DOBOT

OnRobot Gripper
User Guide

Issue: V2.0

Date: 2024-01-26

Disclaimer

To the maximum extent permitted by applicable law, the products described (including its hardware, software and firmware, etc.) in this document are provided **AS IS**, which may have flaws, errors or faults. Dobot makes no warranties of any kind, express or implied, including but not limited to, merchantability, satisfaction of quality, fitness for a particular purpose and non-infringement of third party rights. In no event will Dobot be liable for any special, incidental, consequential or indirect damages resulting from the use of our products and documents.

Before using our product, please thoroughly read and understand the contents of this document and related technical documents that are published online, to ensure that the robot arm is used on the premise of fully understanding the robot arm and related knowledge. Please use this document with technical guidance from professionals. Even if follow this document or any other related instructions, damages or losses may happen in the using process. Dobot shall not be considered as a guarantee regarding all security information contained in this document.

The user has the responsibility to make sure following the relevant practical laws and regulations of the country, in order that there is no significant danger in the use of the robot arm.

Shenzhen Yuejiang Technology Co., Ltd.

Address: Room 1003, Building 2, Chongwen Garden, Nanshan iPark, Liuxian Blvd, Nanshan District, Shenzhen, Guangdong Province, China

Website: www.dobot-robots.com

Preface

Purpose

    This document introduces the installation and operations of OnRobot grippers used on Dobot six-axis robot, which is convenient for users to understand and use OnRobot grippers.

Intended audience

    This document is intended for:

- Customer
- Sales Engineer
- Installation and Commissioning Engineer
- Technical Support Engineer

Revision history

Date	Version	Revised content
2024/01/26	V2.0	Adapted to V4 controller, supporting the plugin V2-0-X
2023/03/10	V1.0	The first release

Symbol conventions

    The symbols that may be found in this document are defined as follows.

Symbol	Description
⚠ DANGER	Indicates a hazard with a high level of risk which, if not avoided, could result in death or serious injury.
⚠ WARNING	Indicates a hazard with a medium level or low level of risk which, if not avoided, could result in minor or moderate injury, robot arm damage.
⚠ NOTICE	Indicates a potentially hazardous situation which, if not avoided, could result in robot arm damage, data loss, or unanticipated result.
ⓘ NOTE	Provides additional information to emphasize or supplement important points in the main text.

Contents

1. Overview

Dobot adapts most of the OnRobot grippers, which communicate with the robot arm via Modbus protocol. This document mainly introduces how to use the OnRobot grippers on the robot.

The firmware and software versions for using OnRobot grippers are shown below.

Software	Version
Plug-in version	2-0-X
Controller firmware	4.5.0.1 and above
DobotStudio Pro	4.5.0.1 and above

1.1   2FG7 parallel gripper

The 2FG7 parallel gripper is easy to deploy in tight spaces and handles demanding payload requirements.

1.2   VG10 / VGC10 vacuum gripper

The VG10 / VGC10 vacuum gripper needs no external air supply, it features two independently controlled air channels.

2. Hardware Installation

2.1　Tool fixing

The OnRobot grippers can be installed on the end flange of the robot through the Quick Changer. For specific procedure, see the corresponding installation guide.

2.2　Cable connection

After installation, you can connect the gripper to the robot using the following two methods.

- Connect the aviation interface at the end of robot to the communication interface of Quick Changer using the **Dobot customized aviation cable** (procurement code: 3303000062).
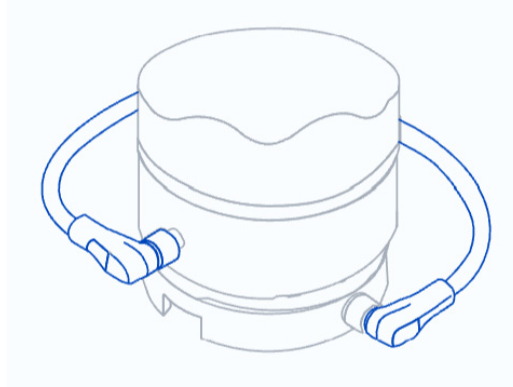
- Purchase an OnRobot Compute Box, connect the Compute Box to the communication interface of Quick Changer using the aviation cable provided by OnRobot, and then connect the Compute Box to the LAN1 interface of the robot controller using the network cable.
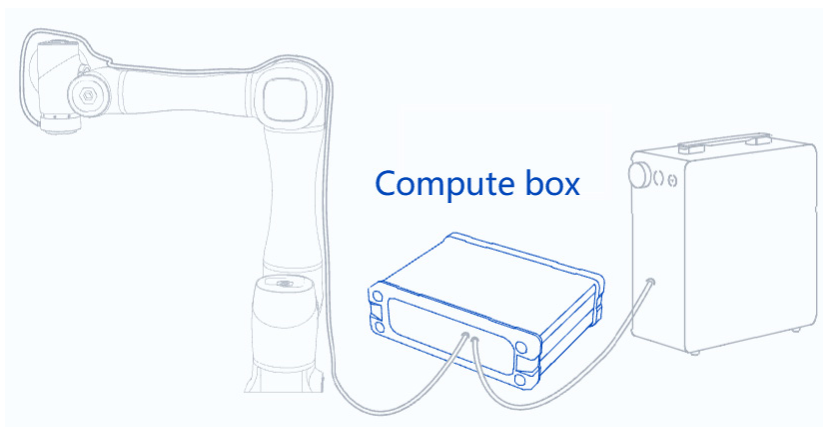
When using this method for connection, you need to change the IP of the Compute Box and the robot controller to the same network segment. For the IP modification method of the Compute Box, please refer to its user manual. For the IP modification method of the robot controller, please refer to the help document of DobotStudio Pro.

3. Plugin Installation and Configuration

1) Open the DobotStudio Pro and enter the Dobot+ page, import and install the OnRobot plugin.
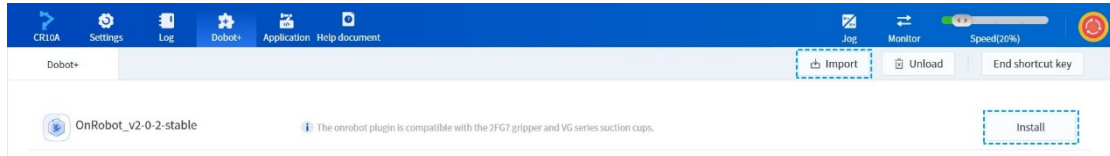
2) After installation, enter the **OnRobot Configuration Guide** page.

3) Click **Next step**, and select the connection method based on the actual cable connection.
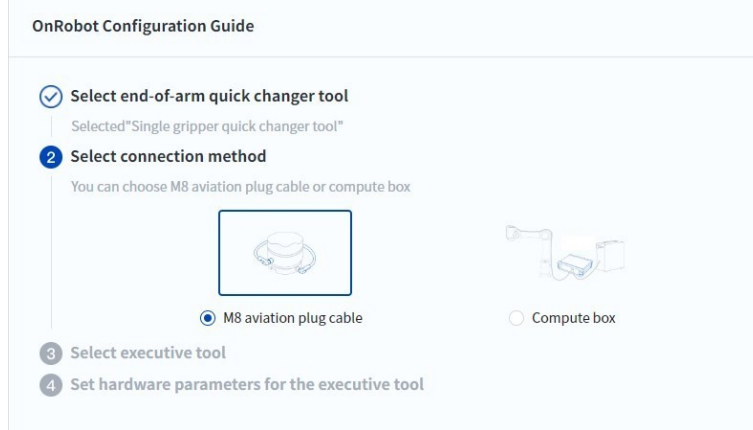
4) Click **Next step**, and select the executive tool based on the actual tool installed.

5) Click **Next step**, and set the hardware parameters by referring to the instructions for each tool below.

OnRobot Configuration Guide

✓ **Select end-of-arm quick changer tool**
  Selected"Single gripper quick changer tool"
✓ **Select connection method**
  Selected"M8 aviation plug cable"
✓ **Select executive tool**
  Selected"2FG7"
④ **Set hardware parameters for the executive tool**
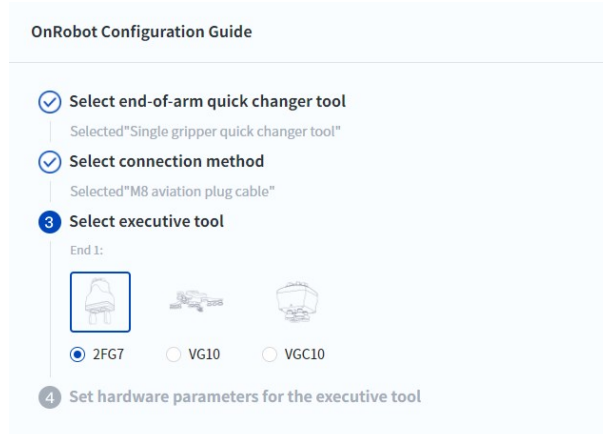
2FG7(1)

| Installation type | | ⊤ Inward | ⊤⊤ Outward |

Image1 Finger width   Image2 Finger length   Image3 Finger height
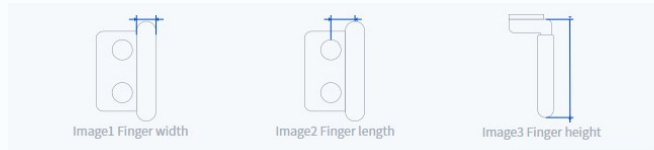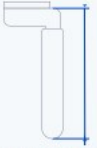
Finger height	⦿ 5mm	○ 7mm	○ Custom	0 mm
Finger length		⦿ 8.5mm	○ Custom	0 mm
Finger height		⦿ 45mm	○ Custom	0 mm

4. 2FG7 Gripper Instructions

4.1 Hardware parameter settings

- **Installation type**: Select the gripper angle according to the actual installation method of the finger (Inward or Outward).
- **Finger width**: Set the width of the fingertips. The width of the standard metal fingertip is 5mm, and the silicone rubber fingertip is 7mm. If you use customized fingers, select **Custom** and enter the diameter of the customized finger.
- **Finger length**: The length of the finger is the distance from the center of the mounting hole to the midline of the fingertip. The length of the standard finger is 8.5mm. If you use customized fingers, select **Custom** and enter the length of the customized fingers.
- **Finger height**: The height of the finger is the distance from the fingertip to the mounting plane of the finger root. The height of the standard finger is 45mm. If you use customized fingers, select **Custom** and enter the height of the customized fingers.

  Click **Configuration**, the plugin will try to connect to the gripper. After successful connection, you will enter the Control panel page.

4.2 Control panel

- **Gripping method**: Select according to how the gripper grips the object.
  - External grip: grip the object from the outside. The width refers to the relative distance between the inside of the fingers.

○   Internal grip: grip the object from the inside. The width refers to the relative distance between the outside of the fingers.

• **Current width**: Current width of the gripper, which varies according to the gripping method.

• **Target width**: After setting the target width, the gripper will move to the specified width.

> **ⓘ NOTE**
>
> The width range is related to the finger installation angle, finger length and gripping method (accuracy: 0.1). For detailed values, refer to 2FG7 specifications document.

• **Force**: The target force that the gripper grips or stretches the target object. When the gripper detects the force, it judges that it has gripped the target object and stops the fingertip movement.

> **ⓘ NOTE**
>
> • The range of gripping force is 20 – 140 (integer only).
> • To better reach the target force, it is recommended to set the target width to 1–3mm less than (external grip) or 1–3mm greater than (internal grip) the actual object according to the softness of the target object.

• **Speed: Target speed ratio of the fingertip motion.**

> **ⓘ NOTE**
>
> The speed range is 10 – 100 (integer only).

4.3   Programming commands

After installing the OnRobot plugin, you will see the OnRobot-related commands on Blockly programming page and Script programming page. The current version of the plugin only supports single gripper, the parameters related to dual grippers are not available.

4.3.1   Blockly programming
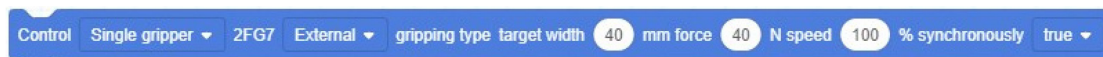
Control Single gripper ▾ 2FG7 External ▾ gripping type target width 40 mm force 40 N speed 100 % synchronously true ▾

Description: Control the 2FG7 gripper to move to the target position, and set the gripping method, target force and target speed. The gripper will stop moving when it detects the target force value before moving to the specified position.

Parameter:
1) Gripper index.
   ○   Single gripper
   ○   End 1 (Dual gripper)
   ○   End 2 (Dual gripper)
2) Gripping method.
   ○   External grip: The target position is the distance inside the fingers
   ○   Internal grip: The target position is the distance outside the fingers
3) Target position of the gripper (accuracy: 0.1), unit: mm.
   For position range, refer to 2FG7 specifications document.
4) Target force of the gripper, unit: N.

The force range is 20 – 140 (integer only).

5) Target speed ratio of fingertip motion, unit: %.

The ratio range is 10 – 100 (integer only).

6) Synchronization.

   ◦ true: Wait until the execution of the current command is completed before executing the next command.

   ◦ false: Execute the next command directly as soon as the command is delivered.

> Get Single gripper ▾ 2FG7 External ▾ gripping type current width

Description: Get the current relative distance of fingertips.

Parameter:

1) Gripper index.

   ◦ Single gripper

   ◦ End 1 (Dual gripper)

   ◦ End 2 (Dual gripper)

2) Measuring method for fingertip distance.

   • External grip: The width is the distance inside the fingers

   • Internal grip: The width is the distance outside the fingers

Return: The current relative distance of the fingertips (accuracy: 0.1), unit: mm.

> Get Single gripper ▾ 2FG7 if it grabs an object and reaches the desired force

Description: Judge whether it has gripped the object according to whether the gripping force reaches the target force.

Parameter: Gripper index.

   ◦ Single gripper

   ◦ End 1 (Dual gripper)

   ◦ End 2 (Dual gripper)

Return:

   • 0: Not gripped

   • 1: Gripped

4.3.2　Script programming

`Onrobot2FGGrip(grip_method, target_width, force, speed, {toolIndex=0, isBlock=true})`

Description: Control the 2FG7 gripper to move to the target position, and set the gripping method, target force and target speed. The gripper will stop moving when it detects the target force value before moving to the specified position.

Parameter:

• **grip_method**: gripping method.

   ◦ 1: External grip. The target position is the distance inside the fingers

   ◦ 2: Internal grip. The target position is the distance outside the fingers

• **target_width**: Target position of the gripper (accuracy: 0.1), unit: mm.

   For width range, refer to 2FG7 specifications document.

• **force**: Target force of the gripper, unit: N.

   The force range is 20 – 140 (integer only).

• **speed**: Target speed ratio of fingertip motion, unit: %.

The ratio range is 10 – 100 (integer only).

- **toolIndex**: Optional parameter, gripper index.
  - ◦ 0 (default): Single gripper
  - ◦ 1: End 1 (Dual gripper)
  - ◦ 2: End 2 (Dual gripper)
- **isBlock**: Optional parameter, block the running of subsequent programs or not.
  - ◦ true (default): Wait until the execution of the current command is completed before executing the next command.
  - ◦ false: Execute the next command directly as soon as the command is delivered.

Example:

```
Onrobot2FGGrip(1, 40, 40, 100, {toolIndex=0, isBlock=true})
```

```
Onrobot2FGWidthGet(measure_method, {toolIndex=0})
```

Description: Get the current relative distance of fingertips.

Parameter:

- **measure_method**: Measuring method for fingertip distance.
  - ◦ 1: External grip. The width is the distance inside the fingers
  - ◦ 2: Internal grip. The width is the distance outside the fingers
- **toolIndex**: Optional parameter, gripper index.
  - ◦ 0 (default): Single gripper
  - ◦ 1: End 1 (Dual gripper)
  - ◦ 2: End 2 (Dual gripper)

Return: The current relative distance of the fingertips (accuracy: 0.1), unit: mm.

Example:

```
Onrobot2FGWidthGet(1, {toolIndex=0})
```

```
Onrobot2FGGripDetected({toolIndex=0})
```

Description: Judge whether it has gripped the object according to whether the gripping force reaches the target force.

Parameter:

- **toolIndex**: Optional parameter, gripper index.
  - ◦ 0 (default): Single gripper
  - ◦ 1: End 1 (Dual gripper)
  - ◦ 2: End 2 (Dual gripper)

Return:

- 0: Not gripped
- 1: Gripped

Example:

```
local isGripped = Onrobot2FGGripDetected({toolIndex=0})
```

5. VG10 / VGC10 Vacuum Gripper Instructions

5.1 Hardware parameter settings

There is no need to set hardware parameters for VG10 and VGC10.

Click **Configuration**, the plugin will try to connect to the suction cup. After successful connection, you will enter the Control panel page.

5.2 Control panel

Despite differences in parameters, VG10 and VGC10 are common in the control mode. This section takes VGC10 as an example to introduce.
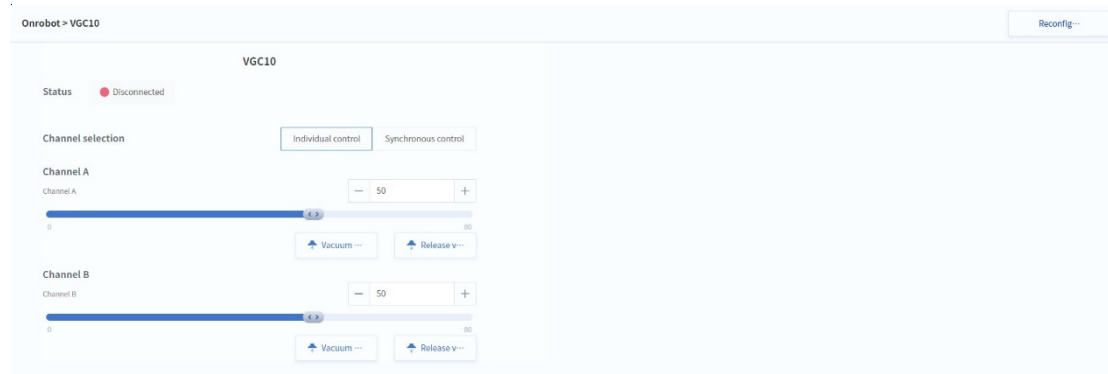
Channel selection: When selecting **Individual control**, you can set the vacuum of Channel A and Channel B respectively; when selecting **Synchronous control**, if you set the vacuum degree of one channel, the other channel will be automatically synchronized.

After setting the target vacuum degree of the corresponding channel, click **Suction** to execute the suction action, and click **Release** to execute the release action.

> ℹ **NOTE**
>
> The range of vacuum degree is 0 – 80 (integer only).

5.3 Programming command

After installing the OnRobot plugin, you will see the OnRobot-related commands on Blockly programming page and Script programming page. The current version of the plugin only supports single suction cup, the parameters related to dual suction cups are not available.

5.3.1 Blockly programming

Description: Control the suction cup to grip.

Parameter:
1) Suction cup index.
   - ◦ Single
   - ◦ End 1 (Dual suction cups)
   - ◦ End 2 (Dual suction cups)
2) Channel to be controlled. You can control one channel or control two channels simultaneously.
3) Target vacuum degree, range: 0 – 80 (integer only), unit: %.
4) Timeout of waiting for the vacuum to reach the target value, unit: ms.

   0: Execute the next command after delivering this command without waiting for reaching the specified vacuum degree.

   Integer greater than 0: Execute the next command after the vacuum degree reaches the target value within the specified time. If the vacuum degree does not reach the target value within the specified time, the robot will alarm.
5) Monitor the vacuum degree continuously or not (Drop detection).
   - ◦ true: After the suction cup reaches the target vacuum degree, the vacuum degree is continuously monitored until the vacuum is released. If the vacuum degree decreases, the plugin reports an error.
   - ◦ false: Vacuum degree will not be monitored after the suction cup reaches the target vacuum degree.

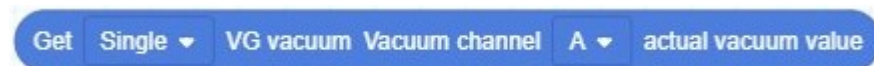**Description**: Control the suction cup to release.

Parameter:
1) Suction cup index.
   - ◦ Single
   - ◦ End 1 (Dual suction cups)
   - ◦ End 2 (Dual suction cups)
2) Channel to be controlled. You can control one channel or control two channels simultaneously.
3) Timeout of waiting for the vacuum to be released, unit: ms.

   0: Execute the next command after delivering this command without waiting for the vacuum to be released.

   Integer greater than 0: Execute the next command after the vacuum degree reaches 0 within the specified time. If the vacuum degree does not reach 0 within the specified time, the robot will alarm.

Description: Get the current vacuum degree of the specified channel.

Parameter:
1) Suction cup index.
   - ◦ Single
   - ◦ End 1 (Dual suction cups)
   - ◦ End 2 (Dual suction cups)
2) The channel to be obtained.

Return: The current vacuum degree of the specified channel.

5.3.2   Script programming

OnrobotVGGrip(channel, vacuum, {toolIndex=0, alert=true, timeout=5000})

Description: Control the suction cup to grip.

Parameter:

- **channel**: The channel to be controlled. Range:
  - ◦ 0: Channel A
  - ◦ 1: Channel B
  - ◦ 2: Channel A and Channel B
- **vacuum**: Target vacuum degree, range: 0 – 80 (integer only), unit: %.
- **toolIndex**: Optional parameter, suction cup index.
  - ◦ 0 (default): Single suction cup
  - ◦ 1: End 1 (Dual suction cups)
  - ◦ 2: End 2 (Dual suction cups)
- **alert**: Optional parameter, monitor the vacuum degree continuously or not (drop detection).
  - ◦ true (default): After the suction cup reaches the target vacuum degree, the vacuum degree is continuously monitored until the vacuum is released. If the vacuum degree decreases, the plugin reports an error.
  - ◦ false: Vacuum degree will not be monitored after the suction cup reaches the target vacuum degree.
- **timeout**: Optional parameter, timeout of waiting for the vacuum to reach the target value, unit: ms. 5000 by default.

  0: Execute the next command after delivering this command without waiting for reaching the specified vacuum degree.

  Integer greater than 0: Execute the next command after the vacuum degree reaches the target value within the specified time. If the vacuum degree does not reach the target value within the specified time, the robot will alarm.

Example:

OnrobotVGGrip(2, 60, {toolIndex=0, alert=true, timeout=5000})

OnrobotVGRelease(channel, {toolIndex=0, timeout=5000})

Description: Control the suction cup to release.

Parameter:

- **channel**: The channel to be controlled. Range:
  - ◦ 0: Channel A
  - ◦ 1: Channel B
  - ◦ 2: Channel A and Channel B
- **toolIndex**: Optional parameter, suction cup index.
  - ◦ 0 (default): Single suction cup
  - ◦ 1: End 1 (Dual suction cups)
  - ◦ 2: End 2 (Dual suction cups)
- **timeout**: Timeout of waiting for the vacuum to be released, unit: ms. 5000 by default.

  0: Execute the next command after delivering this command without waiting for the vacuum to be released.

  Integer greater than 0: Execute the next command after the vacuum degree reaches 0 within the specified time. If the vacuum degree does not reach 0 within the specified time, the robot will alarm.

Example:

```
OnrobotVGRelease(2, {toolIndex=0, timeout=5000})
```

OnrobotVGVacuumGet(channel, {toolIndex=0})

Description: Get the current vacuum degree of the specified channel.

Parameter:
- **channel**: The channel to get the vacuum degree.
  - 0: Channel A
  - 1: Channel B
- **toolIndex**: Optional parameter, suction cup index.
  - 0 (default): Single suction cup
  - 1: End 1 (Dual suction cups)
  - 2: End 2 (Dual suction cups)

Example:

```
local vacuum_a = OnrobotVGVacuumGet(0, {toolIndex=0})
```