



GRIPLINK PLUG-IN FOR ABB

Version 1.0.5

May 2024



Table of contents

1	Introduction	2
1.1	Notation and symbols	2
1.2	Intended use	2
1.3	System requirements	2
1.4	License terms	3
2	Installation	4
2.1	Hardware Installation	4
2.2	Add-In Installation	4
2.3	Add the Add-In to an existing project	5
2.4	Rapid Sockets	8
2.5	Network configuration	9
3	Command Set Reference	10
3.1	Open connection – CONNECT	11
3.2	Close Connection – DISCONNECT	12
3.3	Enable device – ENABLE	13
3.4	Disable device – DISABLE	14
3.5	Get Device State – DEVSTATE	15
3.6	Reference gripper – HOME	16
3.7	Grip part – GRIP	17
3.8	Grip part with multiple grippers – MGRIP	18
3.9	Release part – RELEASE	19
3.10	Release part with multiple grippers – MRELEASE	20
3.11	Wait for state transition and return state – WSTR	21
3.12	Wait for multiple state transitions – MWAITFOR	22
3.13	Get sensor value – VALUE	23
3.14	Control gripping force retention – CLAMP	24
3.15	Control of the LED display – LED	25
3.16	Configure grip settings – GRIPCFG	26
3.17	Assert device type – DEVASSERT	27
4	Wizard Easy Programming	28
5	Error Handling	30
6	Device States	31

1 Introduction

With the GRIPLINK technology, servo-electric and smart pneumatic gripping modules from WEISS ROBOTICS can be controlled by robot controllers of all leading robot brands via simple TCP/IP network connections.

The GRIPLINK plug-in for ABB is the software link between the GRIPLINK interface converter and the robot controller and enables the easy integration of WEISS ROBOTICS' GRIPLINK technology into ABB robot systems.



This manual describes the functions of the GRIPLINK plug-in. For information about installation and operation of the GRIPLINK interface converter please refer to the GRIPLINK user's manual. The manual can be found online on www.griplink.de/manuals

1.1 Notation and symbols

For a better understanding, the following symbols are used in this manual:



Functional or safety relevant information. Non-compliance may endanger the safety of personnel and the system, damage the device or impair its function.



Additional information for a better understanding of the described facts.



Reference to further information.

1.2 Intended use

The software "GRIPLINK plug-in" is intended for communication between the GRIPLINK interface converter from WEISS ROBOTICS and a robot controller. The requirements of the applicable guidelines as well as the installation and operation instructions in this manual must be noted and adhered to. Any other use or use beyond that is considered improper use. The manufacturer is not liable for any damage resulting from this.

1.3 System requirements

To run the GRIPLINK plug-in, the following ABB products are required:

- ABB Robot Controller with RobotWare 7.1 or newer
- ABB RobotStudio® programming environment, version 2022.3 or newer



Please contact ABB or your ABB distributor in order to purchase these products.

1.4 License terms

The GRIPLINK plug-in is protected by copyright. The software package includes the applicable license terms. By installing and using the GRIPLINK plug-in, the user accepts these license terms.

2 Installation

2.1 Hardware Installation

Installing the hardware strongly depends on the type of robot and the robot controller. In general, the following steps are necessary:

1. Attach the mounting plate that comes with the gripper to the robot flange using the screws and centering pins included. Then mount the gripper onto the mounting plate.
2. Connect the electrical connector (4-pin) of the gripper with the appropriate connector (usually 8-pin) on the robot flange using the included cable.
For some robots or grippers, it is necessary to use an external cable instead that must be attached to the outside of the robot arm.
3. When using the internal cable routing, connect the appropriate connector of the robot controller to the GRIPLINK using the included cable. For external wiring, connect the gripper cable directly to the GRIPLINK.
4. Connect the GRIPLINK's Ethernet port (ETH0 or ETH1) to one of the robot controller's Ethernet ports (usually the WAN port must be used). If you intend to access the GRIPLINK's web interface while it is connected to the robot, you can use a common Ethernet Switch in between.
5. Connect the GRIPLINK's power supply port (PWR IN) to a power supply that is sufficient to supply the gripper and the GRIPLINK. Please refer to the device manuals to find out about the total power consumption of your system.



Weiss Robotics recommends using an external power supply unit to power the GRIPLINK Controller or WPG.

Using the robot controller's power supply may lead to unexpected behaviour. Also refer to the GRIPLINK Controller or WPG manual for detailed information.

2.2 Add-In Installation

To operate the GRIPLINK, the GRIPLINK Add-In for ABB RobotStudio® provided by WEISS ROBOTICS is required on the robot controller. To install the GRIPLINK Add-In, please execute the following steps:



Please make sure the most recent version of the GRIPLINK plug-in is used. It can be downloaded from www.griplink.de/software.

1. Open the ABB RobotStudio® software. Choose the ribbon "Add-Ins". Note that you don't need to open a project to install add-ins (cf. Figure 1).

3. Select the controller of your previously opened or created project. This can also be a virtual controller. Note that you might have to adjust the search path for virtual controllers in the preferences on the left (Figure 3).
4. After you've selected the appropriate controller, click on the "Edit" button on the bottom right side of the Installation Manager window (Figure 3).

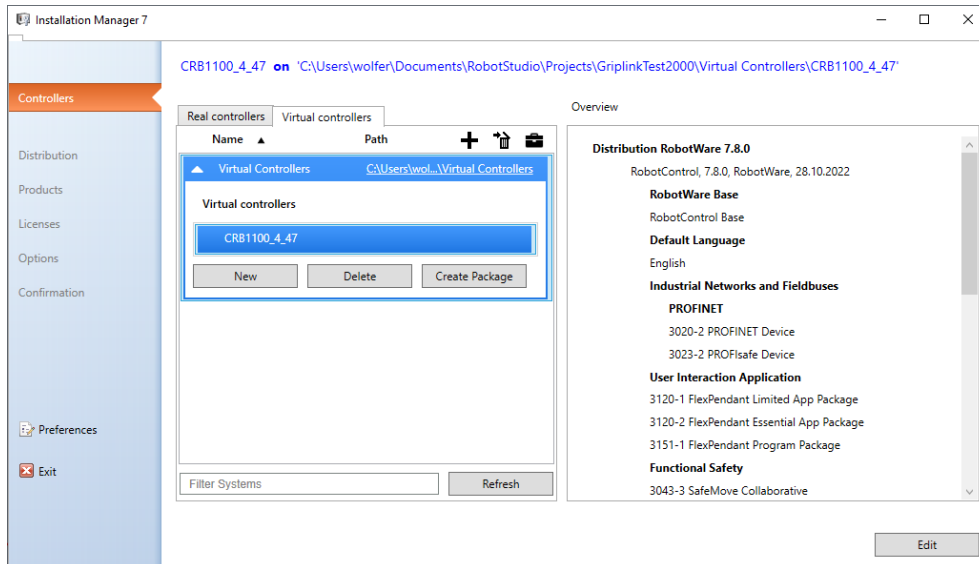


Figure 3

5. Click on the "Add" button to add additional packages to the controller (Figure 4)

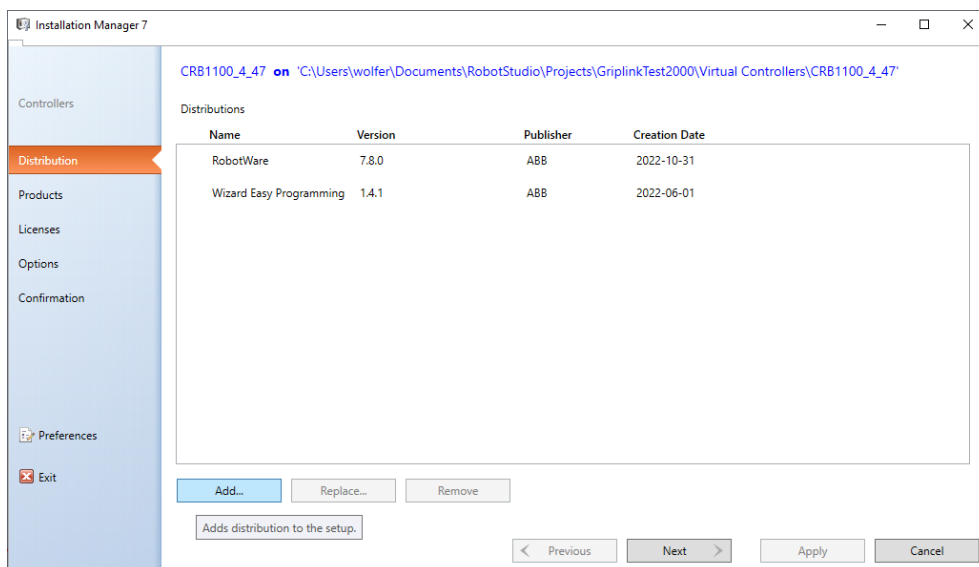


Figure 4

6. Choose "Griplink" and click "OK" (Figure 5)

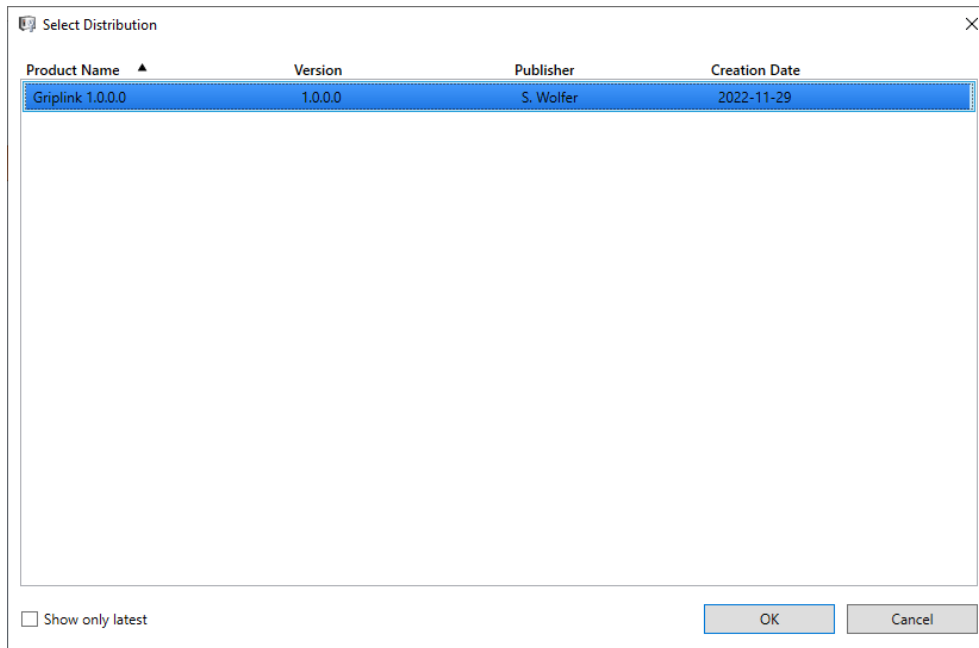


Figure 5

7. Select “Next” until you get to the “Options” panel. Select “Applications” from the tab list and make sure “Griplink” is selected. Click next and finish the dialogue. Close the Installation Manager (Figure 6).

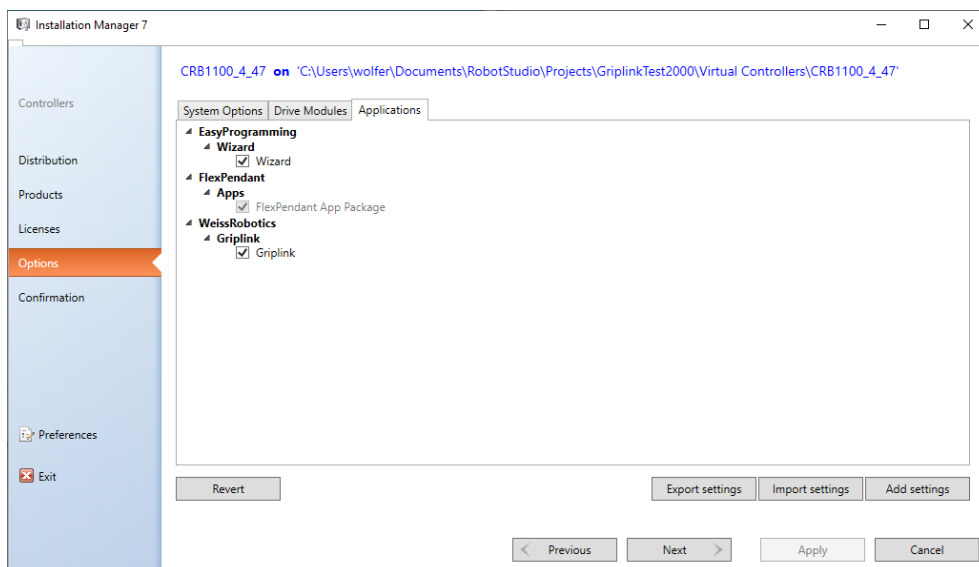


Figure 6

8. For the new settings to take effect, you’ll have to restart your controller (Figure 7). After that, the functions described below will be available for use.

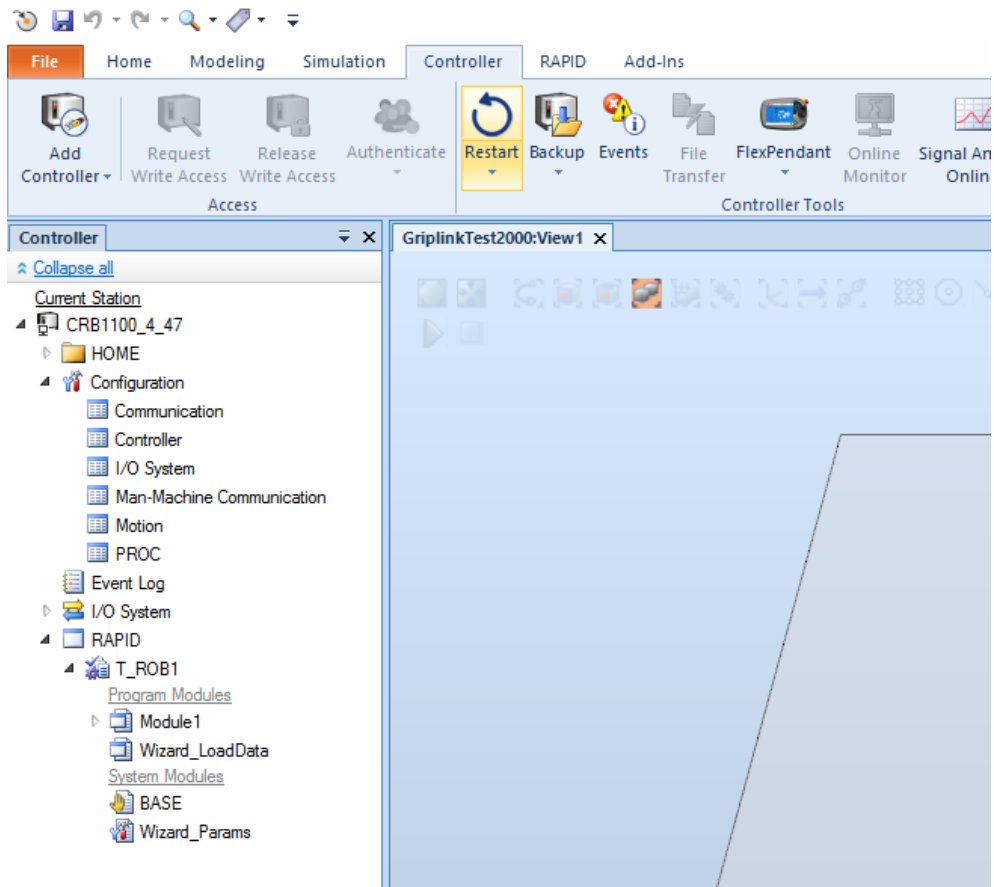


Figure 7: Restart controller

2.4 Rapid Sockets

The GRIPLINK Add-In uses Rapid Sockets to establish a TCP/IP networking connection between the robot controller and the GRIPLINK. In order to use Rapid Sockets, the firewall settings on the robot controller have to be adjusted manually. Using ABB RobotStudio®, find the firewall settings in “Configuration” -> “Communication” -> “Firewall Manager”. Make sure Rapid Sockets are enabled for the appropriate type of network as displayed in Figure 8.

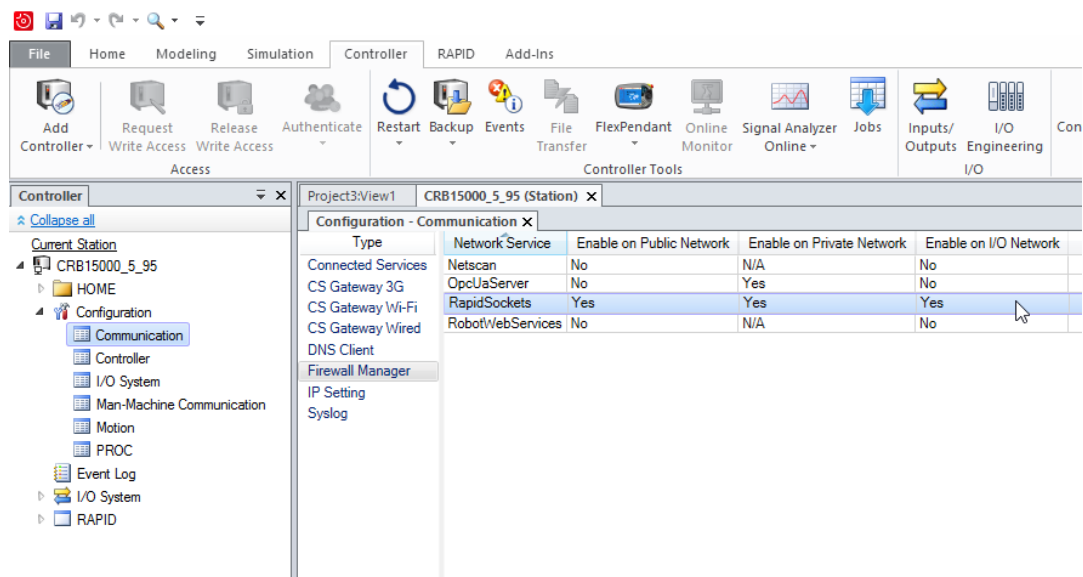


Figure 8: Enabling Rapid Sockets

2.5 Network configuration

The connection between the GRIPLINK and the robot controller is established via TCP/IP networking. To use TCP/IP networking, the robot controller must have an IP address that is compatible with the address assigned to the GRIPLINK. Please refer to the documentation of your ABB robot system to find out how to set an appropriate IP address for your robot controller. When using the simulation in ABB RobotStudio®, the IP address of your computer's network interface will be used to connect to the GRIPLINK. In this case, please make sure the GRIPLINK is connected to your computer and can be accessed via its web interface.

The IP address of the GRIPLINK is set to 192.168.1.40 by default and can be adjusted via its web interface. To change the IP address, connect the GRIPLINK to a PC or laptop and open the web interface in your favorite browser by entering <http://192.168.1.40> into the address bar. You can access the IP settings by pressing the "Config" button.

The GRIPLINK command interface accepts incoming connections on port 10001 (TCP).



For more information about the configuration of the GRIPLINK, please refer to the associated user's manual.



The IP addresses of both the robot controller and the GRIPLINK interface converter must be within the same subnet range. Please contact your network administrator if you experience any problems assigning appropriate IP addresses to the devices.

3 Command Set Reference

The GRIPLINK plug-in provides a number of functions to control gripping modules from WEISS ROBOTICS as well as sensors and actors from several third-party vendors. Both single and multi-gripper commands are available.

Multi-gripper commands

With the multi-gripper commands, several actors can be addressed at once. These commands are particularly suitable for handling large or flexible workpieces with multiple gripping modules simultaneously.

The basic program flow with the GRIPLINK plug-in is always as follows:

1. Establish connection with CONNECT
2. Depending on device: Activate with ENABLE
3. For servo gripping modules without absolute encoder: Reference gripping module with HOME / MHOME
4. Gripping / releasing with GRIP / MGRIP or RELEASE / MRELEASE

The following chapters describe the available commands provided by the plug-in in detail.

Procedure/Function	Description
WRO_GriplinkConnect	Open connection to GRIPLINK
WRO_GriplinkDisconnect	Close connection GRIPLINK
WRO_GriplinkDevAssert	Make sure a specific device is connected to the selected GRIPLINK port
WRO_GriplinkEnable	Enable device
WRO_GriplinkDisable	Disable device
WRO_GriplinkDevState	Query device state
WRO_GriplinkValue	Query sensor value(s)
WRO_GriplinkHome	Reference gripping module
WRO_GriplinkGrip	Grip workpiece
WRO_GriplinkMultiGrip	Grip workpiece with multiple grippers
WRO_GriplinkRelease	Release workpiece
WRO_GriplinkMultiRelease	Release workpiece with multiple grippers
WRO_GriplinkWaitFor	Wait for state transition
WRO_GriplinkMultiWaitFor	Wait for multiple state transitions
WRO_GriplinkClamp	Control gripping force retention PERMAGRIP®
WRO_GriplinkLED	Control LED display
WRO_GriplinkGripCfg	Set grip configuration

3.1 Open connection – CONNECT

This command establishes the connection between the GRIPLINK interface converter and the robot controller. It will also check the command protocol version supported and will raise an error on protocol version mismatch.



The IP address of the GRIPLINK can be changed via the web interface.

Signature

```
PROC WRO_GriplinkConnect( string stIPAddr )
```

Parameters

stIPAddr	IP address of the GRIPLINK interface converter as string
----------	--

Return value

-

Example

Open a connection to the GRIPLINK with IP address 192.168.1.40:

```
WRO_GriplinkConnect "192.168.1.40";
```

3.2 Close Connection – DISCONNECT

This command closes a connection that has been established previously by the *WRO_GriplinkConnect()* procedure.

Signature

```
PROC WRO_GriplinkDisconnect()
```

Parameters

-

Return value

-

Example

Disconnect GRIPLINK from the robot controller:

```
WRO_GriplinkDisconnect;
```

3.3 Enable device – ENABLE

This command activates the device connected to the given port of the GRIPLINK. Note that not all devices have to be enabled explicitly before use. The command, however, can be useful to re-enable a device that previously has been disabled with the DISABLE command to acknowledge an error.

Signature

```
PROC WRO_GriplinkEnable( num nDevIdx )
```

Parameters

nDevIdx	Port index (0 to 31)
---------	----------------------

Return value

-

Example

Enable device connected at GRIPLINK port 0:

```
WRO_GriplinkEnable 0;
```

3.4 Disable device – DISABLE

This command deactivates the device connected to the given port of the GRIPLINK. This function can be used, for example, when using an automatic tool changer. If a gripping module from WEISS ROBOTICS is connected, it deactivates the drive of the selected gripping module, e.g. when using a tool changer. On some devices, the DISABLE command will acknowledge error states. Use the ENABLE command to re-enable the device again.

Signature

```
PROC WRO_GriplinkDisable( num nDevIdx )
```

Parameters

nDevIdx	Port index (0 to 31)
---------	----------------------

Return value

-

Example

Change gripping module at port 1:

```
! Open connection between robot controller and GRIPLINK
WRO_GriplinkConnect "192.168.1.40";
! Enable drive for gripping module at port 1
WRO_GriplinkEnable 1;
! Execute program (e. g. pick & place)
! ...
! Disable gripping module at port 1
WRO_GriplinkDisable 1;
! Now the gripping module can be removed.
! ...
! Enable new gripping module
WRO_GriplinkEnable 1;
! Continue with program
! ...
```

3.5 Get Device State – DEVSTATE

This function returns the device state of the selected device. The device state is provided as an integer value. To simplify and improve readability, pre-defined constants are available that can be used when processing device states.

Signature

```
PROC WRO_GriplinkDevState( num nDevIdx, VAR num nDevState )
```

Parameters

nDevIdx	Port index (0 to 31)
nDevState	Variable (reference) to save device state to

Return value

Current device state (cf. chapter 6)

Example

Wait until the device state of the gripping module at port 2 changes to HOLDING (4):

```
! Query gripping state of gripping module at port 2
VAR num nDevState;
! The numeric constant DS_HOLDING is defined in the Griplink program
! module
WRO_GriplinkDevState 0, nDevState;
WHILE nDevState <> DS_HOLDING DO
    WRO_GriplinkDevState 0, nDevState;
    IF nDevState = DS_FAULT THEN
        ! An error has occurred
    ENDIF
ENDWHILE
```

3.6 Reference gripper – HOME

References the selected servo gripper. The command executes a reference run of the gripping module and waits until it's completed. After the command has been executed, the fingers of the gripping module are without force and must be moved to a defined position by using *WRO_GriplinkGrip()* / *WRO_GriplinkMultiGrip()* or *WRO_GriplinkRelease()* / *WRO_GriplinkMultiRelease()*.



The direction of the reference run can be configured via the web interface of the GRIPLINK interface converter.

Signature

```
PROC WRO_GriplinkHome( num nDevIdx )
```

Parameters

nDevIdx	Port index (0 to 31)
---------	----------------------

Return value

-

Example

Reference gripping module at port 3:

```
! Reference gripping module at port 3  
WRO_GriplinkHome 3;
```

Device state polling to check, if the homing sequence finished successfully or with error:

```
! Reference gripping module at port 0  
WRO_GriplinkHome 0;  
  
! Wait for gripper in state DS_DISABLED or DS_FAULT  
WRO_GriplinkDevState 0, nDevState;  
WHILE ( nDevState <> DS_DISABLED and nDevState <> DS_FAULT ) DO  
    WaitTime 0.05;  
    WRO_GriplinkDevState 0, nDevState;  
  
    ! Note: Additional timeout handling might be required here  
ENDWHILE  
  
! Check device state DS_FAULT  
IF nDevState == DS_FAULT THEN  
    ! Error: Gripper in fault state!  
    ! Stop program execution and notify user here!  
ENDIF  
  
! At this point, the gripper is referenced and ready to be used
```

3.7 Grip part – GRIP

Grips a workpiece with the selected gripping module and grip preset. The command waits until the gripping state changes to either HOLDING or NO PART.



The gripping parameters can be configured via the web interface of the GRIPLINK interface converter.

Signature

```
PROC WRO_GriplinkGrip( num nDevIdx, num nGripIdx )
```

Parameters

nDevIdx	Port index (0 to 31)
nGripIdx	Grip preset to be executed (range depends on gripper type)

Return value

-

Example

Gripping module at port 0 shall execute grip preset 2. After gripping, check device state:

```
VAR num nDevState;  
! Grip part  
WRO_GriplinkGrip 0, 2;  
! Check device state  
WRO_GriplinkDevState 0, nDevState;  
! The constants DS_NO_PART and DS_HOLDING are defined in the  
! Griplink program module  
IF nDevState = DS_NO_PART THEN  
    ! No part found  
ELSEIF nDevState = DS_HOLDING THEN  
    ! Part gripped  
ELSE  
    ! Unexpected grip state  
ENDIF
```

3.8 Grip part with multiple grippers – MGRIP

This function executes a grip command with the selected gripping modules. The function waits until all gripping modules have reached either HOLDING or NO PART state.



The gripping parameters can be configured via the web interface of the GRIPLINK interface converter.

Signature

```
PROC WRO_GriplinkMultiGrip( bool bSelected{*}, num nGripIdx )
```

Parameters

bSelected{*}	Array of boolean values selecting the desired devices that should execute the GRIP command
nGripIdx	Grip preset to be executed on the bSelected grippers

Return value

-

Example

Gripping modules at port 1 and 2 shall grip a workpiece. Display a message if both workpieces were gripped correctly:

```
VAR bool bSelected{4};
VAR num nDevStates{4};
! Select devices 2 and 3 (1 and 2 in GRIPLINK internal counting)
bSelected := [FALSE, TRUE, TRUE, FALSE];
! Grip with selected devices
WRO_GriplinkMultiGrip bSelected, 2;
FOR i FROM 1 TO Dim( bSelected, 1 ) DO
    ! Get device state. Note that the GRIPLINK starts counting
    ! ports from 0 whereas RAPID array indices start with 1
    WRO_GriplinkDevState i - 1, nDevStates[i];
ENDFOR
! The constants DS_NO_PART and DS_HOLDING are defined in the
! Griplink program module
IF nDevStates{2} = DS_HOLDING AND nDevStates{3} = DS_HOLDING THEN
    ! Holding parts
ENDIF
```

3.9 Release part – RELEASE

Releases the workpiece previously gripped with the selected gripping module. The command waits until the workpiece has been released.



The gripping parameters can be configured via the web interface of the GRIPLINK interface converter.

Signature

```
PROC WRO_GriplinkRelease( num nDevIdx, num nGripIdx )
```

Parameters

nDevIdx	Port index (0 to 31)
nGripIdx	Selected grip preset (range depends on gripper type)

Return value

-

Example

Gripping module at port 0 shall release a workpiece previously gripped with grip preset 2:

```
VAR num nDevState;  
WRO_GriplinkRelease 0, 2;  
WRO_GriplinkDevState 0, nDevState;  
! The constant DS_RELEASED is defined in the Griplink program module  
IF nDevState = DS_RELEASED THEN  
    ! Part released  
ENDIF
```

3.10 Release part with multiple grippers – MRELEASE

Releases the workpiece(s) gripped with the selected gripping modules. The function waits until all gripping modules have reached the RELEASED state.



The gripping parameters can be configured via the web interface of the GRIPLINK interface converter.

Signature

```
PROC WRO_GriplinkMultiRelease( bool bSelected{*}, num nGripIdx )
```

Parameters

bSelected{*}	Array of boolean values selecting the desired devices that should execute the GRIP command
nGripIdx	Grip preset to be executed on the bSelected grippers

Return value

-

Example

Gripping module at ports 1, 2 and 3 release a workpiece previously gripped with grip preset 3:

```
VAR bool bSelected{4};  
bSelected := [TRUE, TRUE, TRUE, FALSE];  
WRO_GriplinkMultiRelease bSelected, 3;
```

3.11 Wait for state transition and return state – WSTR

This command waits for a state transition, e. .g. after issuing a motion command for a gripper device, and returns the new state after the state transition.

The robot program will be blocked waiting for at most 8 seconds. If there is no state transition within this time, an error will be raised and the robot program will be stopped.

Signature

```
PROC WRO_GriplinkWaitFor( num nDevIdx, VAR num nDevState )
```

Parameters

nDevIdx	Port index (0 to 31)
nDevState	New device state after state transition

Return value

-

Example

Read the current finger position of the gripper at port 0:

```
VAR num nDevState;  
! Grip part  
WRO_GriplinkGrip 0, 0;  
! Wait for state transition  
WRO_GriplinkWaitFor 0, nDevState;  
IF nDevState = DS_HOLDING THEN  
    ! Workpiece has been gripped, proceed  
ENDIF
```

3.12 Wait for multiple state transitions – MWAITFOR

This command waits for a state transition on the selected ports, e. .g. after issuing a motion command for a gripper device. Unlike the single WSTR command, it won't return the new state.

The robot program will be blocked waiting for at most 8 seconds. If there are no state transitions within this time, an error will be raised and the robot program will be stopped.

Signature

```
PROC WRO_GriplinkMultiWaitFor( bool bSelected{*} )
```

Parameters

bSelected{*}	Array of boolean values selecting the desired devices that should be waited for
--------------	---

Return value

-

Example

```
VAR num nDevState;
VAR bool bSelected{2};
! Select devices 1 and 2 (0 and 1 in GRIPLINK internal counting)
bSelected := [TRUE, TRUE];
! Grip with selected devices
WRO_GriplinkMultiGrip bSelected, 2;
! Block until a state transition has been detected on selected ports
WRO_GriplinkMultiWaitFor bSelected;
FOR i FROM 1 TO Dim( bSelected, 1 ) DO
    ! Get device state. Note that the GRIPLINK starts counting
    ! ports from 0 whereas RAPID array indices start with 1
    WRO_GriplinkDevState i - 1, nDevState;
    IF nDevState <> DS_HOLDING THEN
        ! Raise an error as workpiece has not been gripped
    ENDIF
ENDFOR
```

3.13 Get sensor value – VALUE

This command returns current measured value of a connected sensor. If a WEISS ROBOTICS gripping module is connected, the command can be used to read the current finger position in micrometers. If more than one sensor value per device is available, the desired value can be selected by using the parameter *nValIdx*.

For Weiss Robotics grippers, value index 0 returns the finger position in micrometers (μm).

Signature

```
PROC WRO_GriplinkValue( num nDevIdx, num nValIdx, VAR dnum dnValue )
```

Parameters

nDevIdx	Port index (0 to 31)
nValIdx	Value index (range depends on device)
dnValue	Reference to write read value to

Return value

Sensor value (e. g. Finger position in μm)

Example

Read the current finger position of the gripper at port 0:

```
VAR dnum dnPos_um;  
WRO_GriplinkValue 0, 0, dnPos_um;  
IF dnPos_um >= 80000 THEN  
    ! Finger position is greater than 80 mm  
ENDIF
```

3.14 Control gripping force retention – CLAMP

The innovative gripping force retention PERMAGRIP® developed by WEISS ROBOTICS keeps up the gripping force on the workpiece, even if the power supply to the gripping module is interrupted unexpectedly. PERMAGRIP® also enables permanent gripping without the gripping module overheating.

This command activates or deactivates the PERMAGRIP® gripping force retention for the selected gripping module.



This command is not necessary to keep up the gripping force in case the power supply gets interrupted. The gripping force retention in case of an interrupted power supply is always active.



PERMAGRIP® is not available on all gripping modules.

Signature

```
PROC WRO_GriplinkClamp( num nDevIdx, bool bEnable)
```

Parameters

nDevIdx	Port index (0 to 31)
bEnable	Enable (<i>TRUE</i>) oder disable (<i>FALSE</i>) gripping force retention

Return value

-

Example

Grip workpiece with grip preset 0 on the gripper connected at port 2. Activate PERMAGRIP if a workpiece has been detected:

```
VAR num nDevState;  
! Grip workpiece  
WRO_GriplinkGrip 2, 0;  
! The constant DS_HOLDING is defined in the Griplink program  
! module  
WRO_GriplinkDevState 2, nDevState;  
IF nDevState = DS_HOLDING THEN  
    ! Activate gripping force retention  
    WRO_GriplinkClamp 2, True;  
ENDIF
```

3.15 Control of the LED display – LED

This command changes the color and the pattern of the illuminated ring of a selected WEISS ROBOTICS CRG gripping module.



Light patterns and colors can be configured via the web interface of the GRIPLINK interface converter.



This function is only available for CRG gripping modules by WEISS ROBOTICS.

Signature

```
PROC WRO_GriplinkLED( num nDevIdx, num nLEDIdx )
```

Parameters

nDevIdx	Port index (0 to 31)
nLEDIdx	Index of the pre-configured light pattern (range 0 to 7)

Return value

-

Example

Use the gripping module at port 3 and change the color of the light ring according to the finger position. Display light pattern 0 if the finger position is greater than or equal to 8.1 mm, otherwise display light pattern 1:

```
VAR num nDevState;  
VAR dnum dnPos_um;  
WRO_GriplinkGrip 3, 0;  
WRO_GriplinkDevState 3, nDevState;  
IF nDevState = DS_HOLDING THEN  
    WRO_GriplinkValue3, 0, dnPos_um;  
    IF dnPos_um > 8100 THEN  
        ! Set light pattern 0  
        WRO_GriplinkLED 3, 0;  
    ELSE  
        ! Set light pattern 1  
        WRO_GriplinkLED 3, 1;  
    ENDIF  
ELSE  
    ! Failed to grip part  
ENDIF
```

3.16 Configure grip settings – GRIPCFG

Configure grip settings for the given device. Each preset consists of a tag string and 3 parameters that configure the grip. The tag string can be used to give the preset a meaningful name for later identification.



The parameters may have specific limits depending on the connected device. See the device manual for details

Signature

```
PROC WRO_GriplinkGripCfg( num nDevIdx, num nGripIdx, string stTag, dnum
dnParams{*} )
```

Parameters

nDevIdx	Port index (0 to 31)
nGripIdx	Grip to be executed (range depends on gripper type)
stTag	Label string
dnParams{*}	Array of 8 configuration parameters



For servo-electric gripping modules from Weiss Robotics, parameter 0 represents the no part limit in micrometers (μm), parameter 1 represents the release limit in micrometers (μm) and parameter 2 represents the gripping force in percent multiplied by a factor of 1000 (1% equals parameter value 1.000).

Return value

-

Example

Set the configuration for grip preset 2 of the gripper connected to port 1:

```
VAR dnum dnParams{8};
dnParams {1} := 2000; ! No Part Limit 2.0 mm
dnParams {2} := 10000; ! Release Limit 10.0 mm
dnParams {3} := 80000; ! Gripping force 80%
dnParams {4} := 0;
...
dnParams {8} := 0;
WRO_GriplinkGripCfg 1, 2, "Workpiece 1", dnParams;
```

3.17 Assert device type – DEVASSERT

Assert that a device with the specified Vendor and Device ID is connected to the selected port. The procedure will raise an error and stop the robot program if a device with a different VID or PID is connected.



The necessary parameters IO-Link Vendor ID and Device ID can be found in the device's manual.

Signature

```
PROC WRO_GriplinkAssertDevice( num nDevIdx, num nVID, num nPID )
```

Parameters

nDevIdx	Port index (0 to 31)
nVID	Device's IO-Link Vendor ID
nPID	Device's IO-Link Device ID

Return value

-

Example

Check, if a gripper of type IEG55-020 (Device ID 20) by Weiss Robotics (Vendor ID 815) is connected to port 2:

```
WRO_GriplinkDevAssert 2, 815, 20;
```

4 Wizard Easy Programming

GRIPLINK provides a set of nodes to support ABB's Wizard Easy Programming feature.

Available nodes can be listed by choosing "Griplink" from the menu on the left side of the screen.

The connection to the GRIPLINK will be opened automatically.

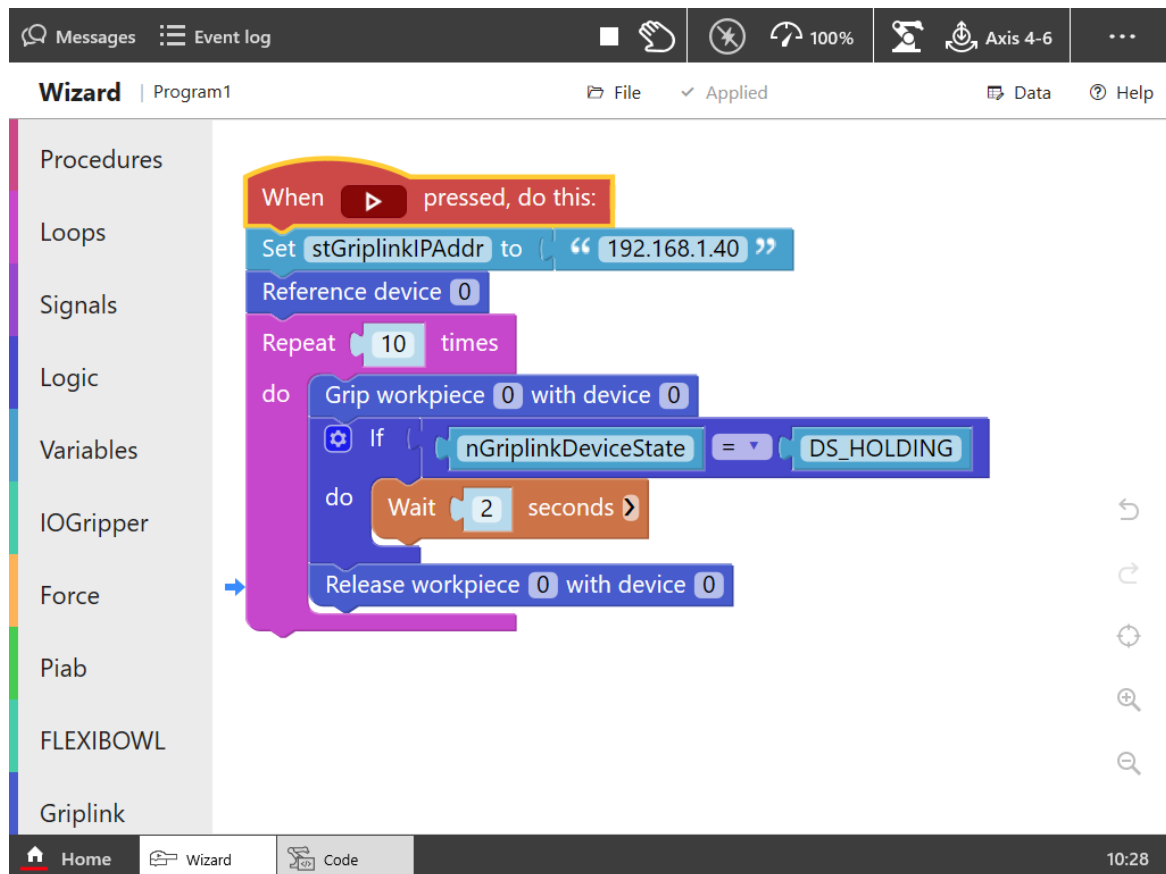


Figure 9: Wizard example

Global variables are available to evaluate the device state of a gripper or sensor connected to the GRIPLINK or to read sensor values. The variables are explained in Table 1.

Variable	Type	Description
stGriplinkIPAddr	String	IP address string representing the IP address of the GRIPLINK. Modify this variable if you changed the GRIPLINK's IP address.
nGriplinkDeviceState	num	This variable holds the device state after executing the last command. It can be used for example to check if a workpiece has been gripped successfully or if it has been released as intended. The variable will be updated automatically after executing the commands <i>Home</i> , <i>Grip</i> and <i>Release</i> . It can be updated anytime manually with the <i>Get device state</i> command.

		Note that when working with multiple devices, the variable will always contain the device state of the device last worked with.
nGriplinkValue	num	This variable holds the sensor value after executing the <i>Get value</i> command. It can be used for example to determine the grip position after gripping a workpiece.

Table 1: Global variables to be used with Wizard

5 Error Handling

If a problem occurs within the GRIPLINK plug-in, an error will be raised on the robot controller. If this error won't be caught by the user, the robot program will stop.



It is the responsibility of the programmer to handle these errors and to bring the system into a safe state after an error occurred. The ABB programming environment provides appropriate commands to catch errors.

In order to differentiate between different types of errors, the GRIPLINK plug-in uses several different error numbers from the robot controller's range of user-defined error numbers. These error numbers are registered using the *BookErrNo* command when calling the *WRO_GriplinkConnect* procedure. The robot controller will then automatically assign an error number from the range of user-defined errors. Table 1 shows the constants used for error numbers and their meaning.

Error	Description
GL_ERR_CONN_FAILED	Connection error
GL_ERR_CMD_FAILED	Failed to execute command
GL_ERR_DEVICE	Device error on connected sensor or gripping module (e. g. overtemperature)
GL_ERR_PARAM	Parameter error when calling a Griplink function (e. g. value out of range)
GL_ERR_TIMEOUT	Timeout error when executing a Griplink command.

Table 2: Error numbers defined by the GRIPLINK plug-in



For more information on error handling in RAPID programs, please refer to the documentation of your ABB robot controller.

6 Device States

Device State	Constant	Code	Description
INVALID	DS_INVALID	0	Port not connected
NOT INITIALIZED	DS_NOT_INITIALIZED	1	Device not initialized
DISABLED	DS_DISABLED	2	Ready for operation, not active
RELEASED	DS_RELEASED	3	Workpiece released
NO PART	DS_NO_PART	4	Workpiece not found
HOLDING	DS_HOLDING	5	Holding workpiece
ENABLED	DS_ENABLED	6	Ready for operation, active
FAULT	DS_FAULT	7	Error

© 2022 WEISS ROBOTICS GmbH & Co. KG. All rights reserved.

GRIPLINK and PERMAGRIP are registered trademarks of WEISS ROBOTICS GmbH & Co. KG. All weiteren other brands are the property of their respective owners.

The technical data mentioned in this document can be changed to improve our products without prior notice. Used trademarks are the property of their respective trademark owners. Our products are not intended for use in life support systems or systems whose failure can lead to personal injury.

