



## GRILINK-PLUGIN FÜR ABB

Version 1.9.9

Februar 2025



# Inhalt

1	Einführung.....	3
1.1	Notation und Symbole .....	3
1.2	Bestimmungsgemäße Verwendung.....	3
1.3	Systemanforderungen .....	4
1.4	Lizenzbestimmungen .....	4
1.5	Demo-Programme .....	4
2	Installation .....	5
2.1	Installation des Plugins .....	5
2.2	Netzwerk-Einstellungen.....	12
2.3	Firewall-Einstellungen.....	13
3	Funktionsumfang .....	14
3.1	RAPID .....	14
3.2	Wizard .....	15
3.3	Globale Variablen für Rückgabewerte .....	16
3.4	Gerätezustände.....	17
3.5	Fehlerbehandlung.....	18
3.6	Genereller Programmablauf .....	19
3.7	Demo-Programme .....	20
4	Befehlsreferenz.....	21
4.1	GRIPLINK_Connect: Verbindung aufbauen.....	22
4.2	GRIPLINK_Disconnect: Verbindung trennen.....	23
4.3	GRIPLINK_AssertDevice: Angeschlossenes Gerät prüfen .....	24
4.4	GRIPLINK_DeviceTag: Geräte-Tag abfragen .....	25
4.5	GRIPLINK_SetDeviceTag: Geräte-Tag setzen .....	26
4.6	GRIPLINK_Enable: Gerät aktivieren .....	27
4.7	GRIPLINK_Disable: Gerät deaktivieren .....	28
4.8	GRIPLINK_Home: Gerät referenzieren.....	29
4.9	GRIPLINK_Grip: Werkstück greifen .....	30
4.10	GRIPLINK_Release: Werkstück freigeben .....	32
4.11	GRIPLINK_FlexGrip: Werkstück greifen mit flexiblen Parametern .....	34
4.12	GRIPLINK_FlexRelease: Werkstück freigeben mit flexiblen Parametern.....	36

4.13	GRIPLINK_LED: LED-Visualisierung wählen.....	38
4.14	GRIPLINK_Clamp: Mechanische Greifkraftherhaltung einstellen.....	39
4.15	GRIPLINK_WaitForStateTransition: Auf Zustandsänderung warten.....	40
4.16	GRIPLINK_SetValue: Gerätewert setzen.....	42
4.17	GRIPLINK_WaitValue: Auf Gerätewert warten.....	43
4.18	GRIPLINK_DeviceState: Gerätezustand abfragen.....	45
4.19	GRIPLINK_Value: Gerätewert abfragen.....	47
4.20	GRIPLINK_SetGripConfig: Greifparameter konfigurieren.....	49

# 1 Einführung

Mit der GRIPLINK-Technologie können IO-Link-kompatible Automationskomponenten über eine TCP/IP-Netzwerkverbindung mit Robotersystemen führender Hersteller verbunden werden. Das GRIPLINK-Plugin für ABB ist das steuerungsseitige Bindeglied und ermöglicht die einfache Einbindung der GRIPLINK-Technologie von WEISS ROBOTICS in Robotersysteme des Herstellers ABB.



Diese Anleitung beschreibt den Inhalt und die Funktionen des GRIPLINK-Plugins für Roboter von ABB.

Informationen über Montage, Inbetriebnahme und Betrieb der GRIPLINK Hardware finden Sie in der jeweiligen Betriebsanleitung auf <https://weiss-robotics.com>.

## 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.



Spezifische Informationen zu Greifmodulen von WEISS ROBOTICS.

Das GRIPLINK-Plugin ist kompatibel mit allen Geräten, die das GRIPLINK-Protokoll über TCP/IP unterstützen. Dies umfasst den GRIPLINK-ET4 sowie die Greifmodule der WPG-Serie. Für bessere Lesbarkeit werden diese Geräte unter dem Begriff *GRIPLINK-Controller* zusammengefasst.

## 1.2 Bestimmungsgemäße Verwendung

Die Software *GRIPLINK-Plugin* ist zur Kommunikation zwischen einem GRIPLINK-Controller 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 Systemanforderungen**

Das GRIPLINK-Plugin ist kompatibel zum GRIPLINK-ET4 ab Firmware-Version 5.1.0 und WPG-Greifmodulen ab Firmware-Version 2.2.0.

Zum Betrieb wird eine ABB-Robotersteuerung mit ABB RobotWare (Version 7.1 oder neuer) benötigt. Zur einfacheren Einrichtung und Programmierung wird außerdem die Programmierumgebung ABB RobotStudio® (Version 2022.3 oder neuer) empfohlen.



Kontaktieren Sie ABB oder Ihren ABB-Distributor, um diese Produkte zu erwerben.

### **1.4 Lizenzbestimmungen**

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

### **1.5 Demo-Programme**

Die im Softwarepaket enthaltenen Demo-Programme zeigen beispielhaft die Anwendung des Plugins. Sie sind ausschließlich zu Demonstrations- und Testzwecken vorgesehen!

## 2 Installation

Um das GRIPLINK-Plugin verwenden zu können, ist neben dem ordnungsgemäßen Anschluss der Hardware die Installation des RobotWare Add-Ins auf der Robotersteuerung sowie deren korrekte Konfiguration erforderlich. Die notwendigen Schritte werden im folgenden Kapitel erläutert.



Detaillierte Informationen zur Montage der Greifmodule und dem Anschluss des GRIPLINK-Controllers finden Sie in den zugehörigen Bedienungsanleitungen.

### 2.1 Installation des Plugins

Das GRIPLINK-Plugin kann entweder über einen PC mit ABB RobotStudio® oder mit einem USB-Stick direkt am FlexPendant auf der Robotersteuerung installiert werden.



Allgemeine Informationen zur Installation von RobotWare-Add-Ins finden Sie in der entsprechenden Dokumentation von ABB.

#### 2.1.1 Installation über ABB RobotStudio®

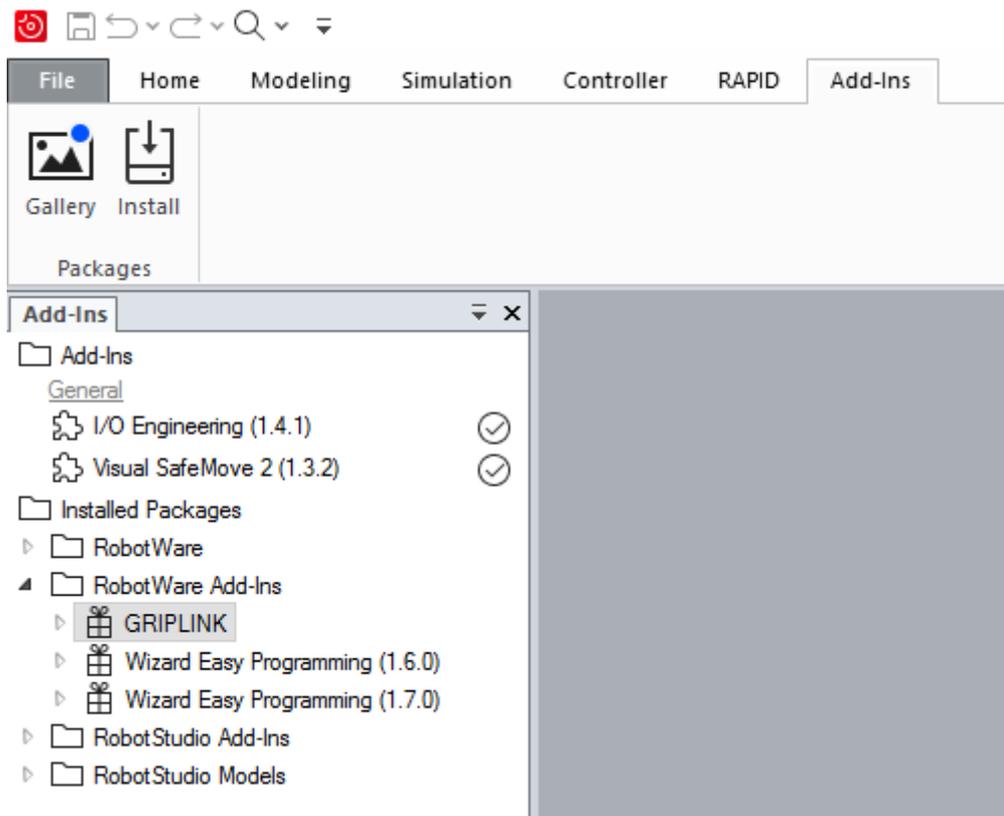
Um das GRIPLINK-Plugin mithilfe von RobotStudio zu installieren, wird ein PC benötigt, auf dem die Programmier- und Simulationsumgebung ABB RobotStudio® installiert ist und der mit der Steuerung des Roboters verbunden ist. Führen Sie dann folgende Schritte aus:

1. Laden Sie unter [www.griplink.de](http://www.griplink.de) die aktuelle Version des GRIPLINK-Plugins für ABB herunter und entpacken Sie die ZIP-Datei.
2. Starten Sie RobotStudio und die Robotersteuerung.
3. Wählen Sie den Reiter *Add-Ins* und klicken Sie auf *Install*.



4. Navigieren Sie zum Speicherort des heruntergeladenen GRIPLINK-Plugins. Wählen Sie dort das RobotStudio-Paket, also die Datei mit der Endung *.rspak*, aus.

- Nach der Installation in RobotStudio erscheint das GRIPLINK-Plugin in der Übersicht verfügbarer Add-Ins am linken Rand des Fensters.

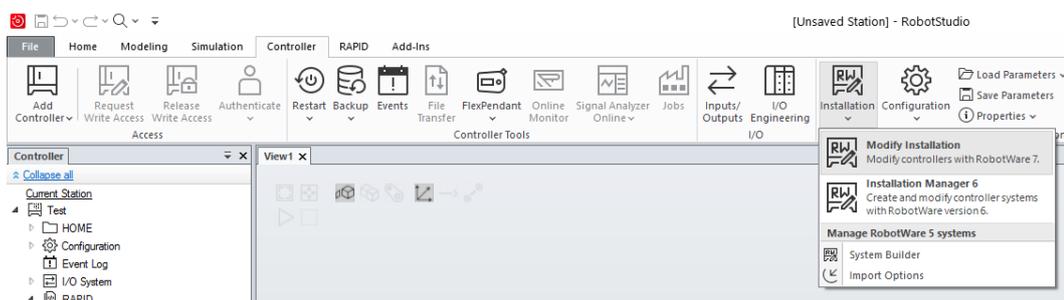


- Öffnen oder erstellen Sie ein Projekt mit der realen oder virtuellen Robotersteuerung, auf der das GRIPLINK-Plugin installiert werden soll, oder stellen Sie direkt eine Verbindung zur entsprechenden Steuerung her.

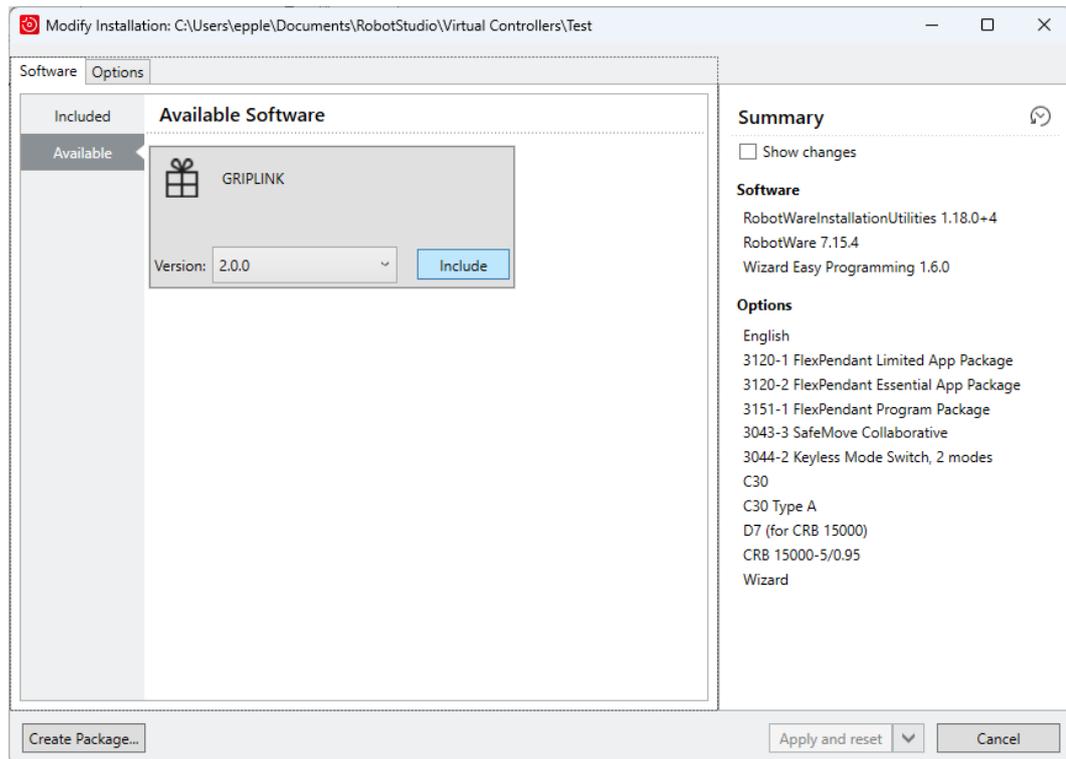


Details zur Bedienung von ABB RobotStudio® finden Sie in der entsprechenden Dokumentation von ABB.

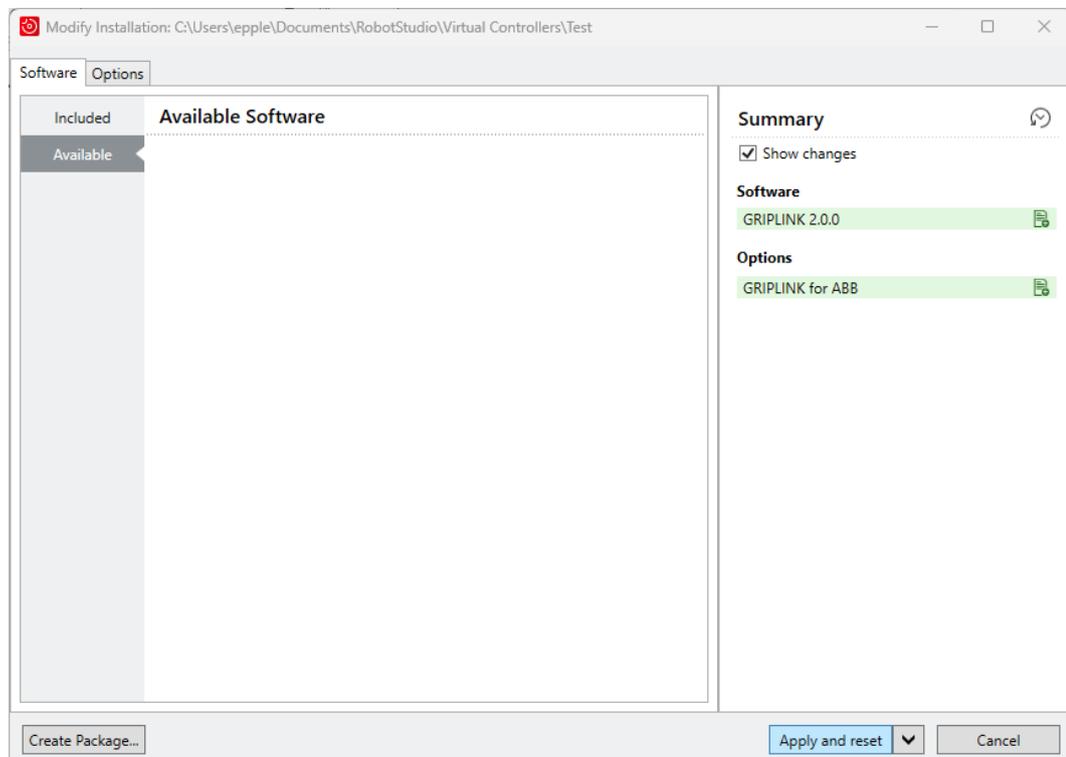
- Wählen Sie den Reiter *Controller* und klicken Sie unter *Installation* auf *Modify Installation*.



8. Wählen Sie den Reiter *Software*. Unter *Available* wird das GRIPLINK-Plugin aufgeführt. Wählen Sie gegebenenfalls eine Version aus und klicken Sie auf *Include*.



9. Unter *Summary* am rechten Rand des Fensters wird grün hinterlegt das GRIPLINK-Plugin angezeigt. Klicken Sie auf *Apply and Reset* und bestätigen Sie den Neustart im Popup-Fenster.

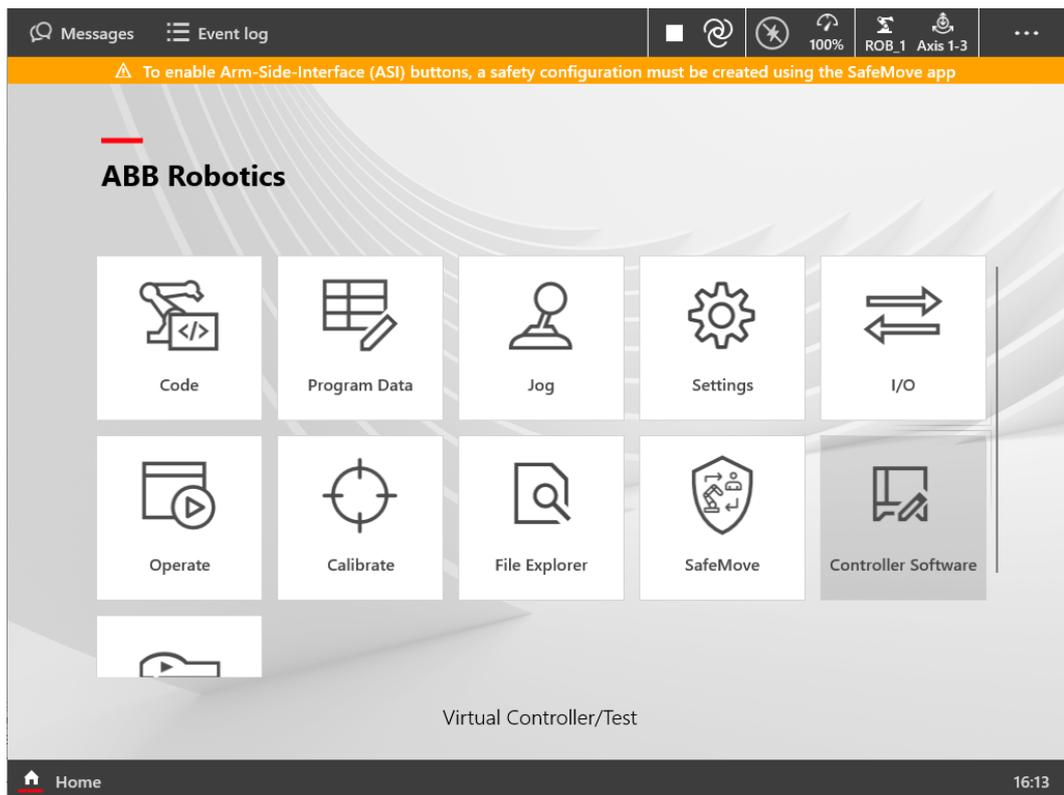


Nach dem Neustart der Robotersteuerung ist das GRIPLINK-Plugin installiert und die entsprechende Funktionalität steht zur Verfügung.

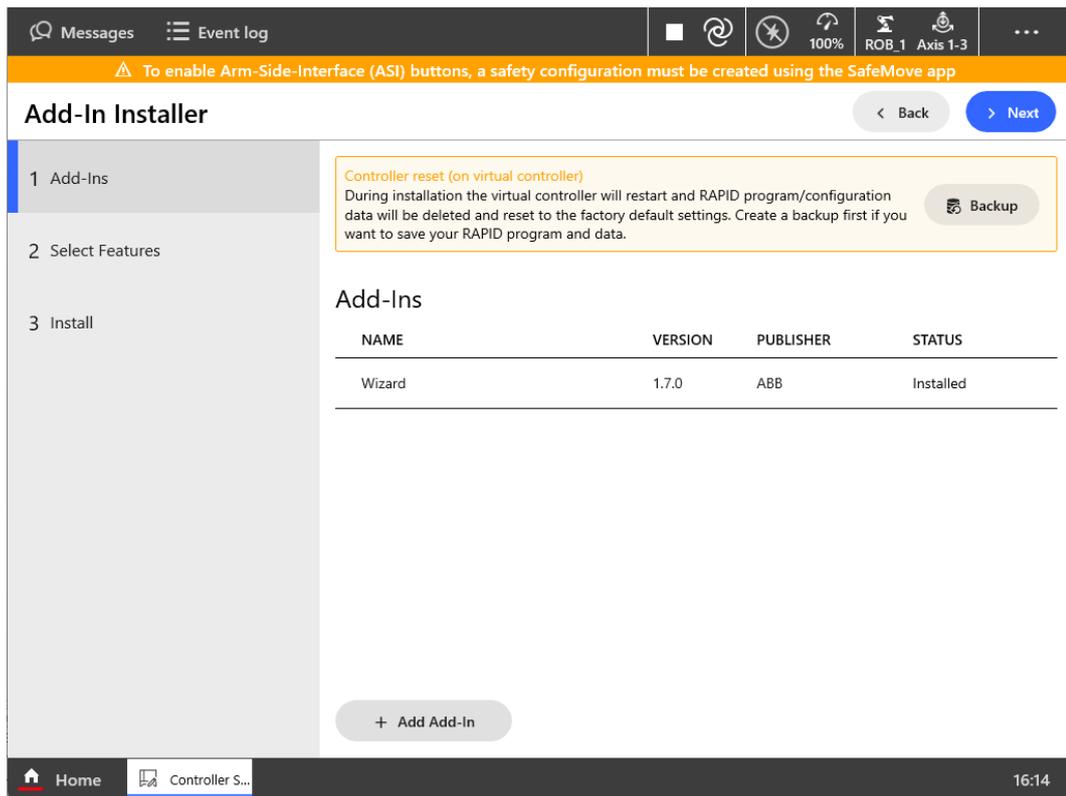
## 2.1.2 Installation am FlexPendant

Um das GRIPLINK-Plugin über das FlexPendant zu installieren, ist kein mit der Robotersteuerung verbundener PC nötig. Das Plugin wird stattdessen mit einem USB-Stick übertragen, der am FlexPendant angeschlossen wird. Führen Sie dazu folgende Schritte aus:

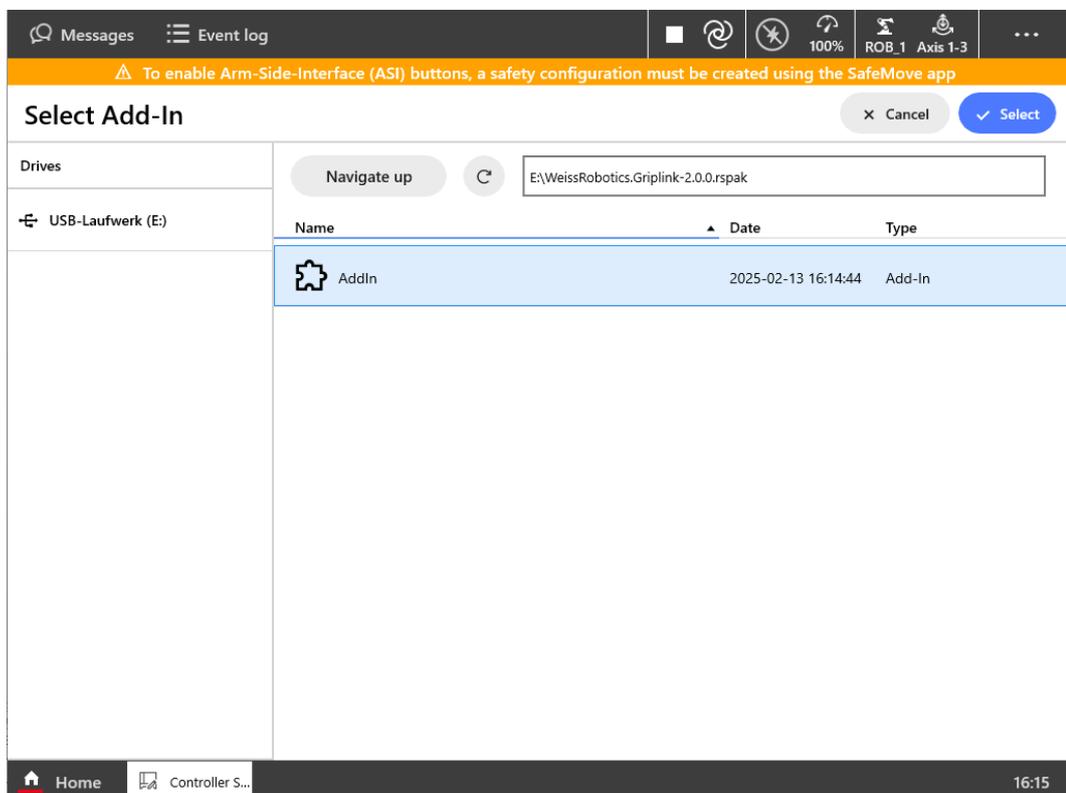
1. Laden Sie unter [www.griplink.de](http://www.griplink.de) die aktuelle Version des GRIPLINK-Plugins für ABB herunter, entpacken Sie die ZIP-Datei und kopieren Sie das Plugin auf den USB-Stick.
2. Starten Sie die Robotersteuerung und schließen Sie den USB-Stick am FlexPendant an.
3. Starten Sie im Hauptmenü des FlexPendant die App *Controller Software* und öffnen Sie dort den *Add-In Installer*.



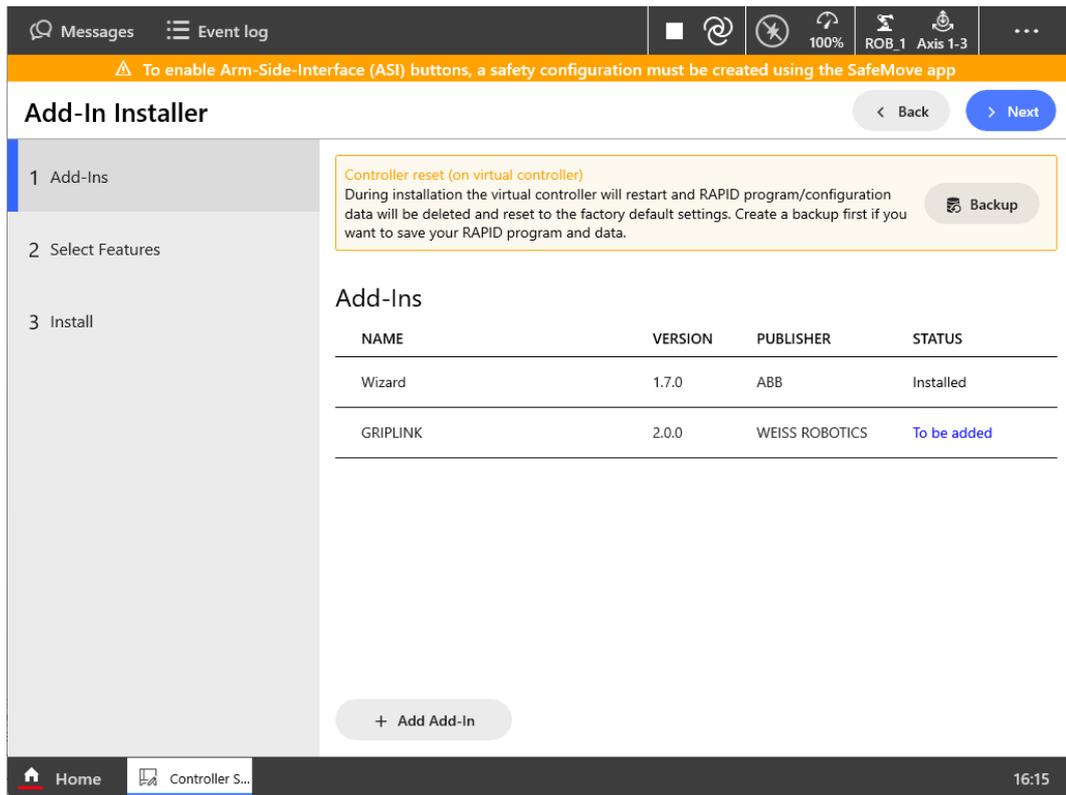
4. Im *Add-In Installer* wählen Sie unten die Schaltfläche *Add Add-In*.



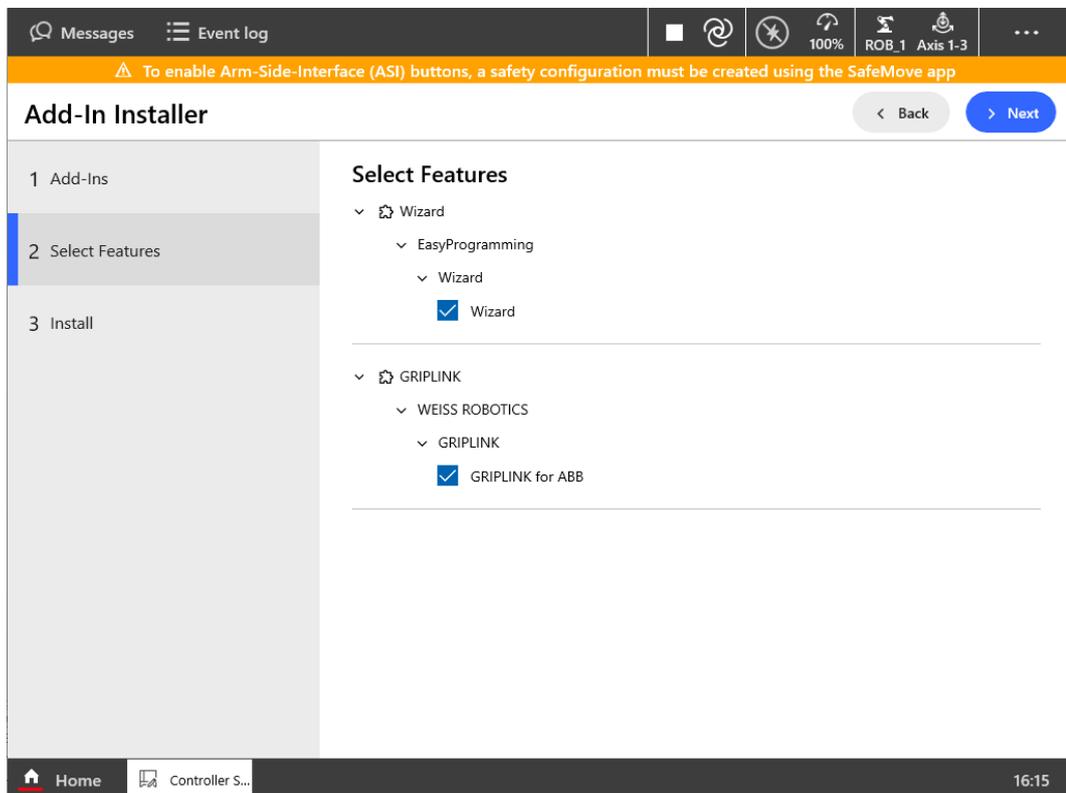
5. Navigieren Sie zum Speicherort des Plugins auf dem USB-Stick. Wählen Sie dort in der Datei mit der Endung *.rspak* das enthaltene *AddIn* aus und bestätigen Sie die Wahl oben rechts mit *Select*.



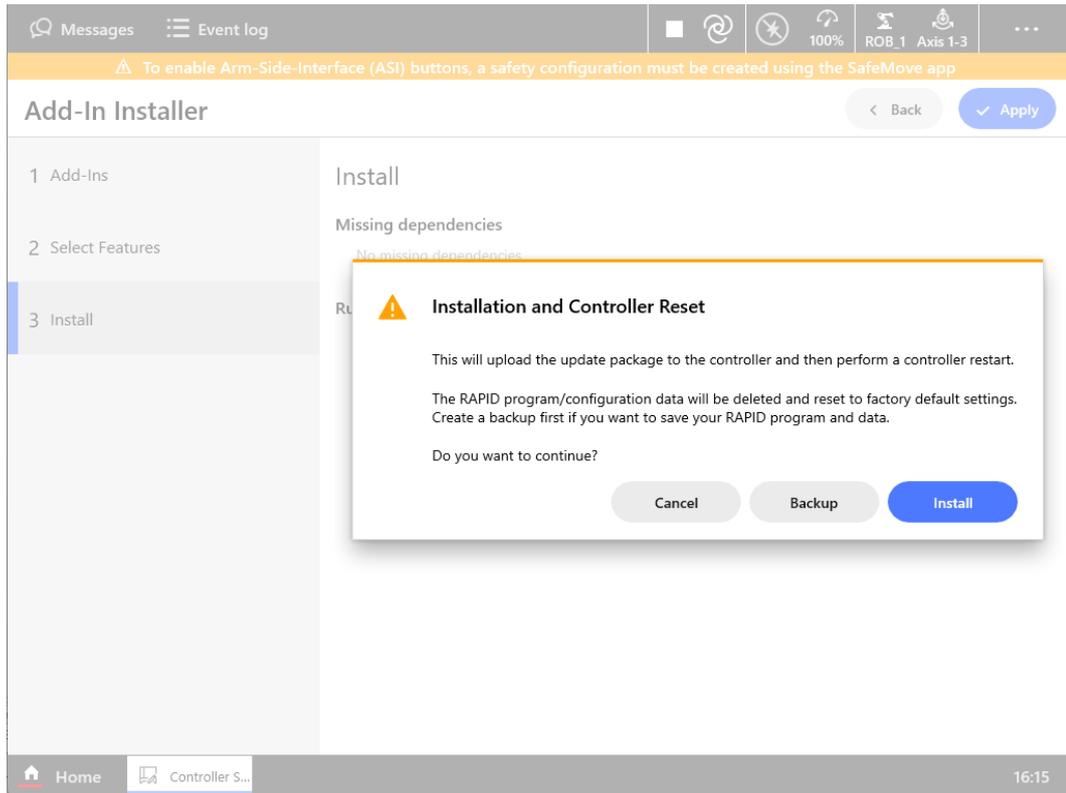
- Im *Add-In Installer* wird unter *Add-Ins* das *GRIPLINK*-Plugin mit dem Status *To be added* angezeigt. Klicken Sie oben rechts auf *Next*.



- Unter *Select Features* muss der Haken bei *GRIPLINK for ABB* gesetzt sein. Klicken Sie dann oben rechts auf *Next*.



8. Unter *Install* klicken Sie oben rechts auf *Apply* und bestätigen Sie im Popup-Fenster mit *Install*, dass die Installation und der damit verbundene Neustart ausgeführt werden soll.



Nach dem Neustart der Robotersteuerung ist das GRIPLINK-Plugin installiert und die entsprechende Funktionalität steht zur Verfügung.

## 2.2 Netzwerk-Einstellungen

Die Verbindung zwischen dem GRIPLINK-Controller und der Robotersteuerung basiert auf TCP/IP und erfolgt über TCP-Port 10001. Um eine Verbindung herstellen zu können ist es notwendig, dass die IP-Adressen des GRIPLINK-Controllers und der Steuerung kompatibel sind. Werkseitig hat der GRIPLINK-Controller folgende Standard-Netzwerkeinstellungen:

IP-Adresse:	192.168.1.40
Subnetzmaske:	255.255.255.0



Die IP-Adresse des GRIPLINK-Controllers sowie die Subnetzmaske können über die Weboberfläche konfiguriert werden.



Informationen zur Konfiguration der IP-Adresse der Robotersteuerung finden Sie in der zugehörigen Bedienungsanleitung.



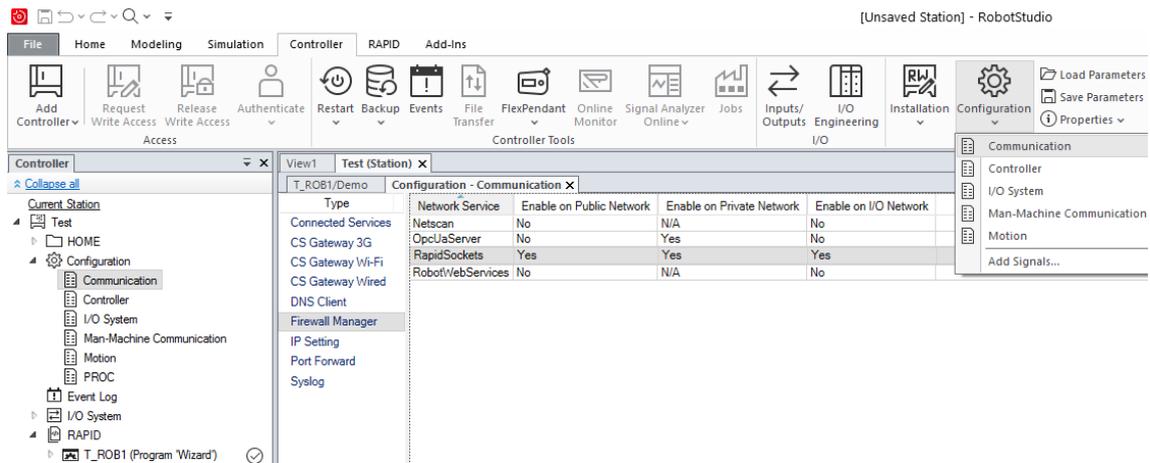
Für virtuelle Steuerungen in ABB RobotStudio® wird die IP-Adresse der Netzwerkschnittstelle des Computers verwendet.



Die IP-Adressen des GRIPLINK-Controllers und der Robotersteuerung müssen im selben Subnetz liegen.

## 2.3 Firewall-Einstellungen

Um die TCP/IP-Verbindung zwischen dem GRIPLINK-Controller und der Robotersteuerung aufzubauen, werden im Plugin RAPID Sockets verwendet. Diese werden daher beim Installationsprozess in der Firewall der Steuerung aktiviert. In manchen Fällen kann es allerdings notwendig sein, die Firewall manuell für die Nutzung von RAPID Sockets zu konfigurieren. Die entsprechenden Einstellungen werden in Robot Studio vorgenommen und befinden sich dort unter dem Reiter *Controller* bei *Configuration > Communication > Firewall Manager > RapidSockets*. Stellen Sie für eine ordnungsgemäße Funktion des Plugins sicher, dass diese zumindest für den verwendeten Netzwerk-Typ aktiviert sind.



## 3 Funktionsumfang

Das GRIPLINK-Plugin stellt eine Sammlung von Befehlen bereit, um einen GRIPLINK-Controller und die daran angeschlossenen Geräte, sowohl Greifmodule von WEISS ROBOTICS als auch Sensoren und Aktoren von diversen Drittanbietern, direkt über die ABB-Robotersteuerung zu bedienen.

### 3.1 RAPID

Im GRIPLINK-Plugin sind diverse Befehle des GRIPLINK-Protokolls als RAPID-Prozeduren implementiert. Nach Installation des Plugins stehen diese Prozeduren zur Verwendung in Nutzerprogrammen zur Verfügung.



Die Funktionsweise der einzelnen Befehle wird im Kapitel **Befehlsreferenz** (4) detailliert erläutert.



Informationen zur Programmierung mit RAPID finden Sie in der entsprechenden Dokumentation von ABB.

```
3 PROC main()
4
5     GRIPLINK_Connect "192.168.1.40";
6     GRIPLINK_AssertDevice 0, 815, 6060;
7     GRIPLINK_SetGripConfig 0, 1, "Demo", 2.5, 100, 50, 0, 0, 0, 0;
8     GRIPLINK_Home 0;
9     GRIPLINK_Release 0, 1, TRUE;
10
11 FOR nIndex FROM 1 TO 10 DO
12     WaitTime 1;
13     GRIPLINK_Grip 0, 1, TRUE;
14     WaitTime 1;
15     IF nGRIPLINK_State = GRIPLINK_DS_HOLDING THEN
16         WaitTime 3;
17         GRIPLINK_Release 0, 1, TRUE;
18     ELSEIF nGRIPLINK_State = GRIPLINK_DS_NO_PART THEN
19         WaitTime 1;
20         GRIPLINK_Release 0, 1, TRUE;
21     ELSE
22         Stop;
23     ENDIF
24 ENDFOR
25
26 GRIPLINK_Disconnect;
27
28 ENDPROC
```

## 3.2 Wizard

Neben den klassischen RAPID-Prozeduren sind alle Befehle auch als Blöcke für die graphische Programmieroberfläche Wizard Easy Programming im Plugin enthalten. Damit können einfachere Roboteranwendungen direkt auf dem FlexPendant eines ABB-Roboters programmiert werden.

Die enthaltenen Funktionsblöcke befinden sich in der Kategorie *GRIPLINK*, welche nach Installation des Plugins im Menü auf der linken Seite des Wizard-Bildschirms verfügbar ist. Jeder Block entspricht dabei einer der RAPID-Prozeduren, sodass auch ein Wechsel zwischen der Programmierung in RAPID und mit Wizard in einem gewissen Rahmen möglich ist („Roundtripping“).



Die Funktionsweise der einzelnen Blöcke wird im Kapitel **Befehlsreferenz** (4) detailliert erläutert.



Allgemeine Informationen zur grafischen Programmierung mit Wizard finden Sie in der entsprechenden Dokumentation von ABB.

The screenshot displays the ABB Wizard Easy Programming interface. At the top, there is a status bar with a warning message: "To enable Arm-Side-Interface (ASI) buttons, a safety configuration must be created using the SafeMove app". Below this, the "Wizard" menu is visible on the left, with categories like Message, Motion, Procedures, Flow control, Signals, Variables, Script, and GRIPLINK. The main workspace shows a sequence of blocks:

- When play pressed, do this:** (Trigger)
- Connect to GRIPLINK with IP address** (192.168.1.40)
- Assert device with vendor ID** (815) **and product ID** (6060) **at port** (0)
- Perform homing sequence on device at port** (0)
- Grip workpiece with device at port** (0) **using preset** (1) **and wait for state change**
- If** **nGRIPLINK\_State** **=** **GRIPLINK\_DS\_HOLDING**
  - do**
    - Wait** (3) **seconds**
    - Release workpiece with device at port** (0) **using preset** (1) **and wait for state change**
  - Else if** **nGRIPLINK\_State** **=** **GRIPLINK\_DS\_NO\_PART**
    - do**
      - Wait** (1) **seconds**
      - Release workpiece with device at port** (0) **using preset** (1) **and wait for state change**
    - Else** **Stop**
- Disconnect from GRIPLINK**

The bottom of the screen shows a navigation bar with "Home" and "Wizard" buttons, and a clock showing "11:48".

### 3.3 Globale Variablen für Rückgabewerte

Etwaige Rückgabewerte von Befehlen werden standardmäßig in globale Variablen geschrieben. In der folgenden Tabelle sind die dazu im Plugin (im Modul *GRIPLINK\_Data*) definierten globalen Variablen aufgelistet.

Globale Variable	Typ	Bedeutung
nGRIPLINK_State	num	letzter ausgelesener Gerätezustand
nGRIPLINK_Value	num	letzter ausgelesener Gerätewert
stGRIPLINK_Tag	string	letzter ausgelesener Geräte-Tag



In RAPID kann alternativ direkt beim Aufruf eines Befehls eine andere Variable für den Rückgabewert spezifiziert werden.

### 3.4 Gerätezustände

Die folgende Tabelle listet die möglichen Zustandswerte angeschlossener Geräte auf. Die in der ersten Spalte angegebenen Konstanten sind im Plugin (im Modul *GRIPLINK\_Data*) definiert und können im Roboterprogramm verwendet werden, um mit einem abgefragten Gerätezustand zu vergleichen.

Gerätezustand	Wert	Bedeutung
GRIPLINK_DS_NOT_CONNECTED	0	Gerät nicht verbunden
GRIPLINK_DS_NOT_INITIALIZED	1	Gerät nicht initialisiert
GRIPLINK_DS_DISABLED	2	Gerät deaktiviert
GRIPLINK_DS_RELEASED	3	Werkstück freigegeben
GRIPLINK_DS_NO_PART	4	kein Werkstück gefunden
GRIPLINK_DS_HOLDING	5	Werkstück gegriffen
GRIPLINK_DS_ENABLED	6	Gerät aktiviert
GRIPLINK_DS_FAULT	7	Fehler

### 3.5 Fehlerbehandlung

Wenn bei der Ausführung von Code des GRIPLINK-Plugins Probleme auftreten, wird auf der Robotersteuerung ein Fehler ausgelöst. Wenn dieser Fehler nicht im Roboterprogramm aufgefangen und verarbeitet wird, wird die Ausführung gestoppt.



Es liegt in der Verantwortung des Programmierers, auf ausgelöste Fehler angemessen zu reagieren und das System in einen sicheren Zustand zu bringen! Die dafür nötigen Werkzeuge werden durch die ABB-Programmierungsumgebung bereitgestellt.

Um zwischen unterschiedlichen Arten von Fehlern unterscheiden zu können, werden im GRIPLINK-Plugin mehrere Konstanten für die RAPID-Fehlernummer (*ERRNO*) verwendet. Die entsprechenden Werte werden beim Aufruf des Befehls *GRIPLINK\_Connect* über die Instruktion *BookErrNo* reserviert. So bekommen die Konstanten automatisch einen nicht anderweitig vergebenen Wert aus dem verfügbaren Wertebereich für benutzerdefinierte Fehlernummern zugewiesen. In der folgenden Tabelle sind die verfügbaren Konstanten und die jeweiligen Bedeutungen aufgelistet.

Fehlernummer (ERRNO)	Fehlerbeschreibung
ERR_GRIPLINK_COMMAND	Fehler bei der Ausführung eines Befehls
ERR_GRIPLINK_CONNECTION	Fehler bei der Verbindung mit dem GRIPLINK-Controller
ERR_GRIPLINK_DEVICE	Fehler eines angeschlossenen Geräts
ERR_GRIPLINK_PARAMETER	Fehler durch ungültige Parameter



Weitere Informationen zum Error-Handling in RAPID-Programmen finden Sie in den Anleitungen zu Ihrer ABB-Robotersteuerung und der Dokumentation zu RAPID.

### 3.6 Genereller Programmablauf

Der grundlegende Ablauf eines Programms unter Verwendung des GRIPLINK-Plugins sieht typischerweise folgendermaßen aus:

#### **Vorbereitung**

- Verbindungsaufbau zum GRIPLINK:  
*GRIPLINK\_Connect*
- Überprüfung der angeschlossenen Geräte:  
*GRIPLINK\_AssertDevice*
- Initialisierung der angeschlossenen Geräte:  
*GRIPLINK\_Enable / GRIPLINK\_Home*
- Konfiguration der angeschlossenen Geräte:  
*GRIPLINK\_SetGripConfig / ...*

#### **Hauptprogramm (gegebenenfalls als Schleife)**

- Befehle zum Greifen und Freigeben:  
*GRIPLINK\_Grip / GRIPLINK\_Release / GRIPLINK\_FlexGrip / GRIPLINK\_FlexRelease*
- Abfrage von Zuständen und Werten (und entsprechende Verarbeitung):  
*GRIPLINK\_DeviceState / GRIPLINK\_Value / ...*

#### **Programmende**

- Deaktivieren der angeschlossenen Geräte:  
*GRIPLINK\_Disable*
- Trennen der Verbindung zum GRIPLINK:  
*GRIPLINK\_Disconnect*

### 3.7 Demo-Programme

Um die typische Verwendung des GRIPLINK-Plugins zu demonstrieren und einfache Tests des Plugins zu ermöglichen, sind im Modul *GRIPLINK\_Demo* mehrere Demo-Programme enthalten. Die jeweiligen Prozeduren können entweder im Roboterprogramm aufgerufen werden oder als Vorlage für eigene Programme verwendet und angepasst werden.



Die Demo-Programme sind ausschließlich zu Demonstrations- und Testzwecken vorgesehen! Speziell die rudimentäre Fehlerbehandlung ist für den Produktionseinsatz gegebenenfalls nicht ausreichend!

## 4 Befehlsreferenz

Im folgenden Kapitel werden die im GRIPLINK-Plugin enthaltenen Befehle im Detail beschrieben. Dabei wird sowohl auf die Prozeduren in RAPID, als auch auf die entsprechenden Blöcke für die grafische Programmierung mit Wizard eingegangen.

Die zur Verfügung stehenden Befehle sind in der folgenden Tabelle aufgeführt:

Befehl	Beschreibung
GRIPLINK_Connect	Verbindung zu GRIPLINK-Controller aufbauen
GRIPLINK_Disconnect	Verbindung zu GRIPLINK-Controller trennen
GRIPLINK_AssertDevice	Angeschlossenes Gerät prüfen
GRIPLINK_DeviceTag	Geräte-Tag abfragen
GRIPLINK_SetDeviceTag	Geräte-Tag setzen
GRIPLINK_Enable	Gerät aktivieren
GRIPLINK_Disable	Gerät deaktivieren
GRIPLINK_Home	Gerät referenzieren
GRIPLINK_Grip	Werkstück greifen
GRIPLINK_Release	Werkstück freigeben
GRIPLINK_FlexGrip	Werkstück mit bestimmten Parametern greifen
GRIPLINK_FlexRelease	Werkstück mit bestimmten Parametern freigeben
GRIPLINK_LED	LED-Visualisierung wählen
GRIPLINK_Clamp	Mechanische Greifkraftherhaltung einstellen
GRIPLINK_WaitForStateTransition	Auf Zustandsänderung eines Geräts warten
GRIPLINK_SetValue	Gerätewert setzen
GRIPLINK_WaitValue	Auf Gerätewert warten
GRIPLINK_DeviceState	Gerätezustand abfragen
GRIPLINK_Value	Gerätewert abfragen
GRIPLINK_SetGripConfig GRIPLINK_SetGripConfig: Greifparameter konfigurieren	Greifparameter konfigurieren

## 4.1 GRIPLINK\_Connect: Verbindung aufbauen

Dieser Befehl stellt die Verbindung zwischen einem GRIPLINK-Controller mit der angegebenen IPv4-Adresse und der ABB-Robotersteuerung her. Es wird außerdem die vom GRIPLINK-Controller unterstützte Protokoll-Version überprüft und ein Fehler ausgelöst, falls diese nicht den Mindestanforderungen entspricht.



Die werkseitige Standard-IP-Adresse ist 192.168.1.40 (Subnetz-Maske: 255.255.255.0).



Die IP-Adresse des GRIPLINK-Controllers sowie die Subnetzmaske können über die Weboberfläche konfiguriert werden.



Die IP-Adresse des GRIPLINK-Controllers muss im selben Subnetz liegen wie die der Robotersteuerung.

### RAPID-Befehl

```
PROC GRIPLINK_Connect (  
    string stIPAddr  
)
```

Parameter	Typ	Bedeutung
stIPAddr	string	IPv4-Adresse des GRIPLINK-Controllers

### Wizard-Block

Connect to GRIPLINK with IP address 192.168.1.40

### Beispiel

Aufbau einer Verbindung zu einem GRIPLINK-Controller mit der IP-Adresse 192.168.1.40:

```
GRIPLINK_Connect "192.168.1.40";
```

## 4.2 GRIPLINK\_Disconnect: Verbindung trennen

Dieser Befehl trennt die aktuell bestehende Verbindung zwischen einem GRIPLINK-Controller und der ABB-Robotersteuerung.



Nach dem Trennen der Verbindung können bis zum erneuten Aufruf des Befehls *GRIPLINK\_Connect* (4.1) keine anderen Befehle gesendet werden!

### RAPID-Befehl

```
PROC GRIPLINK_Disconnect()
```

### Wizard-Block



Disconnect from GRIPLINK

### Beispiel

Trennen der aktuell bestehenden Verbindung zu einem GRIPLINK-Controller:

```
GRIPLINK_Disconnect;
```

### 4.3 GRIPLINK\_AssertDevice: Angeschlossenes Gerät prüfen

Dieser Befehl prüft, ob an einem bestimmten Port ein Gerät mit der angegebenen IO-Link-Vendor-ID und Product-ID angeschlossen ist. Sollte dies nicht der Fall sein, wird ein Fehler ausgelöst und das Programm angehalten. Diese Überprüfung ist generell empfehlenswert, bevor ein Gerät im Programmablauf verwendet wird, um sicherzustellen, dass tatsächlich das erwartete Gerät angeschlossen ist.



Die Vendor- und Product-ID finden sie in der Betriebsanleitung des jeweiligen Geräts oder in der auf [www.griplink.de](http://www.griplink.de) unter *Downloads* verfügbaren Übersichtstabelle.

#### RAPID-Befehl

```
PROC GRIPLINK_AssertDevice(  
    num nPort,  
    num nVID,  
    num nPID  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
nVID	num	IO-Link-Vendor-ID des erwarteten Geräts
nPID	num	IO-Link-Device-ID des erwarteten Geräts

#### Wizard-Block

Assert device with vendor ID  and product ID  at port

#### Beispiel

Prüfen auf einen IEG PLUS 260-030 (Product-ID 24) von WEISS ROBOTICS (Vendor-ID 815) am Port 0 eines GRIPLINK-Controllers:

```
GRIPLINK_AssertDevice 0, 815, 24;
```

## 4.4 GRIPLINK\_DeviceTag: Geräte-Tag abfragen

Dieser Befehl liest den Tag des Geräts an einem bestimmten GRIPLINK-Port aus. Dieser Tag kann zum Beispiel verwendet werden, um das Gerät in komplexeren Szenarien mit vielen Geräten eindeutig zu identifizieren.

### RAPID-Befehl

```
PROC GRIPLINK_DeviceTag(  
    num nPort,  
    \VAR string stTag  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
\stTag	string	optionale Variable für den Rückgabewert

Rückgabewert	Typ	Bedeutung
stGRIPLINK_Tag	string	aktueller Geräte-Tag



Wird das optionale Argument `\stTag` spezifiziert, wird der Rückgabewert nur in diese Variable geschrieben.

### Wizard-Block

Get tag of device at port 0

### Beispiel

Auslesen des Tags eines Geräts am Port 1 eines GRIPLINK-Controllers:

```
GRIPLINK_DeviceTag 1;
```

## 4.5 GRIPLINK\_SetDeviceTag: Geräte-Tag setzen

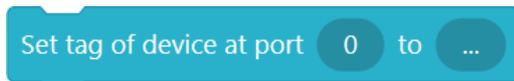
Dieser Befehl setzt den Tag des Geräts an einem bestimmten GRIPLINK-Port. Dieser Tag kann zum Beispiel verwendet werden, um das Gerät in komplexeren Szenarien mit vielen Geräten eindeutig zu identifizieren.

### RAPID-Befehl

```
PROC GRIPLINK_SetDeviceTag(  
    num nPort,  
    string stTag  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
stTag	string	zu setzender Geräte-Tag

### Wizard-Block



### Beispiel

Setzen des Tags eines Geräts am Port 1 eines GRIPLINK-Controllers:

```
GRIPLINK_SetDeviceTag 1, "Device_1";
```

## 4.6 GRIPLINK\_Enable: Gerät aktivieren

Dieser Befehl aktiviert das an einem bestimmten GRIPLINK-Port angeschlossene Gerät. Der genaue Effekt dieses Kommandos hängt vom jeweiligen Gerät ab. Bei manchen Geräten können Fehler durch Deaktivieren und anschließendes Aktivieren quitiert werden.



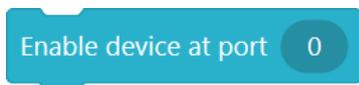
Greifer von WEISS ROBOTICS werden beim Referenzieren sowie durch die Ausführung von Greif- oder Freigabebefehlen automatisch (wieder) aktiviert.

### RAPID-Befehl

```
PROC GRIPLINK_Enable(  
    num nPort  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port

### Wizard-Block



### Beispiel

Aktivieren des Geräts am Port 2 eines GRIPLINK-Controllers:

```
GRIPLINK_Enable 2;
```

## 4.7 GRIPLINK\_Disable: Gerät deaktivieren

Dieser Befehl deaktiviert das an einem bestimmten GRIPLINK-Port angeschlossene Gerät. Der genaue Effekt dieses Kommandos hängt vom jeweiligen Gerät ab. Bei manchen Geräten können Fehler durch Deaktivieren und anschließendes Aktivieren quitiert werden.



Bei Greifern von WEISS ROBOTICS wird durch diesen Befehl der Antrieb deaktiviert, sodass sich die Finger des Greifers von Hand bewegen lassen.

### RAPID-Befehl

```
PROC GRIPLINK_Disable(  
    num nPort  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port

### Wizard-Block

Disable device at port 0

### Beispiel

Deaktivieren des Geräts am Port 2 eines GRIPLINK-Controllers:

```
GRIPLINK_Disable 2;
```

## 4.8 GRIPLINK\_Home: Gerät referenzieren

Dieser Befehl führt die Referenzierung eines an einem bestimmten GRIPLINK-Port angeschlossenen Gerätes durch. Eine solche Initialisierung kann je nach Gerät notwendig sein, um eine ordnungsgemäße Funktion zu gewährleisten. Bei Greifern werden üblicherweise die Finger zum inneren oder äußeren mechanischen Anschlag bewegt, um die Position der Finger zu referenzieren. Der Befehl blockiert die weitere Ausführung des Programms, bis die Referenzierung abgeschlossen ist.



Weitere Details zur Referenzierung finden Sie in der Bedienungsanleitung des jeweiligen Geräts.



Bei Greifmodulen kann über die Weboberfläche des GRIPLINK-Controllers eingestellt werden, ob die Referenzierung mithilfe des inneren oder des äußeren Anschlags durchgeführt wird.

### RAPID-Befehl

```
PROC GRIPLINK_Home (  
    num nPort  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port

### Wizard-Block

Perform homing sequence on device at port 0

### Beispiel

Referenzierung des Geräts am Port 3 eines GRIPLINK-Controllers:

```
GRIPLINK_Home 3;
```

## 4.9 GRIPLINK\_Grip: Werkstück greifen

Dieser Befehl führt einen Griff mit einem an einen bestimmten GRIPLINK-Port angeschlossenen Greifer aus. Dafür werden die im angegebenen Greif-Rezept hinterlegten Parameter verwendet. Die weitere Ausführung des Programms kann entweder sofort nach der Übermittlung des Kommandos und damit vor Abschluss der Bewegung der Finger geschehen, oder bis zu einer Änderung des Gerätezustands und damit dem Abschluss des Greifprozesses unterbrochen werden. Wird auf einen Zustandsübergang gewartet, wird der neue Gerätezustand zurückgegeben.



Zur Konfiguration der Greif-Rezepte kann entweder die Weboberfläche des GRIPLINK-Controllers oder der Befehl *GRIPLINK\_SetGripConfig* (4.20) verwendet werden.

### RAPID-Befehl

```
PROC GRIPLINK_Grip(
    num nPort,
    num nIndex,
    bool bBlocking,
    \VAR num nDevState
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
nIndex	num	Index des Greif-Rezepts
bBlocking	bool	TRUE: auf einen Zustandswechsel des Geräts warten FALSE: nicht auf einen Zustandswechsel des Geräts warten
\nDevState	num	optionale Variable für den Rückgabewert

Rückgabewert	Typ	Bedeutung
nGRIPLINK_State	num	Geräte-Zustand nach Ende des Greifprozesses: GRIPLINK_DS_HOLDING: Werkstück gegriffen GRIPLINK_DS_NO_PART: kein Werkstück gefunden GRIPLINK_DS_FAULT: Fehler



Der neue Gerätezustand wird nur in die Rückgabevariable geschrieben, wenn auf einen Zustandsübergang gewartet wird (`bBlocking = TRUE`).



Wird das optionale Argument `\nDevState` spezifiziert, wird der Rückgabewert nur in diese Variable geschrieben.

### Wizard-Block

Grip workpiece with device at port  using preset  and

### Beispiel

Greifen mit dem Greifer am Port 0 eines GRIPLINK-Controllers unter Verwendung des Greif-Rezepts 1. Dabei soll auf den Abschluss des Greifprozesses gewartet werden, um das weitere Vorgehen vom neuen Gerätezustand abhängig zu machen:

```
! Variable für Gerätezustand anlegen
VAR num nDeviceState;

! Greifen mit Greifer an Port 0 und Rezept 1
GRIPLINK_Grip 0, 1, TRUE, \nDevState := nDeviceState;

! Prüfen des neuen Gerätezustands
IF nDeviceState = GRIPLINK_DS_HOLDING THEN
  ! Werkstück gegriffen, zur Zielposition bewegen
ELSEIF nDeviceState = GRIPLINK_DS_NO_PART THEN
  ! kein Werkstück gefunden, warten und erneut versuchen
ELSE
  ! Fehler, Programm anhalten
ENDIF
```

## 4.10 GRIPLINK\_Release: Werkstück freigeben

Dieser Befehl führt eine Freigabe mit einem an einen bestimmten GRIPLINK-Port angeschlossenen Greifer aus. Dafür werden die im angegebenen Greif-Rezept hinterlegten Parameter verwendet. Die weitere Ausführung des Programms kann entweder sofort nach der Übermittlung des Kommandos und damit vor Abschluss der Bewegung der Finger geschehen, oder bis zu einer Änderung des Gerätezustands und damit dem Abschluss des Freigabeprozesses unterbrochen werden. Wird auf einen Zustandsübergang gewartet, wird der neue Gerätezustand zurückgegeben.



Zur Konfiguration der Greif-Rezepte kann entweder die Weboberfläche des GRIPLINK-Controllers oder der Befehl *GRIPLINK\_SetGripConfig* (4.20) verwendet werden.

### RAPID-Befehl

```
PROC GRIPLINK_Release(
    num nPort,
    num nIndex,
    bool bBlocking,
    \VAR num nDevState
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
nIndex	num	Index des Greif-Rezepts
bBlocking	bool	TRUE: auf einen Zustandswechsel des Geräts warten FALSE: nicht auf einen Zustandswechsel des Geräts warten
\nDevState	num	optionale Variable für den Rückgabewert

Rückgabewert	Typ	Bedeutung
nGRIPLINK_State	num	Geräte-Zustand nach Ende des Freigabeprozesses: GRIPLINK_DS_RELEASED: Werkstück freigegeben GRIPLINK_DS_FAULT: Fehler



Der neue Gerätezustand wird nur in die Rückgabewariable geschrieben, wenn auf einen Zustandsübergang gewartet wird (`bBlocking = TRUE`).



Wird das optionale Argument `\nDevState` spezifiziert, wird der Rückgabewert nur in diese Variable geschrieben.

### Wizard-Block

Release workpiece with device at port  using preset  and

### Beispiel

Freigeben eines Werkstücks mit dem Greifer am Port 0 eines GRIPLINK-Controllers unter Verwendung des Greif-Rezepts 1. Dabei soll auf den Abschluss des Freigabeprozesses gewartet werden, um zu prüfen, ob das Werkstück korrekt freigegeben wurde:

```
! Freigeben mit Greifer an Port 0 und Rezept 1
GRIPLINK_Release 0, 1, TRUE;

! Prüfen des neuen Gerätezustands
IF nGRIPLINK_State = GRIPLINK_DS_RELEASED THEN
    ! Werkstück freigegeben, fortfahren
ELSE
    ! Fehler, Programm anhalten
ENDIF
```

## 4.11 GRIPLINK\_FlexGrip: Werkstück greifen mit flexiblen Parametern

Dieser Befehl führt einen Griff mit einem an einen bestimmten GRIPLINK-Port angeschlossenen Greifer aus. Die Greifparameter können dabei unabhängig von den hinterlegten Greif-Rezepten direkt angegeben werden. Die weitere Ausführung des Programms kann entweder sofort nach der Übermittlung des Kommandos und damit vor Abschluss der Bewegung der Finger geschehen, oder bis zu einer Änderung des Gerätezustands und damit dem Abschluss des Greifprozesses unterbrochen werden. Wird auf einen Zustandsübergang gewartet, wird der neue Gerätezustand zurückgegeben.



Weitere Informationen zu den Greifparametern und den zulässigen Wertebereichen finden Sie in der Betriebsanleitung des jeweiligen Greifmoduls.



Diese Funktion ist nicht bei allen Geräten verfügbar!

### RAPID-Befehl

```
PROC GRIPLINK_FlexGrip(  
    num nPort,  
    num nPosition,  
    num nForce,  
    num nSpeed,  
    num nAcceleration,  
    bool bBlocking,  
    \VAR num nDevState  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
nPosition	num	Zielposition (No-Part-Limit) in mm
nForce	num	Greifkraft in N
nSpeed	num	Geschwindigkeit in mm/s (0: automatisch berechnen)
nAcceleration	num	Beschleunigung in mm/s <sup>2</sup> (0: Standardwert verwenden)
bBlocking	bool	TRUE: auf einen Zustandswechsel des Geräts warten FALSE: nicht auf einen Zustandswechsel des Geräts warten
\nDevState	num	optionale Variable für den Rückgabewert

Rückgabewert	Typ	Bedeutung
nGRIPLINK_State	num	Geräte-Zustand nach Ende des Greifprozesses: GRIPLINK_DS_HOLDING: Werkstück gegriffen GRIPLINK_DS_NO_PART: kein Werkstück gefunden GRIPLINK_DS_FAULT: Fehler



Der neue Gerätezustand wird nur in die Rückgabewariable geschrieben, wenn auf einen Zustandsübergang gewartet wird (bBlocking = TRUE).



Wird das optionale Argument \nDevState spezifiziert, wird der Rückgabewert nur in diese Variable geschrieben.

### Wizard-Block

FlexGrip workpiece with device at port  using position  mm, force  N, speed  mm/s, acceleration  mm/s<sup>2</sup>, and

### Beispiel

Greifen eines Werkstücks größer 12,5 mm mit dem Greifer am Port 0 eines GRIPLINK-Controllers mit einer Greifkraft von 60 N, einer Geschwindigkeit von 45 mm/s und Standard-Beschleunigung. Dabei soll auf den Abschluss des Greifprozesses gewartet werden, um das weitere Vorgehen vom neuen Gerätezustand abhängig zu machen:

```
! Greifen mit Greifer an Port 0 und den entsprechenden Parametern
GRIPLINK_FlexGrip 0, 12.5, 60, 45, 0, TRUE;
```

```
! Prüfen des neuen Gerätezustands
IF nGRIPLINK_State = GRIPLINK_DS_HOLDING THEN
  ! Werkstück gegriffen, zur Zielposition bewegen
ELSEIF nGRIPLINK_State = GRIPLINK_DS_NO_PART THEN
  ! kein Werkstück gefunden, warten und erneut versuchen
ELSE
  ! Fehler, Programm anhalten
ENDIF
```

## 4.12 GRIPLINK\_FlexRelease: Werkstück freigeben mit flexiblen Parametern

Dieser Befehl führt eine Freigabe mit einem an einen bestimmten GRIPLINK-Port angeschlossenen Greifer aus. Die Greifparameter können dabei unabhängig von den hinterlegten Greif-Rezepten direkt angegeben werden. Der Befehl kann auch zum Vorpositionieren der Finger eines Greifers verwendet werden. Die weitere Ausführung des Programms kann entweder sofort nach der Übermittlung des Kommandos und damit vor Abschluss der Bewegung der Finger geschehen, oder bis zu einer Änderung des Gerätezustands und damit dem Abschluss des Freigabeprozesses unterbrochen werden. Wird auf einen Zustandsübergang gewartet, wird der neue Gerätezustand zurückgegeben.



Weitere Informationen zu den Greifparametern und den zulässigen Wertebereichen finden Sie in der Betriebsanleitung des jeweiligen Greifmoduls.



Diese Funktion ist nicht bei allen Geräten verfügbar!

### RAPID-Befehl

```
PROC GRIPLINK_FlexRelease (  
    num nPort,  
    num nPosition,  
    num nSpeed,  
    num nAcceleration,  
    bool bBlocking,  
    \VAR num nDevState  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
nPosition	num	Zielposition (Release-Limit) in mm
nSpeed	num	Geschwindigkeit in mm/s (0: Standardwert verwenden)
nAcceleration	num	Beschleunigung in mm/s <sup>2</sup> (0: Standardwert verwenden)
bBlocking	bool	TRUE: auf einen Zustandswechsel des Geräts warten FALSE: nicht auf einen Zustandswechsel des Geräts warten
\nDevState	num	optionale Variable für den Rückgabewert

Rückgabewert	Typ	Bedeutung
nGRIPLINK_State	num	Geräte-Zustand nach Ende des Freigabeprozesses: GRIPLINK_DS_RELEASED: Werkstück freigegeben GRIPLINK_DS_FAULT: Fehler

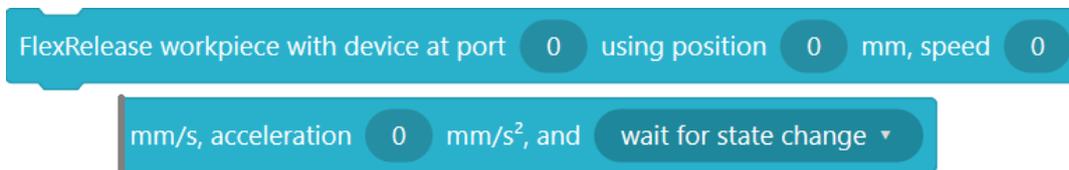


Der neue Gerätezustand wird nur in die Rückgabevariable geschrieben, wenn auf einen Zustandsübergang gewartet wird (`bBlocking = TRUE`).



Wird das optionale Argument `\nDevState` spezifiziert, wird der Rückgabewert nur in diese Variable geschrieben.

### Wizard-Block



### Beispiel

Vorsichtiges Greifen eines Werkstücks größer 12,5 mm mit dem Greifer am Port 0 eines GRIPLINK-Controllers mit Greifkraft 80 N, Geschwindigkeit 25 mm/s und Beschleunigung 25 mm/s<sup>2</sup>. Dabei sollen die Finger zuerst mit größerer Geschwindigkeit mit *GRIPLINK\_FlexRelease* auf eine Position von 20 mm vorpositioniert werden:

```
! Vorpositionieren der Finger
GRIPLINK_FlexRelease 0, 20, 150, 300, FALSE;

! Auf Erreichen der Fingerposition warten
GRIPLINK_WaitValue 0, 0, 20, 0.2, 2500;
WaitTime 0.1;

! Greifen mit den entsprechenden Parametern
GRIPLINK_FlexGrip 0, 12.5, 50, 25, 25, TRUE;
```

### 4.13 GRIPLINK\_LED: LED-Visualisierung wählen

Dieser Befehl konfiguriert die LED-Statusanzeige eines an einem bestimmten GRIPLINK-Port angeschlossenen Geräts. Unterstützte Geräte verfügen über mehrere konfigurierbare Sets von Einstellungen, zwischen denen mit diesem Kommando gewählt werden kann.



Weitere Informationen zur LED-Visualisierung und deren Konfigurationsmöglichkeiten finden Sie in der Betriebsanleitung des entsprechenden Geräts.



Diese Funktion ist nicht bei allen Geräten verfügbar!

#### RAPID-Befehl

```
PROC GRIPLINK_LED (  
    num nPort,  
    num nIndex  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
nIndex	num	Index des LED-Muster-Rezepts

#### Wizard-Block

Select LED visualization preset  on device at port

#### Beispiel

Auswahl der LED-Visualisierung 2 des Geräts am Port 3 eines GRIPLINK-Controllers:

```
GRIPLINK_LED 3, 2;
```

## 4.14 GRIPLINK\_Clamp: Mechanische Greifkraftherhaltung einstellen

Dieser Befehl aktiviert oder deaktiviert die mechanische Greifkraftherhaltung eines an einem bestimmten GRIPLINK-Port angeschlossenen Greifers. Durch diese Greifkraftherhaltung ist es möglich, die Motorregelung während längerem Greifen von Bauteilen abzuschalten und das Bauteil dennoch weiterhin sicher zu halten.



Weitere Informationen zur mechanischen Greifkraftherhaltung finden Sie in der Bedienungsanleitung des entsprechenden Greifers.



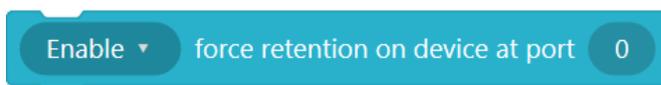
Diese Funktion ist nicht bei allen Geräten verfügbar!

### RAPID-Befehl

```
PROC GRIPLINK_Clamp(  
    num nPort,  
    bool bEnable  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
bEnable	bool	TRUE: mechanische Greifkraftherhaltung aktivieren FALSE: mechanische Greifkraftherhaltung deaktivieren

### Wizard-Block



### Beispiel

Greifen eines Werkstücks mit dem Greifer am Port 0 eines GRIPLINK-Controllers für eine längere Zeit bei aktivierter mechanischer Greifkraftherhaltung:

```
! Aktivieren der Greifkraftherhaltung  
GRIPLINK_Clamp 0, TRUE;
```

```
! Greifen mit Greifer an Port 0 und Rezept 1  
GRIPLINK_Grip 0, 1, TRUE;
```

```
! Längere Zeit warten, Greifkraftherhaltung wird aktiv  
WaitTime 600;
```

## 4.15 GRIPLINK\_WaitForStateTransition: Auf Zustandsänderung warten

Dieser Befehl wartet auf eine Änderung des Gerätezustands des Geräts an einem bestimmten GRIPLINK-Port. Er kann beispielsweise verwendet werden, um das Ende eines Greif- oder Freigabeprozesses zu detektieren. Nach erfolgtem Zustandswechsel wird der neue Gerätezustand zurückgegeben. Tritt keine Zustandsänderung ein, wird nach einer gewissen Zeit ein Fehler ausgelöst.



Das Warten auf eine Zustandsänderung kann alternativ auch direkt in den Greif- und Freigabebefehlen ausgelöst werden.

### RAPID-Befehl

```
PROC GRIPLINK_WaitForStateTransition(  
    num nPort,  
    \VAR num nDevState  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
\nDevState	num	optionale Variable für den Rückgabewert

Rückgabewert	Typ	Bedeutung
nGRIPLINK_State	num	Geräte-Zustand nach Zustandswechsel



Wird das optionale Argument \nDevState spezifiziert, wird der Rückgabewert nur in diese Variable geschrieben.

### Wizard-Block

Wait for state transition on device at port 0

## Beispiel

Gleichzeitiges Greifen mit zwei Greifern an den Ports 1 und 2 eines GRIPLINK-Controllers. Dabei soll das Programm erst weiter ausgeführt werden, wenn beide Greifprozesse erfolgreich abgeschlossen sind:

```
! Variablen für Gerätezustand anlegen
VAR num nState1;
VAR num nState2;

! Gleichzeitig Greifen (ohne auf Zustandsänderung zu warten)
GRIPLINK_Grip 1, 0, FALSE;
GRIPLINK_Grip 2, 0, FALSE;

! Auf Zustandsänderungen der Greifer warten
GRIPLINK_WaitForStateTransition 1, \nDevState := nState1;
GRIPLINK_WaitForStateTransition 2, \nDevState := nState2;

! Neue Zustände der Greifer prüfen
IF nState1 <> GRIPLINK_DS_HOLDING THEN
  ! Werkstück 1 nicht gegriffen
ENDIF
IF nState2 <> GRIPLINK_DS_HOLDING THEN
  ! Werkstück 2 nicht gegriffen
ENDIF
```

## 4.16 GRIPLINK\_SetValue: Gerätewert setzen

Dieser Befehl setzt einen Wert auf einem Gerät an einem bestimmten GRIPLINK-Port. Die verfügbaren Werte werden durch einen zugehörigen Index ausgewählt. Der genaue Effekt dieses Kommandos hängt vom jeweiligen Gerät ab.



Eine Übersicht über die bei verschiedenen Geräten verfügbaren Werte und deren Indices finden Sie in der auf [www.griplink.de](http://www.griplink.de) unter *Downloads* verfügbaren Tabelle.



Diese Funktion ist nicht bei allen Geräten verfügbar!

### RAPID-Befehl

```
PROC GRIPLINK_SetValue(  
    num nPort,  
    num nIndex,  
    num nValue  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
nIndex	num	Index des zu setzenden Werts
nValue	num	zu setzender Wert

### Wizard-Block

Set value with index  of device at port  to

### Beispiel

Setzen des Wertes mit dem Index 0 am mit Port 1 eines GRIPLINK-Controllers verbundenen Gerät auf 42:

```
GRIPLINK_SetValue 0, 1, 42;
```

## 4.17 GRIPLINK\_WaitValue: Auf Gerätewert warten

Dieser Befehl wartet darauf, dass ein angegebener Wert auf einem Gerät an einem bestimmten GRIPLINK-Port erreicht wird. Konkret wird für eine gewisse Zeit gewartet, bis der Gerätewert in einen gewissen Bereich (Fenster) um den Zielwert (Schwellenwert) eintritt. Bei Zeitüberschreitung wird ein Fehler ausgelöst.



Eine Übersicht über die bei verschiedenen Geräten verfügbaren Werte und deren Indices finden Sie in der auf [www.griplink.de](http://www.griplink.de) unter *Downloads* verfügbaren Tabelle.



Die Fenstergröße sollte je nach Änderungsrate des Gerätewerts passend gewählt werden, um eine zuverlässige und genaue Erkennung zu ermöglichen.

### RAPID-Befehl

```
PROC GRIPLINK_WaitValue(  
    num nPort,  
    num nIndex,  
    num nValue,  
    num nWindow,  
    num nTimeout  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
nIndex	num	Index des zu setzenden Werts
nValue	num	Schwellenwert, der erreicht werden soll
nWindow	num	Größe des Fensters um den Schwellenwert, innerhalb dessen der Wert als erreicht gilt
nTimeout	num	maximale Dauer in ms, die auf das Erreichen des Schwellenwerts gewartet wird

### Wizard-Block

Wait up to  ms until value with index  on device at port

reaches window of size  around

**Beispiel**

Warten bis der Wert mit Index 1 des Geräts am Port 0 eines GRIPLINK-Controllers den Wert  $10 \pm 1,5$  erreicht. Dabei soll maximal 20 Sekunden gewartet werden:

```
GRIPLINK_WaitValue 0, 1, 10, 1.5, 20000;
```

## 4.18 GRIPLINK\_DeviceState: Gerätezustand abfragen

Dieser Befehl liest den aktuellen Gerätezustand des Geräts an einem bestimmten GRIPLINK-Port aus. Dieser Zustand gibt Aufschluss über den aktuellen Status des Geräts.



Eine Tabelle mit einer Übersicht der möglichen Gerätezustände und den entsprechenden vordefinierten Konstanten finden Sie im Abschnitt **Gerätezustände** (3.4).

### RAPID-Befehl

```
PROC GRIPLINK_DeviceState(  
    num nPort,  
    \VAR num nDevState  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
\nDevState	num	optionale Variable für den Rückgabewert

Rückgabewert	Typ	Bedeutung
nGRIPLINK_State	num	aktueller Geräte-Zustand



Wird das optionale Argument \nDevState spezifiziert, wird der Rückgabewert nur in diese Variable geschrieben.

### Wizard-Block

Get state of device at port

## Beispiel

Abfragen des Zustands eines Greifers am Port 3 eines GRIPLINK-Controllers. Der weitere Ablauf des Programms soll dann vom Gerätezustand abhängen:

```
! Abfragen des Gerätezustands
GRIPLINK_DeviceState 3;

! Prüfen des Gerätezustands
IF nGRIPLINK_State = GRIPLINK_DS_HOLDING THEN
  ! Werkstück gegriffen
ELSEIF nGRIPLINK_State = GRIPLINK_DS_NO_PART THEN
  ! kein Werkstück gegriffen
ELSEIF nGRIPLINK_State = GRIPLINK_DS_RELEASED THEN
  ! Werkstück freigegeben
ENDIF
```

## 4.19 GRIPLINK\_Value: Gerätewert abfragen

Dieser Befehl liest einen Gerätewert von einem Gerät an einem bestimmten GRIPLINK-Port aus. Die verfügbaren Werte werden durch einen zugehörigen Index ausgewählt.



Eine Übersicht über die bei verschiedenen Geräten verfügbaren Werte und deren Indices finden Sie in der auf [www.griplink.de](http://www.griplink.de) unter *Downloads* verfügbaren Tabelle.

### RAPID-Befehl

```
PROC GRIPLINK_Value (  
    num nPort,  
    num nIndex,  
    \VAR num nValue  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
nIndex	num	Index des auszulesenden Gerätewerts
\nValue	num	optionale Variable für den Rückgabewert

Rückgabewert	Typ	Bedeutung
nGRIPLINK_Value	num	aktueller Gerätewert



Wird das optionale Argument \nValue spezifiziert, wird der Rückgabewert nur in diese Variable geschrieben.

### Wizard-Block

Get value with index  of device at port

## Beispiel

Abfragen der Größe eines gegriffenen Werkstücks mit einem Greifer von WEISS ROBOTICS am Port 0 eines GRIPLINK-Controllers. Dafür wird der Gerätwert mit Index 0 (Öffnungsweite in mm) verwendet. Der weitere Programmablauf soll von der ermittelten Größe abhängig sein:

```
! Greifen mit Greifer an Port 0 und Rezept 0
GRIPLINK_Grip 0, 0, TRUE;

! Prüfen des neuen Gerätezustands
IF nGRIPLINK_State = GRIPLINK_DS_HOLDING THEN

    ! Werkstück gegriffen, Fingerposition abfragen
    GRIPLINK_Value 0, 0;

    ! Fingerposition prüfen
    IF nGRIPLINK_Value > 9.5 AND nGRIPLINK_Value < 10.5 THEN
        ! Korrektes Werkstück gegriffen
    ELSE
        ! Falsches Werkstück gegriffen
    ENDIF

ELSE
    ! kein Werkstück gegriffen
ENDIF
```

## 4.20 GRIPLINK\_SetGripConfig: Greifparameter konfigurieren

Dieser Befehl setzt ein Set von Greifparametern für ein Gerät an einem bestimmten GRIPLINK-Port. Greifmodule verfügen über mehrere, frei konfigurierbare Sätze von Parametern, die über einen Index ausgewählt werden. Diese Sets bestehen aus bis zu acht Parametern und einem Tag zur Identifikation und Beschreibung.



Informationen zur Bedeutung der einzelnen Parameter auf dem jeweiligen Gerät und den zulässigen Wertebereichen finden Sie in der Bedienungsanleitung des Geräts, sowie im entsprechenden GRIPLINK-Gerätetreiber.



Zur Konfiguration der Parameter der Greif-Rezepte kann alternativ auch die Weboberfläche des GRIPLINK-Controllers verwendet werden.

Bei Greifern von WEISS ROBOTICS haben die Greifparameter folgende Bedeutungen:

- 0 Zielposition beim Greifen (No-Part-Limit) in mm
- 1 Zielposition beim Freigeben (Release-Limit) in mm
- 2 Kraft-Faktor in %



Weitere optionale Parameter werden bei Greifern der WPG-Serie verwendet:

- 3 Greifgeschwindigkeit in mm/s
- 4 Greifbeschleunigung in mm/s<sup>2</sup>
- 5 Freigabegeschwindigkeit in mm/s
- 6 Freigabebeschleunigung in mm/s<sup>2</sup>

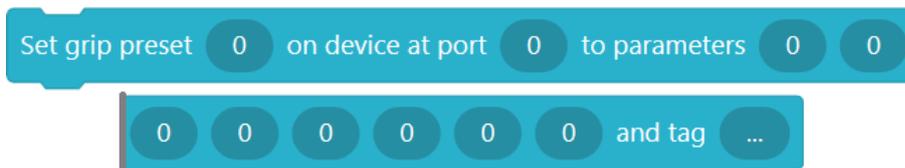
Nicht verwendete Parameter werden einfach auf 0 gesetzt.

### RAPID-Befehl

```
PROC GRIPLINK_SetGripConfig(  
    num nPort,  
    num nIndex,  
    string stTag,  
    num nParam0,  
    num nParam1,  
    num nParam2,  
    num nParam3,  
    num nParam4,  
    num nParam5,  
    num nParam6,  
    num nParam7  
)
```

Parameter	Typ	Bedeutung
nPort	num	GRIPLINK-Port
nIndex	num	Index des zu setzenden Greif-Rezepts
stTag	string	zu setzender Rezept-Tag
nParamX	num	zu setzender Parameterwert (0: nicht verwendet)

### Wizard-Block



### Beispiel

Konfigurieren des Rezepts 1 eines Greifers von WEISS ROBOTICS am Port 0 eines GRIPLINK-Controllers. Dabei sollen ein No-Part-Limit von 3,5 mm, ein Release-Limit von 12 mm und eine Greifkraft von 50 % eingestellt werden:

```
GRIPLINK_SetGripConfig 0, 1, "Grip_1", 3.5, 12, 50, 0, 0, 0, 0, 0;
```



© 2025 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.