Software manual
2D Grasping Kit
SCHUNK software module FANUC CRX

Translation of original software manual

Hand in hand for tomorrow

Imprint

Technical changes:
We reserve the right to make alterations for the purpose of technical improvement.

Document number: 1592171

Version: 03.00 | 02/12/2025 | en

Dear Customer,

Thank you for trusting our products and our family-owned company, the leading technology supplier of robots and production machines.

Our team is always available to answer any questions on this product and other solutions. Ask us questions and challenge us. We will find a solution!

Best regards,

Your SCHUNK team

Customer Management

Tel. +49-7133-103-2503
Fax +49-7133-103-2189

cmg@de.schunk.com

Please read the operating manual in full and keep it close to the product.

Table of Contents

1 General

1.1 About this manual

This manual contains information about the SCHUNK software module for FANUC CRX cobots and how to use it.

The software is used for simple integration and control of the following products in FANUC CRX applications:

- 2D Grasping Kit

Definition of terms "Product"

The term "product" replaces the product names listed above in this manual.

This manual describes the software environment for a FANUC CRX robot.

NOTE: The illustrations in this manual are intended to provide a basic understanding and may deviate from the actual version.

In addition to these instructions, the documents listed under ▸ 1.7 [🗋 6] are applicable.

Validity

These instructions apply for the following software versions:

SVC Firmware Version	FANUC CRX version
>=3.0.0	V9.40P/56 Option: R648 User Socket Msg Option: R640 MROT (optional)

1.2 Target group

This manual is intended for robot integrators who have basic mechanical and electrical training skills and who are also familiar with elementary programming concepts.

Commissioning and troubleshooting may only be performed by qualified personnel with appropriate training.

The following knowledge is required:

- Basic knowledge of robotics
- Knowledge in handling FANUC robots

Electrical installations may only be carried out by a suitably trained electrician.

1.3 Symbol definition

The following symbols are used in this manual:

- ■ Prerequisite for an action
1. Action 1
2. Action 2
   - ⇨ Intermediate results
- ⇨ Final results

▸ 1.3 [🗋 5]: chapter number and [page number] in hyperlinks

1.4 Abbreviations

SVC	SCHUNK Vision Controller (Industrial PC)
AI	Artificial Intelligence
TCP	Tool Center Point
CMOS	Control Memory Operating System (memory)

1.5 Brands

- FANUC is a registered trademark of FANUC CORPORATION (Japan).

1.6 Presentation of Warning Labels

To make risks clear, the following signal words and symbols are used for safety notes.

⚠ DANGER

Dangers for persons!

Non-observance will inevitably cause irreversible injury or death.

⚠ WARNING

Dangers for persons!

Non-observance can lead to irreversible injury and even death.

⚠ CAUTION

Dangers for persons!

Non-observance can cause minor injuries.

NOTICE

Material damage!

Information about avoiding material damage.

1.7 Applicable documents

- Commissioning instructions for 2D Grasping Kit *
- Operating manual for FANUC CRX robots

The documents labeled with an asterisk (*) can be downloaded from **schunk.com/downloads**.

2 Functional description

The software module facilitates operation and creation of applications for SCHUNK products on a collaborative FANUC CRX robot.

All necessary controls are installed via the software module. After the installation is complete, the programming elements are deployed within the graphical user interface (GUI). The GUI supports the entire configuration and parameterization of SCHUNK products as well as the necessary control and programming options.

The following functions are available in the software module and can be used in a robot program:

- **Connect**: A connection is established to the SVC
- **Set Pre Pos**: Saves the robot configuration.
- **Set Reply Reg**: Configures the register in which the reply from the SVC is saved.
- **Grasp**: Starts object detection and receives the grasping pose for an object.
- **Feedback**: Sends feedback to SVC regarding the gripping action performed.
- **SetProject**: Requests a project change based on the project ID.
- **Set Tool**: Requests a tool change based on the tool ID.
- **Object Count**: Analyzes the scene. The number of all objects and the number of instances of a specified object class are determined.
- **Get State**: Queries the status of the SVC.
- **Robot Pose**: Transmits the current robot position to the SVC.
- **Timeout State**: Displays the status of the command timeout.
- **Conn State**: Sends feedback on the connection status.
- **View**: Starts the output of the image evaluated from the last gripping process.
- **Disconnect**: Ends the connection to the SVC.

Further Information on the functions ▶ 7 [🗋 16].

3 Connecting the product to the robot control system

Before connecting or commissioning the product, read the operating manual of the robot and observe the instructions in this manual!

⚠ WARNING

Risk of injury due to unexpected movements!

If the power supply is switched on or residual energy remains in the system, components can move unexpectedly and cause serious injuries.

- Before starting any work on the product: Switch off the power supply and secure against restarting.
- Make sure, that no residual energy remains in the system.

⚠ CAUTION

Risk of injury from electric shock due to contact with live parts!

- Follow the operating manual for the robot.
- Before starting any work on the product: Switch off the energy supply and secure against re-connection.

NOTE

Safety-relevant signals (e.g. emergency stop) must be wired externally, e.g. via safety relays, thus completely disconnecting the product from the power supply.

- Perform a risk assessment for the entire robotic application based on legal requirements to evaluate all safety-related aspects of the application.

- SVC (industrial PC) is connected to the robot via a network cable.
- IP address of the SVC is known (factory settings 192.168.1.76)
- IP address of the robot is in the same subnet (example 192.168.1.100)

- For information on connecting the product, see the commissioning instructions.

4 Installing the software module

NOTICE

Possible damage to product!

The product or the robot may get damaged if electrical cables are connected or disconnected during operation.

- Connect or disconnect electrical connections only when the device is switched off.

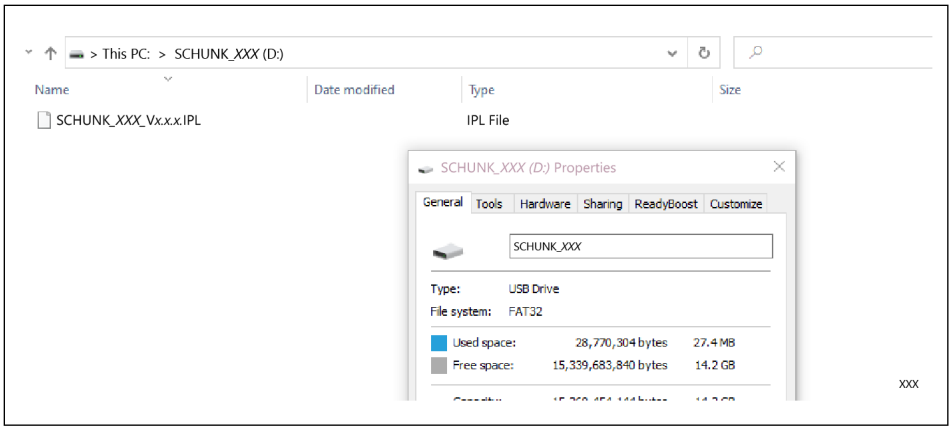
Preparing the installation

NOTE

A USB data carrier (type A) is required for installation.

The USB data carrier must comply with the following requirements:

- Formatted in FAT32 format
- Designation: "SCHUNK_*XXX*", (*XXX* = product type)



Installing

NOTE

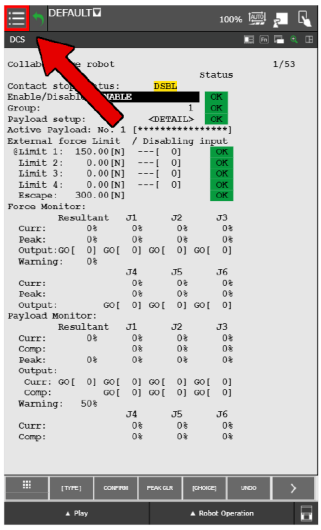
To avoid malfunctions, SCHUNK recommends installing the current version of the software module.

- Product is mounted and connected to the robot control system.

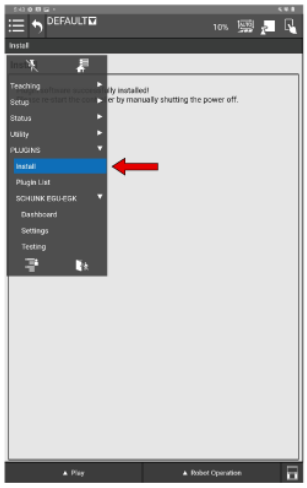
1. Download the current version of the software module at **schunk.com/downloads** and copy it to the USB stick.
   ⇨ The CRX control software must be compatible with the version of the software module. Information on this can be found in the download area.

2. Connect the USB stick to the robot control system (not to the USB port of the Tablet Teach Pendant).

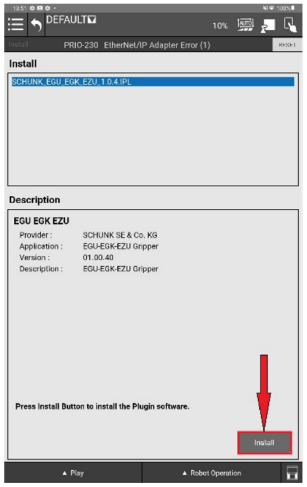
3. Select the "Menu" button at the top left of the Tablet Teach Pendant screen.



4. Select "PLUGINS" > "Install" in the menu.

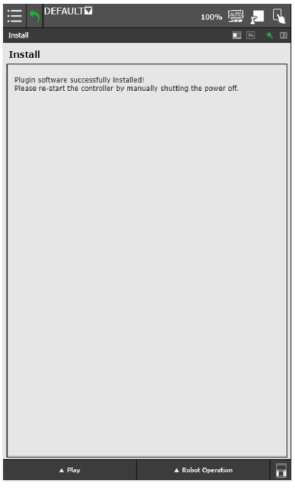


5. Select the "Install" button.



⇨ The installation is executed.

6. Restart the robot controller as soon as the installation is complete.



5 Uninstalling the software module

1. Select the "Menu" button at the top right of the Tablet Teach Pendant screen.
2. In the "PLUGINS" menu > "select Plugin List".
3. Select software module from the list.
4. Select "Uninstall" on the lower right-hand side.
5. Confirm selection.
   ⇨ The message appears saying the software module has been uninstalled.
6. Restart the robot controller.

6 Configuring the software module

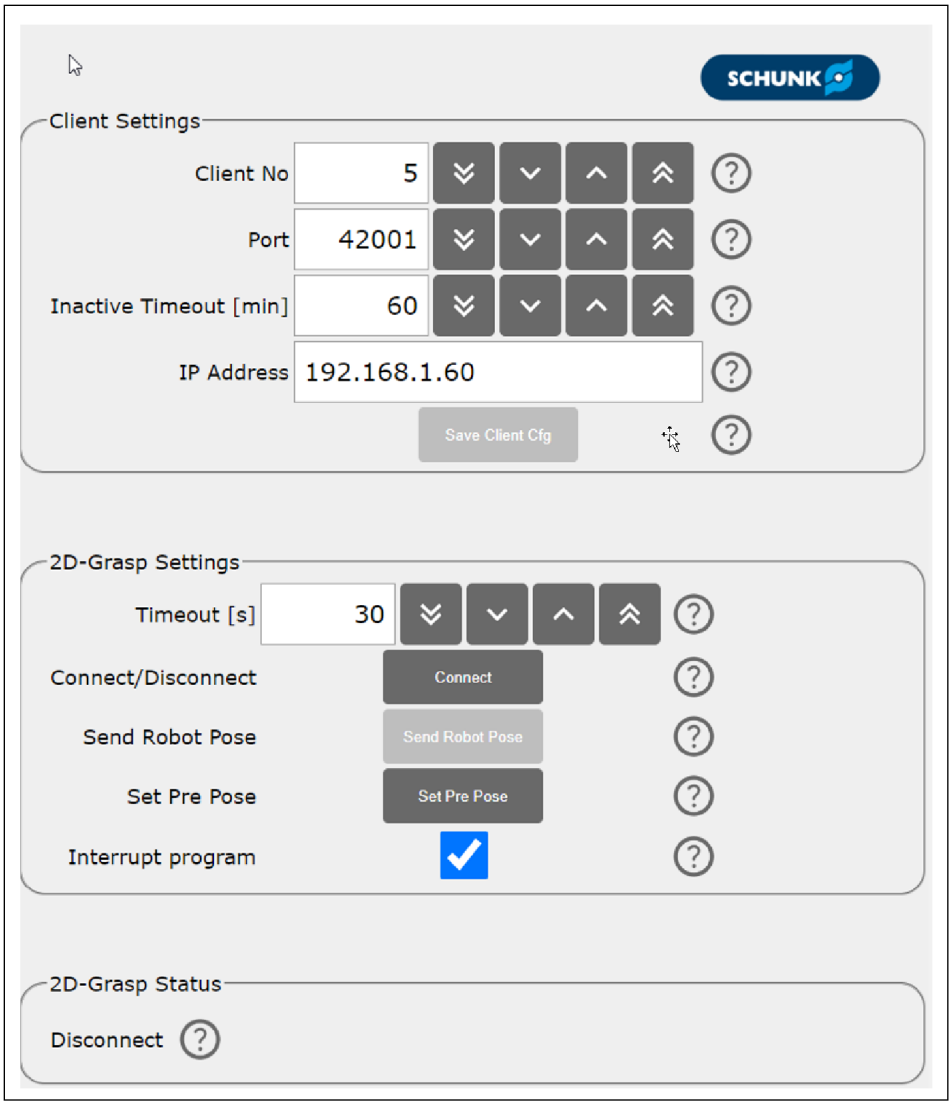
⚠ WARNING

Risk of injury due to sudden movements!

Components could move unexpectedly and result in serious injuries.

- During commissioning, observe all warnings displayed on the software interface.
- Keep a safe distance and wear suitable protective equipment.

- Robot and SVC are switched on and connected with a network cable.
- Robot and SVC are in the same network.
- *Select Menu bar > PLUGINS > SCHUNK 2D-Grasp > Settings.*



The following settings are possible in the menu item:

- Client no. of the FANUC controller
  Factory setting: 5
- Selection of the port SVC
  Factory setting: 42001
- Inactive timeout
  Factory setting: 60 [min]
- Selection of the IP address SVC
  Factory setting: 192.168.1.76
- Connect/Disconnect: A connection is established/terminated to the SVC
- Timeout: Exact limit for a timeout (in seconds) between the robot request and the SVC response.
- Sending the robot position: The current position (TCP) in the robot's base coordinate system is transmitted to the SVC.
- Setting the robot pre-position: The configuration for the gripping position is set (F,U,T,0,0,0). The value is stored in the CMOS.
- Program interruption: By activating this option, the main program is interrupted as soon as a function of this PlugIn detects an error.

Connecting robot and SVC

1. Select IP address or use factory setting. This depends on the settings of the SVC.
2. Select the "Connect" button.
   ⇨ Connection is established.

Possible messages

The following messages may appear in the installation area:

Message	Action
Warning: Device not connected! Please enter parameters and connect.	• Set IP address and port. • Select the "Connect" button in the installation window.
Error: Connection cannot be established!	• Check connection cable. • Check the network settings of the robot. • Check network settings on SVC *Settings > Network > Robot*.
Error: Incorrect protocol version	• SVC Update. Request new version from SCHUNK.

7 Creating robot program

After installing the software module, the following functions can be inserted into a robot program.

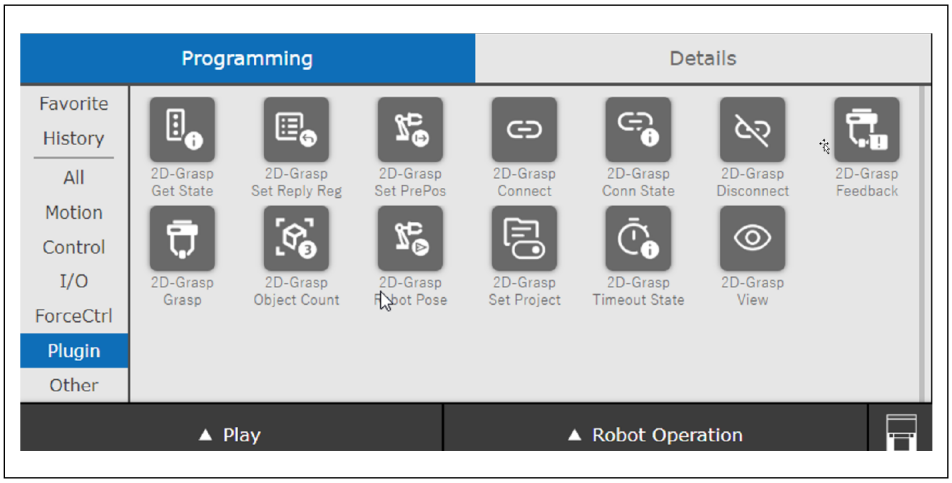
Function	Description
Set Pre Pos 	The robot configuration is saved (CMOS). This configuration (e.g. F,U,T,0,0,0) is added to all positions reported by the SVC. The command can be executed at any time. No connection to the SVC is required.
Connect 	A connection to the SVC is established. The connection parameters are taken from the "Settings" (▶ 6 [🗋 13]). If no connection can be established, an error message appears.
Disconnect 	The connection to the SVC is disconnected.
▶ Set Reply Reg [🗋 19] 	Configuration of the register in which the response of the SVC commands is saved
▶ Grasp [🗋 20] 	Starts object detection and receives the grasping pose for an object. All necessary information is provided to handle a grasping task. The function is parameterized by specifying the grasping mode and object ID.
▶ Feedback [🗋 21] 	Sends a feedback signal to the SVC in relation to the grasping action performed.
▶ Set Project [🗋 21] 	Requests a project change based on the project ID.
▶ Set Tool [🗋 22] 	Requests a tool change based on the tool ID.

Function	Description
▶ Object Count [🗋 22] 	This function makes it possible to analyze the scene. The number of all objects and the number of instances of a specified object class are determined.
▶ Get State [🗋 23] 	This function queries the status of the SVC.
▶ Timeout State [🗋 23] 	The command monitoring status is determined.
Robot Pose 	The current robot position (X,Y,Z,W,P,R) is transmitted to the SVC. The robot configuration is not taken into account here. For further information, see the SVC commissioning instructions.
▶ View [🗋 23] 	The evaluated image of the last grasping command is updated. The image is displayed under "PLUGINS->SCHUNK 2D GRASP->VIEW".
▶ Conn State [🗋 24] 	The connection status is queried and the result is written to a numerical register.

Refer to the FANUC CRX manuals for more information about programming using the visual editor.

Add function to robot program

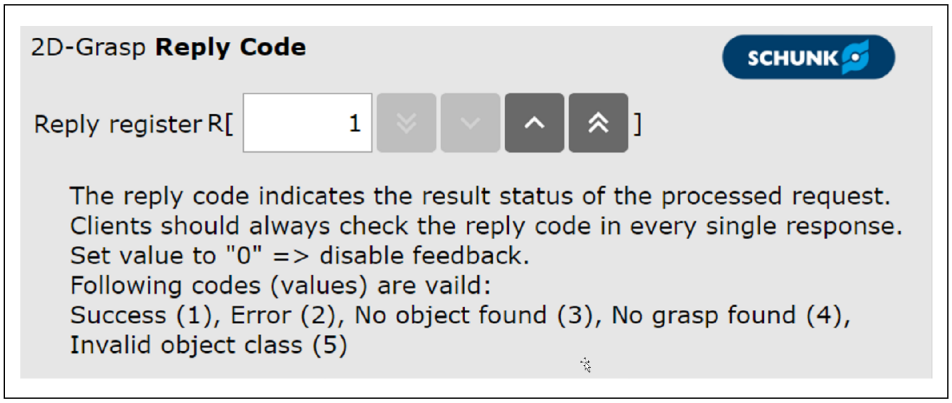
1. Select the "PlugIn" button in the CRX program editor.
   ⇨ All available functions are displayed in the menu.



2. Drag the desired command into the program.
3. Adjust parameters. To do this, select the command in the program and choose the "Details" tab.
   ⇨ For further information on the functions, see the following sections.

7.1 Set Reply Reg

Configuration of the register in which the response of the SVC commands is saved.

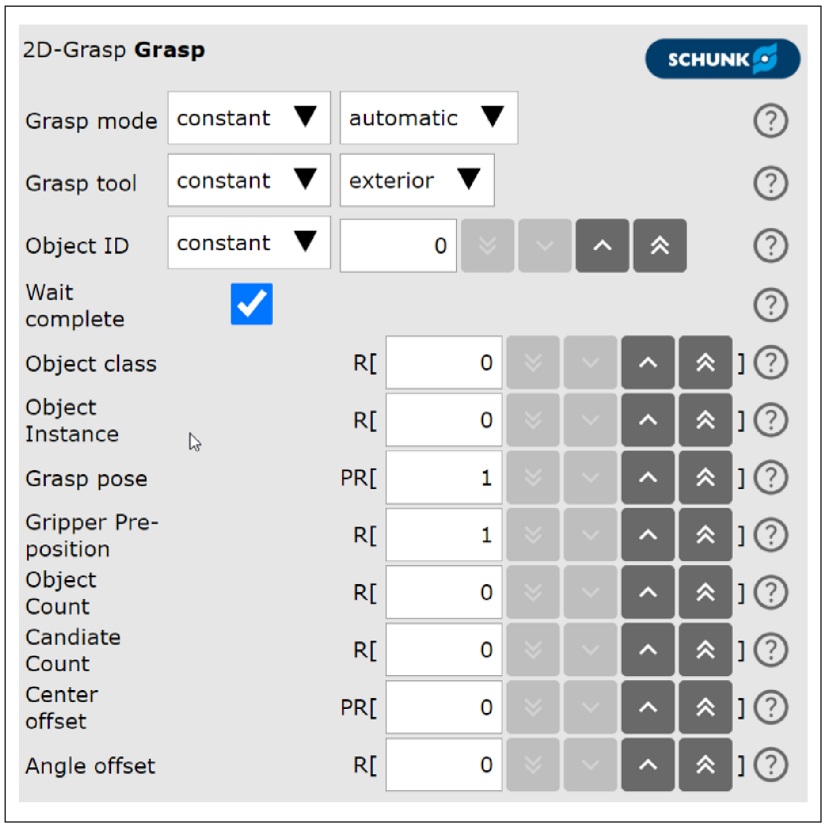


The numeric register is always written to when a command is sent.

Note: This command only needs to be called once or whenever the response register is to be changed.

7.2 Grasp

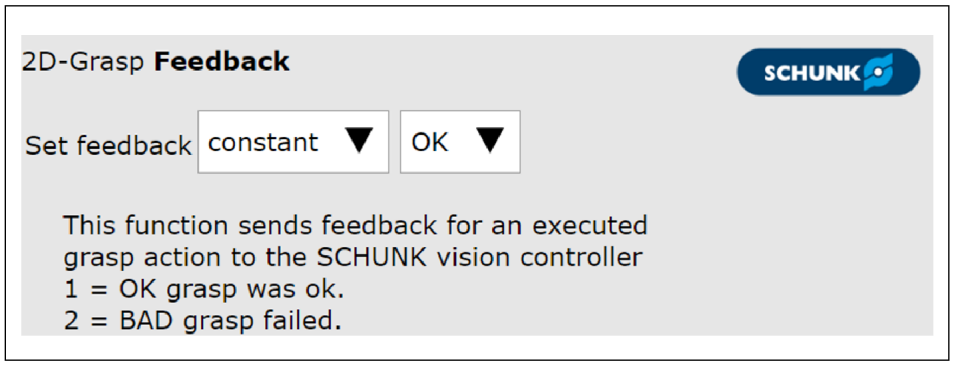
This command is used to provide all the information required to handle a grasping task.



- Select gripper mode, tool and object class
  - As a fixed value (constant)
  - Dynamic via register
- Wait until the command has been completely processed
  - If this option is **activated**, the program sequence is stopped until the command has been processed and a response is available. The reply is stored in a register that was defined with the "Reply Code" command.
  - If this option is **deactivated**, further actions can be carried out after the command has been sent. The user is responsible for correct processing. (Monitor corresponding status information)
- Select the register in which the answer is written
  - Gripping position and center offset are written to position registers. The configuration of the robot (e.g. B. F,U,T,0,0,0,0) is taken from the position saved with "SetPrePos".
  - If register "0" is selected, this value is not reported back.
  - The gripping position and gripper position must always be read.

7.3 Feedback

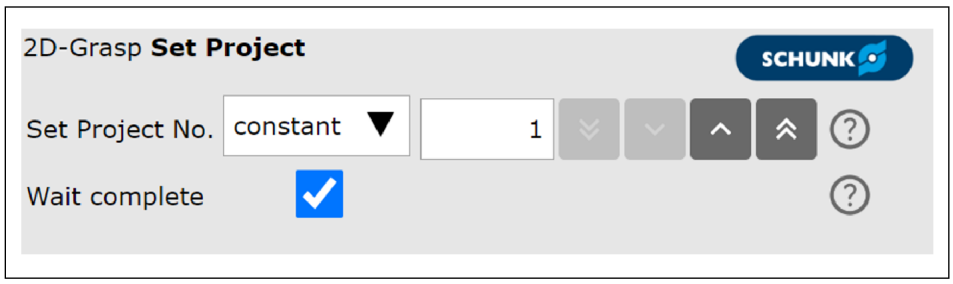
Sends a feedback signal in relation to the grasping action performed to the SVC.



Depending on the quality of the grasping action performed, select "OK" or "Poor" as feedback. The return value can also be written via a register.

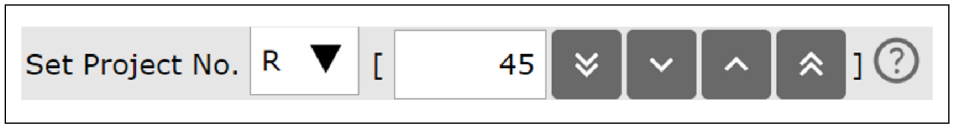
7.4 Set Project

This command switches the active project. It takes some time for this command to be fully processed.



- Variable for the project ID with the ID of the project to be activated.

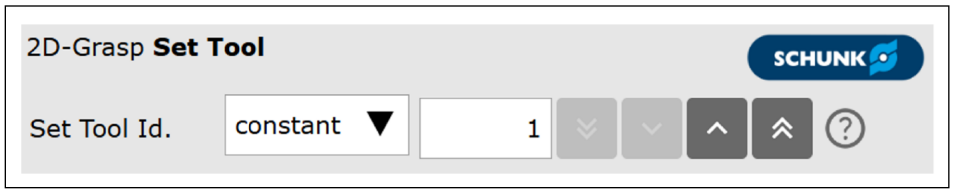
  The project selection can also be made via a numerical register.



- Wait until the command has been completely processed
  - If this option is **activated**, the program sequence is stopped until the command has been processed and a response is available. The reply is stored in a register that was defined with the "Reply Code" command.
  - If this option is **deactivated**, further actions can be carried out after the command has been sent. The user is responsible for correct processing. (Monitor corresponding status information)

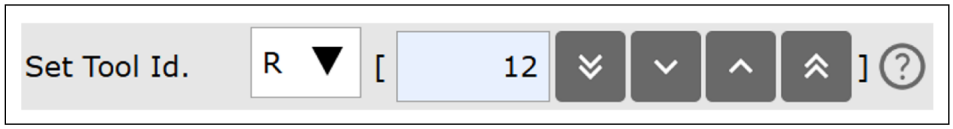
7.5 Set Tool

This command switches the active tool.



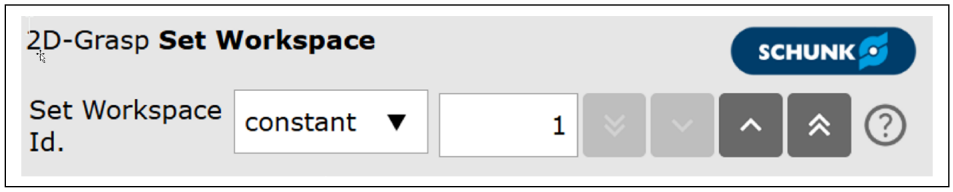
- Describe the variable for the tool ID with the ID of the tool to be activated.

  The tool selection can also be made via a numerical register.



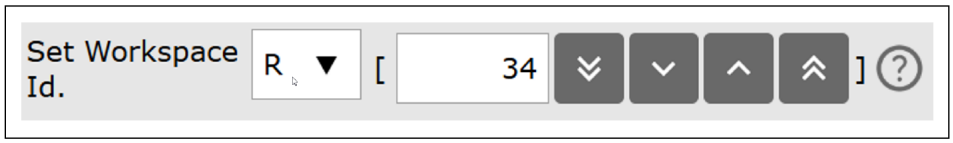
7.6 Set Workspace

This command switches the active workspace.



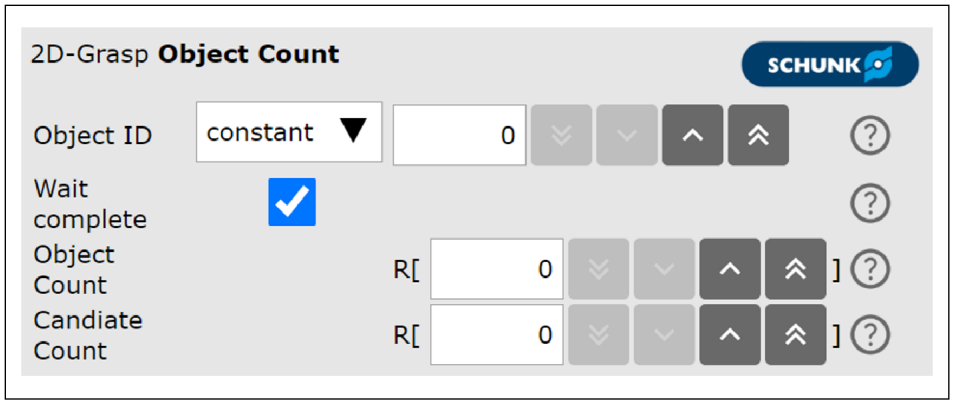
- Describe the variables for the workspace ID with the ID of the workspace to be activated.

  The workspace can also be selected via a numerical register.



7.7 Object Count

This command determines the number of all objects and the number of instances of a specified object class.



- Write the variable for the object class ID using the ID of the target object or select "0" for all objects.

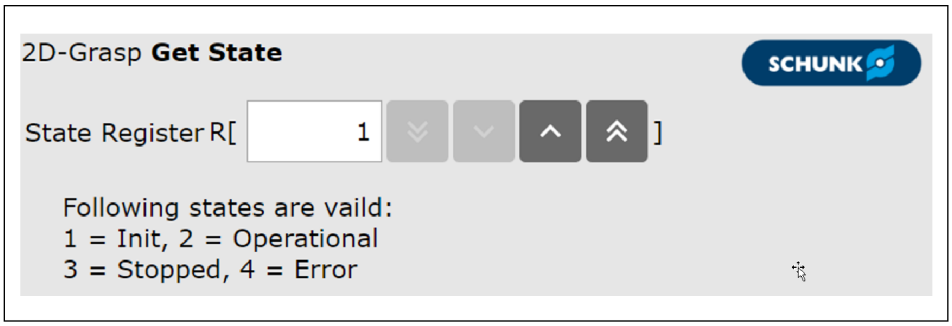
The result is written to the configured numerical register. If the register is set to "0", the result is ignored.

- Object selection can also be done via a numerical register.

- Wait until the command has been completely processed
  - If this option is **activated**, the program sequence is stopped until the command has been processed and a response is available. The reply is stored in a register that was defined with the "Reply Code" command.
  - If this option is **deactivated**, further actions can be carried out after the command has been sent. The user is responsible for correct processing. (Monitor corresponding status information)

7.8 Get State

The command queries the SVC status and writes the result to the configured numeric register.



7.9 Timeout State

The "command monitoring" status (setting timeout) is written to a numerical register.

If the time is exceeded, the time that has elapsed until the error occurs is displayed in negative. Otherwise, you can query the time (positive in sec.) that has elapsed since a command was sent.

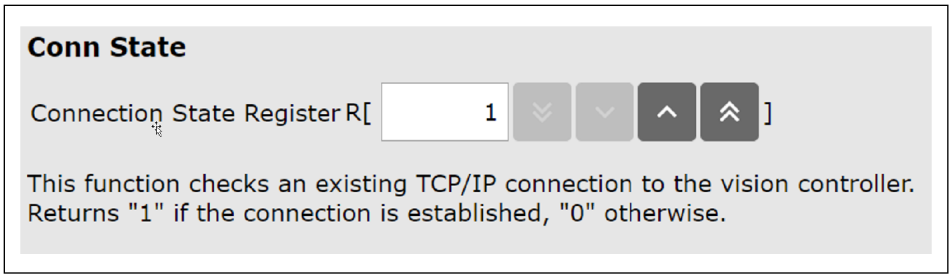
In the event of an error, an alarm is always triggered (warning, error depending on configuration, ▸ 6 [🗋 13])

7.10 View

The image evaluated from the last "Grasp" or "Object Count" command is updated. The image is visible at *PLUGINS -> SCHUNK 2D-Grasp -> View* .

7.11 Conn State

The connection status is queried and the result is written to a numerical register.



8 Example of a robot program

This "Pick&Place" example program can be used as a guide for creating individual applications.

Prerequisites

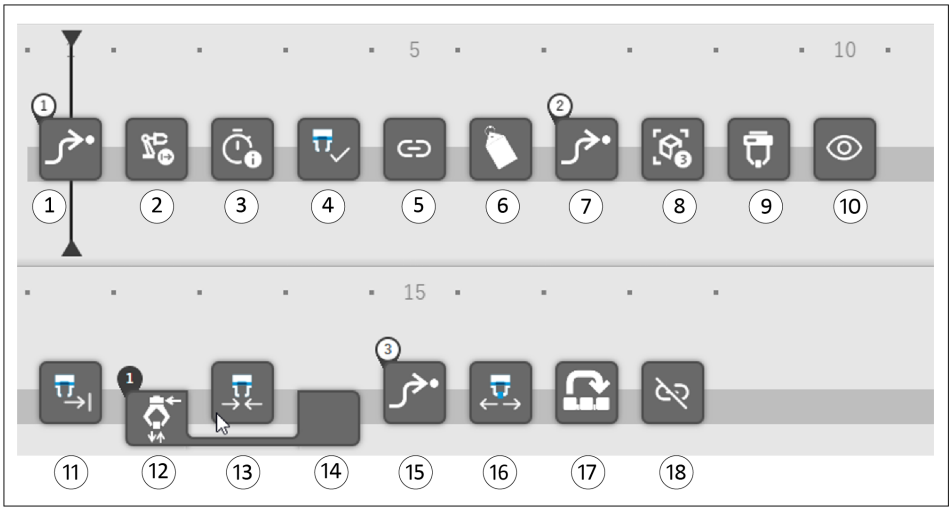
The following preconditions must be met:

- Option package *User Socket Msg (R648)* is installed.
- Gripper is installed.
- Tool[1] is set up.
- SVC PlugIn is installed.
- EGU/EKG Gripper PlugIn is installed.
  Note: Any gripper can be used. The gripper-specific commands in the example must be adapted accordingly.
- SVC is configured and ready for operation.
- Gripper is configured and ready for operation.

NOTE

If the gripping position is approached via a joint command, the option package *MROT (R640)* is recommended.

Graphical implementation of the sample program



TP code

```
  1:J  P[1] 100% FINE
  2:   CALL IPL_SCHUNK_2DG_ROBPREPOS
  3:   CALL IPL_SCHUNK_2DG_TIMSTAT(21)
  4:   CALL IPL_SCHUNK_EGK_ACK(1)
  5:   CALL IPL_SCHUNK_2DG_CON
  6:   LBL[1]
  7:J  P[2] 100% FINE
  8:   CALL IPL_SCHUNK_2DG_OBJCNT(0,1,8,9)
  9:   CALL IPL_SCHUNK_2DG_GRASP(3,1,0,1,5,6,1,10,0,0,0,0)
 10:   CALL IPL_SCHUNK_2DG_VIEW
 11:   CALL IPL_SCHUNK_EGK_MOVE_ABS(1,R[10],80,1,0)
 12:   CALL -INST_BASICPICK_START(1,(-1),PR[1],0,100,75,150)
 13:   CALL IPL_SCHUNK_EGK_BASIC_GRIP(1,0,150,25,1,1,0)
 14:   CALL -INST_BASICPICK_END(1,(-1),PR[1],0,100,75,150)
 15:J  P[3] 100% FINE
 16:   CALL IPL_SCHUNK_EGK_RELEASE(1,1,0
 17:   JMP LBL[1]
 18:   CALL IPL_SCHUNK_2DG_DISCON
[End]
```

Tab.: Command Explanations

1.		Drive the robot close to the grasping task.
2.		Adopt the current robot configuration for the grasping task.
3.		Use register 21 for timeout monitoring.
4.		• Initialize the gripper. The command structure differs if a different gripper is used. • Acknowledge the gripper error messages 1.
5		Establish a connection to the SCHUNK Vision Controller (SVC).
6.		Set jump mark.
7.		Move robot to "observation position". The camera image must not be obscured.

8.		• Show found objects (optional). • Show all objects (0). • Wait until the command has been completely processed (1). • Write the result in registers 8 and 9: – Register 8 contains all objects found. – Register 9 contains all objects found that belong to the selected object ID. In this example, this corresponds to all objects, as the object ID has been set to "0".
9.		Perform a calculation for a grasp: • Gripper mode: "Automatic" (3) • Gripper tool: "O.D. gripping" (1) • Object ID to be gripped: "Any object" (0) • Wait until this command has been completed (1). • Write the object ID of the planned grasping object in register 5. • Write the object instance of the planned grasping object in register 6. • Write the calculated gripping position in position register 1. • Write the calculated gripper finger position in position register 10. • Ignore the information about the number of objects found. (0) • Ignore the information about the number of objects found that belong to the selected object ID. (0) • Ignore the information about the offset of the calculated gripping point to the "center point" of the gripping object. (0) • Ignore the information about the rotation of the model relative to the robot flange. May be required if the orientation is relevant when storing the object. (0)
10.		Update the evaluated image (optional). Image is shown under *PLUGINS -> SCHUNK 2D-Grasp->View*.
11.		Move the gripper finger to the calculated position. The command structure differs if a different gripper is used. In this example SCHUNK EGU/EKG gripper 1: • Move the gripper finger to the position saved in the register 10. • at a speed of 80 mm/s • Wait until the target position is reached (1). • Do not use a brake (0).
12.		Grasp the object. The FANUC Basic Pick/Place command is used here. • The position stored in position register 1 is approached. • Further information on this command can be found in the FANUC robot manual.

13.		Grasp the object. The command structure differs if a different gripper is used. In this example SCHUNK EGU/EKG gripper 1: • O.D. gripping (0). • Use 150% gripping force. • Move at 25 mm/s. • Wait until the command has been completed (1). • Use a brake (1). • "Soft Mode" is not used (0).
14.		End of the FANUC Basic Pick/Place command. Further information on this command can be found in the FANUC robot manual.
15.		Drive to the storage position.
16.		Put the object down. The command structure differs if a different gripper is used. In this example SCHUNK EGU/EKG gripper 1: • Wait until the command has been completed (1). • Do not use a brake (0).
17.		Jump to jump mark 1 line 6 The next object is gripped.
18.		Terminates the connection to the SVC.

9 Troubleshooting

Message	Action
No connection to vision controller	• Check connection cable. • Check the network settings of the robot. • Check network settings on SVC (*Settings > Network > Robot*).
(IPL_SCHUNK_2DG-MAIN, xxx) Undefined build in	• Install option package *User Socket Msg (R648)*.
Wrong protocol version	• SVC Update. Request new version from SCHUNK.
Command timeout	• Check the timeout setting: • Check the connection to the SVC
Invalid object class	• Correct object class.
Camera not connected	• Check the connection to the camera • Check the connection of the camera to the SVC (correct port?).
Camera not calibrated	• Calibrate camera via SVC interface.
Robot not calibrated	• Calibrate robot via SVC interface.
Workspace not calibrated	• Calibrate workspace via SVC interface.
No active project	• Activate project via SVC interface. • Select project via "Set Project" command.
No active tool	• Activate gripper via SVC interface. • Select gripper via "Set Tool" command.
No grasp found	• Check SVC project (gripper settings). • Check the position of the gripper objects. • Check configuration of the robot.
No object found	• Check SCV project. • Check the camera's field of view. • Are objects available?
Error processing request	• Check last SCV command.
Unitialized data is used (after grasp command)	• Check robot configuration. • Use "Set Pre Pos".

SCHUNK SE & Co. KG
Spanntechnik | Greiftechnik | Automatisierungstechnik

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

Folgen Sie uns | *Follow us*

Wir drucken nachhaltig | *We print sustainable*

12-2025 03.00 | 2D Grasping Kit | Software manual | en | 159217