

**Inbetriebnähmeanleitung, Firmware 5.3.1**  
**EGU mit Modbus RTU-Schnittstelle**  
**Elektrischer Universalgreifer**

Original Inbetriebnähmeanleitung

Hand in hand for tomorrow

## Impressum

### **Urheberrecht:**

Diese Anleitung ist urheberrechtlich geschützt. Urheber ist die SCHUNK SE & Co. KG.  
Alle Rechte vorbehalten.

### **Technische Änderungen:**

Änderungen im Sinne technischer Verbesserungen sind uns vorbehalten.

**Dokumentenummer:** 1514033-EGU-MB-FW5.3.1

**Auflage:** 01.00 | 23.09.2025 | de

Sehr geehrte Kundin,  
sehr geehrter Kunde,  
vielen Dank, dass Sie unseren Produkten und unserem Familienunternehmen als führendem  
Technologieausrüster für Roboter und Produktionsmaschinen vertrauen.  
Unser Team steht Ihnen bei Fragen rund um dieses Produkt und weiteren Lösungen jederzeit  
zur Verfügung. Fragen Sie uns und fordern Sie uns heraus. Wir lösen Ihre Aufgabe!  
Mit freundlichen Grüßen  
Ihr SCHUNK-Team

Customer Management  
Tel. +49-7133-103-2503  
Fax +49-7133-103-2189  
cmg@de.schunk.com



**Betriebsanleitung bitte vollständig lesen und produktnah aufbewahren.**

# Inhaltsverzeichnis

|   |           |
|---|-----------|
| Änderungen vom Software-Release V5.3.0 zu V5.3.1 .....                        | 5         |
| Änderungen vom Software-Release V5.2 zu V5.3.....                             | 6         |
| <b>1 Allgemein.....</b>   | <b>7</b>  |
| 1.1 Zu diesem Dokument .....  | 7         |
| 1.2 Definitionen .....  | 8         |
| 1.2.1 Minimale und maximale Position .....                                    | 8         |
| 1.2.2 Bewegungs- und Greifrichtungen .....                                    | 9         |
| 1.2.3 Greifmodi .....   | 10        |
| 1.2.4 Greifkraft- und Positionserhaltung (GPE).....                           | 11        |
| 1.2.5 Nullpunkt.....  | 11        |
| <b>2 Kommunikation .....</b>  | <b>12</b> |
| 2.1 Datenaustausch .....  | 12        |
| 2.1.1 Zyklischer Datenaustausch .....   | 12        |
| 2.1.2 Azyklischer Datenaustausch Modbus RTU .....                             | 21        |
| <b>3 Modulfunktionen .....</b>  | <b>23</b> |
| 3.1 Booten, Abschalten und Neustart.....                                      | 23        |
| 3.1.1 Booten und Betriebsbereitschaft herstellen .....                        | 23        |
| 3.1.2 Abschalten.....   | 24        |
| 3.1.3 Neustart.....   | 26        |
| 3.2 Bewegungsfunktionen .....   | 27        |
| 3.2.1 Tipp-Betrieb .....  | 27        |
| 3.2.2 Positionsfahrt absolut .....  | 28        |
| 3.2.3 Positionsfahrt relativ.....   | 30        |
| 3.2.4 Kontrolliert anhalten .....   | 32        |
| 3.2.5 Bewegung abbrechen .....  | 33        |
| 3.3 Handhabung eines Werkstücks.....  | 33        |
| 3.3.1 Werkstück-Greifen (einfache Greiffahrt) .....                           | 33        |
| 3.3.2 Werkstück-Greifen an erwarteter Position (kombinierte Greiffahrt) ..... | 35        |
| 3.3.3 Werkstück-Nachgreifen .....   | 41        |
| 3.3.4 Werkstückverlusterkennung.....  | 43        |
| 3.3.5 Werkstück-Freigeben .....   | 44        |
| 3.3.6 Werkstück manuell entnehmen .....                                       | 46        |
| 3.4 Weitere Funktionen .....  | 46        |
| 3.4.1 Nullpunktverschiebung.....  | 46        |
| 3.4.2 Handshake .....   | 48        |
| 3.4.3 Bremsentest.....  | 48        |
| 3.4.4 LifeSign .....  | 49        |
| 3.4.5 Steuerbefehl zeitoptimiert wiederholen .....                            | 49        |
| 3.4.6 Werkseinstellung.....   | 50        |
| <b>4 Systemparameter .....</b>  | <b>51</b> |

|   |           |
|---|-----------|
| 4.1 Wertebereiche .....                                       | 51        |
| 4.2 Parameterliste .....                                      | 51        |
| <b>5 Inbetriebnahme.....</b>                                  | <b>67</b> |
| 5.1 Sicherheit.....   | 67        |
| 5.2 Systemintegration .....                                   | 67        |
| 5.3 Inbetriebnahme Modbus RTU .....                           | 68        |
| 5.3.1 Allgemeine Kommunikationseinstellungen .....            | 68        |
| 5.3.2 Datenspezifische Kommunikationseinstellungen .....      | 69        |
| 5.3.3 Datenaustausch .....                                    | 70        |
| <b>6 Diagnose.....</b>  | <b>75</b> |
| 6.1 Warnungen.....  | 75        |
| 6.2 Fehler.....   | 77        |
| <b>7 Anhang .....</b>   | <b>81</b> |
| 7.1 Anwendungsbeispiele .....                                 | 81        |
| 7.2 Steuerdoppelwort.....                                     | 90        |
| 7.3 Statusdoppelwort.....                                     | 93        |
| 7.4 Zusatzcode bei anliegender Warnung WRN_NOT_FEASIBLE ..... | 96        |
| 7.5 Software Copyright Hinweise.....                          | 99        |

## Änderungen vom Software-Release V5.3.0 zu V5.3.1

Gegenüber dem Software-Release V5.3.0 ergeben sich die folgenden Erweiterungen/Verbesserungen:

- Das Verhalten beim Freigeben von elastischen Werkstücken wurde verbessert.
- Verbessertes Jog-Verhalten: Beim Verlassen der Softwarelimits wurde das Überschwingen reduziert.
- Fehlerkorrektur (Bugfixing):
  - Fälschlicherweise generierte Absolutwertgeber-Fehler "0xB9" beim Greifer-Start wurden behoben.
  - In seltenen Fällen wurde das Vorpositionieren beim "Werkstück greifen an erwarteter Position" übersprungen. Das wurde behoben.
  - Die Modulfunktion "Neustart" wird auch im Zustand "ready for shutdown" korrekt ausgeführt.
  - Beim Ausführen der Modulfunktion "Abschalten" wird die Regelung des Moduls deaktiviert, und falls vorhanden, fällt die Bremse ein.
  - Die Statusbits "workpiece pre-grip started" und "workpiece lost" werden nach einem Werkstückverlust korrekt zurückgesetzt.
  - Die Statusbits "workpiece pre-grip started" und "workpiece gripped" werden am Ende des Nachgreifens korrekt zurückgesetzt.
  - Das Setzen des reservierten Steuerbits 15 wird ignoriert.
  - Inkrementalgeber-Fehler werden im Diagnosespeicher gespeichert.

## Änderungen vom Software-Release V5.2 zu V5.3

Gegenüber dem Software-Release V5.2 ergeben sich die folgenden Erweiterungen/Verbesserungen:

- Defaultwert Parameter <wp\_lost\_dst> (► 4.2 [□ 55]) wurde auf 2 mm geändert
- zeitlicher Ablauf beim Werkstück-Greifen an erwarteter Position wurde optimiert
- Name des Zusatzcodes 0x28 bei anliegender Warnung WRN\_NOT\_FEASIBLE wurde geändert
- Erkennung von MODBUS-Kommunikationsabbrüchen wurde toleranter gestaltet (dadurch wurde die Kompatibilität zu UR Polycope Steuerungen verbessert)
- Fehlerkorrektur (Bugfixing):
  - Statusbit "workpiece pre-grip started" wird nach Anstoßen einer Funktion korrekt zurückgesetzt
  - nach Anstoßen von "release workpiece" während des Nachgreifens wird das Fehlerereignis ERR\_MOVE\_BLOCKED nicht mehr angezeigt (trat nur in seltenen Fällen auf)
  - Geschwindigkeitsschwingen nach Neu-/Start ist behoben (trat nur in seltenen Fällen auf)
  - Verfahren zu Ermittlung der Motortemperatur wurde robuster (trat nur in seltenen Fällen auf)

# 1 Allgemein

## 1.1 Zu diesem Dokument

Diese Anleitung beschreibt die Inbetriebnahme sowie die Bedienungs- und Parametriermöglichkeiten eines elektrischen Greifers EGU mit folgender Schnittstelle:

- Modbus RTU (MB)

### Gültigkeit

In dieser Ausführung der Anleitung sind die Funktionen für die Firmware-Versionen mit der Versionsnummer 5.3.1 beschrieben. Die Firmware-Version kann ausgelesen werden. Informationen zum entsprechenden Parameter sind enthalten im Abschnitt ▶ 4.2 [📄 65].

### Konventionen

Für diese Anleitung gelten folgende Konventionen:

- Der Greifer wird im Folgenden als "Modul" bezeichnet.
- Vom Benutzer angestoßene Aktionen, die das Modul ausführen soll, werden im Folgenden als "Steuerbefehl" bezeichnet.
- Kennzeichnung von Parametern: <parameter>
- Kennzeichnung von Ereignissen: WARNING
- Seitenzahl in Verweisen: [ ▶ 4]

**HINWEIS:** Abbildungen in dieser Anleitung dienen dem grundsätzlichen Verständnis und können von der tatsächlichen Ausführung abweichen.

### Abkürzungen

Folgende Abkürzungen werden verwendet:

- GPE: Greifkraft- und Positionserhaltung
- Allgemeine Geschäftsbedingungen \*
- Montage- und Betriebsanleitung des Moduls \*

### Mitgeltende Unterlagen

Die mit Stern (\*) gekennzeichneten Unterlagen können unter [schunk.com/downloads](https://www.schunk.com/downloads) heruntergeladen werden.

---

### HINWEIS

Inbetriebnahmeanleitungen zu älteren Firmware-Versionen können bei SCHUNK angefordert werden.

---

## 1.2 Definitionen

### 1.2.1 Minimale und maximale Position

Durch die Parameter <min\_pos> und <max\_pos> werden die Positionsgrenzen festgelegt, innerhalb derer Bewegungen zulässig sind.

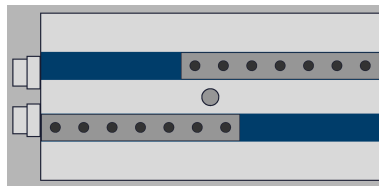
Der Wert des Parameters <min\_pos> entspricht hierbei dem *kleinsten* Positionswert, der angefahren werden kann.

Der Wert des Parameters <max\_pos> entspricht hierbei dem *größten* Positionswert, der angefahren werden kann.

Im Auslieferungszustand entsprechen die Positionswerte der Parameter <min\_pos> und <max\_pos> den im Folgenden dargestellten Stellungen der Grundbacken.

#### Minimale Position

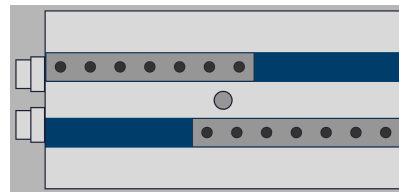
In der Draufsicht steht die obere Grundbacke rechts und die untere links.



Im Auslieferungszustand entspricht diese Stellung dem **Nullpunkt** des Moduls.

#### Maximale Position

In der Draufsicht steht die obere Grundbacke links und die untere rechts.




---

### HINWEIS

Erfolgt eine Bewegung der Greiferfinger über den Verfahrbereich hinaus, wechselt das Modul in den Fehlerzustand und meldet das Diagnose-Ereignis ERR\_SOFT\_LOW bzw. ERR\_SOFT\_HIGH zurück.

---

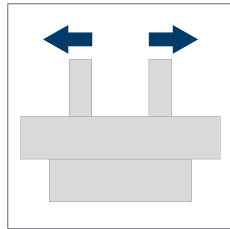
## 1.2.2 Bewegungs- und Greifrichtungen

Im Folgenden werden Bewegungs- und Greifrichtungen dargestellt.

### Bewegungsrichtungen

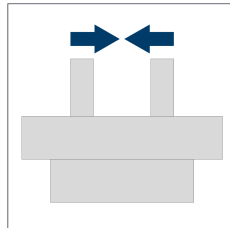
#### nach außen

Die Bewegung vom minimalen zum maximalen Positionswert entspricht der Bewegung *nach außen*.



#### nach innen

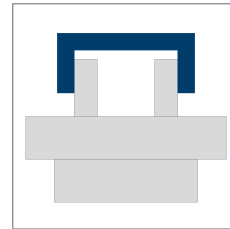
Die Bewegung vom maximalen zum minimalen Positionswert entspricht der Bewegung *nach innen*.



### Greifrichtungen

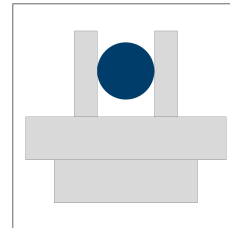
#### Innengreifen

Durch eine Bewegung nach außen kann ein Werkstück von *innen* gegriffen werden, daher die Bezeichnung *Innengreifen*.



#### Außengreifen

Durch eine Bewegung nach innen kann ein Werkstück von *außen* gegriffen werden, daher die Bezeichnung *Außengreifen*.



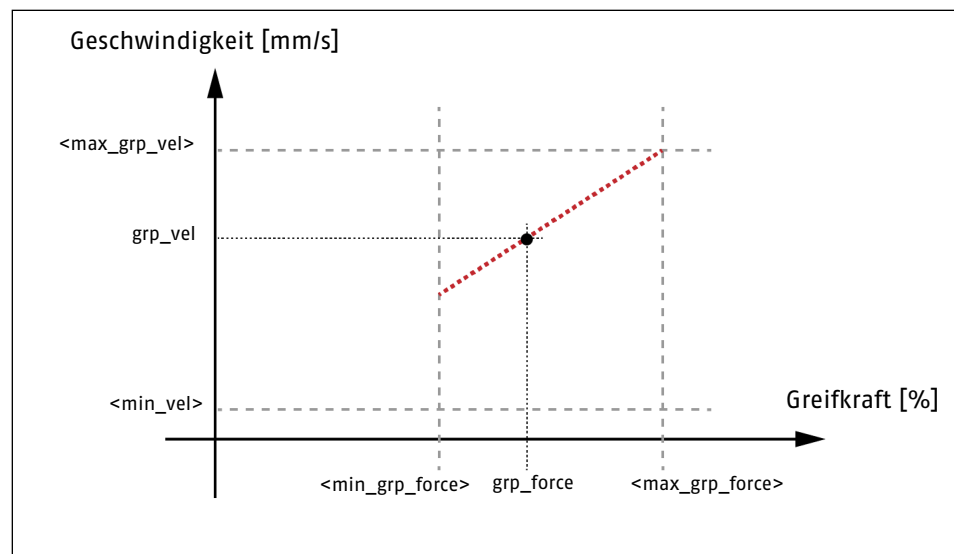
### 1.2.3 Greifmodi

Zum Greifen von Werkstücken stellt das Modul unterschiedliche Greifmodi zur Verfügung:

- BasicGrip
- StrongGrip

#### BasicGrip

BasicGrip ist der Standardgreifmodus des Moduls. Das Modul berechnet in Abhängigkeit der übergebenen Greifkraft die Greifgeschwindigkeit, mit der das Werkstück gegriffen wird. Dadurch wird der beim Greifen des Werkstücks entstehende Kraftimpuls reduziert.



*Greifgeschwindigkeit in Abhängigkeit von der Greifkraft*

#### Berechnung der Greifgeschwindigkeit im BasicGrip-Modus

Greifgeschwindigkeit = Greifkraft [%] \* <max\_grp\_vel> [mm/s]

#### StrongGrip

Dieser Modus ist nur bei Modulen der Variante "M" (mit GPE) verfügbar, ► [1.2.4 \[11\]](#).

Der StrongGrip-Modus kann verwendet werden, um Werkstücke mit einer Greifkraft größer 100 % zu greifen.

In diesem Modus verwendet das Modul immer die maximale Greifgeschwindigkeit <max\_grp\_vel>.

Aus technischen Gründen kann das Modul bei dieser Funktion maximal 2 Sekunden Nachgreifen, bevor automatisch die GPE aktiviert wird.

### 1.2.4 Greifkraft- und Positionserhaltung (GPE)

Werkstücke und Positionen werden standardmäßig durch die Antriebsregelung des Moduls gehalten. Module der Variante "M" verfügen über eine Greifkraft- und Positionserhaltung (GPE). Bei diesen Modulen kann beim Senden von Steuerbefehlen angegeben werden, ob Werkstücke und Positionen durch die Antriebsregelung *oder* durch die GPE gehalten werden sollen. Die Auswahl, auf welche Art Werkstücke und Positionen gehalten werden sollen, erfolgt über das Steuerbit "Activate grip force and position maintenance", ▶ 7.2 [📄 93].

---

#### HINWEIS

Bei Modulen ohne GPE *muss* das Steuerbit "Activate grip force and position maintenance" immer gleich 0 sein. Beim Versuch die GPE zu aktivieren, wird das Statusbit "not feasible" und der Diagnose Code WRN\_NOT\_FEASIBLE inklusive Zusatzcode zurückgemeldet.

---

### 1.2.5 Nullpunkt

Der Nullpunkt des Moduls entspricht einer Stellung der Greiferfinger, bei der der Positionswert 0 mm ausgegeben wird. Der Nullpunkt kann innerhalb einer Applikation individuell an die Gegebenheiten angepasst werden, ▶ 3.4.1 [📄 46].

## 2 Kommunikation

### 2.1 Datenaustausch

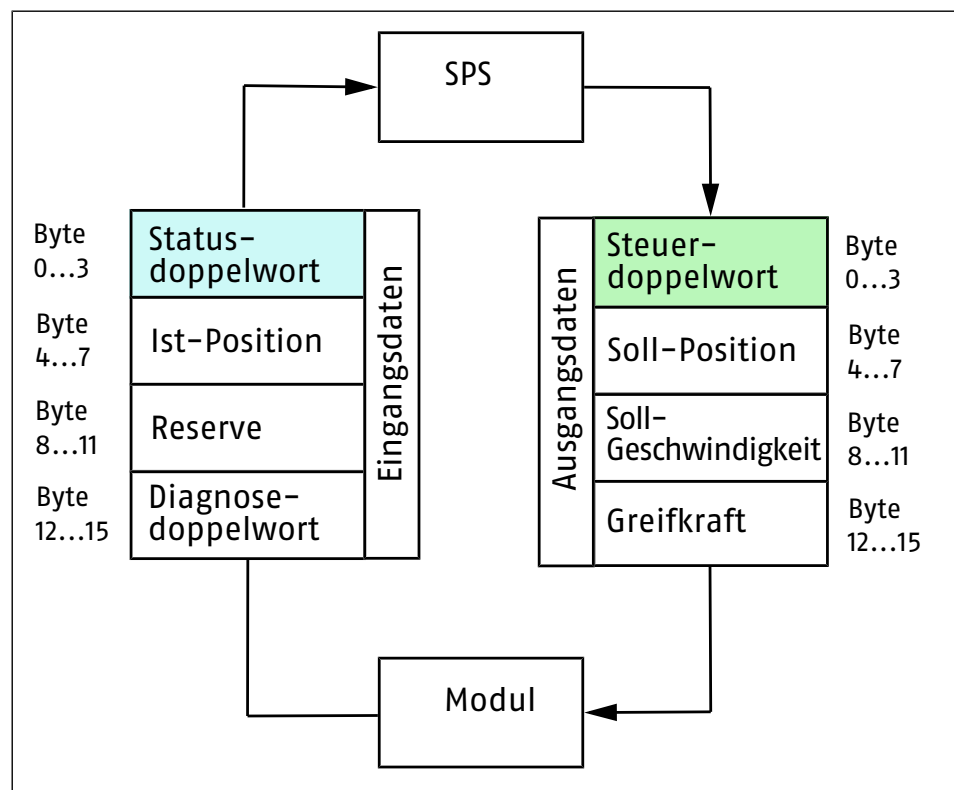
Über die integrierte Feldbus-Schnittstelle können zwischen Modul und Steuerung Daten zyklisch und azyklisch ausgetauscht werden.

#### HINWEIS

Bei Unterbrechung der Kommunikation zwischen Modul und Steuerung z. B. durch einen Kabelbruch oder durch Wechsel der Steuerung in den Zustand "Stop" führt das Modul einen Schnellstopp aus. Weiterhin wird das Diagnose-Ereignis ERR\_COMM\_LOST zurückgemeldet.

#### 2.1.1 Zyklischer Datenaustausch

Für den zyklischen Datenaustausch ist ein fester Datenrahmen für Aus- und Eingangsdaten definiert. Der Datenrahmen basiert auf der Verwendung von Datendoppelworten und ist auf eine Datenlänge von vier Doppelworten festgelegt.



Zyklischer Datenaustausch

Modbus RTU unterscheidet **nicht** zwischen zyklischem und azyklischem Datenaustausch. Es gibt nur die beschriebenen Registerzugriffe, ▶ 2.1.2 [21].

Für eine zyklische Kommunikation muss entsprechend explizit wiederholt auf die zyklischen Parameter <plc\_sync\_input> (► 4.2 [□ 52]) und <plc\_sync\_output> (► 4.2 [□ 65]) lesend bzw. schreibend zugegriffen werden.

Weiterführende Informationen zur Datenübertragung- und Interpretation siehe folgende Abschnitte.

### 2.1.1.1 Zyklische Ausgangsdaten

Die zyklischen Ausgangsdaten werden von der SPS an das Modul übertragen und somit Steuerbefehle an das Modul gesendet. Praktische Anwendungsbeispiele hierzu siehe Kapitel ► 7.1 [□ 81].

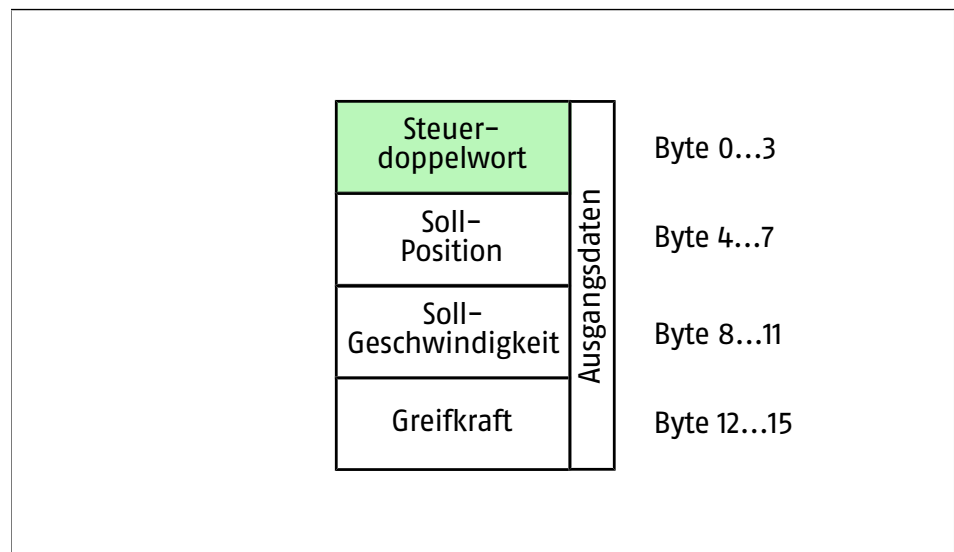
An das Modul gesendete Steuerbefehle können zulässig oder unzulässig sein.

- Zulässige Steuerbefehle werden vom Modul umgesetzt.
- Unzulässige Steuerbefehle werden nicht umgesetzt. Der SPS wird dies durch Setzen des Statusbits "not feasible" angezeigt. Weiterhin wird über das Diagnosedoppelwort die Warnung WRN\_NOT\_FEASIBLE inklusive Zusatzcode übertragen, ► 6.1 [□ 76].

**Umsetzung der Steuerbefehle**

**Datenrahmen**

Der Datenrahmen zyklischer Ausgangsdaten setzt sich zusammen aus dem Steuerdoppelwort und Bewegungsparametern.



*Datenrahmen zyklischer Ausgangsdaten*



## Übersicht Steuerdoppelwort

| Wort | Byte | Bit | Zyklische Ausgangsdaten   |
|------|------|-----|---|
| 0    | 0    | 0   | ▶ fast stop [ 90]<br>EN – kurz: fast stop<br>DE – kurz: Schnellstopp  |
|      |      | 1   | ▶ stop [ 90]<br>EN – kurz: stop<br>DE – kurz: Stopp   |
|      |      | 2   | ▶ acknowledge [ 90]<br>EN – kurz: ack<br>DE – kurz: Quittieren  |
|      |      | 3   | ▶ prepare for shutdown [ 90]<br>EN – kurz: prep shutdown<br>DE – kurz: Herunterfahren vorbereiten           |
|      |      | 4   | ▶ softreset [ 90]<br>EN – kurz: softreset<br>DE – kurz: Neustart  |
|      |      | 5   | ▶ release for manual movement [ 91]<br>EN – kurz: release manual movement<br>DE – kurz: Man. Bwg. freigeben |
|      |      | 6   | ▶ repeat command toggle [ 91]<br>EN – kurz: rpt cmd tgl<br>DE – kurz: Kdo. wiederh.                         |
| 0    | 1    | 8   | ▶ jog mode negative [ 91]<br>EN – kurz: jog -<br>DE – kurz: Tipp -  |
|      |      | 9   | ▶ jog mode positive [ 91]<br>EN – kurz: jog +<br>DE – kurz: Tipp +  |
|      |      | 10  | reserved  |
|      |      | 11  | ▶ release workpiece [ 92]<br>EN – kurz: release wp<br>DE – kurz: Werkst. freigeben                          |
|      |      | 12  | ▶ grip workpiece [ 92]<br>EN – kurz: grp wp<br>DE – kurz: Werkst. greifen                                   |

| Wort | Byte | Bit | Zyklische Ausgangsdaten  |
|------|------|-----|--|
|      |      | 13  | ▶ move to absolute position [ 92]<br>EN – kurz: pos absolute<br>DE – kurz: Pos. absolut                            |
|      |      | 14  | ▶ move to relative position [ 92]<br>EN – kurz: pos relative<br>DE – kurz: Pos. relativ                            |
|      |      | 15  | reserved   |
| 1    | 2    | 16  | ▶ grip workpiece at expected position [ 92]<br>EN – kurz: grp wp at pos<br>DE – kurz: Werkst. greifen an erw. Pos. |
|      |      | 17  | reserved   |
|      |      | 18  | reserved   |
|      |      | 19  | reserved   |
|      |      | 20  | reserved   |
|      |      | 21  | reserved   |
|      |      | 22  | reserved   |
|      |      | 23  | reserved   |
| 1    | 3    | 24  | reserved   |
|      |      | 25  | reserved   |
|      |      | 26  | reserved   |
|      |      | 27  | reserved   |
|      |      | 28  | reserved   |
|      |      | 29  | reserved   |
|      |      | 30  | ▶ brake test [ 93]<br>EN – kurz: brake test<br>DE – kurz: Bremsentest  |
|      |      | 31  | ▶ Activate grip force and position maintenance [ 93]<br>EN – kurz: activate GPE<br>DE – kurz: GPE aktivieren       |

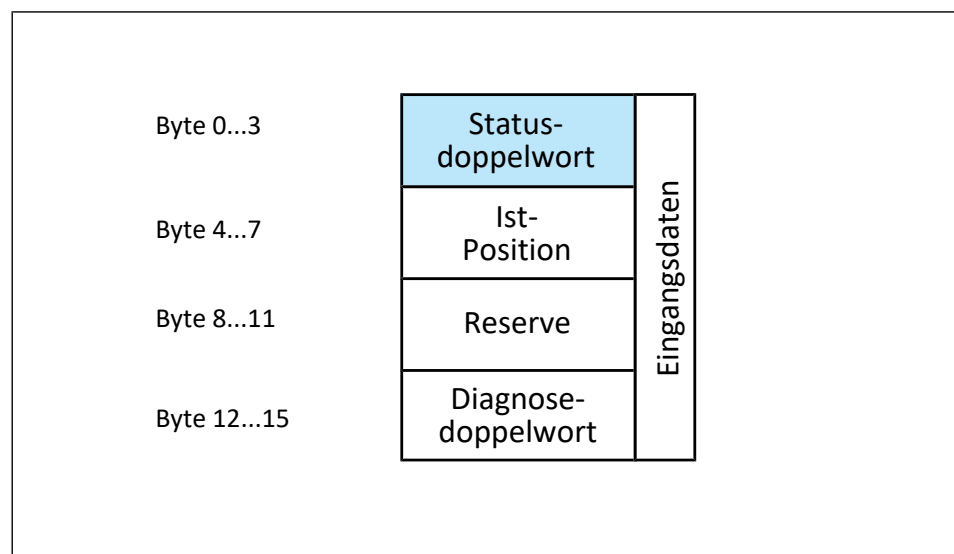
- Soll-Position**
  - In den Bytes 4 – 7 der zyklischen Ausgangsdaten werden Daten übertragen, die zu Positionierungszwecken genutzt werden, ▶ 4.2 [ 51].
  - Das Datenformat des Parameters ist *signed 32 Bit* und stellt einen Wert in Mikrometer [µm] dar. (1000 µm ≙ 1 mm)
- Soll-Geschwindigkeit**
  - In den Bytes 8 – 11 der zyklischen Ausgangsdaten wird der Soll-Geschwindigkeitswert einer Bewegung übertragen, ▶ 4.2 [ 51].
  - Das Datenformat des Parameters ist *signed 32 Bit* und stellt einen Wert in Mikrometer pro Sekunde [µm/s] dar. (1000 µm/s ≙ 1 mm/s)
- Greifkraft**
  - In den Bytes 12 – 15 der zyklischen Ausgangsdaten wird die Greifkraft übertragen, mit der ein Werkstück gegriffen werden soll, ▶ 3.3.1 [ 33].
  - Das Datenformat des Parameters ist *signed 32 Bit* und stellt einen Wert in Prozent [%] dar. Der prozentuale Wert bezieht sich auf den Parameter <max\_grp\_force>, ▶ 4.2 [ 59]. Bei Einstellung einer ungültigen Greifkraft wird im Diagnosespeicher ein Eintrag erstellt, der sich ebenfalls auf die Kraftgrenzen in Newton bezieht.

### 2.1.1.2 Zyklische Eingangsdaten

Die zyklischen Eingangsdaten werden vom Modul an die Steuerung übertragen. Dadurch bekommt die SPS eine Rückmeldung vom Modul und kann darauf entsprechend reagieren.

#### Datenrahmen

Der Datenrahmen zyklischer Eingangsdaten setzt sich zusammen aus dem Statusdoppelwort und Modulrückmeldungen.



Datenrahmen zyklischer Eingangsdaten



## Übersicht Statusdoppelwort

| Wort | Byte | Bit | Zyklische Eingangsdaten   |
|------|------|-----|---|
| 0    | 0    | 0   | ▶ ready for operation [ 93]<br>EN – kurz: ready for op<br>DE – kurz: Betriebsbereit                             |
|      |      | 1   | ▶ control authority fieldbus [ 93]<br>EN – kurz: ctrl authority fb<br>DE – kurz: Feldbus                        |
|      |      | 2   | ▶ ready for shutdown [ 93]<br>EN – kurz: ready for sd<br>DE – kurz: Abschaltbereit                              |
|      |      | 3   | ▶ not feasible [ 94]<br>EN – kurz: not feasible<br>DE – kurz: Nicht durchführb.                                 |
|      |      | 4   | ▶ command successfully processed [ 94]<br>EN – kurz: cmd success<br>DE – kurz: Kdo. erfolgreich                 |
|      |      | 5   | ▶ command received toggle [ 94]<br>EN – kurz: cmd rcvd tgl<br>DE – kurz: Kommandowechsel                        |
|      |      | 6   | ▶ warning [ 94]<br>EN – kurz: warning<br>DE – kurz: Warnung   |
| 0    | 1    | 7   | ▶ error [ 94]<br>EN – kurz: error<br>DE – kurz: Fehler  |
|      |      | 8   | ▶ released for manual movement [ 94]<br>EN – kurz: manual movement released<br>DE – kurz: Man. Bwg. freigegeben |
|      |      | 9   | ▶ software limit reached [ 95]<br>EN – kurz: softlimit reached<br>DE – kurz: Softlimit                          |
|      |      | 10  | reserved  |
|      |      | 11  | ▶ no workpiece detected [ 95]<br>EN – kurz: no wp detected<br>DE – kurz: Kein Werkstück                         |
|      |      | 12  | ▶ workpiece gripped [ 95]<br>EN – kurz: wp gripped<br>DE – kurz: Gegriffen                                      |

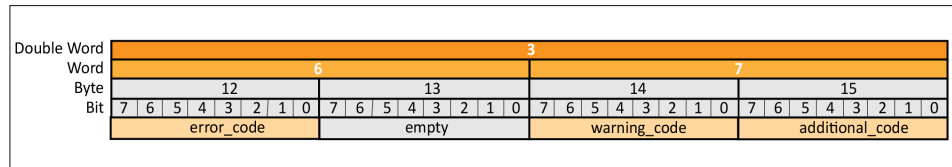
| Wort | Byte | Bit | Zyklische Eingangsdaten  |
|------|------|-----|--|
|      |      | 13  | ▶ position reached [📄 95]<br>EN – kurz: pos reached<br>DE – kurz: Positioniert                                 |
|      |      | 14  | ▶ workpiece pre-grip started [📄 95]<br>EN – kurz: wp pre-grip started<br>DE – kurz: Nachgreifen                |
|      |      | 15  | reserved   |
| 1    | 2    | 16  | ▶ workpiece lost [📄 95]<br>EN – kurz: wp lost<br>DE – kurz: Werkst. verloren                                   |
|      |      | 17  | ▶ wrong workpiece gripped [📄 96]<br>EN – kurz: wrong wp gripped<br>DE – kurz: Falsches Werkst.                 |
|      |      | 18  | reserved   |
|      |      | 19  | reserved   |
|      |      | 20  | reserved   |
|      |      | 21  | reserved   |
|      |      | 22  | reserved   |
|      |      | 23  | reserved   |
| 1    | 3    | 24  | reserved   |
|      |      | 25  | reserved   |
|      |      | 26  | reserved   |
|      |      | 27  | reserved   |
|      |      | 28  | reserved   |
|      |      | 29  | reserved   |
|      |      | 30  | reserved   |
|      |      | 31  | ▶ Grip force and position maintenance activated [📄 96]<br>EN – kurz: GPE activated<br>DE – kurz: GPE aktiviert |

**Ist-Position**

- In den Bytes 4 – 7 der zyklischen Eingangsdaten wird die aktuelle Ist-Position des Moduls übertragen, ▶ 4 [□ 51].
- Das Datenformat des Parameters ist *signed 32 Bit* und stellt einen Wert in Mikrometer [ $\mu\text{m}$ ] dar ( $1000 \mu\text{m} \triangleq 1 \text{ mm}$ ).

**Reserve**

- In den Bytes 8 – 11 der zyklischen Eingangsdaten werden derzeit keine Nutzdaten übertragen.

**Diagnosedoppelwort***Codereihenfolge Diagnosedoppelwort*

- In den Bytes 12 – 15 der zyklischen Eingangsdaten wird das Diagnosedoppelwort übertragen.
- Eine Verwechslung zwischen Warnungs- und Fehlercodes ist ausgeschlossen, da jeder Code nur einmal vergeben ist.

**HINWEIS**

Falls über das Byte 14 die Warnung

WRN\_NOT\_FEASIBLE, ▶ 6.1 [□ 76] ausgegeben wird, kann über das Byte 15 (additional\_code) die Ursache für die

Nichtdurchführbarkeit ausgelesen werden. Die Liste der Ursachen befindet sich im Anhand dieses Dokuments, ▶ 7.4 [□ 96].

**2.1.2 Azyklischer Datenaustausch Modbus RTU**

Für den Datenaustausch implementiert die Firmware das Modbus RTU Kommunikationsprotokoll nach [www.modbus.com](http://www.modbus.com).

Modbus RTU organisiert sich anhand von Registern, auf die lesend oder schreibend zugegriffen werden kann. Ein Register ist dabei immer 16-Bit breit.

**Funktionscodes**

Der Zugriff auf die Register erfolgt über Funktionscodes (function-codes), welche dann wiederum entweder auf einzelne oder mehrere Register zugreifen. Folgende Funktionscodes werden von der Firmware unterstützt:

- "Read Holding Registers" dezimal 03 (0x03 hexadezimal) zum Lesen von mehreren Registern
- "Write Multiple Register" dezimal 16 (0x10 hexadezimal) zum Schreiben von mehreren Registern

**Ansatz**

Das Modul verwendet dabei einen einfachen Ansatz, um seine Funktionalität über Modbus RTU zur Verfügung zu stellen:

- Register-Adressen entsprechen den bekannten Parameter-IDs, ▶ 4.2 [□ 51].

- Parameter haben unterschiedliche Datentypen (16-Bit Integer, 32-Bit Float, Arrays von 32-Bit Integer, ...) und somit auch unterschiedliche Längen. Parameter können durch Zugriff auf mehrere aufeinander folgende Register gelesen oder geschrieben werden. Da Modbus RTU in Einheiten von 16-Bit Registern rechnet, müssen die Größenangaben entsprechend angepasst werden:
  - Ein 32-Bit Float ist daher 2-Register-groß, ein 32-Bit-Integer Array mit 4 Array-Elementen entsprechend 8 Register.
- Zugriffsbeschränkungen wie z. B. bei nur lesbaren Parametern gelten weiterhin.
- Parameter können immer nur vollständig gelesen oder geschrieben werden. Auch bei Array-Parametern kann daher nicht auf einzelne Array-Elemente zugegriffen werden, sondern es müssen immer alle Elemente gemeinsam gelesen oder geschrieben werden.
- Parameter-Werte werden im Little-Endian Format übertragen, d. h. mit dem niederwertigsten Byte zuerst.
- Modbus RTU spezifische Protokolldaten, wie z. B. Register-Adressen werden im Big-Endian Format übertragen, d. h. mit dem höchstwertigen Byte zuerst.

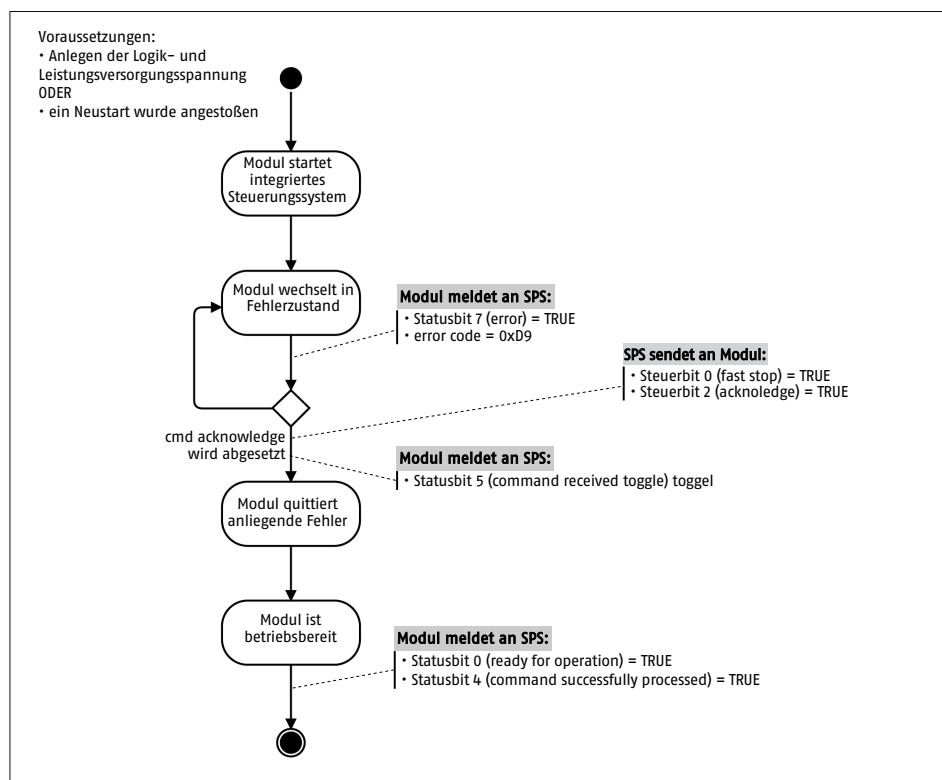
## 3 Modulfunktionen

### 3.1 Booten, Abschalten und Neustart

#### 3.1.1 Booten und Betriebsbereitschaft herstellen

##### Kurzbeschreibung

Beim Booten werden nach dem Hochfahren der Elektronik die interne Hardware und die angeschlossenen Kommunikationsschnittstellen überprüft. Das Modul befindet sich nach dem Booten im Fehlerzustand. Aus diesem Zustand kann durch Quittieren die Betriebsbereitschaft hergestellt werden. Folgendes Beispiel zeigt den Ablauf zum Herstellen der Betriebsbereitschaft:



Modul booten und Betriebsbereitschaft herstellen

##### Anstoßen

Das Booten kann hardwareseitig durch Anlegen der Logik-Versorgungsspannung oder softwareseitig durch einen Neustart angestoßen werden, ▶ 3.1.3 [26].

Befindet sich das Modul nach dem Booten im Fehlerzustand, wird das Herstellen der Betriebsbereitschaft durch Setzen des Steuerbits "acknowledge" (Bit 2) angestoßen, ▶ 7.2 [90].

---

## HINWEIS

- Um ein unerwartetes Verhalten des Moduls zu verhindern, sollten während des Bootens alle Steuerbits gleich 0 zyklisch an das Modul übertragen werden.
  - Das Modul setzt beim Booten alle Steuerbits intern auf den Zustand 1. Dadurch können bei weiterlaufender zyklischer Übertragung des Steuerworts durch die externe Steuerung keine ungewollten Anfragen ausgelöst werden. Nur das Statusbit "command received toggle" spiegelt den invertierten Wert des initial empfangenen Steuerbits "fast stop" wider.  
Hintergrund: Überträgt die Steuerung während bzw. nach dem Booten eine 0 in das Steuerbit "fast stop", so wird dieser Übergang 1 -> 0 als ein Anfordern eines "fast stop" interpretiert und entsprechend wird "command received toggle" vom Startwert 0 auf 1 gesetzt.
- 

## Modulrückmeldung

- Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt. Diese Bestätigung erfolgt unabhängig davon, ob der Steuerbefehl im Anschluss erfolgreich beendet oder überhaupt bearbeitet werden kann.
- War das Herstellen der Betriebsbereitschaft *erfolgreich*, wird dies durch Setzen des Statusbits "ready for operation" angezeigt. Das Statusbit "error" und der angezeigte Diagnose Code werden zurückgesetzt.
- War das Herstellen der Betriebsbereitschaft *erfolglos*, bleibt das Modul im Fehlerzustand. Das Statusbits "error" und ein entsprechender Diagnose Code werden weiterhin angezeigt. In diesem Fall den SCHUNK Service kontaktieren.

### 3.1.2 Abschalten

## Kurzbeschreibung

Beim kontrollierten Abschalten des Moduls werden für den Betrieb notwendige Daten dauerhaft abgespeichert. Meldet das Modul zurück, dass es zum Abschalten bereit ist, kann die Logik-Versorgungsspannung abgezogen oder softwareseitig ein Neustart angestoßen werden.

---

## HINWEIS

Sobald das Modul die Abschaltbereitschaft anzeigt, können ausgenommen vom Neustart (► [3.1.3 \[26\]](#)) keine weiteren Funktionen angestoßen werden. Weiterhin werden keine Änderungen von Parameterwerten übernommen.

---

**Anstoßen**

Kontrolliertes Abschalten ist aus definierten Systemzuständen heraus zulässig und wird durch Setzen des Steuerbits "prepare for shutdown" (Bit 3) angestoßen, ▶ 7.2 [📄 90].

**Systemzustand**

Um das Vorbereiten zum Abschalten anzustoßen, muss sich das Modul in einem der folgenden Zustände befinden:

- Positionserhaltung
- Werkstück-Halten
- Fehlerzustand

---

**HINWEIS**

- Wird bei Modulen *mit* GPE das Abschalten aus dem Werkstück-Halten heraus angestoßen, speichert das Modul diese Information. Nach dem Neustart wird das entsprechende Statusbit "workpiece gripped" oder "wrong workpiece gripped" wieder angezeigt.
  - Wird bei Modulen *ohne* GPE das Abschalten aus dem Werkstück-Halten heraus angestoßen, speichert das Modul keine Informationen bezüglich eines gegriffenen Werkstücks.
- 

---

**HINWEIS**

- Bei Modulen *ohne* GPE oder bei einem harten Neustart (Spannung abziehen, Spannung wieder anlegen) kann der zuletzt gesendete Greifbefehl wiederholt werden.
  - Wurde das Werkstück nicht verloren, wird dies durch das Statusbit "workpiece gripped" oder "wrong workpiece gripped" angezeigt.
  - Wurde das Werkstück verloren, wird dies durch das Statusbit "no workpiece detected" angezeigt.
- 

**Modulrückmeldung**

- Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt. Diese Bestätigung erfolgt unabhängig davon, ob der Steuerbefehl im Anschluss erfolgreich beendet oder überhaupt bearbeitet werden kann.
- War das Vorbereiten zum Abschalten des Moduls *erfolgreich*, wird dies durch Setzen des Statusbits "ready for shutdown" angezeigt
- War das Vorbereiten zum Abschalten des Moduls *erfolglos*, wird dies durch Setzen des Statusbits "error" und dem entsprechenden Diagnose Code angezeigt. In diesem Fall den SCHUNK Service kontaktieren.

### 3.1.3 Neustart

|                         |   |
|-------------------------|---|
| <b>Kurzbeschreibung</b> | Beim Neustart des Moduls werden für den Betrieb notwendige Daten dauerhaft abgespeichert und anschließend wird das Booten eingeleitet, siehe Kapitel ▶ 3.1.1 [☐ 23].  |
| <b>Anstoßen</b>         | Der Neustart des Moduls ist aus definierten Systemzuständen heraus zulässig und wird durch Setzen des Steuerbits "softreset" (Bit 4) angestoßen, ▶ 7.2 [☐ 90].  |
| <b>Systemzustand</b>    | Um den Neustart anzustoßen, muss sich das Modul in einem der folgenden Zustände befinden: <ul style="list-style-type: none"><li>• Bereitschaft zum Abschalten</li><li>• Positionserhaltung</li><li>• Werkstück-Halten</li><li>• Fehlerzustand</li></ul> |

---

#### HINWEIS

- Wird bei Modulen *mit* GPE das Abschalten aus dem Werkstück-Halten heraus angestoßen, speichert das Modul diese Information. Nach dem Neustart wird das entsprechende Statusbit "workpiece gripped" oder "wrong workpiece gripped" wieder angezeigt.
  - Wird bei Modulen *ohne* GPE das Abschalten aus dem Werkstück-Halten heraus angestoßen, speichert das Modul keine Informationen bezüglich eines gegriffenen Werkstücks.
- 

#### HINWEIS

- Bei Modulen *ohne* GPE oder bei einem harten Neustart (Spannung abziehen, Spannung wieder anlegen) kann der zuletzt gesendete Greifbefehl wiederholt werden.
  - Wurde das Werkstück nicht verloren, wird dies durch das Statusbit "workpiece gripped" oder "wrong workpiece gripped" angezeigt.
  - Wurde das Werkstück verloren, wird dies durch das Statusbit "no workpiece detected" angezeigt.
-

## 3.2 Bewegungsfunktionen

### 3.2.1 Tipp-Betrieb

|  |   |
|--|---|
| <b>Kurzbeschreibung</b>  | Beim Tipp-Betrieb wird eine Bewegungsfahrt nach außen oder innen ausgeführt, solange eines der entsprechenden Steuerbits gesetzt ist. Falls durch die Bewegung die minimale oder maximale Position erreicht wird, endet die Bewegung automatisch.   |
| <hr/>  |   |
| <b>HINWEIS</b>   |   |
| Der Tipp-Betrieb ist ausschließlich eine Funktion zur Inbetriebnahme des Moduls. Diese Funktion <b>nicht</b> während des automatisierten Betriebs verwenden! |   |
| <hr/>  |   |
| <b>Anstoßen</b>  | <ul style="list-style-type: none"> <li>• Der Tipp-Betrieb nach außen wird durch Setzen des Steuerbits "jog mode positive" (Bit 9) angestoßen, ▶ 7.2 [ 91].</li> <li>• Der Tipp-Betrieb nach innen wird durch Setzen des Steuerbits "jog mode negative" (Bit 8) angestoßen, ▶ 7.2 [ 91].</li> </ul>  |
| <b>Bewegungsparameter</b>  | <p>Folgende Bewegungsparameter müssen an das Modul zyklisch übertragen werden:</p> <ul style="list-style-type: none"> <li>• <math>\langle \text{min\_vel} \rangle \leq \text{Bewegungsgeschwindigkeit} [\mu\text{m/s}] \leq \langle \text{max\_grp\_vel} \rangle</math></li> <li>• Verwendung GPE <ul style="list-style-type: none"> <li>– Modul mit GPE: Verwendung der GPE wird durch den Zustand des Steuerbits "Activate grip force and position maintenance" (Bit 31) angegeben, ▶ 7.2 [ 93].</li> <li>– Modul ohne GPE: Steuerbit "Activate grip force and position maintenance" muss gleich 0 sein.</li> </ul> </li> </ul> |
| <b>Beenden</b>   | <p>Der Tipp-Betrieb wird durch folgende Ereignisse beendet:</p> <ul style="list-style-type: none"> <li>• Zurücksetzen des Steuerbits "jog mode positive" bzw. "jog mode negative"</li> <li>• Durch Erreichen der minimalen oder maximalen Position.</li> </ul>  |
| <b>Modulrückmeldung</b>  | <ul style="list-style-type: none"> <li>• Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt. Diese Bestätigung erfolgt unabhängig davon, ob der Steuerbefehl im Anschluss erfolgreich beendet oder überhaupt bearbeitet werden kann.</li> <li>• Das Beenden des Tipp-Betriebs durch <i>Zurücksetzen eines der Steuerbits</i> wird durch Setzen der Statusbits "position reached" und "command successfully processed" angezeigt. Weiterhin wechselt das Statusbit "command received toggle" den Zustand.</li> </ul>   |

- Das Beenden des Tipp-Betriebs durch *Erreichen der minimalen oder maximalen Position* wird durch Setzen der Statusbits "position reached" und "command successfully processed" angezeigt. Weiterhin meldet das Modul den Diagnose Code WRN\_POS\_LIMIT zurück.

**Mögliche Diagnose-Ereignisse**

Ereignisse, die zu Warnungen und/oder Fehlern führen, werden durch die Diagnose erkannt. Im Folgenden sind alle möglichen Diagnose-Ereignisse aufgelistet.

| Diagnose Ereignis                        | Diagnose Code *               |
|--|-------------------------------|
| Unteres Softwarelimit wird erreicht.     | ERR_SOFT_LOW                  |
| Oberes Softwarelimit wird erreicht.      | ERR_SOFT_HIGH                 |
| Antrieb blockiert während Bewegung.      | ERR_MOVE_BLOCKED              |
| Senden eines unzulässigen Steuerbefehls. | WRN_NOT_FEASIBLE + Zusatzcode |
| Bewegungsabbruch durch Benutzer.         | ERR_FAST_STOP                 |

\* Weitere Informationen siehe Kapitel ▶ 6 [87].

**3.2.2 Positionsfahrt absolut**

**Kurzbeschreibung**

Beim absoluten Positionieren verfährt das Modul auf den zyklisch übergebenen Positionswert. Dieser Positionswert bezieht sich auf den parametrisierten Nullpunkt des Moduls. Ein praktisches Anwendungsbeispiel dazu ist beschrieben im Kapitel ▶ 7.1 [82], Beispiel 1.

**HINWEIS**

Das Verwenden einer Positionsfahrt zum Greifen von Werkstücken ist eine Fehlanwendung und führt zu einem Fehler des Moduls.

**Anstoßen**

Das absolute Positionieren wird durch Setzen des Steuerbits "move to absolute position" (Bit 13) angestoßen, ▶ 7.2 [92]. Bei gesetztem Steuerbit "move to absolute position" kann ein erneutes absolutes Positionieren durch einen Wechsel des Steuerbits "repeat command toggle" (Bit 6) angestoßen werden, ▶ 7.2 [91].

- Bewegungsparameter** Folgende Bewegungsparameter müssen an das Modul zyklisch übertragen werden:
- $\langle \text{min\_pos} \rangle \leq \text{absolute Position } [\mu\text{m}] \leq \langle \text{max\_pos} \rangle$
  - $\langle \text{min\_vel} \rangle \leq \text{Bewegungsgeschwindigkeit } [\mu\text{m/s}] \leq \langle \text{max\_vel} \rangle$
  - Verwendung GPE
    - Modul mit GPE: Verwendung der GPE wird durch den Zustand des Steuerbits "Activate grip force and position maintenance" (Bit 31) angegeben, ▶ 7.2 [93].
    - Modul ohne GPE: Steuerbit "Activate grip force and position maintenance" muss gleich 0 sein.
- Beenden** Das absolute Positionieren wird durch folgende Ereignisse beendet:
- Zielposition wurde erreicht
  - Setzen des Steuerbits "stop"
- Modulrückmeldung**
- Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt. Diese Bestätigung erfolgt unabhängig davon, ob der Steuerbefehl im Anschluss erfolgreich beendet oder überhaupt bearbeitet werden kann.
  - Das Erreichen der Zielposition wird durch Setzen der Statusbits "position reached" und "command successfully processed" angezeigt.
- Mögliche Diagnose-Ereignisse** Ereignisse, die zu Warnungen und/oder Fehlern führen, werden durch die Diagnose erkannt. Im Folgenden sind alle möglichen Diagnose-Ereignisse aufgelistet.

| Diagnose Ereignis                                  | Diagnose Code *               |
|--|-------------------------------|
| Positionieren dauert zu lange.                     | ERR_MOV_ABORT_TO              |
| Unteres Softwarelimit wird erreicht.               | ERR_SOFT_LOW                  |
| Oberes Softwarelimit wird erreicht.                | ERR_SOFT_HIGH                 |
| Antrieb ist bei Bewegungsbeginn bereits blockiert. | ERR_MOVE_BLOCKED              |
| Antrieb blockiert während Bewegung.                | ERR_MOVE_BLOCKED              |
| Senden eines unzulässigen Steuerbefehls.           | WRN_NOT_FEASIBLE + Zusatzcode |
| Bewegungsabbruch durch Benutzer.                   | ERR_FAST_STOP                 |

\* Weitere Informationen siehe Kapitel ▶ 6 [75].

### 3.2.3 Positionsfahrt relativ

#### Kurzbeschreibung

Beim relativen Positionieren verfährt das Modul von der aktuellen Position um den zyklisch übergebenen und vorzeichenbehafteten Positionswert. Ein praktisches Anwendungsbeispiel dazu ist beschrieben im Kapitel ▶ 7.1 [83], Beispiel 2.

---

#### HINWEIS

Das Verwenden einer Positionsfahrt zum Greifen von Werkstücken ist eine Fehlanwendung und führt zu einem Fehler des Moduls.

---

#### Anstoßen

Das relative Positionieren wird durch Setzen des Steuerbits "move to relative position" (Bit 14) angestoßen, ▶ 7.2 [92].

Bei gesetztem Steuerbit "move to relative position" kann ein erneutes relatives Positionieren durch einen Wechsel des Steuerbits "repeat command toggle" (Bit 6) angestoßen werden, ▶ 7.2 [91].

#### Bewegungsparameter

Folgende Bewegungsparameter müssen an das Modul zyklisch übertragen werden:

- vorzeichenbehaftete relative Position [ $\mu\text{m}$ ]
- $\langle \text{min\_vel} \rangle \leq \text{Bewegungsgeschwindigkeit} [\mu\text{m/s}] \leq \langle \text{max\_vel} \rangle$
- Verwendung GPE
  - Modul mit GPE: Verwendung der GPE wird durch den Zustand des Steuerbits "Activate grip force and position maintenance" (Bit 31) angegeben, ▶ 7.2 [93].
  - Modul ohne GPE: Steuerbit "Activate grip force and position maintenance" muss gleich 0 sein.

---

#### HINWEIS

- Die übertragene vorzeichenbehaftete relative Position muss so gewählt werden, dass der zulässige Bewegungsbereich von  $\langle \text{min\_pos} \rangle$  bis  $\langle \text{max\_pos} \rangle$  nicht verlassen wird.
  - Falls die Position nicht innerhalb des zulässigen Bewegungsbereich liegt, setzt das Modul das Statusbit "not feasible" und meldet den Diagnose Code WRN\_NOT\_FEASIBLE inklusive Zusatzcode zurück.
- 

#### Beenden

Das relative Positionieren wird durch folgende Ereignisse beendet:

- Zielposition wurde erreicht
- Setzen des Steuerbits "stop"

**Modulrückmeldung**

- Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt. Diese Bestätigung erfolgt unabhängig davon, ob der Steuerbefehl im Anschluss erfolgreich beendet oder überhaupt bearbeitet werden kann.
- Das Erreichen der Zielposition wird durch Setzen der Statusbits "position reached" und "command successfully processed" angezeigt.

**Mögliche Diagnose-Ereignisse**

Ereignisse, die zu Warnungen und/oder Fehlern führen, werden durch die Diagnose erkannt. Im Folgenden sind alle möglichen Diagnose-Ereignisse aufgelistet.

| Diagnose Ereignis                                  | Diagnose Code *               |
|--|-------------------------------|
| Positionieren dauert zu lange.                     | ERR_MOV_ABORT_TO              |
| Unteres Softwarelimit wird erreicht.               | ERR_SOFT_LOW                  |
| Oberes Softwarelimit wird erreicht.                | ERR_SOFT_HIGH                 |
| Antrieb ist bei Bewegungsbeginn bereits blockiert. | ERR_MOVE_BLOCKED              |
| Antrieb blockiert während Bewegung.                | ERR_MOVE_BLOCKED              |
| Senden eines unzulässigen Steuerbefehls.           | WRN_NOT_FEASIBLE + Zusatzcode |
| Bewegungsabbruch durch Benutzer.                   | ERR_FAST_STOP                 |

\* Weitere Informationen siehe Kapitel ▶ 6 [75].

### 3.2.4 Kontrolliert anhalten

- Kurzbeschreibung** Beim kontrollierten Anhalten wird die aktuelle Bewegung schnellstmöglich bis zum Stillstand geregelt verzögert.
- Anstoßen** Das kontrollierte Anhalten wird durch Setzen des Steuerbits "stop" (Bit 1) angestoßen, ▶ 7.2 [📄 90].
- Bewegungsparameter** Folgende Bewegungsparameter müssen an das Modul zyklisch übertragen werden:
- Verwendung GPE
    - Modul mit GPE: Verwendung der GPE wird durch den Zustand des Steuerbits "Activate grip force and position maintenance" (Bit 31) angegeben, ▶ 7.2 [📄 93].
    - Modul ohne GPE: Steuerbit "Activate grip force and position maintenance" muss gleich 0 sein.
- Beenden** Das kontrollierte Anhalten wird automatisch mit Erreichen des Stillstandes beendet.
- Modulrückmeldung**
- Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt. Diese Bestätigung erfolgt unabhängig davon, ob der Steuerbefehl im Anschluss erfolgreich beendet oder überhaupt bearbeitet werden kann.
  - Das kontrollierte Anhalten einer aktiven Bewegung wird durch Setzen der Statusbits "position reached" und "command successfully processed" angezeigt.
- Mögliche Diagnose-Ereignisse** Ereignisse, die zu Warnungen und/oder Fehlern führen, werden durch die Diagnose erkannt. Im Folgenden sind alle möglichen Diagnose-Ereignisse aufgelistet.

| Diagnose Ereignis                        | Diagnose Code *               |
|--|-------------------------------|
| Kontrolliertes Anhalten dauert zu lange  | ERR_MOV_ABORT_TO              |
| Senden eines unzulässigen Steuerbefehls. | WRN_NOT_FEASIBLE + Zusatzcode |
| Bewegungsabbruch durch Benutzer.         | ERR_FAST_STOP                 |

\* Weitere Informationen siehe Kapitel ▶ 6 [📄 75].

### 3.2.5 Bewegung abbrechen

|                         |   |
|-------------------------|---|
| <b>Kurzbeschreibung</b> | Beim Bewegungsabbruch wird die aktuelle Bewegung in den Stillstand gezwungen.   |
| <b>Anstoßen</b>         | Da das Steuerbit "fast stop" drahtbruchsicher und damit "low-aktiv" umgesetzt ist, wird der Abbruch einer aktiven Bewegung durch Zurücksetzen des Steuerbits "fast stop" (1 → 0) angestoßen, ▶ 7.2 [☐ 90].  |
| <b>Modulrückmeldung</b> | <ul style="list-style-type: none"> <li>• Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt. Diese Bestätigung erfolgt unabhängig davon, ob der Steuerbefehl im Anschluss erfolgreich beendet oder überhaupt bearbeitet werden kann.</li> <li>• Der Bewegungsabbruch wird durch Setzen des Statusbits "error" in Verbindung mit dem Diagnose Code ERR_FAST_STOP angezeigt.</li> </ul> |

## 3.3 Handhabung eines Werkstücks

### 3.3.1 Werkstück-Greifen (einfache Greiffahrt)

|                         |   |
|-------------------------|---|
| <b>Kurzbeschreibung</b> | Beim Werkstück-Greifen wird ein Werkstück <b>ohne</b> Angabe der Werkstückposition mit dem angegebenen Greifkraftwert gegriffen. Praktische Anwendungsbeispiele dazu sind beschrieben im Kapitel ▶ 7.1 [☐ 84], Beispiele 3 – 5. |
| <b>Anstoßen</b>         | Das Werkstück-Greifen wird durch Setzen des Steuerbits "grip workpiece" (Bit 12) angestoßen, ▶ 7.2 [☐ 92].  |

---

#### HINWEIS

Es ist zulässig, solange ein Werkstück gehalten wird, ein Werkstück-Greifen mit veränderten Bewegungsparametern anzustoßen.

---

|   |   |
|---|---|
| <b>Bewegungsparameter für BasicGrip</b> | <p>Um im BasicGrip-Modus zu greifen, müssen folgende Bewegungsparameter und Informationen an das Modul zyklisch übertragen werden:</p> <ul style="list-style-type: none"> <li>• <math>50 \leq \text{Greifkraft [\%]} \leq 100</math></li> <li>• Greifrichtung wird durch das Steuerbit "grip direction" (Bit 7) angegeben, ▶ 7.2 [☐ 91].</li> <li>• Verwendung GPE <ul style="list-style-type: none"> <li>– Modul mit GPE: Verwendung der GPE wird durch den Zustand des Steuerbits "Activate grip force and position maintenance" (Bit 31) angegeben, ▶ 7.2 [☐ 93].</li> <li>– Modul ohne GPE: Steuerbit "Activate grip force and position maintenance" muss gleich 0 sein.</li> </ul> </li> </ul> |
|---|---|

### **Bewegungsparameter für StrongGrip**

Um im StrongGrip-Modus zu greifen, müssen folgende Bewegungsparameter und Informationen an das Modul zyklisch übertragen werden:

- EGU 50, 60, 80:  $101 \leq \text{Greifkraft [\%]} \leq 200$
- EGU 70:  $101 \leq \text{Greifkraft [\%]} \leq 150$
- Greifrichtung wird durch das Steuerbit "grip direction" (Bit 7) angegeben, ▶ 7.2 [91].
- Verwendung der GPE wird durch den Zustand des Steuerbits "Activate grip force and position maintenance" angegeben, ▶ 7.2 [93].

### **Beenden**

Das Werkstück-Greifen wird durch folgende Ereignisse beendet:

- Werkstück wurde erfolgreich gegriffen, das Modul wechselt automatisch zum Werkstück-Halten.
- Werkstück wurde erfolgreich erfasst und muss nachgegriffen werden, das Modul wechselt automatisch zum Werkstück-Nachgreifen
- Automatisch bei Erreichen der minimalen oder maximalen Position
- Setzen des Steuerbits "stop"

### **Modulrückmeldung**

- Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt. Diese Bestätigung erfolgt unabhängig davon, ob der Steuerbefehl im Anschluss erfolgreich beendet oder überhaupt bearbeitet werden kann.
- Das erfolgreiche Greifen eines Werkstücks wird durch Setzen der Statusbits "workpiece gripped" und "command successfully processed" angezeigt.
- Der Wechsel ins Werkstück-Nachgreifen wird durch Setzen des Statusbits "workpiece pre-grip started" angezeigt.
- Das Erreichen der minimalen oder maximalen Position wird durch Setzen des Statusbits "no workpiece detected" angezeigt.

## Mögliche Diagnose-Ereignisse

Ereignisse, die zu Warnungen und/oder Fehlern führen, werden durch die Diagnose erkannt. Im Folgenden sind alle möglichen Diagnose-Ereignisse aufgelistet.

| Diagnose Ereignis                        | Diagnose Code *               |
|--|-------------------------------|
| Senden eines unzulässigen Steuerbefehls. | WRN_NOT_FEASIBLE + Zusatzcode |
| Bewegungsabbruch durch Benutzer.         | ERR_FAST_STOP                 |

\* Weitere Informationen siehe Kapitel ▶ 6 [D 75].

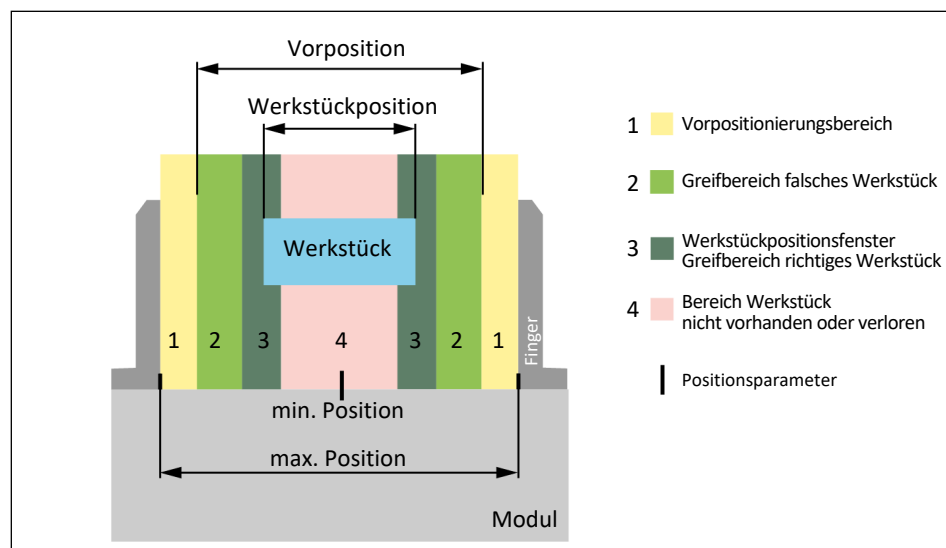
### 3.3.2 Werkstück-Greifen an erwarteter Position (kombinierte Greiffahrt)

#### Kurzbeschreibung

Beim Werkstück-Greifen an erwarteter Position wird durch eine kombinierte Greiffahrt ein Werkstück an der angegebenen Werkstückposition mit dem angegebenen Greifkraftwert gegriffen. Praktische Anwendungsbeispiele dazu sind beschrieben im Kapitel ▶ 7.1 [D 87], Beispiele 6 – 8.

#### HINWEIS

Folgende Beispieldarstellung zeigt den Greifmodus Außengreifen, die getroffenen Aussagen gelten ebenso für den Greifmodus Innengreifen.



Werkstück greifen an erwarteter Position, Beispiel Außengreifen

Die Kombination besteht aus einem optionalen Vorpositionieren (Abb.: gelber Bereich 1) und der Greifbewegung (Abb.: hell- und dunkelgrüner Bereich 2 und 3).

Die Entscheidung, ob das richtige oder falsche Werkstück gegriffen wurde, wird aufgrund der erkannten Greifposition getroffen:

- Das *richtige* Werkstück wird innerhalb des Werkstückpositionsfensters gegriffen (Abb.: dunkelgrüner Bereich 3).
- Das *falsche* Werkstück wird zwischen Vorposition und Werkstückpositionsfenster gegriffen (Abb.: hellgrüner Bereich 2).

Das Werkstückpositionsfenster ist ein "virtuelles Fenster", das um die erwartete Werkstückposition aufgespannt ist. Wird das Werkstückpositionsfenster überfahren (Abb. hellroter Bereich 4), dann wurde kein Werkstück erkannt oder das Werkstück wurde während des Nachgreifens verloren.

---

### HINWEIS

- Die Vorposition und das Werkstückpositionsfenster berechnen sich aus der zyklisch übertragenen Werkstückposition und der Parametrierung des Moduls.
- Das Vorpositionieren wird mit der maximalen Positionierungsgeschwindigkeit ausgeführt.
- Liegt die Startposition beim Anstoßen der Greiffahrt zwischen der Vorposition und der Greifposition, wird auf das Vorpositionieren verzichtet und unmittelbar die Greifbewegung ausgeführt.
- Abhängig von der Parametrierung wechselt das Modul nach einem erfolgreichen kombinierten Greifvorgang in den Zustand ▶ [Werkstück-Nachgreifen \[41\]](#) oder Werkstück-Halten.

## Anstoßen

---

Die kombinierte Greiffahrt wird durch Setzen des Steuerbits "grip workpiece at expected position" (Bit 16) angestoßen, ▶ [7.2 \[92\]](#).

---

### HINWEIS

Es ist zulässig, solange ein Werkstück gehalten wird, ein Werkstück-Greifen mit veränderten Bewegungsparametern anzustoßen.

---

- Bewegungsparameter für BasicGrip** Um im BasicGrip-Modus zu greifen, müssen folgende Bewegungsparameter und Informationen an das Modul zyklisch übertragen werden:
- $50 \leq \textit{Greifkraft} [\%] \leq 100$
  - $\langle \textit{min\_pos} \rangle \leq \textit{Werkstückposition} [\mu\text{m}] \leq \langle \textit{max\_pos} \rangle$
  - Greifrichtung wird durch das Steuerbit "grip direction" (Bit 7) angegeben, ▶ 7.2 [91].
  - Verwendung GPE
    - Modul mit GPE: Verwendung der GPE wird durch den Zustand des Steuerbits "Activate grip force and position maintenance" (Bit 31) angegeben, ▶ 7.2 [93].
    - Modul ohne GPE: Steuerbit "Activate grip force and position maintenance" muss gleich 0 sein.
- Bewegungsparameter für StrongGrip** Um im StrongGrip-Modus zu greifen, müssen folgende Bewegungsparameter und Informationen an das Modul zyklisch übertragen werden:
- EGU 50, 60, 80:  $101 \leq \textit{Greifkraft} [\%] \leq 200$
  - EGU 70:  $101 \leq \textit{Greifkraft} [\%] \leq 150$
  - $\langle \textit{min\_pos} \rangle \leq \textit{Werkstückposition} [\mu\text{m}] \leq \langle \textit{max\_pos} \rangle$
  - Greifrichtung wird durch das Steuerbit "grip direction" (Bit 7) angegeben, ▶ 7.2 [91].
  - Verwendung der GPE wird durch den Zustand des Steuerbits "Activate grip force and position maintenance" angegeben, ▶ 7.2 [93].

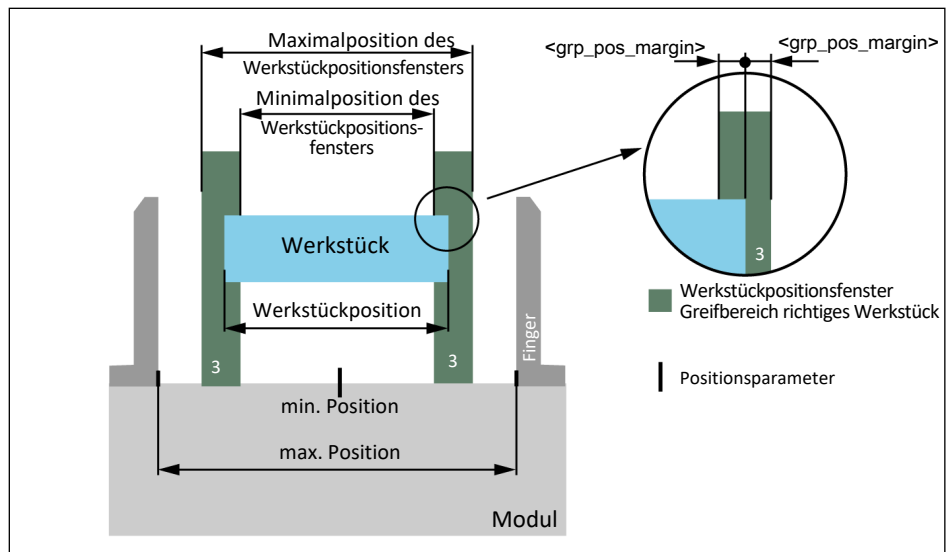
## Parametrierung

### 1. Werkstückpositionsfenster

Mit dem Parameter `<grp_pos_margin>` (► 4.2 [ 55]) kann der Wert parametrierung werden, aus dem sich die Minimal- und Maximalposition des Werkstückpositionsfensters berechnen.

#### HINWEIS

- Die Minimalposition des Werkstückpositionsfensters berechnet sich nach: *Werkstückposition* - `<grp_pos_margin>`.
- Die Maximalposition des Werkstückpositionsfensters berechnet sich nach: *Werkstückposition* + `<grp_pos_margin>`.



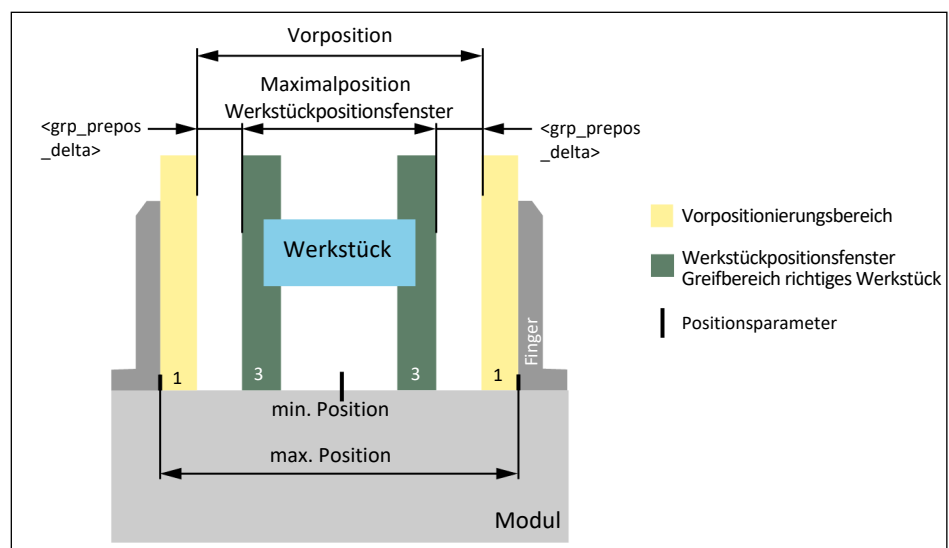
Minimal- und Maximalposition des Werkstückpositionsfensters

## 2. Vorposition

Mit dem Parameter `<grp_prepos_delta>` (► 4.2 [📄 56]) kann die betragsmäßige Positionsdifferenz zwischen Werkstückpositionsfenster und Vorposition parametrisiert werden.

### HINWEIS

- Die Vorposition wird berechnet von der Minimal- oder Maximalposition des Werkstückpositionsfensters in Abhängigkeit von der Richtung, aus der ein Werkstück gegriffen wird.
- Die Vorposition beim Innengreifen berechnet sich nach: Minimalposition Werkstückpositionsfenster - `<grp_prepos_delta>`.
- Die Vorposition beim Außengreifen berechnet sich nach: Maximalposition Werkstückpositionsfenster + `<grp_prepos_delta>`



*Vorpositionierungsbereich für das Außengreifen*

### Beenden

Das Werkstück-Greifen an erwarteter Position wird durch folgende Ereignisse beendet:

- Erwartetes Werkstück wurde gegriffen
- Nicht erwartetes Werkstück wurde gegriffen
- Automatisches Umschalten in das Nachgreifen
- Greifposition wurde überfahren
- Automatisch bei Erreichen der minimalen oder maximalen Position
- Setzen des Steuerbits "stop"

## Modulrückmeldung

- Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt. Diese Bestätigung erfolgt unabhängig davon, ob der Steuerbefehl im Anschluss erfolgreich beendet oder überhaupt bearbeitet werden kann.
- Das Greifen des erwarteten Werkstücks wird durch Setzen der Statusbits "workpiece gripped" und "command successfully processed" angezeigt.
- Das Greifen eines nicht erwarteten Werkstücks wird durch Setzen der Statusbits "wrong workpiece gripped" und "command successfully processed" angezeigt.
- Der automatisierte Wechsel ins Nachgreifen wird durch Setzen des Statusbits "workpiece pre-grip started" angezeigt.
- Das Überfahren des Werkstückpositionsfensters wird durch Setzen des Statusbits "no workpiece detected" angezeigt.

## Mögliche Diagnose-Ereignisse

Ereignisse, die zu Warnungen und/oder Fehlern führen, werden durch die Diagnose erkannt. Im Folgenden sind alle möglichen Diagnose-Ereignisse aufgelistet.

| Diagnose Ereignis                                 | Diagnose Code *               |
|---|-------------------------------|
| Senden eines unzulässigen Steuerbefehls.          | WRN_NOT_FEASIBLE + Zusatzcode |
| Bewegungsabbruch durch Benutzer.                  | ERR_FAST_STOP                 |
| Beim Vorpositionieren wird der Antrieb blockiert. | ERR_MOVE_BLOCKED              |

\* Weitere Informationen siehe Kapitel ▶ 6 [ 75 ].

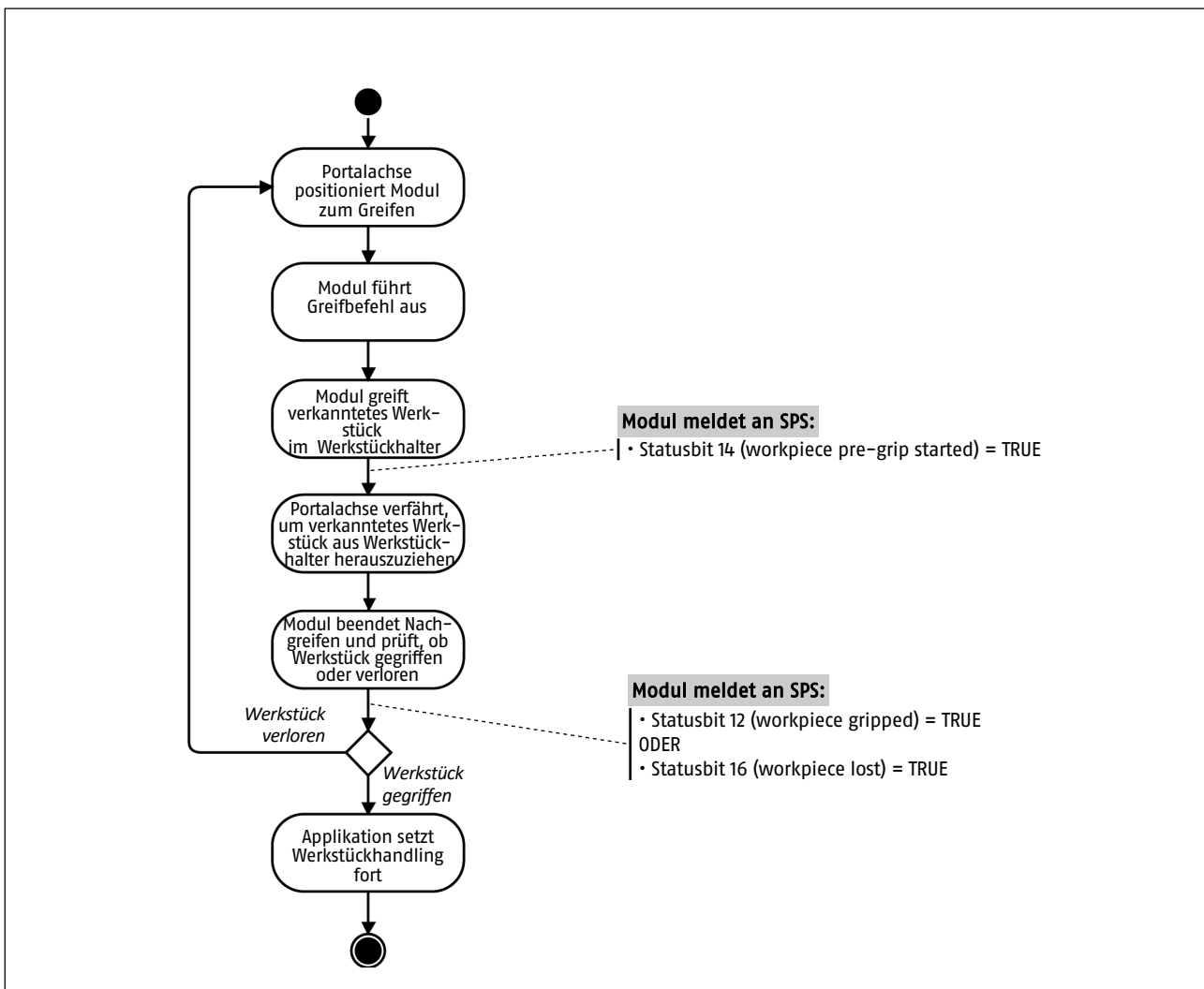
### 3.3.3 Werkstück-Nachgreifen

#### Kurzbeschreibung

Das Werkstück-Nachgreifen ist eine optionale Erweiterung für die Greifmodi (► 1.2.3 [10]) des Moduls. Das Modul kann dadurch Werkstücke greifen, die sich während der Greifbewegung verzögert ausrichten. Das Modul erkennt den ersten Kontakt mit einem Werkstück und startet daraufhin das Nachgreifen. Am Ende des Nachgreifens meldet das Modul dem Anwender zurück, ob das Werkstück gegriffen oder verloren wurde. Praktische Anwendungsbeispiele dazu sind beschrieben im Kapitel ► 7.1 [81].

#### BEISPIEL

#### Werkstück-Nachgreifen in einer Linearportal-Applikation



Werkstück-Nachgreifen in einer Linearportal-Applikation

---

## HINWEIS

Im oben dargestellten Applikationsbeispiel ist zu erkennen, dass das Starten des Nachgreifens als Auslöser für eine Aktion einer anderen Applikationskomponente dienen kann (SPS liest Statusbit 14 = TRUE). Die Rückmeldung vom Modul, ob das Werkstück gegriffen oder verloren wurde, kann wiederum als Auslöser für weitere Fallunterscheidungen in der Applikation verwendet werden.

---

### Anstoßen

Das Nachgreifverhalten wird durch den Parameter `<grp_prehold_time>` bestimmt. Ist in dem Parameter eine Nachgreifzeit eingestellt, erfolgt bei allen ausgeführten Greifbefehlen ein Nachgreifen mit der eingestellten Zeit.

### Parametrierung

Mit dem Parameter `<grp_prehold_time>` (► 4.2 [54]) kann die Zeitspanne des Nachgreifens parametrierbar werden. Die maximale Zeitspanne für das Nachgreifen beträgt 60000 ms (1 Minute).

---

## HINWEIS

Ist in diesem Parameter die Zeit 0 ms hinterlegt (Werkseinstellung), wird das Nachgreifen beim Ausführen einer Greifbewegung *nicht* verwendet.

---

### Beenden

Das Nachgreifen von Werkstücken wird durch folgende Ereignisse beendet:

- Zeitspanne des Nachgreifens ist abgelaufen
  - Setzen des Steuerbits "stop"
  - Setzen des Steuerbits "release workpiece"
  - Setzen des Steuerbits "move to absolute position"
  - Setzen des Steuerbits "move to relative position"
- 

## HINWEIS

Wird das Nachgreifen durch Setzen des Steuerbits "stop" unterbrochen, ist von einem Werkstückverlust auszugehen, da das Nachgreifen nicht erfolgreich beendet wurde. Dies wird durch Setzen des Statusbits "workpiece lost" angezeigt. Weiterhin wird das Statusbit "workpiece pre-grip started" zurückgesetzt.

---

### Modulrückmeldung

- Der Beginn des Nachgreifens wird durch Setzen des Statusbits "workpiece pre-grip started" angezeigt.

**Rückmeldungen nach vorangegangenem Werkstück-Greifen:**

- Das Greifen eines Werkstücks wird durch Setzen der Statusbits "workpiece gripped" und "command successfully processed" angezeigt.
- Ein erfolgloses Nachgreifen wird durch Setzen des Statusbits "workpiece lost" angezeigt.

**Rückmeldungen nach vorangegangenem Werkstück-Greifen an erwarteter Position:**

- Das Greifen des erwarteten Werkstücks wird durch Setzen des Statusbits "workpiece gripped" und "command successfully processed" angezeigt.
- Das Greifen eines nicht erwarteten Werkstücks wird durch Setzen des Statusbits "wrong workpiece gripped" angezeigt.
- Das Überfahren des Werkstückpositionsfensters wird durch Setzen des Statusbits "workpiece lost" angezeigt.

**Mögliche Diagnose-Ereignisse**

Ereignisse, die zu Warnungen und/oder Fehlern führen, werden durch die Diagnose erkannt. Im Folgenden sind alle möglichen Diagnose-Ereignisse aufgelistet.

| Diagnose Ereignis                        | Diagnose Code *               |
|--|-------------------------------|
| Senden eines unzulässigen Steuerbefehls. | WRN_NOT_FEASIBLE + Zusatzcode |
| Bewegungsabbruch durch Benutzer.         | ERR_FAST_STOP                 |

\* Weitere Informationen siehe Kapitel ▶ 6 [75].

**3.3.4 Werkstückverlusterkennung****Kurzbeschreibung**

Das Modul kann den Verlust des Werkstücks erkennen. Durch einen Werkstückverlust führt das Modul eine Bewegung aus. Sobald sich durch diese Bewegung eine Positionsänderung größer einer parametrierbaren Strecke ergibt, wird das Werkstück als verloren erkannt.

**HINWEIS**

Bei Modulen mit GPE: Aufgrund der fixen Nachgiebigkeit des Antriebsstrangs und der Einstellbarkeit des Parameters <wp\_lost\_dst> kann es unter Umständen dazu kommen, dass bei aktivierter GPE ein Werkstückverlust nicht erkannt wird. Bei Unsicherheit kann der letzte Greifbefehl erneut ausgeführt werden. Der Werkstückverlust wird in diesem Fall durch Setzen des Statusbits "no workpiece detected" angezeigt.

|                         |   |
|-------------------------|---|
| <b>Anstoßen</b>         | Die Werkstückverlusterkennung muss nicht angestoßen werden. Sie wird automatisch aktiviert, sobald ein Werkstück gehalten wird.   |
| <b>Parametrierung</b>   | Mit dem Parameter <wp_lost_dst> (► 4.2 [□ 55]) kann die Strecke parametrierung werden, welche die Greiferfinger nach dem Verlust des Werkstücks verfahren dürfen, bevor ein Werkstückverlust erkannt wird.  |
| <b>Modulrückmeldung</b> | <ul style="list-style-type: none"><li>• Ein Werkstückverlust wird durch Setzen des Statusbits "workpiece lost" angezeigt.</li><li>• Ein gesetztes Statusbits "workpiece pre-grip started" wird zurückgesetzt.</li><li>• Ein gesetztes Statusbits "workpiece gripped" wird zurückgesetzt.</li><li>• Ein gesetztes Statusbits "wrong workpiece gripped" wird zurückgesetzt.</li></ul> |

### 3.3.5 Werkstück-Freigeben

|                         |   |
|-------------------------|---|
| <b>Kurzbeschreibung</b> | Das Modul führt beim Werkstück-Freigeben eine relative Positionsfahrt aus. Ausgehend von der aktuellen Position wird eine definierte Strecke des Parameters <wp_release_delta> entgegengesetzt zur Greifrichtung der letzten Greifbewegung verfahren. |
|-------------------------|---|

---

#### HINWEIS

- Da beim Werkstück-Freigeben alle notwendigen Bewegungsparameter intern berechnet werden, müssen maximal zwei Steuerbits (Bit 11, optional Bit 31) geändert werden.
  - Werkstücke können ebenfalls durch Anstoßen einer absoluten oder relativen Positionsfahrt freigegeben werden.
- 

|                           |  |
|---------------------------|--|
| <b>Anstoßen</b>           | Das Freigeben von Werkstücken ist ausschließlich aus dem Werkstück-Halten heraus zulässig und wird durch Setzen des Steuerbits "release workpiece" (Bit 11) angestoßen, ► 7.2 [□ 92].  |
| <b>Bewegungsparameter</b> | Folgende Bewegungsparameter müssen an das Modul zyklisch übertragen werden: <ul style="list-style-type: none"><li>• Verwendung der GPE wird durch den Zustand des Steuerbits "Activate grip force and position maintenance" angegeben, ► 7.2 [□ 93].</li></ul> |
| <b>Parametrierung</b>     | Mit dem Parameter <wp_release_delta> (► 4.2 [□ 55]) kann die Strecke parametrierung werden, die das Modul beim Freigeben relativ verfährt.   |

**Beenden**

Das Werkstück-Freigeben wird durch folgende Ereignisse beendet:

- Berechnete Freigabeposition wurde erreicht
- Setzen des Steuerbits "stop"

**Modulrückmeldung**

- Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt. Diese Bestätigung erfolgt unabhängig davon, ob der Steuerbefehl im Anschluss erfolgreich beendet oder überhaupt bearbeitet werden kann.
- Das Freigeben von Werkstücken wird durch Setzen der Statusbits "position reached" und "command successfully processed" angezeigt.
- Ein gesetztes Statusbits "workpiece pre-grip started" wird zurückgesetzt.
- Ein gesetztes Statusbits "workpiece gripped" wird zurückgesetzt.
- Ein gesetztes Statusbits "wrong workpiece gripped" wird zurückgesetzt.

**Mögliche Diagnose-Ereignisse**

Ereignisse, die zu Warnungen und/oder Fehlern führen, werden durch die Diagnose erkannt. Im Folgenden sind alle möglichen Diagnose-Ereignisse aufgelistet.

| Diagnose Ereignis                                  | Diagnose Code *               |
|--|-------------------------------|
| Freigeben dauert zu lange.                         | ERR_MOV_ABORT_TO              |
| Antrieb ist bei Bewegungsbeginn bereits blockiert. | ERR_MOVE_BLOCKED              |
| Antrieb blockiert während Bewegung.                | ERR_MOVE_BLOCKED              |
| Senden eines unzulässigen Steuerbefehls.           | WRN_NOT_FEASIBLE + Zusatzcode |
| Bewegungsabbruch durch Benutzer.                   | ERR_FAST_STOP                 |

\* Weitere Informationen siehe Kapitel ▶ 6 [ 75 ].

### 3.3.6 Werkstück manuell entnehmen

**Kurzbeschreibung** Befindet sich das Modul im **Fehlerzustand**, kann die GPE des Moduls deaktiviert werden. Der Anwender kann ein gegriffenes Werkstück manuell entnehmen.

---

#### HINWEIS

Da der Benutzer unmittelbar am Modul arbeitet, ist das manuelle **Entnehmen** von Werkstücken **nur im Notfall zulässig**. Um sicherzustellen, dass das Modul keine unerwarteten Bewegungen durchführt, ist das Anstoßen dieser Funktion ausschließlich im Fehlerzustand des Moduls möglich!

---

**Anstoßen** Das manuelle Entnehmen von Werkstücken wird durch Setzen des Steuerbits "release for manual movement" (Bit 5) angestoßen, ► 7.2 [91].

Sollte sich das Modul **nicht** im Fehlerzustand befinden, muss wie folgt vorgegangen werden:

- Zurücksetzen des Steuerbits "fast stop" (Bit 0)
- Setzen des Steuerbits "fast stop"(Bit 0)
- Setzen des Steuerbits "release for manual movement" (Bit 5)

**Bewegungsparameter** Um das manuelle Freigeben von Werkstücken durchzuführen, müssen keine Bewegungsparameter übertragen werden.

**Beenden** Das manuelle Entnehmen von Werkstücken wird durch folgende Ereignisse beendet:

- Zurücksetzen des Steuerbits "fast stop" auf 0
- Automatisch nach 30 Minuten

**Modulrückmeldung**

- Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt. Diese Bestätigung erfolgt unabhängig davon, ob der Steuerbefehl im Anschluss erfolgreich beendet oder überhaupt bearbeitet werden kann.
- Die Freigabe zur manuellen Werkstückentnahme wird durch Setzen des Statusbits "released for manual movement" angezeigt.

## 3.4 Weitere Funktionen

### 3.4.1 Nullpunktverschiebung

Bei Verwendung von applikationsspezifischen Greiferfingern kann der Nullpunkt "verschoben" werden, sodass die angezeigten Positionswerte zur Geometrie der Greiferfinger passen. Mit Verschieben des Nullpunkts werden automatisch die Werte der Parameter <actual\_pos>, <min\_pos> und <max\_pos> geändert.

**HINWEIS**

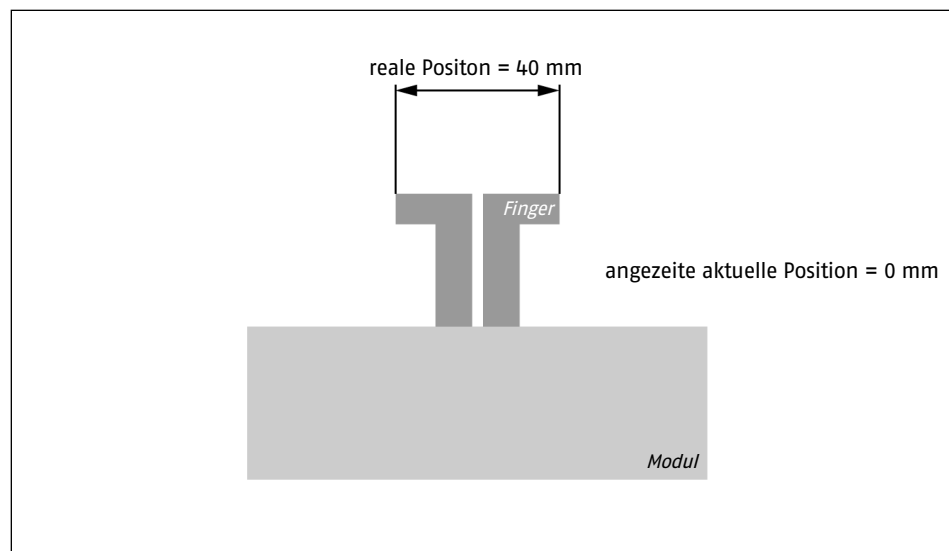
Im Auslieferungszustand entspricht der Nullpunkt der minimalen Position der Grundbacken, siehe ▶ 1.2.1 [8].

**Parametrierung**

Mit dem Parameter <zero\_pos\_ofs> (▶ 4.2 [58]) kann vorzeichenbehaftet die Strecke parametrieren, um die der Nullpunkt verschoben wird.

**Beispiel: Nullpunktverschiebung EGU 50**

- Anwendung:
  - Werte der aktuellen, minimalen und maximalen Position **vor** der Verschiebung.
    - > actual\_pos = 0 mm
    - > min\_pos = 0 mm
    - > max\_pos = 102 mm
- Das Modul soll mit den dargestellten Greiferfingern als Innengreifer verwendet werden.



*Nullpunktverschiebung*

- Verschiebung des Nullpunkts:
  - In den Parameter <zero\_pos\_ofs> Wert +40 mm schreiben.
  - Werte der aktuellen, minimalen und maximalen Position **nach** der Verschiebung:
    - > actual\_pos = 40 mm
    - > min\_pos = 40 mm
    - > max\_pos = 142 mm

### 3.4.2 Handshake

|                         |   |
|-------------------------|---|
| <b>Kurzbeschreibung</b> | Erkennt das Modul den Eingang eines Steuerbefehls, meldet es den Eingang an die Steuerung zurück.                       |
| <b>Modulrückmeldung</b> | Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt. |

### 3.4.3 Bremsentest

|                         |  |
|-------------------------|--|
| <b>Kurzbeschreibung</b> | Das Modul überprüft beim Bremsentest die Haltekraft der Bremse, indem ein definiertes Moment abwechselnd in beide Richtungen gegen die eingefallene Bremse wirkt.  |
| <b>Anstoßen</b>         | <ul style="list-style-type: none"><li>• Der Bremsentest wird durch Setzen des Steuerbits "brake test" (Bit 30) angestoßen, ▶ 7.2 [📄 93].</li><li>• Das Modul muss sich im Stillstand befinden und kein Werkstück halten.</li></ul>   |
| <b>Modulrückmeldung</b> | <ul style="list-style-type: none"><li>• Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt. Diese Bestätigung erfolgt unabhängig davon, ob der Steuerbefehl im Anschluss erfolgreich beendet oder überhaupt bearbeitet werden kann.</li><li>• Ein erfolgreich durchgeführter Bremsentest wird durch Setzen des Statusbits "command successfully processed" angezeigt.</li><li>• Ein erfolglos durchgeführter Bremsentest wird durch Setzen des Statusbits "error" in Verbindung mit dem Diagnose Code ERR_BT_FAILED, ▶ 6.2 [📄 78].</li></ul> |

### 3.4.4 LifeSign

|                         |  |
|-------------------------|--|
| <b>Kurzbeschreibung</b> | Das Quittieren (acknowledge) kann dazu verwendet werden, um die Kommunikation zwischen Steuerung und Modul zu überprüfen. Sobald das Kommando angestoßen wird, meldet das Modul den Eingang des Steuerbefehls an die Steuerung zurück. Erfolgt keine Rückmeldung, ist von einer Störung in der Kommunikation auszugehen. |
| <b>Anstoßen</b>         | Das Quittieren wird durch Setzen des Steuerbits "acknowledge" (Bit 2) angestoßen, ▶ 7.2 [📄 90].  |
| <b>Modulrückmeldung</b> | Der Eingang des Steuerbefehls wird durch sofortigen Zustandswechsel des Statusbits "command received toggle" angezeigt.  |

### 3.4.5 Steuerbefehl zeitoptimiert wiederholen

|                         |   |
|-------------------------|---|
| <b>Kurzbeschreibung</b> | Diese Funktion ermöglicht es, aufeinanderfolgende gleiche Steuerbefehle zeitoptimiert an das Modul zu senden. |
|-------------------------|---|

---

#### HINWEIS

Standardmäßig werden Steuerbefehle an das Modul durch 0 -> 1 Flanken einzelner Steuerbits gesendet. Soll die gleiche Funktion erneut ausgeführt werden, muss dazu das entsprechende Steuerbit erst zurückgesetzt und anschließend wieder gesetzt werden.

---

|                 |   |
|-----------------|---|
| <b>Anstoßen</b> | Das zeitoptimierte Senden gleicher Steuerbefehle wird bei gesetztem Steuerbit durch einen Toggle des Steuerbits "repeat command toggle" angestoßen. |
|-----------------|---|

### 3.4.6 Werkseinstellung

**Kurzbeschreibung** Das Modul kann, aus dem Fehlerzustand heraus, softwareseitig auf die Werkseinstellung zurückgesetzt werden. Damit wird die Standardparametrierung wie bei Auslieferung des Moduls wiederhergestellt. Zusätzlich wird der Diagnosespeicher gelöscht.

---

**HINWEIS**

Der Parameter <enable\_softreset> wird beim Herstellen der Werkseinstellung nicht zurückgesetzt.

---

**Anstoßen** Das Zurücksetzen auf die Werkseinstellungen wird angestoßen, indem der Wert HEX 0x8C (= 140 dezimal) in den Parameter HEX 0x0100 (= 256 dezimal) geschrieben wird. Der Datentyp des Parameters HEX 0x100 ist UINT16.

**Modulrückmeldung** Das erfolgreiche Zurückstellen auf Werkseinstellungen wird durch Setzen des Statusbits "ready for shutdown" angezeigt.

---

**HINWEIS**

Das Setzen dieses Bits muss zwingend abgewartet werden, bevor das Modul ausgeschaltet oder neu gestartet wird.

---

**Mögliche Diagnose-Ereignisse** Ereignisse, die zu Warnungen und/oder Fehlern führen, werden durch die Diagnose erkannt. Im Folgenden sind alle möglichen Diagnose-Ereignisse aufgelistet.

| Diagnose Ereignis   | Diagnose Code *               |
|---|-------------------------------|
| Das Zurücksetzen auf Werkseinstellungen ist nicht möglich | WRN_NOT_FEASIBLE + Zusatzcode |

---

**ACHTUNG**

**Sachschaden durch fehlerhafte Anwendung!**

Nach dem Zurücksetzen auf Werkseinstellungen müssen applikationsspezifische Parameter wieder angepasst werden. Nichtbeachtung kann zu Schäden am Modul selbst oder an benachbarten Maschinenteilen führen.

---

## 4 Systemparameter

### 4.1 Wertebereiche

#### Wertebereiche

Folgende interne Datentypen werden verwendet:

| Datentyp | Grenzwert  | Zahlenwert    |
|----------|------------|---------------|
| BOOL     | MIN_BOOL   | 0             |
|          | MAX_BOOL   | 1             |
| UINT8    | MIN_UINT8  | 0             |
|          | MAX_UINT8  | 255           |
| UINT16   | MIN_UINT16 | 0             |
|          | MAX_UINT16 | 65535         |
| UINT32   | MIN_UINT32 | 0             |
|          | MAX_UINT32 | 4294968295    |
| INT32    | MIN_INT32  | -2147483648   |
|          | MAX_INT32  | 2147483647    |
| FLOAT    | MIN_FLOAT  | -3.402823E+38 |
|          | MAX_FLOAT  | 3.402823E+38  |
| CHAR     | MIN_CHAR   | 0             |
|          | MAX_CHAR   | 255           |
| ENUM     | MIN_ENUM   | 0             |
|          | MAX_ENUM   | 255           |

### 4.2 Parameterliste

Im Folgenden sind alle systemrelevanten Parameter nach dem Schema "HEX-Code/DEC-Code <Parametername>" aufgelistet.

#### HINWEIS

Die Parameterliste bezieht sich auf Parameter, die azyklisch ausgelesen bzw. geschrieben werden können.

Einige der hier als nur "lesbar" aufgelisteten Parameter können prinzipiell geändert werden, jedoch hat der Benutzer nicht das Recht diese Parameter zu ändern.

Alle Parameter, die in dieser Liste nicht auftauchen, sind interne oder reservierte Parameter.

#### Parameterkonfiguration

Alle Systemparameter, bei denen der Benutzer Schreibrechte besitzt, lassen sich über den azyklischen Datenaustausch parametrieren.

**HEX 0x0040**  
**DEC 64**

**<plc\_sync\_input>**

Kurzbeschreibung: Dieser Parameter bildet die zyklischen Eingangsdaten ab (PLC Frame Eingangsdaten).  
Parametername: PLC Frame Eingangsdaten  
Zugriffsrecht: Lesen  
Datentyp: 1x UINT32, 1x INT32, 2x UINT32 (zusammen 16 Byte)  
Anzahl Register: 8

**HEX 0x0048**  
**DEC 72**

**<plc\_sync\_output>**

Kurzbeschreibung: Dieser Parameter bildet die zyklischen Ausgangsdaten ab (PLC Frame Ausgangsdaten).  
Parametername: PLC Frame Ausgangsdaten  
Zugriffsrecht: Lesen und Schreiben  
Datentyp: 1x UINT32, 2x INT32, 1x UINT32 (zusammen 16 Byte)  
Anzahl Register: 8

**HEX 0x0118**  
**DEC 280**

**<err\_code>**

Kurzbeschreibung: Mit diesem Parameter kann der anliegende Fehlercode ausgelesen werden.  
Parametername: Fehlercode  
Zugriffsrecht: Lesen  
Datentyp: ENUM  
Enumeration: siehe Kapitel ▶ 6.2 [[📄 77](#)]  
Anzahl Register: 1

**HEX 0x0120**  
**DEC 288**

**<wrn\_code>**

Kurzbeschreibung: Mit diesem Parameter kann der anliegende Warnungscode ausgelesen werden.  
Parametername: Warnungscode  
Zugriffsrecht: Lesen  
Datentyp: ENUM  
Enumeration: siehe Kapitel ▶ 6.1 [[📄 75](#)]  
Anzahl Register: 1

HEX 0x0128  
DEC 296

### <sys\_msg\_req>

Kurzbeschreibung: Mit diesem Parameter kann durch Schreiben eines Index ein Eintrag im Diagnosespeicher zum Auslesen über <sys\_msg\_buffer> ausgewählt werden

Parametername: Diagnosespeichereintrag anfordern

Zugriffsrecht: Lesen und Schreiben

Datentyp: UINT16

Anzahl Register: 1

---

### HINWEIS

Im Diagnosespeicher werden die 32 (Index 0 - 31) zuletzt aufgetretenen Diagnoseereignisse abgespeichert.

---

HEX 0x0130  
DEC 304

### <sys\_msg\_buffer>

Kurzbeschreibung: Über diesem Parameter kann der angeforderte Diagnosespeichereintrag ausgelesen werden.

Parametername: Diagnosespeichereintrag

Zugriffsrecht: Lesen

Datentyp: CHAR[214]

Format: ASCII-String

Anzahl Register: 107

HEX 0x0230  
DEC 560

### <actual\_pos>

Kurzbeschreibung: Mit diesem Parameter kann die aktuelle Ist-Position ausgelesen werden.

Parametername: Ist-Position

Zugriffsrecht: Lesen

Datentyp: FLOAT

Einheit: Millimeter [mm]

Anzahl Register: 2

HEX 0x0238  
DEC 568

### <actual\_vel>

Kurzbeschreibung: Mit diesem Parameter kann die aktuelle Ist-Geschwindigkeit ausgelesen werden.

Parametername: Ist-Geschwindigkeit

Zugriffsrecht: Lesen

Datentyp: FLOAT

Einheit: Millimeter pro Sekunde [mm/s]

Anzahl Register: 2

**HEX 0x0380**  
**DEC 896**

**<grp\_prehold\_time>**

Kurzbeschreibung: Mit diesem Parameter kann die Zeitspanne für das Nachgreifen ausgelesen und geschrieben werden.

Parametername: Nachgreifzeit  
Zugriffsrecht: Lesen und Schreiben  
Datentyp: UINT16  
Einheit: Millisekunde [ms]  
Min; Max; Default 0; 5000; 0  
Anzahl Register: 1

**HEX 0x03A8**  
**DEC 936**

**<dead\_load\_kg>**

Kurzbeschreibung: Mit diesem Parameter kann die Eigenmasse des Moduls ausgelesen und geschrieben werden.

Parametername: Eigenmasse des Greifers  
Zugriffsrecht: Lesen  
Datentyp: FLOAT  
Einheit: Kilogramm [kg]  
Anzahl Register: 2

**HEX 0x03B0**  
**DEC 944**

**<tool\_cent\_point>**

Kurzbeschreibung: Mit diesem Parameter kann der Werkzeugmittelpunkt (TCP = Tool Center Point) des Moduls ausgelesen und geschrieben werden.

Parametername: Tool center point 6D-frame  
Zugriffsrecht: Lesen  
Datentyp: 6x FLOAT (24 Byte)  
Einheit: x [mm], y [mm], z [mm], a [°], b [°], c [°]  
Anzahl Register: 12

**HEX 0x03B8**  
**DEC 952**

**<cent\_of\_mass>**

Kurzbeschreibung: Mit diesem Parameter können der Masseschwerpunkt und die Massenträgheitsmomente des Moduls ausgelesen und geschrieben werden.

Parametername: Masseschwerpunkt – 6D-frame  
Zugriffsrecht: Lesen  
Datentyp: 6x FLOAT (24 Byte)  
Einheit: x [mm], y [mm], z [mm], a [kg\*m<sup>2</sup>], b [kg\*m<sup>2</sup>], c [kg\*m<sup>2</sup>]  
Anzahl Register: 12

**HEX 0x0500**  
**DEC 1280**

**<module\_type>**

Kurzbeschreibung: Mit diesem Parameter kann der Modultyp ausgelesen werden.

Parametername: Modultyp

Zugriffsrecht: Lesen

Datentyp: ENUM

Enumeration: Ausgelesen wird der zum Modul passende Enumerationswert.

Anzahl Register: 1

**HEX 0x0528**  
**DEC 1320**

**<wp\_lost\_dst>**

Kurzbeschreibung: Mit diesem Parameter kann eingestellt werden, ab welchem Verfahrensweg ein Werkstückverlust erkannt wird.

Parametername: Max. Strecke nach Werkstückverlust

Zugriffsrecht: Lesen und Schreiben

Datentyp: FLOAT

Einheit: Millimeter [mm]

Min; Max; Default 0.1; 50; 1

Anzahl Register: 2

**HEX 0x0540**  
**DEC 1344**

**<wp\_release\_delta>**

Kurzbeschreibung: Mit diesem Parameter kann das relative Positionsdelta zwischen Greifposition und Freigabe-position ausgelesen und geschrieben werden.

Parametername: Relativer Positionswert beim Werkstück-Freigeben

Zugriffsrecht: Lesen und Schreiben

Datentyp: FLOAT

Einheit: Millimeter [mm]

Min; Max; Default 1; 50; 2

Anzahl Register: 2

**HEX 0x0580**  
**DEC 1408**

**<grp\_pos\_margin>**

Kurzbeschreibung: Mit diesem Parameter kann der Toleranzwert des Werkstückpositionsfensters gelesen und geschrieben werden.

Parametername: Positionstoleranz der Werkstückerkennung

Zugriffsrecht: Lesen und Schreiben

Datentyp: FLOAT

Einheit: Millimeter [mm]

Min; Max; Default 1; 10; 2

Anzahl Register: 2

**HEX 0x0588**  
**DEC 1416**

**<max\_phys\_stroke>**

Kurzbeschreibung: Mit diesem Parameter kann der maximale physikalische Hub des Moduls ohne Finger gelesen werden.

Parametername: Max. physikalischer Hub

Zugriffsrecht: Lesen

Datentyp: FLOAT

Einheit: Millimeter [mm]

Anzahl Register: 2

---

**HINWEIS**

In Abhängigkeit von der Baugröße sind die maximalen Hübe wie folgt:

- EGU50 = 102 mm / EGU50-SD = 82 mm
  - EGU60 = 120 mm / EGU60-SD = 100 mm
  - EGU70 = 140 mm / EGU70-SD = 120 mm
  - EGU80 = 160 mm / EGU80-SD = 140 mm
- 

**HEX 0x05A8**  
**DEC 1448**

**<grp\_prepos\_delta>**

Kurzbeschreibung: Mit diesem Parameter kann das relative Positionsdelta zwischen Vorposition und Greifposition ausgelesen und geschrieben werden.

Parametername: Relativer Vorpositionswert beim Greifen

Zugriffsrecht: Lesen und Schreiben

Datentyp: FLOAT

Einheit: Millimeter [mm]

Min; Max; Default 1; 50; 5

Anzahl Register: 2

HEX 0x0600  
DEC 1536

### <min\_pos>

Kurzbeschreibung: Mit diesem Parameter kann der kleinste Positionswert ausgelesen und geschrieben werden, der durch das Modul angefahren werden kann.

Parametername: Min. absoluter Positionswert

Zugriffsrecht: Lesen und Schreiben

Datentyp: FLOAT

Einheit: Millimeter [mm]

Anzahl Register: 2

---

### HINWEIS

- In diesen Parameter können Werte innerhalb der folgenden Grenzen geschrieben werden:  
 $\langle \text{zero\_pos\_ofs} \rangle \leq \text{Wert} < \langle \text{max\_phys\_stroke} \rangle + \langle \text{zero\_pos\_ofs} \rangle$
  - Weiterhin muss der Wert kleiner als der Wert des Parameters  $\langle \text{max\_pos} \rangle$  sein.
- 

HEX 0x0608  
DEC 1544

### <max\_pos>

Kurzbeschreibung: Mit diesem Parameter kann der größte Positionswert ausgelesen und geschrieben werden, der durch das Modul angefahren werden kann.

Parametername: Max. absoluter Positionswert

Zugriffsrecht: Lesen und Schreiben

Datentyp: FLOAT

Einheit: Millimeter [mm]

Anzahl Register: 2

---

### HINWEIS

- In diesen Parameter können Werte innerhalb der folgenden Grenzen geschrieben werden:  
 $\langle \text{zero\_pos\_ofs} \rangle < \text{Wert} \leq \langle \text{max\_phys\_stroke} \rangle + \langle \text{zero\_pos\_ofs} \rangle$
  - Weiterhin muss der Wert größer als der Wert des Parameters  $\langle \text{min\_pos} \rangle$  sein.
-

**HEX 0x0610**  
**DEC 1552**

**<zero\_pos\_ofs>**

Kurzbeschreibung: Mit diesem Parameter kann der Nullpunkt an die Applikation angepasst werden.

Parametername: Nullpunkt Offset

Zugriffsrecht: Lesen und Schreiben

Datentyp: FLOAT

Einheit: Millimeter [mm]

Min; Max; Default -10000; 10000; 0

Anzahl Register: 2

**HEX 0x0628**  
**DEC 1576**

**<min\_vel>**

Kurzbeschreibung: Mit diesem Parameter kann die minimale Bewegungs-/Greifgeschwindigkeit ausgelesen werden, mit der das Modul verfahren werden kann.

Parametername: Min. Geschwindigkeit

Zugriffsrecht: Lesen

Datentyp: FLOAT

Einheit: Millimeter pro Sekunde [mm/s]

Anzahl Register: 2

---

**HINWEIS**

Unabhängig von der Baugröße beträgt die minimale Bewegungsgeschwindigkeiten 10 mm/s.

---

**HEX 0x0630**  
**DEC 1584**

**<max\_vel>**

Kurzbeschreibung: Mit diesem Parameter kann die maximale Positionierungsgeschwindigkeit ausgelesen werden, mit der das Modul verfahren werden kann.

Parametername: Max. Geschwindigkeit

Zugriffsrecht: Lesen

Datentyp: FLOAT

Einheit: Millimeter pro Sekunde [mm/s]

Anzahl Register: 2

---

**HINWEIS**

In Abhängigkeit von der Baugröße sind die maximalen Bewegungsgeschwindigkeiten wie folgt:

- EGU50 = 110 mm/s
  - EGU60 = 110 mm/s
  - EGU70 = 70 mm/s
  - EGU80 = 70 mm/s
-

HEX 0x0650  
DEC 1616

#### <max\_grp\_vel>

Kurzbeschreibung: Mit diesem Parameter kann die maximale Greifgeschwindigkeit ausgelesen werden, mit der das Modul verfahren werden kann.

Parametername: Max. Greifgeschwindigkeit  
Zugriffsrecht: Lesen  
Datentyp: FLOAT  
Einheit: Millimeter pro Sekunde [mm/s]  
Anzahl Register: 2

---

#### HINWEIS

- Unabhängig von der Baugröße beträgt die maximale Greifgeschwindigkeit 25 mm/s.
  - Weitere Informationen zur Greifgeschwindigkeit siehe Kapitel ► [Greifmodi \[10\]](#).
- 

HEX 0x0658  
DEC 1624

#### <min\_grp\_force>

Kurzbeschreibung: Mit diesem Parameter kann die minimale Greifkraft ausgelesen werden.

Parametername: Min. Greifkraft  
Zugriffsrecht: Lesen  
Datentyp: FLOAT  
Einheit: Newton [N]  
Anzahl Register: 2

HEX 0x0660  
DEC 1632

#### <max\_grp\_force>

Kurzbeschreibung: Mit diesem Parameter kann die maximale Greifkraft ausgelesen werden.

Parametername: Max. Greifkraft  
Zugriffsrecht: Lesen  
Datentyp: FLOAT  
Einheit: Newton [N]  
Anzahl Register: 2

HEX 0x06A8  
DEC 1704

#### <max\_allow\_force>

Kurzbeschreibung: Mit diesem Parameter kann die maximale Greifkraft für den StrongGrip-Modus ausgelesen werden.

Parametername: Max. zulässige Greifkraft StrongGrip  
Zugriffsrecht: Lesen  
Datentyp: FLOAT  
Einheit: Newton [N]  
Anzahl Register: 2

**HEX 0x0800**  
**DEC 2048**

**<min\_err\_mot\_volt>**

Kurzbeschreibung: Mit diesem Parameter kann der untere Fehlergrenzwert der Versorgungsspannung des Motors ausgelesen werden.

Parametername: Unterer Fehlergrenzwert Versorgungsspannung Leistungsteil

Zugriffsrecht: Lesen

Datentyp: FLOAT

Einheit: Volt [V]

Anzahl Register: 2

**HEX 0x0808**  
**DEC 2056**

**<max\_err\_mot\_volt>**

Kurzbeschreibung: Mit diesem Parameter kann der obere Fehlergrenzwert der Versorgungsspannung des Motors ausgelesen werden.

Parametername: Oberer Fehlergrenzwert Versorgungsspannung Leistungsteil

Zugriffsrecht: Lesen

Datentyp: FLOAT

Einheit: Volt [V]

Anzahl Register: 2

**HEX 0x0810**  
**DEC 2064**

**<min\_err\_lgc\_volt>**

Kurzbeschreibung: Mit diesem Parameter kann der untere Fehlergrenzwert der Versorgungsspannung des Logikteils ausgelesen werden.

Parametername: Unterer Fehlergrenzwert Versorgungsspannung Logikteil

Zugriffsrecht: Lesen

Datentyp: FLOAT

Einheit: Volt [V]

Anzahl Register: 2

**HEX 0x0818**  
**DEC 2072**

**<max\_err\_lgc\_volt>**

Kurzbeschreibung: Mit diesem Parameter kann der obere Fehlergrenzwert der Versorgungsspannung des Logikteils werden.

Parametername: Oberer Fehlergrenzwert Versorgungsspannung Logikteil

Zugriffsrecht: Lesen

Datentyp: FLOAT

Einheit: Volt [V]

Anzahl Register: 2

**HEX 0x0820**  
**DEC 2080**

**<min\_err\_lgc\_temp>**

**Kurzbeschreibung:** Mit diesem Parameter kann der untere Fehlergrenzwert der Temperatur des Logikteils ausgelesen werden.

**Parametername:** Unterer Fehlergrenzwert Versorgungsspannung Logikteil

**Zugriffsrecht:** Lesen

**Datentyp:** FLOAT

**Einheit:** Grad Celsius [°C]

**Anzahl Register:** 2

**HEX 0x0828**  
**DEC 2088**

**<max\_err\_lgc\_temp>**

**Kurzbeschreibung:** Durch diesen Parameter kann der obere Fehlergrenzwert der Temperatur des Logikteils ausgelesen werden.

**Parametername:** Oberer Fehlergrenzwert Temperatur Logikteil

**Zugriffsrecht:** Lesen

**Datentyp:** FLOAT

**Einheit:** Grad Celsius [°C]

**Anzahl Register:** 2

**HEX 0x0840**  
**DEC 2112**

**<meas\_lgc\_temp>**

**Kurzbeschreibung:** Mit diesem Parameter kann die aktuell gemessene Temperatur des Logikteils ausgelesen werden.

**Parametername:** Gemessene Temperatur Logikteil

**Zugriffsrecht:** Lesen

**Datentyp:** FLOAT

**Einheit:** Grad Celsius [°C]

**Anzahl Register:** 2

**HEX 0x0870**  
**DEC 2160**

**<meas\_lgc\_volt>**

**Kurzbeschreibung:** Mit diesem Parameter kann die aktuell gemessene Versorgungsspannung des Logikteils ausgelesen werden.

**Parametername:** Gemessene Versorgungsspannung Logikteil

**Zugriffsrecht:** Lesen

**Datentyp:** FLOAT

**Einheit:** Volt [V]

**Anzahl Register:** 2

**HEX 0x0878**  
**DEC 2168**

**<meas\_mot\_volt>**

Kurzbeschreibung: Mit diesem Parameter kann die aktuell gemessene Versorgungsspannung des Motors ausgelesen werden.

Parametername: Gemessene Versorgungsspannung Leistungsteil

Zugriffsrecht: Lesen

Datentyp: FLOAT

Einheit: Volt [V]

Anzahl Register: 2

**HEX 0x0880**  
**DEC 2176**

**<min\_wrn\_mot\_volt>**

Kurzbeschreibung: Mit diesem Parameter kann der untere Warnungsgrenzwert der Versorgungsspannung des Motors ausgelesen und geschrieben werden.

Parametername: Unterer Warnungsgrenzwert  
Versorgungsspannung Leistungsteil

Zugriffsrecht: Lesen und Schreiben

Datentyp: FLOAT

Einheit: Volt [V]

Anzahl Register: 2

**HEX 0x0888**  
**DEC 2184**

**<max\_wrn\_mot\_volt>**

Kurzbeschreibung: Mit diesem Parameter kann der obere Warnungsgrenzwert der Versorgungsspannung des Motors ausgelesen und geschrieben werden.

Parametername: Oberer Warnungsgrenzwert  
Versorgungsspannung Leistungsteil

Zugriffsrecht: Lesen und Schreiben

Datentyp: FLOAT

Einheit: Volt [V]

Anzahl Register: 2

**HEX 0x0890**  
**DEC 2192**

**<min\_wrn\_lgc\_volt>**

Kurzbeschreibung: Mit diesem Parameter kann der untere Warnungsgrenzwert der Versorgungsspannung des Logikteils ausgelesen und geschrieben werden.

Parametername: Unterer Warnungsgrenzwert  
Versorgungsspannung Logikteil

Zugriffsrecht: Lesen und Schreiben

Datentyp: FLOAT

Einheit: Volt [V]

Anzahl Register: 2

**HEX 0x0898**  
**DEC 2200**

**<max\_wrn\_lgc\_volt>**

Kurzbeschreibung: Mit diesem Parameter kann der obere Warnungsgrenzwert der Versorgungsspannung des Logikteils ausgelesen und geschrieben werden.

Parametername: Oberer Warnungsgrenzwert Versorgungsspannung Logikteil

Zugriffsrecht: Lesen und Schreiben

Datentyp: FLOAT

Einheit: Volt [V]

Anzahl Register: 2

**HEX 0x08A0**  
**DEC 2208**

**<min\_wrn\_lgc\_temp>**

Kurzbeschreibung: Mit diesem Parameter kann der untere Warnungsgrenzwert der Temperatur des Logikteils ausgelesen werden.

Parametername: Unterer Warnungsgrenzwert Temperatur Logikteil

Zugriffsrecht: Lesen

Datentyp: FLOAT

Einheit: Grad Celsius [°C]

Anzahl Register: 2

**HEX 0x08A8**  
**DEC 2216**

**<max\_wrn\_lgc\_temp>**

Kurzbeschreibung: Mit diesem Parameter kann der obere Warnungsgrenzwert der Temperatur des Logikteils ausgelesen werden.

Parametername: Oberer Warnungsgrenzwert Temperatur Logikteil

Zugriffsrecht: Lesen

Datentyp: FLOAT

Einheit: Grad Celsius [°C]

Anzahl Register: 2

**HEX 0x1000**  
**DEC 4096**

**<serial\_no\_txt>**

Kurzbeschreibung: Mit diesem Parameter kann die Seriennummer des Moduls ausgelesen werden.

Parametername: Geräteseriennummer

Zugriffsrecht: Lesen

Datentyp: CHAR[16]

Format: ASCII-String

Anzahl Register: 8

|                                      |   |
|--------------------------------------|---|
| <b>HEX 0x1008</b><br><b>DEC 4104</b> | <p><b>&lt;order_no_txt&gt;</b></p> <p>Kurzbeschreibung: Mit diesem Parameter kann die Bestellnummer des Moduls ausgelesen werden.</p> <p>Parametername: Bestellnummer</p> <p>Zugriffsrecht: Lesen</p> <p>Datentyp: CHAR[16]</p> <p>Format: ASCII-String</p> <p>Anzahl Register: 8</p>                             |
| <b>HEX 0x1020</b><br><b>DEC 4128</b> | <p><b>&lt;serial_no_num&gt;</b></p> <p>Kurzbeschreibung: Mit diesem Parameter kann die Seriennummer des Moduls numerisch ausgelesen werden.</p> <p>Parametername: Geräteseriennummer kodiert</p> <p>Zugriffsrecht: Lesen</p> <p>Datentyp: UINT32</p> <p>Anzahl Register: 2</p>                                    |
| <b>HEX 0x1100</b><br><b>DEC 4352</b> | <p><b>&lt;fw_app_date&gt;</b></p> <p>Kurzbeschreibung: Mit diesem Parameter kann das Erstelldatum der Firmware-Version ausgelesen werden.</p> <p>Parametername: Applikation-Firmware Erstelldatum</p> <p>Zugriffsrecht: Lesen</p> <p>Datentyp: CHAR[12]</p> <p>Format: ASCII-String</p> <p>Anzahl Register: 6</p> |
| <b>HEX 0x1108</b><br><b>DEC 4360</b> | <p><b>&lt;fw_app_time&gt;</b></p> <p>Kurzbeschreibung: Mit diesem Parameter kann die Erstellzeit der Firmware-Version ausgelesen werden.</p> <p>Parametername: Applikation-Firmware Erstellzeit</p> <p>Zugriffsrecht: Lesen</p> <p>Datentyp: CHAR[9]</p> <p>Format: ASCII-String</p> <p>Anzahl Register: 5</p>    |
| <b>HEX 0x1110</b><br><b>DEC 4368</b> | <p><b>&lt;fw_app_ver_num&gt;</b></p> <p>Kurzbeschreibung: Mit diesem Parameter kann die Version der Software als Zahl ausgelesen werden.</p> <p>Parametername: Applikation-Firmware Version kurz</p> <p>Zugriffsrecht: Lesen</p> <p>Datentyp: UINT16</p> <p>Anzahl Register: 1</p>                                |

**HEX 0x1118**  
**DEC 4376**

**<fw\_app\_ver\_txt>**

Kurzbeschreibung: Mit diesem Parameter kann die Version der Software als Text ausgelesen werden.

Parametername: Application-Firmware Version

Zugriffsrecht: Lesen

Datentyp: CHAR[22]

Format: ASCII-String

Anzahl Register: 11

**HEX 0x1138**  
**DEC 4408**

**<mac\_addr>**

Kurzbeschreibung: Mit diesem Parameter kann die MAC-Adresse des Moduls ausgelesen werden.

Parametername: MAC-Adresse

Zugriffsrecht: Lesen

Datentyp: UINT8

Format: MAC

Anzahl Register: 3

**HEX 0x11A0**  
**DEC 4512**

**<baudrate>**

Kurzbeschreibung: Mit diesem Parameter kann die der Modbus-RTU Kommunikation zugrunde liegende RS-485 Kommunikation gelesen und geschrieben werden

Parametername: Modbus-RTU Baudrate

Zugriffsrecht: Lesen und Schreiben

Datentyp: UINT32

Zulässige Werte: 19200, 115200, 230400, 460800, 921600

Besonderheiten: Eine Änderung wird erst nach einem Neustart übernommen.

Anzahl Register: 2

**HEX 0x11A8**  
**DEC 4520**

**<slave\_id>**

Kurzbeschreibung: Mit diesem Parameter kann die Modbus-Slave-ID des Moduls geschrieben und ausgelesen werden.

Parametername: Modbus-RTU Slave ID

Zugriffsrecht: Lesen und Schreiben

Datentyp: UINT8

Besonderheiten: Modbus-RTU spezifisch sind lediglich Werte von 1 bis 247 erlaubt.

Anzahl Register: 1

**HEX 0x1330**  
**DEC 4912**

**<enable\_softreset>**

Kurzbeschreibung: Mit diesem Parameter kann die Funktion "Neustart" freigeschaltet werden.

Parametername: Softwareneustart aktivieren

Zugriffsrecht: Lesen und Schreiben

Datentyp: BOOL

Werte: 0 = Funktion ausgeschaltet  
1 = Funktion eingeschaltet

Anzahl Register: 1

**HEX 0x1400**  
**DEC 5120**

**<system\_uptime>**

Kurzbeschreibung: Mit diesem Parameter kann die Betriebszeit ausgelesen werden, die seit letztem (Neu-) Start des Moduls verstrichen ist.

Parametername: Systembetriebszeit

Zugriffsrecht: Lesen

Datentyp: UINT32

Einheit: Sekunden [s]

Anzahl Register: 2

## 5 Inbetriebnahme

### 5.1 Sicherheit

Die Inbetriebnahme des Moduls darf nur durch Fachpersonal mit Programmier- und Schnittstellenkenntnissen durchgeführt werden!



#### ⚠️ WARNUNG

##### Verletzungsgefahr durch Quetschen und Stoßen!

Beim Verfahren der Grundbacken, durch Bruch oder Lösen der Greiferfinger oder bei Werkstückverlust kann es zu schweren Verletzungen kommen.

- Geeignete Schutzausrüstung tragen.
- Nicht in die offene Mechanik und in den Bewegungsbereich des Produkts greifen.



#### ⚠️ VORSICHT

##### Verletzungsgefahr durch elektromagnetische Störungen!

Elektromagnetische Störungen können Fehlfunktionen verursachen und zu unerwarteten Bewegungen führen.

- Elektrische Komponenten z. B. Sensoren, Steuerungen etc. nach EN 61000-5-7 verwenden.

### 5.2 Systemintegration

Für den Betrieb innerhalb der Anlage steht das Kommunikationsprotokoll "SCHUNK Flexible Protocol" zur Verfügung.

Weiterführende Informationen zur Kommunikation, Modulfunktionen und Parametern siehe entsprechende Abschnitte in dieser Anleitung.

#### Überblick

- Modul ist montiert und elektrisch angeschlossen. Weitere Hinweise siehe Montage- und Betriebsanleitung, ▶ 1.1 [📄 7].
- 1. Logik- und Leistungsspannung aktivieren.
  - ⇒ LED LOG und PWR leuchten grün.
- 2. Kabel für Kommunikation anschließen.
  - ⇒ Kommunikation wird durch LED-Statusanzeige zurückgemeldet.
- 3. Steuerung und Modul konfigurieren, ▶ 5.3 [📄 68].
- 4. Programmablauf festlegen.

## 5.3 Inbetriebnahme Modbus RTU

### 5.3.1 Allgemeine Kommunikationseinstellungen

Für die Inbetriebnahme eines Moduls als Modbus-RTU Slave müssen dem Modbus-RTU Master die grundsätzlichen Kommunikationseinstellungen (Betriebsart, Übertragungsmodus, Baudrate und Slave-ID) bekannt gemacht werden.

#### Betriebsart

Das Modul verwendet die Betriebsart "Halbduplex" über RS485 (EIA/TIA-485) im Zweidraht-Betrieb. Bei der Kommunikation werden damit 3 Leitungen verwendet: Masse (Ground) und die beiden differentiellen Signalleitungen A und B.

#### Übertragungsmodus

Der Übertragungsmodus des Moduls ist auf 8 Datenbits, gerade (even) Parität und 1 Stopp-Bit (8E1) festgelegt. Damit werden pro Datenframe 11 Bits (1 Start-Bit, 8 Daten-Bits, 1 Even-Parity-Bit und 1 Stop-Bit) übertragen. Flusskontrolle ist "keine", RTS-Ein- und Ausschaltverzögerung ist 0, keine Verzögerung.

---

#### HINWEIS

Im Modbus Kommunikationsnetzwerk darauf achten, dass der RTU Übertragungsmodus bei allen Bus-Teilnehmern gleich ist.

---

#### Baudrate

Im Auslieferungszustand ist die Baudrate des Moduls auf 115200 Bit/s eingestellt. Die Baudrate kann durch Einstellen des Parameters <baudrate> (► 4.2 [□ 65]) oder über Modbus-RTU Registerzugriffe geändert werden. Mögliche Baudraten sind 19200, 115200 (Default), 230400, 460800, 921600 Bit/s.

---

#### HINWEIS

Im Modbus Kommunikationsnetzwerk darauf achten, dass die Baudrate bei allen Bus-Teilnehmern gleich ist.

---

#### Slave-ID

Im Auslieferungszustand ist die Slave-ID des Moduls dezimal 12 (= 0x0C hexadezimal) eingestellt. Die Slave-ID kann durch Ändern des Parameters <slave\_id> (► 4.2 [□ 65]) oder über Modbus-RTU Registerzugriffe eingestellt werden. Mögliche Slave-IDs sind dezimal 1 - 247 (0x01 - 0xF7 hexadezimal).

---

#### HINWEIS

Im Modbus Kommunikationsnetzwerk darauf achten, dass die Slave-IDs bei allen Bus-Teilnehmern unterschiedlich sind.

---

Folgende Tabelle zeigt eine Zusammenfassung der allgemeiner Kommunikationseinstellungen. Die Bezeichnungen können sich je nach Steuerungshersteller unterscheiden, daher werden auch geläufige alternative Bezeichnungen aufgeführt.

| Modbus-Bezeichnung | Default-Wert | Beschreibung                                     | Alternative Bezeichnungen                              |
|--------------------|--------------|--|--|
| Übertragungsmodus  | 8E1          | Anzahl Datenbits, Parität, Anzahl Stopp-Bits     | Transmission mode, Kommunikationsparameter / CommParam |
| Baudrate           | 115200       | Übertragungsrate in Bit/s                        |  |
| SlaveID            | 12           | Eindeutige Identifikation eines Modbus-RTU Slave | Stationsadresse / UnitID / Modbus-Adresse / mbAddr     |

### HINWEIS

Bei den Steuerungsherstellern überschneiden sich teilweise die alternativen Bezeichnungen. So bezeichnet Siemens die *SlaveID* mit *mbAddr*, während *MBAddr* bei Beckhoff die *Register-Adresse* bezeichnet.

### 5.3.2 Datenspezifische Kommunikationseinstellungen

Für den azyklischen Datenaustausch mit dem Modul werden die Modbus Funktionscodes "Read Holding Registers" zum Lesen und "Write Multiple Registers" zum Schreiben von Parametern verwendet, ▶ 2.1.2 [21].

Weiterhin muss beim Datenaustausch Folgendes angegeben werden:

- Über die Register-Adresse muss die Startadresse des zu übertragenden Parameters angegeben werden.
- Die Anzahl (Quantität) der zu übertragenden 16-Bit Register *und/oder* ein Byte-Count mit der Anzahl der übertragenen 8-Bit Bytes muss angegeben werden.

Die Datenübertragung wird mit einer 16-Bit Checksumme (CRC – Cyclic Redundancy Check) abgesichert, siehe dazu "Appendix B CRC Generation" unter [www.modbus.org/docs/](http://www.modbus.org/docs/).

Folgende Tabelle zeigt eine Zusammenfassung datenspezifischer Kommunikationseinstellungen. Die Bezeichnungen können sich je nach Steuerungshersteller unterscheiden, daher werden auch geläufige alternative Bezeichnungen aufgeführt.

| Modbus Bezeichnung       | Beschreibung  | Alternative Bezeichnung   |
|--------------------------|---|---|
| Read Holding Registers   | Funktionscode 3: Eine oder mehrere 16-Bit Register lesen      | ReadInputRegs   |
| Write Multiple Registers | Funktionscode 16: Eine oder mehrere 16-Bit Register schreiben | WriteRegs   |
| Register-Adresse         | Adresse der zu lesenden oder schreibenden Daten               | Modbus-Datenadresse / MAddr (Beckhoff),<br>Datenanfangsadresse / dataAddr (Siemens) |
| Quantity                 | Anzahl der zu übertragenden 16-Bit Register                   | Quantity (Beckhoff),<br>Datenlänge / dataLen (Siemens)                              |
| Byte-Count               | Anzahl der übertragenen 8-Bit Bytes                           | cbLength (Beckhoff)   |
| CRC                      | 16-Bit Checksumme   | CRC   |

### HINWEIS

Bei den Steuerungsherstellern überschneiden sich teilweise die alternativen Bezeichnungen. So bezeichnet Siemens die *SlaveID* mit *mbAddr*, während *MAddr* bei Beckhoff die *Register-Adresse* bezeichnet.

### 5.3.3 Datenaustausch

Mit den Funktionscodes "Read Holding Registers" und "Write Multiple Registers" kann prinzipiell auf jeden Parameter zugegriffen werden. Lese- und Schreibbeschränkungen bleiben dabei erhalten, so kann z. B. auf nur-Lese-Parameter *nicht* schreibend zugegriffen werden.

Für die Ansteuerung des Moduls ist neben dem zyklischen Auslesen des Parameters <plc\_sync\_input> (► 4.2 [□ 52]) auch das zyklische Schreiben des Parameters <plc\_sync\_output> (► 4.2 [□ 52]) erforderlich:

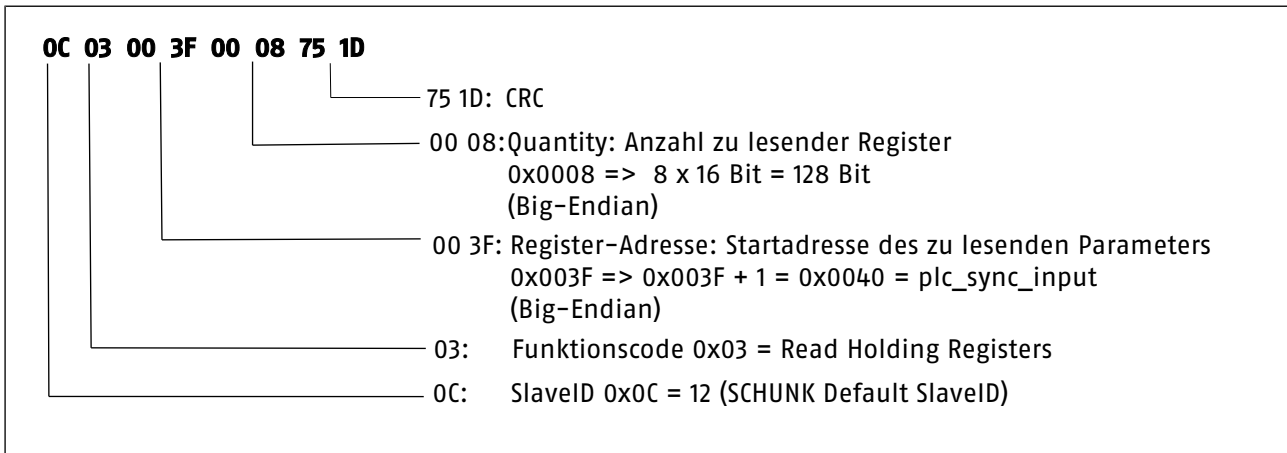
- Der Master erhält Informationen über den Zustand des Moduls durch Lesen des im Parameter <plc\_sync\_input> enthaltenen Status-Doppelworts, der aktuellen Position sowie des Diagnose-Doppelworts (Warnungs- und Fehlercode).
- Entsprechend kann der Master das Modul steuern durch Schreiben des im Parameter <plc\_sync\_output> enthaltenen Steuer-Doppelworts, dem Positionswert, dem Geschwindigkeitswert und dem Greifkraftwert, ► 2 [□ 12].

Folgende Beispiele zeigen einen möglichen Modbus-Datenaustausch. Dabei werden die Daten per RS485 übermittelt, der Master kommuniziert mit dem Modul (Standard-SlaveID 12 (0x0C)).

**Beispiel 1**

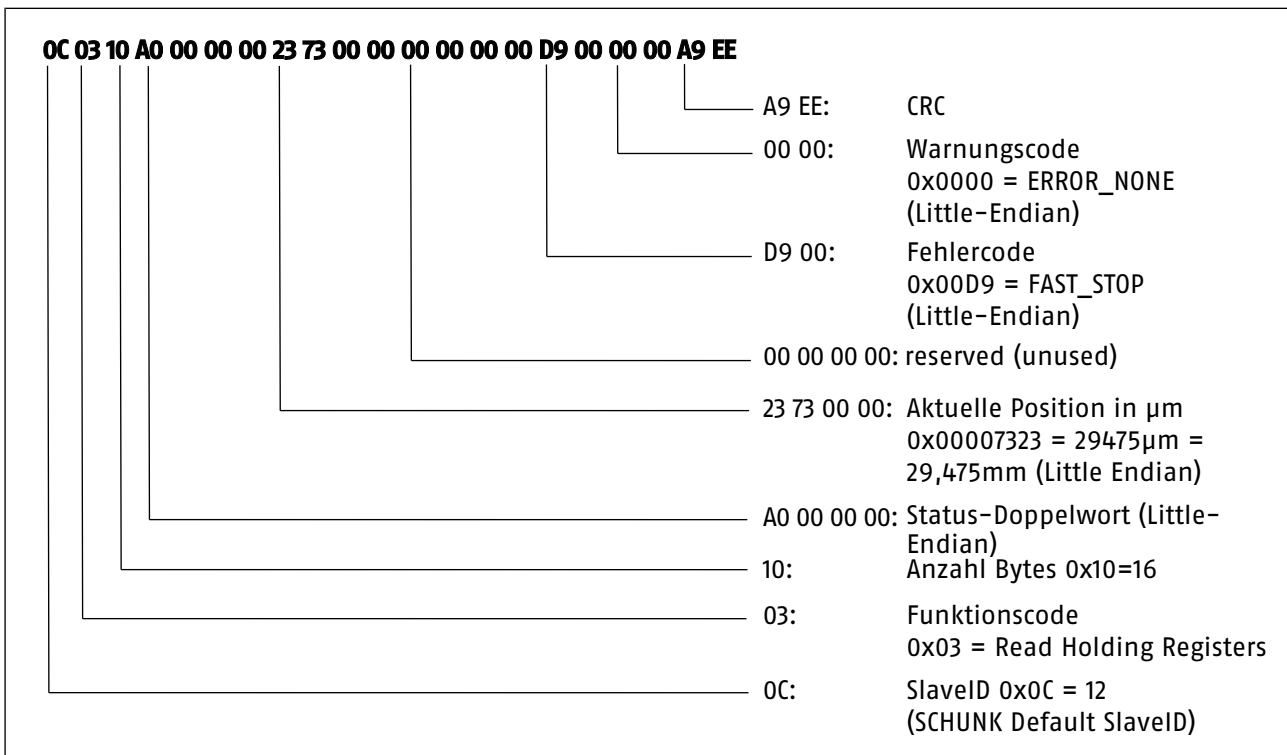
**Lesen der zyklischen Eingangsdaten aus dem Parameter <plc\_sync\_input>**

Anfrage (Request) vom Master zum Slave (8 Bytes):



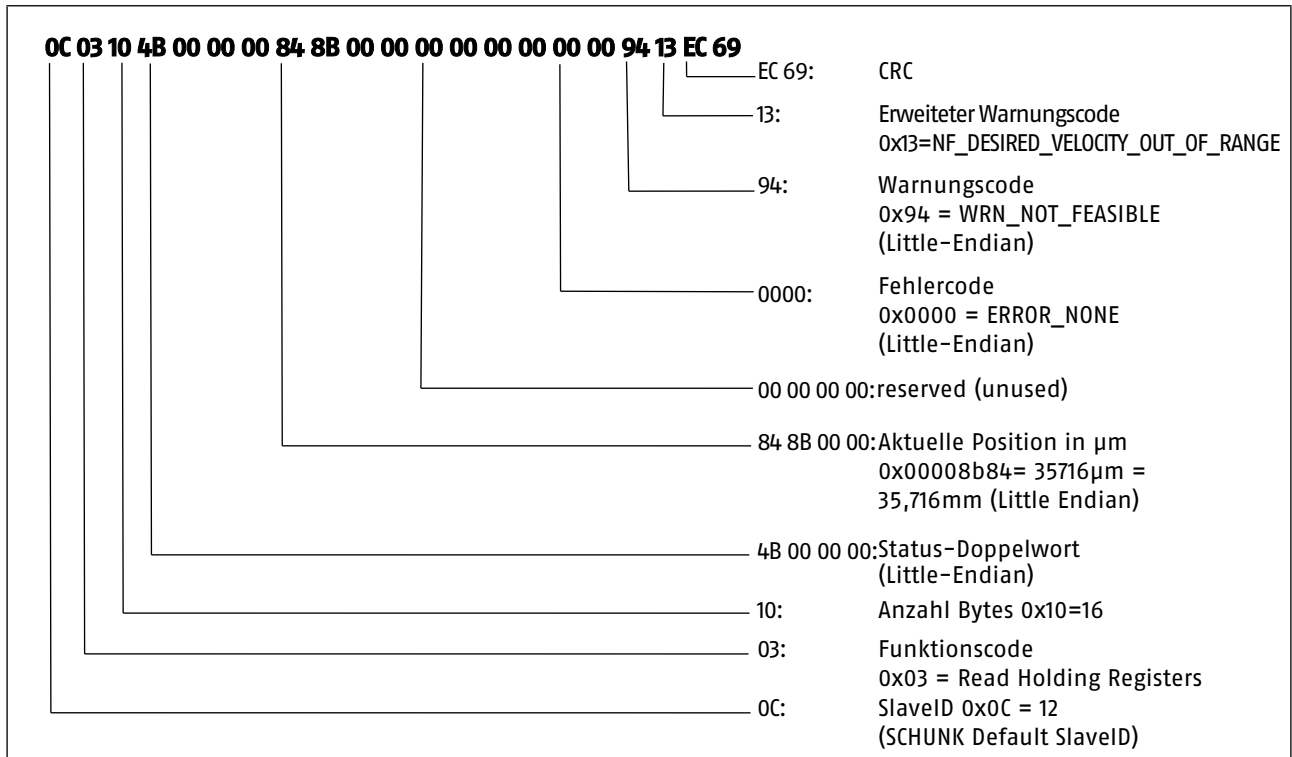
Anfrage vom Master zum Slave beim Lesen der Eingangsdaten

Antwort (Response) vom Slave an den Master (21 Bytes):



Antwort vom Slave an den Master beim Lesen der Eingangsdaten

Alternative Antwort (Response) vom Slave an den Master  
(21 Bytes) mit erweitertem Warnungscode:

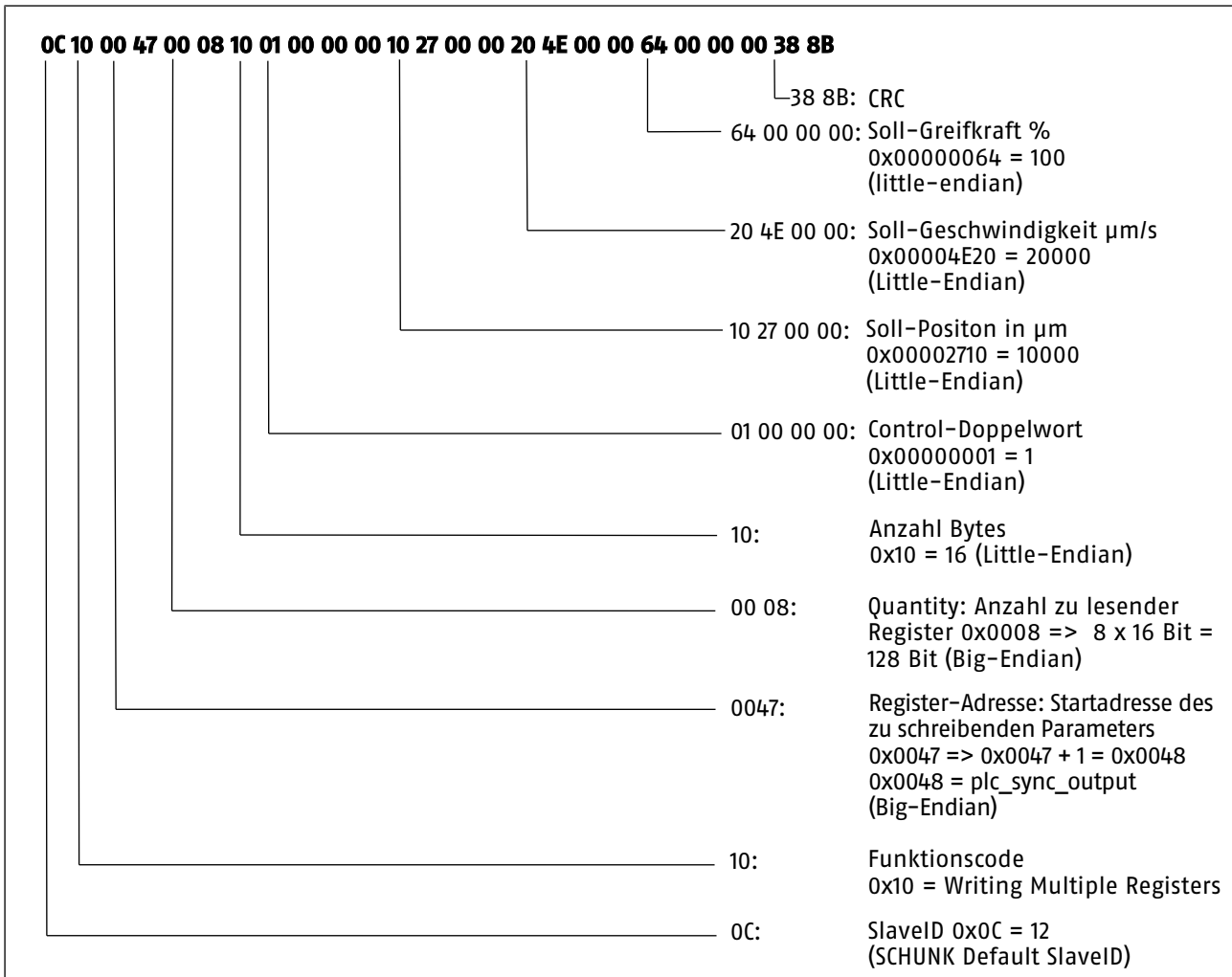


Antwort vom Slave an den Master beim Lesen der Eingangsdaten mit erweitertem Warnungscode

**Beispiel 2**

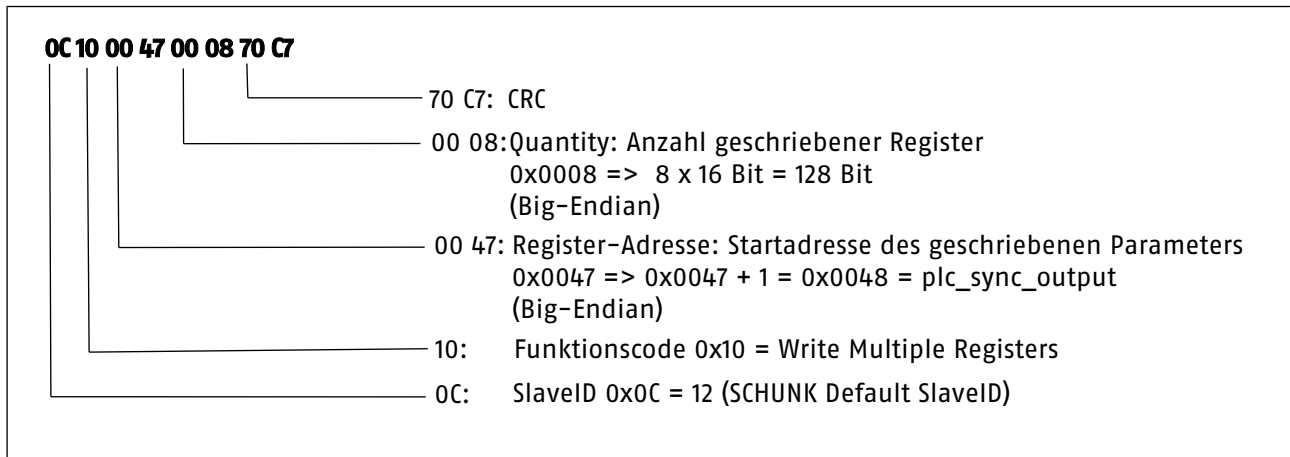
**Schreiben der zyklischen Ausgangsdaten in den Parameter  
<plc\_sync\_output>**

Anfrage (Request) vom Master zum Slave (25 Bytes):



Anfrage vom Master zum Slave beim Schreiben der Ausgangsdaten

Antwort (Response) vom Slave an den Master (8 Bytes):



Antwort vom Slave an den Master beim Schreiben der Ausgangsdaten

---

**HINWEIS**

- Aus den Beispielen ist ersichtlich, dass die jeweils auf dem Bus übertragene Register-Adresse um eins kleiner ist als die im Slave tatsächlich angesprochene Adresse. Hierbei handelt es sich um eine Modbus-Konvention. Je nach Steuerungshersteller muss entweder die um eins verkleinerte Register-Adresse oder aber die Standard-Register-Adresse angegeben werden. Im zweiten Fall passt die Steuerung die Adresse bei der Übertragung automatisch an.
  - Modbus RTU Protokoll-Daten mit mehr als 8 Bit, z. B. Quantity (Anzahl Register) oder die Register-Adresse werden im Big-Endian Format übertragen.
  - Modulspezifische Parameter werden im Little-Endian Format übertragen.
-

## 6 Diagnose

Die Diagnose dient der Systemüberwachung und reagiert mit dem Generieren von entsprechenden Diagnose Codes auf erkannte Diagnoseereignisse. Die Diagnose des Moduls läuft permanent im Hintergrund.

### Diagnoseereignisse

Diagnoseereignisse unterteilen sich in Warnungs- und Fehlerereignisse. Informationen zu aufgetretenen Diagnoseereignissen werden in den zyklischen Eingangsdaten übertragen.

### 6.1 Warnungen

Wird durch die Diagnose erkannt, dass ein Warnungsereignis aufgetreten ist, wechselt das Modul in den Warnungszustand. Ein Warnungscode wird generiert und zyklisch übertragen. Das Anliegen einer Warnung wird durch Setzen des Statusbits "warning" angezeigt.

---

#### HINWEIS

- Liegt mehr als eine Warnung an, wird der zuletzt aufgetretene Warnungscode zyklisch übertragen.
  - Liegt eine Warnung an, die in folgender Liste nicht aufgeführt ist, SCHUNK Service kontaktieren.
- 

### Warnungszustand

Im Warnungszustand ist das Modul weiterhin betriebsbereit, wird allerdings unter Umständen an der Grenze zum Fehlerzustand betrieben.

### Warnungscode

Zu jedem erkennbaren Warnungsereignis gehört ein eindeutiger Warnungscode, der in den zyklischen Eingangsdaten übertragen wird.

### Quittieren

Warnungen sind sowohl quittierbar als auch selbstquittierend. Durch Setzen des Steuerbits "acknowledge" (Bit 2) wird das Quittieren einer anliegenden Warnung angestoßen, ▶ 7.2 [📄 90]. Ist die Ursache des Warnungsereignisses zu diesem Zeitpunkt nicht mehr vorhanden, wird die Warnung quittiert. Sollte die Ursache des Warnungsereignisses noch immer vorhanden sein, kann die Warnung zu diesem Zeitpunkt nicht quittiert werden und liegt weiterhin an. Wird vom Modul erkannt, dass die Ursache eines anliegenden Warnungsereignisses nicht mehr vorhanden ist, quittiert sich diese Warnung selbstständig.

### Erkennbare Warnungsereignisse

Im Folgenden sind alle Warnungsereignisse und die dazugehörigen Warnungscodes aufgelistet, die durch das Modul erkannt werden können.

**HEX 0x90 / DEC 144**

**WRN\_LGC\_TEMP\_LO**  
**WARNING\_LOGIC\_TEMP\_LOW**

Diagnose Ereignis: Die gemessene Logik-Temperatur ist zu niedrig.  
Quittierbarkeit: selbstquittierend

**HEX 0x92 / DEC 146**

**WRN\_MOT\_TEMP\_LO**  
**WARNING\_MOTOR\_TEMP\_LOW**

Diagnose Ereignis: Die gemessene Motor-Temperatur ist zu niedrig.  
Quittierbarkeit: selbstquittierend

**HEX 0x93 / DEC 147**

**WRN\_MOT\_TEMP\_HI**  
**WARNING\_MOTOR\_TEMP\_HIGH**

Diagnose Ereignis: Die gemessene Motor-Temperatur ist zu hoch.  
Quittierbarkeit: selbstquittierend

**HEX 0x94 / DEC 148**

**WRN\_NOT\_FEASIBLE**  
**WARNING\_CMD\_NOT\_FEASIBLE**

Diagnose Ereignis: Der an das Modul gesendete Steuerbefehl ist nicht durchführbar.  
Quittierbarkeit: quittierbar/selbstquittierend

---

**HINWEIS**

- Im Diagnosespeicher werden weitere Informationen zur Ursache dieser Warnung abgespeichert. Diagnosespeicher auslesen, siehe Parameter <sys\_msg\_req> (► 4.2 [ 53]).
- Falls diese Warnung angezeigt wird, kann über die zyklischen Daten die Ursache für die Nichtdurchführbarkeit ausgelesen werden, ► 2.1.1 [ 12]. Die Liste der Ursachen befindet sich im Anhang dieses Dokuments, ► 7.4 [ 96].

**HEX 0x95 / DEC 149**

**WRN\_POS\_LIMIT**  
**WARNING\_POSITION\_LIMIT\_REACHED**

Diagnose Ereignis: Die Bewegung beim Tipp-Betriebs wurde durch Erreichen der minimalen oder maximalen Position automatisch beendet.  
Quittierbarkeit: selbstquittierend

**HEX 0x96 / DEC 150**

**WRN\_LGC\_VOLT\_LO**  
**WARNING\_LOGIC\_VOLTAGE\_LOW**

Diagnose Ereignis: Die gemessene Logik-Versorgungsspannung ist zu niedrig.  
Quittierbarkeit: selbstquittierend

|                           |  |
|---------------------------|--|
| <b>HEX 0x97 / DEC 151</b> | <b>WRN_LGC_VOLT_HI</b><br><b>WARNING_LOGIC_VOLTAGE_HIGH</b><br>Diagnose Ereignis: Die gemessene Logik-Versorgungsspannung ist zu hoch.<br>Quittierbarkeit: selbstquittierend   |
| <b>HEX 0x98 / DEC 152</b> | <b>WRN_MOT_VOLT_LO</b><br><b>WARNING_MOTOR_VOLTAGE_LOW</b><br>Diagnose Ereignis: Die gemessene Motor-Versorgungsspannung ist zu niedrig.<br>Quittierbarkeit: selbstquittierend |
| <b>HEX 0x99 / DEC 153</b> | <b>WRN_MOT_VOLT_HI</b><br><b>WARNING_MOTOR_VOLTAGE_HIGH</b><br>Diagnose Ereignis: Die gemessene Motor-Versorgungsspannung ist zu hoch.<br>Quittierbarkeit: selbstquittierend   |

## 6.2 Fehler

Wird durch die Diagnose erkannt, dass ein Fehlerereignis aufgetreten ist, wechselt das Modul in den Fehlerzustand. Ein Fehlercode wird generiert und zyklisch übertragen. Das Anliegen eines Fehlers wird durch Setzen des Statusbits "error" angezeigt.

---

### HINWEIS

- Liegt mehr als ein Fehler an, wird der zuerst aufgetretene Fehlercode übertragen.
  - Liegt ein Fehler an, der in folgender Liste nicht aufgeführt ist, SCHUNK Service kontaktieren.
- 

|                      |   |
|----------------------|---|
| <b>Fehlerzustand</b> | Im Fehlerzustand ist das Modul nicht mehr betriebsbereit. Mit dem Wechsel in den Fehlerzustand wird das Modul in den Stillstand gezwungen.<br>Bei Modulen mit GPE: Die GPE wird aktiviert.  |
| <b>Fehlercode</b>    | Zu jedem erkennbaren Fehlerereignis gehört ein eindeutiger Fehlercode, der in den zyklischen Eingangsdaten übertragen wird.   |
| <b>Quittieren</b>    | Fehler werden unterschieden in quittierungspflichtige und nicht quittierbare Fehler.<br><b>Quittierungspflichtige Fehler:</b> Durch Setzen des Steuerbits "acknowledge" wird das Quittieren eines anliegenden quittierungspflichtigen Fehlers angestoßen. |

Ist die Ursache des Fehlereignisses zu diesem Zeitpunkt nicht mehr vorhanden, wird der Fehler quittiert. Sollte die Ursache des Fehlerereignisses noch immer vorhanden sein, kann der Fehler zu diesem Zeitpunkt nicht quittiert werden und liegt weiterhin an.

**Nicht quittierbare Fehler:** Bei auftretenden schwerwiegenden Fehlern kann das Modul bei Wiederinbetriebnahme beschädigt oder zerstört werden. Der Fehlerzustand kann nicht verlassen werden. In solchen Fällen den SCHUNK Service kontaktieren.

Im Folgenden sind alle Fehlerereignisse und die dazugehörigen Fehlercodes aufgelistet, die durch das Modul erkannt werden können.

**Erkennbare Fehlerereignisse**

**HEX 0x28 / DEC 040**

**ERR\_BT\_FAILED**  
**ERROR\_BRAKE\_TEST\_FAILED**

Diagnose Ereignis: Der Bremsentest wurde erfolglos durchgeführt.

Quittierbarkeit: quittierungspflichtig

**HEX 0x6C / DEC 108**

**ERR\_MOT\_TEMP\_LO**  
**ERROR\_MOTOR\_TEMP\_LOW**

Diagnose Ereignis: Die gemessene Motor-Temperatur ist zu niedrig.

Quittierbarkeit: quittierungspflichtig

**HEX 0x6D / DEC 109**

**ERR\_MOT\_TEMP\_HI**  
**ERROR\_MOTOR\_TEMP\_HIGH**

Diagnose Ereignis: Die gemessene Motor-Temperatur ist zu hoch.

Quittierbarkeit: quittierungspflichtig

**HEX 0x70 / DEC 112**

**ERR\_LGC\_TEMP\_LO**  
**ERROR\_LOGIC\_TEMP\_LOW**

Diagnose Ereignis: Die gemessene Logik-Temperatur ist zu niedrig.

Quittierbarkeit: quittierungspflichtig

**HEX 0x71 / DEC 113**

**ERR\_LGC\_TEMP\_HI**  
**ERROR\_LOGIC\_TEMP\_HIGH**

Diagnose Ereignis: Die gemessene Logik-Temperatur ist zu hoch.

Quittierbarkeit: quittierungspflichtig

**HEX 0x72 / DEC 114**

**ERR\_LGC\_VOLT\_LO**  
**ERROR\_LOGIC\_VOLTAGE\_LOW**

Diagnose Ereignis: Die gemessene Logik-Versorgungsspannung ist zu niedrig.

Quittierbarkeit: quittierungspflichtig

HEX 0x73 / DEC 115

**ERR\_LGC\_VOLT\_HI**  
**ERROR\_LOGIC\_VOLTAGE\_HIGH**

Diagnose Ereignis: Die gemessene Logik-Versorgungsspannung ist zu hoch.

Quittierbarkeit: quittierungspflichtig

HEX 0x74 / DEC 116

**ERR\_MOT\_VOLT\_LO**  
**ERROR\_MOTOR\_VOLTAGE\_LOW**

Diagnose Ereignis: Die gemessene Motor-Versorgungsspannung ist zu niedrig.

Quittierbarkeit: quittierungspflichtig

---

**HINWEIS**

Bei Modulen *mit* GPE: Solange GPE aktiviert ist, wird dieser Fehler *nicht* überwacht.

Bei Modulen *ohne* GPE: Die Überwachung dieses Fehlers ist permanent aktiv.

---

HEX 0x75 / DEC 117

**ERR\_MOT\_VOLT\_HI**  
**ERROR\_MOTOR\_VOLTAGE\_HIGH**

Diagnose Ereignis: Die gemessene Motor-Versorgungsspannung ist zu hoch.

Quittierbarkeit: quittierungspflichtig

---

**HINWEIS**

Bei Modulen *mit* GPE: Solange GPE aktiviert ist, wird dieser Fehler *nicht* überwacht.

Bei Modulen *ohne* GPE: Die Überwachung dieses Fehlers ist permanent aktiv.

---

HEX 0xD5 / DEC 213

**ERR\_SOFT\_LOW**  
**ERROR\_SOFT\_LOW**

Diagnose Ereignis: Das untere Softwarelimit wurde erreicht oder überfahren.

Quittierbarkeit: quittierungspflichtig

HEX 0xD6 / DEC 214

**ERR\_SOFT\_HIGH**  
**ERROR\_SOFT\_HIGH**

Diagnose Ereignis: Das obere Softwarelimit wurde erreicht oder überfahren.

Quittierbarkeit: quittierungspflichtig

**HEX 0xD9 / DEC 217**

**ERR\_FAST\_STOP**

**ERROR\_FAST\_STOP**

Diagnose Ereignis: Ein Schnellstopp wurde ausgelöst.

Quittierbarkeit: quittierungspflichtig

**HEX 0xE4 / DEC 228**

**ERR\_TOO\_FAST**

**ERROR\_TOO\_FAST**

Diagnose Ereignis: Die maximal erlaubte Geschwindigkeit wurde um Faktor 1.2 überschritten.

Quittierbarkeit: quittierungspflichtig

**HEX 0xEF / DEC 239**

**ERR\_COMM\_LOST**

**ERROR\_COMMUNICATION\_LOST**

Diagnose Ereignis: Die Kommunikationsverbindung zwischen dem Modul und der Gegenstelle (Steuerung oder App *Mechatronische Greifer*, Link MTSN) wurde unterbrochen.

Quittierbarkeit: quittierungspflichtig

**HEX 0xF1 / DEC 241**

**ERR\_MOV\_ABORT\_TO**

**ERROR\_MOVE\_ABORT\_TIMEOUT**

Diagnose Ereignis: Das Positionieren konnte nicht in dem dafür erwarteten Zeitraum durchgeführt werden.

Quittierbarkeit: quittierungspflichtig

**HEX 0xF4 / DEC 244**

**ERR\_MOVE\_BLOCKED**

**ERROR\_MOVE\_BLOCKED**

Diagnose Ereignis: Der Antrieb wurde blockiert.

Quittierbarkeit: quittierungspflichtig

## 7 Anhang

### 7.1 Anwendungsbeispiele

Die folgenden Anwendungsbeispiele beschreiben die Bedienung und das Verhalten des Moduls.

#### HINWEIS

Aufgrund der teilweise starken Abweichungen zwischen den Modulen kann es u.U. vorkommen, dass die hier beschriebenen Anwendungsbeispiele bei einem spezifischen Modul zu anderen Ergebnissen führen können. Zum Beispiel ist nicht jedes Modul in der Lage, die absolute Position 80 mm anzufahren. An den grundlegenden Aussagen der Anwendungsbeispiele ändert dies jedoch nichts.

| Szenariobeschreibung  | Beispiel            |
|---|---------------------|
| Eine absolute Positionsfahrt wird ausgeführt.   | ► Beispiel 1 [📄 82] |
| Eine relative Positionsfahrt wird ausgeführt.   | ► Beispiel 2 [📄 83] |
| Ein Werkstück wird gegriffen: <ul style="list-style-type: none"> <li>• ohne Nachgreifen</li> <li>• Werkstück-Halten durch Antriebsregelung</li> </ul>                                 | ► Beispiel 3 [📄 84] |
| Ein Werkstück wird gegriffen: <ul style="list-style-type: none"> <li>• mit Nachgreifen</li> <li>• Werkstück-Halten durch GPE</li> </ul>   | ► Beispiel 4 [📄 85] |
| Ein Werkstück wird im StrongGrip-Modus gegriffen: <ul style="list-style-type: none"> <li>• ohne Nachgreifen</li> <li>• Werkstück-Halten durch GPE</li> </ul>                          | ► Beispiel 5 [📄 86] |
| Ein Werkstück wird an erwarteter Position gegriffen: <ul style="list-style-type: none"> <li>• ohne Nachgreifen</li> <li>• Werkstück-Halten durch GPE</li> </ul>                       | ► Beispiel 6 [📄 87] |
| Ein Werkstück wird an erwarteter Position gegriffen: <ul style="list-style-type: none"> <li>• mit Nachgreifen</li> <li>• Werkstück-Halten durch Antriebsregelung</li> </ul>           | ► Beispiel 7 [📄 88] |
| Ein Werkstück wird an erwarteter Position wird StrongGrip-Modus gegriffen: <ul style="list-style-type: none"> <li>• ohne Nachgreifen</li> <li>• Werkstück-Halten durch GPE</li> </ul> | ► Beispiel 8 [📄 89] |

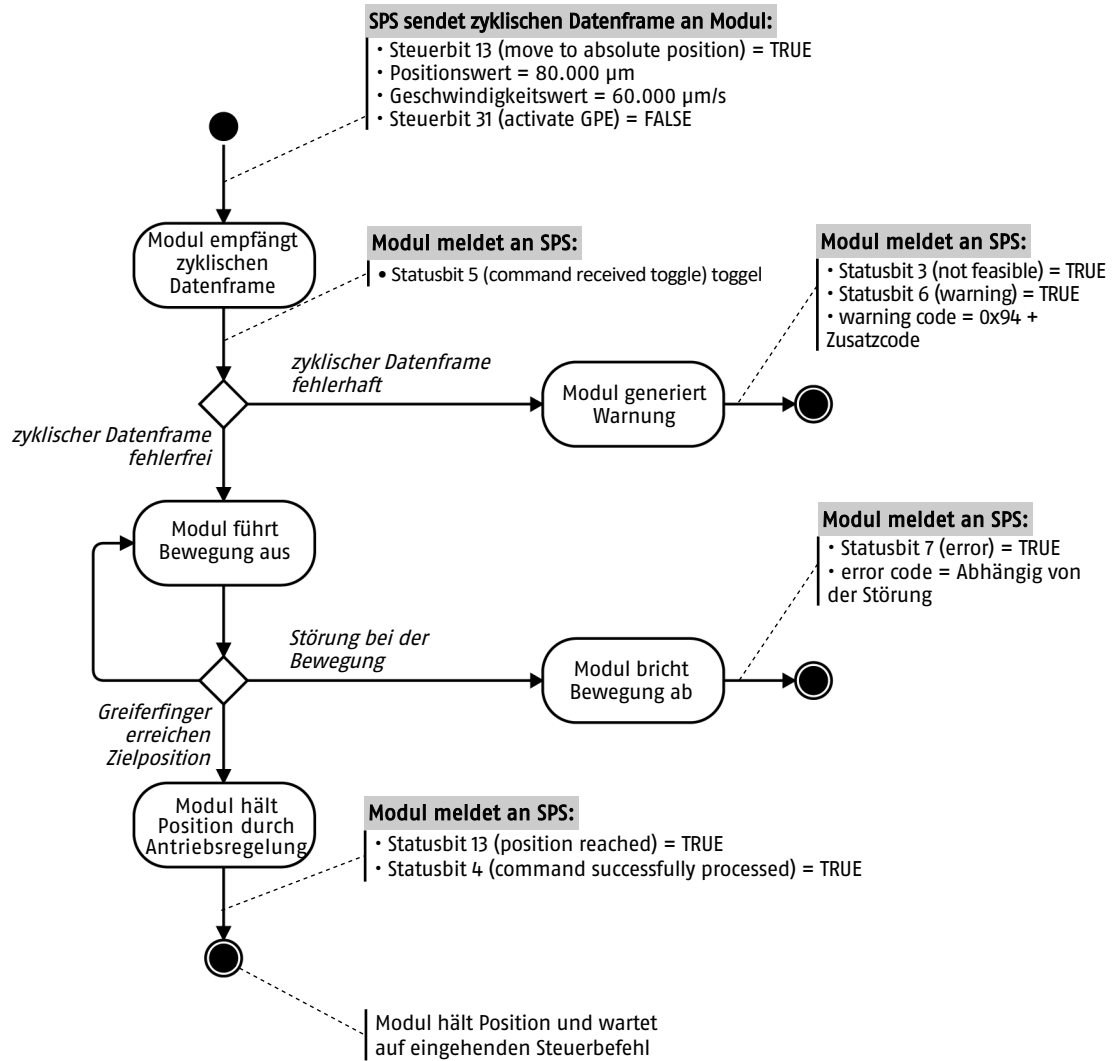
## BEISPIEL 1 Positionsfahrt absolut

### Anwendungsbeispiel - move to absolute position

- Bewegung auf absolute Position 80 mm
- Maximal zulässige Geschwindigkeit der Bewegung 60 mm/s
- Position-Halten durch Antriebsregelung

Voraussetzungen:

- Statusbit 0 (ready for operation) = TRUE
- keine Bewegung der Greiferfinger



Weitere Informationen siehe Kapitel ▶ 3.2.2 [ 28 ].

## BEISPIEL 2

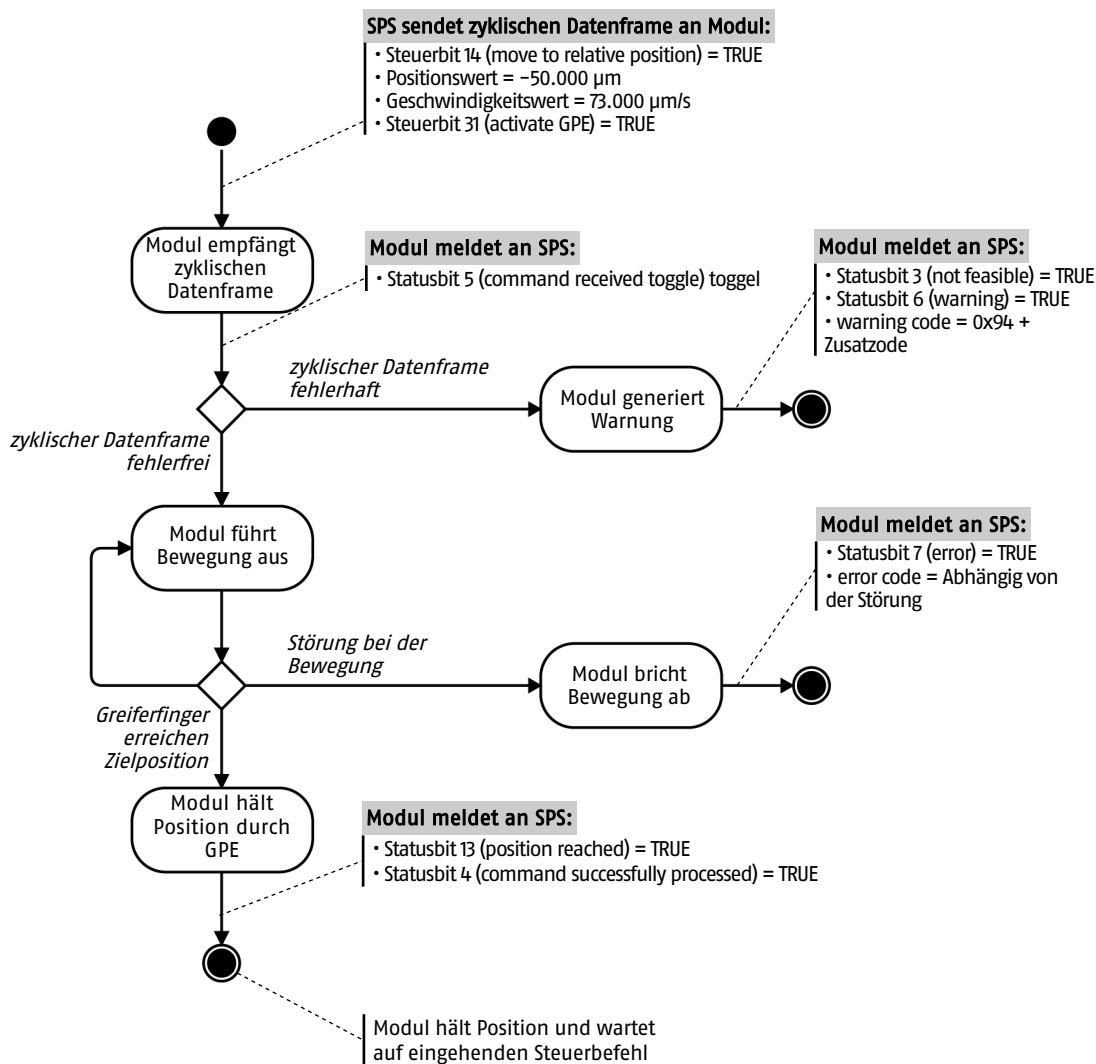
## Positionsfahrt relativ

**Anwendungsbeispiel – move to relative position**

- Bewegung von aktueller Position -50 mm
- Maximal zulässige Geschwindigkeit der Bewegung 73 mm/s
- Position-Halten durch GPE

Voraussetzungen:

- Statusbit 0 (ready for operation) = TRUE
- keine Bewegung der Greiferfinger



Weitere Informationen siehe Kapitel ▶ 3.2.3 [ 30 ].

### BEISPIEL 3 Werkstück-Greifen (1)

#### Anwendungsbeispiel – grip workpiece

- Werkstück mit 64 % Greifkraft greifen
- Werkstück von außen greifen
- ohne Nachgreifen
- Werkstück-Halten durch Antriebsregelung

Voraussetzungen:

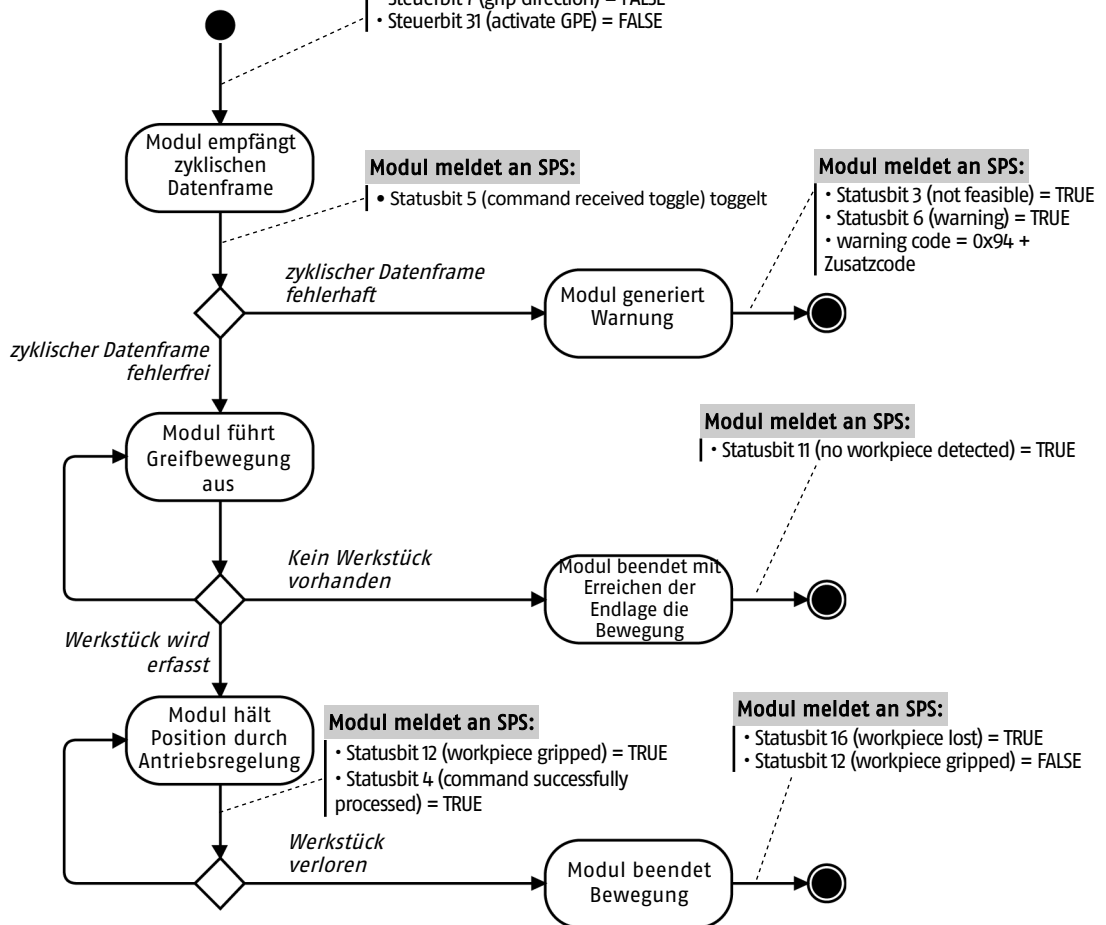
- Statusbit 0 (ready for operation) = TRUE
- keine Bewegung der Greiferfinger

**Parametrierung des Moduls:**

- grp\_prehold\_time = 0 ms

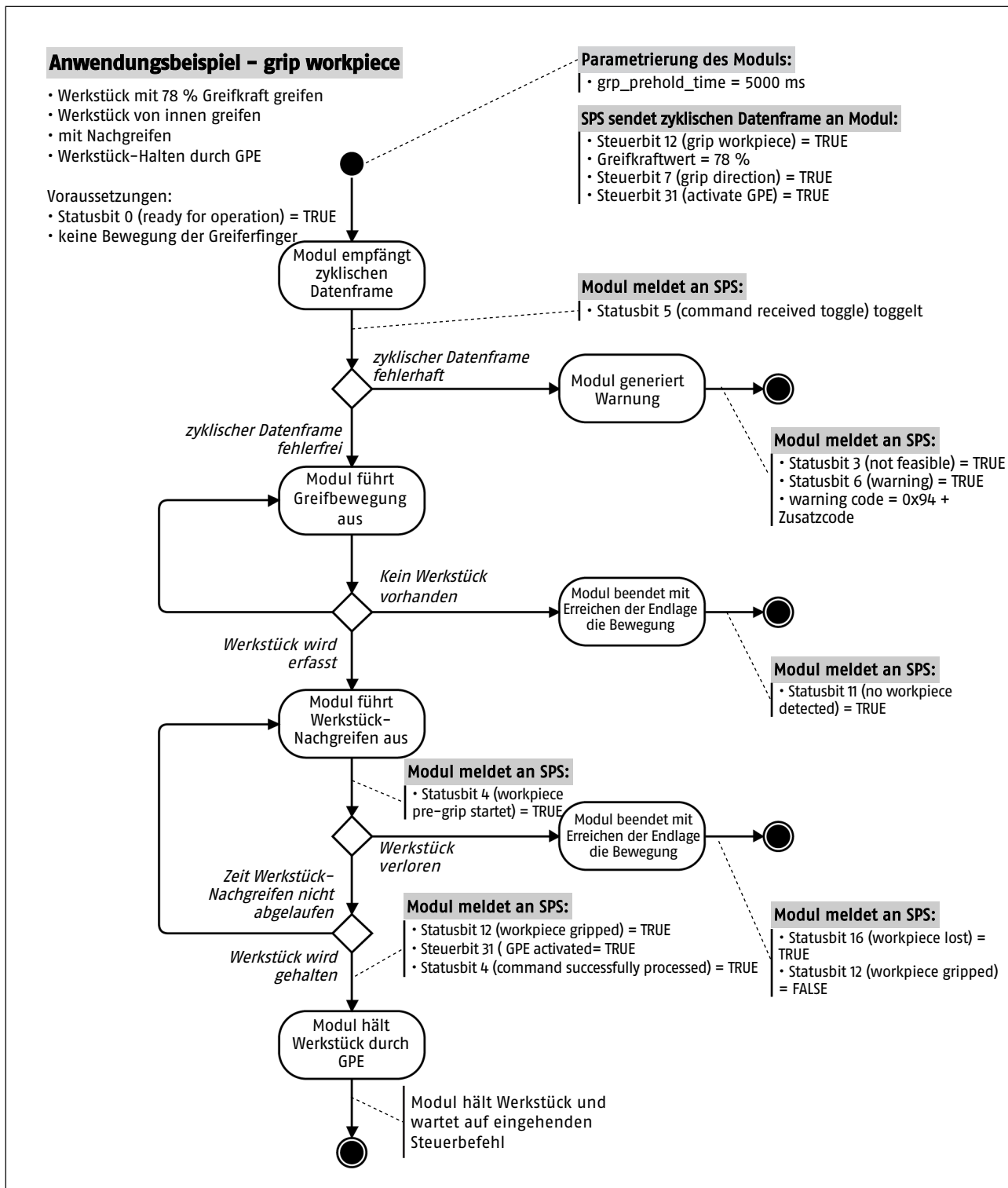
**SPS sendet zyklischen Datenframe an Modul:**

- Steuerbit 12 (grip workpiece) = TRUE
- Greifkraftwert = 64 %
- Steuerbit 7 (grip direction) = FALSE
- Steuerbit 31 (activate GPE) = FALSE



Weitere Informationen siehe Kapitel ▶ 3.3.1 [ 33 ].

### BEISPIEL 4 Werkstück-Greifen (2)



Weitere Informationen siehe Kapitel ▶ 3.3.1 [ 33 ].

## BEISPIEL 5 Werkstück-Greifen im StrongGrip-Modus

### Anwendungsbeispiel – grip workpiece im StrongGrip-Modus

- Werkstück mit 149% Greifkraft greifen
- Werkstück von außen greifen
- ohne Nachgreifen
- Werkstück-Halten durch GPE wird im StrongGrip automatisch aktiviert

Voraussetzungen:

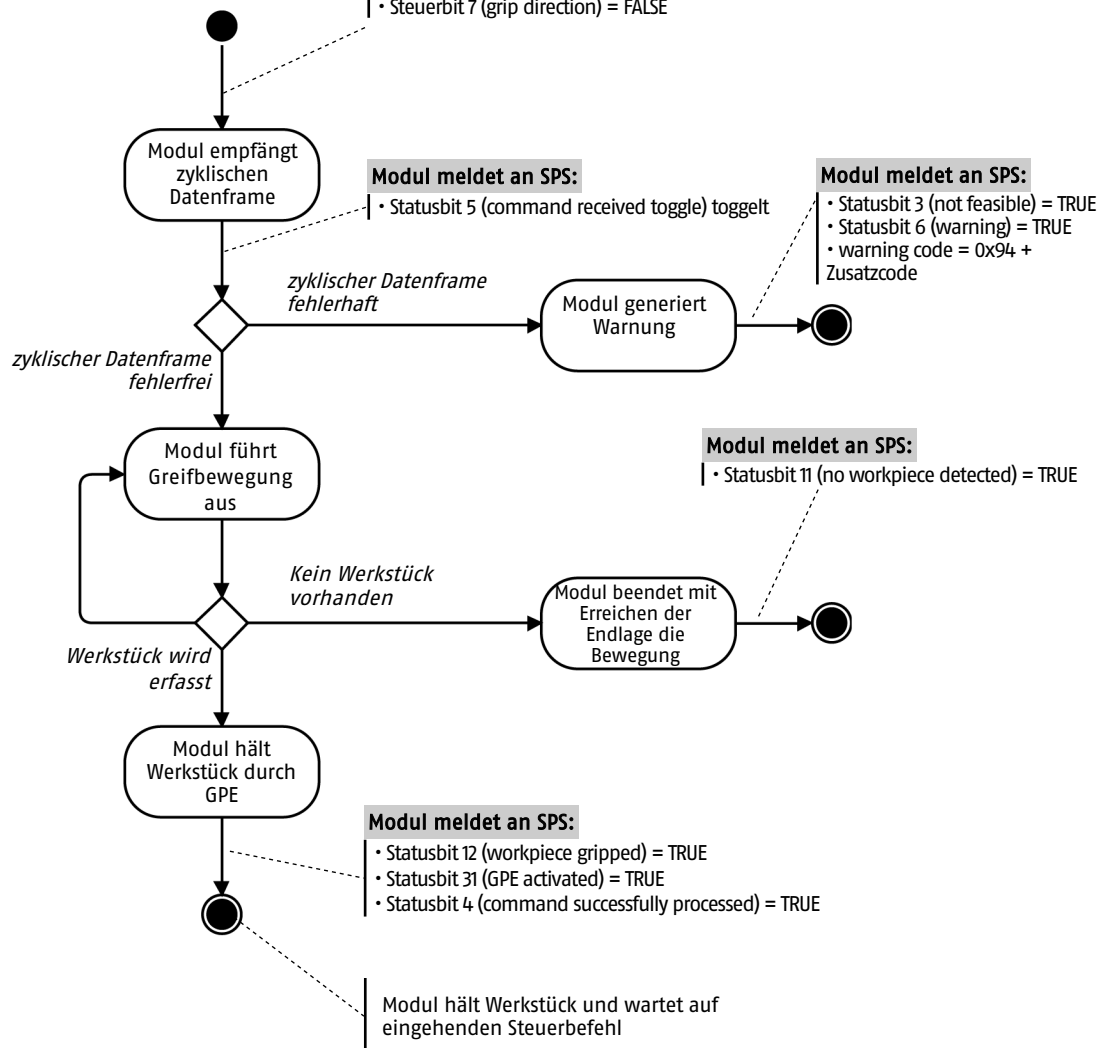
- Statusbit 0 (ready for operation) = TRUE
- keine Bewegung der Greiferfinger

#### Parametrierung des Moduls:

- `grp_prehold_time` = 0 ms

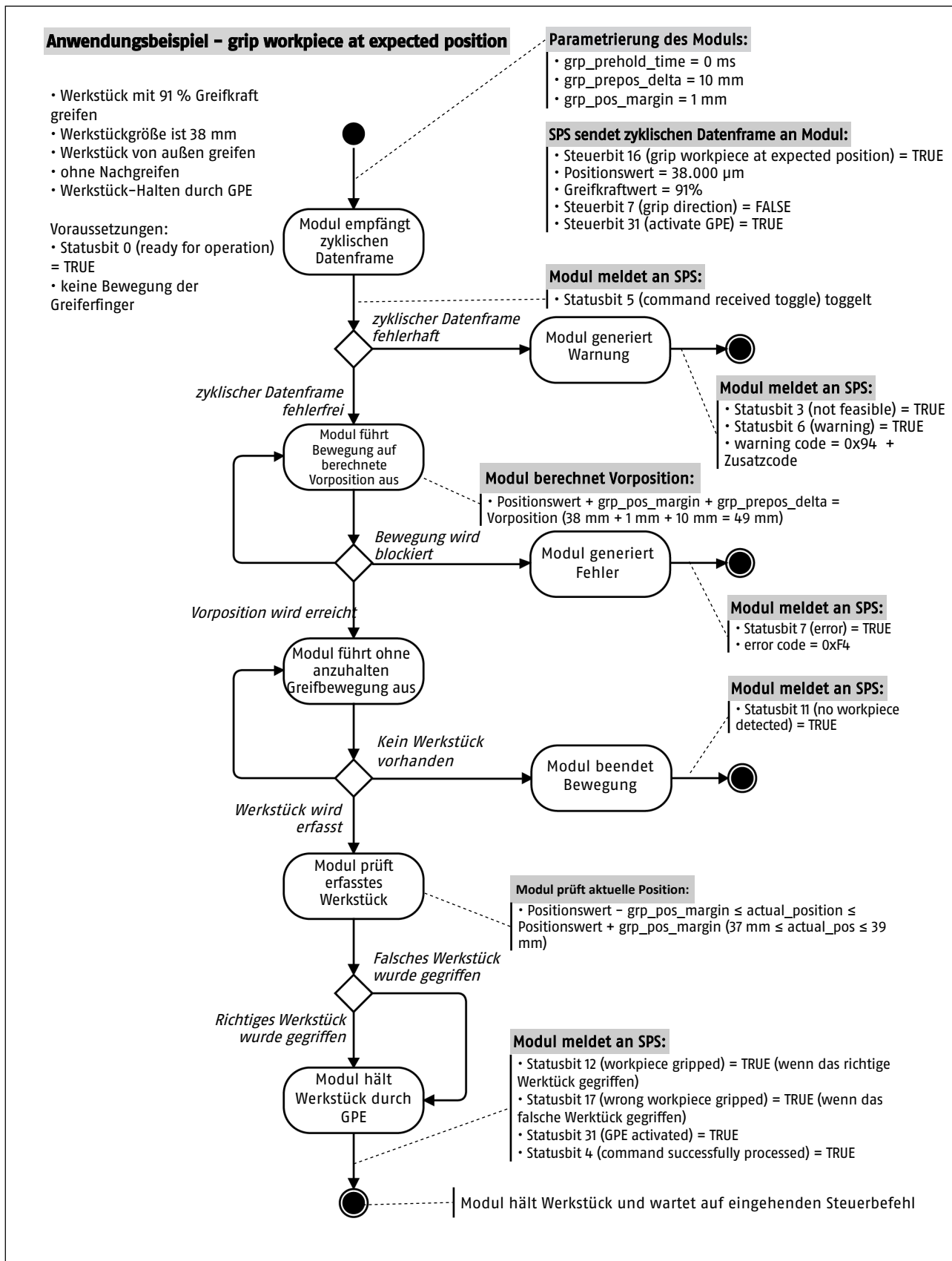
#### SPS sendet zyklischen Datenframe an Modul:

- Steuerbit 12 (grip workpiece) = TRUE
- Greifkraftwert = 149%
- Steuerbit 7 (grip direction) = FALSE



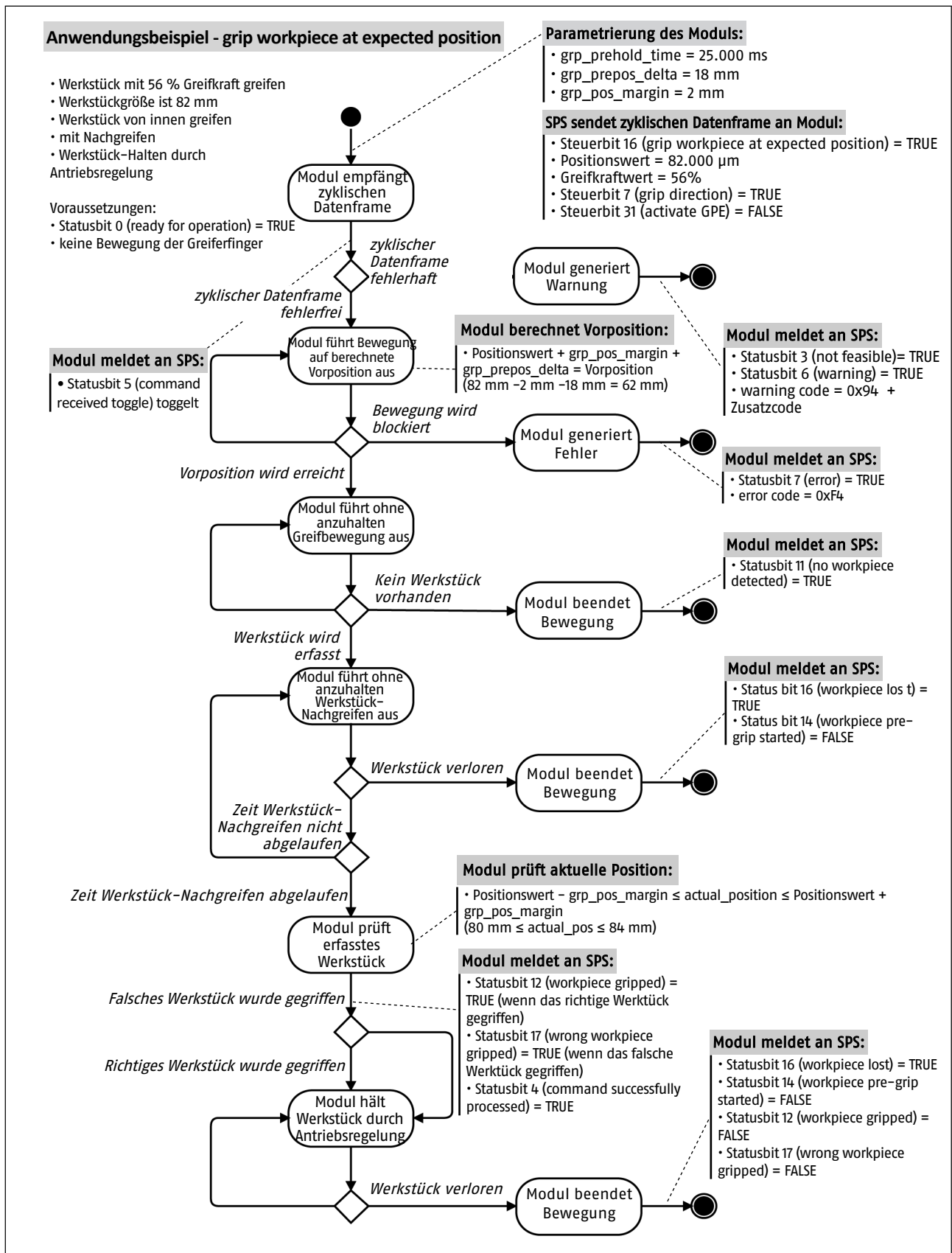
Weitere Informationen siehe Kapitel ▶ 3.3.1 [ 33 ].

## BEISPIEL 6 Werkstück-Greifen an erwarteter Position (1)



Weitere Informationen siehe Kapitel ▶ 3.3.2 [ 35 ].

## BEISPIEL 7 Werkstück-Greifen an erwarteter Position (2)



Weitere Informationen siehe Kapitel ▶ 3.3.2 [35].

## BEISPIEL 8 Werkstück-Greifen an erwarteter Position im StrongGrip-Modus

### Anwendungsbeispiel – grip workpiece at expected position im StrongGrip

- Werkstück mit 138 % Greifkraft greifen
- Werkstückgröße ist 21 mm
- Werkstück von außen greifen
- mit Nachgreifen
- Werkstück-Halten durch GPE (im StrongGrip zwingend notwendig)

- Voraussetzungen:
- Statusbit 0 (ready for operation) = TRUE
  - keine Bewegung der Greiferfinger

#### Parametrierung des Moduls:

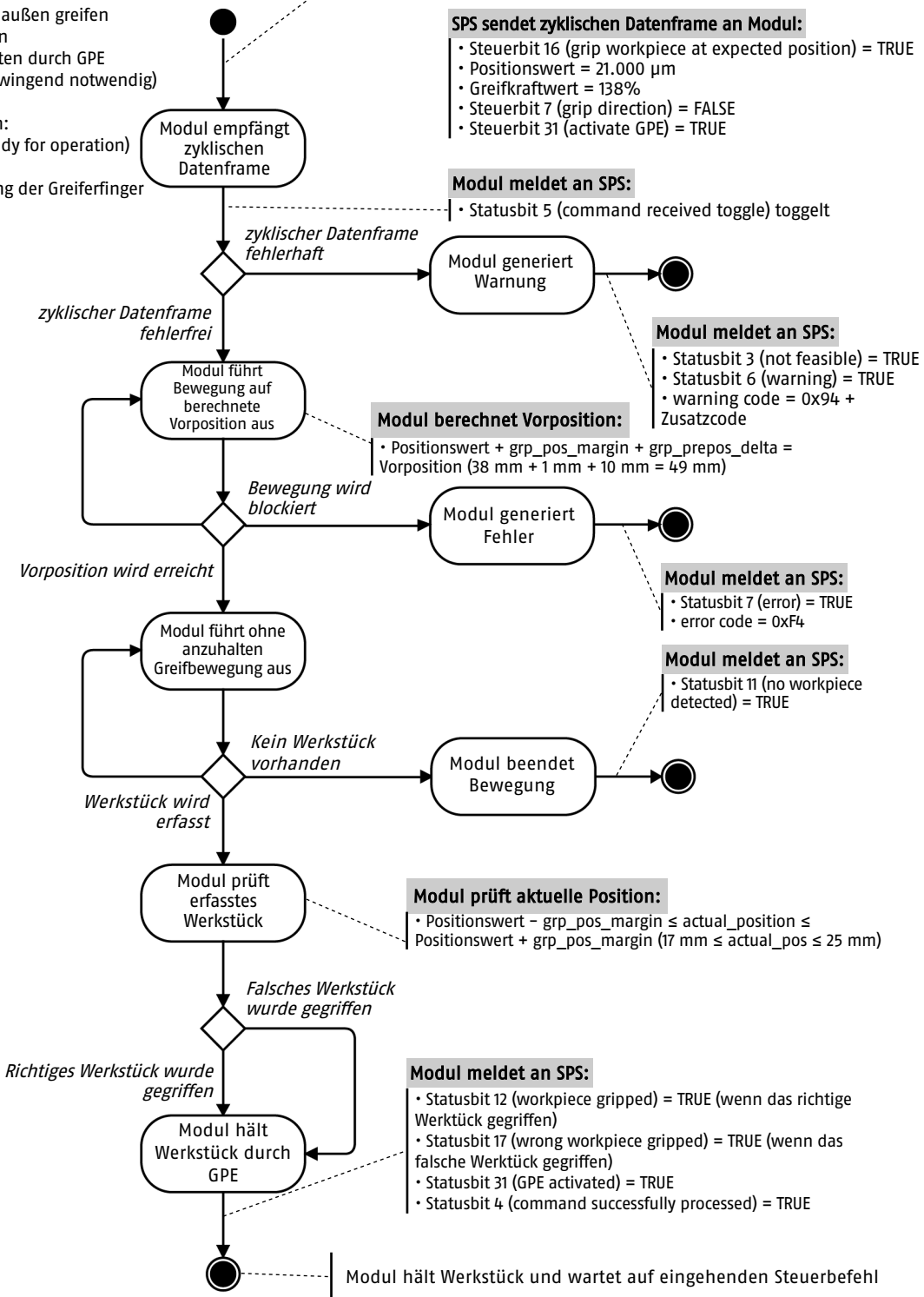
- grp\_prehold\_time = 2000 ms (max. Nachgreifzeit im StrongGrip= 2000 ms)
- grp\_prepos\_delta = 5mm
- grp\_pos\_margin = 4 mm

#### SPS sendet zyklischen Datenframe an Modul:

- Steuerbit 16 (grip workpiece at expected position) = TRUE
- Positionswert = 21.000 µm
- Greifkraftwert = 138%
- Steuerbit 7 (grip direction) = FALSE
- Steuerbit 31 (activate GPE) = TRUE

#### Modul meldet an SPS:

- Statusbit 5 (command received toggle) toggelt



Weitere Informationen siehe Kapitel ▶ 3.3.2 [35].

## 7.2 Steuerdoppelwort

Im Folgenden sind die Steuerbits des Steuerdoppelworts detailliert beschrieben. Eine übersichtliche Darstellung des Steuerworts siehe Kapitel ▶ 2.1.1.1 [14].

### Bit 0 – fast stop

| Flankenwechsel | Modulreaktion   |
|----------------|---|
| 0 -> 1         | keine Reaktion  |
| 1 -> 0         | Das Modul führt einen Schnellstopp durch, ▶ 3.2.5 [33]. |

### Bit 1 – stop

| Flankenwechsel | Modulreaktion  |
|----------------|--|
| 0 -> 1         | Das Modul führt ein kontrolliertes Anhalten durch, ▶ 3.2.4 [32]. |
| 1 -> 0         | keine Reaktion   |

### Bit 2 – acknowledge

| Flankenwechsel | Modulreaktion   |
|----------------|---|
| 0 -> 1         | Das Modul versucht alle anliegenden Warnungen und Fehler zu quittieren, ▶ 6.1 [75], ▶ 6.2 [77]. |
| 1 -> 0         | keine Reaktion  |

### Bit 3 – prepare for shutdown

| Flankenwechsel | Modulreaktion  |
|----------------|--|
| 0 -> 1         | Das Modul wird zum Abschalten vorbereitet, ▶ 3.1.2 [24]. |
| 1 -> 0         | keine Reaktion   |

### Bit 4 – softreset

| Flankenwechsel | Modulreaktion  |
|----------------|--|
| 0 -> 1         | Das Modul wird softwareseitig neu gestartet, ▶ 3.1.3 [26]. |
| 1 -> 0         | keine Reaktion   |

**Bit 5 – release for manual movement**

| Flankenwechsel | Modulreaktion  |
|----------------|--|
| 0 -> 1         | GPE wird deaktiviert, um ein Werkstück manuell zu entnehmen, ▶ 3.3.6 [46]. |
| 1 -> 0         | keine Reaktion   |

**Bit 6 – repeat command toggle**

| Flankenwechsel | Modulreaktion  |
|----------------|--|
| 0 -> 1         | Das Modul führt erneut den Steuerbefehl aus, dessen Bit noch ansteht |
| 1 -> 0         | Das Modul führt erneut den Steuerbefehl aus, dessen Bit noch ansteht |

Hinweis: Abhängig vom aktuellen Zustand des Moduls kann es zu Rückmeldungen kommen, dass Bewegungen nicht erneut ausgeführt werden können.

**Bit 7 – grip direction**

| Zustand | Modulreaktion                                    |
|---------|--|
| 0       | Bei einem Greifvorgang wird von außen gegriffen. |
| 1       | Bei einem Greifvorgang wird von innen gegriffen. |

**Bit 8 – jog mode negative**

| Flankenwechsel | Modulreaktion   |
|----------------|---|
| 0 -> 1         | Solange das Bit gesetzt ist, führt das Modul eine Bewegungsfahrt in negative Bewegungsrichtung aus, ▶ 3.2.1 [27]. |
| 1 -> 0         | keine Reaktion  |

**Bit 9 – jog mode positive**

| Flankenwechsel | Modulreaktion   |
|----------------|---|
| 0 -> 1         | Solange das Bit gesetzt ist, führt das Modul eine Bewegungsfahrt in positive Bewegungsrichtung aus, ▶ 3.2.1 [27]. |
| 1 -> 0         | keine Reaktion  |

**Bit 10 – reserved**

| Flankenwechsel | Modulreaktion  |
|----------------|----------------|
| 0 -> 1         | keine Reaktion |
| 1 -> 0         | keine Reaktion |

**Bit 11 – release workpiece**

| Flankenwechsel | Modulreaktion                                    |
|----------------|--|
| 0 -> 1         | Das Modul gibt ein Werkstück frei, ▶ 3.3.5 [44]. |
| 1 -> 0         | keine Reaktion                                   |

**Bit 12 – grip workpiece**

| Flankenwechsel | Modulreaktion   |
|----------------|---|
| 0 -> 1         | Das Modul führt ein Werkstück-Greifen durch, ▶ 3.3.1 [33] |
| 1 -> 0         | keine Reaktion  |

**Bit 13 – move to absolute position**

| Flankenwechsel | Modulreaktion   |
|----------------|---|
| 0 -> 1         | Das Modul führt eine Positionsfahrt auf eine absolute Position durch, ▶ 3.2.2 [28]. |
| 1 -> 0         | keine Reaktion  |

**Bit 14 – move to relative position**

| Flankenwechsel | Modulreaktion   |
|----------------|---|
| 0 -> 1         | Das Modul führt eine Positionsfahrt auf eine relative Position durch, ▶ 3.2.3 [30]. |
| 1 -> 0         | keine Reaktion  |

**Bit 15 – reserved**

| Flankenwechsel | Modulreaktion  |
|----------------|----------------|
| 0 -> 1         | keine Reaktion |
| 1 -> 0         | keine Reaktion |

**Bit 16 – grip workpiece at expected position**

| Flankenwechsel | Modulreaktion   |
|----------------|---|
| 0 -> 1         | Das Modul führt ein Werkstück-Greifen an erwarteter Position durch. |
| 1 -> 0         | keine Reaktion  |

**Bit 17 – 29 – reserved**

| Flankenwechsel | Modulreaktion  |
|----------------|----------------|
| 0 -> 1         | keine Reaktion |
| 1 -> 0         | keine Reaktion |

**Bit 30 – brake test**

| Flankenwechsel | Modulreaktion  |
|----------------|--|
| 0 -> 1         | Das Modul führt einen Bremsentest aus, ▶ 3.4.3 [48]. |
| 1 -> 0         | keine Reaktion                                       |

**Bit 31 – Activate grip force and position maintenance**

| Zustand | Modulreaktion  |
|---------|--|
| 0       | Greifkräfte und Positionen werden durch die Antriebsregelung gehalten. |
| 1       | Greifkräfte und Positionen werden durch die GPE gehalten.              |

**7.3 Statusdoppelwort**

Im Folgenden sind die Statusbits des Statusdoppelworts detailliert beschrieben. Eine übersichtliche Darstellung des Statusdoppelworts siehe Kapitel ▶ 2.1.1.2 [18].

**Bit 0 – ready for operation**

| Zustand | Modulrückmeldung                    |
|---------|-------------------------------------|
| 0       | Das Modul ist nicht betriebsbereit. |
| 1       | Das Modul ist betriebsbereit.       |

**Bit 1 – control authority fieldbus**

| Zustand | Modulrückmeldung                    |
|---------|-------------------------------------|
| 0       | Der Feldbus hat keine Steuerhoheit. |
| 1       | Der Feldbus besitzt Steuerhoheit.   |

**Bit 2 – ready for shutdown**

| Zustand | Modulrückmeldung                       |
|---------|--|
| 0       | Keine Information wird zurückgemeldet. |
| 1       | Das Modul ist bereit zum Abschalten.   |

**Bit 3 – not feasible**

| Zustand | Modulrückmeldung   |
|---------|--|
| 0       | Keine Information wird zurückgemeldet.                                     |
| 1       | Der an das Modul gesendete Steuerbefehl ist nicht durchführbar. ▶ 6.1 [76] |

**Bit 4 – command successfully processed**

| Zustand | Modulrückmeldung   |
|---------|--|
| 0       | Keine Information wird zurückgemeldet.   |
| 1       | Die folgenden an das Modul gesendeten Steuerbefehle wurden erfolgreich <i>abgearbeitet</i> : <ul style="list-style-type: none"> <li>• Bit 1 – stop</li> <li>• Bit 8 – jog mode negative</li> <li>• Bit 9 – jog mode positive</li> <li>• Bit 11 – release workpiece</li> <li>• Bit 12 – grip workpiece</li> <li>• Bit 13 – move to absolute position</li> <li>• Bit 14 – move to relative position</li> <li>• Bit 16 – grip workpiece at expected position</li> </ul> |

**Bit 5 – command received toggle**

| Zustandswechsel | Modulrückmeldung                                     |
|-----------------|--|
| 0 -> 1          | Das Modul bestätigt den Empfang eines Steuerbefehls. |
| 1 -> 0          | Das Modul bestätigt den Empfang eines Steuerbefehls. |

**Bit 6 – warning**

| Zustand | Modulrückmeldung           |
|---------|----------------------------|
| 0       | Es liegt keine Warnung an. |
| 1       | Eine Warnung liegt an.     |

**Bit 7 – error**

| Zustand | Modulrückmeldung         |
|---------|--------------------------|
| 0       | Es liegt kein Fehler an. |
| 1       | Ein Fehler liegt an.     |

**Bit 8 – released for manual movement**

| Zustand | Modulrückmeldung   |
|---------|--|
| 0       | Keine Information wird zurückgemeldet.                     |
| 1       | Modul ist zum manuellen Entnehmen eines Werkstücks bereit. |

**Bit 9 – software limit reached**

| Zustand | Modulrückmeldung                       |
|---------|--|
| 0       | Keine Information wird zurückgemeldet. |
| 1       | Ein Softwarelimit wurde überfahren.    |

**Bit 10 – reserved**

| Zustand | Modulrückmeldung                       |
|---------|--|
| 0       | Keine Information wird zurückgemeldet. |
| 1       | Keine Information wird zurückgemeldet. |

**Bit 11 – no workpiece detected**

| Zustand | Modulrückmeldung                       |
|---------|--|
| 0       | Keine Information wird zurückgemeldet. |
| 1       | Der Greifvorgang war erfolglos.        |

**Bit 12 – workpiece gripped**

| Zustand | Modulrückmeldung   |
|---------|--|
| 0       | Keine Information wird zurückgemeldet.   |
| 1       | Der vorangegangene Greifvorgang war erfolgreich bzw. das richtige Werkstück wurde gegriffen. |

**Bit 13 – position reached**

| Zustand | Modulrückmeldung                           |
|---------|--|
| 0       | Keine Information wird zurückgemeldet.     |
| 1       | Das Modul hat die Zielposition angefahren. |

**Bit 14 – workpiece pre-grip started**

| Zustand | Modulrückmeldung                         |
|---------|--|
| 0       | Keine Information wird zurückgemeldet.   |
| 1       | Das Modul hat das Nachgreifen gestartet. |

**Bit 15 – reserved**

| Zustand | Modulrückmeldung                       |
|---------|--|
| 0       | Keine Information wird zurückgemeldet. |
| 1       | Keine Information wird zurückgemeldet. |

**Bit 16 – workpiece lost**

| Zustand | Modulrückmeldung                         |
|---------|--|
| 0       | Keine Information wird zurückgemeldet.   |
| 1       | Das gegriffene Werkstück wurde verloren. |

**Bit 17 – wrong workpiece gripped**

| Zustand | Modulrückmeldung   |
|---------|--|
| 0       | Keine Information wird zurückgemeldet.   |
| 1       | Beim Werkstück-Greifen an erwarteter Position wurde das falsche Werkstück gegriffen. |

**Bit 18 – 30 – reserved**

| Zustand | Modulrückmeldung                       |
|---------|--|
| 0       | Keine Information wird zurückgemeldet. |
| 1       | Keine Information wird zurückgemeldet. |

**Bit 31 – Grip force and position maintenance activated**

| Zustand | Modulrückmeldung     |
|---------|----------------------|
| 0       | GPE ist nicht aktiv. |
| 1       | GPE ist aktiv.       |

**7.4 Zusatzcode bei anliegender Warnung WRN\_NOT\_FEASIBLE**

Falls die Warnung WRN\_NOT\_FEASIBLE ausgegeben wird, kann ein Zusatzcode ausgelesen werden. Im Folgenden sind die möglichen Ursachen für die Nichtausführbarkeit aufgelistet.

HEX 0x00

**NF\_NO\_REASON**

Die Warnung liegt nicht an.

HEX 0x01

**NF\_IOLINK\_FUNKTION\_NOT\_SUPPORTED**

Über den I0-Link Master wurde azyklisch eine kundenspezifische Funktion angestoßen, die vom Modul nicht unterstützt wird.

HEX 0x02

**NF\_IOLINK\_RESET\_CONDITION\_ONLY\_ALLOWED\_IN\_ERROR\_STATE**

Das Modul befand sich nicht im Fehlerzustand, als über den I0-Link Master eine der folgenden Funktion angestoßen wurde:

- Application Reset
- Restore Factory Settings
- Back-To-Box

HEX 0x03

**NF\_SHUTDOWN\_NOT\_FEASIBLE\_IN\_CURRENT\_STATE**

Aus dem aktuellen Zustand darf das Abschalten nicht angestoßen werden, ▶ 3.1.2 [ 24].

HEX 0x04

**NF\_RESET\_NOT\_FEASIBLE\_IN\_CURRENT\_STATE**

Aus dem aktuellen Zustand darf der Neustart nicht angestoßen werden, ▶ 3.1.3 [ 26].

HEX 0x05

**NF\_COMMAND\_NOT\_ALLOWED\_IN\_CURRENT\_STATE**

Aus dem aktuellen Zustand darf das Zurücksetzen auf Werkseinstellung nicht angestoßen werden, ▶ 3.4.6 [ 50].

|                 |   |
|-----------------|---|
| <b>HEX 0x06</b> | <p><b>NF_COMMAND_NOT_ALLOWED_IN_ERROR_STATE</b><br/>Das Modul befand sich im Fehlerzustand, als die Funktion angestoßen wurde.</p>  |
| <b>HEX 0x08</b> | <p><b>NF_SOFT_RESET_DISABLED_BY_PARAMETER</b><br/>Der Neustart ist durch den Parameter &lt;enable_softreset&gt; nicht freigeschaltet, ▶ 4.2 [ 66].</p>  |
| <b>HEX 0x09</b> | <p><b>NF_CANNOT_TRIGGER_COMMAND_WHILE_FAST_STOP_IS_ACTIVE</b><br/>Das Statusbit "fast stop" (Bit 0) war zurückgesetzt, als eine Funktion angestoßen wurde.</p>  |
| <b>HEX 0x0A</b> | <p><b>NF_MULTIPLE_COMMANDS_TRIGGERED_SIMULTANEOUSLY</b><br/>Mehrere Funktionen sollten gleichzeitig angestoßen werden.</p>  |
| <b>HEX 0x0C</b> | <p><b>NF_COMMAND_NOT_ALLOWED_DURING_BRAKE_TEST</b><br/>Das Modul führte einen Bremsentest aus, als eine unzulässige Funktion angestoßen wurde.</p>  |
| <b>HEX 0x0D</b> | <p><b>NF_RELEASE_BRAKE_ONLY_ALLOWED_IN_ERROR_STATE</b><br/>Das Modul befand sich nicht im Fehlerzustand, als die manuelle Werkstückentnahme angestoßen wurde, ▶ 3.3.6 [ 46].</p>  |
| <b>HEX 0x0E</b> | <p><b>NF_RELEASE_WORKPIECE_COMMAND_ONLY_ALLOWED_WHILE_GRIPPING</b><br/>Das Modul hat kein Werkstück gehalten, als das Werkstück-Freigeben angestoßen wurde, ▶ 3.3.5 [ 44].</p>  |
| <b>HEX 0x0F</b> | <p><b>NF_COMMAND_NOT_ALLOWED_WHILE_HOLDING_A_WORKPIECE</b><br/>Während das Modul ein Werkstück hält, wurde eine unzulässige Funktion angestoßen.</p>  |
| <b>HEX 0x10</b> | <p><b>NF_DESIRED_POSITION_OUT_OF_RANGE</b><br/>Die sich aus dem zyklisch übertragenen Positionswert ergebene Zielposition liegt außerhalb der Positionsgrenzen, ▶ 1.2.1 [ 8].</p>   |
| <b>HEX 0x11</b> | <p><b>NF_CURRENT_POSITION_ALREADY_INSIDE_WORKPIECE</b><br/>Das Werkstück-Greifen an erwarteter Position wurde mit einer ungültigen Kombination aus Werkstückposition und Greifrichtung angestoßen.<br/>Unzulässige Kombination sind:</p> <ul style="list-style-type: none"> <li>• Werkstückposition &gt; aktueller Position und Greifrichtung nach innen</li> <li>• Werkstückposition &lt; aktueller Position und Greifrichtung nach außen</li> </ul> |
| <b>HEX 0x12</b> | <p><b>NF_COMMAND_NOT_ALLOWED_DURING_GRIPPING_COMMAND</b><br/>Während das Modul eine Greifbewegung ausgeführt hat, wurde eine unzulässige Funktion angestoßen.</p>   |

|                 |  |
|-----------------|--|
| <b>HEX 0x13</b> | <b>NF_DESIRED_VELOCITY_OUT_OF_RANGE</b><br>Der zyklisch übertragene Geschwindigkeitswert ist außerhalb der zulässigen Geschwindigkeitsgrenzen, ▶ 4.2 [458] und ▶ 4.2 [458].                          |
| <b>HEX 0x15</b> | <b>NF_COMMAND_NOT_ALLOWED_DURING_MOVE_TO_POSITION</b><br>Während das Modul eine Positionierbewegung ausgeführt hat, wurde eine unzulässige Funktion angestoßen.                                      |
| <b>HEX 0x1D</b> | <b>NF_COMMAND_NOT_ALLOWED_DURING_RELEASE_BRAKE_COMMAND</b><br>Während das Modul in dem Zustand ist, in dem ein Werkstück händisch entnommen werden kann, wurde eine unzulässige Funktion angestoßen. |
| <b>HEX 0x1E</b> | <b>NF_COMMAND_NOT_ALLOWED_DURING_RELEASE_WORKPIECE_COMMAND</b><br>Während ein Werkstück freigegeben wurde, wurde eine unzulässige Funktion angestoßen.   |
| <b>HEX 0x22</b> | <b>NF_MINIMUM_POSITION_OUT_OF_RANGE</b><br>In den Parameter <min_pos> sollte ein Wert geschrieben werden, der außerhalb der zulässigen Grenzwerte liegt, ▶ 4.2 [457].                                |
| <b>HEX 0x23</b> | <b>NF_MINIMUM_POSITION_ABOVE_MAXIMUM_POSITION</b><br>In den Parameter <min_pos> sollte ein Wert geschrieben werden, der größer als der Wert des Parameters <max_pos> ist.                            |
| <b>HEX 0x24</b> | <b>NF_MAXIMUM_POSITION_OUT_OF_RANGE</b><br>In den Parameter <max_pos> sollte ein Wert geschrieben werden, der außerhalb der zulässigen Grenzwerte liegt, ▶ 4.2 [457].                                |
| <b>HEX 0x25</b> | <b>NF_MAXIMUM_POSITION_BELOW_MINIMUM_POSITION</b><br>In den Parameter <max_pos> sollte ein Wert geschrieben werden, der kleiner als der Wert des Parameters <min_pos> ist.                           |
| <b>HEX 0x26</b> | <b>NF_RELEASE_WORKPIECE_WOULD_VIOLATE_SOFTWARE_LIMIT</b><br>Die Zielposition beim Werkstück-Freigeben liegt außerhalb der Positionsgrenzen, ▶ 3.3.5 [444].   |
| <b>HEX 0x27</b> | <b>NF_MOVEMENT_INTO_WORKPIECE_NOT_ALLOWED</b><br>Die Zielposition beim Freigeben eines Werkstücks durch eine absolute oder relative Positionsfahrt befindet sich innerhalb des Werkstücks.           |
| <b>HEX 0x28</b> | <b>NF_GPE_FEATURE_NOT_AVAILABLE_ON_GRIPPER_WITHOUT_BRAKE</b><br>Die GPE sollte zum Halten eines Werkstücks oder einer Position verwendet werden, obwohl das Modul über keine GPE verfügt.            |

**HEX 0x29****NF\_DESIRED\_FORCE\_OUT\_OF\_RANGE**

Der zyklisch übertragene Greifkraftwert ist außerhalb der zulässigen Greifkraftgrenzen, ▶ 4.2 [📄 59] und ▶ 4.2 [📄 59].

**7.5 Software Copyright Hinweise****Modbus-RTU Stack**

FreeModbus Library: A portable Modbus implementation for Modbus ASCII/RTU.

Copyright (c) 2006-2018 Christian Walter <cwalter@embedded-solutions.at>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

**Texas Instruments  
F2837xS Support  
Library, F021 Flash API**

TI Release: F2837xS Support Library v3.12.00.00

Release Date: Fri Feb 12 19:06:50 IST 2021

Copyright:

Copyright (C) 2014-2021 Texas Instruments Incorporated - <http://www.ti.com/>

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Texas Instruments Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## TMG IOL Stack

Copyright (c) 2017, TMG Technologie und Engineering GmbH

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of TMG Technologie und Engineering GmbH nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF

MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## Loki AssocVector

The Loki Library

Copyright (c) 2001 by Andrei Alexandrescu

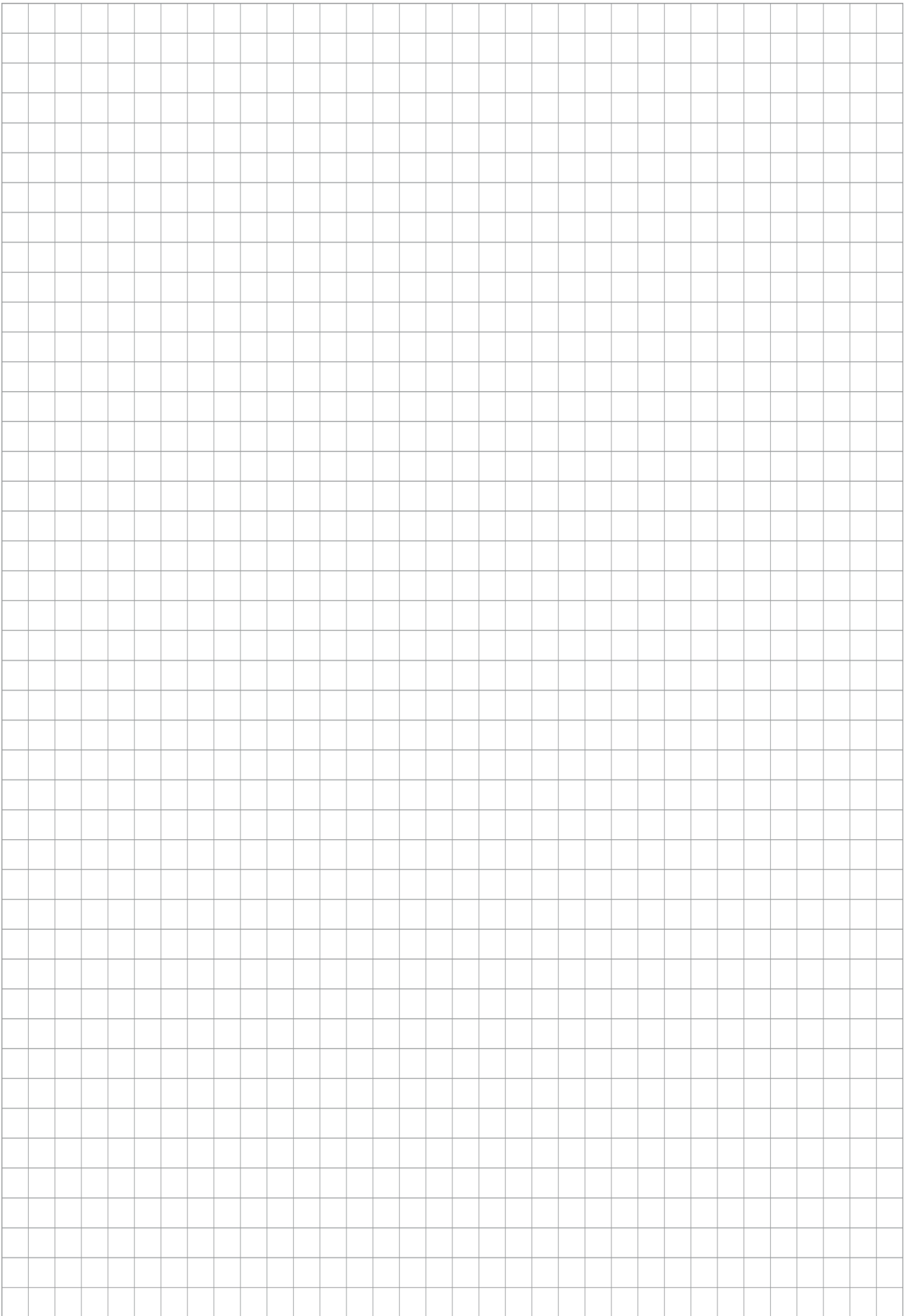
This code accompanies the book:

Alexandrescu, Andrei. "Modern C++ Design: Generic Programming and Design Patterns Applied". Copyright (c) 2001. Addison-Wesley.

Permission to use, copy, modify, distribute and sell this software for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

The author or Addison-Wesley Longman make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.







**SCHUNK SE & Co. KG**  
Spanntechnik | Greiftechnik | Automatisierungstechnik

Bahnhofstr. 106 - 134  
D-74348 Lauffen/Neckar  
Tel. +49-7133-103-0  
info@de.schunk.com  
schunk.com

Folgen Sie uns | *Follow us*



Wir drucken nachhaltig | *We print sustainable*