

# Modicon M221 Logic Controller

## Bibliothekshandbuch zu erweiterten Funktionen

Übersetzung der Originalbetriebsanleitung

EIO0000003307.03

09/2025

# Rechtliche Hinweise

Die in diesem Dokument enthaltenen Informationen umfassen allgemeine Beschreibungen, technische Merkmale und Kenndaten und/oder Empfehlungen in Bezug auf Produkte/Lösungen.

Dieses Dokument ersetzt keinesfalls eine detaillierte Analyse bzw. einen betriebs- und standortspezifischen Entwicklungs- oder Schemaplan. Es darf nicht zur Ermittlung der Eignung oder Zuverlässigkeit von Produkten/Lösungen für spezifische Benutzeranwendungen verwendet werden. Es liegt im Verantwortungsbereich eines jeden Benutzers, selbst eine angemessene und umfassende Risikoanalyse, Risikobewertung und Testreihe für die Produkte/Lösungen in Übereinstimmung mit der jeweils spezifischen Anwendung bzw. Nutzung durchzuführen bzw. von entsprechendem Fachpersonal (Integrator, Spezifikateur oder ähnliche Fachkraft) durchführen zu lassen.

Die Marke Schneider Electric sowie alle anderen in diesem Dokument enthaltenen Markenzeichen von Schneider Electric SE und seinen Tochtergesellschaften sind das Eigentum von Schneider Electric SE oder seinen Tochtergesellschaften. Alle anderen Marken können Markenzeichen ihrer jeweiligen Eigentümer sein.

Dieses Dokument und seine Inhalte sind durch geltende Urheberrechtsgesetze geschützt und werden ausschließlich zu Informationszwecken bereitgestellt. Ohne die vorherige schriftliche Genehmigung von Schneider Electric darf kein Teil dieses Dokuments in irgendeiner Form oder auf irgendeine Weise (elektronisch, mechanisch, durch Fotokopieren, Aufzeichnen oder anderweitig) zu irgendeinem Zweck vervielfältigt oder übertragen werden.

Schneider Electric gewährt keine Rechte oder Lizenzen für die kommerzielle Nutzung des Dokuments oder dessen Inhalts, mit Ausnahme einer nicht-exklusiven und persönlichen Lizenz, es „wie besehen“ zu konsultieren.

Schneider Electric behält sich das Recht vor, jederzeit ohne entsprechende schriftliche Vorankündigung Änderungen oder Aktualisierungen mit Bezug auf den Inhalt bzw. am Inhalt dieses Dokuments oder dessen Format vorzunehmen.

**Soweit nach geltendem Recht zulässig, übernehmen Schneider Electric und seine Tochtergesellschaften keine Verantwortung oder Haftung für Fehler oder Auslassungen im Informationsgehalt dieses Dokuments oder für Folgen, die aus oder infolge der sachgemäßen oder missbräuchlichen Verwendung der herein enthaltenen Informationen entstehen.**

---

# Inhaltsverzeichnis

Sicherheitshinweise .....	7
Bevor Sie beginnen .....	7
Start und Test .....	8
Betrieb und Einstellungen .....	9
Informationen zum Dokument .....	10
<b>Einführung: Erweiterte Funktionen .....</b>	<b>15</b>
Einführung .....	16
Experten-E/A .....	17
Zuordnung der integrierten Experten-E/A .....	19
Allgemeine Informationen zur Funktionsbausteinverwaltung .....	21
<b>Erweiterte Eingangsfunktionen für Experten .....</b>	<b>22</b>
Schnellzähler (%FC) .....	23
Beschreibung .....	23
Konfiguration .....	24
Programmierbeispiel .....	26
Hochgeschwindigkeitszähler (%HSC) .....	28
Beschreibung .....	28
Hochgeschwindigkeitszähler in Zählmodi .....	32
Hochgeschwindigkeitszähler im Frequenzmesser-Modus .....	38
<b>Erweiterte Ausgangsfunktionen für Experten .....</b>	<b>40</b>
Impuls (%PLS) .....	41
Beschreibung .....	41
Funktionsblock-Konfiguration .....	42
Programmierbeispiel .....	47
Impulsbreitenmodulation (%PWM) .....	48
Beschreibung .....	48
Funktionsblock-Konfiguration .....	49
Programmierbeispiel .....	52
Antrieb (%DRV) .....	54
Beschreibung .....	54
Zustand der Antriebe und Logiksteuerungen .....	56
Hinzufügen eines Antriebsfunktionsbausteins .....	58
Funktionsblock-Konfiguration .....	59
MC_Power_ATV: Endstufe aktivieren/deaktivieren .....	59
MC_Jog_ATV: Tipbetrieb starten .....	62
MC_MoveVel_ATV: Bewegen mit einer bestimmten Geschwindigkeit .....	64
MC_Stop_ATV: Bewegung anhalten .....	67
MC_ReadStatus_ATV: Gerätestatus lesen .....	70
MC_ReadMotionState_ATV: Bewegungsstatus lesen .....	72
MC_Reset_ATV: Quittieren und Zurücksetzen eines Fehlers .....	74
Fehlercodes .....	77
Impulswellenausgang (%PTO) .....	80
Beschreibung .....	80
Impulswellenausgang (PTO) .....	80
Impulsausgangsmodi .....	82
Rampen Hochlaufzeit/Verzögerungszeit .....	83
Sondenergebnis .....	85

Spelausgleich .....	87
Positionierungsgrenzen .....	88
Konfiguration .....	90
PTO-Konfiguration .....	91
Motion Task Table .....	91
Programmierung .....	98
Hinzufügen oder Entfernen eines Funktionsbausteins .....	98
PTO-Funktionsbausteine .....	99
Homing-Modi .....	101
Homing-Modi .....	101
Positionseinstellung .....	103
Ausführliche Referenz .....	103
Kurze Referenz ohne Umkehr .....	104
Kurze Referenz mit Umkehr .....	105
Homing-Offset .....	107
Datenparameter .....	107
Objektcodes für Funktionsbausteine .....	107
Betriebsmodi .....	112
Bewegungszustandsdiagramm .....	112
Puffermodus .....	113
<b>Bewegungs-Funktionsbausteine .....</b>	<b>116</b>
<i>MC_MotionTask_PTO</i> Funktionsbaustein .....	116
<i>MC_Power_PTO</i> Funktionsbaustein .....	119
<i>MC_MoveVel_PTO</i> Funktionsbaustein .....	122
<i>MC_MoveRel_PTO</i> Funktionsbaustein .....	126
<i>MC_MoveAbs_PTO</i> Funktionsbaustein .....	129
<i>MC_Home_PTO</i> Funktionsbaustein .....	133
<i>MC_SetPos_PTO</i> Funktionsbaustein .....	135
<i>MC_Stop_PTO</i> Funktionsbaustein .....	137
<i>MC_Halt_PTO</i> Funktionsbaustein .....	139
<b>Administrative Funktionsbausteine .....</b>	<b>141</b>
<i>MC_ReadActVel_PTO</i> Funktionsbaustein .....	141
<i>MC_ReadActPos_PTO</i> Funktionsbaustein .....	143
<i>MC_ReadSts_PTO</i> Funktionsbaustein .....	144
<i>MC_ReadMotionState_PTO</i> Funktionsbaustein .....	146
<i>MC_ReadAxisError_PTO</i> Funktionsbaustein .....	147
<i>MC_Reset_PTO</i> Funktionsbaustein .....	149
<i>MC_TouchProbe_PTO</i> Funktionsbaustein .....	150
<i>MC_AbortTrigger_PTO</i> Funktionsbaustein .....	153
<i>MC_ReadPar_PTO</i> Funktionsbaustein .....	154
<i>MC_WritePar_PTO</i> Funktionsbaustein .....	155
Frequenzgenerator (%FREQGEN) .....	157
Beschreibung .....	157
Konfiguration .....	159
Erweiterte Softwarefunktion .....	161
PID-Funktion .....	162
PID-Betriebsarten .....	162
PID-Betriebsarten .....	162
PID-Auto-Tuning-Konfiguration .....	163
Konfiguration der PID-Auto-Tuning-Funktion .....	163
PID-Standardkonfiguration .....	166

---

PID-Konfiguration über eine Wortadresse .....	166
PID-Einstellung mittels Auto-Tuning (AT).....	169
Manueller Modus .....	173
Ermittlung der Abtastperiode (Ts) .....	174
PID-Assistent .....	176
Zugriff auf den PID-Assistenten .....	176
Registerkarte „Allgemein“ .....	177
Registerkarte „Eingang“ .....	179
Registerkarte „PID“ .....	180
Registerkarte „AT“ .....	182
Registerkarte „Ausgang“ .....	183
PID-Programmierung .....	185
Beschreibung.....	185
Programmierung und Konfiguration .....	186
PID-Zustände und erkannte Fehlercodes .....	187
Anhang .....	190
PID-Parameter .....	191
Aufgabe und Einfluss von PID-Parametern .....	191
Methode zur Anpassung der PID-Parameter .....	193
Glossar .....	195
Index .....	197



# Sicherheitshinweise

## Wichtige Informationen

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.

## Bevor Sie beginnen

Dieses Produkt nicht mit Maschinen ohne effektive Sicherheitseinrichtungen im Arbeitsraum verwenden. Das Fehlen effektiver Sicherheitseinrichtungen im Arbeitsraum einer Maschine kann schwere Verletzungen des Bedienpersonals zur Folge haben.

## ▲ WARNUNG

### UNBEAUF SICHTIGTE GERÄTE

- Diese Software und zugehörige Automatisierungsgeräte nicht an Maschinen verwenden, die nicht über Sicherheitseinrichtungen im Arbeitsraum verfügen.
- Greifen Sie bei laufendem Betrieb nicht in das Gerät.

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

Dieses Automatisierungsgerät und die zugehörige Software dienen zur Steuerung verschiedener industrieller Prozesse. Der Typ bzw. das Modell des für die jeweilige Anwendung geeigneten Automatisierungsgeräts ist von mehreren Faktoren abhängig, z. B. von der benötigten Steuerungsfunktion, der erforderlichen Schutzklasse, den Produktionsverfahren, außergewöhnlichen Bedingungen, behördlichen Vorschriften usw. Für einige Anwendungen werden möglicherweise mehrere Prozessoren benötigt, z. B. für ein Backup-/Redundanzsystem.

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. des Prozesses zum Tragen kommen. Demzufolge sind allein Sie in der Lage, die Automatisierungskomponenten und zugehörigen Sicherheitsvorkehrungen und Verriegelungen zu identifizieren, die einen ordnungsgemäßen Betrieb gewährleisten. Bei der Auswahl der Automatisierungs- und Steuerungsgeräte sowie der zugehörigen Software für eine bestimmte Anwendung sind die einschlägigen örtlichen und landesspezifischen Richtlinien und Vorschriften zu beachten. Das National Safety Council's Accident Prevention Manual (Handbuch zur Unfallverhütung; in den USA landesweit anerkannt) enthält ebenfalls zahlreiche nützliche Hinweise.

Für einige Anwendungen, z. B. Verpackungsmaschinen, sind zusätzliche Vorrichtungen zum Schutz des Bedienpersonals wie beispielsweise Sicherheitseinrichtungen im Arbeitsraum erforderlich. Diese Vorrichtungen werden benötigt, wenn das Bedienpersonal mit den Händen oder anderen Körperteilen in den Quetschbereich oder andere Gefahrenbereiche gelangen kann und somit einer potenziellen schweren Verletzungsgefahr ausgesetzt ist. Software-Produkte allein können das Bedienpersonal nicht vor Verletzungen schützen. Die Software kann daher nicht als Ersatz für Sicherheitseinrichtungen im Arbeitsraum verwendet werden.

Vor Inbetriebnahme der Anlage sicherstellen, dass alle zum Schutz des Arbeitsraums vorgesehenen mechanischen/elektronischen Sicherheitseinrichtungen und Verriegelungen installiert und funktionsfähig sind. Alle zum Schutz des Arbeitsraums vorgesehenen Sicherheitseinrichtungen und Verriegelungen müssen mit dem zugehörigen Automatisierungsgerät und der Softwareprogrammierung koordiniert werden.

**HINWEIS:** Die Koordinierung der zum Schutz des Arbeitsraums vorgesehenen mechanischen/elektronischen Sicherheitseinrichtungen und Verriegelungen geht über den Umfang der Funktionsbaustein-Bibliothek, des System-Benutzerhandbuchs oder andere in dieser Dokumentation genannten Implementierungen hinaus.

## Start und Test

Vor der Verwendung elektrischer Steuerungs- und Automatisierungsgeräte ist das System zur Überprüfung der einwandfreien Funktionsbereitschaft einem Anlauftest zu unterziehen. Dieser Test muss von qualifiziertem Personal durchgeführt werden. Um einen vollständigen und erfolgreichen Test zu gewährleisten, müssen die entsprechenden Vorkehrungen getroffen und genügend Zeit eingeplant werden.

## ▲ WARNUNG

### GEFAHR BEIM GERÄTEBETRIEB

- Überprüfen Sie, ob alle Installations- und Einrichtungsverfahren vollständig durchgeführt wurden.
- Vor der Durchführung von Funktionstests sämtliche Blöcke oder andere vorübergehende Transportsicherungen von den Anlagekomponenten entfernen.
- Entfernen Sie Werkzeuge, Messgeräte und Verschmutzungen vom Gerät.

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

Führen Sie alle in der Dokumentation des Geräts empfohlenen Anlauffests durch. Die gesamte Dokumentation zur späteren Verwendung aufbewahren.

### **Softwaretests müssen sowohl in simulierten als auch in realen Umgebungen stattfinden.**

Sicherstellen, dass in dem komplett installierten System keine Kurzschlüsse anliegen und nur solche Erdungen installiert sind, die den örtlichen Vorschriften entsprechen (z. B. gemäß dem National Electrical Code in den USA). Wenn Hochspannungsprüfungen erforderlich sind, beachten Sie die Empfehlungen in der Gerätedokumentation, um eine versehentliche Beschädigung zu verhindern.

Vor dem Einschalten der Anlage:

- Entfernen Sie Werkzeuge, Messgeräte und Verschmutzungen vom Gerät.
- Schließen Sie die Gehäusetür des Geräts.
- Alle temporären Erdungen der eingehenden Stromleitungen entfernen.
- Führen Sie alle vom Hersteller empfohlenen Anlauffests durch.

## Betrieb und Einstellungen

Die folgenden Vorsichtsmaßnahmen stammen aus der NEMA Standards Publication ICS 7.1-1995:

(Im Falle einer Abweichung oder eines Widerspruchs zwischen einer Übersetzung und dem englischen Original hat der Originaltext in der englischen Sprache Vorrang.)

- Ungeachtet der bei der Entwicklung und Fabrikation von Anlagen oder bei der Auswahl und Bemessung von Komponenten angewandten Sorgfalt, kann der unsachgemäße Betrieb solcher Anlagen Gefahren mit sich bringen.
- Gelegentlich kann es zu fehlerhaften Einstellungen kommen, die zu einem unbefriedigenden oder unsicheren Betrieb führen. Für Funktionseinstellungen stets die Herstelleranweisungen zu Rate ziehen. Das Personal, das Zugang zu diesen Einstellungen hat, muss mit den Anweisungen des Anlagenherstellers und den mit der elektrischen Anlage verwendeten Maschinen vertraut sein.
- Nur die vom Bediener unbedingt vorzunehmenden betriebsspezifischen Einstellungen sollten für den Bediener zugänglich sein. Der Zugriff auf andere Steuerungsfunktionen sollte eingeschränkt sein, um unbefugte Änderungen der Betriebskenngrößen zu vermeiden.

# Informationen zum Dokument

## Umfang der Dokumentation

In diesem Dokument finden Sie Beschreibungen der erweiterten EcoStruxure Machine Expert - Basic Funktionen und ihrer Beziehung zum M221 Logic Controller Experten-E/A- und PID-Support. Hier finden Sie Beschreibungen der Funktionen, Eigenschaften und Leistungsmerkmale der erweiterten M221 Logic Controller Funktionen.

## Gültigkeitshinweis

Dieses Dokument wurde für EcoStruxure™ Machine Expert – Basic V1.4. aktualisiert.

## Produktbezogene Informationen

### **▲ WARNUNG**

#### **STEUERUNGS AUSFALL**

- Führen Sie vor der Implementierung eine Fehlermodus- und Effektanalyse (FMEA, Failure Mode and Effects Analysis) oder eine gleichwertige Risikoanalyse Ihrer Anwendung durch und wenden Sie Vorbeugemaßnahmen und Kontrollen an.
- Stellen Sie einen Fallback-Zustand für den Fall unerwünschter Steuerungsereignisse oder -sequenzen bereit.
- Sorgen Sie für separate oder redundante Steuerungspfade, wann immer erforderlich.
- Stellen Sie geeignete Parameter bereit, insbesondere für Grenzwerte.
- Überprüfen Sie die Auswirkungen von Übertragungsverzögerungen und ergreifen Sie Maßnahmen, um diese zu mindern.
- Überprüfen Sie die Auswirkungen von Unterbrechungen der Kommunikationsverbindung und ergreifen Sie Maßnahmen, um diese zu mindern.
- Stellen Sie unabhängige Pfade für Steuerungsfunktionen bereit (z. B. Not-Aus, Bedingungen bei Grenzüberschreitung und Fehler), die Ihrer Risikobewertung sowie den geltenden Vorschriften entsprechen.
- Wenden Sie lokale Unfallverhütungsvorschriften und -richtlinien an.<sup>1</sup>
- Jede Implementierung eines Systems muss auf ihre ordnungsgemäße Funktion getestet werden, bevor sie in Betrieb genommen 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.**

## Allgemeine Informationen zur Cybersicherheit

In den letzten Jahren hat sich durch die wachsende Anzahl an vernetzten Maschinen und Produktionsanlagen das Potenzial für Cyberbedrohungen wie unbefugter Zugriff, Datenverletzungen und Betriebsunterbrechungen entsprechend erhöht. Sie müssen daher alle möglichen Maßnahmen zur Cybersicherheit in Betracht ziehen, um Anlagen und Systeme vor solchen Bedrohungen zu schützen.

Um die Sicherheit und den Schutz Ihrer Schneider Electric-Produkte zu gewährleisten, ist es in Ihrem Interesse, die Best Practices für die Cybersicherheit umzusetzen, die im Dokument *Cybersecurity Best Practices* beschrieben sind.

Schneider Electric bietet zusätzliche Informationen und Unterstützung:

- Abonnieren Sie den *Sicherheits-Newsletter* von Schneider Electric.
- Besuchen Sie die *Webseite Cybersecurity Support Portal*, um:
  - Sicherheitshinweise zu suchen
  - Schwachstellen und Vorfälle zu melden
- Besuchen Sie die *Webseite Schneider Electric Cybersecurity and Data Protection Posture*, um:
  - auf den Cybersicherheitsstatus zuzugreifen
  - mehr über Cybersicherheit in der *Cybersecurity Academy* zu erfahren
  - die *Cybersicherheits-Services* von Schneider Electric zu entdecken

## Verfügbare Sprachen des Dokuments

Dieses Dokument ist in folgenden Sprachen verfügbar:

- Englisch (EIO0000003305)
- Französisch (EIO0000003306)
- Deutsch (EIO0000003307)
- Spanisch (EIO0000003308)
- Italienisch (EIO0000003309)
- Chinesisch (EIO0000003310)
- Portugiesisch (EIO0000003311)
- Türkisch (EIO0000003312)

## Weiterführende Dokumente

Titel der Dokumentation	Referenznummer
EcoStruxure Machine Expert - Basic – Betriebshandbuch	EIO0000003281 (ENG) EIO0000003282 (FRA) EIO0000003283 (GER) EIO0000003284 (SPA) EIO0000003285 (ITA) EIO0000003286 (CHS) EIO0000003287 (POR) EIO0000003288 (TUR)
EcoStruxure Machine Expert - Basic Allgemeine Funktionen – Bibliothekshandbuch	EIO0000003289 (ENG) EIO0000003290 (FRA) EIO0000003291 (GER) EIO0000003292 (SPA) EIO0000003293 (ITA) EIO0000003294 (CHS) EIO0000003295 (POR) EIO0000003296 (TUR)
Modicon M221 Logic Controller – Programmierhandbuch	EIO0000003297 (ENG) EIO0000003298 (FRE) EIO0000003299 (GER) EIO0000003300 (SPA) EIO0000003301 (ITA) EIO0000003302 (CHS) EIO0000003304 (TUR) EIO0000003303 (POR)
Modicon M221 Logic Controller – Hardwarehandbuch	EIO0000003313 (ENG) EIO0000003314 (FRA) EIO0000003315 (GER) EIO0000003316 (SPA) EIO0000003317 (ITA) EIO0000003318 (CHS) EIO0000003319 (POR) EIO0000003320 (TUR)

Um Dokumente online zu finden, besuchen Sie das Schneider Electric Download-Center ([www.se.com/ww/en/download/](http://www.se.com/ww/en/download/)).

## Informationen zu nicht-inklusive oder unsensibler Terminologie

Als verantwortungsbewusstes, integratives Unternehmen aktualisiert Schneider Electric kontinuierlich seine Kommunikationen und Produkte, die nicht-integrative oder unsensible Terminologie enthalten. Trotz dieser Bemühungen können unsere

Inhalte jedoch nach wie vor Begriffe enthalten, die von einigen Kunden als unangemessen betrachtet werden.

## Terminologie gemäß den geltenden Standards

Die technischen Begriffe, Terminologie, Symbole und die entsprechenden Beschreibungen in den hierin enthaltenen oder in bzw. auf den Produkten selbst angegebenen Informationen sind im Allgemeinen von den Begriffen oder Definitionen internationaler Normen abgeleitet.

Im Bereich der funktionalen Sicherheitssysteme, Antriebe und allgemeinen Automatisierungssysteme kann dies unter anderem Begriffe wie *Sicherheit*, *Sicherheitsfunktion*, *Sicherer Zustand*, *Störung*, *Fehlerreset*, *Fehlfunktion*, *Versagen/Ausfall*, *Fehler*, *Fehlermeldung*, *Gefährlich* usw. umfassen.

Zu diesen Normen und Standards zählen unter anderem:

Norm/Standard	Beschreibung
IEC 61131-2:2007	Speicherprogrammierbare Steuerungen, Teil 2: Betriebsmittelanforderungen und Prüfungen.
ISO 13849-1:2023	Sicherheit von Maschinen: Sicherheitsbezogene Teile von Steuerungen Allgemeine Gestaltungsleitsätze.
EN 61496-1:2020	Sicherheit von Maschinen: Berührungslos wirkende Schutzeinrichtung Teil 1: Allgemeine Anforderungen und Tests
ISO 12100:2010	Sicherheit von Maschinen – Allgemeine Gestaltungsleitsätze – Risikobeurteilung und Risikominderung
EN 60204-1:2006	Sicherheit von Maschinen – Elektrische Ausrüstung von Maschinen – Teil 1: 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 62061:2021	Sicherheit von Maschinen – Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer 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 für sicherheitsbezogene elektrische/elektronische/programmierbare elektronische Systeme
IEC 61508-3:2010	Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme: Softwareanforderungen
IEC 61784-3:2021	Industrielle Kommunikationsnetzwerke – Profile – Teil 3: Funktional sichere Übertragung bei Feldbussen – Allgemeine Regeln und Festlegungen für Profile
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/Standards entnommen, u. a.:

Norm/Standard	Beschreibung
IEC 60034-Reihe	Drehende elektrische Maschinen
IEC 61800-Reihe	Drehzahlveränderbare elektrische Umrichter
IEC 61158-Reihe	Industrielle Kommunikationsnetze – Feldbus für industrielle Steuerungssysteme

Des Weiteren kann der Begriff *Betriebsbereich* in Verbindung mit der Beschreibung spezifischer Gefahren verwendet werden und wird in diesem Fall für eine *Gefahrenzone* bzw. einen *Gefahrenbereich* in folgenden *Maschinenrichtlinien* definiert: 2006/42/EC und ISO 12100:2010.

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

---

# Einführung: Erweiterte Funktionen

## Inhalt dieses Abschnitts

Einführung.....	16
-----------------	----

## Überblick

Dieser Teil der Dokumentation enthält eine Beschreibung der verfügbaren Modi und Funktionalitäten sowie der Leistung der verschiedenen erweiterten Funktionen.

# Einführung

## Inhalt dieses Kapitels

Experten-E/A .....	17
Zuordnung der integrierten Experten-E/A .....	19
Allgemeine Informationen zur Funktionsbausteinverwaltung .....	21

## Überblick

In diesem Dokument finden Sie Beschreibungen der erweiterten EcoStruxure Machine Expert - Basic Funktionen und ihrer Beziehung zum M221 Experten-E/A- und PID-Support. Hier finden Sie Beschreibungen der Funktionen, Eigenschaften und Leistungsmerkmale der Eingänge und Ausgänge Fast Counter (%FC), High Speed Counter (%HSC), Pulse (%PLS), Pulse Width Modulation (%PWM) und Pulse Train Output (%PTO). Zudem finden Sie eine vollständige Beschreibung der erweiterten PID-Softwarefunktionalität. Informationen zu den benutzerdefinierten Funktionen und den benutzerdefinierten Funktionsbausteinen finden Sie unter Benutzerdefinierte Funktionen (siehe EcoStruxure Machine Expert - Basic, Betriebshandbuch) und Benutzerdefinierte Funktionsbausteine (siehe EcoStruxure Machine Expert - Basic, Betriebshandbuch).

Diese Funktionen stellen einfache und dabei leistungsstarke Lösungen für Ihre Anwendung bereit. 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 oder Software die geltenden örtlichen, regionalen oder 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 den erweiterten EcoStruxure Machine Expert - Basic Funktionen bereitstehende Funktionalität für die M221 Steuerungen 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.**

## Experten-E/A

### Einführung

Der M221 Logic Controller stellt Folgendes bereit:

- Vier Schnelleingänge (%I0.0, %I0.1, %I0.6 und %I0.7)
- Zwei Schnellausgänge in Steuerungsreferenzen, die Transistorausgänge enthalten (%Q0.0 und %Q0.1)
- Vier Schnellausgänge auf Steuerungsreferenzen TM221C40U und TM221CE40U (%Q0.0, %Q0.1, %Q0.2 und %Q0.3).

**HINWEIS:** Schnellausgangsfunktionen sind nicht auf Steuerungsreferenzen verfügbar, die Relaisausgänge haben.

Die logische Steuerung M221 unterstützt die folgenden E/A-Expertenfunktionen (abhängig von der Referenz):

Funktionen		Beschreibung
Zähler	Schneller Zähler, Seite 23	Die FC-Funktion kann schnelle Zählungen von Impulsen durchführen, die von Sensoren, Schaltern usw. ausgehen.
	High Speed Counter (Hochgeschwindigkeitszähler), Seite 28	Die HSC-Funktion kann schnelle Zählungen von Impulsen durchführen, die von Sensoren, Gebern, Schaltern usw. ausgehen, die an dedizierte Schnelleingänge angeschlossen sind.
Impulsgeneratoren	Pulse (Impuls), Seite 41	Die PLS-Funktion generiert ein Rechteckwellen-Signal auf dedizierten Ausgangskanälen.
	Impulsbreitenmodulation, Seite 48	Die PWM-Funktion generiert ein moduliertes Wellensignal auf dedizierten Ausgangskanälen mit variablem Arbeitszyklus.
	Impulswellenausgang, Seite 80	Die PTO-Funktion generiert Impulswellen-Ausgänge, um lineare, einachsige Stepper- oder Servoantriebe im Open-Loop-Betrieb zu steuern.
	Frequenzgenerator, Seite 157	Die Funktion FREQGEN generiert ein Rechtecksignal auf einem dedizierten Ausgangskanal mit programmierbarer Frequenz und einem Arbeitszyklus gleich 50 %.

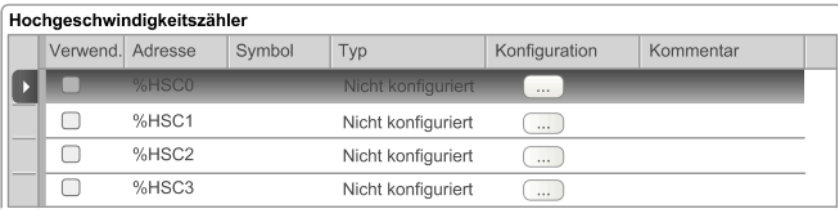
**HINWEIS:**

- Wenn ein Eingang für Ausführung/Stopp eingesetzt wird, kann er nicht von einer Expertenfunktion verwendet werden.
- Wenn ein Eingang zum Alarm eingesetzt wird, kann er nicht von einer Expertenfunktion verwendet werden.

Weitere Informationen finden Sie unter Konfiguration der integrierten Ein-/Ausgänge (siehe Modicon M221, Logic Controller, Programmierhandbuch).

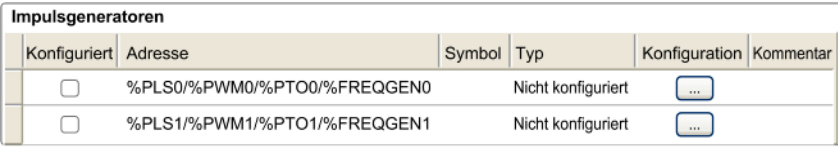
## Konfigurieren einer Experteneingangsfunktion

Gehen Sie wie folgt vor, um eine Experteneingangsfunktion zu konfigurieren:

Schritt	Beschreibung
1	<p>Klicken Sie in der Hardware-Baumstruktur auf den Knoten <b>Hochgeschwindigkeitszähler</b>.  <b>Ergebnis:</b> Die Liste <b>Hochgeschwindigkeitszähler</b> wird angezeigt:</p> 
2	<p>Klicken Sie auf ... in der Spalte <b>Konfiguration</b>, um den Typ des Hochgeschwindigkeitszählers auszuwählen und um das Fenster <b>Hochgeschwindigkeitszähler-Assistent</b> aufzurufen.</p>

## Konfigurieren einer Expertenausgangsfunktion

Gehen Sie wie folgt vor, um eine Expertenausgangsfunktion zu konfigurieren:

Schritt	Beschreibung
1	<p>Klicken Sie in der Hardware-Baumstruktur auf den Knoten <b>Impulsgeneratoren</b>.  <b>Ergebnis:</b> Die Liste <b>Impulsgeneratoren</b> wird angezeigt:</p> 
2	<p>Klicken Sie auf ... in der Spalte <b>Konfiguration</b>, um den Typ des Impulsgenerators auszuwählen und um das Fenster <b>Impulsgenerator-Assistent</b> aufzurufen.</p>

## Konfigurationseigenschaften der Experten-E/A-Funktion

- Eingänge können über Standard-Speichervariablen gelesen werden, auch wenn sie gemeinsam mit E/A-Expertenfunktionen konfiguriert sind.
- Die Kurzschlussverwaltung gilt dennoch für alle Expertenausgänge.
- Alle E/A, die nicht von E/A-Expertenfunktionen verwendet werden, können als Standard-E/A verwendet werden.
- Auf Ausgänge, die von *Pulse*, *Pulse Train Output*, *Pulse Width Modulation* und *High Speed Counters* verwendet werden, kann nur über den E/A-Experten-Funktionsbaustein zugegriffen werden. Sie können nicht direkt von der Anwendung gelesen oder geschrieben werden.

# Zuordnung der integrierten Experten-E/A

## Eingangszuordnung für Expertenfunktionen für M221 Logic Controller

Die integrierten Digitaleingänge können Funktionen zugewiesen werden (Run/ Stop, Speicherung, Ereignis, Schnellzähler, HSC, PTO). Die keinen Funktionen zugewiesenen Eingänge werden als normale Eingänge verwendet. In der folgenden Tabelle werden die möglichen Zuweisungen der integrierten Digitaleingänge des M221 Logic Controller beschrieben:

Funktion		Einfache Eingangsfunktion			Erweiterte Eingangsfunktion		
		Run/Stop	Statusspeicherung	Ereignis	Schneller Zähler	HSC	PTO <sup>(3)</sup>
Schnelleingang	%I0.0	X	–	–	–	%HSC0	–
	%I0.1	X	–	–	–	%HSC0 oder %HSC2 <sup>(1)</sup>	–
Normaler Eingang	%I0.2	X	X	X	%FC0	Preset-Eingang für %HSC0	Ref or probe for %PTO0 to %PTO3
	%I0.3	X	X	X	%FC1	Erfassungseingang für %HSC0	
	%I0.4	X	X	X	%FC2	Erfassungseingang für %HSC1	
	%I0.5	X	X	X	%FC3	Preset-Eingang für %HSC1	
Schnelleingang	%I0.6	X	–	–	–	%HSC1	–
	%I0.7	X	–	–	–	%HSC1 or %HSC3 <sup>(2)</sup>	–
Normaler Eingang (abhängig von Steuerungsreferenz)	%I0.8	X	–	–	–	–	Ref oder Probe für %PTO0 bis %PTO3 auf TM221C40U- und TM221CE40U-Steuerungen
	%I0.9	X	–	–	–	–	
	%I0.10	X	–	–	–	–	–
	%I0.11	X	–	–	–	–	–
	%I0.12	X	–	–	–	–	–
	%I0.13	X	–	–	–	–	–
	%I0.14	X	–	–	–	–	–
	%I0.15	X	–	–	–	–	–
	%I0.16	X	–	–	–	–	–
	%I0.17	X	–	–	–	–	–
	%I0.18	X	–	–	–	–	–
	%I0.19	X	–	–	–	–	–
	%I0.20	X	–	–	–	–	–
%I0.21	X	–	–	–	–	–	
%I0.22	X	–	–	–	–	–	
%I0.23	X	–	–	–	–	–	

X Ja

– Nein

<sup>(1)</sup> %HSC2 ist verfügbar, wenn %HSC0 als Einphasig oder Not Configured konfiguriert ist.

<sup>(2)</sup> %HSC3 ist verfügbar, wenn %HSC1 als Einphasig oder Not Configured konfiguriert ist.

<sup>(3)</sup> PTO-Funktion ist auf Steuerungsreferenzen verfügbar, die Transistorausgänge haben.

## Ausgangszuordnung bei Expertenfunktionen für M221 Logic Controller

Die nachstehenden Informationen gelten für die Standard- und die schnellen Transistorausgänge am M221 Logic Controller:

Funktion		Alarmausgang	HSC	PLS / PWM / PTO / FREQGEN
<b>Schneller Ausgang<sup>(1)</sup></b>	%Q0.0	X	–	<ul style="list-style-type: none"> <li>• %PLS0</li> <li>• %PWM0</li> <li>• %PTO0</li> <li>• %FREQGEN0</li> </ul>
	%Q0.1	X	–	<ul style="list-style-type: none"> <li>• %PLS1</li> <li>• %PWM1</li> <li>• %PTO<sup>(2)</sup></li> <li>• %FREQGEN1</li> </ul>
<b>Normaler Ausgang<sup>(3)</sup></b> (abhängig von Steuerungsreferenz)	%Q0.2	X	Reflexausgang 0 für %HSC0 oder %HSC2	<ul style="list-style-type: none"> <li>• %PTO<sup>(4)</sup></li> <li>• %FREQGEN2</li> </ul>
	%Q0.3	X	Reflexausgang 1 für %HSC0 oder %HSC2	<ul style="list-style-type: none"> <li>• %PTO<sup>(5)</sup></li> <li>• %FREQGEN3</li> </ul>
	%Q0.4	X	Reflexausgang 0 für %HSC1 oder %HSC3	%PTOx Richtung
	%Q0.5	X	Reflex output 1 for %HSC1 or %HSC3	%PTOx Richtung
	%Q0.6	X	–	%PTOx Richtung
	%Q0.7	X	–	%PTOx Richtung
	%Q0.8	–	–	%PTOx Richtung
	%Q0.9	–	–	%PTOx Richtung
	%Q0.10	–	–	%PTOx Richtung
	%Q0.11	–	–	%PTOx Richtung
	%Q0.12	–	–	%PTOx Richtung
	%Q0.13	–	–	%PTOx Richtung
	%Q0.14	–	–	%PTOx Richtung
	%Q0.15	–	–	%PTOx Richtung

**X** Ja  
**–** Nein

(1) Schnellausgangsfunktionen sind nur auf Steuerungsreferenzen verfügbar, die Transistorausgänge haben.

(2) %PTO0-Richtung in CW/CCW-Ausgangsmodus oder %PTO1 (nicht verfügbar, wenn %PTO0 in CW/CCW-Ausgangsmodus konfiguriert ist) oder %PTOx-Richtung in anderen Fällen.

(3) %Q0.2 Und %Q0.3 sind Schnellausgänge bei TM221C40U- und TM221CE40U-Steuerungen

(4) %PTO2 an Steuerungen vom Typ TM221C40U und TM221CE40U oder %PTOx-Richtung in anderen Fällen.

(5) %PTO2-Richtung im CW/CCW-Ausgangsmodus an Steuerungen vom Typ TM221C40U und TM221CE40U oder %PTO3 (nicht verfügbar, wenn %PTO2 im CW/CCW-Ausgangsmodus konfiguriert ist) an Steuerungen vom Typ TM221C40U und TM221CE40U %PTOx-Richtung in anderen Fällen.

# Allgemeine Informationen zur Funktionsbausteinverwaltung

## Verwaltung der Funktionsbausteineingänge und Eingangsobjekte

Die Variablen (Funktionsbausteineingänge und Eingangsobjekte) werden bei einer steigenden Flanke am Eingang *Execute* verwendet. Um eine Variable zu ändern, müssen Sie die Eingangsvariablen ändern und den Funktionsbaustein erneut auslösen. Es gibt jedoch einige Funktionsbausteine, die eine Funktion für kontinuierliche Aktualisierungen besitzen.

## Verwaltung der Funktionsbausteinausgänge und Ausgangsobjekte

Die Ausgänge *Done*, *Error*, *Busy* und *CmdAborted* 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 am Eingang *Execute* wird der Ausgang *Busy* auf TRUE gesetzt. Der Ausgang bleibt für die Dauer der Ausführung des Funktionsbausteins TRUE und wird bei einer steigenden Flanke an einem der anderen Ausgänge (*Done*, *Error* und *CmdAborted*) zurückgesetzt.

Der Ausgang *Done* ist TRUE, sobald die Ausführung des Funktionsbausteins erfolgreich abgeschlossen wurde.

Bei Feststellung eines Fehlers wird der Funktionsbaustein beendet, indem der Ausgang *Error* auf TRUE gesetzt wird und der Fehlercode im Ausgang *ErrId* enthalten ist.

Die Ausgänge *Done*, *Error* und *CmdAborted* werden auf TRUE oder FALSE gesetzt mit der fallenden Flanke des Eingangs *Execute* gemäß den folgenden Bedingungen:

- werden für einen Task-Zyklus gesetzt, wenn die Ausführung des Funktionsbausteins abgeschlossen ist und der Eingang *Execute* FALSE ist, und dann auf die Standardwerte zurückgesetzt.
- behalten ihren Wert bei, wenn die Ausführung des Funktionsbausteins abgeschlossen ist und der Eingang *Execute* TRUE 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. Der neue Befehl wird jedoch am Funktionsbaustein gestartet (Status ist *Busy*).

## Fehlerbehandlung

Alle Funktionsbausteine haben zwei Ausgänge, die einen Fehler bei der Ausführung des Funktionsbausteins melden können:

- *Error*= Die steigende Flanke dieses Ausgangs zeigt an, dass ein Fehler erkannt wurde.
- *ErrID*= Der Fehlercode des erkannten Fehlers.

---

# Erweiterte Eingangsfunktionen für Experten

## Inhalt dieses Abschnitts

Schnellzähler (%FC) .....	23
Hochgeschwindigkeitszähler (%HSC).....	28

## Überblick

In diesem Abschnitt werden die erweiterten Eingangsfunktionen für Experten beschrieben.

# Schnellzähler (%FC)

## Inhalt dieses Kapitels

Beschreibung.....	23
Konfiguration .....	24
Programmierbeispiel .....	26

## Verwendung der Schnellzähler-Funktionsbausteine

Dieser Abschnitt enthält Beschreibungen und Programmierrichtlinien für die Verwendung von *Fast Counter*-Funktionsbausteinen.

## Beschreibung

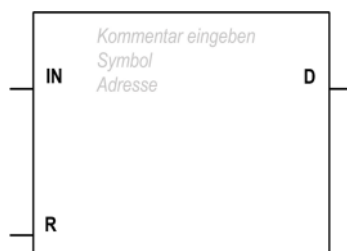
### Einführung

Der *Fast Counter*-Funktionsbaustein **1123** dient entweder als Aufwärts- oder Abwärtszähler. Er kann die steigenden Flanken an Digitaleingängen mit Frequenzen von bis zu 5 kHz im Einzelwort- oder Doppelwort-Verarbeitungsmodus zählen. Da die *Fast Counter*-Funktionsbausteine von bestimmten Hardware-Interrupts verwaltet werden, richtet sich die Beibehaltung von Abtastraten mit maximaler Frequenz nach der jeweiligen Anwendung und der Hardwarekonfiguration.

Die *Fast Counter*-Funktionsbausteine %FC0, %FC1, %FC2, und %FC3 verwenden die dedizierte Eingänge %IO.2, %IO.3, %IO.4 bzw. %IO.5. Diese Bits sind nicht für eine exklusive Nutzung reserviert. Bei der Zuordnung dieser zweckbestimmten Ressourcen muss die Verwendung durch andere Funktionsbausteine berücksichtigt werden.

## Abbildung

Die folgende Abbildung zeigt einen Funktionsbaustein *Fast Counter* im Einzelwortmodus:



## Eingänge

Der *Fast Counter*-Funktionsbaustein verfügt über folgende Eingänge:

Bezeichnung	Beschreibung	Wert
IN	Aktivieren	Im Zustand 1 wird der Wert entsprechend den Impulsen am physischen Eingang aktualisiert. Im Zustand 0 verbleibt der Wert auf dem letzten Wert.
R	Reset (optional)	Dient der Initialisierung des Bausteins. Im Zustand 1: <ul style="list-style-type: none"> <li>Die Werte %FC.P und %FC.PD werden berücksichtigt.</li> <li>Der Wert wird auf 0 zurückgesetzt, wenn ein Aufwärtszähler konfiguriert ist, oder er wird auf %FC.P oder %FC.PD gesetzt, wenn ein Abwärtszähler konfiguriert ist.</li> <li>Das Fertig-Bit %FC.D wird auf den voreingestellten Wert zurückgesetzt.</li> </ul>

## Ausgänge

Der *Fast Counter*-Funktionsbaustein verfügt über den folgenden Ausgang:

Bezeichnung	Beschreibung	Wert
D	Fertig (%FCi.D)	Dieses Bit wird auf 1 gesetzt, wenn <ul style="list-style-type: none"> <li>%FCi.V oder %FCi.VD den vordefinierten Wert %FCi.P erreicht oder %FCi.PD als Aufwärtszähler konfiguriert wurde.</li> <li>oder wenn %FCi.V bzw. %FCi.VD den Wert 0 erreicht und ein Abwärtszähler konfiguriert wurde.</li> </ul> Dieses schreibgeschützte Bit wird nur zurückgesetzt, wenn %FCi.R auf 1 gesetzt wird.

## Konfiguration

### Parameter

Halten Sie sich für die Konfiguration der Parameter an die unter Konfiguration eines Funktionsbausteins (siehe EcoStruxure Machine Expert - Basic, – Bibliothekshandbuch zu Generischen Funktionen) beschriebene Vorgehensweise und lesen Sie sich die Beschreibung der Speicherzuweisungsmodi im EcoStruxure Machine Expert - Basic Betriebshandbuch (siehe EcoStruxure Machine Expert - Basic, Betriebshandbuch) durch.

Der *Fast Counter*-Funktionsbaustein verfügt über folgende Parameter:

Parameter	Beschreibung	Wert
<b>Verwendet</b>	Adresse verwendet	Wenn diese Option ausgewählt ist, wird die Adresse derzeit in einem Programm verwendet.
<b>Adresse</b>	%FCi <i>Fast Counter</i> -Adresse	Die Instanz-ID, wobei i einer Zahl zwischen 0 und der Gesamtzahl der mit der Steuerung verfügbaren Objekte entspricht. Siehe Tabelle Maximale Anzahl Objekte (siehe Modicon M221, Logic Controller, Programmierhandbuch) für die maximale Anzahl der <i>Schnellen Zähler</i> .
<b>Eingang</b>	%IO.i	Der dedizierte Eingang, der dieser Instanz des Funktionsbausteins zugeordnet ist.  %IO.2...%IO.5
<b>Symbol</b>	Symbol	Das diesem Objekt zugeordnete Symbol. Detaillierte Informationen finden Sie im EcoStruxure Machine Expert - Basic Betriebshandbuch (Definition und Verwendung von Symbolen).
<b>Konfiguriert</b>	Aufwärts- oder Abwärtszählen	Wird auf einen der folgenden Werte gesetzt: <ul style="list-style-type: none"> <li>• <b>Nicht verwendet</b></li> <li>• <b>Aufwärtszähler</b></li> <li>• <b>Abwärtszähler</b></li> </ul>
<b>Preset</b>	Preset-Wert (%FCi.P oder %FCi.PD)	Der Initialwert kann gesetzt werden: <ul style="list-style-type: none"> <li>• über ein zugeordnetes Objekt %FCi.P von 1 bis 65535 im Einzelwortformat</li> <li>• Verwenden eines zugeordneten Objekts %FCi.PD von 1 bis 4294967295 im Doppelwortmodus.</li> </ul>
<b>Double Word</b>	Doppelwortmodus	Wenn dieser Wert ausgewählt ist, verwenden Sie den Doppelwortmodus. Andernfalls verwenden Sie den Einzelwortmodus.
<b>Kommentar</b>	Kommentar	Diesem Objekt kann ein optionaler Kommentar zugeordnet werden.  Doppelklicken Sie in die Spalte <b>Kommentar</b> und geben Sie einen Kommentar ein.

## Objekte

Der *Fast Counter*-Funktionsbaustein ist mit folgenden Objekten verknüpft:

Objekt	Beschreibung	Wert
%FCi.V %FCi.VD	Aktueller Wert	Der aktuelle Wert wird je nach eingestellter Zählrichtung (Aufwärts oder Abwärts) inkrementiert oder dekrementiert. Beim Aufwärtszählen wird der aktuelle Zählwert bis 65535 im Einzelwortmodus (%FCi.V) und 4294967295 im Doppelwortmodus (%FCi.VD) aktualisiert. Für die Abwärtszählung ist der aktuelle Wert der voreingestellte Wert %FC.P oder %FC.PD und kann bis 0 verringert werden.
%FCi.P %FCi.PD	Preset-Wert	Ein neuer Preset-Wert wird nur dann berücksichtigt, wenn der R-Eingang aktiv ist. Siehe hierzu die Beschreibung in der obigen Parametertabelle.
%FCi.D	Fertig	Siehe Beschreibung in der Tabelle der Ausgänge oben.

## Operation

In der nachstehenden Tabelle werden die Hauptphasen der Operationen des Funktionsbausteins *Fast Counter* beschrieben:

Operation	Aktion	Ergebnis
Aufwärtszählen	Eine steigende Flanke tritt am Aufwärtszähleingang auf.	Der aktuelle Wert %FCi.V wird um 1 Einheit inkrementiert.
	Der Preset-Wert erreicht %FCi.P oder %FCi.PD.	Das „Fertig“-Ausgangsbit %FCi.D wird auf 1 gesetzt.
Abwärtszählen	Eine steigende Flanke tritt am Abwärtszähleingang auf.	Der aktuelle Wert %FCi.V wird um 1 Einheit dekrementiert.
	Der Wert entspricht 0	Das „Fertig“-Ausgangsbit %FCi.D wird auf 1 gesetzt.

## Sonderfälle

Die folgende Tabelle enthält eine Liste von Sonderfällen bei Verwendung des Funktionsbausteins *Fast Counter*.

Sonderfall	Beschreibung
Auswirkung eines Kaltstarts (%S0=1)	Setzt die Attribute des <i>Fast Counter</i> mit den zurück, die konfiguriert sind, oder mit der Benutzeranwendung (siehe Modicon M221, Logic Controller, Programmierhandbuch).
Auswirkung eines Warmstarts (%S1=1)	Keine Auswirkung (siehe Modicon M221 Logic Controller, Programmierhandbuch).
Auswirkung eines Steuerungsstopps	Der <i>Fast Counter</i> stoppt das Zählen, wenn die Steuerung in den <i>STOPPED</i> -Zustand versetzt wird und zählt weiter, wenn sie in den <i>RUNNING</i> -Zustand zurückkehrt. Der Zähler nimmt das Zählen beim letzten Wert auf, bevor in den <i>STOPPED</i> -Zustand übergegangen wurde.

## Programmierbeispiel

### Einführung

In diesem Beispiel zählt die Anwendung eine Anzahl von Elementen bis 5000, wobei %I0.1 auf 1 gesetzt ist. Der Eingang für %FC1 ist der dedizierte Eingang %I0.3. Ist der voreingestellte Wert erreicht, wird %FC1.D auf 1 gesetzt und behält diesen Wert, bis %FC1.R durch das Ergebnis der AND auf %I0.2 und %M0 gesetzt wird.

## Programmierung

Dieses Beispiel entspricht einem *Fast Counter*-Funktionsbaustein:

Programm- baustein	Anweisung
0	BLK %FC1 LD %I0.1 IN LD %I0.2 AND %M0 R OUT_BLK LD D ST %Q0.0 END_BLK

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

# Hochgeschwindigkeitszähler (%HSC)

## Inhalt dieses Kapitels

Beschreibung.....	28
Hochgeschwindigkeitszähler in Zählmodi .....	32
Hochgeschwindigkeitszähler im Frequenzmesser-Modus.....	38

## Verwendung der Funktionsbausteine des Typs Hochgeschwindigkeitszähler

Dieser Abschnitt enthält Beschreibungen und Programmierrichtlinien für die Verwendung von *High Speed Counter*-Funktionsbausteinen.

## Beschreibung

### Einführung

Der Funktionsbaustein des Typs *High Speed Counter* **11123** kann in EcoStruxure Machine Expert - Basic für die folgenden Funktionen eingerichtet werden:

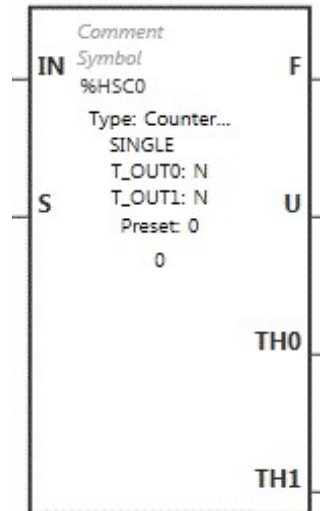
- Zweiphasig [Impuls / Richtung]
- Zweiphasig [Quadratur X1]
- Zweiphasig [Quadratur X2]
- Zweiphasig [Quadratur X4]
- Einphasig
- Frequenzmesser

Der Funktionsbaustein des Typs *High Speed Counter* funktioniert mit einer maximalen Frequenz von 100 kHz in allen Zählmodi und in einem Bereich von 0 bis 65535 im Einzelwortmodus bzw. von 0 bis 4294967295 im Doppelwortmodus.

Der Funktionsbaustein des Typs *High Speed Counter* verwendet dedizierte E/A-Eingänge sowie Hilfsein- und -ausgänge. Weitere Informationen zu Ein- und Ausgängen finden Sie im M221 Logic Controller - Hardwarehandbuch.

Bevor Sie eine Instanz des Funktionsbausteins verwenden, müssen Sie die Funktion *High Speed Counter* auf der Registerkarte **Konfiguration** mithilfe des **Hochgeschwindigkeitszähler-Assistenten** initialisieren. Siehe Konfigurieren von Hochgeschwindigkeitszählern (siehe Modicon M221 Logic Controller, Programmierhandbuch).

## Grafische Darstellung



## Eingänge

Der High Speed Counter-Funktionsbaustein verfügt über folgende Eingänge:

Bezeichnung	Beschreibung	Wert
<b>IN</b>	Aktivieren (erforderlich) Im Zustand 1 sind die Zählfunktion bzw. die Frequenzmessung aktiviert. Im Zustand 0 verbleibt der aktuelle Wert auf dem letzten Wert.	0 oder 1
<b>S</b>	Preset-Eingang. Im Zustand 1: <ul style="list-style-type: none"> <li>• Initialisiert den Wert mit dem Preset-Wert für:                             <ul style="list-style-type: none"> <li>◦ Zweiphasig [Quadratur X1],</li> <li>◦ Zweiphasig [Quadratur X2],</li> <li>◦ Zweiphasig [Quadratur X4] oder</li> <li>◦ Zweiphasig [Impuls / Richtung] mit aktiver Abwärtsfunktion</li> </ul> </li> <li>• Setzt den Wert auf 0 zurück für:                             <ul style="list-style-type: none"> <li>◦ Einphasig oder</li> <li>◦ Zweiphasig [Impuls / Richtung] mit aktiver Aufwärtsfunktion</li> </ul> </li> </ul> Außerdem werden die Schwellenwertausgänge initialisiert und alle vom Benutzer vorgenommenen Änderungen an den Schwellenwerten berücksichtigt, die über das Eigenschaftsfenster oder das Benutzerprogramm eingestellt wurden.	0 oder 1

Der High Speed Counter-Funktionsbaustein ist mit folgenden Eingangsobjekten verknüpft:

Objekt	Typ	Beschreibung	Wert
%HSCi.P	WORD	Preset-Wert	Siehe Hilfeeingänge, Seite 33
%HSCi.PD	DOUBLE WORD		
%HSCi.S0	WORD	Schwellenwert 0	Siehe Ausgangsschwellenwerte in Zählmodi, Seite 33
%HSCi.S0D	DOUBLE WORD		
%HSCi.S1	WORD	Schwellenwert 1	Siehe Ausgangsschwellenwerte in Zählmodi, Seite 33
%HSCi.S1D			

Objekt	Typ	Beschreibung	Wert
	DOUBLE WORD		
%HSCi.T	WORD	Zeitbasis	Siehe Hochgeschwindigkeitszähler im Frequenzmessermodus, Seite 38
%HSCi.R	BOOL	Reflexausgang 0 freigeben	Im Zustand 1 wird Reflexausgang 0 aktiviert.
%HSCi.S	BOOL	Reflexausgang 1 freigeben	Im Zustand 1 wird Reflexausgang 1 aktiviert.

**HINWEIS:** Die Bits %HSCi.R und %HSCi.S aktivieren bzw. deaktivieren die Reflexausgänge nur dann, wenn der HSC-Funktionsbaustein aktiviert ist, also wenn %HSCi.IN auf 1 gesetzt ist.

## Ausgänge

Der Funktionsbaustein *High Speed Counter* verfügt über folgende Ausgänge:

Bezeichnung	Beschreibung	Wert
<b>F</b>	Überlauf Wird auf 1 gesetzt, falls ein arithmetischer Überlauf auftritt.	0 oder 1
<b>U</b>	Zählrichtung Wird vom System gesetzt. Dieses Bit wird von den <i>Dual Phase</i> -Funktionen zum Auf-/Abwärtszählen verwendet, um die Zählrichtung anzuzeigen.	0: Abwärtszählen 1: Aufwärtszählen
<b>TH0</b>	Schwellenwert-Bit 0 Wird auf 1 gesetzt, wenn der aktuelle Wert größer ist als der Schwellenwert S0 bzw. diesem entspricht (%HSCi.S0). Testen Sie dieses Bit nur einmal im Programm, da es in Echtzeit aktualisiert wird. Die Benutzeranwendung ist verantwortlich für die Gültigkeit des Werts bei der Verwendung.	0 oder 1
<b>TH1</b>	Schwellenwert-Bit 1 Wird auf 1 gesetzt, wenn der aktuelle Wert größer ist als der Schwellenwert S1 bzw. diesem entspricht (%HSCi.S1). Testen Sie dieses Bit nur einmal im Programm, da es in Echtzeit aktualisiert wird.	0 oder 1

Der *High Speed Counter*-Funktionsbaustein ist mit folgenden Ausgangsobjekten verknüpft:

Objekt	Typ	Beschreibung	Wert
%HSCi.V %HSCi.VD	WORD DOUBLE WORD	Aktueller Wert	Siehe Hochgeschwindigkeitszähler in Zählmodi, Seite 32 und Hochgeschwindigkeitszähler im Frequenzmesser-Modus, Seite 38  <b>HINWEIS:</b> Der aktuelle Wert kann unabhängig vom Funktionsblock-Aufruf %HSC aktualisiert werden. %HSCi.V/%HSCi.VD kann im selben Taskzyklus zweimal gelesen werden und unterschiedliche Ergebnisse liefern.
%HSCi.C %HSCi.CD	WORD DOUBLE WORD	Erfassungswert	Siehe Hilfeeingänge, Seite 33
%HSCi.U	BOOL	Zählrichtung	0: Abwärtszählen 1: Aufwärtszählen
%HSCi.F	BOOL	Überlauf	0: Kein Überlauf 1: Zählerüberlauf

## Eigenschaften

Der *High Speed Counter*-Funktionsbaustein hat die folgenden Eigenschaften:

Eigen-schaft	Wert	Beschreibung
Verwendet	Aktiviertes/deaktiviertes Kontrollkästchen	Gibt an, ob die Adresse verwendet wird.
Adresse	%HSCi, wobei i die Werte 0 bis 3 annehmen kann, abhängig von den konfigurierten Zählertypen.	i ist der Bezeichner für die Instanz.  Die maximale Anzahl an %HSC-Objekten finden Sie in der Tabelle Maximale Anzahl Objekte (siehe Modicon M221, Logic Controller, Programmierhandbuch).
Symbol	Benutzerdefinierter Text	Das Symbol, das dieses Objekt eindeutig kennzeichnet. Weitere Informationen hierzu finden Sie unter Definieren und Verwenden von Symbolen (siehe EcoStruxure Machine Expert – Basic, Betriebshandbuch).
Preset	<ul style="list-style-type: none"> <li>Von 0 bis 65535 für %HSCi.P</li> <li>Von 0 bis 4294967295 für %HSCi.PD</li> </ul>	Preset-Wert zur Initialisierung des HSC-Istwertes %HSCi.P, %HSCi.PD).  Nicht gültig für die Funktion Frequenzmesser.
S0	<ul style="list-style-type: none"> <li>Von 1 bis 65535 für %HSCi.S0</li> <li>Von 1 bis 4294967295 für %HSCi.S0D</li> </ul>	Der Schwellenwert 0 wird zum Vergleich mit dem Istwert verwendet.  Der Wert von S0 muss kleiner sein als S1 (%HSCi.S1).
S1	<ul style="list-style-type: none"> <li>Von 2 bis 65535 für %HSCi.S1</li> <li>Von 2 bis 4294967295 für %HSCi.S1D</li> </ul>	Der Schwellenwert 1 wird zum Vergleich mit dem Istwert verwendet.  Der Wert von S1 muss größer sein als S0 (%HSCi.S0).
Zeitbasis	100 ms oder 1 s für %HSCi.T	Zeitbasis der Frequenzmessung.
Kommen-tar	Benutzerdefinierter Text	Ein mit diesem Objekt zu verknüpfender Kommentar.

## Sonderfälle

Die folgende Tabelle enthält eine Liste von Sonderfällen für den Betrieb des Funktionsbausteins *High Speed Counter*.

Sonderfall	Beschreibung
Auswirkung eines Kaltstarts (%S0=1)	Setzt die <i>High Speed Counter</i> -Attribute auf die vom Programm konfigurierten Werte zurück.
Auswirkung eines Warmstarts (%S1=1)	Keine Auswirkungen.
Auswirkung eines Steuerungsstopps	<p><i>High Speed Counter</i> stoppt seine Funktion und die Ausgänge verbleiben im aktuellen Zustand.</p> <p><b>HINWEIS:</b> Wenn die Steuerung stoppt, werden die Reflexausgänge auf 0 gesetzt, wenn für die Ausgänge <b>Werte beibehalten</b> ausgewählt wurde. Wenn <b>Werte beibehalten</b> jedoch nicht ausgewählt ist, nehmen die Reflexausgänge die Fehlerausweichwerte an. Detaillierte Informationen zur Konfigurierung des Fehlerausweichverhaltens finden Sie unter Fehlerausweichverhalten (siehe EcoStruxure Machine Expert - Basic, Betriebshandbuch).</p>

## Hochgeschwindigkeitszähler in Zählmodi

### Einführung

Der Funktionsbaustein des Typs *High Speed Counter* funktioniert mit einer maximalen Frequenz von 100 kHz in allen Zählmodi und in einem Bereich von 0 bis 65535 im Einzelwortmodus bzw. von 0 bis 4294967295 im Doppelwortmodus.

Die zu zählenden Impulse werden auf folgende Weise angelegt:

Funktion	Beschreibung	Eingangstyp	%HSC0	%HSC1	%HSC2	%HSC3
Zweiphasig [Impuls / Richtung]	Die Impulse werden auf die mit <b>Impulseingang</b> verknüpften physischen Eingänge angewendet.	<b>Impulseingang</b>	%I0.0	%I0.6	–	–
	Die aktuelle Funktionsweise (Auf-/Abwärtszählen) wird vom Zustand des <b>Richtungseingangs</b> vorgegeben: <ul style="list-style-type: none"> <li>• 0 = Aufwärtszählen</li> <li>• 1 = Abwärtszählen</li> </ul>	<b>Richtungseingang</b>	%I0.1	%I0.7	–	–
Zweiphasig [Quadratur X1], Zweiphasig [Quadratur X2] oder Zweiphasig [Quadratur X4]	Die 2 Phasen des Gebers werden auf die mit <b>Impulseingang Phase A</b> und <b>Impulseingang Phase B</b> verknüpften physischen Eingänge angewendet.	<b>Impulseingang Phase A</b>	%I0.0	%I0.6	–	–
		<b>Impulseingang Phase B</b>	%I0.1	%I0.7	–	–
Einphasig	Die Impulse werden auf die mit <b>Impulseingang</b> verknüpften physischen Eingänge angewendet.	<b>Impulseingang</b>	%I0.0	%I0.6	%I0.1	%I0.7

**HINWEIS:** Die E/A-Zuordnung ist zwischen der Twido-Plattform und dem Bereich des M221 Logic Controller unterschiedlich. Auf dem M221 Logic Controller ist der Hauptimpulseingang %I0.0 für %HSC0 und %I0.6 für %HSC1. Auf der Twido-Plattform ist der Hauptimpulseingang %I0.1 für %HSC0 und %I0.7 für %HSC1.

## Ausgangsschwellenwerte

Während des Zählens wird der aktuelle Wert mit zwei Schwellenwerten verglichen:  $\%HSCi.S0$  oder  $\%HSCi.S0D$  und  $\%HSCi.S1$  oder  $\%HSCi.S1D$ .

Änderungen an diesen Schwellenwerten werden unabhängig vom Wert des **Preset**-Eingangs berücksichtigt.

Änderungen des Schwellenwerts werden im Logic Controller (Objekte  $\%HSCi.S0$ ,  $\%HSCi.S1$ ,  $\%HSCi.S0D$  and  $\%HSCi.S1D$ ), jedoch nicht im Fenster **Konfiguration** von EcoStruxure Machine Expert - Basic gespeichert.

Je nach Ergebnis der Vergleiche werden die Bitobjekte  $\%HSCi.TH0$  und  $\%HSCi.TH1$ :

- auf 1 gesetzt, wenn der aktuelle Wert größer als oder gleich dem entsprechenden Schwellenwert ist,
- auf 0 zurückgesetzt, wenn der aktuelle Wert kleiner als der entsprechenden Schwellenwert ist.

Physische Reflexausgänge können so konfiguriert werden, dass sie im Kontext der Vergleichsergebnisse der Schwellenwerte und der aktuellen Werte der Zähler differentiell reagieren.

**HINWEIS:** Es können keine, 1 oder 2 Reflexausgänge konfiguriert werden.

Weitere Informationen zur Konfiguration von Reflexausgängen finden Sie unter Konfigurieren von Zweiphasen- und Einphasen-Zählern (siehe Modicon M221, Logic Controller, Programmierhandbuch).

$\%HSCi.U$  ist ein Ausgang des FB, der die Richtung des zugehörigen Zählers angibt (1 bei AUF, 0 bei AB).

## Hilfseingänge

Die Zählvorgänge werden bei steigenden Flanken von Impulsen durchgeführt, und zwar nur dann, wenn der Zähl-Funktionsbaustein aktiviert ist (**IN**-Eingang bei Zustand 1).

Es gibt zwei Eingänge, die für den Zählmodus verwendet werden:

**Erfassungseingang** und **Preset-Eingang**:

- Eine steigende Flanke am **Erfassungseingang** wird verwendet, um den aktuellen Wert ( $\%HSCi.V$  oder  $\%HSCi.VD$ ) zu erfassen und ihn in  $\%HSCi.C$  oder  $\%HSCi.CD$  zu speichern. Die Erfassungseingänge sind angegeben als  $\%I0.3$  für  $\%HSC0$  und  $\%I0.4$  für  $\%HSC1$ , falls verfügbar.
- Eine steigende Flanke am **Preset-Eingang** initialisiert den  $\%HSCi.V$  - oder  $\%HSCi.VD$ -Wert mit dem Preset-Wert für:
  - Zweiphasig [Quadratur X1]
  - Zweiphasig [Quadratur X2]
  - Zweiphasig [Quadratur X4]
  - Zweiphasig [Impuls / Richtung] mit aktiver Abwärtsfunktion

Der **Preset-Eingang** setzt den Wert auf 0 zurück für:

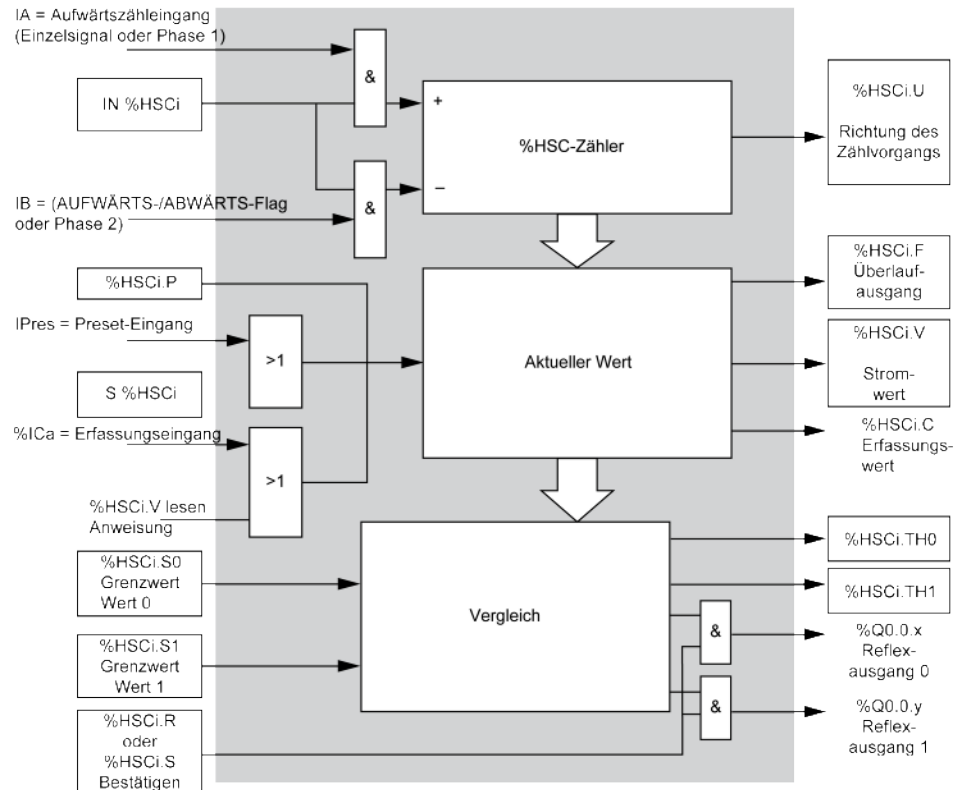
- Einphasig
- Zweiphasig [Impuls / Richtung] mit aktiver Aufwärtsfunktion

Wenn der Hilfs-**Preset-Eingang** auf 1 und der Eingang *IN* auf 0 gesetzt ist (die Funktion ist gehemmt), werden die Ausgänge nicht überwacht und behalten ihre Werte bei.

**HINWEIS:**  $\%HSCi.F$  wird ebenfalls auf 0 gesetzt. Der **Preset-Eingang** wird als  $\%I0.2$  für  $\%HSC0$  und/oder  $\%I0.5$  für  $\%HSC1$  angegeben.

## Aktion

Die folgende Abbildung zeigt das Ablaufdiagramm des Zählmodus im Einzelwortmodus (im Doppelwortmodus müssen die Variablen des Doppelwortmodus verwendet werden):



**HINWEIS:** Reflexausgänge werden unabhängig von der Zykluszeit der Steuerung verwaltet.

## Zeitdiagramm für Zweiphasig [Impuls / Richtung]

Beispiel für Reflexausgangskonfiguration:

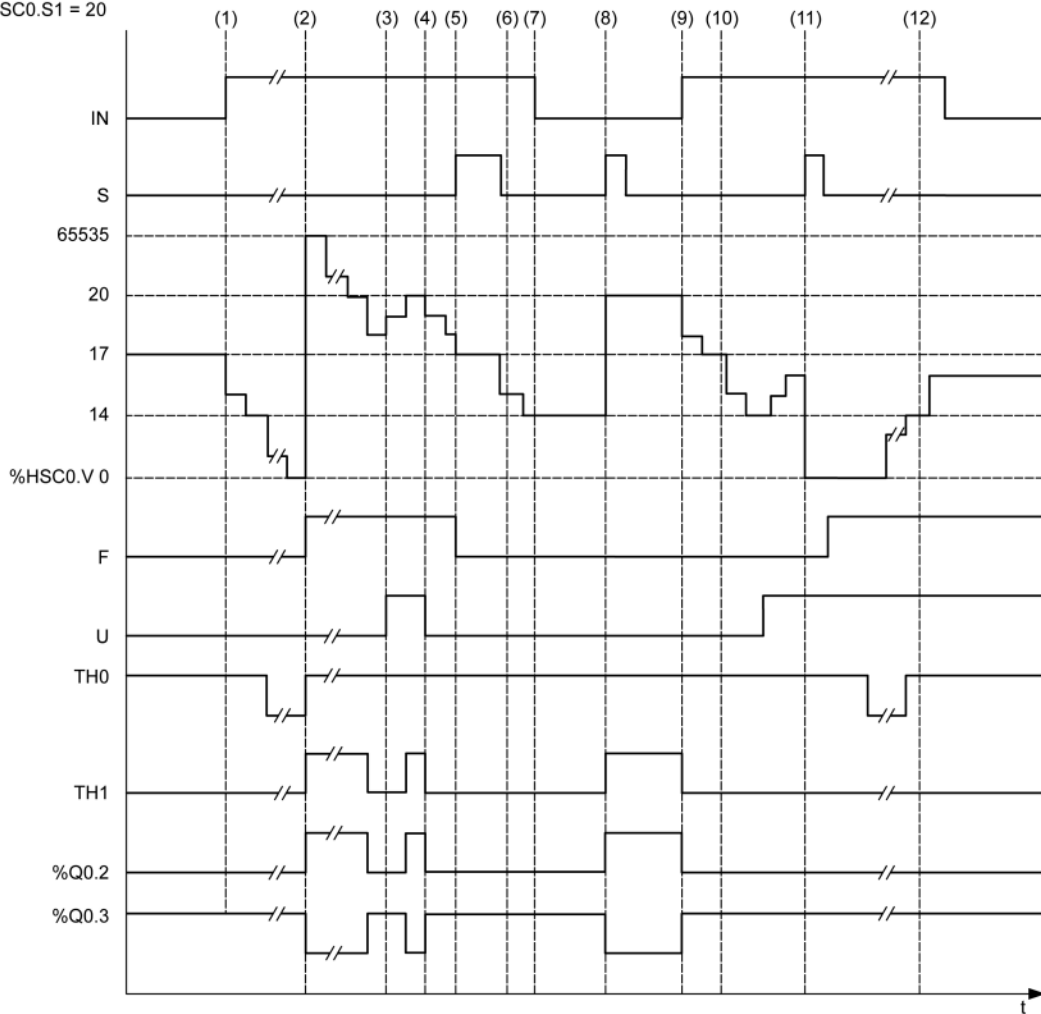
Reflexausgang	Wert < %HSC0.S0	%HSC0.S0 <= Wert < %HSC0.S1	Wert >= %HSC0.S1
%Q0.2	0	0	1
%Q0.3	1	1	0

## Zeitiagramm:

%HSC0.P = 17

%HSC0.S0 = 14

%HSC0.S1 = 20



- (1) Der Eingang IN wird auf 1 gesetzt. Daraufhin startet der Abwärtszählmodus ( $\%HSC0.U = 0$ , das heißt  $IB = 1$ )
- (2) Der aktuelle Wert erreicht 0, sodass der Flag des F-Ausgangs auf 1 gesetzt wird und  $\%HSC0.V$  bei der nächsten Zählung auf 65535 gesetzt wird.
- (3) Änderung am IB-Eingang, der Zähler ist nun im Aufwärtszählmodus und  $\%HSC0.U = 1$
- (4) Der IB-Eingang wird auf 1 gesetzt, d. h. der Zähler befindet sich jetzt im Abwärtszählmodus und  $\%HSC0.U$  wird auf 0 gesetzt
- (5) Der Eingang S wird auf 1 gesetzt, während der Abwärtszählvorgang läuft, d. h.  $\%HSC0.V$  wird auf den Preset-Wert  $\%HSC0.P = 17$  initialisiert
- (6) S wird auf 0 zurückgesetzt und der Preset-Wert  $\%HSC0.P$  wird in 20 geändert
- (7) Der Eingang IN wird auf 0 gesetzt, sodass die Funktion gesperrt wird,  $\%HSC0.V$  wird gehalten
- (8) S wird auf 1 gesetzt, sodass der neue Preset-Wert ( $\%HSC0.P = 20$ ) berücksichtigt wird und die Reflex-Ausgänge aktualisiert werden. **Hinweis:** Wenn ein Hilfs-Preset-Eingang anstelle von S verwendet wird, werden die Reflexausgänge nicht in Übereinstimmung mit der Twido-Familie an Steuerungen aktualisiert.
- (9) Der IN-Eingang wird auf 1 gesetzt und die Funktion startet erneut im Abwärtszählmodus
- (10) Der Schwellenwert  $\%HSC0.S1$  wird auf 17 gesetzt
- (11) Wenn der Eingang S aktiv ist, wird der neue Schwellenwert S1 beim nächsten Zählen zugelassen und  $\%HSC0.V$  wird auf 0 zurückgesetzt
- (12) Eine Erfassung des aktuellen Werts  $\%HSC0.V$  wird hierdurch  $\%HSC0.C = 14$

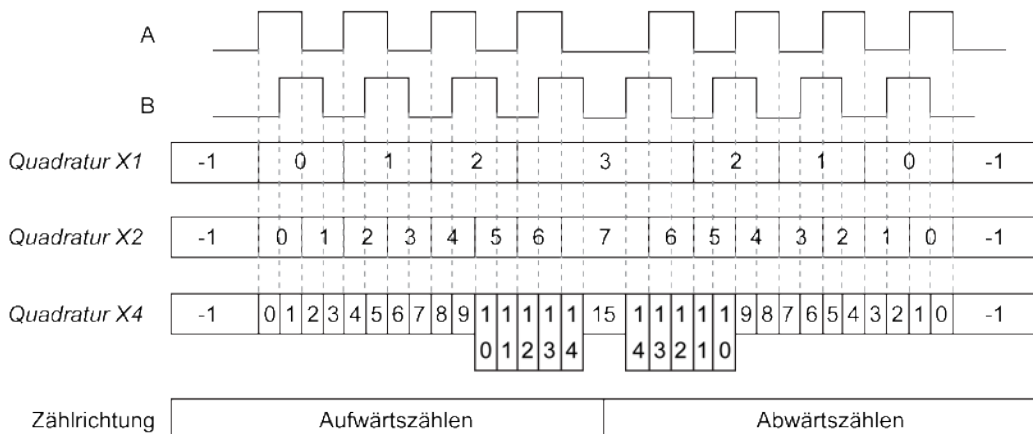
**HINWEIS:**  $\%HSC0.R$  und  $\%HSC0.S$  müssen auf *TRUE* eingestellt werden, damit die konfigurierten Reflexausgänge aktiv sind.

## Zeitdiagramm für Zweiphasig [Quadratur X1], Zweiphasig [Quadratur X2], Zweiphasig [Quadratur X4]

Ein physischer Geber stellt zwei um 90° verschobene Signale bereit, die dem Zähler das Zählen der Impulse und die Erkennung der Zählrichtung ermöglichen:

X1	1 Zählung pro Geberzyklus
X2	2 Zählungen pro Geberzyklus
X4	4 Zählungen pro Geberzyklus

Zeitdiagramm:



**Quadratur X1** Wenn Kanal A Kanal B anführt, dann wird der Zähler bei steigender Flanke an Kanal A inkrementiert. Wenn hingegen Kanal B Kanal A vorangeht, dann wird der Zähler bei fallender Flanke an Kanal A dekrementiert.

**Quadratur X2** Der Zähler wird bei jeder Flanke an Kanal A inkrementiert oder dekrementiert, je nachdem, welcher Kanal den anderen anführt. In jedem Zyklus werden zwei Inkrementierungen bzw. Dekrementierungen durchgeführt.

**Quadratur X4** Der Zähler wird bei jeder Flanke an den Kanälen A und B inkrementiert oder dekrementiert, je nachdem, welcher Kanal den anderen anführt. In jedem Zyklus werden vier Inkrementierungen bzw. Dekrementierungen durchgeführt.

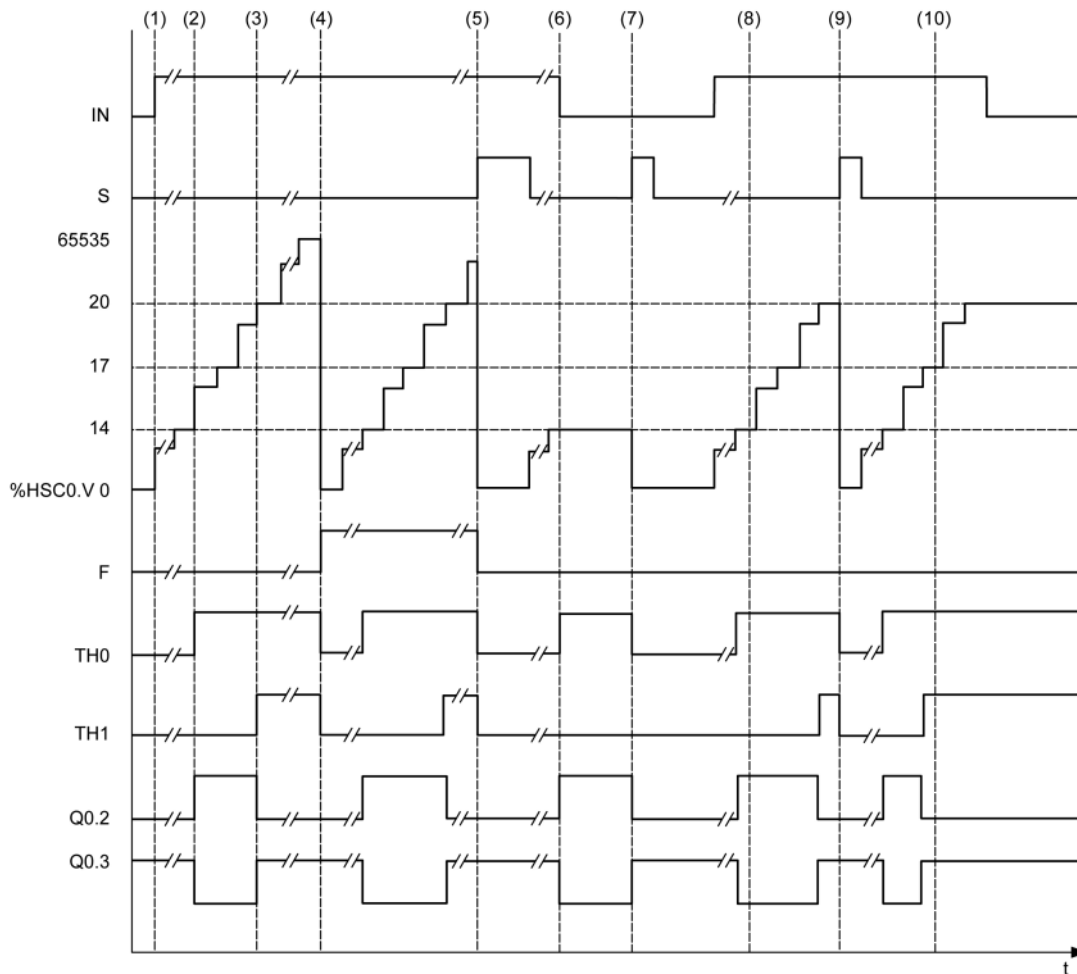
## Zeitdiagramm für Einphasig

Beispiel für Reflexausgangskonfiguration:

Reflexausgang	Wert < %HSC0.S0	%HSC0.S0 <= Wert < %HSC0.S1	Wert >= %HSC0.S1
%Q0.2	0	1	0
%Q0.3	1	0	1

Zeitdiagramm:

%HSC0.P = 17  
 %HSC0.S0 = 14  
 %HSC0.S1 = 20



- (1) *IN* wird auf 1 gesetzt: Die Zählfunktion wird aktiviert ( $\%HSC0.U = 1$  da  $\%HSC0$  ein Aufwärtszähler ist)
- (2)  $\%Q0.2$ (Reflexausgang) und *TH0* werden auf 1 gesetzt
- (3) *TH1* ist auf 1 eingestellt
- (4) Der Maximalwert wird erreicht, sodass bei der nächsten Zählung  $\%HSC0.V$  auf 0 zurückgesetzt wird und *F* auf 1 gesetzt wird.
- (5) *S* wird auf 1 gesetzt, der aktuelle Wert,  $\%HSC0.V$ , wird auf 0 gesetzt
- (6) Die aktuelle Funktion wird gesperrt, solange *IN* auf 0 gesetzt ist.
- (7) Während die Funktion gesperrt ist, wird *S* auf 1 gesetzt, sodass der aktuelle Wert auf 0 zurückgesetzt wird.
- (8) Änderung des Schwellenwerts *S1* auf 17
- (9) *S* wird auf 1 gesetzt, sodass der neue Wert von *S1* bei der nächsten Zählung vergeben wird
- (10) Der Erfassungseingang wird auf 1 gesetzt, somit gilt  $\%HSC0.C = 17$

# Hochgeschwindigkeitszähler im Frequenzmesser-Modus

## Einführung

Der Frequenzmesser-Modus eines Hochgeschwindigkeitszählers (*High Speed Counter*) dient der Messung der Frequenz eines periodischen Signals in Hz am Eingang IA (Impulseingang Phase A).

Der messbare Frequenzbereich reicht von 1 Hz bis 100 kHz mit einem Bereich von 0 bis 4294967295 im Doppelwortmodus.

Es kann zwischen 2 Zeitbasen gewählt werden. Die Wahl erfolgt anhand des Objekts `%HSC.T` (Zeitbasis):

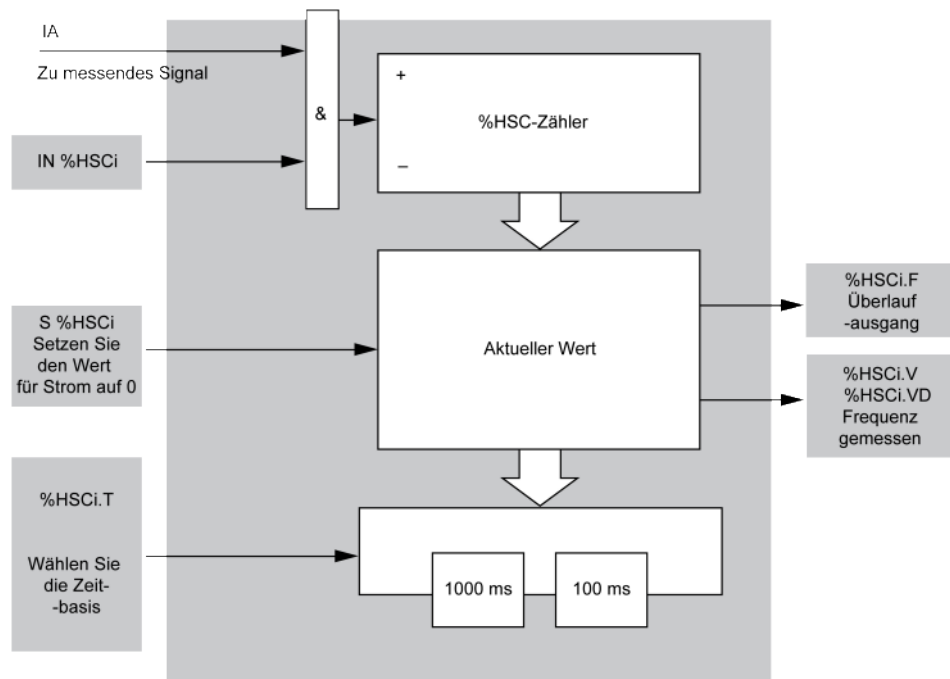
Zeitbasis	Genauigkeit	Aktualisieren
100 ms	0,01% für 100 kHz 10% für 100 Hz	10 Mal pro Sekunde
1 s	0,001% für 100 kHz 10% für 10 Hz	1 Mal pro Sekunde

## Genauigkeitsmessung

$$\text{Genauigkeit}(\%) = \frac{1}{f[\text{Hz}]} \times \frac{1}{TB[\text{s}]} \times 100$$

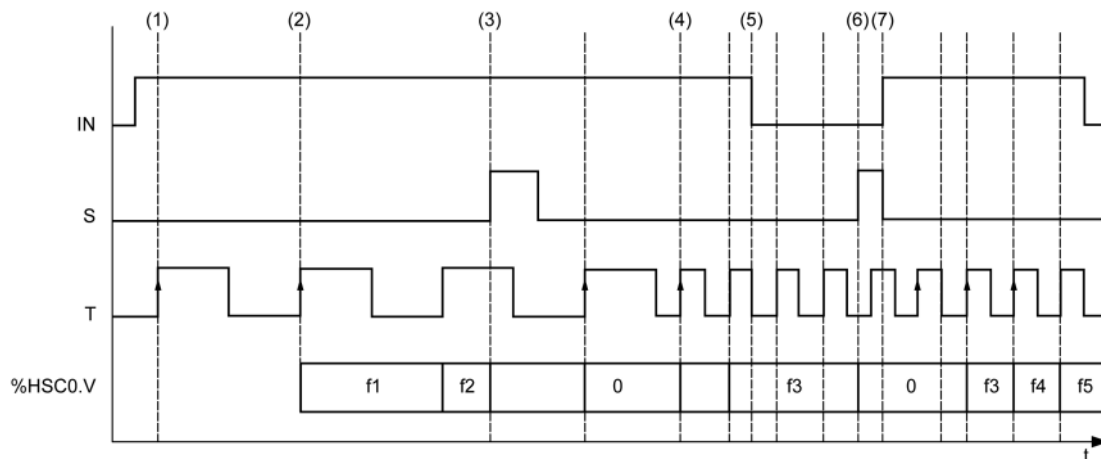
## Operation

Diese Abbildung zeigt das Betriebsdiagramm des Frequenzmesser-Modus:



## Zeitdiagramm

Dieses Zeitdiagramm ist ein Beispiel für die Verwendung einer *High Speed Counter*-Funktion im Frequenzmesser-Modus:



- (1) Die erste Frequenzmessung startet mit einer steigenden Flanke des *TB*-Signals
- (2) *%HSC0.V* (oder *%HSC0.VD*) wird nach einer Periode des *TB* aktualisiert
- (3) Eingang *IN* und Eingang *S* werden auf 1 gesetzt, während *%HSC0.V* (oder *%HSC0.VD*) auf 0 gesetzt wird
- (4) *%HSC0.T* wird auf 100 ms gesetzt, folglich wird die aktuelle Messung abgebrochen und eine neue Messung gestartet
- (5) Eingang *IN* wird auf 0 gesetzt, sodass die Frequenzmessfunktion gesperrt und *%HSC0.V* (oder *%HSC0.VD*) gehalten wird
- (6) *S* wird auf 1 gesetzt, folglich wird der Wert *%HSC0.V* (oder *%HSC0.VD*) auf 0 gesetzt
- (7) *S* wird auf 0 und *IN* auf 1 gesetzt, sodass die Messung mit der nächsten steigenden Flanke des *TB*-Signals startet.

---

# Erweiterte Ausgangsfunktionen für Experten

## Inhalt dieses Abschnitts

Impuls (%PLS).....	41
Impulsbreitenmodulation (%PWM) .....	48
Antrieb (%DRV) .....	54
Impulswellenausgang (%PTO) .....	80
Frequenzgenerator (%FREQGEN) .....	157

## Überblick

In diesem Abschnitt werden die erweiterten Ausgangsfunktionen für Experten beschrieben.

# Impuls (%PLS)

## Inhalt dieses Kapitels

Beschreibung.....41  
 Funktionsblock-Konfiguration .....42  
 Programmierbeispiel .....47

## Verwendung der Pulse-Funktionsbausteine

Dieser Abschnitt enthält Beschreibungen und Programmierrichtlinien für die Verwendung von *Pulse*-Funktionsbausteinen.

### Beschreibung

### Einführung

Der *Pulse*-Funktionsbaustein  dient der Erzeugung von Rechtecksignalen.

Zwei *Pulse*-Funktionsbausteine sind am dedizierten Ausgangskanal %Q0.0 oder %Q0.1 verfügbar. Logische Steuerungen mit Relaisausgängen für diese beiden Kanäle werden vom *Pulse*-Funktionsbaustein nicht unterstützt. Weitere Informationen zu Ein- und Ausgängen finden Sie im M221 Logic Controller - Hardwarehandbuch.

Der *Pulse*-Funktionsbaustein erlaubt nur eine einzelne Signalbreite bzw. einen Arbeitszyklus von 50 %.

Sie können die Anzahl der Impulse oder den Zeitraum, während dem die Impulsfolge ausgeführt wird, begrenzen. Diese Einstellungen können bei der Konfiguration vorgenommen und/oder vom Programm aktualisiert werden.

Sie müssen den *Pulse*-Funktionsbaustein unter **Konfiguration > Impulsgeneratoren** konfigurieren, bevor Sie eine Instanz des Funktionsbausteins verwenden – siehe Konfigurieren von Impulsgeneratoren (siehe Modicon M221, Logic Controller, Programmierhandbuch).

Die PLS-Funktion hat die folgenden Leistungsmerkmale:

Kenndaten	Wert
Anzahl der Kanäle	2
Minimale Frequenz	1 Hz
Maximale Frequenz	10000 Hz
Genauigkeit der Frequenz	1 %

## Abbildung

Die folgende Abbildung zeigt einen Funktionsbaustein *Pulse*:



## Eingänge

Der *Pulse*-Funktionsbaustein verfügt über folgende Eingänge:

Bezeichnung	Beschreibung	Wert
IN	Aktivieren	Im Zustand 1 findet die Impulserzeugung am dedizierten Ausgangskanal statt. Im Zustand 0 wird der Ausgangskanal auf 0 gesetzt.
L	Rücksetzen auf 0 (optional)	Im Zustand 1 werden die Ausgänge %PLSi.Q und %PLSi.D auf 0 gesetzt. Die Anzahl der im Zeitraum T erzeugten Impulse wird auf 0 gesetzt.

## Ausgänge

Der Funktionsbaustein *Pulse* verfügt über folgende Ausgänge:

Bezeichnung	Objekt	Beschreibung	Wert
Q	%PLSi.Q	Generierung läuft	Der Zustand 1 zeigt an, dass das <i>Impulssignal</i> am zweckbestimmten Ausgangskanal erzeugt wird.
D	%PLSi.D	Generierung abgeschlossen (optional)	Im Zustand 1 ist die Signalerzeugung beendet. Die Anzahl der gewünschten Impulse wurde erzeugt.

## Funktionsblock-Konfiguration

### Überblick

Siehe Konfigurieren von Impulsgeneratoren (siehe Modicon M221, Logic Controller, Programmierhandbuch) zur Konfiguration der Ressource des *Impulsgenerators*.

Siehe Konfigurieren des Impulses (siehe Modicon M221, Logic Controller, Programmierhandbuch) zur Konfiguration der Ressource des *Impulsgenerators* als eine PLS.

## Parameter

Der *Pulse*-Funktionsbaustein verfügt über folgende Parameter:

Parameter	Beschreibung	Wert
<b>Verwendet</b>	Adresse verwendet	Wenn diese Option ausgewählt ist, wird die Adresse derzeit in einem Programm verwendet.
<b>Adresse</b>	%PLSi Pulse-Adresse	Die Instanz-ID, wobei i einer Zahl zwischen 0 und der Gesamtanzahl der mit der Steuerung verfügbaren Objekte entspricht. Die maximale Anzahl an <i>Pulse</i> -Objekten finden Sie in der Tabelle Maximale Anzahl Objekte (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<b>Symbol</b>	Symbol	Das diesem Objekt zugeordnete Symbol. Weitere Informationen finden Sie unter Definieren und Verwenden von Symbolen (siehe EcoStruxure Machine Expert - Basic, Betriebshandbuch).
<b>Preset</b>	Voreinstellung der Periode (%PLSi.P)	<ul style="list-style-type: none"> <li>Zeitbasis = 1 s, %PLSi.P = 1 oder 2</li> <li>Zeitbasis = 10 ms, 1 &lt;= %PLSi.P &lt;= 200</li> <li>Zeitbasis = 1 ms, 1 &lt;= %PLSi.P &lt;= 2000</li> <li>Zeitbasis = 0,1 ms, 1 &lt;= %PLSi.P &lt;= 20000</li> </ul>
<b>Anz. Impuls</b>	Anzahl Impulse (%PLSi.N, %PLSi.ND)	Um eine unbegrenzte Anzahl von Impulsen zu erzeugen, setzen Sie %PLS.N oder %PLS.ND auf 0.
<b>Strom</b>	Stromausgang (%PLSi.Q)	0 oder 1.
<b>Fertig</b>	Signalerzeugung beendet (%PLSi.D)	Im Zustand 1 ist die Signalerzeugung beendet. Die Anzahl der gewünschten Impulse wurde erzeugt. Wird zurückgesetzt, wenn einer der Eingänge IN oder R auf 1 gesetzt wird.
<b>Kommentar</b>	Kommentar	Diesem Objekt kann ein optionaler Kommentar zugeordnet werden.  Doppelklicken Sie in die Spalte <b>Kommentar</b> und geben Sie einen Kommentar ein.

## Objekte

Der *Pulse*-Funktionsbaustein ist mit folgenden Objekten verknüpft:

Objekt	Beschreibung	Größe (Bit)	Standardwert	Bereich	
%PLSi.P	Preset-Wert	16	Preset (unter Konfiguration > Impulsgeneratoren)	Preset %PLSi.P	Zeitbasis
				1...20000	0,1 ms
				1...2000	1 ms
				1...200	10 ms
				1 oder 2	1 s (Standard)
%PLSi.N	Anzahl Impulse	16	0	0...32767	
%PLSi.ND		32	0	0...2147483647	

## Verwendungsregeln

Die Ausgangssignalperiode *T* wird mit **Preset** und den **Zeitbasis**-Parametern wie beispielsweise  $T = \%PLSi.P \times \text{Zeitbasis}$  eingestellt.

Die nachstehende Tabelle enthält den Bereich verfügbarer Perioden:

Zeitbasis	Frequenz
0,1 ms	0,5 Hz...10000 Hz
1 ms	0,5 Hz...1000 Hz
10 ms	0,5 Hz...100 Hz
1 s	0,5 Hz...1 Hz

Die **Zeitbasis** wird in **Konfiguration > Impulsgeneratoren** eingestellt und kann nicht geändert werden. Detaillierte Informationen hierzu finden Sie in Konfigurieren von Impulsgeneratoren (siehe Modicon M221, Logic Controller, Programmierhandbuch).

Wenn *%PLSi.P*:

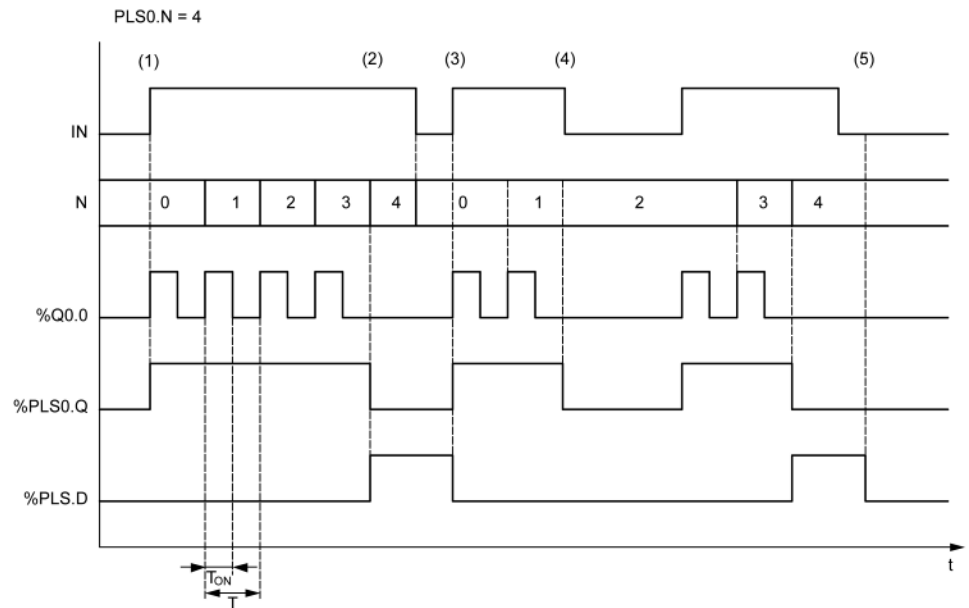
- geändert wird, wird die Ausgangssignalperiode am Ende der aktuellen Periode geändert.
- auf 0 gesetzt wird, wird die Funktion zur Impulsgenerierung gestoppt.
- außerhalb des Bereichs liegt, wird der Parameter auf 0 forciert und die Funktion zur Impulsgenerierung wird gestoppt.

Wenn *%PLSi.N* (bzw. *%PLSi.ND* im **Doppelwort**-Modus):

- geändert wird, wird die Anzahl zu generierender Impulse bei der nächsten Ausführung der Funktion zur Impulsgenerierung verwendet (*%PLSi.D* = 1 oder nach *%PLSi.R* = 1).
- auf 0 gesetzt wird, wird eine unbegrenzte Anzahl Impulse generiert.
- außerhalb des Bereichs liegt, wird der Parameter auf 0 forciert.

## Zeitdiagramm

Dieses Diagramm veranschaulicht den Zeitablauf für den Funktionsbaustein *Pulse*:



- (1)** Der *IN*-Eingang wird auf 1 gesetzt, das Impulssignal wird am zweckbestimmten Ausgang (*%Q0.0*) erzeugt, sodass *%PLSi.Q* auf 1 gesetzt wird.
- (2)** Die Anzahl der Impulse erreicht *%PLS0.N* (=4), der Ausgang (*%PLS0.D*) mit dem Flag „Done“ (Fertig) wird auf 1 gesetzt und die Impulsgenerierung wird angehalten (*%PLS0.Q* = 0).
- (3)** Der Eingang *IN* wird auf 1 gesetzt und *%PLS0.D* auf 0 zurückgesetzt.
- (4)** Der Eingang *IN* wird auf 0 gesetzt, der Ausgangskanal wird auf 0 gesetzt und *%PLS0.Q* = 0 gibt an, dass die Signalerzeugung nicht aktiv ist.
- (5)** *%PLS0.D* wird auf 0 gesetzt, indem der Eingang *R* auf 1 gesetzt wird.

## Sonderfälle

Sonderfall	Beschreibung
Auswirkung eines Kaltstarts (%S0=TRUE)	<ul style="list-style-type: none"> <li>• Impulsgenerierung wird gestoppt.</li> <li>• Während der Steuerungsinitialisierung wird der Ausgang auf 0 zurückgesetzt.</li> <li>• Verhaltensweisen nach der Steuerungsinitialisierung:               <ul style="list-style-type: none"> <li>◦ Wenn die Steuerung in den Status <i>STOPPED</i> (GESTOPPT) wechselt, wird die Fehlerausweichstrategie auf den Ausgang angewendet.</li> <li>◦ Wenn die Steuerung in den Status <i>RUNNING</i> (IN BETRIEB) wechselt, werden die Konfigurationsparameter wiederhergestellt.</li> </ul> </li> </ul>
Auswirkung eines Warmstarts (%S1=TRUE)	<ul style="list-style-type: none"> <li>• Impulsgenerierung wird gestoppt.</li> <li>• Während der Steuerungsinitialisierung wird der Ausgang auf 0 zurückgesetzt.</li> <li>• Verhaltensweisen nach der Steuerungsinitialisierung:               <ul style="list-style-type: none"> <li>◦ Wenn die Steuerung in den Status <i>STOPPED</i> (GESTOPPT) wechselt, wird die Fehlerausweichstrategie auf den Ausgang angewendet.</li> <li>◦ Wenn die Steuerung in den Status <i>RUNNING</i> (LÄUFT) wechselt, werden die Konfigurationsparameter wiederhergestellt. Die Anzahl der Impulse, die evtl. bereits gesendet wurden, wird jedoch auf 0 zurückgesetzt.<sup>(1)</sup></li> </ul> </li> </ul>
Auswirkung bei Steuerungsstopp	<ul style="list-style-type: none"> <li>• Impulsgenerierung wird gestoppt.</li> <li>• Das Fehlerausweichverhalten ist von der konfigurierten Fehlerausweichstrategie abhängig:               <ul style="list-style-type: none"> <li>◦ <b>Wert beibehalten:</b> Die Ausgänge werden auf 0 zurückgesetzt.</li> <li>◦ <b>Fehlerausweichwert:</b> Die Ausgänge werden auf die konfigurierte Fehlerausweichwerte (siehe EcoStruxure Machine Expert - Basic, Betriebshandbuch).</li> </ul> </li> </ul>
Auswirkung einer Online-Änderung	Keine
Auswirkung eines Kurzschlusses oder von Überspannung auf einen Ausgang, der von einem <i>Pulse</i> -Funktionsbaustein adressiert wird, während eine begrenzte Anzahl an Impulsen generiert wird	<ul style="list-style-type: none"> <li>• Impulsgenerierung wird gestoppt.</li> <li>• Wenn der Kurzschluss oder die Überspannung behoben ist, fährt die Impulsgenerierung bei der letzten Sequenz vor dem Stopp fort.</li> </ul>
<p><sup>(1)</sup> Wenn zum Zeitpunkt des Warmstarts eine Impulsausgangsanweisung effektiv ist, berücksichtigt die Impulsgenerierung beim Neustart der Steuerung nicht die Anzahl der Impulse, die vor dem Warmstart gesendet wurden.</p>	

### **WARNUNG**

#### **UNBEABSICHTIGTER GERÄTEBETRIEB**

- Verwenden Sie keinen Warmstartbefehl (%S1=TRUE), während ein PLS-Befehl aktiv ist.
- Wenn ein Warmstart nicht zu vermeiden ist, müssen Sie alle Impulse berücksichtigen, die vor diesem Warmstart gesendet wurden.

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

# Programmierbeispiel

## Einführung

Der *Pulse*-Funktionsbaustein kann wie in diesem Programmierbeispiel gezeigt konfiguriert werden.

## Programmierung

Dieses Beispiel ist ein *Pulse*-Funktionsbaustein:

Programm- baustein	Anweisung
0	BLK %PLS0 LD %M1 IN LD %M0 R OUT_BLK LD Q ST %Q0.5 LD D ST %M10 END_BLK

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

# Impulsbreitenmodulation (%PWM)

## Inhalt dieses Kapitels


Beschreibung .....	48
Funktionsblock-Konfiguration .....	49
Programmierbeispiel .....	52

## Verwendung der Funktionsbausteine des Typs Impulsbreitenmodulation

Dieser Abschnitt enthält Beschreibungen und Programmierrichtlinien für die Verwendung von *Pulse Width Modulation*-Funktionsbausteinen.

### Beschreibung

#### Einführung

Der *Pulse Width Modulation*-Funktionsbaustein  erzeugt ein moduliertes Wellensignal am dedizierten Ausgangskanal, %Q0.0 oder %Q0.1, mit variabler Breite und folglich variablem Arbeitszyklus.

Steuerungen mit Relaisausgängen für diese beiden Kanäle unterstützen diese Funktion nicht.

%PWM0 verwendet den dedizierten Ausgang %Q0.0, und %PMW1 verwendet den dedizierten Ausgang %Q0.1. Die Pulse-Funktionsbausteine %PLS können auch zur Verwendung derselben dedizierten Ausgänge konfiguriert werden. Für einen bestimmten dedizierten Ausgang kann jeweils eine dieser zwei Funktionen konfiguriert werden, jedoch nicht beide.

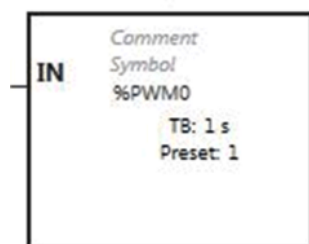
Sie müssen den *Pulse Width Modulation*-Funktionsbaustein unter **Konfiguration > Impulsgeneratoren** konfigurieren, bevor Sie eine Instanz des Funktionsbausteins verwenden. Siehe Konfigurieren von Impulsgeneratoren (siehe Modicon M221, Logic Controller, Programmierhandbuch).

Die PWM-Funktion hat die folgenden Leistungsmerkmale:

Merkmal	Wert
Anzahl der Kanäle	2
Minimale Frequenz	1 Hz
Maximale Frequenz	10000 Hz
Genauigkeit der Frequenz	1 %

## Abbildung

Diese Abbildung zeigt den *Pulse Width Modulation*-Funktionsbaustein.



## Eingänge

Der *Pulse Width Modulation*-Funktionsbaustein verfügt über folgenden Eingang:

Bezeichnung	Objekt	Beschreibung	Wert
IN	%PWMi.IN	Aktivieren	Im Zustand 1 wird das <i>Pulse Width Modulation</i> -Signal am Ausgangskanal erzeugt. Im Zustand 0 wird der Ausgangskanal auf 0 gesetzt.

## Funktionsblock-Konfiguration

### Überblick

Siehe Konfigurieren von Impulsgeneratoren (siehe Modicon M221, Logic Controller, Programmierhandbuch) zur Konfiguration der Ressource des *Impulsgenerators*.

Siehe Konfigurieren der Impulsbreitenmodulation (siehe Modicon M221, Logic Controller, Programmierhandbuch) zur Konfiguration der Ressource des *Impulsgenerators* als eine PWM.

## Eigenschaften

Der *Pulse Width Modulation*-Funktionsbaustein hat die folgenden Eigenschaften:

Eigenschaft	Wert	Beschreibung
<b>Verwendet</b>	Aktiviertes/deaktiviertes Kontrollkästchen	Gibt an, ob die Adresse verwendet wird.
<b>Adresse</b>	%PWMi, wobei i 0 oder 1 ist.	i ist der Bezeichner für die Instanz. Die maximale Anzahl an PWM-Objekten finden Sie in der Tabelle Maximale Anzahl Objekte (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<b>Symbol</b>	Benutzerdefinierter Text	Das diesem Objekt zugeordnete Symbol. Weitere Informationen finden Sie unter Definieren und Verwenden von Symbolen (siehe EcoStruxure Machine Expert - Basic, Betriebshandbuch).
<b>Preset</b>	<ul style="list-style-type: none"> <li>• %PWMi.P=1, wenn <b>Zeitbasis</b>=1 s</li> <li>• 1&lt;=%PWMi.P&lt;=100, wenn <b>Zeitbasis</b>=10 ms</li> </ul>	Voreinstellung der Periode

Eigenschaft	Wert	Beschreibung
	<ul style="list-style-type: none"> <li>• <math>1 \leq \%PWMi.P \leq 1000</math>, wenn <b>Zeitbasis</b>=1 ms</li> <li>• <math>1 \leq \%PWMi.P \leq 10000</math>, wenn <b>Zeitbasis</b>=0,1 ms</li> </ul>	
<b>Arbeitszyklus</b>	Von 0 bis 100 <b>HINWEIS:</b> Werte über 100 werden als gleich 100 betrachtet.	Der <b>Arbeitszyklus</b> wird durch das Objekt <b>%PWMi.R</b> gesteuert und ist der Prozentsatz des Signals in Zustand 1 innerhalb der Periode. Die Breite von Zustand 1 (Tp) ist deshalb gleich:  $TP = T \times (\%PWMi.R/100)$ . Die Anwendung schreibt den Wert für <b>%PWMi.R</b> .
<b>Kommentar</b>	Benutzerdefinierter Text	Ein mit diesem Objekt zu verknüpfender Kommentar.

**HINWEIS:** Die Eigenschaften **Anz.Impuls**, **Aktuell** und **Fertig**, die in der Tabelle der **Eigenschaften von Impulsgeneratoren** auf der Registerkarte **Programmierung** aufgeführt werden, sind für die PWM-Funktion nicht relevant.

## Objekte

Der *Pulse Width Modulation*-Funktionsbaustein ist mit folgenden Objekten verknüpft:

Objekt	Beschreibung	Größe (Bit)	Standardwert	Bereich	
				Preset %PWMi.P	Zeitbasis
%PWMi.P	Preset-Wert	16	Preset (unter <b>Konfiguration &gt; Impulsgeneratoren</b> )	1 bis 10000	0,1 ms
				1 bis 1000	1 ms
				1 bis 100	10 ms
				1	1 s (Standard)
%PWMi.R	Arbeitszyklus (Ratio)	16	0	0...100	

Wenn **%PWMi.P**:

- geändert wird, ist die Ausgangssignalperiode am Ende der aktuellen Periode betroffen.
- auf 0 gesetzt wird, wird die Funktion zur Impulsgenerierung gestoppt.
- außerhalb des Bereichs liegt, wird der Parameter auf 0 forciert und die Funktion zur Impulsgenerierung wird gestoppt.

Wenn **%PWMi.R**:

- auf 0 gesetzt wird, wird die Funktion zur Impulsgenerierung gestoppt (Ausgang wird auf 0 gesetzt).
- auf 100 gesetzt wird, wird das Ausgangssignal auf 1 gesetzt.
- geändert wird, wird die Ausgangssignalverhältnis am Ende der aktuellen Periode geändert.
- außerhalb des Bereichs liegt, wird der Parameter auf 0 forciert.

## Zeitbasis

Die **Zeitbasis** wird im Menü **Konfiguration > Impulsgeneratoren** festgelegt und kann nur auf der Registerkarte **Konfiguration** geändert werden. Detaillierte

Informationen hierzu finden Sie in Konfigurieren von Impulsgeneratoren (siehe Modicon M221, Logic Controller, Programmierhandbuch).

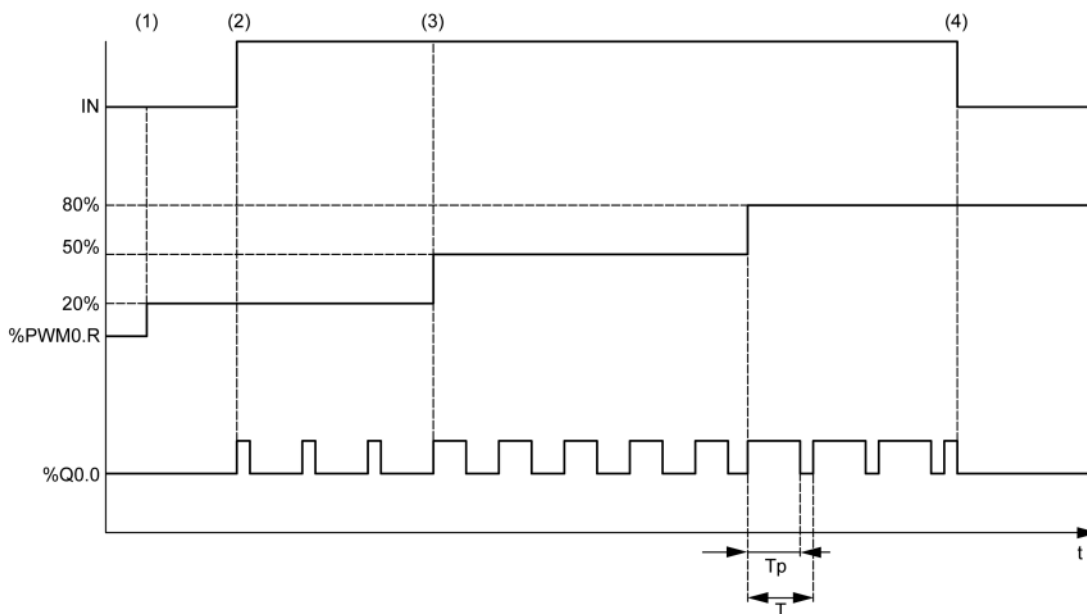
Die Ausgangssignalperiode  $T$  wird mit **Preset** und den **Zeitbasis**-Parametern so eingestellt, dass gilt:  $T = \%PWMi.P \times \text{Zeitbasis}$ .

Die nachstehende Tabelle zeigt den Bereich verfügbarer Perioden:

Zeitbasis	Frequenzbereich
0,1 ms	1 Hz...10000 Hz
1 ms	1 Hz...1000 Hz
10 ms	1 Hz...100 Hz
1 s	1 Hz...1 Hz

## Zeitdiagramm

Dieses Diagramm veranschaulicht den Zeitablauf für den Funktionsbaustein *Pulse Width Modulation*:



- (1) Das PWM-Verhältnis ( $\%PWMi.R$ ) wird auf 20 % gesetzt,  $IN = 0$ , d. h. die Impulsgenerierung ist nicht aktiv
- (2)  $IN$  wird auf 1 gesetzt, so dass der PWM-Ausgang aktiviert wird
- (3) Die programmierbare Breite ( $T_p$ ) ändert sich mit  $\%PWM.R$
- (4)  $IN$  wird auf 0 gesetzt, so dass die PWM-Funktion gesperrt ist

## Sonderfälle

Sonderfall	Beschreibung
Auswirkung eines Kaltstarts (%S0=TRUE)	<ul style="list-style-type: none"> <li>Impulsgenerierung wird gestoppt.</li> <li>Während der Steuerungsinitialisierung wird der Ausgang auf 0 zurückgesetzt.</li> <li>Verhaltensweisen nach der Steuerungsinitialisierung: <ul style="list-style-type: none"> <li>Wenn die Steuerung in den Status <i>STOPPED</i> (GESTOPPT) wechselt, wird die Fehlerausweichstrategie auf den Ausgang angewendet.</li> <li>Wenn die Steuerung in den Status <i>RUNNING</i> (IN BETRIEB) wechselt, werden die Konfigurationsparameter wiederhergestellt.</li> </ul> </li> </ul>
Auswirkung eines Warmstarts (%S1=TRUE)	<ul style="list-style-type: none"> <li>Impulsgenerierung wird gestoppt.</li> <li>Während der Steuerungsinitialisierung wird der Ausgang auf 0 zurückgesetzt.</li> <li>Wenn die Steuerung nach der Initialisierung in den Status <i>STOPPED</i> (GESTOPPT) wechselt, wird die Fehlerausweichstrategie auf den Ausgang angewendet.</li> </ul>
Auswirkung bei Steuerungsstopp	<ul style="list-style-type: none"> <li>Impulsgenerierung wird gestoppt.</li> <li>Das Fehlerausweichverhalten ist von der konfigurierten Fehlerausweichstrategie abhängig: <ul style="list-style-type: none"> <li><b>Wert beibehalten:</b> Die Ausgänge werden auf 0 zurückgesetzt.</li> <li><b>Fehlerausweichwert:</b> Die Ausgänge werden auf die konfigurierte Fehlerausweichwerte (siehe EcoStruxure Machine Expert - Basic, Betriebshandbuch).</li> </ul> </li> </ul>
Auswirkung einer Online-Änderung	Keine

## Programmierbeispiel

### Einführung

Der Funktionsbaustein des Typs *Pulse Width Modulation* kann wie in diesem Programmierbeispiel konfiguriert werden.

### Programmierbeispiel

In diesem Beispiel:

- Die Signalbreite wird vom Programm in Übereinstimmung mit dem Status der Steuerungseingänge *%I0.0* und *%I0.1* konfiguriert.
- Die Zeitbasis wird auf 10 ms gesetzt.
- Der Preset-Wert *%PWM0.P* wird auf 50 gesetzt, sodass das Schrittverhältnis 2 % entspricht.
- Die konfigurierbare Periode T ist gleich 500 ms.

Das Ergebnis ist:

- Wenn *%I0.0* und *%I0.1* auf 0 gesetzt sind, wird das *%PWM0.R*-Verhältnis auf 20 % gesetzt und die Länge des Signals im Zustand 1 ist dann: 20 % x 500 ms = 100 ms.
- Werden *%I0.0* auf 1 und *%I0.1* auf 0 gesetzt, dann wird das *%PWM0.R*-Verhältnis auf 50 % gesetzt (Dauer 250 ms).
- Werden *%I0.0* und *%I0.1* auf 1 gesetzt, dann wird das *%PWM0.R*-Verhältnis auf 80 % gesetzt (Dauer 400 ms).

Beispiele für *Pulse Width Modulation*-Anweisungen:

Pro-gramm-baustein	Anweisung
0	LDN %I0.0 ANDN %I0.1 [%PWM0.R:=20]
1	LD %I0.0 ANDN %I0.1 [%PWM0.R:=50]
2	LD %I0.0 AND %I0.1 [%PWM0.R:=80]
3	BLK %PWM0 LD %I0.2 IN END_BLK

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

# Antrieb (%DRV)

## Inhalt dieses Kapitels

Beschreibung.....	54
Zustand der Antriebe und Logiksteuerungen .....	56
Hinzufügen eines Antriebsfunktionsbausteins .....	58
Funktionsblock-Konfiguration .....	59
MC_Power_ATV: Endstufe aktivieren/deaktivieren.....	59
MC_Jog_ATV: Tippbetrieb starten .....	62
MC_MoveVel_ATV: Bewegen mit einer bestimmten Geschwindigkeit .....	64
MC_Stop_ATV: Bewegung anhalten .....	67
MC_ReadStatus_ATV: Gerätestatus lesen .....	70
MC_ReadMotionState_ATV: Bewegungsstatus lesen .....	72
MC_Reset_ATV: Quittieren und Zurücksetzen eines Fehlers .....	74
Fehlercodes.....	77

## Beschreibung

### Beschreibung

Durch Antriebsfunktionsbausteine  können Antriebsgeräte wie Altivar Frequenzumrichter durch einen M221 Logic Controller gesteuert werden. Beispiel:

- Steuern der Geschwindigkeit eines durch einen ATV-Antrieb verwalteten Motors und kontinuierliches Aktualisieren
- Überwachung des Zustands des ATV-Antriebs und Motors
- Verwalten von im ATV-Antrieb festgestellten Fehlern.

Kommunikationen finden über eine der folgenden Methoden statt.

- Konfiguration einer der seriellen Leitungen der Logiksteuerung als ein Modbus Serial-E/A-Scanner (siehe Modicon M221, Logic Controller, Programmierhandbuch) über das Modbus RTU-Protokoll.
- Konfiguration des Ethernet-Ports als ein Modbus TCP IOScanner.

Fügen Sie in EcoStruxure Machine Expert - Basic zunächst die erwünschten ATV-Antriebsarten zum Modbus Serial E/A-Scanner oder Modbus TCP IOScanner hinzu. Dadurch werden vordefinierte Kanäle und Initialisierungsanfragen eingerichtet, die das Lesen und Schreiben von Daten auf bestimmten Registern auf dem ATV-Antrieb ermöglichen, wie zum Beispiel:

- **ETA**-Statuswort
- Erweitertes **ETI**-Statuswort
- **RFRD** Aktuelle Geschwindigkeit (RPM)
- **DP0** Fehlercode des letzten Fehlers
- **CMD**-Steuerwort

Die Datenübertragung wird unter Verwendung des Modbus-Anfragetyps **FC23 - Lesen/Schreiben mehrerer Register** ausgeführt. Dadurch kann das Programm beispielsweise mit einer einzelnen Modbus-Anfrage von **ETA**-, **ETI**- und **DP0**-Registern lesen und ins **CMD**-Register schreiben.

Die folgenden Antriebsfunktionsbausteine für Einzelachsen sind in der Registerkarte **Programmierung** von EcoStruxure Machine Expert - Basic verfügbar:

Funktionsbaustein	Beschreibung
MC_Power_ATV, Seite 59	Aktiviert oder deaktiviert die Endstufe eines Geräts.
MC_Jog_ATV, Seite 62	Startet den Jog-Betriebsmodus eines Geräts.
MC_MoveVel_ATV, Seite 64	Spezifiziert die Zielgeschwindigkeit eines Geräts.
MC_Stop_ATV, Seite 67	Stoppt die aktuelle Bewegung eines Geräts.
MC_ReadStatus_ATV, Seite 70	Gibt Zustandsinformationen eines Geräts wieder.
MC_ReadMotionState_ATV, Seite 72	Gibt Zustandsinformationen über die aktuelle Bewegung eines Geräts wieder.
MC_Reset_ATV, Seite 74	Setzt den Gerätefehler hinsichtlich des Antriebzustands, Seite 56 zurück und bestätigt die MC_Power_ATV, Seite 59-Fehler.

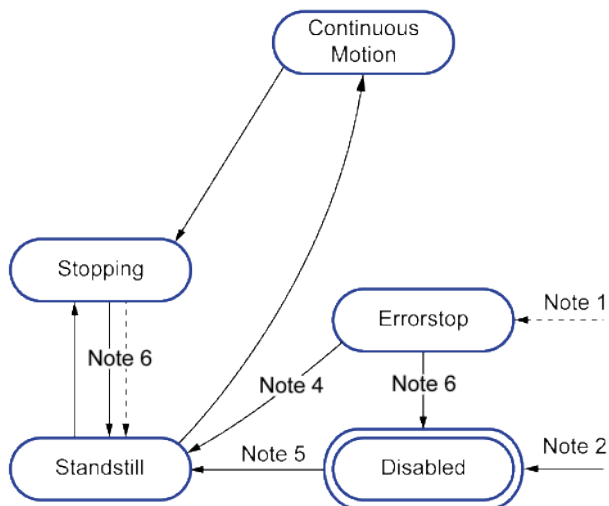
Es können maximal 16 Instanzen jedes Antriebsfunktionsbausteins gleichzeitig in einem Programm verwendet werden.

Wenn dem Modbus Serial E/A-Scanner oder dem Modbus TCP IOScanner ein Gerät hinzugefügt wird, dann weist EcoStruxure Machine Expert - Basic dem Gerät unter Verwendung eines %DRVn-Objekts eine Achse zu, wobei *n* der Nummer des ATV-Antriebs entspricht. Sie müssen jeden Antriebsfunktionsbaustein, den Sie Ihrem Programm hinzufügen, mit einer Achse verknüpfen; Sie müssen also eine Verbindung zwischen dem Funktionsbaustein, der Achse und dem im Modbus Serial E/A-Scanner oder Modbus TCP IOScanner definierten Zielgerät herstellen.

# Zustand der Antriebe und Logiksteuerungen

## Antriebszustandsdiagramm

Der Antrieb befindet sich immer in einem der im Diagramm unten beschriebenen Zustände. Die Ausführung eines Antriebsfunktionsbausteins oder das Auftreten eines Fehlers können einen Zustandsübergang zur Folge haben:



**Hinweis 1** Von allen Zuständen aus, wenn ein Fehler auftritt.

**Hinweis 2** Von jedem Zustand aus (wenn kein *ErrorAxis*), wenn *%MC\_Power\_ATV.status* 0 ist.

**Hinweis 3** Übergang vom Zustand *ErrorStop* zu *Disabled*, nur wenn *%MC\_Reset\_ATV.Done* = 1 und *%MC\_Power\_ATV.status* = 0.

**Hinweis 4** Übergang vom Zustand *ErrorStop* zu *Standstill*, nur wenn *%MC\_Reset\_ATV.Done* = 1 und *%MC\_Power\_ATV.Enable* = 1 und *%MC\_Power\_ATV.Status* = 1.

**Hinweis 5** Übergang vom Zustand *DISABLED* zu *Standstill*, nur wenn *%MC\_Power\_ATV.Enable* = 1 und *%MC\_Power\_ATV.Status* = 1.

**Hinweis 6** Übergang vom Zustand *Stopping* zu *Standstill*, nur wenn *%MC\_Stop\_ATV.Done* = 1 und *%MC\_Stop\_ATV.Execute* = 0.

Die nachstehende Tabelle beschreibt die Antriebszustände:

Status	Beschreibung
<i>Deaktiviert</i>	Ausgangszustand. Der Antrieb befindet sich nicht in einem Betriebszustand oder Fehlerzustand.
<i>Standstill</i>	Der Antrieb befindet sich in einem Betriebszustand (ETA = 16#xx37) und <i>Velocity</i> = 0 (RFRD = 0).
<i>ErrorStop</i>	Der Antrieb befindet sich in einem Fehlerzustand (ETA = 16#xxx8)
<i>Kontinuierliche Bewegung</i>	Der Antrieb befindet sich in einem Betriebszustand (ETA = 16#xx37) und <i>Velocity</i> ≠ 0 (RFRD ≠ 0).
„ <i>Stopping</i> “ ( <i>Anhalten</i> )	Funktionsbaustein <i>MC_Stop_ATV</i> wird ausgeführt.

Der Funktionsbaustein *MC\_ReadStatus\_ATV*, Seite 70 kann zum Lesen des ATV-Antriebs verwendet werden.

## Zustandsübergänge der Logiksteuerungen

Die folgende Tabelle beschreibt wie Antriebsfunktionsbausteine durch Änderungen der Logiksteuerungszustände beeinflusst werden:

Zustand der Logiksteuerung	Einfluss auf Antriebsfunktionsbausteine
<i>RUNNING</i>	Antriebsfunktionsbausteine werden normalerweise gemäß der Benutzerlogik ausgeführt.
<i>STOPPED (GESTOPPT)</i>	Die konfigurierten Antriebsachsen werden gestoppt, wenn die Steuerung in den <i>STOPPED</i> -Zustand übergeht, es sei denn, die Option <b>Fehlerausweichverhalten</b> ist auf <b>Werte beibehalten</b> eingestellt.  Wenn die Option <b>Fehlerausweichverhalten</b> auf <b>Fehlerausweichwerte</b> eingestellt ist, dann wird der Befehl 0x00 zum ATV-Antrieb gesendet, was zu einem Switch on Disabled (NST)-Zustand führt. Wenn das <b>Fehlerausweichverhalten</b> auf <b>Werte beibehalten</b> eingestellt ist, dann wird keine Maßnahme ergriffen (der Befehl wird nicht geändert).
<i>HALTED</i>	Die konfigurierten Antriebsachsen werden gestoppt, wenn die Steuerung in den <i>HALTED</i> -Zustand übergeht, es sei denn, die Option <b>Fehlerausweichverhalten</b> ist auf <b>Werte beibehalten</b> eingestellt.  Wenn die Option <b>Fehlerausweichverhalten</b> auf <b>Fehlerausweichwerte</b> eingestellt ist, dann wird der Befehl 0x00 zum ATV-Antrieb gesendet, was zu einem Switch on Disabled (NST)-Zustand führt. Wenn das <b>Fehlerausweichverhalten</b> auf <b>Werte beibehalten</b> eingestellt ist, dann wird keine Maßnahme ergriffen (der Befehl wird nicht geändert).
<i>POWERLESS, EMPTY</i>	Antriebsfunktionsbausteine werden nicht ausgeführt (der Modbus Serial E/A-Scanner oder Modbus TCP IOScanner wird gestoppt).  Dies ist auch der Fall, wenn die Anwendung in der Steuerung aktualisiert wird.

**HINWEIS:** Falls sich die Steuerung im Zustand *HALTED* oder *STOPPED* befindet und die Option **Werte beibehalten** ausgewählt ist, dann erhält der Antrieb keine weiteren Befehle von der Steuerung. Deswegen muss der Antrieb den geeigneten zu übernehmenden Zustand bestimmen. Wenn Sie für Ihren Antrieb **Werte beibehalten** auswählen, müssen Sie dies in Ihrer Gefahren- und Risikoanalyse bezüglich möglicher Folgeschäden oder Gefahren berücksichtigen

### **⚠️ 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.**

## Hinzufügen eines Antriebsfunktionsbausteins

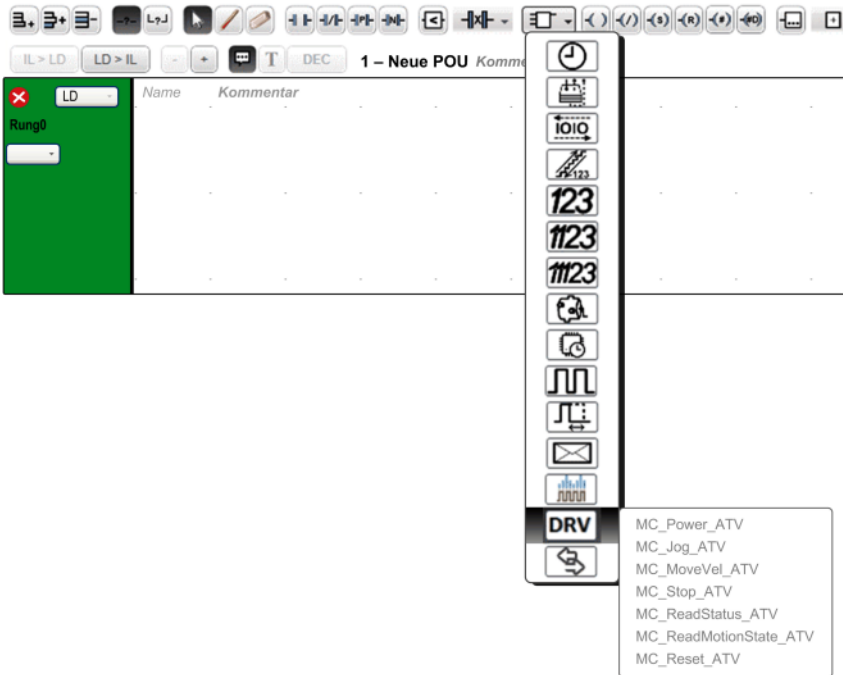
### Voraussetzungen

Voraussetzungen für das Hinzufügen eines Antriebsfunktionsbausteins:

- Ein Modbus Serial E/A-Scanner oder Modbus TCP IOScanner muss auf einer seriellen Leitung oder im Ethernet-Netzwerk konfiguriert werden
- Die zu steuernden ATV-Antriebe müssen auf dem Modbus Serial-E/A-Scanner oder Modbus TCP IOScanner hinzugefügt und konfiguriert werden (siehe Modicon M221, Logic Controller, Programmierhandbuch).

## Hinzufügen eines Antriebsfunktionsbausteins

Gehen Sie wie folgt vor, um eine Instanz eines Antriebsfunktionsbausteins hinzuzufügen:

Schritt	Aktion
1	Wählen Sie die Registerkarte <b>Programmierung</b> aus.
2	Wählen Sie <b>Funktionsbausteine &gt; Antrieb</b> aus, wie in der folgenden Abbildung gezeigt: 
3	Klicken Sie in den Programmbaustein, um den ausgewählten Funktionsbaustein zu platzieren.
4	Verknüpft die Eingänge/Ausgänge des Funktionsbausteins.

## Entfernen eines Funktionsbausteins

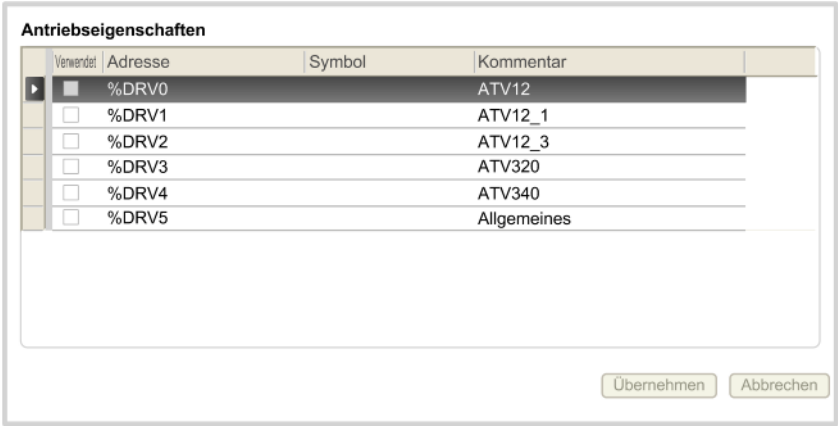
Gehen Sie wie folgt vor, um eine Instanz eines Antriebsfunktionsbausteins zu entfernen:

Schritt	Aktion
1	Klicken Sie auf der Registerkarte <b>Programmierung</b> auf die Instanz des Funktionsbausteins.
2	Drücken Sie <b>Entf.</b> , um den ausgewählten Funktionsbaustein zu löschen.

# Funktionsblock-Konfiguration

## Konfigurieren von Antriebsobjekten

Jeder Antriebsfunktionsbaustein ist mit einem Antriebsobjekt (%DRV) verknüpft. Zur Anzeige einer Liste der konfigurierten Antriebsobjekte:

Schritt	Aktion
1	<p>Wählen Sie die Registerkarte <b>Programmieren &gt; Tools</b> aus und klicken Sie auf <b>Antriebsobjekte &gt; Antrieb</b>, um die Eigenschaften der Antriebsobjekte anzuzeigen.</p> 
2	Aktualisieren Sie die Eigenschaften wie erfordert und klicken Sie auf <b>Anwenden</b>

Antriebsfunktionsbausteine haben die folgenden Eigenschaften:

Parameter	Veränderbar	Wert	Standardwert	Beschreibung
<b>Verwendet</b>	Nein	Aktiviert/ Nicht aktiviert	Unzulässig	Gibt an, ob das Antriebsobjekt im Programm verwendet wird.
<b>Adresse</b>	Nein	%DRVn	%DRVn	Die Adresse des Antriebsobjekts, wobei <i>n</i> der Objektnummer entspricht.
<b>Symbol</b>	Ja	–	–	Hiermit kann ein Symbol angegeben werden, das mit dem Antriebsobjekt verknüpft werden soll.  Klicken Sie doppelt auf die Zelle, um ein Symbol zu definieren oder bearbeiten.
<b>Kommentar</b>	Ja	–	–	Hiermit kann ein Kommentar angegeben werden, der mit dem Antriebsobjekt verknüpft werden soll.  Klicken Sie doppelt auf die Zelle, um einen Kommentar zu definieren oder bearbeiten.

## MC\_Power\_ATV: Endstufe aktivieren/deaktivieren

### Beschreibung

Der Funktionsbaustein aktiviert oder deaktiviert die Endstufe des Antriebs.

Eine steigende Flanke des Eingangs *Enable* aktiviert die Endstufe. Sobald die Endstufe aktiviert ist, wird der Ausgang *Status* auf 1 gesetzt.

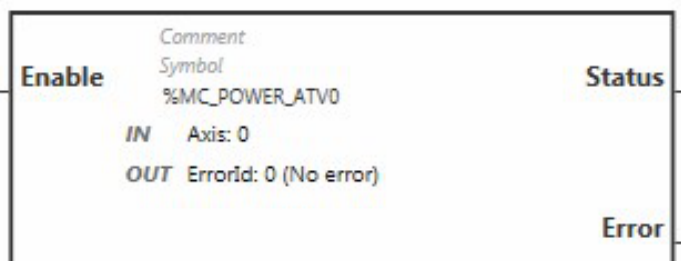
Eine fallende Flanke des Eingangs *Enable* deaktiviert die Endstufe (*Shutdown-Befehl ohne Error*). Sobald die Endstufe deaktiviert ist, wird der Ausgang *Status* zurück auf 0 gesetzt.

Wenn das interne Zustandsregister ETA des ATV-Antriebs vor dem Überschreiten des Timeout-Werts keinen Betriebszustand erreicht hat, wird ein *Timeout Error* generiert. Die Timeout-Zeit ist entweder das Vierfache der Kanalzykluszeit oder 10 Sekunden, je nachdem, welcher Wert größer ist. Es ist eine Dauer von minimal 10 Sekunden erforderlich, damit der Antrieb reagieren kann.

Wenn während der Ausführung des Funktionsbausteins Fehler erkannt werden, wird der Ausgang *Error* auf 1 gesetzt. Dies führt zu einem Shutdown-Befehl (CMD = 16#0006) zum Deaktivieren des ATV-Antriebs (Ready to switch on-Zustand, ETA = 16#xx21).

Wenn ein Fehler auftritt, kann die Endstufe nur durch eine erfolgreiche Ausführung des Funktionsbausteins *MC\_Reset\_ATV*, Seite 74 wiederhergestellt werden.

## Grafische Darstellung



## Eingänge

In dieser Tabelle werden die Eingänge des Funktionsbausteins beschrieben:

Bezeichnung	Objekt	Initialwert	Beschreibung
<i>Aktivieren</i>	-	0	Wird auf 1 gesetzt, um die Ausführung des Funktionsbausteins zu starten und die Endstufe zu aktivieren.  Wird auf 0 gesetzt, um die Ausführung des Funktionsbausteins anzuhalten und die Endstufe zu deaktivieren.
<i>Axis</i>	<i>%MC_POWER_ATVi.AXIS</i> wobei „i“ einem Wert von 0 bis 15 entspricht.	-	Bezeichner der Achse ( <i>%DRV0</i> bis <i>%DRV15</i> ), über der der Funktionsbaustein ausgeführt werden soll.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Bezeichnung	Objekt	Initialwert	Wert
Status	%MC_POWER_ATVi.STATUS wobei „i“ einem Wert von 0 bis 15 entspricht.	0	Standardwert: 0 <ul style="list-style-type: none"> <li>0: Endstufe ist deaktiviert.</li> <li>1: Endstufe ist aktiviert.</li> </ul> Wird auf 1 gesetzt, wenn der ATV-Antrieb einen Betriebszustand erreicht (ETA = 16#xx37).
Fehler	%MC_POWER_ATVi.ERROR wobei „i“ einem Wert von 0 bis 15 entspricht.	0	Auf 0 gesetzt, wenn kein Fehler erkannt wurde. Auf 1 gesetzt, wenn während der Ausführung ein Fehler auftritt. Die Ausführung des Funktionsbausteins ist beendet. Das <i>ErrorId</i> -Ausgangsobjekt gibt die Ursache des Fehlers an.
ErrorId	%MC_POWER_ATVi.ERRORID wobei „i“ einem Wert von 0 bis 15 entspricht.	0 (Kein Fehler)	Vom Funktionsbaustein angegebener Fehlercode, wenn der Ausgang <i>Error</i> auf 1 gesetzt ist.  Für weitere Informationen zu Fehlern, siehe Fehlercodes, Seite 77.  Bereich: 0 bis 65535

## Parameter

Klicken Sie doppelt auf den Funktionsbaustein, um die Funktionsbausteinparameter anzuzeigen.

Der *MC\_Power\_ATV*-Funktionsbaustein verfügt über folgende Parameter:

Parameter	Wert	Beschreibung
<b>Verwendet</b>	Adresse verwendet	Wenn diese Option ausgewählt ist, wird die Adresse derzeit in einem Programm verwendet.
<b>Adresse</b>	%MC_Power_ATVi	Die Instanz-ID, wobei i einer Zahl zwischen 0 und der Gesamtanzahl der mit diesem Logic Controller verfügbaren Objekte entspricht. Die maximale Anzahl an Antriebsobjekten finden Sie in der Tabelle Maximale Anzahl Objekte (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<b>Symbol</b>	Symbol	Das diesem Objekt zugeordnete Symbol. Weitere Informationen hierzu finden Sie unter Definieren und Verwenden von Symbolen (siehe EcoStruxure Machine Expert – Basic, Betriebshandbuch).
<b>Achse</b>	%DRVn, wobei n einem Wert von 0 bis 15 entspricht  <b>Keine</b>	Wählen Sie die Achse (Antriebsobjektinstanz) aus, für die der Funktionsbaustein ausgeführt wird.  Das Antriebsobjekt muss zuvor auf dem Modbus TCP I/O-Scanner oder Modbus Serial-E/A-Scanner (siehe Modicon M221, Logic Controller, Programmierhandbuch) konfiguriert worden sein.
<b>Kommentar</b>	Kommentar	Diesem Objekt kann ein optionaler Kommentar zugeordnet werden.  Doppelklicken Sie in die Spalte <b>Kommentar</b> und geben Sie einen Kommentar ein.

Aktualisieren Sie die Parameter nach Bedarf und klicken Sie auf **Übernehmen**.

## MC\_Jog\_ATV: Tippbetrieb starten

### Beschreibung

Der Funktionsbaustein Jog startet den Betriebsmodus. Eine Jog-Operation befiehlt dem Gerät sich mit einer spezifischen Geschwindigkeit vorwärts oder rückwärts zu bewegen.

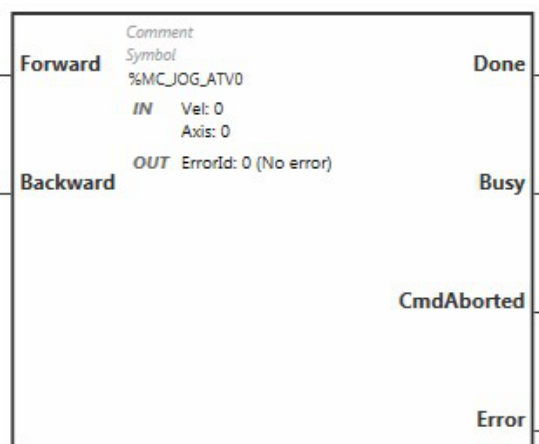
Wenn der Funktionsbaustein MC\_MoveVel\_ATV, Seite 64 oder MC\_Stop\_ATV, Seite 67 während der Ausführung des Funktionsbausteins aktiviert ist (*Busy*-Ausgang auf 1 gesetzt), dann befiehlt der Funktionsbaustein MC\_Jog\_ATV die Bewegung. Der Ausgang *Busy* wird auf 0 zurückgesetzt und der Ausgang *CmdAborted* wird auf 1 gesetzt.

Wenn eine Jog-Operation ausgeführt wird, dann wird eine Änderung des Geschwindigkeitswerts (*Vel*) nur beim Erkennen einer fallenden/steigenden Flanke am Eingang *Forward* oder *Backward* angewandt.

Wenn der Ausgang *Error* oder *CmdAborted* auf 1 gesetzt ist, dann müssen die Eingänge *Forward* und *Backward* erst auf 0 zurückgesetzt werden und eine neue steigende Flanke muss am Eingang *Forward* und/oder *Backward* angewandt werden, um die Bewegung erneut zu starten.

Das Starten einer Jog-Operation während der Funktionsbaustein MC\_Stop\_ATV, Seite 67 ausgeführt wird, führt zu einem Stop Active Error. Das Starten einer Jog-Operation, wenn der Antrieb nicht in einem Betriebszustand ist (ETA ≠ 16#xx37), führt zu einem Not Run Error.

### Grafische Darstellung



## Eingänge

In dieser Tabelle werden die Eingänge des Funktionsbausteins beschrieben:

Eingang	Objekt	Initialwert	Beschreibung
<i>Forward</i>	-	0	Das Einstellen des Eingangs <i>Forward</i> oder <i>Backward</i> auf 1 startet die Jog-Bewegung.  Wenn beide Eingänge <i>Forward</i> und <i>Backward</i> auf 1 gesetzt sind, dann bleibt der Betriebsmodus aktiv, die Jog-Bewegung wird angehalten und der Ausgang <i>Busy</i> behält den Wert 1.  Wenn beide Eingänge <i>Forward</i> und <i>Backward</i> auf 0 gesetzt werden, wird der Betriebsmodus beendet und der Ausgang <i>Done</i> wird für einen Zyklus auf 1 gesetzt.
<i>Backward</i>	-	0	
<i>Vel</i>	%MC_JOG_ATVi.VEL wobei „i“ einem Wert von 0 bis 15 entspricht	0	Zielgeschwindigkeit für den Betriebsmodus Jog, in Umdrehungen pro Minute (RPM).  Wenn eine Jog-Operation ausgeführt wird, dann wird eine Änderung des Geschwindigkeitswerts <i>Vel</i> nur beim Erkennen einer fallenden/steigenden Flanke am Eingang <i>Forward</i> oder <i>Backward</i> angewandt.  Bereich: -32768 bis 32767
<i>Axis</i>	%MC_JOG_ATVi.AXIS wobei „i“ einem Wert von 0 bis 15 entspricht.	-	Bezeichner der Achse (%DRV0 bis %DRV15), über der der Funktionsbaustein ausgeführt werden soll.  Die Achse muss zuerst auf der Registerkarte <b>Konfiguration</b> deklariert werden.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Ausgangsobjekt	Initialwert	Beschreibung
<i>Done</i>	%MC_JOG_ATVi.DONE	0	Wird für einen Zyklus auf 1 gesetzt, wenn beide Eingänge <i>Forward</i> und <i>Backward</i> auf 0 gesetzt sind.  Wird auf 1 gesetzt, um anzuzeigen, dass der Betriebsmodus Jog beendet wurde.
<i>Busy</i>	%MC_JOG_ATVi.BUSY	0	Wird auf 1 eingestellt, wenn: <ul style="list-style-type: none"> <li>• <i>Jog</i> ist in Bearbeitung (<i>Forward</i> =1 oder <i>Backward</i> =1)</li> <li>• Beide Eingänge <i>Forward</i> und <i>Backward</i> werden auf 1 gesetzt, was angibt, dass der Jog-Betriebsmodus aktiv bleibt und die Jog-Bewegung angehalten wird.</li> </ul>
<i>CmdAborted</i>	%MC_JOG_ATVi.CMDABORTED	0	Wird auf 1 gesetzt, wenn der Funktionsbaustein wegen der Ausführung eines anderen Befehls beendet wird.
<i>Error</i>	%MC_JOG_ATVi.ERROR	0	Auf 0 gesetzt, wenn kein Fehler erkannt wurde. Auf 1 gesetzt, wenn während der Ausführung ein Fehler auftritt. Die Ausführung des Funktionsbausteins ist beendet. Das <i>ErrorId</i> -Ausgangsobjekt gibt die Ursache des Fehlers an.
<i>ErrorId</i>	%MC_JOG_ATVi.ERRORID	0 (Kein Fehler)	Vom Funktionsbaustein angegebener Fehlercode, wenn der Ausgang <i>Error</i> auf 1 gesetzt ist.  Für weitere Informationen zu Fehlern, siehe Fehlercodes, Seite 77.  Bereich: 0 bis 65535

## Parameter

Klicken Sie doppelt auf den Funktionsbaustein, um die Funktionsbausteinparameter anzuzeigen.

Der *MC\_Jog\_ATV*-Funktionsbaustein verfügt über folgende Parameter:

Parameter	Wert	Beschreibung
<b>Verwendet</b>	Adresse verwendet	Wenn diese Option ausgewählt ist, wird die Adresse derzeit in einem Programm verwendet.
<b>Adresse</b>	<i>%MC_Jog_ATVi</i>	Die Instanz-ID, wobei i einer Zahl zwischen 0 und der Gesamtanzahl der mit diesem Logic Controller verfügbaren Objekte entspricht. Die maximale Anzahl an Antriebsobjekten finden Sie in der Tabelle Maximale Anzahl Objekte (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<b>Symbol</b>	Symbol	Das diesem Objekt zugeordnete Symbol. Weitere Informationen hierzu finden Sie unter Definieren und Verwenden von Symbolen (siehe EcoStruxure Machine Expert – Basic, Betriebshandbuch).
<b>Achse</b>	<i>%DRVn</i> , wobei n einem Wert von 0 bis 15 entspricht  <b>Keine</b>	Wählen Sie die Achse (Antriebsobjektinstanz) aus, für die der Funktionsbaustein ausgeführt wird.  Das Antriebsobjekt muss zuvor auf dem Modbus TCP IOScanner oder Modbus Serial-E/A-Scanner (siehe Modicon M221, Logic Controller, Programmierhandbuch) konfiguriert worden sein.
<b>Geschw.</b>	Zielgeschwindigkeit	Geben Sie die Zielgeschwindigkeit für den Betriebsmodus Jog ein und drücken Sie auf „Eingabe“.  Standardwert: 0  Bereich: –32768 bis 32767
<b>Kommentar</b>	Kommentar	Diesem Objekt kann ein optionaler Kommentar zugeordnet werden.  Doppelklicken Sie in die Spalte <b>Kommentar</b> und geben Sie einen Kommentar ein.

Aktualisieren Sie die Parameter nach Bedarf und klicken Sie auf **Übernehmen**.

## MC\_MoveVel\_ATV: Bewegen mit einer bestimmten Geschwindigkeit

### Beschreibung

Dieser Funktionsbaustein startet den Profile Velocity-Betriebsmodus mit einer spezifischen Geschwindigkeit. Sobald die Endgeschwindigkeit erreicht ist, wird der Ausgang *InVel* auf 1 gesetzt.

Wenn der Funktionsbaustein *MC\_Jog\_ATV*, Seite 62 oder *MC\_Stop\_ATV*, Seite 67 beim Ausführen dieses Funktionsbausteins aktiviert ist (*Busy*-Ausgang auf 1 gesetzt), dann befiehlt *MC\_MoveVel\_ATV* die Bewegung. In diesem Fall wird der Ausgang *Busy* auf 0 zurückgesetzt und der Ausgang *CmdAborted* wird auf 1 gesetzt.

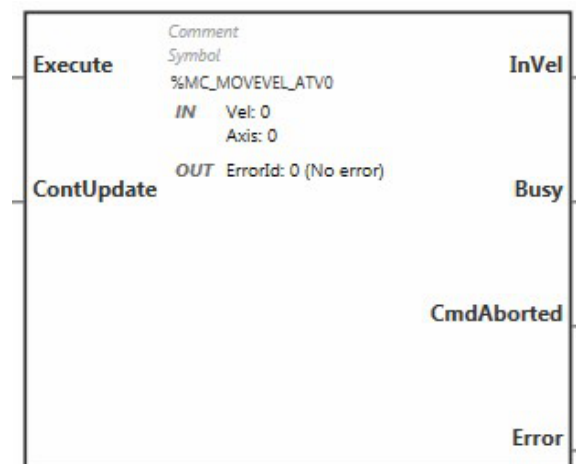
Die Eingangswerte *ContUpdate* und *Vel* werden bei einer steigenden Flanke an Eingang *Execute* angewandt.

Wenn einer der Ausgänge *Error* oder *CmdAborted* von *MC\_MoveVel\_ATV* auf 1 gesetzt wurde, dann ist eine neue steigende Flanke an *Execute* notwendig, um die Bewegung wiederaufzunehmen.

Das Starten dieses Funktionsbausteins während der Funktionsbaustein *MC\_Stop\_ATV*, Seite 67 ausgeführt wird, führt zu einem Stop Active Error.

Das Starten dieses Funktionsbausteins, wenn der Antrieb nicht in einem Betriebszustand ist (ETA ≠ 16#xx37), führt zu einem Not Run Error.

## Grafische Darstellung



## Eingänge

In dieser Tabelle werden die Eingänge des Funktionsbausteins beschrieben:

Eingang	Objekt	Initialwert	Beschreibung
<i>Execute</i>	-	0	Zum Start der Ausführung des Funktionsbausteins auf 1 einstellen.
<i>ContUpdate</i>	-	0	Wird vor der Ausführung des Funktionsbausteins auf 1 gesetzt, um die kontinuierliche Aktualisierung des <i>Vel</i> -Parameterwerts zu aktivieren.
<i>Vel</i>	<code>%MC_MOVEVEL_ATVi.VEL</code>  wobei „i“ einem Wert von 0 bis 15 entspricht.	0	Zielgeschwindigkeit für den Betriebsmodus in Einheiten von Umdrehungen pro Minute (RPM).  Bereich: -32 768...32 767. Ein negativer Wert erzwingt die Bewegung in die entgegengesetzte Richtung.
<i>Axis</i>	<code>%MC_MOVEVEL_ATVi.AXIS</code>  wobei „i“ einem Wert von 0 bis 15 entspricht.	-	Bezeichner der Achse ( <code>%DRV0</code> bis <code>%DRV15</code> ), über der der Funktionsbaustein ausgeführt werden soll.  Die Achse muss zuerst auf der Registerkarte <b>Konfiguration</b> deklariert werden.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Objekt	Initialwert	Beschreibung
<i>InVel</i>		0	0 gibt an, dass die Zielgeschwindigkeit ( <i>Vel</i> ) nicht erreicht wurde.  Wird auf 1 gesetzt, wenn die Zielgeschwindigkeit ( <i>Vel</i> ) erreicht wurde.
<i>Busy</i>	<i>%MC_MOVEVEL_ATVi.BUSY</i>	0	Wird auf 1 gesetzt, wenn der Funktionsbaustein ausgeführt wird.  Bleibt auf 1, selbst wenn die Zielgeschwindigkeit erreicht wurde. Wird auf 0 gesetzt, wenn der Funktionsbaustein gestoppt oder abgebrochen wird.
<i>CmdAborted</i>	<i>%MC_MOVEVEL_ATVi.CMDABORTED</i>	0	Wird auf 1 gesetzt, wenn die Ausführung des Funktionsbausteins wegen der Ausführung eines anderen Befehls beendet wurde.
<i>Error</i>	<i>%MC_MOVEVEL_ATVi.ERROR</i>	0	Auf 0 gesetzt, wenn kein Fehler erkannt wurde. Auf 1 gesetzt, wenn während der Ausführung ein Fehler auftritt. Die Ausführung des Funktionsbausteins ist beendet. Das <i>ErrorId</i> -Ausgangsobjekt gibt die Ursache des Fehlers an.
<i>ErrorId</i>	<i>%MC_MOVEVEL_ATVi.ERRORID</i>	0 (Kein Fehler)	Vom Funktionsbaustein angegebener Fehlercode, wenn der Ausgang <i>Error</i> auf 1 gesetzt ist.  Für weitere Informationen zu Fehlern, siehe Fehlercodes, Seite 77.  Bereich: 0 bis 65535

**HINWEIS:** Wenn der Geschwindigkeitsbefehl des ATV-Antriebs niedrig (< 10) ist, dann sind die Parameter *InVel* und *ConstantVel* möglicherweise ungültig, da der Geschwindigkeitsbereich des ATV-Antriebs selbst nicht korrekt sein könnte.

## Parameter

Klicken Sie doppelt auf den Funktionsbaustein, um die Funktionsbausteinparameter anzuzeigen.

Der *MC\_MoveVel\_ATV*-Funktionsbaustein verfügt über folgende Parameter:

Parameter	Wert	Beschreibung
<b>Verwendet</b>	Adresse verwendet	Wenn diese Option ausgewählt ist, wird die Adresse derzeit in einem Programm verwendet.
<b>Adresse</b>	<i>%MC_MoveVel_ATVi</i>	Die Instanz-ID, wobei i einer Zahl zwischen 0 und der Gesamtanzahl der mit diesem Logic Controller verfügbaren Objekte entspricht. Die maximale Anzahl an Antriebsobjekten finden Sie in der Tabelle Maximale Anzahl Objekte (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<b>Symbol</b>	Symbol	Das diesem Objekt zugeordnete Symbol. Weitere Informationen hierzu finden Sie unter Definieren und Verwenden von Symbolen (siehe EcoStruxure Machine Expert – Basic, Betriebshandbuch).
<b>Achse</b>	<i>%DRVn</i> , wobei n einem Wert von 0 bis 15 entspricht  <b>Keine</b>	Wählen Sie die Achse (Antriebsobjektinstanz) aus, für die der Funktionsbaustein ausgeführt wird.  Das Antriebsobjekt muss zuvor auf dem Modbus TCP IOScanner oder Modbus Serial-E/A-Scanner (siehe Modicon M221, Logic Controller, Programmierhandbuch) konfiguriert worden sein.
<b>Geschw.</b>	Zielgeschwindigkeit	Geben Sie die Zielgeschwindigkeit für den Betriebsmodus ein und drücken Sie auf „Eingabe“.  Standardwert: 0  Bereich: -32768...32767. Ein negativer Wert erzwingt die Bewegung in die entgegengesetzte Richtung.
<b>Kommentar</b>	Kommentar	Diesem Objekt kann ein optionaler Kommentar zugeordnet werden.  Doppelklicken Sie in die Spalte <b>Kommentar</b> und geben Sie einen Kommentar ein.

Aktualisieren Sie die Parameter nach Bedarf und klicken Sie auf **Übernehmen**.

## MC\_Stop\_ATV: Bewegung anhalten

### Beschreibung

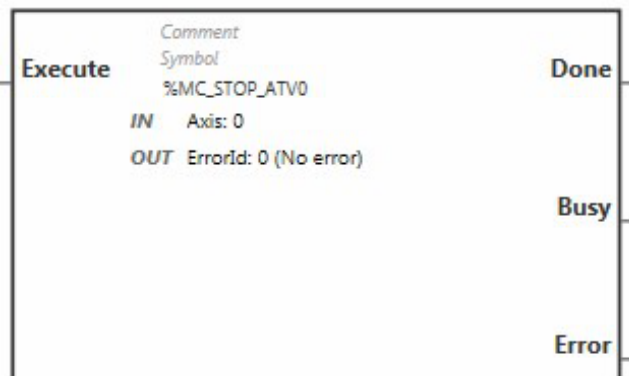
Dieser Funktionsbaustein stoppt die aktuelle Bewegung des spezifizierten Antriebs.

Antriebsspezifische Stop-Parameter, beispielsweise Verzögerung, werden in der Konfiguration des Antriebs zur Verfügung gestellt.

Wenn dies durch eine steigende Flanke am Eingang *Execute* ausgelöst wird, dann wird jede weitere Aktivität am Eingang *Execute* ignoriert, bis *Done* auf TRUE gesetzt wird. Durch die Ausführung eines anderen Bewegungsfunktionsbausteins, während *MC\_Stop\_ATV* beschäftigt („Busy“) ist, wird der Stopp-Vorgang nicht abgebrochen. Der Funktionsblock *MC\_Stop\_ATV* bleibt beschäftigt und der andere Funktionsblock endet mit einem Fehler.

Das Stop-Verfahren kann nur durch ein Deaktivieren der Endstufe unterbrochen werden oder wenn ein Fehler auftritt (beispielsweise ATV Not Run-Fehler oder Modbus TCP IOScanner-Fehler oder Modus Serial E/A-Scannerfehler).

## Grafische Darstellung



## Eingänge

In dieser Tabelle werden die Eingänge des Funktionsbausteins beschrieben:

Eingang	Objekt	Initialwert	Beschreibung
Ausführen	-	0	Zum Start der Ausführung des Funktionsbausteins auf 1 einstellen.  Die Ausführung eines anderen Bewegungsfunktionsbausteins ist nicht möglich, wenn der Ausgang <i>Busy</i> auf 1 gesetzt ist. In diesem Fall wird der andere Funktionsbaustein einen Fehler wiedergeben.
Axis	%MC_STOP_ATVi.AXIS wobei „i“ einem Wert von 0 bis 15 entspricht.	-	Bezeichner der Achse (%DRV0 bis %DRV15), über der der Funktionsbaustein ausgeführt werden soll.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Ausgangsobjekt	Initialwert	Beschreibung
Done	%MC_STOP_ATVi.DONE	0	Wird auf 1 gesetzt, um anzuzeigen, dass der Funktionsbaustein komplett ist.
Busy	%MC_STOP_ATVi.BUSY	0	Wird auf 1 gesetzt, wenn der Funktionsbaustein mit der Ausführung startet.
Error	%MC_STOP_ATVi.ERROR	0	Auf 0 gesetzt, wenn kein Fehler erkannt wurde. Auf 1 gesetzt, wenn während der Ausführung ein Fehler auftritt. Die Ausführung des Funktionsbausteins ist beendet. Das <i>ErrorId</i> -Ausgangsobjekt gibt die Ursache des Fehlers an.
ErrorId	%MC_STOP_ATVi.ERRORID	0 (Kein Fehler)	Vom Funktionsbaustein angegebener Fehlercode, wenn der Ausgang <i>Error</i> auf 1 gesetzt ist.  Für weitere Informationen zu Fehlern, siehe Fehlercodes, Seite 77.  Bereich: 0 bis 65535

## Parameter

Klicken Sie doppelt auf den Funktionsbaustein, um die Funktionsbausteinparameter anzuzeigen.

Der *MC\_Stop\_ATV*-Funktionsbaustein verfügt über folgende Parameter:

Parameter	Wert	Beschreibung
<b>Verwendet</b>	Adresse verwendet	Wenn diese Option ausgewählt ist, wird die Adresse derzeit in einem Programm verwendet.
<b>Adresse</b>	<i>%MC_Stop_ATVi</i>	Die Instanz-ID, wobei i einer Zahl zwischen 0 und der Gesamtanzahl der mit diesem Logic Controller verfügbaren Objekte entspricht. Die maximale Anzahl an Antriebsobjekten finden Sie in der Tabelle Maximale Anzahl Objekte (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<b>Symbol</b>	Symbol	Das diesem Objekt zugeordnete Symbol. Weitere Informationen hierzu finden Sie unter Definieren und Verwenden von Symbolen (siehe EcoStructure Machine Expert – Basic, Betriebshandbuch).
<b>Achse</b>	<i>%DRVn</i> , wobei n einem Wert von 0 bis 15 entspricht  <b>Keine</b>	Wählen Sie die Achse (Antriebsobjektinstanz) aus, für die der Funktionsbaustein ausgeführt wird.  Das Antriebsobjekt muss zuvor auf dem Modbus TCP IOScanner oder Modbus Serial-E/A-Scanner (siehe Modicon M221, Logic Controller, Programmierhandbuch) konfiguriert worden sein.
<b>Kommentar</b>	Kommentar	Diesem Objekt kann ein optionaler Kommentar zugeordnet werden.  Doppelklicken Sie in die Spalte <b>Kommentar</b> und geben Sie einen Kommentar ein.

Aktualisieren Sie die Parameter nach Bedarf und klicken Sie auf **Übernehmen**.

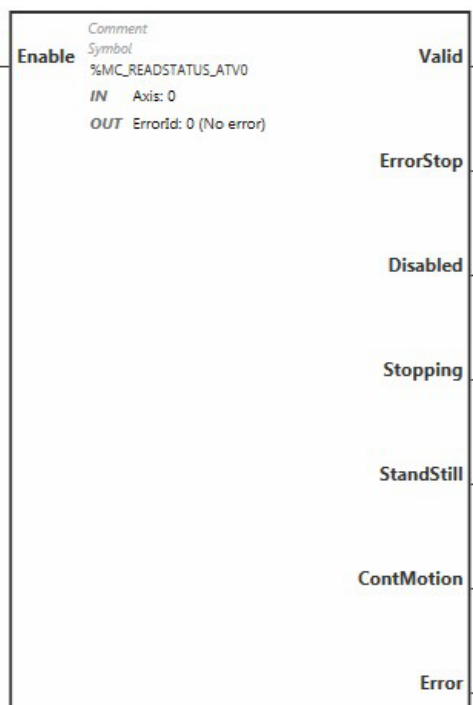
# MC\_ReadStatus\_ATV: Gerätestatus lesen

## Beschreibung

Der Funktionsbaustein liest des Zustand des ATV-Antriebs.

Für Details zu den Zuständen, siehe das Antriebszustandsdiagramm, Seite 56.

## Grafische Darstellung



## Eingänge

In dieser Tabelle werden die Eingänge des Funktionsbausteins beschrieben:

Bezeichnung	Objekt	Initialwert	Beschreibung
Aktivieren	-	0	Zum Aktivieren des Funktionsbausteins auf 1 setzen.
Axis	%MC_READSTATUS_ATVi. AXIS  wobei „i“ einem Wert von 0 bis 15 entspricht.	-	Bezeichner der Achse (%DRV0 bis %DRV15), über der der Funktionsbaustein ausgeführt werden soll.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Bezeichnung	Objekt	Initialwert	Beschreibung
<i>Valid</i>	%MC_READSTATUS_ATVi. VALID	0	Wird auf 1 gesetzt, während der Funktionsbaustein ohne Fehler ausgeführt wird.
<i>ErrorStop</i>	%MC_READSTATUS_ATVi. ERRÖRSTOP	0	Wird auf 1 gesetzt, wenn sich der ATV-Antrieb in einem Fehlerzustand (ETA = 16#xxx8) befindet.
<i>Disabled</i>	%MC_READSTATUS_ATVi. DISABLED	0	Wird auf 1 gesetzt, wenn sich der ATV-Antrieb nicht in einem Betriebszustand oder Fehlerzustand befindet.
<i>Stopping</i>	%MC_READSTATUS_ATVi. STOPPING	0	Wird auf 1 gesetzt, wenn der Funktionsbaustein MC_Stop_ATV ausgeführt oder die Bewegung angehalten wird.
<i>Standstill</i>	%MC_READSTATUS_ATVi. STANDSTILL	0	Wird auf 1 gesetzt, wenn sich der ATV-Antrieb in einem Betriebszustand befindet und die Geschwindigkeit 0 ist (ETA = 16#xx37 und RFRD = 0).
<i>ContMotion</i>	%MC_READSTATUS_ATVi. CONTMOTION	0	Wird auf 1 gesetzt, wenn sich der ATV-Antrieb in einem Betriebszustand befindet und die Geschwindigkeit nicht 0 ist (ETA = 16#xx37 und RFRD ≠ 0).
<i>Error</i>	%MC_READSTATUS_ATVi. ERROR	0	Auf 0 gesetzt, wenn kein Fehler erkannt wurde. Auf 1 gesetzt, wenn während der Ausführung ein Fehler auftritt. Die Ausführung des Funktionsbausteins ist beendet. Das <i>ErrorId</i> -Ausgangsobjekt gibt die Ursache des Fehlers an.
<i>ErrorId</i>	%MC_READSTATUS_ATVi. ERRÖRID	0 (Kein Fehler)	Vom Funktionsbaustein angegebener Fehlercode, wenn der Ausgang <i>Error</i> auf 1 gesetzt ist.  Für weitere Informationen zu Fehlern, siehe Fehlercodes, Seite 77.  Bereich: 0 bis 65535

## Parameter

Klicken Sie doppelt auf den Funktionsbaustein, um die Funktionsbausteinparameter anzuzeigen.

Der *MC\_ReadStatus\_ATV*-Funktionsbaustein verfügt über folgende Parameter:

Parameter	Wert	Beschreibung
<b>Verwendet</b>	Adresse verwendet	Wenn diese Option ausgewählt ist, wird die Adresse derzeit in einem Programm verwendet.
<b>Adresse</b>	%MC_ReadStatus_ATVi	Die Instanz-ID, wobei i einer Zahl zwischen 0 und der Gesamtanzahl der mit diesem Logic Controller verfügbaren Objekte entspricht. Die maximale Anzahl an Antriebsobjekten finden Sie in der Tabelle Maximale Anzahl Objekte (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<b>Symbol</b>	Symbol	Das diesem Objekt zugeordnete Symbol. Weitere Informationen hierzu finden Sie unter Definieren und Verwenden von Symbolen (siehe EcoStruxure Machine Expert – Basic, Betriebshandbuch).
<b>Achse</b>	%DRVn, wobei n einem Wert von 0 bis 15 entspricht  <b>Keine</b>	Wählen Sie die Achse (Antriebsobjektinstanz) aus, für die der Funktionsbaustein ausgeführt wird.  Das Antriebsobjekt muss zuvor auf dem Modbus TCP IOScanner oder Modbus Serial-E/A-Scanner (siehe Modicon M221, Logic Controller, Programmierhandbuch) konfiguriert worden sein.
<b>Kommentar</b>	Kommentar	Diesem Objekt kann ein optionaler Kommentar zugeordnet werden.  Doppelklicken Sie in die Spalte <b>Kommentar</b> und geben Sie einen Kommentar ein.

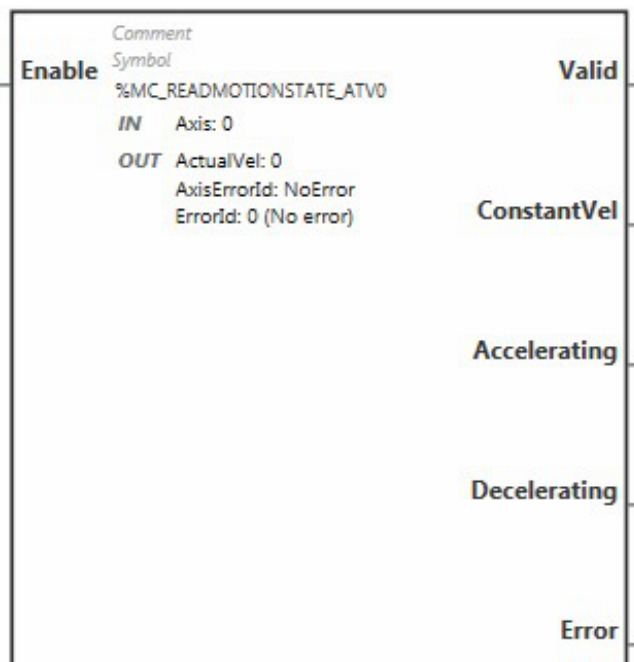
Aktualisieren Sie die Parameter nach Bedarf und klicken Sie auf **Übernehmen**.

# MC\_ReadMotionState\_ATV: Bewegungsstatus lesen

## Beschreibung

Dieser Funktionsbaustein gibt Zustandsinformationen über die vom ATV-Antrieb gelesene Bewegung wieder.

## Grafische Darstellung



## Eingänge

In dieser Tabelle werden die Eingänge des Funktionsbausteins beschrieben:

Eingang	Objekt	Initialwert	Beschreibung
<i>Enable</i>	-	0	Zum Start der Ausführung des Funktionsbausteins auf 1 setzen.
<i>Axis</i>	%MC_READMOTIONSTATE_ATVi. AXIS  wobei „i“ einem Wert von 0 bis 15 entspricht.	-	Bezeichner der Achse (%DRV0 bis %DRV15), über der der Funktionsbaustein ausgeführt werden soll.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Objekt	Initialwert	Beschreibung
<i>Valid</i>	%MC_READMOTIONSTATE_ATVi.VALID	0	Wird auf 1 gesetzt, während der Funktionsbaustein ohne Fehler ausgeführt wird.
<i>ConstantVel</i>	%MC_READMOTIONSTATE_ATVi.CONSTANTVEL	0	Wird auf 1 gesetzt, wenn eine Bewegung mit konstanter Geschwindigkeit ausgeführt wird (ETA-Register)
<i>Accelerating</i>	%MC_READMOTIONSTATE_ATVi.ACCELERATING	0	Wird auf 1 gesetzt, wenn der Motor beschleunigt wird (ETI-Register).
<i>Decelerating</i>	%MC_READMOTIONSTATE_ATVi.DECCELERATING	0	Wird auf 1 gesetzt, wenn der Motor verlangsamt wird (ETI-Register).
<i>Error</i>	%MC_READMOTIONSTATE_ATVi.ERROR	0	Auf 0 gesetzt, wenn kein Fehler erkannt wurde. Auf 1 gesetzt, wenn während der Ausführung ein Fehler auftritt. Die Ausführung des Funktionsbausteins ist beendet. Das <i>ErrorId</i> -Ausgangsobjekt gibt die Ursache des Fehlers an.
<i>ActualVel</i>	%MC_READMOTIONSTATE_ATVi.ACTUALVEL	0	Die vom ATV-Antrieb wiedergegebene Geschwindigkeit (RFRD-Register).  Bereich: -32768 bis 32767
<i>AxisErrorId</i>	%MC_READMOTIONSTATE_ATVi.AXISERRORID	0	Der vom ATV-Antrieb wiedergegebene Bezeichner des Achsenfehlers (DP0-Register). Wenn sich der Antrieb in einem Fehlerzustand befindet, tritt ein Achsenfehler auf.  Wird auf 0 gesetzt, wenn sich der Antrieb nicht in einem Fehlerzustand (ETA-Register ≠ 16#xxx8) befindet  Für weitere Informationen zu Achsenfehlern, siehe <i>AxisErrorId</i> Fehlercodes, Seite 77.  Bereich: -32768 bis 32767
<i>ErrorId</i>	%MC_READMOTIONSTATE_ATVi.ERRORID	Kein Fehler (nOF)	Vom Funktionsbaustein angegebener Fehlercode, wenn der Ausgang <i>Error</i> auf 1 gesetzt ist.  Für weitere Informationen zu Fehlern, siehe Fehlercodes, Seite 77.  Bereich: 0 bis 65535

**HINWEIS:** Wenn der Geschwindigkeitsbefehl des ATV-Antriebs niedrig (< 10) ist, dann sind die Parameter *InVel* und *ConstantVel* möglicherweise ungültig, da der Geschwindigkeitsbereich des ATV-Antriebs selbst nicht korrekt sein könnte.

## Parameter

Klicken Sie doppelt auf den Funktionsbaustein, um die Funktionsbausteinparameter anzuzeigen.

Der *MC\_ReadMotionState\_ATV*-Funktionsbaustein verfügt über folgende Parameter:

Parameter	Wert	Beschreibung
<b>Verwendet</b>	Adresse verwendet	Wenn diese Option ausgewählt ist, wird die Adresse derzeit in einem Programm verwendet.
<b>Adresse</b>	<i>%MC_ReadMotionState_ATVi</i>	Die Instanz-ID, wobei i einer Zahl zwischen 0 und der Gesamtanzahl der mit diesem Logic Controller verfügbaren Objekte entspricht. Die maximale Anzahl an Antriebsobjekten finden Sie in der Tabelle Maximale Anzahl Objekte (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<b>Symbol</b>	Symbol	Das diesem Objekt zugeordnete Symbol. Weitere Informationen hierzu finden Sie unter Definieren und Verwenden von Symbolen (siehe EcoStruxure Machine Expert – Basic, Betriebshandbuch).
<b>Achse</b>	<i>%DRVn</i> , wobei n einem Wert von 0 bis 15 entspricht  <b>Keine</b>	Wählen Sie die Achse (Antriebsobjektinstanz) aus, für die der Funktionsbaustein ausgeführt wird.  Das Antriebsobjekt muss zuvor auf dem Modbus TCP IOScanner oder Modbus Serial-E/A-Scanner (siehe Modicon M221, Logic Controller, Programmierhandbuch) konfiguriert worden sein.
<b>Kommentar</b>	Kommentar	Diesem Objekt kann ein optionaler Kommentar zugeordnet werden.  Doppelklicken Sie in die Spalte <b>Kommentar</b> und geben Sie einen Kommentar ein.

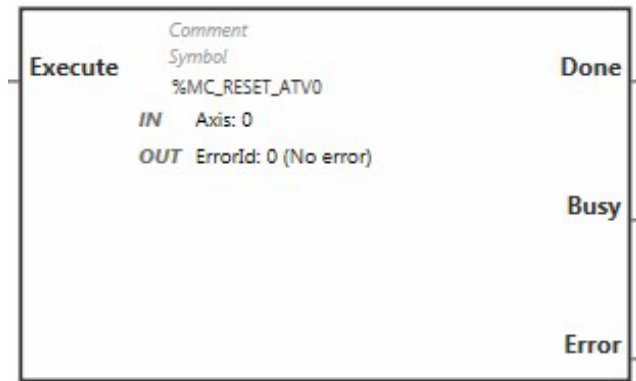
Aktualisieren Sie die Parameter nach Bedarf und klicken Sie auf **Übernehmen**.

## MC\_Reset\_ATV: Quittieren und Zurücksetzen eines Fehlers

### Beschreibung

Dieser Funktionsbaustein wird verwendet, um einen Fehler zu bestätigen und die Fehlerbedingung auf dem Antrieb zu reinitialisieren. Weitere Informationen finden Sie im Antriebszustandsdiagramm, Seite 56.

## Grafische Darstellung



## Eingänge

In dieser Tabelle werden die Eingänge des Funktionsbausteins beschrieben:

Bezeichnung	Objekt	Initialwert	Beschreibung
<i>Ausführen</i>	-	0	Zum Start der Ausführung des Funktionsbausteins auf 1 einstellen.
<i>Axis</i>	<i>%MC_RESET_ATVi.AXIS</i> wobei „i“ einem Wert von 0 bis 15 entspricht.	-	Bezeichner der Achse ( <i>%DRV0</i> bis <i>%DRV15</i> ), über der der Funktionsbaustein ausgeführt werden soll.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Ausgangsobjekt	Initialwert	Beschreibung
<i>Fertig</i>	<i>%MC_RESET_ATVi.DONE</i>	0	Wird auf 1 gesetzt, wenn <i>Reset</i> ohne Fehler beendet wurde.
<i>Belegt</i>	<i>%MC_RESET_ATVi.BUSY</i>	0	Wird auf 1 gesetzt, wenn der Funktionsbaustein mit der Ausführung startet.
<i>Fehler</i>	<i>%MC_RESET_ATVi.ERROR</i>	0	Wird auf 1 gesetzt, wenn das Gerät nach Überschreitung des Timeouts im Fehlerzustand verbleibt. Die Timeout-Zeit ist entweder das Vierfache der Kanalzykluszeit oder 200 ms, je nachdem, welcher Wert größer ist. Es ist eine Dauer von minimal 200 ms erforderlich, damit der Antrieb reagieren kann.  Informationen zur Konfiguration der Kanalzykluszeit finden Sie unter Konfigurieren von Kanälen (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<i>ErrorId</i>	<i>%MC_RESET_ATVi.ERRORID</i>	0 (Kein Fehler)	Vom Funktionsbaustein angegebener Fehlercode, wenn der Ausgang <i>Error</i> auf 1 gesetzt ist.  Für weitere Informationen zu Fehlern, siehe Fehlercodes, Seite 77.  Bereich: 0 bis 65535

## Parameter

Klicken Sie doppelt auf den Funktionsbaustein, um die Funktionsbausteinparameter anzuzeigen.

Der *MC\_Reset\_ATV*-Funktionsbaustein verfügt über folgende Parameter:

Parameter	Wert	Beschreibung
<b>Verwendet</b>	Adresse verwendet	Wenn diese Option ausgewählt ist, wird die Adresse derzeit in einem Programm verwendet.
<b>Adresse</b>	<i>%MC_Reset_ATVi</i>	Die Instanz-ID, wobei i einer Zahl zwischen 0 und der Gesamtanzahl der mit diesem Logic Controller verfügbaren Objekte entspricht. Die maximale Anzahl an Antriebsobjekten finden Sie in der Tabelle Maximale Anzahl Objekte (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<b>Symbol</b>	Symbol	Das diesem Objekt zugeordnete Symbol. Weitere Informationen hierzu finden Sie unter Definieren und Verwenden von Symbolen (siehe EcoStruxure Machine Expert – Basic, Betriebshandbuch).
<b>Achse</b>	<i>%DRVn</i> , wobei n einem Wert von 0 bis 15 entspricht  <b>Keine</b>	Wählen Sie die Achse (Antriebsobjektinstanz) aus, für die der Funktionsbaustein ausgeführt wird.  Das Antriebsobjekt muss zuvor auf dem Modbus TCP IOScanner oder Modbus Serial-E/A-Scanner (siehe Modicon M221, Logic Controller, Programmierhandbuch) konfiguriert worden sein.
<b>Kommentar</b>	Kommentar	Diesem Objekt kann ein optionaler Kommentar zugeordnet werden.  Doppelklicken Sie in die Spalte <b>Kommentar</b> und geben Sie einen Kommentar ein.

Aktualisieren Sie die Parameter nach Bedarf und klicken Sie auf **Übernehmen**.

## Fehlercodes

### FehlerId Fehlercode

In dieser Tabelle werden die möglichen Fehlercodes der Funktionsbausteine aufgelistet:

Wert	Name	Beschreibung
0	Kein Fehler	Kein Fehler erkannt.
1	E/A-Scanner-Fehler	Fehler auf E/A-Scanner erkannt <sup>(1)</sup> .
2	ATV befindet sich im Fehlerzustand	Der ATV-Antrieb befindet sich in einem Fehlerzustand (ETA = 16#xxx8).
3	Timeout-Fehler	Eine Zeitüberschreitung (Timeout) ist eingetreten bevor der <i>MC_Power_ATV</i> -Funktionsbaustein den richtigen Zustand vom Antrieb empfangen hat.
4	Ungültiger ATV-Zustand	Der ATV-Antrieb hat einen ungültigen ETA-Wert.
5	Reset-Fehler	Der <i>MC_Reset_ATV</i> -Funktionsbaustein wird angefragt, während sich der ATV-Antrieb in einem Fehlerzustand befindet.
6	Fehler: Stop-Modus aktiv	Der Funktionsbaustein <i>MC_Jog_ATV</i> oder <i>MV_MoveVelocity_ATV</i> wird angefragt, während <i>MC_Stop</i> aktiv ist.
7	Fehler: ATV nicht in Run-Modus	Der Funktionsbaustein <i>MC_Jog_ATV</i> oder <i>MV_MoveVelocity_ATV</i> wird angefragt, während der ATV-Antrieb nicht betriebsbereit ist.
8	Fehler: Ungültiger AxisRef-Parameter	<i>AxisRef</i> -Eingang <i>%DRV</i> des Funktionsbausteins ist ungültig (nicht vorhanden in Modbus TCP IOScanner oder Konfiguration des Modbus seriellen E/A-Scanners (siehe Modicon M221, Logic Controller, Programmierhandbuch)).
9	Interner Fehler	Ein Firmware-Fehler ist aufgetreten.

(1) Nur für Modbus TCP IOScanner.

Wenn der Funktionsbaustein *%MC\_Power\_ATV* einen E/A-Scannerfehler auslöst, während das Gerät abgefragt wird, kann das auf eine Überlastung des Ethernet-Netzwerkes zurückgeführt werden. Gehen Sie wie folgt vor, um die Fehlerursache zu identifizieren:

- Prüfen Sie den Status des E/A-Scanners: *%SW212*.
- Prüfen Sie den Antriebsstatus: *%IWNS (300+x)*.
- Prüfen Sie den Kanalstatus: *%IWNS (300+x).y*.
- Erhöhen Sie das **Timeout für Antwort** des Antriebs.

### AxisErrorId Fehlercodes

In dieser Tabelle werden die möglichen Achsenfehlercodes des Funktionsbausteins aufgelistet, die vom *MC\_ReadMotionStatus*-Funktionsbaustein wiedergegeben werden:

Wert	Name
0	Kein Fehler (nOF)
2	EEPROM-Steuerung (EEF1)
3	Falsche Konfiguration (CFF)
4	Ungültige Konfiguration (CFI)
5	Modbus-Kommunikationsunterbrechung (SLF1)
6	Interner Verbindungsfehler (ILF)
7	Fieldbus-Kommunikationsunterbrechung (CnF)

Wert	Name
8	Externer Fehler (EPF1)
9	Überstrom (OCF)
10	Kondensator mit Vorladung (CrF)
13	AI2 4-20 mA Verlust (LFF2)
15	Überhitzung des Eingangs (IHF)
16	Überhitzung des Antriebs (OHF)
17	Überlast des Motors (OLF)
18	DC-Bus-Überspannung (ObF)
19	Überspannung der Hauptversorgung (OSF)
20	Phasenausfall eines einzelnen Ausgangs (OPF1)
21	Phasenausfall am Eingang (PHF)
22	Unterspannung der Hauptversorgung (USF)
23	Kurzschluss am Motor (SCF1)
24	Übergeschwindigkeit des Motors (SOF)
25	Auto-Tuning-Fehler
26	Interner Fehler 1 (InF1)
27	Interner Fehler 2 (InF2)
28	Interner Fehler 3 (InF3)
29	Interner Fehler 4 (InF4)
30	EEPROM-ROM-Leistung (EEF2)
32	Masse-Kurzschluss (SCF3)
33	Phasenausfall am Ausgang (OPF2)
37	Interner Fehler (InF7)
38	Fieldbus-Fehler (EPF2)
40	Interner Fehler 8 (InF8)
42	PC-Kommunikationsunterbrechung (SLF2)
45	HMI-Kommunikationsunterbrechung (SLF3)
51	Interner Fehler 9 (InF9)
52	Interner Fehler 10 (InFA)
53	Interner Fehler 11 (InFb)
54	IGBT-Überhitzung (tJF)
55	IGBT-Kurzschluss (SCF4)
56	Kurzschluss am Motor (SCF5)
60	Interner Fehler 12 (InFC)
64	Eingangsschalterschütz (LCF)
68	Interner Fehler 6 (InF6)
69	Interner Fehler 14 (InFE)
71	AI3 4-20mA Verlust (LFF3)
72	AI4 4-20mA Verlust (LFF4)
73	Plattenkompatibilität (HCF)
77	Conf-Übertragungsfehler (CFI2)

Wert	Name
79	AI5 4-20mA Verlust (LFF5)
99	Kanalumschaltfehler (CSF)
100	Prozess-Unterlast (ULF)
101	Prozess-Überlast (OLC)
105	Winkelfehler (ASF)
106	AI1 4-20mA Verlust (LFF1)
107	Fehler der Sicherheitsfunktion (SAFF)
110	AI2 Th Erkannter Fehler (th2F)
111	AI2 Th Fehler des Temperatursensors (t2CF)
112	AI3 Th Erkannter Fehler (th3F)
113	AI3 Th Fehler des Temperatursensors (t3CF)
114	Startfehler des Pumpenzyklus (PCPF)
119	Pumpenfehler: Niedriger Durchfluss (PLFF)
120	AI4 Th Erkannter Fehler (th4F)
121	AI4 Th Fehler des Temperatursensors (t4CF)
122	AI5 Th Erkannter Fehler (th5F)
123	AI5 Th Fehler des Temperatursensors (t5CF)
126	Fehler: Trockenlauf (drYF)
127	PID Feedback-Fehler (PFMF)
128	Programmladefehler (PGLF)
129	Programmausführungsfehler (PGLF)
130	Fehler der Lead-Pumpe (MPLF)
131	Fehler: Niedrige Stufe (LCLF)
132	Fehler: Hohe Stufe (LCLF)
142	Interner Fehler 16 (InFG)
143	Interner Fehler 17 (InFH)
144	Interner Fehler 0 (InF0)
146	Interner Fehler 13 (InFd)
149	Interner Fehler 21 (InFL)
151	Interner Fehler 15 (InFF)
152	Aktualisierungsfehler der Firmware (FEr)
153	Interner Fehler 22 (InFM)
154	Interner Fehler 25 (InFP)
155	Interner Fehler 20 (InF)
157	Interner Fehler 27 (InFr)

# Impulswellenausgang (%PTO)

## Inhalt dieses Kapitels

Beschreibung .....	80
Konfiguration .....	90
Programmierung .....	98
Homing-Modi .....	101
Datenparameter .....	107
Betriebsmodi .....	112
<b>Bewegungs-Funktionsbausteine</b> .....	116
<b>Administrative</b> Funktionsbausteine .....	141

## Verwenden von PTO-Funktionsbausteinen (Pulse Train Output)

Dieser Abschnitt enthält Beschreibungen und Programmierrichtlinien für die Verwendung von `Pulse Train Output`-Funktionsbausteinen.

### Beschreibung

#### Überblick

Dieser Abschnitt beschreibt die Funktion *Pulse Train Output*.

## Impulswellenausgang (PTO)

### Einführung

Die M221 *PTO*-Funktion stellt zwei Impulswellenausgänge für eine bestimmte Anzahl von Impulsen und eine bestimmte Geschwindigkeit (Frequenz) bereit. Die *PTO*-Funktion dient dazu, die Geschwindigkeit bzw. Positionierung von unabhängigen linearen, einachsigen Stepper- oder Servoantrieben im Open-Loop-Modus zu steuern. Die *PTO*-Funktion erhält keine Positionsrückmeldungen vom Prozess. Deshalb müssen die Positionsinformationen in den Antrieb integriert werden. Die Funktionen von *PLS* (Impuls), *PWM* (Puls-Weiten-Modulation), *PTO* (Impulsfolgenausgabe), *FREQGEN* (Frequenzgenerator) verwenden dieselben dedizierten Ausgänge. Doch jede dieser vier Funktionen muss einen eigenen Kanal nutzen.

Ein *PTO*-Kanal kann optionale Schnittstellensignale für Homing (*Ref*), Ereignis (*Probe*), Grenzwerte (*LimP*, *LimN*) oder Antriebsschnittstelle (*DriveReady*, *DriveEnable*) verwenden.

Darüber hinaus wird der Ursprungs-Offset- und Spielausgleich automatisch verwaltet, was zu einer Verbesserung der Genauigkeit bei der Positionierung beiträgt. Diagnosefunktionen stehen zur Überwachung des Status zur Verfügung.

## Unterstützte Funktionen

Die *PTO*-Kanäle unterstützen die folgenden Funktionen:

- Zwei Ausgangsmodi (zwei Kanäle für Impuls und Richtung und ein Kanal für CW/CCW)
- Einachsenbewegungen (Geschwindigkeit und Position)
- Relative und absolute Positionierung mit automatischer Richtungsverwaltung
- Trapez- und S-Kurven-Beschleunigung und -Verzögerung
- Homing (vier Modi, mit Offset-Ausgleich)
- Dynamische Änderung von Beschleunigung, Verzögerung, Geschwindigkeit und Position
- Wechsel vom Geschwindigkeits- zum Positionsmodus
- Bewegungswarteschlange (Puffer für eine Bewegung)
- ereignisgesteuerte Positionserfassung und Bewegungsauslösung (über den Probe-Eingang)
- Spielausgleich
- Grenzwerte (Hardware- und Software-basiert)
- Diagnose

**HINWEIS:** Motion-Funktionsbausteine, Seite 116 und administrative Funktionsbausteine, Seite 141 helfen bei der Programmierung dieser Funktionen.

## Eigenschaften des PTO

*PTO*-Kanäle verfügen über bis zu fünf physische Eingänge.

- Zwei werden der *PTO*-Funktion über die Konfiguration zugeordnet und werden bei einer steigenden Flanke auf dem Eingang berücksichtigt:
  - REF-Eingang
  - Sondeneingang
- Drei sind dem Funktionsbaustein *MC\_Power\_PTO*, Seite 119 zugeordnet. Sie haben keine feste Zuordnung (sind nicht auf dem Konfigurationsbildschirm konfiguriert) und werden mit allen anderen Eingängen gelesen:
  - *DriveReady*-Eingang
  - "Grenzwert positiv"-Eingang
  - "Grenzwert negativ"-Eingang

**HINWEIS:** Diese Eingänge werden wie alle anderen Eingänge verwaltet, sie werden jedoch von der *PTO*-Funktion genutzt, wenn sie zusammen mit einem *MC\_Power\_PTO*, Seite 119-Funktionsbaustein verwendet werden.

**HINWEIS:** Die „Grenzwert positiv“- und „Grenzwert negativ“-Eingänge sind erforderlich, um ein Überfahren zu vermeiden.

### ▲ **WARNUNG**

#### **UNBEABSICHTIGTER GERÄTEBETRIEB**

- Stellen Sie sicher, dass die Positionsschalter der Steuerungshardware in die Konzeption und Logik Ihrer Anwendung integriert sind.
- Montieren Sie die Positionsschalter der Steuerungshardware in einer Position, die einen angemessenen Bremsabstand ermöglicht.

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

Ein PTO-Kanal verfügt über bis zu drei physische Ausgänge.

- Zwei Ausgänge sind für die Verwaltung des Ausgangsmodus der PTO-Funktion vorgeschrieben. Sie haben eine feste Zuordnung und müssen in der Konfiguration aktiviert werden:
  - CW / CCW
  - Im/Gegen den Uhrzeigersinn
- Der andere Ausgang, *DriveEnable*, ist mit dem Funktionsbaustein *MC\_Power\_PTO*, Seite 119 verbunden. Er hat keine feste Zuordnung und wird am Ende eines MAST-Zyklus als reguläre Ausgänge geschrieben.

Die PTO-Funktion hat die folgenden Leistungsmerkmale:

Merkmal	Wert
Anzahl der Kanäle	2 oder 4, abhängig vom Modul
Anzahl der Achsen	1 pro Kanal
Positionsbereich	-2.147.483.648...2.147.483.647 (32 Bit)
Minimale Geschwindigkeit	0 Hz
Maximale Geschwindigkeit	100 kHz (für einen 40/60-Arbeitszyklus und max. 200 mA)
Minimale Schrittweite	1 Hz
Genauigkeit bei Geschwindigkeit	1 %
Beschleunigung/Verzögerung min.	1 Hz/ms
Beschleunigung/Verzögerung max.	100 kHz/ms
Ursprungsoffset	-2.147.483.648...2.147.483.647 (32 Bit)
Bereiche der Software-Grenzwerte	-2.147.483.648...2.147.483.647 (32 Bit)

## Impulsausgangsmodi

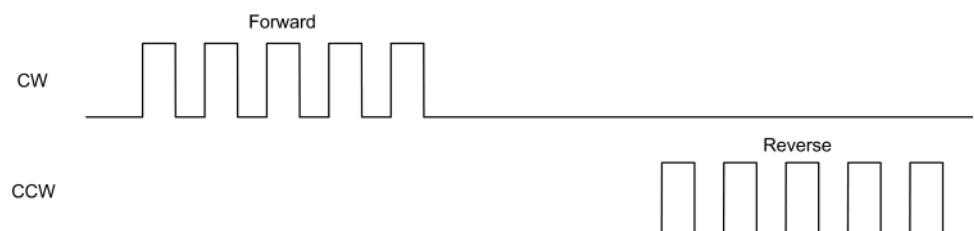
### Überblick

Es gibt zwei verschiedene Ausgangsmodi:

- Im Uhrzeigersinn (CW) / Gegen den Uhrzeigersinn (CCW)
- Impuls / Richtung

### Modus im Uhrzeigersinn/ClockWise (CW) / Modus gegen den Uhrzeigersinn/CounterClockwise (CCW)

Dieser Modus generiert ein Signal, das die Betriebsgeschwindigkeit und Richtung des Motors definiert. Dieses Signal wird nur auf dem ersten PTO-Kanal implementiert (nur PTO0).



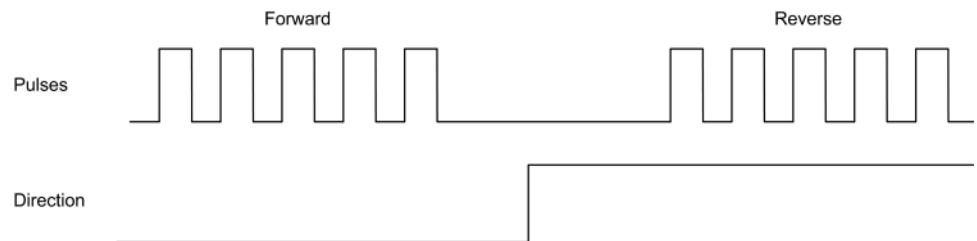
**HINWEIS:** PTO1 ist bei Auswahl dieses Modus nicht verfügbar.

## Modus Impuls / Richtung

Dieser Modus generiert zwei Signale an den PTO-Kanälen:

- Der Impulsausgang gibt die Betriebsgeschwindigkeit des Motors an (*Pulses*).
- Der Richtungsausgang dient zur Definition der Drehrichtung des Motors (*Direction*).

**HINWEIS:** Der Bewegungsausgang kann deaktiviert werden, wenn er für die Anwendung nicht notwendig ist.



## Sonderfälle

Sonderfall	Beschreibung
Auswirkung eines Kaltstarts (%S0=TRUE)	<ul style="list-style-type: none"> <li>• Die Achse wechselt in den Zustand <i>Disabled</i>.</li> <li>• Die PTO-Funktionsbausteine werden initialisiert.</li> </ul>
Auswirkung eines Warmstarts (%S1=TRUE)	<ul style="list-style-type: none"> <li>• Die Achse wechselt in den Zustand <i>Disabled</i>.</li> <li>• Die PTO-Funktionsbausteine werden initialisiert.</li> </ul>
Auswirkung bei Steuerungsstopp	<ul style="list-style-type: none"> <li>• Die Achse wechselt in den Status <i>ErrorStop</i>.</li> <li>• Die Ausgänge werden auf 0 zurückgesetzt.</li> </ul>
Auswirkung einer Online-Änderung	Keine

## Rampen Hochlaufzeit/Verzögerungszeit

### Startgeschwindigkeit

Die **Startgeschwindigkeit** ist die minimale Frequenz, bei der ein belasteter Schrittmotor anfahren kann, ohne Arbeitsschritte zu verlieren.

Der Parameter **Startgeschwindigkeit** wird verwendet, wenn eine Bewegung von der Geschwindigkeit 0 startet.

Die **Startgeschwindigkeit** muss im Bereich  $0 \dots \text{MaxVelocityAppl}$  liegen.

Der Wert 0 bedeutet, dass der Parameter **Startgeschwindigkeit** nicht verwendet wird. In diesem Fall startet die Bewegung mit einer Geschwindigkeit = Beschleunigungsrate x 1 ms.

### Stoppgeschwindigkeit

Die **Stoppgeschwindigkeit** ist die maximale Frequenz, bei der ein belasteter Schrittmotor stoppen kann, ohne Arbeitsschritte zu verlieren.

Die **Stoppgeschwindigkeit** wird nur verwendet, wenn von einer höheren Geschwindigkeit als der **Stoppgeschwindigkeit** auf die Geschwindigkeit 0 heruntergebremst wird.

Die **Stoppgeschwindigkeit** muss im Bereich  $0 \dots \text{MaxVelocityAppl}$  liegen.

Der Wert 0 bedeutet, dass der Parameter **Stoppgeschwindigkeit** nicht verwendet wird. In diesem Fall stoppt die Bewegung mit einer Geschwindigkeit = Verzögerungsrate x 1 ms.

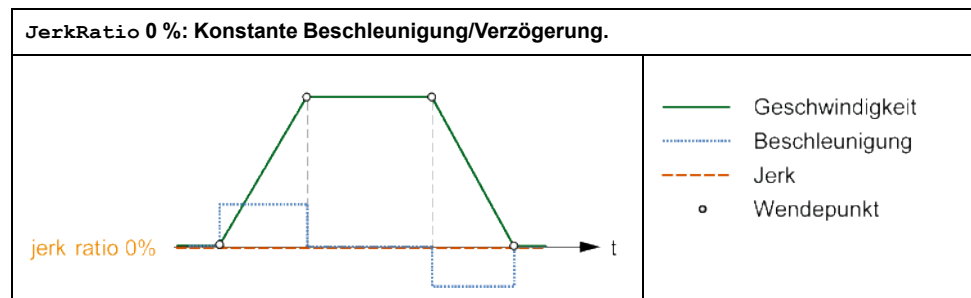
## Beschleunigung/Verzögerung

Beschleunigung ist die Rate der Geschwindigkeitsänderung, von der **Startgeschwindigkeit** bis zur Zielgeschwindigkeit. Verzögerung ist die Rate der Geschwindigkeitsänderung, von der **Zielgeschwindigkeit** bis zur Stoppgeschwindigkeit. Diese Geschwindigkeitsänderungen werden implizit von der *PTO*-Funktion in Übereinstimmung mit den Parametern *Acceleration*, *Deceleration* und *JerkRatio* gemäß eines **trapezförmigen** oder **S-Kurvenprofils** verwaltet.

## Beschleunigungs-/Verzögerungsrampe mit trapezförmigem Profil

Wenn der Parameter *JerkRatio* auf 0 gesetzt ist, hat die Beschleunigungs-/Verzögerungsrampe ein trapezförmiges Profil.

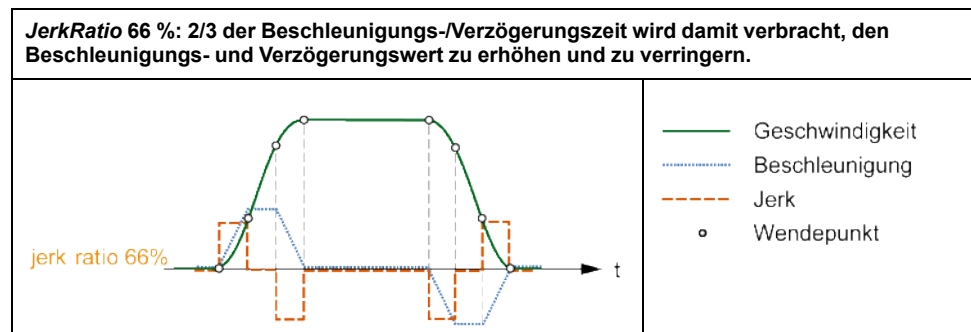
Die Parameter *Acceleration* und *Deceleration* verkörpern die Rate der Geschwindigkeitsänderung, ausgedrückt in Hz/ms.

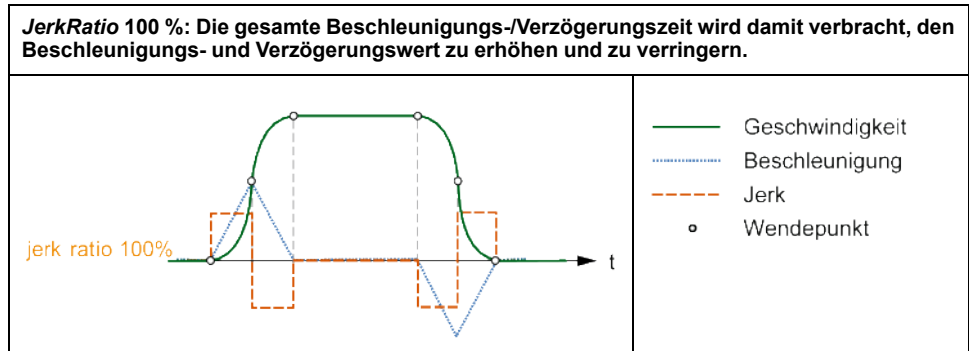


## Beschleunigungs-/Verzögerungsrampe mit S-kurvenförmigem Profil

Wenn der Parameter *JerkRatio* größer als 0 ist, hat die Beschleunigungs-/Verzögerungsrampe ein S-kurvenförmiges Profil.

Die S-kurvenförmige Rampe wird in Anwendungen verwendet, die hohe Trägheit steuern oder in denen fragile Objekte oder Flüssigkeiten gehandhabt werden. Die S-kurvenförmige Rampe ermöglicht eine gleichmäßigere und progressive Beschleunigung/Verzögerung, wie in den folgenden Abbildungen veranschaulicht wird:





**HINWEIS:** Der Wert des *JerkRatio*-Parameters gilt sowohl für Beschleunigung als auch Verzögerung, sodass die konkave und konvexe Zeit gleich sind.

## Auswirkung der S-kurvenförmigen Rampe auf Beschleunigung/Verzögerung

Die Dauer für die Beschleunigung/Verzögerung wird beibehalten, unabhängig vom jeweiligen *JerkRatio*-Parameter. Um diese Dauer aufrechtzuerhalten, unterscheiden sich Beschleunigung oder Verzögerung von den im Funktionsbaustein konfigurierten Werten (Parameter *Acceleration* oder *Deceleration*).

Bei Anwendung von *JerkRatio* ist die Beschleunigung/Verzögerung betroffen.

Wenn das *JerkRatio* mit 100 % angewendet wird, ist die Beschleunigung/Verzögerung doppelt so groß wie in den konfigurierten *Acceleration/Deceleration*-Parametern.

**HINWEIS:** Wenn der *JerkRatio*-Parameterwert ungültig ist, wird der Wert unter Berücksichtigung der Parameter *MaxAccelerationAppl* und *MaxDecelerationAppl* neu berechnet.

*JerkRatio* ist ungültig, wenn:

- sein Wert 100 überschreitet. In diesem Fall wird eine *Jerkratio* von 100 angewandt.
- sein Wert 0 unterschreitet. In diesem Fall wird eine *Jerkratio* von 0 angewandt.

## Sondenereignis

### Beschreibung

Der *Probe*-Eingang wird über die Konfiguration aktiviert und über den Funktionsbaustein *MC\_TouchProbe\_PTO* aktiviert.

Der *Probe*-Eingang wird als Ereignis verwendet, um folgende Aufgaben auszuführen:

- Erfassen der Position
- Starten einer Bewegung unabhängig von der Task

Beide Funktionen können gleichzeitig aktiv sein, d. h. dasselbe Ereignis erfasst die Position und startet einen Bewegungsfunktionsbaustein, Seite 80.

**HINWEIS:** Nur das erste Ereignis nach einer steigenden Flanke am *Busy*-Ausgang des Funktionsbausteins *MC\_TouchProbe\_PTO* ist gültig. Wenn der *Done*-Ausgang auf TRUE gesetzt ist, werden die nachfolgenden Ereignisse ignoriert. Der Funktionsbaustein muss neu aktiviert werden, um auf andere Ereignisse reagieren zu können.

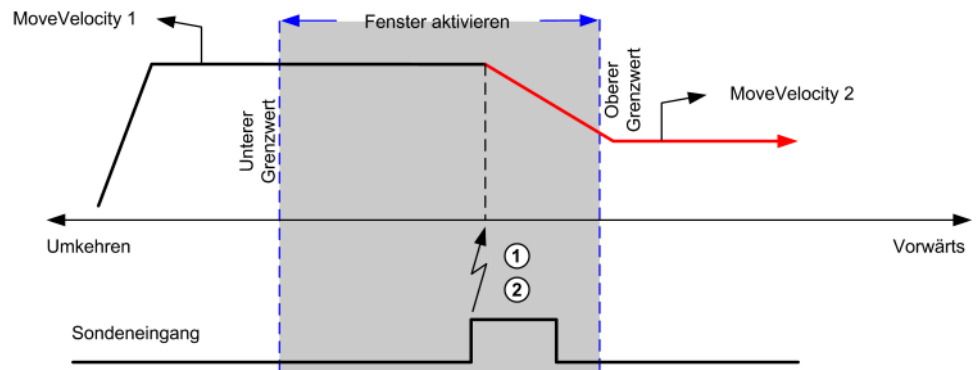
## Positionserfassung

Die erfasste Position ist in `%MC_TouchProbe_PTO.RecordedPos` verfügbar.

## Bewegungsauslöser

Der BufferMode-Eingang von Bewegungs-Funktionsblöcken muss auf `seTrigger` gesetzt sein.

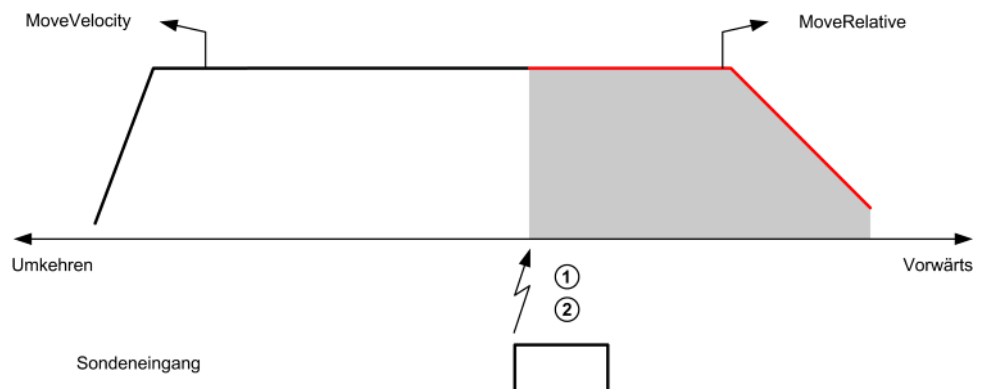
Dieses Beispiel veranschaulicht eine Änderung der Zielgeschwindigkeit unter Verwendung eines Aktivierungsfensters:



1 Wert des Positionszählers erfassen

2 Auslösen des Funktionsbausteins *Move Velocity*

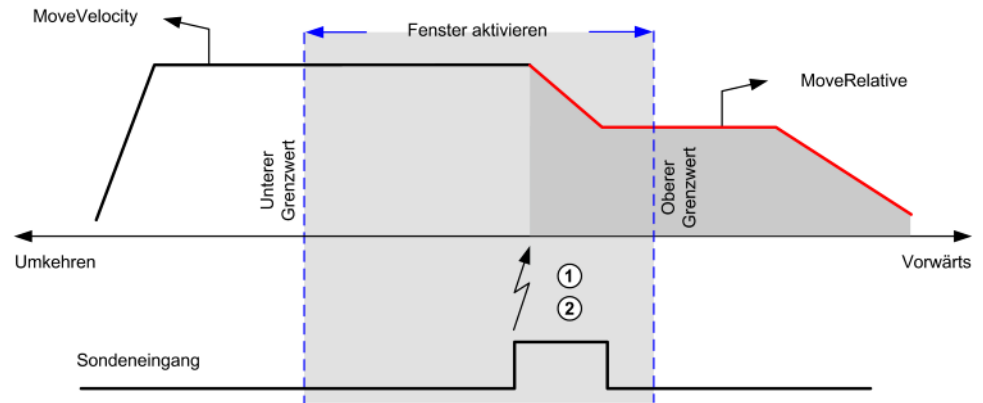
Dieses Beispiel illustriert eine Bewegung zu einer vorprogrammierten Entfernung, mit einem einfachen Profil und ohne ein Aktivierungsfenster zu verwenden:



1 Wert des Positionszählers erfassen

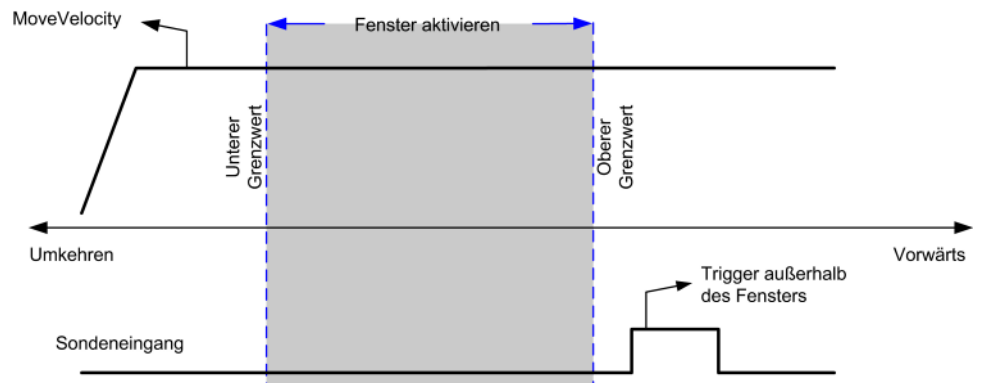
2 Auslösen des Funktionsbausteins *Move Relative*

Dieses Beispiel illustriert eine Bewegung zu einer vorprogrammierten Entfernung, mit einem komplexen Profil und unter Verwendung eines Aktivierungsfensters:



- 1 Wert des Positionszählers erfassen
- 2 Auslösen des Funktionsbausteins *Move Relative*

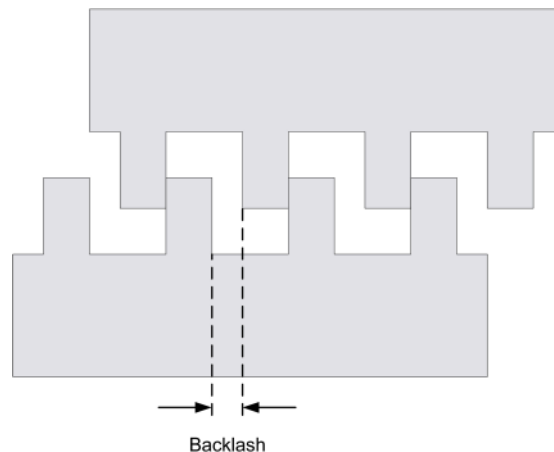
Dieses Beispiel illustriert ein auslösendes Ereignis, das innerhalb eines Aktivierungsfensters eintritt:



## Spielausgleich

### Beschreibung

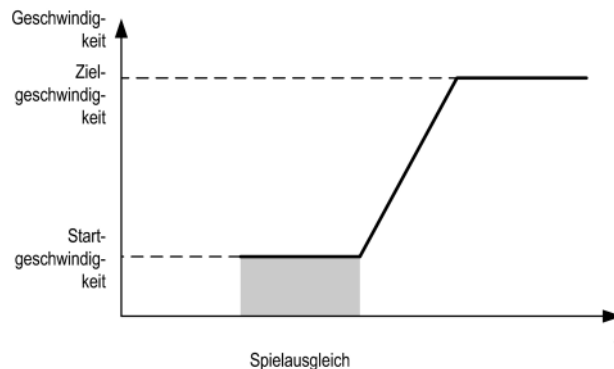
Der Parameter *Backlash Compensation* ist als der Bewegungsbetrag definiert, um den mechanischen Abstand der Geräte (Spielausgleich) zu kompensieren, wenn die Bewegung umgekehrt wird:



**HINWEIS:** Die Funktion berücksichtigt keine externen Bewegungsauslöser, wie z. B. Trägheitsbewegungen oder andere Formen der induzierten Bewegung.

Der Spieldausgleich wird als Impulsanzahl angegeben (0...65535, Standardwert ist 0). Wenn dieser Parameter gesetzt ist, wird bei jeder Richtungsumkehrung zunächst die angegebene Anzahl an Impulsen bei Startgeschwindigkeit ausgegeben und anschließend wird die programmierte Bewegung ausgeführt. Die Impulse für den Spieldausgleich werden dem Positionszähler nicht hinzugefügt.

Die folgende Abbildung stellt den Spieldausgleich dar:



#### HINWEIS:

- Vor dem Start der ersten Bewegung ist der Funktion nicht bekannt, wie groß das auszugleichende Spiel ist. Deswegen ist ein Spieldausgleich nur dann aktiv, wenn bereits eine erste Bewegung ausgeführt wurde, und der Ausgleich wird bei der ersten Richtungsumkehrung angewendet.
- Wenn vor dem Abschluss des Nachschwingens ein Abbruchbefehl empfangen oder ein Fehler erkannt wurde, bleibt die absolute Position unverändert.
- Nach einem Abbruchbefehl fährt das Nachschwingen von der aktuellen Nachschwingposition fort, wenn eine neue Bewegung gestartet wird.

Detaillierte Informationen hierzu finden Sie in Konfigurieren von Impulsfolgenausgabe (siehe Modicon M221, Logic Controller, Programmierhandbuch).

## Positionierungsgrenzen

### Einführung

Es können positive und negative Grenzwerte gesetzt werden, um die Bewegungsgrenzen in beiden Richtungen zu kontrollieren. Sowohl Hardware- als auch Softwaregrenzen werden von der Steuerung verwaltet.

Hardware- und Software-Grenzwertschalter werden ausschließlich eingesetzt, um Grenzwerte in der Steuerungsanwendung zu verwalten. Sie ersetzen keinesfalls mit dem Antrieb verdrahtete funktionale Sicherheitsenschalter. Die Grenzwertschalter der Steuerungsanwendung müssen zwangsläufig vor der Verdrahtung der funktionalen Sicherheitsenschalter mit dem Antrieb aktiviert werden. Der Typ der von Ihnen implementierten funktionalen Sicherheitsarchitektur sprengt die Grenzen des vorliegenden Dokuments. Er ist von Ihrer Sicherheitsanalyse abhängig, einschließlich, jedoch nicht beschränkt auf:

- Risikoanalyse nach EN/ISO 12100
- FMEA nach EN 60812

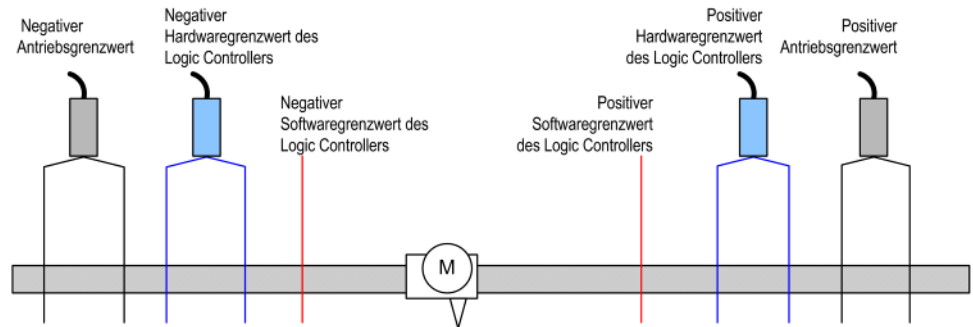
## ▲ WARNUNG

### 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.**

Die folgende Abbildung stellt Hardware- und Software-Grenzwertschalter dar:

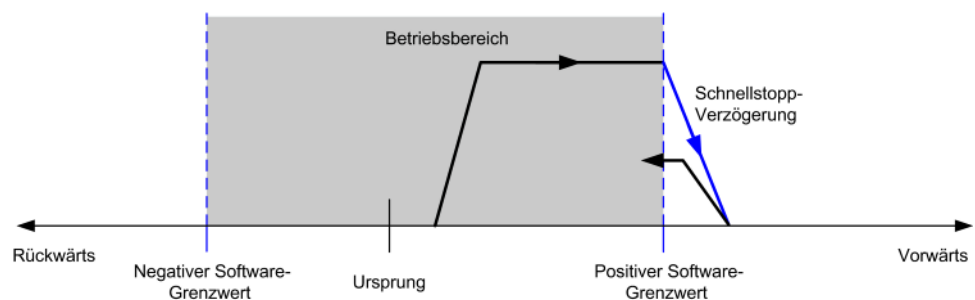


Sobald die negativen oder positiven Software-Grenzwerte der Steuerung überschritten werden, wird ein Fehler erkannt und eine Schnellstopp-Verzögerung durchgeführt.

- Die Achse wechselt in den Zustand *ErrorStop*, mit *AxisErrorId* 1002 bis 1005. Weitere Informationen finden Sie unter *MC\_ReadAxisError\_PTO*, Seite 147 und *PTO-Achsenfehlercodes*, Seite 110.
- Die aktuelle Richtung wird ungültig und der entsprechende PTO-Parameter *EnableDirPos* (1004) oder *EnableDirNeg* (1005) wird vom System auf 0 zurückgesetzt.
- Der gerade ausgeführte Funktionsbaustein erkennt den Fehlerzustand.
- Auf anderen betroffenen Funktionsbausteinen werden die *CmdAborted*-Ausgänge auf TRUE gesetzt.

Um den Achsenfehlerzustand zu löschen und in den Zustand *Standstill* zurückzukehren, ist die Ausführung von *MC\_Reset\_PTO* erforderlich, da jeder Bewegungsbefehl zurückgewiesen wird (siehe PTO-Parameter, Seite 109 *EnableDirPos* oder *EnableDirNeg*), während die Achse außerhalb der Grenzwerte bleibt (der Funktionsbaustein wird mit *ErrorId = InvalidDirectionValue* beendet). Unter diesen Umständen kann ein Bewegungsbefehl nur in der umgekehrten Richtung ausgeführt werden.

Sobald sich die Achse innerhalb der Grenzwerte befindet werden die Parameter *EnableDirPos* oder *EnableDirNeg* vom System wieder auf 1 (gültig) zurückgesetzt.



**HINWEIS:** Im vorherigen Diagramm ist das Zurücksetzen der Achse innerhalb der Grenzwerte das Ergebnis der Ausführung von *MC\_Reset\_PTO* (die Ausführung erfolgt nicht automatisch).

## Software-Grenzwerte

Es können Software-Grenzwerte (Software-Endschalter) gesetzt werden, um die Grenzen der Bewegung in beiden Richtungen zu steuern.

Grenzwerte werden über den Konfigurationsbildschirm aktiviert und unter Beachtung der folgenden Bedingungen festgelegt:

- Positiver Grenzwert > negativer Grenzwert
- Wert im Bereich -2.147.483.648...2.147.483.647

Sie können auch im Anwendungsprogramm aktiviert, deaktiviert und geändert werden (*MC\_WritePar\_PTO* und PTO-Parameter, Seite 109).

**HINWEIS:** Wenn die Software-Grenzwerte aktiviert sind, werden sie nach einem erfolgreichen ersten Homing-Prozess gültig (d. h. die Achse wird mit *MC\_Home\_PTO* an den Ursprung zurückgesetzt).

## Hardware-Grenzwerte

Hardware-Grenzwerte sind für den Homing-Prozess erforderlich sowie um eine Beschädigung der Maschine zu vermeiden. Es müssen die geeigneten Eingänge an den Eingängen *%MC\_Power\_PTO.LimP* und *%MC\_Power\_PTO.LimN* verwendet werden. Die Hardware-Begrenzungsvorrichtungen müssen vom Typ Öffner sein, sodass der Eingang am Funktionsbaustein den Wert FALSE annimmt, sobald die entsprechende Grenze erreicht wird.

**HINWEIS:** Die Bewegungsbeschränkungen gelten, während die Grenzeingänge FALSE sind. Dies gilt unabhängig von der Richtung. Wenn sie wieder TRUE sind, werden die Bewegungsbeschränkungen entfernt, und die Hardware-Grenzwerte werden wiederhergestellt. Deshalb müssen Sie vor dem Funktionsbaustein fallende Flanken als Kontakte verwenden, die zu RESET-Ausgangsweisungen führen. Anschließend können Sie mit diesen Bits die Funktionsbausteineingänge steuern. Wenn alle Vorgänge abgeschlossen sind, setzen Sie die Bits so, dass der normale Betrieb wiederhergestellt wird.

### ⚠ WARNUNG

#### UNBEABSICHTIGTER GERÄTEBETRIEB

- Stellen Sie sicher, dass die Positionsschalter der Steuerungshardware in die Konzeption und Logik Ihrer Anwendung integriert sind.
- Montieren Sie die Positionsschalter der Steuerungshardware in einer Position, die einen angemessenen Bremsabstand ermöglicht.

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

**HINWEIS:** Ein angemessener Bremsabstand ist von der Höchstgeschwindigkeit, der Höchstbelastung (Masse) der zu bewegenden Geräte und dem Schnellhalt-Parameter abhängig.

## Konfiguration

### Überblick

In diesem Abschnitt wird die Vorgehensweise beim Konfigurieren eines PTO-Kanals und der zugehörigen Parameter beschrieben.

## PTO-Konfiguration

### Überblick

Informationen zur Konfiguration der *Pulse Generator*-Ressource finden Sie im Modicon M221 Logic Controller Programmierhandbuch unter Konfigurieren des Impulsgenerators (siehe Modicon M221, Logic Controller, Programmierhandbuch).

Informationen zur Konfiguration der *Pulse Generator*-Ressource als PTO finden Sie im Modicon M221 Logic Controller Programmierhandbuch unter PTO-Konfiguration (siehe Modicon M221, Logic Controller, Programmierhandbuch).

## Motion Task Table

### Übersicht

Die Motion Task Table ist eine Programmiermöglichkeit für Bewegungsfunktionsbausteine, die für repetitive Bewegungsabfolgen gedacht ist. Eine Bewegungsabfolge wird für eine Achse bei der Konfigurationszeit definiert (eine Abfolge kann mit einem Rezept verglichen werden, das verschiedene Bewegungen mischt.)

Die Motion Task Table kann verschiedenen Achsen zugewiesen sein und bietet eine grafische Übersicht der konfigurierten Bewegungsabfolge.

Verwenden Sie den Funktionsbaustein *MC\_MotionTask\_PTO*, um eine Motion Task Table auszuführen. Wenn die Tabelle durch den *MC\_MotionTask\_PTO*-Funktionsbaustein angerufen wird, muss sie einer bestimmten Achse zugewiesen sein. Die Motion Task Table wird der Achse zugewiesen, die vom Funktionsbaustein *MC\_MotionTask\_PTO* verwendet wird. Verschiedene *MC\_MotionTask\_PTO*-Funktionsbausteine können dieselben %MT Motion Task Table-Instanzen gleichzeitig ausführen.

### Merkmale

Die maximale Anzahl der Motion Task Table (%MT)-Instanzen ist 4.

Eine Motion Task Table enthält eine Sequenz einfacher Achsenbewegungen:

- Eine Sequenz ist eine Abfolge an Schritten.
- Jeder Schritt definiert die Parameter einer Bewegung.
- Jeder Schritt verwendet eine dedizierte Bewegungsfunktionsblockinstanz.

Bewegungen, die in der Motion Task Table verwendet werden können:

- Absolute Bewegung
- Relative Bewegung
- Halt
- Position einstellen
- Bewegungsgeschwindigkeit

## Konfigurieren einer Motion Task Table

Der **Motion Task Table Assistant** (Hilfe für die Bewegungstask-Tabelle) ermöglicht Ihnen die Konfigurierung jeder Bewegung in einer geordneten Sequenz und visualisiert ein geschätztes globales Bewegungsprofil.

Zur Anzeige des **Motion Task Table Assistant** gehen Sie wie folgt vor:

Schritt	Aktion																														
1	<p>Wählen Sie die Modulregisterkarte <b>Programmieren &gt; Tools</b> aus und klicken Sie im Hardware-Baum auf <b>PTO-Objekte &gt; Bewegungstask-Tabellen</b>, um die Motion Task Table-Eigenschaften anzuzeigen.</p> <p><b>tooltipMotionTaskTabelleneigenschaften</b></p> <table border="1"> <thead> <tr> <th></th> <th>Konfiguriert</th> <th>Adresse</th> <th>Symbol</th> <th>Konfiguration</th> <th>Kommentar</th> </tr> </thead> <tbody> <tr> <td></td> <td><input type="checkbox"/></td> <td>%MT0</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td><input type="checkbox"/></td> <td>%MT1</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td><input type="checkbox"/></td> <td>%MT2</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td><input type="checkbox"/></td> <td>%MT3</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		Konfiguriert	Adresse	Symbol	Konfiguration	Kommentar		<input type="checkbox"/>	%MT0					<input type="checkbox"/>	%MT1					<input type="checkbox"/>	%MT2					<input type="checkbox"/>	%MT3			
	Konfiguriert	Adresse	Symbol	Konfiguration	Kommentar																										
	<input type="checkbox"/>	%MT0																													
	<input type="checkbox"/>	%MT1																													
	<input type="checkbox"/>	%MT2																													
	<input type="checkbox"/>	%MT3																													
2	Klicken Sie auf zum Konfigurieren der Motion Task Table auf [...].																														

Beschreibung des Fensters Motion Task Table-Eigenschaften:

Parameter	Veränderbar	Wert	Standardwert	Beschreibung
<b>Konfiguriert</b>	Nein	True/False	False	Zeigt an, ob Motion Task Table konfigurierte Schritte enthält.
<b>Adresse</b>	Nein	%MTx	%MTx	Zeigt die Adresse von Motion Task Table an, wobei x die Tabellenummer ist.
<b>Symbol</b>	Ja	–	–	Hiermit kann ein Symbol angegeben werden, das mit dem Motion Task Table verknüpft werden soll.  Doppelklicken Sie auf die Zelle, um das Feld zu bearbeiten.
<b>Konfiguration</b>	Ja	[...] (Schaltfläche)	Aktiviert	Hiermit kann mithilfe des <b>Motion Task Table Assistant</b> eine Sequenz an Bewegungen konfiguriert werden.
<b>Kommentar</b>	Ja	–	–	Hiermit kann ein Befehl angegeben werden, das mit dem Motion Task Table verknüpft werden soll.  Doppelklicken Sie auf die Zelle, um das Feld zu bearbeiten.

**Motion Task Table Assistant:**

Motion Task Tabellenassistent
✖

**Schritte**

Schritt	Typ	Pos	Distanz	Geschw.	Beschl.	Verz.	Ruckanteil	Nächster Schritt	Ereignis	Verzögerung	Softwareobjekt
1	MC_MoveAbs_P	2000		5000	20	50	0	Fertig		10	%MC_MOVEA
2	MC_MoveRel_P*		5000	7500	20	100	0	Fertig		0	%MC_MOVER
3	MC_MoveRel_P*		5000	4000	20	200	0	SW Ereignis	%M1	1000	%MC_MOVER
4	MC_Halt_PTO						1	Fertig		0	%MC_HAL_PT
5	Kein										
6	Kein										

PROBE-Ereignisbereich verwenden
 Erste Position 
 Letzte Position

**Bewegungsübersicht**

Die nachstehende Abbildung gibt nicht unbedingt reale Ereignisse wieder. Weitere Informationen finden Sie in der produktspezifischen Dokumentation.

Übernehmen

Abbrechen

**Hauptbereiche des Motion Task Table Assistant:**

Parameter	Beschreibung
<b>Schritte</b>	Listet die Sequenz von einfachen Achsenbewegungen und Eingangsparametern für jede Bewegung auf.
<b>Bewegungsübersicht</b>	<p>Klicken Sie auf die Schaltfläche „Aktualisieren“ oder auf <b>F5</b>, um eine grafische Ansicht der Bewegungen zu generieren, die in der Schrittsequenz implementiert wurden.</p> <p>Die Kurve bietet eine allgemeine Übersicht der Bewegung. Die Kurve basiert auf den folgenden Annahmen:</p> <ul style="list-style-type: none"> <li>Die Initialposition ist 0.</li> <li>Positionsgrenzen sind nicht aktiviert.</li> <li>Es werden standardmäßige Bewegungskonfigurationsparameter verwendet.</li> <li>Ein Ereignis (Sondeneingang, POU) tritt auf nachdem der Schritt abgeschlossen ist und mit einer 100 ms Verzögerung.</li> <li>Eine %MWx-Verzögerung wird grafisch durch eine 100 ms-Verzögerung dargestellt.</li> </ul>

Beschreibung des Fensters **Schritte**:

Parameter	Wert	Standardwert	Beschreibung
<b>Schritt</b>	1...16	–	Nummer der einfachen Achsenbewegung in der Sequenz.
<b>Typ</b>	<b>Keine</b> <b>MC_MoveAbs_PTO</b> (Absolute Bewegung) <b>MC_MoveRel_PTO</b> (Relative Bewegung) <b>MC_Halt_PTO</b> (Halt) <b>MC_SetPos_PTO</b> (Sollposition) <b>MC_MoveVel_PTO</b> (Bewegungsgeschwindigkeit)	<b>Keine</b>	Bewegungsbehl. Der Bewegungsbehl verwendet eine Bewegungsfunktionsbausteininstanz, die im Parameter <b>Software-Objekte</b> angegeben ist.
<b>Pos</b>	Siehe jeden Parameterwert eines Software-Objekt-Funktionsbausteins.	<i>Leer</i>	Die Bewegungsparameter sind die Parameter des Software-Objekts, das dem Schritt zugewiesen ist. Parameterbeschreibung: <ul style="list-style-type: none"> <li>• <b>Pos</b>: Position</li> <li>• <b>Abstand</b>: Distanz</li> <li>• <b>Geschw.</b>: Geschwindigkeit</li> <li>• <b>Beschl.</b>: Beschleunigung</li> <li>• <b>Verz.</b>: Verzögerung</li> <li>• <b>Ruckanteil</b>: Ruckanteil</li> </ul> <b>HINWEIS:</b> <i>Geschw.</i> -Parameter für den Bewegungsbehl der Bewegungsgeschwindigkeit ist eine Kombination von Geschwindigkeit und Richtung. In der Tabelle ist der Geschwindigkeitsbereich für den <i>MC_MoveVel_PTO</i> -Bewegungsbehl: - Max. Geschwindigkeit... + Max. Geschwindigkeit. Eine negative Geschwindigkeit zeigt eine negative Richtung an. Eine positive Geschwindigkeit zeigt eine positive Richtung an.
<b>Abstand</b>			
<b>Geschw.</b>			
<b>Beschl.</b>			
<b>Dez.</b>			
<b>Ruckanteil</b>			

Parameter	Wert	Standardwert	Beschreibung
<b>Nächster Schritt</b>	<b>Done / In velocity</b> <b>Blending previous</b> <b>Probe input event</b> <b>%M event</b> <b>Verzögerung</b>	<i>Leer</i>	<p>Die Bedingung, die erfüllt werden muss, um zum nächsten Schritt in der Tabellensequenz zu gehen.</p> <p>Beschreibung der Bedingung:</p> <ul style="list-style-type: none"> <li>• <b>Done / In velocity:</b> <ul style="list-style-type: none"> <li>◦ <b>Done:</b> Mit dem nächsten Schritt fortfahren, wenn der aktuelle Schritt abgeschlossen ist. Dieser Parameter ist für verschiedene Bewegungsbefehle außer der Bewegungsgeschwindigkeit verfügbar.</li> <li>◦ <b>In velocity:</b> Mit dem nächsten Schritt fortfahren, wenn die angefragte Geschwindigkeit erreicht wurde. Dieser Parameter ist nur für den Bewegungsbefehl der Bewegungsgeschwindigkeit verfügbar.</li> </ul> </li> <li>• <b>Blending previous:</b> Die Geschwindigkeit des nächsten Schritts überschneidet sich mit der Geschwindigkeit der Endposition dieses Schritts.</li> <li>• <b>Probe input event:</b> Der Vorgang wird mit dem nächsten Schritt fortgesetzt, wenn ein definiertes Ereignis am Sondereingang festgestellt wird. Die Flanke wird im Parameter <b>Ereignis</b> definiert. Ein Eingangsfeld öffnet sich am unteren Ende des Fensters <b>Schritte: Sondereignisbereich verwenden</b>, wie in der nächsten Tabelle beschrieben. <b>HINWEIS:</b> Es kann nur eine Instanz des <b>Sondereingangereignisses</b> pro Motion Task Table verwendet werden.</li> <li>• <b>%M event:</b> Fahren Sie mit dem nächsten Schritt fortfahren, wenn die Speicher-Bit-Adresse <b>%Mx</b> im Parameter <b>Event</b> („Ereignis“) auf 1 eingestellt ist.</li> <li>• <b>Delay:</b> Mit dem nächsten Schritt fortfahren, wenn die Verzögerung (startet am Anfang des Schritts) abläuft. Die Verzögerung wird im Parameter <b>Verzögerung</b> definiert. <b>HINWEIS:</b> Wenn das Ereignis <b>Probe input event</b>, <b>%M event</b> oder <b>Delay</b> auftritt, wird der nächste Schritt gestartet, selbst wenn der aktuelle Schritt noch nicht abgeschlossen ist.</li> </ul>
<b>Ereignis</b>	– <b>0/1</b> <b>%Mx</b>	<i>Leer</i>	<p>Der <b>Ereignis</b>-Wert ergänzt die Bedingungen, die im Parameter <b>Nächster Schritt</b> beschrieben sind.</p> <p>Die Auswahl <b>Nächster Schritt</b> und die entsprechende <b>Ereignis</b>-Auswahl:</p> <ul style="list-style-type: none"> <li>• <b>Probe input event:</b> <ul style="list-style-type: none"> <li>◦ 0: Fallende Flanke</li> <li>◦ 1: Steigende Flanke <b>HINWEIS:</b> Das Sondereingangereignis ist unabhängig vom Anwendungstask-Zyklus und dem Bewegungstask-Zyklus.</li> </ul> </li> <li>• <b>%M event:</b> Speicherbit <b>%Mx</b>. <b>HINWEIS:</b> <b>%Mx</b> wird alle 4 ms evaluiert.</li> </ul>

Parameter	Wert	Standardwert	Beschreibung
<b>Verzögerung</b>	0...65535  %MWx	Leer	<p>Der Wert <b>Verzögerung</b> stellt die Zeit dar, die verstreicht, bevor mit dem nächsten Schritt fortgefahren wird. Abhängig vom Parameterwert <b>Nächster Schritt</b> wird die <b>Verzögerung</b> vom Start oder Ende des Schritts evaluiert:</p> <ul style="list-style-type: none"> <li>• <b>Done / In velocity</b>: Die Verzögerung beginnt, wenn der aktuelle Schritt <b>Done</b> oder <b>In velocity</b> ist.</li> <li>• <b>Blending previous</b>: Nicht verfügbar</li> <li>• <b>Probe input event</b> und <b>%M event</b>: Die Verzögerung startet am Anfang des Schritts. <ul style="list-style-type: none"> <li>◦ Eine angelaufene Verzögerung generiert eine Zeitüberschreitung, wenn das Ereignis nicht auftrat und der nächste Schritt wird gestartet.</li> <li>◦ Wenn das Ereignis vor dem Ende der Verzögerung auftritt, wird der nächste Schritt gestartet und die Zeitüberschreitung der Verzögerung wird abgebrochen.</li> </ul> </li> </ul> <p><b>HINWEIS:</b> Wenn die <b>Verzögerung</b> auf ihrem Standardwert (0) bleibt, wartet der Bewegungsbefehl auf das Auftreten eines Sondeneingangs- oder Software-Ereignisses ohne Zeitüberschreitung.</p> <ul style="list-style-type: none"> <li>• <b>Delay</b>: Die Verzögerung startet am Anfang des Schritts. Der nächste Schritt wird gestartet, wenn die Verzögerung abgelaufen ist.</li> </ul> <p><b>HINWEIS:</b> Ein unmittelbarer Wert kann in einer Anwendung POU nicht geändert werden. Ein %MWx-Wert muss jedoch durch die Anwendung POU eingestellt werden. Der Parameter Motion Task Table <b>Verzögerung</b> wird nicht geändert, wenn <i>MC_ReadPar_PTO</i> oder <i>MC_WritePar_PTO</i> für die Verwendung von <i>ParNumber</i> = 1000 (Verzögerung) eingestellt sind.</p>
<b>Softwareobjekte</b>	%MC_MOVEABS_PTOx  %MC_MOVEREL_PTOx  %MC_HALT_PTOx  %MC_SETPOS_PTOx  %MC_MOVEVEL_PTOx	Leer	Zeigt die Softwareobjekte, die dem Schritt zugewiesen sind. Er wird vom System zugewiesen und ist ein schreibgeschützter Parameter. Diese Softwareobjekte sind Funktionsblockinstanzen.
<b>Symbol</b>	–	Leer	<p>Hiermit kann ein Symbol angegeben werden, das mit dem Schritt Softwareobjekt verknüpft werden soll.</p> <p>Doppelklicken Sie auf die Zelle, um das Feld zu bearbeiten.</p>

Der Parameter **Sondeneignisbereich** verwenden im Fenster **Schritte**:

Parameter	Wert	Standardwert	Beschreibung
<b>Sondeneignisbereich verwenden</b>	True/False	False	<p>Wenn TRUE, wird ein Trigger-Ereignis nur innerhalb des Positionsbereiches erkannt, der zwischen der <b>Ersten Position</b> und der <b>Letzten Position</b> definiert ist.</p> <p>Der Parameter kann geändert werden, wenn der <b>Nächste Schritt</b> in der Motion Task Table auf <b>Probe input event</b> eingestellt ist.</p>
<b>Letzte Position</b> <sup>(1)</sup>	- 2147483648... 2147483647  %MDx	2147483647	Oberer Grenzwert des Positionsbereichs.
<b>Erste Position</b> <sup>(1)</sup>	- 2147483648... 2147483647  %MDx	- 2147483648	Unterer Grenzwert des Positionsbereichs.
<p>(1) Der Wert an der <b>ersten Position</b> muss kleiner sein als der Wert an der <b>letzten Position</b>.</p> <p><b>HINWEIS:</b> Eine Illustration der Auswirkung des Positionsbereichs auf die Auslösung wird im Abschnitt zum <b>Sondeneignis</b>, Seite 85 zur Verfügung gestellt.</p> <p><b>HINWEIS:</b> Die Position, an der die Auslösung des Ereignisses festgestellt wurde, wird nicht aufgezeichnet.</p>			

## Verwalten von Schritt-Parametern und Ereignissen

Die Parameter und Ereignisse, die in einem Schritt definiert sind, sind nur am Anfang der Ausführung des Schritts gültig. Deswegen:

- Wenn der Parameterwert eines Schritts durch die Anwendung geändert wurde, ist dies nur gültig, wenn die Änderung vor der Aktivierung des Schritts vorgenommen wurde. Der Parameter kann geändert werden, indem man vom System zugewiesene Softwareobjektparameter in einem POU verwendet.
- Der Wert eines Speicherobjekts (%MW oder %MWx) ist nur gültig, wenn er vor der Aktivierung des Schritts aktualisiert wird.
- Ein Ereignis wird nur evaluiert, wenn der Schritt aktiv ist. Im Falle eines *Probe input event*, kann ein Ereignis, das vor der Aktivierung des Schritts auftrat, nicht erkannt werden.

## Verwalten von in Motion Task Table verwendeten Funktionsblockinstanzen

Dem System zugewiesene Softwareobjektinstanzen:

- können nicht in einer Anwendung POU verwendet werden, um die Achsenbewegung zu kontrollieren.
- Ausgangsparameter werden während der Ausführung von Motion Task Table nicht vom System aktualisiert. Anders gesagt, die Ausgangs-Bits und Ausgangsparameter sind nicht gültig.
- Eingangsparameter:
  - können nicht im Editor für Softwareobjektinstanzen oder in der Registerkarte **Programmieren** geändert werden.
  - können verwendet werden, um die Motion Task Table in einer Anwendung POU dynamisch zu ändern. Um die Eingangsparameter der vom System zugewiesene Softwareobjektinstanzen dynamisch zu ändern, verwenden Sie die Parameteradresse oder das entsprechende Symbol.

**HINWEIS:** Der Ausführungsschritt kann geändert werden, aber diese Änderungen werden erst bei der nächsten Ausführung des Schritts berücksichtigt.

Beispiel einer Bewegung beschrieben in einer Motion Task Table:

- Schritt: 2
- Bewegungstyp: Relative Bewegung
- Softwareobjekt: %MC\_MOVEREL\_PTO1
- Symbol: *Move\_Relative\_Label2*

Im vorigen Beispiel kann der Eingangsparameter der Geschwindigkeit durch das Programm geändert werden, indem eine der folgenden Syntaxen verwendet wird:

- %MC\_MOVEREL\_PTO1.Vel
- *Move\_Relative\_Label2.Vel*

Verwalten der Funktionsbausteininstanzen, die in einer Motion Task Table verwendet werden:

- Wenn eine Motion Task Table konfiguriert wurde, werden die reservierten Funktionsblockinstanzen als **Verwendet** eingestellt.
- Wenn alle Instanzen eines bestimmten Funktionsbausteins reserviert sind, kann der zugehörige Bewegungstyp nicht mehr verwendet werden.

# Programmierung

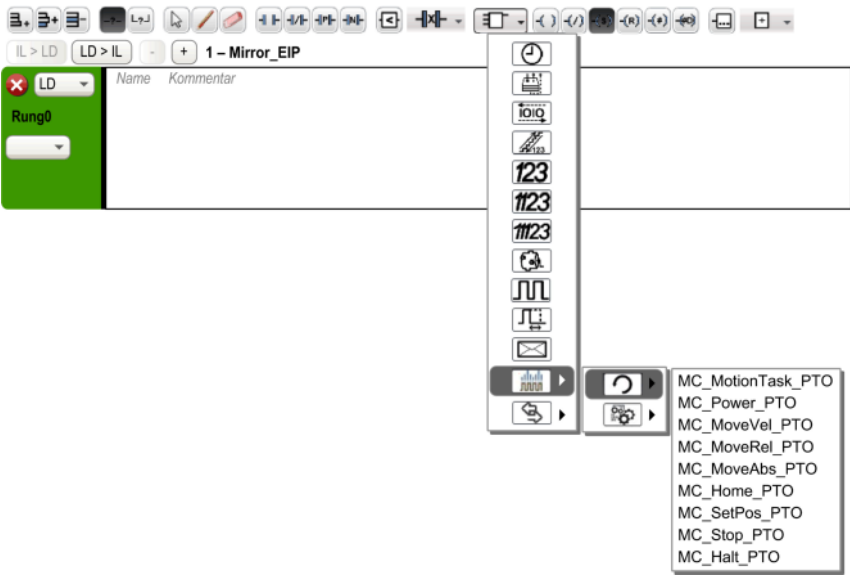
## Überblick

Dieser Abschnitt listet die Funktionsbausteine auf, die zur Programmierung der *PTO*-Funktion verwendet werden und beschreibt das Hinzufügen und Entfernen dieser Funktionsbausteine.

## Hinzufügen oder Entfernen eines Funktionsbausteins

### Hinzufügen eines Funktionsbausteins

Gehen Sie wie folgt vor, um eine Instanz des *PTO*-Funktionsbausteins hinzuzufügen:

Schritt	Aktion
1	Wählen Sie die Registerkarte <b>Programmierung</b> aus.
2	Wählen Sie <b>Funktionsbausteine &gt;&gt; PTOAdministrativ</b> oder <b>Funktionsbausteine &gt; PTO &gt; Bewegung</b> aus, wie in der folgenden Grafik dargestellt: 
3	Klicken Sie in den Programmbaustein, um den ausgewählten Funktionsbaustein zu platzieren.
4	Ordnen Sie die Eingangs- und Ausgangsvariablen des Funktionsbausteins zu.

**HINWEIS:** Legen Sie auf der Registerkarte **Konfiguration** die Parameter fest.

Weitere Details finden Sie im Modicon M221 Logic Controller - Programmierhandbuch im Kapitel *PTO*-Konfiguration.

### Entfernen eines Funktionsbausteins

Gehen Sie wie folgt vor, um eine Instanz des *PTO*-Funktionsbausteins zu entfernen:

Schritt	Aktion
1	Klicken Sie auf der Registerkarte <b>Programmierung</b> auf die Instanz des Funktionsbausteins.
2	Drücken Sie <b>Entf</b> , um den ausgewählten Funktionsbaustein zu löschen.

## PTO-Funktionsbausteine

### Funktionsbausteine

Die PTO-Funktion ist in EcoStruxure Machine Expert - Basic mit den folgenden Funktionsbausteinen programmiert:

Kategorie	Funktionsbaustein	Beschreibung
Bewegung (einachsig), Seite 116	<i>MC_MotionTask_PTO</i> , Seite 116	Ruft einen Motion Task Table an.
	<i>MC_Power_PTO</i> , Seite 119	Versorgt die Achse mit Strom, der Achsenstatus wechselt von <i>Disabled</i> zu <i>Standstill</i> . Wenn das <i>%MC_Power_PTO.Status</i> -Bit <i>FALSE</i> ist, kann für diese Achse kein Bewegungsfunktionsbaustein ausgeführt werden.
	<i>MC_MoveVel_PTO</i> , Seite 122	Diese Funktion veranlasst die angegebene Achse zu einer Bewegung mit der angegebenen Geschwindigkeit und zu einem Übergang in den Zustand <i>Continuous</i> . Die kontinuierliche Bewegung wird beibehalten, bis ein Software-Grenzwert erreicht oder eine abbrechende Bewegung ausgelöst wird oder ein Übergang in den Zustand <i>ErrorStop</i> erkannt wurde.
	<i>MC_MoveRel_PTO</i> , Seite 126	Diese Funktion veranlasst die angegebene Achse zu einer Bewegung mit der angegebenen Geschwindigkeit und über eine inkrementelle Entfernung sowie zu einem Übergang in den Zustand <i>Discrete</i> .  Die Zielposition wird von der aktuellen Position zur Ausführungszeit referenziert und um eine bestimmte Entfernung erhöht.
	<i>MC_MoveAbs_PTO</i> , Seite 129	Diese Funktion veranlasst die angegebene Achse zu einer Bewegung mit der angegebenen Geschwindigkeit bis zu einer bestimmten Position sowie zu einem Übergang in den Zustand <i>Discrete</i> .  Der Funktionsbaustein terminiert mit <i>Error</i> auf <i>TRUE</i> , wenn die Achse nicht genullt wurde (d.h. noch keine absolute Referenzposition definiert ist). In diesem Fall wird <i>ErrorId</i> auf <i>InvalidAbsolute</i> gesetzt.
	<i>MC_Home_PTO</i> , Seite 133	Dieser Funktionsbaustein veranlasst die angegebene Achse, eine Sequenz auszuführen, die eine absolute Referenzposition definiert, und weist die Achse anschließend an, in den Zustand <i>Homing</i> , Seite 101 überzugehen. Die Details dieser Sequenz richten sich nach der Konfigurationseinstellung für den <i>Homing</i> -Vorgang.
	<i>MC_SetPos_PTO</i> , Seite 135	Verändert die Koordinaten der Achse ohne physische Bewegung.
	<i>MC_Stop_PTO</i> , Seite 137	Dieser Funktionsbaustein veranlasst einen kontrollierten Stopp der Bewegung und weist die Achse anschließend an, in den Zustand <i>Stopping</i> überzugehen. Die eventuelle Ausführung von Bewegungen wird abgebrochen.
	<i>MC_Halt_PTO</i> , Seite 139	Dieser Funktionsbaustein veranlasst einen kontrollierten Stopp der Bewegung bis zur Geschwindigkeit 0 und weist die Achse anschließend an, in den Zustand <i>Discrete</i> überzugehen. Der Ausgang <i>Done</i> wird auf <i>TRUE</i> gesetzt, und der Zustand geht in <i>Standstill</i> über.
Administrativ, Seite 141	<i>MC_ReadActVel_PTO</i> , Seite 141	Dieser Funktionsbaustein gibt den Geschwindigkeitswert der Achse zurück.
	<i>MC_ReadActPos_PTO</i> , Seite 143	Dieser Funktionsbaustein gibt den Positionswert der Achse zurück.
	<i>MC_ReadSts_PTO</i> , Seite 144	Dieser Funktionsbaustein gibt den Wert des Zustandsdiagramms, Seite 112 der Achse zurück.
	<i>MC_ReadMotionState_PTO</i> , Seite 146	Dieser Funktionsbaustein gibt den Bewegungszustand der Achse zurück.
	<i>MC_ReadAxisError_PTO</i> , Seite 147	Gibt einen Achsensteuerungsfehler zurück, falls vorhanden.
	<i>MC_Reset_PTO</i> , Seite 149	Setzt alle achsenbezogenen Fehler zurück, falls die Bedingungen dies zulassen, um einen Übergang vom Status <i>ErrorStop</i> in <i>Standstill</i> zu erreichen. Dieser Vorgang hat keinerlei Auswirkungen auf die Instanzen der Funktionsblöcke.
	<i>MC_TouchProbe_PTO</i> , Seite 150	Aktiviert ein auslösendes Ereignis auf dem Probe-Eingang. Das auslösende Ereignis ermöglicht die Aufzeichnung der Achsenposition und das Starten einer gepufferten Bewegung.
	<i>MC_AbortTrigger_PTO</i> , Seite 153	Dieser Funktionsbaustein hält die Ausführung anderer Funktionsbausteine an, wenn diese mit auslösenden Ereignissen verbunden sind (beispielsweise <i>MC_TouchProbe_PTO</i> ).
	<i>MC_ReadPar_PTO</i> , Seite 154	Ruft Parameter vom PTO ab.
	<i>MC_WritePar_PTO</i> , Seite 155	Schreibt Parameter in den PTO.

**HINWEIS:** Die Bewegungsfunktionsbausteine handeln auf der Position der Achse entsprechend dem Bewegungszustandsdiagramm. Administrative Funktionsbausteine haben keine Auswirkungen auf den Bewegungszustand.

**HINWEIS:** Der Funktionsbaustein *MC\_Power\_PTO*, Seite 119 muss aufgerufen werden, damit ein anschließender Bewegungsbefehl ausgegeben werden kann.

## ⚠️ WARNUNG

### UNBEABSICHTIGTER GERÄTEBETRIEB

- Eine Funktionsbausteininstanz darf nicht in verschiedenen Programm-Tasks verwendet werden.
- Ändern Sie die Referenz des Funktionsbausteins (AXIS) nicht, während der Funktionsbaustein ausgeführt wird.

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

## Homing-Modi

### Überblick

In diesem Abschnitt werden die PTO-Homingmodi beschrieben.

## Homing-Modi

### Beschreibung

Homing ist die Methode zum Festlegen des Referenzpunkts oder Ursprungs für eine absolute Bewegung.

Homing-Bewegungen können unter Verwendung verschiedener Methoden durchgeführt werden. Die M221 PTO-Kanäle bieten verschiedene standardmäßige Homing-Bewegungstypen:

- Positionseinstellung, Seite 103
- Ausführliche Referenz, Seite 103
- Kurze Referenz mit Umkehr, Seite 105
- Kurze Referenz ohne Umkehr, Seite 104

Eine Homing-Bewegung muss ohne Unterbrechung beendet werden, damit der neue Referenzpunkt als gültig anerkannt wird.

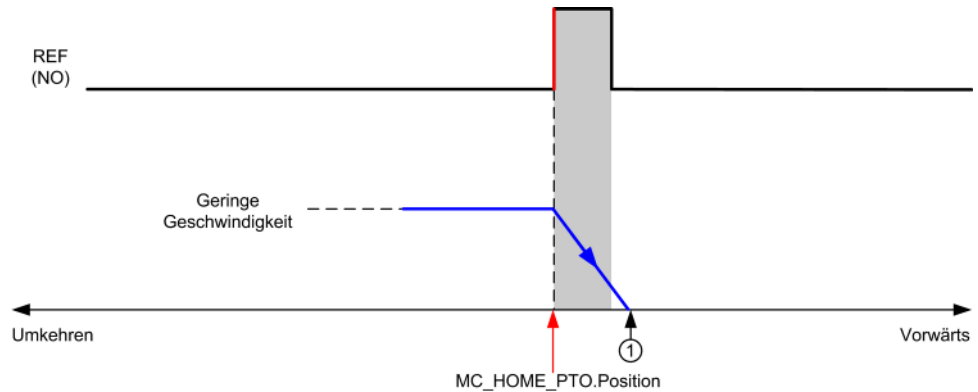
- *%MC\_ReadSts\_PTO.IsHomed* wird auf TRUE gesetzt, wenn eine Homing-Bewegung erfolgreich beendet wurde. Wenn die Homing-Bewegung unterbrochen wird, muss sie erneut durchgeführt werden.
- *%MC\_ReadSts\_PTO.IsHomed* wird auf FALSE gesetzt, wenn der Achsenzustand DISABLED ist oder wenn keine Homing-Bewegung erfolgreich beendet wurde.

Siehe die *MC\_Home\_PTO*, Seite 133 und Objektcodes des Funktionsbausteins Homing-Modi, Seite 108.

## Nullposition

Wenn das Homing (Nullen) anhand eines externen Schalters erfolgt, ist die Homing-Position über die Kante des Schalters definiert. Anschließend wird die Bewegung bis zum Anhalt verzögert.

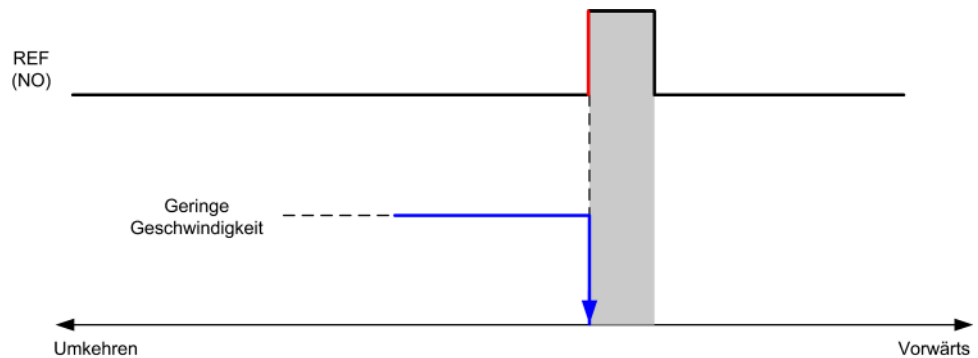
Ist die tatsächliche Position zum Abschluss der Bewegungssequenz weicht daher möglicherweise von dem in dem Funktionsbaustein eingestellten Positionsparameter ab.



**REF (NO)** Referenzpunkt (Normal Offen)

1 Position zum Ende der Bewegung =  $\%MC\_HOME\_PTO.Position + \text{Entfernung „Verzögerung-bis-Stopp“}$ .

Um die Darstellung eines Halts in Homing-Modi-Diagrammen zu vereinfachen, wird die tatsächliche Position der Achse folgendermaßen angegeben:



**REF (NO)** Referenzpunkt (Normal Offen)

## Grenzwerte

Hardwaregrenzen sind für den ordnungsgemäßen Betrieb des Funktionsbausteins *MC\_Home\_PTO* erforderlich (*Positioning Limits*, Seite 88 und *MC\_Power\_PTO*). Je nach Art der im Homing-Modus angeforderten Bewegung stellen die Hardwaregrenzen sicher, dass das Ende der Verfahrbewegung vom Funktionsbaustein eingehalten wird.

Wenn eine Homing-Bewegung in eine Richtung ausgelöst wird, die vom Referenzschalter wegführt, ermöglichen die Hardwaregrenzen Folgendes:

- Sie signalisieren, dass eine Richtungsumkehr erforderlich ist (Bewegung der Achse in Richtung des Referenzschalters), oder:
- Sie signalisieren die Erkennung eines Fehlers, da der Referenzschalter vor Erreichen des Endes der Verfahrstrecke nicht gefunden wurde.

Bei Homing-Bewegungen, die eine Richtungsumkehr zulassen, sobald die Hardwaregrenze erreicht ist, wird die Achse über die konfigurierte Verzögerung gestoppt und in der umgekehrten Richtung wieder in Bewegung gesetzt.

Bei Homing-Bewegungen, die keine Richtungsumkehr zulassen, sobald der Hardware-Grenzwert erreicht ist, wird der Homing-Prozess mit Fehler abgebrochen und die Achse über die Schnellstopp-Verzögerung angehalten.

## ▲ **WARNUNG**

### UNBEABSICHTIGTER GERÄTEBETRIEB

- Stellen Sie sicher, dass die Positionsschalter der Steuerungshardware in die Konzeption und Logik Ihrer Anwendung integriert sind.
- Montieren Sie die Positionsschalter der Steuerungshardware in einer Position, die einen angemessenen Bremsabstand ermöglicht.

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

**HINWEIS:** Ein angemessener Bremsabstand ist von der Höchstgeschwindigkeit, der Höchstbelastung (Masse) der zu bewegenden Geräte und dem Schnellhalt-Parameter abhängig.

## Positionseinstellung

### Beschreibung

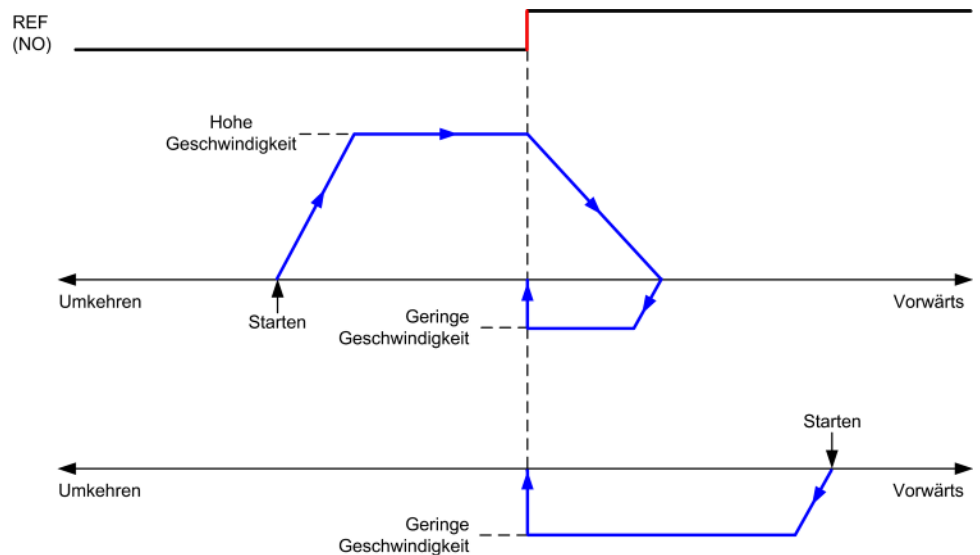
In der aktuellen Positionseinstellung wird die aktuelle Position auf die festgelegte Positionswerte eingestellt. Es wird keine Bewegung ausgeführt.

## Ausführliche Referenz

### Ausführliche Referenz: positive Richtung

Homing erfolgt anhand der fallenden Flanke des Referenzschalters in umgekehrter Richtung.

Die initiale Bewegungsrichtung ist von dem Zustand des Referenzschalters abhängig:

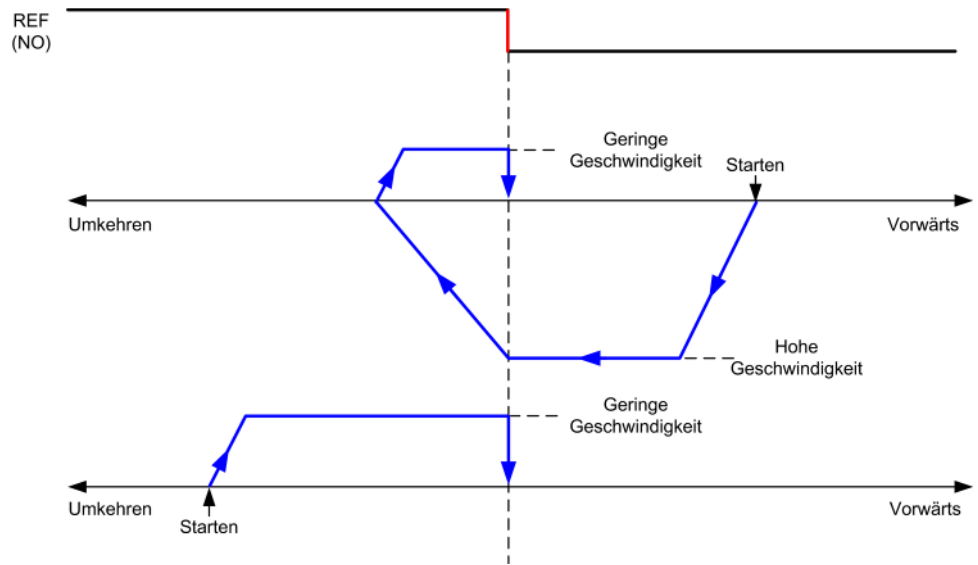


**REF (NO)** Referenzpunkt (Schließer)

### Ausführliche Referenz: negative Richtung

Homing erfolgt anhand der fallenden Flanke des Referenzschalters in Vorwärts-Richtung.

Die initiale Bewegungsrichtung ist von dem Zustand des Referenzschalters abhängig:

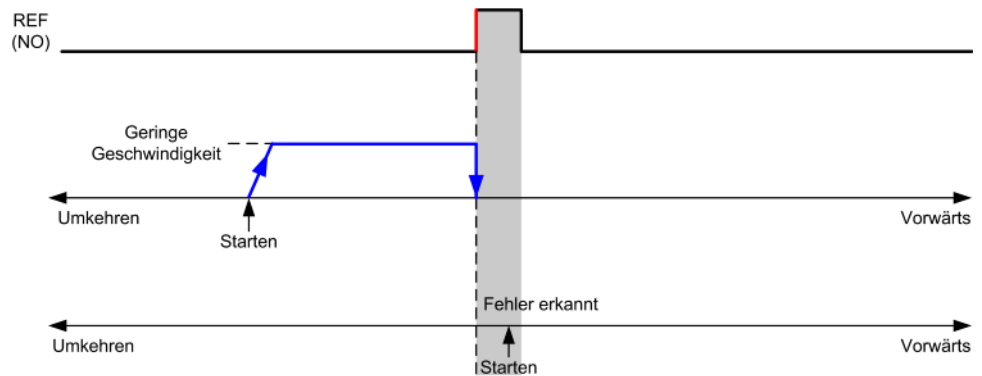


REF (NO) Referenzpunkt (Schließer)

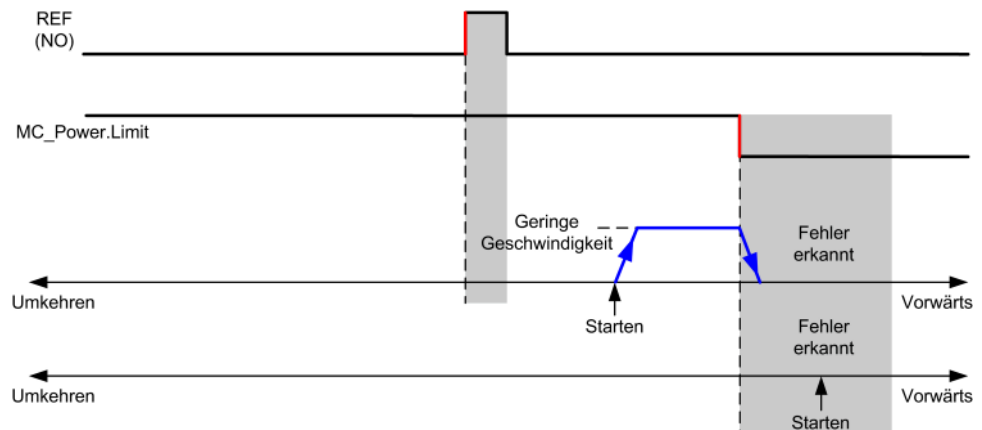
## Kurze Referenz ohne Umkehr

### Kurze Referenz ohne Umkehr: positive Richtung

Homing erfolgt mit geringer Geschwindigkeit anhand der steigenden Flanke des Referenzschalters in Vorwärts-Richtung.



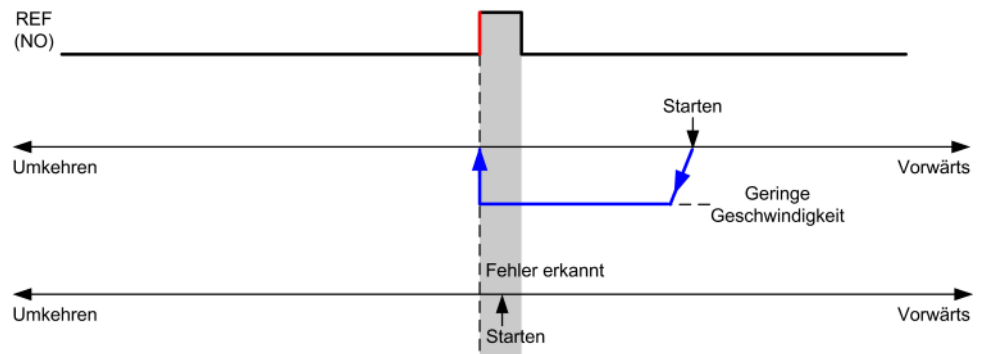
REF (NO) Referenzpunkt (Schließer)



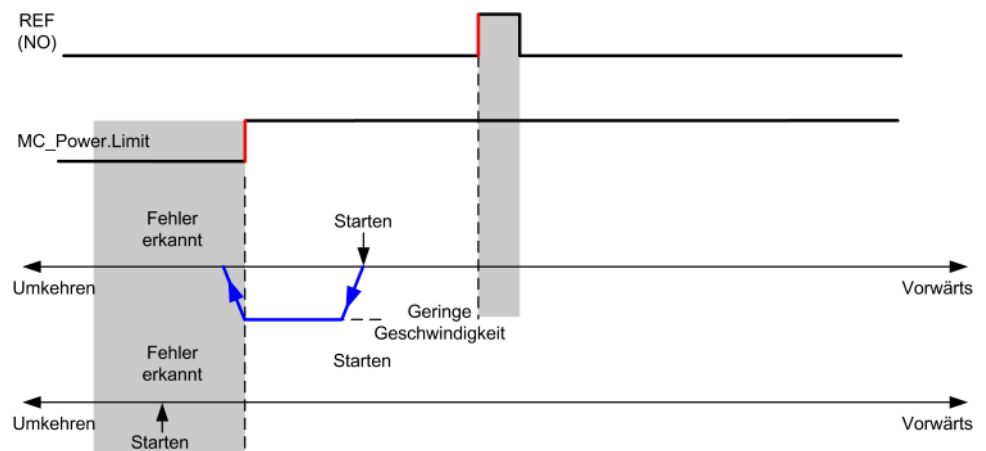
REF (NO) Referenzpunkt (Schließer)

## Kurze Referenz ohne Umkehr: negative Richtung

Homing erfolgt mit geringer Geschwindigkeit anhand der steigenden Flanke des Referenzschalters in Vorwärts-Richtung, ohne Umkehrung.



REF (NO) Referenzpunkt (Schließer)



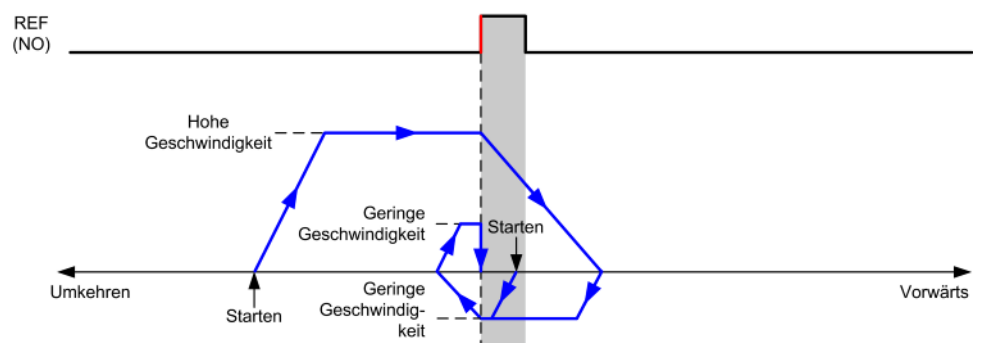
REF (NO) Referenzpunkt (Schließer)

## Kurze Referenz mit Umkehr

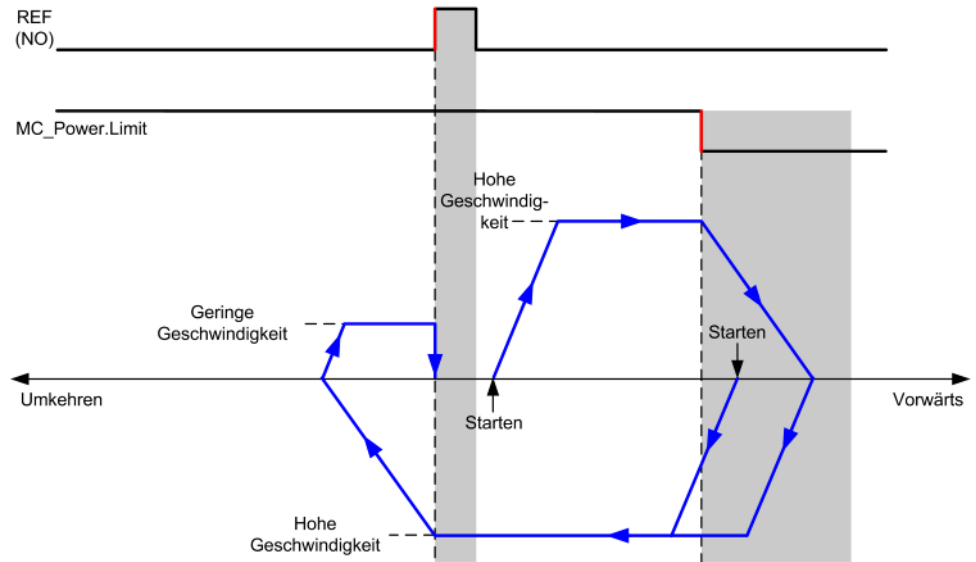
### Kurze Referenz mit Umkehr: positive Richtung

Homing erfolgt anhand der steigenden Flanke des Referenzschalters in Vorwärts-Richtung.

Die initiale Bewegungsrichtung ist von dem Zustand des Referenzschalters abhängig:



REF (NO) Referenzpunkt (Schließer)

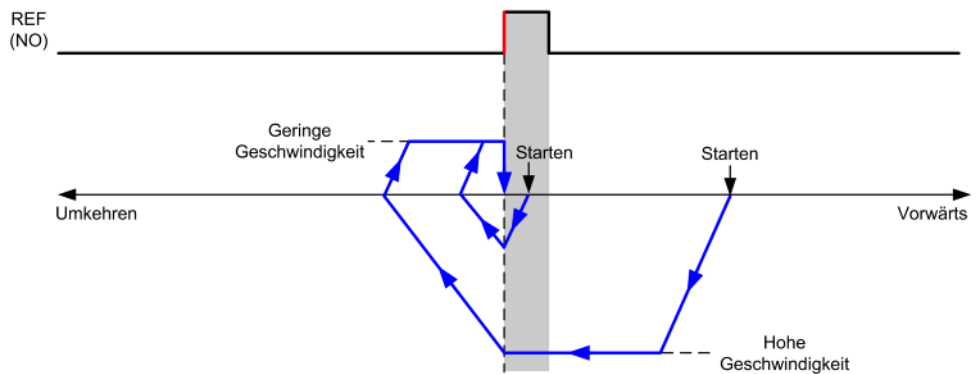


REF (NO) Referenzpunkt (Schließer)

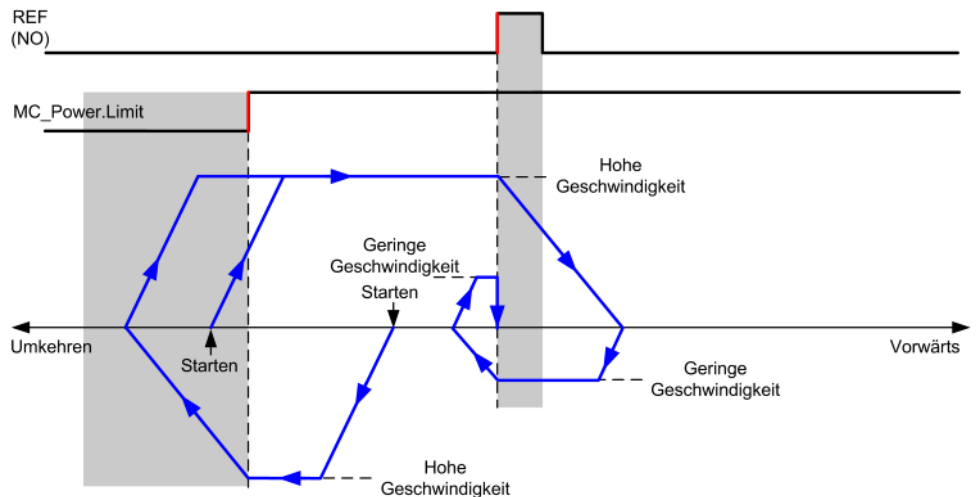
### Kurze Referenz mit Umkehr: negative Richtung

Homing erfolgt anhand der steigenden Flanke des Referenzschalters in Vorwärts-Richtung.

Die initiale Bewegungsrichtung ist von dem Zustand des Referenzschalters abhängig:



REF (NO) Referenzpunkt (Schließer)



REF (NO) Referenzpunkt (Schließer)

## Homing-Offset

### Beschreibung

Wenn der Ursprung nicht anhand der Schalter mit hinreichender Genauigkeit festgelegt werden kann, kann die Achse an eine bestimmte Position von dem Ursprungsschalter weg bewegt werden. Mit dem Homing-Offset können der mechanische und der elektrische Ursprung voneinander abweichend definiert werden.

Der Wert für das Homing-Offset wird als Impulsanzahl angegeben (-2.147.483.648...2.147.483.647, Standardwert 0). Wenn per Konfiguration festgelegt, wird zunächst der Befehl *MC\_Home\_PTO* ausgeführt und anschließend wird die angegebene Anzahl an Impulsen bei unterer Homing-Geschwindigkeit in der angegebenen Richtung ausgegeben.

**HINWEIS:** Die Wartezeit zwischen der Ausführung des *MC\_Home\_PTO*-Stoppbefehls über dem Ursprungsschalter und dem Start der Offsetbewegung ist unveränderlich auf 500 ms festgelegt. Die "In Ausführung"-Flag des *MC\_Home\_PTO*-Befehls wird erst dann freigegeben, wenn der Ursprungs-Offset ausgeführt ist.

## Datenparameter

### Überblick

In diesem Abschnitt werden die Datenparameter der *PTO*-Funktion beschrieben.

## Objektcodes für Funktionsbausteine

### Direction

In dieser Tabelle werden die Werte für die Objektcodes des Richtungsfunktionsbausteins aufgeführt.

Name	Wert	Beschreibung
<i>mcPositiveDirection</i>	1	CW (Im Uhrzeigersinn), vorwärts, positiv (entsprechend der Konfigurationseinstellung <b>Ausgangsmodus</b> ).
<i>mcNegativeDirection</i>	-1	CCW (Gegen den Uhrzeigersinn), rückwärts, negativ (entsprechend der Konfigurationseinstellung <b>Ausgangsmodus</b> ).

## Buffer Modes

In dieser Tabelle werden die Werte für die Objektcodes des Puffermodus-Funktionsbausteins aufgeführt.

Name	Wert	Beschreibung
<i>mcAborting</i>	0	Startet den Funktionsbaustein sofort (Standardmodus). Aktuelle Bewegungen werden abgebrochen. Die Bewegungswarteschlange wird geleert.
<i>mcBuffered</i>	1	Startet den Funktionsbaustein, nachdem die aktuelle Bewegung beendet wurde ( <i>Done</i> - bzw. <i>InVel</i> -Bit wurde auf TRUE gesetzt). Es werden keine Daten vermischt.
<i>mcBlendingPrevious</i>	3	Die Geschwindigkeit wird mit der Geschwindigkeit des ersten Funktionsbausteins gemischt (Mischung mit der Geschwindigkeit von <i>FB1</i> an der Endposition von <i>FB1</i> ).
<i>seTrigger</i>	10	Startet den Funktionsbaustein, sobald am Probeneingang ein Ereignis erkannt wird. Aktuelle Bewegungen werden abgebrochen. Die Bewegungswarteschlange wird geleert.
<i>seBufferedDelay</i>	11	Startet den Funktionsbaustein, nachdem die aktuelle Bewegung beendet wurde ( <i>Done</i> - bzw. <i>InVel</i> -Ausgang wurde auf TRUE gesetzt) und die Zeitverzögerung abgelaufen ist. Es werden keine Daten vermischt.  Der Parameter <i>Delay</i> wird mit <i>MC_WritePar_PTO</i> eingestellt, mit <i>ParameterNumber</i> 1000.

## Homing-Modi

In dieser Tabelle werden die Werte für die Objektcodes des Homing-Modi-Funktionsbausteins aufgeführt.

Name	Wert	Beschreibung
<i>PositionSetting</i>	0	Position.
<i>LongReference</i>	1	Ausführliche Referenz.
<i>ShortReference_Reversal</i>	20	Kurze Referenz.
<i>ShortReference_NoReversal</i>	21	Kurze Referenz ohne Umkehr.

## PTO-Parameter

In dieser Tabelle werden die Werte für die Objektcodes des PTO-Parameter-Funktionsbausteins aufgeführt.

Name	Parameternummer	L/S	Beschreibung
<i>CommandedPosition</i>	1	L	Angewiesene Position.
<i>SWLimitPos</i> (Oberer Grenzwert)	2	L/S	Positive Software-Positionsgrenzen
<i>SWLimitNeg</i> (Unterer Grenzwert)	3	L/S	Negative Software-Positionsgrenzen
<i>EnableLimitPos</i> (Positionsgrenzwerte der Software aktivieren)	4	L/S	Positive Software-Grenzwertschalter aktivieren (0 bis 1).
<i>EnableLimitNeg</i> (Positionsgrenzwerte der Software aktivieren)	5	L/S	Negative Software-Grenzwertschalter aktivieren (0 bis 1).
<i>MaxVelocityAppl</i> (Max. Geschwindigkeit)	9	L/S	Maximal zulässige Geschwindigkeit der Achse in der Anwendung (0 bis 100.000).
<i>ActualVelocity</i>	10	L	Geschwindigkeit der Achse.
<i>CommandedVelocity</i>	11	L	Angewiesene Geschwindigkeit.
<i>MaxAccelerationAppl</i> (Max. Beschl.)	13	L/S	Maximal zulässige Beschleunigung der Achse in der Anwendung (0 bis 100.000).
<i>MaxDecelerationAppl</i> (Max. Verz.)	15	L/S	Maximal zulässige Verzögerung der Achse in der Anwendung (0 bis 100.000).
Reserviert	16 bis 999	-	Reserviert für den PLCopen-Standard
<i>Delay</i>	1000	L/S	Zeit in ms (0...65.535) Standardwert: 0
<i>EnableDirPos</i>	1004	L/S	Positive Richtung aktivieren.  Wenn der Wert = 0 ist, ist die positive Richtung auf der Achse nicht erlaubt. Eine Bewegung des Funktionsbausteins, die eine Bewegung in positiver Richtung erzeugen würde, endet mit einem erkannten <i>InvalidDirectionValue</i> -Fehler (3006). Wenn eine Bewegung in negativer Richtung stattfindet, die von einem Bewegungsbefehl in positiver Richtung unterbrochen wird, wird der Fehler erst am Ende der Verzögerung der aktuellen negativen Bewegung erkannt.  Standardwert: 1 <b>HINWEIS:</b> Eine Wertänderung wird nur beim nächsten Bewegungsbefehl oder dem nächsten Auftreten von Geschwindigkeit = 0 berücksichtigt.
<i>EnableDirNeg</i>	1005	L/S	Negative Richtung aktivieren.  Wenn der Wert = 0 ist, ist die negative Richtung auf der Achse nicht erlaubt. Eine Bewegung des Funktionsbausteins, die eine Bewegung in negativer Richtung erzeugen würde, endet mit einem erkannten <i>InvalidDirectionValue</i> -Fehler (3006). Wenn eine Bewegung in positiver Richtung stattfindet, die von einem Bewegungsbefehl in negativer Richtung unterbrochen wird, wird der Fehler erst am Ende der Verzögerung der aktuellen positiven Bewegung erkannt.  Standardwert: 1 <b>HINWEIS:</b> Eine Wertänderung wird nur beim nächsten Bewegungsbefehl oder dem nächsten Auftreten von Geschwindigkeit = 0 berücksichtigt.

## Fehlercodes der PTO-Achse

In dieser Tabelle werden die Werte für die PTO-Achsen-Fehlercodes aufgeführt.

Name	Wert	Beschreibung
<i>NoError</i>	0	Kein Fehler erkannt.
<b>Achssteuerungsfehler</b>		
<i>InternalError</i>	1000	Interner Fehler in der Bewegungssteuerung entdeckt.
<i>DisabledAxis</i>	1001	Die Bewegung konnte nicht gestartet werden oder wurde abgebrochen, weil die Achse nicht bereit ist.
<i>HwPositionLimitP</i>	1002	Positiver Hardware-Positionsgrenzwert <i>limP</i> überschritten.
<i>HwPositionLimitN</i>	1003	Negativer Hardware-Positionsgrenzwert <i>limN</i> überschritten.
<i>SwPositionLimitP</i>	1004	Positive Software-Positionsgrenze überschritten.
<i>SwPositionLimitN</i>	1005	Negative Software-Positionsgrenze überschritten.
<i>ApplicationStopped</i>	1006	Anwendungsausführung wurde gestoppt (Steuerung im <i>STOPPED</i> - oder <i>HALT</i> -Zustand).
<i>OutputProtection</i>	1007	Der Kurzschlusschutz des Ausgangs ist auf den PTO-Kanälen aktiviert. Siehe die Beschreibung von <i>%S10</i> und <i>%SW139</i> im „Modicon M221 Logic Controller – Programmierhandbuch, Systembits und Systemwörter“ (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<i>OutputReset</i>	1008	<i>%S9</i> hat alle Ausgänge auf 0 forciert. Siehe Systembits.
<b>Achsensteuerungswarnungen</b>		
<i>WarningVelocityValue</i>	1100	Der angeforderte Geschwindigkeitsparameter ist außerhalb des Bereichs. Die Geschwindigkeit ist deshalb auf die konfigurierte maximale Geschwindigkeit begrenzt.
<i>WarningAccelerationValue</i>	1101	Der angeforderte Beschleunigungsparameter ist außerhalb des Bereichs. Die Beschleunigung ist deshalb auf die konfigurierte maximale Beschleunigung begrenzt.
<i>WarningDecelerationValue</i>	1102	Der angeforderte Verzögerungsparameter ist außerhalb des Bereichs. Die Verzögerung ist deshalb auf die konfigurierte maximale Verzögerung begrenzt.
<i>WarningJerkRatioValue</i>	1103	Der angeforderte jerk ratio-Parameter ist durch die konfigurierte maximale Beschleunigung oder Verzögerung begrenzt. In diesem Fall wird jerk ratio neu berechnet, um diese maximalen Werte zu berücksichtigen.

Eine **Achssteuerungswarnung** schaltet die Achse in den Zustand **ErrorStop** (*MC\_Reset\_PTO* ist zwingend erforderlich, um den Zustand **ErrorStop** zu verlassen). Der resultierende Achsenzustand kann an *MC\_ReadSts\_PTO* und *MC\_ReadAxisError\_PTO* abgelesen werden.

## PTO-Bewegungsbefehl-Fehlercodes

In dieser Tabelle werden die Werte für die Fehlercodes der PTO-Bewegungsbefehle aufgeführt.

Name	Wert	Beschreibung
<i>NoError</i>	0	Kein Fehler erkannt.
<b>Bewegungsstatusfehler</b>		
<i>ErrorStopActive</i>	2000	Die Bewegung konnte nicht gestartet werden oder wurde abgebrochen, da die Bewegung durch einen Zustand <b>ErrorStop</b> blockiert wird.
<i>StoppingActive</i>	2001	Die Bewegung konnte nicht gestartet werden, da die Bewegung durch <i>MC_Stop_PTO</i> mit Steuerung der Achse blockiert wird (entweder wird die Achse angehalten oder der Eingang <i>MC_Stop_PTO.Execute</i> verbleibt im Zustand TRUE).
<i>InvalidTransition</i>	2002	Es ist kein Übergang erlaubt. Weitere Informationen erhalten Sie im Bewegungsstatusdiagramm.
<i>InvalidSetPosition</i>	2003	<i>MC_SetPos_PTO</i> kann nicht ausgeführt werden, während sich die Achse bewegt.
<i>HomingError</i>	2004	Homing-Sequenz kann im aktuellen Modus nicht anhand der Referenznocke gestartet werden.
<i>InvalidProbeConf</i>	2005	Der Sondeneingang (Probe) muss konfiguriert sein.
<i>InvalidHomingConf</i>	2006	Der Referenzeingang muss für diesen Homing-Modus konfiguriert sein.
<i>InvalidAbsolute</i>	2007	Absolute Bewegungen können nur dann ausgeführt werden, wenn die Achse erfolgreich an die Ursprungsposition bewegt wurde. Zuerst muss eine Referenzierungssequenz ( <i>MC_Home_PTO</i> ) ausgeführt werden.
<i>MotionQueueFull</i>	2008	Die Bewegung konnte nicht gepuffert werden, weil die Bewegungswarteschlange voll ist.
<i>InvalidTransitionMotionTask</i>	2009	Eine Bewegungstask und andere Bewegungsfunktionsbausteine, die mit derselben Achse verbunden sind, können nicht nebeneinander ausgeführt werden.
<b>Bereichsfehler</b>		
<i>InvalidAxis</i>	3000	Der Funktionsbaustein kann für die angegebene Achse nicht verwendet werden.
<i>InvalidPositionValue</i>	3001	Positionsparameter liegt außerhalb der zulässigen Grenzen oder der Entfernungparameter führt zu einer solchen Position.
<i>InvalidVelocityValue</i>	3002	Geschwindigkeitsparameter liegt außerhalb des gültigen Bereichs.
<i>InvalidAccelerationValue</i>	3003	Beschleunigungsparameter liegt außerhalb des gültigen Bereichs.
<i>InvalidDecelerationValue</i>	3004	Verzögerungsparameter liegt außerhalb des gültigen Bereichs.
<i>InvalidBufferModeValue</i>	3005	Der Puffermodus entspricht keinem gültigen Wert.
<i>InvalidDirectionValue</i>	3006	Die Richtung entspricht keinem gültigen Wert oder ist ungültig, weil die Software- oder Hardware-Positionsgrenzwerte überschritten werden.
<i>InvalidHomeMode</i>	3007	Der Homing-Modus kann nicht verwendet werden.
<i>InvalidParameter</i>	3008	Die Parameternummer existiert für die angegebene Achse nicht.
<i>InvalidParameterValue</i>	3009	Parameterwert außerhalb des Bereichs.
<i>ReadOnlyParameter</i>	3010	Parameter ist schreibgeschützt.
<i>InvalidStepMotionTask</i>	3011	Motiontask-Schritttyp ist nicht definiert.

Ein **Bewegungszustandsfehler** oder ein **Bereichsfehler** hat auf den Achsenzustand, aktuell ausgeführte Bewegungen und die Bewegungswarteschlange keinerlei Auswirkungen. In diesem Fall wirkt sich der erkannte Fehler nur lokal auf den betroffenen Funktionsbaustein aus: Der *Error*-Ausgang wird auf TRUE gesetzt, und der *ErrorId*-Objektausgang wird auf den entsprechenden PTO-Bewegungsfehlercode gesetzt.

# Betriebsmodi

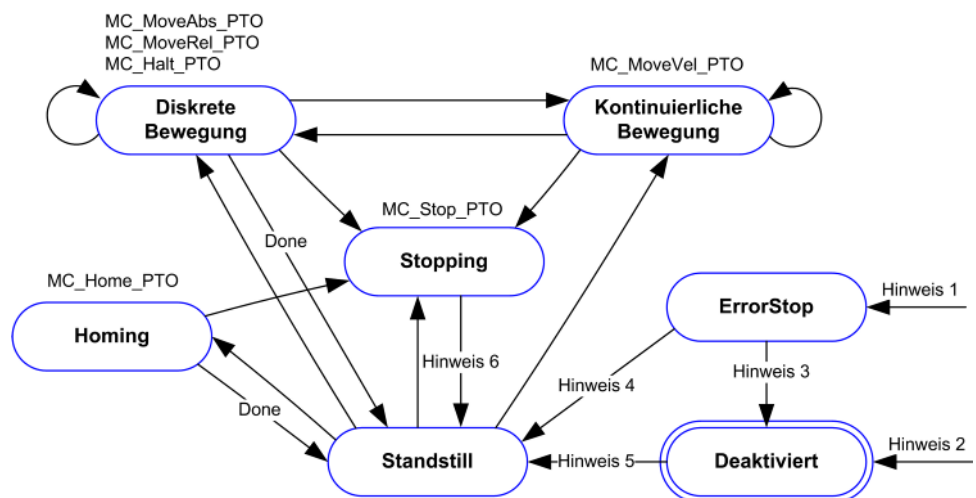
## Überblick

In diesem Abschnitt werden die Betriebsmodi beschrieben.

## Bewegungszustandsdiagramm

### Zustandsdiagramm

Die Achse befindet sich stets in einem in dem Diagramm angezeigten festgelegten Zustand.



**Hinweis 1** Von allen Zuständen aus, falls ein Fehler erkannt wurde.

**Hinweis 2** Von jedem Zustand außer *ErrorStop*, wenn `%MC_Power_PTO.Status = FALSE`.

**Hinweis 3** `%MC_Reset_PTO.Done = TRUE` und `%MC_Power_PTO.Status = FALSE`.

**Hinweis 4** `%MC_Reset_PTO.Done = TRUE` und `%MC_Power_PTO.Status = TRUE`.

**Hinweis 5** `%MC_Power_PTO.Status = TRUE`.

**Hinweis 6** `%MC_Stop_PTO.Done = TRUE` und `%MC_Stop_PTO.Execute = FALSE`.

In der folgenden Tabelle werden die Zustände der Achsen beschrieben:

Status	Beschreibung
<i>Disabled</i>	Initialer Zustand der Achse, es sind keine Bewegungsbefehle erlaubt. Die Achse ist nicht genullt.
<i>Standstill</i>	Der Strom ist eingeschaltet, es wird kein Fehler erkannt, und auf der Achse sind keine Bewegungsbefehle aktiv. Bewegungsbefehle sind erlaubt.
<i>ErrorStop</i>	Höchste Priorität; gilt im Falle eines an der Achse oder in der Steuerung erkannten Fehlers. Alle aktuellen Bewegungen werden durch eine <b>Schnellstopp-Verzögerung</b> abgebrochen. Der <i>Error</i> -Ausgang ist auf den entsprechenden Funktionsbausteinen auf TRUE gesetzt, und eine <i>ErrorId</i> legt den Fehlercode fest. Der Zustand verbleibt in <i>ErrorStop</i> , solange noch Fehler nicht behoben sind. Es sind keine Bewegungsbefehle möglich, bis das System unter Verwendung von <i>MC_Reset_PTO</i> zurückgesetzt wurde.
<i>Homing</i>	Gilt, wenn <i>MC_Home_PTO</i> die Achse steuert.
<i>Discrete</i>	Gilt, wenn die Achse von <i>MC_MoveRel_PTO</i> , <i>MC_MoveAbs_PTO</i> oder <i>MC_Halt_PTO</i> gesteuert wird.

Status	Beschreibung
<i>Continuous</i>	Gilt, wenn <i>MC_MoveVel_PTO</i> die Achse steuert.
<i>Stopping</i>	Gilt, wenn <i>MC_Stop_PTO</i> die Achse steuert.

**HINWEIS:** Funktionsbausteine, die im Diagramm nicht aufgeführt werden, haben keine Auswirkungen auf eine Statusänderung der Achse.

Der gesamte Bewegungsbefehl, einschließlich der Beschleunigungs- und Verzögerungsrampen, kann 4.294.967.295 Impulse nicht überschreiten. Bei der maximalen Frequenz von 100 kHz sind die Beschleunigungs- und Verzögerungsrampen auf 80 Sek. begrenzt.

## Bewegungsübergangstabelle

Der PTO-Kanal ist entsprechend der folgenden Tabelle in der Lage, auf einen neuen Befehl zu reagieren, während er den aktuellen Befehl ausführt (und bevor er diesen abschließt).

Befehl		Nächster Zustand					
		Home	MoveVel	MoveRel	MoveAbs	Halt	Stop
Strom	Standstill	Zulässig	Zulässig <sup>(1)</sup>	Zulässig <sup>(1)</sup>	Zulässig <sup>(1)</sup>	Zulässig	Zulässig
	Home	Unzulässig	Unzulässig	Unzulässig	Unzulässig	Unzulässig	Zulässig
	MoveVel	Unzulässig	Zulässig	Zulässig	Zulässig	Zulässig	Zulässig
	MoveRel	Unzulässig	Zulässig	Zulässig	Zulässig	Zulässig	Zulässig
	MoveAbs	Unzulässig	Zulässig	Zulässig	Zulässig	Zulässig	Zulässig
	Halt	Unzulässig	Zulässig	Zulässig	Zulässig	Zulässig	Zulässig
	Stop	Unzulässig	Unzulässig	Unzulässig	Unzulässig	Unzulässig	Unzulässig

<sup>(1)</sup> Wenn sich die Achse im Standstill-Status befindet, beginnt für die Puffermodi *mcAborting/mcBuffered/mcBlendingPrevious* die Bewegung sofort.

**Erlaubt** Der neue Befehl wird ausgeführt, auch wenn der vorherige Befehl noch nicht vollständig abgeschlossen ist.

**Abgelehnt** Der neue Befehl wird ignoriert, und es wird ein Fehler ausgegeben.

**HINWEIS:** Wenn beim Bewegungsübergang ein Fehler erkannt wird, geht die Achse in den Zustand **ErrorStop** über. *ErrorId* wird auf *InvalidTransition* gesetzt.

## Puffermodus

### Beschreibung

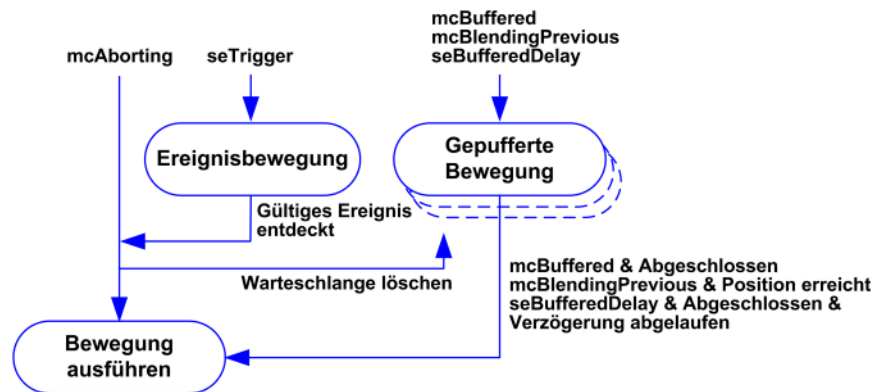
Einige der Bewegungs-Funktionsbausteine verfügen über ein Eingangsobjekt mit dem Namen *BufferMode*. Über dieses Eingangsobjekt kann der Funktionsbaustein wahlweise sofort oder beim Eintreffen eines Sondereignisses gestartet werden, oder das Eingangssignal kann gepuffert werden.

Die verfügbaren Optionen werden in den Objektcodes der Puffermodi des Funktionsbausteins, Seite 108 definiert:

- Eventuelle Abbruchbewegungen (*mcAborting*) werden sofort ausgeführt, die aktuelle Bewegung wird abgebrochen und die Bewegungswarteschlange wird gelöscht.
- Ereignisgesteuerte Bewegungen (*seTrigger*) sind Abbruchbewegungen und werden über ein Sondereignis, Seite 85 gestartet.
- Gepufferte Bewegungen (*mcBuffered*, *mcBlendingPrevious*, *seBufferedDelay*) werden in die Warteschlange eingestellt und an die Liste der aktuellen bzw. anstehenden Bewegungen angehängt und ausgeführt, nachdem die vorangehende Bewegung abgearbeitet ist.

## Bewegungswarteschlangendiagramm

Die folgende Abbildung stellt ein Bewegungswarteschlangendiagramm dar:



Der Puffer kann nur einen Bewegungs-Funktionsbaustein enthalten:

Die Ausführungsbedingung des Bewegungs-Funktionsbausteins in dem Puffer ist wie folgt gegeben:

- *mcBuffered*: wenn die aktuelle kontinuierliche Bewegung *InVel* ist oder wenn die aktuelle diskrete Bewegung angehalten wird.
- *seBufferedDelay*: wenn die angegebene Verzögerung abgelaufen ist, die aktuelle kontinuierliche Bewegung *InVel* ist oder die aktuelle diskrete Bewegung angehalten wird.
- *mcBlendingPrevious*: Wenn die Positions- und die Geschwindigkeitsvorgaben des aktuellen Funktionsbausteins erreicht sind.

Die Bewegungswarteschlange wird geleert (alle gepufferten Bewegungen werden gelöscht), wenn eine der folgenden Bedingungen zutrifft:

- Wenn eine Abbruchbewegung ausgelöst wird (*mcAborting* oder *seTrigger*): *CmdAborted* wird bei gepufferten Funktionsbausteinen auf TRUE gesetzt.
- Wenn eine *MC\_Stop\_PTO*-Funktion ausgeführt wird: *Error*-Ausgang wird bei geleerten gepufferten Funktionsbausteinen auf TRUE gesetzt, wobei *ErrorId=StoppingActive*.
- Wenn ein Übergang in den Zustand **ErrorStop** erkannt wird: *Error*-Ausgang wird bei gepufferten Funktionsbausteinen auf TRUE gesetzt, wobei *ErrorId=ErrorStopActive*.

### HINWEIS:

- Es können ausschließlich gültige Bewegungen in die Warteschlange eingestellt werden. Wenn die Ausführung des Funktionsbausteins mit gesetztem *Error*-Ausgang beendet wird, wird die Bewegung nicht in die Warteschlange eingestellt. Aktuell ausgeführte Bewegungen werden nicht beeinträchtigt, und die Warteschlange wird nicht geleert.
- Wenn die Warteschlange bereits voll ist, wird der *Error*-Ausgang in dem zuständigen Funktionsbaustein auf TRUE gesetzt, und der *ErrorId=MotionQueueFull* zurück.



# Bewegungs-Funktionsbausteine

## Übersicht

In diesem Kapitel werden die **Bewegungs**-Funktionsbausteine beschrieben.

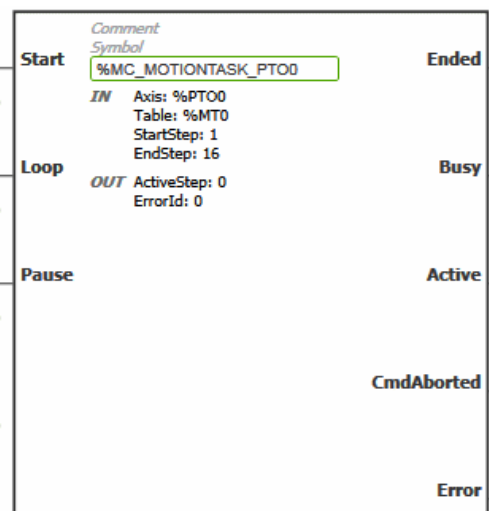
## MC\_MotionTask\_PTO Funktionsbaustein

### Beschreibung der Funktion

Sowohl die Funktionsbausteine mit einfacher Bewegung als auch der Motion Task Table-Funktionsbaustein (*MC\_MotionTask\_PTO*) kann für eine Achse verwendet werden.

Der *MC\_MotionTask\_PTO*-Funktionsbaustein kann jedoch nicht konkurrierend mit anderen Bewegungsfunktionsbausteinen ausgeführt werden. In diesem Fall wird ein Fehler erkannt und *ErrorId* wird auf *InvalidTransitionMotionTask* (2009), Seite 111 eingestellt.

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse und motion task table nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

## Eingänge

In dieser Tabelle werden die Eingänge des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Start</i>	FALSE	<p>Startet die Ausführung des Funktionsbausteins an der steigenden Flanke.</p> <p>Die <i>Loop</i>- und <i>Pause</i>-Eingänge können während der Ausführung des Funktionsbausteins geändert werden, was Auswirkungen auf die laufende Ausführung hat.</p> <p>Die <i>Axis</i>-, <i>Table</i>-, <i>StartStep</i>- und <i>EndStep</i>-Eingangsobjektwerte definieren die Bewegungsabfolge, wenn eine steigende Flanke auftritt. Eine spätere Änderung an diesen Eingangsparametern wirkt sich nicht auf die aktuelle Ausführung aus.</p> <p>Die Ausgänge werden festgelegt, wenn der Funktionsbaustein beendet wird.</p> <p>Bei FALSE:</p> <ul style="list-style-type: none"> <li>• Wenn die Ausführung noch läuft (Bewegung ist <i>Busy</i> und <i>Active</i>), werden die Ausgänge aktualisiert.</li> <li>• Wenn die Ausführung beendet ist, werden die Ausgänge einen Zyklus später zurückgesetzt.</li> </ul>
<i>Loop</i>	FALSE	<p>Bei TRUE, startet die Bewegungstaskabfolge wieder bei <i>StartStep</i>, sobald die Ausführung des Funktionsbausteins ohne Fehler beendet ist. Der <i>Ended</i>-Ausgang wird für einen Zyklus festgelegt.</p> <p>Der Eingang wird getestet, wenn die Ausführung des Funktionsbausteins ohne erkannten Fehler (<i>Ended</i>-Ausgang ist TRUE) beendet ist.</p>
<i>Pause</i>	FALSE	<p>Bei TRUE:</p> <ul style="list-style-type: none"> <li>• <i>Active</i> = 1 und <i>Busy</i> = 1</li> <li>• Forciert die Achse in den <b>Halt</b>-Zustand.</li> </ul> <p>Um den <b>Halt</b>-Zustand zu erreichen, wird die Achse im <b>Discrete motion</b>-Zustand verzögert und geht dann in den <b>Standstill</b>-Zustand über, wenn die Geschwindigkeit = 0 ist.</p> <ul style="list-style-type: none"> <li>• Der <b>Halt</b>-Zustand wird solange beibehalten, bis der <i>Pause</i>-Eingang auf TRUE steht.</li> <li>• Lässt den <i>Active</i>-Ausgang bestehen, selbst wenn die Geschwindigkeit gleich 0 ist.</li> </ul> <p>Wenn nach einer Einstellung auf TRUE ein Zurücksetzen auf FALSE erfolgt, fährt die Ausführung der Bewegungstask unter den folgenden Bedingungen fort:</p> <ul style="list-style-type: none"> <li>• Die Bewegungstask beginnt mit dem Wert der aktuellen Geschwindigkeit.</li> <li>• Die aktiven Schrittparameter werden verwendet.</li> <li>• Die absolute Zielposition wird nicht geändert. Wenn die Bewegungstask ein relativer Bewegungstyp ist, wird keine Distanz hinzugefügt.</li> <li>• Im aktuellen Schritt wird die Bedingung <b>Nächster Schritt</b> zurückgesetzt (Beispiel: die Verzögerung wird von 0 wieder gestartet, <i>Probe input event</i> wird aktiviert und wartet auf eine konfigurierte Flanke).</li> </ul>

In dieser Tabelle werden die Eingangsobjekte des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	%PTOx	–	Instanz der PTO-Achse, für die der Funktionsbaustein ausgeführt wird. Der Parameter wird in der Funktionsblockinstanz festgelegt, wie in der Modulregisterkarte <b>Programmieren &gt; Tools</b> zu finden. Wählen Sie den Parameter <b>Achse</b> im Dialogfenster <b>PTO-Objekte &gt; Bewegung &gt; MC_MotionTask_PTO &gt; MC_MotionTask_PTO Eigenschaften</b> .
<i>Table</i>	%MT	–	Tabelleninstanz, für die der Funktionsbaustein ausgeführt wird. Der Parameter wird in der Funktionsblockinstanz festgelegt, wie in der Modulregisterkarte <b>Programmieren &gt; Tools</b> zu finden. Wählen Sie den Parameter <b>Tabelle</b> im Dialogfenster <b>PTO-Objekte &gt; Bewegung &gt; MC_MotionTask_PTO &gt; MC_MotionTask_PTO Eigenschaften</b> .
<i>StartStep</i>	Byte	1	Schrittnummer, welche den ersten ausgeführten Schritt in Motion Task Table definiert.  Die Sequenz ist von <i>StartStep</i> bis <i>EndStep</i> ausführbar.  Einschränkung: $StartStep \leq EndStep$ .
<i>EndStep</i>	Byte	16	Schrittnummer, welche den letzten ausgeführten Schritt in Motion Task Table definiert.  Die Sequenz ist von <i>StartStep</i> bis <i>EndStep</i> ausführbar.  Einschränkung: $StartStep \leq EndStep$ .  <b>HINWEIS:</b> Wenn <i>EndStep</i> größer als die in Motion Task Table definierte maximale Anzahl an Schritten ist, wird der letzte Schritt der Tabelle verwendet.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Ended</i>	0	Wenn TRUE, wird die Ausführung des Funktionsbausteins ohne erkannte Fehler beendet.  <i>Ended</i> -Ausgangsverhalten: <ul style="list-style-type: none"> <li>Wenn der letzte Schritt einer Bewegungsabfolge eine <b>diskrete</b> Bewegung ist, verhält sich der Ausgang wie ein <i>Done</i>-Ausgang, die anderen Ausgänge (<i>Busy</i>, <i>Active</i>, <i>CmdAborted</i>, <i>Error</i>) werden auf 0 zurückgesetzt.</li> <li>Wenn der letzte Schritt der Bewegungsabfolge eine <b>kontinuierliche</b> Bewegung (Bewegungsgeschwindigkeit), dann verhält sich der Ausgang wie ein <i>InVel</i>-Ausgang.</li> </ul> Verhalten der anderen Ausgänge: <ul style="list-style-type: none"> <li><i>Busy</i> und <i>Active</i> sind TRUE (1).</li> <li><i>CmdAborted</i> und <i>Error</i> sind FALSE (0).</li> </ul> Wenn ein Loop angefragt wird ( <i>Loop</i> -Eingang), dann ist der <i>Ended</i> -Ausgang für einen Task-Zyklus TRUE.
<i>Busy</i>	-	Wenn TRUE, wird der Funktionsbaustein ausgeführt.  Wenn FALSE, wurde die Ausführung des Funktionsbausteins abgeschlossen.
<i>Active</i>	-	Wenn TRUE, hat die Funktionsbausteininstanz Kontrolle über die Achse. Es kann jeweils nur ein Funktionsbaustein <i>Active</i> für eine Achse auf TRUE setzen.
<i>CmdAborted</i>	-	Wenn TRUE, wird die Ausführung des Funktionsbausteins aufgrund eines anderen Bewegungsbefehls ( <i>MC_Stop_PTO</i> ) beendet oder ein Achsenfehler wird erkannt.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle werden die Ausgangsobjekte des Funktionsbausteins beschrieben:

Ausgangsobjekt	Typ	Initialwert	Beschreibung
<i>ActiveStep</i>	Byte	0	Anzahl der Schritte, die in Motion Task Table ausgeführt werden.
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

## Betriebsarten

Die Ausführung eines durch den *MC MotionTask PTO*-Funktionsbaustein angerufenen Motion Task Table stimmt mit dem Bewegungszustandsdiagramm, Seite 112 überein.

*MC\_MotionTask\_PTO*-Start: Der Funktionsbaustein kann nur aus dem Zustand **Standstill** gestartet werden.

*MC\_MotionTask\_PTO*-Stopp: Der Funktionsbaustein kann mit einer der folgenden Aktionen gestoppt werden:

- Einstellen des *Pause*-Eingangs auf TRUE.
- Ausführen von *MC\_Stop\_PTO*

Verhalten von Funktionsbausteinen bei erkannten Fehlern:

- Wenn während der Ausführung des Funktionsbausteins ein Fehler des Bewegungszustands oder Bereichs erkannt wird:
  - Ein Bewegungsstopp-Befehl wird an der Bewegungstask angewendet. Dabei wird der Parameterwert der aktuellen Schrittverzögerung verwendet. Wenn der Parameter der Schrittverzögerung nicht gültig ist, wird eine Schnellstopp-Verzögerung angewendet.
  - Während des kontrollierten Bewegungsstopps bleiben die Funktionsbausteinausgänge *Active* und *Busy* TRUE, mit dem Ausgangsobjekt *ActiveStep* = 0.
  - Wenn die Bewegung gestoppt ist, ist die Ausführung des Funktionsbausteins beendet mit *Error* = 1 und der Wert des *ErrorId*-Ausgangsobjekts wird auf den Wert gesetzt, der dem erkannten Fehlerwert entspricht.
- Wenn ein Achsenkontrollfehler erkannt wird, wechselt die Achse in den **ErrorStop**-Zustand. Die Ausführung des Funktionsbausteins wird mit *Error* = 1 und *ErrorId* = 2000 beendet.

## MC\_Power\_PTO Funktionsbaustein

### Verhalten

Die Achse ist deaktiviert, wenn ...

- *%MC\_Power\_PTO.Enable* = FALSE oder
- *%MC\_Power\_PTO.DriveReady* = FALSE oder
- ein Hardware-Grenzwertfehler entdeckt wird (*HwPositionLimitP* / *HwPositionLimitN*)

Wenn die Achse deaktiviert ist ...

- wechselt die Achse vom Status *Standstill* in den Status *Disabled* oder aus einer beliebigen aktuellen Bewegung in den Status *ErrorStop* und anschließend in den Status *Disabled* (wenn der erkannte Fehler zurückgesetzt wird).
- `%MC_ReadSts_PTO.IsHomed` wird auf 0 zurückgesetzt (ein neues Homing-Verfahren ist erforderlich).

## Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

## Eingänge

In dieser Tabelle werden die Eingänge des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Enable</i>	FALSE	Wenn TRUE, wird der Funktionsbaustein ausgeführt. Die Werte der anderen Funktionsbausteineingänge können kontinuierlich geändert werden, wobei die Ausgänge dann kontinuierlich aktualisiert werden.  Wenn FALSE, wird die Ausführung des Funktionsbausteins beendet, und die Ausgänge werden zurückgesetzt.
<i>DriveReady</i>	FALSE	Signal vom Antrieb, der seine Bereitschaft ankündigt.  Ist TRUE, wenn der Antrieb bereit ist, mit der Ausführung der Bewegung zu beginnen.  Wenn das Antriebssignal mit der Steuerung verbunden wird, verwenden Sie den geeigneten Steuerungseingang. Wenn der Antrieb dieses Signal nicht bereitstellt, können Sie mit einer beliebigen Booleschen TRUE-Variablen den Wert TRUE für diesen Eingang erzwingen.
<i>LimP</i>	TRUE	Hardware-Grenzwert-Schalterinformationen in positiver Richtung.  Ist FALSE, wenn der Hardware-Grenzwert erreicht wird.  Wenn das Hardware-Grenzwertsignal mit der Steuerung verbunden wird, verwenden Sie den geeigneten Steuerungseingang. Wenn der Antrieb dieses Signal nicht bereitstellt, können Sie mit einer beliebigen Booleschen TRUE-Variablen den Wert TRUE für diesen Eingang erzwingen.
<i>LimN</i>	TRUE	Hardware-Grenzwert-Schalterinformationen in negativer Richtung.  Ist FALSE, wenn der Hardware-Grenzwert erreicht wird.  Wenn das Hardware-Grenzwertsignal mit der Steuerung verbunden wird, verwenden Sie den geeigneten Steuerungseingang. Wenn der Antrieb dieses Signal nicht bereitstellt, können Sie mit einer beliebigen Booleschen TRUE-Variablen den Wert TRUE für diesen Eingang erzwingen.

In dieser Tabelle wird das Eingangsobjekt des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

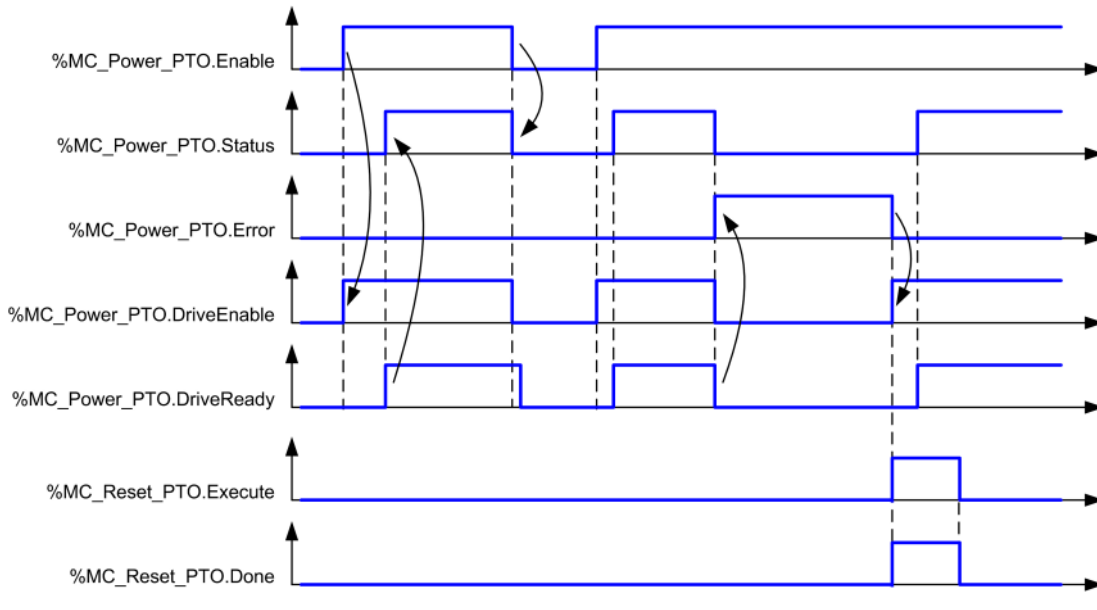
Ausgang	Initialwert	Beschreibung
<i>Status</i>	FALSE	Wenn TRUE, wird der Antrieb als bereit für Bewegungsbefehle angegeben.
<i>DriveEnable</i>	FALSE	Wenn TRUE, wird dem Antrieb vermittelt, dass er Bewegungsbefehle annehmen kann und deshalb die Stromversorgung aktivieren muss.  Wenn das Antriebssignal mit der Steuerung verbunden wird, verwenden Sie den geeigneten Steuerungsausgang. Wenn der Antrieb keinen Eingang für dieses Signal hat, können Sie diesen Ausgang des Funktionsbausteins ungenutzt lassen.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle wird das Ausgangsobjekt des Funktionsbausteins beschrieben:

Ausgangsobjekt	Typ	Initialwert	Beschreibung
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

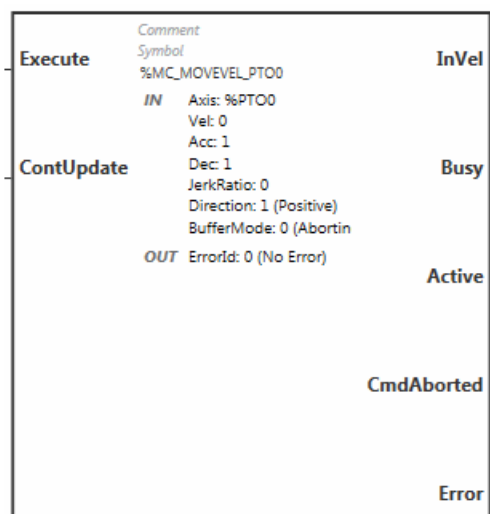
## Beispiel-Zeitdiagramm

Das nachstehende Diagramm illustriert die Funktionsweise des *MC\_Power\_PTO*-Funktionsbausteins:



## MC\_MoveVel\_PTO Funktionsbaustein

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

## Eingänge

In dieser Tabelle werden die Eingänge des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Execute</i>	FALSE	<p>Startet die Ausführung des Funktionsbausteins an der steigenden Flanke. Die Werte der anderen Funktionsbausteineingänge steuern die Ausführung des Funktionsbausteins an der steigenden Flanke von <i>Execute</i>. Eine spätere Änderung an diesen Eingangsparametern wirkt sich nicht auf die aktuelle Ausführung aus, es sei denn, der Eingang <i>ContUpdate</i> ist auf TRUE gesetzt.</p> <p>Die Ausgänge werden festgelegt, wenn der Funktionsbaustein beendet wird.</p> <p>Wenn während der Ausführung des Funktionsbausteins eine zweite steigende Flanke erkannt wird, wird die aktuelle Ausführung abgebrochen und der Funktionsbaustein wird erneut ausgeführt.</p>
<i>ContUpdate</i>	FALSE	<p>Wenn TRUE, wird der Funktionsbaustein angewiesen, geänderte Werte der Eingangsobjekte (<i>Vel</i>, <i>Acc</i>, <i>Dec</i> und <i>Direction</i>) zu verwenden und diese auf den aktuell ausgeführten Befehl anzuwenden.</p> <p>Dieser Eingang muss vor der steigenden Flanke auf dem Eingang <i>Execute</i> auf TRUE gesetzt sein, damit er berücksichtigt wird.</p> <p><b>HINWEIS:</b> Eine Änderung des <i>Axis</i>-Parameterwerts wird nicht berücksichtigt. Sie müssen <i>Execute</i> auf 0 und dann auf 1 setzen, um die <i>Axis</i> zu ändern.</p>

In dieser Tabelle werden die Eingangsobjekte des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.
<i>Vel</i>	DINT	0	Zielgeschwindigkeit. Bereich Hz: 0 bis <i>MaxVelocityAppl</i> , Seite 109
<i>Acc</i>	DINT	0	Beschleunigung in Hz/ms Bereich (Hz/ms): 1... <i>MaxAccelerationAppl</i> , Seite 109
<i>Dec</i>	DINT	0	Verzögerung in Hz/ms Bereich (Hz/ms): 1... <i>MaxDecelerationAppl</i> , Seite 109
<i>JerkRatio</i>	INT	0	Prozentsatz der verwendeten Beschleunigungs-/Verzögerungsanpassung für die Erstellung des S-Kurven-Profiles, Seite 84. Bereich: 0...100
<i>Direction</i>	INT	<i>mcPositiveDirection</i>	Bewegungsrichtung für den PTO-Typ CW/CCW vorwärts (CW) = 1 ( <i>mcPositiveDirection</i> ) rückwärts (CCW) = -1 ( <i>mcNegativeDirection</i> )
<i>BufferMode</i>	INT	<i>mcAborting</i>	Übergangsmodus aus der aktuellen Bewegung. Weitere Informationen finden Sie in der Tabelle zu Puffermodi, Seite 108.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>InVel</i>	FALSE	Wenn TRUE, wurde die Zielgeschwindigkeit erreicht.
<i>Busy</i>	-	Wenn TRUE, wird der Funktionsbaustein ausgeführt. Wenn FALSE, wurde die Ausführung des Funktionsbausteins abgeschlossen. Der Funktionsbaustein muss mindestens so lange wie <i>Busy</i> = TRUE in einer aktiven Task des Anwendungsprogramms belassen werden.
<i>Active</i>	-	Wenn TRUE, hat die Funktionsbausteininstanz Kontrolle über die Achse. Es kann jeweils nur ein Funktionsbaustein <i>Active</i> für eine Achse auf TRUE setzen.
<i>CmdAborted</i>	-	Wenn TRUE, wird die Ausführung des Funktionsbausteins aufgrund eines anderen Bewegungsbefehls beendet.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle wird das Ausgangsobjekt des Funktionsbausteins beschrieben:

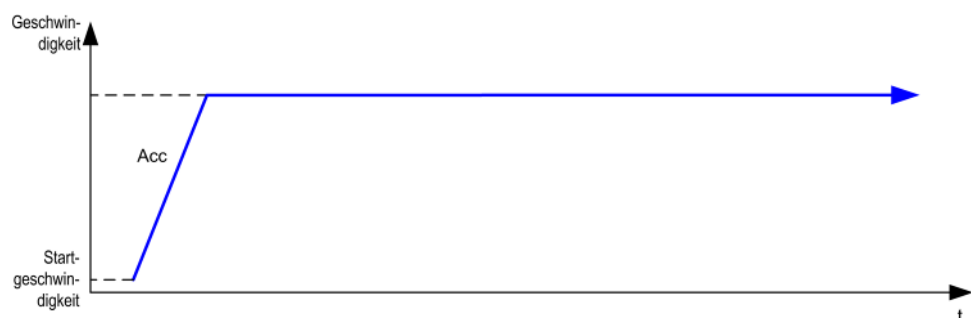
Ausgangs-objekt	Typ	Initialwert	Beschreibung
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

### HINWEIS:

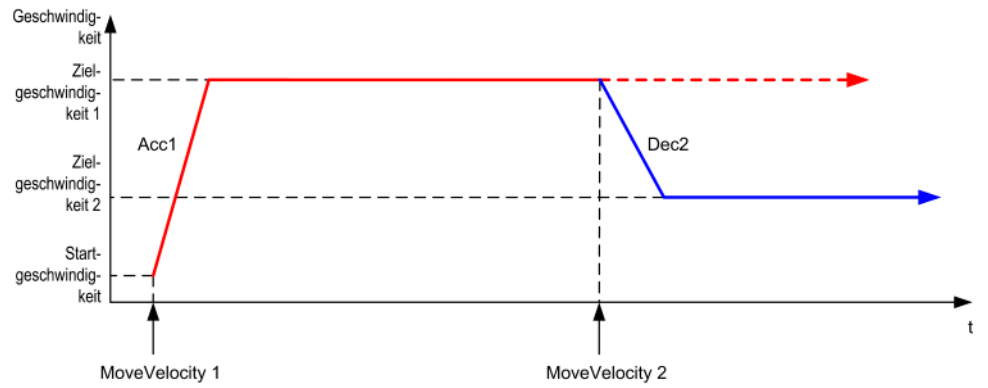
- Um die Bewegung anzuhalten, muss der Funktionsbaustein durch einen anderen Funktionsbaustein unterbrochen werden, der einen neuen Befehl ausgibt.
- Wenn aktuell eine Bewegung ausgeführt wird, und die Richtung wird umgekehrt, wird die Bewegung zunächst mit der Verzögerung des Funktionsbausteins *MC\_MoveVel\_PTO* angehalten und anschließend wird die Bewegung in umgekehrter Richtung wiederaufgenommen.
- Die Dauer der Beschleunigung bzw. Verzögerung des Segmentbausteins darf 80 Sekunden nicht überschreiten.

## Beispiel-Zeitdiagramm

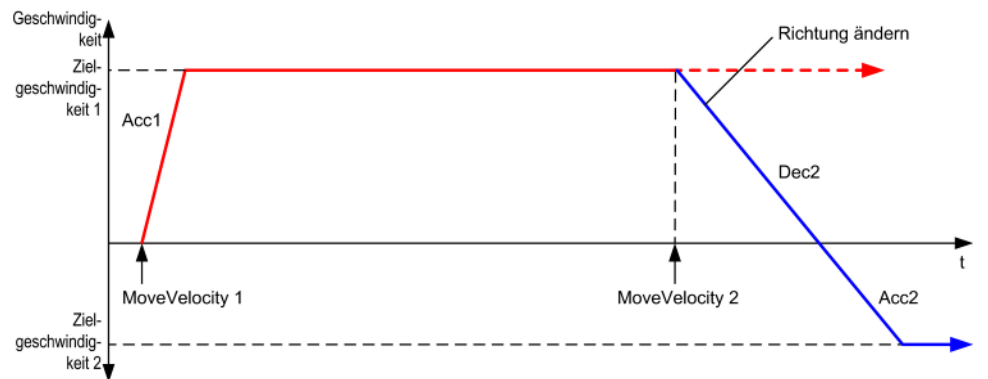
Die folgende Abbildung zeigt ein einfaches Profil aus dem Zustand **Standstill**:



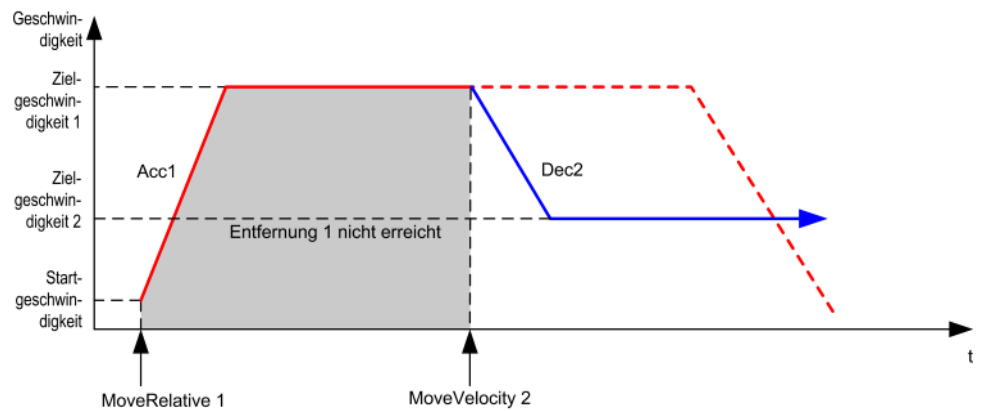
Die folgende Abbildung zeigt ein komplexes Profil aus dem Zustand **Continuous**:



Die folgende Abbildung zeigt ein komplexes Profil aus dem Zustand **Continuous** mit Richtungsänderung:

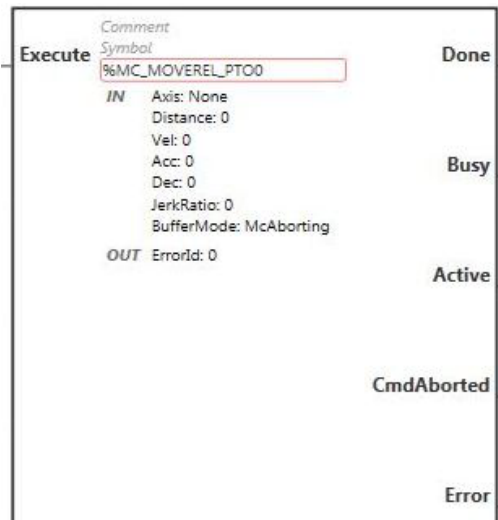


Die folgende Abbildung zeigt ein komplexes Profil aus dem Zustand **Discrete**:



## MC\_MoveRel\_PTO Funktionsbaustein

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

### Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Execute</i>	FALSE	Startet die Ausführung des Funktionsbausteins an der steigenden Flanke. Die Werte der anderen Funktionsbausteineingänge steuern die Ausführung des Funktionsbausteins an der steigenden Flanke von <i>Execute</i> . Eine spätere Änderung an diesen Eingangsparametern wirkt sich nicht auf die aktuelle Ausführung aus.  Die Ausgänge werden festgelegt, wenn der Funktionsbaustein beendet wird.

In dieser Tabelle werden die Eingangsobjekte des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.
<i>Distance</i>	DINT	0	Relative Entfernung für die Bewegung in Impulsen. Das Vorzeichen gibt die Bewegungsrichtung an.
<i>Vel</i>	DINT	0	Zielgeschwindigkeit. Bereich Hz: 0 bis <i>MaxVelocityAppl</i> , Seite 109
<i>Acc</i>	DINT	0	Beschleunigung in Hz/ms Bereich (Hz/ms): 1... <i>MaxAccelerationAppl</i> , Seite 109
<i>Dec</i>	DINT	0	Verzögerung in Hz/ms Bereich (Hz/ms): 1... <i>MaxDecelerationAppl</i> , Seite 109
<i>JerkRatio</i>	INT	0	Prozentsatz der verwendeten Beschleunigungs-/Verzögerungsanpassung für die Erstellung des S-Kurven-Profiles, Seite 84. Bereich: 0...100
<i>BufferMode</i>	INT	<i>mcAborting</i>	Übergangsmodus aus der aktuellen Bewegung. Weitere Informationen finden Sie in der Tabelle zu Puffermodi, Seite 108.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Done</i>	FALSE	Wenn TRUE, wird die Ausführung des Funktionsbausteins ohne erkannte Fehler beendet. Wenn eine Bewegung auf einer Achse durch eine andere Bewegung auf derselben Achse unterbrochen wird, bevor die befohlene Aktion beendet ist, wird <i>CmdAborted</i> auf TRUE und <i>Done</i> auf FALSE gesetzt.
<i>Busy</i>	-	Wenn TRUE, wird der Funktionsbaustein ausgeführt. Wenn FALSE, wurde die Ausführung des Funktionsbausteins abgeschlossen. Der Funktionsbaustein muss mindestens so lange wie <i>Busy</i> = TRUE in einer aktiven Task des Anwendungsprogramms belassen werden.
<i>Active</i>	-	Wenn TRUE, hat die Funktionsbausteininstanz Kontrolle über die Achse. Es kann jeweils nur ein Funktionsbaustein <i>Active</i> für eine Achse auf TRUE setzen.
<i>CmdAborted</i>	-	Wenn TRUE, wird die Ausführung des Funktionsbausteins aufgrund eines anderen Bewegungsbefehls beendet.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle wird das Ausgangsobjekt des Funktionsbausteins beschrieben:

Ausgangsobjekt	Typ	Initialwert	Beschreibung
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

**HINWEIS:**

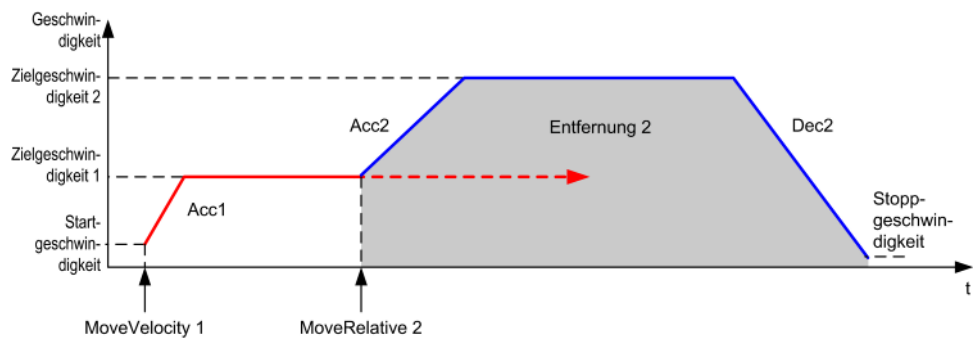
- Der Funktionsbaustein wird mit einer Geschwindigkeit von 0 beendet, wenn keine weiteren Funktionsbausteine zur Ausführung anstehen.
- Wenn die Entfernung für die Zielgeschwindigkeit zu gering ist, ist das Bewegungsprofil dreieckig statt trapezförmig.
- Wenn aktuell eine Bewegung ausgeführt wird und eine angewiesene Entfernung aufgrund der aktuellen Bewegungsparameter unerreichbar ist, erfolgt die Verwaltung der Richtungsumkehrung automatisch: Die Bewegung wird zunächst mit der Verzögerung des Funktionsbausteins *MC\_MoveRel\_PTO* angehalten, und anschließend wird die Bewegung in umgekehrter Richtung wiederaufgenommen.
- Die Dauer der Beschleunigung bzw. Verzögerung des Segmentbausteins darf 80 Sekunden nicht überschreiten.

**Beispiel-Zeitdiagramm**

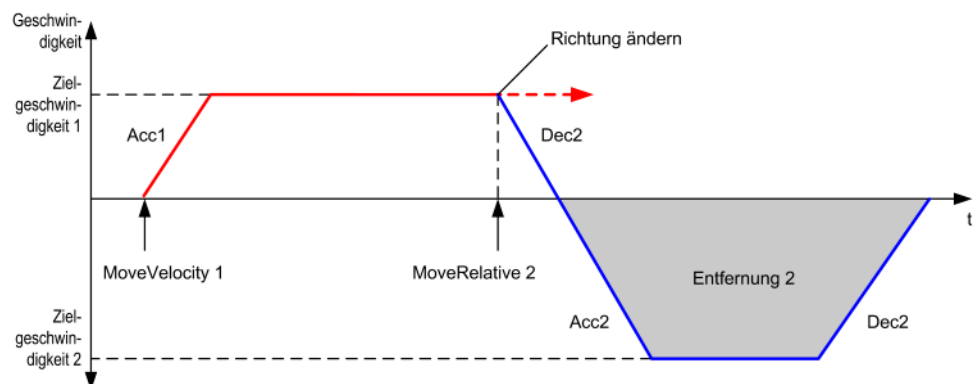
Die folgende Abbildung zeigt ein einfaches Profil aus dem Zustand **Standstill**:



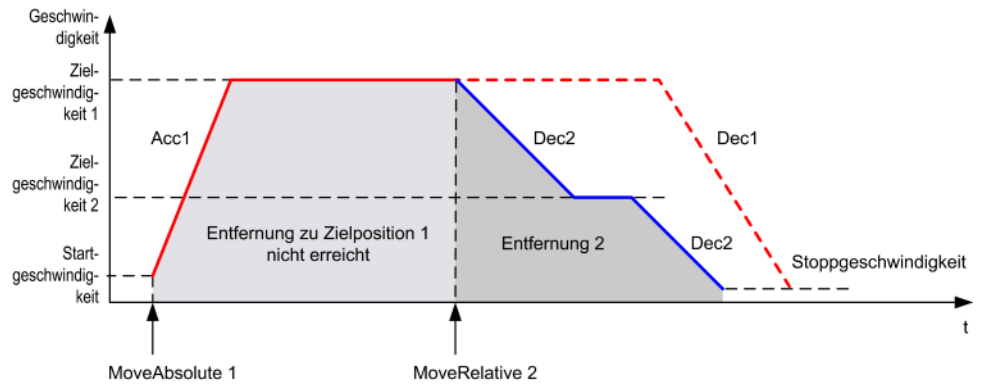
Die folgende Abbildung zeigt ein komplexes Profil aus dem Zustand **Continuous**:



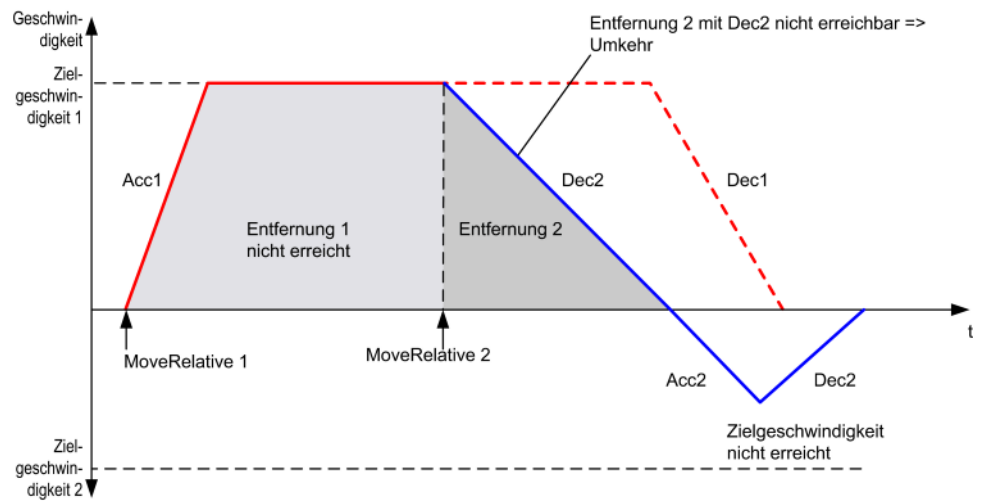
Die folgende Abbildung zeigt ein komplexes Profil aus dem Zustand **Continuous** mit Richtungsänderung:



Die folgende Abbildung zeigt ein komplexes Profil aus dem Zustand **Discrete**:

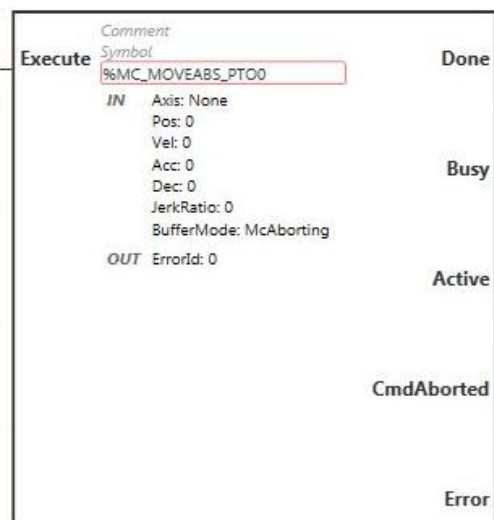


Die folgende Abbildung zeigt ein komplexes Profil aus dem Zustand **Discrete** mit Richtungsänderung:



## MC\_MoveAbs\_PTO Funktionsbaustein

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

## Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Execute</i>	FALSE	Startet die Ausführung des Funktionsbausteins an der steigenden Flanke. Die Werte der anderen Funktionsbausteineingänge steuern die Ausführung des Funktionsbausteins an der steigenden Flanke von <i>Execute</i> . Eine spätere Änderung an diesen Eingangsparametern wirkt sich nicht auf die aktuelle Ausführung aus.  Die Ausgänge werden festgelegt, wenn der Funktionsbaustein beendet wird.

In dieser Tabelle werden die Eingangsobjekte des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.
<i>Pos</i>	DINT	0	Position der Achse.
<i>Vel</i>	DINT	0	Zielgeschwindigkeit. Bereich Hz: 0 bis <i>MaxVelocityAppl</i> , Seite 109
<i>Acc</i>	DINT	0	Beschleunigung in Hz/ms Bereich (Hz/ms): 1... <i>MaxAccelerationAppl</i> , Seite 109
<i>Dec</i>	DINT	0	Verzögerung in Hz/ms Bereich (Hz/ms): 1... <i>MaxDecelerationAppl</i> , Seite 109
<i>JerkRatio</i>	INT	0	Prozentsatz der verwendeten Beschleunigungs-/Verzögerungsanpassung für die Erstellung des S-Kurven-Profiles, Seite 84. Bereich: 0...100
<i>BufferMode</i>	INT	<i>mcAborting</i>	Übergangsmodus aus der aktuellen Bewegung. Weitere Informationen finden Sie in der Tabelle zu Puffermodi, Seite 108.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Done</i>	FALSE	Wenn TRUE, wird die Ausführung des Funktionsbausteins ohne erkannte Fehler beendet.  Wenn eine Bewegung auf einer Achse durch eine andere Bewegung auf derselben Achse unterbrochen wird, bevor die befohlene Aktion beendet ist, wird <i>CmdAborted</i> auf TRUE und <i>Done</i> auf FALSE gesetzt.
<i>Busy</i>	-	Wenn TRUE, wird der Funktionsbaustein ausgeführt.  Wenn FALSE, wurde die Ausführung des Funktionsbausteins abgeschlossen.  Der Funktionsbaustein muss mindestens so lange wie <i>Busy</i> = TRUE in einer aktiven Task des Anwendungsprogramms belassen werden.
<i>Active</i>	-	Wenn TRUE, hat die Funktionsbausteininstanz Kontrolle über die Achse. Es kann jeweils nur ein Funktionsbaustein <i>Active</i> für eine Achse auf TRUE setzen.
<i>CmdAborted</i>	-	Wenn TRUE, wird die Ausführung des Funktionsbausteins aufgrund eines anderen Bewegungsbefehls beendet.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle wird das Ausgangsobjekt des Funktionsbausteins beschrieben:

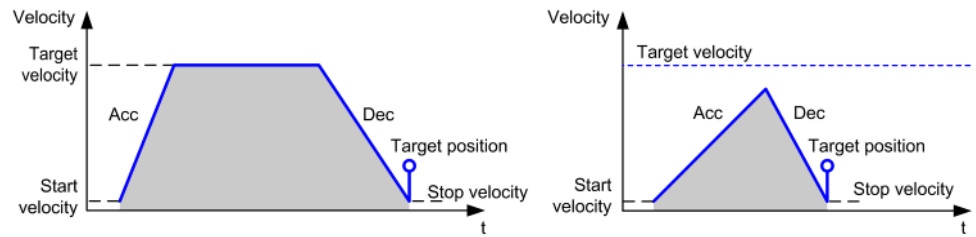
Ausgangs-objekt	Typ	Initialwert	Beschreibung
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

### HINWEIS:

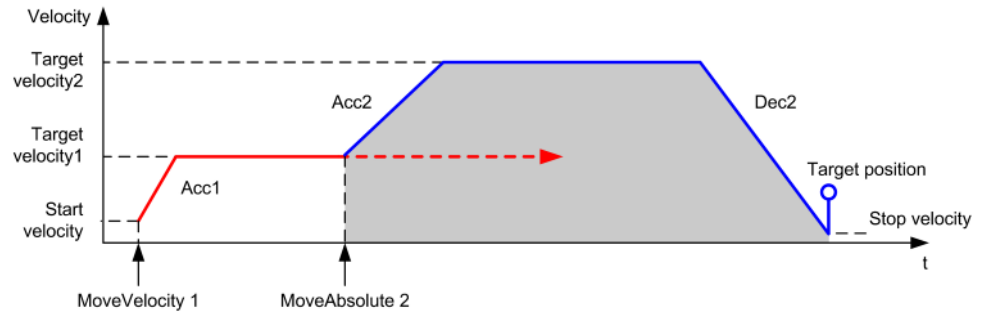
- Der Funktionsbaustein wird mit einer Geschwindigkeit von 0 beendet, wenn keine weiteren Funktionsbausteine zur Ausführung anstehen.
- Die Richtung der Bewegung wird entsprechend der aktuellen und der Zielposition automatisch eingestellt.
- Wenn die Entfernung für die Zielgeschwindigkeit zu gering ist, ist das Bewegungsprofil dreieckig statt trapezförmig.
- Wenn die Position in der aktuellen Bewegungsrichtung nicht erreicht werden kann, wird die Richtungsumkehrung automatisch verwaltet. Wenn aktuell eine Bewegung ausgeführt wird, und die Richtung wird umgekehrt, wird die Bewegung zunächst mit der Verzögerung des Funktionsbausteins *MC\_MoveAbsolute\_PTO* angehalten und anschließend wird die Bewegung in umgekehrter Richtung wiederaufgenommen.
- Die Dauer der Beschleunigung bzw. Verzögerung des Segmentbausteins darf 80 Sekunden nicht überschreiten.

## Beispiel-Zeitdiagramm

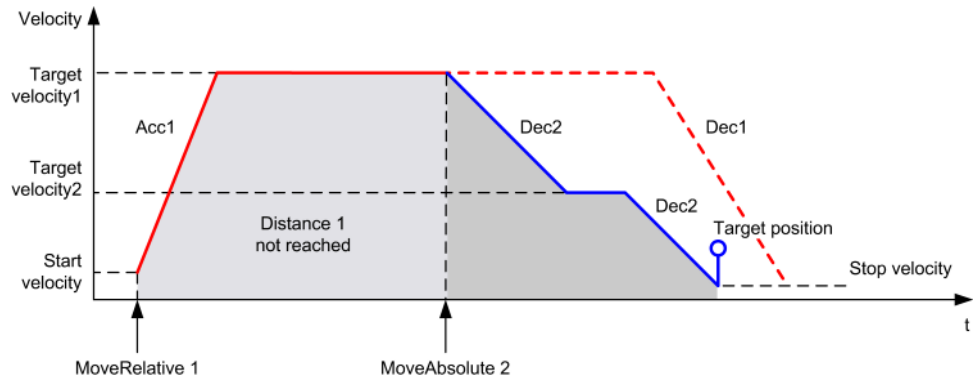
Die folgende Abbildung zeigt ein einfaches Profil aus dem Zustand **Standstill**:



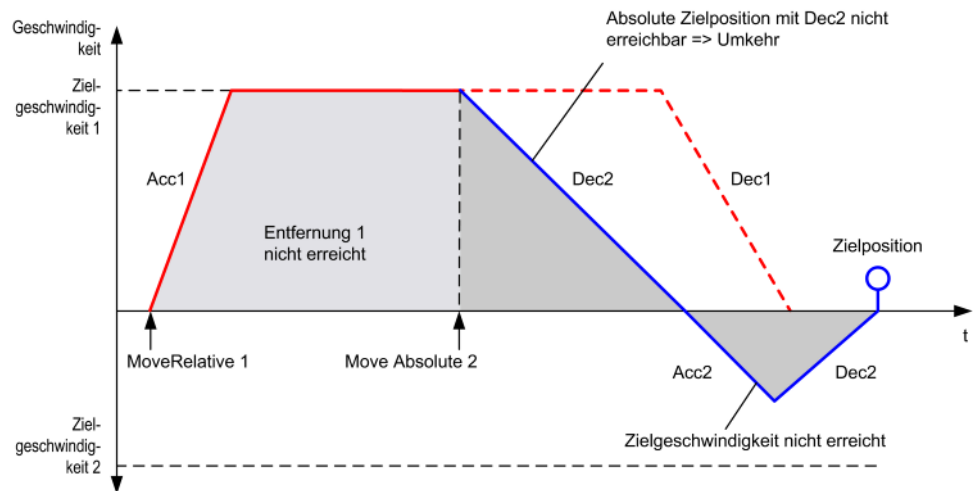
Die folgende Abbildung zeigt ein komplexes Profil aus dem Zustand **Continuous**:



Die folgende Abbildung zeigt ein komplexes Profil aus dem Zustand **Discrete**:

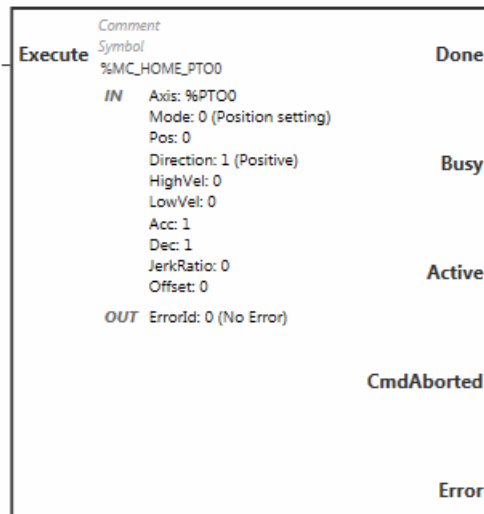


Die folgende Abbildung zeigt ein komplexes Profil aus dem Zustand **Discrete** mit Richtungsänderung:



## MC\_Home\_PTO Funktionsbaustein

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

### Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Execute</i>	FALSE	Startet die Ausführung des Funktionsbausteins an der steigenden Flanke. Die Werte der anderen Funktionsbausteineingänge steuern die Ausführung des Funktionsbausteins an der steigenden Flanke von <i>Execute</i> . Eine spätere Änderung an diesen Eingangsparametern wirkt sich nicht auf die aktuelle Ausführung aus.  Die Ausgänge werden festgelegt, wenn der Funktionsbaustein beendet wird.

In dieser Tabelle werden die Eingangsobjekte des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.
<i>Mode</i>	BYTE	0	Vordefinierter Homing-Sequenztyp, Seite 108.
<i>Pos</i>	DINT	0	Position der Achse.
<i>HighVel</i>	DINT	0	Ziel-Homing-Geschwindigkeit bei der Abtastung von Grenzwerten bzw. dem Referenzschalter. Bereich Hz: 1... <i>MaxVelocityAppl</i> , Seite 109
<i>LowVel</i>	DINT	0	Ziel-Homing-Geschwindigkeit bei der Abtastung des Referenzschaltersignals. Die Bewegung stoppt, wenn der Grenzwert oder Referenzschalter entdeckt wird. Bereich Hz: 1... <i>HighVelocity</i>
<i>Acc</i>	DINT	0	Beschleunigung in Hz/ms Bereich (Hz/ms): 1... <i>MaxAccelerationAppl</i> , Seite 109
<i>Dec</i>	DINT	0	Verzögerung in Hz/ms Bereich (Hz/ms): 1... <i>MaxDecelerationAppl</i> , Seite 109
<i>JerkRatio</i>	INT	0	Prozentsatz der verwendeten Beschleunigungs-/Verzögerungsanpassung für die Erstellung des S-Kurven-Profiles, Seite 84. Bereich: 0...100
<i>Direction</i>	INT	<i>mcPositiveDirection</i>	Bewegungsrichtung für den PTO-Typ CW/CCW vorwärts (CW) = 1 ( <i>mcPositiveDirection</i> ) rückwärts (CCW) = -1 ( <i>mcNegativeDirection</i> )
<i>Offset</i>	DINT	0	Entfernung vom Ursprung. Wenn der Ursprung erreicht wurde, wird die Bewegung wiederaufgenommen, bis die angegebene Entfernung erreicht ist. Die Richtungsangabe ist vorzeichenbehaftet (Homing-Offset, Seite 107). Bereich: -2.147.483.648 bis 2.147.483.647

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Done</i>	FALSE	Wenn TRUE, wird die Ausführung des Funktionsbausteins ohne erkannte Fehler beendet.  Wenn eine Bewegung auf einer Achse durch eine andere Bewegung auf derselben Achse unterbrochen wird, bevor die befohlene Aktion beendet ist, wird <i>CmdAborted</i> auf TRUE und <i>Done</i> auf FALSE gesetzt.
<i>Busy</i>	-	Wenn TRUE, wird der Funktionsbaustein ausgeführt.  Wenn FALSE, wurde die Ausführung des Funktionsbausteins abgeschlossen.  Der Funktionsbaustein muss mindestens so lange wie <i>Busy</i> = TRUE in einer aktiven Task des Anwendungsprogramms belassen werden.
<i>Active</i>	-	Wenn TRUE, hat die Funktionsbausteininstanz Kontrolle über die Achse. Es kann jeweils nur ein Funktionsbaustein <i>Active</i> für eine Achse auf TRUE setzen.
<i>CmdAborted</i>	-	Wenn TRUE, wird die Ausführung des Funktionsbausteins aufgrund eines anderen Bewegungsbefehls beendet.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle wird das Ausgangsobjekt des Funktionsbausteins beschrieben:

Ausgangsobjekt	Typ	Initialwert	Beschreibung
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

**HINWEIS:** Die Dauer der Beschleunigung bzw. Verzögerung des Segmentbausteins darf 80 Sekunden nicht überschreiten.

## Beispiel-Zeitdiagramm

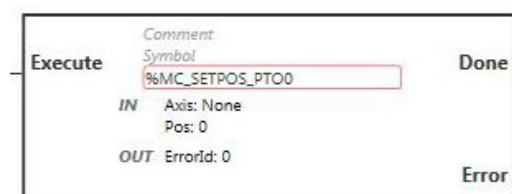
Homing-Modi, Seite 101

## MC\_SetPos\_PTO Funktionsbaustein

### Verhalten

Dieser Funktionsbaustein verändert den Wert der tatsächlichen Position der Achse ohne physische Bewegung. Er kann nur verwendet werden, wenn die Achse sich im Zustand *Standstill* befindet.

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

## Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Execute</i>	FALSE	Startet die Ausführung des Funktionsbausteins an der steigenden Flanke. Die Werte der anderen Funktionsbausteineingänge steuern die Ausführung des Funktionsbausteins an der steigenden Flanke von <i>Execute</i> . Eine spätere Änderung an diesen Eingangsparametern wirkt sich nicht auf die aktuelle Ausführung aus.  Die Ausgänge werden festgelegt, wenn der Funktionsbaustein beendet wird.

In dieser Tabelle werden die Eingangsobjekte des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.
<i>Pos</i>	DINT	0	Position der Achse.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

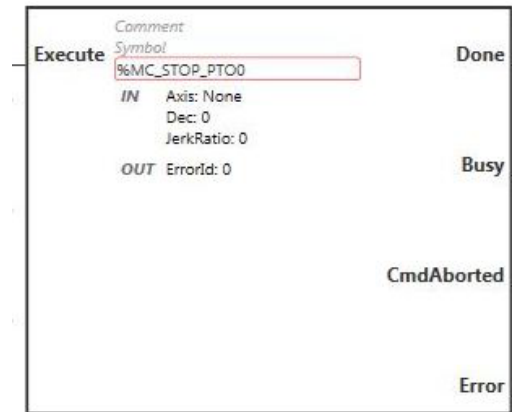
Ausgang	Initialwert	Beschreibung
<i>Done</i>	FALSE	Wenn TRUE, wird die Ausführung des Funktionsbausteins ohne erkannte Fehler beendet.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle wird das Ausgangsobjekt des Funktionsbausteins beschrieben:

Ausgangsobjekt	Typ	Initialwert	Beschreibung
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

# MC\_Stop\_PTO Funktionsbaustein

## Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

## Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Execute</i>	FALSE	Startet die Ausführung des Funktionsbausteins an der steigenden Flanke. Die Werte der anderen Funktionsbausteineingänge steuern die Ausführung des Funktionsbausteins an der steigenden Flanke von <i>Execute</i> . Eine spätere Änderung an diesen Eingangsparametern wirkt sich nicht auf die aktuelle Ausführung aus.  Die Ausgänge werden festgelegt, wenn der Funktionsbaustein beendet wird.

In dieser Tabelle werden die Eingangsobjekte des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.
<i>Dec</i>	DINT	0	Verzögerung in Hz/ms Bereich (Hz/ms): 1... <i>MaxDecelerationAppl</i> , Seite 109
<i>JerkRatio</i>	INT	0	Prozentsatz der verwendeten Beschleunigungs-/ Verzögerungsanpassung für die Erstellung des S-Kurven-Profiles, Seite 84. Bereich: 0...100

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Done</i>	FALSE	Wenn TRUE, wird die Ausführung des Funktionsbausteins ohne erkannte Fehler beendet.  Wenn eine Bewegung auf einer Achse durch eine andere Bewegung auf derselben Achse unterbrochen wird, bevor die befohlene Aktion beendet ist, wird <i>CmdAborted</i> auf TRUE und <i>Done</i> auf FALSE gesetzt.
<i>Busy</i>	-	Wenn TRUE, wird der Funktionsbaustein ausgeführt.  Wenn FALSE, wurde die Ausführung des Funktionsbausteins abgeschlossen.  Der Funktionsbaustein muss mindestens so lange wie <i>Busy</i> = TRUE in einer aktiven Task des Anwendungsprogramms belassen werden.
<i>CmdAborted</i>	-	Wenn TRUE, wird die Ausführung des Funktionsbausteins aufgrund eines anderen Bewegungsbefehls beendet.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle wird das Ausgangsobjekt des Funktionsbausteins beschrieben:

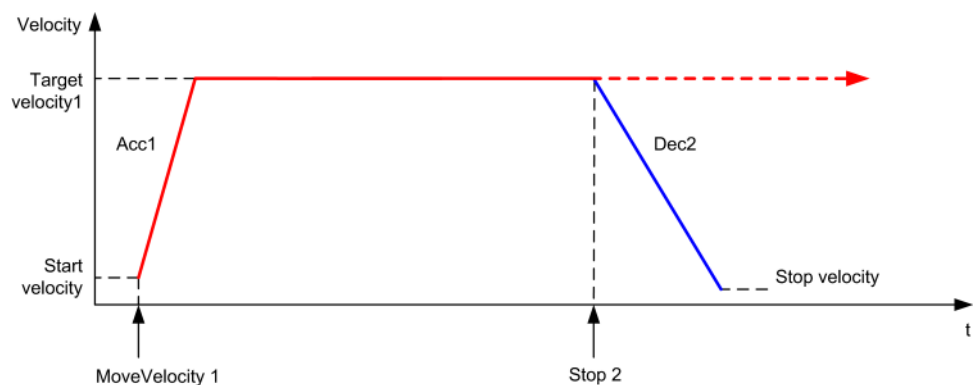
Ausgangs-objekt	Typ	Initialwert	Beschreibung
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

### HINWEIS:

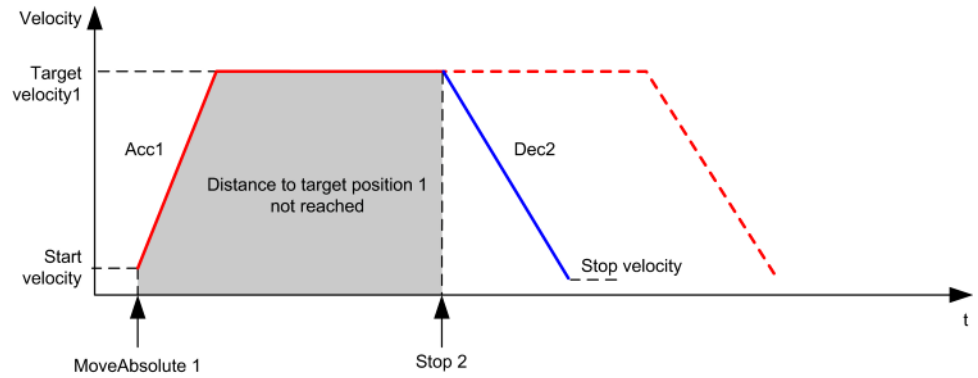
- Wenn der Funktionsbaustein im Zustand **Standstill** aufgerufen wird, ändert sich der Zustand zu **Stopping** und dann zurück zu **Standstill**, sobald *Execute* auf FALSE gesetzt ist.
- Der Status **Stopping** wird solange beibehalten, bis der Eingang *Execute* auf TRUE steht.
- Der Ausgang *Done* wird gesetzt, wenn die Halterampe abgeschlossen ist.
- Wenn der Wert für *Deceleration* 0 ist, wird die Schnellhalt-Verzögerung verwendet.
- Der Funktionsbaustein wird mit einer Geschwindigkeit von 0 beendet.
- Die Dauer der Verzögerung des Segmentbausteins darf 80 Sekunden nicht überschreiten.

## Beispiel-Zeitdiagramm

Die folgende Abbildung zeigt ein einfaches Profil aus dem Zustand **Continuous**:



Die folgende Abbildung zeigt ein einfaches Profil aus dem Zustand **Discrete**:



## MC\_Halt\_PTO Funktionsbaustein

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

### Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Execute</i>	FALSE	Startet die Ausführung des Funktionsbausteins an der steigenden Flanke. Die Werte der anderen Funktionsbausteineingänge steuern die Ausführung des Funktionsbausteins an der steigenden Flanke von <i>Execute</i> . Eine spätere Änderung an diesen Eingangsparametern wirkt sich nicht auf die aktuelle Ausführung aus.  Die Ausgänge werden festgelegt, wenn der Funktionsbaustein beendet wird.

In dieser Tabelle werden die Eingangsobjekte des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.
<i>Dec</i>	DINT	0	Verzögerung in Hz/ms Bereich (Hz/ms): 1... <i>MaxDecelerationAppl</i> , Seite 109
<i>JerkRatio</i>	INT	0	Prozentsatz der verwendeten Beschleunigungs-/Verzögerungsanpassung für die Erstellung des S-Kurven-Profiles, Seite 84. Bereich: 0...100
<i>BufferMode</i>	INT	<i>mcAborting</i>	Übergangsmodus aus der aktuellen Bewegung. Weitere Informationen finden Sie in der Tabelle zu Puffermodi, Seite 108.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Done</i>	FALSE	Wenn TRUE, wird die Ausführung des Funktionsbausteins ohne erkannte Fehler beendet. Wenn eine Bewegung auf einer Achse durch eine andere Bewegung auf derselben Achse unterbrochen wird, bevor die befohlene Aktion beendet ist, wird <i>CmdAborted</i> auf TRUE und <i>Done</i> auf FALSE gesetzt.
<i>Busy</i>	-	Wenn TRUE, wird der Funktionsbaustein ausgeführt. Wenn FALSE, wurde die Ausführung des Funktionsbausteins abgeschlossen. Der Funktionsbaustein muss mindestens so lange wie <i>Busy</i> = TRUE in einer aktiven Task des Anwendungsprogramms belassen werden.
<i>Active</i>	-	Wenn TRUE, hat die Funktionsbausteininstanz Kontrolle über die Achse. Es kann jeweils nur ein Funktionsbaustein <i>Active</i> für eine Achse auf TRUE setzen.
<i>CmdAborted</i>	-	Wenn TRUE, wird die Ausführung des Funktionsbausteins aufgrund eines anderen Bewegungsbefehls beendet.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

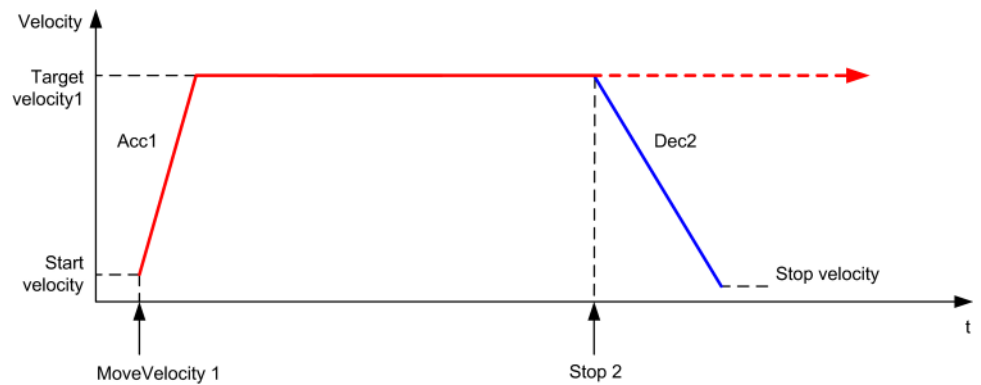
In dieser Tabelle wird das Ausgangsobjekt des Funktionsbausteins beschrieben:

Ausgangsobjekt	Typ	Initialwert	Beschreibung
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

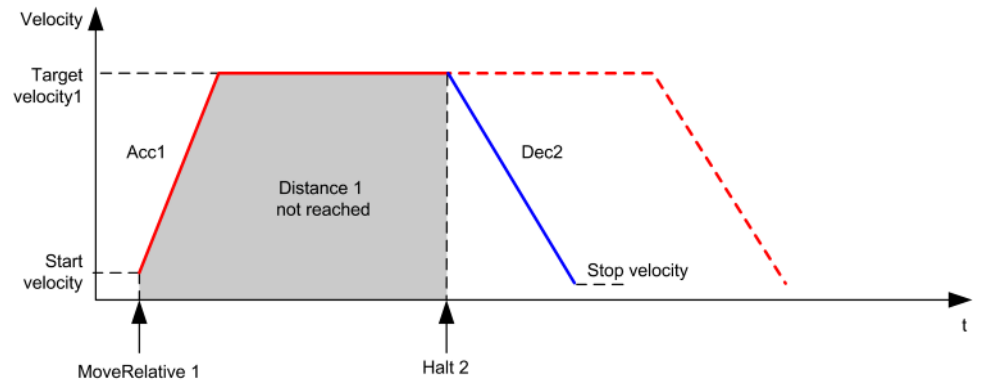
**HINWEIS:** Der Funktionsbaustein wird mit einer Geschwindigkeit von 0 beendet.

## Beispiel-Zeitdiagramm

Die folgende Abbildung zeigt ein einfaches Profil aus dem Zustand **Continuous**:



Die folgende Abbildung zeigt ein einfaches Profil aus dem Zustand **Discrete**:



## Administrative Funktionsbausteine

### Übersicht

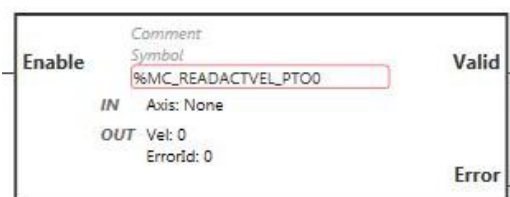
In diesem Kapitel werden die **Administrativen** Funktionsbausteine beschrieben.

### MC\_ReadActVel\_PTO Funktionsbaustein

#### Beschreibung der Funktion

Dieser Funktionsbaustein gibt den Wert der tatsächlichen Geschwindigkeit der Achse zurück.

#### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

## Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Enable</i>	FALSE	Wenn TRUE, wird der Funktionsbaustein ausgeführt. Die Werte der anderen Funktionsbausteineingänge können kontinuierlich geändert werden, wobei die Ausgänge dann kontinuierlich aktualisiert werden.  Wenn FALSE, wird die Ausführung des Funktionsbausteins beendet, und die Ausgänge werden zurückgesetzt.

In dieser Tabelle wird das Eingangsobjekt des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Valid</i>	-	Wenn TRUE, sind die Objektdaten des Funktionsbausteins gültig.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle werden die Ausgangsobjekte des Funktionsbausteins beschrieben:

Ausgangsobjekt	Typ	Initialwert	Beschreibung
<i>Vel</i>	DINT	-	Geschwindigkeit der Achse.
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

# MC\_ReadActPos\_PTO Funktionsbaustein

## Beschreibung der Funktion

Dieser Funktionsbaustein gibt den Wert der tatsächlichen Position der Achse zurück.

## Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

## Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Enable</i>	FALSE	Wenn TRUE, wird der Funktionsbaustein ausgeführt. Die Werte der anderen Funktionsbausteineingänge können kontinuierlich geändert werden, wobei die Ausgänge dann kontinuierlich aktualisiert werden.  Wenn FALSE, wird die Ausführung des Funktionsbausteins beendet, und die Ausgänge werden zurückgesetzt.

In dieser Tabelle wird das Eingangsobjekt des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Valid</i>	-	Wenn TRUE, sind die Objektdaten des Funktionsbausteins gültig.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle werden die Ausgangsobjekte des Funktionsbausteins beschrieben:

Ausgangs-objekt	Typ	Initialwert	Beschreibung
<i>Pos</i>	DINT	-	Position der Achse.
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

## MC\_ReadSts\_PTO Funktionsbaustein

### Beschreibung der Funktion

Dieser Funktionsbaustein gibt den Status des Zustandsdiagramms der Achse zurück.

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

### Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Enable</i>	FALSE	Wenn TRUE, wird der Funktionsbaustein ausgeführt. Die Werte der anderen Funktionsbausteineingänge können kontinuierlich geändert werden, wobei die Ausgänge dann kontinuierlich aktualisiert werden.  Wenn FALSE, wird die Ausführung des Funktionsbausteins beendet, und die Ausgänge werden zurückgesetzt.

In dieser Tabelle wird das Eingangsobjekt des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Valid</i>	-	Wenn TRUE, sind die Objektdaten des Funktionsbausteins gültig.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.
<i>IsHomed</i>	FALSE	Wenn TRUE, wurde die Achse auf Pos 1 gesetzt, sodass der absolute Referenzpunkt gültig ist und absolute Bewegungsbefehle zulässig sind.
<i>AxisWarning</i>	FALSE	Wenn TRUE, wurde durch einen Bewegungsbefehl ein Fehler oder Warnhinweis ausgegeben. Verwenden Sie den <i>MC_ReadAxisError_PTO</i> -Funktionsbaustein, um weitere Informationen zu erhalten., Seite 147
<i>QueueFull</i>	FALSE	Wenn TRUE, ist die Bewegungswarteschlange voll, und es sind keine weiteren gepufferten Bewegungsbefehle möglich.

In dieser Tabelle werden die Ausgangsobjekte des Funktionsbausteins beschrieben:

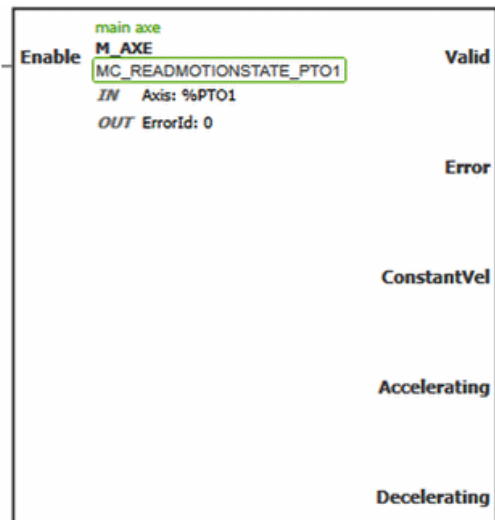
Ausgangsobjekt	Typ	Initialwert	Beschreibung
<i>AxisState</i>	-	-	Code für den Achsenstatus: 0 = Achse nicht konfiguriert 1 = ErrorStop 2 = Deaktiviert 4 = Stoppt 8 = Homing 16 = Standstill 32 = Diskrete Bewegung 64 = Kontinuierliche Bewegung Weitere Informationen finden Sie in der Statusbeschreibungstabelle, Seite 112.
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

## MC\_ReadMotionState\_PTO Funktionsbaustein

### Beschreibung der Funktion

Dieser Funktionsbaustein gibt den Wert des aktuellen Bewegungsstatus der Achse zurück.

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

### Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Enable</i>	FALSE	Wenn TRUE, wird der Funktionsbaustein ausgeführt. Die Werte der anderen Funktionsbausteineingänge können kontinuierlich geändert werden, wobei die Ausgänge dann kontinuierlich aktualisiert werden.  Wenn FALSE, wird die Ausführung des Funktionsbausteins beendet, und die Ausgänge werden zurückgesetzt.

In dieser Tabelle wird das Eingangsobjekt des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Valid</i>	-	Wenn TRUE, sind die Objektdaten des Funktionsbausteins gültig.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.
<i>ConstantVel</i>	-	Wenn TRUE, ist die Geschwindigkeit der Achse konstant.
<i>Accelerating</i>	-	Wenn TRUE, steigt die Geschwindigkeit der Achse an.
<i>Decelerating</i>	-	Wenn TRUE, verringert sich die Geschwindigkeit der Achse.

In dieser Tabelle wird das Ausgangsobjekt des Funktionsbausteins beschrieben:

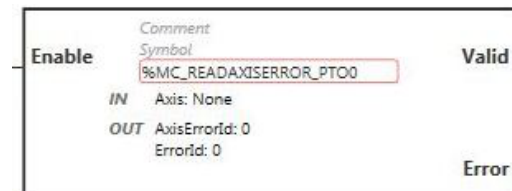
Ausgangsobjekt	Typ	Initialwert	Beschreibung
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

## MC\_ReadAxisError\_PTO Funktionsbaustein

### Beschreibung der Funktion

Dieser Funktionsbaustein empfängt den Achsensteuerungsfehler. Wenn kein Achsensteuerungsfehler vorliegt, gibt der Funktionsbaustein *AxisErrorId* = 0 aus.

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

## Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Enable</i>	FALSE	Wenn TRUE, wird der Funktionsbaustein ausgeführt. Die Werte der anderen Funktionsbausteineingänge können kontinuierlich geändert werden, wobei die Ausgänge dann kontinuierlich aktualisiert werden.  Wenn FALSE, wird die Ausführung des Funktionsbausteins beendet, und die Ausgänge werden zurückgesetzt.

In dieser Tabelle wird das Eingangsobjekt des Funktionsbausteins beschrieben:

Eingangs-objekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Valid</i>	-	Wenn TRUE, sind die Objektdaten des Funktionsbausteins gültig.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle werden die Ausgangsobjekte des Funktionsbausteins beschrieben:

Ausgangs-objekt	Typ	Initialwert	Beschreibung
<i>AxisErrorId</i>	-	-	Achsenfehlercodes, gültig wenn der Ausgang <i>AxisWarning</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Achsenfehlercodes, Seite 110.
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

## MC\_Reset\_PTO Funktionsbaustein

### Verhalten

Dieser Funktionsbaustein setzt alle achsenbezogenen Fehler zurück, falls die Bedingungen dies zulassen, um einen Übergang vom Status **ErrorStop** in **Standstill** zu erreichen. Dieser Vorgang hat keinerlei Auswirkungen auf die Instanzen der Funktionsblöcke.

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

### Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Execute</i>	FALSE	Startet die Ausführung des Funktionsbausteins an der steigenden Flanke. Die Werte der anderen Funktionsbausteineingänge steuern die Ausführung des Funktionsbausteins an der steigenden Flanke von <i>Execute</i> . Eine spätere Änderung an diesen Eingangsparametern wirkt sich nicht auf die aktuelle Ausführung aus.  Die Ausgänge werden festgelegt, wenn der Funktionsbaustein beendet wird.

In dieser Tabelle wird das Eingangsobjekt des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.

### Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Done</i>	FALSE	Wenn TRUE, wird die Ausführung des Funktionsbausteins ohne erkannte Fehler beendet.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle wird das Ausgangsobjekt des Funktionsbausteins beschrieben:

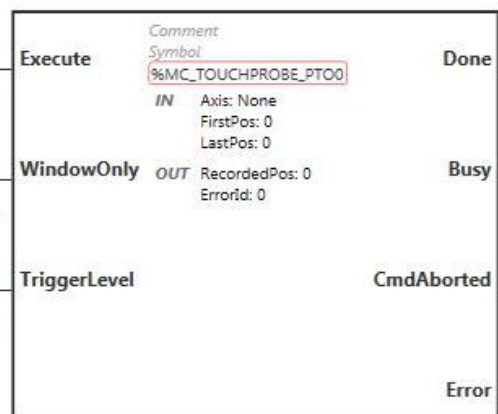
Ausgangs-objekt	Typ	Initialwert	Beschreibung
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

## MC\_TouchProbe\_PTO Funktionsbaustein

### Beschreibung der Funktion

Dieser Funktionsbaustein ermöglicht das Aktivieren eines auslösenden Ereignisses am Probe-Eingang. Das auslösende Ereignis ermöglicht die Aufzeichnung der Achsenposition und das Starten einer gepufferten Bewegung.

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

## Eingänge

In dieser Tabelle werden die Eingänge des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Execute</i>	FALSE	<p>Startet die Ausführung des Funktionsbausteins an der steigenden Flanke. Die Werte der anderen Funktionsbausteineingänge steuern die Ausführung des Funktionsbausteins an der steigenden Flanke von <i>Execute</i>. Eine spätere Änderung an diesen Eingangsparametern wirkt sich nicht auf die aktuelle Ausführung aus.</p> <p>Die Ausgänge werden festgelegt, wenn der Funktionsbaustein beendet wird.</p> <p>Wenn während der Ausführung des Funktionsbausteins eine zweite steigende Flanke erkannt wird, wird die aktuelle Ausführung abgebrochen und der Funktionsbaustein wird erneut ausgeführt.</p> <p>Wenn der Eingang <i>Execute</i> anschließend auf 0 gesetzt wird, dann wird die Achsenposition aufgezeichnet und der Ausgang <i>Done</i> wird für einen MAST-Zyklus auf 1 gesetzt. Die Achsenposition wird anschließend zurückgesetzt und der Ausgang <i>Done</i> wird auf 0 gesetzt.</p>
<i>WindowOnly</i>	FALSE	Wenn TRUE, wird ein Trigger-Ereignis nur innerhalb des Positionsbereichs (Fenster) erkannt, der durch <i>FirstPosition</i> und <i>LastPosition</i> definiert ist.
<i>TriggerLevel</i>	FALSE	<p>Wenn TRUE, wird die Position an der steigenden Flanke erfasst oder ein Ereignis ausgelöst.</p> <p>Wenn FALSE, wird die Position an der fallenden Flanke erfasst oder ein Ereignis ausgelöst.</p>

In dieser Tabelle werden die Eingangsobjekte des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.
<i>FirstPos</i>	DINT	0	Start der absoluten Position, von der aus auslösende Ereignisse akzeptiert werden (Wert im Aktivierungsfenster).
<i>LastPos</i>	DINT	0	Ende der absoluten Position, von der aus auslösende Ereignisse akzeptiert werden (Wert im Aktivierungsfenster).

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Done</i>	FALSE	Wenn TRUE, wird die Ausführung des Funktionsbausteins ohne erkannte Fehler beendet.  Wenn eine Bewegung auf einer Achse durch eine andere Bewegung auf derselben Achse unterbrochen wird, bevor die befohlene Aktion beendet ist, wird <i>CmdAborted</i> auf TRUE und <i>Done</i> auf FALSE gesetzt.
<i>Busy</i>	-	Wenn TRUE, wird der Funktionsbaustein ausgeführt.  Wenn FALSE, wurde die Ausführung des Funktionsbausteins abgeschlossen.  Der Funktionsbaustein muss mindestens so lange wie <i>Busy</i> = TRUE in einer aktiven Task des Anwendungsprogramms belassen werden.
<i>CmdAborted</i>	-	Wenn TRUE, wird die Ausführung des Funktionsbausteins aufgrund eines anderen Bewegungsbefehls beendet.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle werden die Ausgangsobjekte des Funktionsbausteins beschrieben:

Ausgangs-objekt	Typ	Initialwert	Beschreibung
<i>RecordedPos</i>	-	-	Position, an der das auslösende Ereignis erkannt wurde.
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

### HINWEIS:

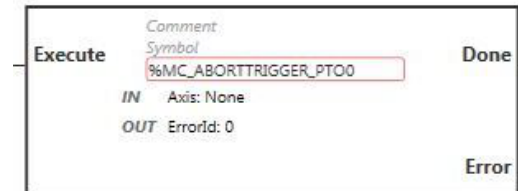
- Es ist nur eine Instanz dieses Funktionsbausteins auf derselben Achse zulässig.
- Nur das erste Ereignis nach einer steigenden Flanke am *Busy*-Ausgang des Funktionsbausteins *MC\_TouchProbe\_PTO* ist gültig. Wenn der *Done*-Ausgang auf TRUE gesetzt ist, werden die nachfolgenden Ereignisse ignoriert. Der Funktionsbaustein muss neu aktiviert werden, um auf andere Ereignisse reagieren zu können.

## MC\_AbortTrigger\_PTO Funktionsbaustein

### Beschreibung der Funktion

Dieser Funktionsbaustein ermöglicht, die Ausführung anderer Funktionsbausteine anzuhalten, wenn diese mit auslösenden Ereignissen verbunden sind (beispielsweise *MC\_TouchProbe\_PTO*).

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

### Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Execute</i>	FALSE	Startet die Ausführung des Funktionsbausteins an der steigenden Flanke. Die Werte der anderen Funktionsbausteineingänge steuern die Ausführung des Funktionsbausteins an der steigenden Flanke von <i>Execute</i> . Eine spätere Änderung an diesen Eingangsparametern wirkt sich nicht auf die aktuelle Ausführung aus.  Die Ausgänge werden festgelegt, wenn der Funktionsbaustein beendet wird.

In dieser Tabelle wird das Eingangsobjekt des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.

### Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Done</i>	FALSE	Wenn TRUE, wird die Ausführung des Funktionsbausteins ohne erkannte Fehler beendet.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle wird das Ausgangsobjekt des Funktionsbausteins beschrieben:

Ausgangsobjekt	Typ	Initialwert	Beschreibung
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

## MC\_ReadPar\_PTO Funktionsbaustein

### Beschreibung der Funktion

Dieser Funktionsbaustein ermöglicht das Abrufen beliebiger Parameter aus der PTO.

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

### Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Enable</i>	FALSE	Wenn TRUE, wird der Funktionsbaustein ausgeführt. Die Werte der anderen Funktionsbausteineingänge können kontinuierlich geändert werden, wobei die Ausgänge dann kontinuierlich aktualisiert werden.  Wenn FALSE, wird die Ausführung des Funktionsbausteins beendet, und die Ausgänge werden zurückgesetzt.

In dieser Tabelle werden die Eingangsobjekte des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.
<i>ParNumber</i>	DINT	0	Code für den Parameter, den Sie lesen oder schreiben möchten. Weitere Informationen finden Sie in der PTO-Parametertabelle, Seite 109.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Valid</i>	-	Wenn TRUE, sind die Objektdaten des Funktionsbausteins gültig.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle werden die Ausgangsobjekte des Funktionsbausteins beschrieben:

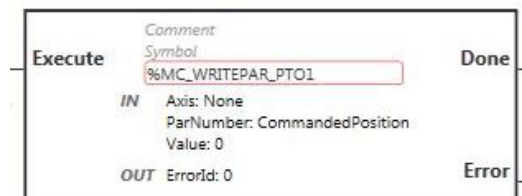
Ausgangs-objekt	Typ	Initialwert	Beschreibung
<i>Value</i>	DINT	0	Der Wert des angeforderten Parameters.
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.

## MC\_WritePar\_PTO Funktionsbaustein

### Beschreibung der Funktion

Dieser Funktionsbaustein ermöglicht das Schreiben von Parameter aus der PTO.

### Grafische Darstellung



**HINWEIS:** Wenn Sie den Funktionsbaustein zum ersten Mal verwenden, müssen Sie ihn so konfigurieren, dass er die gewünschte Achse nutzt. Doppelklicken Sie auf den Funktionsbaustein, um seine Eigenschaften anzuzeigen, wählen Sie die Achse aus, und klicken Sie auf **Übernehmen**.

### Eingänge

In dieser Tabelle wird der Eingang des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<i>Execute</i>	FALSE	Startet die Ausführung des Funktionsbausteins an der steigenden Flanke. Die Werte der anderen Funktionsbausteineingänge steuern die Ausführung des Funktionsbausteins an der steigenden Flanke von <i>Execute</i> . Eine spätere Änderung an diesen Eingangsparametern wirkt sich nicht auf die aktuelle Ausführung aus.  Die Ausgänge werden festgelegt, wenn der Funktionsbaustein beendet wird.

In dieser Tabelle werden die Eingangsobjekte des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<i>Axis</i>	PTOx	-	Instanz, für die der Funktionsbaustein ausgeführt wird. Der Name steht in der Steuerungskonfiguration.
<i>ParNumber</i>	DINT	0	Code für den Parameter, den Sie lesen oder schreiben möchten. Weitere Informationen finden Sie in der PTO-Parametertabelle, Seite 109.
<i>Value</i>	DINT	0	Wert, der in den Parameter geschrieben wird, der für das Eingangsobjekt <i>ParNumber</i> ausgewählt ist.

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<i>Done</i>	FALSE	Wenn TRUE, wird die Ausführung des Funktionsbausteins ohne erkannte Fehler beendet.
<i>Error</i>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle wird das Ausgangsobjekt des Funktionsbausteins beschrieben:

Ausgangsobjekt	Typ	Initialwert	Beschreibung
<i>ErrorId</i>	Word	<i>NoError</i>	Bewegungsfehlercodes, gültig wenn der Ausgang <i>Error</i> TRUE ist. Weitere Informationen finden Sie in der Tabelle mit PTO-Bewegungsfehlercodes, Seite 111.


# Frequenzgenerator (%FREQGEN)

## Inhalt dieses Kapitels

Beschreibung..... 157  
 Konfiguration ..... 159

## Beschreibung

### Einführung

Der Funktionsbaustein  Frequenzgenerator *FREQGEN* steuert einen Rechteckwellen-Signalausgang mit der angegebenen Frequenz.

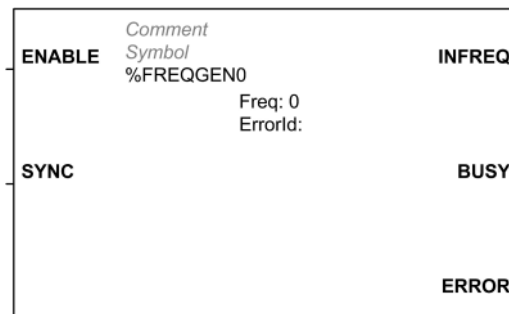
Frequenz, konfigurierbar von 0 Hz bis 100 kHz in Schritten von 1 Hz.

Die *FREQGEN*-Funktion hat die folgenden Leistungsmerkmale:

Merkmal	Wert
Anzahl der Kanäle	2 oder 4, abhängig von der Referenz
Minimale Frequenz	0 Hz
Maximale Frequenz	100 kHz
Genauigkeit der Frequenz	1 %

## Abbildung

Die folgende Abbildung zeigt einen Funktionsbaustein *FREQGEN*:



## Eingänge

In dieser Tabelle werden die Eingänge des Funktionsbausteins beschrieben:

Eingang	Initialwert	Beschreibung
<b>ENABLE</b>	FALSE	Wenn TRUE, dann wird der Funktionsbaustein ausgeführt. Die Werte der anderen Funktionsbausteineingänge können kontinuierlich geändert werden, wobei die Ausgänge dann kontinuierlich aktualisiert werden.  Wenn FALSE, dann wird die Ausführung des Funktionsbausteins beendet, und die Ausgänge werden zurückgesetzt.
<b>SYNC</b>	FALSE	Wenn eine steigende Flanke festgestellt wird, wird die Zielfrequenz ausgegeben, ohne auf ein Ende der aktuellen Ausgangsperiode zu warten.

In dieser Tabelle wird das Eingangsobjekt des Funktionsbausteins beschrieben:

Eingangsobjekt	Typ	Initialwert	Beschreibung
<b>Freq</b>	DWORD	-	Frequenz des Ausgangssignals des <i>Frequency Generator</i> in Hz.  Spezifizieren der Frequenz in der Tabelle <i>Eigenschaften der Impulsgeneratoren</i> , Seite 159  (Bereich: Minimum 0 (0 Hz)...Maximum 100000 (100 kHz))

## Ausgänge

In dieser Tabelle werden die Ausgänge des Funktionsbausteins beschrieben:

Ausgang	Initialwert	Beschreibung
<b>INFREQ</b>	-	Wenn TRUE, wird das Signal des Frequenzgenerators in der Frequenz ausgegeben, die im <b>Freq</b> -Eingangsobjekt festgelegt ist.
<b>BUSY</b>	-	Wenn TRUE, wird der Funktionsbaustein ausgeführt.  Wenn FALSE, wurde die Ausführung des Funktionsbausteins abgeschlossen.  Der Funktionsbaustein muss mindestens so lange wie <b>BUSY</b> = TRUE in einer aktiven Task des Anwendungsprogramms belassen werden.
<b>ERROR</b>	FALSE	TRUE gibt an, dass ein Fehler erkannt wurde. Die Ausführung des Funktionsbausteins ist beendet.

In dieser Tabelle wird das Ausgangsobjekt des Funktionsbausteins beschrieben:

Ausgangsobjekt	Typ	Initialwert	Beschreibung
<b>ErrorId</b>	Word	<i>NoError</i>	Fehlercodes, gültig wenn der Ausgang <b>ERROR</b> TRUE ist. Weitere Informationen finden Sie in der nachfolgenden Tabelle mit <b>ErrorId-Fehlercodes</b> .

## FehlerId Fehlercode

In dieser Tabelle werden die Werte für die Fehlercodes der Funktionsbausteine aufgelistet.

Name	Wert	Beschreibung
<i>NoError</i>	0	Keine Fehler erkannt.
<i>OutputProtection</i>	1007	Der Impulsausgang verfügt über einen aktiven Schutz für Digitalausgänge. Weitere Informationen finden Sie in den Systemobjekten %S10 und %SW139 (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<i>OutputReset</i>	1008	%S9 hat alle Ausgänge auf 0 forciert. Siehe Systembits (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<i>InvalidFrequencyValue</i>	3002	Das Eingangsobjekt Frequenz <b>Freq</b> ist außerhalb des erlaubten Bereichs.

# Konfiguration

## Überblick

Siehe Konfigurieren von Impulsgeneratoren (siehe Modicon M221, Logic Controller, Programmierhandbuch) zur Konfiguration der Ressource des *Pulse Generator*.

Siehe Konfigurieren des Impulses (siehe Modicon M221, Logic Controller, Programmierhandbuch) zur Konfiguration der Ressource des *Pulse Generator* als eine *FREQGEN*.

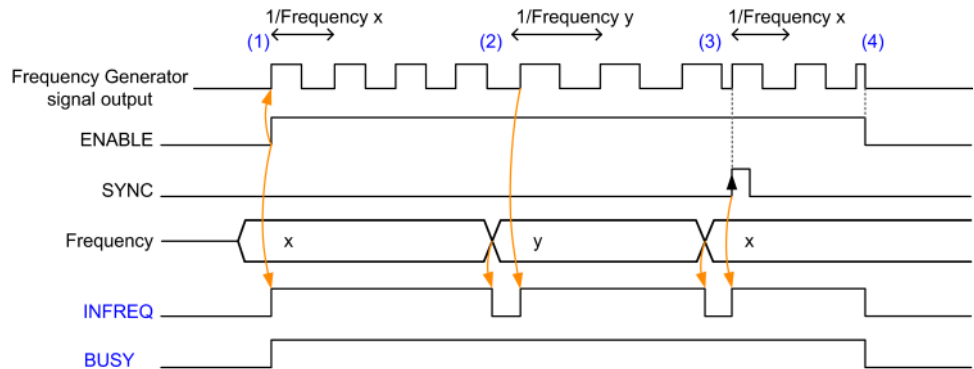
## Eigenschaften

Der *FREQGEN*-Funktionsbaustein hat die folgenden Eigenschaften:

Eigenschaft	Beschreibung	Wert
<b>Verwendet</b>	Adresse verwendet	Wenn diese Option ausgewählt ist, wird die Adresse in einem Programm verwendet.
<b>Adresse</b>	%FREQGENi Frequenzgeneratoradresse	Die Instanz-ID, wobei i einer Zahl zwischen 0 und der Gesamtanzahl der mit der Steuerung verfügbaren Objekte entspricht. Die maximale Anzahl an <i>FREQGEN</i> -Objekten finden Sie in der Tabelle Maximale Anzahl Objekte (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<b>Symbol</b>	Symbol	Das diesem Objekt zugeordnete Symbol. Weitere Informationen finden Sie unter Definieren und Verwenden von Symbolen (siehe EcoStruxure Machine Expert - Basic, Betriebshandbuch).
<b>Freq</b>	Frequenz	Frequenz des Ausgangssignals des Frequenzgenerators in Hz.  Minimalwert: 0 (0 Hz). Maximaler Wert: 100000 (100 kHz).  Der Standardwert beträgt 0.
<b>Kommentar</b>	Kommentar	Diesem Objekt kann ein optionaler Kommentar zugeordnet werden.  Doppelklicken Sie in die Spalte <b>Kommentar</b> und geben Sie einen Kommentar ein.

## Zeitdiagramm

Dieses Diagramm veranschaulicht den Zeitablauf für den Funktionsbaustein *FREQGEN*:



**(1)** Der *ENABLE*-Eingang wird auf 1 gesetzt. Das Frequenzgeneratorsignal wird am dedizierten Ausgang generiert. Der *INFREQ*-Ausgang wird auf 1 gesetzt. Der *BUSY*-Ausgang wird auf 1 gesetzt.

**(2)** Der Frequenzwert wird geändert. Der *INFREQ*-Ausgang wird auf 0 gesetzt bis die neue Frequenz am dedizierten Ausgang generiert wird. Der *BUSY*-Ausgang bleibt auf 1 gesetzt.

**(3)** Der *SYNC*-Eingang wird auf 1 gesetzt. Der aktuelle Frequenzgeneratorzyklus stoppt und ein neuer Zyklus wird gestartet. Der *INFREQ*-Ausgang wird auf 1 gesetzt. Der *BUSY*-Ausgang bleibt auf 1 gesetzt.

**(4)** Der *ENABLE*-Eingang wird auf 0 gesetzt. Der Frequenzgenerator stoppt. Der *INFREQ*-Ausgang wird auf 0 gesetzt. Der *BUSY*-Ausgang wird auf 0 gesetzt.

Wenn die Anwendung gestoppt ist, stoppt der Frequenzgenerator ohne auf das Ende des Impulsgenerierungszyklus zu warten. Der *Error*-Ausgang bleibt *FALSE*.

Wenn ein Fehler festgestellt wird, wird er automatisch anerkannt, wenn die Fehlerbedingung verlassen wird.

---

# Erweiterte Softwarefunktion

## Inhalt dieses Abschnitts

PID-Funktion.....	162
-------------------	-----

## Überblick

In diesem Abschnitt wird die PID-Funktion beschrieben.

# PID-Funktion

## Inhalt dieses Kapitels

PID-Betriebsarten.....	162
PID-Auto-Tuning-Konfiguration.....	163
PID-Standardkonfiguration.....	166
PID-Assistent.....	176
PID-Programmierung.....	185

## PID-Betriebsarten

### PID-Betriebsarten

#### Einführung

Der EcoStruxure Machine Expert - Basic *PID*-Regler stellt 4 separate Betriebsarten zur Auswahl, die auf der Registerkarte **Allgemein**, Seite 177 im **PID-Assistenten** von EcoStruxure Machine Expert - Basic konfiguriert werden können.

Folgende *PID*-Betriebsarten sind verfügbar:

- PID-Modus
- AT + PID-Modus
- AT-Modus
- Wortadresse

#### PID-Modus

Der einfache *PID*-Reglermodus ist standardmäßig aktiv, wenn der *PID*-Regler eingeschaltet wird. Die auf der Registerkarte **PID**, Seite 180 einzustellenden Verstärkungswerte  $K_p$ ,  $T_i$  und  $T_d$  müssen im Vorfeld bekannt sein, damit der Prozess erfolgreich gesteuert werden kann. Sie können den Korrekturtyp des Reglers (PID oder PI) auf der Registerkarte **PID** im Fenster, Seite 176 des **PID-Assistenten** auswählen. Bei Auswahl des PI-Korrekturtyps wird das Feld **Td** mit der Differentialzeit deaktiviert.

Wenn der *PID*-Modus verwendet wird, wird die Auto-Tuning-Funktion deaktiviert, d. h. die Registerkarte **AT**, Seite 182 im Konfigurationsfenster des **PID-Assistenten** ist nicht verfügbar.

#### AT + PID-Modus

In diesem Modus ist die Auto-Tuning-Funktion aktiv, wenn der *PID*-Regler startet. Die Auto-Tuning-Funktion berechnet dann die Verstärkungswerte  $K_p$ ,  $T_i$  und  $T_d$ , Seite 180 und den Typ der PID-Aktion, Seite 183. Am Ende der Auto-Tuning-Sequenz schaltet die Steuerung in den *PID*-Modus für den angepassten Sollwert unter Verwendung der von der Auto-Tuning-Funktion berechneten Parameter.

Wenn der Auto-Tuning-Algorithmus einen Fehler, Seite 187 erkennt:

- Es wird kein *PID*-Parameter berechnet.
- Der Auto-Tuning-Ausgang wird auf den Ausgang eingestellt, der vor Start des Auto-Tuning-Vorgangs auf den Prozess angewendet wurde.
- Im Dropdown-Listefeld **Liste der PID-Zustände** wird eine Fehlermeldung angezeigt.
- Die *PID*-Regelung wird abgebrochen.

Im *AT + PID*-Modus erfolgt die Umschaltung von Auto-Tuning zu *PID* automatisch und übergangslos.

## AT-Modus

In diesem Modus ist die Auto-Tuning-Funktion aktiv, wenn der *PID*-Regler startet. Die Funktion berechnet dann automatisch sowohl die Verstärkungswerte  $K_p$ ,  $T_i$  und  $T_d$ , Seite 180 als auch den Typ der *PID*-Aktion, Seite 183. Nach der Konvergenz des Auto-Tuning-Prozesses und dem erfolgreichen Abschluss mit Bestimmung der Parameter  $K_p$ ,  $T_i$  und  $T_d$  und des Typs der *PID*-Aktion, Seite 183 (oder nach Erkennung eines Fehlers im Auto-Tuning-Algorithmus) wird der numerische Auto-Tuning-Ausgang auf 0 gesetzt und im Dropdown-Feld Liste der *PID*-Zustände, Seite 187 die Meldung **Auto-Tuning abgeschlossen** angezeigt. Daraufhin stoppt der *PID*-Regler und wartet. Die berechneten *PID*-Koeffizienten  $K_p$ ,  $T_i$  und  $T_d$  sind in den entsprechenden Speicherwörtern ( $\%MWx$ ) verfügbar.

## Wortadresse

Dieser *PID*-Modus wird durch Zuweisung des gewünschten Werts zu einer Wortadresse ausgewählt:

- $\%MWxx = 0$ : Der Regler wird deaktiviert.
- $\%MWxx = 1$ : Der Regler läuft im einfachen *PID*-Modus.
- $\%MWxx = 2$ : Der Regler läuft im *AT+ PID*-Modus.
- $\%MWxx = 3$ : Der Regler läuft im ausschließlichen *AT*-Modus.
- $\%MWxx = 4$ : Der Regler läuft im einfachen *PID*-Modus mit Korrektortyp PI.

Mit dem Modus *word address* können Sie die Betriebsart des *PID*-Reglers über die Anwendung verwalten und sie dadurch an Ihre spezifischen Anforderungen anpassen.

# PID-Auto-Tuning-Konfiguration

## Konfiguration der PID-Auto-Tuning-Funktion

### Einführung

Dieser Abschnitt enthält detaillierte Anweisungen zur Konfiguration der Steuerung EcoStruxure Machine Expert - Basic *PID* mit Auto-Tuning (*AT*).

Im weiteren Verlauf werden folgende Konfigurationsschritte beschrieben:

Schritt	Aspekt
1	Konfiguration des Analogkanals, Seite 163
2	Voraussetzungen für die <i>PID</i> -Konfiguration, Seite 164
3	Konfiguration der <i>PID</i> -Funktion, Seite 164
4	Einrichtung des Regelprozesses, Seite 165

### Schritt 1: Konfiguration des Analogkanals

Ein *PID*-Regler verwendet ein analoges Rückführungssignal (auch Prozessvariable genannt), um den zur Regelung des Prozesses benötigten Algorithmus zu berechnen. Die Steuerung verfügt über einen integrierten Analogeingang, der zur Erfassung der Prozessvariablen (Messwert) eingesetzt

werden kann. Detaillierte Informationen zur Konfiguration von Analogeingängen finden Sie im M221 Logic Controller - Programmierhandbuch.

Wenn für den Antrieb des zu regelnden Systems ein Analogeingang verwendet wird, müssen Sie sicherstellen, dass dieser Eingang ordnungsgemäß konfiguriert ist. Siehe das analoge Ausgangserweiterungsmodul Ihrer Steuerung.

## Schritt 2: Voraussetzungen für die PID-Konfiguration

Bevor Sie den *PID*-Regler konfigurieren, müssen Sie sicherstellen, dass folgende Phasen abgeschlossen wurden:

Phase	Beschreibung
1	Die PID-Funktion wurde im Programm aktiviert., Seite 185
2	Der <b>Abfragemodus</b> wurde auf <i>periodisch</i> , Seite 186 eingestellt.

## Schritt 3: Konfiguration der PID-Funktion

Verwenden Sie einen Festkörperausgang in Verbindung mit der PID-Funktion. Bei Verwendung eines Relaisausgangs ist die Lebensdauer des Ausgangs unter Umständen in kürzester Zeit überschritten, sodass das Relais aufgrund von in offener oder geschlossener Stellung blockierten Kontakten (eingefroren/gelötet) nicht mehr betriebsfähig ist.

### **▲ WARNUNG**

#### **UNBEABSICHTIGTER GERÄTEBETRIEB ODER FUNKTIONSunFÄHIGE GERÄTE**

- Verwenden Sie keinesfalls Relaisausgänge in Verbindung mit der PID-Funktion.
- Verwenden Sie ausschließlich Festkörperausgänge, wenn ein Digitalausgang für den Antrieb des zu steuernden Systems benötigt wird.

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

Gehen Sie zur Implementierung eines *PID*-Reglers mit Auto-Tuning vor wie folgt:

Schritt	Aktion
1	Wählen Sie auf der <b>General</b> tab, Seite 177 im Fenster <b>PID-Assistent</b> (im Offline-Modus) die Betriebsart <b>AT + PID</b> (oder <b>AT</b> ) bzw. die Einstellung <b>Wortadresse</b> aus und setzen Sie das zugehörige Wort aus den <i>Operating Modes</i> , Seite 162 auf 2 oder 3.
2	Aktivieren Sie das Kontrollkästchen <b>PID-Zustände</b> und geben Sie die Adresse des Speicherworts im Feld ein.
3	Geben Sie auf der <b>Input</b> tab, Seite 179 die Adresse des als Messwert verwendeten Analogeingangs an.
4	Sollten eine <b>Konvertierung</b> oder <b>Alarmer</b> erforderlich sein, siehe die Registerkarte <b>Eingang</b> , Seite 179 im Fenster <b>PID-Assistent</b> .
5	Geben Sie auf der Registerkarte <b>PID</b> , Seite 180 den Sollwert ein. Im Allgemeinen handelt es sich bei diesem Wert um eine Speicheradresse oder einen Analogeingang.
6	Der <b>Korrekturtyp</b> auf der Registerkarte <b>PID</b> muss auf <b>PID</b> oder <b>PI</b> eingestellt werden.
7	Stellen Sie die <b>Parameter</b> auf der Registerkarte <b>PID</b> ein: <b>Kp (x0,01)</b> , <b>Ti (x0,1s)</b> und <b>Td (x0,1s)</b> . Wenn <b>AT+PID</b> oder <b>AT</b> als Betriebsart, Seite 162 ausgewählt wurde, sollten die Parameter <b>Speicherwortadressen (%MWxx)</b> sein, damit der Auto-Tuning-Algorithmus automatisch den für die Parameter berechneten Wert einsetzt.
8	Geben Sie die <b>PID-Abtastperiode</b> ( <b>Ts</b> , Seite 169) für die-Funktion auf der Registerkarte <b>PID</b> ein. Die <b>Abtastperiode</b> ist ein zentraler Parameter und muss mit Bedacht festgelegt werden.
9	Auf der Registerkarte <b>AT</b> muss der <b>AT-Modus</b> standardmäßig auf <b>Genehmigen</b> eingestellt werden. Geben Sie die <b>Min.</b> - und <b>Max.</b> -Werte ein, wenn der <b>Messbereich</b> aktiviert ist (Kontrollkästchen <b>Genehmigen</b> ). Wählen Sie die <b>Dynamische AT-Korrektur</b> in der Liste aus. Verfügbare Korrekturtypen: <b>Schnell</b> , <b>Mittel</b> , <b>Langsam</b> und <b>Wortadresse</b> . Weitere Informationen hierzu finden Sie auf der Registerkarte <b>AT</b> im Fenster <b>PID-Assistent</b> , Seite 176.
10	Geben Sie auf der Registerkarte <b>AT</b> das <b>AT-Trigger</b> -Speicherbit ein, in dem der Wert des Schrittwechsels während des Auto-Tuning-Vorgangs gespeichert wird. Weitere Informationen finden Sie in Verbindung mit der Registerkarte <b>AT</b> im Fenster <b>PID-Assistent</b> , Seite 176.
11	Legen Sie auf der <b>Output</b> tab, Seite 183 die <b>Aktion</b> durch Auswahl in der Liste fest.  Bei der Auswahl von <b>Bitadresse</b> müssen Sie die Speicherbitadresse im Feld <b>Bit</b> angeben.  Bei Bedarf können Sie <b>Grenzwerte</b> konfigurieren.  Bei aktiviertem <b>manuellem Modus</b> muss ein Speicherwort oder Analogausgang ausgewählt werden. Bei der Auswahl von <b>Bitadresse</b> müssen Sie ein <b>Bit</b> eingeben. Detaillierte Informationen zur Funktionsweise des manuellen Modus finden Sie bei der <b>Output</b> tab, Seite 183.  Setzen Sie im Feld <b>Analogausgang</b> die Option <b>Output PWM</b> , Seite 183 auf <b>Genehmigen</b> .  Legen Sie einen Analogausgang oder ein Speicherwort als Adresse des Worts fest.  Geben Sie einen Wert in das Feld <b>Periode (x0,1 s)</b> ein und geben Sie das Speicherbit bzw. den Digitalausgang an.
12	Klicken Sie auf <b>OK</b> , um die Konfiguration des <i>PID</i> -Reglers zu bestätigen.

### Schritt 4: Einrichtung der Steuerung

Verwenden Sie einen Festkörperausgang in Verbindung mit der *PID*-Funktion. Bei Verwendung eines Relaisausgangs ist die Lebensdauer des Ausganges unter Umständen in kürzester Zeit überschritten, sodass das Relais aufgrund von in offener oder geschlossener Stellung blockierten Kontakten (eingefroren/gelötet) nicht mehr betriebsfähig ist.

<b>⚠️ WARNUNG</b>
<p><b>UNBEABSICHTIGTER GERÄTEBETRIEB ODER FUNKTIONSUNFÄHIGE GERÄTE</b></p> <ul style="list-style-type: none"> <li>• Verwenden Sie keinesfalls Relaisausgänge in Verbindung mit der <i>PID</i>-Funktion.</li> <li>• Verwenden Sie ausschließlich Festkörperausgänge, wenn ein Digitalausgang für den Antrieb des zu steuernden Systems benötigt wird.</li> </ul> <p><b>Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.</b></p>

Um den Betrieb im *AT+PID* Betriebsmodus, Seite 162 zu starten, führen Sie die folgenden Schritte aus:

Schritt	Aktion
1	Schließen Sie den PC an die Steuerung an und übertragen Sie die Anwendung.
2	Umschalten der Steuerung in den Zustand RUNNING.

**HINWEIS:** Bevor Sie die Steuerung in den RUNNING-Zustand setzen, müssen Sie sicherstellen, dass die Betriebsbedingungen der Maschine einen RUNNING-Zustand für die restliche Anwendung zulassen.

Schritt	Aktion
1	Erstellen Sie eine Animationstabelle mit den bei der Konfiguration definierten Objekten. Detaillierte Informationen zur Erstellung von Animationstabellen finden Sie im <i>Ecostruxure Machine Expert - Basic Betriebshandbuch</i> .
2	Überprüfen Sie die Kohärenz zwischen der gemessenen Prozessvariablen und den Werten der Anwendung. Dieser Test ist von grundlegender Bedeutung, da ein erfolgreicher Betrieb des <i>PID</i> -Reglers von der Genauigkeit der Messung abhängig ist. Wenn Sie Zweifel bezüglich der Genauigkeit der Messwerte haben, setzen Sie die Steuerung in den STOP-Zustand und überprüfen Sie die Verdrahtung der Analogkanäle.  Wenn das Stellglied nicht gesteuert wird: <ul style="list-style-type: none"> <li>• Überprüfen Sie für einen analogen Ausgang die Ausgangsspannung bzw. den Ausgangsstrom vom Analogkanal.</li> <li>• Überprüfen Sie für einen PWM-Ausgang Folgendes: <ul style="list-style-type: none"> <li>◦ Die LED des dedizierten Ausgangs leuchtet.</li> <li>◦ Die Verdrahtung der Spannungsversorgungen und der 0-V-Schaltung.</li> <li>◦ Das Stellglied wird mit Spannung versorgt.</li> </ul> </li> </ul>
3	Überprüfen Sie in der Animationstabelle Folgendes: <ul style="list-style-type: none"> <li>• Der Ausgangsmodus ist auf „Automatisch“ eingestellt.</li> <li>• Alle Parameter, die für Ihre Anwendung erforderlich sind, sind auf angemessene Werte eingestellt.</li> </ul>
4	Stellen Sie die Zykluszeit des Logic Controllers so ein, dass die <b>Abtastperiode</b> ( $T_s$ ) genau einem Vielfachen der Zykluszeit des <i>PID</i> -Reglers entspricht. Detaillierte Informationen zur Ermittlung der Abtastperiode finden Sie unter Einstellung der <i>PID</i> -Funktion, Seite 169.
5	Nach Abschluss der Auto-Tuning-Sequenz werden die Parameter <b>K<sub>p</sub></b> , <b>T<sub>i</sub></b> und <b>T<sub>d</sub></b> in den RAM-Speicher des Logic Controllers geschrieben. Die Werte werden gespeichert, solange die Anwendung gültig ist (Spannungsunterbrechung kürzer als 30 Tage) und kein Kaltstart durchgeführt wird.

Der Auto-Tuning-Prozess wird jedes Mal neu ausgeführt, wenn am Speicherbit **AT-Trigger** eine steigende Flanke erkannt wird.

**HINWEIS:** Wenn *PID*-Auto-Tuning gerade eine Kalibrierung durchführt, um die neuen Parameter für **K<sub>p</sub>**, **T<sub>i</sub>** und **T<sub>d</sub>** zu finden, und die manuelle Ausgangskontrolle aktiviert ist, müssen Sie nach dem Ende der manuellen Ausgangskontrolle *PID*-Auto-Tuning neu starten, damit die Parameter aktualisiert werden.

## PID-Standardkonfiguration

## PID-Konfiguration über eine Wortadresse

### Einführung

In diesem Abschnitt werden Sie durch alle Schritte zur Konfiguration des *EcoStruxure Machine Expert - Basic PID*-Reglers mithilfe der Betriebsart, Seite 162 **Wortadresse** geführt. Diese Betriebsart bietet größere Flexibilität bei der Verwendung als die anderen *PID*-Modi.

Dieser Abschnitt enthält die folgenden Schritte:

Schritt	Aspekt
1	Voraussetzungen für die PID-Konfiguration, Seite 167
2	Konfiguration der PID-Funktion, Seite 167
3	Einrichtung des Regelprozesses, Seite 168

## Schritt 1: Voraussetzungen für die PID-Konfiguration

Stellen Sie vor der Konfiguration des *PID* sicher, dass die folgenden Phasen ausgeführt worden sind:

Phase	Beschreibung
1	Ein Analogeingang wurde konfiguriert sowie nach Bedarf auch ein Analogausgang. Weitere Informationen finden Sie im M221 Logic Controller - Programmierhandbuch.
2	Die PID-Funktion wurde im Programm aktiviert., Seite 185
3	Der Abfragemodus wurde auf <i>periodisch</i> , Seite 186 eingestellt.

## Schritt 2: Konfiguration der PID-Funktion

Verwenden Sie einen Festkörperausgang in Verbindung mit der PID-Funktion. Bei Verwendung eines Relaisausgangs ist die Lebensdauer des Ausgangs unter Umständen in kürzester Zeit überschritten, sodass das Relais aufgrund von in offener oder geschlossener Stellung blockierten Kontakten (eingefroren/gelötet) nicht mehr betriebsfähig ist.

⚠ <b>WARNUNG</b>
<p><b>UNBEABSICHTIGTER GERÄTEBETRIEB ODER FUNKTIONSUNFÄHIGE GERÄTE</b></p> <ul style="list-style-type: none"> <li>• Verwenden Sie keinesfalls Relaisausgänge in Verbindung mit der PID-Funktion.</li> <li>• Verwenden Sie ausschließlich Festkörperausgänge, wenn ein Digitalausgang für den Antrieb des zu steuernden Systems benötigt wird.</li> </ul> <p><b>Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.</b></p>

In den folgenden Schritten wird die Implementierung eines *PID*-Reglers im Modus **Wortadresse** erläutert. Detaillierte Informationen zur Konfiguration der *PID*-Funktion finden Sie im **Abschnitt zum PID-Assistenten**.

Geben Sie für die dynamische Änderung der *PID*-Parameter (im Offline- und Online-Modus) die Speicheradressen in die zugehörigen Felder ein. Dadurch muss für betriebsbegleitende Werteänderungen nicht in den Offline-Modus geschaltet werden.

Schritt	Aktion
1	Wählen Sie auf der Registerkarte <b>Allgemein</b> im Fenster <b>PID-Assistent</b> (im Offline-Modus) die Option <b>Wortadresse</b> im Dropdown-Listefeld <b>Betriebsarten</b> aus. Aktivieren Sie das Kontrollkästchen der PID-Zustände und geben Sie die Adresse des Speicherworts in das Feld ein.
2	Geben Sie auf der Registerkarte <b>Eingang</b> , Seite 179 die Adresse des als Messwert verwendeten Analogeingangs ein. Sollten eine <b>Konvertierung</b> oder <b>Alarmerfordernis</b> erforderlich sein, siehe die Registerkarte <b>Eingang</b> , Seite 179 im Fenster <b>PID-Assistent</b> , Seite 176.
3	Geben Sie auf der Registerkarte <b>PID</b> den <b>Sollwert</b> ein. Im Allgemeinen handelt es sich bei diesem Wert um eine Speicheradresse oder einen Analogeingang. Die <b>Parameter</b> (Kp, Ti, und Td) sollten <b>Speicherwortadressen (%MWxx)</b> entsprechen.  Geben Sie die <b>Abtastperiode</b> (Ts, Seite 181) für die PID-Funktion auf der Registerkarte <b>PID</b> , Seite 180 ein. Dieser Parameter kann ebenfalls ein Speicherwort sein (der Wert kann dann über die Animationstabelle eingestellt werden).  In der Betriebsart <b>Wortadresse</b> wird der <b>Korrekturtyp</b> auf <b>Auto</b> gesetzt und grau abgeblendet (er kann nicht manuell geändert werden).
4	Auf der Registerkarte <b>AT</b> sollte der <b>AT-Modus</b> auf die Option <b>Genehmigen</b> gesetzt werden. Geben Sie die <b>Dynamische Korrektur</b> und den <b>AT-Trigger</b> ein. Weitere Informationen hierzu finden Sie auf der Registerkarte <b>AT</b> , Seite 182 im Fenster <b>PID-Assistent</b> .
5	Auf der Registerkarte <b>Ausgang</b> muss <b>Aktion</b> auf den Wert <b>Bitadresse</b> eingestellt werden. Geben Sie eine <b>Speicherbitadresse</b> ein. Bei Bedarf können auf der Registerkarte „Ausgang“, Seite 183 <b>Grenzwerte</b> konfiguriert werden. Definieren Sie im Feld <b>Analogausgang</b> die Wortadresse: Analogausgang oder Speicherwort. Stellen Sie bei Bedarf den <b>Ausgang PWM</b> ein (siehe Registerkarte <b>Ausgang</b> , Seite 183 im Fenster <b>PID-Assistent</b> , Seite 176).
6	Klicken Sie auf <b>OK</b> , um die Konfiguration des <b>PID</b> -Reglers zu bestätigen.

### Schritt 3: Überprüfen des Setup

Schritt	Aktion
1	Schließen Sie den PC an die Logiksteuerung an und übertragen Sie die Anwendung.
2	Umschalten der Logiksteuerung in den Zustand <b>RUNNING</b> .

**HINWEIS:** Bevor Sie die Logiksteuerung in den **RUNNING**-Zustand setzen, müssen Sie sicherstellen, dass die Betriebsbedingungen der Maschine einen **RUNNING**-Zustand für die restliche Anwendung zulassen. Das Verfahren ist dasselbe wie in den Betriebsmodi **AT** und **AT+PID**. Die Konfiguration per **Wortadresse** ermöglicht Ihnen eine Änderung der **PID**-Betriebsarten über die Software. Im **PID**-Modus gestaltet sich das Verfahren grundlegend einfacher, sofern die Parameter (Kp, Ti, Td und Ts) bekannt sind und kein **Auto-Tuning** durchgeführt werden muss.

Die folgende Tabelle zeigt das allgemeine Verfahren zur Einrichtung des *PID*-Reglers:

Schritt	Aktion
1	Erstellen Sie eine Animationstabelle mit den bei der Konfiguration definierten Objekten. Detaillierte Informationen hierzu können Sie dem <i>EcoStruxure Machine Expert - Basic Betriebshandbuch</i> entnehmen.
2	Überprüfen Sie die Kohärenz zwischen der gemessenen Prozessvariablen und den anderen in der Animationstabelle definierten Werten. Wenn Sie Zweifel bezüglich der Genauigkeit der Messwerte haben, setzen Sie die Logiksteuerung in den STOP-Zustand und überprüfen Sie die Verdrahtung der Analogkanäle.  Wenn Sie erkennen, dass das Stellglied nicht gesteuert wird: <ul style="list-style-type: none"> <li>• Überprüfen Sie für einen Analogausgang die Ausgangsspannung bzw. den Ausgangsstrom vom Analogkanal.</li> <li>• Überprüfen Sie für einen PWM-Ausgang Folgendes: <ul style="list-style-type: none"> <li>◦ Die LED des dedizierten Ausgangs leuchtet.</li> <li>◦ Die Verdrahtung der Spannungsversorgungen und der 0-V-Schaltung ist fehlerfrei.</li> <li>◦ Das Stellglied wird mit Spannung versorgt.</li> </ul> </li> </ul>
3	Stellen Sie die Zykluszeit des Logic Controllers so ein, dass die <b>Abtastperiode</b> ( $T_s$ ) genau einem Vielfachen der Zykluszeit des <i>PID</i> -Reglers entspricht. Weitere Informationen zur <b>Abtastperiode</b> finden Sie unter Ermittlung der Abtastperiode, Seite 174.
4	Wenn Sie die Auto-Tuning, Seite 169-Funktion verwenden möchten, müssen Sie ggf. den manuellen Modus, Seite 173 ausführen, um die <b>dynamische Korrektur</b> und den auf der Registerkarte <b>AT</b> , Seite 182 im Fenster <b>PID-Assistent</b> definierten <b>AT-Trigger</b> zu ermitteln.
5	Schalten Sie die Steuerung über die Animationstabelle ein: <ul style="list-style-type: none"> <li>• Stellen Sie die Betriebsart, Seite 162 ein.</li> <li>• Aktivieren Sie den <i>PID</i>-Regler, Seite 186.</li> <li>• Stellen Sie die bei der Konfiguration, Seite 167 definierten Parameter auf die geeigneten Werte in Übereinstimmung mit der ausgewählten Betriebsart ein.</li> </ul>

## PID-Einstellung mittels Auto-Tuning (AT)

### Einführung

Der Auto-Tuning-Modus ermöglicht die automatische Einstellung der  $K_p$ -,  $T_i$ - und  $T_d$ -Werte sowie der Aktionsparameter, um eine präzisere Konvergenz der *PID*-Funktion zu erreichen. Die in *EcoStruxure Machine Expert - Basic* verfügbare Auto-Tuning-Funktion eignet sich insbesondere für die automatische Einstellung thermischer Prozesse.

In diesem Abschnitt werden folgende Themen behandelt:

- Auto-Tuning-Anforderungen
- Beschreibung des Auto-Tuning-Prozesses
- Speicherung der berechneten Koeffizienten
- Einstellung der *PID*-Parameter
- Auto-Tuning wird gestartet
- Einschränkungen bei der Verwendung der Auto-Tuning-Funktion und der *PID*-Regelung

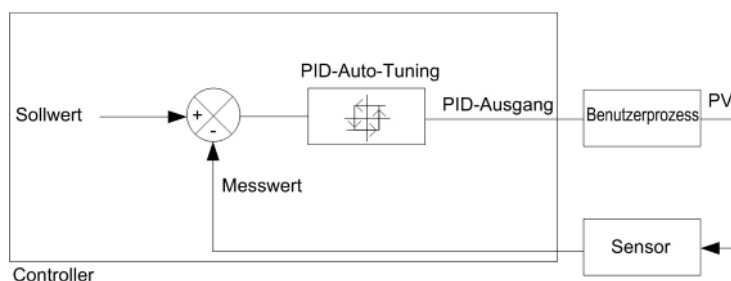
## Auto-Tuning-Anforderungen

Bei Verwendung der Auto-Tuning-Funktion ist sicherzustellen, dass der Regelprozess und die Steuerung folgenden Anforderungen gerecht werden:

- Prozessanforderungen:
  - Der Prozess muss ein stabiles offenes System ohne Rückführung sein.
  - Der Prozess muss im gesamten Betriebsbereich größtenteils linear verlaufen.
  - Die Prozessreaktion auf eine Pegeländerung des Analogausgangs folgt einem vorübergehend asymptotischen Muster.
  - Der Prozess befindet sich in einem stabilen Zustand mit einem Null-Eingang am Anfang der Auto-Tuning-Sequenz.
  - Der Prozess muss während des gesamten Prozessverlaufs störungsfrei sein. Andernfalls sind die berechneten Parameter ungültig oder der Auto-Tuning-Prozess wird nicht ordnungsgemäß durchgeführt.
- Konfigurationsanforderungen:
  - Konfigurieren Sie die Steuerung im periodischen Abfragemodus, um sicherzustellen, dass die Auto-Tuning-Funktion fehlerfrei ausgeführt wird.
  - Verwenden Sie die Auto-Tuning-Funktion nur, wenn keine anderen *PID*-Steuerungen ausgeführt werden.
  - Konfigurieren Sie die  $K_p$ -,  $T_i$ - und  $T_d$ -Koeffizienten als Speicherwortadressen (%MWxx).
  - Stellen Sie den Aktionstyp auf der Registerkarte **Ausgang** auf eine Speicherbitadresse (%Mxx) ein.

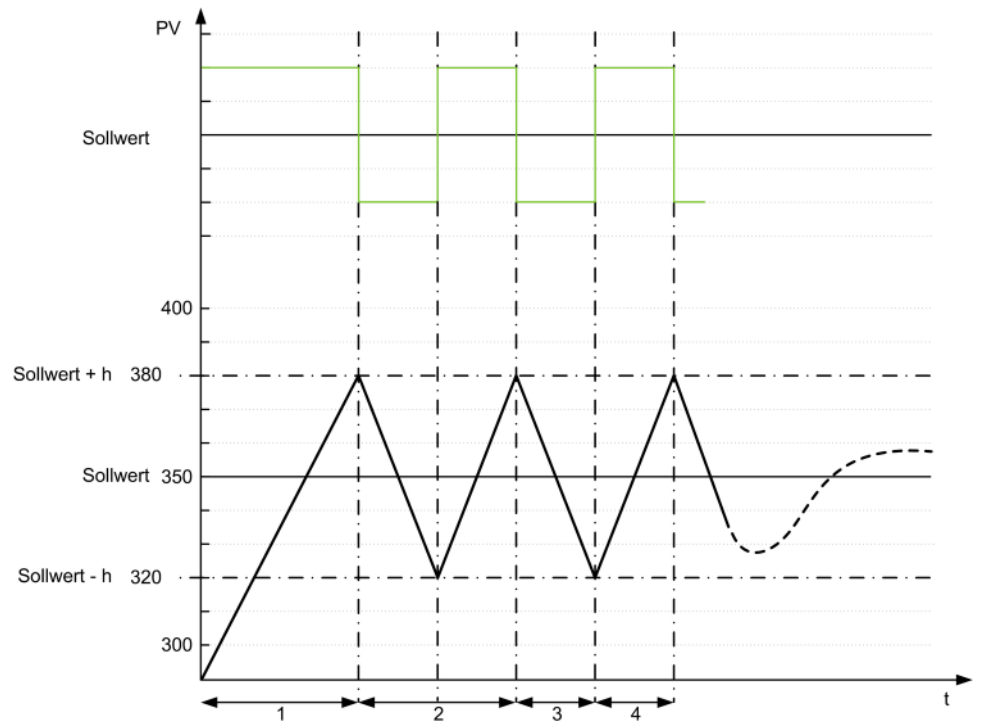
## Beschreibung des Auto-Tuning-Prozesses

Die folgende Abbildung beschreibt das Auto-Tuning in der Steuerung und in der Anwendung:



## Beschreibung des Kalibrierungsvorgangs des Auto-Tuning

Der Kalibrierungsvorgang des Auto-Tuning ist in vier aufeinander folgende Phasen unterteilt. Jede Phase des Prozesses muss erfüllt sein, damit der Auto-Tuning-Prozess erfolgreich abgeschlossen werden kann. Die folgenden Prozessansprechkurven und die Tabelle beschreiben die vier Phasen der EcoStruxure Machine Expert - Basic  $PID$  Auto-Tuning-Funktion:



**PV** Prozesswert

■ PID-Ausgang

**h** = 1 % (**Max.**-Wert - **Min.**-Wert) des Feldes **Messbereich** auf der Registerkarte **AT**

---- PID aktiv

1...4 Auto-Tuning-Phasen (siehe nachfolgende Tabelle)

Die folgende Tabelle beschreibt die Auto-Tuning-Phasen:

AT-Phase	Beschreibung
1	Der PID-Ausgang wird auf den <b>Max.</b> -Wert des Feldes <b>Grenzwerte</b> auf der Registerkarte, Seite 184 <b>Ausgang</b> forciert, bis der Prozesswert den Sollwert + h erreicht.
2	In der Auto-Tuning-Phase 2 gibt es zwei Schritte: 1. Der PID-Ausgang wird auf den <b>Min.</b> -Wert des Feldes <b>Grenzwerte</b> auf der Registerkarte, Seite 184 <b>Ausgang</b> forciert, bis der Prozesswert den Sollwert - h erreicht. 2. Der PID-Ausgang wird auf den <b>Max.</b> -Wert des Feldes <b>Grenzwerte</b> auf der Registerkarte, Seite 184 <b>Ausgang</b> forciert, bis der Prozesswert den Sollwert + h erreicht.
3	Der PID-Ausgang wird auf den <b>Min.</b> -Wert des Feldes <b>Grenzwerte</b> auf der Registerkarte, Seite 184 <b>Ausgang</b> forciert, bis der Prozesswert den Sollwert - h erreicht.
4	In der Auto-Tuning-Phase 4 gibt es zwei Schritte: 1. Der PID-Ausgang wird auf den <b>Max.</b> -Wert des Feldes <b>Grenzwerte</b> auf der Registerkarte, Seite 184 <b>Ausgang</b> forciert, bis der Prozesswert den Sollwert + h erreicht. 2. Der PID-Ausgang wird auf den <b>Min.</b> -Wert des Feldes <b>Grenzwerte</b> auf der Registerkarte, Seite 184 <b>Ausgang</b> forciert, die PID-Parameter werden berechnet und der PID wird aktiv.
(1) Der zuletzt vor Beginn des Auto-Tunings auf den Prozess angewendete Ausgang wird sowohl als Startpunkt als auch als Ruhepunkt für den Auto-Tuning-Prozess verwendet.	

**HINWEIS:** Die Kp- Ti- und Td-Parameter können nicht berechnet werden, wenn die manuelle Ausgangssteuerung während des Kalibrierungsvorgangs des Auto-Tuning aktiviert ist. Starten Sie den Kalibrierungsvorgang des Auto-Tuning erneut, wenn die manuelle Ausgangssteuerung beendet ist.

## Speicherung der berechneten Koeffizienten

Nach Abschluss der Auto-Tuning-Sequenz werden die den Kp-, Ti- und Td-Koeffizienten zugeordneten Speicherwörter und der Aktionstyp mithilfe der berechneten Werte eingestellt. Diese Werte werden in den RAM-Speicher geschrieben und in der Logiksteuerung gespeichert, solange die Anwendung gültig ist und kein Kaltstart durchgeführt wird (%S0).

Wenn das System nicht durch äußere Störungen beeinflusst wird, können die berechneten Werte in die Einstellungen des *PID*-Reglers geschrieben werden (siehe die Registerkarte **PID** im **PID-Assistent**, Seite 183). Auf diese Weise kann die *PID*-Betriebsart auf den *PID*-Modus eingestellt werden.

## Anpassung der PID-Parameter

Die Auto-Tuning-Methode kann sich als überaus dynamische Steuerung erweisen, die zu unerwünschtem Überschwingen beim Schrittwechsel der Sollwerte führen kann. Um die von den *PID*-Parametern (Kp, Ti, Td) per Auto-Tuning gelieferte Prozessregelung feiner anzupassen, haben Sie auch die Möglichkeit, diese Parameterwerte direkt ausgehend von der Registerkarte **PID** des **PID-Assistenten** oder über die entsprechenden Speicherwörter (%MW) manuell anzupassen. Detaillierte Informationen zur manuellen Parameteranpassung finden Sie im Anhang, Seite 191.

## Auto-Tuning wird gestartet

Auf der Registerkarte **AT** kann über den **AT-Trigger** eine Wiederholung der Auto-Tuning-Sequenz aktiviert werden. Der Auto-Tuning-Prozess wird bei jeder steigenden Flanke des mit dem **AT-Trigger** verknüpften Signals ausgelöst.

Informationen zur Konfiguration des Auto-Tuning finden Sie auf der Registerkarte **AT**, Seite 182.

## Einschränkungen bei der Verwendung der Auto-Tuning-Funktion

Thermische Prozesse können in vielen Fällen mit reinen Verzögerungsmodellen der ersten Rangordnung gleichgesetzt werden. Diese Art von Modell lässt sich anhand von zwei Schlüsselparametern beschreiben:

- Zeitkonstante  $\tau$
- Verzögerung  $\theta$

Das **Auto-Tuning** ist am besten für Prozesse geeignet, in denen die Zeitkonstante ( $\tau$ ) und die Verzögerung ( $\theta$ ) folgenden Kriterien entsprechen:

- $10 \text{ s} < (\tau + \theta) < 2700 \text{ s}$  (d. h.: 45 min)
- $2 < \tau / \theta < 20$

## Manueller Modus

### Einführung

Der manuelle Modus kann im Fenster **PID-Assistent** aufgerufen werden (**Output** tab, Seite 183). In diesem Modus können Sie die Befehle der *PID* umgehen. Mit der Verwendung des manuellen Modus werden 2 Hauptziele verfolgt:

- Initialisierung der Konfiguration
- Ermittlung der Abtastperiode

### Beschreibung

Im manuellen Modus können Sie den **Output value**, Seite 183 angeben. Dieses Verfahren kann sich besonders für den Test der Systemreaktion als überaus nützlich erweisen.

Durch Einstellen der **Bitadresse** auf der Registerkarte **Ausgang**, Seite 183 auf den Wert 1 wird der manuelle Modus aktiviert. Wenn die Option **Aktivieren** ausgewählt ist, steht ausschließlich der manuelle Modus zur Auswahl.

### Anwendung

Wenn der manuelle Modus aktiv ist, wird dem Ausgang der von Ihnen definierte Festwert zugewiesen. Dieser Ausgangswert liegt zwischen 0 und 10.000 (0 und 100 % für einen PWM-Ausgang).

Sie können den manuellen Modus auch für die Durchführung von Tests zur Bestimmung der minimalen/maximalen Ausgangsbegrenzung verwenden.

Der manuelle Modus ist darüber hinaus erforderlich, wenn die *Methode der Prozessansprechkurve*, Seite 174 verwendet wird, die der Ermittlung der korrekten Abtastzeit ( $T_s$ ) dient.

## Starten des manuellen Modus

Bevor Sie den manuellen Modus starten, sollten Sie sicherstellen, dass sich der RUN/STOP-Schalter der Steuerung in der RUN-Position befindet.

Gehen Sie zum Start des manuellen Modus mithilfe einer Animationstabelle vor wie folgt:

Schritt	Beschreibung
1	Aktivieren Sie den manuellen Modus, indem Sie das dedizierte Speicherbit auf 1 setzen. Detaillierte Informationen hierzu finden Sie auf der Registerkarte <b>Ausgang</b> , Seite 183.
2	Stellen Sie bei Verwendung der PWM-Funktion die PWM- <b>Periode</b> auf den gewünschten Wert ein.
3	Setzen Sie das der Betriebsart zugeordnete Speicherwort auf der Registerkarte <b>Allgemein</b> , Seite 177 im Fenster <b>PID-Assistent</b> auf den Wert 1 ( <i>PID</i> -Modus). Detaillierte Informationen zu den Betriebsarten mit Wortadresse finden Sie unter Beschreibung der Betriebsarten, Seite 162.
4	Stellen Sie das dem manuellen Ausgang zugeordnete Speicherwort auf der Registerkarte <b>Ausgang</b> , Seite 183 auf den gewünschten Wert ein. Dieser manuelle Sollwert kann mehrere Male ausgewählt werden, wenn das System in seinem Initialzustand gehalten wird.
5	Aktivieren Sie die Schleifen-Steuerung, Seite 166.

## Anhalten des manuellen Modus

Gehen Sie zum Stopp des manuellen Modus mithilfe einer Animationstabelle vor wie folgt:

Schritt	Beschreibung
1	Deaktivieren Sie die Schleifen-Steuerung, Seite 166.
2	Unterbinden Sie den manuellen Modus, indem Sie das dedizierte Speicherbit auf 0 setzen. Detaillierte Informationen hierzu finden Sie auf der Registerkarte <b>Ausgang</b> , Seite 183.
3	Setzen Sie das der Betriebsart zugeordnete Speicherwort auf der Registerkarte <b>Allgemein</b> , Seite 177 für den <i>PID</i> -Regler auf 0. Detaillierte Informationen zu den Betriebsarten mit Wortadresse finden Sie unter Beschreibung der Betriebsarten, Seite 162.
4	Setzen Sie das dem manuellen Ausgang zugeordnete Speicherwort auf der Registerkarte <b>Ausgang</b> , Seite 183 auf 0.

## Ermittlung der Abtastperiode ( $T_s$ )

### Einführung

Die Abtastperiode ( $T_s$ ) ist der wichtigste Parameter für die *PID*-Regelung. Sie sollte auf der Registerkarte **PID**, Seite 180 im Fenster **PID-Assistent** mit Bedacht eingestellt werden. Dieser Parameter ist eng mit der Zeitkonstante ( $\tau$ ) des zu steuernden Prozesses verknüpft.

In diesem Abschnitt wird die Verwendung des Online-Modus beschrieben, gleichzeitig werden zwei Methoden zur Bestimmung der Abtastperiode ( $T_s$ ) vorgestellt:

- Methode der Prozessansprechkurve
- Trial-and-Error-Methode (Versuch und Irrtum)

### Methode der Prozessansprechkurve

Diese Methode besteht aus einer offenen Regelschleife (ohne Rückführung), über die die Zeitkonstante des zu steuernden Prozesses ermittelt wird. Zunächst muss sichergestellt werden, dass der Prozess über ein Zeitverzögerungsmodell der ersten Rangordnung beschrieben werden kann. Das Prinzip ist äußerst einfach: Wenden Sie einen Schrittwechsel am Eingang des Prozesses an, während die

Angangskurve des Prozesses überwacht wird. Verwenden Sie dann eine grafische Methode zur Ermittlung der Zeitverzögerung des Prozesses.

Gehen Sie vor wie folgt, um die Abtastperiode (Ts) über die Methode der Prozessansprechkurve zu ermitteln:

Schritt	Aktion
1	Es wird davon ausgegangen, dass Sie bereits die verschiedenen Einstellungen auf den Registerkarten <b>Allgemein</b> , <b>Eingang</b> , <b>PID</b> , <b>AT</b> und <b>Ausgang</b> der <i>PID</i> -Funktion konfiguriert haben.
2	Wählen Sie die Registerkarte <b>Ausgang</b> , Seite 183 im Fenster <b>PID-Assistent</b> aus.
3	Wählen Sie in der Dropdown-Liste <b>Manueller Modus</b> den Eintrag <b>Aktivieren</b> oder <b>Bitadresse</b> aus, um den manuellen Modus zu autorisieren.
4	Stellen Sie das Feld <b>Ausgang</b> auf einen hohen Wert ein (innerhalb des Bereichs [5.000...10.000]).
5	Laden Sie Ihre Anwendung auf den Logic Controller herunter. Detaillierte Informationen zum Herunterladen einer Anwendung finden Sie im <i>EcoStruxure Machine Expert - Basic Betriebshandbuch</i> .
6	Führen Sie die <i>PID</i> -Funktion aus und prüfen Sie den Anstieg der Ansprechkurve.
7	Sobald die Ansprechkurve einen stabilen Zustand erreicht hat, halten Sie die <i>PID</i> -Messung an.
8	Verwenden Sie die folgende grafische Methode, um die Zeitkonstante ( $\tau$ ) des Steuerungsprozesses zu bestimmen: <ol style="list-style-type: none"> <li>Berechnen Sie mittels der folgenden Formel den Prozessvariablenausgang bei einer Steigung von 63 % (<math>S_{[63\%]}</math>): <math>S_{[63\%]} = S_{[initial]} + (S_{[final]} - S_{[initial]}) \times 63\%</math></li> <li>Berechnen Sie grafisch die Zeitabszisse (<math>t_{[63\%]}</math>), die <math>S_{[63\%]}</math> entspricht.</li> <li>Ermitteln Sie grafisch die Initialzeit (<math>t_{[initial]}</math>), die dem Start des Anstiegs der Prozessansprechkurve entspricht.</li> <li>Berechnen Sie mittels der folgenden Beziehung die Zeitkonstante (<math>\tau</math>) des Regelprozesses: <math>\tau = t_{[63\%]} - t_{[initial]}</math></li> </ol>
9	Berechnen Sie mittels der folgenden Formel die Abtastzeit (Ts) <sup>(1)</sup> auf der Grundlage des Werts von ( $\tau$ ), den Sie im vorigen Schritt ermittelt haben: $Ts = \tau/75$
10	Stellen Sie die <b>Zykluszeit</b> für den periodischen <b>Abtastmodus</b> so ein, dass die Abtastzeit (Ts) ein exaktes Vielfaches der Zykluszeit ist: $Zykluszeit = Ts / n$ , wobei „n“ eine positive Ganzzahl <sup>(2)</sup> ist
<p><b>(1)</b> Die Basiseinheit für die Abtastperiode ist 10 ms. Daher müssen Sie den Wert von Ts auf die nächsten vollen 10 ms auf- bzw. abrunden.</p> <p><b>(2)</b> Sie müssen „n“ so wählen, dass die resultierende Zykluszeit eine positive Ganzzahl im Bereich [1...150] ms ist.</p>	

## Trial-and-Error-Methode (Versuch und Irrtum)

Bei der Trial-and-Error-Methode wird die Abtastperiode der Auto-Tuning-Funktion kontinuierlich geschätzt, bis der Algorithmus zu zufrieden stellenden Werten für Kp, Ti und Td konvergiert.

**HINWEIS:** Im Gegensatz zur Prozessansprechkurven-Methode basiert die Trial-and-Error-Methode nicht auf einer auf die Prozessreaktion angewandten Näherungsregel. Sie hat jedoch den Vorteil, dass der Wert zu einem Wert der Abtastzeit konvergiert, der sich in derselben Größenordnung wie der aktuelle Wert befindet.

Gehen Sie zur Durchführung einer auf Versuch und Irrtum basierenden Schätzung der Auto-Tuning-Funktion vor wie folgt:

Schritt	Aktion
1	Wählen Sie im <i>PID</i> -Konfigurationsbildschirm die Registerkarte <b>AT</b> aus.
2	Setzen Sie die <b>Ausgangsbegrenzung</b> der Auto-Tuning-Funktion auf <b>10.000</b> .
3	Laden Sie Ihre Anwendung auf den Logic Controller herunter. Detaillierte Informationen zum Herunterladen einer Anwendung finden Sie im <i>EcoStruxure Machine Expert - Basic Betriebshandbuch</i> .
4	Wählen Sie die Registerkarte <b>PID</b> im Fenster <b>PID-Assistent</b> aus.
5	Geben Sie die erste oder n-te Schätzung in das Feld <b>Abtastzeit</b> <sup>(1)</sup> ein.
6	Starten Sie Auto-Tuning, Seite 163.
7	Warten Sie, bis der Auto-Tuning-Vorgang abgeschlossen ist.
8	Zwei Fälle sind möglich: <ul style="list-style-type: none"> <li>• <b>Auto-Tuning wird erfolgreich abgeschlossen:</b> Fahren Sie mit Schritt 10 fort.</li> <li>• <b>Auto-Tuning nicht erfolgreich:</b> Siehe <i>Auto-Tuning-Fehlercodes</i>, Seite 189. Das bedeutet, dass die aktuelle Schätzung der Abtastperiode (Ts) falsch ist. Führen Sie eine neue Schätzung von Ts durch und wiederholen Sie die Schritte 3 bis 8 so oft wie erforderlich, bis der Auto-Tuning-Vorgang letztendlich konvergiert.</li> </ul>
9	Halten Sie sich an folgende Richtlinien, um eine neue Schätzung von Ts durchzuführen: <ul style="list-style-type: none"> <li>• Der Auto-Tuning-Vorgang wird mit dem Fehlercode 800C hex beendet. Das bedeutet, die Abtastperiode (Ts) ist zu groß. Reduzieren Sie den Wert von Ts für eine neue Schätzung.</li> <li>• Der Auto-Tuning-Vorgang wird mit dem Fehlercode 800A hex beendet. Das bedeutet, die Abtastperiode (Ts) ist zu gering. Erhöhen Sie den Wert von Ts für eine neue Schätzung.</li> </ul>
10	Passen Sie die <i>PID</i> -Regelparameter <sup>(2)</sup> (Kp, Ti und Td) auf der Registerkarte <b>PID</b> , Seite 180 im Fenster <b>PID-Assistent</b> nach Bedarf an.

**(1)** Wenn Sie über keinen ersten Anhaltspunkt für den möglichen Bereich für die Abtastperiode verfügen, setzen Sie diesen Wert auf den zulässigen Mindestwert: 1 (1 Einheit von 10 ms).

**(2)** Wenn die *PID*-Einstellung dieser Regelparametern keine vollständig zufrieden stellenden Ergebnisse liefert, können Sie die Versuch-und-Irrtum-Schätzung der Abtastperiode weiter verfeinern, bis Sie die richtigen Kp-, Ti- und Td-Regelparameter erhalten.

## Online-Modus

Wenn sich der Logic Controller im Online-Modus befindet und die periodische Task ausgeführt wird, kann sich der im Ts-Feld angezeigte Wert (im Fenster, Seite 176 **PID-Assistent**) vom eingegebenen Parameter (%MW) unterscheiden. Der Ts-Wert ist ein Vielfaches der periodischen Task, während der %MW-Wert dem von der Logiksteuerung gelesenen Wert entspricht.

## PID-Assistent

### Zugriff auf den PID-Assistenten

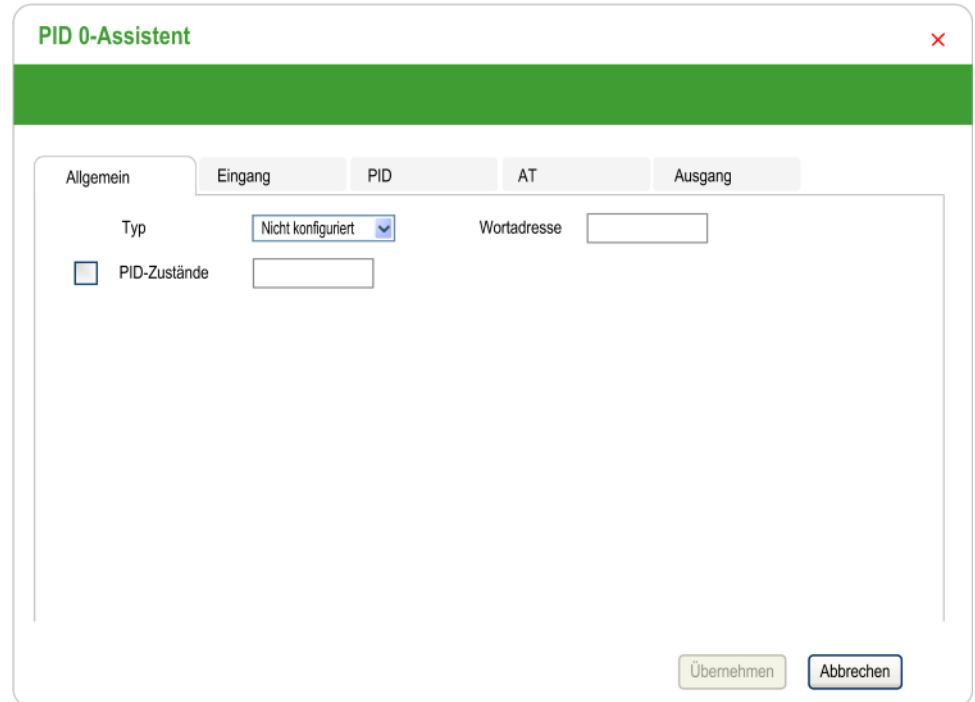
#### Einführung

Verwenden Sie das Fenster **PID-Assistent** von *EcoStruxure Machine Expert - Basic* zur Konfiguration des *PID*-Reglers.

## Konfigurationsassistent


Klicken Sie in der *PID*-Eigenschaftstabelle auf die Schaltfläche **Konfiguration [...]**. Dadurch wird das Fenster **PID-Assistent** aufgerufen.

Die nachstehende Abbildung zeigt das Fenster **PID-Assistent**.



Das Fenster **PID-Assistent** umfasst verschiedene Registerkarten, je nachdem, ob Sie sich im Offline- oder Online-Modus befinden:

Registerkarte	Zugriff	Verbindung
<b>Allgemeines</b>	Offline	Registerkarte „Allgemein“, Seite 177
<b>Eingang</b>	Offline	Registerkarte „Eingang“, Seite 179
<b>PID</b>	Offline	Registerkarte „PID“, Seite 180
<b>AT</b>	Offline	Registerkarte „AT“, Seite 182
<b>Ausgang</b>	Offline	Registerkarte „Ausgang“, Seite 183

Nach der Auswahl der Betriebsart werden die Registerkarten mit leeren Feldern, in die Werte eingegeben werden müssen, mit dem Symbol  gekennzeichnet und die betroffenen Felder rot unterlegt.

## Registerkarte „Allgemein“

### Einführung

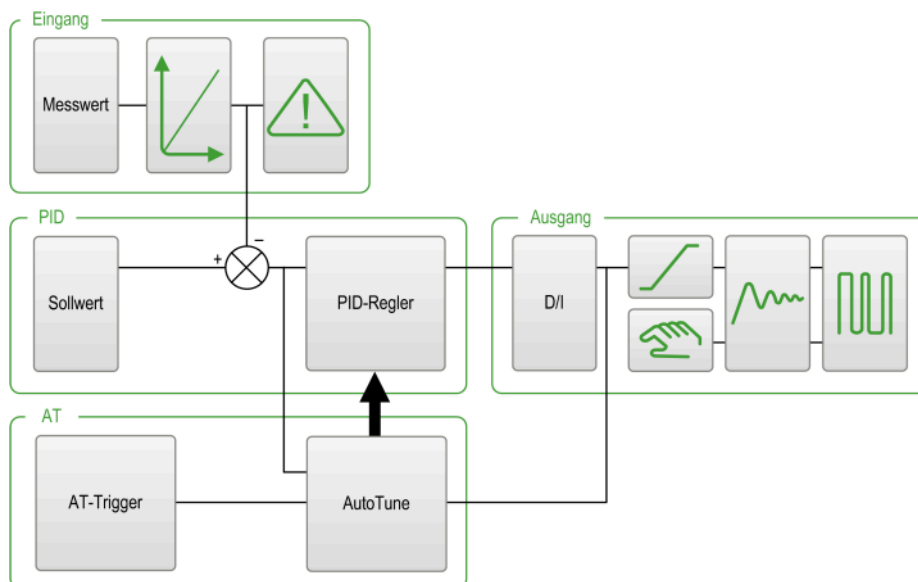
In diesem Kapitel wird die Registerkarte **Allgemein** des *PID* beschrieben. Die Registerkarte **Allgemein** wird standardmäßig angezeigt, wenn Sie den *PID*-Assistenten im Offline-Modus aufrufen.

## Beschreibung

In der nachstehenden Tabelle werden die verschiedenen Einstellungen auf der Registerkarte **Allgemein** beschrieben.





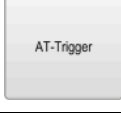
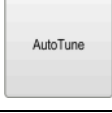






Parameter	Beschreibung
<b>Betriebsart</b>	<p>Stellt den zu verwendenden <i>PID</i>-Modus dar:</p> <ul style="list-style-type: none"> <li>• Nicht konfiguriert</li> <li>• PID</li> <li>• AT + PID</li> <li>• AT</li> <li>• Wortadresse</li> </ul> <p>Detaillierte Informationen zu den Betriebsarten finden Sie unter <i>PID</i>-Betriebsarten, Seite 162.</p>
<b>Wortadresse</b>	<p>Sie können in dieses Textfeld ein Speicherwort (%MWxx) eingeben, das zur Festlegung der Betriebsart über das Programm verwendet wird. Das Speicherwort kann abhängig von der Betriebsart, die Sie einstellen möchten, mögliche Werte annehmen:</p> <ul style="list-style-type: none"> <li>• %MWx = 0 (PID deaktiviert)</li> <li>• %MWx = 1 (zum Einstellen von „Nur PID“)</li> <li>• %MWx = 2 (zum Einstellen von „AT + PID“)</li> <li>• %MWx = 3 (zum Einstellen von „Nur AT“)</li> <li>• %MWx = 4 (zum Einstellen von „Nur PI“)</li> </ul>
<b>PID-Zustände</b>	<p>Wenn Sie das Kontrollkästchen markieren, um diese Option zu aktivieren, können Sie ein Speicherwort im zugehörigen Feld (%MWxx) angeben, das vom <i>PID</i>-Regler verwendet wird, um den aktuellen <i>PID</i>-Status während der Ausführung des <i>PID</i>-Reglers und/oder der Auto-Tuning-Funktion zu speichern. Detaillierte Informationen hierzu finden Sie unter <i>PID</i>-Zustände und Fehlercodes, Seite 187.</p>

## Grafischer Assistent



Der grafische Assistent unterstützt Sie bei der Visualisierung der Generierung der *PID*-Funktion. Es handelt sich um eine dynamische Grafik, die in Übereinstimmung mit der Konfiguration aktualisiert wird.

In der nachstehenden Tabelle werden die Verfügbarkeit der verschiedenen Symbole und die Aktionen beschrieben, die durch Klicken auf die Symbole ausgeführt werden:

Anzeige	Beschreibung
	Klicken Sie auf diese Schaltfläche, um das Sollwert-Feld auf der Registerkarte <b>PID</b> , Seite 180 anzuzeigen.
	Klicken Sie auf diese Schaltfläche, um die Registerkarte <b>PID</b> , Seite 180 anzuzeigen.
	Klicken Sie auf diese Schaltfläche, um die Registerkarte <b>Ausgang</b> , Seite 183 anzuzeigen.
	Klicken Sie auf diese Schaltfläche, um die Registerkarte <b>Eingang</b> , Seite 179 anzuzeigen.
	Klicken Sie auf diese Schaltfläche, um die Registerkarte <b>AT</b> , Seite 182 anzuzeigen.
	Klicken Sie auf diese Schaltfläche, um die Registerkarte <b>AT</b> , Seite 182 anzuzeigen.
	Diese Schaltfläche wird angezeigt, wenn die Option <b>Genehmigen</b> im Bereich <b>Konvertierung</b> auf der Registerkarte <b>Eingang</b> , Seite 179 ausgewählt ist.
	Diese Schaltfläche wird angezeigt, wenn die Option <b>Genehmigen</b> im Bereich <b>Alarmer</b> auf der Registerkarte <b>Eingang</b> , Seite 179 ausgewählt ist.
	Diese Schaltfläche wird angezeigt, wenn die <b>Grenzwerte</b> im <b>Grenzwertbereich</b> auf der Registerkarte <b>Ausgang</b> , Seite 183 nicht auf <b>Sperren</b> eingestellt wurden.
	Diese Schaltfläche wird angezeigt, wenn der <b>manuelle Modus</b> im Bereich <b>manueller Modus</b> auf der Registerkarte <b>Ausgang</b> , Seite 183 nicht auf <b>Sperren</b> eingestellt wurde.
	Klicken Sie auf diese Schaltfläche, um die Registerkarte <b>Ausgang</b> , Seite 183 anzuzeigen.
	Diese Schaltfläche wird angezeigt, wenn die Option <b>Genehmigen</b> im <b>PWM-Ausgangsbereich</b> auf der Registerkarte <b>Ausgang</b> , Seite 183 ausgewählt wurde.

## Registerkarte „Eingang“

### Einführung

In diesem Abschnitt wird die Registerkarte **Eingang** von *PID* beschrieben. Die Registerkarte **Eingang** dient der Eingabe der *PID*-Eingangsparameter.

Diese Registerkarte ist nur im Offline-Modus und nur bei Auswahl einer Betriebsart auf der Registerkarte **Allgemein** verfügbar.

## Beschreibung

In der folgenden Tabelle werden die Parameter beschrieben, die Sie definieren können:

Parameter	Beschreibung	
<b>Messung</b>	Geben Sie hier die Variable an, die den Messwert des zu regulierenden Prozesses enthält.  Die Standardskala reicht von 0 bis 10000. Sie können ein Speicherwort (%MWxx) oder einen Analogeingang eingeben.	
<b>Konvertie- rung</b>	<b>Autorisieren</b>	Aktivieren Sie dieses Kontrollkästchen, um den Prozesswert <b>[0...10000]</b> in einen linearen Bereich <b>Min. ... Max.</b> zu konvertieren.  Die Konvertierung betrifft auch den Sollwert.
	<b>Min. Max.</b>	Geben Sie den Minimal- und Maximalwert der Konvertierungsskala an. Der Prozesswert wird dann automatisch innerhalb des Intervalls <b>Min. ... Max.</b> neu berechnet.  <b>Min.</b> oder <b>Max.</b> können Speicherwörter (%MWxx), konstante Wörter (%KWxx) oder ein Wert zwischen -32768 und +32767 sein.  <b>HINWEIS:</b> Hinweis: <b>Min.</b> muss kleiner als <b>Max.</b> sein.
<b>Filter</b>	<b>Autorisieren</b>	Aktivieren Sie dieses Kontrollkästchen, um dem gemessenen Eingang einen Filter zuzuweisen.
	<b>(100 ms)</b>	Geben Sie einen Filterwert von 0 bis 10000 oder eine Speicherwortadresse (%MWxx) an. Die Basiseinheit der Filterzeit ist 100 ms.
<b>Alarme</b>	<b>Autorisieren</b>	Aktivieren Sie dieses Kontrollkästchen, um Alarme in Eingangsvariablen zu aktivieren.  Die Alarmwerte sind anhand des nach der Konvertierungsphase erhaltenen Prozesswerts festzulegen. Bei aktiver Konvertierung müssen die Alarmwerte von <b>Min.</b> bis <b>Max.</b> reichen. Andernfalls gilt für die Alarmwerte der Bereich 0 bis 10000
	<b>Niedrig Ausgang</b>	Geben Sie den unteren Alarmwert in das Feld <b>Niedrig</b> ein.  Bei diesem Wert kann es sich um ein Speicherwort (%MWxx), eine Konstante (%KWxx) oder einen Direktwert handeln.  <b>Ausgang</b> muss die Adresse des Bits enthalten, das auf 1 gesetzt wird, wenn der untere Grenzwert erreicht ist. <b>Ausgang</b> kann ein Speicherbit (%Mxx) oder ein Ausgang sein.
	<b>Hoch Ausgang</b>	Geben Sie den oberen Alarmwert in das Feld <b>Hoch</b> ein.  Bei diesem Wert kann es sich um ein Speicherwort (%MWxx), eine Konstante (%KWxx) oder einen Direktwert handeln.  <b>Ausgang</b> muss die Adresse des Bits enthalten, das auf 1 gesetzt wird, wenn der obere Grenzwert erreicht ist. <b>Ausgang</b> kann ein Speicherbit (%Mxx) oder ein Ausgang sein.

## Registerkarte „PID“

### Einführung

Verwenden Sie die Registerkarte **PID** zur Eingabe der internen *PID*-Parameter.

Diese Registerkarte ist nur im Offline-Modus und nur bei Auswahl einer Betriebsart auf der Registerkarte **Allgemein** verfügbar.

## Beschreibung

In der folgenden Tabelle werden die Einstellungen beschrieben, die Sie definieren können:

Parameter	Beschreibung
<b>Sollwert</b>	<p>Legen Sie den <i>PID</i>-Sollwert fest. Bei diesem Wert kann es sich um ein Speicherwort (%<i>MWxx</i>), ein konstantes Wort (%<i>KWxx</i>) oder einen Direktwert handeln.</p> <p>Dieser Wert muss zwischen 0 und 10000 liegen, wenn die Konvertierung deaktiviert ist. Andernfalls muss der Wert für die Konvertierung zwischen dem <b>Min.</b>- und dem <b>Max.</b>-Wert liegen.</p>
<b>Korrektortyp</b>	<p>Wenn die Betriebsart <b>PID</b> oder <b>AT + PID</b> zuvor in der Eigenschaftstabelle <i>PID</i> ausgewählt wurde, können Sie im Dropdown-Listefeld den gewünschten Korrektortyp (<b>PID</b> oder <b>PI</b>) auswählen. Wurden andere Betriebsarte (<b>AT</b> oder <b>Wortadresse</b>) ausgewählt, dann wird der <b>Korrektortyp</b> auf <b>Auto</b> gesetzt und grau abgeblendet (d. h. er kann manuell nicht geändert werden).</p> <p>Wenn <b>PI</b> im Dropdown-Listefeld ausgewählt wird, wird der Td-Parameter auf 0 forciert und dieses Feld ist deaktiviert.</p>
<b>Parameter</b> <sup>(1)</sup>	<p><b>Kp (x0,01s)</b></p> <p>Geben Sie die <i>PID</i>-Proportionalverstärkung an, multipliziert mit 100.</p> <p>Bei diesem Wert kann es sich um ein Speicherwort (%<i>MWxx</i>), ein konstantes Wort (%<i>KWxx</i>) oder einen Direktwert handeln.</p> <p>Der gültige Bereich für den <i>Kp</i>-Parameter lautet: <math>0 &lt; Kp &lt; 10000</math>.</p> <p><b>HINWEIS:</b> Wenn <i>Kp</i> versehentlich auf 0 gesetzt wird (<math>Kp \leq 0</math> ist ungültig), wird automatisch der Standardwert <math>Kp=100</math> von der <i>PID</i>-Funktion zugewiesen.</p>
	<p><b>Ti (x0,1s)</b></p> <p>Geben Sie die Integralzeit für eine Zeitbasis von 0,1 Sekunden ein.</p> <p>Bei diesem Wert kann es sich um ein Speicherwort (%<i>MWxx</i>), ein konstantes Wort (%<i>KWxx</i>) oder einen Direktwert handeln.</p> <p>Er muss zwischen 0 und 36000 liegen.</p> <p><b>HINWEIS:</b> Setzen Sie diesen Koeffizienten auf 0, um die integrale Aktion des <i>PID</i> zu deaktivieren.</p>
	<p><b>Td (x0,1s)</b></p> <p>Geben Sie die Differentialzeit für eine Zeitbasis von 0,1 Sekunden ein.</p> <p>Bei diesem Wert kann es sich um ein Speicherwort (%<i>MWxx</i>), ein konstantes Wort (%<i>KWxx</i>) oder einen Direktwert handeln.</p> <p>Er muss zwischen 0 und 10000 liegen.</p> <p><b>HINWEIS:</b> Setzen Sie diesen Koeffizienten auf 0, um die abgeleitete Aktion des <i>PID</i> zu deaktivieren.</p>
<b>Abtastperiode</b>	<p>Geben Sie hier die <i>PID</i>-Abtastperiode für eine Zeitbasis von <math>10^{-2}</math> Sekunden (10 ms) ein.</p> <p>Bei diesem Wert kann es sich um ein Speicherwort (%<i>MWxx</i>), ein konstantes Wort (%<i>KWxx</i>) oder einen Direktwert handeln.</p> <p>Er muss zwischen 1 (0,01 s) und 10000 (100 s) liegen.</p>
<p><b>(1)</b> Wenn Auto-Tuning aktiviert ist, brauchen Sie die Parameter <i>Kp</i>, <i>Ti</i> und <i>Td</i> nicht mehr zu definieren, da sie automatisch über das Programm vom Auto-Tuning-Algorithmus eingestellt werden. In diesem Fall dürfen Sie in diese Felder nur eine interne Wortadresse (%<i>MWxx</i>) eingeben. Geben Sie bei aktivierter Auto-Tuning-Funktion kein konstantes Wort und keinen Direktwert ein.</p>	

## Registerkarte „AT“

### Einführung

Die Registerkarte **AT** ist mit der Auto-Tuning-Funktion verbunden. Detaillierte Informationen finden Sie unter PID-Einstellung per Auto-Tuning, Seite 169.

Diese Registerkarte ist nur im Offline-Modus und nur bei Auswahl einer Betriebsart auf der Registerkarte **Allgemein** verfügbar.

### Beschreibung

PID Auto-Tuning ist ein rückführungsfreier Prozess, der ohne Vorschriften oder andere als die durch den Prozessvariablen-Grenzwert (PV) oder den Ausgangs-Sollwert vorgegebenen Einschränkungen direkt auf den Regelprozess einwirkt. Daher müssen beide Werte sorgfältig innerhalb des zulässigen, durch den Prozess vorgegebenen Bereichs ausgewählt werden, um eine mögliche Überlastung des Prozesses zu vermeiden.

Wenn die PID-Regelung mit Auto-Tuning durchgeführt wird, wirkt sich der Parameter **Dynamische AT-Korrektur** auf den Wert der Proportionalverstärkung ( $K_p$ ) aus. Die Berechnung der Proportionalverstärkung im Auto-Tuning-Prozess ist von der ausgewählten Geschwindigkeit der Dynamikkorrektur abhängig. Sie können eine der folgenden Optionen auswählen:

- **Schnell**
- **Mittel**
- **Langsam**
- **Wortadresse**

Weitere Informationen finden Sie in den Beschreibungen der Optionen in der nachfolgenden Tabelle:

<b>▲ WARNUNG</b>
<p><b>INSTABILER PID-BETRIEB</b></p> <ul style="list-style-type: none"> <li>• Beim Setzen der Istwert-Grenze (PV: Prozesswert) und der Ausgangssollwert-Grenzen müssen die jeweils damit verbundenen Auswirkungen auf die Maschine bzw. den Prozess berücksichtigt werden.</li> <li>• Überschreiten Sie in keinem Fall den zulässigen Wertebereich für den Istwert und die Ausgangssollwerte.</li> </ul> <p><b>Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.</b></p>

<b>▲ WARNUNG</b>
<p><b>UNBEABSICHTIGTER GERÄTEBETRIEB</b></p> <p>Verwenden Sie keinen Relaisausgang mit der PID-Funktion.</p> <p><b>Die Nichtbeachtung dieser Anweisungen kann Tod, schwere Verletzungen oder Sachschäden zur Folge haben.</b></p>

In der folgenden Tabelle werden die Einstellungen beschrieben, die Sie definieren können:

Feld	Beschreibung	
<b>AT-Modus</b>	<b>Genehmigen</b>	<p>Aktivieren Sie dieses Kontrollkästchen, um Auto-Tuning zu aktivieren.</p> <p>Es gibt zwei Möglichkeiten, dieses Kontrollkästchen zu verwenden, je nachdem, ob Sie die Betriebsart manuell oder über eine Wortadresse auf der Registerkarte <b>Allgemein</b> der <i>PID</i>-Funktion einstellen:</p> <ul style="list-style-type: none"> <li>• Wenn Sie den <b>Betriebsmodus</b> auf der Registerkarte <b>Allgemein</b>, Seite 177 auf <b>PID + AT</b> oder <b>ATE</b> einstellen, wird die Option <b>Genehmigen</b> aktiviert und kann nicht bearbeitet werden.</li> <li>• Wenn Sie den Betriebsmodus über eine Wortadresse <math>%MWx</math> (<math>%MWx = 2</math>: <b>PID + AT</b>; <math>%MWx = 3</math>: <b>AT</b>), festlegen, müssen Sie die Option <b>Genehmigen</b> manuell aktivieren, um die Auto-Tuning-Parameter konfigurieren zu können.</li> </ul>
<b>Messbereich</b>	<b>Genehmigen</b>	<p>Aktivieren Sie dieses Kontrollkästchen, um die Bereichsmessung zu aktivieren.</p> <p><b>HINWEIS:</b> Wenn die Bereichsmessung deaktiviert ist, wird der Wert <b>Min.</b> auf 0 und der Wert <b>Max.</b> auf 10000 eingestellt.</p>
	<b>Min.</b> <b>Max</b>	<p>Legen Sie die Werte <b>Min.</b> und <b>Max</b> basierend auf dem Messbereich von 1 % über oder unter dem Sollwert fest.</p> <p>Bei den Werten kann es sich um Direktwerte von 1 bis 10000 oder ein Speicherwort <math>%MWx</math> handeln.</p> <p><b>HINWEIS:</b> Der <b>Min.</b>-Wert muss kleiner sein als der <b>Max</b>-Wert.</p> <p><b>Beispiel:</b> Wenn der Prozesswert bei etwa <math>35\text{ °C} \pm 3\text{ °C}</math> liegen muss, gilt Folgendes:</p> <ul style="list-style-type: none"> <li>• Der Sollwert entspricht 350.</li> <li>• <math>\pm 3\text{ °C}</math> entspricht <math>h</math>, Seite 171 und sollte bei 30 liegen.</li> <li>• Folglich <math>1\% \times (\text{Max.} - \text{Min.}) = 30</math></li> <li>• Folglich <math>1\% \times 3000 = 30</math></li> <li>• Folglich <b>Max.</b> = 3100 und <b>Min.</b> = 100</li> </ul>
<b>Dynamische AT-Korrektur</b>	<b>Schnell</b> <b>Mittel</b> <b>Langsam</b> <b>Wortadresse</b>	<p>Dieser Parameter wirkt sich auf den vom <b>AT</b>-Prozess berechneten Wert der Proportionalverstärkung (<math>K_p</math>) aus.</p> <ul style="list-style-type: none"> <li>• <b>Schnell</b> bewirkt eine schnelle Antwortzeit mit größerem Überschwingen als „Mittel“.</li> <li>• <b>Mittel</b> bewirkt eine mittlere Antwortzeit mit mittlerem Überschwingen.</li> <li>• <b>Langsam</b> bewirkt eine längere Antwortzeit mit geringerem Überschwingen als „Mittel“.</li> <li>• <b>Wortadresse</b> bewirkt die im angegebenen Wortobjekt (<math>%MW</math>) konfigurierte Antwortzeit.</li> </ul>
<b>AT-Trigger</b>	<b>AT-Trigger</b>	<p>Über diesen Parameter kann der Start des <b>AT</b>-Prozesses bei Erkennung einer steigenden Flanke am dedizierten Bit (Speicherbit oder Digitaleingangsbit) ausgelöst werden.</p> <p>Der <b>AT</b>-Prozess wird angehalten, wenn eine fallende Flanke am Eingang <b>AT-Trigger</b> erkannt wird. Darüber hinaus wird im Modus <b>PID + AT</b> auch die <i>PID</i>-Einstellung gestoppt.</p>

## Berechnete $K_p$ -, $T_i$ - und $T_d$ -Koeffizienten

Nach Abschluss des Auto-Tuning-Vorgangs werden die berechneten *PID*-Koeffizienten  $K_p$ ,  $T_i$  und  $T_d$  in den entsprechenden Speicherwörtern ( $%MWx$ ) abgelegt.

## Registerkarte „Ausgang“

### Einführung

Diese Registerkarte ermöglicht die Eingabe der *PID*-Ausgangsparameter.

Diese Registerkarte ist nur im Offline-Modus und nur bei Auswahl einer Betriebsart auf der Registerkarte **Allgemein** verfügbar.

## Beschreibung

In der folgenden Tabelle werden die Einstellungen beschrieben, die Sie definieren können:

Feld	Beschreibung
<b>Aktion</b>	<p>Geben Sie hier den Typ der auf den Prozess wirkende <i>PID</i>-Aktion an. Drei Optionen sind verfügbar: <b>Invers</b>, <b>Direkt</b> und <b>Bitadresse</b>. Wenn eine Erhöhung des Ausgangswerts eine Erhöhung des gemessenen Prozesswerts bewirkt, definieren Sie eine invertierte Aktion (<b>Invers</b>). Wenn jedoch eine Reduzierung des Messwerts verursacht wird, machen Sie <i>PID</i> direkt (<b>Direkt</b>).</p> <p>Bei der Auswahl von <b>Bitadresse</b><sup>(1)</sup> können Sie den Aktionstyp durch Änderung des zugeordneten Bits anpassen, bei dem es sich entweder um ein Speicherbit (%Mxx) oder um eine Eingangsadresse (%I.x.y) handelt.</p> <p>Das Speicherbit wird bei Auswahl der Aktion <b>Direkt</b> auf 1 und bei Auswahl der Aktion <b>Invers</b> auf 0 gesetzt.</p>
<b>Grenzwerte</b>	<p>Geben Sie an, ob der <i>PID</i>-Ausgang mit Einschränkungen belegt werden soll. Es stehen 3 Optionen zur Verfügung: <b>Aktivieren</b>, <b>Deaktivieren</b> und <b>Bitadresse</b>.</p> <p>Wählen Sie <b>Aktivieren</b> aus, um das <b>Bit</b> auf 1 zusetzen, bzw. <b>Deaktivieren</b>, damit das <b>Bit</b> auf 0 gesetzt wird.</p> <p>Wählen Sie <b>Bitadresse</b> aus, wenn die Verwaltung der Grenzwerte über eine Änderung des zugeordneten Bits erfolgen soll, bei dem es sich um ein Speicherbit (%Mxx) oder um eine Eingangsadresse (%I.x.y) handeln kann.</p> <p>Legen Sie den oberen und unteren Grenzwert für den <i>PID</i>-Ausgang fest.</p> <p><b>Min.</b> oder <b>Max.</b> können Speicherwörter (%MWxx), konstante Wörter (%KWxx) oder Werte zwischen 1 und 10000 (0,01 % bis 100 % der PWM-Periode) sein.</p> <p><b>HINWEIS:</b> Der <b>Min.</b>-Wert muss kleiner sein als der <b>Max.</b>-Wert.</p>
<b>Manueller Modus</b>	<p>Geben Sie an, ob der <i>PID</i> in den manuellen Modus wechseln soll. Es stehen 3 Optionen zur Verfügung: <b>Aktivieren</b>, <b>Deaktivieren</b> und <b>Bitadresse</b>.</p> <p>Wenn Sie <b>Bitadresse</b> auswählen, können Sie über das Programm in den manuellen Modus (Bit auf 1) oder in den automatischen Modus (Bit auf 0) wechseln, indem Sie das zugeordnete Bit ändern, wobei es sich entweder um ein Speicherbit (%Mxx) oder um einen Eingang handeln kann.</p> <p>Der <b>Ausgang</b> des manuellen Modus muss den Wert enthalten, den Sie dem Analogausgang zuweisen möchten, wenn sich die <i>PID</i>-Funktion im manuellen Modus, Seite 173 befindet. Dieser <b>Ausgang</b> kann entweder ein Wort (%MWxx) oder ein direkter Wert im Format [0...10.000] sein.</p>
<b>Analogausgang</b>	<p>Geben Sie den <i>PID</i>-Ausgang an, der im Auto-Tuning-Modus verwendet werden soll.</p> <p>Bei diesem <b>Analogausgang</b><sup>(2)</sup> kann es sich um eine Speicherwortadresse oder um eine Analogausgangsadresse handeln. Bei der Verwendung der PWM-Funktion von <i>PID</i> sind nur Speicherwortadressen erlaubt.</p>
<b>Ausgang PWM</b>	<p>Aktivieren Sie dieses Kontrollkästchen, um die PWM-Funktion des <i>PID</i> zu verwenden.</p> <p>Geben Sie im Textfeld <b>Periode (x0,1 s)</b> die Modulationsperiode ein. Diese Periode muss zwischen 1 und 500 liegen und kann ein Speicherwort (%MWxx) oder ein konstantes Wort (%KWxx) sein. Die Genauigkeit der PWM-Funktion ist sowohl von der PWM-Periode als auch von der Abfrageperiode abhängig. Sie fällt besser aus, wenn das PWM-Verhältnis (<i>PWM.R</i>) über die größtmögliche Anzahl an Werten verfügt.                      Beispiel: Abfrageperiode = 20 ms und PWM-Periode = 200 ms. In diesem Fall kann <i>PWM.R</i> folgende Werte annehmen: 0 %, 10 %, 20 %, 30 %, 40 %, 50 %, 60 %, 70 %, 80 %, 90 %, 100 %. Bei einer Abfrageperiode = 50 ms und PWM-Periode = 200 ms kann <i>PWM.R</i> folgende Werte annehmen: 0 %, 25 %, 50 %, 75 % und 100 % des <i>PWM.P</i>-Periodenwerts.</p> <p><b>Beispiel:</b> im Falle von <i>PWM.R</i> = 75 %</p> <p>The diagram shows a horizontal axis labeled '(ms)'. A series of rectangular pulses are shown. The first pulse is labeled 'Ts' with a double-headed arrow below it. A longer double-headed arrow below the first pulse is labeled 'Ton'. A longer double-headed arrow below the first two pulses is labeled 'PWM.P'. The text 'Ts = Abfrageperiode' and 'Ton = Summe von Ts' is written to the right of the diagram.</p> <p>Geben Sie das PWM-Ausgangsbit als Wert in <b>Ausgang</b> an. Dabei kann es sich um ein Speicherbit (%Mxx) oder um eine Ausgangsadresse handeln. Detaillierte Informationen zur PWM-Funktion finden Sie im Kapitel Impulsbreitenmodulation/Pulse Width Modulation (%PWM), Seite 48.</p>
<p>(1) Wenn Auto-Tuning aktiviert ist, bestimmt der Auto-Tuning-Algorithmus automatisch den richtigen direkten oder inversen Aktionstyp für den Regelprozess. Sie müssen dann im zugehörigen Textfeld <b>Bitadresse</b> ein Speicherbit (%Mxx) eingeben.</p> <p>(2) Geben Sie eine Speicheradresse (%MWxx) oder eine Analogausgangsadresse (%QWx.y) ein.</p>	

# PID-Programmierung

## Verwendung der PID-Funktion

Dieser Abschnitt enthält der Beschreibungen und Programmierrichtlinien für die Verwendung der **PID**-Funktion.

## Beschreibung

### Einführung

Ein Proportional-Integral-Differential-Regler (*PID*) ist ein generischer Rückführmechanismus (Steuerung) für Regelkreise, der in industriellen Steuerungssystemen weitläufig zum Einsatz kommt. Der *PID*-Regler verwendet einen Algorithmus mit 3 separaten konstanten Parametern: den Proportional-, den Integral- und den Differentialwerten, die jeweils mit P, I und D gekennzeichnet werden.

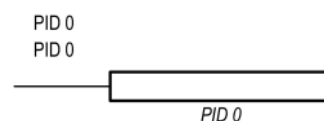
## Hauptmerkmale

Die EcoStruxure Machine Expert - Basic-PID-Funktion stellt folgende Hauptmerkmale bereit:

- Analogeingang
- Lineare Konvertierung des konfigurierbaren Messwerts
- Überwachung des oberen und unteren Grenzwerts am konfigurierbaren Eingang
- Analog- oder PWM-Ausgang (Impulsbreitenmodulation)
- Abschaltung für den konfigurierbaren Ausgang
- Konfigurierbare direkte oder inverse Aktion
- Auto-Tuning-Funktion

## Abbildung

Nachstehend eine Abbildung der *PID*-Funktion im KOP-Editor von EcoStruxure Machine Expert - Basic:



**HINWEIS:** Zwischen PID und der PID-Nummer muss ein Leerzeichen eingefügt werden (Beispiel: PID<Leerzeichen>0).

## Parameter

Im Gegensatz zu den *Timer*- oder *Counter*-Funktionsbausteinen ist in EcoStruxure Machine Expert - Basic kein *PID*-Funktionsbaustein vorhanden. Die Anweisung `[PID x]` aktiviert lediglich die *PID*-Rückführungsfunktion, wobei x der *PID*-Nummer entspricht.

Um die *PID*-Funktion zu konfigurieren, navigieren Sie zum Fenster **Programmierung**, klicken Sie auf **Tools > PID** und bearbeiten Sie dann die *PID*-Eigenschaften bedarfsgerecht (die verschiedenen Konfigurationsparameter werden in der nachstehenden Tabelle beschrieben).

Die *PID*-Funktion verfügt über folgende Parameter:

Parameter	Beschreibung	Wert
<b>Verwendet</b>	Ausgewählt, wenn die E/A im Projekt verwendet werden.	True/False False (Standard)
<b>PID</b>	Name des aktuellen <i>PID</i> -Objekts	Ein Programm kann nur eine begrenzte Anzahl an <i>PID</i> -Funktionen enthalten. Die maximale Anzahl an <i>PID</i> -Objekten finden Sie in der Tabelle Maximale Anzahl Objekte (siehe Modicon M221, Logic Controller, Programmierhandbuch).
<b>Symbol</b>	Symbol des aktuellen <i>PID</i> -Objekts	Das diesem <i>PID</i> -Objekt zugeordnete Symbol. Weitere Informationen finden Sie unter Definieren und Verwenden von Symbolen (siehe EcoStruxure Machine Expert - Basic, Betriebshandbuch).
<b>[...]</b>	Schaltfläche zum Start des Assistenten	Klicken Sie auf die Schaltfläche, um das Fenster <b>PID-Assistent</b> aufzurufen. Detaillierte Informationen hierzu finden Sie unter PID-Assistent, Seite 176.
<b>Kommentar</b>	Kommentar	Diesem Objekt kann ein Kommentar zugeordnet werden.

## Programmierung und Konfiguration

### Einführung

In diesem Abschnitt werden die Programmierung und Konfiguration des EcoStruxure Machine Expert - Basic *PID*-Reglers beschrieben.

### Aktivieren des *PID*-Reglers

Das folgende Beispiel aktiviert die *PID 0*-Reglerschleife, wenn Bit *%M0* auf 1 gesetzt ist:

Programm- baustein	Anweisung
0	LD %M0 [PID 0]

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

### Analoge *PID*-Messung

Die *PID*-Funktion führt eine *PID*-Korrektur anhand einer Analogmessung und eines Sollwerts durch und gibt entweder einen analogen Befehl im gleichen Format oder eine PWM an einem Digitalausgang aus.

Um den *PID*-Regler mit voller Skala (höchste Auflösung) zu verwenden, konfigurieren Sie den der *PID*-Regelmessung zugeordneten Analogeingang im Format [0...10.000]. Wenn Sie die Standardkonfiguration [0...4095] verwenden, funktioniert die *PID*-Regelung dennoch ordnungsgemäß.

### Konfigurieren der Abfrageperiode

Wenn Sie EcoStruxure Machine Expert - Basic *PID*-Regler verwenden, müssen Sie den Abfragemodus der Logiksteuerung in den Abfragemodus **Periodisch**

(Registerkarte **Programm, Tasks > Master-Task**) konfigurieren. Im periodischen Abfragemodus werden die Abfragen durch die Steuerung in regelmäßigen Zeitintervallen gestartet, sodass die Abtastrate während der gesamten Messung konstant bleibt. Detaillierte Informationen zur Konfiguration des Abfragemodus finden Sie im *EcoStruxure Machine Expert - Basic Betriebshandbuch*.

Im periodischen Abfragemodus wird das Systembit %S19 vom System auf 1 gesetzt, wenn die Abfragezeit der Logiksteuerung die über das Benutzerprogramm definierte Zeit überschreitet.

## PID-Zustände und erkannte Fehlercodes

### Einführung

Der *PID*-Regler von EcoStruxure Machine Expert - Basic kann den aktuellen Zustand sowohl der *PID*-Regelung als auch des Auto-Tuning-Prozesses in ein benutzerdefiniertes Speicherwort schreiben. Detaillierte Informationen zur Aktivierung und Konfiguration des Speicherworts für die *PID*-Zustände finden Sie in Verbindung mit der Registerkarte **Allgemein**, Seite 177 im Fenster *PID-Assistent*, Seite 176.

Im Speicherwort für die *PID*-Zustände können folgende Typen von *PID*-Informationen aufgezeichnet werden:

- Aktueller Zustand des *PID*-Reglers
- Aktueller Zustand des Auto-Tuning-Prozesses
- *PID*-Fehlercodes
- Auto-Tuning-Fehlercodes

**HINWEIS:** Das Zustände-Speicherwort ist schreibgeschützt.

### Zustände-Speicherwort

Zustände	Beschreibung
0000 hex	<i>PID</i> -Regelung nicht aktiv
2000 hex	<i>PID</i> -Regelung aktiv
4000 hex	<i>PID</i> -Sollwert erreicht

### Speicherwort für die Auto-Tuning-Zustände

Zustand des AT-Prozesses	Beschreibung
0100 hex	Auto-Tuning-Phase 1, Seite 171 aktiv
0200 hex	Auto-Tuning-Phase 2, Seite 171 aktiv
0400 hex	Auto-Tuning-Phase 3, Seite 171 aktiv
0800 hex	Auto-Tuning-Phase 4, Seite 171 aktiv
1000 hex	Auto-Tuning-Phase abgeschlossen

## PID-Fehlercodes

In der nachstehenden Tabelle werden die möglicherweise erkannten Fehler beschrieben, die im Verlauf der *PID*-Steuerung auftreten können:

Fehlercode	Beschreibung
8001 hex	Betriebsmoduswert außerhalb des zulässigen Bereichs
8002 hex	Lineare Konvertierung min. und max. gleich
8003 hex	Oberer Grenzwert für den Digitalausgang geringer als unterer Grenzwert
8004 hex	Sollwertbegrenzung außerhalb des linearen Konvertierungsbereichs
8005 hex	Sollwertbegrenzung geringer als 0 oder größer als 10000
8006 hex	Sollwert außerhalb des linearen Konvertierungsbereichs
8007 hex	Sollwert geringer als 0 oder größer als 10000
8008 hex	Regelaktion nicht mit der beim AT-Start festgelegten Aktion identisch

## Auto-Tuning-Fehlercodes

Die folgende Tabelle enthält eine Liste der Auto-Tuning-Fehlermeldungen sowie deren mögliche Ursachen und Maßnahmen zur Behebung der Fehler:

Fehlercode	Beschreibung
8009 hex	Der Prozessvariablen Grenzwert (PV) wurde erreicht. Da es sich beim Auto-Tuning um einen offenen Prozess ohne Rückführung handelt, fungiert der Prozessvariablen Grenzwert (PV) als maximal zulässiger Wert.
800A hex	Entweder die Abtastperiode ist zu kurz oder der Ausgangssollwert ist zu niedrig. Erhöhen Sie entweder die Abtastperiode oder den Sollwert des AT-Ausgangs.
800B hex	Kp gleich Null.
800C hex	Die Zeitkonstante ist negativ, sodass die Abtastperiode u. U. zu lang ist. Detaillierte Informationen finden Sie unter <b>Einschränkungen bei der Verwendung der Auto-Tuning-Funktion</b> , Seite 192.
800D hex	Verzögerung ist negativ.
800E hex	<p>Fehler bei der Berechnung des Kp-Werts erkannt. Der AT-Algorithmus ist instabil (keine Konvergenz). Mögliche Ursachen:</p> <ul style="list-style-type: none"> <li>• Störungen am Prozess während des Auto-Tunings haben zu einer Verzerrung der Entwicklung der statischen Prozessverstärkung geführt.</li> <li>• Die vorübergehende Antwort der Prozessvariablen ist nicht groß genug, sodass das Auto-Tuning die statische Verstärkung nicht ermitteln kann.</li> <li>• Eine Kombination aus beiden vorgenannten Ursachen.</li> </ul> <p>Prüfen Sie die Parameter der <i>PID</i>- und der Auto-Tuning-Funktion und nehmen Sie bedarfsgerecht Anpassungen vor, um die Konvergenz zu verbessern. Stellen Sie zudem sicher, dass keine Störung vorliegt, die die Prozessvariable beeinflussen könnte. Versuchen Sie, folgende Werte zu ändern:</p> <ul style="list-style-type: none"> <li>• Sollwert des Ausgangs</li> <li>• Abtastperiode</li> </ul> <p>Vergewissern Sie sich, dass während der Ausführung des Auto-Tunings keine Störung des Prozesses auftritt.</p>
800F hex	Die Zeitkonstante überschreitet die Verzögerungsrate: $\tau/\theta > 20$ . Die <i>PID</i> -Regelung ist möglicherweise nicht länger stabil. Detaillierte Informationen finden Sie unter <b>Einschränkungen bei der Verwendung der Auto-Tuning-Funktion</b> , Seite 192.
8010 hex	Die Zeitkonstante überschreitet die Verzögerungsrate: $\tau/\theta < 2$ . Die <i>PID</i> -Regelung ist möglicherweise nicht länger stabil. Detaillierte Informationen finden Sie unter <b>Einschränkungen bei der Verwendung der Auto-Tuning-Funktion</b> , Seite 192.
8011 hex	Der Grenzwert für die statische Verstärkung Kp wurde überschritten: $K_p > 10000$ . Die Messempfindlichkeit einiger Anwendungsvariablen ist eventuell zu gering. Der Messbereich der Anwendung muss innerhalb des Intervalls [0...10000] neu eingestellt werden.
8012 hex	Der berechnete Werte der Integralzeit-Konstanten Ti wurde überschritten: $T_i > 20000$ .
8013 hex	Der berechnete Werte der Differentialzeit-Konstanten Td wurde überschritten: $T_d > 10000$ .
8014 hex	Ungültige Eingabensollwertwerte (außerhalb des Bereichs, der durch Alarmlimits, Seite 180 für niedrige und hohe Ausgänge definiert ist).
8015 hex	<p>Fehler bei der Filterverarbeitung:</p> <ul style="list-style-type: none"> <li>• Zykluszeit außerhalb des Bereichs.</li> <li>• Filterzeit <math>&lt; 10 \times</math> Zykluszeit</li> </ul>

---

# Anhang

## Inhalt dieses Abschnitts

PID-Parameter .....	191
---------------------	-----

## Überblick

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

# PID-Parameter

## Inhalt dieses Kapitels

Aufgabe und Einfluss von PID-Parametern.....	191
Methode zur Anpassung der PID-Parameter .....	193

## Aufgabe und Einfluss von PID-Parametern

### Einführung

In diesem Abschnitt werden die Aufgabe und der Einfluss der PID-Parameter beschrieben.

### PID-Regelungsmodell

Der EcoStruxure Machine Expert - Basic PID-Regler implementiert eine kombinierte (seriell-parallele) PID-Korrektur. Die Integral- und die Differentialaktionen agieren sowohl unabhängig voneinander als auch parallel zueinander. Die Proportionalaktion agiert in Bezug auf den kombinierten Ausgang der Integral- und der Differentialaktionen.

### Rechenalgorithmen

Es werden abhängig vom Wert der integralen Zeitkonstante ( $T_i$ ) zwei verschiedene Rechenalgorithmen verwendet:

- Wenn  $T_i \neq 0$ , dann wird ein Inkrementierungsalgorithmus verwendet.
- Wenn  $T_i = 0$ , dann wird ein Positionierungsalgorithmus in Verbindung mit einem Offset von +5000 verwendet, der auf den PID-Ausgang angewendet wird.

### Einfluss der Aktionen

Eine proportionale Aktion wird verwendet, um die Prozessansprechgeschwindigkeit zu beeinflussen. Eine Erhöhung der Proportionalaktion bewirkt Folgendes:

- Schnellere Antwort
- Weniger statische Fehler
- Reduzierung der Stabilität

Eine integrale Aktion wird verwendet, um statische Fehler auszugrenzen. Eine Erhöhung der Integralaktion (d. h. eine Reduzierung der Integralzeit  $T_i$ ) bewirkt Folgendes:

- Schnellere Antwort
- Reduzierung der Stabilität

Eine differentiale Aktion ist vorausgreifend. In der Praxis fügt sie einen Begriff hinzu, der die Geschwindigkeit der Variation in der Abweichung berücksichtigt (was es ermöglicht, Änderungen durch die Beschleunigung der Prozessansprechzeiten vorwegzunehmen, wenn die Abweichung zunimmt, und die Ansprechzeit dann wieder zu erhöhen, wenn die Abweichung abnimmt). Eine Erhöhung der Differentialaktion (d. h. eine Steigerung der Differentialzeit) bewirkt Folgendes:

- Langsamere Antwort
- Reduziertes Überschwingen

**HINWEIS:** In Bezug auf die Differentialzeit wird  $T_d$  zur Vorwegnahme der Variation in der Abweichung verwendet. Bei zu niedrigen oder zu hohen  $T_d$ -Werten kann es zu unerwünschten Oszillationen kommen.

Für jede Aktion muss ein geeigneter Kompromiss zwischen Geschwindigkeit und Stabilität gefunden werden.

## Grenzen der PID-Regelschleife

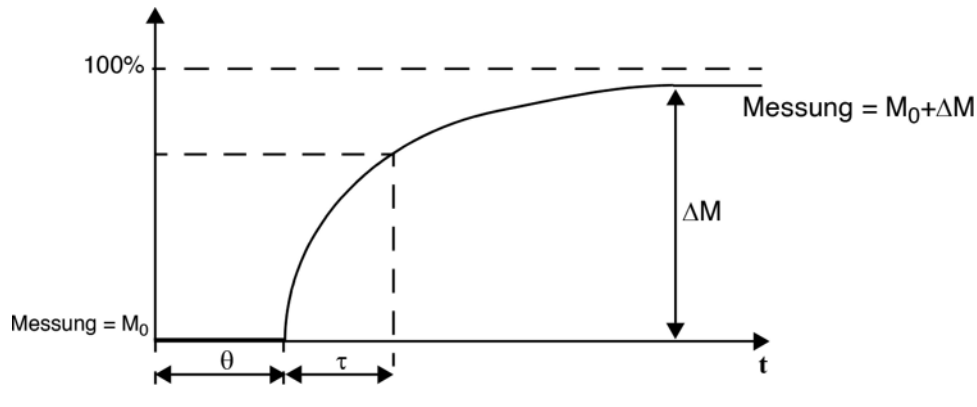
Wenn der Prozess an eine reine Verzögerung ersten Rangs mit einer Übertragungsfunktion angeglichen ist:

$$H(p) = K \times \frac{e^{-\theta p}}{1 + \tau p}$$

Hierbei gilt Folgendes:

$\tau$ : Konstante der Modellzeit

$\theta$ : Modellverzögerung



Die Leistung der Prozessregelung ist abhängig vom Verhältnis  $\frac{\tau}{\theta}$ .

Eine geeignete PID-Prozesssteuerung wird in folgendem Bereich erreicht:  $2 < \frac{\tau}{\theta} < 20$ .

Der PID-Regler ist ideal für die Regelung von Prozessen, die folgende Bedingung erfüllen:

- Für  $\frac{\tau}{\theta} < 2$ , mit anderen Worten für schnelle Regelschleifen (niedriger  $\theta$ ) oder für Prozesse mit einer hohen Verzögerung (hoher  $\tau$ ) ist die PID-Prozessregelung nicht mehr geeignet. In solchen Fällen müssen komplexere Algorithmen verwendet werden.
- Für  $\frac{\tau}{\theta} > 20$  ist eine Prozessregelung ausreichend, die einen Grenzwert plus Hysterese verwendet.

# Methode zur Anpassung der PID-Parameter

## Einführung

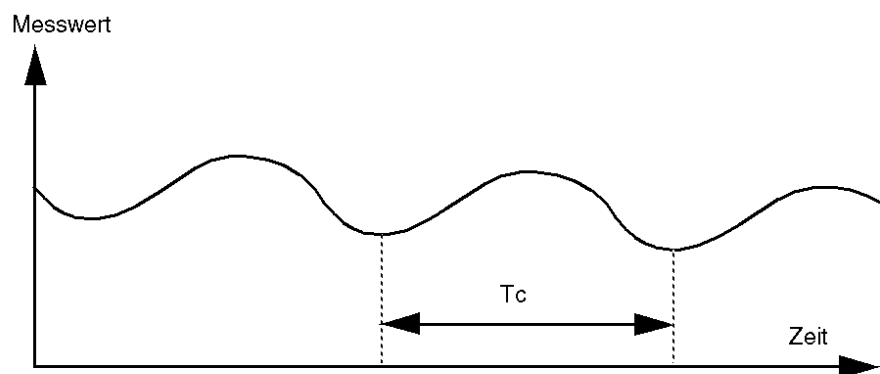
Für die Anpassung der PID-Parameter sind zahlreiche Verfahren verfügbar. Bei der bevorzugten Methode handelt es sich um die Ziegler-Nichols-Methode, für die es 2 Varianten gibt:

- Adaptive Regelung mit Rückführung
- Adaptive Regelung ohne Rückführung

Vor der Durchführung eines dieser Verfahren müssen Sie die PID-Aktion, Seite 183 festlegen.

## Adaptive Regelung mit Rückführung

Bei dieser Methode wird ein Proportionalbefehl ( $T_i = 0$ ,  $T_d = 0$ ) zum Start des Prozesses durch Erhöhung eines Proportionalkoeffizienten verwendet, bis nach Anwendung eines Pegels auf den Sollwert der PID-Korrektur erneut eine Oszillation festzustellen ist. All das ist erforderlich, um die kritische Proportionalverstärkung ( $K_{pc}$ ) zu erhöhen, die dazu geführt hat, dass die nicht gedämpfte Oszillation und die Oszillationsdauer ( $T_c$ ) die Werte verringert haben, was zu einer optimalen Einstellung des Reglers führt.

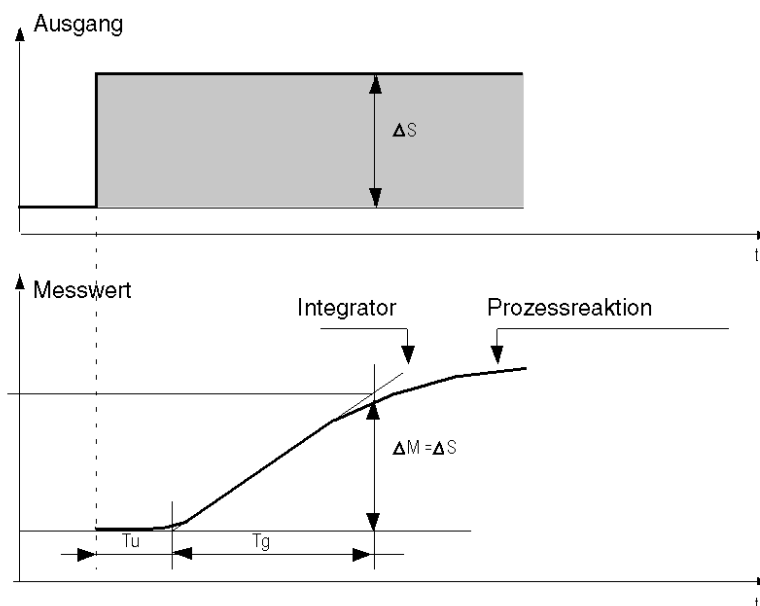


Je nach verwendetem Korrekturtyp (PID oder PI) erfolgt die Anpassung der Koeffizienten mit folgenden Werten:

Korrektur	Kp: P-Faktor	Ti: Integrationszeit	Td: Differential-
PID	$K_{pc}/1,7$	$T_c/2$	$T_c/8$
PI	$K_{pc}/2,22$	$0,83 \times T_c$	–

## Adaptive Regelung ohne Rückführung

Da sich der Regler im manuellen Modus, Seite 173 befindet, legen Sie einen Pegel am Ausgang an und lassen die Prozessantwort genauso starten wie ein Integrator mit reiner Verzögerungszeit.



Der Kreuzungspunkt auf der rechten Seite, der repräsentativ für den Integrator mit der Zeitachse ist, bestimmt die Zeit  $T_u$ . Als nächstes wird die Zeit  $T_g$  als die für die gesteuerte Variable (Messwert) erforderliche Zeit bestimmt, um dieselbe Variationsgröße (% der Skala) wie der Reglerausgang zu erhalten.

Je nach verwendetem Korrekturtyp (PID oder PI) erfolgt die Anpassung der Koeffizienten mit folgenden Werten:

Korrektur	Kp: P-Faktor	Ti: Integrationszeit	Td: Differential-
PID	$-1,2 T_g/T_u$	$2 \times T_u$	$0,5 \times T_u$
PI	$-0,9 T_g/T_u$	$3,3 \times T_u$	–

**HINWEIS:** Detaillierte Informationen zu Parametereinheiten finden Sie in Verbindung mit der Registerkarte, Seite 180 **PID**.

Dieses Anpassungsverfahren bietet außerdem eine äußerst dynamische Steuerung, was sich in einem unerwünschten Überschwingen während des Wechsels der Sollwertimpulse äußern kann. Verringern Sie in diesem Fall die Proportionalverstärkung, bis Sie das gewünschte Verhalten erreichen. Diese Methode erfordert keine Annahmen über die Art und Weise des Verfahrens. Sie können es sowohl auf das stabile Verfahren als auch auf reale integrierende Verfahren anwenden. Bei langsamen Verfahren (beispielsweise in der Glasindustrie) benötigt der Benutzer lediglich den Anfang der Antwort, um die Koeffizienten  $K_p$ ,  $T_i$  und  $T_d$  regeln zu können.

## A

### **Absolute Bewegung:**

Eine Bewegung zu einer Position, die von einem Referenzpunkt aus definiert wird.

## B

### **Beschleunigung/Verzögerung:**

Beschleunigung ist die Rate der Geschwindigkeitsänderung, von der **Startgeschwindigkeit** bis zur Zielgeschwindigkeit. Verzögerung ist die Rate der Geschwindigkeitsänderung, von der **Zielgeschwindigkeit** bis zur Stoppgeschwindigkeit. Diese Geschwindigkeitsänderungen werden implizit von der PTO-Funktion gemäß den Parametern Beschleunigung, Verzögerung und Rückanteil verwaltet, entsprechend einem S-Kurven- oder trapezförmigen Profil.

## C

### **CW/CCW:**

ClockWise / Counter ClockWise

## D

### **DWORD:**

(*Double Word: Doppelwort*) Im 32-Bit-Format codierter Typ.

## F

### **Funktion:**

Programmereinheit, 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).

## H

### **Homing:**

Die Methode zum Festlegen des Referenzpunkts für eine absolute Bewegung.

## J

### **jerk ratio:**

Das Verhältnis der Änderung von Beschleunigung und Verzögerung als Zeitfunktion.

---

## 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.

## S

### **S-Kurve Rampe:**

Eine Beschleunigungs-/Verzögerungsrampe, wenn der Parameter `JerkRatio` über 0 % liegt.

### **Startgeschwindigkeit:**

Die minimale Frequenz, bei der ein belasteter Schrittmotor anfahren kann, ohne Arbeitsschritte zu verlieren.

### **Stoppgeschwindigkeit:**

Die maximale Frequenz, bei der ein belasteter Schrittmotor stoppen kann, ohne Arbeitsschritte zu verlieren.

## T

### **Trapezrampe:**

Eine Beschleunigungs-/Verzögerungsrampe, wenn der Parameter `JerkRatio` bei 0 % liegt.

# Index

## A

Aktivieren/Deaktivieren der Endstufe	
MC_Power_ATV .....	59
Antriebsfunktionsbausteine	
Konfiguration .....	59
Antriebsfunktionsbausteine: Fehlercodes .....	77

## B

Beschleunigungsrampe .....	83
Bewegung anhalten	
MC_Stop_ATV .....	67
BUFFER_MODE .....	108

## C

configuring	
Drive function blocks .....	59

## D

DIRECTION .....	107
%DRV .....	54

## F

%FC .....	23
Fehlerbehandlung	
ErrID .....	21
Fehler .....	21
Fehlercodes	
Antriebsfunktionsbausteine .....	77
FREQGEN	
Funktionsbaustein .....	157
Funktionsblock-Konfiguration .....	159
Frequenzgenerator	
Funktionsbaustein .....	157
Funktionen	
PTO .....	80
Funktionsbausteine	
FC (Schneller Zähler) .....	23
Frequenzgenerator (%FREQGEN) .....	157
HSC (Hochgeschwindigkeitszähler) .....	28
Impuls .....	41
Impulsbreitenmodulation .....	48
MC_Halt_PTO .....	139
MC_Home_PTO .....	133
MC_Jog_ATV .....	62
MC_Motion_PTO .....	116
MC_MoveAbs_PTO .....	129
MC_MoveRel_PTO .....	126
MC_MoveVel_ATV .....	64
MC_MoveVel_PTO .....	122
MC_Power_ATV .....	59
MC_Power_PTO .....	119
MC_ReadMotionState_ATV .....	72
MC_ReadStatus_ATV .....	70
MC_Reset_ATV .....	74
MC_SetPost_PTO .....	135
MC_Stop_ATV .....	67
MC_Stop_PTO .....	137
MV_AbortTrigger_PTO .....	153
MV_ReadActPos_PTO .....	143

MV_ReadActVel_PTO .....	141
MV_ReadAxis_PTO .....	147
MV_ReadMotionState_PTO .....	146
MV_ReadPar_PTO .....	154
MV_ReadSts_PTO .....	144
MV_Reset_PTO .....	149
MV_TouchProbe_PTO .....	150
MV_WritePar_PTO .....	155

## H

Hochgeschwindigkeitszähler	
Beschreibung .....	28
Frequenzmesser-Modus .....	38
Zählmodus .....	32
HOMING_MODE .....	108
%HSC	
Hochgeschwindigkeitszähler .....	28

## I

Impulsbreitenmodulation	
Beschreibung .....	48
Funktionsblock-Konfiguration .....	49
Programmierbeispiel .....	52

## J

JerkRatio .....	83
-----------------	----

## L

Lesen des Bewegungszustands	
MC_ReadMotionState_ATV .....	72
Lesen des Gerätestatus	
%MC_ReadStatus_ATV .....	70

## M

MC_Halt_PTO	
Gesteuerter Bewegungsstopp bis die	
Geschwindigkeit 0 beträgt .....	139
MC_Home_PTO	
Befehlsachse zum Ausführen der Homing-	
Sequenz .....	133
MC_Jog_ATV	
Starten des Tippbetriebs .....	62
MC_Motion_PTO	
Aufrufen einer Motion Task Table .....	116
MC_MoveAbs_PTO	
Bewegen der Achse zu einer bestimmten Position	
mit der vorgegebenen Geschwindigkeit .....	129
MC_MoveRel_PTO	
Bewegen der Achse um einen inkrementellen	
Abstand mit der vorgegebenen	
Geschwindigkeit .....	126
MC_MoveVel_ATV	
Mit einer bestimmten Geschwindigkeit bewegen ..	64
MC_MoveVel_PTO	
Bewegen einer Achse mit einer bestimmten	
Geschwindigkeit .....	122
MC_Power_ATV	
Aktivieren/Deaktivieren der Endstufe .....	59
MC_Power_PTO	
Aktivieren der Stromversorgung einer Achse .....	119
MC_ReadMotionState_ATV	

Lesen des Bewegungszustands.....	72	Funktionen .....	80
MC_ReadStatus_ATV		Konfiguration .....	91
Lesen des Gerätestatus .....	70	motion task table.....	91
MC_Reset_ATV		PTO_ERROR .....	110–111
Quittieren und Zurücksetzen eines Fehlers .....	74	PTO_PARAMETER.....	109
MC_SetPost_PTO		Pulse	
Die Achse an eine vorgegebene Position		Beschreibung .....	41
bewegen.....	135	Funktionsblock-Konfiguration.....	42
MC_Stop_ATV		Programmierbeispiel .....	47
Bewegung anhalten .....	67	%PWM .....	48
MC_Stop_PTO			
Steuerung eines gesteuerten			
Bewegungsstopps.....	137		
Mit einer bestimmten Geschwindigkeit bewegen			
MC_MoveVel_ATV.....	64		
motion task table			
PTO .....	91		
MV_AbortTrigger_PTO			
Abbruch von Funktionsbausteinen, die mit			
Auslöseereignissen verbunden sind.....	153		
MV_ReadActPos_PTO			
Abrufen der Achsenposition .....	143		
MV_ReadActVel_PTO			
Abrufen der Achsengeschwindigkeit.....	141		
MV_ReadAxisError_PTO			
Abrufen eines Achsensteuerungsfehlers.....	147		
MV_ReadMotionState_PTO			
Abrufen des Bewegungszustands der Achse .....	146		
MV_ReadPar_PTO			
Abrufen der Parameter vom PTO .....	154		
MV_ReadSts_PTO			
Abrufen des Status der Achse .....	144		
MV_Reset_PTO			
Zurücksetzen achsenbezogener Fehler .....	149		
MV_TouchProbe_PTO			
Aktiviert ein auslösendes Ereignis auf dem Probe-			
Eingang.....	150		
MV_WritePar_PTO			
Schreiben von Parametern in den PTO.....	155		

## O

Objektcodes für Funktionsbausteine	
BUFFER_MODE.....	108
DIRECTION .....	107
HOMING_MODE .....	108
PTO_PARAMETER.....	109

## P

PID	
Adaptive Regelung mit Rückführung.....	193
Adaptive Regelung ohne Rückführung.....	193
Auto-Tuning.....	163
Beschreibung .....	185
Betriebsarten.....	162
Konfigurationsassistent .....	176
Parameter .....	191
Programmierung und Konfiguration .....	186
Registerkarte „Allgemein“ .....	177
Registerkarte „AT“ .....	182
Registerkarte „Ausgang“.....	183
Registerkarte „Eingang“ .....	179
Registerkarte „PID“ .....	180
Standardkonfiguration .....	166
Zustände und erkannte Fehlercodes .....	187
%PLS.....	41
PTO	

## Q

Quittieren und Zurücksetzen eines Fehlers	
MC_Reset_ATV .....	74

## S

Schneller Zähler	
Beschreibung .....	23
Konfiguration .....	24
Programmierbeispiel .....	26
Spiel.....	87
Starten des Tippbetriebs, MC_Jog_ATV.....	62

## V

Verwaltung der Funktionsbausteinausgänge und	
Ausgangsobjekte	
Busy .....	21
CmdAborted .....	21
ErrID .....	21
Fehler .....	21
Fertig .....	21
Verwaltung der Funktionsbausteineingänge und	
Eingangsobjekte	
Ausführung.....	21
Verzögerungsrampe .....	83



Schneider Electric  
35 rue Joseph Monier  
92500 Rueil Malmaison  
France

+ 33 (0) 1 41 29 70 00

[www.se.com](http://www.se.com)

Da Normen, Spezifikationen und Bauweisen sich von Zeit zu Zeit ändern, sollten Sie um Bestätigung der in dieser Veröffentlichung gegebenen Informationen nachsuchen.

© 2025 Schneider Electric. Alle Rechte vorbehalten.

EIO0000003307.03