



## GRILINK-PLUGIN FÜR YASKAWA MOTOMAN

Version 1.0.0



## Inhalt

1	Einführung.....	2
1.1	Notation und Symbole .....	2
1.2	Bestimmungsgemäße Verwendung.....	2
1.3	Systemvoraussetzungen .....	2
1.4	Lizenzbestimmungen .....	3
2	Installation des GRIPLINK-Plugins .....	4
2.1	Benutzervariablen.....	5
2.2	Greifteilerkennung und -überwachung .....	5
2.3	Verhalten im Fehlerfall .....	6
3	Befehlsreferenz.....	8
3.1	Verbindung aufbauen – CONNECT .....	9
3.2	Antrieb des Greifmoduls einschalten – ENABLE .....	10
3.3	Greifzustand abfragen – GET STATE .....	11
3.4	Greifzustand von mehreren Greifmodulen – GET MSTATE .....	12
3.5	Antrieb des Greifmoduls deaktivieren – DISABLE.....	13
3.6	Greifmodul referenzieren – HOME.....	14
3.7	Mehrere Greifmodule referenzieren – MHOME .....	15
3.8	Werkstück greifen – GRIP .....	16
3.9	Gleichzeitiges Greifen von Werkstücken – MGRIP .....	17
3.10	Werkstück freigeben – RELEASE .....	18
3.11	Gleichzeitiges Freigeben von Werkstücken – MRELEASE .....	19
3.12	Aktuelle Fingerposition auslesen – GET POSITION .....	20
3.13	Fingerposition aller Greifmodule abfragen – GET MPOSITION .....	21
3.14	Greifkraftherhaltung steuern – PERMAGRIP .....	22
3.15	Gleichzeitiges Steuern der Greifkraftherhaltung – MPERMAGRIP .....	23
3.16	Ansteuerung der LED-Anzeige – GRIPLINK LED .....	24
Anhang A.	Greifzustand.....	25

# 1 Einführung

Mit der GRIPLINK-Technologie können servoelektrische und smart pneumatische Greifmodule von WEISS ROBOTICS über eine Netzwerkverbindung mit Robotersystemen führender Hersteller verbunden werden. Das GRIPLINK-Plugin für YASKAWA Motoman ist das steuerungsseitige Bindeglied und ermöglicht die einfache Einbindung der GRIPLINK-Technologie von WEISS ROBOTICS in Robotersysteme des Herstellers YASKAWA.



Diese Anleitung beschreibt die Funktionen des GRIPLINK-Plugins. Informationen über Montage, Inbetriebnahme und Betrieb des GRIPLINK Schnittstellenwandlers entnehmen Sie der Betriebsanleitung des jeweiligen Moduls. Diese finden Sie online unter [www.griplink.de/manuals](http://www.griplink.de/manuals)

## 1.1 Notation und Symbole

Zur besseren Übersicht werden in dieser Anleitung folgende Symbole verwendet:



Funktions- oder sicherheitsrelevanter Hinweis. Nichtbeachtung kann die Sicherheit von Personal und Anlage gefährden, das Gerät beschädigen oder die Funktion des Gerätes beeinträchtigen.



Zusatzinformation zum besseren Verständnis des beschriebenen Sachverhalts.



Verweis auf weiterführende Informationen.

## 1.2 Bestimmungsgemäße Verwendung

Die Software „GRIPLINK-Plugin“ ist zur Kommunikation zwischen dem GRIPLINK Schnittstellenwandler von WEISS ROBOTICS und einer Robotersteuerung bestimmt. Die Anforderungen der zutreffenden Richtlinien sowie die Installations- und Betriebshinweise in dieser Anleitung müssen beachtet und eingehalten werden. Eine andere oder darüberhinausgehende Verwendung gilt als nicht bestimmungsgemäß. Für hieraus resultierende Schäden haftet der Hersteller nicht.

## 1.3 Systemvoraussetzungen

Zum Betrieb wird eine der folgenden YASKAWA Robotersteuerungen benötigt:

- DX 100
- DX 200
- YRC 1000

Folgende Roboter-Optionen werden zum Betrieb der Software benötigt:

- MotoPlus Softwareoption.



Kontaktieren Sie Ihren YASKAWA Händler zum Bezug der Roboter-Optionen.



Die IP-Adresse des GRIPLINK Schnittstellenwandlers muss im selben Subnetz liegen wie die der Robotersteuerung. In der Anleitung des GRIPLINK Schnittstellenwandlers ist der genaue Vorgang beschrieben, wie Sie die IP-Adresse ändern.

## **1.4 Lizenzbestimmungen**

Das GRIPLINK-Plugin ist urheberrechtlich geschützt. Die jeweils gültigen Lizenzbestimmungen liegen dem Softwarepaket bei. Mit der Installation akzeptieren Sie diese Lizenzbestimmungen.

## 2 Installation des GRIPLINK-Plugins

Zum Betrieb des GRIPLINK-ET4 wird das GRIPLINK-Plugin auf der Robotersteuerung benötigt. Zur Installation des GRIPLINK-Plugins folgen Sie den Punkten 1 bis 4.



Stellen Sie sicher, dass Sie die aktuelle Version des GRIPLINK-Plugins verwenden. Die aktuelle Version kann unter [www.griplink.de/software](http://www.griplink.de/software) heruntergeladen werden.

1. Entpacken Sie das zuvor heruntergeladene ZIP-Archiv mit dem GRIPLINK-Plugin in das Stammverzeichnis eines USB-Sticks und stecken Sie diesen in den USB-Slot des Teach Pendants.
2. Kopieren Sie das GRIPLINK-Plugin (MotoPlus-Programm) „griplink\_xxxx.out“ vom USB-Stick auf die Robotersteuerung, wobei xxxx für die verwendete Steuerung steht:
  - Starten Sie die Steuerung im Maintenance-Mode, indem Sie die MENU Taste am Teach-Pendant beim Einschalten gedrückt halten
  - Wechseln Sie in den Management-Mode. Wählen Sie dafür im „Main Menu“ unter „SYSTEM“ die Option „SECURITY“ und wählen Sie mit der SELECT-Taste den „MANAGEMENT MODE“ aus.
  - Geben Sie das Passwort ein. Das default Passwort ist 16-mal die Neun (9999999999999999)
  - Laden Sie im Main Menu unter „MotoPlus APL.“ bei der Option „LOAD (USER APPLICATION)“ mit der Select-Taste das GRIPLINK-Plugin Softwarepaket.

Das GRIPLINK-Plugin darf nicht in einem Unterordner liegen.

3. Starten Sie die Steuerung neu.
4. Kopieren Sie die GRIPLINK-Jobs vom USB-Stick auf die Robotersteuerung.

## 2.1 Benutzervariablen

Das GRIPLINK-Plugin nutzt einen Block von acht Benutzervariablen des Typs „double“ zur Kommunikation zwischen GRIPLINK-Plugin und Roboterprogramm. Der Variablenblock wird standardmäßig ab Adresse 90 gespeichert. Der Zweck der einzelnen Variablen ist in der folgenden Tabelle aufgelistet.

Adress-Offset	Inhalt
0	Rückgabewert (RETVL0) der aufgerufenen Funktion
1	Rückgabewert (RETVL1) der aufgerufenen Funktion
2	Rückgabewert (RETVL2) der aufgerufenen Funktion
3	Rückgabewert (RETVL3) der aufgerufenen Funktion
4	Diagnosezähler
5	
6	Reserviert für die Kommunikation mit dem GRIPLINK-Plugin Softwarepaket
7	

Die Variablen 0 bis 3 liefern die Rückgabewerte des ausgeführten Befehls zurück.

Mit Variable 4 ist ein Diagnosezähler realisiert, welcher vom GRIPLINK-Plugin periodisch hochgezählt wird und über dessen Zustand Auskunft gibt.

Die Variablen 5 bis 7 sind für die interne Kommunikation zwischen den Bibliotheksbefehlen und dem GRIPLINK-Plugin reserviert und dürfen nicht verändert werden.

### ***Bereich der Benutzervariablen ändern***

Je nach Roboterprogramm kann es notwendig sein, dass der vom GRIPLINK-Plugin genutzte Speicherbereich verschoben werden muss. Hierzu gehen Sie wie folgt vor:

1. Wählen Sie im Main Menu das Feld „JOB“ aus und anschließend das Feld „SELECT JOB“.
2. Wählen Sie nun in der „JOB LIST“ den Job „GRIPLINK\_EXECUTE\_CMD“ aus und bestätigen Sie Ihre Auswahl mit der SELECT-Taste.
3. Legen Sie die Startadresse des Variablenblocks fest (Standardwert ist 90).
4. Starten Sie die Robotersteuerung neu (das GRIPLINK-Plugin übernimmt die Änderungen beim Neustart).

## 2.2 Greifteilerkennung und -überwachung

Greifmodule von WEISS ROBOTICS verfügen über eine integrierte Greifteilerkennung und -überwachung, die es ermöglicht, den gesamten Greifprozess ohne zusätzliche Sensoren permanent zu überwachen und so die Zuverlässigkeit des Handhabungsprozesses wesentlich zu erhöhen.

Das GRIPLINK-Plugin kommuniziert im Hintergrund kontinuierlich mit dem Greifmodul und fragt dessen Zustand permanent ab. Auf diese Weise kann im Roboterprogramm unmittelbar auf nicht vorhandene oder verloren gegangene Werkstücke reagiert werden.

### 2.2.1 Greifteilerkennung

Die Greifteilerkennung ermöglicht es, nach dem Greifen sofort zu erkennen, ob ein Werkstück korrekt gegriffen wurde oder nicht. Sind die Greiffinger innerhalb des vorgegebenen Positionsfensters blockiert und konnte die gewünschte Greifkraft aufgebaut werden, wechselt das Greifmodul in den Zustand HOLDING. Andernfalls wechselt das Greifmodul in den Zustand NO PART. Der Greifzustand nach Ausführung des Griiffs wird von der Funktion GRIP (vgl. Kapitel 3.8) als Rückgabewert zurück geliefert. Außerdem kann der Greifzustand jederzeit mittels der Funktion GET\_STATE (vgl. Kapitel 3.3) abgefragt werden.



Eine detaillierte Beschreibung der Greifzustände finden Sie in der Betriebsanleitung des jeweiligen Greifmoduls.

### 2.2.2 Greifteilüberwachung

Wurde ein Werkstück korrekt gegriffen, d.h. der Zustand HOLDING wurde erreicht, startet auf dem Greifmodul automatisch die integrierte Greifteilüberwachung. Wenn das Werkstück nun verloren geht oder entnommen wird, bevor der Befehl RELEASE (vgl. Kapitel 3.10) zur Freigabe des Werkstücks ausgeführt wurde, wechselt der Zustand des Greifmoduls nach PART LOST.

Das Verhalten des Roboterprogramms im Falle eines Werkstückverlusts kann beim Aktivieren des Greifmoduls mittels des Befehls ENABLE (vgl. Kapitel 3.2) festgelegt werden. Wird hier die Greifteilüberwachung aktiviert, wird im Falle eines Werkstückverlusts das Roboterprogramm mittels Fehler gestoppt und so der Roboter angehalten. Wird die Greifteilüberwachung dagegen nicht aktiviert, wird das GRIPLINK-Plugin den Werkstückverlust ignorieren und ohne Fehlermeldung fortfahren.

## 2.3 Verhalten im Fehlerfall

Tritt innerhalb des GRIPLINK-Plugins ein Fehler auf, so wird das laufende Roboterprogramm grundsätzlich mittels Fehler gestoppt. Dies führt in der Regel dazu, dass laufende Bewegungen des Roboters abgebrochen werden.

Geht die Verbindung zwischen Robotersteuerung und GRIPLINK oder zwischen GRIPLINK und Greifmodul unerwartet verloren oder tritt auf dem Greifmodul ein schwerer Fehler auf, so werden ebenfalls alle laufenden Roboterbewegungen automatisch angehalten.

Die folgende Tabelle gibt eine Übersicht über die Fehlerreaktion:

Fehler	Bedingung	Reaktion
Kommunikationsunterbrechung zwischen GRIPLINK Schnittstellenwandler und Roboter	-	Roboter wird angehalten
Kommunikationsunterbrechung zwischen Greifmodul und GRIPLINK Schnittstellenwandler	Greifer aktiviert über ENABLE-Befehl	Roboter wird angehalten
	Greifer nicht aktiviert	keine
Fehler am Greifmodul (Hardwarefehler, Übertemperatur)	Greifer aktiviert über ENABLE-Befehl	Roboter wird angehalten
	Greifer nicht aktiviert	Keine
Greifteilverlust	Greifteilüberwachung aktiv	Roboter wird angehalten
	Greifteilüberwachung nicht aktiv	keine



### 3 Befehlsreferenz

Das GRIPLINK-Plugin stellt dem Anwender eine Sammlung an Greifmodul-spezifischen Funktionen bereit. Die Befehle sind als Roboter-Jobs realisiert, welche ihre Parameter als Übergabewerte erhalten und über einen gemeinsamen Registerbereich mit einem MotoPlus-Programm im Hintergrund kommunizieren. Die Rückgabewerte der Befehle werden in den Benutzervariablen gespeichert (siehe Abschnitt 2.1). Zur Ausführung eines Befehls muss der entsprechende Roboter-Job aufgerufen werden. Die Jobs stehen sowohl als Einzel- wie auch als Mehrfachbefehle zur Verfügung.

#### ***Mehrfachbefehle***

Bei den Mehrfachbefehlen (Präfix M) können mehrere Greifmodule gleichzeitig parallel angesprochen werden. Diese Befehle eignen sich insbesondere für die Handhabung großer oder biegeschlaffer Werkstücke mit mehreren Greifmodulen.

***Der prinzipielle Programmablauf mit dem GRIPLINK-Plugin ist stets wie folgt:***

1. Verbindung herstellen mit GRIPLINK\_CONNECT
2. Greifmodul und Verbindungsüberwachung aktivieren mit GRIPLINK\_ENABLE
3. Bei Servogreifmodulen ohne Absolutgeber: Greifmodul referenzieren mit GRIPLINK\_HOME/MHOME
4. Greifen/Freigeben mit GRIPLINK\_GRIP bzw. GRIPLINK\_RELEASE

Im Folgenden sind die verfügbaren Befehle des GRIPLINK-Plugins beschrieben.

### 3.1 Verbindung aufbauen – CONNECT

Dieser Befehl stellt die Verbindung zwischen GRIPLINK Schnittstellenwandler und der Robotersteuerung her. Als Übergabeparameter wird die IP-Adresse des GRIPLINK Schnittstellenwandlers übergeben. Der Befehl wartet, bis die Verbindung hergestellt ist. Kann der GRIPLINK Schnittstellenwandler unter der angegebenen Adresse nicht erreicht werden, wird ein User Alarm ausgelöst und die Programmausführung angehalten.

Wenn GRIPLINK-ET4-Befehle vor einem GRIPLINK\_CONNECT ausgeführt werden, löst dies einen User Alarm aus.



Die IP-Adresse des GRIPLINK Schnittstellenwandlers kann über die GRIPLINK-ET4 Website geändert werden.

#### **Syntax**

```
CALL GRIPLINK_CONNECT(< IPADDR.0 >, < IPADDR.1 >, < IPADDR.2 >, < IPADDR.3 >, < RES >)
```

#### **Parameter**

< IPADDR.0 >	Erstes Byte der IP-Adresse
< IPADDR.1 >	Zweites Byte der IP-Adresse
< IPADDR.2 >	Drittes Byte der IP-Adresse
< IPADDR.3 >	Viertes Byte der IP-Adresse
< Res >	Reservierter Befehl, setze auf 0

#### **Rückgabewerte**

RETVAL 0 bis 3      nicht verwendet (0)

#### **Beispiel**

Verbindung zwischen Roboter und dem GRIPLINK mit der IP-Adresse 192.168.1.40 herstellen:

```
CALL GRIPLINK_CONNECT(192,168,1,40,0)
```

### 3.2 Antrieb des Greifmoduls einschalten – ENABLE

Dieser Befehl aktiviert das Greifmodul und die Verbindungsüberwachung. Wird die Verbindung zum Greifmodul getrennt (z.B. durch einen Kabelbruch), löst dies einen User Alarm aus und das Roboterprogramm wird gestoppt.

#### Greifteilüberwachung

Über einen Parameter kann die Greifteilüberwachung ein-/ bzw. ausgeschaltet werden. Ist die Greifteilüberwachung aktiv und verliert das Greifmodul ein zuvor aufgenommenes Werkstück, löst dies einen User Alarm aus und das Roboterprogramm wird gestoppt.

GRIPLINK\_ENABLE muss nach GRIPLINK\_CONNECT für alle Greifmodule ausgeführt werden.

Wenn GRIPLINK-Befehle vor einem GRIPLINK\_ENABLE ausgeführt werden, löst dies einen User Alarm.

#### **Syntax**

```
CALL GRIPLINK_ENABLE(< GRIPPER_ID >,< PARTMON >)
```

#### **Parameter**

< GRIPPER_ID >	Index des Greifmoduls (0 bis 3)
< PARTMON >	Greifteilüberwachung: 1 = an, 0 = aus

#### **Rückgabewerte**

RETVAL 0	aktueller Greifzustand (siehe Anhang AAnhang A <b>Fehler! Verweisquelle konnte nicht gefunden werden.</b> )
RETVAL 1 bis 3	nicht verwendet (0)Anhang A

#### **Beispiel**

Aktiviere Antrieb und Greifteilüberwachung des Greifmodul an Port 0:

```
CALL GRIPLINK_ENABLE (0, 1)
```

### 3.3 Greifzustand abfragen – GET STATE

Dieser Befehl gibt den Greifzustand des ausgewählten Greifmoduls zurück.

#### **Syntax**

```
CALL GRIPLINK_GET_STATE(< GRIPPER_ID >)
```

#### **Parameter**

< GRIPPER\_ID >      Index des Greifmoduls (0 bis 3)

#### **Rückgabewerte**

RETVAL 0            aktueller Greifzustand (siehe Anhang A)  
RETVAL 1 bis 3      nicht verwendet (0)

#### **Beispiel**

Warte, bis der Greifzustand von Greifmodul an Port 2 „HOLDING“ (4) ist:

```
*WAIT  
TIMER T=0.01  
CALL GRIPLINK_GET_STATE(2)  
IF(D090<>4) THEN  
    JUMP *WAIT  
ENDIF
```

### 3.4 Greifzustand von mehreren Greifmodulen – GET MSTATE

Dieser Befehl gibt den Greifzustand aller Greifmodule zurück.

#### **Syntax**

```
CALL GRIPLINK_GET_MSTATE
```

#### **Parameter**

Es werden keine Parameter verwendet.

#### **Rückgabewerte**

RETVAL 0	aktueller Greifzustand Greifmodul an Port 0 (siehe Anhang A)
RETVAL 1	aktueller Greifzustand Greifmodul an Port 1 (siehe Anhang A)
RETVAL 2	aktueller Greifzustand Greifmodul an Port 2 (siehe Anhang A)
RETVAL 3	aktueller Greifzustand Greifmodul an Port 3 (siehe Anhang A)

#### **Beispiel**

Warte, bis alle Greifmodule im Zustand „RELEASED“ (3) sind:

```
*WAIT  
TIMER T=0.01  
CALL GRIPLINK_GET_MSTATE  
IF (D090<>3 OR 091<>3 OR D092<>3 OR D093<>3) THEN  
    JUMP *WAIT  
ENDIF
```

### 3.5 Antrieb des Greifmoduls deaktivieren – DISABLE

Dieser Befehl kann beispielsweise zum Werkzeugwechsel genutzt werden. Er deaktiviert den Antrieb des ausgewählten Greifmoduls. Ist der Antrieb deaktiviert, ist auch die Verbindungsüberwachung ausgeschaltet und eine Verbindungsunterbrechung zwischen GRIPLINK-ET4 und Greifmodul führt nicht zu einem User Alarm. Die Verbindungsüberwachung kann über GRIPLINK\_ENABLE wieder aktiviert werden.

#### **Syntax**

```
CALL GRIPLINK_DISABLE(< GRIPPER_ID >)
```

#### **Parameter**

< GRIPPER\_ID >      Index des Greifmoduls (0 bis 3)

#### **Rückgabewerte**

RETVAL 0            aktueller Greifzustand (siehe Anhang A)  
RETVAL 1 bis 3      nicht verwendet (0)

#### **Beispiel**

Greifmodul an Port 1 wechseln:

```
'Connect to GRIPLINK
CALL GRIPLINK_CONNECT(192,168,1,40,0)
'Activate gripper 1
CALL GRIPLINK_ENABLE(1,1)
'DO something
'...
'Prepare tool change:
'Disable gripper 1:
CALL GRIPLINK_DISABLE(1)
'Operate the tool changer here
'Activate new gripper:
CALL GRIPLINK_ENABLE(1,1)
```

### 3.6 Greifmodul referenzieren – HOME

Referenziert den ausgewählten Servogreifer. Der Befehl führt eine Referenzfahrt des Greifmoduls aus und wartet, bis diese abgeschlossen ist. Nachdem der GRIPLINK\_HOME Befehl ausgeführt wurde, sind die Finger des Greifmoduls kraftlos und müssen mit einem GRIPLINK\_GRIP/MGRIP oder GRIPLINK\_RELEASE/MRELEASE in eine definierte Position verfahren werden.



Die Referenzfahrt kann über die Weboberfläche des GRIPLINK Schnittstellenwandlers konfiguriert werden.

#### **Syntax**

```
CALL GRIPLINK_HOME(< GRIPPER_ID >)
```

#### **Parameter**

< GRIPPER\_ID >      Index des Greifmoduls (0 bis 3)

#### **Rückgabewerte**

RETVAL 0            aktueller Greifzustand (siehe Anhang A)  
RETVAL 1 bis 3      nicht verwendet (0)

#### **Beispiel**

Referenziere das Greifmodul an Port 2:

```
CALL GRIPLINK_HOME (2)
```

### 3.7 Mehrere Greifmodule referenzieren – MHOME

Referenziert die ausgewählten Servogreifer. Der Befehl führt bei allen ausgewählten Greifmodulen eine Referenzfahrt aus und wartet, bis diese bei allen abgeschlossen ist. Nachdem der GRIPLINK\_MHOME Befehl ausgeführt wurde, sind die Finger des Greifmoduls kraftlos und müssen mit einem GRIPLINK\_GRIP oder GRIPLINK\_RELEASE in eine definierte Position verfahren werden.



Die Referenzfahrt kann über die Weboberfläche des GRIPLINK Schnittstellenwandlers konfiguriert werden.

#### **Syntax**

```
CALL GRIPLINK_MHOME(< GRIPPERS >)
```

#### **Parameter**

< GRIPPERS >      Ausgewählte Greifmodule als Bitvektor:  
Bit 0:      1 = Greifmodul an Port 0 selektiert, 0 = nicht selektiert  
Bit 1:      1 = Greifmodul an Port 1 selektiert, 0 = nicht selektiert  
Bit 2:      1 = Greifmodul an Port 2 selektiert, 0 = nicht selektiert  
Bit 3:      1 = Greifmodul an Port 3 selektiert, 0 = nicht selektiert  
Bit 31...4:    reserviert (auf 0 setzen)

#### **Rückgabewerte**

RETVAL 0      aktueller Greifzustand von Greifmodul an Port 0 (siehe Anhang **AFehler! V erweisquelle konnte nicht gefunden werden.**)  
RETVAL 1      aktueller Greifzustand von Greifmodul an Port 1 (siehe Anhang **AFehler! V erweisquelle konnte nicht gefunden werden.**)  
RETVAL 2      aktueller Greifzustand von Greifmodul an Port 2 (siehe Anhang **AFehler! V erweisquelle konnte nicht gefunden werden.**)  
RETVAL 3      aktueller Greifzustand von Greifmodul an Port 3 (siehe Anhang **AFehler! V erweisquelle konnte nicht gefunden werden.**)

#### **Beispiele**

Referenziere gleichzeitig die Greifmodule 0 und 2:

```
CALL GRIPLINK_MHOME (5)
```



### 3.8 Werkstück greifen – GRIP

Greift mit dem ausgewählten Greifmodul und dem ausgewählten Griff ein Werkstück. Der Befehl wartet, bis der Greifzustand entweder auf „HOLDING“ oder auf „NO PART“ wechselt.



Die Greifparameter können über die Weboberfläche des GRIPLINK Schnittstellenwandlers konfiguriert werden.

#### **Syntax**

```
CALL GRIPLINK_GRIP(< GRIPPER_ID >, < PRESET >)
```

#### **Parameter**

< GRIPPER_ID >	Index des Greifmoduls (0 bis 3)
< PRESET >	Ausgewählter Griff (0 bis 3 bzw. 0 bis 7 bei CRG Greifmodulen)

#### **Rückgabewerte**

RETVAL 0	aktueller Greifzustand (siehe Anhang A)
RETVAL 1 bis 3	nicht verwendet (0)

#### **Beispiele**

Greifmodul an Port 0 soll mit Griff 2 greifen. Wurde kein Werkstück gefunden, soll der Greifer wieder öffnen und es erneut versuchen:

```
*LOOP
CALL GRIPLINK_GRIP(0,2)
IF(D090<>5) THEN
'No part, open and try again:
CALL GRIPLINK_RELEASE(0,2)
JUMP *LOOP
ENDIF
'Part gripped!
```

### 3.9 Gleichzeitiges Greifen von Werkstücken – MGRIP

Dieser Befehl führt mit den ausgewählten Greifmodulen einen Griff aus. Der Befehl wartet, bis alle Greifmodule jeweils einen der Zustände „HOLDING“ oder „NO PART“ erreicht haben.



Die Greifparameter können über die Weboberfläche des GRIPLINK Schnittstellenwandlers konfiguriert werden.

#### **Syntax**

```
CALL GRIPLINK_MGRIP(< GRIPPERS >, < PRESET >)
```

#### **Parameter**

< GRIPPERS >	Ausgewählte Greifmodule als Bitvektor: Bit 0: 1 = Greifmodul an Port 0 selektiert, 0 = nicht selektiert Bit 1: 1 = Greifmodul an Port 1 selektiert, 0 = nicht selektiert Bit 2: 1 = Greifmodul an Port 2 selektiert, 0 = nicht selektiert Bit 3: 1 = Greifmodul an Port 3 selektiert, 0 = nicht selektiert Bit 31...4: reserviert (auf 0 setzen)
< PRESET >	Ausgewählter Griff (0 bis 3 bzw. 0 bis 7 bei CRG Greifmodulen)

#### **Rückgabewerte**

RETVAL 0	aktueller Greifzustand von Greifmodul an Port 0 (siehe Anhang A)
RETVAL 1	aktueller Greifzustand von Greifmodul an Port 1 (siehe Anhang A)
RETVAL 2	aktueller Greifzustand von Greifmodul an Port 2 (siehe Anhang A)
RETVAL 3	aktueller Greifzustand von Greifmodul an Port 3 (siehe Anhang A)

#### **Beispiele**

Greifmodule an Port 1 und 2 greifen Werkstück mit Griff 2:

```
CALL GRIPLINK_MGRIP(6,2)
```

### 3.10 Werkstück freigeben – RELEASE

Gibt das mit dem ausgewählten Greifmodul gegriffene Werkstück wieder frei. Der Befehl wartet, bis das Werkstück freigegeben wurde.



Die Greifparameter können über die Weboberfläche des GRIPLINK Schnittstellenwandlers konfiguriert werden.

#### **Syntax**

```
CALL GRIPLINK_RELEASE(< GRIPPER_ID >, < PRESET >)
```

#### **Parameter**

< GRIPPER_ID >	Index des Greifmoduls (0 bis 3)
< PRESET >	Ausgewählter Griff (0 bis 3 bzw. 0 bis 7 bei CRG Greifmodulen)

#### **Rückgabewerte**

RETVAL 0	aktueller Greifzustand (siehe Anhang A)
RETVAL 1 bis 3	nicht verwendet (0)

#### **Beispiele**

Gebe das mit Greifmodul an Port 0 und Griff 2 gegriffene Werkstück wieder frei:

```
CALL GRIPLINK_RELEASE (0, 2)
```

### 3.11 Gleichzeitiges Freigeben von Werkstücken – MRELEASE

Gibt das mit den ausgewählten Greifmodulen gegriffene Werkstück gleichzeitig wieder frei. Der Befehl wartet, bis alle Greifmodule jeweils den Zustand „RELEASED“ erreicht haben.

#### **Syntax**

```
CALL GRIPLINK_MRELEASE(< GRIPPERS >, < PRESET >)
```

#### **Parameter**

< GRIPPERS >	Ausgewählte Greifmodule als Bitvektor: Bit 0: 1 = Greifmodul an Port 0 selektiert, 0 = nicht selektiert Bit 1: 1 = Greifmodul an Port 1 selektiert, 0 = nicht selektiert Bit 2: 1 = Greifmodul an Port 2 selektiert, 0 = nicht selektiert Bit 3: 1 = Greifmodul an Port 3 selektiert, 0 = nicht selektiert Bit 31...4: reserviert (auf 0 setzen)
< PRESET >	Ausgewählter Griff (0 bis 3 bzw. 0 bis 7 bei CRG Greifmodulen)

#### **Rückgabewerte**

RETVAL 0	aktueller Greifzustand Greifmodul an Port 0 (siehe Anhang A)
RETVAL 1	aktueller Greifzustand Greifmodul an Port 1 (siehe Anhang A)
RETVAL 2	aktueller Greifzustand Greifmodul an Port 2 (siehe Anhang A)
RETVAL 3	aktueller Greifzustand Greifmodul an Port 3 (siehe Anhang A)

#### **Beispiele**

Greifmodul an Port 1,2 und 3 geben Werkstück mit Griff 3 frei:

```
CALL GRIPLINK_MRELEASE (14, 3)
```

### 3.12 Aktuelle Fingerposition auslesen – GET POSITION

Dieser Befehl gibt die aktuelle Fingerposition des ausgewählten Greifmoduls zurück.

#### **Syntax**

```
CALL GRIPLINK_GET_POSITION(< GRIPPER_ID >)
```

#### **Parameter**

< GRIPPER\_ID >     Index des Greifmoduls (0 bis 3)

#### **Rückgabewerte**

RETVAL 0             Fingerposition in 0,01 mm Schritten  
RETVAL 1 bis 3       nicht verwendet (0)

#### **Beispiel**

Führe das Unterprogramm „PROG“ aus, wenn die Fingerposition des Greifmoduls an Port 2 größer als 10,5 mm ist:

```
CALL GRIPLINK_GET_POSITION(2)  
IF (D090>1050) THEN  
    CALL PROG  
ENDIF
```

### 3.13 Fingerposition aller Greifmodule abfragen – GET MPOSITION

Dieser Befehl gibt die Fingerposition aller Greifmodule zurück.

#### **Syntax**

```
CALL GRIPLINK_GET_MPOSITION
```

#### **Parameter**

Es werden keine Parameter verwendet.

#### **Rückgabewerte**

RETVAL 0	Position von Greifmodul an Port 0 in 0,01 mm Schritten
RETVAL 1	Position von Greifmodul an Port 1 in 0,01 mm Schritten
RETVAL 2	Position von Greifmodul an Port 2 in 0,01 mm Schritten
RETVAL 3	Position von Greifmodul an Port 3 in 0,01 mm Schritten

#### **Beispiel**

Führe das Unterprogramm „PROG“ aus, wenn die Fingerposition des Greifmoduls an Port 0 größer als 4,2 mm und die Fingerposition des Greifmoduls an Port 1 kleiner als 11,9 mm ist:

```
CALL GRIPLINK_GET_MPOSITION
IF (D090>420 AND D091<1190) THEN
  CALL PROG
ENDIF
```

### 3.14 Greifkraftherhaltung steuern – PERMAGRIP

Die von Weiss Robotics entwickelte innovative Greifkraftsicherung erhält die Greifkraft am Werkstück, auch wenn die Stromzufuhr zum Greifmodul unerwartet unterbrochen wird. Dank der integrierten Absolutsensorik kann die Produktion bei Wiederherstellung der Stromversorgung auch ohne Referenzieren gleich weitergehen. Des Weiteren ermöglicht PERMAGRIP dauerhaftes Greifen, ohne dass das Greifmodul dabei heiß wird.

Dieser Befehl aktiviert oder deaktiviert die PERMAGRIP-Greifkraftherhaltung bei dem ausgewählten Greifmodul.



PERMAGRIP ist nicht bei allen Greifmodulen verfügbar.

#### **Syntax**

```
CALL GRIPLINK_PERMAGRIP(< GRIPPER_ID >, < PERMAGRIP >)
```

#### **Parameter**

< GRIPPER\_ID >      Index des Greifmoduls (0 bis 3)  
< PERMAGRIP >      PERMAGRIP: 1 = ein, 0 = aus

#### **Rückgabewerte**

RETVAL 0 bis 3      nicht verwendet (0)

#### **Beispiel**

Aktiviert, PERMAGRIP bei Greifmodul an Port 2, wenn der Greifstatus „HOLDING“ ist:

```
CALL GRIPLINK_GRIP(2,0)  
IF(D090=5) THEN  
    CALL GRIPLINK_PERMAGRIP(2,1)  
ENDIF
```

### 3.15 Gleichzeitiges Steuern der Greifkraftherhaltung – MPERMAGRIP

Die von Weiss Robotics entwickelte innovative Greifkraftsicherung erhält die Greifkraft am Werkstück, auch wenn die Stromzufuhr zum Greifmodul unerwartet unterbrochen wird. Dank der integrierten Absolutsensorik kann die Produktion bei Wiederherstellung der Stromversorgung auch ohne Referenzieren gleich weitergehen. Des Weiteren ermöglicht PERMAGRIP® dauerhaftes Greifen, ohne dass das Greifmodul dabei heiß wird.

Dieser Befehl aktiviert oder deaktiviert PERMAGRIP bei den ausgewählten Greifmodulen.



PERMAGRIP ist nicht bei allen Greifmodulen verfügbar.

#### **Syntax**

```
CALL GRIPLINK_MPERMAGRIP(< GRIPPERS >, < PERMAGRIP >)
```

#### **Parameter**

< GRIPPERS >	Ausgewählte Greifmodule als Bitvektor: Bit 0: 1 = Greifmodul an Port 0 selektiert, 0 = nicht selektiert Bit 1: 1 = Greifmodul an Port 1 selektiert, 0 = nicht selektiert Bit 2: 1 = Greifmodul an Port 2 selektiert, 0 = nicht selektiert Bit 3: 1 = Greifmodul an Port 3 selektiert, 0 = nicht selektiert Bit 31...4: reserviert (auf 0 setzen)
< PERMAGRIP >	PERMAGRIP: 1 = ein, 0 = aus

#### **Rückgabewerte**

RETVAL 0 bis 3	nicht verwendet (0)
----------------	---------------------

#### **Beispiele**

Aktiviert PERMAGRIP gleichzeitig für die Greifmodule an Port 0 und 1:

```
CALL GRIPLINK_MPERMAGRIP (3, 1)
```



### 3.16 Ansteuerung der LED-Anzeige – GRIPLINK LED

Dieser Befehl ändert die Farbe und das Muster des Leuchtrings eines selektierten CRG Greifmoduls. Diese Funktion ist ausschließlich für Greifmodule der CRG-Serie verfügbar.



Leuchtmuster können über die Weboberfläche des GRIPLINK Schnittstellenwandlers konfiguriert werden.

#### **Syntax**

```
CALL GRIPLINK_LED(< GRIPPER_ID >, < PATTERN >)
```

#### **Parameter**

< GRIPPER\_ID >      Index des Greifmoduls (0 bis 3)  
< PATTERN >        Index des vordefinierten Leuchtmusters

#### **Rückgabewerte**

RETVAL 0 bis 3      nicht verwendet (0)

#### **Beispiel**

Greife mit dem Greifmodul an Port 3 und ändere die Farbe des Leuchtrings auf das Leuchtmuster 0, wenn die Fingerposition danach größer gleich 8,1 mm und auf Leuchtmusters 1, wenn kleiner:

```
CALL GRIPLINK_GRIP(3,0)  
CALL GRIPLINK_GET_POSITION(3)  
IF(D090>=810) THEN  
    CALL GRIPLINK_LED(3,0)  
ELSE  
    CALL GRIPLINK_LED(3,1)  
ENDIF
```

## Anhang A. Greifzustand

Greifzustand	Code	Beschreibung
NOT CONNECTED	0	Greifmodul nicht verbunden
NOT INITIALIZED	1	Greifmodul nicht initialisiert
IDLE	2	Betriebsbereit, nicht aktiv
RELEASED	3	Werkstück freigegeben
NO PART	4	Kein Werkstück gefunden
HOLDING	5	Werkstück wird gehalten
PART LOST	6	Werkstück verloren
FAULT	7	Fehlerzustand

© 2020 WEISS ROBOTICS GmbH & Co. KG. Alle Rechte vorbehalten.

GRIPLINK und PERMAGRIP sind eingetragene Marken der WEISS ROBOTICS GmbH & Co. KG. Alle weiteren Marken sind Eigentum ihrer jeweiligen Inhaber.

Die in diesem Dokument angegebenen technischen Daten können zum Zwecke der Produktverbesserung ohne Vorankündigung geändert werden. Warenzeichen sind Eigentum des jeweiligen Eigentümers. Unsere Produkte sind nicht für den Einsatz in lebenserhaltenden Systemen oder für Systeme, bei denen ein Fehlverhalten zu Personenschäden führen könnte, vorgesehen.