung aufweist. Stellen Sie sicher, dass Op1 eine ausreichende Mindestlänge für die Operation aufweist. |                        |

Direkte Zeichenfolgen werden nicht akzeptiert.

Die folgende Tabelle zeigt den Verkettungsvorgang:

| Schritt | Beschreibung  |
|---------|---|
| 1       | Die Anwendung kopiert Op1 in Op2.   |
| 2       | Der Kopiervorgang wird angehalten, wenn: <ul style="list-style-type: none"> <li>• Das Endezeichen von Op2 wurde erreicht.</li> <li>• Der Op2 zugewiesene Speicherplatz wird kopiert. %S28 wird gesetzt. Siehe Beschreibung der Systembits.</li> <li>• Der gesamte Speicherplatz von Op1 ist belegt. %S28 wird gesetzt.</li> </ul> |
| 3       | Wenn der Speicherplatz von Op1 nicht belegt ist, dann setzt die Anwendung das Kopieren von Op3 in Op1 fort.   |
| 4       | Der Kopiervorgang wird angehalten, wenn: <ul style="list-style-type: none"> <li>• Das Endezeichen von Op3 wurde erreicht.</li> <li>• Der Op3 zugewiesene Speicherplatz wird kopiert. %S28 wird gesetzt.</li> <li>• Der gesamte Speicherplatz von Op1 ist belegt. %S28 wird gesetzt.</li> </ul>                                    |
| 5       | Durch die Anwendung wird sichergestellt, dass Op1 mit dem Endezeichen endet. Das letzte Zeichen von Op1 kann durch das Endezeichen ersetzt werden, wenn der Speicherplatz belegt ist.   |

### Vergleichen von zwei Zeichenfolgen

Verwenden Sie zum Vergleichen von zwei Zeichenfolgen die folgende Syntax: `Op1 := EQUAL_STR(Op2, Op3)`.

Die folgende Tabelle zeigt die für Op1, Op2 und Op3 autorisierten Speicherobjekte:

| Parameter                                  | Beschreibung           |
|--|------------------------|
| Op1  | %MWx                   |
| Op2  | %MWy : A oder %KWy : A |
| Op3  | %MWz : B oder %KWz : B |
| <b>x, y, z</b> Indizes des Speicherobjekts |                        |

Wenn die Anwendung ein unterschiedliches Zeichen erkennt, dann entspricht Op1 der Indexposition des ersten festgestellten unterschiedlichen Zeichens von links nach rechts.

Die folgende Tabelle zeigt Beispiele des Ergebnisses eines Zeichenfolgenvergleichs:

| Op2             | Op3             | Unterschiedliches Zeichen | Op1 |
|-----------------|-----------------|---------------------------|-----|
| azerty          | qwertz          | Der erste                 | 0   |
| 123456          | 124356          | Der dritte                | 2   |
| SoMachine Basic | SoMachine Basic | –                         | -1  |

Die folgende Tabelle zeigt den Vorgang eines Zeichenfolgenvergleichs:

| Aktion   | Und wenn  | Dann   |
|--|-----------|--|
| Die Anwendung erreicht das Endezeichen von Op2   | Op2 = Op3 | Op1 := -1  |
|  | Op2 ≠ Op3 | Op1 entspricht der Position des unterschiedlichen Zeichens.  |
| Die Anwendung findet ein unterschiedliches Zeichen vor Erreichen des Endes von Op2 oder Op3. | –         | Op1 entspricht der Position des unterschiedlichen Zeichens.  |
| Das Ende des Op2 oder Op3 zugewiesenen Speicherplatzes wurde erreicht                        | A ≠ B     | Op1 entspricht der Position des unterschiedlichen Zeichens und %S28 wird gesetzt. Siehe Beschreibung der Systembits. |
|  | A = B     | Op1 := -1 und %S28 werden gesetzt.   |

# Abschnitt 6.9

## Periodische Task

---

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema  | Seite |
|--|-------|
| Erstellen periodischer Tasks                         | 143   |
| Konfigurieren der Abfragedauer des periodischen Task | 145   |

## Erstellen periodischer Tasks

### Übersicht

Bei einem periodischen Task handelt es sich um ein Unterprogramm, in der Regel von kurzer Dauer, das regelmäßig (periodisch) verarbeitet wird. In SoMachine Basic wird dieses Unterprogramm als Freie POU (*siehe Seite 109*) implementiert. Unterprogramme können in allen von SoMachine Basic unterstützten Programmiersprachen geschrieben werden.

### Zuweisen eines Unterprogramms zu einer periodischen Task

| Schritt | Aktion   |
|---------|--|
| 1       | Erstellen Sie eine neue freie POU ( <i>siehe Seite 110</i> ) mit einer periodischen Task (Unterprogramm).  |
| 2       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 3       | <p>Fügen Sie mithilfe einer der folgenden Methoden ein neues Unterprogramm hinzu:</p> <ul style="list-style-type: none"> <li>• Wählen Sie die <b>Periodische Task</b> aus und klicken Sie auf  (Schaltfläche <b>Freie POU zuweisen</b>) in der Symbolleiste am oberen Rand der Registerkarte <b>Tasks</b>.</li> <li>• Klicken Sie mit der rechten Maustaste auf die <b>Periodische Task</b> und wählen Sie im Kontextmenü <b>Freie POU zuweisen</b> aus.</li> </ul> <p><b>Ergebnis:</b> Daraufhin wird das Fenster <b>Freie POU auswählen</b> angezeigt.</p> <div data-bbox="349 841 875 1291" style="border: 1px solid gray; padding: 10px; margin: 10px 0;"> <p style="text-align: center; color: green; font-weight: bold;">Freie POU auswählen <span style="float: right;">✕</span></p> <hr style="border: 1px solid green;"/> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0;"> <p>Periodischer Task (SR0)<br/>Freie POU_0 (SR1)</p> </div> <div style="text-align: center; margin-top: 20px;"> <input type="button" value="OK"/> <input type="button" value="Abbrechen"/> </div> </div> <p><b>HINWEIS:</b> Sie können eine freie POU direkt in der periodischen Task hinzufügen. Klicken Sie mit der rechten Maustaste auf die <b>Periodische Task</b> und wählen Sie im Kontextmenü <b>Freie POU hinzufügen</b> aus. In diesem Fall wird eine freie POU erstellt und der periodischen Task zugewiesen.</p> |

| Schritt | Aktion  |
|---------|---|
| 4       | <p>Wählen Sie eine Freie POU aus, die der periodischen Task zugewiesen werden soll, und klicken Sie auf <b>OK</b>.</p> <p><b>Ergebnis:</b> Das ausgewählte Unterprogramm wird der <b>Periodischen Task</b> zugewiesen und ist im Zweig <b>Freie POU</b> der Registerkarte <b>Tasks</b> nicht mehr verfügbar.</p> <p>Beispiel: Wenn die freie POU „Freie POU_0“, die das Unterprogramm <b>SR4</b> enthält, der periodischen Task zugewiesen wird, wird das Unterprogramm <b>Freie POU_0 (%SR4)</b> aus dem Zweig <b>Freie POU</b>s in den Zweig <b>Periodische Task</b> der Registerkarte <b>Tasks</b> verschoben.</p> |

### Entfernen eines Unterprogramms aus einer periodischen Task

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | <p>Entfernen Sie das Unterprogramm mit einer der folgenden Methoden aus der <b>Periodischen Task</b>:</p> <ul style="list-style-type: none"> <li>• Wählen Sie die <b>Periodische Task</b> aus und klicken Sie auf  (Schaltfläche <b>Zuweisung der freien POU aufheben</b>) in der Symbolleiste am oberen Rand der Registerkarte <b>Tasks</b>.</li> <li>• Klicken Sie mit der rechten Maustaste auf die <b>Periodische Task</b> und wählen Sie im Kontextmenü <b>Zuweisung der freien POU aufheben</b> aus.</li> </ul> <p><b>Ergebnis:</b> Das ausgewählte Unterprogramm wird aus der <b>Periodischen Task</b> entfernt und ist als freie POU im Zweig <b>Freie POU</b> der Registerkarte <b>Tasks</b> verfügbar.</p> |

## Konfigurieren der Abfragedauer des periodischen Task

### Vorgehensweise

Gehen Sie wie folgt vor, um die Abfragedauer des periodischen Task zu konfigurieren:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Wählen Sie das Element <b>Periodische Task</b> aus.<br><b>Ergebnis:</b> Im unteren mittleren Bereich des SoMachine Basic-Fensters wird daraufhin das Eigenschaftsfenster der <b>Periodischen Task</b> angezeigt. |
| 3       | Nehmen Sie an den Eigenschaften die gewünschten Änderungen vor.  |
| 4       | Klicken Sie auf <b>Übernehmen</b> , um die Änderungen zu speichern.  |

### Eigenschaften der periodischen Task

Legen Sie die **Periode** für die periodische Task mit einem Wert zwischen 1 und 255 ms fest. Der Standardwert beträgt 255 ms.

# Abschnitt 6.10

## Ereignis-Task

---

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                            | Seite |
|----------------------------------|-------|
| Übersicht über die Ereignistasks | 147   |
| Ereignisquellen                  | 148   |
| Ereignisprioritäten              | 149   |
| Betrachten von Ereignistasks     | 150   |

## Übersicht über die Ereignistasks

### Einführung

Eine Ereignistask:

- Ist ein Programmteil, der ausgeführt wird, wenn eine bestimmte Bedingung (Ereignisquelle) erfüllt ist.
- Besitzt eine höhere Priorität als das Hauptprogramm.
- Garantiert eine kurze Antwortzeit, wodurch die globale Antwortzeit des Systems verkürzt werden kann.

### Beschreibung eines Ereignisses

Ein Ereignis besteht aus folgenden Elementen:

- *Ereignisquelle*: Software- oder hardware-spezifische Bedingung, die das Programm bei Auslösung des Ereignisses unterbricht..
- *POU*: Eine unabhängige Programmeinheit (Unterprogramm), die mit einem Ereignis verknüpft ist.
- *Prioritätsstufe*: Eine Ereignissen zugewiesene Priorität, die die Reihenfolge deren Ausführung vorgibt.

## Ereignisquellen

### Übersicht

Es stehen 8 Ereignisquellen zur Verfügung:

- 4 in Verbindung mit ausgewählten physischen Eingängen der Steuerung
- 4 in Verbindung mit den Schwellenwerten des HSC-Funktionsbausteins

Eine Ereignisquelle ist stets mit einem einzigen Ereignis verknüpft. Sobald ein Ereignis ausgelöst wird, wird die Steuerung unterbrochen, welche dann das dem Ereignis zugeordnete Unterprogramm ausführt.

### Physische Eingangereignisse einer Steuerung

Die integrierten Digitaleingänge %I0.2, %I0.3, %I0.4 und %I0.5 einer Steuerung können als Ereignisquellen konfiguriert werden (Filtern muss deaktiviert sein).

Sie können für jede der Ereignisquellen das folgende ausführen:

- Auslösung eines Ereignisses bei steigender Flanke, fallender Flanke oder beidem
- Zuweisung einer Priorität für das Ereignis
- Identifizierung des dem Ereignis zugeordneten Unterprogramms

Weitere Informationen zur Konfiguration von Eingangereignissen finden Sie im *Programmierhandbuch* der Steuerung.

### Schwellenwert des Ausgangereignisses eines %HSC-Funktionsbausteins

Die Schwellenwertausgänge TH0 und TH1 des Funktionsbausteins %HSC können als Ereignisquellen verwendet werden.. Die Ausgänge TH0 und TH1 sind wie folgt eingestellt:

- TH0 = 0 und TH1 = 0, wenn der Wert unter dem Grenzwert S0 und S1 liegt
- TH0 = 1 und TH1 = 0, wenn der Wert über dem Grenzwert S0 und unter dem Grenzwert S1 liegt
- TH0 = 1 und TH1 = 1, wenn der Wert über dem Grenzwert S0 und S1 liegt

Sie können für jede der Ereignisquellen das folgende ausführen:

- Auslösung eines Ereignisses bei steigender Flanke, fallender Flanke oder beidem
- Zuweisung einer Priorität für das Ereignis.
- Identifizierung des dem Ereignis zugeordneten Unterprogramms

Eine steigende oder fallende Flanke an diesen Ausgängen kann eine Ereignisverarbeitung auslösen.

Weitere Informationen zur Konfiguration von Ausgangereignissen finden Sie im *Programmierhandbuch* der Steuerung.

## Ereignisprioritäten

### Ereignisprioritäten

Ereignisse können eine von 8 möglichen Prioritäten aufweisen, von der Priorität 7 (der niedrigsten Priorität) bis zur Priorität 0 (der höchsten Priorität).

Weisen Sie jeder Ereignisquelle eine Priorität zu. Zwei Ereignisse können nicht dieselbe Priorität haben. Die Reihenfolge der Ausführung ist dementsprechend von den relativen Prioritäten und der Reihenfolge ihrer Erkennung abhängig.

Ereignistasks unterbrechen sowohl die Master-Task- als auch die periodische Taskausführung. Weitere Informationen finden Sie unter Priorität von Ereignissen vor Master- und periodischen Tasks (*siehe Seite 97*).

**HINWEIS:** Beim Schreiben in globale Speicherbereiche bzw. bei der Einwirkung auf E/A-Werte ist besondere Vorsicht geboten, wenn Ereignistasks während der Ausführung anderer Tasks aufgerufen werden. Eine Änderung von Werten, die in den anderen Tasks zum Einsatz kommen, kann das logische Ergebnis dieser Tasks negativ beeinflussen.

 **WARNUNG**

**UNBEABSICHTIGTER GERÄTEBETRIEB**

Sie müssen alle Tasks (Master-Task, periodische und Ereignistasks) sowie die interaktive Wirkung, die diese aufeinander ausüben, einer umfassenden Prüfung und Validierung unterziehen, bevor Sie Ihre Anwendung in Betrieb nehmen.

**Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.**

Schlagen Sie Informationen zur Konfiguration der Priorität von Ereignistasks im *Programmierhandbuch* Ihrer Steuerung nach.

### Ereignisverwaltung

Jedes Mal, wenn eine mit einer Ereignisquelle verknüpfte Unterbrechung auftritt, wird folgende Sequenz ausgeführt:

| Schritt | Beschreibung  |
|---------|---|
| 1       | Ereignisunterbrechung tritt auf.  |
| 2       | Speicherung des Kontextes   |
| 3       | Ausführung der mit dem Ereignis verknüpften Programm-Section (Unterprogramm mit der Bezeichnung SRi:) |
| 4       | Aktualisieren der integrierten Ausgänge.  |
| 5       | Wiederherstellung des Kontextes.  |

## Betrachten von Ereignistasks

### Überblick

Ereignistasks werden in der Registerkarte **Konfiguration** angezeigt. Siehe Konfigurieren von Digitaleingängen.

Mithilfe von Systembits und -wörtern können Sie konfigurierte Ereignisquellen sowie die den Ereignissen zugeordneten Unterprogramme anzeigen und den aktuellen Status der Ereignisse prüfen.

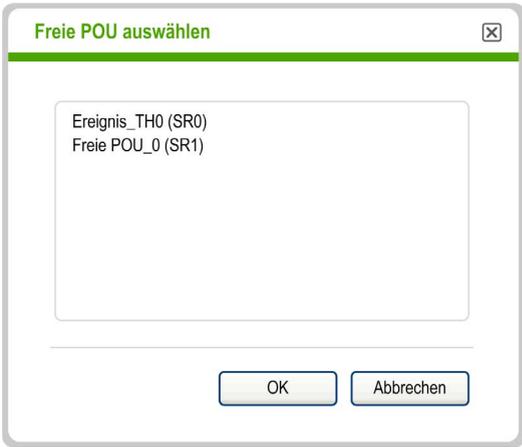
Gehen Sie zur Anzeige der Ereignissen zugeordneten Ereignisquellen und Unterprogrammen (freie POU's) vor wie folgt:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | <p>Wählen Sie <b>Ereignisse</b> aus:</p> <ul style="list-style-type: none"> <li><input type="checkbox"/> Ereignisse                             <ul style="list-style-type: none"> <li>%HSC0.TH0 :</li> <li>%HSC0.TH1 :</li> <li>%I0.2 :</li> </ul> </li> <li><input type="checkbox"/> %I0.3 : Freie POU_0                             <ul style="list-style-type: none"> <li>Rung0</li> </ul> </li> </ul> <p><b>HINWEIS:</b> Konfigurierte Ereignisquellen, denen noch kein Unterprogramm zugewiesen wurde, werden rot dargestellt.</p> |

**HINWEIS:** In einem Ereignisunterprogramm können nur integrierte Steuerungseingänge und -ausgänge verwendet werden.

### Zuweisen einer freien POU zu einer Ereignisquelle

Gehen Sie für die Zuweisung einer freien POU zu einer konfigurierten Ereignisquelle vor wie folgt:

| Schritt | Aktion  |
|---------|---|
| 1       | Erstellen Sie eine neue freie POU ( <i>siehe Seite 110</i> ) mit dem für das Ereignis zu verwendenden Unterprogramm.  |
| 2       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 3       | <p>Fügen Sie der Ereignisquelle mit einer der folgenden Methoden ein neues Unterprogramm hinzu:</p> <ul style="list-style-type: none"> <li>Wählen Sie die Ereignisquelle in der Liste <b>Ereignisse</b> aus und klicken Sie auf  (Schaltfläche <b>Freie POU zuweisen</b>) in der Symbolleiste am oberen Rand der Registerkarte <b>Tasks</b>.</li> <li>Klicken Sie mit der rechten Maustaste auf die Ereignisquelle in der Liste <b>Ereignisse</b> und wählen Sie im Kontextmenü <b>Freie POU zuweisen</b> aus.</li> </ul> <p><b>Ergebnis:</b> Daraufhin wird das Fenster <b>Freie POU auswählen</b> angezeigt.</p>  <p><b>HINWEIS:</b> Sie können eine freie POU direkt zur Ereignisquelle hinzufügen. Klicken Sie mit der rechten Maustaste auf die Ereignisquelle in der Liste <b>Ereignisse</b> und wählen Sie im Kontextmenü <b>Freie POU hinzufügen</b> aus. In diesem Fall wird eine freie POU erstellt und der Ereignisquelle zugewiesen.</p> |
| 4       | <p>Wählen Sie eine Freie POU aus, die der Ereignisquelle zugewiesen werden soll, und klicken Sie auf <b>OK</b>.</p> <p><b>Ergebnis:</b> Das ausgewählte Unterprogramm wird der Ereignisquelle zugewiesen und ist im Zweig <b>Freie POU</b> der Registerkarte <b>Tasks</b> nicht mehr verfügbar.<br/>         Beispiel: Wenn die freie POU „Freie POU_0“, die das Unterprogramm <b>SR1</b> enthält, der Ereignisquelle zugewiesen wird, wird das Unterprogramm <b>Freie POU_0 (%SR1)</b> aus dem Zweig <b>Freie POU</b>s in den Ereignisquellenzweig der Registerkarte <b>Tasks</b> verschoben.</p>  |

### Entfernen eines Unterprogramms aus einem Ereignis

Gehen Sie wie folgt vor, um die Zuordnung zwischen einem Unterprogramm und einer Ereignisquelle aufzuheben:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tasks</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | <p>Entfernen Sie das Unterprogramm mit einer der folgenden Methoden aus der Ereignisquelle:</p> <ul style="list-style-type: none"> <li>Wählen Sie die Ereignisquelle in der Liste <b>Ereignisse</b> aus und klicken Sie auf  (Schaltfläche <b>Zuweisung der freien POU aufheben</b>) in der Symbolleiste am oberen Rand der Registerkarte <b>Tasks</b>.</li> <li>Klicken Sie mit der rechten Maustaste auf die Ereignisquelle in der Liste <b>Ereignisse</b> und wählen Sie im Kontextmenü <b>Zuweisung der freien POU aufheben</b> aus.</li> </ul> <p><b>Ergebnis:</b> Das ausgewählte Unterprogramm wird aus der Ereignisquelle entfernt und ist als freie POU im Zweig <b>Freie POU</b>s der Registerkarte <b>Tasks</b> verfügbar.</p> |

### Prüfen von Ereignissen über Systembits und -wörter

Folgende Systembits werden zur Prüfung der Ereignisse verwendet:

| Systembit | Beschreibung   |
|-----------|--|
| %S38      | Zum Aktivieren (%S38 = 1) oder Deaktivieren (%S38 = 0) der Ereignisverarbeitung. |
| %S39      | Hiermit kann bestimmt werden, ob Ereignisse verloren gegangen sind.              |

Folgende Systemwörter werden zur Prüfung der Ereignisse verwendet:

| Systemwort | Beschreibung   |
|------------|--|
| %SW48      | Die Anzahl an ausgeführten Ereignissen seit dem letzten Kaltstart der Steuerung. |

Im Anschluss an einen Kaltstart oder nach dem Laden einer Anwendung werden die Werte von %S39 und %SW48 auf 0 zurückgesetzt und der Wert des Systembits %S38 wird auf seinen Initialstatus 1 gesetzt. Bei einem Warmstart bleiben die Werte unverändert.

# Abschnitt 6.11

## Verwenden von Tools

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                       | Seite |
|-----------------------------|-------|
| Meldungen                   | 154   |
| Animationstabellen          | 157   |
| Speicherobjekte             | 163   |
| Systemobjekte               | 168   |
| E/A-Objekte                 | 169   |
| Netzwerkobjekte             | 170   |
| Software-Objekte            | 171   |
| PTO-Objekte                 | 172   |
| Antriebsobjekte             | 173   |
| Kommunikationsobjekte       | 174   |
| Suchen und Ersetzen         | 175   |
| Querverweis                 | 177   |
| Symbolliste                 | 178   |
| Ansicht der Speichernutzung | 183   |

## Meldungen

### Überblick

Während das Programm bearbeitet wird, analysiert SoMachine Basic den Quellcode in der Registerkarte **Programmierung**.

SoMachine Basic Analysiert auch, ob die Schaltfläche **Kompilieren** ✓ in der Werkzeugleiste angeklickt ist.

Wenn Fehler oder Hinweise entdeckt werden, wird auf der Registerkarte **Programmierung** ein anklickbares Symbol angezeigt:



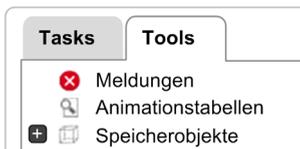
Das Anklicken des Symbols öffnet das Fenster ‚Meldungen‘.

Welches Symbol angezeigt wird, hängt von dem Schweregrad der Meldung ab:

| Symbol | Bedeutung  |
|--------|--|
|        | Warnung. Der Programmbaustein ist unvollständig. |
|        | Es wurde ein Syntaxfehler erkannt.               |

Wenn sowohl Hinweis- und Fehlermeldungen entdeckt werden, wird nur das Fehlersymbol angezeigt.

Das Symbol wird ebenfalls auf der Registerkarte **Tools** neben den **Meldungen** angezeigt:



## Anzeigen der Meldungen

Anzeigen einer Liste mit Hinweis- und Fehlermeldungen:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf das Symbol auf der Registerkarte <b>Programmierung</b> .<br>oder:<br>Klicken Sie auf <b>Tools → Meldungen</b> .<br>Im unteren mittleren Bereich des Fensters <b>Programmierung</b> wird eine Liste der Meldungen angezeigt.       |
| 2       | Klicken Sie im Bereich <b>Meldungen</b> auf die Schaltfläche <b>Hinweis</b> , um Hinweismeldungen anzuzeigen, oder auf <b>Fehler</b> , um Fehlermeldungen anzuzeigen. Klicken Sie erneut auf die Schaltfläche, um die Meldungsliste auszublenden. |

## Status des Programmbausteins

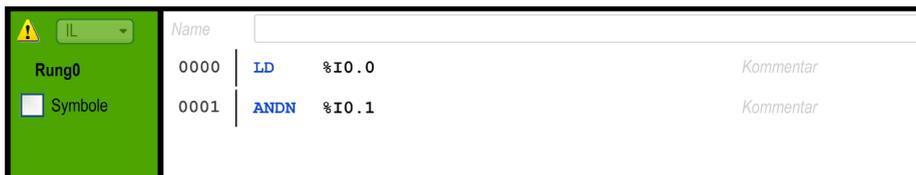
SoMachine Basic zeigt ebenfalls den Status der einzelnen Bausteine im Programm an.

Wenn der Programmbaustein syntaktisch gültig und vollständig ist, dann werden keine Meldungen angezeigt. Stattdessen erscheint ein grünes Häkchensymbol:



| Name |    |       |           |
|------|----|-------|-----------|
| 0000 | LD | %I0.0 | Kommentar |
| 0001 | ST | %Q0.0 | Kommentar |

Ein Hinweissymbol wird angezeigt, wenn der Programmbaustein unvollständig ist, beispielsweise weil er keine Endanweisung enthält, wie z. B. END, CALL oder Jump.



| Name |      |       |           |
|------|------|-------|-----------|
| 0000 | LD   | %I0.0 | Kommentar |
| 0001 | ANDN | %I0.1 | Kommentar |

Ein Fehlersymbol erscheint, wenn SoMachine Basic einen oder mehrere Syntaxfehler erkennt, die eine erfolgreiche Kompilierung des Programmbausteins verhindern würden:



| Name |     |       |           |
|------|-----|-------|-----------|
| 0000 | BLK | %SBR0 | Kommentar |
| 0001 | LD  | %S6   | Kommentar |
| 0002 | CU  |       | Kommentar |

Hinweis- und Fehlersymbole werden auch neben dem jeweiligen Namen der fehlerhaften Programmbausteine auf der Registerkarte **Tasks** angezeigt:

- ▣ ✖ 1 - M\_ZeroPressureAccumulator
  - ✖ Rung0
  - Rung1
  - Rung2
  - Rung3 - Rung\_1
  - ✖ Rung4 - Rung\_3
  - Rung5
  - Rung6 - Rung\_2
  - Rung7
  - ✖ Rung8

## Animationstabellen

### Überblick

Sie können in Animationstabellen Objekte hinzufügen zum:

- Anzeigen von Symbolen und Kommentaren, die zu Objekten gehören.
- Anzeigen und Ändern der Werte bestimmter Objekttypen in Echtzeit, sofern SoMachine Basic mit der Steuerung verbunden ist (Online-Modus).
- Auswählen von Objekten zur Anzeige im **Trace**-Fenster (*siehe Seite 246*).

Animationstabellen sind ein Komponent einer SoMachine Basic-Applikation und werden zusammen mit dem Programm als Teil der nicht-Programmdaten auf die Steuerung geladen. Somit können die in Animationstabellen gespeicherten Objekte abgerufen werden, wenn eine Anwendung zu einem späteren Zeitpunkt aus der Steuerung heruntergeladen wird.

| Animationstabelle                   |                                     |   |   |             |      |   |              |
|-------------------------------------|-------------------------------------|---|---|-------------|------|---|--------------|
| %I0.0                               |                                     | <input type="button" value="Hinzufügen"/> | <input type="button" value="Einfügen"/> | Zeitbasis   | 5    | <input type="button" value="Trace-Fenster öffnen"/> |              |
|                                     | Verwendet                           | Trace                                     | Adresse                                 | Symbol      | Wert | Forcieren   | Kommentar    |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            |   | %MW50                                   | ADDRESS_MEM | 0    |   |              |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |   | %MW610                                  | CONTROL_CMD | 0    |   |              |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            |   | %M16                                    | MODBUS_READ | 0    |   |              |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            |   | %MW61                                   | SPEED_VALUE | 0    |   |              |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |   | %MW40                                   | CMD         | 0    |   | Control Word |

Wenn Sie ein Objekt hinzufügen, das nicht in der Animationstabelle besteht, dann wird das Feld **Wert** mit einer roten Umrandung angezeigt. Wenn Sie beispielsweise %Q1.0 hinzufügen, ohne dass ein entsprechendes digitales Ausgangsmodul in der Konfiguration besteht.

|                          |                          |       |   |
|--------------------------|--------------------------|-------|---|
| <input type="checkbox"/> | <input type="checkbox"/> | %Q1.0 | <input style="border: 2px solid red;" type="text"/> |
| <input type="checkbox"/> | <input type="checkbox"/> | %M0   | 0   |

### Erstellen einer Animationstabelle

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tools</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Klicken Sie mit der rechten Maustaste auf <b>Animationstabellen</b> und wählen Sie im angezeigten Kontextmenü die Option <b>Neue Animationstabelle hinzufügen</b> aus.<br><b>Ergebnis:</b> Unter dem Bereich <b>Animationstabellen</b> im Fenster <b>Tools</b> wird eine neue Animationstabelle angezeigt und im unteren mittleren Bereich erscheint ein Eigenschaftsfenster. |

## Hinzufügen individueller Objekte in eine Animationstabelle

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tools</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Wählen Sie im Bereich <b>Animationstabellen</b> des Fensters <b>Tools</b> die zu konfigurierende Animationstabelle aus.<br><b>Ergebnis:</b> Das Eigenschaftsfenster wird im unteren mittleren Bereich des Fensters angezeigt.   |
| 3       | Um ein neues Objekt am Ende der Animationstabelle hinzuzufügen, geben Sie den Objektnamen in das Textfeld ein und drücken Sie die Eingabetaste bzw. klicken Sie auf <b>Hinzufügen</b> .<br>In einer Animationstabelle können folgende Objekte hinzugefügt werden: <ul style="list-style-type: none"> <li>• E/A-Objekte</li> <li>• Funktionsbausteinobjekte. Beispielsweise für einen Timer-Funktionsbaustein %TM0, %TM0.V, %TM0.P und %TM0.Q werden der Animationstabelle automatisch hinzugefügt.</li> <li>• Bitzeichenfolgen (Beispiel: %Mx:L, wobei L der Bitanzahl entspricht, einem Vielfachen von 8)</li> <li>• Worttabellen (Beispiel: %MWx:L, wobei L der Wortanzahl entspricht)</li> <li>• Wortbits (Beispiel: %MWx:X, wobei X dem Bit-Offset entspricht)</li> <li>• Netzwerkobjekte (%QWE, %IWE, %QWM, %IWM)</li> </ul> <b>HINWEIS:</b> Netzwerkobjekte sind nur verfügbar, wenn entweder der Ethernet/IP-Adapter ( <i>siehe Modicon M221, Logic Controller, Programmierhandbuch</i> ) der Steuerung aktiviert ist oder wenn <b>Modbus-Zuordnung</b> in der Modbus TCP-Konfiguration aktiviert ist. |
| 4       | Um ein neues Objekt direkt über einem bereits vorhandenen Objekt hinzuzufügen, wählen Sie die entsprechende Zeile in der Animationstabelle aus, geben Sie den Namen des hinzuzufügenden Objekts in das Textfeld ein und klicken Sie dann auf <b>Einfügen</b> .  |

Adressen von E/A-Objekten in Animationstabellen werden nach Konfigurationsänderungen nicht automatisch geändert. Beispielsweise wird %Q3.0 nicht automatisch zu %Q1.0 geändert, wenn sich die Position des entsprechenden Moduls in der Konfiguration ändert. Sie müssen alle Anpassungen berücksichtigen, die an E/A-Speicherzuweisungen in Ihrer Applikation vorgenommen werden und diese entsprechend aktualisieren.

### **WARNUNG**

#### **UNBEABSICHTIGTER GERÄTEBETRIEB**

Überprüfen und passen Sie nach Bedarf alle in der Anwendung verwendeten direkten E/A-Adressen an, nachdem Sie Änderungen an der Konfiguration vorgenommen haben.

**Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.**

Verifizieren und aktualisieren Sie Animationstabellen nach einer Konfigurationsänderung.

## Hinzufügen aller verwendeter Objekte in einem Programmbaustein an eine Animationstabelle

| Schritt | Aktion   |
|---------|--|
| 1       | Wenn mehr als eine (1) Animationstabelle besteht, wählen Sie eine Animationstabelle im Bereich <b>Animationstabellen</b> im Fenster <b>Tools</b> aus.<br><b>Ergebnis:</b> Das Eigenschaftsfenster der Animationstabelle wird im unteren mittleren Bereich des Fensters angezeigt.                                    |
| 2       | Wählen Sie das Fenster <b>Tasks</b> aus.   |
| 3       | Klicken Sie mit der rechten Maustaste auf einen Programmbaustein und wählen Sie im angezeigten Kontextmenü die Option <b>Programmbausteinobjekte zu aktueller Animationstabelle hinzufügen</b> aus.<br><b>Ergebnis:</b> Objekte, die im Programmbaustein verwendet werden, werden der Animationstabelle hinzugefügt. |

### HINWEIS:

- Der Programmbaustein darf keine erkannten Fehler enthalten (das Fehlersymbol  erscheint nicht).
- Es werden nur die ersten 64 im Programmbaustein verwendeten Objekte hinzugefügt (maximale Größe einer Animationstabelle).
- Wenn dasselbe Objekt mehr als einmal in einem Programmbaustein erscheint, wird nur das erste Auftreten zur Animationstabelle hinzugefügt.

## Eigenschaften der Animationstabelle

In der folgenden Tabelle werden die Eigenschaften von Animationstabellen beschrieben:

| Parameter        | Veränderbar       | Wert                | Beschreibung  |
|------------------|-------------------|---------------------|---|
| <b>Verwendet</b> | Nein              | True/False          | Gibt an, ob das Objekt derzeit in einem Programm verwendet wird.  |
| <b>Trace</b>     | Ja <sup>(1)</sup> | True/False          | Wählen Sie im Trace-Fenster ( <i>siehe Seite 246</i> ) das Objekt für den Tracing-Vorgang aus.  |
| <b>Adresse</b>   | Nein              | Objektadresse       | Zeigt die Adresse des Objekts an.   |
| <b>Symbol</b>    | Nein              | Ein gültiges Symbol | Der Name des diesem Objekt zugeordneten Symbols, sofern definiert.  |
| <b>Wert</b>      | Ja <sup>(2)</sup> | Aktueller Wert      | Der Wert eines Objekts.<br>Wenn der Objekttyp mit einem Lese-/Schreibzugriff ausgestattet ist und Sie sich im Online-Modus ( <i>siehe Seite 31</i> ) befinden, doppelklicken Sie und geben Sie, falls erforderlich, einen neuen Objektwert ein. Der Wert des Objekts wird dann in dem in der Steuerung ausgeführten Programm in Echtzeit aktualisiert.<br>Detaillierte Informationen hierzu finden Sie unter Ändern von Echtzeitwerten ( <i>siehe Seite 249</i> ) |

(1) Sie können bis zu 8 Objekte auswählen.

(2) Je nach Objekttyp und ob der Online-Modus aktiv ist.

| Parameter        | Veränderbar       | Wert  | Beschreibung  |
|------------------|-------------------|---|---|
| <b>Forcieren</b> | Ja <sup>(2)</sup> | <b>Auf 0 forcieren</b><br><b>Auf 1 forcieren</b><br><b>Nicht forciert</b> | Wird nur für digitale Ein- und Ausgänge angezeigt. Kann nur im Online-Modus ( <i>siehe Seite 31</i> ) bearbeitet werden. Ermöglicht Ihnen die bedarfsgerechte Forcierung des Ein- bzw. Ausgangswerts auf 0 oder 1.<br>Wählen Sie <b>Nicht forciert</b> aus, um jede derzeit auf die Adresse angewandte Forcierung zu entfernen.<br><b>HINWEIS:</b> Die Forcierung wird am Ende des Abfragezyklus durchgeführt. Die Abbildtabelle der Ausgänge kann jedoch aufgrund der Logik Ihres Programms geändert werden und kann sich, entgegen dem von Ihnen ausgewählten geforcten Zustand, in Animationstabellen und anderen Datenanzeigen befinden. Am Ende der Abfrage wird dies durch Ausführen des angeforderten geforcten Zustands korrigiert und der physikalische Ausgang wird diesen geforcten Zustand wiedergeben. |
| <b>Kommentar</b> | Nein              | Ein gültiger Kommentar  | Der diesem Objekt zugeordnete Kommentar, sofern definiert.  |

(1) Sie können bis zu 8 Objekte auswählen.  
 (2) Je nach Objekttyp und ob der Online-Modus aktiv ist.

### Konfigurieren von Elementen in einer Animationstabelle

Um ein Objekt in einer Animationstabelle zu suchen und nach Bedarf zu ersetzen, klicken Sie mit der rechten Maustaste auf das Objekt und wählen Sie **Suchen und Ersetzen** aus. Weitere Informationen finden Sie unter Suchen und Ersetzen (*siehe Seite 175*).

Um ein Objekt aus einer Animationstabelle zu entfernen, klicken Sie mit der rechten Maustaste auf das Objekt und wählen Sie die Option **Aus Animationstabelle entfernen** aus.

### Kopieren/Ausschneiden und Einfügen von vorhandenen Animationstabellen

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tools</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Wählen Sie unter <b>Animationstabellen</b> eine oder mehrere Animationstabellen aus, indem Sie die Strg-Taste gedrückt halten.  |
| 3       | Klicken Sie mit der rechten Maustaste auf eine der ausgewählten Animationstabellen unter <b>Animationstabellen</b> und wählen Sie <b>Animationstabelle kopieren</b> oder <b>Animationstabelle ausschneiden</b> aus. |

| Schritt | Aktion  |
|---------|---|
| 4       | <p>Gehen Sie zum Einfügen der Animationstabelle folgendermaßen vor:</p> <ul style="list-style-type: none"> <li>• Klicken Sie mit der rechten Maustaste auf <b>Animationstabellen</b> und wählen Sie <b>Animationstabelle einfügen</b> aus.</li> <li>• Klicken Sie mit der rechten Maustaste auf eine vorhandene Animationstabelle und wählen Sie <b>Animationstabelle einfügen</b> aus.</li> </ul> <p><b>Ergebnis:</b> Das Fenster <b>Bestätigung</b> wird angezeigt. Deaktivieren Sie das Kontrollkästchen, um die Symbole und Kommentare beizubehalten und klicken Sie anschließend auf <b>Ok</b>.</p> <p><b>Ergebnis:</b> Eine oder mehrere Animationstabellen werden am Ende der <b>Animationstabellen</b> oder nach der ausgewählten Animationstabelle hinzugefügt.</p> <p>Beim Kopieren/Einfügen einer Animationstabelle weist SoMachine Basic automatisch einen neuen Namen zu. Beispiel: <b>Animationstabelle_2</b> wird zu <b>Animationstabelle_2_0</b>.</p> |

Wenn Sie eine Animationstabelle in ein Projekt mit einer niedrigeren Funktionsebene (*siehe Seite 94*) einfügen, dann werden nur die von dieser Funktionsebene unterstützten Objektkonfigurationen kopiert.

Wenn die in der eingefügten Animationstabelle enthaltenen Symbole bereits im Projekt verwendet werden, dann ersetzt SoMachine Basic das eingefügte Symbol.

### Löschen einer Animationstabelle

| Schritt | Aktion   |
|---------|--|
| 1       | <p>Klicken Sie im Bereich <b>Animationstabellen</b> des Fensters <b>Tools</b> mit der rechten Maustaste auf die Animationstabelle, die Sie löschen möchten, und wählen Sie dann die Option <b>Animationstabelle löschen</b> aus.</p> |

### Umbenennen einer Animationstabelle

| Schritt | Aktion   |
|---------|--|
| 1       | <p>Klicken Sie im Bereich <b>Animationstabellen</b> des Fensters <b>Tools</b> mit der rechten Maustaste auf die Animationstabelle, die Sie umbenennen möchten, und wählen Sie dann die Option <b>Animationstabelle umbenennen</b> aus.</p> |
| 2       | <p>Geben Sie einen neuen Namen für die Animationstabelle ein und drücken Sie die Eingabetaste.</p>   |

## Exportieren von Animationstabellen

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tools</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Gehen Sie zum Auswählen der Animationstabelle folgendermaßen vor: <ul style="list-style-type: none"> <li>● Klicken Sie mit der rechten Maustaste auf <b>Animationstabellen</b>.</li> <li>● Wählen Sie eine oder mehrere vorhandene Animationstabellen aus, indem Sie die Strg-Taste gedrückt halten und führen Sie anschließend einen Rechtsklick aus.</li> </ul> |
| 3       | Klicken Sie auf <b>Animationstabelle exportieren</b> .  |
| 4       | Wählen Sie einen Ordner aus und speichern Sie die Animationstabellen (.smbf).   |

## Importieren von Animationstabellen

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tools</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Gehen Sie zum Auswählen der Animationstabelle folgendermaßen vor: <ul style="list-style-type: none"> <li>● Klicken Sie mit der rechten Maustaste auf <b>Animationstabellen</b>.</li> <li>● Klicken Sie mit der rechten Maustaste auf eine vorhandene Animationstabelle.</li> </ul> |
| 3       | Klicken Sie auf <b>Animationstabelle importieren</b> .   |
| 4       | Navigieren Sie zu dem Ordner, der die Animationstabellendatei enthält (*.smbf).  |
| 5       | Doppelklicken Sie auf die Animationstabelle.<br><b>Ergebnis:</b> Die Animationstabelle wird am Ende der <b>Animationstabellen</b> oder vor der ausgewählten vorhandenen Animationstabelle hinzugefügt.   |

Wenn die in der importierten Animationstabelle enthaltenen Symbole bereits im Projekt verwendet werden, dann ersetzt SoMachine Basic das importierte Symbol.

## Öffnen des Trace-Fensters

| Schritt | Aktion  |
|---------|---|
| 1       | Wählen Sie in der Spalte <b>Trace</b> einer Animationstabelle bis zu 8 Objekte aus.   |
| 2       | Stellen Sie eine Verbindung ( <i>siehe Seite 265</i> ) zur Steuerung her oder starten Sie den Simulator ( <i>siehe Seite 286</i> ).                                 |
| 3       | Wählen Sie in der Liste <b>Zeitbasis</b> einen Wert aus. Dieser Wert gibt die Aktualisierungsfrequenz des Trace-Fensters ( <i>siehe Seite 246</i> ) in Sekunden an. |
| 4       | Klicken Sie auf <b>Trace</b> .<br>Das <b>Trace</b> -Fenster wird angezeigt.   |

## Speicherobjekte

### Überblick

Speicherobjekte beinhalten Folgendes:

- Speicherbits
- Speicherwörter
- Konstantwörter

### Auswählen des Speicherzuweisungsmodus

Vor der Anzeige oder Aktualisierung der Eigenschaften von Speicherobjekten müssen Sie den zu verwendenden Speicherzuweisungsmodus (*siehe Seite 83*) auswählen.

### Eigenschaften von Speicherbits

In der folgenden Tabelle werden die verschiedenen Parameter im Fenster **Speicherbits** beschrieben:

| Parameter        | Veränderbar | Wert                   | Standardwert | Beschreibung   |
|------------------|-------------|------------------------|--------------|--|
| <b>Verwendet</b> | Nein        | True/False             | False        | Gibt an, ob das Speicherbit derzeit in einem Programm verwendet wird.  |
| <b>Adresse</b>   | Nein        | Siehe Bitobjekte       | -/-          | Gibt die Adresse des Speicherbits an, wobei <i>x</i> der Anzahl der von der Steuerung unterstützten Speicherbits entspricht. |
| <b>Symbol</b>    | Ja          | Ein gültiges Symbol    | <i>Kein</i>  | Ermöglicht Ihnen die Zuordnung eines Symbols für dieses Speicherbit.   |
| <b>Wert</b>      | Ja          | Siehe Bitobjekte       | 0            | Wert des Speicherbits.   |
| <b>Kommentar</b> | Ja          | Ein gültiger Kommentar | Kein         | Ermöglicht Ihnen die Zuordnung eines Kommentars für dieses Speicherbit.  |

### Eigenschaften von Speicherwörtern



Wählen Sie zuerst den Typ des Speicherworts aus, dessen Eigenschaften Sie anzeigen möchten:

- **%MW**: Speicherwörter
- **%MD**: Doppelwörter
- **%MF**: Gleitkommawörter

In der folgenden Tabelle werden die Eigenschaften der **Speicherwörter** beschrieben:

| Parameter                  | Veränderbar | Wert  | Standardwert | Beschreibung   |
|----------------------------|-------------|---|--------------|--|
| <b>Verwendet</b>           | Nein        | True/False  | False        | Gibt an, ob das Speicherwort derzeit in einem Programm verwendet wird.   |
| <b>Gleichung verwendet</b> | Nein        | True/False  | False        | Äquivalent verwendet. Zeigt an, ob ein Teil des Speicherbereichs des Speicherworts momentan verwendet wird. Siehe Möglichkeit der Überlappung zwischen Objekten ( <i>siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen</i> ). |
| <b>Adresse</b>             | Nein        | Siehe Wortobjekte ( <i>siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen</i> ) | -/-          | Gibt die Adresse des Speicherworts an.   |
| <b>Symbol</b>              | Ja          | Ein gültiges Symbol   | <i>Kein</i>  | Ermöglicht Ihnen die Zuordnung eines Symbols für dieses Speicherwort.  |
| <b>Wert</b>                | Ja          | Siehe Wortobjekte ( <i>siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen</i> ) | 0            | Wert des Speicherworts.  |
| <b>Kommentar</b>           | Ja          | Ein gültiger Kommentar  | Kein         | Ermöglicht Ihnen die Zuordnung eines Kommentars für dieses Speicherwort.   |

### Eigenschaften von Konstantwörtern

Eigenschaften der Konstantwörter

%KW
  %KD
  %KF

Wählen Sie zuerst den Typ des Konstantworts aus, dessen Eigenschaften Sie anzeigen möchten:

- **%KW**: Konstantwörter
- **%KD**: Konstantwörter im Doppelformat
- **%KF**: Konstantwörter im Gleitkommaformat

In der folgenden Tabelle werden die verschiedenen Parameter im Fenster **Konstantwörter** beschrieben:

| Parameter                  | Veränderbar | Wert  | Standardwert       | Beschreibung  |
|----------------------------|-------------|---|--------------------|---|
| <b>Verwendet</b>           | Nein        | True/False  | False              | Gibt an, ob das Konstantwort derzeit in einem Programm verwendet wird.  |
| <b>Gleichung verwendet</b> | Nein        | True/False  | False              | Äquivalent verwendet. Zeigt an, ob ein Teil des Speicherbereichs des konstanten Worts momentan verwendet wird. Siehe Möglichkeit der Überlappung zwischen Objekten ( <i>siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen</i> ). |
| <b>Adresse</b>             | Nein        | Siehe Wortobjekte ( <i>siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen</i> )                                 | -/-                | Gibt die Adresse des Konstantworts an.  |
| <b>Symbol</b>              | Ja          | Ein gültiges Symbol   | Kein               | Ermöglicht Ihnen die Zuordnung eines Symbols für dieses Konstantwort.   |
| <b>Dezimal</b>             | Ja          | Dezimale Darstellung des Werts. Siehe Wortobjekte ( <i>siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen</i> ) | 0                  | Der Dezimalwert des Konstantworts.  |
| <b>Binär</b>               | Ja          | Binäre Darstellung des Werts. Siehe Wortobjekte ( <i>siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen</i> )   | 2#0000000000000000 | Der Binärwert des Konstantworts.  |

| Parameter   | Veränderbar | Wert   | Standardwert    | Beschreibung   |
|-------------|-------------|--|-----------------|--|
| Hexadezimal | Ja          | Hexadezimale Darstellung des Werts. Siehe Wortobjekte (siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen) | 16#0000         | Der Hexadezimalwert des Konstantworts.                                   |
| ASCII       | Ja          | ASCII-Darstellung des Werts. Siehe Wortobjekte (siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen)        | keine Bedeutung | Der ASCII-Wert des Konstantworts.  |
| Kommentar   | Ja          | Ein gültiger Kommentar   | Kein            | Ermöglicht Ihnen die Zuordnung eines Kommentars für dieses Konstantwort. |

### Exportieren/Importieren der Eigenschaften der Konstantwörter

Sie können in eine CSV-Datei exportieren und die Eigenschaften von **Adresse**, **Symbol**, **Wert** und **Kommentar** im Offline- oder Online-Modus importieren.

Exportieren der Eigenschaften der Konstantwörter:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tools</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | Klicken Sie auf <b>Speicherobjekte → Konstantwörter</b> .   |
| 3       | Klicken Sie in den <b>Eigenschaften der Konstantwörter</b> auf <b>Exportieren</b> .<br><b>Ergebnis:</b> Das Fenster <b>Konstanten exportieren</b> wird angezeigt.   |
| 4       | Im Fenster <b>Konstanten exportieren</b> :<br><ol style="list-style-type: none"> <li>1. Wählen Sie den <b>Exporttyp</b> aus.</li> <li>2. Wählen Sie den <b>Dateipfad</b> aus, indem Sie auf  klicken.</li> <li>3. Geben Sie Folgendes ein: <ul style="list-style-type: none"> <li><input type="radio"/> Den <b>Dateinamen</b></li> <li><input type="radio"/> Den <b>Ersten Index</b> (numerisch)</li> <li><input type="radio"/> Den <b>Letzten Index</b> (numerisch)</li> </ul>                     Der <b>Erste Index</b> muss kleiner oder gleich dem <b>Letzten Index</b> sein.                 </li> </ol> |

| Schritt | Aktion  |
|---------|---|
| 5       | Klicken Sie auf  <b>Exportoptionen</b> , um die Exportparameter zu ändern.<br><b>1.</b> Wählen Sie <b>Kopfzeilen</b> aus, wenn der Name der Kopfzeilen angezeigt werden sollen.<br><b>2.</b> Wählen Sie als Trennzeichen <b>Semikolon</b> oder <b>Komma</b> aus. |
| 6       | Klicken Sie auf <b>Exportieren</b> .  |

Importieren der Eigenschaften der Konstantwörter:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Registerkarte <b>Tools</b> im linken Bereich des Fensters <b>Programmierung</b> .  |
| 2       | Klicken Sie auf <b>Speicherobjekte → Konstantwörter</b> .  |
| 3       | Klicken Sie in den <b>Eigenschaften der Konstantwörter</b> auf <b>Importieren</b> .<br><b>Ergebnis:</b> Das Fenster <b>Konstanten importieren</b> wird angezeigt.  |
| 4       | Klicken Sie auf  und navigieren Sie zu dem Ordner, der die Datei (*.csv) enthält und doppelklicken Sie auf die Datei.   |
| 5       | Um die Importparameter zu ändern, klicken Sie auf  <b>Importoptionen</b> und wählen Sie das in der .csv-Datei verwendete Trennzeichen aus: <b>Semikolon</b> oder <b>Komma</b> . |
| 6       | Klicken Sie auf <b>Importieren</b> .   |

Bei doppelten Werten wird der letzte doppelte Wert importiert.

## Systemobjekte

### Überblick

Systemobjekte sind spezifisch für die Logiksteuerung. Ausführliche Informationen finden Sie im *Programmierhandbuch* Ihrer Logiksteuerung.

## E/A-Objekte

### Übersicht

Die folgenden Objekttypen sind hardwarespezifisch und von der jeweils verwendeten Steuerung abhängig:

- Digitale Ein- und Ausgänge
- Analoge Ein- und Ausgänge
- Erweiterte Funktionsbausteine, wie z. B. schnelle Zähler, Hochgeschwindigkeitszähler und Impulsgeneratoren

Ausführliche Informationen finden Sie im *Programmierhandbuch* und im *Bibliothekshandbuch zu erweiterten Funktionen* Ihrer Logiksteuerung.

## Netzwerkobjekte

### Beschreibung

Netzwerkobjekte werden verwendet, um über EtherNet/IP, Modbus TCP oder Modbus Serial-E/A-Scanner zu kommunizieren.

Für die EtherNet/IP-Kommunikation sind folgende zwei Typen von Netzwerkobjekten verfügbar:

- %QWE: Input Assembly
- %IWE: Output Assembly

Für die Modbus TCP-Kommunikation sind folgende zwei Typen von Netzwerkobjekten verfügbar:

- %QWM: Eingangsregister
- %IWM: Ausgangsregister

Die folgenden Typen von Netzwerkobjekten werden für den Modbus Serial-E/A-Scanner verwendet:

- %IN: Digitaleingänge (E/A-Scanner)
- %QN: Digitalausgänge (E/A-Scanner)
- %IWN: Eingangsregister (E/A-Scanner)
- %QWN: Ausgangsregister (E/A-Scanner)
- %IWNS: Netzwerkdiagnosecodes des E/A-Scanners

**HINWEIS:** Verweise auf Ein- und Ausgänge erfolgen vom Standpunkt des EtherNet/IP-Masters oder des Modbus TCP-Clients aus.

Weitere Informationen zur Konfiguration von Netzwerkobjekten finden Sie im Programmierhandbuch Ihrer Steuerung.

## Software-Objekte

### Übersicht

SoMachine Basic unterstützt folgende generische Softwareobjekte:

| Objekt               | Beschreibung   |
|----------------------|--|
| Timer                | Timer werden zur Angabe eines Zeitraums vor einer Aktion verwendet, z. B. der Auslösung eines Ereignisses.   |
| Zähler               | Zähler ermöglichen die Auf- und Abwärtszählung von Ereignissen.  |
| Meldungen            | Meldungen ermöglichen die Kommunikation mit externen Geräten.  |
| LIFO/FIFO-Register   | Speicherbaustein, der bis zu 16 Wörter mit je 16 Bits nach dem FIFO- oder LIFO-Prinzip aufnehmen kann.   |
| Drums                | Ein Drum-Register funktioniert vom Prinzip her ähnlich wie eine elektromechanische Trommelsteuerung, die nach externen Ereignissen von Schritt zu Schritt wechselt. Bei jedem Schritt gibt der höchste Punkt der Nocke einen Befehl ab, der von der Steuerung ausgeführt wird. |
| Schiebebitregister   | Diese Register ermöglichen eine Verschiebung binärer Datenbits (0 oder 1) nach links oder rechts.  |
| Schrittzähler        | Schrittzähler stellen eine Reihe von Schritten bereit, denen Aktionen zugewiesen werden können.  |
| Zeitplan-Bausteine   | Diese Bausteine werden zur Steuerung von Aktionen zu einem über Monat, Tag und Uhrzeit vordefinierten Zeitpunkt verwendet.   |
| RTC                  | Wird zum Lesen von Datum und Uhrzeit von RTC oder zum Aktualisieren von RTC in der Logiksteuerung mit benutzerdefinierten Werten für Datum und Uhrzeit verwendet.  |
| PID                  | Ein PID übernimmt die Regelung der PID-Funktion (Proportional-Integral-Differential).  |
| Datenprotokollierung | Ermöglicht das permanente Speichern von Daten aus Objekten oder Zeichenfolgen.   |
| Grafcet-Schritte     | Listet die Grafcet-Bit-Adressvariablen (%Xi) auf, um Symbole oder Kommentare hinzuzufügen oder ändern zu können.   |

Diese Funktionsbausteine werden im SoMachine Basic – Allgemeine Funktionen Bibliothekshandbuch (*siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen*) beschrieben.

### Auswählen des Speicherzuweisungsmodus

Vor der Anzeige oder Aktualisierung der Eigenschaften von Softwareobjekten müssen Sie den zu verwendenden Speicherzuweisungsmodus (*siehe Seite 83*) auswählen.

## PTO-Objekte

### Überblick

Die PTO-Objekte stellen die Funktionsbausteine bereit, die zur Programmierung der PTO-Funktionen verwendet werden. Die PTO-Funktionsbausteine sind in folgende Kategorien unterteilt:

- **Bewegungs-Task-Tabellen**  
Ermöglicht die Konfiguration individueller PTO-Bewegungen in einer geordneten Reihenfolge und die Visualisierung eines geschätzten globalen Bewegungsprofils.
- **Bewegung**  
Diese Funktionsbausteine steuern Bewegungen der Achse. Beispiel: Spannungszufuhr zur Achse, Bewegung der Achse usw.
- **Administrativ**  
Diese Funktionsbausteine steuern Status und Diagnose der Achsenbewegung. Beispiel: Status und Wert der tatsächlichen Geschwindigkeit, tatsächliche Position, erkannte Fehler der Achsensteuerung usw.

Ausführliche Informationen zu den PTO-Funktionsbausteinen können Sie im *Bibliothekshandbuch zu erweiterten Funktionen* Ihrer Steuerung nachschlagen.

## Antriebsobjekte

### Überblick

Antriebsobjekte steuern ATV-Antriebe und andere auf dem Modbus Serial-E/A-Scanner konfigurierte Objekte. Oder Modbus TCP IOScanner.

Schlagen Sie Informationen im *Bibliothekshandbuch zu erweiterten Funktionen* für Ihren Logic Controller nach.

## Kommunikationsobjekte

### Überblick

Kommunikationsobjekte werden zur Kommunikation mit Modbus-Geräten, zum Senden/Empfangen von Meldungen im Zeichenmodus (ASCII) und zum Senden/Empfangen von SMS-Mitteilungen verwendet.

Weitere Einzelheiten finden Sie im Kapitel Kommunikationsobjekte.

## Suchen und Ersetzen

### Übersicht

Über die Funktion „Suchen und Ersetzen“ können Sie alle in einem Programm verwendeten Instanzen eines Objekts identifizieren und nach Bedarf durch ein anderes Objekt ersetzen.

### Suchen und Ersetzen von Elementen

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tools</b> im linken Bereich des Fensters <b>Programmierung</b> . Sie können die Funktion „Suchen und Ersetzen“ auch von anderen Stellen in SoMachine Basic aus aufrufen, z. B. über einen Rechtsklick auf einen Eintrag in einer Animationstabelle ( <i>siehe Seite 157</i> ) und die anschließende Auswahl der Option <b>Suchen und Ersetzen</b> .  |
| 2       | <p>Das Fenster <b>Suchen und Ersetzen</b> kann mit jeder der folgenden Methoden aufgerufen werden:</p> <ul style="list-style-type: none"> <li>● Klicken Sie auf <b>Suchen und Ersetzen</b> auf der Registerkarte <b>Tools</b> des Fensters <b>Programmierung</b>.</li> <li>● Klicken Sie mit der rechten Maustaste auf einen Programmbaustein bzw. ein markiertes Element in dem Programmbaustein und klicken Sie im daraufhin angezeigten Kontextmenü auf <b>Suchen und Ersetzen</b>.</li> <li>● Klicken Sie mit der rechten Maustaste auf eine Zeile im Eigenschaftsfenster eines beliebigen Objekts und klicken Sie dann im Kontextmenü auf <b>Suchen und Ersetzen</b>.</li> </ul> <p>Diese Abbildung zeigt das Fenster <b>Suchen und Ersetzen</b>:</p>  |
| 3       | <p>Geben Sie im Feld <b>Suchen</b> den Objekt- oder Symbolnamen ein, der gesucht werden soll. Das Feld <b>Suchen</b> ist bereits ausgefüllt, wenn die Suche durch Rechtsklicken auf ein markiertes Element in einem Programmbaustein oder auf ein Element in dem Eigenschaftsfenster eines Objekts gestartet wurde.</p> <p>Dabei können Sie folgende Platzhalterzeichen verwenden:</p> <ul style="list-style-type: none"> <li>● Sternchen (*): Ersetzt 0 oder mehr Zeichen in einem Suchbegriff. Beispielsweise würde <code>%MW1*</code> sowohl <code>%MW1</code> als auch <code>%MW101</code> finden.</li> <li>● Fragezeichen (?): Ersetzt genau 1 Zeichen in einem Suchbegriff. Beispiel: Mit <code>COIL?2</code> würde <code>COIL12</code> finden, aber nicht <code>COIL012</code>.</li> </ul>   |
| 4       | Im Feld <b>Ersetzen</b> können Sie nach Bedarf den Namen eines Ersatzobjekts bzw. -symbols eingeben.  |
| 5       | Wählen Sie <b>Programm</b> aus, um das Element im Quellcode des aktuellen Programms zu suchen. Wählen Sie <b>Kommentare</b> aus, um das Element in den Kommentaren zum Programm zu suchen.  |

| Schritt | Aktion  |           |                  |      |       |        |        |       |        |           |
|---------|---|-----------|------------------|------|-------|--------|--------|-------|--------|-----------|
| 6       | <p>Klicken Sie auf <b>Suchen</b> oder <b>Ersetzen</b>. Sie können auch die Eingabetaste (ENTER) drücken, um die Suche zu starten.</p> <p>Die Schaltfläche <b>Ersetzen</b> ist nur dann zur Auswahl verfügbar, wenn im Feld <b>Ersetzen</b> der Name des Ersatzobjekts bzw. -symbols angegeben ist.</p> <p>Alle gefundenen Elemente werden in der Liste <b>Ergebnisse</b> aufgeführt:</p> <p>Ergebnis <input type="checkbox"/> Symbole anzeigen</p> <table border="1"> <thead> <tr> <th>POU</th> <th>Programmbaustein</th> <th>Code</th> </tr> </thead> <tbody> <tr> <td>POU_0</td> <td>Rung_0</td> <td>%Q0.0</td> </tr> <tr> <td>POU_0</td> <td>Rung_1</td> <td>LD %Q0.0</td> </tr> </tbody> </table> | POU       | Programmbaustein | Code | POU_0 | Rung_0 | %Q0.0  | POU_0 | Rung_1 | LD %Q0.0  |
| POU     | Programmbaustein  | Code      |                  |      |       |        |        |       |        |           |
| POU_0   | Rung_0  | %Q0.0     |                  |      |       |        |        |       |        |           |
| POU_0   | Rung_1  | LD %Q0.0  |                  |      |       |        |        |       |        |           |
| 7       | <p>Wählen Sie nach Bedarf die Option <b>Symbole anzeigen</b> aus, um stattdessen alle für Objekte definierten Symbole anzuzeigen:</p> <p>Ergebnis <input checked="" type="checkbox"/> Symbole anzeigen</p> <table border="1"> <thead> <tr> <th>POU</th> <th>Programmbaustein</th> <th>Code</th> </tr> </thead> <tbody> <tr> <td>POU_0</td> <td>Rung_0</td> <td>OUTPUT</td> </tr> <tr> <td>POU_0</td> <td>Rung_1</td> <td>LD OUTPUT</td> </tr> </tbody> </table>   | POU       | Programmbaustein | Code | POU_0 | Rung_0 | OUTPUT | POU_0 | Rung_1 | LD OUTPUT |
| POU     | Programmbaustein  | Code      |                  |      |       |        |        |       |        |           |
| POU_0   | Rung_0  | OUTPUT    |                  |      |       |        |        |       |        |           |
| POU_0   | Rung_1  | LD OUTPUT |                  |      |       |        |        |       |        |           |
| 8       | <p>Klicken Sie auf einen der aufgeführten Ergebnisse, um direkt zur entsprechenden Codezeile im Programm zu springen.</p>   |           |                  |      |       |        |        |       |        |           |

## Querverweis

### Überblick

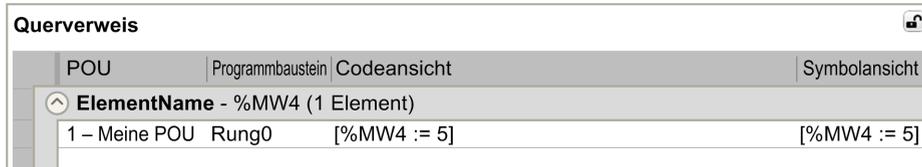
Die Querverweis-Ansicht ermöglicht das Anzeigen des in einer POU enthaltenen Programms. Wenn ein Objekt von einem anderen Objekt derselben POU abhängig ist, dann werden die entsprechenden Programmbausteine angezeigt.

Die Querverweis-Ansicht ist im Offline- und Online-Modus verfügbar.

### Anzeigen der Querverweis-Ansicht

Um die Querverweis-Ansicht anzuzeigen, klicken Sie auf **Programmierung** → **Tools** → **Querverweis** und wählen Sie anschließend ein oder mehrere Objekte im Aktionsbereich aus.

### Querverweis-Ansicht



Die folgende Tabelle zeigt das Element der Querverweis-Ansicht:

| Element                 | Beschreibung  |
|-------------------------|---|
| <b>POU</b>              | Name der POU, die das Objekt enthält.               |
| <b>Programmbaustein</b> | Name des Programmbausteins, der das Objekt enthält. |
| <b>Codeansicht</b>      | Programmiercode des Objekts.                        |
| <b>Symbolansicht</b>    | Symbol des Objekts.                                 |

## Symbolliste

### Übersicht

Sie können eine Liste aller Symbole anzeigen, die Objekte in Ihrem Programm zugeordnet wurden. Es werden alle Objekte mit Symbolen angezeigt, mit Ausnahme von Symbolen, die automatisch mit Systembits (%S) und Systemwörtern (%SW) assoziiert sind. Sie können Symbole und Kommentare auf Systembits (%S) und Systemwörtern (%SW) überschreiben, indem Sie die Eigenschaften der Systemobjekte verwenden oder durch einen Import Ihrer eigenen Symbolliste (siehe unten). Überschriebene Symbole erscheinen dann in der Symbolliste.

Unter Definieren und Verwenden von Symbolen (*siehe Seite 80*) wird die Erstellung von Symbolen und deren Verwendung in Programmen beschrieben.

### Symbolliste

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tools</b> im linken Bereich des Fensters <b>Programmierung</b> .   |
| 2       | <p>Klicken Sie auf <b>Symbolliste</b>.</p> <p><b>Ergebnis:</b> Das Fenster <b>Symbolliste</b> wird angezeigt. Für jedes Element werden folgende Informationen angezeigt:</p> <ul style="list-style-type: none"> <li>● <b>Verwendet:</b> Ob das Symbol derzeit im Programm verwendet wird.</li> <li>● <b>Adresse</b> Die Adresse des Objekts, dem das Symbol zugeordnet ist.</li> <li>● <b>Symbol:</b> Der Name des Symbols.</li> <li>● <b>Kommentar:</b> Der diesem Objekt zugeordnete Kommentar, sofern definiert.</li> <li>● <b>Symbol zuweisen:</b> SoMachine Basic weist automatisch jedem Objekt, das ohne Symbol verwendet wird, ein Symbol der folgenden Typen zu: %M, %MW, %MD, %MF, %S, %SW, , %KW, %KD, %I, %IW, %Q und %QW.</li> <li>● <b>Standardsymbole löschen:</b> Löschen von zugewiesenen Standardsymbolen.</li> <li>● <b>Importieren:</b> Importieren von Symbolen.</li> <li>● <b>Exportieren:</b> Exportieren von Symbolen.</li> </ul> |

### Erstellen von Standardsymbolen

Gehen Sie wie folgt vor, um Standardsymbole zu erstellen:

#### 1. Klicken Sie auf **Symbol zuweisen**.

**Ergebnis:** Standardsymbole werden allen Speicherobjekten zugewiesen, (%M, %MW, %MD, %MF, %S, %SW, %KW, %KD, %KF, %I, %IW, %Q, %QW) die im Programm verwendet werden und für die noch keine Symbole definiert wurden.

Symbole werden wie folgt bezeichnet: `symbolname = objectname_i`, wobei `objectname` dem Objekttyp ohne % entspricht und `i` den Index des Objekts darstellt.

Beispiel: Im Programm werden die folgenden Objekte verwendet, für die jedoch noch keine Symbole definiert wurden:

| Objekt | Zugewiesenes Symbol |
|--------|---------------------|
| %MW0   | MW_0                |
| %MW2   | MW_2                |
| %M0    | M_0                 |

### Löschen von Standardsymbolen

Gehen Sie wie folgt vor, um Standardsymbole zu löschen:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf <b>Standardsymbole löschen</b> .  |
| 2       | Klicken Sie im angezeigten Bestätigungsfenster auf <b>Ja</b> .<br><b>Ergebnis:</b> Alle zugewiesenen Standardsymbole werden gelöscht. |

**HINWEIS:** Wenn ein Objekt mit einem zugewiesenen Standardsymbol im Programm nicht länger verwendet wird, dann behält es sein Standardsymbol bei.

### Importieren von Symbolen

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie entweder auf die Schaltfläche <b>Importieren</b> , oder klicken Sie mit der rechten Maustaste auf eine beliebige Stelle in der Symbolliste, und wählen Sie die Option <b>Symbole importieren</b> aus.<br><b>Ergebnis:</b> Daraufhin wird das Fenster <b>Symbole importieren</b> angezeigt. |
| 2       | Suchen und wählen Sie den <b>Dateipfad</b> der CSV-Datei (Comma Separated Values) mit den zu importierenden Symbolen aus.  |
| 3       | Klicken Sie nach Bedarf auf <b>Importoptionen</b> und konfigurieren Sie Formatierungsoptionen für die importierten Symbole:<br>   |

| Schritt       | Aktion  |       |         |       |        |               |   |   |   |            |                                 |   |   |               |                    |   |   |
|---------------|---|-------|---------|-------|--------|---------------|---|---|---|------------|---------------------------------|---|---|---------------|--------------------|---|---|
| 4             | <p>Klicken Sie auf <b>Importieren</b>.<br/> <b>Ergebnis:</b> Alle in der ausgewählten CSV-Datei enthaltenen Symbole werden erstellt und im Fenster <b>Symbolliste</b> mit den angegebenen Formatierungsoptionen angezeigt.<br/>                 Falls beim Import Fehler festgestellt werden, wird ein Bericht mit den erkannten Fehlern angezeigt:</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p style="color: green; text-align: center;">Bericht über 'symbols_timer_drum.csv' <span style="float: right;">×</span></p> <div style="background-color: #008000; height: 20px; margin-bottom: 5px;"></div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Typ</th> <th>Meldung</th> <th>Zeile</th> <th>Spalte</th> </tr> </thead> <tbody> <tr> <td>Informationen</td> <td>Backup 'H:\Nigel\SoMachine Basic\Test projects\SymBackup_20130724.csv' suc...</td> <td>0</td> <td>0</td> </tr> <tr> <td>Erstellung</td> <td>Symbol 'M2' zugeordnet zu '%M2'</td> <td>2</td> <td>0</td> </tr> <tr> <td>Informationen</td> <td>Import erfolgreich</td> <td>0</td> <td>0</td> </tr> </tbody> </table> <div style="text-align: right; margin-top: 10px;"> <input type="button" value="Speichern"/> <input type="button" value="Schließen"/> </div> </div> | Typ   | Meldung | Zeile | Spalte | Informationen | Backup 'H:\Nigel\SoMachine Basic\Test projects\SymBackup_20130724.csv' suc... | 0 | 0 | Erstellung | Symbol 'M2' zugeordnet zu '%M2' | 2 | 0 | Informationen | Import erfolgreich | 0 | 0 |
| Typ           | Meldung   | Zeile | Spalte  |       |        |               |   |   |   |            |                                 |   |   |               |                    |   |   |
| Informationen | Backup 'H:\Nigel\SoMachine Basic\Test projects\SymBackup_20130724.csv' suc...   | 0     | 0       |       |        |               |   |   |   |            |                                 |   |   |               |                    |   |   |
| Erstellung    | Symbol 'M2' zugeordnet zu '%M2'   | 2     | 0       |       |        |               |   |   |   |            |                                 |   |   |               |                    |   |   |
| Informationen | Import erfolgreich  | 0     | 0       |       |        |               |   |   |   |            |                                 |   |   |               |                    |   |   |
| 5             | Klicken Sie auf <b>Speichern</b> , um den Inhalt des Berichts in eine Textdatei (.txt) zu schreiben.  |       |         |       |        |               |   |   |   |            |                                 |   |   |               |                    |   |   |

### Exportieren der Symbolliste

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie entweder auf die Schaltfläche <b>Exportieren</b> , oder klicken Sie mit der rechten Maustaste auf eine beliebige Stelle in der Symbolliste, und wählen Sie die Option <b>Symbole exportieren</b> aus. Sie werden dann zur Speicherung Ihrer Änderungen aufgefordert. Daraufhin wird das Fenster <b>Symbole exportieren</b> angezeigt.   |
| 2       | Suchen und wählen Sie einen <b>Dateipfad</b> und <b>Dateinamen</b> für die zu erstellende CSV-Datei (Comma Separated Values) aus.   |
| 3       | <p>Klicken Sie nach Bedarf auf <b>Exportoptionen</b> und konfigurieren Sie Formatierungsoptionen für die exportierten Werte:</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;"> <p>^ Exportoptionen</p> <p>Einbeziehen <input checked="" type="checkbox"/> Kopfzeilen <input checked="" type="checkbox"/> Kommentare</p> <p>Trennzeichen <input type="text" value="Semikolon"/></p> <p>Codepage <input type="text" value="Unicode"/></p> </div> |
| 4       | <p>Klicken Sie auf <b>Exportieren</b>.<br/> <b>Ergebnis:</b> Dadurch wird eine CSV-Datei mit dem angegebenen Formatierungsoptionen erstellt.</p>  |

### Freigeben von Symbolen zwischen einem SoMachine Basic-Projekt und einem Vijeo-Designer-Projekt

Bevor Sie die Symbole für ein Vijeo-Designer-Projekt freigeben, sollten Sie sicherstellen, dass diese alle im SoMachine Basic-Projekt definiert sind. Ist dies nicht der Fall, erstellen bzw. öffnen Sie ein Projekt in SoMachine Basic, definieren Sie die Symbolnamen, und speichern Sie das Projekt. Sie können Vijeo-Designer-Standardsymbole für alle Speicherobjekte im Projekt erstellen, siehe Erstellen von Standardsymbolen (*siehe Seite 178*).

Halten Sie sich an die folgenden Schritte, um SoMachine Basic-Symbole für ein Vijeo-Designer-Projekt freizugeben:

| Schritt | Aktion  |
|---------|---|
| 1       | Starten Sie Vijeo-Designer.   |
| 2       | Erstellen bzw. öffnen Sie ein Projekt in Vijeo-Designer.  |
| 3       | Klicken Sie auf die Registerkarte <b>Projekt</b> im Fenster <b>Navigator</b> , klicken Sie mit der rechten Maustaste auf <b>E/A-Manager</b> , und wählen Sie <b>Neuer Treiber ... Einfügen</b> aus.<br><b>Ergebnis:</b> Das Fenster <b>Neuer Treiber</b> wird geöffnet.   |
| 4       | Wählen Sie aus der Liste <b>Treiber</b> einen Treiber aus. Wählen Sie aus der Liste <b>Gerät</b> ein Gerät aus, und klicken Sie auf <b>OK</b> .<br>Beispiel:<br><ul style="list-style-type: none"> <li>● Treiber: <b>Modbus TCP/IP</b></li> <li>● Gerät: <b>Modbus-Gerät</b></li> </ul> <b>Ergebnis:</b> Das Fenster <b>Gerätekonfiguration</b> wird geöffnet.  |
| 5       | Geben Sie die Details für die jeweiligen Parameter ein, und klicken Sie auf <b>OK</b> . Beispielsweise <b>IP-Adresse</b> , <b>Einheit-ID</b> , <b>IP-Protokoll</b> usw.<br><b>Ergebnis:</b> Es wird ein neuer Treiber erstellt, um die Kommunikation mit der Steuerung zu starten. Der ausgewählte Treiber und das ausgewählte Gerät werden unter dem Knoten <b>E/A-Manager</b> auf der Registerkarte <b>Projekt</b> im Fenster <b>Navigator</b> angezeigt. |
| 6       | Klicken Sie in der Menüleiste Vijeo-Designer auf <b>Variable → Variablen verbinden</b> .<br><b>Ergebnis:</b> Das Fenster <b>Variablen verknüpfen</b> wird geöffnet.   |
| 7       | Setzen Sie den Filter <b>Dateityp</b> auf <b>SoMachine Basic-Projektdateien (*.SMBP)</b> und setzen Sie den Filter <b>Gerät</b> auf den Treiber, den Sie für die Kommunikation erstellt haben.  |
| 8       | Wählen Sie das SoMachine Basic-Projekt aus, in dem Sie die Symbole definiert haben, und klicken Sie auf <b>Öffnen</b> .<br><b>Ergebnis:</b> Alle Symbole werden automatisch aus dem Projekt extrahiert und mit dem erstellten Treiber verknüpft.  |
| 9       | Wählen Sie die zu verwendenden Variablen aus, und fügen Sie sie zur HMI-Anwendung hinzu.<br><b>Ergebnis:</b> Alle Variablen mit denselben Namen als Symbole werden der Liste der verfügbaren Variablen hinzugefügt. Die Variablenliste werden unter dem Knoten <b>Variablen</b> auf der Registerkarte <b>Projekt</b> im Fenster <b>Navigator</b> angezeigt.   |

**HINWEIS:** Wenn Sie die Symbole bereits für ein anderes Vijeo-Designer-Projekt freigegeben haben, aber die vorhandenen Symbole geändert und/oder zu Ihrem SoMachine Basic-Projekt neue Symbole hinzugefügt haben, müssen Sie die Symbole im Vijeo-Designer-Projekt aktualisieren.

Um die Symbole in einem Vijeo-Designer-Projekt zu aktualisieren, definieren Sie zuerst neue Symbole bzw. ändern Sie die vorhandenen Symbole, und speichern Sie das SoMachine Basic-Projekt. Öffnen Sie dann das Vijeo-Designer-Projekt, und halten Sie sich an die folgenden Schritte:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf der Registerkarte <b>Projekt</b> im Fenster <b>Navigator</b> mit der rechten Maustaste auf <b>Variablen</b> , und wählen Sie <b>Link aktualisieren</b> aus.<br><b>Ergebnis:</b> Der Gerätetreiber und die vorhandenen Symbole werden aktualisiert.   |
| 2       | Klicken Sie erneut mit der rechten Maustaste auf <b>Variablen</b> , wählen Sie <b>Neue Variablen von Geräten</b> und anschließend die neuen Variablen aus, die Sie im SoMachine Basic-Projekt erstellt haben.<br><b>Ergebnis:</b> Die neuen Variablen aus dem SoMachine Basic-Projekt werden der Liste der Variablen hinzugefügt. Diese Variablen werden unter dem Knoten <b>Variablen</b> auf der Registerkarte <b>Projekt</b> im Fenster <b>Navigator</b> angezeigt. |

## Ansicht der Speichernutzung

### Überblick

Sie können Informationen zu dem von der Anwendung, vom Programm und von den zugehörigen Benutzerdaten beanspruchten SPS-Speicher anzeigen.

### Anzeigen der Detailansicht der Speichernutzung

Das Programm muss zuerst ohne Fehler kompiliert werden, damit diese Funktion verwendet werden kann. Den aktuellen Programmstatus finden Sie im Meldungsfenster (*siehe Seite 154*).

Gehen Sie vor wie folgt, um die **Detailansicht der Speichernutzung** aufzurufen:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Registerkarte <b>Tools</b> im linken Bereich des Fensters <b>Programmierung</b> . |
| 2       | Klicken Sie auf <b>Speichernutzung</b> .<br>Das Fenster „Speichernutzung“ wird geöffnet.              |

### Beschreibung Detailansicht der Speichernutzung

**HINWEIS:** Diese Ansicht ist nur verfügbar, wenn eine gültige Kompilierung vorhanden ist.

In den folgenden Tabellen werden die Felder in der **Detailansicht der Speichernutzung** beschrieben:

| Feld                       | Beschreibung   |
|----------------------------|--|
| <b>Letzte Kompilierung</b> | <p>Datum und Uhrzeit, zu der das Programm zuletzt kompiliert wurde.</p> <p><b>HINWEIS:</b><br/>Dieser Wert wird aktualisiert, sobald:</p> <ul style="list-style-type: none"> <li>● auf die Schaltfläche <b>Kompilieren</b> ✓ in der Symbolleiste geklickt wird.</li> <li>● eine Anmeldung bei einer Steuerung erfolgt.</li> <li>● ein Programm-Upload gestartet wird.</li> <li>● eine Programmänderung im Online-Modus an die Steuerung gesendet wird.</li> <li>● der Simulator gestartet wird.</li> </ul> |

| Programmzeilen   |  |
|--|--|
| Feld   | Beschreibung   |
| <b>Verwendet</b>   | Anzahl der vom Programm verwendeten Codezeilen.  |
| <b>Verbleibend</b>   | Maximale Anzahl der für das Programm verfügbaren Zeilen, abzüglich der bereits verwendeten Zeilen. |
| <b>HINWEIS:</b> Es besteht keine direkte Verbindung zwischen der Anzahl verwendeter Programmzeilen und der Gesamtanzahl an IL-Code-Zeilen in Programmbausteinen in der Registerkarte <b>Programmieren</b> . Beispielsweise können 2 Zeilen an IL-Code 6 Programmzeilen erzeugen. |  |

| Cachespeicher                        |   |
|--------------------------------------|---|
| Feld                                 | Beschreibung  |
| <b>Periodische und Ereignistasks</b> | Umfang des von den periodischen und den Ereignistasks beanspruchten Cachespeichers (in Byte). |
| <b>Reserviert für System</b>         | Umfang des für eine Nutzung durch das System reservierten Cachespeichers (in Byte).           |
| <b>Verbleibender Speicher</b>        | Umfang des noch für das Programm verfügbaren Cache-Speichers (in Byte).                       |

| RAM-Speicher                         |   |
|--------------------------------------|---|
| Feld                                 | Beschreibung  |
| <b>Mastertask und Unterprogramme</b> | Umfang des von der Mastertask des Programms und allen Unterprogrammen beanspruchten RAM-Speichers (in Byte).  |
| <b>Konfiguration</b>                 | Umfang des zur Speicherung der Hardwarekonfiguration der Steuerung und der Erweiterungsmodule verwendeten RAM-Speichers (in Byte).  |
| <b>Speicherobjekte</b>               | Umfang des RAM-Speichers, der von den von der Anwendung verwendeten Speicherobjekten beansprucht wird (in Byte).  |
| <b>Anzeige</b>                       | Größe der Dezentrale Grafikanzeige: Remote Graphic Display-Anwendung (in Byte). Null, wenn die Steuerung keine Unterstützung für Dezentrale Grafikanzeige: Remote Graphic Display bietet. |
| <b>Verbleibender Speicher</b>        | Umfang des noch für das Programm verfügbaren RAM-Speichers (in Byte).   |

| Keine Programmdaten |   |
|---------------------|---|
| Feld                | Beschreibung  |
| <b>Verwendet</b>    | Die Größe des Speichers, der durch Projekteigenschaften, Symbole, Kommentare und Animationstabellen belegt ist. |
| <b>Verbleibend</b>  | Die Größe des Speichers, der für keine Programmdaten zur Verfügung steht.                                       |

# Abschnitt 6.12

## Programmieren in Kontaktplan

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                                  | Seite |
|--|-------|
| Einführung in Kontaktplan              | 186   |
| Programmiergrundlagen für Kontaktplan  | 188   |
| Farbkodierungen von Programmbausteinen | 190   |
| Grafikelemente in Kontaktplan          | 192   |
| Vergleichsbausteine                    | 199   |
| Operationsbausteine                    | 200   |
| Hinzufügen von Kommentaren             | 204   |
| Best Practices für die Programmierung  | 205   |

## Einführung in Kontaktplan

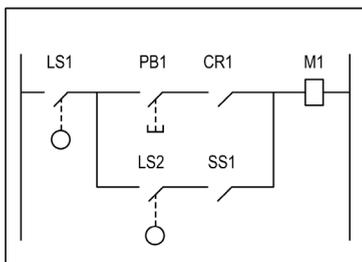
### Einführung

Kontaktplan-Diagramme ähneln Schaltplänen, in denen Relaisschaltungen dargestellt werden. Der Hauptunterschied besteht in den folgenden Funktionen der Programmierung in Kontaktplan, die Relaisschaltungen nicht bieten:

- Alle Eingänge und binären Logikbits werden durch Kontaktsymbole dargestellt (  ).
- Alle Ausgänge und binären Logikbits werden durch Spulensymbole dargestellt (  ).
- Der grafische KOP-Anweisungssatz enthält numerische Operationen.

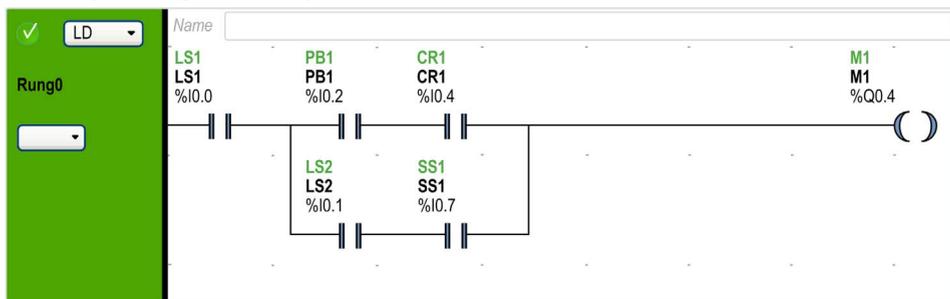
### KOP-Entsprechungen zu Relaisschaltungen

Die nachstehende Abbildung zeigt den vereinfachten Schaltplan einer Relaisschaltung:



Relay logic circuit

Die Entsprechung in Kontaktplan:



Beachten Sie, dass in der Abbildung oben alle Eingänge, denen in der Relaisschaltung ein Schalter zugewiesen ist, in Kontaktplan als Kontakte dargestellt werden. Die Ausgangsspule M1 im Relais-Logikdiagramm wird anhand eines Ausgangsspulensymbols in Kontaktplan dargestellt. Die Adressen über den Kontakt-/Spulensymbolen in Kontaktplan verweisen auf die Positionen der externen Eingangs- und Ausgangsverbindungen des Logic Controllers.

## Programmbausteine in Kontaktplan

Ein in Kontaktplan geschriebenes Programm besteht aus Programmbausteinen, bei denen es sich jeweils um Kombinationen verschiedener Grafikanweisungen handelt, die zwischen zwei vertikalen Potenzialschienen angeordnet sind. Die Programmbausteine werden vom Logic Controller der Reihe nach ausgeführt.

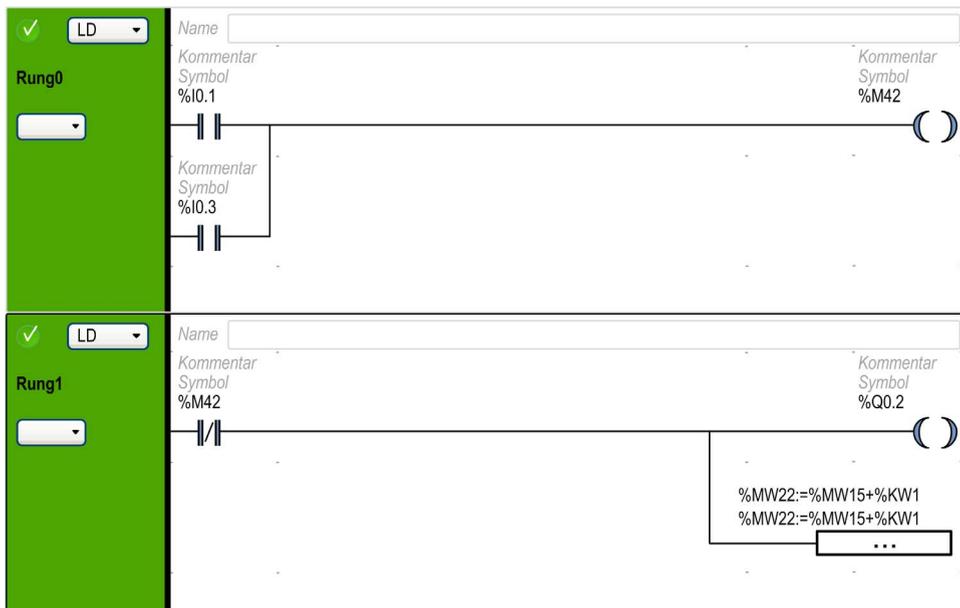
Die Grafikanweisungen stellen folgende Funktionen dar:

- Eingänge und Ausgänge der Steuerung (Drucktaster, Sensoren, Relais, Anzeigeleuchten usw.)
- Funktionen der Steuerung (Timer, Zähler usw.)
- Mathematische und logische Operationen (Addition, Division, AND, XOR usw.)
- Vergleichsoperatoren und andere numerische Operationen ( $A < B$ ,  $A = B$ , Verschieben, Drehen usw.)
- Interne Variablen der Steuerung (Bits, Wörter usw.)

Diese Grafikanweisungen werden mit vertikalen und horizontalen Verbindungen angeordnet, die schließlich zu einem Ausgang oder zu mehreren Ausgängen und/oder Aktionen führen. Ein Programmbaustein kann nicht mehrere Gruppen verbundener Anweisungen unterstützen.

## Beispiel für Programmbausteine in Kontaktplan

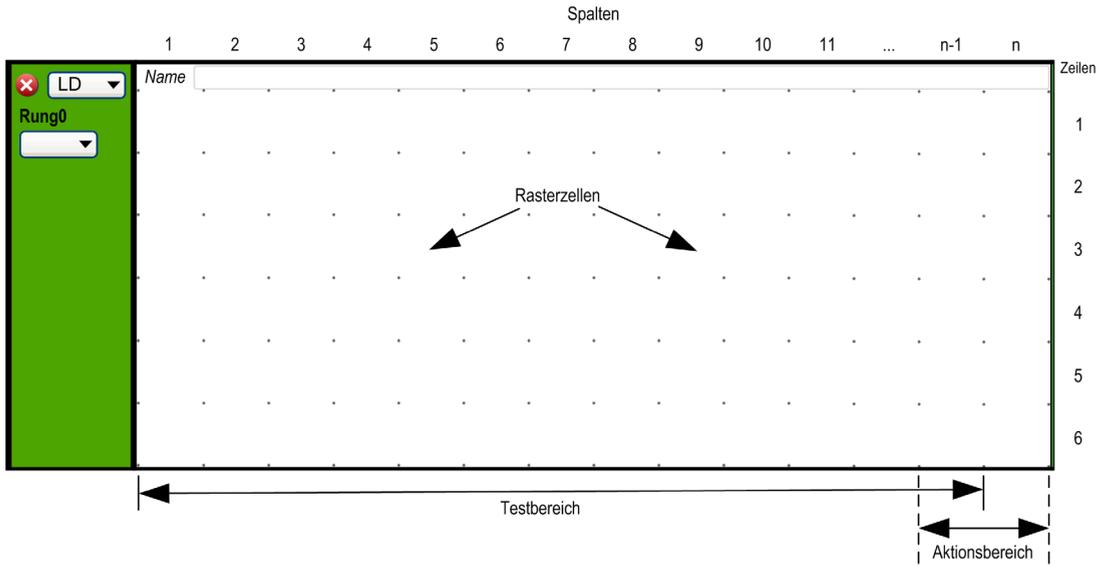
Das folgende Diagramm ist ein Beispiel für ein KOP-Programm, das aus zwei Programmbausteinen besteht.



## Programmiergrundlagen für Kontaktplan

### Programmerraster

Jeder Programmbaustein in Kontaktplan besteht aus einem Raster mit bis zu 255 Zeilen und 11 bis 30 Spalten, das in zwei Bereiche aufgeteilt ist, wie in folgender Abbildung illustriert:



n Anzahl konfigurierter Spalten (11 bis 30) Weitere Informationen zur Anzahl der Spalten finden Sie unter Anpassen des Kontaktplan-Editors (*siehe Seite 60*).

### Rasterzellen

Anhand der Zellen können Sie Grafikelemente im Raster positionieren. Jede Zelle im Raster ist durch 4 Punkte in den Zellecken begrenzt.

## Rasterbereiche

Standardmäßig ist das Programmiergitter in Kontaktplan in zwei Bereiche unterteilt:

- **Testbereich**  
Enthält die zu testenden Bedingungen für die Ausführung von Aktionen. Besteht aus den Spalten 1 bis n-1, wobei n der Anzahl konfigurierter Spalten entspricht und Kontakte, Funktionsbausteine und Vergleichsbausteine umfasst.
- **Aktionsbereich**  
Enthält den Ausgang bzw. die Operation, die entsprechend den Testergebnissen der Bedingungen im Testbereich ausgeführt wird. Besteht aus den Spalten n-1 to n, wobei n der Anzahl konfigurierter Spalten entspricht und Spulen und Operationsbausteine umfasst.

## Anpassen des Kontaktplan-Editors



Verwenden Sie die folgenden Objekte am oberen Ende des Ladder-Editors, um den Inhalt des Editors anzupassen:

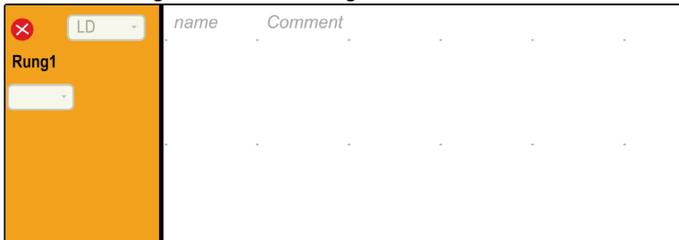
| Objekt                                | Beschreibung  |
|---------------------------------------|---|
| IL > LD                               | Wechselt die Anzeige aller Programmbausteine von IL zu Kontaktplan (Ladder).  |
| LD > IL                               | Wechselt die Anzeige aller Programmbausteine von Kontaktplan (Ladder) zu IL.  |
| -                                     | Löscht eine Spalte aus dem Kontaktplan-Gitter. Die Schaltfläche wird deaktiviert, wenn die minimale Anzahl an Spalten (11) erreicht wurde.  |
| +                                     | Fügt dem Kontaktplan-Gitter eine Spalte hinzu. Die Schaltfläche wird deaktiviert, wenn die maximale Anzahl an Spalten (30) erreicht wurde.  |
| <b>Kommentare anzeigen/ausblenden</b> | Klicken zum Anzeigen/Ausblenden der Kommentare in den Programmbausteinen. Wenn <b>T</b> freigegeben wird, werden die Kommentare auf zwei Zeilen angezeigt.  |
| <b>T</b>                              | Klicken zum Anzeigen oder Ausblenden der Symbole in den Programmbausteinen. Wenn <b>Kommentare anzeigen/ausblenden</b> freigegeben wird, werden die Symbole auf zwei Zeilen angezeigt.  |
| <b>DEZ/HEX</b>                        | Nur im Online-Modus angezeigt. Klicken, um die numerischen Werte in den Programmbausteinen abwechselnd im Dezimal- und Hexadezimalformat anzuzeigen.  |
| <b>1 – Neue POU</b>                   | Doppelklicken zum Bearbeiten des Standardnamens der <b>POU</b> , der im Bereich <b>Tools → Mastertask</b> des Bildschirms erscheint.  |
| <b>Kommentar</b>                      | Doppelklicken zum Schreiben von Text als Kommentar in dieser <b>POU</b> .   |
| <b>Zoom-Schieberegler</b>             | Vergrößern oder Verkleinern des Kontaktplan-Editors.<br>Vergrößern oder verkleinern mithilfe des Tastaturkürzels <b>Strg + Mausrad</b> .<br>Die Zoomposition wird auch dann beibehalten, wenn Sie durch das Projekt navigieren. |



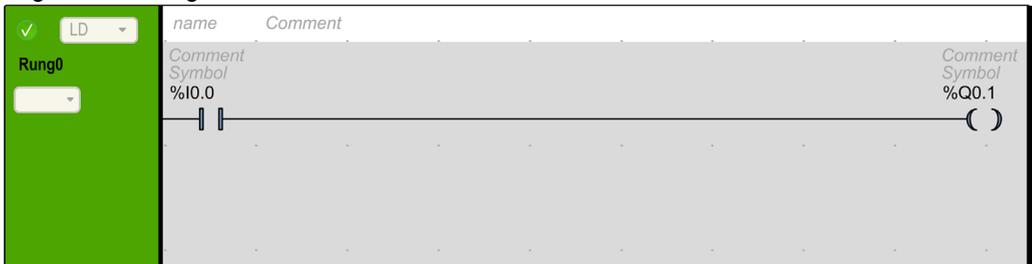
## Online-Modus

Wenn im Online-Modus:

- Unveränderte Programmbausteine erscheinen mit einem grünen Hintergrund.
- Programmbausteine, die im Online-Modus hinzugefügt oder geändert wurden, erscheinen mit einem orangefarbenen Hintergrund:



- Programmbausteine ohne veränderbare Elemente sind gesperrt und erscheinen mit einem grauen Vordergrund:



## Grafikelemente in Kontaktplan

### Einführung

In Kontaktplan werden Anweisungen durch Ziehen und Ablegen von Grafikelementen aus der Symbolleiste eingefügt. Die Symbolleiste ist über dem Programmierungsbereich in einer Rasterzelle verfügbar.

### Einfügen eines Grafikelements

Gehen Sie zum Einfügen eines Grafikelements vor wie folgt:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf das Grafikelement in der Symbolleiste, um es einzufügen. Wenn es sich bei dem Grafikelement um ein Menü handelt, werden die Grafikoptionen des Menüs angezeigt. Klicken Sie auf die Menüoption, die Sie einfügen möchten.   |
| 2       | Bewegen Sie den Mauszeiger bis zu der Position im Programmbaustein, an der das Grafikelement eingefügt werden soll, und klicken Sie.<br>Hinweis: Einige Elemente müssen in den Test- oder Aktionsbereich des Programmbausteins eingefügt werden. Detaillierte Informationen hierzu finden Sie in der Beschreibung der einzelnen Grafikelemente. |
| 3       | Falls erforderlich, klicken Sie auf das Grafikelement <b>[Auswahlmodus]</b>  in der Symbolleiste, um die Auswahl zurückzusetzen.  |

### Programmbausteine

Zur Verwaltung der Programmbausteine in einem Programm stehen Ihnen folgende Grafikelemente zur Verfügung:

| Grafikelement   | Name   | Funktion  |
|---|--|---|
|  | Programmbaustein erstellen<br><i>(siehe Seite 103)</i> | Fügt einen neuen, leeren Programmbaustein unter dem letzten Programmbaustein im Programmierungsbereich ein.   |
|  | Programmbaustein einfügen<br><i>(siehe Seite 104)</i>  | Fügt einen neuen, leeren Programmbaustein direkt über dem derzeit ausgewählten Programmbaustein ein.  |
|  | Programmbaustein löschen<br><i>(siehe Seite 106)</i>   | Entfernt den derzeit ausgewählten Programmbaustein aus dem Programm. Wenn der Programmbaustein nicht leer ist, werden Sie aufgefordert, das Löschen des Programmbaustein-Inhalts zu bestätigen. |

## Verzweigungsmodi

Zur Verwaltung der Verzweigung in Kontaktplan stehen Ihnen folgende Grafikelemente zur Verfügung:

| Grafikelement | Name              | Funktion  |
|---------------|-------------------|---|
|               | Normalbetrieb     | Ermöglicht Ihnen die Platzierung von Programmier-elementen (z. B. Kontakte, Spulen usw., mit Ausnahme von Funktionsbauseinen) In-line mit der Leiterlinie.        |
|               | Verzweigungsmodus | Ermöglicht Ihnen die Platzierung von Programmier-elementen (z. B. Kontakte, Spulen usw., mit Ausnahme von Funktionsbauseinen) in Verzweigung mit der Leiterlinie. |

## Auswahlen und Linien

Zur Auswahl von Grafikelementen und zum Zeichnen von Linien stehen Ihnen folgende Grafikelemente zur Verfügung:

| Grafikelement | Name           | Funktion  |
|---------------|----------------|---|
|               | Auswahlmodus   | Auswahlmodus.   |
|               | Linie zeichnen | Zeichnet eine Leiterlinie zwischen 2 Grafikelementen. |
|               | Linie löschen  | Löscht eine Leiterlinie.                              |

## Kontakte

Zum Einfügen von Kontakten (Höhe: 1 Zeile, Breite: 1 Spalte) stehen Ihnen folgende Grafikelemente zur Verfügung:

| Grafikelement | Name  | Anweisungsliste | Funktion  |
|---------------|---|-----------------|---|
|               | NO-Kontakt (Schließer)                        | LD              | Der Kontakt ist stromleitend, wenn sich das Befehlsbitobjekt im Zustand 1 befindet. |
|               | NC-Kontakt (Öffner)                           | LDN             | Der Kontakt ist stromleitend, wenn sich das Befehlsbitobjekt im Zustand 0 befindet. |
|               | Kontakt zur Erkennung einer steigenden Flanke | LDR             | Steigende Flanke: Erkennt den Übergang des Befehlsbitobjekts von 0 auf 1.           |
|               | Kontakt zur Erkennung einer fallenden Flanke  | LDF             | Fallende Flanke: Erkennt den Übergang des Befehlsbitobjekts von 1 auf 0.            |

### Vergleichsbaustein

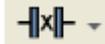
Vergleichsbausteine werden im Testbereich des Programmiergitters abgelegt werden. Der Baustein kann in einer beliebigen Zeile und in einer beliebigen Spalte im Testbereich platziert werden, solange sich die Gesamtlänge der Anweisung innerhalb des Testbereichs befindet.

Das Grafikelement für Vergleichsbausteine beansprucht 2 Zellen (Höhe: 1 Zeile, Breite: 2 Spalten).

| Grafikelement   | Name               | Anweisungsliste                  | Funktion  |
|---|--------------------|----------------------------------|---|
|  | Vergleichsbaustein | Jeder gültige Vergleichsausdruck | Verwenden Sie das Grafikelement der Vergleichsbausteine, um Vergleichsausdrücke ( <i>siehe Seite 199</i> ) in Anweisungsliste in Kontaktplan-Programmbausteine einzufügen. Ein Vergleichsausdruck vergleicht 2 Operanden. Der Ausgang wechselt in den Zustand 1, wenn das Ergebnis überprüft wurde. |

### Boolesche Operationen

Das Grafikelement für boolesche Operationen beansprucht 1 Zelle (Höhe: 1 Zeile, Breite: 1 Spalte).

| Grafikelement   | Name            | Operator                       | Funktion   |
|---|-----------------|--------------------------------|--|
|  | XOR-Anweisungen | XOR,<br>XORN,<br>XORR,<br>XORF | Die Anweisung <code>XOR</code> führt eine exklusive ODER-Operation zwischen dem Operanden und dem booleschen Ergebnis der vorhergehenden Anweisung aus.<br>Die Anweisung <code>XORN</code> führt eine Exklusiv-ODER-Operation zwischen dem inversen Wert des Operanden und dem booleschen Ergebnis der vorhergehenden Anweisung aus.<br>Die Anweisung <code>XORR</code> führt eine Exklusiv-ODER-Operation zwischen der steigenden Flanke des Operanden und dem booleschen Ergebnis der vorhergehenden Anweisung aus.<br>Die Anweisung <code>XORF</code> führt eine exklusive ODER-Operation zwischen der fallenden Flanke des Operanden und dem booleschen Ergebnis der vorhergehenden Anweisung aus. |

## Funktionen

Funktionsbausteine werden immer in der ersten Zeile des Kontaktplan-Programmerrasters angezeigt. Über oder unter einem Funktionsbaustein dürfen weder Kontaktplan-Anweisungen noch Anschlusslinien vorhanden sein. Am linken Rand eines Funktionsbausteins kommen Kontaktplan-Testanweisungen an, vom rechten Rand der Funktion gehen Test- und Aktionsanweisungen ab.

Die Grafikelemente der Funktionsbausteine können nur im Testbereich abgelegt werden und beanspruchen 2, 3 oder 4 Zeilen über 2 Zellen spalten.

| Grafikelement   | Name                         | Funktion   |
|---|------------------------------|--|
|  | Timer, Zähler, Register usw. | Jeder Funktionsbaustein verwendet die Ein- und Ausgänge, die eine Verbindung zu den anderen Grafikelementen ermöglichen.<br><b>HINWEIS:</b> Die Ausgänge von Funktionsbausteinen können nicht miteinander verbunden werden (vertikale Kurzschlüsse). |

## Spulen

Die Grafikelemente der Spulen können nur im Aktionsbereich abgelegt werden und beanspruchen 1 Zeile (Höhe: 1 Zeile, Breite: 1 Spalte).

| Grafikelement   | Name                | Operator | Funktion  |
|---|---------------------|----------|---|
|    | Direkte Spule       | ST       | Das zugewiesene Bitobjekt nimmt den Wert des Testbereichsergebnisses an.                |
|    | Inverse Spule       | STN      | Das zugewiesene Bitobjekt nimmt den invertierten Wert des Testbereichsergebnisses an.   |
|   | Erregte Spule       | S        | Das zugewiesene Bitobjekt wird auf 1 gesetzt, wenn das Ergebnis des Testbereichs 1 ist. |
|  | Nicht erregte Spule | R        | Das zugewiesene Bitobjekt wird auf 0 gesetzt, wenn das Ergebnis des Testbereichs 1 ist. |

### Grafcet (Liste) Anweisungen

Zur Verwaltung der Verzweigung in Kontaktplan stehen Ihnen folgende Grafikelemente zur Verfügung:

| Grafikelement   | Name  | Operator | Funktion  |
|---|---|----------|---|
|  | Grafcet-Schritt-Aktivierung/<br>Grafcet-Schritt-Deaktivierung | #        | Deaktiviert den aktuellen Schritt und aktiviert optional einen anderen Schritt im Grafcet-Programm. |
|  | Grafcet-Schritt-Deaktivierung                                 | #D       | Deaktiviert zusätzlich zum Deaktivieren des aktuellen Schritts einen Schritt im Grafcet-Programm.   |

### Operationsbausteine

Das Grafikelement der Operationsbausteine wird im Aktionsbereich platziert und beansprucht 2 Spalten über 1 Zeile:

| Grafikelement   | Name               | Operator  | Funktion   |
|---|--------------------|---|--|
|  | Operationsbaustein | Jede(r) gültige Operator oder Zuweisungsanweisung | Verwenden Sie das Grafikelement für Operationsbausteine, um Operationen und IL-Zuweisungsanweisungen ( <i>siehe Seite 200</i> ) in Kontaktplan-Programmbausteine einzufügen. |

### Andere Kontaktplan-Elemente



Im Menü **Andere Kontaktplan-Elemente** werden verschiedene Anweisungen gruppiert.

Die Anweisungen `OPEN` und `SHORT` bieten eine praktische Methode für das Debugging und die Fehlerbehebung in KOP-Programmen. Dies Sonderanweisungen ändern die Logik eines Strompfads (Programmbausteins), indem der Stromfluss in einem Strompfad entweder kurzgeschlossen oder geöffnet wird (siehe folgende Tabelle).

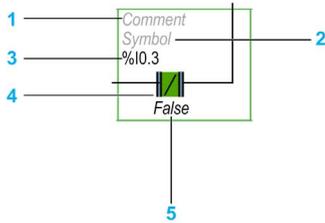
Die Grafikelemente `END/JUMP` werden im Aktionsbereich abgelegt und beanspruchen 1 Zeile (Höhe: 1 Zeile, Breite: 1 Spalte).

| Grafikelement  | Name             | Operator                             | Funktion                                     |
|--|------------------|--------------------------------------|--|
|   | Steigende Flanke | <code>RISING n<sup>(1)</sup></code>  | Evaluert die steigende Flanke des Ausdrucks. |
|   | Fallende Flanke  | <code>FALLING n<sup>(1)</sup></code> | Evaluert die fallende Flanke des Ausdrucks.  |
| <sup>(1)</sup> <i>n</i> ist eine Ganzzahl, die bei jedem Einfügen einer steigenden oder fallenden Flanke inkrementiert wird. |                  |                                      |  |

| Grafikelement   | Name                                      | Operator            | Funktion   |
|---|---|---------------------|--|
|                                      | Logisches NOT                             | N                   | Gibt den umgekehrten Wert seines Operanden weiter.   |
|                                      | OPEN                                      | LD 0<br>AND 0       | Am Anfang des Strompfads.<br>Innerhalb eines Strompfads: Unterbricht den Stromfluss in einem KOP-Strompfad unabhängig von den Ergebnissen der letzten logischen Operation.   |
|                                      | SHORT                                     | LD 1<br>OR 1        | Am Anfang des Strompfads.<br>Innerhalb eines Strompfads: Ermöglicht den Stromfluss im Strompfad unabhängig von den Ergebnissen der letzten logischen Operation.  |
|                                      | Programmstopp                             | END                 | Definiert das Ende des Programms.  |
|                                      | Bedingter Programmstopp                   | ENDCN               | Definiert ein bedingtes Ende des Programms.  |
|                                      | Sprung oder Unterprogrammaufruf           | JMP                 | Stellt eine Verbindung zu einem vor- oder nachgeschalteten markierten Programmbaustein her.<br><br><b>HINWEIS:</b> Bei der Programmierung in IL erfolgt die Verbindung zu einer vor- oder nachgeschalteten markierten Anweisung. |
|                                      | Bedingter Sprung oder Unterprogrammaufruf | JMPCN               | Bedingte Verbindung zu einer vor- oder nachgeschalteten markierten Anweisung.<br><br><b>HINWEIS:</b> Bei der Programmierung in IL erfolgt die Verbindung zu einer vor- oder nachgeschalteten markierten Anweisung.               |
|                                    | Bedingte Elemente                         | IF<br>ELSE<br>ENDIF | Führt eine Gruppe von Anweisungen unter bestimmten Bedingungen aus, abhängig vom Wert eines Ausdrucks.   |
|                                    | Schleifenelemente                         | FOR<br>ENDFOR       | Wiederholt eine Gruppe von Anweisungen.  |
| <sup>(1)</sup> n ist eine Ganzzahl, die bei jedem Einfügen einer steigenden oder fallenden Flanke inkrementiert wird. |   |                     |  |

## Kontakte und Spulen

Nach dem Einfügen in eine Zelle können zusätzliche Informationen zu dem Kontakten und Spulen zugeordneten Objekt angezeigt werden:



| Legende | Element           | Beschreibung  |
|---------|-------------------|---|
| 1       | Benutzerkommentar | Klicken Sie, um einen Kommentar ( <i>siehe Seite 204</i> ) hinzuzufügen.  |
| 2       | Symbol            | Klicken Sie, um den Namen eines Symbols ( <i>siehe Seite 80</i> ) einzugeben, das dem in der Zelle enthaltenen Objekt zugeordnet werden soll. |
| 3       | Adresse           | Klicken Sie, um die Adresse des in der Zelle enthaltenen Objekts einzugeben.  |
| 4       | Grafikelement     | Das grafische Element.  |
| 5       | Echtzeitwert      | Im Online-Modus (Verbindung zu einem Logic Controller vorhanden und Programm aktiv) wird der Echtzeitwert des Objekts in der Zelle angezeigt. |

## Vergleichsbausteine

### Einfügen von AWL-Vergleichsausdrücken in Kontaktplan

Verwenden Sie das Grafikelement **Vergleichsbaustein**, um Vergleichsausdrücke in Anweisungsliste in Kontaktplan-Programmbausteine einzufügen.



Die Operanden müssen vom selben Objekttyp sein: Wörter mit Wörtern, Gleitkommata mit Gleitkommata etc.

Gehen Sie wie folgt vor:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf die Schaltfläche <b>Vergleichsbaustein</b>  in der Symbolleiste.                           |
| 2       | Klicken Sie auf eine beliebige Stelle innerhalb des Programmbausteins, um den <b>Vergleichsbaustein</b> einzufügen.   |
| 3       | Doppelklicken Sie auf die Zeile <b>Vergleichsausdruck</b> .   |
| 4       | Geben Sie eine gültige AWL-Vergleichsoperation ein und drücken Sie die Eingabetaste. Der Ausdruck kann im Online-Modus geändert werden. Siehe Online-Änderungen ( <i>siehe Seite 255</i> ). |

#### HINWEIS:

Wenn die Anwendung mit einer Funktionsebene (*siehe Seite 94*) von mindestens **Ebene 6.0** konfiguriert wird:

- Es können bis zu fünf Operanden und drei Ebenen von Klammern in einem Vergleichsbaustein verwendet werden.
- Es müssen mindestens 20 Speicherwörter (%MW) verfügbar sein, um mehrere Operanden in der Master-Task verwenden zu können. Wenn in einer periodischen Task ebenfalls mehrere Operanden verwendet werden, müssen zusätzliche 20 Speicherwörter verfügbar sein.

**HINWEIS:** Mehrere Operanden-Ausdrücke können in Ereignis-Tasks nicht verwendet werden.

### Hilfe mit der Syntax

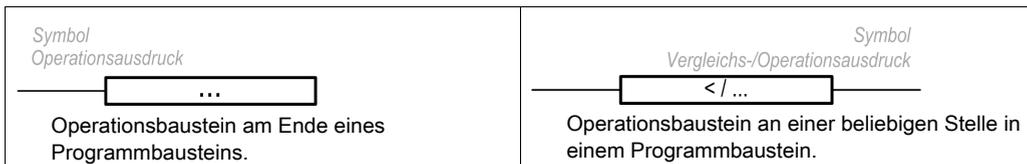
Wenn die Syntax der Vergleichsoperation in Anweisungsliste ungültig ist, wird die Umrandung des Felds **Vergleichsausdruck** rot dargestellt. Sie haben verschiedene Möglichkeiten, um Hilfe zu erhalten:

- Bewegen Sie die Maus über die Zeile **Vergleichsausdruck**., Oder
- Wählen Sie **Tools** → **Programmmeldungen** aus.

## Operationsbausteine

### Einfügen von AWL-Operationen und -Zuweisungsanweisungen in Kontaktplan

Verwenden Sie das Grafikelement **Operationsbaustein**, um Operationen und Zuweisungsanweisungen in Anweisungsliste in Kontaktplan-Programmbausteine einzufügen.



Das Grafikelement **Operationsbaustein** kann an jeder Position in einem Programmbaustein in Kontaktplan eingefügt werden, mit Ausnahme der ersten Spalte, da diese nicht als erster Kontakt in einem Programmbaustein verwendet werden kann.

Wenn mehr als ein Grafikelement **Operationsbaustein** in einem Programmbaustein in Kontaktplan verwendet wird, müssen diese in Reihe angeordnet werden. **Operationsbaustein**-Anweisungen können nicht parallel verwendet werden.

#### HINWEIS:

Wenn die Anwendung mit einer Funktionsebene (*siehe Seite 94*) von mindestens **Ebene 5.0** konfiguriert wird:

- Es können bis zu fünf Operanden und drei Ebenen von Klammern in einem Operationsbaustein verwendet werden. Die Operanden müssen vom selben Objekttyp sein: Wörter mit Wörtern, Gleitkommata mit Gleitkommata etc.
- Es müssen mindestens 20 Speicherwörter (%MW) verfügbar sein, um mehrere Operanden in der Master-Task verwenden zu können. Wenn in einer periodischen Task ebenfalls mehrere Operanden verwendet werden, müssen zusätzliche 20 Speicherwörter verfügbar sein.

Gehen Sie wie folgt vor, um ein Grafikelement Operationsbaustein in einen Programmbaustein in Kontaktplan einzufügen:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie auf die Schaltfläche <b>Operationsbaustein</b>  in der Symbolleiste.  |
| 2       | Klicken Sie auf eine beliebige Stelle innerhalb des Programmbausteins, um den <b>Operationsbaustein</b> einzufügen.  |
| 3       | Klicken Sie auf die Schaltfläche <b>Auswahlmodus</b>  in der Symbolleiste.  |
| 4       | Doppelklicken Sie auf die Zeile <b>Operationsausdruck</b> .<br><br>Die Schaltfläche Intelligente Code-Bearbeitung ( <i>siehe Seite 201</i> )  wird am Ende der Zeile angezeigt. Klicken Sie auf diese Schaltfläche, um Hilfe bei der Auswahl einer Funktion und der Syntax der Anweisung zu erhalten. |

| Schritt | Aktion   |
|---------|--|
| 5       | Geben Sie eine gültige AWL-Operation oder -Zuweisungsanweisung ein und drücken Sie die Eingabetaste.<br>Beispiel: <code>%MF10 := ((SIN( %MF12 + 60.0 ) + COS( %MF13 )) + %MF10 ) + 1.2</code><br>Der Ausdruck kann im Online-Modus geändert werden. Siehe Online-Änderungen<br><i>(siehe Seite 255).</i> |

**HINWEIS:** Mehrere Operanden-Ausdrücke können in Ereignis-Tasks nicht verwendet werden.

### Syntax der OPER-Anweisung

Die OPER-Anweisung entspricht einem Operationsbaustein, der sich an einer beliebigen Stelle in einem Programmbaustein befindet.

Die entsprechende OPER-Anweisung kann direkt in Programmbausteinen in Anweisungsliste verwendet werden.

OPER [ *Ausdruck* ] wobei *Ausdruck* ein beliebiger gültiger Ausdruck ist, der bis zu fünf Operanden und drei Ebenen von Klammern enthält. Beispiel:

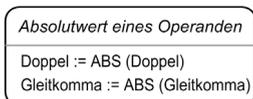
```
OPER [ %MF10 := ((SIN( %MF12 + 60.0 ) + COS( %MF13 )) + %MF10 ) + 1.2]
```

### Tooltips für die Intelligente Code-Bearbeitung in Kontaktplan

Um Sie bei der Auswahl von Funktionen zu unterstützen zeigt SoMachine Basic Tooltips an, während Sie Funktionsnamen in Operationsbausteine eingeben.

Es gibt zwei Arten von Tooltips:

- Eine Liste mit Funktionsnamen, die dynamisch aktualisiert wird mit den Funktionsnamen, die mit den eingegebenen Zeichen beginnen. Beispiel: Bei der Eingabe von "AS" wird ASCII\_TO\_FLOAT, ASCII\_TO\_INT und ASIN angezeigt.
- Hilfe bei der Syntax einer Funktion, die angezeigt wird, wenn Sie eine öffnende Klammer eingeben. Beispiel: Bei der Eingabe von "ABS(" wird folgendes angezeigt:



### Verwendung des Assistenten für Intelligente Code-Bearbeitung

Der Assistent für Intelligente Code-Bearbeitung wird angezeigt, wenn Sie auf die Schaltfläche

Intelligente Code-Bearbeitung  in der Zeile Operationsausdruck klicken:

Funktion einfügen
✕

---

✕

Nach Kategorie filtern Alle Typen

ABS

ACOS

ASCII\_TO\_FLOAT

ASCII\_TO\_INT

ASIN

**Absolutwert eines Operanden**

---

Doppel := ABS (Doppel)

Gleitkomma := ABS (Gleitkomma)

Funktion einfügen

Gehen Sie wie folgt vor:

| Schritt | Aktion   |
|---------|--|
| 1       | Optional kann die Liste nach Kategorie der Funktion gefiltert werden: <ul style="list-style-type: none"> <li>● Alle Typen</li> <li>● ASCII</li> <li>● Gleitkommawert</li> <li>● E/A-Objekte</li> <li>● Gleitkomma</li> <li>● Numerische Verarbeitung</li> <li>● Tabelle</li> <li>● PID</li> <li>● Benutzerdefinierte Funktion</li> </ul> |
| 2       | Wählen Sie eine Funktion aus, die dem Ausdruck hinzugefügt werden soll.  |
| 3       | Klicken Sie auf <b>Funktion einfügen</b> .   |

### Hilfe mit der Syntax

Wenn die Syntax der Operation oder Zuweisungsanweisung in Anweisungsliste ungültig ist, wird die Umrandung des Felds **Operationsausdruck** rot dargestellt. Sie haben verschiedene Möglichkeiten, um Hilfe zu erhalten:

- Bewegen Sie die Maus über die Zeile **Operationsausdruck**. Oder
- Wählen Sie **Tools** → **Programmmeldungen** aus.

## Hinzufügen von Kommentaren

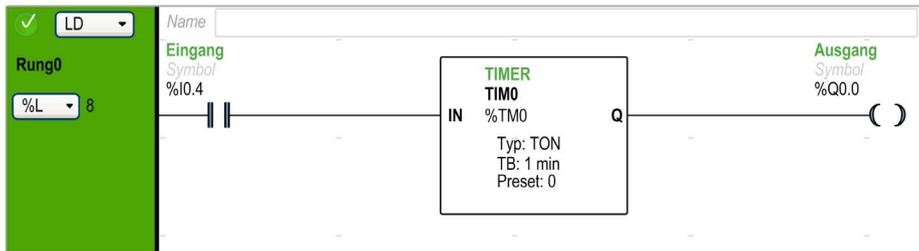
### Hinzufügen von Kommentaren in Kontaktplan

Gehen Sie zum Hinzufügen von Kommentaren in einem KOP-Programm vor wie folgt:

| Schritt | Aktion  |
|---------|---|
| 1       | Fügen Sie ein Grafikelement in den Programmbaustein ein.                                      |
| 2       | Falls erforderlich, klicken Sie auf den Auswahlzeiger oder drücken Sie die Esc-Taste.         |
| 3       | Doppelklicken Sie auf die Zeile <b>Kommentar</b> am oberen Rand des Grafikelements.           |
| 4       | Geben Sie einen Kommentar für das Grafikelement ein und drücken Sie die Eingabetaste (ENTER). |

### Beispiel für Kommentare in Kontaktplan

Die nachstehende Abbildung zeigt ein Beispiele für Kommentare in einem KOP-Programmbaustein:



## Best Practices für die Programmierung

### Programmsprünge

Setzen Sie Programmsprünge mit Vorsicht ein, um lange Schleifen zu vermeiden, die die Zykluszeit erhöhen können. Vermeiden Sie Sprünge zu Anweisungen, die sich vor der Sprunganweisung befinden.

**HINWEIS:** Eine vorgeschaltete Anweisung erscheint vor einem Sprung in einem Programm. Eine nachgeschaltete Anweisung erscheint nach einem Sprung in einem Programm.

### Programmierung von Ausgängen

Physische Ausgänge sowie logische Bits sollten im Programm nur einmal geändert werden. Bei physischen Ausgängen wird nur der letzte abgefragte Wert für die Aktualisierung der Ausgänge berücksichtigt.

### Verwenden direkt verdrahteter Not-Aus-Sensoren

Sensoren, die als Not-Aus-Einrichtungen genutzt werden, dürfen nicht vom Logic Controller verarbeitet werden. Sie müssen direkt mit den entsprechenden Ausgängen verbunden und gemäß örtlichen, nationalen und/oder internationalen Vorschriften angewandt werden.

### Verwalten der Spannungsrückkehr

Veranlassen Sie, dass die Rückkehr der Spannung nur bei manuellem Betrieb erfolgt. Ein automatischer Neustart kann einen unbeabsichtigten Gerätebetrieb in der Installation auslösen (verwenden Sie die Systembits %S0, %S1 und %S49). Andere Systembits und -wörter können unter Umständen ebenfalls zur Verwaltung eines Neustarts nach einem Spannungsausfall verwendet werden. Siehe die Systembits (%S) und Systemwörter (%SW) (*siehe Modicon M221, Logic Controller, Programmierhandbuch*).

## WARNUNG

### UNBEABSICHTIGTER GERÄTEBETRIEB

Verwenden Sie das mit dieser Software konfigurierte und programmierte Gerät nicht für sicherheitskritische Maschinenfunktionen, sofern das Gerät nicht anderweitig explizit für einen Einsatz zur Funktionssicherheit ausgewiesen ist und allen geltenden Vorschriften und Normen entspricht.

**Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.**

### Verwaltung der Zeit- und Abarbeitungsbausteine

Der Status des Systembits %S51, das alle Fehler der Echtzeituhr (RTC) angibt, sollte überprüft werden.

### Syntaxüberprüfung

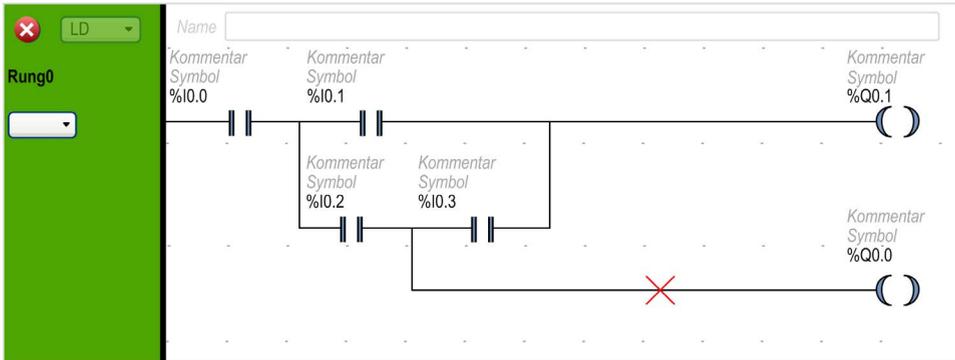
Während Sie programmieren, überprüft SoMachine Basic die Syntax von Anweisungen, Operanden sowie deren Verknüpfungen.

### Zusätzliche Hinweise zur Verwendung von Klammern

Setzen Sie Zuweisungsanweisungen nicht im Klammern:

```
LD    %I0.0
MPS
AND   %I0.1
OR (  %I0.2
)
```

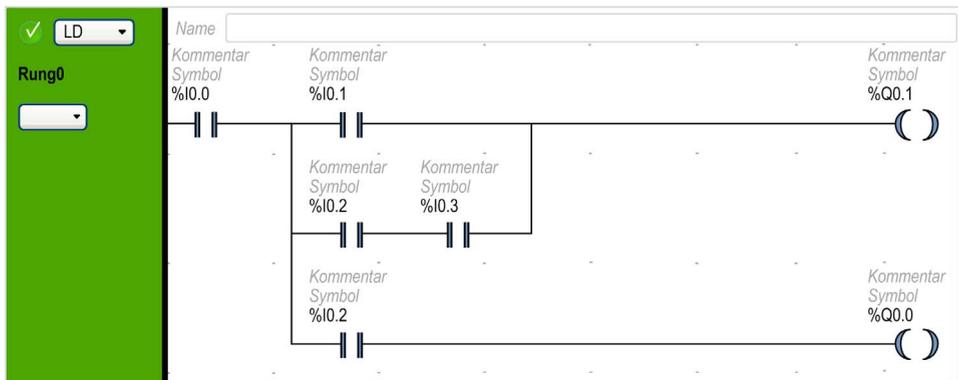
Das entsprechende Diagramm in Kontaktplan löst einen Kurzschluss aus:



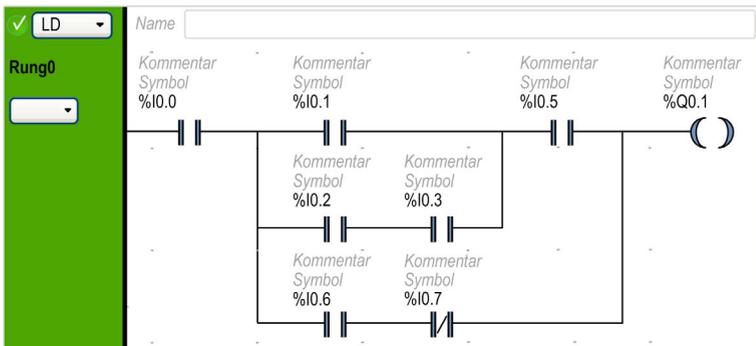
Um die gleiche Funktion zu erfüllen, programmieren Sie die Anweisungen wie folgt:

```
LD    %I0.0
MPS
AND ( %I0.1
OR (  %I0.2
AND  %I0.3
)
)
ST   %Q0.1
MPP
AND  %I0.2
ST   %Q0.0
```

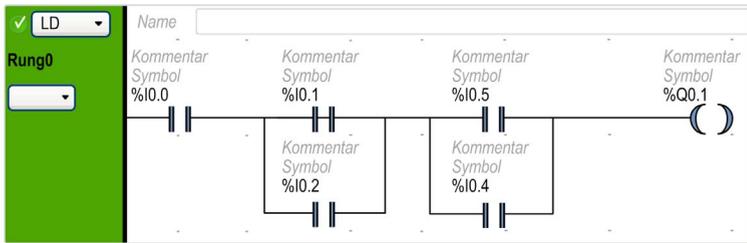
Die Entsprechung in Kontaktplan:



Wenn verschiedene Kontakte parallel zueinander sind, verschachteln Sie sie miteinander:



Oder die Kontakte müssen wie folgt vollständig voneinander getrennt werden:



---

## Abschnitt 6.13

### Programmieren in Anweisungsliste

---

#### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                                       | Seite |
|---|-------|
| Übersicht über Programme in Anweisungsliste | 210   |
| Funktionsweise der Anweisungsliste          | 213   |
| AWL-Anweisungen                             | 214   |
| Verwenden von Klammern                      | 218   |

## Übersicht über Programme in Anweisungsliste

### Einführung

Ein in Anweisungsliste (AWL oder IL: Instruction List) geschriebenes Programm besteht aus einer Folge von Anweisungen, die vom Logic Controller der Reihe nach ausgeführt werden. Jede Anweisung wird durch eine einzelne Programmzeile dargestellt und besteht aus folgenden Komponenten:

- Zeilennummer
- Aktueller Wert (nur im Online-Modus)
- Anweisungsoperator
- Operand(en)
- Optionaler Kommentar

### Beispiel für ein Programm in Anweisungsliste

Nachstehend ein Beispiel für ein AWL-Programm:

| IL                               |      | Name  |       |   |
|----------------------------------|------|-------|-------|---|
| Rung0                            | 0000 | LD    | %M1   | Bit 1 laden                             |
| <input type="checkbox"/> Symbole | 0001 | AND ( | %I0.1 | Abzweig starten und Eingangsbit 1 laden |
|                                  | 0002 | OR (  | %I0.2 | Eingangsbit 2 laden                     |
|                                  | 0003 | ANDN  | %I0.3 | Eingangsbit 3 laden und invertieren     |
|                                  | 0004 | )     |       | Kommentar                               |
|                                  | 0005 | )     |       | Kommentar                               |
|                                  | 0006 | ST    | %Q0.0 | Ausgangsbit 0 setzen                    |

### Zeilennummern

Die 4-stelligen Zeilennummern werden bei der Erstellung einer neuen Programmzeile generiert und von SoMachine Basic automatisch verwaltet.

### Aktuelle Werte

Wenn sich SoMachine Basic im Online-Modus (*siehe Seite 31*) befindet (aktive Verbindung zum Logic Controller und Programm in Betrieb), zeigt SoMachine Basic den aktuellen Wert aller Objekttypen im Fenster des AWL-Editors an.

Die angezeigten Werte dieser Objekte werden aktualisiert.

## Anweisungsoperatoren

Der Anweisungscode ist ein mnemotechnisches Symbol, das als Operator bezeichnet wird und das die mit den Operanden durchzuführende Operation bestimmt. Typische Operatoren beziehen sich auf boolesche und numerische Operationen.

Beispiel: Im obigen Beispielprogramm steht `LD` ist die mnemonische Kurzbezeichnung für den `LOAD`-Operator. Die `LOAD`-Anweisung legt den Wert des Operanden `%M1` in einem internen Register ab, das als Akkumulator bezeichnet wird (bzw. lädt ihn in dieses Register).

Grundsätzlich gibt es 2 Typen von Operatoren:

- Testoperatoren  
Diese Operatoren richten die zur Durchführung einer Aktion erforderlichen Bedingungen ein bzw. testen sie. Beispielsweise `LOAD (LD)` und `AND`.
- Aktionsoperatoren  
Diese Operatoren führen Aktionen auf der Grundlage der vorangegangenen Logik aus.  
Beispiele: Zuweisungsoperatoren wie `STORE (ST)` und `RESET (R)`.

Operatoren in Verbindung mit Operanden bilden Anweisungen.

## Operanden

Ein Operand ist ein Objekt, eine Adresse oder ein Symbol, das bzw. die einen Wert darstellt, der von einem Programm in einer Anweisung bearbeitet werden kann. Im obigen Beispielprogramm ist der Operand `%M1` eine Adresse, der der Wert eines integrierten Eingangs der Logiksteuerung zugewiesen ist. Eine Anweisung kann über 0 bis 3 Operanden verfügen, je nach Typ des Anweisungsoperators.

Operanden können Folgendes darstellen:

- Steuerungseingänge und -ausgänge wie Sensoren, Drucktaster und Relais
- Vordefinierte Systemfunktionen wie Timer und Zähler
- Arithmetische, logische, numerische und Vergleichsoperationen
- Steuerungsinterne Variablen wie Systembits und -wörter

## Kommentare

Gehen Sie zum Hinzufügen von Kommentaren in einem AWL-Programm vor wie folgt:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie nach Bedarf in das Kommentarfeld, das am oberen Rand des Programmbausteins über der ersten Zeile 0000 angezeigt wird, und geben Sie einen Kommentar für den Programmbaustein ein. |
| 2       | Fügen Sie eine Anweisungszeile ein.   |
| 3       | Klicken Sie in den Bereich <b>Kommentar</b> rechts neben der Anweisung.   |
| 4       | Geben Sie einen Kommentar ein und drücken Sie die Eingabetaste ( <code>Enter</code> ).  |

### Anpassen des Kontaktplan/IL-Editors



Verwenden Sie die folgenden Objekte am oberen Ende des IL-Editors, um den Inhalt des Editors anzupassen:

| Objekt                                | Beschreibung   |
|---------------------------------------|--|
| <b>IL &gt; LD</b>                     | Wechselt die Anzeige aller Programmbausteine von IL zu Kontaktplan (Ladder).   |
| <b>LD &gt; IL</b>                     | Wechselt die Anzeige aller Programmbausteine von Kontaktplan (Ladder) zu IL.   |
| <b>-</b>                              | Löscht eine Spalte aus dem IL-Gitter. Die Schaltfläche wird deaktiviert, wenn die minimale Anzahl an Spalten (11) erreicht wurde.  |
| <b>+</b>                              | Fügt dem IL-Gitter eine Spalte hinzu. Die Schaltfläche wird deaktiviert, wenn die maximale Anzahl an Spalten (30) erreicht wurde.  |
| <b>Kommentare anzeigen/ausblenden</b> | Klicken zum Anzeigen/Ausblenden der Kommentare in den Programmbausteinen.  |
| <b>T</b>                              | Klicken, um Objekte abwechselnd in Adressmodus oder Symbolmodus anzuzeigen.  |
| <b>DEZ/HEX</b>                        | Nur im Online-Modus aktiv. Klicken, um die numerischen Werte in den Programmbausteinen abwechselnd im Dezimal- und Hexadezimalformat anzuzeigen.   |
| <b>1 – Neue POU</b>                   | Doppelklicken zum Bearbeiten des Standardnamens der <b>POU</b> , der im Bereich <b>Tools</b> → <b>Master task</b> des Bildschirms erscheint.   |
| <b>Kommentar</b>                      | Doppelklicken zum Schreiben von Text als Kommentar in dieser <b>POU</b> .  |
| <b>Zoom-Schieberegler</b>             | Vergrößern oder Verkleinern des Kontaktplan-Editors.<br>Vergrößern oder verkleinern mithilfe des Tastaturkürzels <b>Strg + Mausrad</b> . Die Zoomposition wird auch dann beibehalten, wenn Sie durch das Projekt navigieren. |

## Funktionsweise der Anweisungsliste

### Einführung

Binäre AWL-Anweisungen haben normalerweise nur einen expliziten Operanden, der andere Operand ist impliziert. Der implizierte Operand ist der Wert im booleschen Akkumulator. Beispiel: Bei der Anweisung `LD %I0.1` ist `%I0.1` der explizite Operand. Ein impliziter Operand wird in den Akkumulator geladen und der vorherige Wert des Akkumulators wird durch den Wert von `%I0.1` überschrieben. Dieser Wert wird nun zum impliziten Wert für die darauffolgende Anweisung.

### Funktionsweise

Eine AWL-Anweisung führt eine bestimmte Operation mit dem Inhalt des Akkumulators und dem expliziten Operanden aus. Der Inhalt des Akkumulators wird durch das Ergebnis ersetzt. Beispiel: Die Operation `AND %I1.2` führt eine logische UND-Verknüpfung mit dem Inhalt des Akkumulators und dem Eingang `1.2` aus und ersetzt den Inhalt des Akkumulators durch das Ergebnis.

Alle booleschen Anweisungen mit Ausnahme von `Load`, `Store` und `Not` verwenden 2 Operanden. Der Wert der beiden Operanden kann entweder „True“ oder „False“ sein. Die Programmausführung der Anweisungen erzeugt einen einzigen Wert: „True“ oder „False“. Die Ladeanweisungen `Load` platzieren den Wert des Operanden im Akkumulator, während die Speicheranweisungen `Store` den Wert im Akkumulator in den Operanden übertragen. Die Anweisung `Not` hat keine expliziten Operanden und invertiert einfach den Zustand des Akkumulators.

### Unterstützte AWL-Anweisungen

Die folgende Tabelle zeigt einige Anweisungen in der AWL-Sprache:

| Typ der Anweisung   | Beispiel                          | Funktion  |
|---------------------|-----------------------------------|---|
| Boolesche Anweisung | <code>LD %M10</code>              | Lädt den Wert des internen Bits <code>%M10</code> in den Akkumulator. |
| Bausteinanweisung   | <code>IN %TM0</code>              | Startet den Timer <code>%TM0</code> .                                 |
| Wortanweisung       | <code>[%MW10 := %MW50+100]</code> | Additionsoperation  |
| Programmanweisung   | <code>SR5</code>                  | Ruft das Unterprogramm #5 auf.  |

## AWL-Anweisungen

### Einführung

Die Sprache „Anweisungsliste“ (AWL, engl.: Instruction List bzw. IL) besteht aus den folgenden Typen von Anweisungen bzw. Blöcken von Anweisungen:

- Testanweisungen
- Aktionsanweisungen
- Funktionsbausteine

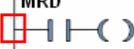
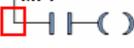
In diesem Abschnitt werden die Anweisungen für die Programmierung in Anweisungsliste aufgeführt und beschrieben.

### Testanweisungen

Die folgende Tabelle beschreibt die Testanweisungen in der Sprache AWL.

| Kurzbezeichnung | Name         | Entsprechendes Grafikelement  | Funktion  |
|-----------------|--------------|---|---|
| LD              | Load         |    | Lädt den booleschen Wert des Operanden in den Akkumulator.  |
| LDN             | Load Not     |    | Lädt den negierten booleschen Wert des Operanden in den Akkumulator.  |
| LDR             | Load Rising  |    | Lädt den booleschen Wert des Operanden in den Akkumulator, wenn der Wert sich von 0 auf 1 ändert (steigende Flanke). Der Wert des Akkumulators wird sodann so lange mit 0 geladen, bis die nächste Transition des Operanden von 0 auf 1 stattfindet.  |
| LDF             | Load Falling |  | Lädt den booleschen Wert des Operanden in den Akkumulator, wenn der Wert sich von 1 auf 0 ändert (fallende Flanke). Der Wert des Akkumulators wird sodann so lange mit 1 geladen, bis die nächste Transition des Operanden von 1 auf 0 stattfindet.   |
| AND             | And          |  | Das boolesche Ergebnis ist gleich der AND-Verknüpfung zwischen dem booleschen Ergebnis der vorherigen Anweisung (das im Akkumulator gespeichert ist) und dem Status des Operanden. Das Ergebnis der Anweisung selbst wird daraufhin implizit in den Akkumulator geladen und überschreibt den vorherigen Wert. |

| Kurzbezeichnung | Name        | Entsprechendes Grafikelement | Funktion  |
|-----------------|-------------|------------------------------|---|
| ANDN            | And Not     |                              | Das boolesche Ergebnis ist gleich der AND-Verknüpfung zwischen dem booleschen Ergebnis der vorherigen Anweisung (das im Akkumulator gespeichert ist) und dem umgekehrten (negierten) Status des Operanden. Das Ergebnis der Anweisung selbst wird daraufhin implizit in den Akkumulator geladen und überschreibt den vorherigen Wert. |
| ANDR            | And Rising  |                              | Das boolesche Ergebnis ist gleich der AND-Logikverknüpfung zwischen dem booleschen Ergebnis der vorherigen Anweisung und der Erkennung der steigenden Flanke des Operanden (1 = steigende Flanke). Das Ergebnis der Anweisung selbst wird daraufhin implizit in den Akkumulator geladen und überschreibt den vorherigen Wert.         |
| ANDF            | And Falling |                              | Das boolesche Ergebnis ist gleich der AND-Logikverknüpfung zwischen dem booleschen Ergebnis der vorherigen Anweisung und der Erkennung der fallenden Flanke des Operanden (1 = fallende Flanke). Das Ergebnis der Anweisung selbst wird daraufhin implizit in den Akkumulator geladen und überschreibt den vorherigen Wert.           |
| OR              | Or          |                              | Das boolesche Ergebnis ist gleich der OR-Verknüpfung zwischen dem booleschen Ergebnis der vorherigen Anweisung und dem Status des Operanden (der im Akkumulator gespeichert ist).   |
| AND(            | And With    |                              | Logisches AND (max. 32 Ebenen von Klammern). Die Klammern geben ein logisches Zwischenergebnis der darin befindlichen Anweisungen an. Dieses Ergebnis wird dann mit der Wert im Akkumulator logisch mit AND verknüpft.  |
| OR(             | Or With     |                              | Logisches OR (max. 32 Ebenen von Klammern). Die Klammern geben ein logisches Zwischenergebnis der darin befindlichen Anweisungen an. Dieses Ergebnis wird dann mit der Wert im Akkumulator logisch mit OR verknüpft.  |

| Kurzbezeichnung             | Name  | Entsprechendes Grafikelement   | Funktion                                     |
|-----------------------------|---|--|--|
| XOR<br>XORN<br>XORR<br>XORF | Ex Or<br>Ex Or Not<br>Ex Or Rising<br>Ex Or Falling | <br><br><br> | Exklusives ODER                              |
| MPS<br>MRD<br>MPP           | Memory Push<br>Store<br>Memory ReaD<br>Memory PoP   | <br><br>  | Verzweigungsoperatoren für Ausgangsaktionen. |
| N                           | Not   |   | Keht den Wert des Operanden um.              |

### Aktionsanweisungen

Die folgende Tabelle beschreibt die Aktionsanweisungen in der Sprache AWL.

| Kurzbezeichnung | Name       | Entsprechendes Grafikelement  | Funktion   |
|-----------------|------------|---|--|
| ST              | Store      |  | Der zugehörige Operand nimmt den Wert des Testbereichsergebnisses an.                |
| STN             | Store Not  |  | Der zugehörige Operand nimmt den umgekehrten Wert des Testbereichsergebnisses an.    |
| S               | Set        |  | Der zugehörige Operand wird auf 1 gesetzt, wenn das Ergebnis des Testbereichs 1 ist. |
| R               | Reset      |  | Der zugehörige Operand wird auf 0 gesetzt, wenn das Ergebnis des Testbereichs 1 ist. |
| JMP             | Jump       |  | Unbedingte Verbindung zu einer markierten Sequenz, vor oder nach dieser Anweisung.   |
| SRn             | Subroutine |  | Verbindung am Anfang eines Unterprogramms (Unterprogrammaufruf).                     |

| Kurzbezeichnung | Name            | Entsprechendes Grafikelement  | Funktion   |
|-----------------|-----------------|---|--|
| END             | End             |  | Ende des Programms.  |
| ENDCN           | End Conditional |  | Bedingtes Beenden des Programms bei einem booleschen Ergebnis von 0. |

### Funktionsbausteine

Die folgende Tabelle beschreibt die Funktionsbausteine in der Sprache AWL.

| Name                         | Entsprechendes Grafikelement  | Funktion  |
|------------------------------|---|---|
| Timer, Zähler, Register usw. |  | <p>Für jeden Funktionsbaustein sind Anweisungen zur Steuerung des Bausteins verfügbar.</p> <p>Eine strukturierte Form wird zur Verbindung der Eingänge und Ausgänge des Bausteins verwendet.</p> <p><b>Hinweis:</b> Ausgänge von Funktionsbausteinen können nicht miteinander verbunden werden (vertikale Kurzschlüsse).</p> <p>Weitere Informationen finden Sie unter Softwareobjekte (<i>siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen</i>).</p> |

## Verwenden von Klammern

### Einführung

Bei logischen AND- und OR-Operatoren werden Klammern zur Verschachtelung von logischen Anweisungen verwendet. Auf diese Weise geben sie Abweichungen (Verzweigungen) im KOP-Editor an. Klammern werden wie folgt mit Anweisungen verbunden:

- Das Öffnen der Klammern wird mit dem AND- oder OR-Operator verknüpft.
- Das Schließen der Klammer entspricht einer Anweisung (einem Operator ohne Operand), die für jede offene Klammer erforderlich ist.

### Beispiel mit einer AND-Anweisung

Die folgenden Beispiele illustrieren die Verwendung von Klammern mit einer AND-Anweisung:

| Programmbaustein | Anweisung  |
|------------------|--|
| 0                | LD %I0.0<br>AND %I0.1<br>OR %I0.2<br>ST %Q0.0        |
| 1                | LD %I0.0<br>AND ( %I0.1<br>OR %I0.2<br>)<br>ST %Q0.1 |

**HINWEIS:** Die Entsprechung in Kontaktplan finden Sie unter Umkehrbarkeit (*siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen*).

### Beispiel mit einer Anweisung OR

Das folgende Beispiel illustriert die Verwendung von Klammern mit einer OR-Anweisung:

| Programmbaustein | Anweisung   |
|------------------|---|
| 0                | LD %I0.0<br>AND %I0.1<br>OR ( %I0.2<br>AND %I0.3<br>)<br>ST %Q0.0 |

**HINWEIS:** Die Entsprechung in Kontaktplan finden Sie unter Umkehrbarkeit (*siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen*).

## Modifikatoren

Die folgende Tabelle führt Modifikatoren auf, die den Klammern zugewiesen werden können.

| Modifikator | Funktion         | Beispiel                     |
|-------------|------------------|------------------------------|
| N           | Negation         | AND(N oder OR(N              |
| F           | Fallende Flanke  | AND(F oder OR(F              |
| R           | Steigende Flanke | AND(R oder OR(R              |
| [           | Vergleich        | Siehe Vergleichsanweisungen. |

**HINWEIS:** Der Modifikator '[' kann auch in Verbindung mit anderen Anweisungen eingesetzt werden und als Operator dienen. Weitere Möglichkeiten zur Verwendung von '[' in anderen Anweisungen finden Sie unter Einführung in numerische Operationen.

## Verschachteln von Klammern

Sie können Klammern auf bis zu 32 Ebenen verschachteln.

Beachten Sie bei der Verschachtelung von Klammern folgende Richtlinien:

- Jede geöffnete Klammer muss über eine entsprechende schließende Klammer verfügen.
- Sprungmarken (%Li:), Unterprogramme (SRi:), JMP-Sprunganweisungen (JMP) und Funktionsbausteinanweisungen dürfen nicht in Ausdrücke zwischen Klammern gesetzt werden.
- Die Speicheranweisungen (ST, STN, S und R) dürfen nicht zwischen Klammern programmiert werden.
- Die Stapelanweisungen (MPS, MRD und MPP) dürfen nicht zwischen Klammern verwendet werden.

## Beispiele für die Verschachtelung von Klammern

Die nachstehenden Beispiele illustrieren die Verschachtelung von Klammern:

| Programmbaustein | Anweisung  |
|------------------|--|
| 0                | LD %I0.0<br>AND ( %I0.1<br>OR (N %I0.2<br>AND %M3<br>)<br>)<br>)<br>ST %Q0.0 |

| Programmbaustein | Anweisung  |
|------------------|--|
| 1                | <pre> LD    %I0.1 AND ( %I0.2 OR (  %I0.5 AND  %I0.6 ) AND  %I0.4 OR (  %I0.7 AND  %I0.8 ) ) ) ST   %Q0.0                     </pre> |

**HINWEIS:** Die Entsprechung in Kontaktplan finden Sie unter Umkehrbarkeit (*siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen*).

---

# Abschnitt 6.14

## Programmierung mit Grafcet (Liste)

---

### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema  | Seite |
|--|-------|
| Beschreibung der Programmierung mit Grafcet (Liste)                          | 222   |
| Grafcet-Programmstruktur (Liste)   | 224   |
| Verwendung von Grafcet-Anweisungen (Liste) in einem SoMachine Basic-Programm | 228   |

## Beschreibung der Programmierung mit Grafcet (Liste)

### Einführung

Grafcet- (Listen-)Programmierung in SoMachine Basic bietet eine einfache Methode, um eine Steuerungssequenz in Schritte zu übersetzen. Sie können Steuerungssequenzen in Grafcet-Schritte übersetzen und diese Schritte dann mittels Grafcet-Anweisungen in einem Programm verwenden.

Die maximale Anzahl an Grafcet-Schritten richtet sich nach dem Typ der Steuerung. Die Anzahl der Schritte, die zu einem beliebigen Zeitpunkt aktiv sein dürfen, ist nur durch die Gesamtzahl der Schritte begrenzt.

### Grafcet-Anweisungen

Ein SoMachine Basic Grafcet-Programm verfügt über folgende Anweisungen:

| Operator | Operand          | IL-Anweisung | Anweisungsname                               | Grafische Entsprechung  | Beschreibung  |
|----------|------------------|--------------|--|---|---|
| =*=      | x                | =*= x        | INITIAL STEP                                 |    | Diese Anweisung definiert den ersten Schritt in dem Programm.   |
| =*= POST | Nicht zutreffend | =*= POST     | POST PROCESSING (impliziter Operand)         |    | Diese Anweisung definiert die Nachverarbeitung sowie die sequenzielle Endverarbeitung.                      |
| -*-      | x                | -*- x        | STEP   |    | Diese Anweisung definiert im Programm einen Schritt zur Transitionsvalidierung.                             |
| #        | Nicht zutreffend | #            | DEACTIVATE CURRENT STEP (impliziter Operand) |    | Diese Anweisung deaktiviert den ersten Schritt in dem Programm.   |
| #        | x                | #x           | DEACTIVATE CURRENT STEP und ACTIVATE STEP x  |  | Diese Anweisung deaktiviert den aktuellen Schritt und aktiviert Schritt x in dem Programm.                  |
| #D       | x                | #D x         | DEACTIVATE CURRENT STEP und STEP x           |  | Diese Anweisung deaktiviert den aktuellen Schritt sowie Schritt x in dem Programm.                          |
| S        | x                | S x          | ACTIVATE STEP x                              |  | Diese Anweisung aktiviert Schritt x im Programm. Diese Aktion hat keinen Effekt auf andere aktive Schritte. |

**x** Grafcet-Schrittnummer (eine Ganzzahl beginnend bei 1).

| Operator  | Operand | IL-Anweisung | Anweisungsname    | Grafische Entsprechung  | Beschreibung  |
|---|---------|--------------|-------------------|---|---|
| R   | x       | R x          | DEACTIVATE STEP x |  | Diese Anweisung deaktiviert Schritt x im Programm. Diese Aktion hat keinen Effekt auf andere aktive Schritte. |
| <b>x</b> Grafcet-Schrittnummer (eine Ganzzahl beginnend bei 1). |         |              |                   |   |   |

## Grafcet-Programmstruktur (Liste)

### Einführung

Ein SoMachine Basic-Grafcet-Programm (Liste) besteht aus folgenden Teilen:

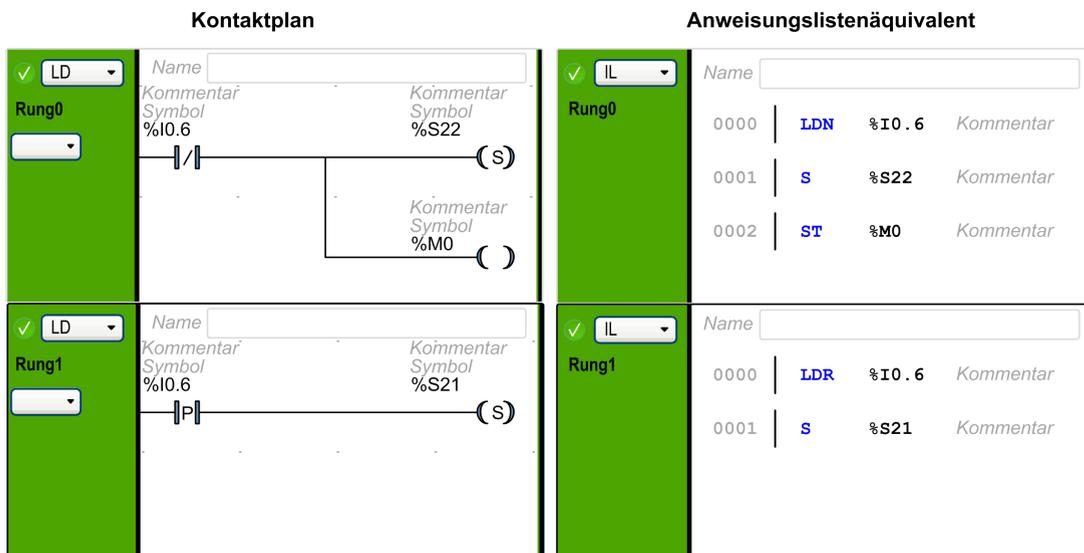
- Vorverarbeitung
- Sequenzielle Verarbeitung
- Nachverarbeitung

### Vorverarbeitung

Während der Vorverarbeitung werden folgende Elemente generiert:

- Wiederkehr der Spannungsversorgung
- Fehlerverwaltung
- Änderungen des Betriebszustands
- Vorpositionierung von Grafcet-Schritten
- Eingangslogik

In diesem Beispiel wird das Systembit %S21 auf 1 gesetzt und wird mit der steigenden Flanke an Eingang %I0.6 (Rung1) verwendet. Dadurch werden die aktiven Schritte deaktiviert und die Initialschritte aktiviert:



Die Vorverarbeitung beginnt mit der ersten Zeile des Programms und endet mit dem ersten Auftreten einer =\*=- oder -\*--Anweisung.

Die Systembits %S21, %S22 und %S23 sind für die Grafcet-Steuerung reserviert. Jedes dieser Systembits wird während der Vorverarbeitung von der Anwendung auf 1 gesetzt (falls notwendig). Die zugehörige Funktion wird vom System am Ende der Vorverarbeitung ausgeführt, und das Systembit wird dann vom System auf 0 zurückgesetzt.

| Systembit | Name                           | Beschreibung  |
|-----------|--------------------------------|---|
| %S21      | Initialisierung des Grafcet    | Alle aktiven Schritte werden deaktiviert und die Initialschritte werden aktiviert.  |
| %S22      | Neuinitialisierung des Grafcet | Alle Schritte werden deaktiviert.   |
| %S23      | Vorpositionierung des Grafcet  | Dieses Bit muss auf 1 gesetzt werden, wenn die %Xi-Objekte während der Vorverarbeitung explizit durch die Anwendung geschrieben werden. Wenn dieses Bit während der Vorverarbeitung ohne explizite Änderung des Objekts %Xi auf 1 gehalten wird, friert Grafcet ein (Aktualisierungen werden nicht berücksichtigt). |

### Sequenzielle Verarbeitung

Die sequenzielle Verarbeitung wird in Ablaufsprache ausgeführt (Anweisungen, die den Ablauf repräsentieren):

- Schritte
- Zu Schritten gehörige Aktionen
- Übergänge
- Übergangsbedingungen

Beispiel:

### Kontaktplan

|                        |  |                           |                         |  |
|------------------------|--|---------------------------|-------------------------|--|
| ✓ LD<br>Rung0<br>== 1  | Name <input type="text"/><br>Kommentar<br>Symbol %I0.2 | Kommentar<br>Symbol %I0.3 | Kommentar<br>Symbol %X2 |  |
| ✓ LD<br>Rung1<br>1     | Name <input type="text"/><br>Kommentar<br>Symbol %I0.3 | Kommentar<br>Symbol %I0.2 | Kommentar<br>Symbol %X3 |  |
| ✓ LD<br>Rung2<br>-*- 2 | Name <input type="text"/><br>Kommentar<br>Symbol %I0.4 | Kommentar<br>Symbol %X1   |                         |  |
| ✓ LD<br>Rung3<br>-*- 3 | Name <input type="text"/><br>Kommentar<br>Symbol %I0.5 | Kommentar<br>Symbol %X1   |                         |  |

### Anweisungslistenäquivalent

|                          |  |
|--------------------------|--|
| ✓ IL<br>Rung0<br>Symbole | Name <input type="text"/><br>0000 == 1 Kommentar<br>0001 LD %I0.2 Kommentar<br>0002 ANDN %I0.3 Kommentar<br>0003 # 2 Kommentar |
| ✓ IL<br>Rung1<br>Symbole | Name <input type="text"/><br>0000 LD %I0.3 Kommentar<br>0001 ANDN %I0.2 Kommentar<br>0002 # 3 Kommentar                        |
| ✓ IL<br>Rung2<br>Symbole | Name <input type="text"/><br>0000 -*- 2 Kommentar<br>0001 LD %I0.4 Kommentar<br>0002 # 1 Kommentar                             |
| ✓ IL<br>Rung3<br>Symbole | Name <input type="text"/><br>0000 -*- 3 Kommentar<br>0001 LD %I0.5 Kommentar<br>0002 # 1 Kommentar                             |

Die sequenzielle Verarbeitung endet mit der Ausführung der **POST**-Anweisung oder der Beendigung des Programms.

## Nachverarbeitung

Während der Nachverarbeitung werden folgende Elemente generiert:

- Befehle zur sequenziellen Verarbeitung für die Regelung der Ausgänge
- Ausgangsspezifische Verriegelungen

Beispiel:

### Kontaktplan

|                               |   |
|-------------------------------|---|
| ✓ LD<br><b>Rung0</b><br>== 1  | Name <input type="text"/><br>Kommentar<br>Symbol %I0.2    Kommentar<br>Symbol %I0.3    Kommentar<br>Symbol %X2        Kommentar<br> |
| ✓ LD<br><b>Rung1</b><br>-.- 2 | Name <input type="text"/><br>Kommentar<br>Symbol %I0.4    Kommentar<br>Symbol %Q0.2    Kommentar<br>                                |
| ✓ LD<br><b>Rung2</b><br>POST  | Name <input type="text"/><br>Kommentar<br>Symbol %X1        Kommentar<br>Symbol %Q0.1    Kommentar<br>                              |

### Anweisungslistenäquivalent

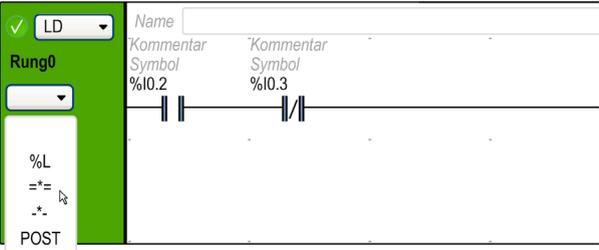
|  |   |
|--|---|
| ✓ IL<br><b>Rung0</b><br><input type="checkbox"/> Symbole | Name <input type="text"/><br>0000    ==*    1        Kommentar<br>0001    LD     %I0.2    Kommentar<br>0002    ANDN %I0.3    Kommentar<br>0003    #        2        Kommentar |
| ✓ IL<br><b>Rung1</b><br><input type="checkbox"/> Symbole | Name <input type="text"/><br>0000    -*-    2        Kommentar<br>0001    LD     %I0.4    Kommentar<br>0002    ST     %Q0.2    Kommentar                                      |
| ✓ IL<br><b>Rung2</b><br><input type="checkbox"/> Symbole | Name <input type="text"/><br>0000    ==*    POST    Kommentar<br>0001    LD     %X1        Kommentar<br>0002    ST     %Q0.1    Kommentar                                     |

## Verwendung von Grafcet-Anweisungen (Liste) in einem SoMachine Basic-Programm

**HINWEIS:** Grafcet-Anweisungen (Liste) können nur in der Master-Task eines Programms verwendet werden.

### Erzeugen von Grafcet-Schritten (Liste) im Kontaktplan

Zum Erzeugen von Grafcet-Schritten in einem Programm gehen Sie wie folgt vor:

| Schritt | Aktion  |
|---------|---|
| 1       | <p>Wählen Sie in einer POU einen Programmbaustein aus und klicken Sie auf die Dropdown-Schaltfläche unter der zugehörigen Sequenz-ID <b>Programmbaustein</b><sub>x</sub>, wobei <i>x</i> die Nummer des Programmbausteins in der POU darstellt.</p>  <p><b>Ergebnis:</b> Es wird ein Menü mit einer Liste der Grafcet-Anweisungen angezeigt.</p>   |
| 2       | <p>Klicken Sie auf eine Anweisung in der Liste, um den Programmbaustein als Initialschritt, Vorverarbeitung oder Schritt des Grafcet-Programms (Liste) zu definieren.</p> <p><b>Ergebnis:</b> Der Programmbaustein wird für eine Grafcet-Anweisung festgelegt. Der Operator der Anweisung erscheint auf der Schaltfläche und der Operand (Schrittnummer) wird als Suffix angezeigt.</p> <p><b>HINWEIS:</b> Die Schrittnummer erhöht sich mit der Definition der nächsten STEP- oder INITIAL STEP-Anweisung um 1. Sie können nur eine POST-Anweisung in einem Programm definieren. Daher weist die POST-Anweisung keine Schrittnummer auf.</p> <p>Zum Ändern der Schrittnummer eines Programmbausteins doppelklicken Sie darauf, geben die neue Nummer ein und drücken die Eingabetaste.</p> |

### Aktivieren oder Deaktivieren von Grafcet-Schritten (Liste) im Kontaktplan

Zum Aktivieren oder Deaktivieren von Grafcet-Schritten (Liste) in einem Programm gehen Sie wie folgt vor:

| Schritt | Aktion  |
|---------|---|
| 1       | Wählen Sie in einer POU einen Programmbaustein im Programm aus.   |
| 2       | <p>Klicken Sie auf  (zur Deaktivierung des aktuellen Schritts und optionaler Aktivierung eines angegebenen Schritts) oder  (zur Deaktivierung des aktuellen Schritts sowie zur Deaktivierung des angegebenen Schritts) und fügen Sie dieses Element in den Aktionsbereich des Programmbausteins ein (siehe Einfügen eines Grafikelements <i>(siehe Seite 192)</i>).</p>   |
| 3       | <p>Alternativ dazu können Sie auch die Anweisungen <b>ACTIVATE</b> oder <b>DEACTIVATE</b> in dem Programmbaustein verwenden, indem Sie <b>ALT+A</b> bzw. <b>ALT+D</b> drücken.<br/> <b>Ergebnis:</b> Das Kontaktplansymbol für Aktivierung bzw. Deaktivierung wird im Aktionsbereich angezeigt.<br/>                     Drücken Sie die Eingabetaste, um das Element einzufügen.</p>   |
| 4       | <p>Doppelklicken Sie im Programmbaustein auf dem Grafcet-Aktivierungs- bzw. -Deaktivierungssymbol auf das Feld <b>Adresse</b> und geben Sie die Grafcet-Bitadresse ein („%Xi“, wobei „i“ die Schrittnummer darstellt).<br/>                     Beispielsweise verweist %X4 auf Schritt 4 des Grafcet-Programms. Wenn %X4 die Adresse für das Deaktivierungssymbol ist und der Ausgang des Programmbausteins, in dem das Symbol verwendet wird, den Wert „true“ aufweist, wird Schritt 4 deaktiviert.<br/> <b>HINWEIS:</b> Der aktuelle Schritt wird in jedem Fall deaktiviert.</p> |

## Abschnitt 6.15

### Grafcet (SFC) Programmierung

---

#### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema   | Seite |
|---|-------|
| Einführung in die Grafcet (SFC)-Programmierung      | 231   |
| Verwenden des Grafischen Grafcet (SFC)-Editors      | 234   |
| Verzweigung   | 239   |
| Beste Verfahren beim Programmieren (Best Practices) | 243   |

## Einführung in die Grafcet (SFC)-Programmierung

### Einführung

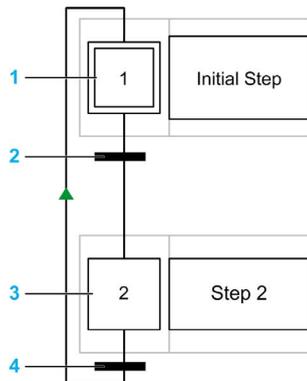
Grafcet (SFC) ist eine grafische Programmiersprache, die eine chronologische Abfolge der Ausführung digitaler Tasks, als *Schritte* bezeichnet, beschreibt. Die Reihenfolge der Ausführung der Schritte wird durch die *Übergänge* bestimmt, welche die Schritte verbinden.

### Elemente einer Grafcet (SFC)-POU

Eine Grafcet (SFC)-POU hat die folgenden Komponenten:

- **Schritt:** Ein Schritt führt eine Aktionsliste aus, die in einem oder mehreren Programmbausteinen definiert sind, welche in der Kontaktplan-/IL-Programmiersprache geschrieben sind. Typen von Schritten:
  - **Initialschritt:** Wird am Beginn eines Programms oder nach einem Neustart der Steuerung ausgeführt. Er wird durch eine Zelle mit Doppelrahmen dargestellt.
  - **Regulärer Schritt:** Schritte, die nach dem Abschluss des Initialschritts gemäß bestimmter Bedingungen ausgeführt werden.
- **Übergang:** Ein boolescher Ausdruck, der zwischen Schritten ausgewertet wird. Er entspricht der Verbindung zwischen zwei oder mehreren Schritten. Der boolesche Ausdruck ist in einem einfachen Übergangsprogrammbaustein definiert, welcher in der Kontaktplan-/IL-Programmiersprache geschrieben ist.

Die folgende Abbildung zeigt ein Beispiel einer Grafcet (SFC)-POU mit einem Initialschritt, einem regulären Schritt und zwei Übergängen:



- 1 Initialschritt
- 2 Übergang von Schritt 1 zu Schritt 2
- 3 Regulärer Schritt
- 4 Übergang von Schritt 2 zurück zu Schritt 1. Ein Pfeil auf der Verbindung zeigt an, dass die Reihenfolge der Schrittausführung nicht dem Standard von links nach rechts oder von oben nach unten folgt.

## Regeln für Grafcet (SFC)-POUs

Grafcet-POUs können nur in der Master-Task eines Programms erstellt werden.

Es können mehrfache Grafcet-POUs erstellt werden.

## Grafcet (SFC)-Verarbeitung

Die folgenden Regeln werden bei der Ausführung von Grafcet (SFC) auf die Steuerung angewandt:

- Der Master-Task-Zyklus startet.
- Die POUs, die dem ersten Grafcet (SFC)-Schritt vorangehen, werden sequentiell ausgeführt.
- Der erste Grafcet (SFC)-Schritt startet die **Grafcet-Überwachung**.
- Wenn die **Grafcet-Überwachung** endet, wird die erste POU nach dem letzten Grafcet (SFC)-Schritt abgerufen.

Verhalten der **Grafcet-Überwachung**:

1. Die Steuerung verarbeitet die zugehörigen Grafcet (SFC)-Systembits %S21, %S22 und %S23.
2. Die Steuerung aktualisiert die Aktivierungszustände jedes Grafcet (SFC)-Schritts.
  - Schritte, die für eine Deaktivierung gekennzeichnet sind, werden deaktiviert.
  - Schritte, die für eine Aktivierung gekennzeichnet sind, werden aktiviert.
  - Schritte, die gleichzeitig für eine Deaktivierung und Aktivierung gekennzeichnet sind, werden oder bleiben aktiviert.
  - Die Aktivierungs- und Deaktivierungslisten werden zurückgesetzt.
3. Die Steuerung tastet die Schritte ab (in einer Schleife von der niedrigsten definierten Schrittnummer zur höchsten definierten Schrittnummer). Wenn ein abgetasteter Schritt aktiviert wird, wird der entsprechende Schrittcode abgerufen.
4. Wenn ein Übergangscod ein Schritt aktiviert oder deaktiviert, dann wird diese Aktion entsprechend in der Aktivierungs- oder Deaktivierungsliste für den nächsten Task-Zyklus platziert.
5. Wenn der letzte aktive Schrittcode ausgeführt wurde, endet die **Grafcet-Überwachung**.

## Multi-Token-Verhalten

SoMachine Basic Grafcet-POU ist ein Multi-Token und erfüllt die Normen von IEC 61131-3 nicht.

Die Ausgangssituation wird durch die als Initialschritte definierten Schritte gesteuert.

In einer Grafcet-POU können mehrere Schritte gleichzeitig aktiviert sein.

Die Prozesse für aktive Signalzustände werden auf den Richtungsverbindungen ausgeführt und durch das Umschalten eines oder mehrerer Übergänge ausgelöst. Die Richtung des Vorgangs folgt den Richtungsverbindungen und verläuft von der Unterseite des Vorgängerschritts zur Oberseite des Folgeschritts.

Ein Übergang wird überprüft, wenn die ihm direkt vorhergehenden Schritte aktiv sind. Ein Übergang wird nicht überprüft, wenn die ihm direkt vorhergehenden Schritte nicht aktiv sind.

Ein Übergang wird ausgelöst, wenn die entsprechenden Übergangsbedingungen erfüllt sind.

Das Auslösen eines Übergangs kennzeichnet die direkt vorhergehenden Schritte, die mit dem Übergang verbunden sind, als deaktiviert, und die direkt folgenden Schritte als aktiviert.

Die tatsächliche Aktivierung oder Deaktivierung von Schritten wird am Anfang eines jeden Master-Task-Zyklus ausgeführt (siehe **Grafcet-Überwachung** (*siehe Seite 232*)).

Wenn mehr als eine Übergangsbedingung in einer Reihe an sequentiellen Schritten erfüllt wurde, wird pro Zyklus ein Schritt verarbeitet.

Wenn ein Schritt gleichzeitig aktiviert und deaktiviert wird, dann wird oder bleibt der Schritt aktiviert.

Es können mehr als eine Verzweigung mit Alternativverzweigungen aktiv sein.

Die Verzweigungen, die ausgeführt werden sollen, werden durch das Ergebnis der Übergangsbedingungen der Übergänge bestimmt, die der Alternativverzweigung folgen.

Verzweigungsübergänge werden parallel verarbeitet.

Verzweigungen mit erfüllten Übergängen werden ausgelöst.

In Schritt-Aktionen können Aufrufe von Unterprogrammen verwendet werden.

### Bits zur Steuerung von Grafcet (SFC)

| Steuerungsbit | Bezeichnung                          | Beschreibung   |
|---------------|--------------------------------------|--|
| %S21          | Initialisierung des Grafcet          | Wenn auf 1 eingestellt, werden die Initialschritte in Grafcet-POU überprüft.   |
| %S22          | Grafcet-Rücksetzung                  | Wenn auf 1 eingestellt, werden die Schritte deaktiviert und die Ausführung beginnt erneut.   |
| %S23          | Grafcet voreinstellen und einfrieren | Wenn auf 1 eingestellt, wird die Ausführung von Grafcet-POU gestoppt bis der Bit auf 0 eingestellt ist.  |
| %Xi           | Grafcet-Schritte                     | Bits %X1 bis %Xi werden Grafcet-Schritten zugeordnet. Das Schrittbit %Xi befindet sich in Zustand 1, wenn der entsprechende Schritt aktiviert ist, und in Zustand 0, wenn der Schritt deaktiviert ist. Der Bit kann nicht geschrieben werden, wenn Grafcet (SFC) verwendet wird. |

Siehe die Beschreibung der Systembits (*siehe Modicon M221, Logic Controller, Programmierhandbuch*) für weitere Informationen.

## Verwenden des Grafischen Grafcet (SFC)-Editors

### Überblick

Der Grafcet-Grafikeditor wird zum Programmieren in Grafcet (SFC) verwendet.

Um den Grafcet-Grafikeditor anzuzeigen, wählen Sie einen beliebiges  $n$  - **Grafcet**-Knoten in der Baumstrukturansicht.

Der Grafcet-Grafikeditor enthält ein Raster an Zellen. Jede Zelle enthält einen Schritt, einen Übergang oder beides.

Die Minimalgröße einer Grafcet-POU ist ein Schritt.

Die maximale Anzahl an Schritten für die Applikation ist 96.

### Trennen des Grafcet Grafischen Editors

Sie können das Grafcet-Grafikeditor-Fenster vom SoMachine Basic-Hauptfenster trennen, so dass es unabhängig verschoben und in der Größe verändert werden kann. Dadurch können Sie es beispielsweise auf einen separaten Monitor verschieben und die Grafcet POU's gleichzeitig mit den IL/Kontaktplan-POUs anzeigen.

Um das Fenster zu trennen, klicken Sie auf die -Schaltfläche in der oberen rechten Ecke des Grafcet-Grafikeditor-Fensters.

Ziehen Sie die Titelleiste des Fensters, um es zu verschieben. Schließen Sie das Fenster, um zur normalen Ansicht zurückzukehren.

### Einfügen von Schritten

Klicken Sie doppelt in eine beliebige Rasterzelle, um einen Schritt hinzuzufügen; oder klicken Sie mit der rechten Maustaste in ein beliebiges Rasterzelle und wählen Sie im erscheinenden Kontextmenü **Einen Schritt hinzufügen** aus.

Die **Anzahl der verwendeten Grafcet-Schritte** wird in der oberen rechten Ecke des Grafcet-Grafikeditor-Fensters angezeigt.

Sie können einen Schritt verschieben, indem Sie ihn mittels Drag-und-Drop (Ziehen und Fallenlassen) in einer anderen Zelle ablegen.

### Ändern eines Schrittyps (initial oder regulär)

Der erste in Grafcet-Grafikeditor erstellte Schritt ist standardmäßig der Initialschritt.

Eine Grafcet-POU muss mindestens einen Initialschritt enthalten. Es ist möglich, mehrere Schritte als Initialschritt zu definieren.

Um den Schrittyp zu ändern (initial/regulär), klicken Sie mit der rechten Maustaste auf den Schritt und wählen Sie **Als Initialschritt festlegen/aufheben** aus.

## Kopieren und Einfügen eines Schritts

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie mit der rechten Maustaste auf den Schritt, den Sie kopieren möchten, und wählen Sie im Kontextmenü die Option <b>Kopieren</b> aus.  |
| 2       | Klicken Sie mit der rechten Maustaste auf eine leere Zelle und wählen Sie <b>Einfügen</b> aus.<br><b>Ergebnis:</b> Eine Kopie des Schritts erscheint. Kopien der Kontaktpläne/IL-Bausteine, die zum Schritt gehören, werden unter dem entsprechenden Subknoten des <b>Schritts</b> in der Baumstruktur hinzugefügt. |

## Erstellen von Übergängen

Verbinden Sie Schritte, um die Reihenfolge der Ausführung der Schritte zu definieren.

Zum Erstellen eines Übergangs zwischen zwei Schritten:

| Schritt | Aktion  |
|---------|---|
| 1       | Bewegen Sie die Maus über das untere Ende eines Schritts.<br><b>Ergebnis:</b> Ein grüner Block erscheint. |
| 2       | Ziehen Sie die Maus zum Schritt, mit dem Sie ihn verbinden wollen.  |
| 3       | Lassen Sie die Maustaste los.<br><b>Ergebnis:</b> Eine Verbindung und ein Übergang erscheint.             |

## Bearbeiten von Bezeichnungen

Zum Bearbeiten der Standard-Bezeichnungen eines beliebigen Schritts oder Übergangs

| Schritt | Aktion   |
|---------|--|
| 1       | Doppelklicken Sie auf die Bezeichnung eines beliebigen Grafnet (SFC)-Schritts oder -Übergangs.   |
| 2       | Geben Sie einen neuen Namen für das Schritt- oder Übergangselement ein und drücken Sie die Eingabetaste (ENTER).<br>Beispielsweise können Sie die standardmäßige Bezeichnung <i>Schritt_1</i> zu <i>INIT</i> ändern. |

### Programmieren der Funktionsweise eines Schritts

Die Funktionsweise eines Schritts ist in einem oder mehreren IL/Ladder-Sprachbausteinen definiert

Zum Definieren der Funktionsweise eines Schritts:

| Schritt | Aktion  |
|---------|---|
| 1       | <p>Führen Sie eine der folgenden Aktionen aus:</p> <ul style="list-style-type: none"> <li>● Klicken Sie doppelt auf einen Schritt im Grafcet-Grafikeditor.</li> <li>● Wählen Sie einen <b>Schritt</b>-Knoten in der Baumstruktur aus, wobei <math>n</math> der Schrittnummer entspricht.</li> </ul> <p><b>Ergebnis:</b> Grafcet-Grafikeditor wird geschlossen.</p>        |
| 2       | <p>Klicken Sie mit der rechten Maustaste auf den <b>Schritt</b>-Knoten und wählen Sie im Kontextmenü die Option <b>Programmbaustein kopieren</b> aus.</p> <p><b>Ergebnis:</b> In der Baumstruktur erscheinen die Programmbausteine als Unterknoten unter dem <b>Schritt</b>-Knoten.</p>   |
| 3       | <p>Programmieren Sie den Programmbaustein in der Ladder- oder IL-Programmiersprache und erstellen Sie zusätzliche Programmbausteine, wenn erforderlich, wie in Programmiersprache Kontaktplan (<i>siehe Seite 185</i>) (Ladder) und Programmiersprache Anweisungsliste (<i>siehe Seite 209</i>) (IL) beschrieben.</p>   |
| 4       | <p>Um Grafcet-Grafikeditor wieder anzuzeigen, gehen Sie wie folgt vor:</p> <ul style="list-style-type: none"> <li>● Klicken Sie auf das Symbol .</li> <li>● Wählen Sie den <math>n</math>- Grafcet-POU Knoten aus, wobei <math>n</math> der Anzahl an Grafcet-POU entspricht.</li> </ul> |

### Programmieren der Funktionsweise eines Übergangs

Die Funktionsweise eines Übergangs ist in einem oder mehreren IL/Ladder-Sprachbausteinen für Übergänge definiert.

Zum Definieren der Funktionsweise eines Übergangsprogrammmbausteins:

| Schritt | Aktion  |
|---------|---|
| 1       | <p>Führen Sie eine der folgenden Aktionen aus:</p> <ul style="list-style-type: none"> <li>● Klicken Sie doppelt auf einen Übergang im Grafcet-Grafikeditor.</li> <li>● Wählen Sie in der Baumstruktur einen <b>Übergänge</b> → <b>Tr<math>n</math></b>-Knoten aus</li> </ul> <p><b>Ergebnis:</b> Der Grafcet-Grafikeditor wird geschlossen und ein Kontaktplan-Sprachbaustein wird angezeigt.</p> |

| Schritt | Aktion  |
|---------|---|
| 2       | <p>Programmieren Sie den Programmbaustein in der Ladder- oder IL-Programmiersprache wie in Programmiersprache Kontaktplan (<i>siehe Seite 185</i>) (Ladder) und Programmiersprache Anweisungsliste (<i>siehe Seite 209</i>) (IL) beschrieben.</p> <p>Funktionsbausteine können in Übergangsprogrammbausteinen verwendet werden, mit Ausnahme derer, die keine Ausgänge haben, wie beispielsweise Schiebebitregister, Schrittzähler.</p> <p>Wenn ein Funktionsbaustein verwendet wird, muss die Anweisung <code>END_BLK</code> der Anweisung <code>ENDT</code> unmittelbar folgen. Beispiel:</p> <pre> Tr1 Comment 0000   BLK  %TM2 0001   LD   0 0002   IN 0003   OUT_BLK 0004   LD   Q 0005   ENDT 0006   END_BLK                     </pre> <p><b>HINWEIS:</b> Der Programmbaustein endet mit einer <code>ENDT</code>-Anweisung (Ende des Übergangs). Die Anweisung kann nicht ausgewählt oder geändert werden und muss die letzte Anweisung im Programmbaustein sein. (Ausnahme: der Programmbaustein enthält ein FB).</p> |
| 3       | <p>Um Grafcet-Grafikeditor wieder anzuzeigen, gehen Sie wie folgt vor:</p> <p></p> <ul style="list-style-type: none"> <li>• Klicken Sie auf das Symbol .</li> <li>• Wählen Sie den <math>n</math>- Grafcet-POU Knoten aus, wobei <math>n</math> der Anzahl an Grafcet-POU entspricht.</li> </ul>  |

### Rückgängig/Wiederherstellen

Sie können die Schaltflächen **Rückgängig** oder **Wiederherstellen** in der Symbolleiste für maximal 10 gespeicherte Aktionen verwenden.

## Löschen eines Schritts oder Übergangs

| Schritt | Aktion   |
|---------|--|
| 1       | <p>Klicken Sie im Grafcet-Grafikeditor:</p> <ul style="list-style-type: none"><li>• Wählen Sie einen Schritt oder Übergang und drücken Sie die <b>ENTF</b>-Taste (Entfernen).</li><li>• Klicken Sie mit der rechten Maustaste auf den Schritt oder Übergang und wählen Sie im Kontextmenü <b>Ausgewählte Elemente löschen</b> aus.</li></ul> <p><b>Ergebnis:</b> Der ausgewählte Schritt oder Übergang wird gelöscht.</p> <p><b>HINWEIS:</b> Sie können in der Ansicht Baumstruktur keine Schritte oder Übergänge löschen.</p> |

## Verzweigung

### Einführung

Eine Grafcet (SFC)-POU kann Verzweigungen enthalten.

Es gibt zwei Arten an Verzweigungen:

- Parallelverzweigung: zwei oder mehr Schritte werden gleichzeitig verarbeitet, wenn der vorhergehende Übergang TRUE ist.
- Alternativverzweigung: einer oder mehrere alternative Schritte werden abhängig vom Ergebnis der Evaluation der vorhergehenden Übergangsbedingungen (Multi-Token-Verhalten) verarbeitet.

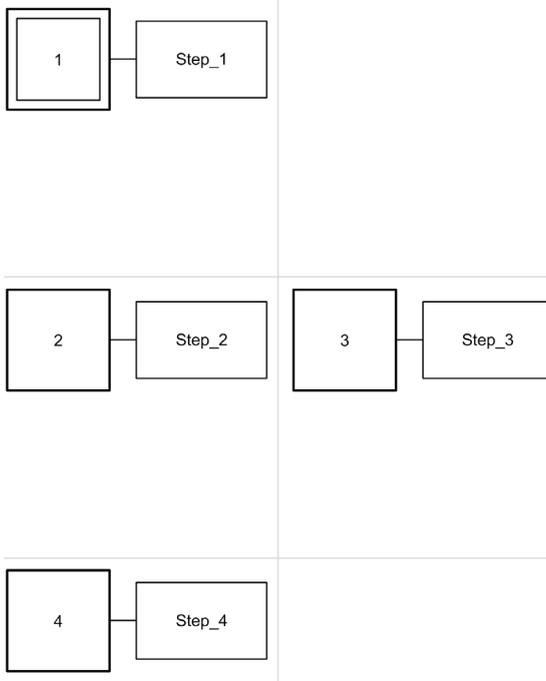
### Parallelverzweigung

Eine Parallelverzweigung ermöglicht den Übergang von einem einfachen Schritt zu mehrfachen Schritten.

Eine Parallelverzweigung muss zwischen zwei Schritten liegen.

Parallelverzweigungen können Alternativ- oder andere Parallelverzweigungen beinhalten.

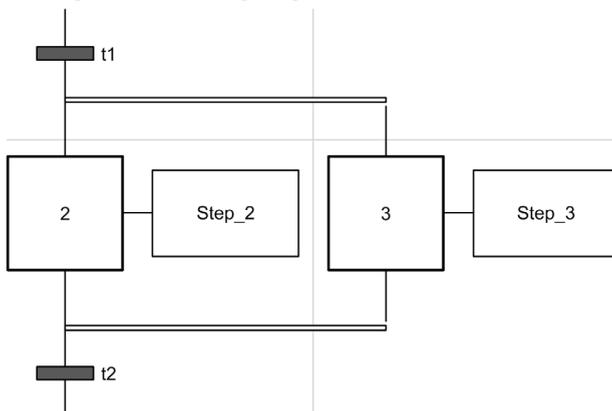
Die folgende Abbildung zeigt eine Beispiel-Grafcet-POU mit 4 Schritten vor der Erstellung einer Parallelverzweigung:



Um eine Parallelverzweigung für die Schritte 2 und 3 zu erstellen:

| Schritt | Aktion   |
|---------|--|
| 1       | Erstellen Sie eine Verbindung zwischen Schritt 1 und Schritt 2: bewegen Sie die Maus zum unteren Ende von Schritt 1, ziehen Sie sie zu Schritt 2 und lassen Sie die Maustaste los. Eine neue Verbindung und ein neuer Übergang erscheint.  |
| 2       | Zeichnen Sie eine Verbindung zwischen Schritt 3 und dem Übergang: bewegen Sie die Maus zum oberen Ende von Schritt 3, ziehen Sie sie zum Übergang und lassen Sie die Maustaste los.<br><b>Ergebnis:</b> eine horizontale Doppellinie erscheint unter dem bestehenden Übergang (siehe folgende Abbildung).<br><b>HINWEIS:</b> Um eine Verbindung zwischen einem Übergang und einem Schritt zu erstellen, der höher in der POU liegt, müssen Sie die Verbindung vom Schritt aus zum Übergang zeichnen. |
| 3       | Um die Verzweigung wieder mit dem Hauptverarbeitungszweig zu verbinden, erstellen Sie einen Übergang zwischen Schritt 2 und Schritt 4.   |
| 4       | Zeichnen Sie eine Verbindung zwischen Schritt 3 und dem neuen Übergang: bewegen Sie die Maus zum oberen Ende von Schritt 3, ziehen Sie sie zum Übergang und lassen Sie die Maustaste los.<br><b>Ergebnis:</b> eine horizontale Doppellinie erscheint unter dem bestehenden Übergang (siehe folgende Abbildung).  |

Die folgende Abbildung zeigt eine Grafcet-POU nach der Erstellung einer Parallelverzweigung:



Die horizontalen Linien vor und nach der Verzweigung sind doppelte Linien.

## Alternativverzweigung

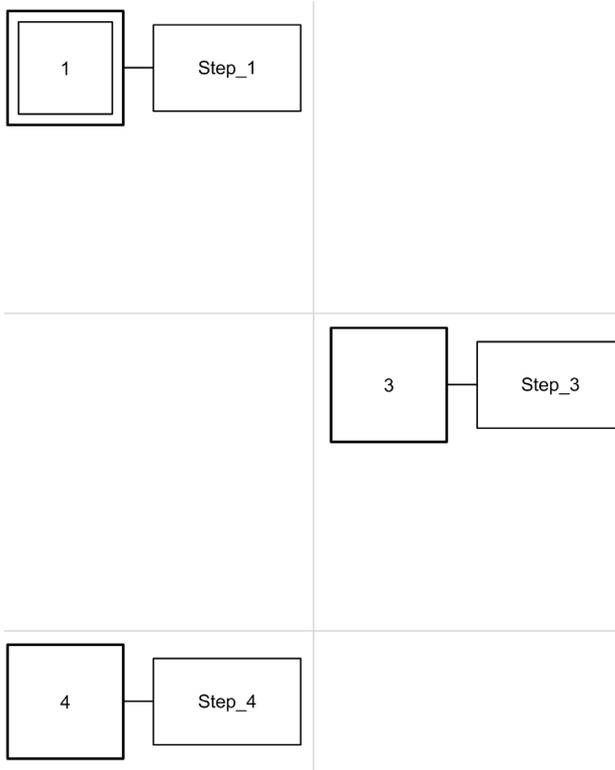
Eine alternative Verzweigung muss mit einem Übergang beginnen und enden.

Alternativverzweigungen können Parallelverzweigungen oder andere alternative Verzweigungen beinhalten.

Bei Multi-Token-Verhalten kann mehr als eine parallele Umschaltung von den Übergängen erstellt werden. Die Verzweigungen, die ausgeführt werden sollen, werden durch das Ergebnis der Übergangsbedingungen der Übergänge bestimmt, die der Alternativverzweigung folgen. Alle Übergänge der Verzweigungen werden verarbeitet. Verzweigungen mit erfüllten Übergängen werden ausgelöst.

Wenn Alternativverzweigungen exklusiv umgeschaltet werden müssen (Mono-Token-Verhalten), muss dies explizit im Übergangscod definiert sein.

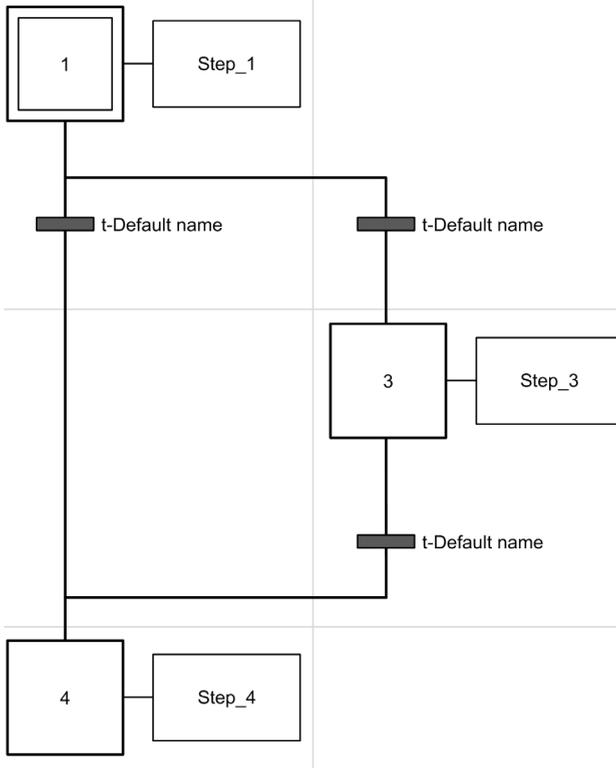
Die folgende Abbildung zeigt eine Beispiel-Grafcet-POU mit 3 Schritten vor der Erstellung einer Alternativverzweigung für Schritt 3 und Schritt 4:



Zur Erstellung einer Alternativverzweigung:

| Schritt | Aktion   |
|---------|--|
| 1       | Erstellen Sie einen Übergang zwischen Schritt 1 und Schritt 4.<br><b>Ergebnis:</b> Eine neue Verbindung und ein neuer Übergang erscheint.  |
| 2       | Zeichnen Sie eine Verbindung zwischen Schritt 1 und Schritt 3: bewegen Sie die Maus zum unteren Ende von Schritt 1, ziehen Sie sie zu Schritt 3 und lassen Sie die Maustaste los.<br><b>Ergebnis:</b> Eine neue Verbindung und ein neuer Übergang erscheint mit einer Verzweigung oberhalb des bestehenden Übergangs (siehe folgende Abbildung). |
| 3       | Erstellen Sie einen Übergang zwischen Schritt 3 und Schritt 4.<br><b>Ergebnis:</b> Eine neue Verbindung und ein neuer Übergang erscheint mit einer Verzweigung unterhalb des bestehenden Übergangs zwischen Schritt 1 und Schritt 4 (siehe folgende Abbildung).  |

Die folgende Abbildung zeigt eine Grafcet-POU nach der Erstellung einer Alternativverzweigung:

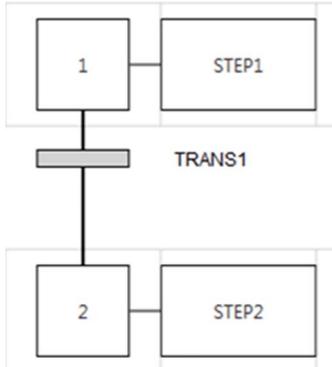


Die horizontalen Linien vor und nach der Verzweigung sind einfache Linien.

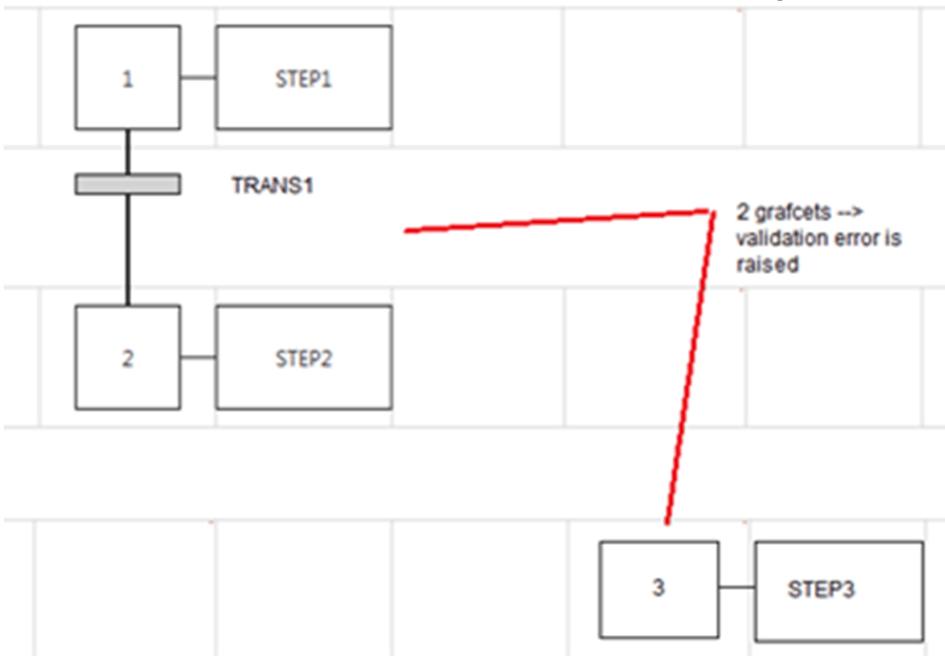
## Beste Verfahren beim Programmieren (Best Practices)

### Grafcet (SFC) Regeln

- Schritte müssen durch einen Übergang verbunden sein:



- Sie können nur eine Grafcet-POU im selben Grafcet-Grafikeditor hinzufügen:



## Überkreuzverbindungen

Sie können aus den folgenden Gründen Überkreuzverbindungen haben:

- Alternative (logisch ODER) Verzweigungen (Gabel oder Kreuzung)
- zum Platzsparen in der Gitterzelle. Wenn Leitungen sich kreuzen, besteht keine Interaktion zwischen den Leitungen. Es dient lediglich der symbolischen Darstellung.

---

## Abschnitt 6.16

### Fehlerbehebung im Online-Modus

---

#### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                      | Seite |
|----------------------------|-------|
| Trace-Fenster              | 246   |
| Ändern von Werten          | 249   |
| Forcieren von Werten       | 250   |
| Änderungen im Online-Modus | 251   |

## Trace-Fenster

### Überblick

Das **Trace**-Fenster ermöglicht Ihnen die Anzeige der Werte bestimmter analoger und/oder digitaler Variablen in Grafikform (die Aufzeichnung ist auf 12 fortlaufende Stunden begrenzt). Jede Animationstabelle kann jeweils nur eine (1) Traceaufzeichnung gleichzeitig enthalten. Es können einer Traceaufzeichnung bis zu 8 Objekte hinzugefügt werden. Sie können die Daten für weitere Analysen in eine Datei exportieren.

**HINWEIS:** Die minimal konfigurierbare Aktualisierungszeit für den Trace ist 1 Sekunde (*siehe Seite 157*). Aus diesem Grund können Änderungen in den Werten boolescher Variablen zwischen den Mastertask-Zyklen nicht nachverfolgt (traced) werden.

Beschreibung des Trace-Fensters

## Trace-Funktion [-] [X]

---

Zeitbasis 5 s

Startzeit des Trace-Vorgangs: 00:00:00

Verstrichene Zeit  SPS-Zeit

Aufzeichnungsdauer: 00:00:41

**Numerisches Tracing**

— %C8.V

— %IW0.1

Autom. Werteskalierung

Maximum

Minimum

**Logisches Tracing**

— %C8.D

— %C8.E

— %M0

— %I0.2

— %M1

Wählen Sie **Verstrichene Zeit** aus, um die Startzeit des Tracing-Vorgangs auf 00:00:00 einzustellen, bzw. **SPS-Zeit**, wenn Sie Uhrzeit und Datum der Steuerung als Startzeit für das Tracing verwenden möchten.

Im Trace-Fenster werden separate Grafiken für jeden in der Animationstabelle zum Tracing ausgewählten Datentyp angezeigt:

- Ganzzahl- und Realzahlwerte werden im Bereich **Numerisches Tracing** dargestellt.  
Für alle numerischen Werte gilt dieselbe Skala in der Grafik.  
Wählen Sie **Autom. Werteskalierung** aus, um die vertikale Achse automatisch anzupassen, sodass alle Werte angezeigt werden. Sie können auch die Werte **Maximum** und **Minimum** eingeben, wenn ein bestimmter Wertebereich angezeigt werden soll.  
**HINWEIS:** Für **Maximum** und **Minimum** können Ganzzahl- oder Realzahlwerte eingegeben werden.
- Im Bereich **Logisches Tracing** werden Binärwerte angezeigt.  
Jeder Binärwert wird auf einer individuellen Skala von 0 und 1 dargestellt.

### Starten, Pausieren und Zurücksetzen des Tracing-Vorgangs

Klicken Sie auf **Start**, um das Tracing der Variablen zu starten.

Klicken Sie auf **Stopp**, um das Echtzeit-Tracing vorübergehend anzuhalten.

Klicken Sie auf **Reset**, um alle bisherigen Tracing-Daten aus den Grafiken zu entfernen und den Wert **Aufzeichnungsdauer** auf 0 zurückzusetzen.

### Exportieren der Tracing-Daten

Klicken Sie auf **Exportieren**, um alle Tracing-Daten in eine Datei auf dem PC zu exportieren.

Die Daten werden im Format CSV (Comma-Separated Value) gespeichert.

---

## Ändern von Werten

### Einführung

Im Online-Modus ermöglicht Ihnen SoMachine Basic die Änderung der Werte bestimmter Objekttypen.

Online-Aktualisierung ist nur möglich, wenn für das betreffende Objekt ein Lese-/Schreibzugriff definiert wurde. Beispiel:

- Der Wert eines Analogeingangs kann nicht geändert werden.
- Der Wert des Parameters `Der Preset-Parameter (%TM0.P-Objekt)` eines `Timer-Funktionsbausteins` kann aktualisiert werden.

Informationen zu Objekten mit Lese-/Schreibzugriff finden Sie in der Objektbeschreibung im *SoMachine Basic – Generische Funktionen – Bibliothekshandbuch* oder im *Programmierhandbuch* für Ihre Hardwareplattform.

Um den Wert eines Objekts zu ändern, fügen Sie das Objekt in einer Animationstabelle (*siehe Seite 159*) hinzu und definieren Sie nach Bedarf dessen Eigenschaften.

## Forcieren von Werten

### Übersicht

Im Online-Modus können Sie die Werte bestimmter boolescher Objekttypen auf „False“ (0) oder „True“ (1) forcieren. Dadurch können Sie Adressen auf bestimmte Werte einstellen und eine Änderung des Werts durch die Programmlogik oder ein externes System unterbinden. Diese Funktion wird für das Debugging und die Feineinstellung von Programmen verwendet.

Um die Werte von booleschen Objekten im Online-Modus zu forcieren haben Sie folgende Möglichkeiten:

- Verwenden einer Animationstabelle (*siehe Seite 157*)
- Ändern von booleschen Objektwerten (*siehe Seite 252*) direkt im Kontaktplan-Editor (LD - Ladder).

Digitaleingänge und -ausgänge können nicht forciert werden, wenn:

- ein Eingang als Run/Stop-Eingang (starten/anhalten) verwendet wird
- sie als Schnellzähler-Eingänge (Fast Counter - FC) konfiguriert sind
- sie als Hochgeschwindigkeitszähler-Eingänge (High Speed Counter - HSC) konfiguriert sind
- sie als Reflexausgänge konfiguriert sind

**HINWEIS:** Die Forcierung wird am Ende des Abfragezyklus durchgeführt. Die Abbildtabelle der Ausgänge kann jedoch aufgrund der Logik Ihres Programms geändert werden und kann sich, entgegen dem von Ihnen ausgewählten geforcten Zustand, in Animationstabellen und anderen Datenanzeigen befinden. Am Ende der Abfrage wird dies durch Ausführen des angeforderten geforcten Zustands korrigiert und der physikalische Ausgang wird diesen geforcten Zustand wiedergeben.

## Änderungen im Online-Modus

### Überblick

Es ist wie folgt möglich, das Programm im Online-Modus zu ändern:

- Programmausteine hinzufügen (*siehe Seite 251*)
- Programmausteine ändern (*siehe Seite 251*)
- Ändern von booleschen Werten in Ladder (Kontaktplan/LD) (*siehe Seite 252*)
- Ändern der Parameter von Funktionsbausteinen (*siehe Seite 254*)
- Ändern von Konstanten im Wortformat (*siehe Seite 255*)
- Ändern von Werten in Operations- und Vergleichsbausteinen (*siehe Seite 255*)
- Löschen von Programmausteinen (*siehe Seite 256*)
- Senden von Änderungen (*siehe Seite 256*)

Alle vorgenommenen Änderungen müssen dann an die Logiksteuerung gesendet (*siehe Seite 256*) werden.

### Hinzufügen von Programmausteinen

Sie können im Online-Modus neue Programmausteine hinzufügen (*siehe Seite 103*).

**HINWEIS:** Die Anwendung muss mit einer Funktionsebene (*siehe Seite 94*) von mindestens **Ebene 4.1** konfiguriert werden, damit neue Programmausteine im Online-Modus hinzugefügt werden können.

Es bestehen die folgenden Einschränkungen bis der neue Programmaustein erfolgreich an die Logiksteuerung gesendet wurde:

- Programmausteine, die Fehler (  ) enthalten, können nicht an die Logiksteuerung gesendet werden.
- Programmausteine müssen in der Ladder-Sprache (Kontaktplan) geschrieben sein und können vor erfolgreicher Kompilation nicht zu IL konvertiert werden.
- Programmausteine können keine Grafcet-Schritte (Liste) enthalten.
- Programmausteinen können nicht mit Bezeichnungen (Labels) versehen werden.

### Ändern von Programmausteinen

Sie können Ihre Programmausteine in den Editoren für Instruction List (IL) und Ladder (LD) ändern, auch während Sie sich im Online-Modus befinden. Grafcet (SFC) ist online jedoch nicht verfügbar. Geänderte Programmausteine erscheinen mit einem orangefarbenen Hintergrund (*siehe Seite 190*).

Es gibt jedoch gewisse Beschränkungen hinsichtlich der Art der möglichen Bearbeitungen und den Anweisungen, die Sie bearbeiten können; abhängig davon ab, ob sich die Logiksteuerung im Modus `RUNNING` oder `STOPPED` befindet. Diese Beschränkungen sorgen dafür, dass die Steuerung in aktuellen Status verbleiben kann und dass die Integrität des Programms geschützt bleibt.

Sie können die Anzeige eines Programmbausteins auch im Online-Modus zwischen Anweisungsliste (AWL - Instruction List/IL) und Kontaktplan (Ladder/LD) umschalten.

In der folgenden Tabelle wird angegeben, in welchen Fällen Bearbeitungen erlaubt sind:

| Funktionsweise  | Im Modus STOPPED in IL | Im Modus RUNNING in IL | Im Modus STOPPED in Ladder | Im Modus RUNNING in Ladder |
|---|------------------------|------------------------|----------------------------|----------------------------|
| Ereignis: Taskinhalt  | Bearbeitbar            | Unzulässig             | Bearbeitbar                | Nicht bearbeitbar          |
| Master/Periodisch: Taskinhalt   | Bearbeitbar            | Bearbeitbar            | Bearbeitbar                | Bearbeitbar                |
| Freier POU-Inhalt   | Bearbeitbar            | Bearbeitbar            | Nicht bearbeitbar          | Nicht bearbeitbar          |
| Programmbaustein mit Marke  | Bearbeitbar            | Unzulässig             | Nicht bearbeitbar          | Nicht bearbeitbar          |
| Programmbaustein mit Ende, Sprung oder Aufruf einer Subroutine oder Marke | Nicht bearbeitbar      | Nicht bearbeitbar      | Nicht bearbeitbar          | Nicht bearbeitbar          |
| Programm mit beliebiger Grafset-Anweisung                                 | Nicht bearbeitbar      | Nicht bearbeitbar      | Nicht bearbeitbar          | Nicht bearbeitbar          |
| Marke hinzufügen/bearbeiten   | Nicht bearbeitbar      | Nicht bearbeitbar      | Nicht bearbeitbar          | Nicht bearbeitbar          |
| Mehrere Operanden (Operations- und Vergleichsbausteine)                   | Nicht bearbeitbar      | Nicht bearbeitbar      | Bearbeitbar                | Bearbeitbar                |

**HINWEIS:** Diese Tabelle berücksichtigt Änderungen an der Programmstruktur nicht, die im Online-Modus nicht zugelassen sind.

### Ändern von booleschen Werten in Ladder (Kontaktplan/LD)

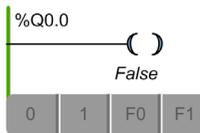
Zur Anzeige von Programmbausteinen in der Ladder-Sprache (Kontaktplan), können die werte bestimmter boolescher Objekttypen zu 1/0 geschrieben, zu 1/0 forciert oder nicht forciert werden.

Die folgenden booleschen Objekttypen können modifiziert werden:

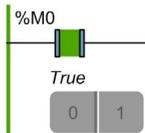
| Objekttyp                        | Schreiben von 1/0 | Zu 1/0 forcieren oder nicht forcieren |
|----------------------------------|-------------------|---------------------------------------|
| Digitaleingang (%Ix.y)           | Nicht zutreffend  | Ja                                    |
| Digitalausgang (%Qx.y)           | Ja                | Ja                                    |
| System Bit (%Si ) <sup>1</sup>   | Ja                | Nicht zutreffend                      |
| Speicher-Bit (%Mi)               | Ja                | Nicht zutreffend                      |
| Speicher-Bit (%MWi :Xj )         | Ja                | Nicht zutreffend                      |
| Analoges Ausgangsbit (%QWi :Xj ) | Ja                | Nicht zutreffend                      |

| Objekttyp   | Schreiben von 1/0 | Zu 1/0 forcieren oder nicht forcieren |
|---|-------------------|---------------------------------------|
| Speicher-Bit (%SWi :Xj) <sup>1</sup>  | Ja                | Nicht zutreffend                      |
| Eingangsbaugruppenbit (%QWEi :Xj)   | Ja                | Nicht zutreffend                      |
| <sup>1</sup> Wenn das Systembit oder Systemwort mit einem Benutzerprogramm geschrieben werden kann. |                   |                                       |

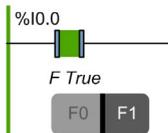
Bewegen Sie den Mauszeiger über das Objekt im Ladder-Editor. Wenn das Objekt zu 1/0 geschrieben und forciert werden kann, erscheinen die folgenden Schaltflächen unter dem grafischen Element:



Wenn das Objekt zu 1/0 geschrieben aber nicht forciert werden kann, erscheinen die folgenden Schaltflächen:



Wenn das Objekt auf 1/0 forciert aber nicht geschrieben werden kann, erscheinen die folgenden Schaltflächen:



Klicken Sie eine Schaltfläche, um den Echtzeitwert des Objekts zu ändern:

- **0.** Schreiben von 0/0.
- **1.** Schreiben von 1.
- **F0.** Auf 0 forcieren.
- **F1.** Auf 1 forcieren.

Die Schaltfläche, die dem aktuellen Status des Objekts entspricht wird in dunklem Grau dargestellt (im obigen Beispiel: **F1**).

Zum Entfernen der Forcierung gehen Sie wie folgt vor:

- Klicken Sie nochmals auf die Schaltfläche **F0/F1**.
- Verwenden Sie eine Animationstabelle (*siehe Seite 157*).

**HINWEIS:** Die Forcierung wird am Ende des Abfragezyklus durchgeführt. Die Abbildtabelle der Ausgänge kann jedoch aufgrund der Logik Ihres Programms geändert werden und kann sich, entgegen dem von Ihnen ausgewählten geforcten Zustand, in Animationstabellen und anderen Datenanzeigen befinden. Am Ende der Abfrage wird dies durch Ausführen des angeforderten geforcten Zustands korrigiert und der physikalische Ausgang wird diesen geforcten Zustand wiedergeben.

### Ändern der Parameterwerte des Funktionsbausteins

Zum Ändern eines Funktionsbausteinparameters im Online-Modus:

| Schritt | Aktion   |
|---------|--|
| 1       | <p>Bewegen Sie den Mauszeiger im Fenster <b>Programmierung</b> über den Funktionsbaustein im Kontaktplan-Editor.</p> <p><b>Ergebnis:</b> Die Quickinfo <b>Konfiguration</b> wird angezeigt.<br/>Die folgende Abbildung zeigt ein Beispiel der Quickinfo <b>Konfiguration</b>:</p>  |
| 2       | Klicken Sie zum Ändern auf den Wert.   |
| 3       | Geben Sie den Wert ein.  |
| 4       | <p>Um die Änderungen zu bestätigen, stehen folgende Methoden zur Auswahl:</p> <ul style="list-style-type: none"> <li>• Klicken Sie auf <b>Übernehmen</b>.</li> <li>• Klicken Sie in den Bereich außerhalb der Quickinfo <b>Konfiguration</b>. <b>Ergebnis:</b> Das Fenster <b>Frage</b> wird angezeigt. Klicken Sie auf <b>OK</b>.</li> </ul>                        |

## Ändern konstanter Doppelwörter

Konfigurationswerte und Laufzeitdatenwerte von konstanten Wort- (%KW), konstanten Doppelwort- (%KD) und konstanten Gleitkommawert- (%KF) Objekten können im Online-Modus geändert werden. Im Raster ‚Eigenschaften‘ können die Spalten **Dezimal**, **Binär**, **Hexadezimal** und **ASCII** bearbeitet werden:

| Eigenschaften der Konstantwörter    |                          | %KW     | %KD    | %KF     |                    |             |                 |  |
|-------------------------------------|--------------------------|---------|--------|---------|--------------------|-------------|-----------------|--|
| Verwendet                           | Gleichung verwendet      | Adresse | Symbol | Dezimal | Binär              | Hexadezimal | ASCII           |  |
| <input type="checkbox"/>            | <input type="checkbox"/> | %KW0    |        | 0       | 2#0000000000000000 | 16#0000     | keine Bedeutung |  |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | %KW1    |        | 0       | 2#0000000000000000 | 16#0000     | keine Bedeutung |  |

Zum Ändern eines konstanten Wort- oder Gleitkommawerts im Online-Modus:

| Schritt | Aktion  |
|---------|---|
| 1       | Wählen Sie auf der Registerkarte <b>Tools</b> des Fensters <b>Programmierung</b> die Optionen <b>Speicherobjekte → Konstanten im Wortformat</b> aus.                    |
| 2       | Klicken Sie auf <b>%KW</b> , <b>%KD</b> oder <b>%KF</b> , um die Art der Konstante auszuwählen, die Sie ändern wollen.  |
| 3       | Ändern Sie den Wert wie erforderlich.<br>Die Konstantwerte können importiert werden. Siehe Importieren der Eigenschaften der Konstantwörter ( <i>siehe Seite 166</i> ). |
| 4       | Klicken Sie auf <b>Übernehmen</b> .<br><b>Ergebnis:</b> Der geänderte Wert wird zur Logiksteuerung gesendet.  |

## Ändern von Werten in Operations- und Vergleichsbausteinen

| Schritt | Aktion  |
|---------|---|
| 1       | Bewegen Sie den Mauszeiger im Fenster <b>Programmierung</b> über einen Operations- oder Vergleichsbaustein im Kontaktplan-Editor.<br><b>Ergebnis:</b> Die Quickinfo <b>Online-Änderung</b> wird angezeigt:<br> |
| 2       | Klicken Sie zum Ändern auf das Objekt oder das Symbol.  |
| 3       | Geben Sie den Wert ein.   |

| Schritt | Aktion  |
|---------|---|
| 4       | <p>Zum Bestätigen stehen folgende Methoden zur Auswahl:</p> <ul style="list-style-type: none"> <li>● Klicken Sie auf die Schaltfläche .</li> <li>● Drücken Sie die <b>Eingabetaste</b>.</li> </ul> <p>Wenn der Wert ungültig ist, bleibt er unverändert.</p> |

### Löschen von Programmbausteinen

Sie können im Online-Modus Programmbausteine aus Ihrem Programm löschen.

**HINWEIS:** Die Anwendung muss mit einer Funktionsebene (*siehe Seite 94*) von mindestens **Ebene 4.1** konfiguriert werden, damit Programmbausteine im Online-Modus gelöscht werden können.

Dabei gelten die folgenden Einschränkungen:

- Der Programmbaustein muss in der Kontaktplan-Sprache (Ladder) angezeigt werden.
- Der Programmbaustein kann nicht der einzige Programmbaustein in einer POU oder Freien POU sein. Diese Einschränkung gilt nicht für Grafcet POU's.
- Der Programmbaustein darf keine Grafcet-Schritte (Liste) enthalten, er darf kein Programmbaustein eines Unterprogramms sein oder eine der folgenden Anweisungen enthalten:
  - JMP
  - JMPC
  - JMPCN
  - END
  - ENDC
  - ENDCN
  - G7
- Es kann nur ein Programmbaustein gleichzeitig gelöscht werden.

### Senden von Änderungen

In IL werden die Bearbeitungen, falls erlaubt, automatisch an die Logiksteuerung gesendet, nachdem die bearbeitete IL-Zeile geprüft wurde. Wenn die Bearbeitung nicht erlaubt ist, wird eine Meldung angezeigt.

Im Kontaktplan (Ladder) werden die Bearbeitungen nicht automatisch gesendet. Im Online-Modus erscheint eine Leiste mit Schaltflächen:



Klicken Sie auf **Senden**, um die Änderungen an die Logiksteuerung zu senden. Diese Schaltfläche ist nur dann aktiv, wenn das Programm im Online-Modus geändert wurde und keine Fehler enthält.

Klicken Sie auf **Rollback**, um die im Online-Modus gemachten Änderungen zu verwerfen und den Original-Programmbaustein wiederherzustellen (also die Version, die in der Logiksteuerung gespeichert ist). Die Hintergrundfarbe des Programmbausteins ändert sich von orangefarben auf grün. Diese Schaltfläche ist nur dann aktiv, wenn das Programm im Online-Modus geändert wurde.

Klicken Sie auf **Herunterladen nicht-ausführbarer Daten**, um alle Aktualisierungen nicht-ausführbarer Daten (Projekteigenschaften, Symbole, Kommentare, Animationstabellen usw.) auf die Logiksteuerung zu laden. Diese Schaltfläche ist nur dann aktiv, wenn die nicht-ausführbaren Daten zwischen dem PC und der Logiksteuerung nicht synchronisiert sind; beispielsweise, wenn eine Animationstabelle geändert wurde bevor der Online-Modus gestartet wurde.

Klicken Sie auf **Sicherung**, um die Inhalte des Flash-Speichers mit denen des RAM-Speichers auf der Logiksteuerung zu synchronisieren. Dieser Status wird im Fenster Steuerungsinformationen (*siehe Seite 280*) angezeigt. Während einer Sicherung werden die laufenden Ethernet-Kommunikationen (beispielsweise die Verwendung von Modbus TCP oder der EXCH3-Anweisung) zeitweise ausgesetzt.

**HINWEIS:** Stellen Sie sicher, dass alle Online-Änderungen auf Flash-Speicher gespeichert wurden, bevor Sie einen Klon erstellen.

Geänderte Programmbausteine werden auf Ihre Validität überprüft, abhängig davon, ob die Steuerung im **RUNNING**- oder **STOPPED**-Zustand ist. Änderungen, die Laufzeitfehler verursachen oder die Struktur des Programmspeichers ändern würden, werden im Online-Modus abgelehnt.



---

# Kapitel 7

## Inbetriebnahme

---

### Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Abschnitte:

| Abschnitt | Thema   | Seite |
|-----------|---|-------|
| 7.1       | Übersicht über das Fenster „Inbetriebnahme“         | 260   |
| 7.2       | <b>Aufbauen einer Verbindung zu einer Steuerung</b> | 261   |
| 7.3       | <b>Steuerungs-Update</b>                            | 272   |
| 7.4       | Speicherverwaltung                                  | 274   |
| 7.5       | <b>SPS-Infos</b>                                    | 280   |
| 7.6       | <b>RTC-Verwaltung</b>                               | 283   |

# Abschnitt 7.1

## Übersicht über das Fenster „Inbetriebnahme“

### Überblick des Fensters ‚Inbetriebnahme‘

#### Einführung

Im Fenster **Inbetriebnahme** können Sie folgende Aufgaben ausführen:

- Sich An- bzw. Abmelden bei einem Logic Controller
- Aktualisieren Sie die Firmware der Logiksteuerung (oder setzen Sie die Version herab).
- Verwalten des Speichers der Logiksteuerung (z. B. zur Durchführung von Sicherungs- und Wiederherstellungsvorgängen)
- Anzeigen von Informationen über die Logiksteuerung, das Erweiterungsmodul (Referenzen und im Falle der TM3-Erweiterungsmodule die Firmware-Versionen) und die Steckmodule, mit denen Sie verbunden sind.
- Verwalten der Echtzeituhr (RTC) der Logiksteuerungen.

| Gefunden: Neues Projekt |            |          |
|-------------------------|------------|----------|
|                         | Referenz   | Firmware |
| Controller              | TM221CE40U | 0.5.0.7  |
| Modul 1                 | TM3AI4-    | 25       |
| Modul 2                 | TM3AQ4-    | 20       |
| Steckmodul 1            | TMC2AI2    |          |
| Steckmodul 2            | TMC2AQ2V   |          |

✔ Die Anwendungen von PC und Steuerung sind identisch.  
Verbindung hergestellt

PC zu Steuerung (Download)

Steuerung zu PC (Upload)

Steuerung stoppen

Steuerung starten

Simulator starten

Simulator stoppen

**HINWEIS:** Die Anwendung muss mit einer Funktionsebene (*siehe Seite 94*) von mindestens Ebene 5.0 konfiguriert werden, damit die Firmware-Version der analogen TM3-Erweiterungsmodule angezeigt werden kann.

---

## Abschnitt 7.2

### Aufbauen einer Verbindung zu einer Steuerung

---

#### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

| Thema                                   | Seite |
|---|-------|
| Verbinden mit einer Logiksteuerung      | 262   |
| Herunter- und Hochladen von Anwendungen | 268   |

## Verbinden mit einer Logiksteuerung

### Überblick

Klicken Sie im Fenster **Inbetriebnahme** auf **Verbinden**, um die Verbindung zur Logiksteuerung zu verwalten.

### Verbundene Steuerungen

Es werden zwei Listen mit Steuerungen angezeigt:

#### 1. Lokale Geräte

Zeigt alle mit dem PC verbundenen logischen Steuerungen an:

- über die physischen COM-Ports des PCs (beispielsweise COM1)
- über USB-Kabel
- über die virtualisierten COM-Ports (über USB-zu-Seriell-Konverter oder Bluetooth-Dongles)
- über eine manuell ausgewählte Modemverbindung Verwenden Sie nur für Überwachungszwecke eine Modemverbindung zwischen SoMachine Basic und der Logiksteuerung.

**HINWEIS:** Wenn ein COM-Port ausgewählt ist und das Kontrollkästchen **Modbus-Treiberparameter beibehalten** aktiviert ist, wird die Kommunikation mit den im Modbus-Treiber definierten Parameter hergestellt.

#### 2. Ethernet-Geräte

In dieser Liste werden alle Steuerungen aufgeführt, die über Ethernet zugänglich sind (im gleichen Teilnetzwerk und nicht über einen Router oder ein anderes Gerät, das UDP-Broadcast-Sendungen blockiert). Die Liste enthält Steuerungen, die automatisch von SoMachine Basic erkannt werden, sowie sämtliche Steuerungen, die Sie manuell hinzufügen.

### Manuelles Hinzufügen von Ethernet-Steuerungen

Gehen Sie zum Hinzufügen einer Steuerung in der Liste **Ethernet-Geräte** vor wie folgt:

| Schritt | Aktion   |
|---------|--|
| 1       | Geben Sie im Feld <b>Dezentraler Lookup</b> die IP-Adresse der hinzuzufügenden Steuerung ein, z. B. 12.123.134.21. |
| 2       | Klicken Sie auf <b>Hinzufügen</b> ; um das Gerät in der Liste <b>Ethernet-Geräte</b> hinzuzufügen.                 |

### Manuelles Hinzufügen von Modemverbindungen

Voraussetzungen für die Verfügbarkeit eines Modems:

- Wenn auf dem PC kein Modem installiert ist, ist die Schaltfläche deaktiviert.
- Stellen Sie in der Option **Telefon und Modem** in der Windows **Systemsteuerung** sicher, dass das Modem installiert ist. Führen Sie einen Test durch. Klicken Sie dafür auf der Registerkarte **Modem** auf das zu testende Modem, und klicken Sie auf **Eigenschaften** → **Diagnose** → **Modem abfragen**.) Die Reaktion des Modems muss gültig sein.

- Wenn es sich bei dem Modem um ein externes Modem handelt, das über einen COM-Port verbunden ist, stellen Sie sicher, dass die folgenden Kommunikationseinstellungen übereinstimmen:
  - die erweiterten Modemparameter
  - die Parameter des Kommunikationsport
  - die Modbus-Treiberparameter

Weitere Details zur Installation und zu den Einstellungen des SR2MOD03-Modems finden Sie im Benutzerhandbuch zum Wireless-Modem SR2MOD02 und SR2MOD03 (*siehe Seite 12*).

Gehen Sie zum manuellen Hinzufügen einer Modemverbindung in der Liste **Lokale Geräte** vor wie folgt:

| Schritt | Aktion   |
|---------|--|
| 2       | <div style="display: flex; align-items: center;">  <p>Klicken Sie auf <b>Modemkonfiguration hinzufügen</b> (Schaltfläche <b>Modemkonfiguration hinzufügen</b>), um das Fenster <b>Modemkonfiguration</b> zu öffnen.<br/><b>Ergebnis:</b> Das Fenster <b>Modemkonfiguration</b> wird angezeigt.</p> </div>   |
| 3       | <p>Wählen Sie aus der Dropdown-Liste den COM-Port des Modems aus:</p> <div style="border: 1px solid gray; padding: 10px; margin: 10px 0;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <span style="color: green;"><b>Konfiguration des Modems</b></span> <span style="color: red; font-size: 1.2em;">✕</span> </div> <hr style="border: 5px solid green; margin: 5px 0;"/> <div style="margin-top: 10px;"> <p><span style="color: green;">- Einstellungen</span></p> <p>COM-Port <span style="float: right;">COM2 ▾</span></p> <p>Gerät <span style="float: right;">ELTIMA Virtual Serial Port (COM2)</span></p> <p>Telefonnummer <span style="float: right;">0512345678</span></p> <p>Modeminitialisierungsbefehl <span style="float: right;">AT&amp;D0</span></p> <p>Escape-Zeichen <span style="float: right;">+</span></p> <p>Baudrate <span style="float: right;">19200 ▾</span></p> <p>Parität <span style="float: right;">Gerade ▾</span></p> <p>Datenbits <span style="float: right;">8 ▾</span></p> <p>Stoppbits <span style="float: right;">1 ▾</span></p> <p>Timeout (ms) <span style="float: right;">15000</span></p> <p>Inter-Char Timeout (ms) <span style="float: right;">10 <input type="checkbox"/> Automatisch</span></p> <div style="text-align: right; margin-top: 10px;"> <span style="border: 1px solid gray; padding: 2px 10px;">Übernehmen</span> <span style="border: 1px solid gray; padding: 2px 10px; margin-left: 10px;">Abbrechen</span> </div> </div> </div> |

| Schritt | Aktion  |
|---------|---|
| 4       | Konfigurieren der Kommunikationsparameter.<br>Weitere Informationen über die Parameter zur Modemkonfiguration finden Sie in der nachstehenden Tabelle.  |
| 5       | Klicken Sie auf <b>Übernehmen</b> .<br><b>HINWEIS:</b> Diese Schaltfläche ist nur aktiviert, wenn alle Einstellungen richtig konfiguriert sind.<br><b>Ergebnis:</b> Die Modemverbindung wird der Liste <b>Lokale Geräte</b> hinzugefügt (z. B. <b>COM2@0612345678,GenericModem</b> ). |

In dieser Tabelle werden die Parameter zur Modemkonfiguration beschrieben:

| Parameter             | Wert  | Standardwert | Beschreibung  |
|-----------------------|---|--------------|---|
| <b>Port</b>           | COMx  | -            | Ermöglicht die Auswahl des COM-Ports des Modems aus der Dropdown-Liste.   |
| <b>Gerät</b>          | -   | -            | Zeigt den Namen des Modems an.  |
| <b>Telefonnummer</b>  | -   | -            | Ermöglicht das Eingeben der Telefonnummer des Remote-Modems, das mit der Logiksteuerung verbunden ist. In dieses Textfeld können alle Zeichen eingegeben werden, insgesamt bis zu 32 Zeichen. Es muss mindestens ein Zeichen eingegeben werden, damit die Konfiguration angewendet werden kann. |
| <b>AT init cmd</b>    | -   | AT&D0        | Ermöglicht das Bearbeiten des AT-Initialisierungsbefehls des Modems. Der AT-Initialisierungsbefehl ist optional (falls das Feld leer ist, wird die Zeichenfolge AT gesendet).   |
| <b>Escape-Zeichen</b> | -   | +            | Ermöglicht das Bearbeiten des Escape-Zeichens für den Auflegvorgang.  |
| <b>Baudrate</b>       | 1200<br>2400<br>4800<br>9600<br>19200<br>38400<br>57600<br>115200 | 19200        | Ermöglicht die Auswahl der Datenübertragungsrate (Bits pro Sekunde) des Modems.   |
| <b>Parität</b>        | Ohne<br>Gerade<br>Ungerade  | Gerade       | Ermöglicht die Auswahl der Parität der übertragenen Daten für die Fehlererkennung.  |
| <b>Datenbits</b>      | 7<br>8  | 8            | Ermöglicht die Auswahl der Anzahl der Datenbits.  |

| Parameter                  | Wert        | Standardwert | Beschreibung   |
|----------------------------|-------------|--------------|--|
| Stoppbits                  | 1<br>2      | 1            | Ermöglicht die Auswahl der Anzahl der Stoppbits.   |
| Timeout (ms)               | 0 bis 60000 | 15000        | Ermöglicht das Festlegen des Übertragungstimeouts (in ms).   |
| Unterbrechungstimeout (ms) | 0 bis 10000 | 10           | Ermöglicht das Festlegen der Eingabeverzögerung (in ms).<br>Wenn das Kontrollkästchen <b>Automatisch</b> aktiviert ist, wird der Wert automatisch berechnet. |

### Aufbauen einer Verbindung zu einer Steuerung

Gehen Sie zur Anmeldung bei einer Steuerung vor wie folgt:

| Schritt | Aktion   |
|---------|--|
| 1       | <p>Klicken Sie auf  (Schaltfläche <b>Geräte aktualisieren</b>), um die Liste verbundener Ethernet-Geräte zu aktualisieren.</p>  |
| 2       | <p>Wählen Sie eine der Logiksteuerungen in der Liste <b>Lokale Geräte</b> oder <b>Ethernet-Geräte</b> aus. Wenn eine Steuerung über Ethernet auf demselben Netzkabel wie Ihr PC verbunden ist, wird die IP-Adresse der Steuerung in der Liste angezeigt. Durch die Auswahl der IP-Adresse in der Liste wird  (Schaltfläche für <b>IP-Adressenkonfiguration</b>) verfügbar. Klicken Sie auf diese Schaltfläche, um die IP-Adresse der Steuerung zu ändern.</p> <p><b>HINWEIS:</b> Wenn Sie das Kontrollkästchen <b>Post-Konfigurationsdatei schreiben</b> aktivieren, werden die Ethernet-Parameter in der Post-Konfigurationsdatei geändert und auch nach dem Ein- und Ausschalten beibehalten.</p> |
| 3       | <p>Falls erforderlich, klicken Sie auf  (Schaltfläche <b>Blinken der LEDs starten</b>), um die LEDs der ausgewählten Steuerung blinken zu lassen, damit die Steuerung physisch anhand der blinkenden LEDs identifiziert werden kann. Klicken Sie erneut auf diese Schaltfläche, um das Blinken der LEDs einzustellen.</p> <p><b>HINWEIS:</b> Sie können die Schaltfläche <b>Blinken der LEDs starten</b> nur für Logiksteuerungen verwenden, die automatisch hinzugefügt werden (bei aktiver Option <b>Automatisches Erkennungsprotokoll aktiviert</b>).</p>  |
| 4       | <p>Klicken Sie auf <b>Login</b>, um sich bei der ausgewählten Steuerung anzumelden. Wenn die Logiksteuerung über Passwortschutz verfügt, werden Sie zur Eingabe des Passworts aufgefordert. Geben Sie das Passwort ein und klicken Sie auf <b>OK</b>, um die Verbindung herzustellen.</p> <p><b>Ergebnis:</b> Es wird eine Statusleiste angezeigt, anhand der Sie den Status des Verbindungsaufbaus mitverfolgen können.</p>   |

| Schritt | Aktion  |
|---------|---|
| 5       | <p>Sobald eine Verbindung aufgebaut wurde, wird im Bereich <b>Ausgewählte Steuerung</b> der Schutzstatus der derzeit in der Steuerung gespeicherten Anwendung angezeigt.</p> <p>Sobald die Verbindung erfolgreich hergestellt wurde, werden Detailinformationen zur Steuerung im Fensterbereich <b>Ausgewählte Steuerung</b> angezeigt:</p> <ul style="list-style-type: none"> <li>● Firmware-Revision</li> <li>● Referenznummer der Steuerung</li> <li>● Referenznummern aller mit der Steuerung verbundenen Erweiterungsmodule</li> <li>● Aktueller Status der Verbindung zwischen SoMachine Basic und Steuerung</li> </ul>   |
| 6       | <p>SoMachine Basic Stellt sicher, dass die Hardwarekonfiguration der Steuerung mit der Konfiguration des aktuellen Projekts kompatibel ist.</p> <p>Ist das der Fall, dann kann die Anwendung in die Steuerung geladen werden. Die Schaltfläche <b>PC zu SPS (Download)</b> wird verfügbar, sodass Sie den Download der Anwendung (<i>siehe Seite 268</i>) starten können.</p> <p>SoMachine Basic prüft, ob die in der Logiksteuerung gespeicherten nicht-ausführbaren Daten (Symbole, Kommentare, Animationstabellen usw.) mit denen der aktuellen Applikation identisch sind. Andernfalls wird eine Meldung mit Empfehlungen angezeigt.</p> <p>SoMachine Basic prüft außerdem, ob eine aktuellere Firmware-Version verfügbar ist und stellt gegebenenfalls einen Link bereit über den Sie die Firmwareaktualisierung starten können.</p> |

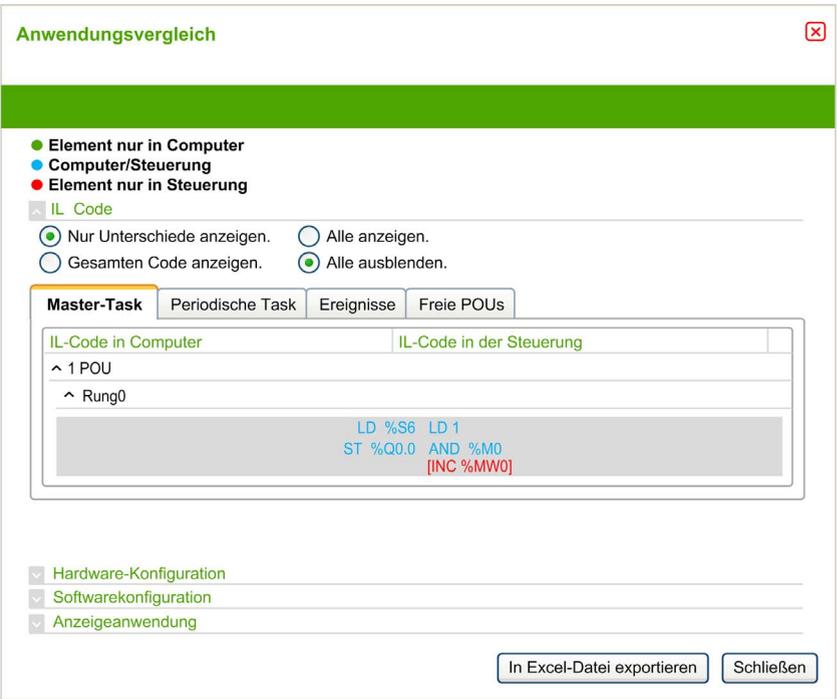
### Vergleich von Projekten bei bestehender Verbindung

Sie können die SoMachine Basic-Applikation mit der Applikation in der Logiksteuerung vergleichen. Unterschiede werden angezeigt und können dann bewertet und berücksichtigt werden.

Wenn sowohl Herunter- als auch Hochladen autorisiert ist, dann sind die Applikation des PCs und der Logiksteuerung nicht identisch und eine Meldung erscheint im Fenster **Inbetriebnahme**:

The screenshot shows a software interface with a warning message on the left and a vertical column of buttons on the right. The warning message is enclosed in a yellow box and reads: "⚠️ PC- und Steuerungsanwendungen sind nicht identisch" followed by a blue link "Computer- und Steuerungsanwendung vergleichen". The buttons on the right are: "PC zu Steuerung (Download)", "Steuerung zu PC (Upload)", "Steuerung stoppen", "Steuerung starten", "Simulator starten", and "Simulator stoppen".

Gehen Sie wie folgt vor:

| Schritt | Aktion   |
|---------|--|
| 1       | Klicken Sie in der Meldung auf <b>Computer- und Steuerungsapplikation vergleichen</b> .<br><b>Ergebnis:</b> Ein Popup-Fenster informiert Sie darüber, dass Sie die Verbindung zur Logiksteuerung trennen müssen bevor Sie den Vergleich betrachten können:   |
| 2       | Klicken Sie auf <b>OK</b> , um fortzufahren und die Verbindung zur Logiksteuerung zu trennen.  |
| 3       | <p>Daraufhin wird das Fenster <b>Applikationsvergleich</b> angezeigt:</p>  <p>Die folgenden Bereiche der Konfiguration und Applikation sind im Vergleich verfügbar:</p> <ul style="list-style-type: none"> <li>● <b>IL-Code</b></li> <li>● <b>Hardware-Konfiguration</b></li> <li>● <b>Softwarekonfiguration</b></li> <li>● <b>Anzeigeangwendung</b></li> </ul> |
| 4       | Sie haben die Möglichkeit, den Vergleich in einem Tabellenkalkulationsformat zu speichern, indem Sie auf <b>Zu Excel-Datei exportieren</b> klicken.  |

## Herunter- und Hochladen von Anwendungen

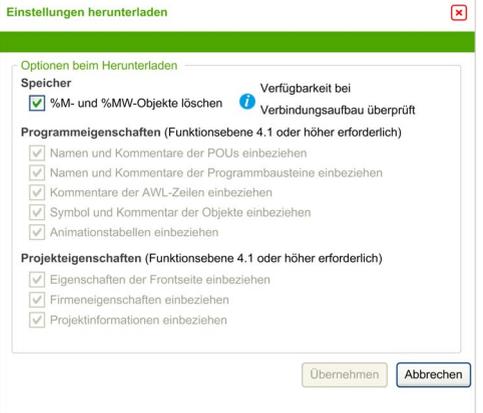
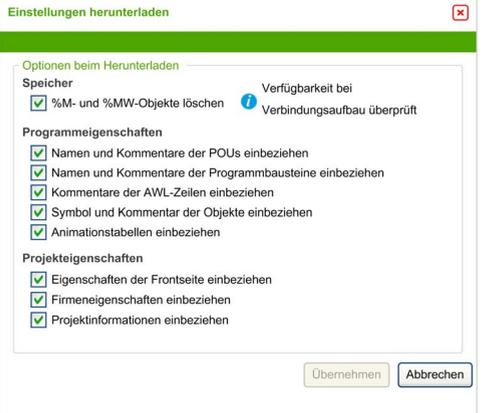
### Herunterladen der Anwendung

Gehen Sie wie folgt vor, um die aktuell in SoMachine Basic geöffnete Anwendung in die Logiksteuerung zu laden:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf <b>Verbinden</b> in der Inbetriebnahmeübersicht des Fensters <b>Inbetriebnahme</b> .  |
| 2       | Wählen Sie eine der Logiksteuerungen in der Liste <b>Lokale Geräte</b> oder <b>Ethernet-Geräte</b> aus.   |
| 3       | Klicken Sie auf <b>Login</b> , um sich bei der ausgewählten Steuerung anzumelden.<br>Wenn die Logiksteuerung mit einem Passwortschutz ausgestattet ist, geben Sie das entsprechende Passwort ein und klicken Sie auf <b>OK</b> , um die Verbindung herzustellen.  |
| 4       | <p>Oder klicken Sie auf  <b>Einstellungen herunterladen</b>.</p> <p>Wenn die Speicherwörter (%MW) und Speicherbits (%M) nach dem Download nicht zurückgesetzt werden sollen, dann deaktivieren Sie die Option <b>Speicher zurücksetzen</b>.</p> <p><b>HINWEIS:</b> Die Option unter <b>Speicher</b> ist nur für Steuerungen mit einer Firmwareversion ab 1.3.3.y verfügbar.<br/>Die Optionen unter <b>Programmeigenschaften</b> und <b>Projekteigenschaften</b> sind nur für Steuerungen mit einer Firmwareversion ab 1.4.1.y verfügbar.</p> |
| 5       | <p>Klicken Sie auf <b>PC zu SPS (Download)</b>.</p> <p>Sollte die Schaltfläche <b>PC zu SPS (Download)</b> nicht verfügbar sein, prüfen Sie Folgendes:</p> <ul style="list-style-type: none"> <li>Die in der Steuerung gespeicherte Anwendung ist mit der SoMachine Basic-Anwendung identisch.</li> <li>Die Hardwarekonfiguration des Steuerungssystems ist nicht mit der Konfiguration in der SoMachine Basic-Anwendung kompatibel.</li> </ul>   |
| 6       | Wenn die nicht-ausführbaren Daten der aktuellen Applikation nicht mit den im Computer gespeicherten Daten identisch sind, dann werden nur die nicht-ausführbaren Daten in die Steuerung heruntergeladen.  |
| 7       | Wenn die Anwendung für <b>Start in RUN</b> konfiguriert ist, wird ein Gefahrenhinweis angezeigt, in dem Sie aufgefordert werden, die entsprechende Konfiguration der Steuerung zu bestätigen. Klicken Sie auf <b>OK</b> , um den Download der Anwendung zu bestätigen, oder klicken Sie auf <b>Abbrechen</b> und ändern Sie die Konfiguration.  |
| 8       | <p>Klicken Sie auf <b>OK</b>, um die Übertragung fortzusetzen und die Anwendung in der Steuerung zu überschreiben.</p> <p><b>Ergebnis:</b> Es wird eine Statusleiste angezeigt, der Sie den Verbindungsstatus entnehmen können.</p>   |
| 9       | <p>Um die heruntergeladene Anwendung auszuführen, klicken Sie auf <b>Steuerung in Betrieb setzen</b> und dann auf <b>OK</b>, um den Vorgang zu bestätigen.</p> <p>Wenn Sie eine Meldung mit dem Hinweis erhalten, dass die Betriebsart nicht geändert werden kann, klicken Sie auf <b>Schließen</b> und prüfen Sie den RUN/STOP-Schalter an der Steuerung und/oder den RUN/STOP-Eingang, da diese Komponenten eventuell den Wechsel der Steuerung in den Zustand <b>RUNNING</b> verhindern. Anderenfalls schlagen Sie weitere Informationen im <i>Hardwarehandbuch</i> der Logiksteuerung nach.</p>                           |

## Einstellen der Download-Optionen

Um die **Download-Optionen** anzuzeigen, klicken Sie auf  **Einstellungen herunterladen** im Fenster **Inbetriebnahme**.

| Funktionsebene ≤ 4.1  | Funktionsebene ≥ 4.1   |
|---|--|
|  |  |
| Die Einstellungen werden nicht hochgeladen  | Die Einstellungen werden hochgeladen.  |

Jede Einstellung wird standardmäßig ausgewählt. Wenn Sie im Online-Modus eine Option auswählen oder deaktivieren, dann klicken Sie auf **PC zu Steuerung (Download)**, um die Änderungen herunterzuladen.

Wenn Sie im Online-Modus den Namen oder Kommentare einer POU, eines Programmbausteins oder einer AWL-Zeile ändern und diese entsprechenden Optionen unter **Einstellungen herunterladen** ausgewählt sind, dann wird der Download automatisch ausgeführt.

Die Option **Speicher zurücksetzen** ist standardmäßig ausgewählt. Diese Option ist sowohl im Offline- als auch im Online-Modus verfügbar.

Wenn **Speicher zurücksetzen** ausgewählt ist, dann werden die Speicherwörter und -Bits beim Herunterladen der Applikation auf 0 zurückgesetzt.

Wenn **Speicher zurücksetzen** nicht aktiv ist, dann behalten die Speicherwörter und -bits ihre Werte.

Wenn sich die Größe des zugewiesenen Speichers der Applikation im PC-Speicher von der Größe des Speichers der Applikation in der Logiksteuerung unterscheidet, wird wie folgt mit dem Speicher umgegangen:

- Wenn der zugewiesene %MWx der Applikation der Logiksteuerung größer ist als der zugewiesene %MWx der Applikation im PC-Speicher, dann wird die Zuweisung der Applikation im PC verwendet und die zusätzlichen %MWx-Wörter werden auf 0 gesetzt.
- Wenn der zugewiesene %MWx in der Applikation der Logiksteuerung kleiner ist als der zugewiesene %MWx der Applikation im PC-Speicher, dann werden die zusätzlichen %MWx-Wörter vom Speicherplatz entfernt.
- Wenn sich in der Logiksteuerung keine Applikation befindet, werden %MW auf 0 gesetzt. Die selben Regeln gelten für %M. Die Download-Einstellungen sind projektabhängig und werden im Projekt gespeichert.

### Hochladen einer Applikation

Gehen Sie vor wie folgt, um die in der Steuerung gespeicherte Anwendung in SoMachine Basic hochzuladen:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf <b>Verbinden</b> in der Inbetriebnahmeübersicht des Fensters <b>Inbetriebnahme</b> .  |
| 2       | Wählen Sie eine der Logiksteuerungen in der Liste <b>Lokale Geräte</b> oder <b>Ethernet-Geräte</b> aus.   |
| 3       | Klicken Sie auf <b>Login</b> , um sich bei der ausgewählten Steuerung anzumelden.<br>Wenn die Logiksteuerung mit einem Passwortschutz ausgestattet ist, geben Sie das entsprechende Passwort ein und klicken Sie auf <b>OK</b> , um die Verbindung herzustellen.                                    |
| 4       | Klicken Sie auf <b>SPS zu PC (Upload)</b> .<br>Wenn die Schaltfläche <b>SPS zu PC (Upload)</b> nicht verfügbar ist, prüfen Sie, ob die in der Steuerung gespeicherte Anwendung mit der SoMachine Basic-Anwendung identisch ist.   |
| 5       | Klicken Sie auf <b>OK</b> , um den Upload aus der Logiksteuerung zu bestätigen.<br><b>Ergebnis:</b> Es wird eine Statusleiste angezeigt, der Sie den Verbindungsstatus entnehmen können. Bei Abschluss der Übertragung wurde die Anwendung aus der Logiksteuerung nach SoMachine Basic hochgeladen. |

**HINWEIS:** Der Wert der Option **Speicher zurücksetzen** wird beim Hochladen einer Applikation nicht gespeichert.

## Herunterladen oder Hochladen einer passwortgeschützten Anwendung

Wenn Sie eine Anwendung herunter- oder hochladen, die in einer früheren Version von SoMachine Basic mit einem Passwortschutz versehen wurde, dann sind die Aktionen, die ausgeführt werden können oder müssen, von den Versionen abhängig:

| Betrieb              | SoMachine Basic-Version | Funktionsebene der Anwendung | Firmware-Version | Aktionen   |
|----------------------|-------------------------|------------------------------|------------------|--|
| <b>Herunterladen</b> |                         |                              |                  |  |
|                      | ≤1,5                    | ≤5,0                         | ≤1,5             | Der Downloadvorgang wendet nicht die neueste Sicherheitsstrategie an.  |
|                      |                         |                              | ≥ 1.5.1          | Der Downloadvorgang ist möglich und das Passwort wird angezeigt.   |
|                      |                         | ≥ 5.1                        | –                | Download ist nicht möglich.  |
|                      | ≥ 1.5 SP1               | ≤5,0                         | ≤ 1,5            | Wenn die Anwendung passwortgeschützt ist, dann ist ein Download nicht möglich.   |
|                      |                         |                              | ≥ 1.5.1          | Sie müssen eine der folgenden Aktionen ausführen: <ul style="list-style-type: none"> <li>● Aktualisieren Sie die Funktionsebene auf <b>5.0</b>.</li> <li>● Lassen Sie das Passwort-Feld leer.</li> <li>● Deaktivieren Sie den Anwendungsschutz.</li> </ul> |
|                      |                         | ≥ 5.1                        | ≤ 1,5            | Download ist nicht möglich.  |
|                      |                         |                              | ≥ 1.5.1          | Der Downloadvorgang wendet die neueste Sicherheitsstrategie an.  |
| <b>Hochladen</b>     |                         |                              |                  |  |
|                      | ≤1,5                    | ≤5,0                         | ≤1,5             | Der Uploadvorgang wendet nicht die neueste Sicherheitsstrategie an.  |
|                      |                         |                              | ≥ 1.5.1          | Sie müssen eine der folgenden Aktionen ausführen: <ul style="list-style-type: none"> <li>● Stufen Sie die Firmwareversion herab.</li> <li>● Aktualisieren Sie die SoMachine Basic-Version.</li> </ul>  |
|                      |                         | ≥ 5.1                        | –                | Upload ist nicht möglich.  |
|                      | ≥ 1.5 SP1               | ≤5,0                         | ≤1,5             | Der Uploadvorgang wendet nicht die neueste Sicherheitsstrategie an.  |
|                      |                         |                              | ≥ 1.5.1          | Der Uploadvorgang wendet die neueste Sicherheitsstrategie an.  |
|                      |                         | ≥ 5.1                        | ≤1,5             | Upload ist nicht möglich.  |
|                      |                         |                              | ≥ 1.5.1          | Der Uploadvorgang wendet die neueste Sicherheitsstrategie an.  |

## Abschnitt 7.3

### Steuerungs-Update

---

#### Firmware-Updates der Steuerung

##### Übersicht

Sie können Firmwareaktualisierungen entweder direkt über SoMachine Basic oder über eine SD-Karte in die Steuerung herunterladen.

##### Herunterladen einer Firmwareaktualisierung in die Steuerung

Durch die Aktualisierung der Firmware wird das in der Steuerung enthaltene Anwendungsprogramm einschließlich der Bootanwendung im nicht-flüchtigen Speicher geschützt.

Gehen Sie vor wie folgt, um Firmwareaktualisierungen in die Steuerung herunterzuladen:

| Schritt | Aktion  |
|---------|---|
| 1       | Stellen Sie sicher, dass Sie nicht mit der Steuerung verbunden sind, wenn Sie das <b>Firmware-Update</b> verwenden. |
| 2       | Klicken Sie auf <b>Inbetriebnahme → SPS-Update</b> .  |
| 3       | Klicken Sie auf <b>Aktualisieren</b> .<br>Die erste Seite des Ausführungsassistenten (OS-Loader) erscheint.         |

Wenn Sie während der Übertragung der Anwendung die Spannungszufuhr zum Gerät trennen oder ein Stromausfall bzw. eine Unterbrechung der Kommunikation auftritt, kann das die Funktionsunfähigkeit des Geräts verursachen. Sollte die Kommunikation unterbrochen werden oder ein Stromausfall auftreten, dann führen Sie die Übertragung erneut durch. Wenn bei der Aktualisierung der Firmware ein Spannungsausfall oder eine Unterbrechung der Kommunikation auftritt, oder wenn eine ungültige Firmware verwendet wird, wird die Maschine betriebsunfähig. In diesem Fall verwenden Sie eine gültige Firmware und starten das Firmware-Update erneut.

## ***HINWEIS***

### **GERÄT NICHT BETRIEBSBEREIT**

- Unterbrechen Sie die Übertragung des Anwendungsprogramms oder einer Firmware-Änderung nicht, nachdem die Übertragung begonnen hat.
- Wenn die Übertragung aus irgendeinem Grund unterbrochen wurde, starten Sie die Übertragung erneut.
- Versuchen Sie keinesfalls, das Gerät (Logiksteuerung, Bewegungssteuerung, HMI-Controller oder Antrieb) in Betrieb zu nehmen, bevor die Dateiübertragung erfolgreich abgeschlossen wurde.

**Die Nichtbeachtung dieser Anweisungen kann Sachschäden zur Folge haben.**

## Abschnitt 7.4

### Speicherverwaltung

#### Verwaltung des Steuerungsspeichers

##### Überblick

In SoMachine Basic können Sie das Sichern, Wiederherstellen oder Löschen verschiedener Elemente von oder auf der Steuerung, mit der Sie verbunden sind, ausführen.

Die Optionen Sichern, Wiederherstellen und Löschen sind nur im Online-Modus verfügbar.

### WARNUNG

#### UNBEABSICHTIGTER GERÄTEBETRIEB

- Stellen Sie sicher, dass die angeschlossene Steuerung das richtige Zielgerät ist, bevor Sie die Operationen ‚Löschen‘ oder ‚Wiederherstellen‘ ausführen.
- Überprüfen Sie den Sicherheitszustand der Maschinen- oder Prozessumgebung, bevor Sie die Operationen ‚Löschen‘ oder ‚Wiederherstellen‘ von einem dezentralen Ort ausführen.

**Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.**

#### Sichern auf einem PC oder einer Steuerungs-SD-Karte

Folgen Sie diesen Schritten, um den Speicher der Steuerung auf einem PC oder einer Steuerungs-SD-Karte zu sichern:

| Schritt | Aktion   |
|---------|--|
| 1       | Melden Sie sich bei der Steuerung an.  |
| 2       | Wählen Sie <b>Speicherverwaltung</b> im linken Bereich des Fensters <b>Inbetriebnahme</b> aus.   |
| 3       | Wählen Sie in der Liste <b>Aktion</b> den Eintrag <b>Von Steuerung sichern</b> aus.  |
| 4       | Zur Sicherung auf dem PC: Wählen Sie unter <b>Ziel PC</b> . Klicken Sie auf die Schaltfläche „Durchsuchen“ und navigieren Sie zu dem Ordner, in dem die Sicherungsdatei geschrieben werden soll.<br>oder<br>Zur Sicherung auf einer SD-Karte: Wählen Sie unter <b>Ziel</b> den Ordner <b>SD-Karte der Steuerung</b> . Schieben Sie die SD-Karte in den SD-Kartensteckplatz der Steuerung ein.<br><br><b>HINWEIS:</b> Die SD-Karte darf weder leer sein, noch eine <code>script.cmd</code> -Datei enthalten, um eine Erstellung eines Klon der Skript-Ausführung zu vermeiden ( <i>siehe Modicon M221, Logic Controller, Programmierhandbuch</i> ). |

| Schritt | Aktion   |
|---------|--|
| 5       | <p>Wählen Sie die zu sichernden Elemente aus, indem Sie die folgende Optionen auswählen:</p> <ul style="list-style-type: none"> <li>● <b>Firmware sichern</b></li> <li>● <b>Programm sichern</b></li> <li>● <b>Speicherwerte sichern</b></li> <li>● <b>Protokolldatei sichern</b></li> <li>● <b>Post-Konfigurationsdatei sichern</b></li> </ul> <p>Wenn <b>Speicherwerte sichern</b> in einer PC-Sicherung ausgewählt ist, spezifizieren Sie das <b>Erste Speicherbit</b>, das <b>Letzte Speicherbit</b>, das <b>Erste Speicherwort</b> und das <b>Letzte Speicherwort</b>, welche in der Sicherung umfasst werden sollen.</p> |
| 6       | <p>Klicken Sie auf <b>Von Steuerung sichern</b>, um den Sicherungsvorgang zu starten.</p> <p>Die Elemente werden im spezifizierten PC-Ordner oder auf der SD-Karte als SD-Karten-Image gespeichert (.smbk).</p> <p>In einem Berichtsfenster wird die Liste der Informations- und Fehlermeldungen zum Sicherungsvorgang angezeigt.</p>  |

**HINWEIS:** Wenn Sie Speicherwerte sichern, können Sie die Sicherung initiieren, wenn sich die Steuerung im **RUNNING**-Zustand befindet. Abhängig davon, wie viele Speichervariablen zur Aufnahme in die Sicherung angegeben wurden, kann es jedoch sein, dass eine Sicherung nicht zwischen den Logikabfragen fertiggestellt werden kann. Folglich wäre die Sicherung nicht unbedingt kohärent, da sich der Wert von Speichervariablen von einer Abfrage zur nächsten verändern könnte. Wenn Sie für die Variablen einen konsistenten Satz von Werten wünschen, müssen Sie die Steuerung u. U. zunächst in den Zustand **STOPPED** setzen.

## Wiederherstellen

Gehen Sie wie folgt vor, um die Elemente der Steuerung von einem PC wiederherzustellen:

| Schritt | Aktion   |
|---------|--|
| 1       | Melden Sie sich bei der Steuerung an.  |
| 2       | Wählen Sie <b>Speicherverwaltung</b> im linken Bereich des Fensters <b>Inbetriebnahme</b> aus.   |
| 3       | Wählen Sie in der Liste <b>Aktion</b> den Eintrag <b>In Steuerung wiederherstellen</b> aus.  |
| 4       | Wählen Sie den Quellordner aus, der die Sicherungsdateien auf dem PC enthält.  |
| 5       | Wählen Sie die Elemente aus, die auf der Steuerung wiederhergestellt werden sollen.  |
| 6       | Klicken Sie auf <b>In Steuerung wiederherstellen</b> , um den Wiederherstellungsvorgang zu starten. In einem Berichtsfenster wird die Liste der Informations- und Fehlermeldungen zum Wiederherstellungsvorgang angezeigt. |

Unvollständige Dateiübertragungen, wie Datendateien, Anwendungsdateien und/oder Firmwaredateien, können ernsthafte Folgen für Ihre Maschine oder Ihre Steuerung haben/ Wenn Sie während der Dateiübertragung den Strom abschalten oder ein Stromausfall oder eine Kommunikationsunterbrechung auftritt, kann Ihre Maschine nicht mehr operativ sein oder Ihre Anwendung kann versuchen, mit beschädigten Dateidaten zu arbeiten. Sollte die Kommunikation unterbrochen werden, dann führen Sie die Übertragung erneut durch. Stellen Sie sicher, dass Sie den Effekt beschädigter Daten in Ihrer Risikoanalyse berücksichtigen.

|   |
|---|
|  <h2 style="margin: 0;">WARNUNG</h2>   |
| <p><b>UNBEABSICHTIGTER GERÄTEBETRIEB, DATENVERLUST ODER DATEIBESCHÄDIGUNG</b></p> <ul style="list-style-type: none"> <li>● Unterbrechen Sie eine laufende Datenübertragung nicht.</li> <li>● Wenn die Übertragung aus einem beliebigen Grund unterbrochen wird, starten Sie sie erneut.</li> <li>● Nehmen Sie Ihre Maschine nicht in Betrieb bis die Dateiübertragung erfolgreich abgeschlossen wurde, es sei denn, sie haben die beschädigten Daten in Ihre Risikoanalyse miteinbezogen und entsprechende Schritte eingeleitet, um mögliche ernste Folgen wegen einer nicht erfolgreichen Datenübertragung zu vermeiden.</li> </ul> <p><b>Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.</b></p> |

Zum Wiederherstellen einer Sicherung von einer Steuerungs-SD-Karte, siehe das *Programmierhandbuch* der Logiksteuerung.

### Löschen von Steuerungselementen

Gehen Sie wie folgt vor, um die Elemente der Steuerung zu löschen:

| Schritt | Aktion  |
|---------|---|
| 1       | Wählen Sie <b>Speicherverwaltung</b> im linken Bereich des Fensters <b>Inbetriebnahme</b> aus.  |
| 2       | Wählen Sie in der Liste <b>Aktion</b> den Eintrag <b>In Steuerung löschen</b> aus.  |
| 3       | <p>Wählen Sie die Elemente aus, die auf der Steuerung gelöscht werden sollen.<br/>Wenn Sie die Option <b>Post-Konfigurationsdatei löschen</b> auswählen, wird die Post-Konfigurationsdatei sofort beim Klicken auf <b>In Steuerung löschen</b> gelöscht. Um bestehende Ethernet-Verbindungen zu erhalten, wird diese Löschung der Datei erst dann von der Steuerung berücksichtigt wenn Ethernet reinitialisiert wird, also nach einem der folgenden Ereignisse:</p> <ul style="list-style-type: none"> <li>● Trennen und erneutes Verbinden des Ethernet-Kabels</li> <li>● Initialisierung der Steuerung</li> <li>● Aus- und Einschalten Sie die Steuerung.</li> </ul> |
| 4       | Klicken Sie auf <b>In Steuerung löschen</b> , um den Löschvorgang zu starten.<br>In einem Berichtsfenster wird die Liste der Informations- und Fehlermeldungen zum Löschvorgang angezeigt.  |

## Erstellen und Lesen von Images (Bilder) der Logiksteuerung

Ein Steuerungsabbild umfasst die Firmware, das Programm und die Postkonfigurationsdatei der Steuerung. Ein Skript ermöglicht die Übertragung dieser Elemente in eine Steuerung.

Wenn Sie beim Erstellen eines Steuerungsabbilds eine SD-Karte als Ziel auswählen, dann kann diese SD-Karte in der Steuerung verwendet werden.

## Erstellen eines Steuerungsabbilds

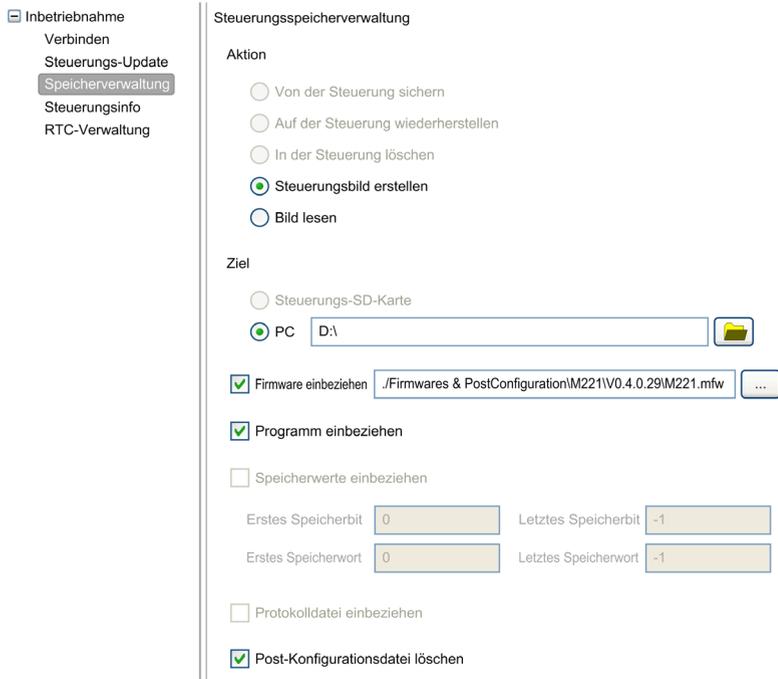
Im Offline-Modus ermöglicht dieses Verfahren die Generierung eines Skripts und das Kopieren von Dateien, die für das Kopieren folgender Elemente auf Ihren PC oder eine SD-Karte erforderlich sind:

- Firmware der installierten SoMachine Basic-Software.
- Programm des aktuell geöffneten Projekts.
- Post-Konfigurationsdatei.

Gehen Sie wie folgt vor, um ein Steuerungsabbild zu erstellen:

| Schritt | Aktion  |
|---------|---|
| 1       | Wenn eine Verbindung zu einer Steuerung besteht, klicken Sie im Fenster <b>Inbetriebnahme</b> auf „Abmelden“.   |
| 2       | Wählen Sie <b>Speicherverwaltung</b> im linken Bereich des Fensters <b>Inbetriebnahme</b> aus.  |
| 3       | Wählen Sie in der Liste <b>Aktion</b> den Eintrag <b>Steuerungsabbild erstellen</b> aus.  |
| 4       | Klicken Sie unter <b>Ziel</b> → <b>PC</b> auf die Schaltfläche „Durchsuchen“ und navigieren Sie zu dem Ordner, in dem die Bilddatei geschrieben werden soll.<br>Sie können eine in Ihren PC eingesetzte SD-Karte als Ziel auswählen.  |
| 5       | Wählen Sie die zu kopierenden Elemente über folgende Auswahl aus: <ul style="list-style-type: none"> <li>● Firmware einbeziehen</li> <li>● Programm einbeziehen</li> </ul>  |
| 6       | Wenn Sie die Postkonfigurationsdatei überschreiben möchten, wählen Sie <b>Post-Konfigurationsdatei löschen</b> aus.   |
| 7       | Klicken Sie auf <b>Erzeuge Controller Image</b> .<br><b>Ergebnis:</b> Die folgenden Ordner und Dateien werden erstellt: <ul style="list-style-type: none"> <li>● <code>script.cmd</code></li> <li>● <code>usr/app/*.smbk</code></li> <li>● <code>sys/os/*.mfw</code></li> </ul> |
| 8       | Wenn Sie das Steuerungsabbild auf Ihrem PC erstellt haben, dann kopieren Sie die Dateien auf eine SD-Karte.   |

Die folgende Abbildung zeigt ein Beispiel der Einstellungen:



### Lesen eines Steuerungsabbilds

Im Offline-Modus ermöglicht dieses Verfahren das Öffnen einer `.smbk`-Bilddatei in SoMachine Basic als Projekt.

**HINWEIS:** Das geöffnete Bild muss zuvor mit der Aktion **Erzeuge Controller Image** oder durch eine Sicherung von der Steuerung (*siehe Seite 277*) erstellt worden sein.

Gehen Sie wie folgt vor, um ein Steuerungsabbild zu lesen:

| Schritt | Aktion   |
|---------|--|
| 1       | Wenn eine Verbindung zu einer Steuerung besteht, klicken Sie im Fenster <b>Inbetriebnahme</b> auf „Abmelden“.  |
| 2       | Wählen Sie <b>Speicherverwaltung</b> im linken Bereich des Fensters <b>Inbetriebnahme</b> aus.   |
| 3       | Wählen Sie in der Liste <b>Aktion</b> den Eintrag <b>Lese Image</b> aus.   |
| 4       | Klicken Sie unter <b>Quelle</b> → <b>PC</b> auf die Schaltfläche „Durchsuchen“ und navigieren Sie zu dem Ordner, der die Bilddatei ( <code>.smbk</code> ) enthält.<br><b>Programm lesen</b> ist standardmäßig ausgewählt. Um eine Bilddatei lesen zu können, muss diese ausgewählt werden. |
| 5       | Klicken Sie auf <b>Lese Image</b> , um das Programm zu lesen und ein Projekt zu öffnen.  |

Die folgende Abbildung zeigt ein Beispiel der Einstellungen:

Inbetriebnahme  
Verbinden  
Steuerungs-Update  
**Speicherverwaltung**  
Steuerungsinfo  
RTC-Verwaltung

Steuerungsspeicherverwaltung

**Aktion**

Von der Steuerung sichern  
 Auf der Steuerung wiederherstellen  
 In der Steuerung löschen  
 Steuerungsbild erstellen  
 Bild lesen

**Quelle**

Steuerungs-SD-Karte  
 PC  

Firmware lesen  
 **Programm lesen**  
 Speicherwerte lesen

Erstes Speicherbit  Letztes Speicherbit

Erstes Speicherwort  Letztes Speicherwort

Protokolldatei lesen  
 Post-Konfigurationsdatei lesen

# Abschnitt 7.5

## SPS-Infos

---

### Informationen zur Steuerung

#### Überblick

Klicken Sie auf **SPS-Infos** im linken Bereich des Fensters **Inbetriebnahme**, um folgende Informationen zum aktuellen Zustand der Steuerung anzuzeigen:

- **Ausführbarer RAM-Speicher:** Mit dieser Option wird geprüft, ob im RAM-Speicher der Steuerung eine gültige Anwendung vorhanden ist. Diese Information kann ebenfalls innerhalb eines Programms durch den Test des Bits 14 Systemworts %SW7 (*siehe Modicon M221, Logic Controller, Programmierhandbuch*) abgerufen werden.
- **Geschützter RAM-Speicher:** Diese Option ist ausgewählt, wenn die Anwendung im RAM-Speicher der Steuerung mit einem Passwort geschützt ist. Diese Information kann ebenfalls innerhalb eines Programms durch den Test des Bits 8 Systemworts %SW7 (*siehe Modicon M221, Logic Controller, Programmierhandbuch*) abgerufen werden.
- **Forcierte E/A:** Diese Option ist aktiviert, wenn 1 oder mehrere digitale Ein- oder Ausgänge der Steuerung auf einen bestimmten Wert forciert wurden (*siehe Seite 159*). In diesem Fall wird der Systembit %S14 (*siehe Modicon M221, Logic Controller, Programmierhandbuch*) (E/A-Forcierung aktiviert) auf 1 gesetzt.
- **RAM mit Flash synchronisiert:** Diese Option ist aktiv, wenn die im nicht-flüchtigen Speicher gespeicherte Applikation nicht mit der Applikation im **RAM-Speicher** übereinstimmt. Diese Option ist nicht aktiv, wenn:
  - Online-Änderungen an der Applikation noch nicht an die Logiksteuerung gesendet wurden (durch Klicken der Schaltfläche **Sicherung** auf dem Programmierungsregister).
  - Die Logiksteuerung noch nicht initialisiert wurde, seitdem die Änderungen vorgenommen wurden (durch Klicken der Schaltfläche **Steuerung initialisieren** in der Werkzeugleiste).
- **Status:** Aktueller Status der Logiksteuerung. Diese Information kann ebenfalls innerhalb eines Programms durch den Test des Systemworts %SW6 abgerufen werden. Weitere Informationen zu Steuerungszuständen finden Sie im *Programmierhandbuch* Ihrer Steuerung.
- **Letzter Halt:** Zeitpunkt (Datum und Uhrzeit), zu dem die Steuerung zuletzt angehalten wurde (STOPPED, HALTED usw.). Diese Information kann ebenfalls innerhalb eines Programms durch den Test des Systemwörter %SW54 bis einschließlich %SW57 abgerufen werden.
- **Grund für letzten Halt:** Gibt den Grund für den letzten Halt der Logiksteuerung an. Diese Information kann ebenfalls innerhalb eines Programms durch den Test des Systemworts %SW58 abgerufen werden.

- **Abfragezeit ( $\mu$ s):** Die folgenden Abfragezeiten:
  - **Minimum** (in Mikrosekunden): Kürzeste Abfragezeit seit dem letzten Einschalten der Logiksteuerung.  
Diese Information kann ebenfalls innerhalb eines Programms durch den Test des Systemworts %SW32 abgerufen werden (in Millisekunden).
  - **Aktuell** (in Mikrosekunden): Die Abfragezeit.  
Diese Information kann ebenfalls innerhalb eines Programms durch den Test des Systemworts %SW30 abgerufen werden (in Millisekunden).
  - **Maximum** (in Mikrosekunden): Längste Abfragezeit seit dem letzten Einschalten der Logiksteuerung.  
Diese Information kann ebenfalls innerhalb eines Programms durch den Test des Systemworts %SW31 abgerufen werden (in Millisekunden).
- **SPS-Zeit:** Die folgenden Informationen werden nur angezeigt, wenn die Logiksteuerung über eine Echtzeituhr (RTC) verfügt:
  - **Datum** (TT/MM/JJJJ): Das aktuelle in der Logiksteuerung gespeicherte Datum.  
Diese Information kann ebenfalls innerhalb eines Programms durch den Test der Systemwörter %SW56 und %SW57 abgerufen werden.
  - **Zeit** (HH:MM:SS): Die aktuelle in der Logiksteuerung gespeicherte Zeit.  
Diese Information kann ebenfalls innerhalb eines Programms durch den Test der Systemwörter %SW54 und %SW55 abgerufen werden.

Datum und Uhrzeit werden im gleichen Format angegeben, wie auf dem PC festgelegt ist.
- **Ethernet-Information:** Die folgenden Informationen werden nur angezeigt, wenn die Logiksteuerung über eine integrierte Ethernet-Verbindung verfügt:
  - **IP-Adresse:** Die IP-Adresse der Logiksteuerung.  
Diese Information kann ebenfalls innerhalb eines Programms durch den Test der Systemwörter (*siehe Modicon M221, Logic Controller, Programmierhandbuch*) %SW33 und %SW34 abgerufen werden.
  - **Subnetzmaske:** Die Subnetzmaske der Logiksteuerung.  
Diese Information kann ebenfalls innerhalb eines Programms durch den Test der Systemwörter %SW35 und %SW36 abgerufen werden.
  - **Gateway-Adresse:** Die Gateway-Adresse der Logiksteuerung.  
Diese Information kann ebenfalls innerhalb eines Programms durch den Test der Systemwörter %SW37 und %SW38 abgerufen werden.
- **Postkonfigurations-Status für SL1:** Die Parameter mit aktiviertem Kontrollkästchen sind in der Post-Konfigurationsdatei definiert. Diese Information kann ebenfalls innerhalb eines Programms durch den Test des Systemworts %SW98 (*siehe Modicon M221, Logic Controller, Programmierhandbuch*) abgerufen werden.

- **Postkonfigurations-Status für SL2:** Die Parameter mit aktiviertem Kontrollkästchen sind in der Post-Konfigurationsdatei definiert. Diese Information kann ebenfalls innerhalb eines Programms durch den Test des Systemworts %SW99 (*siehe Modicon M221, Logic Controller, Programmierhandbuch*) abgerufen werden.
- **Postkonfigurations-Status für ETH:** Die Parameter mit aktiviertem Kontrollkästchen sind in der Post-Konfigurationsdatei definiert. Diese Information kann ebenfalls innerhalb eines Programms durch den Test des Systemworts %SW100 (*siehe Modicon M221, Logic Controller, Programmierhandbuch*) abgerufen werden.

## Abschnitt 7.6

### RTC-Verwaltung

#### Verwalten der RTC

##### Übersicht

Im Fenster **RTC-Verwaltung** können Sie die Echtzeituhr (RTC) der Steuerung einstellen. Dies ist nur möglich, wenn SoMachine Basic mit einer Logiksteuerung verbunden ist, die eine Echtzeituhr (RTC) unterstützt.

##### Aktualisieren der RTC

| Schritt | Aktion   |
|---------|--|
| 1       | Wählen Sie die Option <b>RTC-Verwaltung</b> im linken Bereich des Fensters <b>Inbetriebnahme</b> aus.  |
| 2       | Wenn Sie sich im Online-Modus befinden, wird die <b>Aktuelle SPS-Zeit</b> angezeigt.<br>Wählen Sie den Modus für die Einstellung der Steuerungszeit aus: <ul style="list-style-type: none"><li>● <b>Manuell</b>: In diesem Modus werden Auswahlfelder für das Datum und die Uhrzeit angezeigt, sodass Sie manuell auswählen können, welches Datum und welche Uhrzeit in der Steuerung eingestellt werden sollen.</li><li>● <b>Automatisch</b>: In diesem Modus wird die Uhrzeit in der Steuerung auf die aktuelle Uhrzeit des PC eingestellt, auf dem SoMachine Basic installiert ist.</li></ul> |
| 3       | Klicken Sie auf <b>Übernehmen</b> .  |



---

# Kapitel 8

## Simulator

---

### Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

| Thema   | Seite |
|---|-------|
| Überblick über den SoMachine Basic-Simulator                  | 286   |
| E/A-Manager-Fenster des SoMachine Basic-Simulators            | 288   |
| SoMachine Basic-Simulator – Fenster „ <b>Zeitverwaltung</b> “ | 290   |
| Ändern von Werten mit dem SoMachine Basic-Simulator           | 293   |
| Verwendung des SoMachine Basic-Simulators                     | 299   |
| Starten der Simulation in Vijeo-Designer                      | 300   |

## Überblick über den SoMachine Basic-Simulator

### Einführung

Mit dem SoMachine Basic-Simulator können Sie:

- eine Verbindung zwischen dem PC, der Logiksteuerung und einem beliebigen Erweiterungsmodul simulieren.
- ein Testprogramm ausführen, ohne eine Logiksteuerung und Erweiterungsmodule an den PC anzuschließen.

Der Simulator repliziert das Verhalten der Logiksteuerung und ist eine virtuelle Logiksteuerung, mit der Sie sich über SoMachine Basic verbinden können.

**HINWEIS:** Sicherheitsparameter (*siehe Modicon M221, Logic Controller, Programmierhandbuch*) werden bei der Verwendung des Simulators nicht angewandt.

Sobald Sie den Simulator starten, können Sie diesen verbinden, ausführen, stoppen und andere zugehörige Aktionen durchführen, die Sie bei Verbindung mit einer physischen Steuerung durchführen würden.

**HINWEIS:** Der Simulator unterstützt bis zu 2 Verbindungen: eine für SoMachine Basic und eine andere für Datenzwecke (beispielsweise HMI-Kommunikation).

### Zugriff auf den SoMachine Basic-Simulator

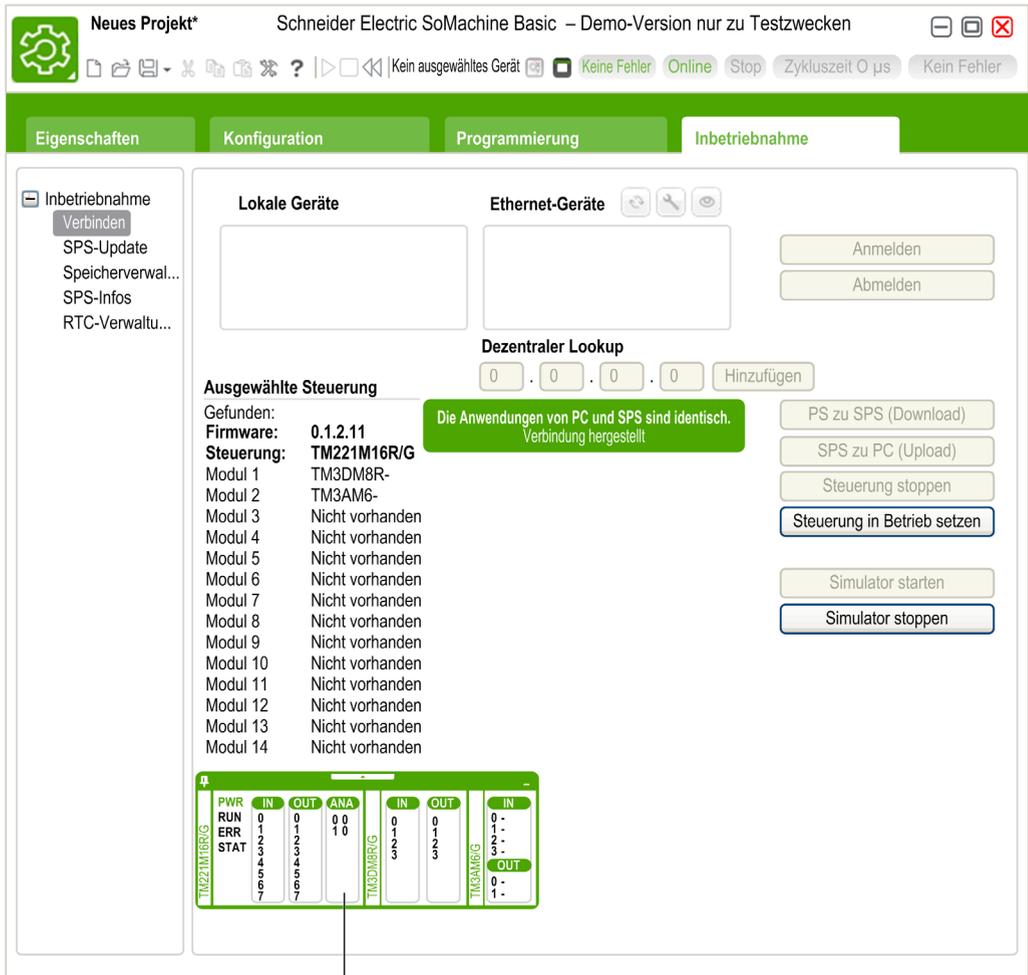
| Schritt | Aktion   |
|---------|--|
| 1       | Stellen Sie sicher, dass das Programm gültig ist. Andernfalls kann es sein, dass der Simulatorstart durch eine Fehlermeldung zu einem Kompilierfehler unterbrochen wird, die auf dem Bildschirm angezeigt wird.  |
| 2       | Starten Sie den Simulator mit einer der folgenden Methoden: <ul style="list-style-type: none"> <li>• Klicken Sie im Taskbereich für die Inbetriebnahme auf <b>Simulator starten</b>.</li> <li>• Drücken Sie im Fenster <b>Inbetriebnahme</b> die Tasten STRG+B.</li> </ul><br><ul style="list-style-type: none"> <li>• Klicken Sie in der SoMachine Basic-Symbolleiste auf  (Simulator starten).</li> </ul> |

### SoMachine Basic-Simulator-Fenster

Der SoMachine Basic-Simulator umfasst die folgenden zwei Fenster:

- **Zeitverwaltungsfenster des Simulators**  
Hiermit können Sie die Echtzeituhr (RTC) der Steuerung steuern, um das Verstreichen von Zeit und die entsprechende Wirkung auf die logischen Konstrukte zu simulieren, die von der Echtzeituhr betroffen sind.
- **E/A-Manager-Fenster des Simulators**  
Dient zur Verwaltung des Zustands von Eingängen/Ausgängen der Steuerung und der Erweiterungsmodule.

Nachdem eine Verbindung zwischen dem PC und der virtuellen Steuerung hergestellt wurde (siehe Verwendung des SoMachine Basic-Simulators (*siehe Seite 299*)), wird das Fenster für den SoMachine Basic-E/A-Simulatormanager auf dem Bildschirm angezeigt:



1

1 E/A-Manager-Fenster des Simulators (*siehe Seite 288*)

## E/A-Manager-Fenster des SoMachine Basic-Simulators

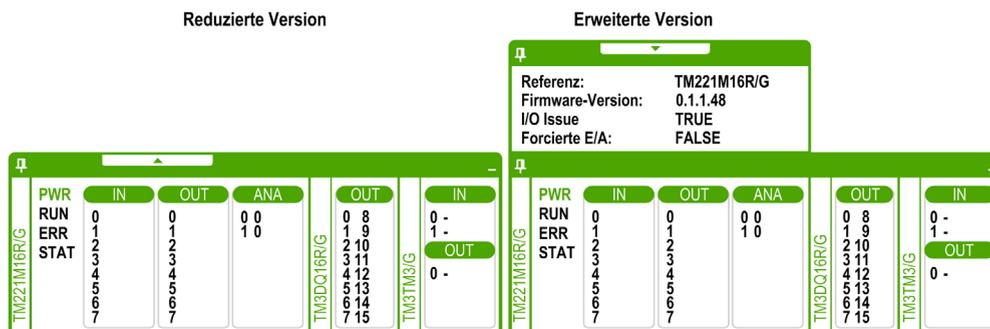
### Überblick

Das E/A-Manager-Fenster des Simulators umfasst folgende Komponenten:

- **LED-Status:**  
Zum Überwachen des LED-Zustands einer simulierten Steuerung.
- **Eingangs-/Ausgangsstatus:**  
Zum Steuern der Ein- und Ausgänge bei laufendem Programm.

### E/A-Manager-Fenster des Simulators

Die folgende Grafik zeigt das E/A-Manager-Fenster des Simulators:



Klicken Sie links oben im Fenster auf das Pinnadelsymbol, um das Fenster im Vordergrund zu fixieren oder die Fixierung aufzuheben.

Klicken Sie rechts oben im Fenster auf das Minimierungssymbol, um das Fenster auf die Taskleiste zu minimieren.

### LED-Status

Die LEDs PWR, RUN, ERR und STAT werden im E/A-Manager-Fenster des SoMachine Basic-Simulators so angezeigt, wie sie an einer angeschlossenen Basissteuerung erscheinen würden.

Die folgenden LED-Zustände werden im E/A-Manager-Fensters des Simulators für einen simulierten Logic Controller angezeigt:

| LED         | Status Informationen   |
|-------------|--|
| <b>PWR</b>  | Zeigt an, ob der simulierte Logic Controller eingeschaltet ist oder nicht. |
| <b>RUN</b>  | Zeigt an, dass sich die simulierte Logiksteuerung im RUN-Zustand befindet. |
| <b>ERR</b>  | Zeigt an, dass sich die simulierte Logiksteuerung im ERR-Zustand befindet. |
| <b>STAT</b> | Die Funktion der LED STAT wird von der Benutzerlogik definiert.            |

## Eingangs-/Ausgangsstatus

Über das E/A-Manager-Fenster des Simulators können Sie die E/A von Steuerungen und Erweiterungsmodulen bei laufendem Programm überwachen und steuern.

Die Ein- und Ausgänge werden in einer Nummernliste angezeigt. Diese Liste ist von den E/A der ausgewählten Steuerungen und Erweiterungsmodulen abhängig. Wenn die Steuerung beispielsweise  $n$  digitale Eingänge hat, beginnt die Nummernliste bei  $0 \dots (n-1)$ , wobei jede Nummer dem digitalen Eingang am entsprechenden Eingangskanal entspricht.

Für eine Steuerung werden die folgenden E/A angezeigt:

- **IN:** digitale Eingänge
- **OUT:** digitale Ausgänge
- **ANA:** analoge Eingänge

Für ein Erweiterungsmodul werden die folgenden E/A angezeigt:

- **IN:** digitale/analoge Eingänge
- **OUT:** digitale/analoge Ausgänge

**HINWEIS:** Die aktuellen Werte der analogen E/A werden jeweils rechts neben der Eingangsnummer angezeigt.

Der Status der digitalen E/A ist an der Textfarbe der E/A-Nummer zu erkennen:

- Grün: E/A ist auf 1 gesetzt
- Schwarz: E/A ist auf 0 gesetzt

Der Status der analogen E/A ist am Wert zu erkennen:

- - (Bindestrich): E/A ist nicht konfiguriert
- Zahl: aktueller Wert des E/A

## SoMachine Basic-Simulator – Fenster „Zeitverwaltung“

### Überblick

Das Simulator-Fenster **Zeitverwaltung** besteht aus folgenden Komponenten:

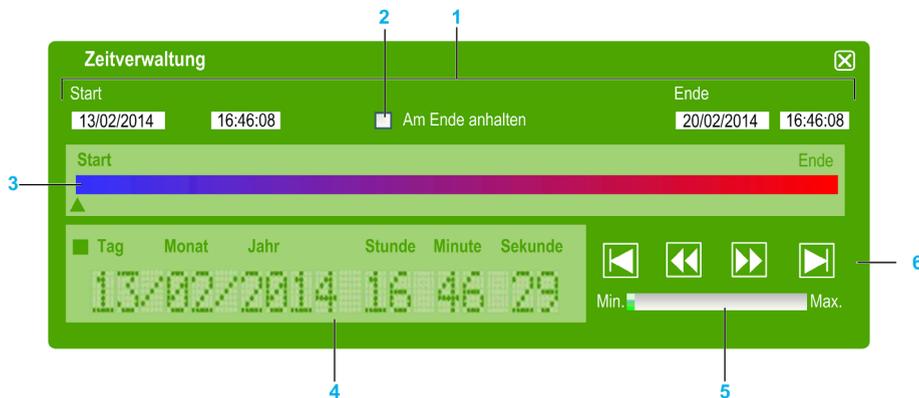
- Datums- und Uhrzeitbereich der Simulation für die Ausführung des Programms im Simulator:
  - **Start**-Datum und -Uhrzeit
  - **End**-Datum und -Uhrzeit
  - Kontrollkästchen **Am Ende anhalten** (stoppt die Ausführung des Programms, wenn das Datum und die Uhrzeit erreicht wurden, die unter **Ende** angegeben sind)
- Zeitsteuerungsleiste:
  - Um die Simulation des Zeitablaufs manuell vor- oder zurückzubewegen
- Anzeige von Datum und Uhrzeit:
  - Datum und Uhrzeit der simulierten Echtzeituhr des Simulators
- Steuerschaltflächen:
  - Zum Zurücksetzen, Vorspulen, Zurückspulen oder Beenden der mit der Echtzeituhr verknüpften Zeitverwaltung
- Inkrementierleiste:
  - Zum Festsetzen der Rate des simulierten Zeitablaufs relativ zur Echtzeit

### Fenster Zeitverwaltung des Simulators

Zum Anzeigen des Fensters **Zeitverwaltung**:

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie mit der rechten Maustaste auf die obere Leiste des Fensters <b>Simulator E/A-Management</b> . |
| 2       | Wählen Sie <b>Zeitverwaltung</b> aus.   |

Diese Abbildung zeigt das Fenster **Zeitverwaltung** des Simulators:



- 1 Datums-/Uhrzeitbereich der Simulation (Start – Ende)
- 2 Kontrollkästchen „Am Ende anhalten“ (des Datums-/Uhrzeitbereichs)
- 3 Zeitsteuerungsleiste
- 4 Datum und Uhrzeit der Echtzeituhr (RTC)
- 5 Inkrementierleiste
- 6 Steuerungsschaltflächen für verstrichene Zeit

### Datums-/Uhrzeitbereich der Simulation

Mithilfe des Simulationszeitbereichs können Sie die Echtzeituhr (RTC) des Simulators einstellen und steuern. Die Echtzeituhr wird mit den Feldern für **Start**-Datum und -Uhrzeit eingestellt, wenn Sie den Simulator in einen RUN-Zustand versetzen. Mit den Felder für **End**-Datum und -Uhrzeit wird das Ende der Simulation festgelegt. Wenn Sie das Kontrollkästchen **Am Ende anhalten** aktivieren, begibt sich der Simulator nach Ablauf des Simulationszeitbereichs in einen STOP-Zustand. Andernfalls laufen der Simulator und auch die Echtzeituhr solange weitern, bis Sie den Simulator manuell über SoMachine Basic anhalten.

### Zeitsteuerungsleiste

Mit der Zeitsteuerungsleiste können Sie Datum und Uhrzeit, die Sie im Simulationszeitbereich festgelegt haben, manuell manipulieren. Zeigen Sie bei gedrückter rechter Maustaste auf den Pfeil unterhalb der Leiste und bewegen Sie die Maus nach rechts, um Datum und Uhrzeit der Echtzeituhr vorzustellen. Wenn Sie genauso vorgehen, aber die Maus nach links bewegen, wird die Zeit der Echtzeituhr zurückgedreht.

### Datum und Uhrzeit der Echtzeituhr (RTC)

Im Bereich für Datum und Uhrzeit der Echtzeituhr wird der RTC-Wert in Bezug auf die stattfindende Simulation angezeigt. Die anfängliche Zeit der Echtzeituhr wird durch die Datums- und Uhrzeitwerte unter **Start** bestimmt, wenn Sie den Simulator in einen RUN-Zustand versetzen. Anschließend wird die Anzeige mit dem kontinuierlichen Takten der Echtzeituhr im Simulator aktualisiert. Sie können die Echtzeituhr entweder über die Zeitsteuerungsleiste oder mit den Schaltflächen zur Steuerung der Geschwindigkeit der verstrichenen Zeit verändern.

### Inkrementierleiste

Mit der Inkrementierleiste können Sie ein relatives Inkrement zum Vor- oder Zurückspulen des Echtzeituhrwerts bei Verwendung der Schaltflächen zur Steuerung der verstrichenen Zeit festlegen. Indem Sie auf die Leiste klicken, können Sie verschiedene Inkremente einstellen, die relativ zu dem von ihnen festgelegten Simulationszeitbereich sind.

### Steuerungsschaltflächen für verstrichene Zeit

Mithilfe dieser Schaltflächen können Sie den Echtzeitwert beeinflussen und somit seine Wirkung auf das im Simulator ausgeführte Programm wie folgt manipulieren:

| Grafikelement   | Befehl         | Beschreibung  |
|---|----------------|---|
|    | Initialisieren | Hiermit können Sie Datum und Uhrzeit wieder auf die Werte zurücksetzen, die im Datums- und Uhrzeitfeld unter <b>Start</b> eingestellt sind. |
|    | Vorspulen      | Hiermit können Sie Uhrzeit und Datum vom aktuellen Wert um die durch die Inkrementierleiste vorgegebenen Inkremente vorspulen.              |
|  | Zurückspulen   | Hiermit können Sie Uhrzeit und Datum vom aktuellen Wert um die durch die Inkrementierleiste vorgegebenen Inkremente zurückspulen.           |
|  | Ende           | Hiermit können Sie Datum und Uhrzeit auf die Werte vorspulen, die im Datums- und Uhrzeitfeld unter <b>Ende</b> eingestellt sind.            |

## Ändern von Werten mit dem SoMachine Basic-Simulator

### Überblick

Im Online-Modus bietet das E/A-Manager-Fenster des SoMachine Basic-Simulators folgende Möglichkeiten:

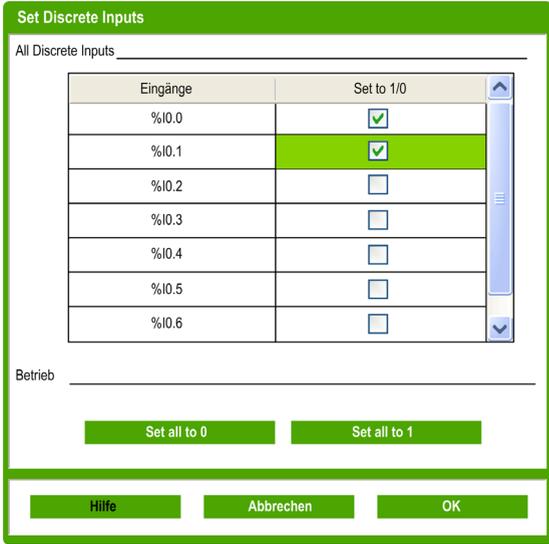
- Ändern der Werte von Eingängen
- Verfolgen von Ausgängen

### Ändern von Werten digitaler Eingänge

Gehen Sie wie folgt vor, um den Wert digitaler Eingänge per einfachen Mausklick zu ändern:

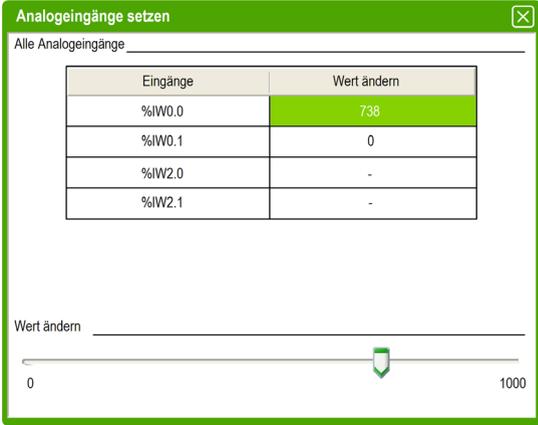
| Schritt | Aktion   |
|---------|--|
| 1       | <p>Klicken Sie im E/A-Manager-Fenster des Simulators auf die Nummer des gewünschten digitalen Eingangs, um dessen Wert zu ändern.</p> <p><b>Ergebnis:</b> Die Textfarbe der Eingangsnummer ändert sich. Die Werte digitaler Eingänge sind an der Textfarbe zu erkennen:</p> <ul style="list-style-type: none"><li>• Grün: E/A ist auf 1 gesetzt</li><li>• Schwarz: E/A ist auf 0 gesetzt</li></ul> |
| 2       | <p>Klicken Sie noch einmal auf dieselbe Eingangsnummer, um den Wert umzuschalten.</p>  |

Gehen Sie wie folgt vor, um die Werte mehrerer digitaler Eingänge gleichzeitig zu ändern:

| Schritt | Aktion   |
|---------|--|
| 1       | <p>Doppelklicken Sie im E/A-Manager-Fenster des Simulators auf die Nummer des gewünschten digitalen Eingangs.<br/> <b>Ergebnis:</b> Auf dem Bildschirm wird das Fenster <b>Digitaleingänge setzen</b> angezeigt, in dem alle digitalen Eingänge aufgelistet sind:</p>   |
| 2       | <p>Klicken Sie im Bereich <b>Operation</b> des Fensters <b>Digitaleingänge setzen</b> auf:</p> <ul style="list-style-type: none"> <li>• <b>Alle auf 0 setzen</b>, um den Wert aller Eingänge auf 0 zu setzen.</li> <li>• <b>Alle auf 1 setzen</b>, um den Wert aller Eingänge auf 1 zu setzen.</li> </ul> <p><b>Ergebnis:</b> Wenn das Kontrollkästchen aktiviert wird, wird der Eingangswert auf 1 gesetzt. Wird das Kontrollkästchen nicht aktiviert, wird der Eingangswert auf 0 gesetzt.</p> |
| 3       | <p>Alternativ dazu können Sie im Bereich <b>Alle Digitaleingänge</b> des Fensters <b>Digitaleingänge setzen</b> auf das Kontrollkästchen der gewünschten Eingänge klicken, um die Werte einzeln zu ändern.</p>   |
| 4       | <p>Klicken Sie auf <b>OK</b>, um die Änderungen zu speichern und das Fenster <b>Digitaleingänge setzen</b> zu schließen.</p>   |

## Ändern von Werten analoger Eingänge

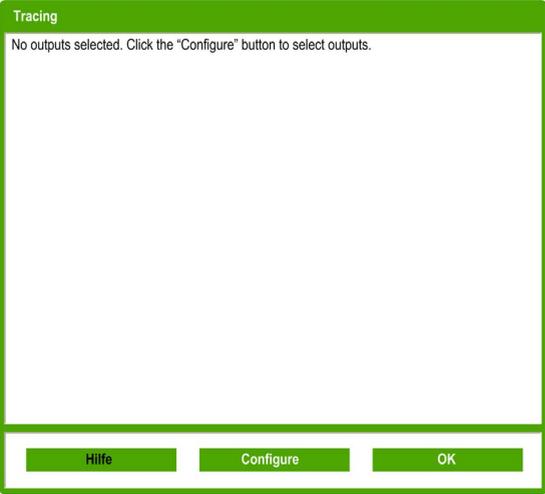
Gehen Sie wie folgt vor, um die Werte analoger Eingänge zu ändern:

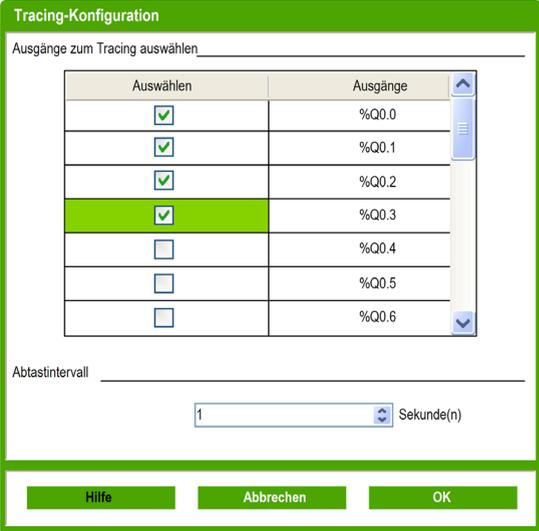
| Schritt | Aktion   |
|---------|--|
| 1       | <p>Doppelklicken Sie im E/A-Manager-Fenster des Simulators auf die Nummer des gewünschten analogen Eingangs.<br/> <b>Ergebnis:</b> Auf dem Bildschirm wird das Fenster <b>Analogeingänge setzen</b> angezeigt, in dem alle analogen Eingänge aufgelistet sind:</p>  |
| 2       | Doppelklicken Sie im Bereich <b>Alle Analogeingänge</b> des Fensters <b>Analogeingänge setzen</b> auf das Wertfeld in der Spalte <b>Wert ändern</b> des Eingangs, der geändert werden soll.  |
| 3       | Geben Sie einen Wert zwischen 0 und 1023 ein und drücken Sie die Eingabetaste.   |
| 4       | Alternativ dazu können Sie im Fenster <b>Analogeingänge setzen</b> einen Eingang aus der Liste <b>Eingänge</b> wählen und den Schieberegler im Bereich <b>Wert ändern</b> auf einen Wert zwischen 0 und 1023 einstellen.<br>Wenn Sie den Schieberegler von links nach rechts bewegen, nimmt der Wert zu und umgekehrt.                               |
| 5       | Klicken Sie auf <b>OK</b> , um die Änderungen zu speichern und das Fenster <b>Analogeingänge setzen</b> zu schließen.  |

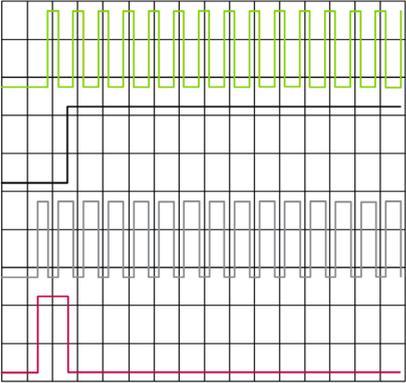
### Verfolgen von Ausgängen

Da Ausgangswerte programmabhängig sind, können Sie sie nicht ändern. Der SoMachine Basic-Simulator bietet Ihnen jedoch die Möglichkeit digitale und analoge Ausgänge zu verfolgen.

Gehen Sie wie folgt vor, um die Werte analoger Eingänge zu ändern:

| Schritt | Aktion   |
|---------|--|
| 1       | <p>Doppelklicken Sie im E/A-Manager-Fenster des Simulators auf die Ausgangsnummer.<br/> <b>Ergebnis:</b> Das Fenster <b>Tracing</b> wird auf dem Bildschirm angezeigt.</p>  |

| Schritt | Aktion   |
|---------|--|
| 2       | <p>Klicken Sie auf die Schaltfläche <b>Konfigurieren</b>, um die zu verfolgenden Ausgänge auszuwählen.<br/> <b>Ergebnis:</b> Das Fenster <b>Tracing-Konfiguration</b> wird auf dem Bildschirm angezeigt.</p>  |
| 3       | <p>Klicken Sie in der Kontrollkästchenspalte <b>Auswählen</b> auf die Kontrollkästchen der Ausgänge, die verfolgt werden sollen.</p>   |
| 4       | <p>Treffen Sie im Dropdown-Menü <b>Abtastintervall</b> eine Auswahl, um das Abtastintervall für die Ausgangsverfolgung festzulegen:</p> <ul style="list-style-type: none"> <li>● 1 Sekunde</li> <li>● 5 Sekunden</li> <li>● 10 Sekunden</li> <li>● 20 Sekunden</li> </ul>                      |

| Schritt | Aktion  |
|---------|---|
| 5       | <p>Klicken Sie auf <b>OK</b>, um die Einstellungen zu speichern und das Fenster <b>Tracing-Konfiguration</b> zu schließen.</p> <p><b>Ergebnis:</b> Die ausgewählten Ausgänge werden dem Fenster <b>Tracing</b> hinzugefügt, das die Verfolgung der Ausgänge mit simulierten Werten anzeigt:</p> <div data-bbox="304 315 865 867" style="border: 2px solid green; padding: 5px;"> <p style="text-align: center; background-color: #008000; color: white; margin: 0;">Tracing</p> <p>Digitale Ausgänge</p>  <p style="font-family: monospace; margin: 5px 0;"> <span style="color: green;">"&gt;%Q0.0: 1</span><br/> <span style="color: green;">"&gt;%Q0.1: 1</span><br/> <span style="color: gray;">"&gt;%Q0.2: 0</span><br/> <span style="color: red;">"&gt;%Q0.3: 0</span> </p> <p style="text-align: center; margin: 0;"> <span style="background-color: #008000; color: white; padding: 2px 10px; margin-right: 5px;">Hilfe</span> <span style="background-color: #008000; color: white; padding: 2px 10px; margin-right: 5px;">Configure</span> <span style="background-color: #008000; color: white; padding: 2px 10px;">OK</span> </p> </div> |
| 6       | <p>Klicken Sie auf <b>OK</b>, um das Fenster <b>Tracing</b> zu schließen.</p>   |

## Verwendung des SoMachine Basic-Simulators

### Vorgehensweise

Gehen Sie wie folgt vor, um den SoMachine Basic-Simulator zum Testen Ihres Programms auszuführen:

| Schritt | Aktion  |
|---------|---|
| 1       | Vergewissern Sie sich, dass Sie über ein gültiges Programm verfügen, indem Sie die Statusmeldung im Statusbereich überprüfen (weitere Informationen finden Sie unter Statusbereich ( <i>siehe Seite 57</i> )). Der Status des Programms sollte <b>Keine Fehler</b> lauten. Sie können den SoMachine Basic-Simulator auch dann ausführen, wenn der Programmstatus <b> Tipp</b> lautet. |
| 2       | Weitere Informationen finden Sie unter Aufrufen des Simulators ( <i>siehe Seite 286</i> ).  |
| 3       | Setzen Sie die Steuerung in Betrieb. Wählen Sie im Fenster <b>Inbetriebnahme</b> die Option <b>Verbinden</b> in der Inbetriebnahmeübersicht aus. Klicken Sie anschließend im Taskbereich für die Inbetriebnahme auf die Schaltfläche <b>Steuerung in Betrieb setzen</b> .   |
| 4       | Steuern Sie Ihr Programm mithilfe des Simulator-Hauptfensters (siehe Steuerschaltflächen ( <i>siehe Seite 291</i> )).   |
| 5       | Überprüfen Sie den LED-Status im Simulator-Hauptfenster (siehe LED-Anzeige ( <i>siehe Seite 289</i> )).   |
| 6       | Überprüfen Sie den Status der Ein- und Ausgänge im Simulator-Fenster zur E/A-Verwaltung (siehe Eingangs-/Ausgangsstatus ( <i>siehe Seite 289</i> )).  |
| 7       | Überprüfen Sie den LED-Status im Simulator-Fenster zur E/A-Verwaltung (siehe LED-Status ( <i>siehe Seite 288</i> )).  |
| 8       | Ändern Sie die E/A-Werte nach Bedarf (siehe Ändern von Werten mit dem Simulator ( <i>siehe Seite 293</i> )).  |
| 9       | Verfolgen Sie die Ausgänge nach Bedarf (siehe Verfolgen der Ausgänge ( <i>siehe Seite 296</i> )).   |
| 10      | Stoppen Sie die Steuerung. Wählen Sie im Fenster <b>Inbetriebnahme</b> die Option <b>Verbinden</b> in der Inbetriebnahmeübersicht aus. Klicken Sie anschließend im Taskbereich für die Inbetriebnahme auf die Schaltfläche <b>Steuerung stoppen</b> .   |
| 11      | Stoppen Sie den Simulator. Wählen Sie im Fenster <b>Inbetriebnahme</b> die Option <b>Verbinden</b> in der Inbetriebnahmeübersicht aus. Klicken Sie anschließend im Taskbereich für die Inbetriebnahme auf die Schaltfläche <b>Steuerung stoppen</b> oder drücken Sie die Tastenkombination STRG+W, um den Simulator zu beenden.   |

## Starten der Simulation in Vijeo-Designer

### Vorgehensweise

Bevor Sie die HMI-Simulation in Vijeo-Designer starten, müssen Sie zuerst den Simulator der Logiksteuerung in SoMachine Basic (*siehe Seite 286*) starten.

Gehen Sie wie folgt vor, um die Simulation in Vijeo-Designer zu starten:

| Schritt | Aktion   |
|---------|--|
| 1       | Starten Sie Vijeo-Designer.  |
| 2       | Öffnen Sie das Vijeo-Designer-Projekt, das die Symbole aus einem SoMachine Basic-Projekt enthält.<br><b>HINWEIS:</b> Wenn das Vijeo-Designer-Projekt nicht vorhanden ist, erstellen Sie in Vijeo-Designer ein Projekt, und geben Sie die Symbole für das SoMachine Basic-Projekt frei. Weitere Informationen finden Sie unter Freigeben von Symbolen zwischen einem SoMachine Basic-Projekt und einem Vijeo-Designer-Projekt ( <i>siehe Seite 181</i> ).       |
| 3       | Klicken Sie auf die Registerkarte <b>Projekt</b> im Fenster <b>Navigator</b> , klicken Sie mit der rechten Maustaste auf den Geräteknoten unter dem Knoten <b>E/A-Manager</b> , und wählen Sie <b>Konfiguration</b> aus.<br><b>Ergebnis:</b> Das Fenster <b>Gerätekonfiguration</b> wird geöffnet.   |
| 4       | Geben Sie die <b>IP-Adresse</b> ein, und klicken Sie auf <b>OK</b> .<br><b>HINWEIS:</b> Die IP-Adresse muss eine lokale Hostadresse oder eine lokale Adresse Ihres PCs sein. Beispiel: 127.0.0.1   |
| 5       | Starten Sie das <b>Geräte-Simulationstool</b> .  |
| 6       | Klicken Sie auf die Registerkarte <b>Variablen</b> , und aktivieren Sie die Kontrollkästchen der Variablen, die Teil der Simulation sein sollen.<br><b>HINWEIS:</b> Wenn das Symbol <b>Alles anzeigen</b> ausgewählt ist, werden alle auf der Registerkarte <b>Variablen</b> ausgewählten Variablen auf der Registerkarte <b>Simulation</b> angezeigt.   |
| 7       | Klicken Sie auf die Registerkarte <b>Simulation</b> .  |
| 8       | Wählen Sie eine Variable aus, wählen Sie eine Operation für die Variable aus, und aktivieren Sie dann das Kontrollkästchen <b>Aktiv</b> .<br><b>HINWEIS:</b> Es kann immer nur eine Simulationsoperation auf eine Variable angewendet werden.  |
| 9       | Definieren Sie die Parameter der Variablensimulationsoperation.  |
| 10      | Klicken Sie auf das Symbol <b>Simulation</b> , um die Simulation zu starten.   |
| 11      | Ändern Sie die Variablenwerte, wie während der Simulation erforderlich:<br><ul style="list-style-type: none"> <li>● Bei Schieberegleroperationen können Sie den Wert verändern, indem Sie den Schieber bewegen, am Rad Ihrer Maus drehen oder die Pfeiltasten auf der Tastatur verwenden.</li> <li>● Bei Wechseloperationen klicken Sie auf <b>Setzen</b> oder <b>Zurücksetzen</b>, um die entsprechende Zeichenfolge in die Variable zu schreiben.</li> </ul> |
| 12      | Klicken Sie erneut auf das Symbol <b>Simulation</b> , um die Simulation zu starten.  |
| 13      | Drücken Sie <b>Strg+Z</b> , um das <b>Geräte-Simulationstool</b> zu beenden.   |

---

# Kapitel 9

## Speichern von Projekten und Schließen von SoMachine Basic

---

### Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

| Thema                                  | Seite |
|--|-------|
| Speichern eines Projekts               | 302   |
| Speichern der Projektdatei als Vorlage | 303   |
| Schließen von SoMachine Basic          | 304   |

## Speichern eines Projekts

### Übersicht

SoMachine Basic-Projekte könne als Dateien auf dem lokalen PC gespeichert werden. Die Datei weist die Erweiterung \*.smbp auf und enthält folgende Daten:

- Quellcode des Programms aus der Registerkarte **Programmierung**
- Aktuelle Hardwarekonfiguration aus der Registerkarte **Konfiguration**
- Im SoMachine Basic-Projekte definierte Parameter und Voreinstellungen

### Speichern eines Projekts

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf <b>Speichern</b>  in der Symbolleiste oder drücken Sie die Tastenkombination <b>Ctrl-S</b> |
| 2       | Wenn Sie das Projekt zum ersten Mal speichern, wählen Sie den Ordner aus, in dem die Projektdatei gespeichert werden soll.  |
| 3       | Geben Sie einen Namen für die Projektdatei ein und klicken Sie dann auf <b>Speichern</b> .  |

### Speichern eines Projekts unter einem anderen Namen

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf den Menüfeil neben der Schaltfläche <b>Speichern</b>  in der Symbolleiste und wählen Sie <b>Speichern unter</b> aus. |
| 2       | Wählen Sie den Ordner aus, in dem die Projektdatei gespeichert werden soll.   |
| 3       | Geben Sie einen neuen Namen für die Projektdatei ein und klicken Sie auf <b>Speichern</b> .   |

## Speichern der Projektdatei als Vorlage

### Übersicht

SoMachine Basic-Projekte können als Vorlage gespeichert werden. Das Projekt wird dann auf der Registerkarte **Vorlagen** der Startseite (*siehe Seite 48*) aufgeführt. Sie können das Projekt dann als Ausgangspunkt für neue Projekt heranziehen.

### Speichern eines Projekts als Vorlage

| Schritt | Aktion  |
|---------|---|
| 1       | Klicken Sie auf den Menüfeil neben der Schaltfläche <b>Speichern</b>  in der Symbolleiste und wählen Sie <b>Als Vorlage speichern</b> aus.<br>Vorlagen werden standardmäßig im folgenden Ordner abgelegt: C:\Users\Public\SoMachine Basic\Examples. |
| 2       | Geben Sie den Namen des Projekts ein.   |
| 3       | Wählen Sie <b>Beispiel-Projektdateien (*.smbe)</b> als <b>Dateityp</b> aus und klicken Sie auf <b>Speichern</b> .   |

## Schließen von SoMachine Basic

### Übersicht

Um SoMachine Basic zu verlassen, klicken Sie auf die Schaltfläche **Schließen** in der oberen rechten Ecke des SoMachine Basic-Fensters.

Sie können ebenfalls auf die Schaltfläche **Beenden** auf der **Startseite** klicken.

---

# Anhang

---



## Inhalt dieses Anhangs

Dieser Anhang enthält die folgenden Kapitel:

| Kapitel | Kapitelname   | Seite |
|---------|---|-------|
| A       | Konvertieren von Twido-Projekten zu SoMachine Basic | 307   |
| B       | SoMachine Basic-Tastaturkombinationen               | 319   |



---

# Anhang A

## Konvertieren von Twido-Projekten zu SoMachine Basic

---

### Konvertieren von Twido-Projekten zu SoMachine Basic

#### Überblick

Wenn Sie ein TwidoSoft- oder TwidoSuite-Projekt öffnen (*siehe Seite 39*), wird dieses automatisch in ein SoMachine Basic-Projekt konvertiert. Es wird ein Konvertierungsbericht erstellt, der alle Fälle auflistet, in denen ein Aspekt eines TwidoSoft- oder TwidoSuite-Projekts nicht automatisch in die entsprechende SoMachine Basic-Funktion umgewandelt werden konnte.

Im Folgenden werden zusätzliche Informationen zur Konvertierung bereitgestellt.

### **WARNUNG**

#### **UNBEABSICHTIGTER GERÄTEBETRIEB**

- Vergewissern Sie sich immer, dass Ihr Anwendungsprogramm weiterhin so funktioniert wie vor der Konvertierung, mit den richtigen Konfigurationen, Parametern, Parameterwerten, Funktionen und Funktionsbausteinen, die erforderlich sind.
- Ändern Sie die Anwendung, so dass es dem vorherigen Betrieb entspricht.
- Testen Sie Ihre neu kompilierte Version sorgfältig und validieren Sie sie, bevor Sie die Anwendung in Betrieb nehmen.

**Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.**

### Twido-Programmtypen, die eine manuelle Anpassung benötigen

Diese Tabelle listet Twido-Projekttypen auf, die Funktionen verwenden, welche keine direkte Entsprechung in M221 Logic Controller haben und gibt Empfehlungen zur Konvertierung dieser Projekte für SoMachine Basic:

| Twido-Programmtyp   | Lösung   | Beschreibung   |
|---|--|--|
| Programm, das CANopen verwendet   | Das Programm kann zur Nutzung des Ethernet-Netzwerkes konvertiert werden.  | Siehe die Vorlagen M221_with_LXM32_Modbus_TCP und M221_with_ILx2T_Modbus_TCP (Durchsuchen Sie das Vorlagenfenster ( <i>siehe Seite 48</i> )).  |
| Programm, das Twido Makro-Kommunikation verwendet.                                      | Der Twido-Code wird automatisch zur Verwendung von EXCH-Anweisungen konvertiert. Das Programm kann zur Verwendung von Kommunikationsfunktionsbausteinen ( <i>siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen</i> ) angepasst werden.  | Siehe die xSample_twido_macro_COMM_Conversion Projektvorlage und entsprechende Dokumentation (Durchsuchen Sie das Vorlagenfenster ( <i>siehe Seite 48</i> )) für Hilfe bei der Anpassung eines konvertierten Programms zur Verwendung von Kommunikationsfunktionsbausteinen. |
| Programm, das Twido Antriebs-Makros verwendet.  | Teile des Twido-Codes können nicht automatisch in den Kontaktplan-Sprachcode konvertiert werden.   | Siehe die Projektvorlagen xSample_ATV Modbus_SL_M221 oder xSample_Twido_Macro_Drive_Conversio für Hilfe bei der Anpassung der Antriebsmanagementfunktion.  |
| Twido Extreme TWDLEDCK1-Projekt   | Dieser Projekttyp kann nicht automatisch umgewandelt werden.   | Zum Abrufen eines Teils des Programms: <ul style="list-style-type: none"> <li>• Ändern Sie die Steuerung im TwidoSuite-Programm von TWDLEDCK1 zu einer anderen Twido-Steuerung</li> <li>• Konvertieren Sie das aktualisierte Projekt</li> </ul>                              |
| Programm, das ein Protokoll zur dezentralen Verbindung (Remote Link Protocol) verwendet | Sie können das Programm zur Nutzung der folgenden M221 Logic Controller-Funktionen anpassen: <ul style="list-style-type: none"> <li>• Modbus TCP-Zuordnung auf dem Ethernet-Netzwerk</li> <li>• Serielles Modbus Protokoll zur Verwendung von Kommunikationsfunktionsbausteinen (<i>siehe SoMachine Basic, – Bibliothekshandbuch zu Generischen Funktionen</i>)</li> </ul> | Das Remote Linke-Protokoll ermöglicht die Verwendung einer Twido-Steuerung als dezentrales E/A-Modul auf einer seriellen Leitung.  |

### Im Konvertierungsbericht aufgelistete Meldungen

Die folgende Tabelle enthält zusätzliche Informationen für spezifische Meldungs-IDs, die im Konvertierungsbericht gemeldet werden:

| Meldungs-ID             | Meldung  | Beschreibung/Lösung  |
|-------------------------|--|--|
| <b>Fehlermeldungen</b>  |  |  |
| TC-001                  | Das Twido-Projekt kann nicht geladen werden  | Die Twido-Projektdatei konnte nicht in SoMachine Basic geöffnet werden.  |
| TC-002                  | Der Ordner mit den Twido-Informationen (mit gleichem Name und Speicherort wie die .xpr-Datei) wurde nicht gefunden | Der angegebene Ordner konnte nicht gefunden werden.  |
| TC-003                  | Die Twido-Datei <filename> weist nicht das richtige Format auf   | Das Twido-Projekt weist nicht das richtige Format auf. Es wird keine Konvertierung vorgenommen.  |
| TC-004                  | Die Twido-Datei <filename> weist ein unerwartetes Format auf   | Das Twido-Projekt ist unvollständig. Es wird keine Konvertierung vorgenommen.  |
| TC-005                  | Gerät <device> wird nicht unterstützt  | Die Twido-Referenz <device> wird nicht unterstützt. Es wird keine Konvertierung vorgenommen.   |
| TC-006                  | Das CANopen-Makro wurde nicht in AWL übersetzt   | Da M221 Logic Controller CANopen nicht unterstützt, werden auch Twido CANopen-Makros nicht unterstützt.  |
| <b>Hinweismeldungen</b> |  |  |
| TC-101                  | Das physische Medium der seriellen Leitung 2 (SL2) wurde zu RS485 geändert   | Auf Referenzen vom Typ TM221M**** kann die serielle Leitung 2 nicht in RS232 konfiguriert werden.<br>Sie können stattdessen Ihr externes Gerät in RS485 konfigurieren. Andernfalls können Sie auch einen externen RS232/RS485-Adapter hinzufügen, die Logiksteuerung mit einer TM221C****-Referenz ersetzen oder ein TMC2-Steckmodul hinzufügen, das RS232 auf der Steuerung unterstützt.  |
| TC-102                  | Die 'Remote Link'-Konfiguration der seriellen Leitung wurde durch das Modbus-Protokoll ersetzt.                    | Das 'Remote Link'-Protokoll wird auf M221 Logic Controller nicht unterstützt.<br>Andere Lösungsmöglichkeiten sind unter anderem die Verwendung von Kommunikationsfunktionsbausteinen auf Modbus oder eine Modbus-Zuordnungstabelle, wenn ein M221 Logic Controller verwendet wird, das Ethernet hat. Siehe außerdem die Informationen unter: Twido-Programmtypen, die eine manuelle Anpassung benötigen ( <i>siehe Seite 308</i> ) |

| Meldungs-ID | Meldung   | Beschreibung/Lösung   |
|-------------|---|---|
| TC-103      | TWDXCPODC-Erweiterung wird in SoMachine Basic nicht unterstützt. Es wurde nicht importiert.   | TWDXCPODC Ist ein Erweiterungsmodul für eine Anzeige, die in SoMachine Basic nicht unterstützt wird.<br>Im Falle von M221 Logic Controller können Sie TMH2GDB Dezentrale Grafikanzeige: Remote Graphic Display verwenden, was eine Benutzerschnittstellenanwendung zur Verfügung stellt.  |
| TC-104      | TWDXCPODM-Erweiterung wird in SoMachine Basic nicht unterstützt. Es wurde nicht importiert.   | TWDXCPODM Ist ein Erweiterungsmodul für eine Anzeige, die in SoMachine Basic nicht unterstützt wird.<br>Im Falle von M221 Logic Controller können Sie TMH2GDB Dezentrale Grafikanzeige: Remote Graphic Display verwenden, was eine Benutzerschnittstellenanwendung zur Verfügung stellt.  |
| TC-105      | Die neue Logiksteuerung <reference> bietet keine Unterstützung für die Funktionen Impulsgenerator (PLS) und Impulsbreitenmodulation (PWM) | Die Twido Compact Base 40 E/A, 240 VAC-Steuerung hat zwei schnelle Transistorausgänge. Im M221 Logic Controller-Bereich haben nur die 24 VDC-betriebenen Steuerungen Transistorausgänge. Die VAC-betriebenen M221-Steuerungen haben nur Relaisausgänge. Wählen Sie beim Ersatz von Steuerungen eine M221 Logic Controller mit einer 24-VDC-Spannungsversorgung. |
| TC-106      | Eine CANopen-Kommunikationserweiterung wird in SoMachine Basic nicht unterstützt. Die Erweiterung wurde nicht importiert.                 | M221 Logic Controller unterstützt CANopen nicht. Wenn Sie CANopen benötigen, sollten Sie Modicon M241 Logic Controller verwenden. Andernfalls können Sie den Kommunikationsbus auf einer seriellen Leitung mit Modbus bzw. auf Ethernet mit Modbus TCP ersetzen.  |
| TC-107      | Die AS-Schnittstellen-Master-Erweiterung wird in SoMachine Basic nicht unterstützt. Es wurde nicht importiert.                            | M221 Logic Controller Stellt kein AS-Schnittstellen-Mastermodul zur Verfügung. Sie können auch ein Ethernet-AS-Schnittstellen-Gateway-Protokoll, ein Modbus-Serial-Protokoll mit einem dezentralen E/A, ein Modbus-TCP-Protokoll oder ein Ethernet/IP-Protokoll verwenden.  |
| TC-108      | Die TM200-HSC--Erweiterung wird in SoMachine Basic nicht unterstützt. Es wurde nicht importiert.  | M221 Logic Controller hat vier schnelle Eingänge, die mit Hochgeschwindigkeitszählern verknüpft werden können.  |

| Meldungs-ID | Meldung   | Beschreibung/Lösung   |
|-------------|---|---|
| TC-109      | Die TWD-PTO-Erweiterung wird in SoMachine Basic nicht unterstützt. Es wurde nicht importiert.   | Die M221 Logic Controller-Referenzen ohne Relaisausgänge haben 2 oder 4 schnelle Ausgänge, die mit Impulswellenausgängen verknüpft werden können.   |
| TC-110      | Die TM2-VCM-Erweiterung wird in SoMachine Basic nicht unterstützt. Es wurde nicht importiert.   | Die TM2-VCM-Erweiterungsmodule werden in SoMachine Basic nicht unterstützt.   |
| TC-111      | Der „anpassbare“ Parameter von Timer 3 wird in SoMachine Basic nicht unterstützt. Es wird auf „True“ forciert.  | In den SoMachine Basic-Funktionsbausteinen wird dieser Parameter nicht unterstützt.   |
| TC-112      | Die %QAS-ASi-Ausgänge werden in SoMachine Basic nicht unterstützt.  | Diese Adressen sind für die Verwaltung der dezentralen AS-Schnittstellen-E/As reserviert. Diese Adressen werden wie das AS-Schnittstellen-Master-Modul auf M221 Logic Controller nicht unterstützt. |
| TC-113      | Der Autotune auf PID ist geändert: der neue Parameter ‚AT Trigger‘ der Registerkarte PID Autotune (AT) wurde hinzugefügt und konfiguriert, und der Parameter ‚Output Setpoint‘ wurde ignoriert. | Konfigurieren Sie PID Autotuning in SoMachine Basic.  |
| TC-114      | Der von HSCn (in Twido: VFCn)) verwendete Eingang ist von <input1> zu <input2> geändert worden.   | Stellen Sie sicher, dass Ihr Programm, den zugewiesenen Eingang verwendet.  |
| TC-115      | Die von HSCn verwendeten Eingänge <input1> und <input2> werden relativ zu Twido VFCn umgekehrt.   | Die HSC-Eingänge <input1> und <input2> werden auf Twido VFC-Steuerungen in SoMachine Basic umgekehrt; machen Sie diese Umkehrung in der Anwendung rückgängig.                                       |
| TC-116      | Die Freie POU <x> ist <y> bereits zugewiesen. Das Ereignis <z> kann diese Freie POU nicht verwenden.  | Weisen Sie die Freie POU einem anderen Ereignis zu.   |
| TC-117      | Das Twido-Objekt <x> wurde zu <y> auf neuen Steuerungen verschoben. Sie müssen das Programm aktualisieren, um die Konsistenz zu erhalten.   | Das Objekt wurde zu einem SoMachine Basic-Objekt mit ähnlicher Funktionalität umgewandelt. <y> kann ein Systembit, Systemwort oder ein anderer Objekttyp wie %IWS sein.                             |
| TC-118      | Das Twido-Objekt <x> wurde auf neuen Steuerungen geändert. Sie müssen prüfen, ob Ihre Steuerung noch konsistent ist.  | Das Objekt wurde konvertiert, seine Funktionalität in SoMachine Basic kann jedoch anders sein. Für Hilfe bei der Aktualisierung Ihres Programms, siehe die Online Hilfe.                            |
| TC-119      | Das Twido-Objekt <x> wird auf neuen Steuerungen nicht mehr unterstützt. Sie müssen das Programm mit neuen Funktionen aktualisieren.   | Das Objekt hat keine Entsprechung in SoMachine Basic. Für Hilfe bei der Aktualisierung Ihres Programms, siehe die Online Hilfe.   |

| Meldungs-ID | Meldung   | Beschreibung/Lösung   |
|-------------|---|---|
| TC-120      | Die Quellensteuerung wird mit 24 VDC gespeist, die Zielsteuerung <reference> jedoch mit 100 bis 240 VAC.  | Die konvertierte M221 Logic Controller hat nicht dieselbe Spannungsversorgung, was aber keinen Effekt auf die Anwendung hat.  |
| TC-121      | Die Quellensteuerung <reference1> mit Transistor- und Relaisausgängen wurde in <reference2> mit lediglich Transistorausgängen konvertiert.  | Die konvertierte M221 Logic Controller hat nicht dieselben Ausgangstypen. Die Konvertierung ermöglicht es, dass die Anwendung unverändert bleibt.   |
| TC-122      | Die Syntax des Symbols <x>, das mit <y> verknüpft ist, ist ungültig.  | Berichtigen Sie die Syntax des spezifischen Symbols.  |
| TC-123      | Das Symbol <x> im Zusammenhang mit <y> ist ein reserviertes Wort und wurde in <z> konvertiert.  | SoMachine Basic hat im Vergleich mit TwidoSuite/TwudiSoft neue Anweisungen. Reservierte Worte werden zu <z> konvertiert.  |
| TC-124      | Die Zeitbasis <w>, die in <x> konfiguriert ist, wurde zu <y> konvertiert. Sie müssen möglicherweise das 'Preset' in der Konfiguration und in <w> anpassen.<z> in der Applikation. | Diese Meldung erscheint, wenn eine Applikation mithilfe von PLS oder PWM konvertiert wurde. Auf Twido sind die Hardware-Zeitbasen 0,412 ms und 0,57 ms. Auf M221 Logic Controller sind die Hardware-Zeitbasen 0,1 ms bzw. 1 ms. Bei PLS- und PWM-Funktionsbausteinen entspricht die Dauer des generierten Signals der Zeitbasis multipliziert mit dem Preset-Wert (PLS . P, PLS . PD oder PWM . P). Preset-Werte (. P oder . PD) müssen möglicherweise in der Konfiguration und im Programm angepasst werden. |
| TC-125      | Das in <y> konfigurierte <x> wurde zu <z> konvertiert.  | HSC wird in M221 Logic Controller im Modus ‚Abwärtszählen‘ nicht unterstützt. Diese Konfigurationen werden in SoMachine Basic zu einem „einfachen Zählmodus“ (also Aufwärtszählen) konvertiert.   |
| TC-126      | Die Schwellenwerte von <x> wurden geändert, da sie nicht gleich sein dürfen.  | Es ist in SoMachine Basic nicht möglich, identische Schwellenwerte zu haben. Wenn die Twido-Applikation keine Schwellenwerte verwendet (kein verknüpftes Ereignis oder konfigurierter Reflex), werden die Werte so geändert, dass Konfigurationsfehler vermieden werden.  |
| TC-127      | Die Schwellenwerte für <x> sind gleich und werden einen Konfigurationsfehler auslösen.  | Es ist in SoMachine Basic nicht möglich, identische Schwellenwerte zu haben. Wenn die Twido-Applikation Schwellenwerte verwendet, wird nichts geändert und ein Konfigurationsfehler wird ausgelöst. Passen Sie die Applikation an, um den Fehler zu berichtigen.  |

| Meldungs-ID                  | Meldung   | Beschreibung/Lösung  |
|------------------------------|---|--|
| TC-128                       | <x> ist sowohl als Run/Stop als auch als Ereignis-Trigger im Twido-Project konfiguriert, was zu einem Konflikt in SoMachine Basic führte; die Konfiguration von Run/Stop-Funktionalität wurde aufgehoben. | Es ist in SoMachine Basic nicht möglich, dass derselbe Eingang gleichzeitig in 2 verschiedenen Funktionen konfiguriert ist.  |
| TC-129                       | Es wurde ein Ethernet-Modul auf einer Twido-Referenz mit integriertem Ethernet-Port gefunden. Die Konfiguration des Ethernet-Moduls wird ignoriert.   | Es ist in SoMachine Basic nicht möglich, zwei Ethernet-Verbindungen zu haben.  |
| TC-130                       | Ein Twido Makro kann nicht von einem Unterprogramm aufgerufen werden. Das Makro, das von SRn-Programmbaustein <x> aufgerufen wurde, ist nicht konvertiert worden.   | Es ist in SoMachine Basic nicht möglich, ein Makro aus einem Unterprogramm aufzurufen.   |
| TC-131                       | Es können nicht alle Ereignisprioritäten konvertiert werden. Eine manuelle Anpassung ist nötig.   | Im Konvertierungsverfahren konnten nicht alle Ereignisprioritäten eingestellt werden.  |
| TC-132                       | Makro <macro> kann nicht konvertiert werden: die maximale Anzahl an Unterprogrammen wird verwendet.   | Das Twido-Projekt verwendet bereits die maximale Anzahl an Unterprogrammen, die zu Freien POU's konvertiert wurden. Das Makro-Konvertierungsverfahren benötigt möglicherweise zusätzliche Freie POU's. |
| TC-133                       | Die Passwörter von Twido-Anwendungen müssen in Großbuchstaben eingegeben werden.  | Das Twido-Passwort wurden in Großbuchstaben von TwidoSuite oder TwidoSoft gespeichert.   |
| <b>Informationsmeldungen</b> |   |  |
| TC-201                       | Die Steuerung <reference1> wurde mit <reference2> ersetzt.  | SoMachine Basic hat eine Standardauswahl für die Ersatzsteuerung getroffen. Wenn diese nicht den erforderlichen Eigenschaften entspricht, ersetzen Sie die Steuerung mit einer anderen Referenz.       |
| TC-202                       | Das Modul <reference1> wurde mit <reference2> ersetzt.  | SoMachine Basic konvertiert TM2-Module zu äquivalenten TM3-Modulen.  |
| TC-203                       | Ein Ethernet-Modul wurde erkannt. Die Steuerung wurde in eine vergleichbare Referenz mit Ethernet-Port geändert.  | Wenn ein Modul 499TWD01100 in TwidoSuite konfiguriert wird, wählt die Konvertierung eine M221 Logic Controller-Referenz mit einem integrierten Ethernet-Port aus.                                      |
| TC-204                       | Eine serielle NAC-Leitungsoption wurde erkannt. Ein SL-Steckmodul wurde in der Konfiguration hinzugefügt.   | Das SL-Steckmodul TMC2SL1 ersetzt einen der 3 seriellen TWDNAC-Adapter von Compact Twido. Prüfen Sie die Konfiguration und Verkablung.   |

| Meldungs-ID | Meldung   | Beschreibung/Lösung   |
|-------------|---|---|
| TC-205      | Eine serielle NOZ-Leitungsoption wurde erkannt. Die zugehörige Konfiguration wurde auf SL2 angewendet.                          | Das SL-Steckmodul TMC2SL1 ersetzt eines der 3 seriellen TWDNOZ-Erweiterungsmodule von Modular Twido. Prüfen Sie die Konfiguration und Verkablung.                 |
| TC-206      | Das Gerät <device> wurde in ein generisches Modem geändert.   | Das Modem TD-33/V90 wird in SoMachine Basic nicht unterstützt.  |
| TC-207      | Das auf SL2 konfigurierte Gerät <device> wurde entfernt, es sind nur SL1-Modems autorisiert.                                    | Es ist nicht möglich, ein Modem auf der seriellen Leitung SL2 in M221 Logic Controller zu konfigurieren. Fügen Sie das Modem auf der seriellen Leitung SL1 hinzu. |
| TC-208      | Die Funktionsebene des Projekts wurde auf <x> eingestellt.  | Prüfen Sie, ob die spezifische Funktionsebene dem Funktionssatz der Logiksteuerungen in Ihrer Konfiguration entspricht.   |
| TC-209      | Die Priorität <x> wurde von <y> zu <z> konvertiert.   | Prüfen Sie die Prioritätsebene, die dem Ereignis zugewiesen ist.  |
| TC-210      | Das Makro <x> in POU <y> - Programmbaustein <z> wurde zu einem äquivalenten Code in POU <a> - Programmbaustein <b> konvertiert. | Prüfen Sie die Funktionalität des konvertierten Codes.  |
| TC-211      | Das Makro <x> in POU <y> - Programmbaustein <z> wurde zu einem äquivalenten Code in der Freien POU <a> konvertiert.             | Prüfen Sie die Funktionalität des konvertierten Codes in der Freien POU.  |

### Systembits

Diese Tabelle listet Twido-Systembits auf, die entweder keine Entsprechung auf M221 Logic Controller oder einen anderen Zweck haben:

| Twido-Systembit | Beschreibung  | M221 Logic Controller-Systembit | Beschreibung                                 |
|-----------------|---|---------------------------------|--|
| %S8             | Verdrahtungsprüfung   | Entfernt                        | In M221 Logic Controller nicht implementiert |
| %S24            | Die Bedienanzeige kann im Bild angehalten sein (eingefroren)  | Entfernt                        | Durch die dezentrale Grafikanzeige ersetzt   |
| %S25            | Auswählen eines Anzeigemodus am Bedienterminal                | Entfernt                        | Durch die dezentrale Grafikanzeige ersetzt   |
| %S26            | Auswählen eines Werts mit oder ohne Vorzeichen an der Anzeige | Entfernt                        | Durch die dezentrale Grafikanzeige ersetzt   |
| %S31            | Ereignismaske   | Entfernt                        | In M221 Logic Controller nicht implementiert |

| Twido-Systembit | Beschreibung   | M221 Logic Controller-Systembit | Beschreibung   |
|-----------------|--|---------------------------------|--|
| %S66            | Aktivieren/Deaktivieren von BAT LED auf einigen Brick 40 | Entfernt                        | In M221 Logic Controller nicht implementiert   |
| %S69            | LED-Anzeige STAT   | Entfernt                        | Es existiert keine Benutzer-STAT-LED auf M221 Logic Controller   |
| %S95            | Wiederherstellung der Speicherworte                      | Zu %S94 verschoben              | Dieses Bit wird auf 1 gesetzt, um die im nicht-flüchtigen Speicher abgelegten Daten wiederherzustellen |
| %S97            | Speichern von %MW OK                                     | Zu %S92 verschoben              | In nicht-flüchtigem Speicher abgelegte %MW-Variablen   |
| %S100           | TwidoSuite-Kommunikationskabelverbindung                 | Entfernt                        | M221 Logic Controller verwendet ein USB-Kabel  |
| %S110           | Datenaustausch über dezentrale Verbindung                | Geändert                        | Setzt den Modbus Serial-E/A-Scanner auf der seriellen Leitung 1 zurück                                 |
| %S111           | Einzelner Datenaustausch über dezentrale Verbindung      | Geändert                        | Setzt den Modbus Serial-E/A-Scanner auf der seriellen Leitung 2 zurück                                 |
| %S112           | Verbindung über dezentrale Verbindung                    | Entfernt                        | Die 'Remote Link'-Funktion ist auf M221 Logic Controller nicht implementiert                           |
| %S113           | Konfiguration/Betrieb der dezentralen Verbindung         | Geändert                        | Setzt den Modbus Serial-E/A-Scanner auf der seriellen Leitung 1 aus                                    |
| %S118           | Fehler dezentrale E/A                                    | Entfernt                        | Die 'Remote Link'-Funktion ist auf M221 Logic Controller nicht implementiert                           |
| %S120           | Überfluss an Eingang PWM0 (%IW0.7) (Twido Extreme)       | Entfernt                        | Keine Eingangs-PWM auf M221 Logic Controller   |
| %S121           | Überfluss an Eingang PWM1 (%IW0.8) (Twido Extreme)       | Entfernt                        | Keine Eingangs-PWM auf M221 Logic Controller   |

Für weitere Informationen, siehe Systembits %S.

## Systemwörter

Diese Tabelle listet Twido-Systemwörter auf, die entweder keine Entsprechung auf M221 Logic Controller oder einen anderen Zweck haben:

| Twido-Systemwort | Beschreibung   | M221 Logic Controller-Systemwort | Beschreibung  |
|------------------|--|----------------------------------|---|
| %SW6             | SPS-Status   | Geändert                         | Steuerungszustand   |
| %SW7             | Steuerungszustand  | Geändert                         | SPS-Status  |
| %SW20...%SW27    | Gibt den Status für CANopen-Slave-Module an  | Entfernt                         | Der CANopen-Bus ist auf M221 Logic Controller nicht verfügbar   |
| %SW49...%SW53    | RTC-Funktionen: Wörter, die Datum und Uhrzeit enthalten (in BCD)   | Geändert                         | RTC-Funktionen: Wörter, die Datum und Uhrzeit enthalten (in BCD).   |
| %SW58            | Zeigt Code an, der die Ursache für den letzten Stopp angibt  | Geändert                         | Zeigt Code an, der die Ursache für den letzten Stopp angibt.  |
| %SW59            | Aktuelles Datum einstellen   | Geändert                         | Aktuelles Datum einstellen.   |
| %SW60            | Echtzeituhr-Korrekturwert  | Entfernt                         | Keine RTC-Korrektur verfügbar.  |
| %SW67            | Funktion und Typ der Steuerung   | Geändert                         | Funktion und Typ der Steuerung.   |
| %SW68            | Auf der 2-Zeilen-Bedienanzeige angezeigte Elemente   | Entfernt                         | Auf M221 Logic Controller existiert keine integrierte Anzeige; Sie wird durch die dezentrale Grafikanzeige ersetzt.   |
| %SW69            | Auf der 2-Zeilen-Bedienanzeige angezeigte Elemente   | Entfernt                         | Auf M221 Logic Controller existiert keine integrierte Anzeige; Sie wird durch die dezentrale Grafikanzeige ersetzt.   |
| %SW73            | Status des AS-Interface-Systems  | Entfernt                         | Der ASI-Bus ist auf M221 Logic Controller nicht verfügbar.  |
| %SW74            | Status des AS-Interface-Systems  | Entfernt                         | Der ASI-Bus ist auf M221 Logic Controller nicht verfügbar.  |
| %SW80            | Status E/A-Grundgerät  | Geändert                         | Status der integrierten Analogeingänge  |
| %SW81...%SW87    | Status der E/A-Erweiterungsmodule 1 bis 7  | Zu %IWS,<br>%QWS<br>verschoben   | Systemobjekte für den Status von Analogeingängen und Analogausgängen  |
| %SW96            | Befehl und/oder Diagnose für die Funktion zum Speichern/Wiederherstellen des Applikationsprogramms und von %MW | Geändert                         | Diagnose für die Funktion ‚Speichern/Wiederherstellen‘ des Programms und %MW (siehe Persistente Variablen ( <i>siehe Modicon M221, Logic Controller, Programmierhandbuch</i> ) für weitere Informationen) |

| Twido-Systemwort | Beschreibung   | M221 Logic Controller-Systemwort | Beschreibung  |
|------------------|--|----------------------------------|---|
| %SW96:X0         | Spezifische Speicherworte müssen im nicht-flüchtigen Speicher gespeichert werden | %S93                             | %SW96:X0 kann nicht auf M221 Logic Controller geschrieben werden; ersetzen Sie in Ihrem Programm %SW96:X0 mit %S93.<br>Ersetzen Sie die Systembits %S95 und %S97 mit %S94 bzw. %S92. Ersetzen Sie das Systemwort %SW97 mit %SW48. Prüfen Sie die Verwendung anderer Bits des Systemworts %SW96. |
| %SW97            | Befehle oder Diagnose für die Funktion Speichern/Wiederherstellen                | Zu %SW148 verschoben             | Anzahl an persistenten Variablen (Maximal 2000 Variablen)   |
| %SW111           | Status dezentrale Verbindung   | Entfernt                         | Die 'Remote Link'-Funktion ist auf M221 Logic Controller nicht implementiert.   |
| %SW112           | Fehlercode (Konfiguration/Betrieb) der dezentralen Verbindung                    | Entfernt                         | Die 'Remote Link'-Funktion ist auf M221 Logic Controller nicht implementiert.   |
| %SW113           | Konfiguration dezentrale Verbindung  | Entfernt                         | Die 'Remote Link'-Funktion ist auf M221 Logic Controller nicht implementiert.   |

Für weitere Informationen, siehe Systembits %SW (siehe *Modicon M221, Logic Controller, Programmierhandbuch*).



# Anhang B

## SoMachine Basic-Tastaturkombinationen

### SoMachine Basic Tastaturkombinationen

#### Liste der Tastaturkombinationen

| Modifikator | Taste  | Befehl                                | Ansicht        | Bedingung   |
|-------------|--------|---------------------------------------|----------------|---|
| STRG        | C      | Kopieren                              | Textfeld       | –   |
| STRG        | V      | Einfügen                              | Textfeld       | –   |
| STRG        | X      | Ausschneiden                          | Textfeld       | –   |
| ALT         | Links  | Zur vorherigen Registerkarte wechseln | Alle           | –   |
| ALT         | Rechts | Zur nächsten Registerkarte wechseln   | Alle           | –   |
|             | F1     | Hilfe anzeigen oder Kontexthilfe      | Alle           | Auswahl unter <b>Systemeinstellungen</b> → <b>Allgemein</b> |
| UMSCHALT    | F1     |                                       |                |   |
| ALT         | F4     | SoMachine Basic beenden               | Alle           | –   |
| STRG        | B      | Simulator starten                     | Alle           | –   |
| STRG        | G      | Einloggen                             | Alle           | –   |
| STRG        | H      | Abmelden                              | Alle           | –   |
| STRG        | L      | Stoppen Reglers                       | Alle           | –   |
| STRG        | M      | Ausführen Reglers                     | Alle           | –   |
| STRG        | N      | Neues Projekt                         | Alle           | –   |
| STRG        | O      | Projekt öffnen                        | Alle           | –   |
| STRG        | P      | Drucken                               | Alle           | –   |
| STRG        | Q      | SoMachine Basic beenden               | Alle           | –   |
| STRG        | S      | Projekt speichern                     | Alle           | –   |
| STRG        | W      | Simulator stoppen                     | Alle           | –   |
| STRG        | J      | Download                              | Inbetriebnahme | –   |
| STRG        | K      | Upload                                | Inbetriebnahme | –   |
|             | ALT    | Ladder-Shortcuts anzeigen             | Programmierung | –   |
|             | Entf   | Löschen                               | Programmierung | Elemente sind ausgewählt                                    |

| Modifikator | Taste      | Befehl  | Ansicht              | Bedingung   |
|-------------|------------|---|----------------------|---|
| STRG        | D          | Alle Bausteine im Programm in Ladder konvertieren   | Programmierung       | –   |
| STRG+ALT    | D          | Alle Bausteine im Programm in IL konvertieren   | Programmierung       | –   |
| STRG        | F          | Suchen  | Programmierung       | –   |
| STRG        | I          | Neuen Programmbaustein vor dem ausgewählten Programmbaustein einfügen                       | Programmierung       | –   |
| STRG        | J          | Wiederherstellen  | Programmierung       | –   |
| STRG        | Z          | Rückgängig  | Programmierung       | –   |
| STRG        | Pfeiltaste | Linie zeichnen  | KOP-Programmbaustein | Zeichentool ist ausgewählt  |
| STRG        | Pfeiltaste | Linie löschen   | KOP-Programmbaustein | Löschttool ist ausgewählt   |
| STRG        | Pfeiltaste | Nächste Kontaktplan-Zelle auswählen/Auswahl aufheben (Zelle für Zelle)                      | KOP-Programmbaustein | Auswahltool ist ausgewählt  |
| UMSCHALT    | Pfeiltaste | Nächste Kontaktplan-Zellen auswählen/Auswahl aufheben (Auswahl nach Bereich)                | KOP-Programmbaustein | Auswahltool ist ausgewählt  |
|             | ESC        | Zeiger auf Auswahltool zurücksetzen   | KOP-Programmbaustein | Ausgewähltes Tool ist nicht „Leiter zeichnen“ oder „Leiter löschen“, es werden keine Elemente gezogen, kein Pop-up-Fenster geöffnet |
|             | ESC        | Ausstehende Linie abbrechen   | KOP-Programmbaustein | Zeichenvorgang findet statt   |
|             | ESC        | Löschen der Linie abbrechen   | KOP-Programmbaustein | Löschvorgang findet statt   |
|             | ESC        | Verschieben der ausgewählten Element(e) abbrechen (ursprüngliche Position wiederherstellen) | KOP-Programmbaustein | KOP-Elemente werden gezogen   |
|             | ESC        | Vorschlagsliste schließen   | KOP-Programmbaustein | Eine Vorschlagsliste ist geöffnet (z. B. die verfügbaren Deskriptoren für einen Kontakt)  |

| Modifikator | Taste      | Befehl  | Ansicht              | Bedingung   |
|-------------|------------|---|----------------------|---|
|             | ESC        | Menüelement der KOP-Symbolleiste schließen    | KOP-Programmbaustein | Ein Menü der KOP-Symbolleiste ist geöffnet (z. B. Funktionsbausteine) |
|             | ENTER      | Verschieben von KOP-Elementen starten/stoppen | KOP-Programmbaustein | Mindestens eine Zelle ist ausgewählt                                  |
|             | Pfeiltaste | Frei bewegliche Zelle verschieben             | KOP-Programmbaustein | Verschieben der Zelle hat begonnen                                    |
|             | Pfeiltaste | Aktuelle Zelle wechseln                       | KOP-Programmbaustein | Standardmäßig   |
|             | F5         | Kontakt öffnen                                | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
|             | F6         | Offene Verzweigung                            | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
| UMSCHALT    | F5         | Kontakt schließen                             | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
| UMSCHALT    | F6         | Zweig schließen                               | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
|             | F7         | Spule   | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
| STRG        | F7         | Negierte Spule                                | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
| STRG        | F5         | Set-Spule                                     | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
| STRG        | F6         | Reset-Spule                                   | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
|             | F8         | Anwendungsanweisung                           | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
|             | F9         | Horizontale Linie zeichnen                    | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
|             | F10        | Vertikale Linie zeichnen                      | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
| STRG        | F9         | Horizontale Linie löschen                     | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
| STRG        | F10        | Vertikale Linie löschen                       | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
| UMSCHALT    | F7         | Offener Kontakt mit steigender Impulsflanke   | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |
| UMSCHALT    | F8         | Offener Kontakt mit fallender Impulsflanke    | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“                            |

| Modifikator   | Taste | Befehl  | Ansicht              | Bedingung                                  |
|---------------|-------|---|----------------------|--|
| ALT           | F7    | Offener Verzweigung mit steigender Impulsflanke | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“ |
| ALT           | F8    | Offener Verzweigung mit fallender Impulsflanke  | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“ |
| STRG+UMSCHALT | O     | Vergleichsbaustein                              | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“ |
|               | X     | XOR-Bausteine                                   | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“ |
|               | F     | Funktionsbausteine                              | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“ |
|               | A     | Schritt aktivieren                              | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“ |
|               | D     | Schritt deaktivieren                            | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“ |
| STRG+ALT      | F10   | Ergebnisse Umkehrbetrieb                        | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“ |
|               | O     | Andere Kontaktplan-Elemente                     | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“ |
| ALT           | F10   | Freihand-Linie zeichnen                         | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“ |
| ALT           | F9    | Freihand-Linie löschen                          | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 1“ |
|               | C     | Neuer Kontakt                                   | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 2“ |
|               | /     | Neuer geschlossener Kontakt                     | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 2“ |
|               | W     | Neues Kontakt-OR                                | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 2“ |
|               | X     | Neues OR für geschlossenen Kontakt              | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 2“ |
| STRG+UMSCHALT | F4    | Steigende Flanke                                | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 2“ |
| STRG+UMSCHALT | F5    | Fallende Flanke                                 | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 2“ |
| STRG+UMSCHALT | O     | Vergleichsbaustein                              | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 2“ |
| ALT           | X     | XOR-Bausteine                                   | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 2“ |
|               | F10   | Neue vertikale Linie                            | KOP-Programmbaustein | KOP-Symbolleiste „Asiatische Definition 2“ |

| Modifikator   | Taste | Befehl                            | Ansicht                  | Bedingung  |
|---------------|-------|-----------------------------------|--------------------------|--|
| ALT           | L     | Neue horizontale Linie            | KOP-<br>Programmbaustein | KOP-Symboleiste<br>„Asiatische Definition 2“             |
|               | O     | Neue Spule                        | KOP-<br>Programmbaustein | KOP-Symboleiste<br>„Asiatische Definition 2“             |
|               | Q     | Neue geschlossene Spule           | KOP-<br>Programmbaustein | KOP-Symboleiste<br>„Asiatische Definition 2“             |
| STRG          | F9    | Set-Spule                         | KOP-<br>Programmbaustein | KOP-Symboleiste<br>„Asiatische Definition 2“             |
| STRG+UMSCHALT | F9    | Reset-Spule                       | KOP-<br>Programmbaustein | KOP-Symboleiste<br>„Asiatische Definition 2“             |
|               | A     | Schritt aktivieren                | KOP-<br>Programmbaustein | KOP-Symboleiste<br>„Asiatische Definition 2“             |
|               | D     | Schritt deaktivieren              | KOP-<br>Programmbaustein | KOP-Symboleiste<br>„Asiatische Definition 2“             |
|               | I     | Neue Anweisung                    | KOP-<br>Programmbaustein | KOP-Symboleiste<br>„Asiatische Definition 2“             |
|               | F     | Neuer Funktionsbaustein           | KOP-<br>Programmbaustein | KOP-Symboleiste<br>„Asiatische Definition 2“             |
| ALT           | O     | Andere Kontaktplan-<br>Elemente   | KOP-<br>Programmbaustein | KOP-Symboleiste<br>„Asiatische Definition 2“             |
|               | F2    | Verzweigungsmodus<br>deaktivieren | KOP-<br>Programmbaustein | Europäische oder<br>amerikanische Ladder-<br>Symboleiste |
| UMSCHALT      | F2    | Verzweigungsmodus<br>aktivieren   | KOP-<br>Programmbaustein | Europäische oder<br>amerikanische Ladder-<br>Symboleiste |
| UMSCHALT      | F3    | Schließerkontakt                  | KOP-<br>Programmbaustein | Europäische KOP-<br>Symboleiste                          |
| UMSCHALT      | F4    | Öffnerkontakt                     | KOP-<br>Programmbaustein | Europäische KOP-<br>Symboleiste                          |
| STRG+UMSCHALT | F4    | Steigende Flanke                  | KOP-<br>Programmbaustein | Europäische KOP-<br>Symboleiste                          |
| STRG+UMSCHALT | F5    | Fallende Flanke                   | KOP-<br>Programmbaustein | Europäische KOP-<br>Symboleiste                          |
| STRG+UMSCHALT | 6     | Operationsbaustein                | KOP-<br>Programmbaustein | Europäische KOP-<br>Symboleiste                          |
| STRG+UMSCHALT | O     | Vergleichsbaustein                | KOP-<br>Programmbaustein | Europäische KOP-<br>Symboleiste                          |
|               | X     | XOR-Bausteine                     | KOP-<br>Programmbaustein | Europäische KOP-<br>Symboleiste                          |

| Modifikator   | Taste | Befehl                            | Ansicht                  | Bedingung                        |
|---------------|-------|-----------------------------------|--------------------------|----------------------------------|
| UMSCHALT      | F7    | Zuweisung                         | KOP-<br>Programmbaustein | Europäische KOP-<br>Symbolleiste |
| STRG+UMSCHALT | F9    | Negierte Spule                    | KOP-<br>Programmbaustein | Europäische KOP-<br>Symbolleiste |
|               | F9    | Set-Spule                         | KOP-<br>Programmbaustein | Europäische KOP-<br>Symbolleiste |
| UMSCHALT      | F9    | Reset-Spule                       | KOP-<br>Programmbaustein | Europäische KOP-<br>Symbolleiste |
|               | A     | Schritt aktivieren                | KOP-<br>Programmbaustein | Europäische KOP-<br>Symbolleiste |
|               | D     | Schritt deaktivieren              | KOP-<br>Programmbaustein | Europäische KOP-<br>Symbolleiste |
| UMSCHALT      | F5    | Funktionsbaustein                 | KOP-<br>Programmbaustein | Europäische KOP-<br>Symbolleiste |
| STRG+UMSCHALT | F6    | Operationsbaustein                | KOP-<br>Programmbaustein | Europäische KOP-<br>Symbolleiste |
|               | F3    | Linie                             | KOP-<br>Programmbaustein | Europäische KOP-<br>Symbolleiste |
|               | F3    | Leiterlinie zeichnen              | KOP-<br>Programmbaustein | Europäische KOP-<br>Symbolleiste |
|               | F4    | Leiterlinie löschen               | KOP-<br>Programmbaustein | Europäische KOP-<br>Symbolleiste |
|               | O     | Andere Kontaktplan-<br>Elemente   | KOP-<br>Programmbaustein | Europäische KOP-<br>Symbolleiste |
| UMSCHALT      | F2    | Verzweigungsmodus<br>aktivieren   | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste   |
|               | F2    | Verzweigungsmodus<br>deaktivieren | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste   |
|               | F3    | Leiterlinie zeichnen              | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste   |
| UMSCHALT      | F3    | Leiterlinie löschen               | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste   |
|               | F4    | Ruhekontakt                       | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste   |
| UMSCHALT      | F4    | Negierter Kontakt                 | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste   |
| STRG          | F9    | Spule                             | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste   |
| STRG+UMSCHALT | F9    | Negative Spule                    | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste   |

| Modifikator   | Taste  | Befehl                          | Ansicht                  | Bedingung                      |
|---------------|--------|---------------------------------|--------------------------|--------------------------------|
|               | F9     | Set-Spule                       | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste |
| UMSCHALT      | F9     | Reset-Spule                     | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste |
| STRG+UMSCHALT | F4     | Steigende Flanke                | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste |
| STRG+UMSCHALT | F5     | Fallende Flanke                 | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste |
| STRG+UMSCHALT | {6, 7} | Operationsbaustein              | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste |
| STRG+UMSCHALT | {O, P} | Vergleichsbaustein              | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste |
| X oder ALT+X  |        | XOR-Bausteine                   | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste |
| O oder ALT+O  |        | Andere Kontaktplan-<br>Elemente | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste |
| A oder ALT+A  |        | Schritt aktivieren              | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste |
| D oder ALT+D  |        | Schritt deaktivieren            | KOP-<br>Programmbaustein | SoMachine-KOP-<br>Symbolleiste |





## !

### %S

Gemäß dem IEC-Standard entspricht %S einem Systembit.

### %SW

Gemäß dem IEC-Standard entspricht %SW einem Systemwort.

## A

### **Animationstabelle**

Software-tabelle zur Anzeige der Echtzeitwerte von Objekten wie Eingangsbits und Speicherwörter. Wenn SoMachine Basic mit einer Logiksteuerung verbunden wird, können die Werte bestimmter Objekttypen in den Animationstabellen auf spezifische Werte forciert werden. Animationstabellen werden als Teil der SoMachine Basic-Anwendungen gespeichert.

### **Anweisungsliste (Programmiersprache)**

Ein in der Programmiersprache Anweisungsliste (AWL oder IL: Instruction List) geschriebenes Programm besteht aus einer Abfolge textbasierter Anweisungen, die von der Steuerung der Reihe nach ausgeführt werden. Jede Anweisung besteht aus einer Zeilennummer, einem Anweisungscode und einem Operanden (siehe IEC 61131-3).

### **Anwendung**

Programm mit Konfigurationsdaten, Symbolen und Dokumentation.

## B

### **Benutzerdefinierte Funktion**

Ermöglicht Ihnen die Erstellung Ihrer eigenen Funktionen mit einem oder mehreren Eingangsparametern, lokalen Variablen und Rückgabewert. Die benutzerdefinierte Funktion kann dann in Operationsbausteinen aufgerufen werden. Eine benutzerdefinierte Funktion wird als Teil des Projekts gespeichert und als Bestandteil der Anwendung in die Steuerung heruntergeladen.

### **Benutzerdefinierter Funktionsbaustein**

Ermöglicht Ihnen die Erstellung eigener Funktionsbausteine mit einem oder mehreren Ein- und Ausgängen, Parametern und lokalen Variablen. Benutzerdefinierte Funktionsbausteine werden als Teil des Projekts gespeichert.

## E

### E/A

*Eingang/Ausgang*

### Erweiterungsbus

Elektronischer Kommunikationsbus zwischen E/A-Erweiterungsmodulen und einer Steuerung.

### EtherNet/IP

(*Ethernet Industrial Protocol*) Offenes Kommunikationsprotokoll für Fertigungsautomatisierungslösungen in industriellen Systemen. EtherNet/IP gehört zu einer Familie von Netzwerken, die CIP (Common Industrial Protocol) in den oberen Schichten implementieren. Die unterstützende Organisation (ODVA) gibt EtherNet/IP für globale Anpassungsfähigkeit und Medienunabhängigkeit vor.

## F

### Flash-Speicher

Nicht flüchtiger Speicher, der überschrieben werden kann. Er wird in einem speziellen EEPROM abgelegt, der gelöscht und neu programmiert werden kann.

### Free POU

Eine programmierbare Objekteinheit (POU: Programmable Object Unit) enthält in der Regel Bibliotheksfunktionen, die unabhängig von der Master-Task eines Programms programmiert und aktualisiert werden können. Freie POU's können ausgehend von Programmen als Unterprogramme oder Sprünge aufgerufen werden. Beispiel: Die *periodische Task* ist ein Unterprogramm, das als freie POU implementiert ist.

## G

### GRAFSET

Funktionsweise eines sequenziellen Vorgangs (Ablauf) in strukturierter und grafischer Form.

Hierbei handelt es sich um ein analytisches Verfahren, bei dem Ablaufsteuerungssysteme in eine Reihe von Schritten unterteilt werden, denen Aktionen, Übergänge und Bedingungen zugewiesen sind.

## K

### Keine Programmdaten

Daten in einer SoMachine Basic-Anwendung, die nicht direkt vom Programm verwendet werden, z. B. Projekteigenschaften, Symbole und Kommentare.

### Konfiguration

Die Anordnung und Vernetzung von Hardwarekomponenten innerhalb eines Systems und die Hardware- und Softwareparameter, die die Betriebsmerkmale des Systems bestimmen.

**Kontaktplan (Programmiersprache)**

Grafische Darstellung der Anweisungen eines Steuerungsprogramms mit Symbolen für Kontakte, Spulen und Bausteine in einer Abfolge von Programmbausteinen, die von der Steuerung der Reihe nach ausgeführt werden (siehe IEC 61131-3).

**M****Master-Task**

Prozessortask, die über die zugehörige Programmiersoftware ausgeführt wird. Die Master-Task besteht aus 2 Sections:

- **IN:** Vor der Ausführung der Master-Task werden die Eingänge in die IN-Section kopiert.
- **OUT:** Nach der Ausführung der Master-Task werden die Ausgänge in die OUT-Section kopiert.

**P****Post-Konfiguration**

Option, mit der Sie einige Anwendungsparameter ändern können, ohne die gesamte Anwendung bearbeiten zu müssen. Die Post-Konfigurationsparameter befinden sich in einer in der Steuerung gespeicherten Datei. Sie überschreiben die Konfigurationsparameter der Anwendung.

**POU**

*(Program Organization Unit: Programmierorganisationseinheit)* Variablendeklaration im Quellcode und der entsprechende Anweisungssatz. POU's ermöglichen die modulare Wiederverwendung von Softwareprogrammen, Funktionen und Funktionsbausteinen. Sobald POU's deklariert sind, stehen sie sich gegenseitig zur Verfügung.

**Programm**

Komponente einer Anwendung, die aus kompiliertem Quellcode besteht und im Speicher einer programmierbaren Steuerung installiert werden kann.

**R****RTC**

*(Real-Time Clock: Echtzeituhr)* Batteriebetriebene Uhr zur Uhrzeit- und Datumsanzeige, die während der gesamten Lebensdauer der Batterie permanent in Betrieb ist, selbst bei ausgeschalteter Steuerung.

**S****Steuerung**

Ermöglicht die Automatisierung industrieller Prozesse (auch als speicherprogrammierbare Steuerung oder SPS bezeichnet).

## Symbol

Zeichenkette mit maximal 32 alphanumerischen Zeichen, von denen das erste Zeichen ein Buchstabe ist. Mit Symbolen können Sie ein Steuerungsobjekt personalisieren, um die Pflfegbarkeit der Anwendung zu erhohen.

## Symbolische Adressierung

Die symbolische Adressierung ist die indirekte Methode zur Adressierung von Speicherobjekten, einschlielich physischer Ein- und Ausgange, die in Programmierbefehlen Verwendung findet. Dabei werden zunachst Symbole fur die Objekte definiert, die dann in Verbindung mit den Programmierbefehlen verwendet werden.

Diese Adressierungsmethode wird der direkten Adressierung vorgezogen, da die Symbole bei einer anderung der Programmkonfiguration automatisch mit den neuen direkten Adresszuordnungen aktualisiert werden. Im Gegensatz dazu werden direkte Adressen, die als Operanden oder Parameter verwendet werden, nicht aktualisiert (siehe *Direkte Adressierung*).

## T

### TCP

(*Transmission Control Protocol*) Verbindungsbasiertes Protokoll der Transportschicht, das die zuverlassige, simultane und bidirektionale ubertragung von Daten unterstutzt. TCP ist Teil der TCP/IP-Protokollreihe.

## W

### Watchdog

Ein Watchdog ist ein spezieller Zeitgeber (Timer), der gewahrleistet, dass Programme nicht die ihnen zugewiesene Abfragezeit uberschreiten. Der Watchdog-Timer wird in der Regel auf einen Wert gesetzt, der groer ist als die Abfragezeit, und am Ende jedes Abfragezyklus auf 0 zuruckgesetzt. Wenn der Watchdog-Timer den voreingestellten Wert (Preset-Wert) erreicht, beispielsweise weil das Programm in einer Endlosschleife gefangen ist, wird ein Fehler signalisiert und das Programm angehalten.



## Specials

## Symbols

Logiksteuerung  
Anzeigen von Informationen über, *280*

## A

Abfrage-Task, Konfigurieren des Watchdogs, *93*  
Abfragemodi, *95, 133*  
Abfragezeit  
Anzeigen der minimalen, maximalen, aktuellen, *280*  
Minimum, Angezeigt im Statusbereich, *57*  
Adressierung  
Symbolisch, *80*  
Akkumulator, *213*  
Aktionsbereich, *188*  
Allgemein, *60*  
Animationstabellen, *157*  
Anpassen, Kontaktplan-Editor, *60*  
Antriebsobjekte, *173*  
Anweisungen  
Vor-/Nachgeschaltet, *205*  
Anweisungsliste, *214*  
Kommentare, *211*  
Übersicht, *210*  
Anwendung  
Definition, *28*  
Herunterladen in die Steuerung, *268*  
Passwortschutz, *67, 70, 280*  
Verhalten, Konfigurieren, *91*  
Applikation  
Von der Logiksteuerung hochladen, *270*  
Ausgänge  
Ändern, *205*  
Auswahl  
Grafikelement, *193*

## B

Beibehalten der Fehlerausweichwerte, *93*  
Benutzerdefinierte Funktion  
Programmierung, *115*  
Benutzerdefinierte Funktionen  
Verwalten, *120*  
Benutzerdefinierter Funktionsbaustein  
Definieren, *125*  
Programmierung, *125*  
Verwalten, *128*  
Benutzeroberfläche  
Einstellen der Sprache, *60*  
Berichte  
Drucken, *62*  
Exportieren, *62*  
Betriebsmodi, *31*  
Boolesch  
Akkumulator, *213*  
Boolesche Operatoren  
Grafikelemente, *194*

## C

Cachespeicher, Nutzung, *183*

## D

Digitaleingänge  
Konfigurieren als Ereignisquellen, *148*  
Drucken von Berichten, *62*

## E

Eigenschaften, *67*  
einfügen  
neuer Graficet-POUs, *107*  
Eingänge  
Ändern, *205*  
Konfigurieren als Ereignisquellen, *148*  
Eingangs-/Ausgangsobjekte, *169*

- Einstellungen
  - Allgemein, *60*
- Ende/Sprung
  - Grafikelemente, *196*
- Entfernen
  - Freie POU, *112*
- Entfernen des Passwortschutzes, *70, 70*
- entfernen von
  - Grafcet-POUs, *108*
- Entwickeln von Programmen, Phasen, *29*
- Entwicklungsphasen, *30*
- Ereignis-Tasks
  - konfigurieren, *95*
- Ereignisquelle
  - Typen, *148*
  - Zuweisen eines Unterprogramms als, *151*
- Ereignisquellen, *148*
- Ereignisse
  - Auslösen von Unterprogrammen über, *148*
  - Seit letztem Kaltstart, *152*
- Ereignistasks
  - Verwalten, *150*
  - Übersicht, *147*
- Ersetzen
  - Steuerung in der Konfiguration, *74*
- Erstellen
  - Freie POU, *110*
- erstellen von
  - Grafcet-POUs, *107*
- Erstellen von Projekten, *28*
- Erweiterungsmodule
  - Unterstützte Geräte, *24*
- Ethernet
  - Konfiguration mithilfe der Post-Konfigurationsdatei, *280*
- EXCEPTION-Status
  - Fehlerausweichverhalten, *93*
- Exportieren
  - Symbolliste, *180*
  - Tracing, *248*

## F

- Fehlerausweich
  - verhalten, Festlegen der, *93*
  - werte, *93*
- Fehlerbehebung im Online-Modus, *245*
- Firmware-Updates, *272*
- Firmware, Herunterladen von Aktualisierungen in die Steuerung, *272*
- Forcieren der Wert
  - E/A, *280*
- Forcieren von Werten
  - In Animationstabellen, *157*
- Freie POU
  - Einführung zu, *99*
  - Erstellen, *110*
  - Für periodische Task, *143*
  - Zuweisen zu einer Ereignisquelle, *151*
  - Zuweisen zu einer periodischen Task, *112*
  - Zuweisen zu Ereignissen, *112*
- Freie POU, *112*
- Freigeben
  - Symbolliste, *181*
- Freigeben von Symbolen
  - Für Vijeo Designer-Projekte, *181*
- Funktionsbausteine
  - Grafikelement, *195*
- Funktionsebenen, *94*

## G

- Grafcet, *222*
  - Anweisungen, *222*
  - grafische Elemente, *196*
  - Nachverarbeitung, *227*
  - Programmstruktur, *224*
  - Sequenzielle Verarbeitung, *225*
  - Verwendung von Anweisungen, *228*
  - Vorverarbeitung, *224*
- Grafcet (SFC)
  - Grafcet Grafischer Editor, *234*
- Grafcet-POU
  - erstellen, *107*
  - umbenennen, *108*

- Grafcet-POUs  
 entfernen, *108*  
 kopieren und einfügen, *107*
- Grafikelemente  
 Kontaktplan, *192*
- H**
- Hardware-Komponente, konfigurieren, *73*  
 Hardwareübersicht, *73*  
 Herunterladen  
 Anwendung direkt in die Steuerung, *46*  
 Benutzeranwendung in die Steuerung, *268*  
 Firmwareaktualisierungen, *272*  
 Herunterladen nicht-ausführbarer Daten, *256*  
 Hilfe  
 Ändern der Tastaturkürzel, *60*  
 hochladen  
 Applikation von Logiksteuerung, *270*  
 Hochladen  
 Verhindern durch Passwortschutz, *70*
- I**
- Importieren  
 Symbolliste, *179*  
 Impulsbreite (TON) , *133*  
 Inbetriebnahme, *30*  
 Fenster ‚Inbetriebnahme‘, *260*  
 Verbinden mit einer Logiksteuerung, *262*  
 Inhalte des Sicherungsspeichers, *256*
- K**
- Katalog, *73*  
 Ersetzen der Steuerung durch eine Katalogreferenz, *74*  
 Keine Programmdateien, *28*  
 Klammern  
 Modifikatoren, *219*  
 Verschalten, *219*  
 Verwenden in Kontaktplan, *206*  
 Verwenden in Programmen, *218*
- Kommentare  
 Hinzufügen in Anweisungsliste, *211*  
 Hinzufügen in Kontaktplan, *204*  
 Kommunikationsobjekte, *174*  
 Kompilieren, Datum und Uhrzeit der letzten Kompilierung, *183*  
 Konfiguration  
 Aktuell, *73*  
 Ersetzen der Steuerung, *74*  
 Konfigurieren  
 Anwendungsverhalten, *91*  
 Dauer des periodischen Task, *145*  
 konfigurieren  
 Hardware-Komponenten mit Konfigurationsfenster, *73*  
 Konfigurieren  
 Master-Task, *132*  
 Projekteigenschaften, *67*  
 konfigurieren  
 Tasks und Abfrage , *95*  
 Kontakte  
 Grafikelemente, *193*  
 Grafische Darstellung von Eingängen, *186*  
 Kontaktplan  
 Einführung, *186*  
 Grafikelemente, *192*  
 Kommentare, *204*  
 Programmausteine, *187*  
 Programmiergrundlagen, *188*  
 Umkehrbar zu Anweisungsliste, *84*  
 Verwenden von Klammern, *206*  
 Kontaktplan-Editor  
 Anpassen, *60*  
 Definieren von Symbolen, *81*  
 Zurücksetzen des Zählers nach Einfügeaktion, *60*  
 Konvertieren von Twido-Projekten zu SoMachine Basic, *307, 307*  
 Kopieren und Einfügen  
 POU, *110*  
 kopieren und einfügen von  
 Grafcet-POUs, *107*

**L**

- Lebenszyklusstatus
  - Steuerung , 57
- Linie
  - Grafikelement, 193
- Logiksteuerung
  - Firmware aktualisieren, 272

**M**

- Master-Task
  - konfigurieren, 95
  - Konfigurieren, 132
  - Steuernde Systembits und -wörter, 133
  - Zuweisen der POU als, 99
- Materialliste, Drucken, 62
- Mindestsystemanforderungen, 23
- Modem
  - Verbindungen, Hinzufügen, 42
- Modi, Betrieb, 31
- Modulbereich, 30

**N**

- Netzwerkobjekte, 170, 170
- neue Graficet-POU
  - einfügen, 107
- nicht-ausführbare Daten, 157
  - Herunterladen, 256
- Normaler Abfragemodus, 133
- Not-Aus-Sensoren, Verdrahtung, 205

**O**

- Objekte
  - Aktualisieren von Werten in Echtzeit, 157
  - Definition von, 79
  - Netzwerk, 170
  - Zum Tracing in Animationstabellen, 157
- Offline-Modus
  - Angezeigt im Statusbereich, 57
  - Überblick, 31

- Online-Modus, 83
  - Aktualisieren der RTC, 283
  - Angezeigt im Statusbereich, 57
  - Animationstabellen in, 157
  - Bearbeiten von Werten in der Animations-tabelle, 159
  - Fehlerbehebung, 245
  - Überblick, 31
- Operanden, 213
- Operationen
  - Einfügen in KOP-Programmbausteine, 200
- Operationsbausteine
  - Einfügen von Zuweisungsanweisungen, 200
  - Grafikelement, 196

**P**

- Passwort
  - Anwendungsschutz mit, 280
  - Entfernen von einem Projekt, 70
  - Entfernen von einer Anwendung, 70
  - Erforderlich zum Öffnen eines Projekts, 69
  - Passwortschutz einer Anwendung , 70
- Passwortschutz für eine Anwendung, 67
- Periode, Abfrage, 133
- Periodisch
  - Abfragemodus, 133
  - Abfrageperiode, 145
  - Tasks, 143
- Periodische Task
  - Zuweisen einer freien POU, 112
- periodischer Task
  - konfigurieren, 95
- Periodischer Task
  - Konfigurieren der Dauer, 145
- Phasen der Programmentwicklung, 30
- Post-Konfiguration
  - Mithilfe der Ethernet-Parameter aus, 280
  - Mithilfe der seriellen Leitungsparameter aus, 280
- Post-Konfigurationsdatei, Schreiben von Ethernet-Parametern in, 44

## POU

- Einfügen, *110*
- Frei, *143*
- Kopieren, *110*
- Übersicht, *99*
- Verwalten mit Tasks, *100*

Prioritätsstufe, Ereignisse, *147*

## Programm

- Anzeigen der Zeilen im Programm, *183*
- Definition, *28*
- Kompilieren, *55*
- Sprünge, *205*

Programm, Konfigurieren der Fehlerabweichverhalten, *93*

## Programmbausteine

- Einfügen, *104*
- Erstellen, *103*
- Grafikelement, *192*
- Kopieren, *105*
- Löschen, *106*
- Umbenennen, *105*
- Verwalten, *103*

Programmentwicklung, Phasen, *29*

## Programmieren

- Arbeitsbereich, *76*

## Programmierung

- Best Practices, *205*
- Raster, *188*
- Sprachen, Unterstützte, *26*

Programmorganisationseinheit (POU), *99*

## Projekt

- Als Vorlage speichern, *303*
- Anzeigen von Berichten für, *62*
- Definition, *28*
- Konfigurieren der Eigenschaften, *67*
- Passwortschutz, *69*
- Speichern, *302*
- Vorlagen, *48*

## Projekten

- erstellen, *28*

PTO-Objekte, *172*

## R

## RAM-Speicher

- Exe-Datei mit Anwendung, *280*

RAM-Speicher, Nutzung, *183*

Rasterlinien, Stil des Kontaktplan-Editors, *60*

Registrierung von SoMachine Basic Software, *36*

Relaisschaltungen, Darstellen in Kontaktplan, *186*

Rollback-Änderungen, *256*

## RTC

- Aktualisieren in der Steuerung, *283*
- Anzeigen von Datum und Uhrzeit, *280*
- Verwalten über Systembits, *205*

## S

Schaltflächen der Symbolleiste, *55*

Schaltflächen, Symbolleiste, *55*

Schwellenwertausgänge (von %HSC)

- Konfigurieren als Ereignisquellen, *148*

## SD-Karte

- Speicherverwaltung mit, *274*

## Sections

- Der Master-Task, *132*
- In Ereignissen, *147*

Senden von Programmänderungen, *256*

## Serielle Leitung

- Konfiguration mithilfe der Post-Konfigurationsdatei, *280*

Simulator, *286*

- Ausgangsverfolgung, *296*
- E/A-Manager-Fenster, *288*
- Modus, Überblick, *31*
- Simulator Fenster, *286*
- Verwendung, *299*
- Werte analoger Eingänge ändern, *295*
- Werte ändern, *293*
- Werte digitaler Eingänge ändern, *293*
- Zeitverwaltungsfenster, *290*
- Zugriff auf den Simulator, *286*

Software-Objekte, *171*

Speichernutzung, Ansicht, *183*

Speicherobjekte, *163*

Speicherverwaltung mit SD-Karte, *274*

- Speicherzuweisung, *83*
  - Sprache,
    - Benutzeroberfläche, *60*
  - SPS-Zeit, Anzeigen im Trace-Fenster, *246*
  - Spulen
    - Grafikelemente, *195*
    - Grafische Darstellung von Ausgängen, *186*
  - Startseite, *30*
  - Status
    - Initialstatus der Steuerung, *91*
    - Status der Steuerung beim Start, *91*
    - Statusbereich, *57*
  - Steuerung
    - Aktualisieren der RTC, *283*
    - Anzeigen des Zustands, *280*
    - Datum und Uhrzeit des letzten Halts, *280*
    - Ersetzen der aktuellen in der Konfiguration, *74*
    - Status beim Start, Konfigurieren, *91*
    - Unterstützte Typen, *24*
  - Steuerung
    - Anwendung direkt herunterladen, *46*
  - STOPPED-Status
    - Fehlerausweichverhalten, *93*
  - Suchen und Ersetzen, *175*
  - Symbole
    - Adressierung über, *80*
    - Definieren im Fenster „Eigenschaften“, *80*
    - Definieren in Grafikelementen im Kontaktplan-Editor, *81*
    - Liste der verwendeten, *178*
    - Speichern in der Steuerung, *82*
  - Symbolische Adressierung, *80*
  - Symbolliste
    - Anzeigen, *178*
    - Exportieren, *180*
    - Freigeben für Vijeo Designer-Projekte, *181*
    - Importieren, *179*
  - Systemanforderungen, *23*
  - Systembits
    - %S0, *205*
    - %S11, *133*
    - %S14, *280*
    - %S19, *133*
    - %S31, *152*
    - %S38, *152*
    - %S39, *152*
    - %S49, *205*
    - %S51, *205*
  - Systembits/-wörter
    - In Symbolliste, *178*
    - Steuern von Ereignissen mit, *152*
  - Systemobjekte, *168*
  - Systemwörter
    - %SW0, *133*
    - %SW27, *133*
    - %SW30, *133*
    - %SW30...%SW32, *280*
    - %SW31, *133*
    - %SW32, *133*
    - %SW35...%SW38, *280*
    - %SW48, *152*
    - %SW54...%SW57, *280*
    - %SW58, *280*
    - %SW6, *280*
- ## T
- Task
    - Ereignis, *147*
    - Periodisch, *143*
  - Tasks
    - konfigurieren, *95*
  - Tastaturkombinationen, *319*
  - Tastaturkürzel, *60*
  - Testbereich, *188*
  - TH0, TH1
    - Konfigurieren als Ereignisquellen, *148*
  - Timer, Watchdog, *93*

## Tools

- Animationstabellen, *157*
- Antriebsobjekte, *173*
- Eingangs-/Ausgangsobjekte, *169*
- Kommunikationsobjekte, *174*
- Netzwerkobjekte, *170*
- PTO-Objekte, *172*
- Software-Objekte, *171*
- Speichernutzung, *183*
- Speicherobjekte, *163*
- Suchen und Ersetzen, *175*
- Symbollisten, *178*
- Systemobjekte, *168*
- verwenden, *153*

## Tracing

- Anzeigen, *246*
- Auswählen einer Zeitbasis für, *162*
- Auswählen von Objekten für, *157*
- Exportieren in PDF, *248*

## U

## umbenennen

- einer Grafnet-POU, *108*

## Umkehrbarkeit

- Einführung zum, *84*

Umkehrbarkeit Kontaktplan/Anweisungsliste, *84*

## Unterprogramm

- Auslösen der Ausführung über ein Ereignis, *148*
- Der Master-Task, *132*
- implementieren als Freie POU, *99*
- Zuweisen zu periodischer Task, *143*
- Zuweisen zu Tasks, *150*

Unterstützte Geräte, *24*

## V

- Verbinden mit einer Logiksteuerung, *262*

- Verdrahtung von Not-Aus-Sensoren, *205*

## Vergleichsausdruck

- Einfügen in KOP-Programmbausteine, *199*

## Vergleichsbaustein

- Grafikelement für, *194*

## Vergleichsbausteine

- Einfügen von AWL-Ausdrücken, *199*

## Verzweigungsmodi

- Grafikelement, *193*

## Vorlage

- Projekt, *48*
- Speichern der Projektdatei als, *303*

## W

- Watchdog-Timer, Konfigurieren, *93*

## X

## XOR

- Grafikelemente, *194*

## Z

- Zeitbasis (für Tracing), *162*

## Zustand

- Steuerung, Anzeigen, *280*

- Zuweisen von Speicher in der Steuerung, *83*

## Zuweisungsanweisungen

- Einfügen in KOP-Programmbausteine, *200*

- Zuweisungsmodus, *83*

