USER MANUAL

HEX
Force Torque Sensor

For the Universal Robots

Edition E12

OnRobot FT URCap Plugin Version 4.0.0

September 2018

Content

1 Preface

1.1 Target Audience

This document is intended for integrators who design and install complete robot applications. Personnel working with the sensor are expected to have the following expertise:

1. Basic knowledge of mechanical systems
2. Basic knowledge of electronic and electrical systems
3. Basic knowledge of the robot system

1.2 Intended Use

The sensor is designed for measuring forces and torques, installed on the end effector of a robot. The sensor can be used within the specified measurement range. Using the sensor outside of its range is considered misuse. OnRobot is not liable for any damage or injury resulting from misuse.

1.3 Important safety notice

The sensor is *partly completed machinery* and a risk assessment is required for each application the sensor is a part of. It is important that all safety instructions herein are followed. The safety instructions are limited to the sensor only and do not cover the safety precautions of a complete application.

The complete application must be designed and installed, in accordance with the safety requirements specified in the standards and regulations of the country where the application is installed.

1.4 Warning Symbols

DANGER:

This indicates a very dangerous situation which, if not avoided, could result in injury or death.

WARNING:

This indicates a potentially hazardous electrical situation which, if not avoided, could result in injury or damage to the equipment.

WARNING:

This indicates a potentially hazardous situation which, if not avoided, could result in injury or major damage to the equipment.

CAUTION:

This indicates a situation which, if not avoided, could result in damage to the equipment.

NOTE:

This indicates additional information such as tips or recommendations.

1.5 Typographic Conventions

The following typographic conventions are used in this document.

Table 1: Conventions

`Courier Text`	File paths and file names, code, user input and computer output.
Italicized text	Citations and marking image callouts in text.
Bold text	UI elements, including text appearing on buttons and menu options.
Bold, blue text	External links, or internal cross-references.
<angle brackets>	Variable names that must be substituted by real values or strings.
1. Numbered lists	Steps of a procedure.
A. Alphabetical lists	Image callout descriptions.

2   Getting Started

2.1   Scope of Delivery

In the Universal Robots OnRobot HEX Sensor Kit everything is provided that is required to connect the OnRobot force/torque sensor to your UR robot.

There are two versions of the OnRobot Universal Robots (UR) Kit, depending on the HW version of the sensor.

2.1.1   OnRobot (OptoForce) UR Kit (v1)

The contents of the OnRobot (OptoForce) UR Kit v1 are the following:

- OnRobot (OptoForce) 6-axis force/torque sensor (variant HEX-E v1 or HEX-H v1)
- OnRobot (OptoForce) Compute Box
- OnRobot (OptoForce) USB drive
- adapter-A
- overload plug
- sensor cable (4 pin M8 - 4 pin M8, 5 m)
- Compute Box power cable (3 pin M8 – open ended)
- Compute Box power supply
- UTP cable (RJ45 - RJ45)
- USB cable (Mini-B – Type A)
- PG16 cable gland
- plastic bag, containing:

4. cable holder
5. M6x30 screws (2 pcs)
6. M6x8 screws (pcs)
7. M5x8 screws (9 pcs)
8. M4x8 screws (7 pcs)
9. M4x12 screws (2 pcs)
10. M4 washer (8 pcs)

2.1.2   OnRobot UR Kit (v2)

The contents of the OnRobot UR Kit v2 are the following:

11. OnRobot 6-axis force/torque sensor (variant HEX-E v2 or HEX-H v2)
12. OnRobot Compute Box
13. OnRobot USB drive
14. adapter-A2

15. sensor cable (4 pin M8 - 4 pin M8, 5 m)

16. Compute Box power cable (3 pin M8 – open ended)

17. Compute Box power supply

18. UTP cable (RJ45 - RJ45)

19. PG16 cable gland

20. plastic bag, containing:

21. cable holder, with integrated screw

22. M6x8 Torx screws (6 pcs)

23. M5x8 Torx screws (9 pcs)

24. M4x6 Torx screws (7 pcs)

25. M6 washer (6 pcs)

26. M5 washer (9 pcs)

NOTE:

From mid-September 2018 the USB cable (Mini-B – Type A) is not provided in the OnRobot UR Kit v2, but could be purchased separately, if necessary.

2.2 Sensor Description

2.2.1 HEX-E v1 and HEX-H v1

The sensor consists of a sensor body, an adapter, and a overload plug. The sensor cable connector, the cable holder, and markers for the reference frame are on the sensor body. The tool is fastened to the sensor body directly, onto the tool interface. The sensor is fastened to the robot tool flange by the adapter.
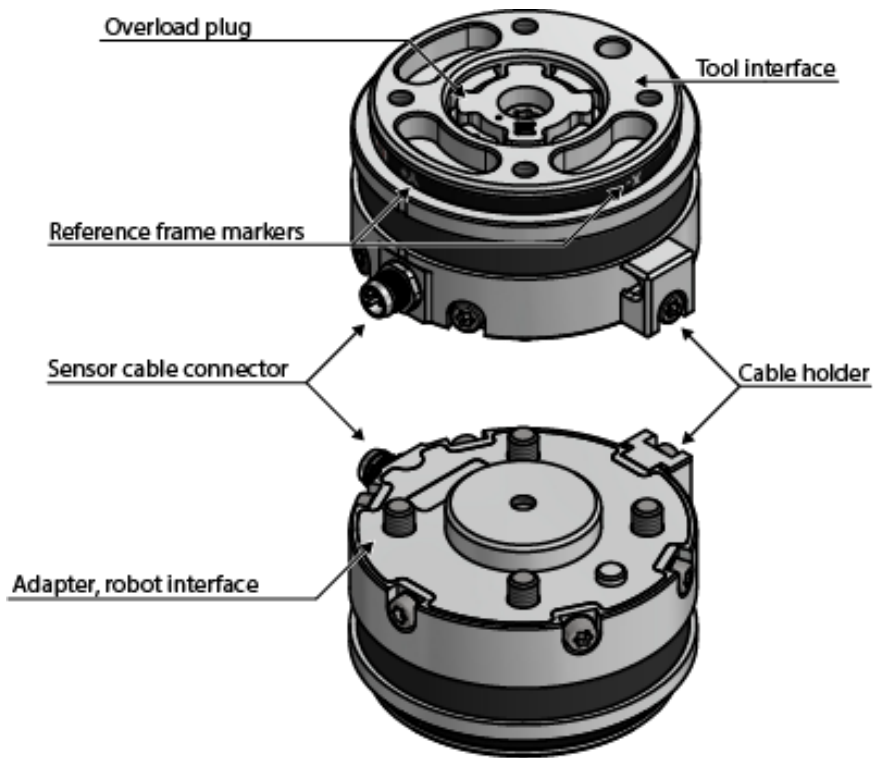
2.2.2 HEX-E v2 and HEX-H v2

The sensor consists of a sensor body, an adapter, and an overload plug. The sensor cable connector, the cable holder, dust sealing, serial number and markers for the reference frame are on the sensor body. The tool is fastened to the sensor body directly, onto the tool interface. The sensor is fastened to the robot tool flange by the adapter.
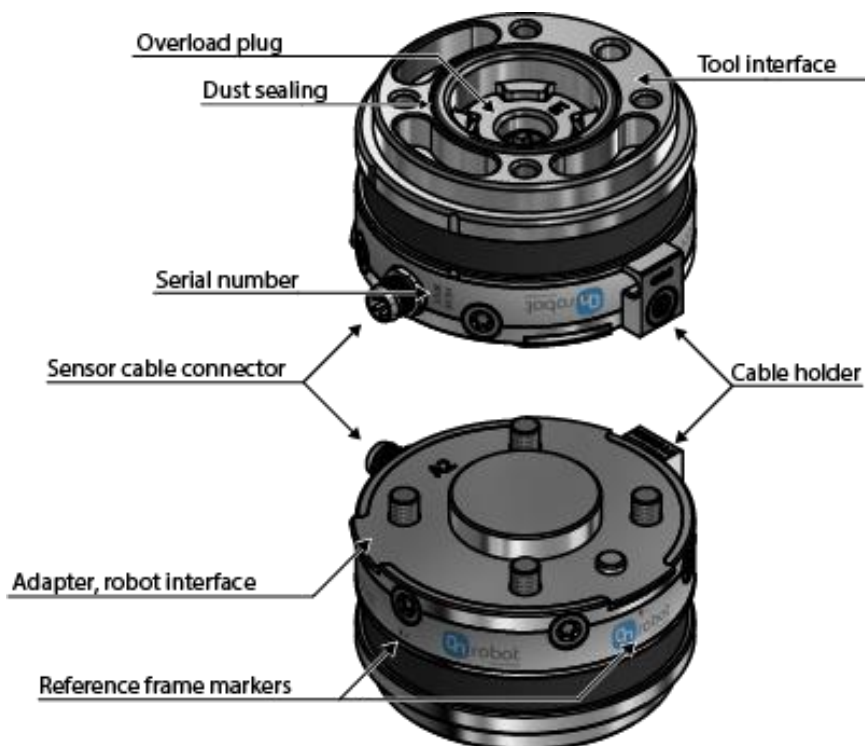
2.3   Mounting

Use only the screws provided with the sensor. Longer screws could damage the sensor or the robot.

2.3.1   HEX-E v1 and HEX-H v1

To mount the sensor, follow this process:

1. Fasten Adapter-A to the Robot by four M6x8 Screws. Use 6 Nm tightening torque.

2. Fasten the Sensor to the adapter by five M4x8 screws with M4 washers. Use 1,5 Nm tightening torque.

3. Fasten the cable to the Sensor with the Cable Holder by one M4x12 screw and M4 washer. Use 1,5 Nm tightening torque.

4. Fasten the Plug to the Sensor by one M6x30 Screw. Use 6 Nm tightening torque.
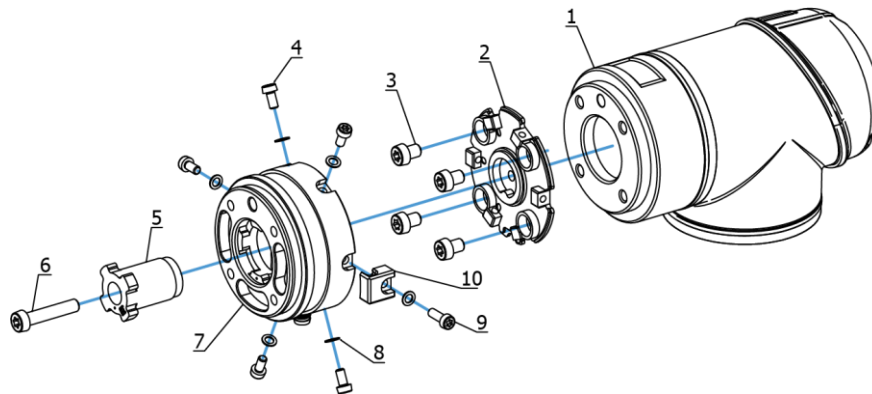
Legend: 1 –robot tool flange, 2 – Adapter A, 3 - M6x8 screws, 4 – M4x8 screws, 5 – overload plug, 6 – M6x30 screw, 7 – sensor, 8 – M4 washer, 9 – M4x12 screw, 10 – cable holder

5. Fasten the tool to the Sensor, according to the instructions from the tool manufacturer.

⚠️ The overload protection is not fully functional, if the tool is not joined to the Sensor with a flat surface.

2.3.2   HEX-E v2 and HEX-H v2

To mount the sensor, follow this process:

1. Fasten Adapter-A2 to the Robot by four M6x8 Torx Screws with M6 washers. Use 6 Nm tightening torque.

2. Fasten the Sensor to the adapter by five M4x6 screws. Use 1,5 Nm tightening torque.

3. Fasten the cable to the Sensor with the Cable Holder by one M4x12 screw. Use 1,5 Nm tightening torque.

4. Fasten the Plug to the Sensor by one M6x30 Screw. Use 6 Nm tightening torque.
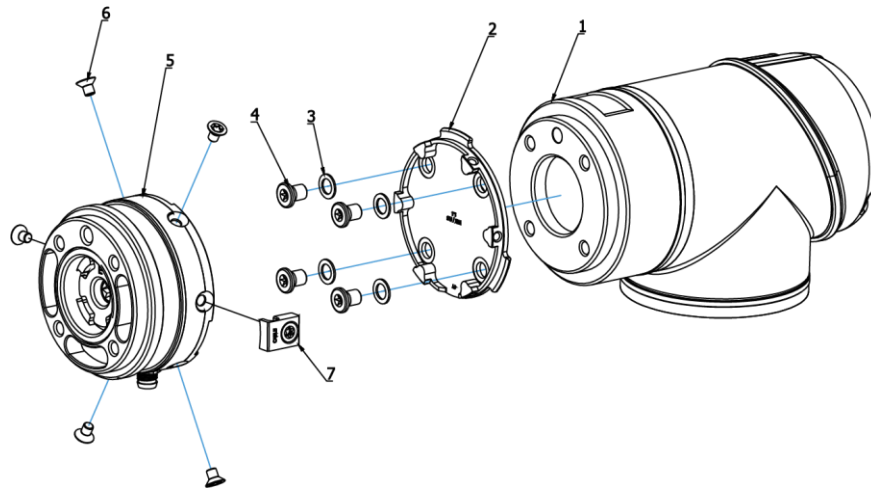
Legend: 1 –robot tool flange, 2 – Adapter- A2, 3 - M6 washer, 4 – M6x8 Torx screws, 5 – sensor, 6 – M4x6 Torx screws, 7 – cable holder

5. Fasten the tool to the Sensor, according to the instructions from the tool manufacturer.

NOTE:

The overload protection is not fully functional, if the tool is not joined to the Sensor with an interface described in ISO 9409-1-50-4-.

2.4 Cable Connections

To connect the sensor, follow this process:

1. Connect the 4 pin M8 cable (5m long) to the sensor. Ensure that the holes of the cable are aligned with the pins of the connector on the sensor.

NOTE:
Do not rotate the cable, only rotate the connector lock.

2. Secure the cable to the robot with cable ties.

NOTE:
Make sure that enough extra cable length is available around the joints for bending.

3. Place the Compute Box somewhere near or inside the UR robot control cabinet and connect the 4 pin M8 sensor cable. The provided cable gland can be used to lead the cable into the UR Control cabinet.

4. Connect the Compute Box's Ethernet interface with the UR controller's Ethernet interface via the supplied UTP cable.

5. Use the 3 pin M8 cable (1m long) to power the Compute Box from the UR's control box. Connect the brown cable to the 24V and the black cable to the 0V.

Power		Configurable Inputs				Configurable Outputs			
PWR	■	24V	■	24V	■	0V	■	0V	■
GND	■	CI0	■	CI4	■	CO0	■	CO4	■
24V	■	24V	■	24V	■	0V	■	0V	■
0V	■	CI1	■	CI5	■	CO1	■	CO5	■
		24V	■	24V	■	0V	■	0V	■
		CI2	■	CI6	■	CO2	■	CO6	■
		24V	■	24V	■	0V	■	0V	■
		CI3	■	CI7	■	CO3	■	CO7	■

For more information, refer to the UR's documentation.

6. Apply the correct network settings to both the Compute Box and the UR robot. The default Compute Box IP address is 192.168.1.1, to change it, see **Changing the IP of the Compute Box**.

2.5 UR Compatibility

Ensure that the robot controller has at least PolyScope version 3.6 (works up to 3.7).

On PolyScope version 3.7 there is a known bug, that the **Save** 🖫 option sometimes does not appear correctly. In this case please use the **Save as** option as a workaround.

2.6 URCap Plugin Installation

To upload the OnRobot examples and install the OnRobot URCap plugin, follow this process:

1. Insert the OnRobot USB drive in the USB slot on the right side of the Teach Pendant.
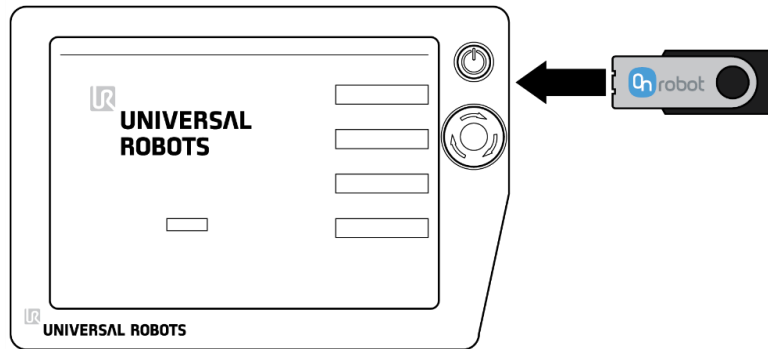
2. A message dialog window appears asking for your permission to continue copying OnRobot examples and URCap file to `programs/OnRobot_UR_Programs` folder.
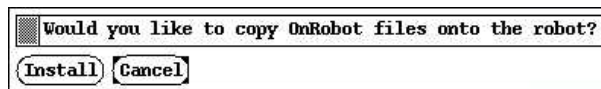
Tap on **Install** to continue.

3. Then select the **Setup Robot** option from the main menu, then the **URCaps Setup** option.

4. Tap on the **+** sign to browse for the newly copied OnRobot URCap file. It can be found in the `programs/OnRobot_UR_Programs` folder. Tap on **Open**.

5. Then the system needs to be restarted for the changes to take effect. Tap on the **Restart** button and then wait for the system to be restarted.
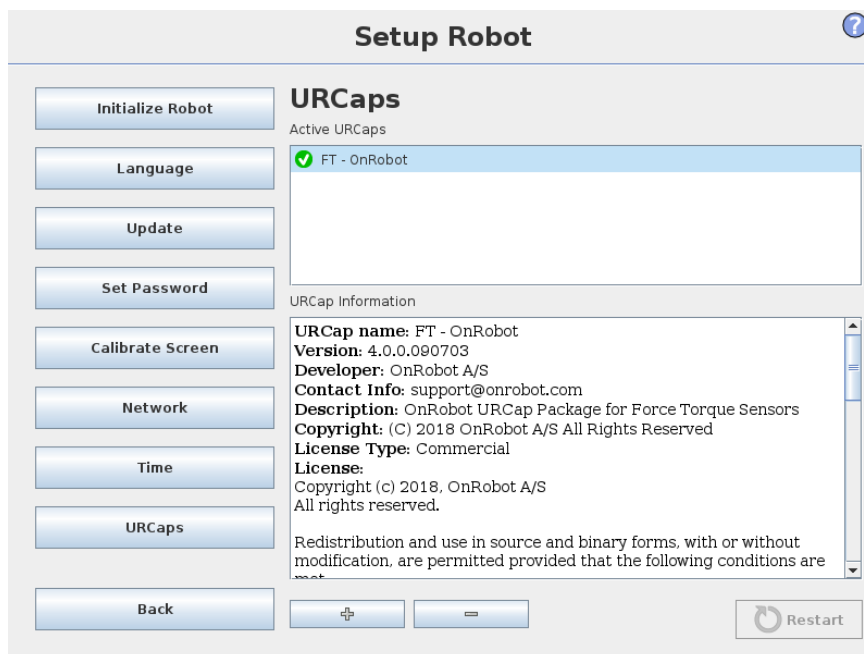
6. Initialize the robot.

> **NOTE:**
>
> For more information on the URCap installation please refer to UR's documentation.

Continue with **URCap Plugin Setup**.

2.7 URCap Plugin Setup

Select the **Installation** tab, then select **OnRobot FT Setup**. The following screen is shown:
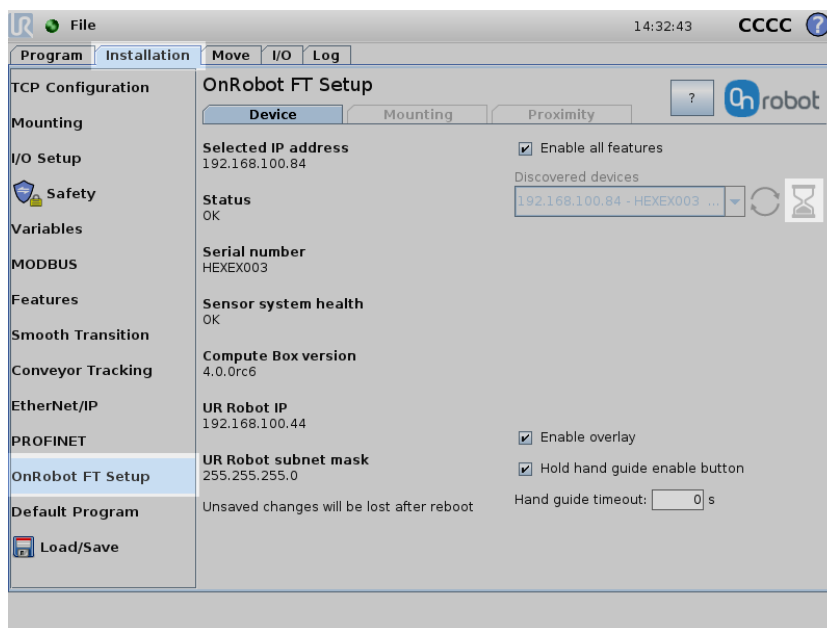
Wait a few seconds while the software is automatically discovering the available OnRobot sensor. The hourglass ⌛ icon indicates that the discovery is still in progress.

Once it finishes the first found device is selected and automatically tested, and the following screen is shown:
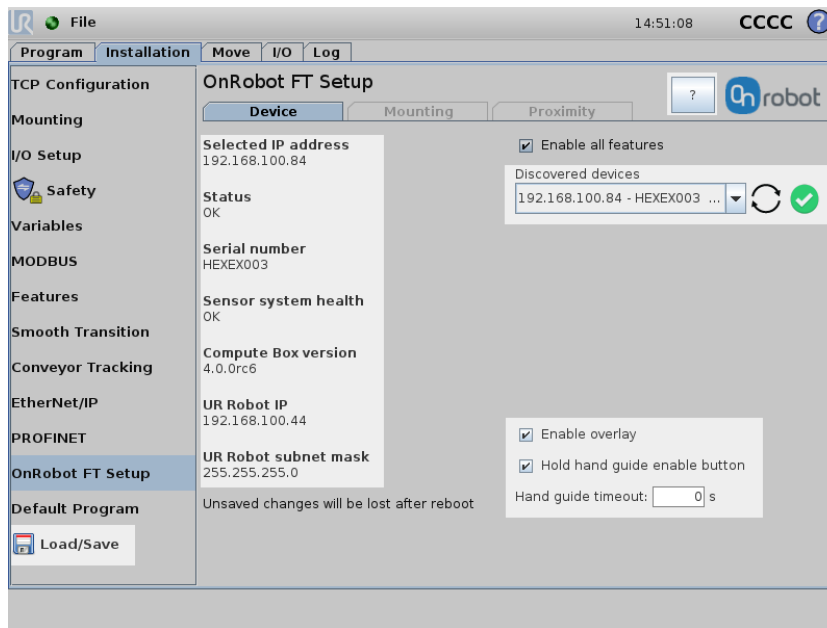
The ok icon shows that the device is found, and the automatic test was successful, so the device is ready to be used.

If no device is found or there was an error during the automatic test, then an error icon is displayed. For troubleshooting, see **URCap Plugin Setup error**.

> **NOTE:**
>
> The discovery can be manually restarted by tapping on the refresh icon.
>
> If multiple devices are available, the pre-selected device could be changed by using the **Discovered devices** drop-down menu.

The status and basic information of the connected device is shown in the left:

Selected IP address: This shows the IP address of the selected device. Using the factory default settings on the Compute Box the value will be 192.168.1.1.

Status: This shows OK or the error message in case of a malfunction.

Serial Number: The serial number of the OnRobot device.

Sensor system health: This shows OK or the error message in case of a malfunction.

Compute Box version: The software version of the Compute Box. This must match the URCap version. If it does not match please update the Compute Box.

The current network settings of the UR robot are showed to help the troubleshooting in case of an error:

UR Robot IP: This shows the current IP address if the robot. Using the factory default settings on the Compute Box the value must be 192.168.1.x.

UR Robot Subnet mask: The current subnet mask of the robot. Using the factory default settings on the Compute Box the value must be 255.255.255.0.

The Hand Guide settings are in the bottom left:

Hold Handguide Enable checkbox: If checked (default value) the Hand Guide enable button needs to be pressed constantly during the hand guiding. If unchecked, the hand guiding could be started by tapping on the enable button and stopped by tapping on the enable button again.

Hand Guide Timeout: After the set timeout value (in seconds) the hand guiding will be stopped automatically. The default value is 0 that sets the timeout to infinite.

NOTE:

After setting up the device the changes need to be saved to be part of the current installation with the Load/Save button.

For embedded help, tap on the question mark icon.

3   Using the URCap Plugin

3.1   OnRobot Feedback Variables

Simple features are demonstrated in this section through an example program. The program shows how to gain data from the OnRobot sensor and how to zero the Force/Torque values of the sensor.

1. Click on Program Robot.
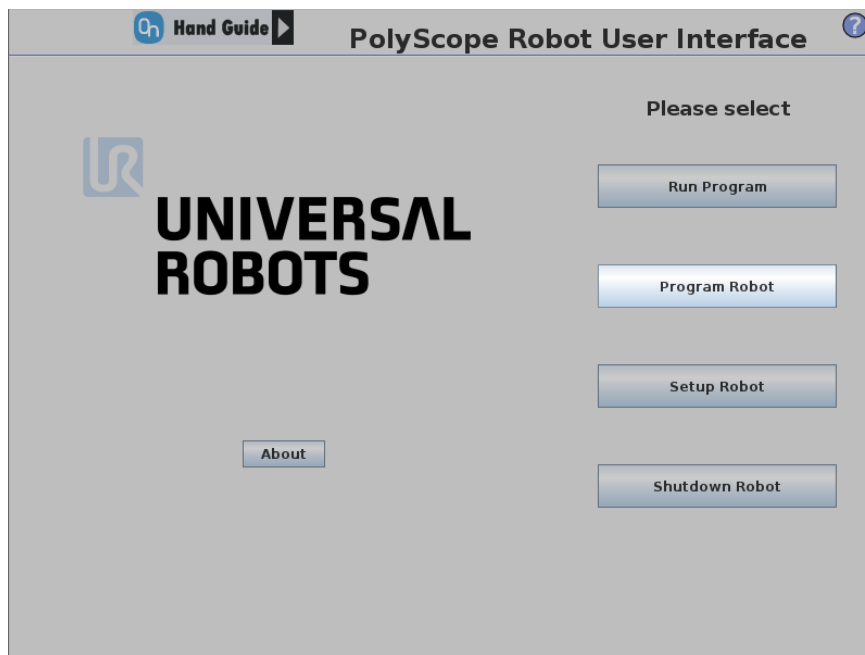
2. Click on Empty Program.
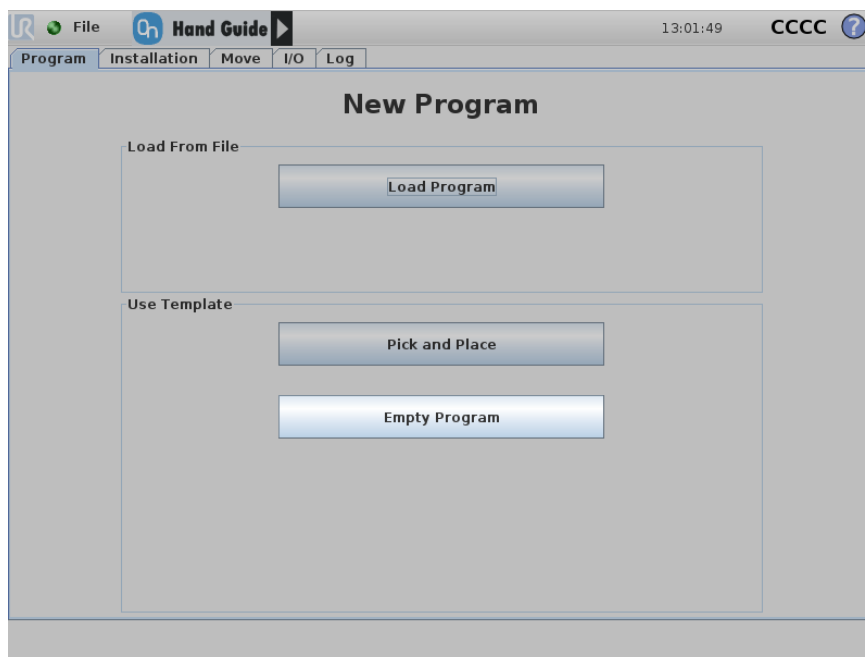
3. Select the **Structure** tab.

4. Press the **Wait** button to avoid infinite loop in the program.
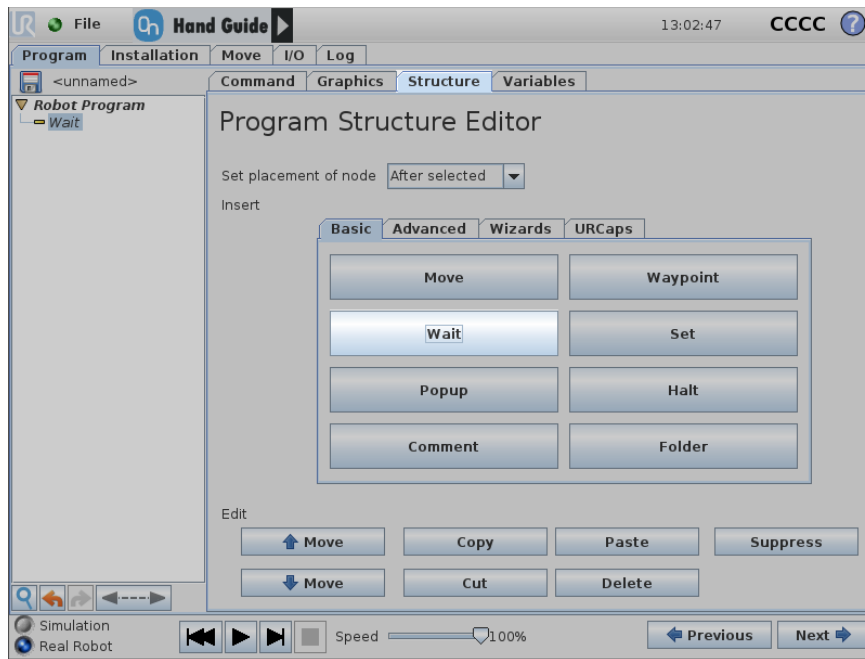
5. Select the **Wait** command in the program structure.

6. Select the **Command** tab.

7. Set the **Wait** to 0.01 seconds.

8. Press the Play button to execute program.
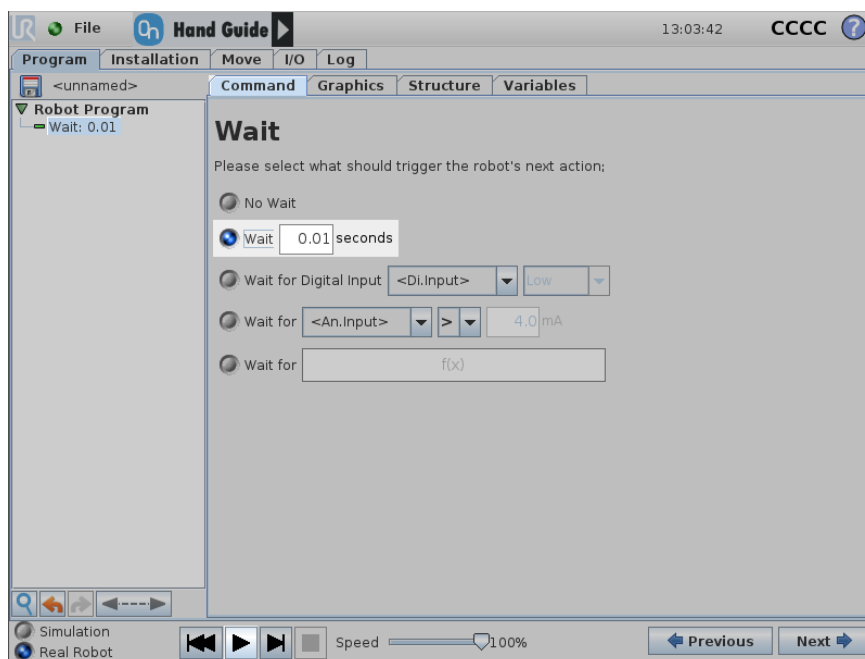
9. Select the **Variables** tab.
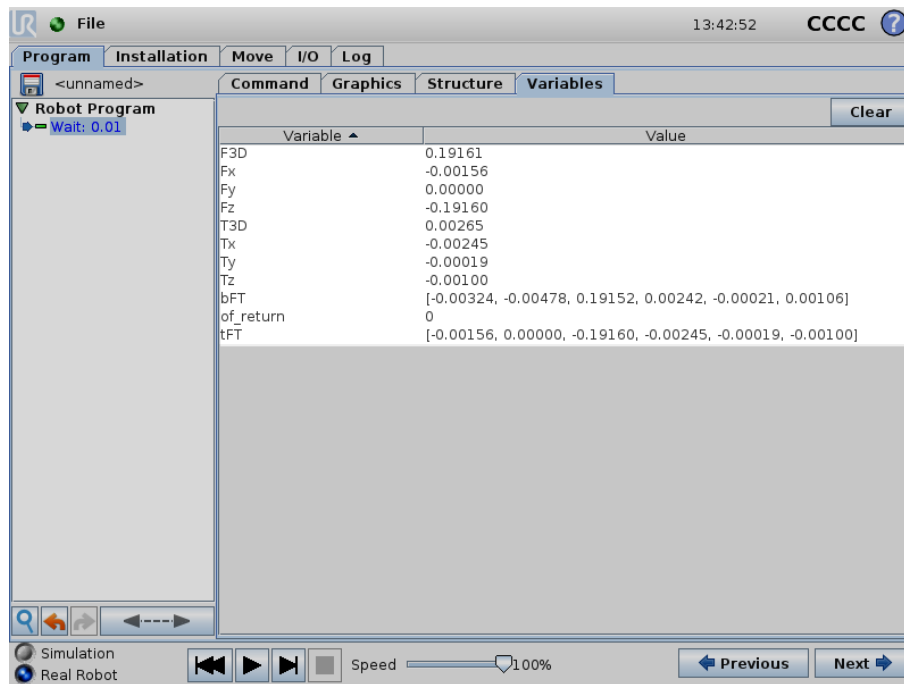
The Force values and Torque values are visible. You can use these variables in any program.

These variables are updated automatically at a rate of approximately 125Hz:

- **F3D**: Length of the 3D force vector F3D = sqrt (Fx^2 +Fy^2 +Fz^2) (N)

- **Fx**: Force vector in the X direction in Newton (N)

- **Fy**: Force vector in the Y direction in Newton (N)

- **Fz**: Force vector in the Z direction in Newton (N)

- **T3D**: Length of the 3D torque vector T3D = sqrt (Tx^2 +Ty^2 +Tz^2) (Nm)

- **Tx**: Torque in the X direction in Newton Meter (Nm)

- **Ty**: Torque in the Y direction in Newton Meter (Nm)

- **Tz**: Torque in the Z direction in Newton Meter (Nm)

- **bFT**: Force and torque values calculated in the Base Coordinate system, in an array in Newton (N) and Newton Meter (Nm)

- **of_return**: the variable used to store the outcome of OnRobot commands

- **tFT**: Force and torque values calculated in the Tool Coordinate system, in an array in Newton (N) and Newton Meter (Nm)

3.1.1 Effects of the TCP Position

The torques are calculated based on the Tool Center Point, meaning the torque exerted by the measured forces is calculated in the Tool Center Point, not on the sensor face. See the effects of the TCP placement on the measured torque in the figure below.
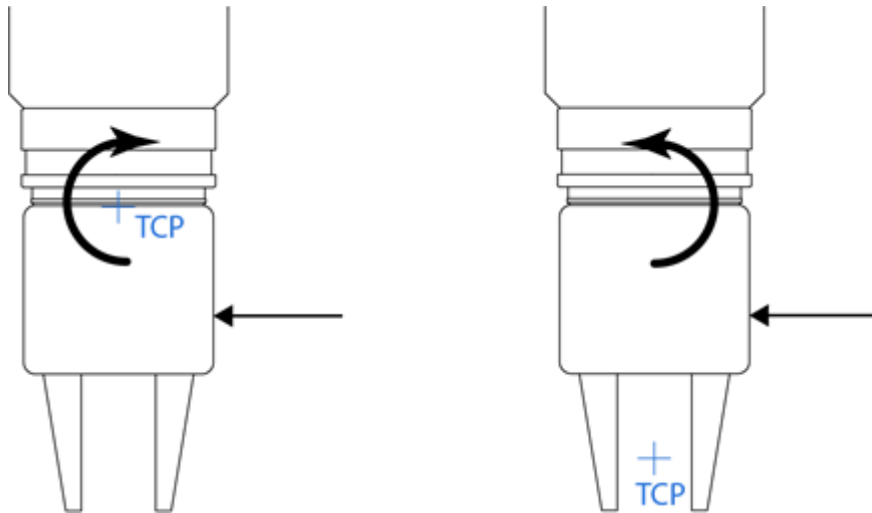
3.2   OnRobot Hand Guide Toolbar

After turning on the UR robot, the start screen of the PolyScope is visible. After 20 seconds, if it is activated, the OnRobot Hand Guide Toolbar appears on the top right.
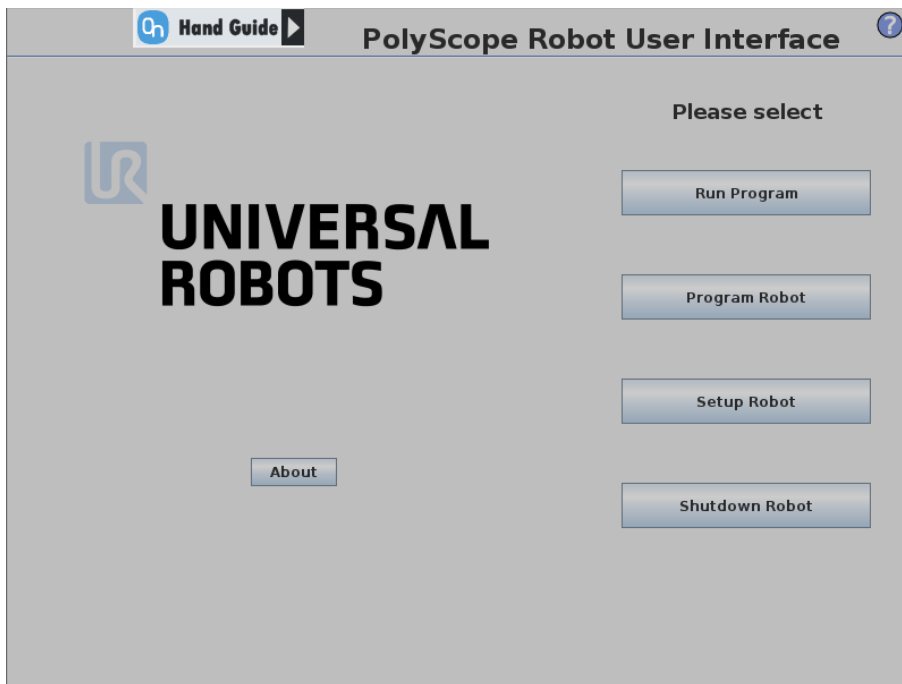
 NOTE:

It is normal to have a yellow warning signal during the boot-up for a few seconds. If it does not disappear than check the device settings at the **URCap plugin setup**.

To activate the functions of the toolbar, tap on any point of the toolbar. The toolbar expands, and the available axes, the enable button, the zero button and the snap to axes button will appear.

To select an axis, press the appropriate item. In the following example, the X and the Y items are selected to restrict the movement along the X and Y axis (planar):

 NOTE:

The used coordinate system is the Tool.

To deactivate any axis that has been selected, press the item again.

OnRobot © 2018   www.onrobot.com

NOTE:

It is possible to activate or deactivate axes during hand guiding.

To start hand guiding the UR robot, first make sure that you do not touch the tool and then press and hold the enable ⏻ button. The button turns to an hourglass ⌛ icon while the hand guiding is being initiated. Wait until the enable ⏻ button turns green and drive the robot by hand with the help of the OnRobot finger sensor.

NOTE:

Make sure that you do not touch the tool before the hand guiding is activated (enable ⏻ button turned to green), otherwise the robot can behave abnormally (e.g..: the robot could move without any external force exerted). In this case, press the zero ⊢0⊣ button, while you are not touching the tool. Make sure that you do not use the zero ⊢0⊣ button while touching the tool.

To stop hand guiding the UR Robot, release the enable ⏻ button. Immediately after deactivating hand guiding, the enable ⏻ button is deactivated for 1 second, and turns into an hourglass ⌛ icon.

NOTE:

Please always set the speed slider of the robot to 100% while using the hand guiding to have an optimal user experience.

The zero ⊢0⊣ button is intended to be used when the tool orientation is changed during hand guiding, so that the effects of gravity or changes in the load of the robot can be neutralized.

The snap to axes button rotates the axes of the tool coordinate system to align with the closest axes of the base coordinate system, disregarding negative or positive directions. This allows the user to set the tool to face precisely horizontally, or vertically, after guiding by hand.

3.3 OnRobot URCap Commands

3.3.1 F/T Center

It moves the robot along the given axis until it finds an obstacle. After the collision, it moves to the opposite direction until another collision is reached. After that the robot calculates the middle of the two boundary points and moves to that point.

> **NOTE:**
>
> To cancel any force/torque offset, execute an `F/T Zero` command at the beginning of the `F/T Control` command and make sure the tool is not in contact with any object before starting the `F/T Control`, otherwise the command may not work properly.
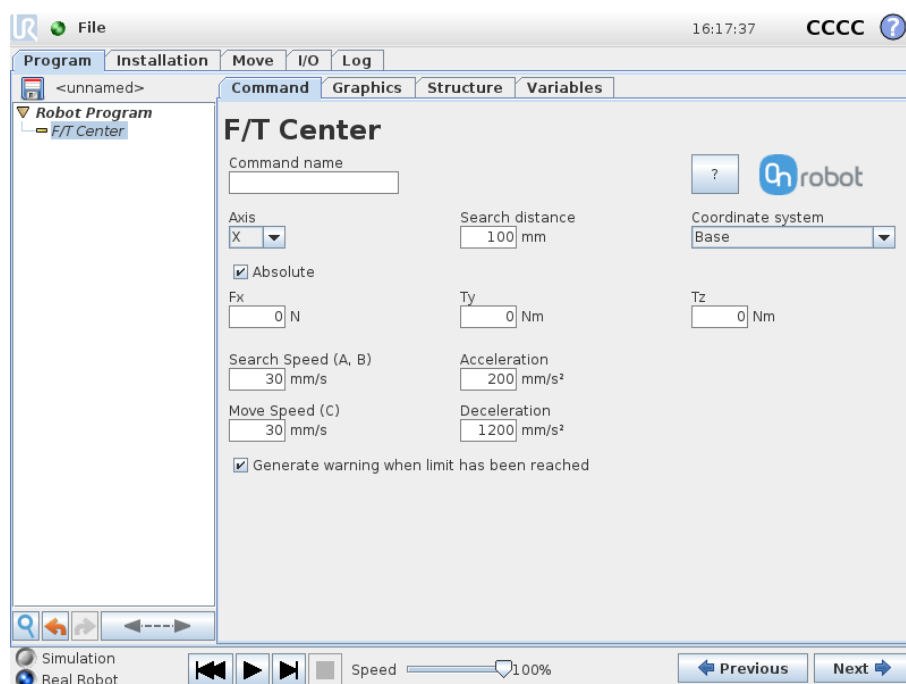
Axis: Defines whether a translational movement will be carried out along the X, Y or Z axis, or a rotational movement (RX, RY or RZ). Only one axis can be selected.

Search distance: The distance from the starting point how far the command can move the robot (in both directions). Make sure that it is big enough, otherwise it will not find the right center point.

Force/torque limits (Fx, Ty, Tz): This is the detection limit. The set axis defines the available force/torque values that can be used as a limit.

> **Absolute** checkbox: If checked the force or torque value sign is going to be checked not just the magnitude.

NOTE:

Only one of the force/torque options can be active at a time. To change the one that is used clear the previous one (delete the contents of the field) and then set the new one.

Search Speed A, B: The movement speed while searching for collision

NOTE:

The slower the speed during the searching phase is the better to work with hard contacts (such as metal surfaces) to avoid overshoots due to the robot's and the tool's momentum.

Move Speed C: The movement speed once the center point is calculated and moves towards that point.

Acceleration: The acceleration parameter of the movement (shared parameters across the A, B, and C section).

Deceleration: The deceleration parameter of the movement (shared parameters across the A, B, and C section).

Coordinate system: The coordinate system used both for the movement and for the sensor reading. It can be set to `Base` or `Tool` (according to the UR's reference frames).

Generate warning (…): If enabled then a pop-up message (blocking) appears once the set limits are reached or exceeded (center point could not be found). If the center point is found, then no warning is displayed.

If disabled, then no pop-up message is shown but the user can handle any possible errors by the return value of the command.

For return values, see **F/T Center Command Return Values**.

3.3.2 `F/T Control`

The main purpose of the `F/T Control` command is to provide easy-to-use functions to application programmers, who want to develop force-controlled applications such as polishing, sanding or grinding. A large subset of these applications may require holding constant force/torque to a defined direction during movements.

The command tries to keep the set force/torque values constant along/about the axes set to be compliant while the commands under the `F/T Control` is executed. The `F/T Control` command does not control forces in the direction that the tool is moving using the `F/T Move,` the `F/T Search,` and the `F/T Path` commands.

NOTE:

UR's built-in Move commands cannot be used under the `F/T Control` command. To move the robot under the force control, use the `F/T Move` or `F/T Search` command instead.

NOTE:

To cancel any force/torque offset, execute an `F/T Zero` command at the beginning of the `F/T Control` command and make sure the tool is not in contact with any object before starting the `F/T Control`, otherwise the command may not work properly.
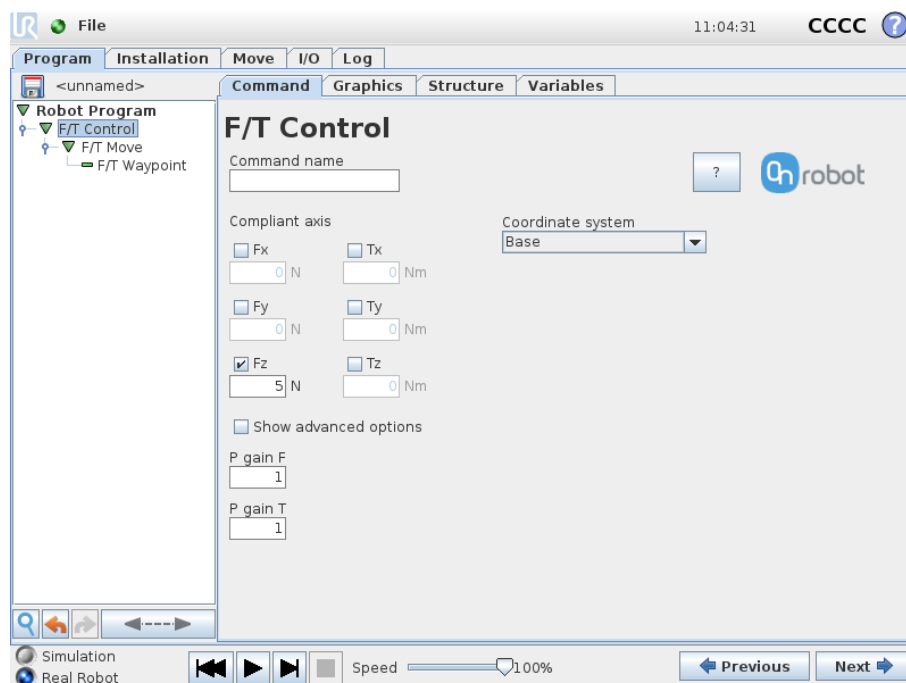
Compliant axis Fx, Fy, Fz, TX, TY, TZ: The axis selection that needs to be compliant. If an axis is enabled (compliant) the movement along/about that axis is force/torque controlled

---

otherwise (non-compliant) position controlled. The enabled axis is controlled to keep the set force/torque value constant. At least one compliant axis must be selected.

Coordinate system: The coordinate system used both for the movement and for the sensor reading. It can be set to `Base, Tool, Custom (Base) Custom (Tool)` (according to the UR's reference frames). The Custom coordinate systems are calculated from the basis coordinate system and the given **Roll**, **Pitch** and **Yaw** values. For the Custom (Base) coordinate system, it is also possible to use the **Get TCP orientation** button to specify the orientation of the coordinate system, by the orientation of the current TCP. To test the given orientation, the **Rotate tool to this orientation [HOLD]** button can be used.

P Gain F: The force controller can be tuned with this proportional gain parameter. If any overshoots or vibrations occur, try lowering the gain value (e.g.: 0.5).

P Gain T: The torque controller can be tuned with this proportional gain parameter. If any overshoots or vibrations occur, try lowering the gain value (e.g.: 0.5).

Show advanced parameters checkbox: If checked then more options become available:
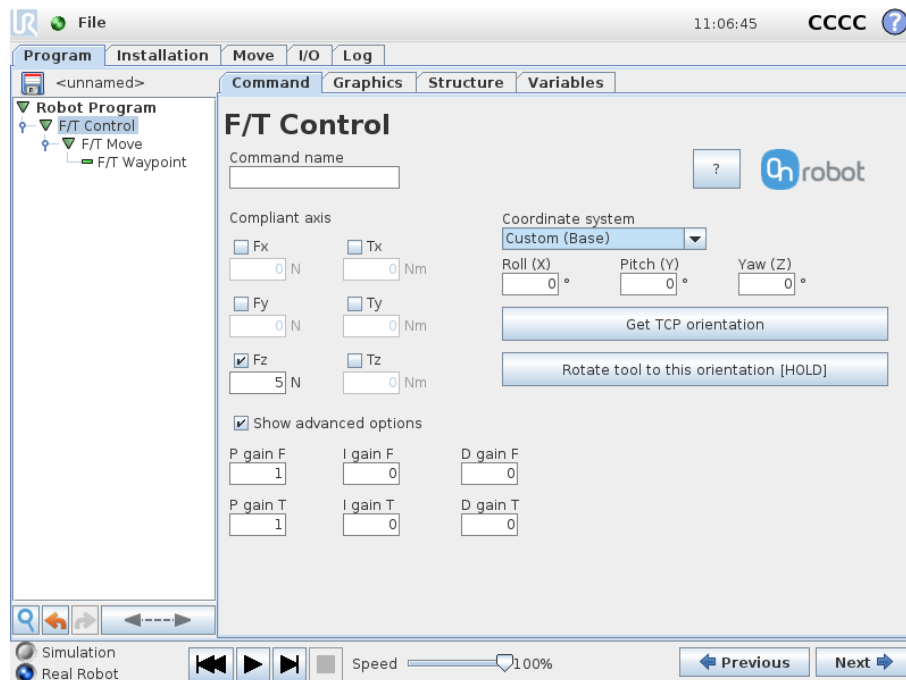
I Gain F: The force controller can be tuned with this integral gain parameter. If any overshoots or vibrations occur, try lowering the gain value.

I Gain T: The torque controller can be tuned with this integral gain parameter. If any overshoots or vibrations occur, try lowering the gain value.

D Gain F: The force controller can be tuned with this derivative gain parameter. If any overshoots or vibrations occur, try lowering the gain value.

D Gain T: The torque controller can be tuned with this derivative gain parameter. If any overshoots or vibrations occur, try lowering the gain value.

This command has no return value.

Guidelines to PID force/torque controller settings:

The PID force/torque controller continuously calculates the error value for the force/torque measured by the sensor, compared to the values set by the `F/T Control` command, and applies correction based on this error.

P Gain: The proportional term produces a correction that is proportional to the current error value. Increasing this parameter, has the following effects: faster reaction, overreaction, lower error, stability degradation.

I Gain: The integral term produces a correction that is proportional to both, the magnitude, and duration of the past error values. Increasing this parameter, has the following effects: faster reaction, overreaction, lower error, stability degradation.

D Gain: The derivative term produces a correction that is proportional to the slope or changing speed of past error values. Increasing this parameter, has the following effects: less overreaction, stability increase.

If the force control is too slow, that is, the tool occasionally leaves the surface instead of continually touching it, try increasing the **P Gain**, and **I gain** values.

If the force control is overreacting to changes, that is, the tool bounces off the surface, try decreasing the **P Gain** (or **D Gain**, if it is above 1).

If the force control is reacting to changes too slowly, that is it keeps pushing the surface hard after touching it, try decreasing the **I Gain**.

As a rule of thumb, it is recommended to use values:

6. P Gain < 5
7. I Gain < 0.25
8. D Gain < 1
9. The ratio of P Gain/I Gain = 10

Values that can be used as a base for tuning are:

P Gain F =1, I Gain F = 0.1, D Gain F = 0.3

P Gain T =0.2, I Gain T = 0, D Gain T = 0

3.3.3 `F/T Stacking`

The `F/T Stacking` command contains Stack and Destack functionality.

Type: The selector between F/T Stack and F/T Destack.

3.3.3.1 `F/T Stack`

The `F/T Stack` command tries to search for the stack top and then executes the user's place sequence (for example opening the gripper) then exits. It tracks how many items are stacked that makes easy to handle if the stack is full. It also works with items that has varying item thickness.

> **NOTE:**
>
> To cancel any force/torque offset, execute an `F/T Zero` command at the beginning of the `F/T Stack` command and make sure the tool is not in contact with any object before starting the `F/T Stack`, otherwise the command may not work properly.
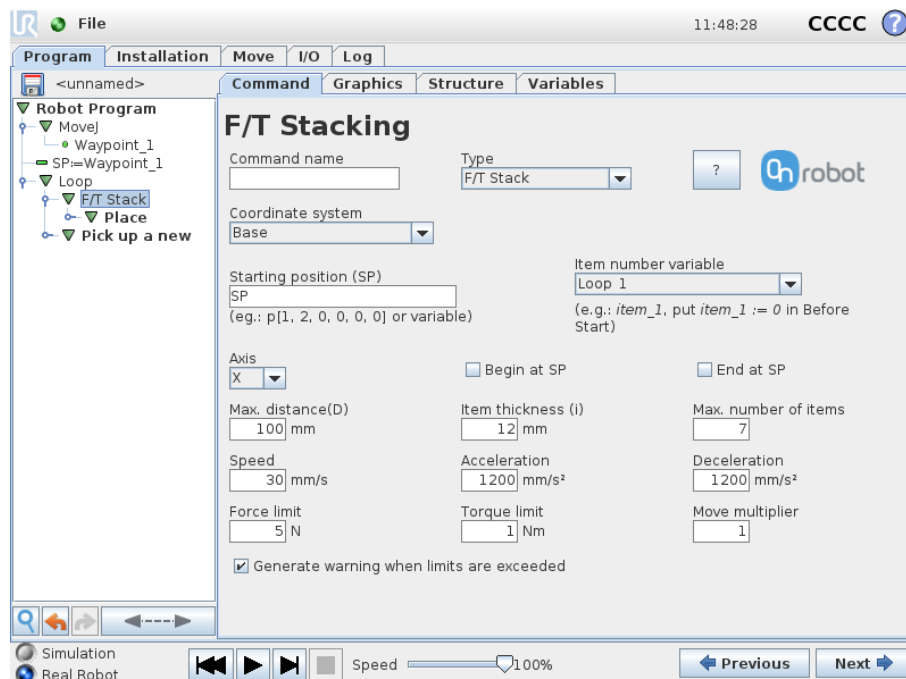
Coordinate system: The coordinate system used both for the movement and for the sensor reading. It can be set to `Base` or `Tool` (according to the UR's reference frames).

Starting position (SP): The starting pose can be defined by a constant such as `p[1,2,3,4,5,6]` or by a variable. It needs to be higher than the full stack's top.

Item number variable: The variable that is used to track how many items are stacked successfully. Enter here the variable name that you have previously defined and set to 0. (E.g.: Use the built-in UR Assignment command `item_1 := 0` in the Before Start section of your program).

Axis: The axis along the stacking is performed (X,Y or Z).

Begin at SP: If enabled, the command will start with moving to the Starting position (SP) at the beginning of its execution.

End at SP: If enabled, the command will exit with moving to the Starting position (SP) at the end of its execution.

Max. distance (D): The stopping distance along the defined axis. It is measured from the Starting position (SP) and needs to be more than the size of the full stack. The sign defines which direction the stacking is performed along the given axis.

Item thickness (i): The stacked items thickness.

Max. number of items: Defines how many items could be stacked, so how many stacked item makes the stack full.

Force limit: The force limit for the collision detection to find the stack top.

Torque limit: The torque limit for the collision detection to find the stack top.

Speed: The movement speed while searching for the stack top. (m/s, rad/s)

NOTE:

The slower the speed during the searching phase is the better to work with hard contacts (such as metal surfaces) to avoid overshoots due to the robot's and the tool's momentum.

Acceleration: The acceleration parameter of the movement.

Deceleration: The deceleration parameter of the movement.

Move multiplier: Defines how many times of the given speed and force/torque limit is used while the robot is not searching the top of the stack but moving to/from the starting point.

Generate warning (…): If enabled then a pop-up message (blocking) appears if the next item is not found or the stack is full.

If disabled, then no pop-up message is shown but the user can handle any possible errors by the return value of the command.

For return values, see **F/T Stacking Command Return Values**.

3.3.3.2 F/T DeStack

The `F/T Destack` command tries to search for the stack top and then executes the user's pick sequence (for example: closing the gripper). It tracks how many items are destacked that makes easy to handle if the stack is empty. It also works with items that has varying item thickness.

NOTE:

To cancel any force/torque offset, execute an `F/T Zero` command at the beginning of the `F/T Stack` command and make sure the tool is not in contact with any object before starting the `F/T Stack`, otherwise the command may not work properly.
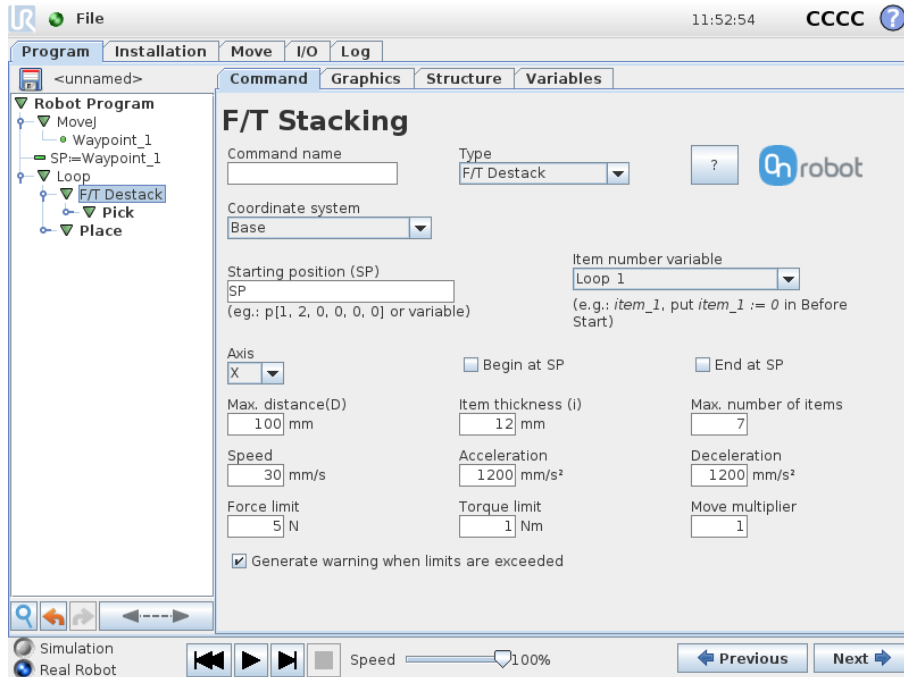
Coordinate system: The coordinate system used both for the movement and for the sensor reading. It can be set to `Base` or `Tool` (according to the UR's reference frames).

Starting position (SP): The starting pose can be defined by a constant such as `p[0.1,0.2,0.3,0.9,0.8,0.7]` or by a variable. It needs to be higher than the full stack's top.

Item number variable: The variable that is used to track how many items are destacked successfully. Enter here the variable name that you have previously defined and set to 0. (E.g.: Use the built-in UR Assignment command `item_1 := 0` in the Before Start section of your program).

Axis: The axis along the destacking is performed (X,Y or Z).

Begin at SP: If enabled, the command will start with moving to the Starting position (SP) at the beginning of its execution.

End at SP: If enabled, the command will exit with moving to the Starting position (SP) at the end of its execution.

Max. distance (D): The stopping distance along the defined axis. It is measured from the Starting position (SP) and needs to be more than the size of the full stack. The sign defines which direction the destacking is performed along the given axis.

Item thickness (i): The stacked items thickness.

Max. number of items: Defines how many items could be destacked, so how many destacked item makes the stack empty.

Force limit: The force limit for the collision detection to find the stack top.

Torque limit: The torque limit for the collision detection to find the stack top.

Speed: The movement speed while searching for the stack top.

NOTE:

The slower the speed during the searching phase is the better to work with hard contacts (such as metal surfaces) to avoid overshoots due to the robot's and the tool's momentum.

Acceleration: The acceleration parameter of the movement.

Deceleration: The deceleration parameter of the movement.

Move multiplier: Defines how many times of the given speed and force/torque limit is used while the robot is not searching the top of the stack but moving to/from the starting point.

Generate warning (…):   If enabled then a pop-up message (blocking) appears if the next item is not found or the stack is empty.

If disabled, then no pop-up message is shown but the user can handle any possible errors by the return value of the command.

For return values, see **F/T Stacking Command Return Values**.

3.3.4 `F/T Fix and Rotate`

First position the object that needs to be inserted to the socket pointing in the right direction and close to the socket entrance. The final position and orientation will be corrected by the `F/T Fix and Rotate` command. It tries to push the object with the predefined force limit until the defined insertion depth is reached, and then adjusts the orientation if necessary.

> **NOTE:**
>
> It is important to set the TCP (Tool Center Point) at the tip of the object.

> **NOTE:**
>
> To cancel any force/torque offset, execute an `F/T Zero` command at the beginning of the `F/T Fix and Rotate` command and make sure the tool is not in contact with any object before starting the `F/T Fix and Rotate`, otherwise the command may not work properly.
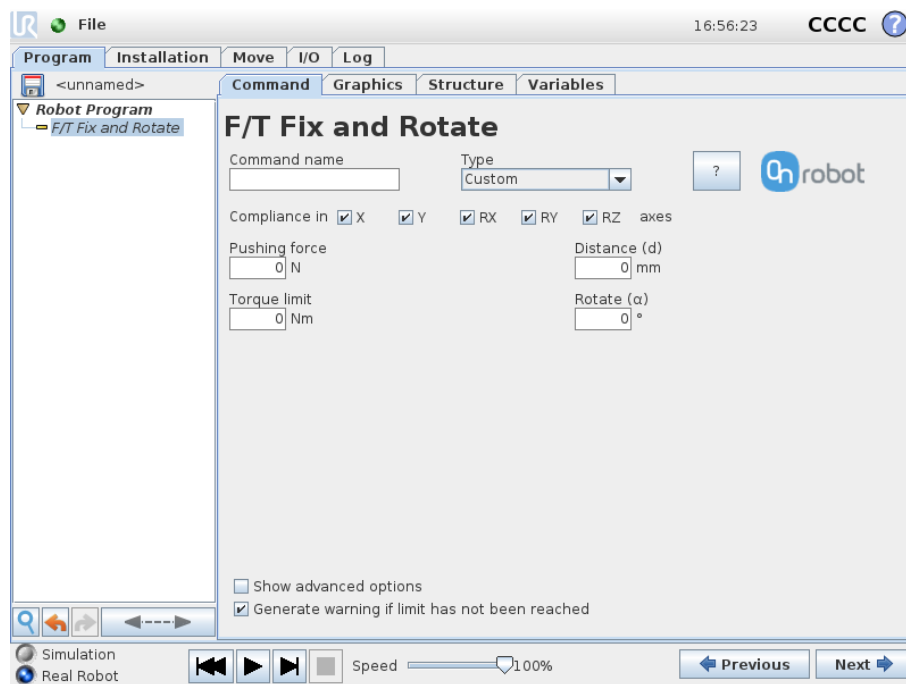
Compliance in X, Y, RX, RY, RZ axes checkboxes: Insertion is performed along the Z axis of the Tool coordinate system. To adopt to any positioning error, the remaining axes (X and Y for the translation and X, Y, and Z for the rotation) can be set to move freely.

Pushing force: The force target used for the force control to gently push the object into the socket.

Distance (d): The distance from the Starting point along the Z axis (in Tool coordinate system).

Torque limit: During the rotation phase this limit is used to finish the movement. The lower the limit the more careful the rotation is.

Rotate (α): The angle for the rotation about the Z axis of the Tool coordinate system.

Show advanced options: If enabled then more options become available:
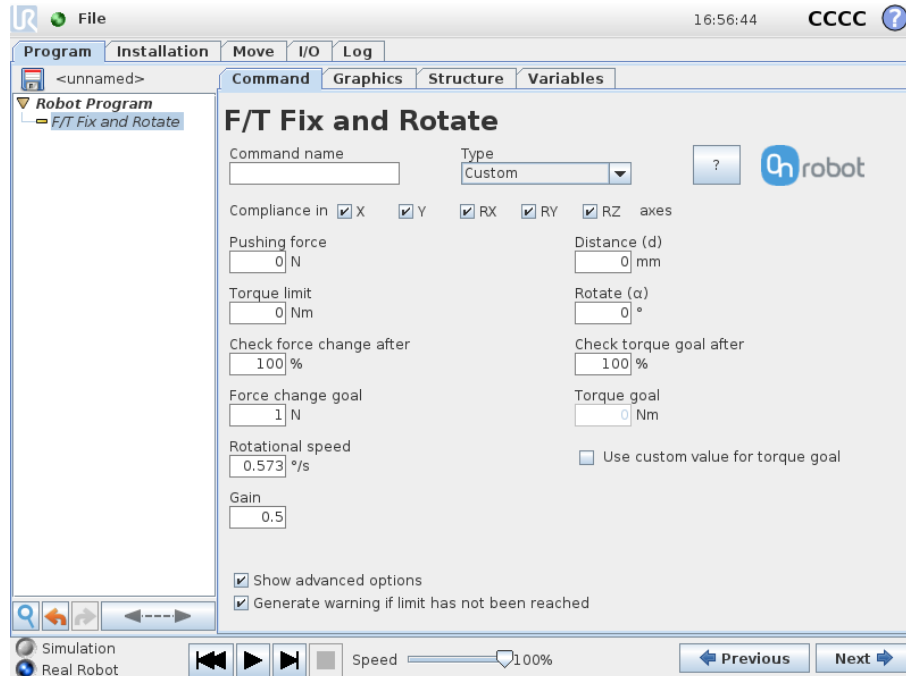
Check force change after: After the object is close to the bottom of the socket, the "bump checking" is activated. The limit how close the object needs to be set by a percent of the **Distance**.

Check torque goal after: During the rotational phase after the set percentage of the **Rotate (α)** angle, the torque goal checking is activated.

Force change goal: During the insertion, after the **Check force change after** percent of the **Distance** is reached the force checking is activated. The force checking is used to monitor whether the connector is pushed to the bottom of the socket. This can be set by an additional force limit that is the **Force change goal** value. The push to the bottom of the socket is reached when the force value equals or exceeds the **Pushing force** + the **Force change goal**.

Torque goal: The set torque value that is going to stop the rotational phase.

> **Use custom value for torque goal:** Check it to be able to set a custom torque goal.

Rotational speed: The rotational speed during the rotational phase.

Gain: The gain parameter of the force and torque control. The default value is 0.5. The smaller the value the more accurate the control of the set pushing force is.

Generate warning (…): If enabled then a pop-up message (blocking) appears if the insertion was not successful.

If disabled, then no pop-up message is shown but the user can handle any possible errors by the return value of the command.

For return values, see **F/T Fix and Rotate Command Return Values**.

3.3.5 `F/T Guard`

Every UR commands that are put under the `F/T Guard` will be executed, but the robot will stop once one of the set limits is reached. The force limiting can be mixed with an external I/O signal (e.g.: stop if Fz>5 AND digital_in[7] == True).

> **NOTE:**
>
> To cancel any force/torque offset, execute an `F/T Zero` command at the beginning of the `F/T Guard` command and make sure the tool is not in contact with any object before starting the `F/T Guard`, otherwise the command may not stop at the given force/torque limit.

Coordinate system: The coordinate system used both for the movement and for the sensor reading. It can be set to `Base` or `Tool` (according to the UR's reference frames).

Force/torque limit: This is the detection limit. From the Fx, Fy, Fz, Tx, Ty, Tz, F3D, T3D available options more than one can be set. In this case, if any of those values reached the set threshold the stop is triggered. The values equal to zero are neglected.

> If the **Absolute** option is enabled, then it is not important whether the entered value is positive or negative (e.g.: stop if |Fz| > = 3), otherwise the sign defines how the threshold is calculated (e.g..: stop if Fz > = 3 or stop if Fz <= -3)

Show advanced options: If enabled then more options become available:

If the **In addition to the force/torque limits…** is enabled, then the set digital I/O will also be monitored and once the condition is met (along with the force/torque limit) the robot will be stopped. (e.g.: stop if Fz>5 AND digital_in[7] == True).

This command has no return value, and halts the program when limits are reached.

3.3.6 `F/T Insert Box`

First position the object close to the entrance of the hole and start from a tilted orientation (α). It will move the object in phase A along the predefined axis (for example Z) if the edge of the hole is not found. Optionally in phase B another edge can be found (for example the side of the hole). In phase α the orientation is changed so that the object is aligned with the hole (user must set the right angle). Finally, the object is being inserted (along the axis defined in phase A) up to the remaining insertion depth. If the force and torque limits are exceeded a warning is generated.

> **NOTE:**
>
> It is important to set the TCP (Tool Center Point) at the tip of the part.

> **NOTE:**
>
> To cancel any force/torque offset, execute an `F/T Zero` command at the beginning of the `F/T Insert Box` command and make sure the tool is not in contact with any object before starting the `F/T Insert Box`, otherwise the command may not stop at the given force/torque limit.
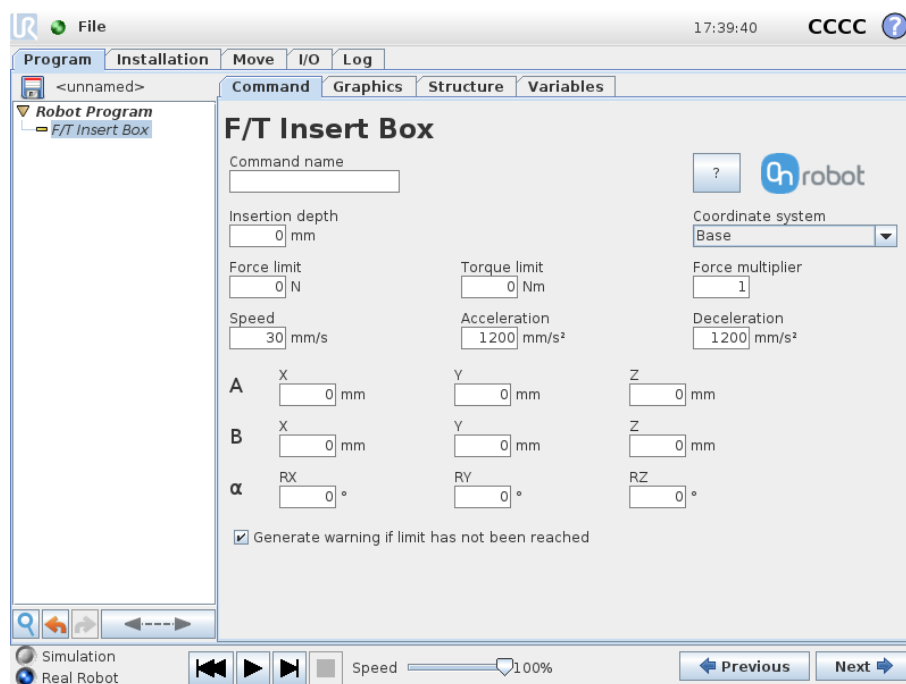
Insertion depth: The distance from the Starting point along the defined axis in phase A.

Coordinate system: The coordinate system used both for the movement and for the sensor reading. It can be set to `Base` or `Tool` (according to the UR's reference frames).

Force limit: The force limit for the edge detection.

Torque limit: The torque limit for the orientation adjustment.

Force multiplier: The force limit for the edge detection is multiplied by this value, to calculate the force limit for the final insertion.

Speed: The movement speed during insertion.

Acceleration: The acceleration parameter of the movement.

Deceleration: The deceleration parameter of the movement.

A: The relative coordinates of the A movement.

B: The relative coordinates of the B movement.

α: The relative angles of the α rotation.

Generate warning (…): If enabled then a pop-up message (blocking) appears if the insertion was not successful.

If disabled, then no pop-up message is shown but the user can handle any possible errors by the return value of the command.

For return values, see **F/T Insert Box Command Return Values**.

3.3.7 `F/T Insert Part`

First position the pin or peg that needs to be inserted to the hole pointing in the right direction and close to the hole entrance. The finial position and orientation will be corrected by the `F/T Insert Part` command. It tries to push the pin with the predefined force limit and adjusts the orientation if necessary. It stops when the defined insertion depth is reached.

> **NOTE:**
>
> It is important to set the TCP (Tool Center Point) at the tip of the part.

> **NOTE:**
>
> To cancel any force/torque offset, execute an `F/T Zero` command at the beginning of the `F/T Insert Part` command and make sure the tool is not in contact with any object before starting the `F/T Insert Part`, otherwise the command may not stop at the given force/torque limit.
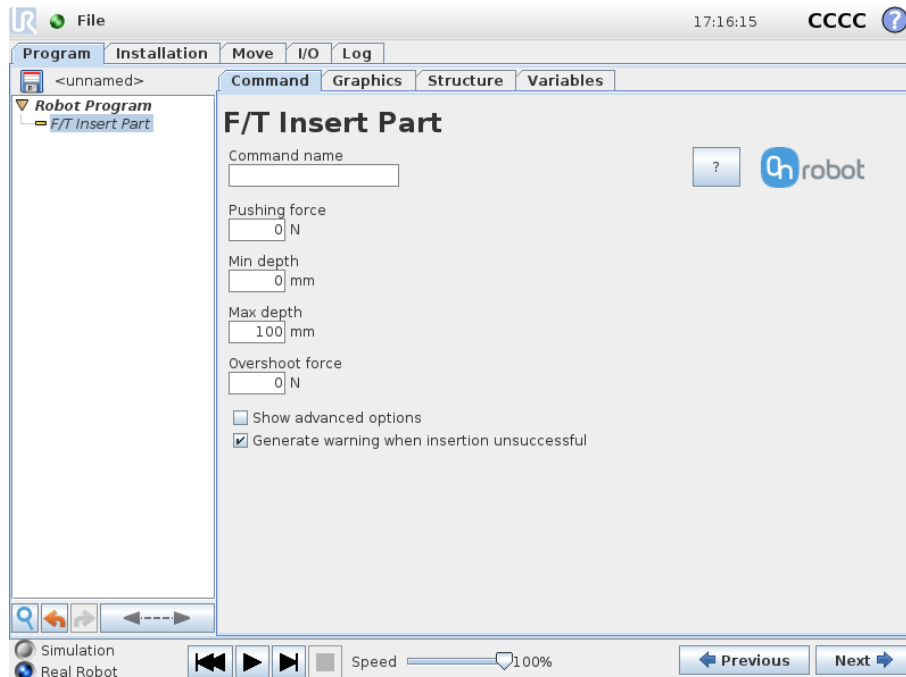
Pushing force: The force target used for the force control to gently push the part into the hole.

Min depth: The minimum distance required to consider the insertion successful, from the starting point along the Z axis (in Tool coordinate system).

Max depth: The maximum distance the insertion can reach, from the starting point along the Z axis (in Tool coordinate system).

Overshoot force: If this parameter is set, after the **Min depth** has been reached, a 'bump', an increase in the pushing force is expected (such as closing a snap-fit joint). This parameter

is the additional force on top of **Pushing force** that the insertion allows, between the min and max depths.

Show advanced parameters checkbox: If checked then more options become available:
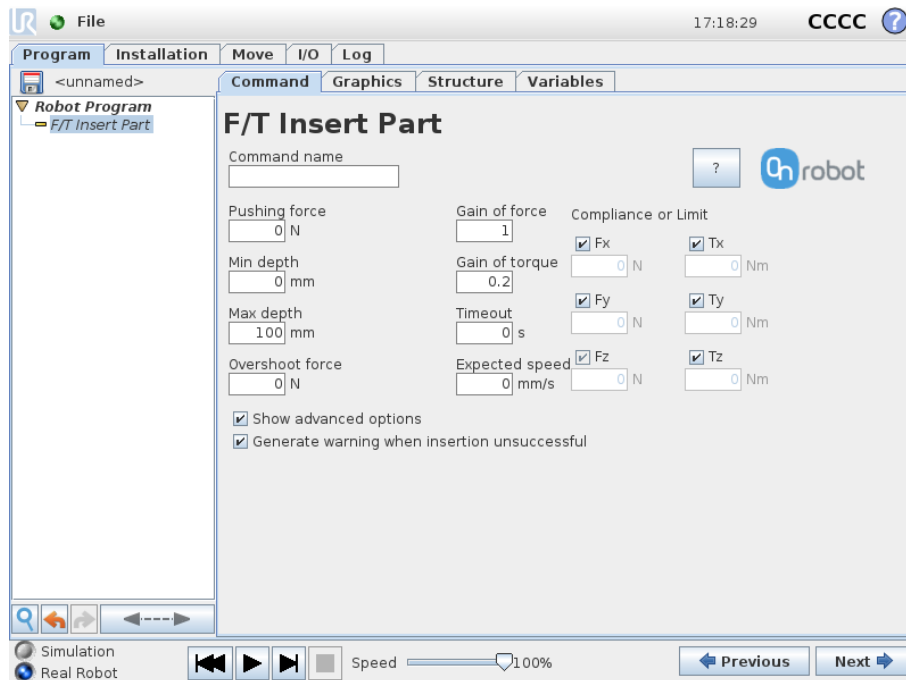
Gain of force: The proportional gain parameter of the force control for the pushing force, and the side forces on compliant axes.

Gain of torque: The proportional gain parameter of the torque control for the compliant axes.

Timeout: The maximal allowed length of time for the whole insertion function. If it is set to zero, this exit criterium is neglected.

Expected speed: The minimum speed the insertion is expected to progress with. If this parameter is set, and the insertion is progressing at a slower pace, it is interrupted and considered unsuccessful. If it is set to zero, this exit criterium is neglected.

Compliance or Limit (Fx, Fy, Tx, Ty, Tz): The axis selection that needs to be compliant. If an axis is enabled (compliant) the movement along/about that axis is force/torque controlled otherwise (non-compliant) position controlled. The enabled axis is controlled to keep the set force/torque value constant. At least one compliant axis must be selected.

Generate warning (…): If enabled then a pop-up message (blocking) appears if the insertion was not successful.

If disabled, then no pop-up message is shown but the user can handle any possible errors by the return value of the command.

For return values, see **F/T Insert Part Command Return Values**.

3.3.8 `F/T Move`

The `F/T Move` command can be used together with the `F/T Waypoint` command, to move the robot along a route, or together with the `F/T Path` to move the robot along a path and stop once the defined force/torque limits are reached (movement interrupted). In this case a warning can be generated. If the movement reaches the last waypoint, the move is successful.

> **NOTE:**
>
> To cancel any force/torque offset, execute an `F/T Zero` command at the beginning of the `F/T Move` command and make sure the tool is not in contact with any object before starting the `F/T Move`, otherwise the command may not stop at the given force/torque limit.
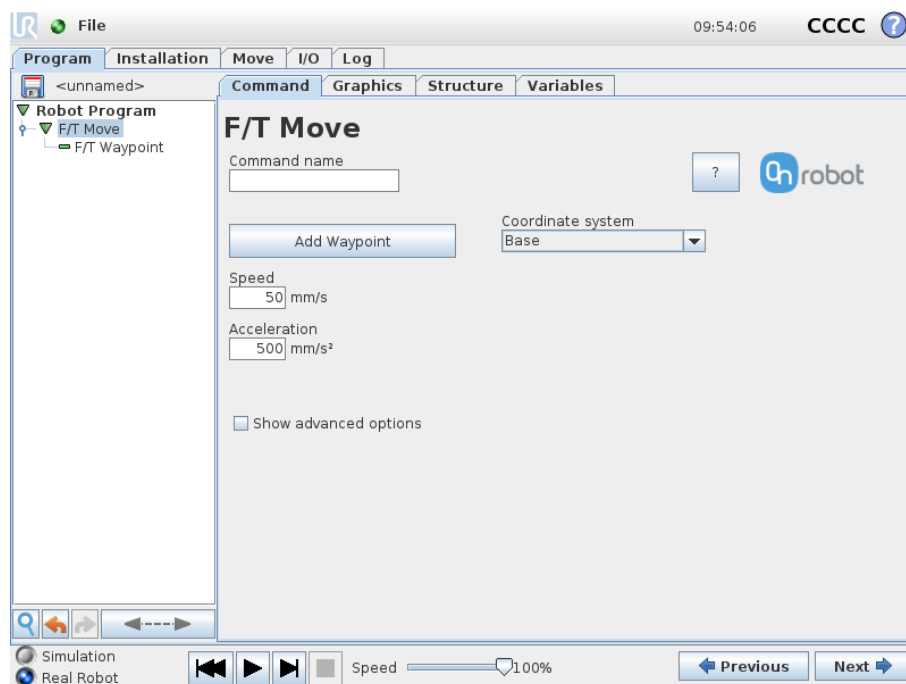
To operate the `F/T Move` command tap on the **Add Waypoint** button to add an `F/T Waypoint` as a child node. More waypoints can be added in the same way. To remove a waypoint, use the **Structure** tab **Delete** button.

Alternatively, `F/T Waypoint` or `F/T Path` could be added as a child node of the `F/T Move` command by using the **Structure** tab.

Speed: The movement speed limit while moving. The movement is carried out in a constant translational speed. If the route or path has sharp changes in the direction, or the orientation, the actual speed of the robot may be less than specified, but still constant throughout the route or Path.

Acceleration: The acceleration and deceleration parameter of the movement.

Coordinate system: The coordinate system used both for the movement and for the sensor reading. It can be set to `Base, Tool, Custom (Base) Custom (Tool)` (according to the UR's reference frames). The Custom coordinate systems are calculated from the basis coordinate system and the given **Roll**, **Pitch** and **Yaw** values. For the Custom (Base) coordinate system, it is also possible to use the **Get TCP orientation** button to specify the orientation of the coordinate system, by the orientation of the current TCP. To test the given orientation, the **Rotate tool to this orientation [HOLD]** button can be used.

Show advanced options checkbox: If checked then more options become available:
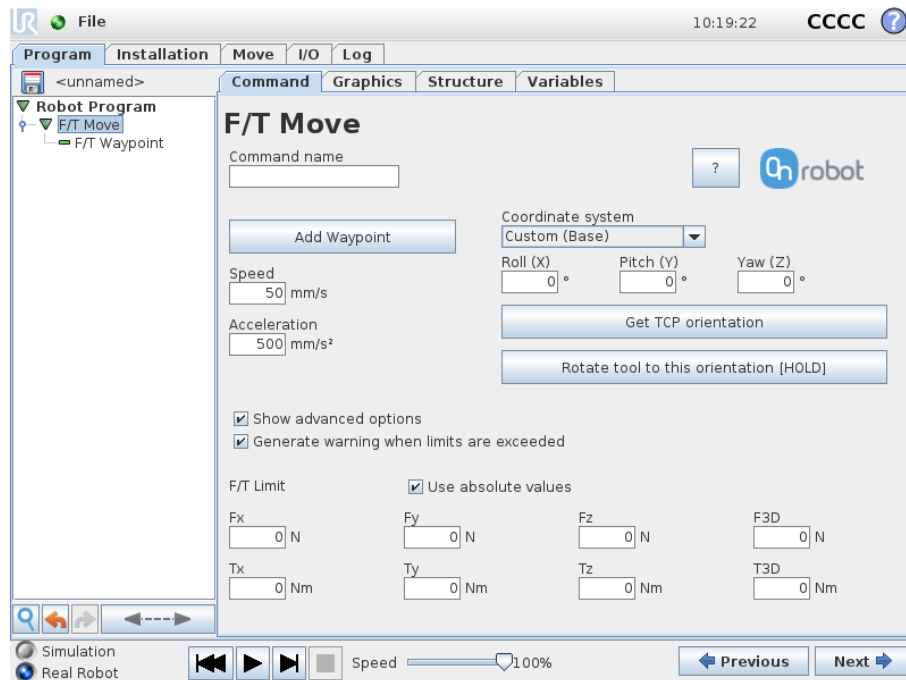
F/T Limit Fx,Fy,Fz,Tx,Ty,Tz,F3D,T3D: This is the detection limit. From the Fx, Fy, Fz, Tx, Ty, Tz, F3D, T3D available options more than one can be set. In this case, if any of the values reach the set threshold the stop is triggered. Values equal to zero are neglected.

> If the **Use absolute values** option is enabled, then it is not important whether the entered value is positive or negative (e.g.: |Fz| > = 3), otherwise the sign defines how the threshold is calculated (e.g.: Fz > = 3 or Fz <= -3)

> **Generate warning (…):** If enabled then a pop-up message (blocking) appears if the target position is not reached (the move is not successful). If the move is successful, then no warning is displayed.

> If disabled, then no pop-up message is shown but the user can handle any possible errors by the return value of the command.

For return values, see **F/T Move Command Return Values**.

3.3.9 `F/T Path`

The `F/T Path` command can be used together with the `F/T Move` or `F/T Search` command, to record and replay a Path.
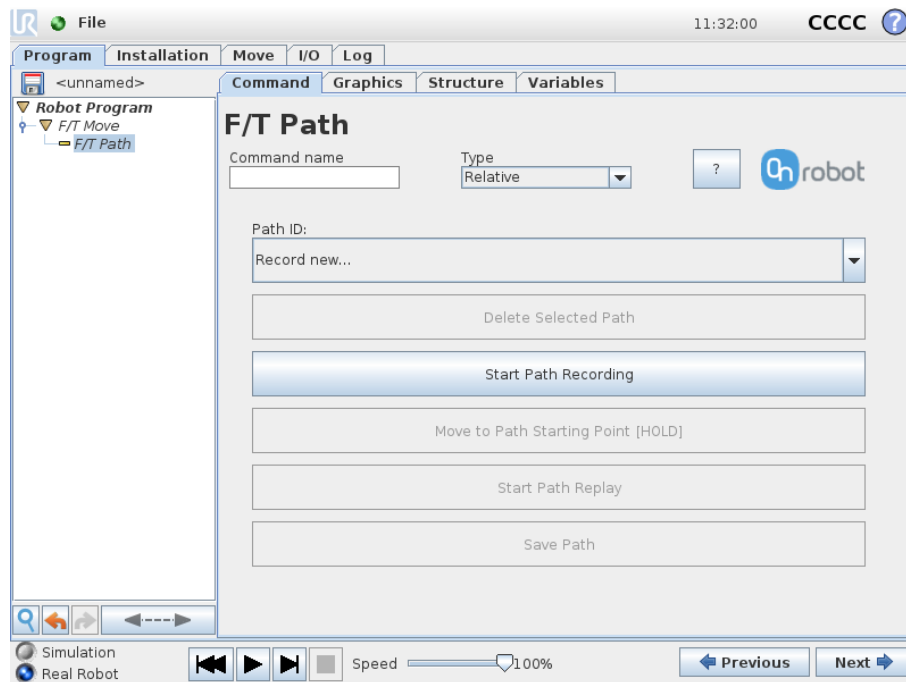
Type: If relative is selected, the Path is replayed starting from the actual position of the Tool, instead of the absolute position where it was recorded. If absolute is selected, the Tool moves to the original starting point, and replays the Path from there.

Path ID dropdown list: Lists identifiers of all the Paths saved on the Compute Box. A Path ID is assigned to a path, when the path is saved. If there is no recorded unsaved path, there is a **Record new…** item, select it to record a new path. If there is a recorded Path that is not saved, there is an **Unsaved** item in the List.

> **NOTE:**
>
> Only one unsaved Path can exist, and it will be overwritten by starting a Path recording while the **Unsaved** Path is selected.

Delete Selected Path button: Deletes the path currently selected in the **Path ID** dropdown list from the Compute Box, permanently.

> **NOTE:**
>
> Do not delete a Path that any other F/T Path commands is using.

Start Path Recording button: Starts recording a Path by automatically enabling the Hand Guide function.

Stop Path Recording button: Stops the Hand Guide function and stores the recording to the memory. It does not permanently save the Path.

Move to Path Starting Point [HOLD] button: moves the Tool to the starting position of the Path, it can only be used if the path is not relative.

Start Path Replay button: Replays the Path, even if it is not saved, only stored in the memory.

Stop Path Replay button: Stops replaying the Path.

Save Path button: Saves the Unsaved Path to the Compute Box.

NOTE:

Rotational movements related to translational movements in Path recording are limited to 2.8 degrees/mm or less, since a larger ratio would cause the Robot to replay the path at a very low translational speed. Rotational movement without translational movement therefore cannot be recorded as Path.

NOTE:

The maximum error of the replayed Path compared to the original recorded movement can be up to 1 mm.

This command has no return value.

3.3.10 `F/T Search`

The `F/T Search` command is used together with the `F/T Waypoint` command, to move the robot along a route, or together with the `F/T Path` to move the robot along a Path and stop once the defined force/torque limits are reached (object found). If the movement reaches the last waypoint or the last point of the path, the search is not successful (the object is not found) and a warning is generated.

> **NOTE:**
>
> To cancel any force/torque offset, execute an `F/T Zero` command at the beginning of the `F/T Search` command and make sure the tool is not in contact with any object before starting the `F/T Search`, otherwise the command may not stop at the given force/torque limit.
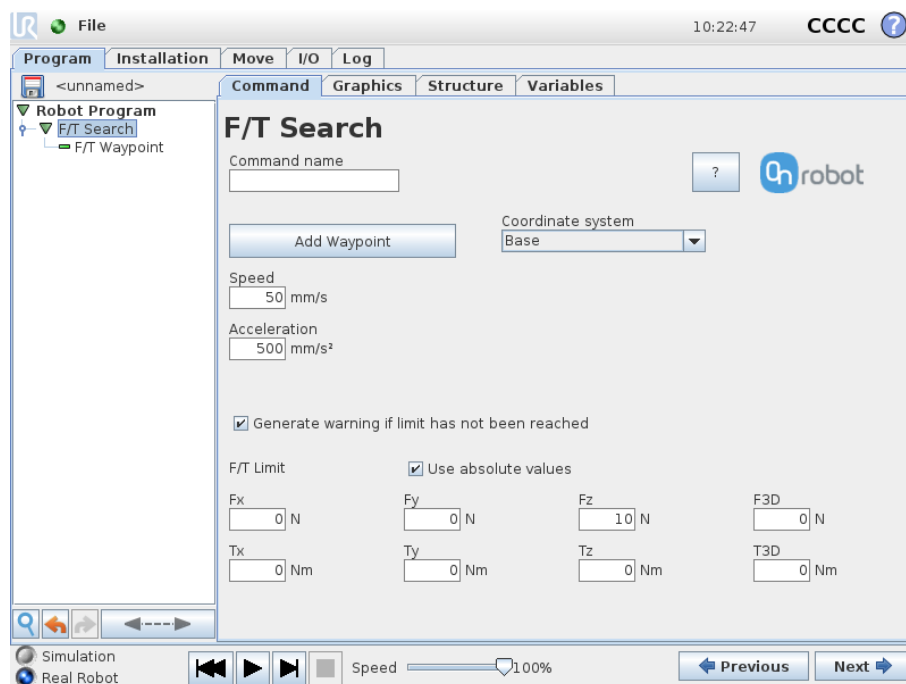
To operate the `F/T Search` command tap on the **Add Waypoint** button to add an `F/T Waypoint` as a child node. More waypoints can be added in the same way. To remove a waypoint, use the **Structure** tab **Delete** button.

Alternatively, `F/T Waypoint` or `F/T Path` could be added as a child node of the `F/T Search` command by using the **Structure** tab.

Speed: The movement speed while searching for collision. The movement is carried out in a constant translational speed. If the route or path has sharp changes in the direction, or the orientation, the actual speed of the robot may be less than specified, but still constant throughout the route or Path.

NOTE:

The slower the speed during the searching phase is the better to work with hard contacts (such as metal surfaces) to avoid overshoots due to the robot's and the tool's momentum.

Acceleration.: The acceleration and deceleration parameter of the movement.

F/T Limit Fx,Fy,Fz,Tx,Ty,Tz,F3D,T3D: This is the detection limit. From the Fx, Fy, Fz, Tx, Ty, Tz, F3D, T3D available options more than one can be set. In this case, if any of the values reach the set threshold the stop is triggered. Values equal to zero are neglected.

If the **Use absolute values** option is enabled, then it is not important whether the entered value is positive or negative (e.g.: |Fz| > = 3), otherwise the sign defines how the threshold is calculated (e.g.: Fz > = 3 or Fz <= -3)

Coordinate system: The coordinate system used both for the movement and for the sensor reading. It can be set to `Base,` `Tool,` `Custom (Base)` `Custom (Tool)` (according to the UR's reference frames). The Custom coordinate systems are calculated from the basis coordinate system and the given **Roll**, **Pitch** and **Yaw** values. For the Custom (Base) coordinate system, it is also possible to use the **Get TCP orientation** button to specify the orientation of the coordinate system, by the orientation of the current TCP. To test the given orientation, the **Rotate tool to this orientation [HOLD]** button can be used.

Generate warning (…): If enabled then a pop-up message (blocking) appears once the target position is reached or was already in collision (so the search is not successful. If the search is successful, then no warning is displayed.

If disabled, then no pop-up message is shown but the user can handle any possible errors by the return value of the command.

For return values, see **F/T Search Command Return Values.**

3.3.11 `F/T Waypoint`

The `F/T Waypoint` command is used together with the `F/T Move` or `F/T Search` command, to move the robot along a route. There are three types of waypoints (Fixed, Relative, and Variable), that can be used in any combination.

NOTE:

Do not use consecutive `F/T Waypoints` that only contain rotations in the same `F/T Move` command. Use more than one `F/T Move` commands to achieve rotations without translational movements.

Waypoint type: The type of the waypoint. It can be set to `Fixed, Relative or Variable`.
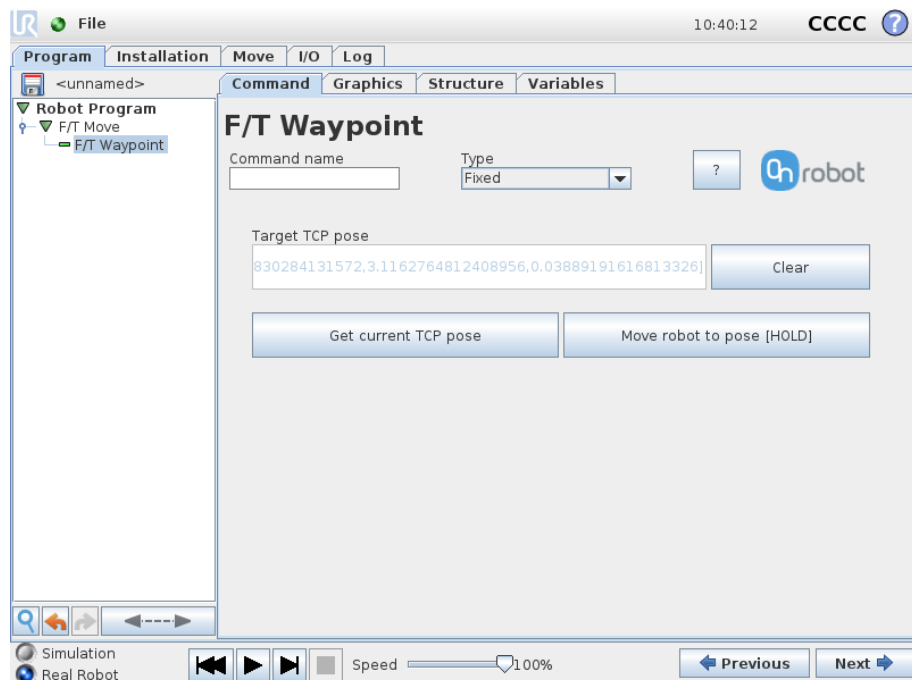
Target TCP pose: The position represented by the waypoint in the robot route. It is a read-only field and can be filled by using the **Get current TCP pose** button.

Clear button: deletes the contents of the **Target TCP pose** field.

Get current TCP pose button: inserts the current TCP coordinates into the **Target TCP pose** field.

Move robot to pose [HOLD] button: moves the robot to the pose set in the **Target TCP pose** field, if the button is pressed. Once it is released the robot stops.
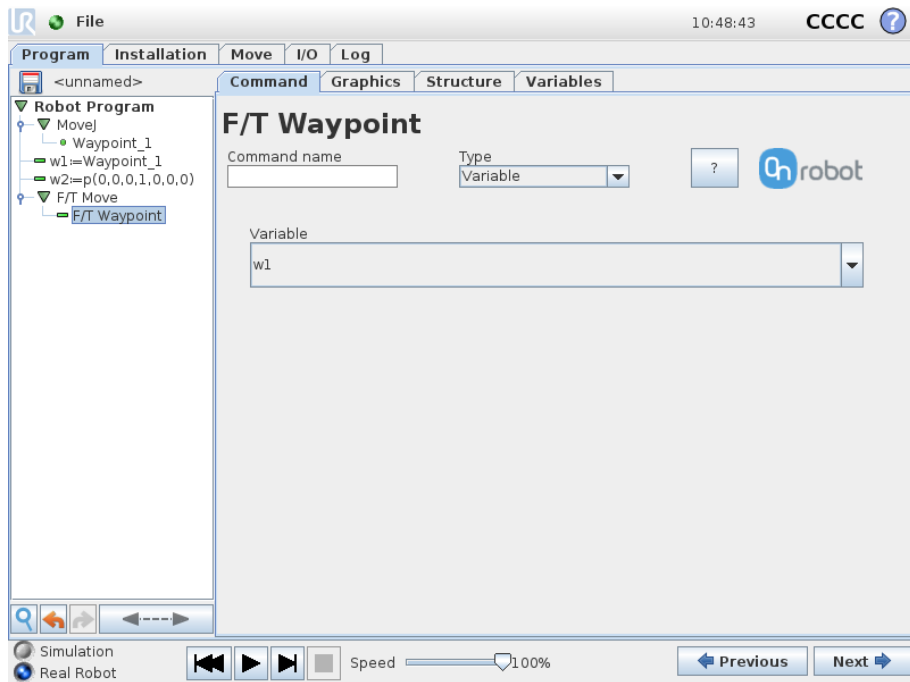
Variable: The position represented by the waypoint in the robot route. A variable can define the target pose. The variable needs to created first.
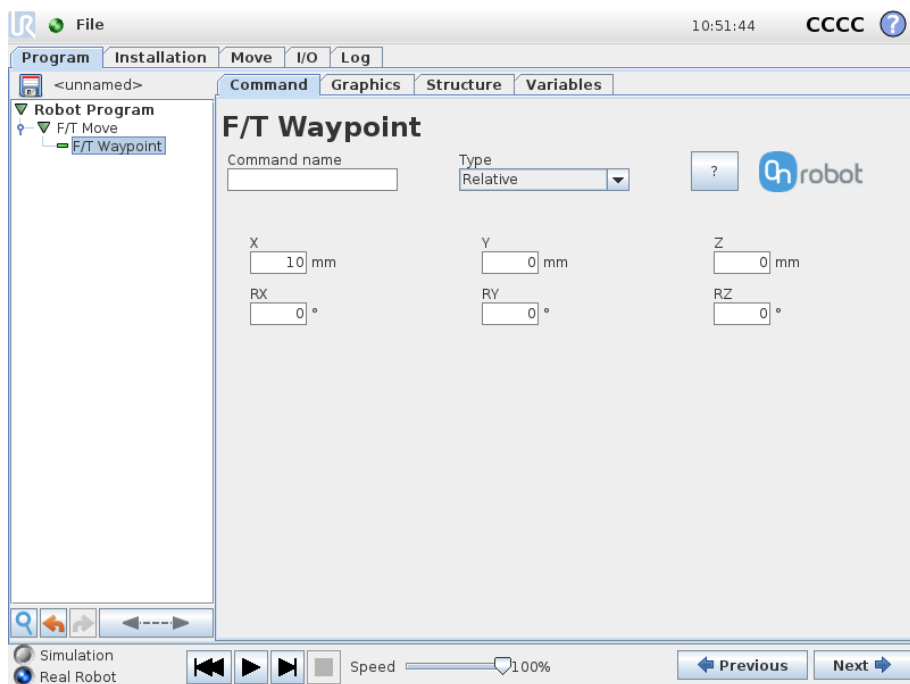
Relative X, Y, Z, RX, RY, RZ: the distances and rotations this waypoint represents, compared to the previous robot position.

This command has no return value.

3.3.12 `F/T Zero`

The `F/T Zero` command can be used to zero the RG2-FT finger sensor force/torque values.
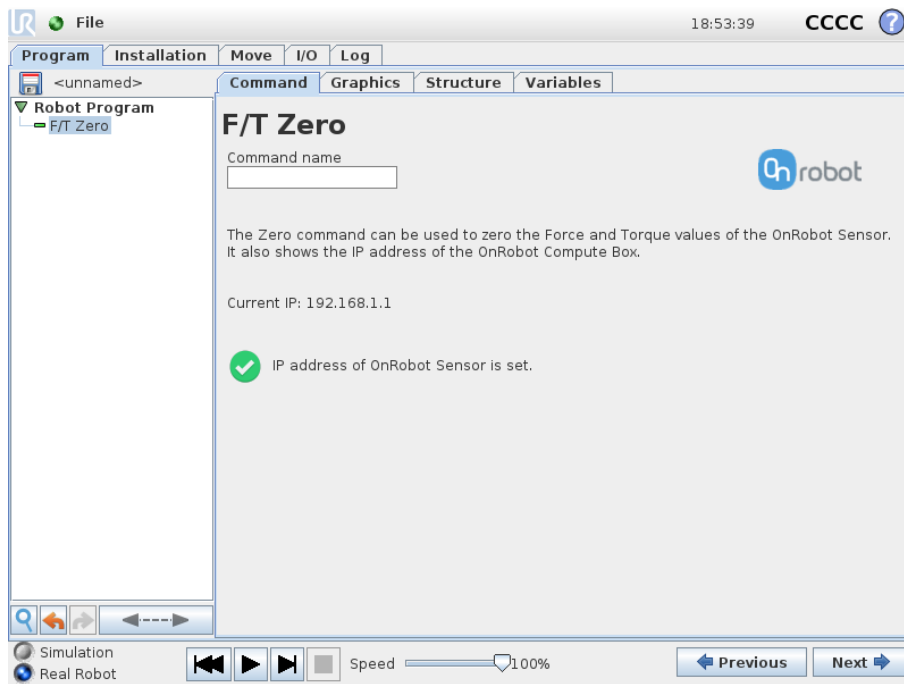
This command has no return value.

3.3.13 `F/T Set Load`

The `F/T Set Load` command can be used to set a new payload and to change the TCP settings within one command.

Either the TCP or the Payload needs to be checked to be set to enable the command to be executed.
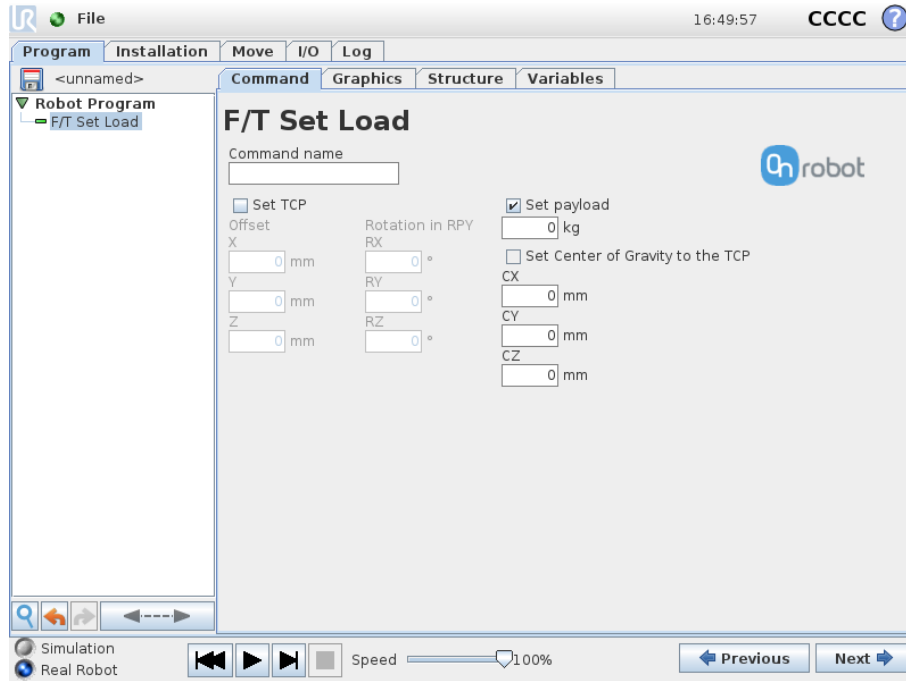
Set TCP offset checkbox: if checked, the installation TCP settings will be overridden with the given values.

> **Offset X, Y, Z**: the translation values of the TCP relative to the tool flange (or the fingertip center).

> **Rotation in RPY RX, RY, RZ**: the rotational values of the TCP relative to the tool flange (or the fingertip center).

Set payload checkbox: if checked, the installation Payload and Center of Gravity settings will be overridden with the given values. The payload must be the overall weight, including the gripper.

> **CX, CY, CZ:** the coordinates of the Center of Gravity in relation to the tool flange

> **Set Center of Gravity to the TCP** checkbox: if checked, CX,CY,CZ values are given by the set TCP offset.

This command has no return value.

3.4 Application Examples

3.4.1 Collision Detection

Collision detection can be implemented by the following commands:

1. `F/T Search`: It can be used for Presence detection. It searches for an object and stops once it is found. If the object could not be found it gives a warning message. If the position of an object is varying it can also be used to easily determine its exact location.

2. `F/T Move`: It can be used for Force/torque limited movements. It is similar to the UR's Move command but with built-in force/torque limitation and supports relative offset type parameters (e.g.: move 1 cm or 1 inch along the Z axis).

3. `F/T Guard`: It can be used in combination with any UR command to limit the exerted force/torque. It monitors the set limits in parallel to your code and once the set limits are reached it stops the robot.

   The `programs/OnRobot_UR_Programs` folder contains a collision detection example UR program, called *OnRobot_Collision_Detection_Example.urp*.

3.4.2 Center Point Detection

With the help of gentle contacts, the robot can be positioned to the geometrical center point of a hole. It also works with shiny metal objects that are usually impossible with camera-based solutions.

The `programs/OnRobot_UR_Programs` folder contains a collision detection example UR program, called *OnRobot_Centerpoint_Detection_Example.urp*.

3.4.3 Polishing and Sanding

For any polishing or sanding task, it is very important to keep the predefined force value constant. This task can be achieved with our force/torque control functions, that requires the following two commands to be used:

1. `F/T Control`: This command is similar to the UR's built-in Force command but uses the OnRobot more precise force/torque sensor as an input to achieve an excellent result even with low forces. The force/torque control tries to keep the defined force/torque constant on the axes that are set to be compliant. The non-compliant axes are position controlled (only with the `F/T Move` command).

2. `F/T Move`: It can be used to position control (move) the robot along/about the non-compliant axis in the `F/T Control`.

The `programs/OnRobot_UR_Programs` folder contains a collision detection example UR program, called *OnRobot_Plastic_Partingline_Removal_Example.urp*.

3.4.4 Palletizing

Palletizing objects that need to be handled with care can be a challenging task. Placing flexible cardboard boxes next to each other requires more than just a simple positioning in a fixed pattern. Using the UR's built-in palletizing command in combination with our `F/T Search` command anyone can easily solve these challenging tasks.

First set up the UR's built-in `Pallet` command, to achieve the required pattern. Make sure that the positions are a little bit further than what is going to be the final position. That allows the `F/T Search` command to find the neighboring item by a gentle touch to adopt to any positioning errors.
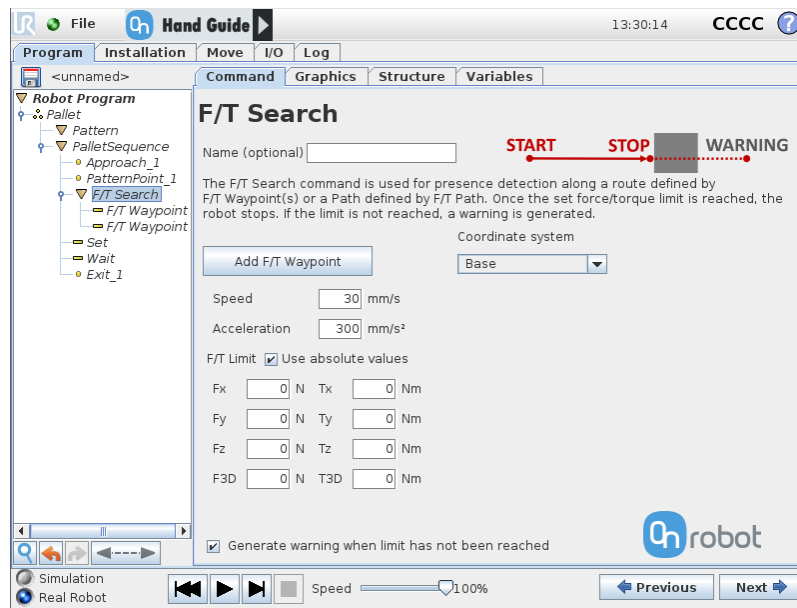
If necessary, more than one F/T Search can be used to align the item horizontally and vertically.

> Make sure to use only the relative offset type of input parameters of the F/T Search command to be always relative to the pattern.

For more information see **F/T Search command**.

The `programs/OnRobot_UR_Programs` folder contains a collision detection example UR program, called *OnRobot_Palletizing_Example.urp*.

3.4.5 Pin Insertion

Inserting pins or pegs into tight holes cannot be achieved with traditional position-based solutions. Even with cameras one cannot have a robust solution.

With the help of the precise OnRobot F/T sensor and the `F/T Insert Pin` command anyone can easily and robustly solve tasks that requires precision fitting.

The `programs/OnRobot_UR_Programs` folder contains a collision detection example UR program, called *OnRobot_Pin_Insertion_Example.urp*.

3.4.6 Box Insertion

Inserting a rectangular object into a rectangular hole is a common task, such as inserting a car radio assembly into the radio bracket or inserting a battery into a phone.

With the help of the `F/T Insert Box` command anyone can easily solve these tasks.

The `programs/OnRobot_UR_Programs` folder contains a collision detection example UR program, called *OnRobot_Box_Insertion_Example.urp*.

3.4.7 Fix and Rotate

With the help of the precise OnRobot F/T sensor and the `F/T Fix and Rotate` command anyone can easily and robustly solve tasks that requires using any Bayonet type of mounting.

4   Glossary of Terms

Term	Description
Compute Box	A unit provided by OnRobot along with the sensor. It performs the calculations needed to use the commands and applications implemented by OnRobot. It needs to be connected to the sensor and the robot controller.
OnRobot Data Visualization	Data visualization software created by OnRobot, to visualize the data provided by the sensor. Can be installed on Windows operating system.

5   List of Acronyms

Acronym	Expansion
DHCP	Dynamic Host Configuration Protocol
DIP	dual in-line package
F/T	Force/Torque
ID	Identifier
IP	Internet Protocol
IT	Information technology
MAC	media access control
PC	Personal Computer
RPY	Roll-Pitch-Yaw
SP	Starting Position
SW	software
TCP	Tool Center Point
UR	Universal Robots
URCap	Universal Robots Capabilities
USB	Universal Serial Bus
UTP	unshielded twisted pair

6 Appendix

6.1 Changing the IP of the Compute Box

To change the IP address of the sensor, connect your laptop or an external PC to the OnRobot Compute Box.

1. Make sure that the device is not powered. Connect the device and the computer with the provided Ethernet cable.

2. If your device is in the factory default settings, proceed to step 3. Otherwise, make sure to switch the DIP switch 3 to the ON position (up) and the DIP switch 4 to the OFF position (down).

3. Power the device from the provided power supply and wait 30 seconds for the device to boot up.

4. Open a web browser (Internet Explorer is recommended) and navigate to **http://192.168.1.1**. The welcome screen is displayed.

5. Click on **Configuration** in the top-side menu. The following screen is displayed:

6. Select the **Static IP** option from the **Network mode** drop down menu.

7. Edit the IP Address.

8. Set DIP switch 3 to off position.

9. Click the **Save** button

10. Open a web browser (Internet Explorer is recommended) and navigate to the IP Address set in step 7.

6.2 Updating the software on the Compute Box

Refer to the Compute Box Description document.

6.3 Software Uninstallation

1. To uninstall (remove) the previously copied OnRobot UR program files, choose from the following options:

   a. Remove the files and folder, using the **Delete** option of the Teach Pendant during file operation (for example, Load Program, Save Program)

   b. Copy the `uninstall.sh` file from the USB drive to a new USB drive, rename it to `urmagic_OnRobot_uninstall.sh` and plug it into the Teach Pendant. The file creates a backup copy to the USB drive, and then it permanently deletes the `OnRobot_UR_Programs` folder from the UR.

2. Uninstall the URCap plugin.

   a. Go to the Welcome screen of the PolyScope.

   b. Click **Setup Robot**.

   c. Click on **URCaps Setup** and locate the `FT – OnRobot` in the list of active URCaps.

   d. Click on the **-** sign at the bottom to uninstall it.

   e. Restart the robot.

6.4 Return Values

Those OnRobot commands that has return values updates the `of_return` variable once the command exit. This global variable can be used with the UR's built in `If` conditional expressions (for example: `if of_return == 1` then do something).

6.4.1 `F/T Center` Command Return Values

0	Arrived successfully to the center point.
1	The first boundary search was unsuccessful. Movement reached the distance limit.
2	The second boundary search was unsuccessful. Movement reached the distance limit.
3	Could not reach the center point. The tool collided during the movement.
4	The search has not been started because of the conditions.
5	The second search has not been started because of the conditions.
99	Do not define more than one directional parameter.

6.4.2 `F/T Fix and Rotate` Command Return Values

0	The Fix and Rotate finished with no error.
11	The Orientation center point search of Ry was unsuccessful.
12	The Orientation center point search of Ry was unsuccessful.
21	The rotation was unsuccessful, collision occurred.
22	The rotation ended without contact.
99	Parameter error.

6.4.3 `F/T Insert Box` Command Return Values

0	The Box Insertion finished with no error.
1	The first direction search was unsuccessful. Movement reached the distance limit.
2	The second direction search was unsuccessful. Movement reached the distance limit.
3	The tilt back movement was unsuccessful. Collision occurred.
4	The tilt movement was unsuccessful. Collision occurred.
5	The box stuck during the insertion state while center pointing of the X axis! Please check the position and orientation.

6 The box stuck during the insertion state while center pointing of the Y axis! Please check the position and orientation.

7 The box stuck during the insertion state while center pointing of the Z axis! Please check the position and orientation.

8 The box cannot be inserted to position, too many collisions occurred. Please check the position and orientation.

6.4.4 F/T `Insert Part` Command Return Values

0 The Insert Part command reached the maximal distance.

1 The Insert Part command exited at a bump after the minimal insertion depth.

2 The Insert Part command got stuck after the minimal insertion depth. Insertion is slower than required.

3 The Insert Part command got stuck before the minimal insertion depth. Insertion is slower than required.

4 The Insert Part command exited with timeout after the minimal insertion depth.

5 The Insert Part command exited with timeout before the minimal insertion depth.

6 The Insert Part command exited due to too high side-forces/-torques at the non-compliant axes after the minimal insertion depth.

7 The Insert Part command exited due to too high side-forces/-torques at the non-compliant axes before the minimal insertion depth.

8 The Insert Part command has a parameter error.

6.4.5 F/T `Move` Command Return Values

0 The move ended without detecting a force or torque greater than the set limit.

1 The move ended, because a force or torque greater than the set limit was detected.

3 The move cannot start due to a force or torque exceeding the set limit.

11 The move cannot start, because there is no recorded Path on the Compute Box with the selected.

12 The move cannot start, because there are no recorded points in this Path.

13 The move cannot start, because the Path file found at this Path ID is empty.

14 The move cannot start, because the Path file is corrupted.

6.4.6 F/T `Search` Command Return Values

0   The search ended successfully, because a force or torque greater than the set limit was detected.

1   The search ended without detecting a force or torque greater than the set limit.

3   The search cannot start due to a force or torque exceeding the set limit.

11   The search cannot start, because there is no recorded Path on the Compute Box with the selected.

12   The search cannot start, because there are no recorded points in this Path.

13   The search cannot start, because the Path file found at this Path ID is empty.

14   The search cannot start, because the Path file is corrupted.

6.4.7 F/T Stacking Command Return Values

Stack return values:

0   One iteration of the stacking is complete.

1   The iteration counter is over the maximum: the stack is full.

2   Stacking is unsuccessful. Next item not found.

3   Stacking cannot start due to a force or torque exceeding the set limit.

4   The movement to the next element was unsuccessful, a collision occurred.

5   The movement to the starting point was unsuccessful, a collision occurred.

Destack return values:

0   One iteration of the destacking is complete.

1   The iteration counter is over the maximum: the stack is empty.

2   Destacking is unsuccessful. Next item not found.

3   Destacking cannot start due to a force or torque exceeding the set limit.

4   The movement to the next element was unsuccessful, a collision occurred.

5   The movement to the starting point was unsuccessful, a collision occurred.

.

6.5 Troubleshooting

6.5.1 URCap Plugin Setup error

There are three possible reason why the error ❌ icon could be displayed.

1. If the **Discovered devices** drop-down menu shows "NO DEVICES FOUND!" error message for troubleshooting, see **"No devices found"**.

2. If the OnRobot device(s) found successfully but the **UR Robot IP** shows "N/A" for troubleshooting, see **UR Robot IP is "N/A"**.

3. If both the OnRobot device(s) found successfully and the UR Robot IP shows a valid IP address, then for troubleshooting see **Device found, and UR has IP**.

6.5.1.1 "No devices found"

If the **Discovered devices** drop-down menu shows "NO DEVICES FOUND!" error check the connections with the Compute Box and the sensor, then try to restart the Compute Box.

After 60 seconds (when both status LEDs of the Compute Box turned to green) try to manually repeat the discovery by tapping on the refresh ↻ icon.

6.5.1.2 UR Robot IP is "N/A"

This error can happen, when the Network Configuration of the UR Robot has not been set.

To resolve the problem, check the Network Configuration of the UR Robot, by doing the following:

1. Press the Setup Robot button.
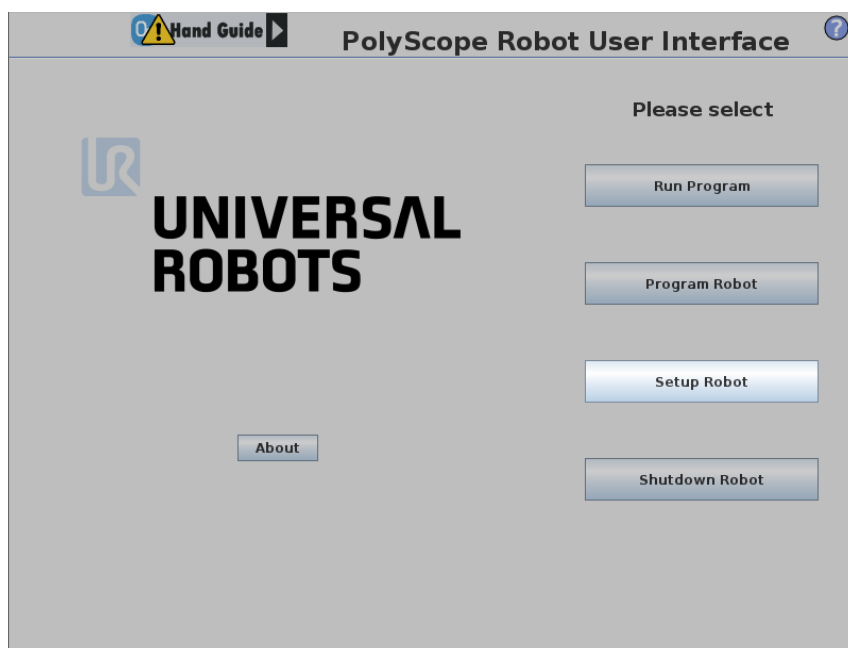
2. Press the Setup Network button.

3. If the network of the UR is disabled:

4. If the OnRobot device is connected to the UR robot directly, select DHCP, and press the Apply button. The OnRobot service assigns an IP.

5. If the OnRobot device is not directly connected to the UR robot, check if the OnRobot device is connected to the same network (router, switch, and so on) as the UR Robot, or consult the Network Supervisor.

6. If DHCP or Static Address is selected, and the problem remains consult your Network Supervisor.
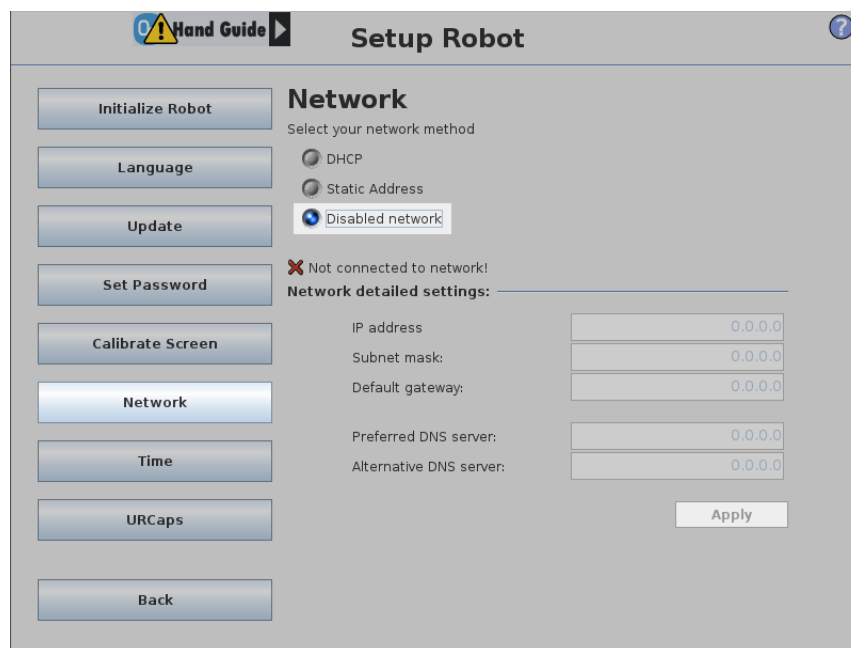
In case of a DHCP, after the proper IP address is assigned to the UR robot switch to Static address mode (the IP address of the UR robot should remain the same) and press the **Apply** button. The IP address is now fixed and does not change later.

Finally start over with the **URCap Plugin Setup**.

6.5.1.3   Device found, and UR has IP

This error can happen when the robot and device are not in the same subnet.

To resolve the problem, follow this procedure:

1. If the OnRobot device is not directly connected to the UR Robot, check if the DIP Switch 3 is in off state on the Compute Box, as shown on the following figure:

2. If the DIP Switch is in on state, set it to off, then restart the OnRobot device (by disconnecting the power) and repeat the steps in **URCap Plugin Setup** section.

If the problem is still present, follow this procedure:

1. Open the Setup Network page of the UR robot explained in **UR Robot IP is "N/A"**.

2. Modify the Subnet mask to "255.0.0.0".

3. Press the Apply button.

Finally start over with the **URCap Plugin Setup**.

6.5.2 Too Close to Singularity

During hand guiding, if the tool is guided too close to the cylindrical volume directly above or below the robot base, a warning message is displayed.
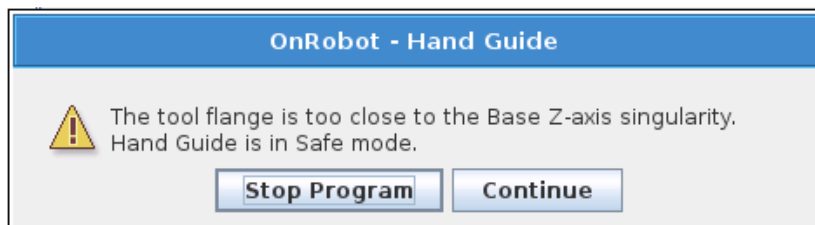
Pressing the **Stop Program** button will disable the Hand Guide function. Pressing the **Continue** button will switch to Safe Mode, which prevents the tool flange from moving into the cylindrical volume directly above or below the robot base by the Hand Guide function. Moving 10 mm away from that volume turns Safe Mode off, enabling movement in all directions again.

> **NOTE:**
>
> For safety and accuracy, the Hand Guide mode keeps the tool flange at a greater distance from the cylindrical volume than the physical possibility of the UR robot. Moving the tool flange closer is possible, by using the PolyScope Move tab, or move commands.

6.5.3 Warning sign on Hand Guide Bar
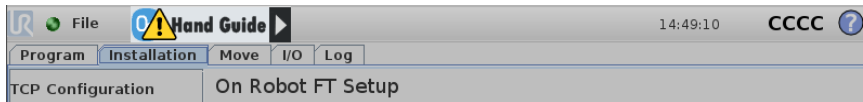
If the OnRobot device is not running properly, a warning sign appears. Repeat the steps of the **URCap Plugin Setup**.

6.5.4 "socket_read_binary_integer: timeout"

If any command is running for more than 2 seconds, a **socket_read_binary_integer: timeout** entry appears in the **Log**.
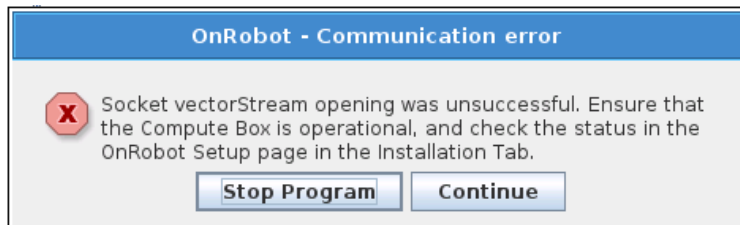
This has no impact on the program execution by the robot.

6.5.5 "Socket vectorStream opening was unsuccessful."

If the Robot Controller cannot connect to the Compute Box, the "Socket vectorStream opening was unsuccessful." error message is displayed.

In this case, ensure that the Compute Box is connected to the Robot Controller, and powered on.

6.5.6 Path Replay is Slower than Expected

When using the `F/T Path` command, it is possible that the recorded Path is not smooth due to the limits of human dexterity. In these cases, the robot can only replay the Path at a very slow speed. To avoid this problem, try recording the Path again, with confident smooth movements with as little variations in translational and rotational speeds, as possible. Also, try to avoid recording Paths that contain rotations without translational elements.

6.5.7 "Error number -2" on Path Saving

If an empty Path is recorded, when trying to save the path, the "Error number: -2" error message is displayed.
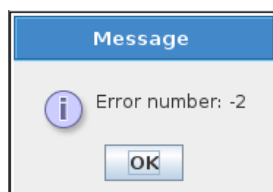
In this case, ensure that the robot is moved between starting and stopping the path recording function.

6.5.8   "Error number -3" on Path Saving

If a Path cannot be saved due to not enough storage space on the Compute Box, the "Error number -3" error message is displayed.
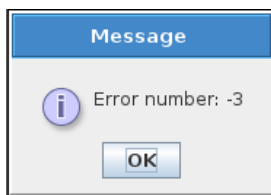
In this case, delete previously recorded paths that are not used anymore.

6.5.9 "Unknown sensor type"

If the Compute Box cannot recognize the connected OnRobot device this error message is shown.
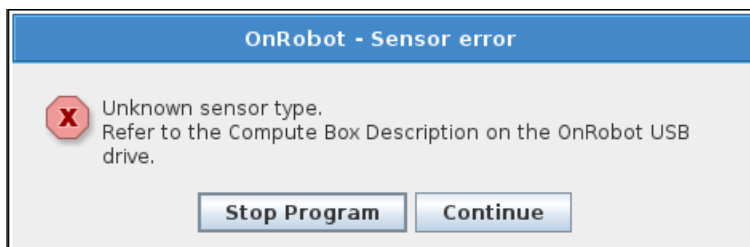
In this case, ensure that the connection between the Compute Box and OnRobot device (sensor) is good and the right device is connected.

6.5.10 "Sensor is not responding."

If the Compute Box has recognized the connected OnRobot device and later the device connection has lost, this error message is shown.
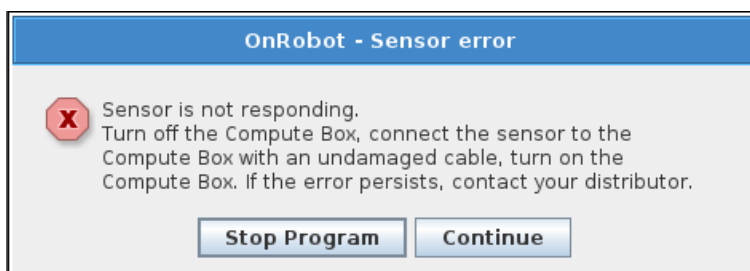
Check that the connection between the Compute Box and OnRobot device (sensor) is good and the right device is connected.

6.6 Declarations and Certificates

CE/EU Declaration of Incorporation (original)

According to the European Machinery Directive 2006/42/EC annex II 1.B.

The manufacturer:

OnRobot A/S
Teglværskvej 47H
DK-5220, Odense SØ
Denmark
+45 53 53 57 37

declares that this product:

Type:	Industrial 6-axis Force/Torque sensor
Model:	HEX-E and HEX-H
Serial number from:	HEXEB001 and HEXHB001

is partly completed machinery according to 2006/42/EC. The product must not be put into service before the complete machine is in full compliance with all essential requirements of 2006/42/EC. A comprehensive risk assessment must be carried out for each application as part of ensuring that all essential requirements are fulfilled. All essential requirements must be assessed. Instructions and guidance provided in the HEX user manual must be followed.

Technical documentation compiled according to 2006/42/EC annex VII part B is available to national authorities upon request.

The product is in conformity with, and CE marked according to, the following directives:

2014/30/EU	— Electromagnetic Compatibility Directive (EMC)
2011/65/EU	— Restriction of the use of certain hazardous substances (RoHS)
2014/35/EU	— Low Voltage Directive (LVD)

Nicolae Gheorghe Tuns
RD Director
Odense, October 17st, 2018

Declaration of EMC test result

T-Network client

OnRobot Hungary Kft.
Aradi u. 16.
1043 Budapest
Hungary

Product identification

OnRobot HEX Force/Torque Sensor
S/N: HEXEX005 with CB1807B018

Manufacturer

OnRobot A/S

Technical report

T-Network Project EMC-180926/1, OnRobot HEX Force/Torque Sensor and Compute Box EMC Test Report,
dated 17 July 2018

Standards/Normative documents

EN 61000-6-2:2005
EN 61000-6-4:2007+A1:2011

T-Network has evaluated the products in various measurements, and the results verify the product's
EMC compliance.

T-Network Kft.
Ungvár u. 64-66.
1142 Budapest
Hungary

Budapest, 05 October 2018

T-Network Kft.
EMC Laboratory
Ungvar u. 64-66. 1142 Budapest, Hungary
Registration num.: 12005222-2-42

Tel. +36 1 460 9000
Fax +36 1 460 9001
E-mail: tnetwork@tnetwork.hu
Web: http://www.tnetwork.hu

Sándor Tatár
Laboratory Leader
T-Network Kft.

SGS

Report No.: SHES180600601401
Date of issue: 2018-09-25

TEST REPORT

Product name...................:	6-axis Force/Torque Sensor
Product model. ....:	HEX-E v2
Product description.............:	Sensor
Electrical Rating ....:	-
Applicant.............................:	OptoForce Ltd.
Address ....:	Aradi utca 16 1043 Budapest Hungary
Manufacturer ....:	OptoForce Ltd.
Address ....:	Aradi utca 16 1043 Budapest Hungary
Testing Laboratory ....:	SGS-CSTC Standards Technical Services (Shanghai) Co., Ltd.
Address ....:	No. 588 West Jindu Rd, Xinqiao Town, Songjiang District, Shanghai, CHINA
Number of Samples received:	1
Date of samples reception ..:	2018-08-31
Date Test Conducted...........:	2018-09-08 to 2018-09-09
Test Requested...................:	IP67 (as client's requirement)
Test Method (standards) ....:	IEC 60529 Clause 13.6 & Clause 14.2.7
Test result..........................:	**Pass**
CONCLUSION	The submitted sample complies with the clauses examined.

Prepared and checked by:

Lewis Hua

Lewis Hua

Reviewed by:

Lucy

Lucy Wang

6.7 Editions

Edition	Comment
Edition 2	Document restructured.
	Glossary of Terms added.
	List of Acronyms added.
	Appendix added.
	Target audience added.
	Intended use added.
	Copyright, Trademark, contact information, original language information added.
	Behavior of F/T Move, F/T Search, F/T Insert Pin and F/T Control commands changed.
	F/T Waypoint command added.
	F/T Move (Ctrl) command removed.
	Application example references added to example UR programs.
Edition 3	Hand Guide Toolbar coordinate system corrected to Tool.
	Note added about TCP Orientation limitation.
	Hand guiding axis activation limit removed.
	Clarification about waypoint type usage added.
Edition 4	TCP Orientation limitation removed.
Edition 5	F/T Search and F/T Move command return values updated.
	Path Recording section removed.
	F/T Path Command section added.
	F/T Insert Connector section removed.
	F/T Insert Connector Return Values section removed.
	F/T Move Command and F/T Search Command sections updated with constant replay speed information, and new command screenshots.
	F/T Control Command section updated with directional force control limitation.
	Editorial changes.
Edition 6	Path replay accuracy added.
	Section "An error occurred in the running program" on Program Continue changed to "An error occurred in the running program" on Program Stop, pausing and continuing program no longer causes alarm.
	Section Effects of the TCP Position added.
	socket_read_byte_list(): timeout log item changed to socket_read_binary_integer: timeout, behavior changed.

	Section "Socket vectorStream opening was unsuccessful." added to Troubleshooting.
	Section Connector Insertion removed.
	Section Path Replay is Slower than Expected added.
	Limitations added for rotation only waypoints.
Edition 7	Editorial changes.
Edition 8	Path recording maximum rotation per translation limit added to section F/T Path Command.
	Section "Error number -2" on Path Saving, and "Error number -3" on Path Saving added.
	Editorial changes.
Edition 9	Important Safety Notice added.
	Warning Symbols added.
	Screenshots updated.
	Note added to warn against rotating the sensor cable in section Cable Connections.
Edition 10	Hex v2 information added.
Edition 11	Sections F/T Stack Command and F/T Destack Command combined to section F/T Stacking Command.
	Sections F/T Stack Command Return Values and F/T Destacking Command Return Values combined to section F/T Stacking Command Return Values.
	Screenshots updated.
Edition 12	USB cable information updated
	URCap Plugin setup updated
	Hand Guide icons updated
	Trouble shooting section is updated
	Error messages are updated