

# Modicon M258 Logic Controller

Impulsbreitenmodulation

M258 Expert I/O - Bibliothekshandbuch

09/2020

EIO0000004155.00

[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!

© 2020 Schneider Electric. Alle Rechte vorbehalten.



	<b>Sicherheitshinweise</b> .....	5
	<b>Über dieses Buch</b> .....	7
<b>Kapitel 1</b>	<b>Einführung</b> .....	13
	Experten-E/A - Überblick .....	14
	Hinzufügen einer Expertenfunktion .....	17
	Zuordnung der integrierten Experten-E/A .....	20
<b>Kapitel 2</b>	<b>Allgemeines</b> .....	23
	Namenskonventionen für PWM .....	25
	Synchronisierungs- und Aktivierungsfunktionen .....	26
<b>Kapitel 3</b>	<b>Impulsbreitenmodulation (PWM)</b> .....	27
	Beschreibung .....	28
	Konfiguration .....	30
	PWM_M258: Steuern eines Impulsbreitenmodulationssignals .....	32
	Programmierung des PWM-Funktionsbausteins .....	34
<b>Kapitel 4</b>	<b>Frequenzgenerator (FreqGen)</b> .....	37
	Beschreibung .....	38
	Konfiguration .....	39
	Frequency_Generator_M258: Steuern eines Rechteckwellensignals .....	41
	Programmieren .....	43
<b>Anhang</b>	.....	45
<b>Anhang A</b>	<b>Allgemeine Informationen</b> .....	47
	Zweckbestimmte Funktionen .....	48
	Allgemeine Informationen zur Funktionsbausteinverwaltung .....	49
<b>Anhang B</b>	<b>Darstellung von Funktionen und Funktionsbausteinen</b> ..	51
	Unterschiede zwischen einer Funktion und einem Funktionsbaustein .....	52
	Verwenden einer Funktion oder eines Funktionsbausteins in der Sprache AWL .....	53
	Verwendung einer Funktion oder eines Funktionsbausteins in der Sprache ST .....	58
<b>Anhang C</b>	<b>Datentypen</b> .....	61
	EXPERT_ERR_TYPE: Typ der Fehlervariable für einen Expertenfunktionsbaustein .....	61
<b>Glossar</b>	.....	63
<b>Index</b>	.....	67





## 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

Diese Dokumentation soll Sie mit den mit dem Modicon M258 Logic Controller verfügbaren Funktionen Impulsbreitenmodulation und Frequenzgenerator vertraut machen.

In der Dokumentation werden die Datentypen und Funktionen der M258 PWM/FreqGen-Bibliothek beschrieben.

Zur Nutzung dieses Handbuchs müssen Sie:

- mit dem M258, d. h. seinem Design, seiner Funktionalität und seiner Implementierung innerhalb von Steuerungssystemen umfassend vertraut sein.
- die folgenden SPS-Programmiersprachen nach IEC 61131-3 beherrschen:
  - Funktionsbausteindiagramm (FBD)
  - Kontaktplan (KOP/LD - Ladder)
  - Strukturierter Text (ST)
  - Anweisungsliste (AWL/IL - Instruction List)
  - Ablaufsteuerung (SFC)

Nur das Expertenmodul DM72F• kann mit der M258 PWM/FreqGen-Bibliothek verwendet werden.

### Gültigkeitsbereich

Dieses Dokument wurde für EcoStruxure™ Machine Expert V1.2.5 aktualisiert.

Die im vorliegenden Dokument beschriebenen technischen Merkmale sind ebenfalls online verfügbar. Um auf die Online-Informationen zuzugreifen, gehen Sie zur Homepage von Schneider Electric <https://www.se.com/ww/en/download/>.

Die in diesem Dokument 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 Dokument und denen online feststellen, nutzen Sie die Online-Informationen als Referenz.

---

## Weiterführende Dokumentation

Titel der Dokumentation	Referenz-Nummer
Modicon M258 Logic Controller – Programmierhandbuch	<a href="#"><u>EIO0000004135 (ENG)</u></a> <a href="#"><u>EIO0000004136 (FRE)</u></a> <a href="#"><u>EIO0000004137 (GER)</u></a> <a href="#"><u>EIO0000004138 (SPA)</u></a> <a href="#"><u>EIO0000004139 (ITA)</u></a> <a href="#"><u>EIO0000004140 (CHS)</u></a>
Modicon M258 Logic Controller – Hardwarehandbuch	<a href="#"><u>EIO0000004159 (ENG)</u></a> <a href="#"><u>EIO0000004160 (FRE)</u></a> <a href="#"><u>EIO0000004161 (GER)</u></a> <a href="#"><u>EIO0000004162 (SPA)</u></a> <a href="#"><u>EIO0000004163 (ITA)</u></a> <a href="#"><u>EIO0000004164 (CHS)</u></a>

Diese technischen Veröffentlichungen sowie andere technische Informationen stehen auf unserer Website <https://www.se.com/ww/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 Steuerfunktionen sind die Notabschaltung (Not-Aus) und der Nachlauf-Stopp, Stromausfall und Neustart.
- Für kritische Steuerfunktionen 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 Normen

Die technischen Begriffe, Terminologie, Symbole und die entsprechenden Beschreibungen in diesem Handbuch, oder die in beziehungsweise auf den Produkten selbst erscheinen, sind im Allgemeinen von den Begriffen und Definitionen der internationalen Normen hergeleitet.

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/Warmmeldung*, *Fehlermeldung*, *gefährlich/gefahrbringend* usw.

Unter anderem schließen diese Normen ein:

Standard	Beschreibung
IEC 61131-2:2007	Speicherprogrammierbare Steuerungen, Teil 2: Betriebsmittelanforderungen und Prüfungen.
ISO 13849-1:2015	Sicherheit von Maschinen: Sicherheitsspezifische Teile von Steuerungen. Allgemeine Gestaltungsleitsätze.
EN 61496-1:2013	Sicherheit von Maschinen: Berührungslos wirkende Schutzeinrichtung. 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 Ausrüstung von Maschinen - Teil1: Allgemeine Anforderungen
ISO 14119:2013	Sicherheit von Maschinen - Verriegelungseinrichtungen in Verbindung mit trennenden Schutzeinrichtungen - Leitsätze für Gestaltung und Auswahl
ISO 13850:2015	Sicherheit von Maschinen - Not-Halt- Gestaltungsleitsätze
IEC 62061:2015	Sicherheit von Maschinen - Funktionale Sicherheit von sicherheitsbezogenen elektrischen, elektronischen und elektronisch programmierbaren Steuerungen.
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 für sicherheitsbezogene elektrische/elektronische/programmierbare elektronische Systeme.
IEC 61508-3:2010	Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme: Softwareanforderungen.
IEC 61784-3:2016	Industrielle Kommunikationsnetze - Profile - Teil 3: Funktional sichere Übertragung bei Feldbussen - Allgemeine Regeln und Festlegungen für Profile.
2006/42/EC	Maschinenrichtlinie
2014/30/EU	EG-Richtlinie Elektromagnetische Verträglichkeit
2014/35/EU	EG-Richtlinie Niederspannung

---

Zusätzlich kann die in vorliegendem Dokument verwendete Nomenklatur tangential verwendet werden, wenn sie aus anderen Normen abgeleitet ist, wie z.B.:

Standard	Beschreibung
Normenreihe IEC 60034	Drehende elektrische Maschinen
Reihe IEC 61800	Elektrische Leistungsantriebssysteme mit einstellbarer Drehzahl
Serie IEC 61158	Digitale Datenkommunikation in der Leittechnik – Feldbus für industrielle Leitsysteme

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. Weitere Informationen über die einzelnen anwendbaren Normen die hier beschriebenen Produkte betreffend, entnehmen Sie den entsprechenden Tabellen dieser Produktbezeichnungen.



---

# Kapitel 1

## Einführung

---

### Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Experten-E/A - Überblick	14
Hinzufügen einer Expertenfunktion	17
Zuordnung der integrierten Experten-E/A	20

## Experten-E/A - Überblick

### Einführung

Das Steuerungsgrundgerät bietet Folgendes:

- 2 integrierte Experten-E/A-Module (DM72F0 und DM72F1) mit:
  - 5 Schnelleingänge
  - 2 Standardeingänge
  - 2 Schnellausgänge
- Stromverteilermodul für die Steuerung (CPDM)

Jedes integrierte Experten-E/A-Modul (DM72F•) kann Expertenfunktionen (*siehe Seite 17*) unterstützen.

### Konfiguration von integrierten Experten-E/A

Um die Experten-E/A zu konfigurieren, doppelklicken Sie auf den Knoten **Expert** in der **Gerätebaumstruktur**.

Die nachstehende Abbildung zeigt die Registerkarte zur Konfiguration:

Parameter	Typ	Wert	Standardwert	Einheit	Beschreibung
Run/Stop-Eingang	Enumeration von BYTE	BLOCK0_I0	Kein		
Alarmausgang	Enumeration von BYTE	BLOCK0_Q1	Kein		
Ausgangsmodus erneut aktivieren	Enumeration von BYTE	Auto	Auto		

Die nachstehende Tabelle zeigt die Funktion der verschiedenen Parameter:

Parameter	Funktion
Run/Stop Input	Definieren eines Eingangs, der als Run/Stop-Eingang ( <i>siehe Seite 15</i> ) dienen soll.
Alarm Output	Definieren eines Ausgangs, der als Alarmausgang ( <i>siehe Seite 15</i> ) dienen soll.
Rearming Output Mode	Definieren des Modus zur erneuten Aktivierung des Ausgangs ( <i>siehe Seite 16</i> ).

## Run/Stop-Eingang

Die folgende Tabelle zeigt die verschiedenen Zustände:

Eingangszustand	Ergebnis
Zustand 0	Hält die Steuerung an und ignoriert den externen Run-Befehl.
Steigende Flanke	Veranlasst im Status STOPPED den Start einer Anwendung im RUNNING-Status.
Zustand 1	Die Anwendung kann durch Folgendes gesteuert werden: <ul style="list-style-type: none"> <li>● EcoStruxure Machine Expert (Run/Stop)</li> <li>● Über die Anwendung (Steuerungsbefehl)</li> <li>● Über einen Netzwerkbefehl</li> </ul>

**HINWEIS:** Der Run/Stop-Eingang wird auch dann verwaltet, wenn die Option **E/A STOP-Zustand aktualisieren** auf der Registerkarte "SPS-Einstellungen" (*siehe Modicon M258 Logic Controller, Programmierhandbuch*) nicht aktiviert ist.

Eingänge, die konfigurierten Expertenfunktionen zugewiesen sind, können nicht als Run/Stop konfiguriert werden.

Weitere Informationen zu Steuerungsstatus und Statusübergängen finden Sie im SPS-Zustandsdiagramm (*siehe Modicon M258 Logic Controller, Programmierhandbuch*).

⚠ <b>WARNUNG</b>
<p><b>UNBEABSICHTIGTER MASCHINEN- ODER PROZESSSTART</b></p> <ul style="list-style-type: none"> <li>● Überprüfen Sie den Sicherheitsstatus Ihrer Maschinen- bzw. Prozessumgebung, bevor Sie den Run/Stop-Eingang unter Spannung setzen.</li> <li>● Verwenden Sie den Run/Stop-Eingang, um den unbeabsichtigten Start von einem entfernten Standort aus zu verhindern.</li> </ul> <p><b>Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.</b></p>

## Alarmausgang

Dieser Ausgang ist auf logisch 1 gesetzt, wenn sich die Steuerung im RUNNING-Status befindet und das Anwendungsprogramm nicht am Haltepunkt angehalten wurde.

Ausgänge, die konfigurierten Expertenfunktionen zugewiesen sind, können nicht als Alarmausgang konfiguriert werden.

### HINWEIS:

Der Alarmausgang wird in folgenden Fällen auf 0 gesetzt:

- Wenn eine Task an einem Haltepunkt angehalten wird; der Alarmausgang signalisiert, dass die Steuerung die Ausführung der Anwendung gestoppt hat.
- Wenn in Verbindung mit den Experten-E/A ein Fehler erkannt wird (Spannungsunterbrechung, Kurzschlusserkennung).

### Ausgangsmodus erneut aktivieren

Die Schnellausgänge von DM72F\*-Modulen nutzen die Push/Pull-Technologie. Wenn ein Fehler erkannt wird, (Kurzschluss oder Übertemperatur), wird der Ausgang in den Tri-State versetzt, und der Zustand wird über ein Statusbit (DM72F\* Kanal IB1.0) und PLC\_R.i\_wLocalIOStatus (siehe Modicon M258 Logic Controller Systemfunktionen und Variablen – PLCSystem-Bibliothekshandbuch).

Es sind zwei Verhaltensweisen möglich:

- **Automatisches Wiedereinschalten:** Sobald der erkannte Fehler behoben wurde, wird der Ausgang erneut in den Zustand versetzt, der dem derzeit zugewiesenen Wert entspricht, und der Diagnosewert wird zurückgesetzt.
- **Manuelles Wiedereinschalten:** Wenn ein Fehler erkannt wird, wird der Zustand gespeichert und der Ausgang in den Tri-State forciert, bis der Benutzer den Zustand manuell aufhebt (siehe E/A-Abbild-Kanal).

Im Falle einer Überlast oder eines Kurzschlusses wird die Gruppe von Ausgängen automatisch gemeinsam in den Temperaturschutzmodus gesetzt (alle Ausgänge in der Gruppe werden auf 0 gesetzt) und dann in regelmäßigen Abständen (jede Sekunde) erneut aktiviert, um den Verbindungsstatus zu testen. Dabei werden allerdings Kenntnisse über die Auswirkungen einer Reaktivierung auf die Maschine und die gesteuerten Prozesse vorausgesetzt.

## **WARNUNG**

### **UNBEABSICHTIGTER MASCHINENSTART**

Unterbinden Sie das automatische Wiedereinschalten der Ausgänge, falls dieses Verhalten für die Maschine oder den Prozess nicht wünschenswert ist.

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



## Hinzufügen einer Expertenfunktion

### Einführung

Alle DM72F\*-Expertenmodule können Expertenfunktionen unterstützen. Expertenfunktionen werden als einfach oder als komplex definiert. Pro Modul kann jeweils nur ein Typ konfiguriert werden:

- Einfache Funktionen:
  - Hochgeschwindigkeitszähler im Simple-Modus (Einfach)
  - E/A-Ereignisspeicherung
- Komplexe Funktionen:
  - Hochgeschwindigkeitszähler im Main-Modus (Haupt)
  - Encoder
  - Frequenzgenerator (FreqGen)
  - Impulsbreitenmodulation (PWM)

Wenn ein E/A nicht von einer Expertenfunktion verwendet wird, kann er als Standard-E/A eingesetzt werden.

#### HINWEIS:

- Wenn ein Standard-E/A als Run/Stop eingesetzt wird, kann er nicht von einer Expertenfunktion verwendet werden.
- Wenn ein Standard-E/A als Alarm eingesetzt wird, kann er nicht von einer Expertenfunktion verwendet werden.

Weitere Einzelheiten finden Sie unter Konfiguration integrierter Experten-E/A (*siehe Seite 14*).

### Hinzufügen einer Expertenfunktion

Wenn Sie Ihrer Steuerung eine Expertenfunktion (Ereignisspeicherung, HSC PWM oder Frequenzgenerator) hinzufügen möchten, wählen Sie die gewünschte Expertenfunktion im **Hardwarekatalog** aus, ziehen Sie sie in die **Gerätebaumstruktur** und legen Sie sie auf einem der optisch hervorgehobenen Knoten ab.

Weitere Informationen zum Hinzufügen von Geräten in einem Projekt finden Sie unter:

- Verwenden der Methode Drag&Dop (*siehe EcoStruxure Machine Expert, Programming Guide*) (Ziehen und Ablegen)
- Verwenden der Kontextmenüs oder Plus-Schaltflächen (*siehe EcoStruxure Machine Expert, Programming Guide*)

Wenn Sie Ihrer Steuerung eine Encoderfunktion hinzufügen möchten, wählen Sie das **Standardgebermodul** im **Hardwarekatalog** aus, ziehen Sie das Modul in die **Gerätebaumstruktur** und legen Sie es dann auf einem der optisch hervorgehobenen Knoten ab.

Die folgenden Expertenfunktionen können verwendet werden:

Funktion	Beschreibung	Siehe...
Event_Latch	Mithilfe der Event_Latch-Funktion zur Ereignisspeicherung können integrierte Expert-Eingänge als Ereignis oder als Statusspeicher konfiguriert werden.	Konfiguration der Ereignisspeicherung (siehe <i>Modicon M258 Logic Controller, Programmierhandbuch</i> )
HSC	HSC-Funktionen können besonders schnell Impulse zählen, die von Sensoren, Gebern, Schaltern usw. ausgehen, die an einen dedizierten schnellen Eingang angeschlossen sind.	M258 HSC-Bibliothek (siehe <i>Modicon M258 Logic Controller, Hochgeschwindigkeitszählung, M258 Expert I/O - Bibliothekshandbuch</i> )
PWM Frequenzgenerator	Die PWM-Funktion generiert ein Rechteckwellen-Signal auf zweckbestimmten Ausgangskanälen mit variablem Arbeitszyklus. Die Frequenzgenerator-Funktion generiert ein Rechteckwellen-Signal auf zweckbestimmten Ausgangskanälen mit festem Arbeitszyklus (50 %).	M258 PWM-Bibliothek
Encoder	Ziel dieser Funktion ist das Anschließen eines Gebers für die Erfassung einer Position. Die Funktion kann an einer Schnittstelle für integrierte Experten-E/A installiert werden und nur einen Inkrementalgeber unterstützen. Sie können eine lineare Achse oder eine Drehachse definieren.	M258 HSC-Bibliothek (siehe <i>Modicon M258 Logic Controller, Hochgeschwindigkeitszählung, M258 Expert I/O - Bibliothekshandbuch</i> )

### Zuweisung von Expertenfunktionen

Zuweisung von Expertenfunktionen entsprechend der Schnittstelle (Spalten schließen sich gegenseitig aus):

I/F-Schnittstelle	Expertenfunktionen				
	Einfache Funktionen: ● Schnelle E/A: Ereignis oder mit Statusspeicher ● HSC Simple (Einfach)	HSC_Main (Haupt)	Encoder	PWM	Frequenzgenerator
DM72F0	Bis zu 4	1	1	1	1
DM72F1	Bis zu 4	1	1	1	1

Weitere Informationen finden Sie unter Experten-E/A-Zuordnung (*siehe Seite 20*).

## Expertenfunktions-E/A in Standard-E/A

Expertenfunktions-E/A in Standard-E/A:

- Eingänge können über eine Standardspeichervariable gelesen werden, auch wenn sie als Expertenfunktion konfiguriert sind.
- Ein Eingang kann nicht als Expertenfunktion konfiguriert werden, wenn er bereits als Run/Stop-Eingang konfiguriert wurde.
- Ein Ausgang kann nicht als Expertenfunktion konfiguriert werden, wenn er bereits als Alarmausgang konfiguriert wurde.
- %Q hat keine Wirkung auf einen Reflexausgang.
- Die Kurzschlussverwaltung gilt dennoch für alle Ausgänge. Der Status der Ausgänge ist verfügbar.
- Alle E/A, die nicht von Expertenfunktionen verwendet werden, sind als Schnell- oder Standard-E/A verfügbar.

Wenn Eingänge in einer Expertenfunktion (Latch, HSC usw.) verwendet werden, wird der Integratorfilter durch einen Antiprellfilter (*siehe Modicon M258, Logic Controller, Hardwarehandbuch*) ersetzt. Der Filterwert wird im Fenster für die Expertenfunktion konfiguriert.

## Zuordnung der integrierten Experten-E/A

### E/A-Zuordnung für Expertenfunktionen für DM72F•

Zuordnung der integrierten Experten-E/A nach Expertenfunktion:

		I0	I1	I2	I3	I4	I5	Q0	Q1
Event_Latch 0/4	Eingang	M							
Event_Latch 1/5	Eingang		M						
Event_Latch 2/6	Eingang			M					
Event_Latch 3/7	Eingang				M				
HSC Simple 0/4	Eingang A	M							
HSC Simple 1/5	Eingang A		M						
HSC Simple 2/6	Eingang A			M					
HSC Simple 3/7	Eingang A				M				
HSC Main 0/1	Eingang A	M							
	Eingang B		C						
	SYNC			C					
	CAP				C				
	EN					C			
	REF						C		
PWM 0/1	Ausgänge							C	C
	SYNC			C					
	EN					C			
Frequenzgenerator 0/1	Ausgänge							M	
	SYNC			C					
	EN					C			
Standardgeber	Eingang A	M							
	Eingang B		M						
	SYNC			C					
	CAP				C				
	EN					C			
	REF						C		
	Ausgänge							C	C
<b>M</b> Zwingend <b>C</b> Abhängig von der Konfiguration									

## E/A-Zusammenfassung

Im Fenster **E/A-Zusammenfassung** werden die DM72F•-E/A und die von den Expertenfunktionen verwendeten E/A angezeigt.

Der Zugriff auf das Fenster **E/A-Zusammenfassung** erfolgt über die **DM72F•**-Knoten:

Schritt	Aktion
1	Erweitern Sie den Knoten <b>Expert</b> in der <b>Gerätebaumstruktur</b> .
2	Klicken Sie mit der rechten Maustaste auf <b>DM72F•</b> und wählen Sie im Kontextmenü die Option <b>E/A-Zusammenfassung</b> aus.

Beispiel einer E/A-Zusammenfassung:

The screenshot shows a software window titled 'E/A-Zusammenfassung' with two main sections: 'Eingänge' (Inputs) and 'Ausgänge' (Outputs). Each section contains a table with columns for 'Kanal' (Channel), 'Adresse' (Address), and 'Nutzung' (Usage). A 'Schließen' (Close) button is located at the bottom right of the window.

Eingänge			Ausgänge		
Kanal	Adresse	Nutzung	Kanal	Adresse	Nutzung
DM72F0 - I0	%IX1.0	HSCMain_1 - Eingang A, DM72F0 - Filter	DM72F0 - Q0	%QX0.0	HSCMain_1 - Reflexausgang 0
DM72F0 - I1	%IX1.1	DM72F0 - Filter	DM72F0 - Q1	%QX0.1	HSCMain_1 - Reflexausgang 1
DM72F0 - I2	%IX1.2	HSCMain_1 - SYNC, DM72F0 - Filter	DM72F1 - Q0	%QX1.0	HSCMain - Reflexausgang 0
DM72F0 - I3	%IX1.3	HSCMain_1 - CAP, DM72F0 - Filter	DM72F1 - Q1	%QX1.1	HSCMain - Reflexausgang 1
DM72F0 - I4	%IX1.4	HSCMain_1 - EN, DM72F0 - Filter			
DM72F0 - I5	%IX1.5	DM72F0 - Filter			
DM72F0 - I6	%IX1.6	DM72F0 - Filter			
DM72F0 - I0	%IX2.0	DM72F0 - Tastenkombinationerkennung			
DM72F1 - I0	%IX3.0	HSCMain_1 - Eingang A, DM72F1 - Filter			
DM72F1 - I1	%IX3.1	DM72F1 - Filter			
DM72F1 - I2	%IX3.2	HSCMain_1 - SYNC, DM72F1 - Filter			
DM72F1 - I3	%IX3.3	HSCMain_1 - CAP, DM72F1 - Filter			
DM72F1 - I4	%IX3.4	HSCMain_1 - EN, DM72F1 - Filter			
DM72F1 - I5	%IX3.5	DM72F1 - Filter			
DM72F1 - I6	%IX3.6	DM72F1 - Filter			
DM72F1 - I0	%IX4.0	DM72F1 - Tastenkombinationerkennung			



---

# Kapitel 2

## Allgemeines

---

### Überblick

Dieses Kapitel enthält allgemeine Informationen zu den Funktionen ) und Impulsbreitenmodulation (PWM).

Diese Funktionen stellen einfache und dabei leistungsstarke Lösungen für Ihre Anwendung bereit. Sie erweisen sich insbesondere für die Bewegungssteuerung als nützlich. Die Nutzung und Anwendung der enthaltenen Informationen setzt allerdings Fachkenntnisse in Bezug auf die Konzeption und Programmierung automatisierter Steuerungssysteme voraus. Nur Sie als Benutzer, Maschinenbauer oder -integrator sind mit allen Bedingungen und Faktoren vertraut, die bei der Installation, der Einrichtung, dem Betrieb und der Wartung der Maschine bzw. der entsprechenden Prozesse zum Tragen kommen. Demzufolge sind allein Sie in der Lage, die Automatisierungskomponenten und zugehörigen Betriebsmittel sowie die angemessenen Sicherheitsvorkehrungen und Sperrvorrichtungen zu identifizieren, die einen effektiven und störungsfreien Betrieb gewährleisten. Beachten Sie bei der Auswahl der Automatisierungs- und Steuerungskomponenten sowie aller zugehörigen Betriebsmittel und Software alle geltenden örtlichen, regionalen und landesspezifischen Normen und/oder Vorschriften.

### **WARNUNG**

#### **INKOMPATIBILITÄT MIT REGULATORISCHEN VORSCHRIFTEN**

Stellen Sie sicher, dass alle eingesetzten Betriebsmittel und entworfenen Systeme die anwendbaren lokalen, regionalen und nationalen Vorschriften und Normen erfüllen.

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

Die mit der Bibliothek der Expertenfunktionen bereitstehenden Funktionen wurden unter der Annahme entwickelt und entworfen, dass Sie die erforderliche Sicherheitshardware in die Anwendungsarchitektur einbauen, einschließlich, jedoch nicht beschränkt auf angemessene Hardware-Grenzwertschalter und -Notausschalter sowie Regelkreise. Es wird implizit davon ausgegangen, dass Sie bei der Konzeption Ihrer Maschine funktionale Sicherheitsvorkehrungen getroffen haben, um ein unerwünschtes Verhalten der Maschine zu verhindern, beispielsweise Überfahren oder andere Arten unkontrollierter Bewegungen. Darüber hinaus wird davon ausgegangen, dass Sie eine für Ihre Maschine bzw. Ihren Prozess geeignete funktionale Sicherheits- und Risikoanalyse durchgeführt haben.

## **WARNUNG**

### **UNBEABSICHTIGTER GERÄTEBETRIEB**

Stellen Sie sicher, dass bei der Konzeption Ihrer Maschine eine Risikoanalyse nach EN/ISO 12100 durchgeführt und respektiert wird.

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

### **Inhalt dieses Kapitels**

Dieses Kapitel enthält die folgenden Themen:

<b>Thema</b>	<b>Seite</b>
Namenskonventionen für PWM	25
Synchronisierungs- und Aktivierungsfunktionen	26



## Namenskonventionen für PWM

### Definition

In diesem Dokument gelten folgende Namenskonventionen:

Name	Beschreibung
SYNC	Synchronisierungsfunktion ( <i>siehe Seite 26</i> ).
EN	Aktivierungsfunktion ( <i>siehe Seite 26</i> ).
IN_SYNC	Physischer Eingang für die SYNC-Funktion.
IN_EN	Physischer Eingang für die EN-Funktion.
OUT_PWM	Physischer Ausgang für die PWM-Funktion.

## Synchronisierungs- und Aktivierungsfunktionen

### Einführung

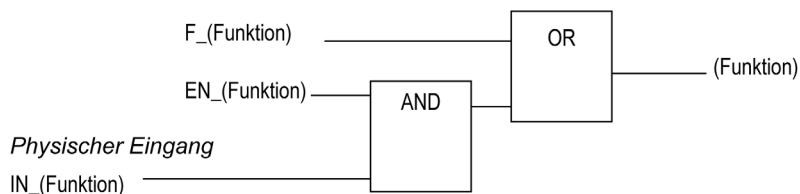
In diesem Abschnitt werden die von der PWM verwendeten Funktionen beschrieben:

- Funktion **Synchronisierung**
- Funktion **Aktivierung** (Enable)

Jede Funktion verwendet die zwei folgenden Funktionsbaustein-Bits:

- **EN\_(Funktion)-Bit:** Wird dieses Bit auf 1 gesetzt, kann diese (Funktion) an einem externen physikalischen Eingang ausgeführt werden, sofern konfiguriert.
- **F\_(Funktion)-Bit:** Wird dieses Bit auf 1 gesetzt, wird die (Funktion) forciert.

Die folgende Abbildung beschreibt, wie die Funktion verwaltet wird:



**HINWEIS:** (Funktion) steht für **Enable** (für die Aktivierungsfunktion) oder **Sync** (für die Synchronisierungsfunktion).

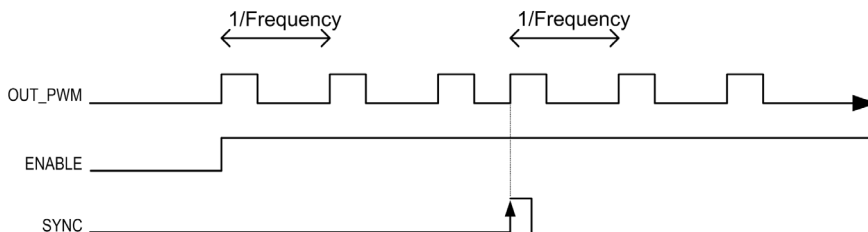
Wenn der physikalische Eingang erforderlich ist, aktivieren Sie ihn in der Konfiguration (*siehe Seite 31*).

### Synchronisierungsfunktion

Die Funktion zur **Synchronisierung** (Sync) dient der Unterbrechung des aktuellen PWM-Zyklus und dem Start eines neuen Zyklus.

### Aktivierungsfunktion

Die **Enable**-Funktion wird zur Aktivierung der PWM verwendet:



---

# Kapitel 3

## Impulsbreitenmodulation (PWM)

---

### Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Beschreibung	28
Konfiguration	30
PWM_M258: Steuern eines Impulsbreitenmodulationssignals	32
Programmierung des PWM-Funktionsbausteins	34

## Beschreibung

### Überblick

Die PWM-Funktion (Pulse Width Modulation, Impulsbreitenmodulation) generiert ein programmierbares Impulswellensignal auf zweckbestimmten Ausgängen mit einem anpassbaren Arbeitszyklus und einer anpassbaren Frequenz.

**HINWEIS:** Die Funktion muss aktiviert werden, indem `F_Enable` auf 1 gesetzt wird, oder durch ein externes Ereignis, wobei der Eingang `IN_EN` und `EN_Enable = 1` sein muss. Andernfalls bleibt der Ausgang (`OUT_PWM`) auf 0.

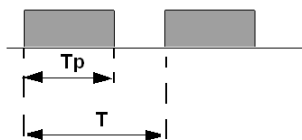
Wenn eine Impulsbreitenmodulation (PWM, Pulse Width Modulation) auf einem Experten-E/A-Modul konfiguriert ist, können keine weiteren Funktionen hinzugefügt werden (*siehe Seite 14*).

### Signalform

Die Signalform hängt von folgenden Eingangsparametern ab:

- **Frequenz**, konfigurierbar von 0,1 Hz bis 20 kHz in Schritten zu je 0,1 Hz
- **Arbeitszyklus** des Ausgangssignals von 0 % bis 100 %

Arbeitszyklus =  $T_p/T$



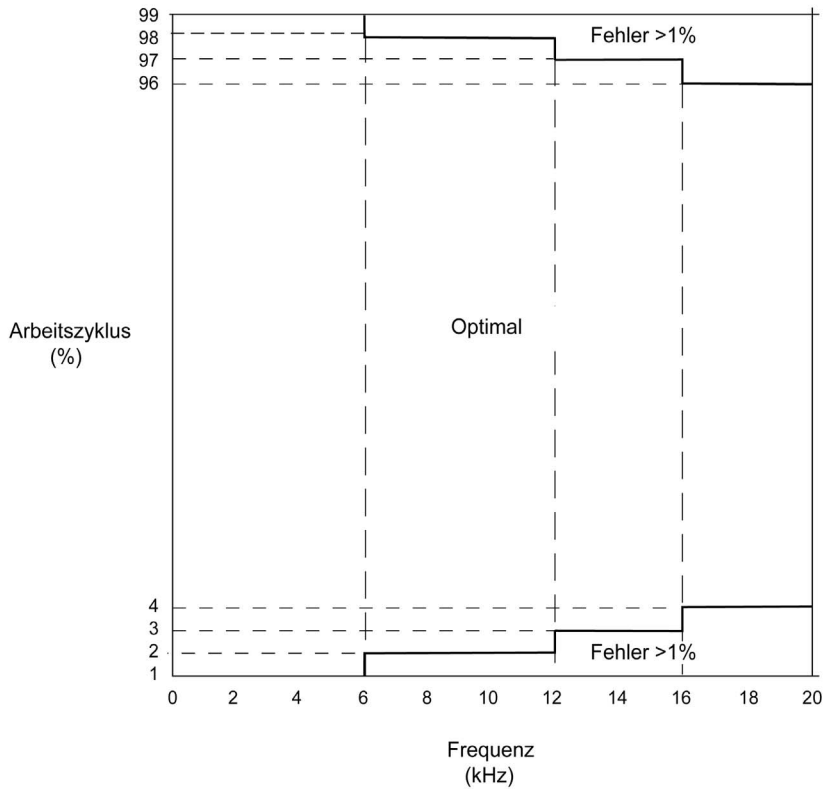
**$T_p$**  Impulsbreite

**$T$**  Impulsdauer (1/Frequenz)

Wenn der Arbeitszyklus im Programm geändert wird, wird die Breite des Signals moduliert. Die nachfolgende Abbildung zeigt ein Ausgangssignal mit variierenden Arbeitszyklen.



Der Arbeitszyklus entspricht einer Genauigkeit von 1 % für jeden Schritt von 80 Hz. Wenn der Arbeitszyklus unter 4 % oder über 96 % liegt, liegt die Abweichung je nach Frequenz über 1 %, wie in der nachstehenden Grafik zu sehen ist:



## Konfiguration

### Überblick

In der Steuerung können 2 Funktionen zur Impulsbreitenmodulation konfiguriert werden.

### Hinzufügen einer Funktion zur Impulsbreitenmodulation

Wenn Sie Ihrer Steuerung eine Funktion zur Impulsbreitenmodulation hinzufügen möchten, wählen Sie im **Hardwarekatalog** den Eintrag **PWM** aus, ziehen Sie ihn in die **Gerätebaumstruktur** und legen Sie ihn dort auf einem der optisch hervorgehobenen Knoten ab.

Weitere Informationen zum Hinzufügen von Geräten in einem Projekt finden Sie unter:

- Verwenden der Methode Drag&Dop (*siehe EcoStruxure Machine Expert, Programming Guide*) (Ziehen und Ablegen)
- Verwenden der Kontextmenüs oder Plus-Schaltflächen (*siehe EcoStruxure Machine Expert, Programming Guide*)

### Zugreifen auf die Parameter

Gehen Sie vor wie folgt, um auf die Parameter der PWM-Funktion zuzugreifen:

Schritt	Aktion
1	Doppelklicken Sie in der <b>Gerätebaumstruktur</b> auf <b>MyController</b> → <b>Expert</b> → <b>DM72Fx</b> → <b>PWM</b> .
2	Wählen Sie die Registerkarte <b>PWM-Konfiguration</b> aus.

## Parameter

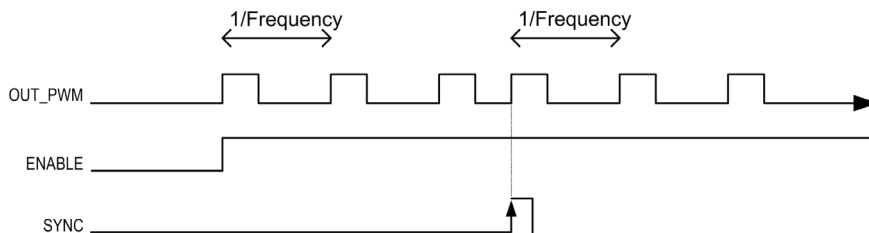
Die Funktion zur Impulsbreitenmodulation verfügt über folgende Parameter:

Parameter	Wert	Standardwert	Beschreibung	
<b>SYNC-Eingang</b>	<b>Position</b>	Deaktiviert 11.2	Deaktiviert	Wählen Sie den zur Voreinstellung der Impulsgenerator-Funktion verwendeten SPS-Eingang aus.
	<b>Prelfilter</b>	0,002 0,004 0,012 0,04 0,12 0,4 1,2 4	0,002	Legen Sie den Filterwert zur Reduzierung des Premeffekts am SYNC-Eingang fest.
	<b>SYNC-Flanke</b>	Steigend Fallend Beide	Steigend	Wählen Sie die Bedingung zur Voreinstellung der Impulsgenerator-Funktion über den SYNC-Eingang aus.
<b>EN-Eingang</b>	<b>Position</b>	Deaktiviert 11.4	Deaktiviert	Wählen Sie den zur Aktivierung der Impulsgenerator-Funktion verwendeten SPS-Eingang aus.
	<b>Prelfilter</b>	0,002 0,004 0,012 0,04 0,12 0,4 1,2 4	0,002	Legen Sie den Filterwert zur Reduzierung des Premeffekts am EN-Eingang fest.

## Synchronisierung mit einem externen Ereignis

Bei einer steigenden Flanke am physischen Eingang IN\_SYNC (mit EN\_Sync = 1) wird der aktuelle Zyklus unterbrochen und die PWM-Funktion startet einen neuen Zyklus.

Die folgende Abbildung zeigt ein Impulsdiagramm für den Funktionsbaustein `Pulse Width Modulation` unter Verwendung des Eingangs IN\_SYNC:



## PWM\_M258: Steuern eines Impulsbreitenmodulationssignals

### Überblick

Der Funktionsbaustein Pulse Width Modulation steuert einen Impulsbreitenmodulation-Signalausgang mit der angegebenen Frequenz und dem angegebenen Arbeitszyklus.

### Grafische Darstellung



### Darstellung in IL (AWL) und ST

Sie finden eine allgemeine Darstellung in AWL- oder ST-Sprache im Kapitel *Unterschiede zwischen Funktionen und Funktionsbausteinen (siehe Seite 52)*.

### Eingangsvariablen

In der nachstehenden Tabelle werden die Eingangsvariablen beschrieben:

Eingänge	Typ	Kommentar
EN_Enable	BOOL	TRUE: Erlaubt die Aktivierung der Impulsbreitenmodulation (PWM, Pulse Width Modulation) über den Eingang IN_EN (sofern konfiguriert).
F_Enable	BOOL	TRUE: Forciert die Aktivierungsfunktion.
EN_SYNC	BOOL	TRUE: Erlaubt den Neustart über den Eingang IN_Sync des internen Zeitgebers mit Bezug zur Zeitbasis (sofern konfiguriert).
F_SYNC	BOOL	Forciert an der steigenden Flanke einen Neustart des internen Zeitgebers mit Bezug zur Zeitbasis.
Frequency	DWORD	Frequenz des Pulse Width Modulation-Ausgangssignals in Zehntel Hz (Bereich: 1 (0,1 Hz) bis 200.000 (20 kHz)).
Duty	BYTE	Arbeitszyklus des Pulse Width Modulation-Ausgangssignals in % (Bereich: 0 bis 100).



## Ausgangsvariablen



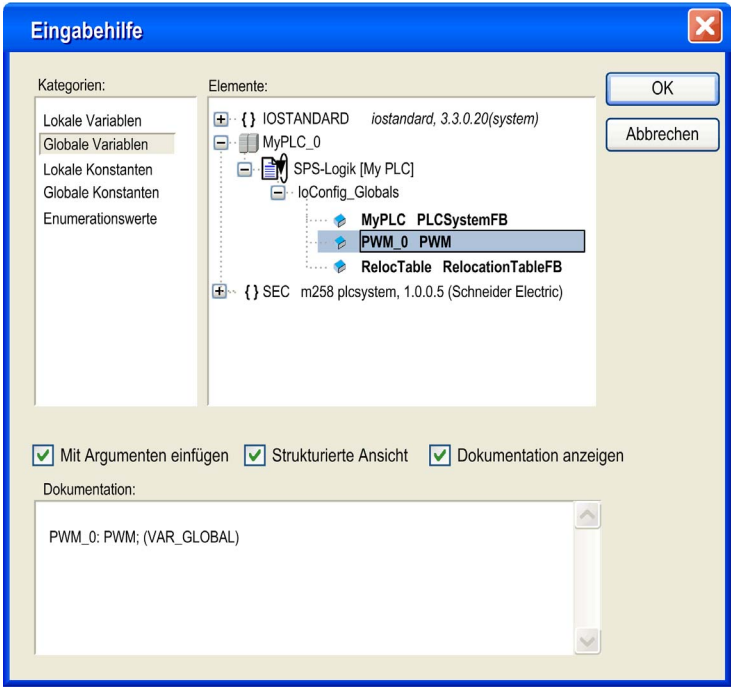
In der nachstehenden Tabelle werden die Ausgangsvariablen beschrieben:

Ausgänge	Typ	Kommentar
InFrequency	BOOL	TRUE: Das Pulse Width Modulation-Signal wird gerade mit der vorgegebenen Frequenz und dem vorgegebenen Arbeitszyklus ausgegeben.
Busy	BOOL	<p>Busy wird verwendet, um anzugeben, dass eine Befehlsänderung stattfindet: Die Frequenz wird geändert.</p> <p>Wird auf TRUE gesetzt, wenn der Enable-Befehl gesetzt und das Frequenzgenerator-Signal nicht mit der angegebenen Frequenz ausgegeben wird.</p> <p>Wird auf FALSE zurückgesetzt, wenn InFrequency oder Error gesetzt oder der Aktivierungsbefehl zurückgesetzt wurde.</p> <p>Wenn die Ausführung einer Befehlsänderung sofort erfolgt, bleibt "Busy" auf FALSE.</p>
Fehler	BOOL	TRUE: Gibt an, dass ein Fehler erkannt wurde.
ErrID	EXPERT_ERR_TYPE (siehe Seite 61)	Wenn Error gesetzt ist: Typ des erkannten Fehlers.

## Programmierung des PWM-Funktionsbausteins

### Vorgehensweise

So programmieren Sie einen **PWM-Funktionsbaustein**:

Schritt	Aktion
1	Wählen Sie die Registerkarte <b>Bibliotheken</b> im <b>Softwarekatalog</b> aus und klicken Sie auf <b>Bibliotheken</b> . Wählen Sie <b>Steuerung</b> → <b>M258</b> → <b>M258 Expert IO</b> → <b>OutputGenerator</b> → <b>PWM_M258</b> in der Liste aus und ziehen Sie das Element in das Fenster <b>POU</b> .
2	<p>Wählen Sie die Instanz des Funktionsbausteins durch einen Klick auf  auf  auf. Das Dialogfeld <b>Eingabehilfe</b> wird angezeigt. Wählen Sie die globale Variable mit Bezug auf die bei der Konfiguration hinzugefügte PWM (<i>siehe Seite 30</i>) aus und bestätigen Sie Ihre Auswahl.</p>  <p><b>HINWEIS:</b> Wenn die Funktionsbausteininstanz nicht sichtbar ist, prüfen Sie die Konfiguration der PWM-Funktion.</p>
3	Die Eingänge/Ausgänge sind im Funktionsbaustein ( <i>siehe Seite 32</i> ) enthalten. Eine Beschreibung der Interaktion zwischen den Eingängen/Ausgängen findet sich unter Allgemeine Informationen ( <i>siehe Seite 47</i> ).

### Programmierbeispiel

Die folgende Abbildung zeigt das Beispiel eines programmierten Funktionsbausteins PWM\_M258.

```
PROGRAM POU_1
```

```
VAR
```

```
MyVar1: BOOL;
```

```
MyVar2: BOOL;
```

```
MyVar3: EXPERT_ERR_TYPE;
```

```
MyVar4: BOOL;
```

```
MyVar5: BOOL;
```

```
MyVar6: BOOL;
```

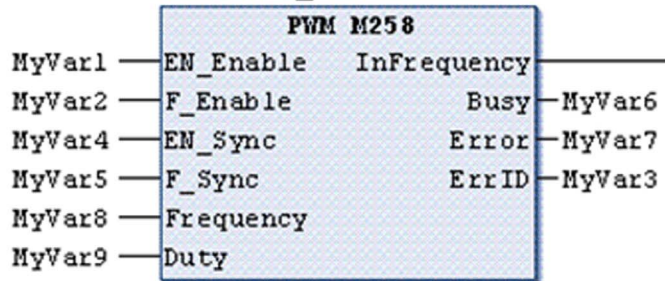
```
MyVar7: BOOL;
```

```
MyVar8: DWORD;
```

```
MyVar9: BYTE;
```

```
END_VAR
```

IoConfig\_Globals.PWM00





---

# Kapitel 4

## Frequenzgenerator (FreqGen)

---

### Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Beschreibung	38
Konfiguration	39
Frequency_Generator_M258: Steuern eines Rechteckwellensignals	41
Programmieren	43

## Beschreibung

### Überblick

Die FG-Funktion (Frequenzgenerator) generiert ein Rechteckwellen-Signal auf zweckbestimmten Ausgangskanälen mit festem Arbeitszyklus (50 %).

**Frequenz** , konfigurierbar von 0,1 Hz bis 100 kHz in Schritten von 0,1 Hz.

Wenn ein Frequenzgenerator auf einem Experten-E/A-Modul konfiguriert wird, können keine weiteren Funktionen hinzugefügt werden (*siehe Seite 14*).

## Konfiguration

### Überblick

In der Steuerung können 2 Frequenzgenerator-Funktionen konfiguriert werden.

### Hinzufügen einer Frequenzgenerator-Funktion

So fügen Sie eine Frequenzgenerator-Funktion hinzu:

Schritt	Aktion
1	Wählen Sie <b>FreqGen</b> im <b>Hardwarekatalog</b> aus, ziehen Sie das Element in die <b>Gerätebaumstruktur</b> und legen Sie es auf einem der optisch hervorgehobenen Knoten ab. Weitere Informationen zum Hinzufügen von Geräten in einem Projekt finden Sie unter: <ul style="list-style-type: none"> <li>• Verwenden der Methode Drag&amp;Dop (<i>siehe EcoStruxure Machine Expert, Programming Guide</i>) (Ziehen und Ablegen)</li> <li>• Verwenden der Kontextmenüs oder Plus-Schaltflächen (<i>siehe EcoStruxure Machine Expert, Programming Guide</i>)</li> </ul>
2	Doppelklicken Sie auf <b>FreqGen</b> und öffnen Sie die Registerkarte <b>Konfiguration des Frequenzgenerators</b> .
3	Stellen Sie die Parameter wie in nachfolgender Tabelle gezeigt ein:
4	Doppelklicken Sie auf den Knoten <b>Impulsgeneratoren</b> Ihrer Steuerung in der <b>Gerätebaumstruktur</b> .
5	Doppelklicken Sie auf den Wert <b>Funktion zur Impulsgenerierung</b> und wählen Sie <b>FreqGen</b> aus. <b>Ergebnis:</b> Die Konfigurationsparameter des Frequenzgenerators werden angezeigt.

### Parameter

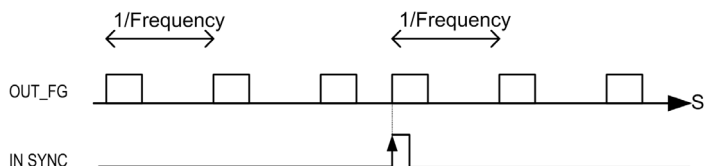
Die Frequenzgenerator-Funktion verfügt über folgende Parameter:

Parameter		Wert	Typ	Beschreibung
SYNC-Eingang	Position	Deaktiviert (Standard) I2	BOOL	Aktiviert den physischen Eingang IN_SYNC zur Verwendung für die Synchronisierung.
	Prellfilter	0,002 (Standard) 0,004 0,012 0,04 0,12 0,4 1,2 4	ENUM	Definiert den Wert für den IN_SYNC-Filterwert.
	SYNC-Flanke	Steigend Fallend Beide	ENUM	Definiert die Flanke des IN_SYNC-Eingangs, an der die Synchronisierung stattfindet.
EN-Eingang	Position	Deaktiviert (Standard) I4	BOOL	Ermöglicht die Verwendung des physischen Eingangs IN_EN zur Aktivierung der Funktion.
	Prellfilter	0,002 (Standard) 0,004 0,012 0,04 0,12 0,4 1,2 4	ENUM	Definiert den Filterwert des Eingangs IN_EN.

### Synchronisierung mit einem externen Ereignis

Bei einer steigenden Flanke am physischen Eingang IN\_SYNC (mit EN\_Sync = 1) wird der aktuelle Zyklus unterbrochen und die FreqGen-Funktion startet einen neuen Zyklus.

Die folgende Abbildung zeigt ein Impulsdiagramm für den Frequenzgenerator-Funktionsbaustein unter Verwendung des Eingangs IN\_SYNC:





## Frequency\_Generator\_M258: Steuern eines Rechteckwellensignals

### Überblick

Der Funktionsbaustein `Frequency Generator` (Frequenzgenerator) steuert einen Rechteckwellen-Signalausgang mit der angegebenen Frequenz.

### Grafische Darstellung (LD/FBD)



### Darstellung in AWL (IL) und ST

Sie finden eine allgemeine Darstellung in AWL- oder ST-Sprache im Kapitel *Unterschiede zwischen Funktionen und Funktionsbausteinen* (siehe Seite 52).

### Eingangsvariablen

In der folgenden Tabelle werden die Eingangsvariablen beschrieben:

Eingänge	Typ	Kommentar
EN_Enable	BOOL	TRUE: Erlaubt die Aktivierung des <code>Frequency Generator</code> über den Eingang <code>IN_EN</code> (sofern konfiguriert).
F_Enable	BOOL	TRUE: Forciert die Aktivierungsfunktion.
EN_SYNC	BOOL	TRUE: Erlaubt den Neustart über den Eingang "IN_SYNC" des internen Zeitgebers relativ zu der Zeitbasis (sofern konfiguriert).
F_SYNC	BOOL	Erzwingt an der steigenden Flanke einen Neustart des internen Zeitgebers relativ zu der Zeitbasis.
Frequency	DWORD	Frequenz des Ausgangssignals des <code>Frequency Generator</code> in Zehntel Hz. (Bereich: min. 1 (0,1 Hz) - max. 1.000.000 (100 kHz))

### Ausgangsvariablen



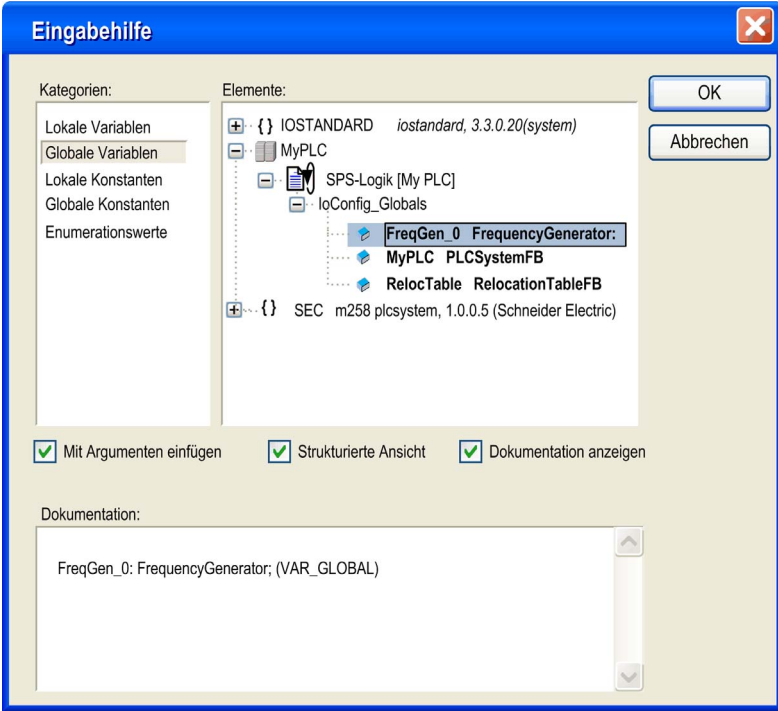
In der folgenden Tabelle werden die Ausgangsvariablen beschrieben:

Ausgänge	Typ	Kommentar
InFrequency	BOOL	TRUE: Das Signal des Frequency Generator wird in der angegebenen Frequenz ausgegeben.
Busy	BOOL	"Busy" wird verwendet, um anzugeben, dass eine Befehlsänderung stattfindet: Die Frequenz wird geändert. Wird auf TRUE gesetzt, wenn der Aktivierungsbefehl gesetzt und das Frequenzgenerator-Signal nicht mit der angegebenen Frequenz ausgegeben wird. Wird auf FALSE zurückgesetzt, wenn InFrequency oder Error gesetzt oder der Aktivierungsbefehl zurückgesetzt wurde. Wenn die Ausführung einer Befehlsänderung sofort erfolgt, bleibt "Busy" auf FALSE.
Error	BOOL	TRUE: Gibt an, dass ein Fehler erkannt wurde.
ErrID	EXPERT_ERR_TYPE <i>(siehe Seite 61)</i>	Wenn Error gesetzt ist: Typ des erkannten Fehlers.

## Programmieren

### Vorgehensweise

So programmieren Sie einen `FrequencyGenerator`-Funktionsbaustein:

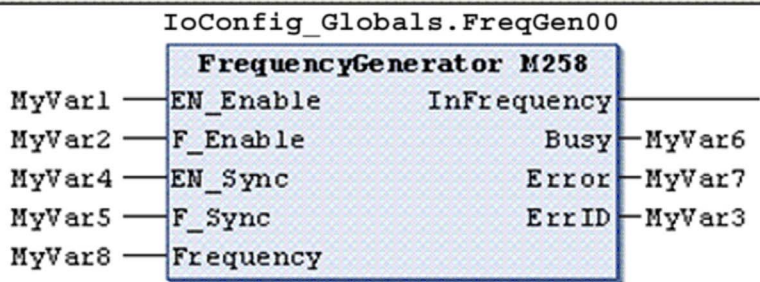
Schritt	Aktion
1	Wählen Sie die Registerkarte <b>Bibliotheken</b> im <b>Softwarekatalog</b> aus und klicken Sie auf <b>Bibliotheken</b> . Wählen Sie <b>Steuerung</b> → <b>M258</b> → <b>M258 Expert IO</b> → <b>OutputGenerator</b> → <b>FrequencyGenerator_M258</b> in der Liste aus und ziehen Sie das Element in das Fenster <b>POU</b> .
2	<p>Wählen Sie die Instanz des Funktionsbausteins durch einen Klick auf   auf. Das Fenster der Eingabehilfe wird angezeigt. Wählen Sie die globale Variable mit Bezug auf den bei der Konfiguration hinzugefügten FreqGen (<i>siehe Seite 39</i>) aus und bestätigen Sie Ihre Auswahl.</p>  <p><b>HINWEIS:</b> Wenn die Funktionsbausteininstanz nicht sichtbar ist, prüfen Sie die Konfiguration des Frequenzgenerators.</p>
3	Die Eingänge/Ausgänge sind im Funktionsbaustein ( <i>siehe Seite 41</i> ) enthalten. Eine Beschreibung der Interaktion zwischen den Eingängen/Ausgängen finden Sie unter Allgemeine Informationen ( <i>siehe Seite 47</i> ).

### Programmierbeispiel

Die nachstehende Abbildung zeigt das Beispiel eines programmierten -Funktionsbausteins **FrequencyGenerator\_M258**.

```

PROGRAM POU_1
VAR
    MyVar1: BOOL;
    MyVar2: BOOL;
    MyVar3: EXPERT_ERR_TYPE;
    MyVar4: BOOL;
    MyVar5: BOOL;
    MyVar6: BOOL;
    MyVar7: BOOL;
    MyVar8: DWORD;
END_VAR
    
```



---

# Anhang

---



## Überblick

Dieser Anhang enthält Auszüge aus dem Programmierhandbuch zum besseren technischen Verständnis der Bibliotheksdokumentation.

## Inhalt dieses Anhangs

Dieser Anhang enthält die folgenden Kapitel:

Kapitel	Kapitelname	Seite
A	Allgemeine Informationen	47
B	Darstellung von Funktionen und Funktionsbausteinen	51
C	Datentypen	61



---

# Anhang A

## Allgemeine Informationen

---

### Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Zweckbestimmte Funktionen	48
Allgemeine Informationen zur Funktionsbausteinverwaltung	49

## Zweckbestimmte Funktionen

### Prellfilter

Die nachstehende Tabelle zeigt maximalen Zählerfrequenzen, die über die Filterwerte festgelegt und zur Reduzierung des Premeffekts an den Eingängen verwendet werden.

Eingang	Filterwert zur Entprellung (ms)	Max. Zählerfrequenz Experte	Max. Zählerfrequenz Standard
A B	0,000	200 kHz	1 kHz
	0,001	200 kHz	1 kHz
	0,002	200 kHz	1 kHz
	0,005	100 kHz	1 kHz
	0,01	50 kHz	1 kHz
	0,05	25 kHz	1 kHz
	0,1	5 kHz	1 kHz
	0,5	1 kHz	1 kHz
	1	500 Hz	500 Hz
	5	100 Hz	100 Hz
A ist der Zählengang des Zählers. B ist der Zählengang des 2-Phasenzählers.			

### Zweckbestimmte Ausgänge

Auf Ausgänge, die von Hochgeschwindigkeitsexpertenfunktionen verwendet werden, kann nur über den Funktionsbaustein zugegriffen werden. Sie können nicht direkt von der Anwendung gelesen oder geschrieben werden.

## WARNUNG

### UNBEABSICHTIGTER GERÄTEBETRIEB

- Eine Funktionsbausteininstanz darf nicht in verschiedenen Programm-Tasks verwendet werden.
- Nehmen Sie an der Funktionsbaustein-Referenz (AXIS) keinerlei Änderungen vor, während der Funktionsbaustein ausgeführt wird.

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



## Allgemeine Informationen zur Funktionsbausteinverwaltung

### Verwaltung der Eingangsvariablen

Die Variablen werden mit der steigenden Flanke an Eingang `Execute` verwendet. Um eine Variable zu ändern, müssen Sie die Eingangsvariablen ändern und den Funktionsbaustein erneut auslösen.

Wenn eine Variable eines Funktionsbausteineingangs fehlt (= offen), wird gemäß der Norm IEC 61131-3 der Wert des vorherigen Aufrufs der Instanz verwendet. Beim ersten Aufruf wird der Initialwert angewendet.

### Verwaltung der Ausgangsvariablen

Die Ausgänge `Done`, `Error`, `Busy` und `CommandAborted` schließen sich gegenseitig aus: Für einen Funktionsbaustein darf nur jeweils einer von ihnen `TRUE` sein. Wenn für den Eingang `Execute` der Wert `TRUE` gilt, entspricht einer dieser Ausgänge dem Wert `TRUE`.

Bei einer steigenden Flanke an Eingang `Execute` wird der Ausgang `Busy` gesetzt. Der Ausgang bleibt für die Dauer der Ausführung des Funktionsbausteins gesetzt und wird bei einer steigenden Flanke an einem der anderen Ausgänge zurückgesetzt (`Done`, `Error`).

Der Ausgang `Done` wird gesetzt, sobald der Funktionsbaustein erfolgreich ausgeführt wurde.

Bei Feststellung eines Fehlers wird der Funktionsbaustein beendet, indem der Ausgang `Error` gesetzt wird. Der Fehlercode ist im Ausgang `ErrID` enthalten.

Die Ausgänge `Done`, `Error`, `ErrID` und `CommandAborted` werden bei einer fallenden Flanke an Eingang `Execute` gesetzt oder zurückgesetzt:

- Zurückgesetzt, wenn die Ausführung des Funktionsbausteins abgeschlossen ist.
- Gesetzt während mindestens eines Taskzyklus, wenn die Ausführung des Funktionsbausteins nicht abgeschlossen ist.

Wenn eine Instanz eines Funktionsbausteins einen neuen `Execute`-Befehl erhält, bevor sie vollständig ausgeführt wurde (bei einer Reihe von Befehlen für die gleiche Instanz), gibt der Funktionsbaustein keine Rückmeldung zurück wie `Done` bei der vorherigen Aktion.

### Fehlerbehandlung

Alle Bausteine haben zwei Ausgänge, die während der Ausführung des Funktionsbausteins die Feststellung eines Fehlers melden können:

- `Error`= Die steigende Flanke dieses Bits weist darauf hin, dass ein Fehler erkannt wurde.
- `ErrID`= Der Fehlercode des erkannten Fehlers.

Bei Auftreten eines `Fehlers` werden andere Ausgangssignale, wie z. B. `Done`, zurückgesetzt.



---

# Anhang B

## Darstellung von Funktionen und Funktionsbausteinen

---

### Übersicht

Jede Funktion kann in den folgenden Sprachen dargestellt werden.

- AWL: Anweisungsliste
- ST: Strukturierter Text
- KOP: Kontaktplan
- FBD: Funktionsbausteindiagramm
- CFC: Continuous Function Chart

Dieses Kapitel enthält Darstellungen von Funktionen und Funktionsbausteinen und erläutert deren Verwendung in den Sprachen AWL und ST.

### Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Unterschiede zwischen einer Funktion und einem Funktionsbaustein	52
Verwenden einer Funktion oder eines Funktionsbausteins in der Sprache AWL	53
Verwendung einer Funktion oder eines Funktionsbausteins in der Sprache ST	58

## Unterschiede zwischen einer Funktion und einem Funktionsbaustein

### Funktion

Eine Funktion hat die folgenden Eigenschaften:

- Ist eine POU (Program Organization Unit), die ein einzelnes direktes Ergebnis zurückgibt
- Wird direkt über ihren Namen aufgerufen (nicht über eine Instanz)
- Ist nicht instanziiert
- Kann als Operand in anderen Ausdrücken verwendet werden

**Beispiele:** Boolesche Operatoren (AND), Berechnungen, Konvertierung (BYTE\_TO\_INT)

### Funktionsbaustein

Ein Funktionsbaustein hat die folgenden Eigenschaften:

- Ist eine POU (Program Organization Unit), die ein oder mehrere direkte Ausgänge zurückgibt
- Muss von einer Instanz aufgerufen werden (Funktionsbausteinkopie mit dediziertem Namen und Variablen)
- Hat für jede Instanz einen persistenten Status (Ausgänge und interne Variablen) von einem Aufruf zum anderen aus einem Funktionsbaustein oder Programm

**Beispiele:** Zeitgeber, Zähler

In dem nachstehenden Beispiel ist `Timer_ON` eine Instanz des Funktionsbausteins `TON`:

```
1  PROGRAM MyProgram_ST
2  VAR
3      Timer_ON: TON; // Function Block Instance
4      Timer_RunCd: BOOL;
5      Timer_PresetValue: TIME := T#5S;
6      Timer_Output: BOOL;
7      Timer_ElapsedTime: TIME;
8  END_VAR
```

```
1  Timer_ON(
2      IN:=Timer_RunCd,
3      PT:=Timer_PresetValue,
4      Q=>Timer_Output,
5      ET=>Timer_ElapsedTime);
```

## Verwenden einer Funktion oder eines Funktionsbausteins in der Sprache AWL

### Allgemeine Informationen

In diesem Abschnitt wird das Implementieren einer Funktion und eines Funktionsbausteins in der Sprache AWL beschrieben.

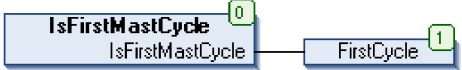

Die Funktionen `IsFirstMastCycle` und `SetRTCDrift` und der Funktionsbaustein `TON` werden als Implementierungsbeispiele verwendet.

### Verwenden einer Funktion in der AWL-Sprache

Im Folgenden wird das Einfügen einer Funktion in der AWL-Sprache beschrieben:

Schritt	Aktion
1	Öffnen oder erstellen Sie eine neue POU in der AWL-Sprache. <b>HINWEIS:</b> Die Vorgehensweise zum Erstellen einer POU wird hier nicht erläutert. Weitere Informationen finden Sie unter Hinzufügen und Aufrufen von POU's ( <i>siehe EcoStruxure Machine Expert, Programming Guide</i> ).
2	Erstellen Sie die Variablen, die für die Funktion erforderlich sind.
3	Wenn die Funktion über mindestens einen Eingang verfügt, beginnen Sie mit dem Laden des ersten Eingangs mithilfe der LD-Anweisung.
4	Fügen Sie unten eine neue Zeile ein, und gehen Sie wie folgt vor: <ul style="list-style-type: none"> <li>● Geben Sie den Namen der Funktion in der Operator-Spalte (linkes Feld) ein.</li> <li>● Oder verwenden Sie die <b>Eingabehilfe</b>, um die Funktion auszuwählen. (Wählen Sie im Kontextmenü <b>Baustein aufruf einfügen</b>.)</li> </ul>
5	Wenn die Funktion über mehr als einen Eingang verfügt und die Eingabehilfe verwendet wird, wird die erforderliche Anzahl von Zeilen automatisch mit ??? in den Feldern rechts erstellt. Ersetzen Sie ??? durch den geeigneten Wert oder die Variable, die der Reihenfolge der Eingänge entspricht.
6	Fügen Sie eine neue Zeile ein, um das Ergebnis der Funktion in der entsprechenden Variable zu speichern: geben Sie ST-Anweisungen in der Bedienspalte (linkes Feld) und den Variablennamen auf der rechten Seite ein.

Die Funktionen `IsFirstMastCycle` (ohne Eingangsparameter) und `SetRTCDrift` (mit Eingangsparametern) werden im Folgenden grafisch dargestellt:

Funktion	Grafische Darstellung
ohne Eingangsparameter: <code>IsFirstMastCycle</code>	 <p>The diagram shows a function block labeled <b>IsFirstMastCycle</b> with a small green box containing the number '0' in the top right corner. Below the block name is the text 'IsFirstMastCycle'. A line connects the right side of the block to another block labeled <b>FirstCycle</b>, which has a small green box containing the number '1' in the top right corner.</p>
mit Eingangsparametern: <code>SetRTCDrift</code>	 <p>The diagram shows a function block labeled <b>SetRTCDrift</b> with a small green box containing the number '0' in the top right corner. Below the block name is the text 'SetRTCDrift'. On the left side, there are four input blocks: <b>myDrift</b>, <b>myDay</b>, <b>myHour</b>, and <b>myMinute</b>. Lines connect each of these input blocks to the left side of the main block, with labels 'RtcDrift', 'Day', 'Hour', and 'Minute' respectively. On the right side, a line connects the main block to an output block labeled <b>myDiag</b>, which has a small green box containing the number '1' in the top right corner.</p>

In der AWL-Sprache wird der Funktionsname direkt in der Operator-Spalte verwendet:

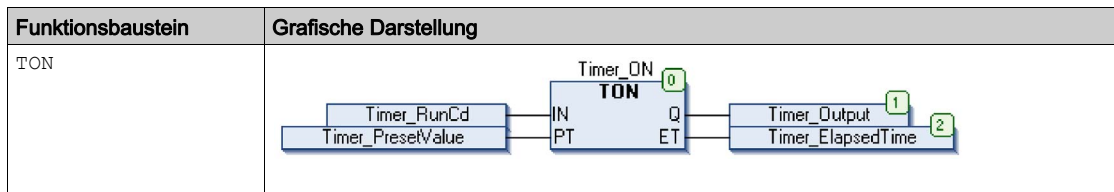
Funktion	Darstellung im POU-Editor in AWL															
<p>Beispiel einer Funktion ohne Eingangsparameter in der AWL-Sprache: IsFirstMastCycle</p>	<pre> 1  PROGRAM MyProgram_IL 2  VAR 3      FirstCycle: BOOL; 4  END_VAR 5 </pre> <hr/> <table border="1" data-bbox="411 454 1008 568"> <tr> <td data-bbox="411 454 768 495">1</td> <td data-bbox="411 495 768 535"><b>IsFirstMastCycle</b></td> <td data-bbox="768 495 1008 535"></td> </tr> <tr> <td data-bbox="411 535 768 568"></td> <td data-bbox="411 568 768 609"><b>ST</b></td> <td data-bbox="768 568 1008 609">FirstCycle</td> </tr> </table>	1	<b>IsFirstMastCycle</b>			<b>ST</b>	FirstCycle									
1	<b>IsFirstMastCycle</b>															
	<b>ST</b>	FirstCycle														
<p>Beispiel einer Funktion mit Eingangsparametern in der AWL-Sprache: SetRTCDrift</p>	<pre> 1  PROGRAM MyProgram_IL 2  VAR 3      myDrift: SINT (-29..29) := 5; 4      myDay: DAY_OF_WEEK := SUNDAY; 5      myHour: HOUR := 12; 6      myMinute: MINUTE; 7      myDiag: RTCSETDRIFT_ERROR; 8  END_VAR 9 </pre> <hr/> <table border="1" data-bbox="411 966 960 1144"> <tr> <td data-bbox="411 966 713 1006">1</td> <td data-bbox="411 1006 713 1047"><b>LD</b></td> <td data-bbox="713 1006 960 1047">myDrift</td> </tr> <tr> <td data-bbox="411 1047 713 1088"></td> <td data-bbox="411 1088 713 1128"><b>SetRTCDrift</b></td> <td data-bbox="713 1088 960 1128">myDay</td> </tr> <tr> <td data-bbox="411 1128 713 1169"></td> <td data-bbox="411 1169 713 1209"></td> <td data-bbox="713 1169 960 1209">myHour</td> </tr> <tr> <td data-bbox="411 1209 713 1250"></td> <td data-bbox="411 1250 713 1291"></td> <td data-bbox="713 1250 960 1291">myMinute</td> </tr> <tr> <td data-bbox="411 1291 713 1331"></td> <td data-bbox="411 1331 713 1372"><b>ST</b></td> <td data-bbox="713 1331 960 1372">myDiag</td> </tr> </table>	1	<b>LD</b>	myDrift		<b>SetRTCDrift</b>	myDay			myHour			myMinute		<b>ST</b>	myDiag
1	<b>LD</b>	myDrift														
	<b>SetRTCDrift</b>	myDay														
		myHour														
		myMinute														
	<b>ST</b>	myDiag														

### Verwenden eines Funktionsbausteins in der AWL-Sprache

Im Folgenden wird das Einfügen eines Funktionsbausteins in der AWL-Sprache beschrieben:

Schritt	Aktion
1	Erstellen Sie eine neue POU in der AWL-Sprache. <b>HINWEIS:</b> Die Vorgehensweise zum Erstellen einer POU wird hier nicht erläutert. Weitere Informationen finden Sie unter Hinzufügen und Aufrufen von POU's ( <i>siehe EcoStruxure Machine Expert, Programming Guide</i> ).
2	Erstellen Sie die Variablen, die für den Funktionsbaustein erforderlich sind, einschließlich des Instanznamens.
3	Funktionsbausteine werden mithilfe einer CAL-Anweisung aufgerufen: <ul style="list-style-type: none"> <li>• Verwenden Sie die <b>Eingabehilfe</b>, um den FB auszuwählen. (Klicken Sie mit der rechten Maustaste, und wählen Sie im Kontextmenü <b>Baustein aufruf einfügen</b> aus.)</li> <li>• Die CAL-Anweisung und der entsprechende E/A werden erstellt.</li> </ul> Jeder Parameter (E/A) ist eine Anweisung: <ul style="list-style-type: none"> <li>• Werte für Eingänge werden mit " := " festgelegt.</li> <li>• Werte für Ausgänge werden mit " =&gt; " festgelegt.</li> </ul>
4	Ersetzen Sie im rechten CAL-Feld die ??? durch den Instanznamen.
5	Ersetzen Sie weitere ??? durch eine geeignete Variable oder einen direkten Wert.

Der grafisch dargestellte Funktionsbaustein TON dient in diesem Beispiel zur Veranschaulichung:





In der AWL-Sprache wird der Name des Funktionsbausteins direkt in der Operator-Spalte verwendet:

Funktionsbaustein	Darstellung im POU-Editor in AWL
TON	<pre>1  PROGRAM MyProgram_IL 2  VAR 3  Timer_ON: TON; // Function Block instance declaration 4  Timer_RunCd: BOOL; 5  Timer_PresetValue: TIME := T#5S; 6  Timer_Output: BOOL; 7  Timer_ElapsedTime: TIME; 8  END_VAR 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000</pre>

## Verwendung einer Funktion oder eines Funktionsbausteins in der Sprache ST

### Allgemeines

In diesem Teil wird die Implementierung einer Funktion oder eines Funktionsbausteins in der ST-Sprache erläutert.

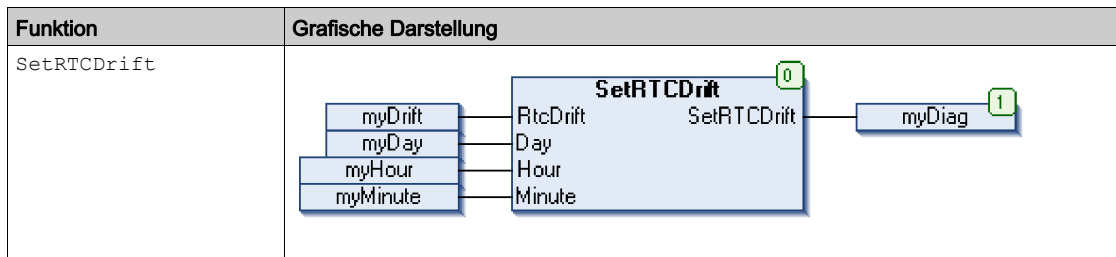
Dabei werden die Funktion `SetRTCDrift` und der Funktionsbaustein `TON` als Beispiele verwendet.

### Verwenden einer Funktion in der ST-Sprache

Im Folgenden wird das Einfügen einer Funktion in der ST-Sprache beschrieben:

Schritt	Aktion
1	Erstellen Sie eine neue POU in der ST-Sprache. <b>HINWEIS:</b> Die Vorgehensweise zum Erstellen einer POU wird hier nicht erläutert. Weitere Informationen finden Sie unter Hinzufügen und Aufrufen von POU's ( <i>siehe EcoStruxure Machine Expert, Programming Guide</i> ).
2	Erstellen Sie die Variablen, die für die Funktion erforderlich sind.
3	Verwenden Sie im <b>POU-ST-Editor</b> die allgemeine Syntax zur Darstellung einer Funktion in der ST-Sprache. Die allgemeine Syntax lautet: Funktionsergebnis:= Funktionsname(VarEingang1, VarEingang2,.. VarEingangx);

Zur Veranschaulichung dieses Verfahrens betrachten wir die grafisch dargestellte Funktion `SetRTCDrift`:



In der ST-Sprache wird diese Funktion folgendermaßen dargestellt:

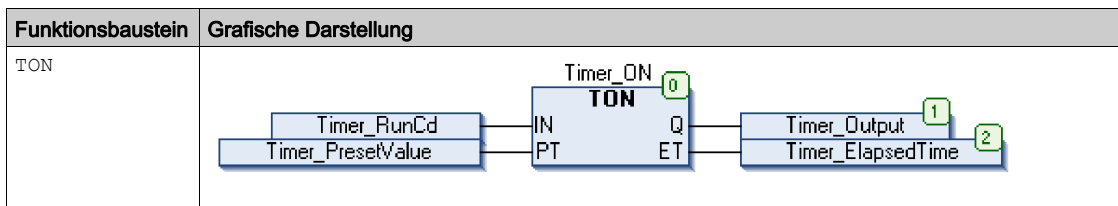
Funktion	Darstellung im POU-Editor in der ST-Sprache
SetRTCDrift	<pre>PROGRAM MyProgram_ST VAR myDrift: SINT(-29..29) := 5; myDay: DAY_OF_WEEK := SUNDAY; myHour: HOUR := 12; myMinute: MINUTE; myRTCAdjust: RTCDRIFT_ERROR; END_VAR myRTCAdjust:= SetRTCDrift(myDrift, myDay, myHour, myMinute);</pre>

### Verwenden eines Funktionsbausteins in der ST-Sprache

Im Folgenden wird das Einfügen eines Funktionsbausteins in der ST-Sprache beschrieben:

Schritt	Aktion
1	Erstellen Sie eine neue POU in der ST-Sprache. <b>HINWEIS:</b> Die Vorgehensweise zum Erstellen einer POU wird hier nicht erläutert. Weitere Informationen finden Sie unter Hinzufügen und Aufrufen von POU's ( <i>siehe EcoStruxure Machine Expert, Programming Guide</i> ).
2	Erstellen Sie die Eingangs- und Ausgangsvariablen und die Instanzen, die für den Funktionsbaustein erforderlich sind: <ul style="list-style-type: none"> <li>Die Eingangsvariablen sind die für den Funktionsbaustein erforderlichen Eingangsparameter.</li> <li>Die Ausgangsvariablen erhalten den vom Funktionsbaustein zurückgegebenen Wert.</li> </ul>
3	Verwenden Sie im <b>POU-ST-Editor</b> die allgemeine Syntax zur Darstellung eines Funktionsbausteins in der ST-Sprache. Die allgemeine Syntax lautet: Funktionsbaustein_InstanceName (Eingang1:=VarEingang1, Eingang2:=VarEingang2, ... Ausgang1=>VarAusgang1, Ausgang2=>VarAusgang2, ... );

Der gradisch dargestellte Funktionsbaustein TON dient in diesem Beispiel der Veranschaulichung des Verfahrens:



Die folgende Tabelle zeigt Beispiele für den Aufruf eines Funktionsbausteins in der ST-Sprache:

Funktionsbaustein	Darstellung im POU-Editor in der ST-Sprache
TON	<pre>1  PROGRAM MyProgram_ST 2  VAR 3      Timer_ON: TON; // Function Block Instance 4      Timer_RunCd: BOOL; 5      Timer_PresetValue: TIME := T#5S; 6      Timer_Output: BOOL; 7      Timer_ElapsedTime: TIME; 8  END_VAR  1  Timer_ON( 2      IN:=Timer_RunCd, 3      PT:=Timer_PresetValue, 4      Q=&gt;Timer_Output, 5      ET=&gt;Timer_ElapsedTime);</pre>

---

# Anhang C

## Datentypen

---

### EXPERT\_ERR\_TYPE: Typ der Fehlervariable für einen Expertenfunktionsbaustein

#### Beschreibung des Enumerationstyps

Die nachstehende Tabelle enthält die Werte für den Enumerationsdatentyp "ENUM":

Enumerator	Wert	Beschreibung
EXPERT_NO_ERROR	00 hex	Kein Fehler erkannt.
EXPERT_UNKNOWN	01 hex	Die Referenz "EXPERT" ist ungültig oder nicht konfiguriert.
EXPERT_UNKNOWN_PARAMETER	02 hex	Die Parameterreferenz ist ungültig.
EXPERT_INVALID_PARAMETER	03 hex	Der Parameterwert ist ungültig.
EXPERT_COM_ERROR	04 hex	Im EXPERT-Modul ist ein Kommunikationsfehler aufgetreten.
EXPERT_CAPTURE_NOT_CONFIGURED	05 hex	Die Erfassung wurde nicht konfiguriert.





## A

### Anwendung

Programm mit Konfigurationsdaten, Symbolen und Dokumentation.

## B

### Byte

In einem 8-Bit-Format codierter Typ. Gültiger Wertebereich: 00 hex bis FF hex.

## C

### CFC

(*Continuous Function Chart*) Grafische Programmiersprache (Erweiterung des Standards IEC 61131-3) auf der Grundlage der FBD-Sprache (Funktionsbausteindiagramm), die wie ein Flussdiagramm aufgebaut ist. Grafische Elemente werden allerdings, sofern möglich, ohne die Verwendung von Netzwerken frei positioniert, sodass Rückkopplungsschleifen möglich sind. Bei jedem Baustein befinden sich die Eingänge links und die Ausgänge rechts. Sie können die Bausteinausgänge mit den Eingängen anderer Bausteine verbinden, um komplexe Ausdrücke zu erstellen.

### CPDM

(*Controller Power Distribution Module: SPS-Stromverteilermodul*) Verbindungspunkt zwischen der Steuerung und den externen 24-VDC-Spannungsversorgungen und Anfangspunkt der Stromverteilung für die lokale Konfiguration.

## E

### E/A

(*Eingang/Ausgang*)

## F

### FB

(*Function Block: Funktionsbaustein*) Nützlicher Programmiermechanismus, der eine Gruppe von Programmieranweisungen zur Durchführung eines spezifischen und normierten Vorgangs konsolidiert, z. B. Drehzahlregelung, Intervallkontrolle oder Zählen. Ein Funktionsbaustein kann Konfigurationsdaten, eine Gruppe interner oder externer Betriebsparameter und in der Regel 1 oder mehrere Dateneingänge und -ausgänge umfassen.

## Funktion

Programmiereinheit, die über 1 Eingang verfügt und 1 unmittelbares Ergebnis zurückgibt. Im Gegensatz zu FBs jedoch wird eine Funktion direkt über ihren Namen (und nicht über eine Instanz) aufgerufen, weist zwischen zwei Aufrufen keinen persistenten Status auf und kann als Operand in anderen Programmierausdrücken verwendet werden.

Beispiele: Boolesche Operatoren (AND), Berechnungen, Konvertierungen (BYTE\_TO\_INT).

## Funktionsbausteindiagramm (Programmiersprache)

Eine von 5 Sprachen für die Logik oder Steuerung, die von dem Standard IEC 61131-3 für Steuerungssysteme unterstützt wird. Es handelt sich hierbei um eine grafisch orientierte Programmiersprache. Sie arbeitet mit einer Liste von Netzwerken, wobei jedes Netzwerk eine grafische Struktur von Feldern und Verbindungslinien enthält, die entweder einen logischen oder einen arithmetischen Ausdruck, den Aufruf eines Funktionsbausteins, einen Sprung oder einen Rückkehrbefehl darstellen.

## G

### Geber

Gerät zur Längen- oder Winkelmessung (lineare oder Drehgeber).

## H

### HSC

*High Speed Counter: Hochgeschwindigkeitszähler* Eine Funktion, die Impulse an der Steuerung oder an Erweiterungsmoduleingängen zählt.

## I

### IEC 61131-3

Teil 3 eines 3-teiligen IEC-Standards für industrielle Automatisierungsanlagen. IEC 61131-3 befasst sich mit den Programmiersprachen für Steuerungen und definiert 2 grafische und 2 textbasierte Programmiersprachenstandards. Grafische Programmiersprachen: Kontaktplan (KOP oder LD: Ladder) und Funktionsbausteindiagramm (FBD oder Function Block Diagram). Textbasierte Programmiersprachen: Strukturierter Text (ST) und Anweisungsliste (AWL oder IL: Instruction List).

### IL

*(Instruction List: Anweisungsliste (AWL))* Ein in Anweisungsliste 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).

### INT

*(Integer: Ganzzahl)* Über 16 Bits codierte Ganzzahl.



**L****LD**

(*Ladder Diagramm: Kontaktplan (KOP)*) 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).

**N****Netzwerk**

Ein Netzwerk umfasst miteinander verbundene Geräte, die einen gemeinsamen Datenpfad und dasselbe Protokoll zur Kommunikation verwenden.

**P****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.

**PWM**

(*Pulse Width Modulation: Impulsbreitenmodulation*) Schneller Ausgang, der innerhalb eines anpassbaren Arbeitszyklus zwischen dem Aus- und Ein-Zustand pendelt und dabei eine Rechteckschwingung erzeugt (obwohl Sie ihn zur Erzeugung eines Rechtecksignals einstellen können).

**R****Reflexausgang**

Zu den HSC-Ausgängen gehören u. a. auch Reflexausgänge. Diese Ausgänge sind einem Schwellenwert zugeordnet, der mit dem Zählerwert in Übereinstimmung mit der HSC-Konfiguration verglichen wird. Reflexausgänge schalten je nach konfigurierter Beziehung zum Schwellenwert entweder in den Ein- oder Aus-Zustand.

**RUN**

Befehl, der die Steuerung zur Abfrage des Anwendungsprogramms, zum Lesen der physischen Eingänge und zum Schreiben der physischen Ausgänge in Übereinstimmung mit der Auflösung der Programmlogik auffordert.

## S

### **Schnelle E/A**

(*Schneller Eingang/Ausgang*) Spezifische E/A-Module mit bestimmten elektrischen Merkmalen (z. B. Antwortzeit), wobei die Verarbeitung dieser Kanäle direkt über die Steuerung erfolgt.

### **ST**

(*Structured Text: Strukturierter Text*) Programmiersprache, die komplexe und verschachtelte Anweisungen umfasst (z. B. Iterationsschleifen, bedingte Ausführungen oder Funktionen). ST ist IEC 61131-3-kompatibel.

### **Steuerung**

Ermöglicht die Automatisierung industrieller Prozesse (auch als speicherprogrammierbare Steuerung oder SPS bezeichnet).

### **Steuerungsnetzwerk**

Ein Netzwerk mit Logic Controllern, SCADA-Systemen, PCs, HMI, Switches usw.

Es werden zwei Arten von Topologien unterstützt:

- Flach: Alle Module und Geräte in diesem Netzwerk gehören demselben Teilnetz an.
- 2-stufig: Das Netzwerk ist in ein Betriebsnetzwerk und ein Steuerungsnetzwerk unterteilt.

Diese beiden Netzwerke sind zwar physisch voneinander unabhängig, in der Regel jedoch über ein Routing-Gerät miteinander verbunden.

### **STOP**

Befehl, der bewirkt, dass die Steuerung die Ausführung eines Anwendungsprogramms stoppt.

## T

### **Task**

Gruppe von Sections und Unterprogrammen, die zyklisch oder periodisch (MAST-Task) bzw. periodisch (FAST-Task) ausgeführt werden.

Eine Task besitzt eine bestimmte Prioritätsstufe und ist den Eingängen und Ausgängen der Steuerung zugeordnet. Diese E/A werden in Abhängigkeit von der Task aktualisiert.

Eine Steuerung kann über mehrere Tasks verfügen.

## V

### **Variable**

Speichereinheit, die von einem Programm adressiert und geändert werden kann.



## D

### Datentypen

EXPERT\_ERR\_TYPE, *61*

## E

### EXPERT\_ERR\_TYPE

Datentypen, *61*

## F

### Fehlerbehandlung

ErrID, *49*

Error, *49*

### FrequencyGenerator\_M258

Steuern eines Rechteckwellensignals, *41*

### Frequenzgenerator

Beschreibung, *38*

Konfiguration, *39*

Programmieren, *43*

### Funktionen

Aktivierung, *26*

Synchronisierung, *26*

Unterschiede zwischen einer Funktion  
und einem Funktionsbaustein, *52*

Verwenden einer Funktion oder eines  
Funktionsbausteins in der Sprache AWL,  
*53*

Verwendung einer Funktion oder eines  
Funktionsbausteins in der Sprache ST, *58*

### Funktionsbausteine

FrequencyGenerator\_M258, *41*

PWM\_M258, *32*

## I

### Impulsbreitenmodulation

Beschreibung, *28*

Konfiguration, *30*

## P

### Programmieren

PWM, *34*

### PWM

Programmieren, *34*

PWM\_M241, *32*

### PWM\_M258

Steuern eines Impulsbreitenmodulations-  
signals, *32*

## V

### Verwaltung der Statusvariablen

Busy, *49*

CommandAborted, *49*

Done, *49*

ErrID, *49*

Error, *49*

Execute, *49*

## Z

Zweckbestimmte Funktionen, *48*

