



UNIVERSAL ROBOTS

Software Handbook

PolyScope X



The information contained herein is the property of Universal Robots A/S and shall not be reproduced in whole or in part without prior written approval of Universal Robots A/S. The information herein is subject to change without notice and should not be construed as a commitment by Universal Robots A/S. This document is periodically reviewed and revised.

Universal Robots A/S assumes no responsibility for any errors or omissions in this document.

Copyright © 2009-2026 by Universal Robots A/S.

The Universal Robots logo is a registered trademark of Universal Robots A/S.



Contents

1. Introduction	10
1.1. About This Document	10
1.2. How to Use This Handbook	10
1.3. Related Documentation	11
2. Safety Information	12
3. First Boot	13
3.1. Powering On the Robot	13
3.2. Inserting the Serial Number	14
3.3. Verifying the Installation	14
3.4. Starting the Robot Arm	18
3.5. Testing Mount and Payload using Freedrive	19
3.6. Powering Down the Robot	20
3.7. Robot Registration and License File	20
4. PolyScope X Components	23
4.1. Header	25
4.1.1. Functions Inside the Hamburger Icon	29
4.2. Main Navigation	31
4.3. Sidebar	33
4.3.1. Sidebar in the Application Tab	34
4.4. Footer	37
5. Network Connection	39
5.1. Network	39
6. Operating the System	41
6.1. PolyScope X User Interface	41
6.1.1. Icons	42
6.1.2. Touch Screen	47
6.1.3. System Manager	47
6.1.4. 3D Viewer	49
6.1.5. Mouse Cursor Support	50
6.1.6. Validation Indicators	51
6.2. Operational Mode	53
6.2.1. Manual Operational Mode	55
6.2.2. Automatic Operational Mode	56
6.2.3. Remote Control	57
6.3. Starting and Stopping the Robot	59
6.3.1. Operator Screen	59



6.3.2. Operator Screen Application	59
6.3.3. Freedrive	61
6.3.4. Constrained Freedrive	64
6.3.5. Backdrive	66
6.4. Programming Workflows	69
6.4.1. Create Program	69
6.4.2. Modify Program	70
6.4.3. Load Program without Safety Password	73
6.4.4. Smart Skills	74
6.4.5. Smart Skills Application	80
6.4.6. Teach Mode	85
6.4.7. Move Menu	88
6.4.8. Export Program	97
6.4.9. Shared Applications in System Manager	100
6.4.10. Logic Program	107
6.4.11. Export and Import of Modules	115
6.4.12. Export to Scripts	119
6.4.13. Move to First Position	123
6.4.14. Startup Program	125
6.4.15. Program Organization	126
6.5. Common Operating Tasks	127
6.5.1. Pick and Place	128
6.5.2. Motion Profiles	132
6.5.3. Global Variables	132
7. Programming Reference	134
7.1. Program Structure	134
7.2. Program Menu	135
7.3. Modules and Functions	137
7.3.1. Gripper Module	137
7.3.2. Movement Module	144
7.4. Command Nodes	149
7.4.1. Comment	151
7.4.2. Simple Force	152
7.5. Move Command Nodes	155
7.5.1. Circular Arc Move	155
7.5.2. Direction	165
7.5.3. Frame	171
7.5.4. Joint Move	174
7.5.5. Linear Move	180
7.5.6. Process Move	187
7.6. Logic Command Nodes	191

7.6.1. Assignment	192
7.6.2. Case	194
7.6.3. Else	195
7.6.4. Else If	196
7.6.5. Function	198
7.6.6. Halt	200
7.6.7. If	201
7.6.8. Loop	201
7.6.9. Script	203
7.6.10. Set	204
7.6.11. Switch	205
7.6.12. Wait	208
7.7. Structure Command Nodes	209
7.7.1. Call	210
7.7.2. Folder	212
7.7.3. Grid Iterator	213
7.7.4. Operator Input	214
7.7.5. Thread	217
7.7.6. Timer	218
7.7.7. Control Actions	219
7.8. Tool Command Nodes	221
7.8.1. Advanced Force	222
7.8.2. Payload	226
7.8.3. TCP	228
7.9. Value-Variable-Expression Tabs	231
7.9.1. Expression Editor	233
7.9.2. Expression Editor Signals	235
7.10. Frames	236
7.11. Grids Application	240
7.12. OptiMove	243
8. First Program	245
9. Configuration	251
9.1. Configuration Settings	251
9.1.1. General Settings	252
9.1.2. Password Management	255
9.1.3. UR Connect	258
9.1.4. Security	263
Ethernet/IP Adapter	267
PROFINET Device	270
9.1.5. Unlock Safety Settings	278



9.2. Configure a 3PE TP in PolyScopeX	279
9.3. Application Tab	279
9.4. Communication	281
9.4.1. Wired IO	282
9.4.2. Tool I/O	283
9.4.3. Analog Input	285
9.4.4. Analog Output	286
9.5. End Effectors Application	287
9.5.1. TCP Position	288
9.5.2. TCP Orientation	290
9.5.3. Payload and Center of Gravity	292
9.5.4. Payload Inertia	294
9.6. Motion Profiles	296
9.7. Application Variables	299
9.7.1. Variable Types	301
9.7.2. Delete Variables	302
9.7.3. Use Variables in Programs	303
9.7.4. Showing Variables	304
9.7.5. Saving Variables	305
9.7.6. Global Variables in Logic Program	306
9.8. Safety Application	308
9.8.1. Safety Menu Settings	309
9.8.2. Robot Limits	309
9.8.3. Joint Limits	313
9.8.4. Safety I/O	316
9.8.5. Safe Home	320
9.8.6. Tool Position Restriction	325
9.8.7. Planes	328
9.8.8. I/O Planes	331
9.8.9. Tool Direction	335
9.8.10. Hardware	338
9.8.11. Three Position	339
10. Integration Guidelines	340
10.1. Communication Protocols Installation	340
10.1.1. Ethernet	341
10.1.2. Input and Output	342
10.1.3. PROFI-safe	348
10.2. API Documentation	355
10.2.1. Robot-API	355
10.3. Custom Scripting and Automation	355
10.3.1. Script Files	356

10.3.2. URScript Control API in Logic Program	356
10.3.3. URCaps	357
11. Troubleshooting	358
11.1. System Info	358
11.1.1. Logic Program Error Messages	360
11.2. Safety Checksum	362
12. Maintenance	363
12.1. Update	363



1. Introduction

1.1. About This Document

Description	This Software Handbook provides the essential information you need to start using your Universal Robots robot. All functions for using PolyScope X are described here.
Intended Audience	For operators, engineers, and integrators.
Software Version	This document applies to PolyScope X version 10.14. Other software versions may behave differently. Check the PolyScope X software version by tapping the About button in the PolyScope X hamburger menu. The latest version is available at https://www.universal-robots.com/manuals/latest/en/html/polyx/sw-handbook/ on the Universal Robots website.

1.2. How to Use This Handbook

Description	Use this handbook to operate robot models using PolyScope X software after powering on the robot.
--------------------	---



NOTICE

Before powering on the robot for the first time, please refer to the **User Manual**:

- Read the safety information in the robot **User Manual** that you can find in the box.
- Make sure that the robot is mounted according to the requirements set in the robot **User Manual**.
- Check that the robot arm and the Teach Pendant are connected to the Control Box, and Control Box is connected to the mains power supply.

1.3. Related Documentation

Description	This Software Handbook is to be used in conjunction with Universal Robots safety-related features found in the User Manual and Service Manual. Please refer to the third-party providers of any URCaps for their proprietary documentation. Refer to the Script Directory if you are going to create scripts for your robot programs.
--------------------	--



2. Safety Information

Description All safety warnings, symbols, and definitions related to this Handbook are found in the **User Manual**.



READ MANUAL

Read the manual for the robot **before** first use. Follow all safety precautions stated in the robot manual.

3. First Boot

Description	The first boot is the initial sequence of actions you can take with the robot after assembly. This initial sequence requires you to: • Power on the robot • Insert the serial number • Verify Installation • Initialize the robot arm • Test the Mount and Payload using Freedrive • Power down the robot
--------------------	--



CAUTION

Failure to verify the payload and installation before starting up the robot arm can lead to injury to personnel and/or property damage.

- Always verify the actual payload and installation are correct before starting up the robot arm.



CAUTION

Incorrect payload and installation settings prevent the robot arm and Control Box from functioning correctly.

- Always verify that the payload and installation settings are correct.



NOTICE

Starting up the robot in low temperatures can result in lower performance, or stops, due to temperature-dependent oil and grease viscosity.

- Starting up the robot in low temperatures can require a warm-up phase.

3.1. Powering On the Robot

To power on the robot	Powering on the robot turns on the Control Box and loads the display on the TP screen. 1. On the Teach Pendant, press the emergency stop button. 2. Press the power button on the Teach Pendant to power on the robot.
------------------------------	---



3.2. Inserting the Serial Number

To insert the serial number

Installing your robot for the first time requires you to enter the serial number on the robot arm. This procedure is also required when you re-install the software. For example, when you install a software update.

1. Select your Control Box.
2. Add the serial number as it is written on the robot arm.
3. Tap **OK** to end.

It can take a few minutes for the start screen to load.

The screenshot shows the Universal Robots start screen. At the top is the Universal Robots logo. Below it, the screen is divided into two main sections. The left section, titled 'Select Control Box', contains three buttons: 'Standard', 'OEM AC', and 'OEM DC'. The right section, titled 'Enter Serial Number', features a large empty text input field at the top. Below this is a numeric keypad with buttons for digits 1 through 9, 0, and a backspace button (represented by a square with an 'x'). To the right of the numeric keypad is a vertical grey bar with a white 'OK' button at the bottom.

3.3. Verifying the Installation

Verify robot mounting

Mounting settings describe whether the robot is mounted vertically, horizontally, or at an angle. This allows the robot to determine the direction of gravity, enabling smooth and precise movement.

This screen also allows you to specify the orientation of the robot base in relation to the viewer, giving the robot an accurate appearance in the visualizations.



WARNING

Always verify the actual payload and installation are correct before starting up the robot arm. If these settings are incorrect, the robot arm and Control Box will not function correctly and may become dangerous to people or equipment.



CAUTION

Ensure the robot arm is not touching an object (e.g., a table) because a collision between the robot arm and an obstacle might damage a joint gearbox.

To change the mounting settings of the robot arm:

1. On the Teach Pendant, press the emergency stop button.
2. On the Teach Pendant, press the power button and allow the system to start and load PolyScope X.
3. On the screen, tap **OK** when the **Robot Emergency Stop** dialog appears.
4. Tap the **Application** icon in the Main Navigation on the left side of the Teach Pendant screen.

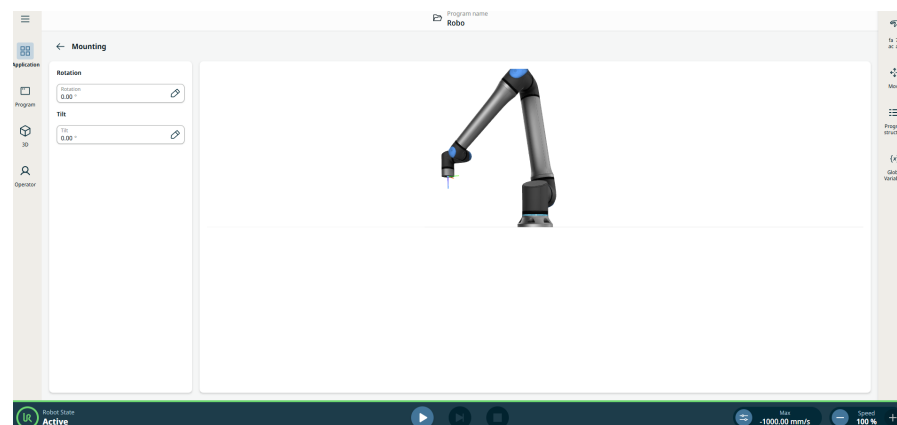
Tap the **Mounting** icon.

A screen appears with two fields on the left side:

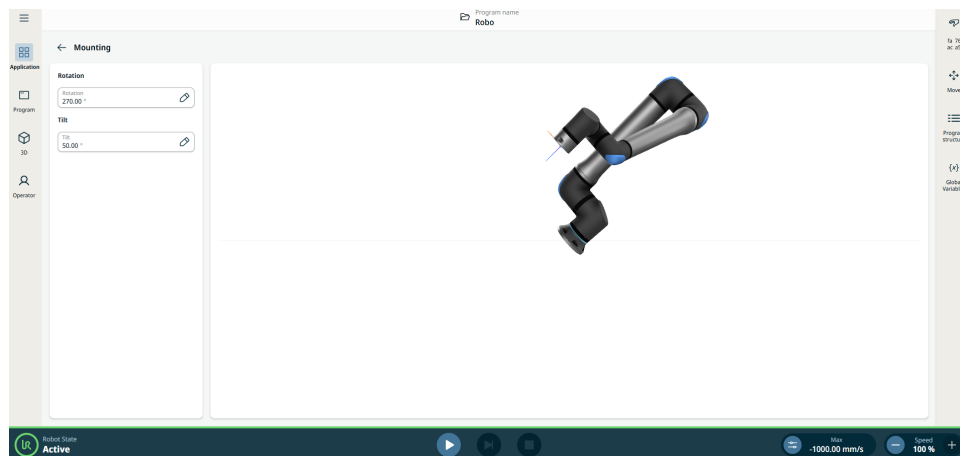
Rotation. The alignment with respect to the power cable.

Tilt. The angle of the base in degrees: 0 being horizontal floor mounting, 90 being wall mounting, and 180 being a ceiling mounting.

The robot arm is seen in the center screen.



5. You can use the following touch gestures to adjust the view of the robot arm. Only the current view is changed.
 - **Press and rotate** clockwise or counterclockwise.
Function: Changes your viewing direction of the robot.
 - **Pinch**. Touch the screen with two fingers and bring them closer together.
Function: Zooms out the robot image.
 - **Spread**. Touch the screen with two fingers and move them apart.
Function: Zooms in the robot image.
6. Tap the **Rotation** field, enter your value in degrees, and **Confirm**.
7. Tap the **Tilt** field, enter your value in degrees, and **Confirm**. The current view is updated to show the corresponding change in the robot arm configuration.



8. Tap the back arrow to return to the previous screen.

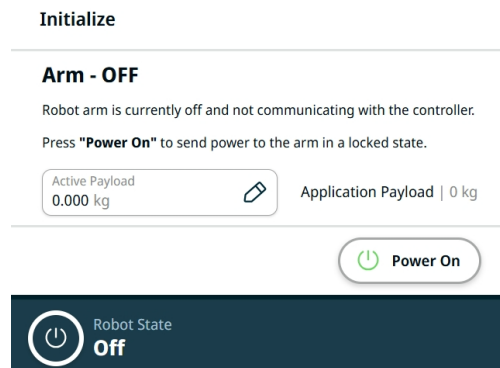
Verify active payload

Starting the robot arm disengages the braking system, allowing you to start moving the robot arm and to start using PolyScope X. Before powering on the robot arm, you need to verify that the correct active payload is configured.

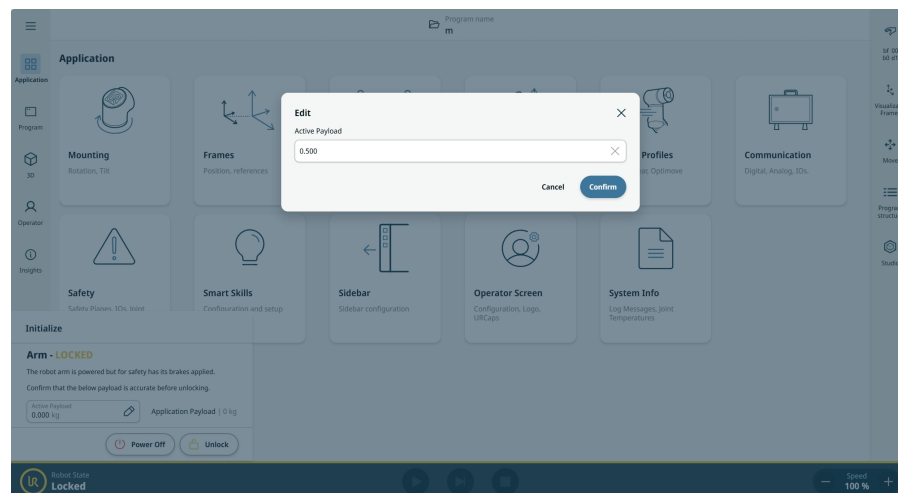
1. Check the **Robot State** in the footer of the screen.

If the state is **Emergency Stopped**, hold and twist the emergency stop button to reset. The arm will then be in **Off** state.

2. Step outside the reach (workspace) of the robot arm.
3. Tap the on-screen **Power** button.



4. In the Initialize box, tap **Power On**. On a new robot or the first boot after the software update, an **Updating Firmware** dialog may appear. When the arm firmware update is complete, the robot state will change to **Locked**.
5. In the **Active Payload** field, verify the payload mass.
6. Tap the **Active Payload** field, and an **Edit** field appears in the main screen.
7. Enter your active payload and tap **Confirm**.



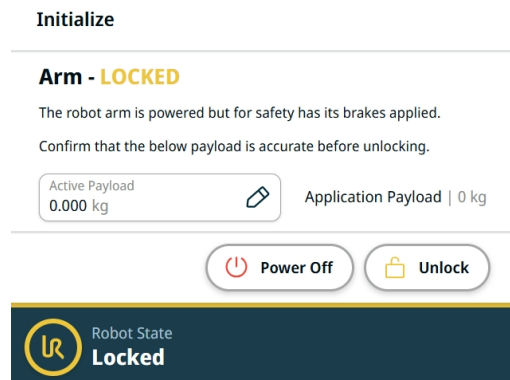
You can now **Unlock** the brakes and start moving the robot arm.

3.4. Starting the Robot Arm

To start the robot

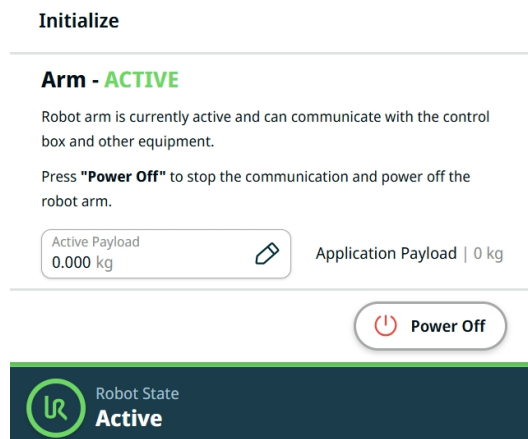
Starting the robot arm disengages the braking system, allowing you to start moving the robot arm and to start using PolyScope X.

1. On the left side of the footer, tap the power or **Robot State** icon.
2. If the robot arm state is **Off**, tap **Power On** in the Initialize box. The robot arm state will change to **Booting** and **Locked**.
3. Tap **Unlock** to release the brakes.



Robot arm initialization is accompanied by sound and slight movements as the joint brakes are released.

4. The robot arm state is now **Active**, and you can start to use the interface.



5. You can tap **Power Off** to turn off the robot arm or check **Freedrive** to verify robot arm mounting.

3.5. Testing Mount and Payload using Freedrive

Description Freedriving the robot allows quickly check if the robot is functioning normally and mounting settings and payload are correctly specified.

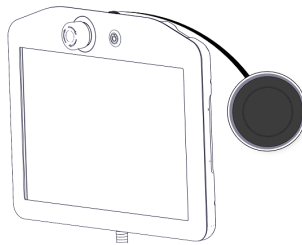
Detailed description of the Freedrive could be found in the Freedrive section.

**To test
Freedrive**

1. Set **Active Payload** to zero if there are no tools attached to the robot arm. Alternatively, set the **Active Payload**, TCP (tool center point) and CoG (center of gravity) according to tool specifications. See **Payload and Center of Gravity** for details.
2. To move the robot arm in Freedrive, see figure below.

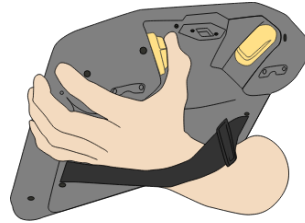
Standard TP

On a standard teach pendant (TP), press and hold the Freedrive button.

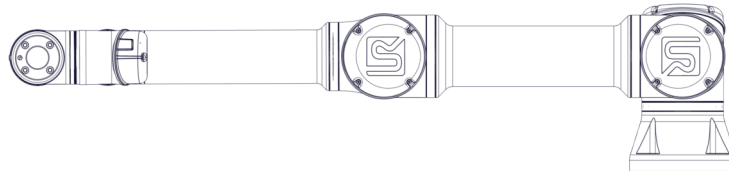


3PE TP

On a 3PE TP, rapidly light-press, release, light-press again, and keep holding the 3PE button in this position.



3. Pull/Push the robot arm to a horizontally elongated position and release. See figure below.



Elongated horizontal position of UR robot

4. Verify the robot arm can maintain this position without support while you are pressing the Freedrive or 3PE button.



3.6. Powering Down the Robot

To power down the robot arm



WARNING

Unexpected startup and/or movement can lead to injury.

- Power down the robot arm to prevent unexpected startup during mounting and dismounting.

1. On the left side of the footer, tap the **Robot State** icon to turn off the robot arm.
The icon color changes from green to white.
2. Press the power button on the Teach Pendant to turn off the Control Box.
3. If a **Shutdown** dialog box appears, tap **Power Off**.

You can now:

- Unplug the mains cable/power cord from the wall socket.
- Wait 30 seconds for the robot to discharge any stored energy.

3.7. Robot Registration and License File

Description

It is necessary to register the robot and download and install the License File, because the license file will include all available software licenses.

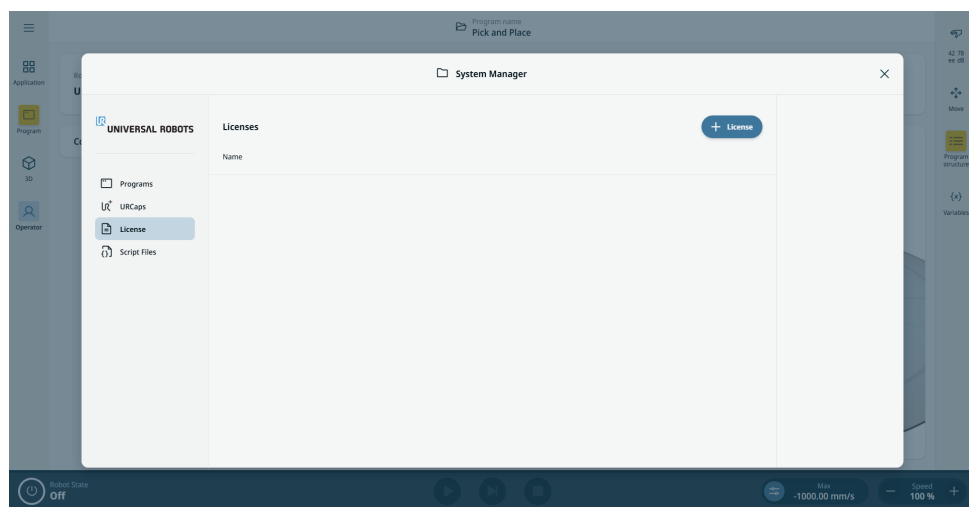
Activate the Software Licenses



NOTICE

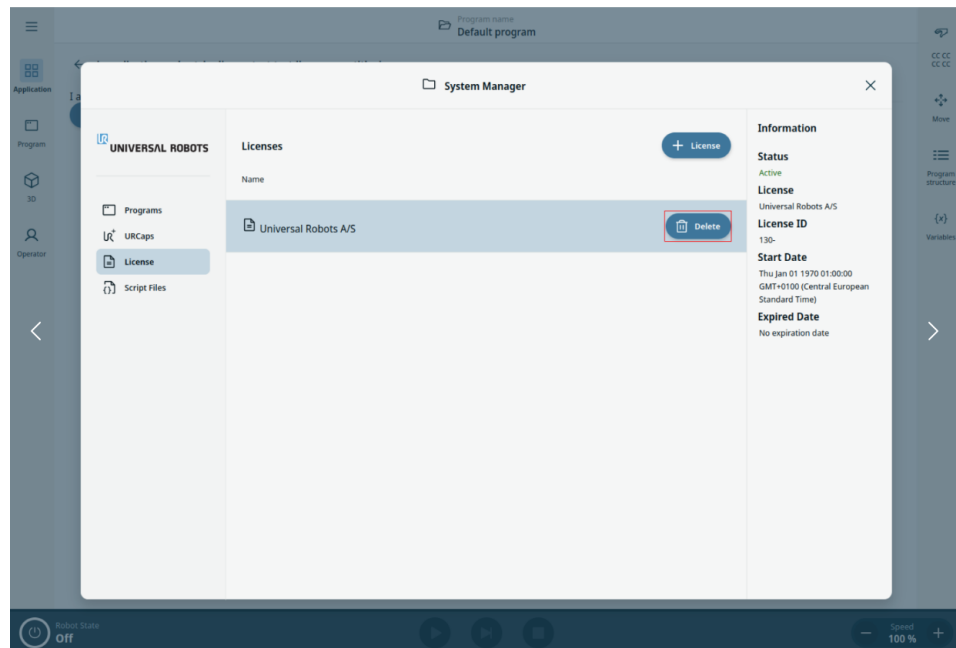
If you have more than one active license, all licenses will be included in the downloaded license file.

1. Download the license file to your PC.
2. Copy the license file to the USB drive and connect it to the Teach Pendant.
3. On the Main navigation, tap the **Hamburger menu** and select **System Manager** to display content.
4. In the list, select the **License** and tap **+License**.
5. Select the license file saved and tap **Open** to confirm robot registration.
6. On the bottom left, tap **Exit**.



Deactivate Software Licenses

- A new license file is required if the robot changes owners. In this case, the license file must be deactivated.
 - If you purchase a new software license for your robot, it is necessary to deactivate and reactivate the license file in order to add the new software license.
1. In the Main navigation, tap the **Hamburger Menu** and select **System Manager**.
 2. In the list, select **Licenses**.
 3. Select the license to deactivate/remove and tap **Delete**.



4. PolyScope X Components

Overview

PolyScope X is the software installed on the teach pendant that operates the robot arm via touch screen. The PolyScope X interface allows you to create, load, and execute programs.



NOTICE

Always update your software to latest version.

To view Main Screen

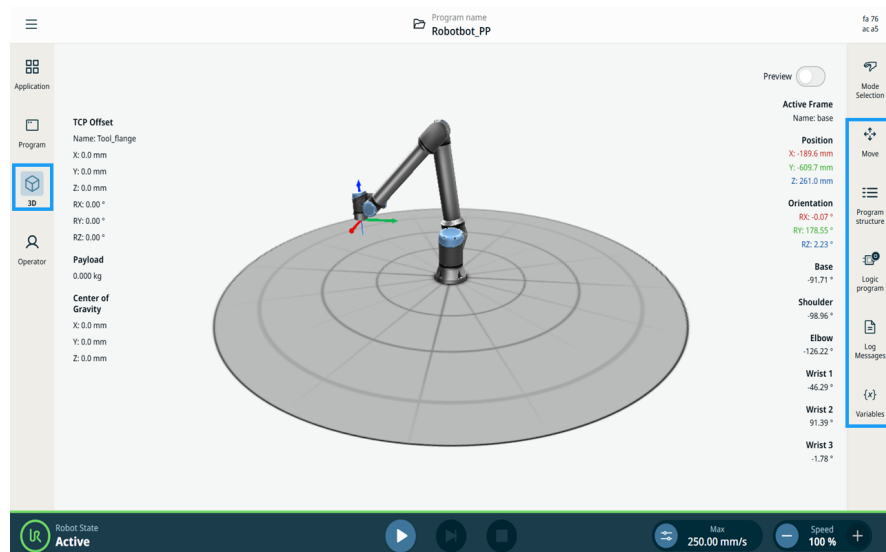


1. Tap the 3D Viewer icon on the main navigation. This gives you a three-dimensional view of the robot arm in X-Y-Z coordinates.
2. To maximize the 3D viewing area, collapse the multitask screen using the sidebar icons:



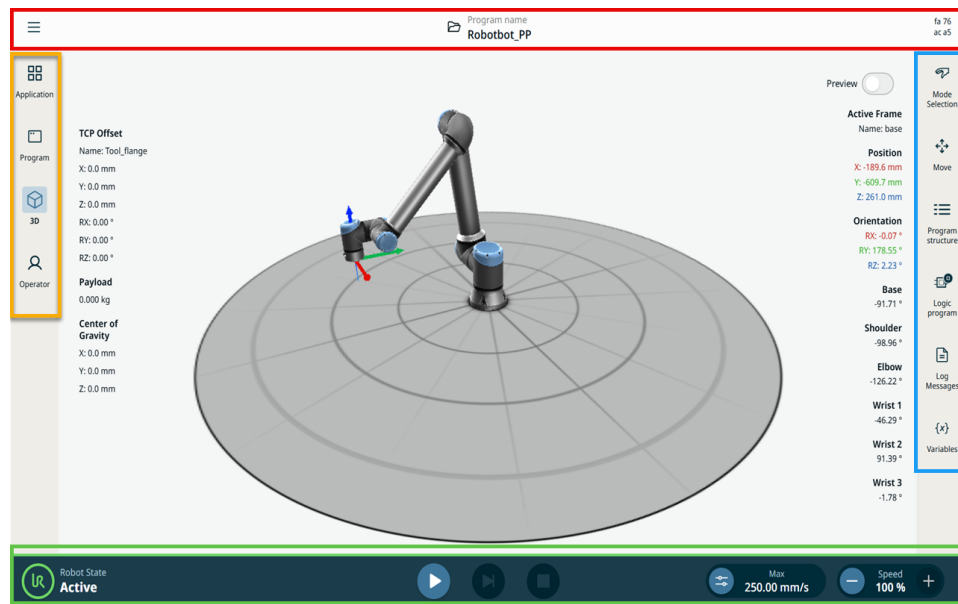
icons:

- Tap once to switch between panels of the multitask screen
- Tap an already-selected icon to close the multitask screen



Screen Layout

The PolyScope X screen interface is divided as shown in the following illustration:



- **Header** - in red-border box. Also called **system manager**, which contains a folder to load, create, and edit programs and access URCaps.

Contains three icons:

- Hamburger icon
- System manager
- Safety checksum

- **Main Navigation** - in orange-border box. Also called **navigation hub**.

Contains icon/fields to select a main screen:

- Application
- Program
- 3D Viewer
- Operator Screen

- **Sidebar** - in blue-border box. Also called **multitask screen**.

Contains icon/fields to select a multitask screen:

- Mode Selection
- Move
- Program structure
- Logic program
- Log messages
- Variables

- **Footer** - in green-border box. Also called **robot control bar**.

Contains buttons to control robot state, speed, and program run/play.

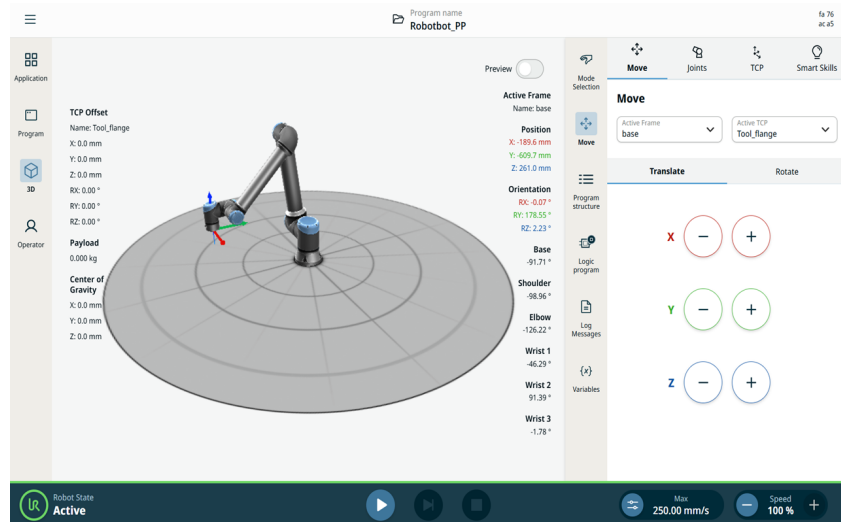
- Robot state initialization
- Play/pause/stop
- Manual high speed mode
- Speed slider

Screen Combinations

The main screen and the multitask screen make up the operating screen combination for the robot.

The multitask screen is independent of the main screen so you can do separate tasks. For example, you can configure a program in the main screen while moving the robot arm in the multitask screen. You also can hide the multitask screen if it is not needed.

- **Main screen**
Contains fields and options to manage and monitor robot actions.
- **Multitask screen**
Contains fields and options often relating to the main screen.



Main screen and Multitask screen

To show/hide the Multitask Screen

1. In the sidebar, tap any icon to show the multitask screen.
The sidebar expands to the middle of the screen so the multitask screen becomes visible.
2. Tap the currently selected icon in the sidebar to hide the multitask screen.

4.1. Header

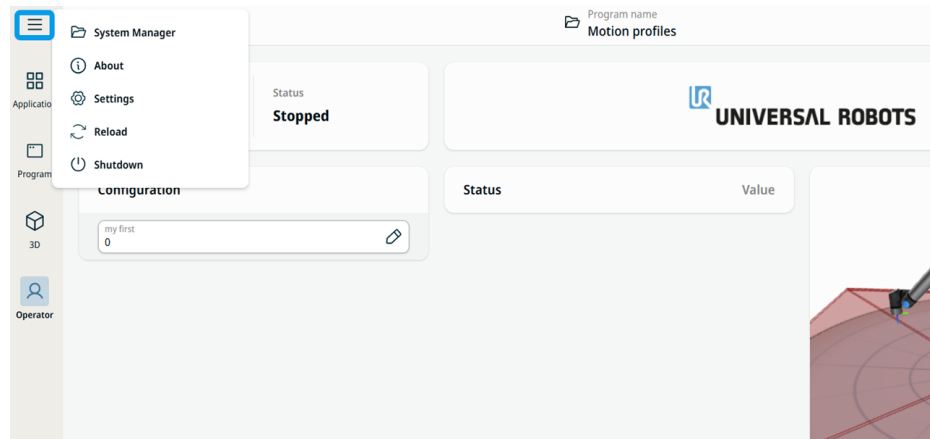
Description

The header contains three functioning menus and feature:

- Hamburger menu
- System manager or Program name menu
- Safety checksum



To open the Hamburger menu

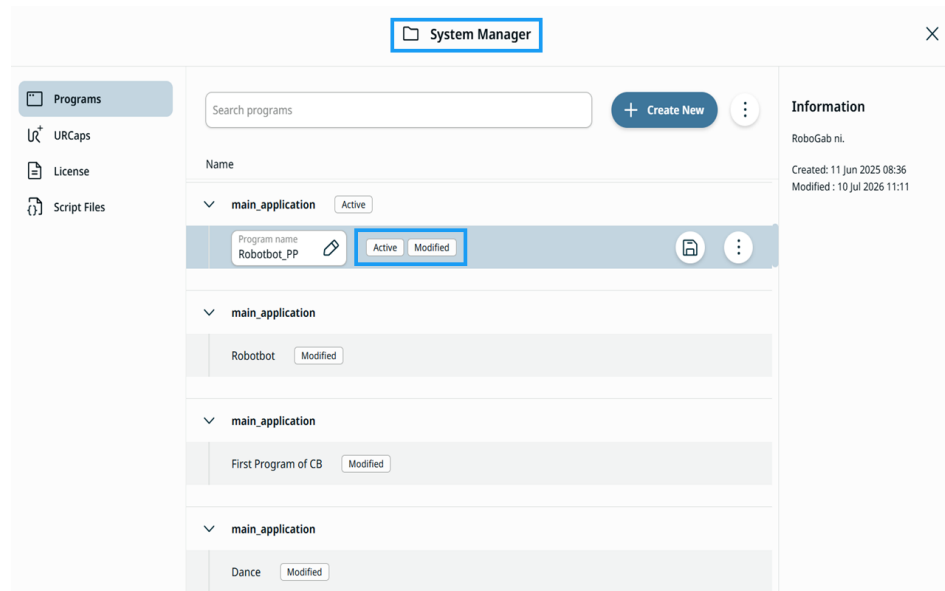


To open Program name

When you tap the **Program name** menu, the **System Manager** screen appears, which enables you to do several things:

- View a list of all installed programs
- See the loaded program, marked **Active**
- Create new programs
- Open a created program
- Search programs
- Import and export a program
- Load a program

- Install URCaps
- Ensure the software license
- Add a script file
- Write a program description
- See program details
- View script



To open

Safety

checksum

Tap the Safety checksum icon on the right side of the header to open the robot's Safety Overview. The items in the Safety Overview panel correspond to the safety settings you configured in the Application tab. Use the scroll bar to review your complete safety configuration.



NOTICE

This icon shows the default Safety checksum for a newly created program in a new application.

Safety Overview

Speed Limit

Power & Force

Joint Position

Joint Speed

Planes

I/O Planes

Tool Direction

Tool Position

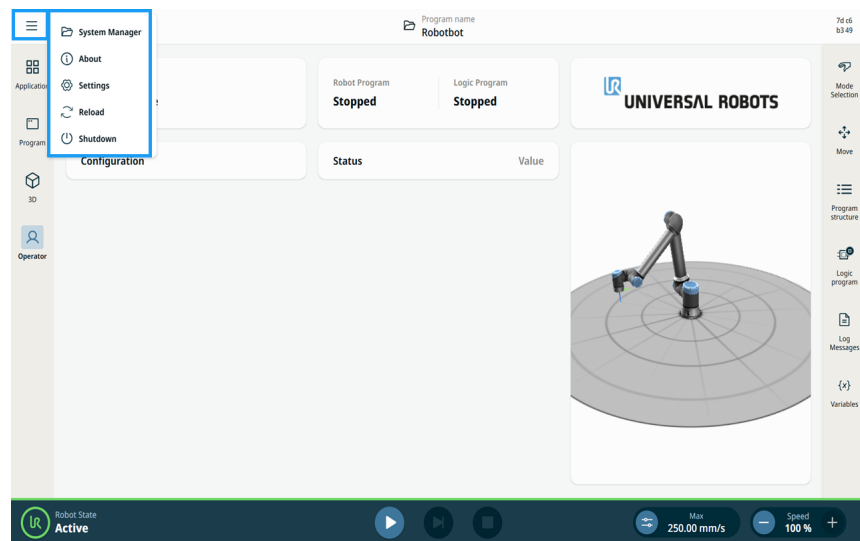
Input

Limit	Normal	Reduced
Tool Speed	1.5 m/s	0.75 m/s
Elbow Speed	1.5 m/s	0.75 m/s
Momentum	25 kg·m/s	10 kg·m/s
Stopping Time	0.4 s	0.3 s
Stopping Distance	0.5 m	0.3 m

4.1.1. Functions Inside the Hamburger Icon

Description The **hamburger icon** contains five functions of PolyScope X:

- System Manager
- About
- Settings
- Reload
- Shutdown



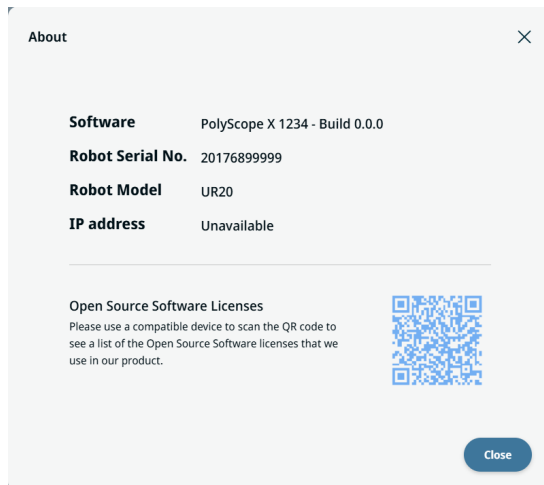
System Manager

The **System Manager** has the same function as with the header. See [Header](#).



About

The **About** function requires no further action to perform in another field or tab. When you tap About, you see the software version, robot serial number, and the QR code of the Open Source Software licenses. The **robot model** and its **IP address** are shown to provide users with quick access to essential system information.



Settings

The **Settings** function gives you access to **General**, **Password**, **Connection**, and **Security** settings. Except for System in the general settings, all are password protected.

General

This setting contains System and Update.

System setting enables you to change the language, theme, and units used in the system.

Update setting is where you are asked to insert a USB storage and check for update.

Password

This setting contains Operational Mode, Safety, and Admin.

Operational Mode password is used when switching between automatic and manual mode.

Safety password is used to change safety settings.

Admin password is used to change system settings. Observe extra care when changing this, as admin password is nonrecoverable. When password is not retrieved, software reinstallation is needed.

Connection

This setting contains Network and UR Connect.

Network enables you to apply network configuration after choosing either DHCP or Static.

UR Connect enables you to connect to myUR Cloud, which allows you to access your robot data anywhere.

Security

This setting contains Secure shell, Permissions, and Services.

Secure shell manages SSH access to the system.

Permissions enables you to select system pages to be protected by Admin password.

Services lets you disable or enable interfaces.

Reload The **Reload** function requires no further action to perform in another field or tab. When you tap Reload, the default settings defined in the application will be applied.

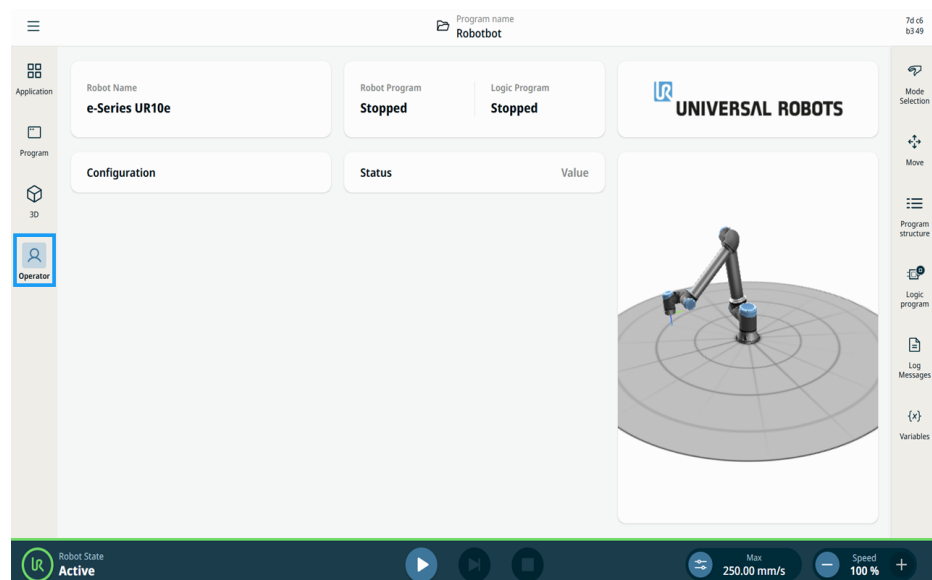
Shutdown The **Shutdown** function requires no further action to perform in another field or tab. When you tap Shutdown, you are given the choice to restart or power off. Program state will be saved before restarting or powering off the robot.

4.2. Main Navigation

Description The main navigation, also called the navigation hub, contains five settings/menus:

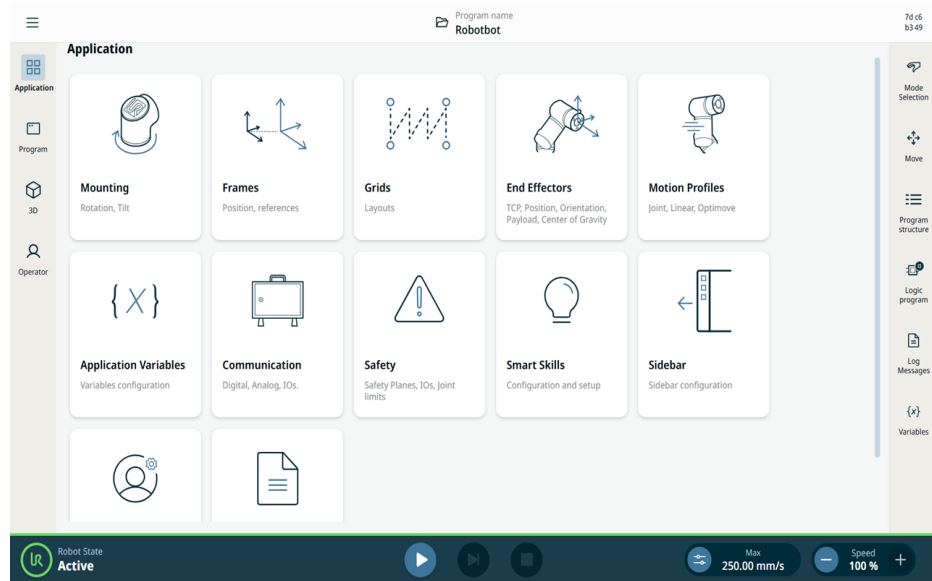
- Application
- Program
- 3D Viewer
- Operator Screen

More about the Main Navigation When you open PolyScope X, your main screen is by default the Operator Screen. The Operator icon, highlighted in blue, indicates this.

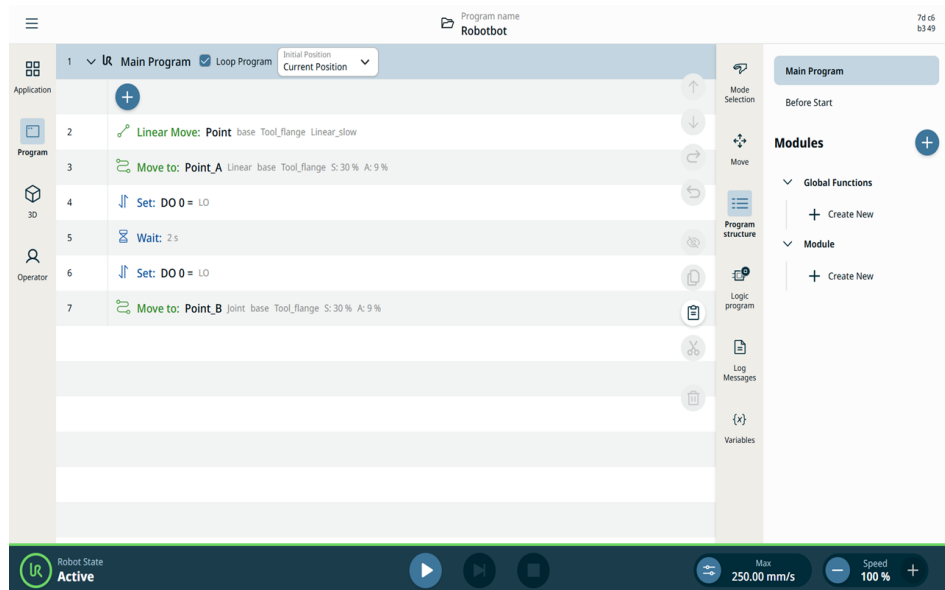


When you tap an icon, it is highlighted blue in the main screen, which indicates it is active. The hamburger icon is an exception to this.

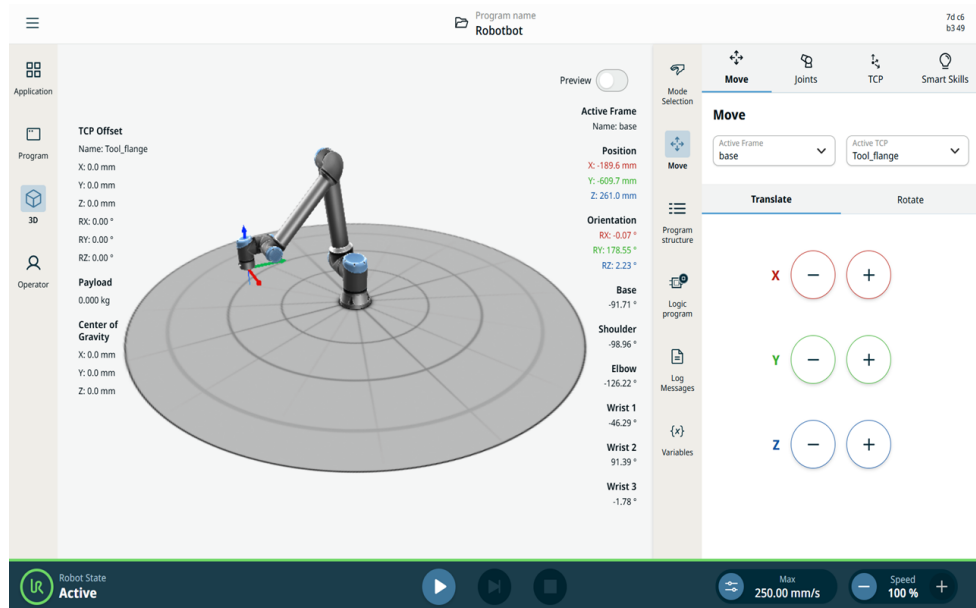
Tap Application



Tap Program



Tap 3D



4.3. Sidebar

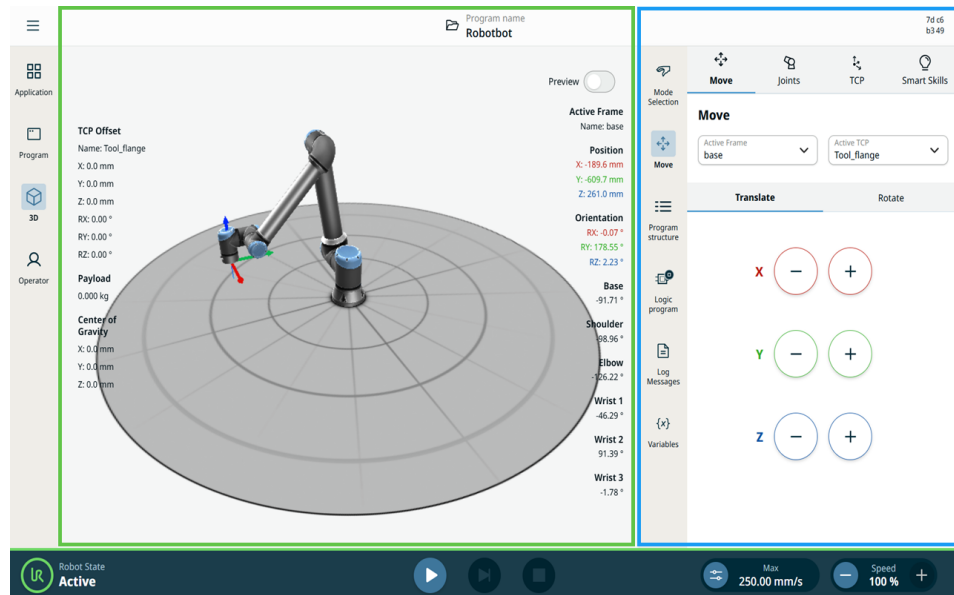
Description The sidebar contains the following menus:

- Mode Selection
- Move
- Program Structure
- Logic Program
- Log Messages
- Variables



Using the Sidebar

When you tap a menu in the sidebar, its content expands into the center of the screen. This center area is the multitask screen (blue box). Any changes you make in the multitask screen appear as graphical responses from the robot arm in the main screen (green box).



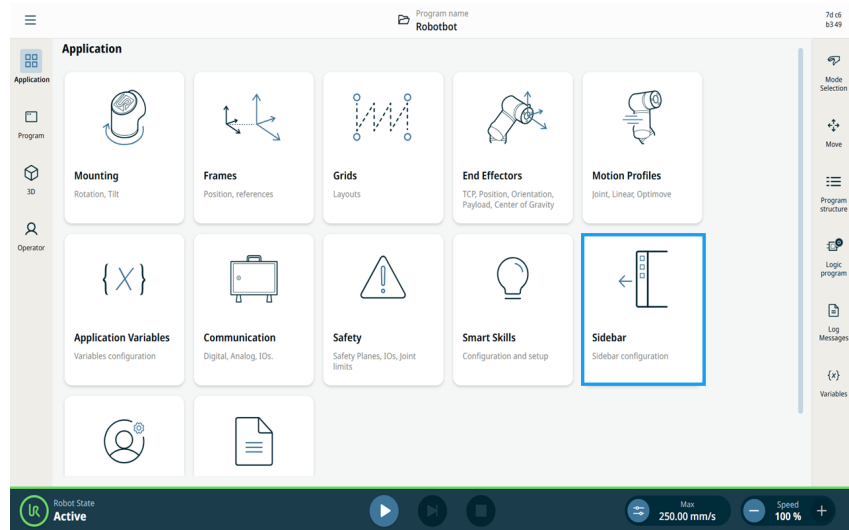
4.3.1. Sidebar in the Application Tab

Description

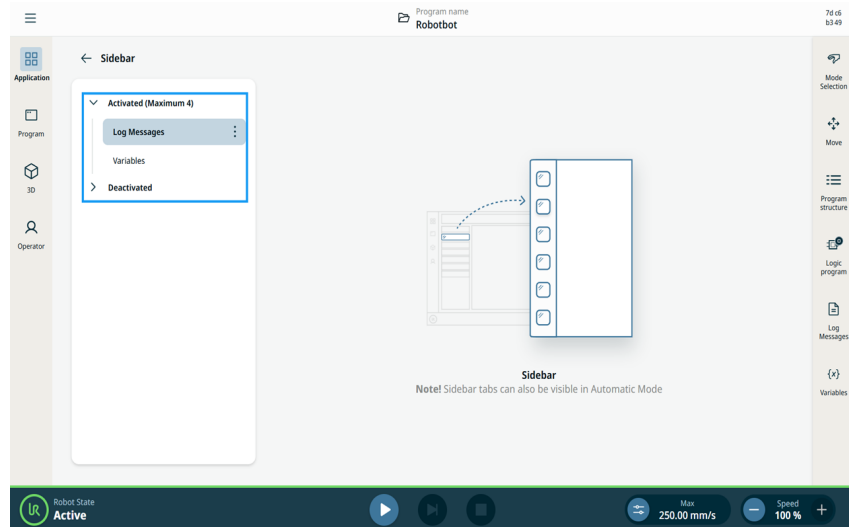
The **Sidebar** in the [Application Tab](#) lets you add tabs in addition to the four main menus in the [multitask screen](#).

Using the Sidebar Functionality

1. Go to the application nodes screen. See [9.3 Application Tab on page 279](#).
2. Tap the **Sidebar** icon.

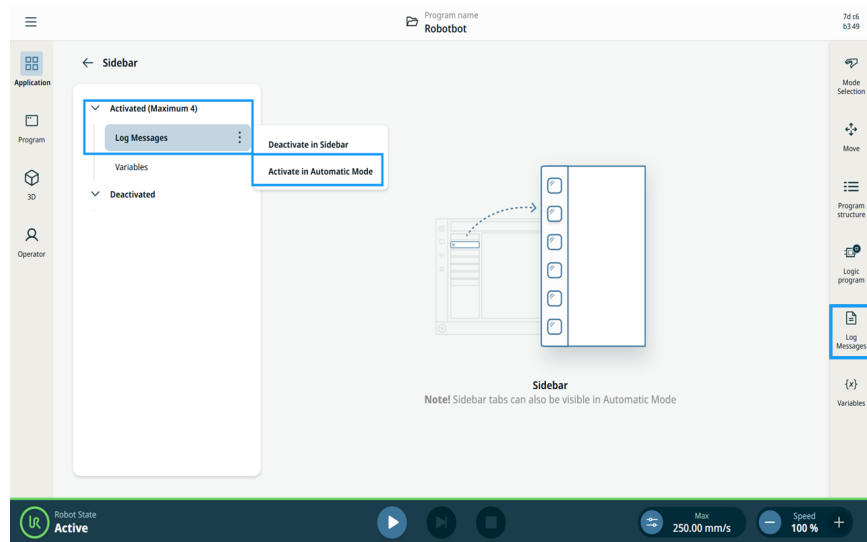


The main screen shows the **Activated** and **Deactivated** sidebar trees.

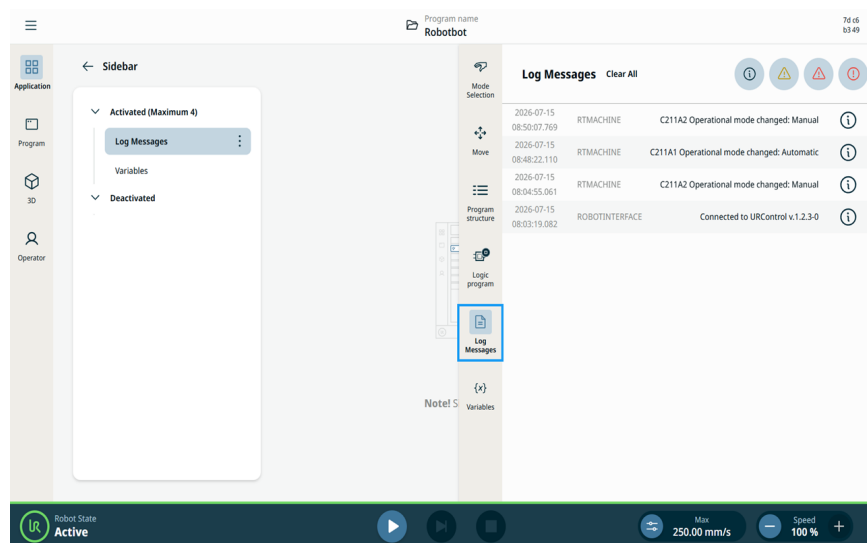


Log Messages

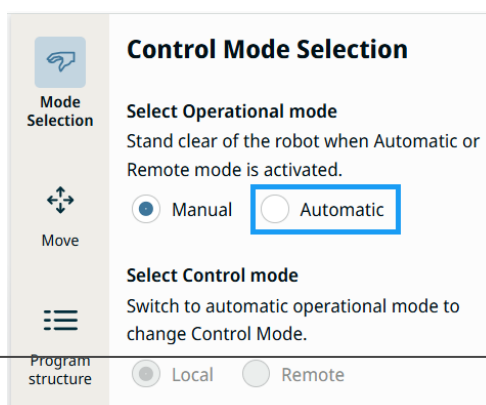
1. Tap the three vertical dots (kebab icon) for **Log Messages**, and choose **Activate in Sidebar** and **Activate in Automatic Mode**. **Log Messages** is then moved to the **Activated** sidebar tree.
2. In the sidebar, **Log Messages** appears.



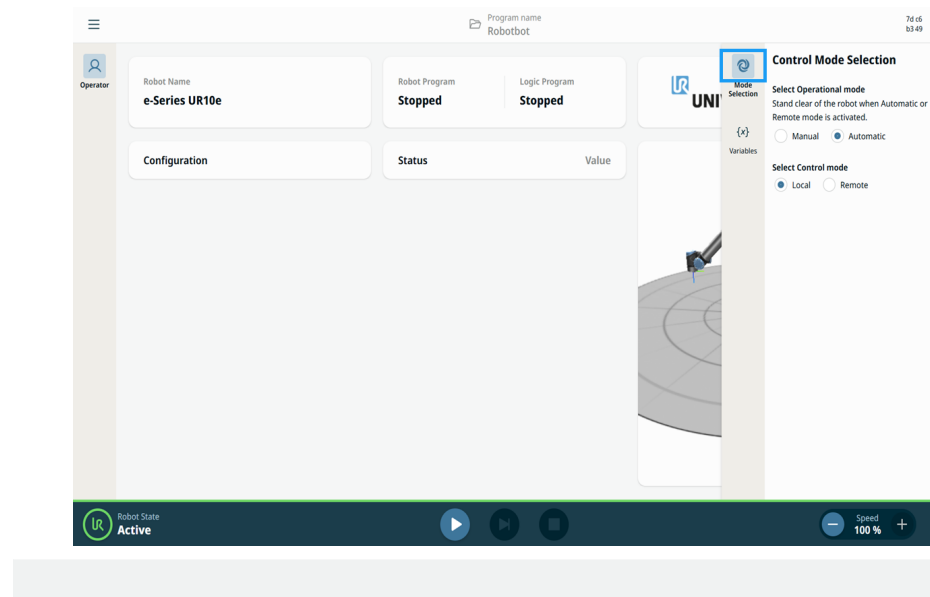
3. Tap **Log Messages** in the sidebar. The multitask screen expands to show **Log Messages**.



4. To switch to automatic mode, tap the **Mode Selection** icon.



5. Choose **Automatic** in the **Mode Selection** screen. Enter the [operational mode password](#) to unlock and confirm.
6. The automatic-mode interface appears, indicated by the swirling icon in **Safety Checksum**, and only the **Variables** remains visible in the sidebar.



Disabling Contribution Function

You can disable a function to control the availability of each contribution at runtime. This function returns a Boolean value that indicates whether the contribution is disabled.

- When a contribution is disabled, it is hidden from the user interface for a cleaner experience.
- Even when it is hidden, the function continues running in the background, so it can maintain state or perform background tasks.
- You can change the disabled state dynamically, which gives you flexible control over feature availability without stopping the underlying processes.

4.4. Footer

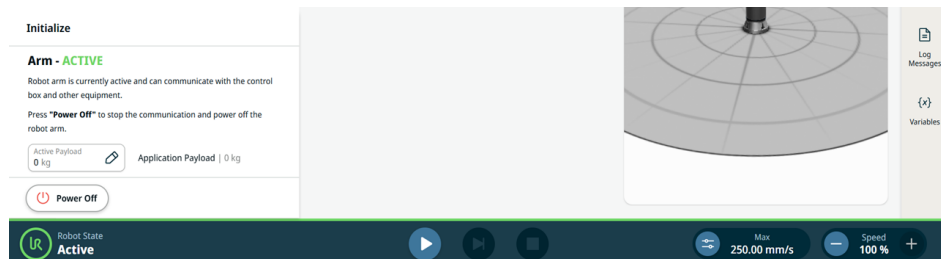
Description

The **Footer**, also called robot control bar, contains buttons to control robot state, program play and stop, and control sliders for manual high speed and robot speed.

Functions in Footer

In the footer, you initialize the robot arm:

- Power On
- Power Off
- Edit the active payload
- Unlock
- Enable manual high speed
- Adjust the robot speed



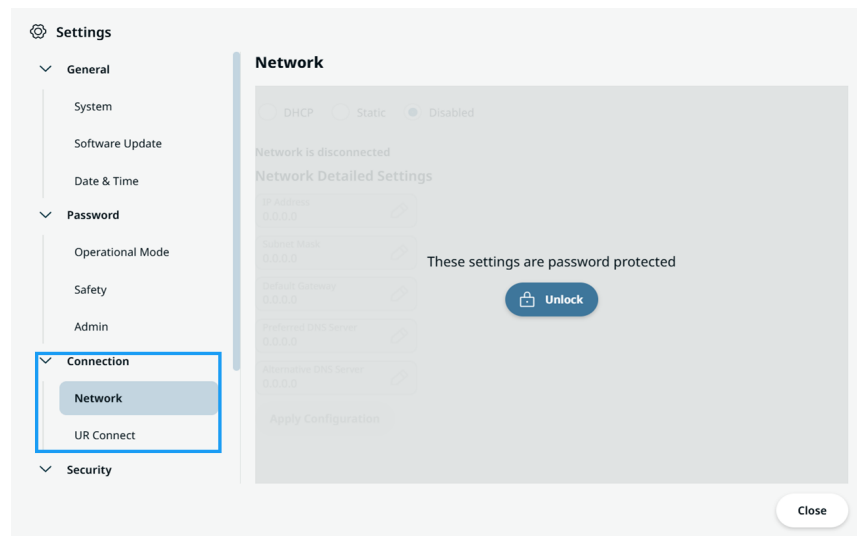
5. Network Connection

Description The **Connection** setting of PolyScope X software enables you to access robot data anytime, anywhere.
You can access the Connection setting via the hamburger menu.

To access Connection tab

The main screen can be seen below.

1. On the left header, tap the hamburger menu  .
Five tabs appear. **Settings** tab is the third.
2. Tap **Settings**.
 - A sidebar opens with the **Connection** tab, below General and Password tabs.
 - You can see **Network** and **UR Connect** bars under the **Connection** tab.
It is in these two functions where the input to connect network to PolyScope X is applied.



5.1. Network

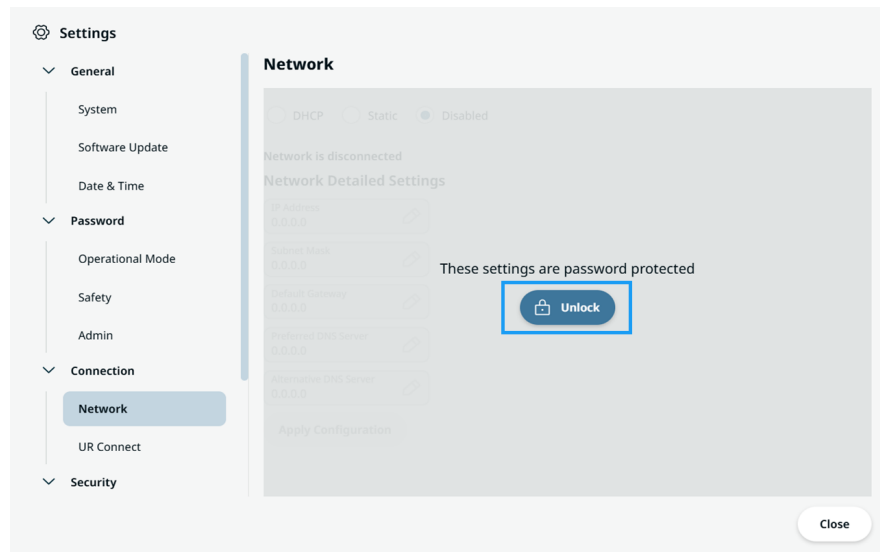
Description The network settings are used to connect the PolyScope X to a secure network.



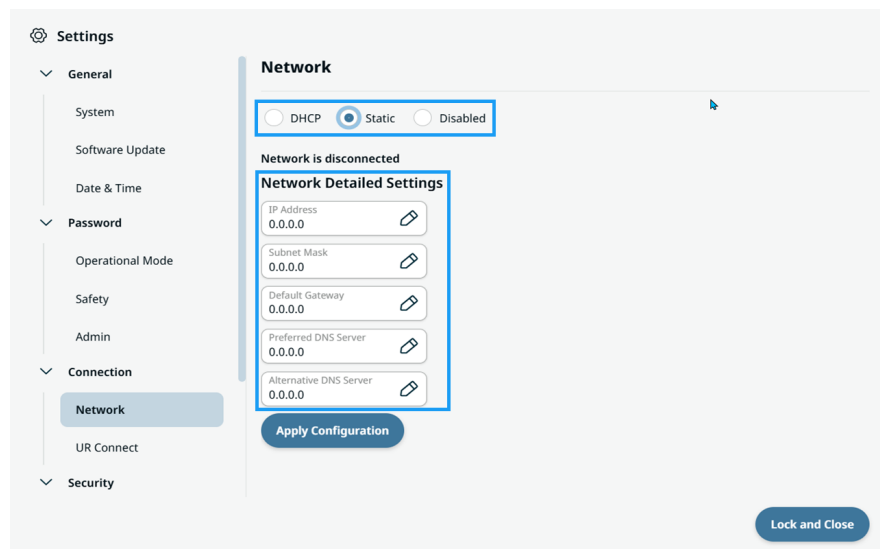
To connect via Network

In the **Network** screen:

1. Tap **Unlock**.



2. Type the admin password.
3. Choose your preferred Network choice and confirm. Your main screen shows you three Network choices.
4. Tap your preferred Network choice.
5. When **Static** is chosen, fill in the box of the five required fields of Network Detailed Settings.



7. Tap **Apply Configuration**.

6. Operating the System

Description	What's in this chapter • PolyScope X User Interface • Operational Mode • Starting and Stopping the Robot • Programming Workflows • Common Operating Tasks
--------------------	--

6.1. PolyScope X User Interface

Description	Use the PolyScope X interface on the teach pendant to create, load, and execute programs that control the robot arm via touch screen. You define robot tasks through node-based programming, which runs on a web-based technology stack.
--------------------	--



6.1.1. Icons

Header Icons	Icon	Title	Description
		Program name	Gives access to System Manager. Allows you to load, save, and add programs and URCaps files.
		More	Access information about robot version, serial number, and settings.
		Safety Checksum	Displays the active safety checksum and gives access to detailed parameters of each robot arm parts, and changes the operational mode.

Main Navigation Icons

Icon	Title	Description
	Application	Configures and sets up the robot arm settings and safety, including end effectors and communication.
	Program	View and modify robot programs.
	3D	Enables control and regulation of robot movement in X, Y, Z coordinates.
	Operator	Operates the robot using prewritten programs and shows the status of the robot.

Icons Inside the Hamburger Icon

Icon	Title	Description
	System Manager	Gives access to System Manager. Allows you to load, save, and add programs and URCaps files.
	About	Displays information about robot version and serial number.
	Settings	Configures system settings, such as language, units, password, and security.
	Reload	A safe function to apply the default settings defined in the application.
	Shutdown	To restart, power on and off the robot.



Sidebar Icons

Icon	Title	Description
	Mode Selection	Provides operational mode and control mode selections.
	Move	Comprehensive function for robot movement, detailing the joints, TCP, flange, base.
	Program structure	Provides structural overview of the main program, modules and functions. Access to add modules.
	Logic program	Enables you to run a logic program and lists the number of running tasks.
	Log Messages	Lists all log activities you performed in the software.
	Global Variables	Provides access to global variable names and their values.

Footer Icons	Icon	Title	Description
		Initialize	Manages the robot state. When RED, press it to make the robot operational. • Black, Power off. The robot arm is in a stopped state. • Orange, Idle. The robot arm is on, but not ready for normal operation. • Orange, Locked. The robot arm is locked. • Green, Normal. The robot arm is on and ready for normal operation. • Red, Error. The robot is in a fault state, such as e-stop. • Blue, Transition. The robot is changing state, such as brake releasing.
			
			
			
			
			
		Play	Starts the current loaded program.
		Step	Allows a program to be run single-stepped.
		Stop	Halts the current loaded program.
		Speed Slider	Shows in real time the relative speed at which the robot arm moves, taking safety settings into account. The High Speed Manual slider is only available in manual mode when a Three-Position Enabling Device is configured. High Speed Manual Mode allows tool speed and elbow speed to temporarily exceed the default speed limit.
		High Speed Manual Mode	

Copyright © 2009-2026 by Universal Robots A/S. All rights reserved.



Main
Screen
Icons

Icon

Title

Description



Move up

To move up a command node in a program tree.



Move down

To move down a command node in a program tree.



Revert

To revert a recent move of a command node in a program tree.



Undo revert

To undo revert a recent move of a command node in a program tree.



Suppress/
Unsuppress

To suppress and unsuppress a command node in a program tree.



Copy

To copy a command node to another program tree.



Paste

To paste a command node to another program tree.



Cut

To cut a command node from a program tree.



Delete

To delete a command node in a program tree.

6.1.2. Touch Screen

Description	The Teach Pendant touch screen is optimized for use in industrial environments. Unlike consumer electronics, teach pendant touch screen sensitivity is, by design, more resistant to environmental factors such as: • Water droplets and/or machine coolant droplets • Radio wave emissions • Other conducted noise from the operating environment
--------------------	--

Using the Touch Screen	The touch sensitivity is designed to avoid false selections on PolyScope X and to prevent unexpected motion of the robot. For best results, use the tip of your finger to make a selection on the screen. In this manual/handbook, this is referred to as a tap. A commercially available stylus may be used to make selections on the screen, if desired. The preceding section lists and defines the icons/tabs and buttons in the PolyScope X interface.
-------------------------------	--

6.1.3. System Manager

Description	The System Manager enables you to do the following functions: • See all created programs and installed URCaps • Create a new program • Import programs • Export programs • View script of selected program • Write a program description • Add a License
--------------------	--

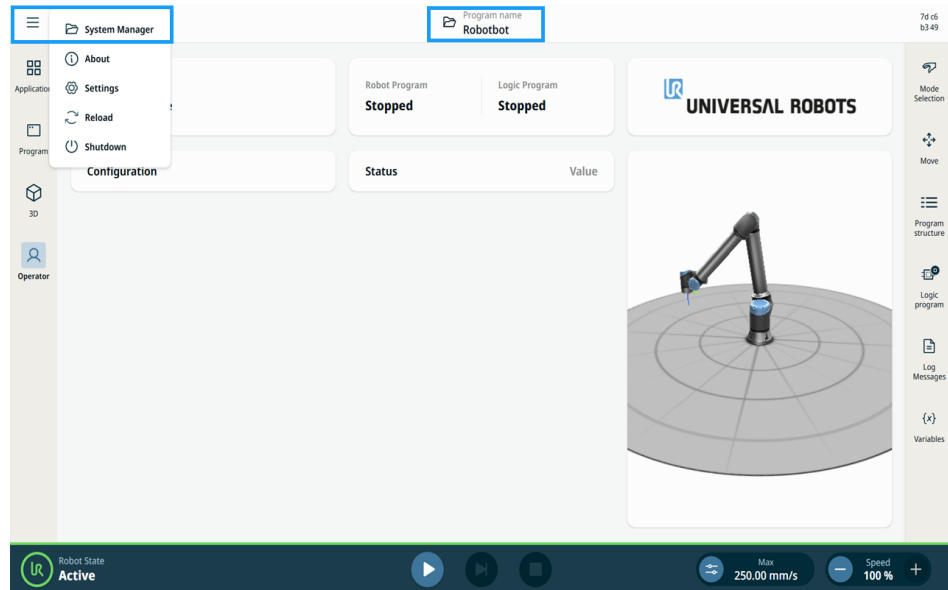


To access System Manager

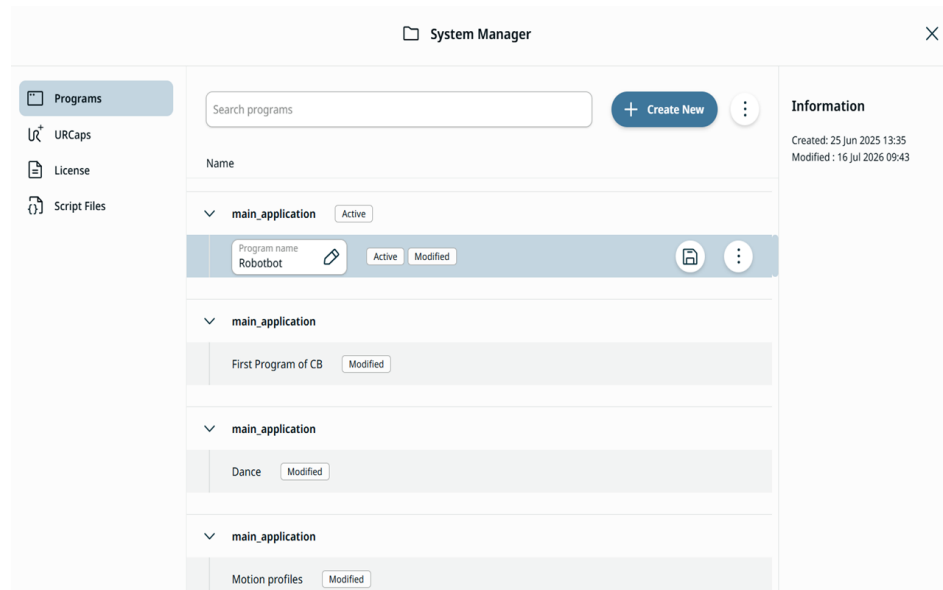
There are two ways to access the **System Manager**:

Program name: Tap the program name in the **Header**.

Hamburger Icon: Tap the hamburger icon and then System Manager.



The **System Manager** screen appears, which contains the following components located in three columns.



On the left column are found the following sections:

- Programs
- URCaps
- License
- Script Files

On the **Programs** tab, Applications and programs are shown in a unified accordion. Each **Application** can be linked to multiple programs. The Application used by the currently open program is shown in **bold** with an **Active** tag. See [Shared Applications in System Manager](#) for details.

The center column lists Applications and their linked programs. The **+ Create New** button/icon enables creation of a new program and importing a program.

On the right column are found the **Information** panel, consisting of the created and modified dates of the program.

6.1.4. 3D Viewer

Description

The **3D** viewer shows in the main screen a three-dimensional view of the robot arm in X, Y, Z coordinates. Measurements of the robot arm parts position, orientation, payload, center of gravity, and TCP offset are given in both sides of the main screen.

You can see the multitask screen, which enables you to move the robot using the editable fields, tabs, and plus-minus button.

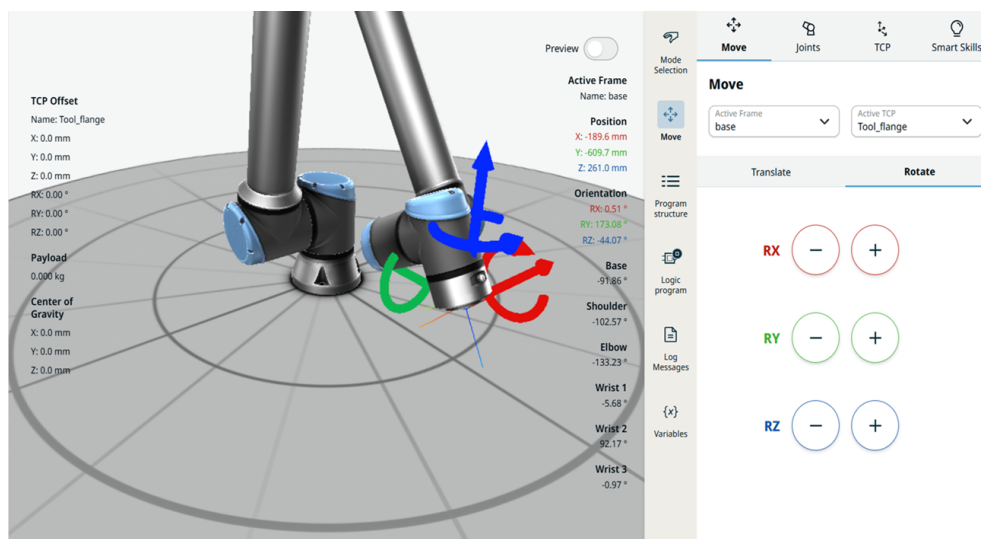


To view in 3D Tap the 3D icon, and the main screen and multitask screen appear simultaneously.
Use the following touch gestures applicable in the **main screen**:

- **Press and rotate** clockwise or counterclockwise.
Function: Changes the viewing direction.
- **Pinch**. Touch the screen with two fingers and bring them closer together.
Function: Zooms out the robot image.
- **Spread**. Touch the screen with two fingers and move them apart.
Function: Zooms in the robot image.

In the **multitask screen**, tap the plus-minus button of X, Y, Z to translate and the RX, RY, RZ to rotate. A responsive, coordinated movement of the robot arm in the main screen is shown simultaneously.

See the color-coded indications of the straight and circular arrow in the robot arm.



6.1.5. Mouse Cursor Support

Description You can use an external mouse on the Teach Pendant starting with SW 10.14.0 for PolyScope X.

Previously, the mouse cursor was disabled at the operating system level, so an external mouse did not work even when one was connected.

How it works

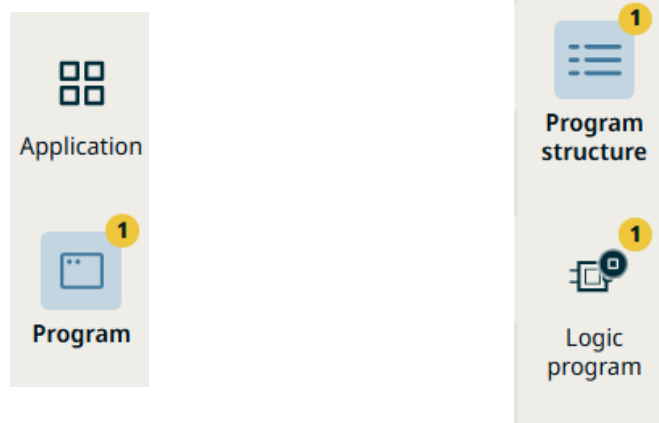
- When you connect an external mouse to the Teach Pendant, PolyScope X enables the mouse cursor automatically.
- When you disconnect the mouse, PolyScope X hides the cursor again.
- This behavior applies to all PolyScope X screens.

6.1.6. Validation Indicators

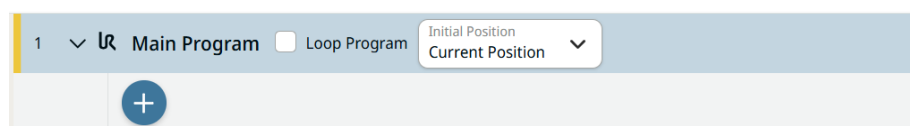
Description Starting in SW 10.14.0 for PolyScope X, validation errors are shown with notification count badges and left-border highlights instead of yellow icon buttons. Validation indicators tell you when program or settings input fails validation—for example, when a command node is incomplete, a reference is missing, or an expression is invalid. Previously, yellow icon buttons marked validation errors in the **Sidebar**, **Main navigation**, program tree, and settings dialogs. The updated design is cleaner and less intrusive while still showing where errors exist. Validation indicators are not the same as safety faults, E-stop states, or controller error dialogs.

How it works

- In the **Sidebar** and **Main navigation**, a notification count badge on an icon shows how many validation errors exist in that area.



- In the program tree, invalid command nodes are marked with a left-border highlight instead of a yellow icon.



- A program tree node is shown as invalid only when that node itself is invalid.
- A parent node in the program tree is shown as invalid only when it is collapsed and contains invalid child nodes.
- In move command settings dialogs—such as **Direction**, **Move C**, **Move P**, **Move To**, **Moves To**, and **Path**—invalid tabs use the standard tab validation state instead of yellow icons.



To find and fix validation errors

1. Look for notification count badges in the **Sidebar** or **Main navigation**, or for left-border highlights in the program tree.
2. Tap the icon or node with the indicator to open the area that contains the error.
3. Review the highlighted node or invalid tab and complete or correct the required input.
4. When the input is valid, the badge, border highlight, or tab indicator is removed automatically.

For an overview of the program screen and program tree, see [Program Menu](#).

6.2. Operational Mode

Description You access and activate different modes using Teach Pendant or the Dashboard Server. If an external mode selector is integrated, it control the modes - not PolyScope or the Dashboard Server.

Automatic Mode When this mode is activated the robot can only execute a program of pre-defined tasks. You cannot modify or save programs and installations.

Manual Mode When this mode is activated you can program the robot. You can modify and save programs and installations. The speeds used in Manual Mode must be limited to prevent injury. When the robot is operating in Manual Mode, a person could be positioned within reach of the robot. The speed must be limited to the value that is appropriate for the application risk assessment.



WARNING

Injury can occur if the speed used, while the robot is operating in Manual Mode, is too high.

Recovery Mode This mode activates when a safety limit from the active limit set is violated, the robot arm performs a Stop Category 0. If an active safety limit, such as a joint position limit or a safety boundary, is violated already when the robot arm is powered on, it starts up in Recovery mode. This makes it possible to move the robot arm back within the safety limits. In Recovery mode, the movement of the robot arm is restricted by a fixed limit that you cannot customize.

High Speed Manual Mode When this mode is enabled, you can temporarily exceed the default speed limit of the tool and the elbow. The robot performs a Safeguard Stop in Manual mode, if a Three-Position Enabling Device is configured, and either released (not pressed) or it is fully compressed.

Switching between Automatic mode to Manual mode requires the Three-Position Enabling Device to be fully released and pressed again to allow the robot to move. When using High Speed Manual Mode, use safety joint limits or safety planes to restrict the robot's moving space.



NOTICE

After five minutes of inactivity the speed limit resets to the default.

To enable High Speed Manual

1. Tap **Application** and select **Safety**.
2. Access the **Three Position** options.
3. On the page, slide the button **Allow manual high speed**.

**Mode switching**

Operational mode	Manual	Automatic
Move robot with +/- on Move Tab	x	
Freedrive	x	
Execute Programs	Reduced speed*	x
Edit & save program	x	

*If a Three-Position Enabling device is configured, the robot operates at Manual Reduced Speed unless High Speed Manual Mode is enabled.

**WARNING**

- Any suspended safeguards must be returned to full functionality before selecting Automatic Mode.
- Wherever possible, Manual Mode shall only be used with all persons located outside the safeguarded space.
- If an external mode selector is used, it must be placed outside the safeguarded space.
- No-one is to enter, or be within, the safeguarded space in Automatic Mode, unless safeguarding is used or the collaborative application is validated for power and force limiting (PFL).

Three-Position Enabling Device

When a Three-Position Enabling Device is used and the robot is in Manual Mode, movement requires pressing the Three-Position Enabling Device to the center-on position. The Three-Position Enabling Device has no effect in Automatic Mode.

**NOTICE**

- Some UR robot sizes might not be equipped with a Three-Position Enabling Device. If the risk assessment requires the enabling device, a 3PE Teach Pendant must be used.

A 3PE Teach Pendant (3PE TP) is recommended for programming. If another person can be within the safeguarded space when in Manual Mode, an additional device can be integrated and configured for the additional person's use.

Switching Modes

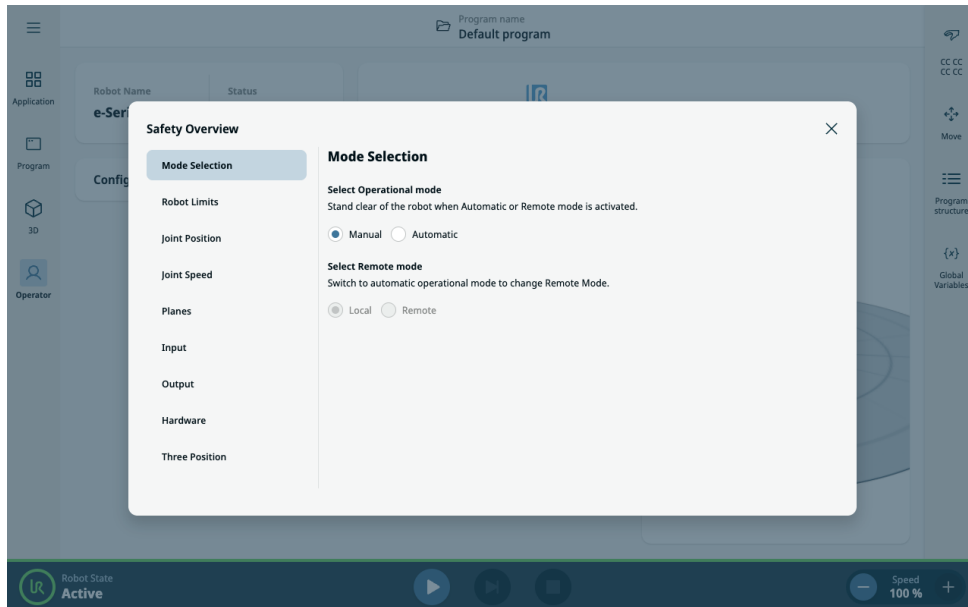
To switch between modes, in the Right Header, select the profile icon to display the Mode Selection.

- Automatic indicates the operational mode of the robot is set to Automatic.
- Manual indicates the operational mode of the robot is set to Manual.

PolyScope X is automatically in Manual Mode when the Safety I/O configuration with Three-Position Enabling Device is enabled.

Select Remote Mode

It is only possible to change the remote mode, when you have changed the operational mode to "Automatic".
If you change remote mode from "remote" to "local" the operational mode will go back to "manual".



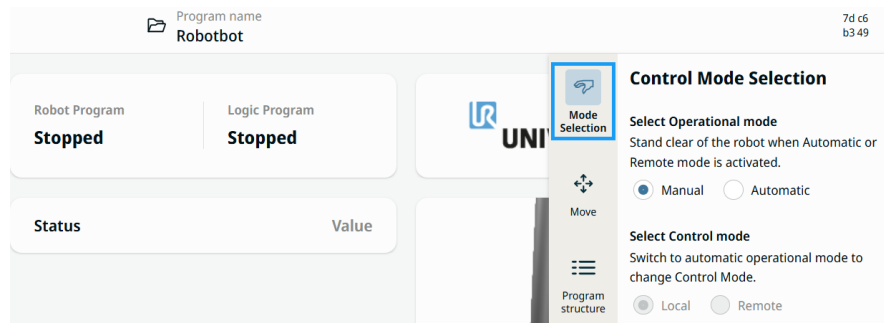
6.2.1. Manual Operational Mode

Description In **Manual Mode**, you can create, modify, and save robot programs, applications, and installations. This has four properties:

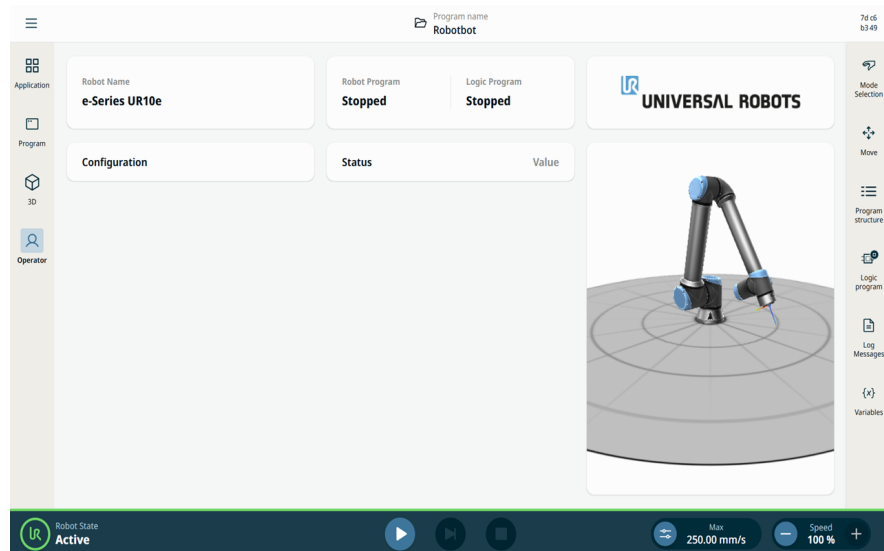
- A 3-position enabling (3PE) device must be held in its center position.
- Modify and save programs.
- Reduced speed control to 250 mm/s.
- Manual high speed is enabled by adjusting the **Manual High Speed** slider in the footer. A confirming action is required before higher speeds can be achieved. If the enabling device has been released for five minutes, manual high speed must be reenabled before higher speeds than 250 mm/s can be obtained.

To use Manual mode

1. Follow the steps of “To access Mode Selection” in Operational Mode. See [Operational Mode](#).



Manual mode is the default setting. The main screen contains all the functionalities of the PolyScope X interface.



6.2.2. Automatic Operational Mode

Description In **Automatic Mode**, the robot can only execute a program of predefined tasks. You cannot modify or save programs and installations. This has two properties:

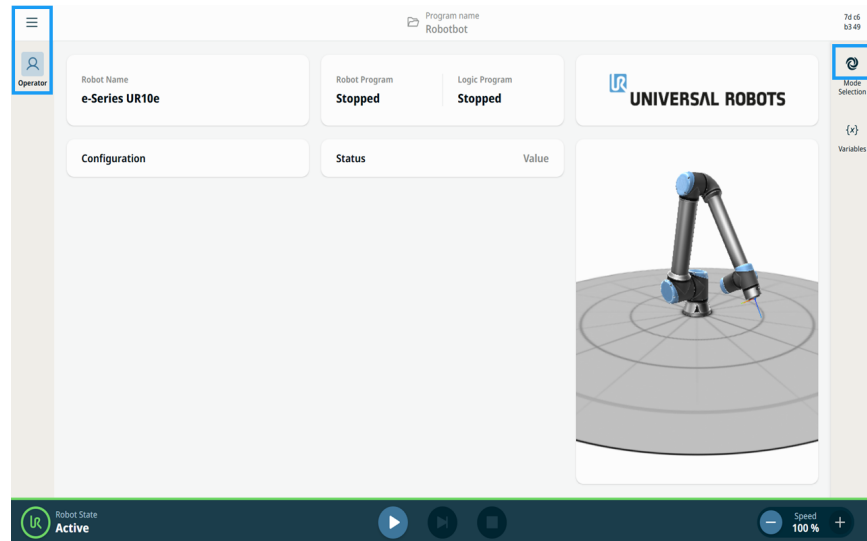
- Automatic program execution is allowed
- Motions at full speed (complying with safety limits)

To use Automatic mode

1. Follow the steps of “Switching Modes” in Operational Mode. See [Operational Mode](#).
2. Tick the **Automatic** button in the Select Operational mode, enter the operational mode password to unlock, and Confirm.

While in automatic mode, the PolyScope X interface is in the operator screen and the tabs in main navigation and sidebar are reduced. Additionally, the automatic

mode icon  is displayed above the Safety checksum .



3. You can tap the **Play button** in the footer or the robot control bar.

6.2.3. Remote Control

Description Remote Control allows you to control the robots via external sources, such as controller sockets, IOs, and the dashboard server. The other control is via the teach pendant (TP), called local control. Remote control can be used to send simple commands to PolyScope X such as starting or loading programs, as well as sending UR Script commands directly to the controller.



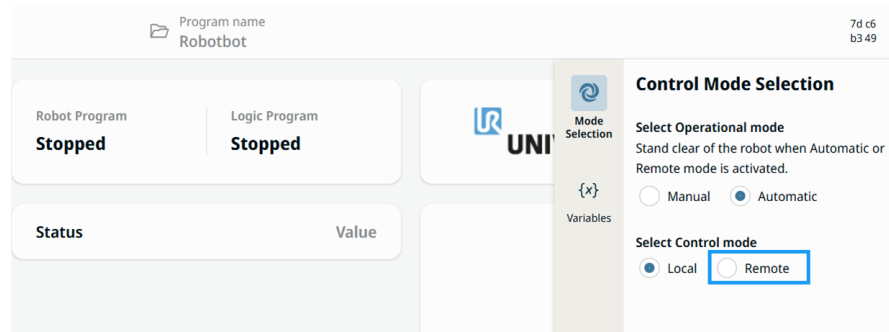
CAUTION

To ensure safe usage, the robot can either be in Remote Control mode or Local Control mode. Local Control mode will ensure that commands sent to the controller from an external source will be rejected while the robot is controlled in person.



To enable Remote Control

1. Follow the steps to switching into Automatic operational mode. See [Automatic Operational Mode](#).
2. Tap the Safety Overview/Checksum icon. The Safety Overview screen appears.



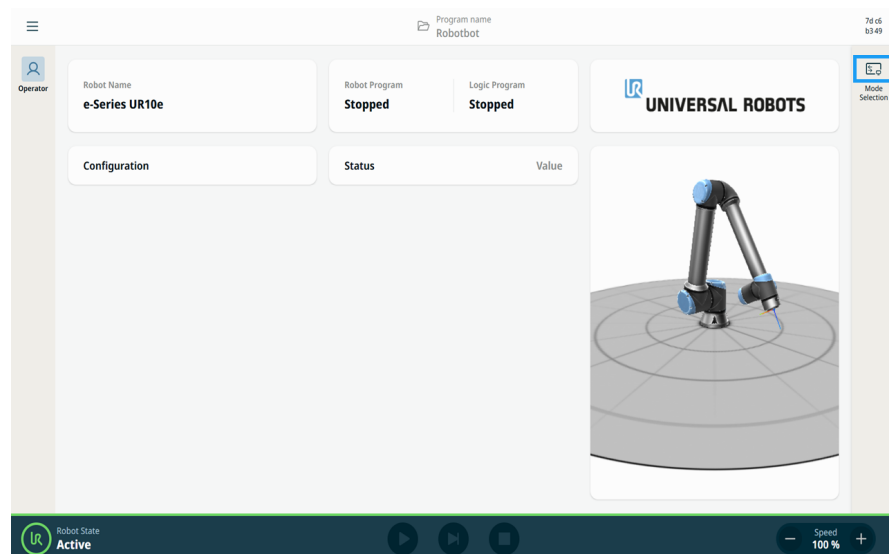
3. In the **Select Remote mode**, tick the Remote button.



NOTICE

Remote Mode is only active when the application is in Automatic mode.

4. The **Remote Mode icon**  is displayed above the Safety checksum to clearly indicate the system is under remote supervision. All editing and control actions are disabled, and only the Operator Screen remains accessible in view-only mode.



NOTICE

- Although Remote Control limits your actions in PolyScope, you can still monitor robot state.
- When a robot system is powered off in Remote Control, it starts up in Remote Control.

Local Control does not allow	• Power on and brake release sent to the robot over network • Receiving and executing robot programs and installation sent to the robot over network • Autostart of programs at boot, controlled from digital inputs • Auto brake release at boot, controlled from digital inputs • Start of programs, controlled from digital inputs
Remote Control does not allow	• Moving the robot from Move Tab • Starting from Teach Pendant • Load programs and installations from the Teach Pendant • Freedrive

6.3. Starting and Stopping the Robot

Description	You initialize the robot arm from the Robot State button after confirming the payload and clearing the arm of fixtures.
--------------------	--

6.3.1. Operator Screen

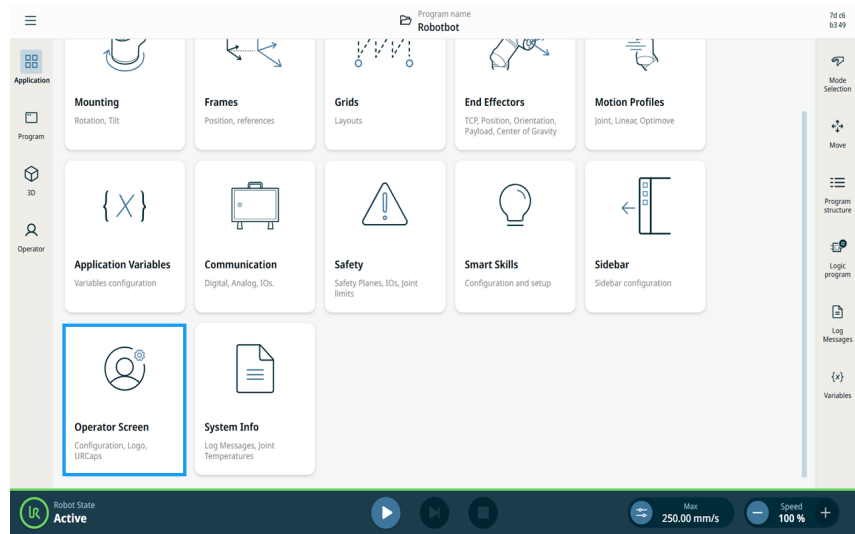
Description	The Operator screen shows information of the robot name, robot status, configuration, status, and a smaller 3D view of the robot arm. The operator screen gives you the following benefits: • Make changes without support or the least complex support in multiple workcells. • Avoid accidental program modification. • Manage one flexible program instead of multiple separate programs across different processes.
--------------------	--

6.3.2. Operator Screen Application

Description	The Operator Screen application enables you to make changes in the contents of the Operator, located in the left toolbar. See Operator Screen .
--------------------	--

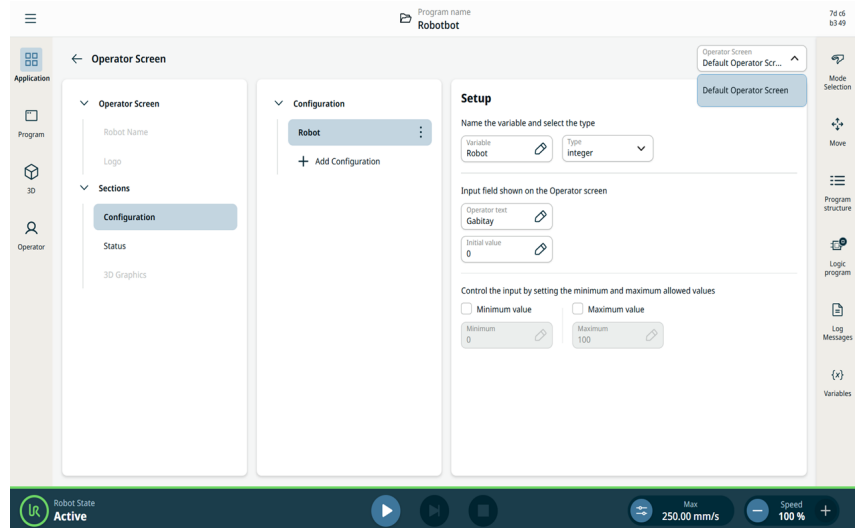
Using the Operator Screen application functionality

1. Go to the application nodes screen. See [9.3 Application Tab on page 279](#).
2. Tap the **Operator Screen** icon.



You see the Operator Screen in the main screen, which is divided into two panels and the Operator Screen field.

The left panel contains the **Operator Screen** and **Sections** tree. The center panel contains the **Configuration** tree. The Operator Screen field contains only **Default Operator Screen** option.



6.3.3. Freedrive

Description

Freedrive lets you manually pull the robot arm into position.

On most robot sizes, enable Freedrive by pressing the **Freedrive** button on the Teach Pendant. Additional ways to enable and use Freedrive are described in the following sections.

In Freedrive, the robot arm joints move with little resistance because the brakes are released. Resistance increases as the robot arm approaches a predefined limit or plane, so pulling the robot into position feels heavier near those boundaries.



WARNING

Injury to personnel can occur due to unexpected motion.

- Verify the configured payload matches the payload in use.
- Verify the correct payload is securely attached to the tool flange.

Enabling Freedrive

You can enable Freedrive in the following ways:

- Use the 3PE Teach Pendant
- Use Freedrive on robot
- Use I/O actions



NOTICE

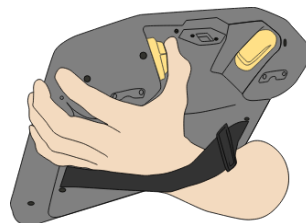
Enabling Freedrive while you are moving the robot arm can cause drift and faults.

- Do not enable Freedrive while you are pushing or touching the robot.

3PE Teach Pendant

Use the 3PE TP button to put the robot arm in Freedrive:

1. Quickly light-press, release, then light-press again and hold the 3PE button in that position.



Now you can pull the robot arm into a desired position, while the light-press is maintained.



Freedrive on robot

Use **Freedrive on robot** to move the robot arm:

1. Press and hold the switch configured for **Freedrive on robot**.
2. When the **Freedrive** panel appears in PolyScope, select the movement type for the robot arm joints, or use the list of axes to customize the movement type.
3. If needed, define the feature type by selecting an option from the **Feature** dropdown list.

The robot arm can stop moving if it approaches a singularity. Tap **All axes are free** in the **Freedrive** panel to resume movement.
4. Move the robot arm as desired.

Backdrive

During initialization of the robot arm, minor vibrations can occur when the robot brakes release. In situations such as when the robot is close to a collision, these vibrations are undesirable. Use Backdrive to force specific joints into position without releasing all brakes in the robot arm.

Heavy Freedrive

If the robot arm is powered on and the brakes are released, the joints can be moved by force. The force required to initiate motion depends on the Force Limits specified in the safety settings.

Freedrive Input

Description

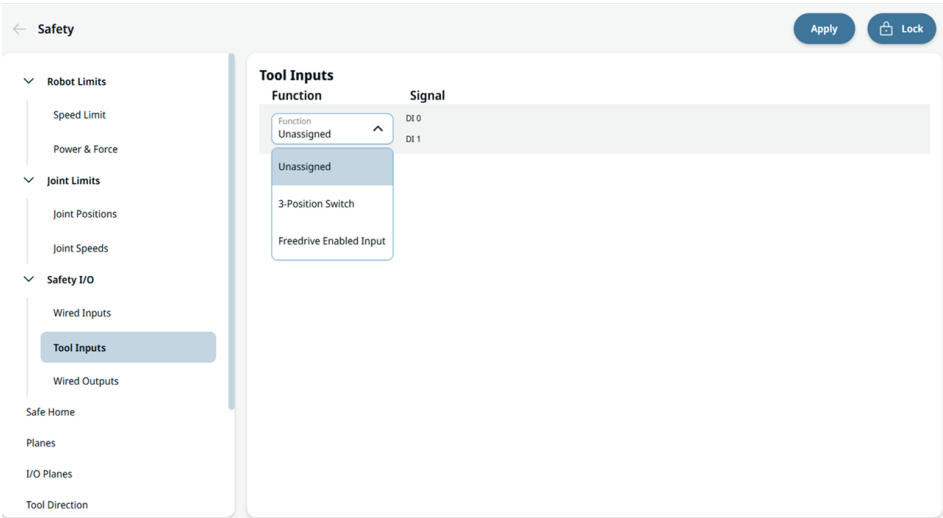
Freedrive input is recommended to be used with 3PE TP and/or 3 Position Enabling Device input.

When Freedrive **INPUT** is **High**, the robot will only enter Freedrive if the following conditions are satisfied:

- 3PE TP button is not pressed
- 3 Position Enabling Device **INPUT** either not configured or not pressed (**INPUT Low**)

Freedrive Enabled Input

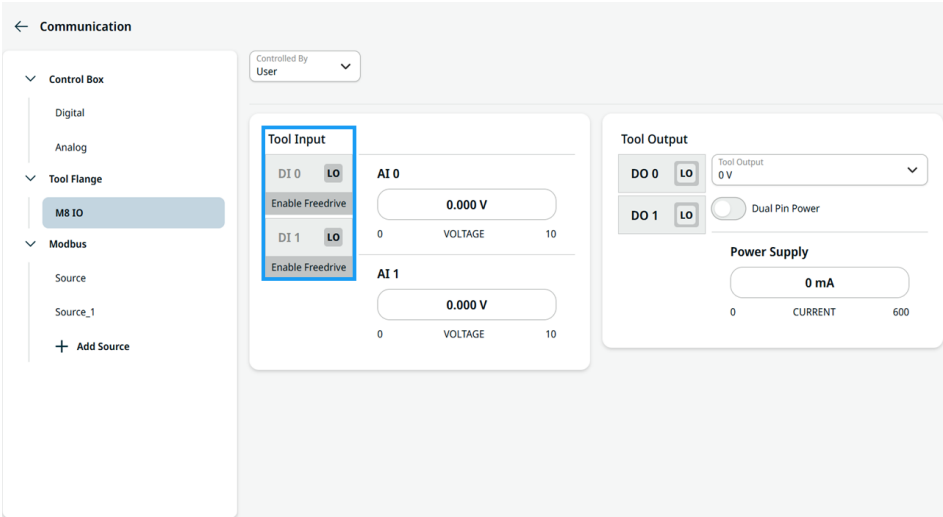
On UR8 Long robots, you can assign the **Freedrive Enabled Input** and **3-Position Switch** safety functions to the **Tool Inputs** on the application **Safety** screen.



NOTICE

This functionality is currently only available on the UR8 Long robots. On all other robot types, the new Tool Inputs section is not shown on the Safety I/O screen.

When the Tool inputs are assigned a Safety Function, it is not possible to rename a Tool input or assign a Tool input an action on the **Communication > Tool IO** page.

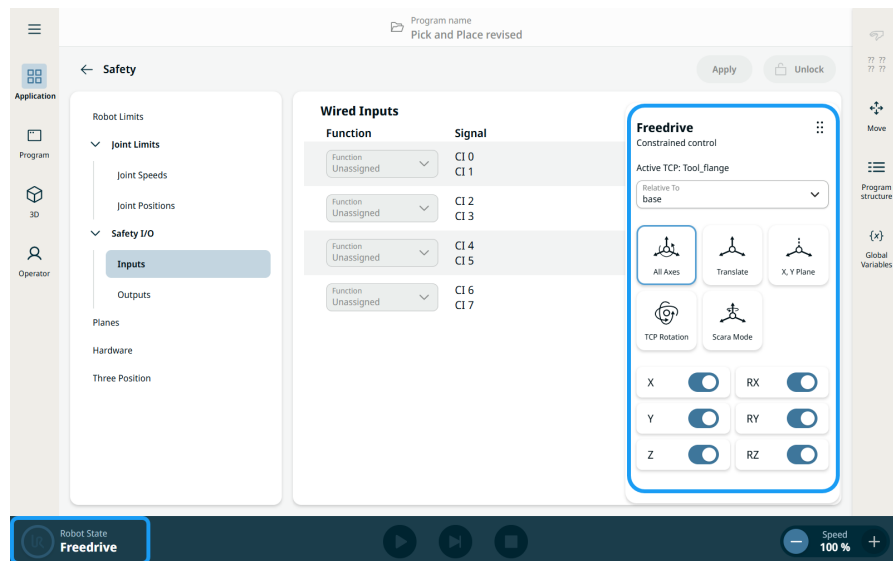


6.3.4. Constrained Freedrive

Description **Constrained Freedrive** mode allows a robot to move freely while being compliant in specific axes. This mode is useful for manually guiding the robot without following a trajectory, but it requires careful management to avoid unintended movements or protective stops due to high acceleration and deceleration.
When the robot arm is in Freedrive, a panel appears in PolyScope X.

To access the Freedrive Panel

1. Use the Teach Pendant with 3-Position Enabling Device button to put the robot arm in [6.3.3 Freedrive on page 61](#). On the left-hand corner of the footer, the Robot State is Freedrive. The Freedrive panel appears in the main screen.

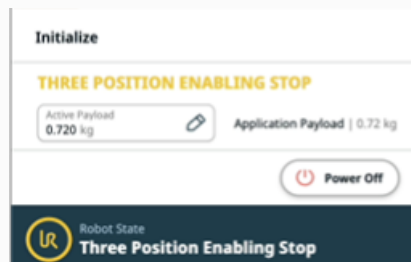


2. In the Freedrive panel, there are five different options to choose from. Tap the icon of your selected Freedrive movement.
3. Tap the sliders to lock or unlock movement in X, Y, Z or RX, RY, RZ.
4. Move the robot arm manually.

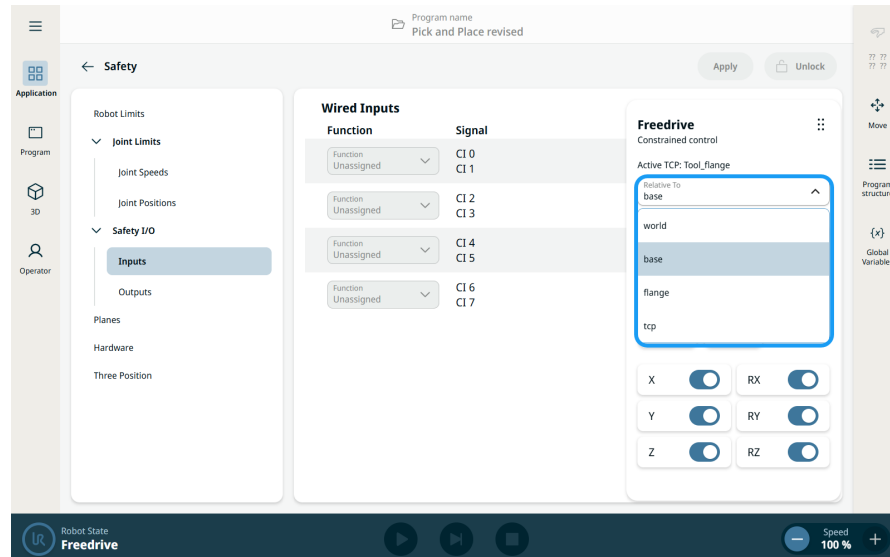


NOTICE

Always press the 3PE button, as indicated, to avoid disruption of Freedrive. Accidental release of the 3PE button changes the Robot State to Three Position Enabling Stop.



- The **Relative To** drop-down in the Freedrive panel allows users to choose what rotations and axes are relative to during constrained Freedrive. When **Base** is selected, all rotations and axes are relative to the base. When **Tool** is selected, all rotations and axes are relative to the tool.



Freedrive Panel icons

You can lock one or more of the axes allowing the TCP to move in a particular direction, as defined in the table below.

	Movement is allowed through all axes.
	Movement is allowed through all axes, without rotation.
	Movement is only allowed through the X-axis and Y-axis.
	Movement is allowed through all axes, in a spherical motion, around the TCP.
	Movement is allowed through all axes and only one rotation, RZ.

**CAUTION**

Moving the robot arm in some axes when a tool is attached, can present a pinch point.

- Use caution when moving the robot arm in any axis.

**Five
Constrained
Freedrive
presets**

There are five presets that the user can use to constrain movement while in freedrive.

Tap the preset icon for your preferred Freedrive constrained movement.

Alternatively, users can toggle the sliders for X, Y, Z and RX, RY, RZ to lock or unlock any combination of axes or rotations. When no preset is matched, the preset icon will not be shown as active.

Preset Icon and Name	X	Y	Z	RX	RY	RZ
 All Axes	Free	Free	Free	Free	Free	Free
 Translate	Free	Free	Free	Locked	Locked	Locked
 X, Y Plane	Free	Free	Locked	Locked	Locked	Locked
 TCP Rotation	Locked	Locked	Locked	Free	Free	Free
 Scara Mode	Free	Free	Free	Locked	Locked	Free

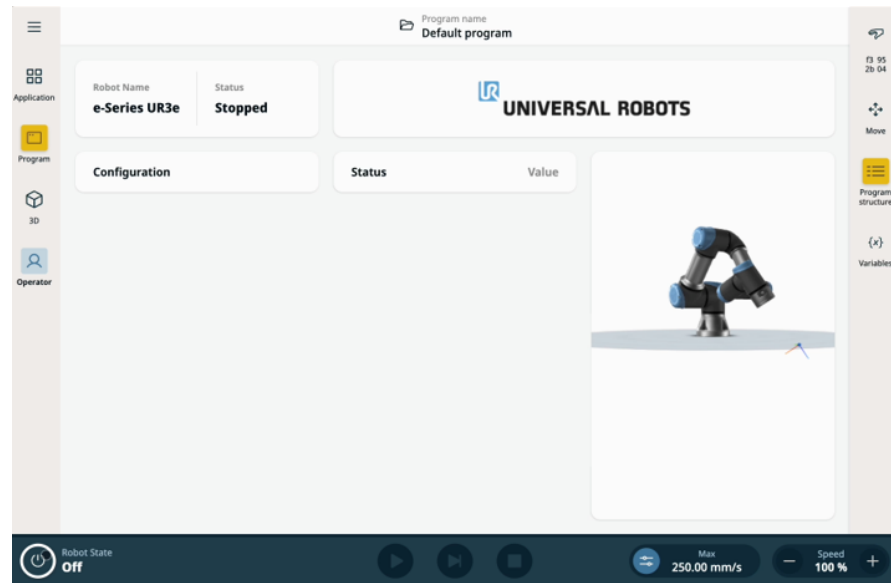
6.3.5. Backdrive

Description

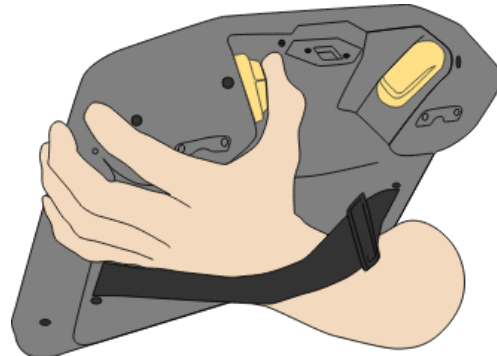
Backdrive is a Manual mode used to move specific joints to a desired position without releasing all brakes in the robot arm. This can be necessary when the robot arm is close to a collision and the motions that accompany a full restart are not desired.

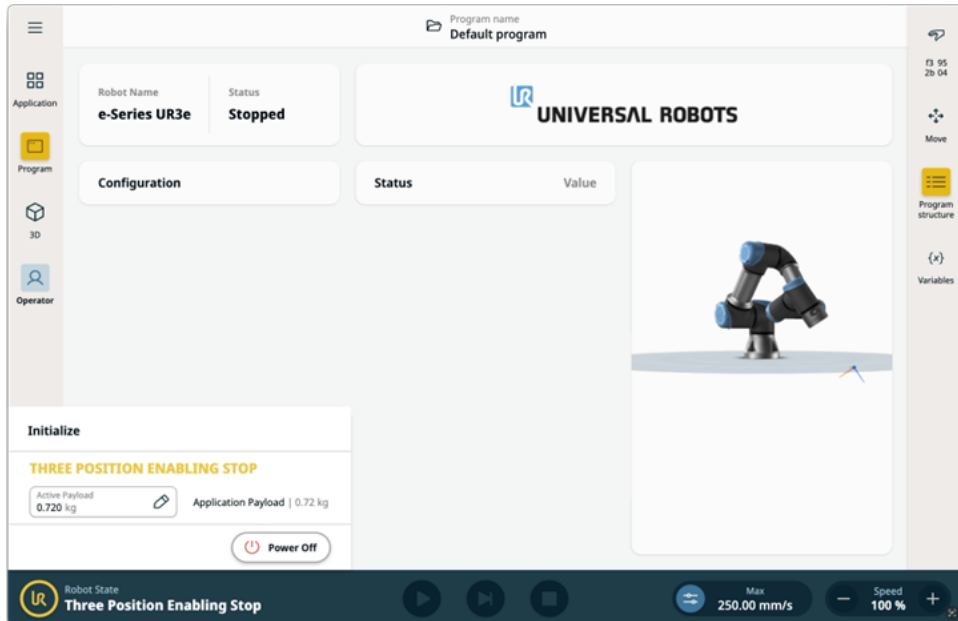
Enable Backdrive in 3PE Teach Pendant

1. Tap the round button in the lower-left corner of the footer to open the control panel and initialize the robot arm. The Robot State is **Off**.

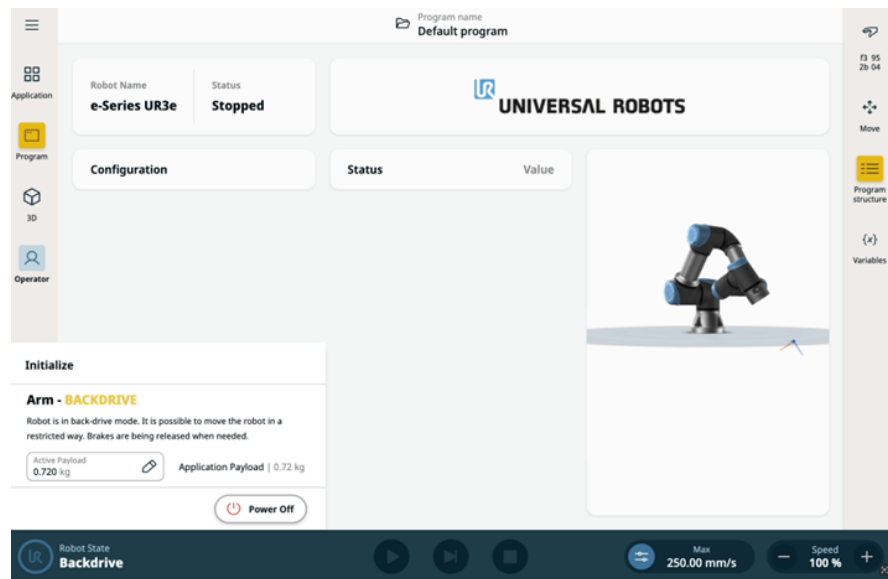


2. Tap **Power On**.
3. Double-tap and hold the physical 3PE switch on the back of the Teach Pendant. The Robot State is **Three Position Enabling Stop**.





4. Apply firm pressure to the joint you want to move. When you hear the brake release, move it in the desired direction. The Robot State is **Backdrive**.



NOTICE

In Backdrive mode, the robot is “heavy” to move.

MANDATORY ACTION

You must test Backdrive mode on all joints.

Safety settings

Verify that the robot safety settings comply with the installation risk assessment.

**Additional
safety inputs
and outputs
are still
functioning**

Check which safety inputs and outputs are active, and verify that they can be triggered via PolyScope or external devices.

6.4. Programming Workflows

Description

You create programs from the Program menu by adding command nodes from the Commands toolbox onto the color-coded program tree, structuring them with folders, functions, and reusable modules. The interface displays only valid nodes at each step to guide you through creating, loading, saving, and exporting programs.

6.4.1. Create Program

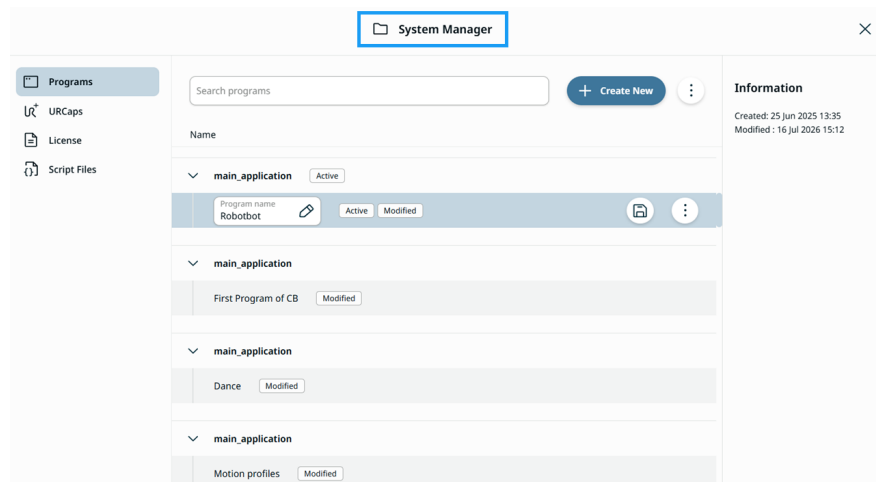
Description

The **Create** program command enables you to create and add programs for the robot.

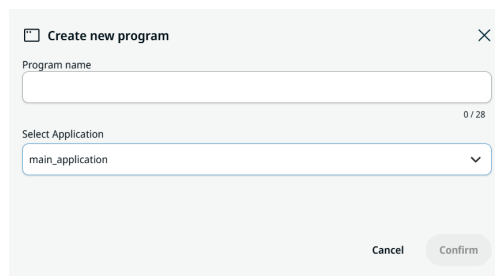


To open Create program menu

1. In the header, tap the **Program name** tab.
The **System Manager** screen appears.



2. Tap the **+ Create New** icon.
3. Tap **Create New Program**.



A new box appears with the heading **Create new program**.

5. Enter your program name in **Program name**.
6. In **Select Application**, select an existing Application for the program to use, or open the dropdown and choose **+ Create New**.
7. If you choose **Create New**, enter a name for the new Application. You can optionally copy the currently active Application settings into the new Application.
8. Tap **Confirm**.

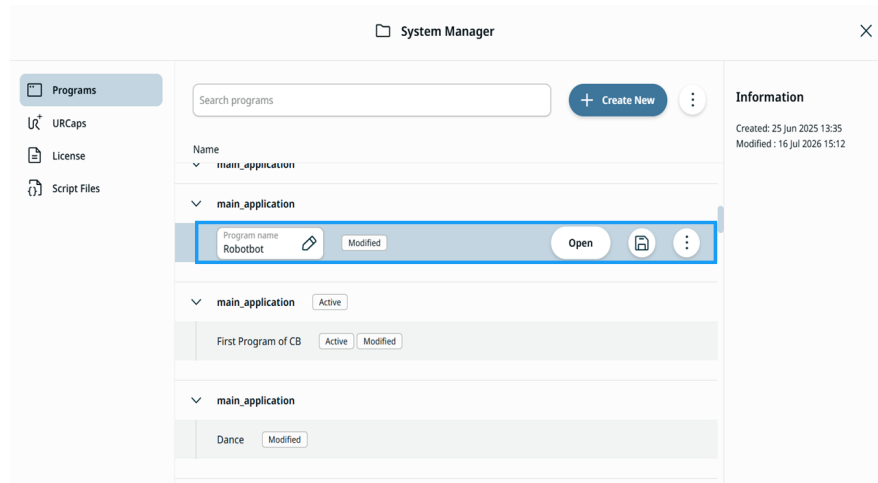
6.4.2. Modify Program

Description The **Modify** program allows you to make changes to the programs you created for the robot.

To access Modify program

1. In the header, tap **Program name**. The **System Manager** screen appears.

In the middle of the screen, expand the Application accordion if needed, then choose and tap the program you want to modify. Once you tap the chosen program, the tab color highlights to blue.



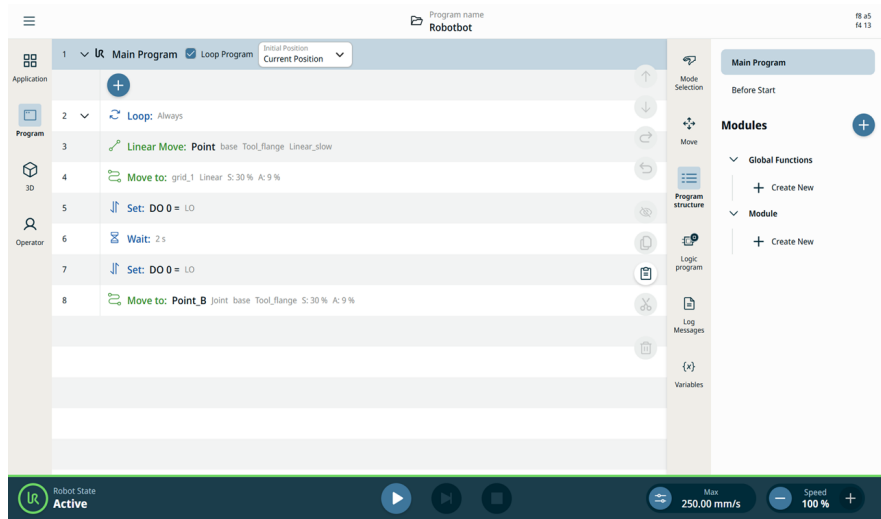
The tab contains the following buttons:

- **Program name**
- **Open**
- **Save icon**
- **Kebab icon** 

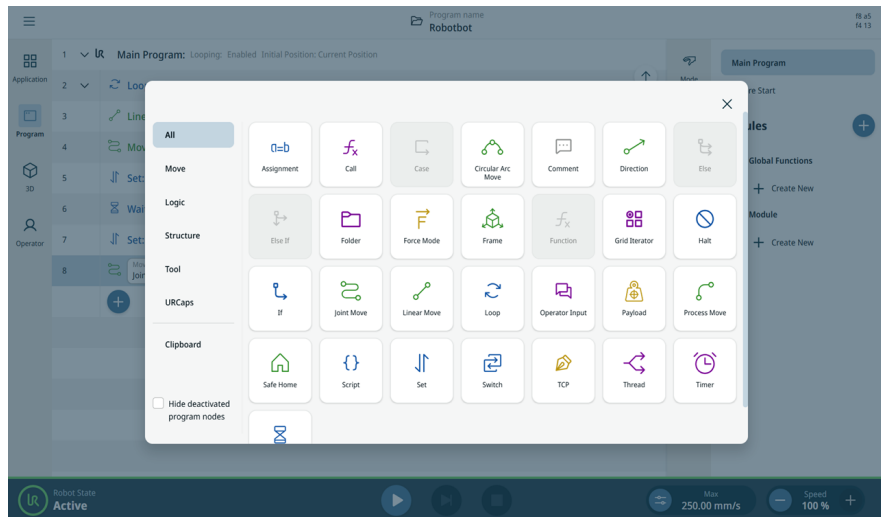
You can view the program's creation and modification dates on the right side of the screen.

2. Tap the **Program name** button to edit the name and **Confirm**.

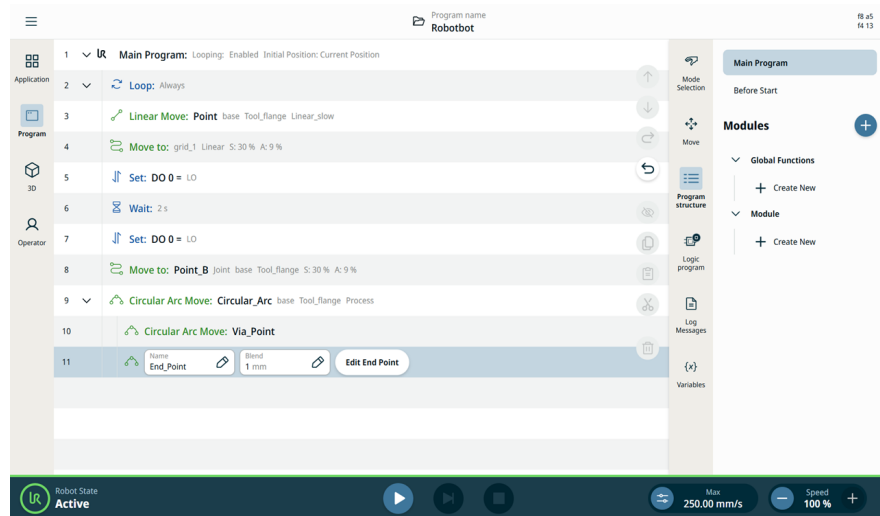
3. Tap the **Open** button. You are directed to the Main Program screen.



4. In the main program screen, tap the **add icon**. A new screen with the **Commands** and **Clipboard** toolboxes. In the Commands toolbox, you can choose which command to use to modify a program.



- After choosing your commands to program a robot, a corresponding list is shown in the Main Program screen.



- Tap again the **Program name** tab found at the topmost.
- Tap the **save icon**, and you have two options:
 - Save**
 - Discard changes**
- Tap the **kebab icon**, and you have these options:
 - Duplicate** – see [Duplicate Program](#)
 - Change Application** – see [Change Application](#)
 - Delete**
 - Export**

6.4.3. Load Program without Safety Password

Description This feature lets you load programs without repeatedly entering the [Safety Password](#), while ensuring that only approved safety configurations can be applied. This improves usability without compromising safety.



User benefit

These are the benefits of this feature:

- You can load programs without entering the safety password.
- You only need the safety password when you modify or apply unapproved safety configurations.
- New programs use the default safety checksum `CCCC` `CCCC`. You must enter the safety password to create a new safety configuration.
- You can create programs by copying settings from a loaded program or duplicating an existing program without entering the safety password.



NOTICE

As in PolyScope 5, approved safety configurations are stored in each program or application.

Safety checksum (CCCC)

PolyScope X displays the safety checksum for the currently loaded program. `CCCC` `CCCC` is the default checksum for the default safety configuration. In PolyScope X, you see the safety checksum in the [Sidebar](#).

6.4.4. Smart Skills

Description

Smart skills are quick-access robot actions that you can start and stop to achieve a smoother and faster programming experience. Through these actions, you can manipulate the robot, any external systems, or end effectors. These are the Smart Skills to choose:

Activated

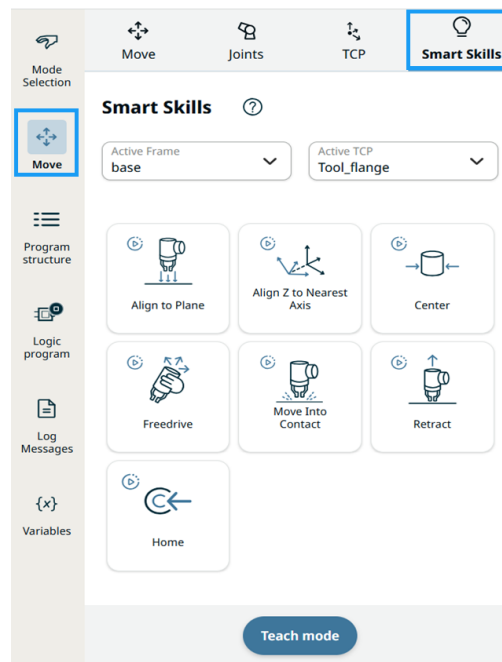
- [Align to Plane on page 76](#)
- [Align Z to Nearest Axis on page 77](#)
- [Center on page 77](#)
- [Freedrive on page 78](#)
- [Move Into Contact on page 79](#)
- [Retract on page 79](#)
- [Home on page 78](#)

Deactivated

- Put into Box
- Custom

- To access Smart Skills**
- There are two ways to access Smart Skills:
- via **Move** section in sidebar
 - via **Application** tab in the main navigation

- Access via Move section**
- Via Move section** is where you press and hold the button on the teach pendant while the smart skill is executing. The corresponding robot movement is seen in the main screen.
1. Tap the Move icon in the sidebar.
 2. Choose the Smart Skills icon at the top right.



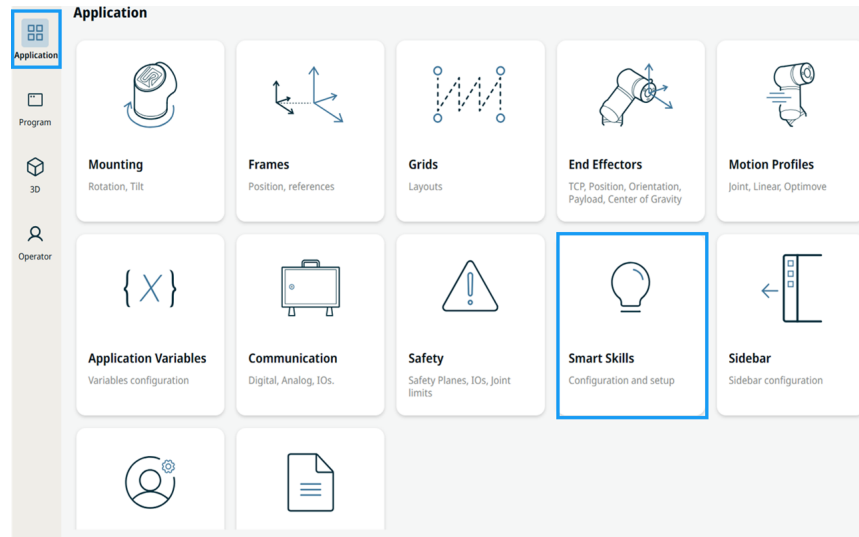
3. In the **Relative To** field, you can choose from the four features or frames or default coordinate systems:
 - **World**. This frame is the same as the base when the robot is mounted in a fixed location. The world frame is always fixed even when the robot is mounted in a moveable location, and you can update the base frame as the robot moves.
 - **Base**. The robot's base element, referring to the frame of the base, which is located in the center of base bottom. If your robot is mounted in a fixed location, this frame will never need to change, and other fixed locations can be defined relative to it.
 - **Flange**. It's placed on the robot's tool flange and can only be moved by moving the robot.
 - **TCP**. The position of the active TCP and updates as the robot moves when jogged or while a program is running.

See [Frames](#) to know more when to use the pertinent default coordinate system.
4. In the **Robot** field, you choose either **Active_TCP** or **Tool_flange**. If Active_TCP is selected, the name of the current active TCP is displayed.

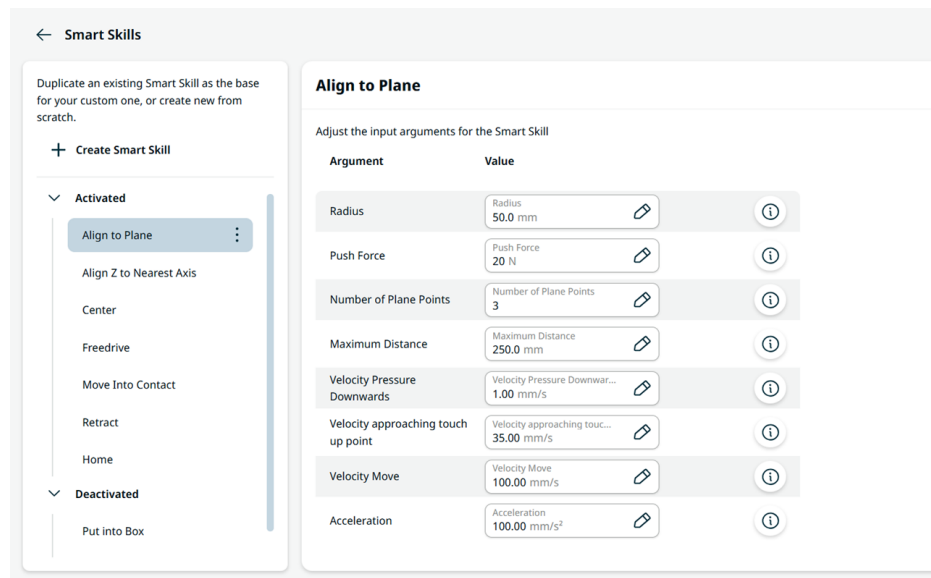
Access via
Application
tab

Via **Application tab** is where you can configure and set up the input arguments.

1. Tap the Application tab in the main navigation. See [Smart Skills Application](#).
Choose the Smart Skills application menu.



2. You are directed to the Smart Skills application screen.



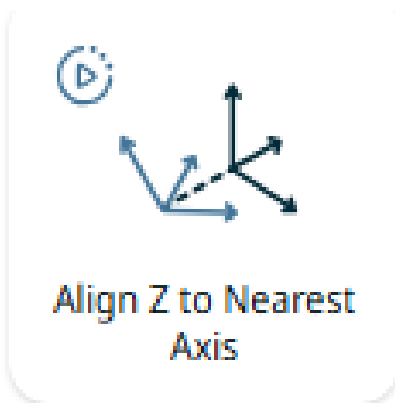
Align to Plane

Description **Align to Plane** assists you in orienting the TCP to a surface. While executing this smart skill, the robot will touch up 3 (default value) points on the surface from which it computes the plane's orientation. The contact points are laid out in a circular pattern which has a default diameter of 2.5 mm. This smart skill is especially useful for creating new frames in the [Frames](#) application.



Align Z to Nearest Axis

Description **Align Z to Nearest Axis** finds the nearest axis of a frame and align the TCP z-axis to that. This functionality is used to align the robot's tool to already existing frames. You can change which frame the robot aligns to by changing the frame in the **Relative To** field.



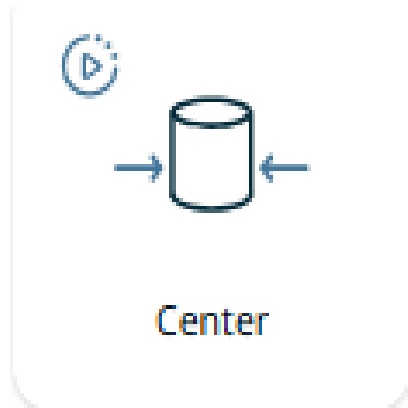
Example If you have a gripper and want to pick up an item, use this smart skill in order to have the robot actually move around in a predefined pattern in order to figure out where is the item positioned and what is the center of that. In this way, you can then center the robot.

Center



Description Center is a dual functionality smart skill used to:

- Center the tool to a part in a fixture
- Center an object to a whole, given that the robot is already holding the object in its tool



Freedrive

Description Freedrive smart skill enables you to set the robot program in a freedrive position.



Freedrive allows the robot arm to be manually pulled into desired positions and/or poses.

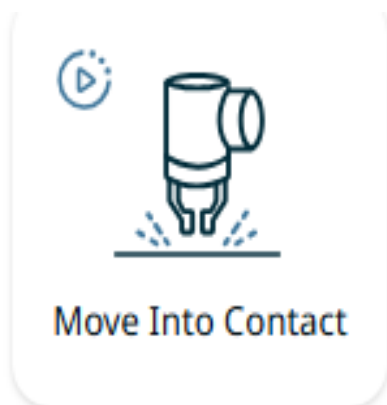
Home

Description The **Home** Smart Skill moves the robot to a predefined home position. Through this, you can define a home position to easily go to a predefined posture from the Smart Skills tab. The upright zero position will be the default but can be changed in the [Smart Skills Application](#) tab, and it is available as a waypoint.



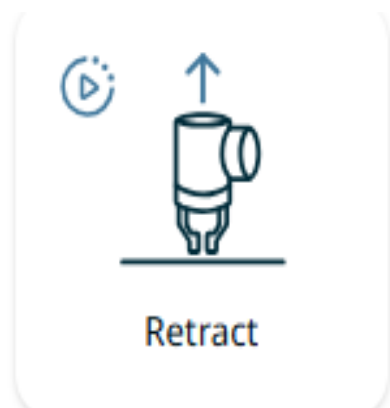
Move Into Contact

Description The **Move Into Contact** smart skill enables you to move the robot in the TCP z-direction until it reaches a contact point. After reaching this contact point, the robot will retract a given distance. By default, this is set to 1 mm retraction distance.



Retract

Description **Retract** smart skill moves the robot in the negative TCP z-direction.



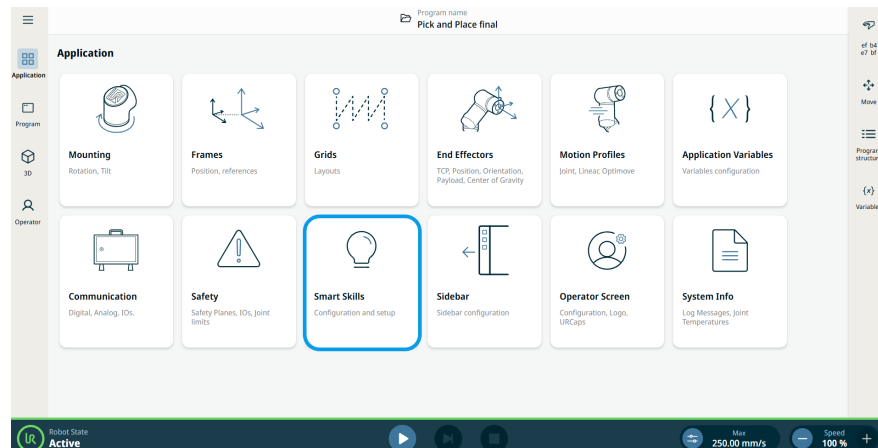
6.4.5. Smart Skills Application

Description All Smart Skills are parametrized, which enables you to configure and set up the Smart Skills in the Application tab. The Smart Skills interface gives you the following benefits:

- A clear and accessible call-to-action button for creating new Smart Skills.
- Eliminates the need to duplicate existing skills as a starting point.
- Simplifies and accelerates the creation process.
- Improves clarity and usability within the Smart Skills configuration interface.

Using the Smart Skills application functionality

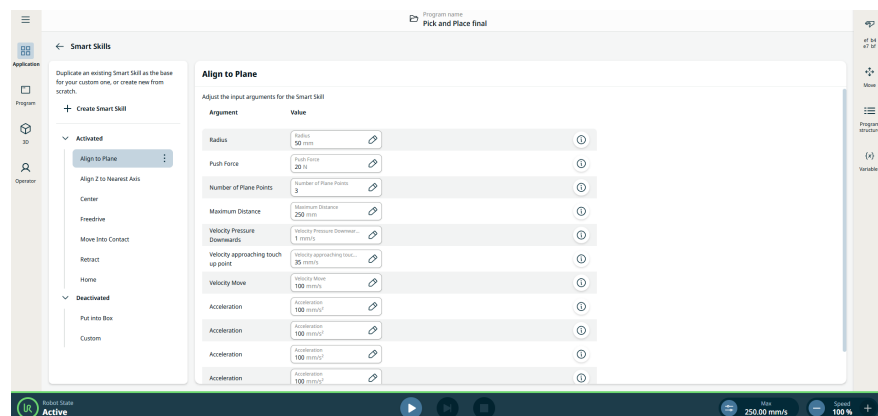
1. Go to the application nodes screen. See [Application Tab](#).
2. Tap the **Smart Skills** icon.



A screen appears with two panels:

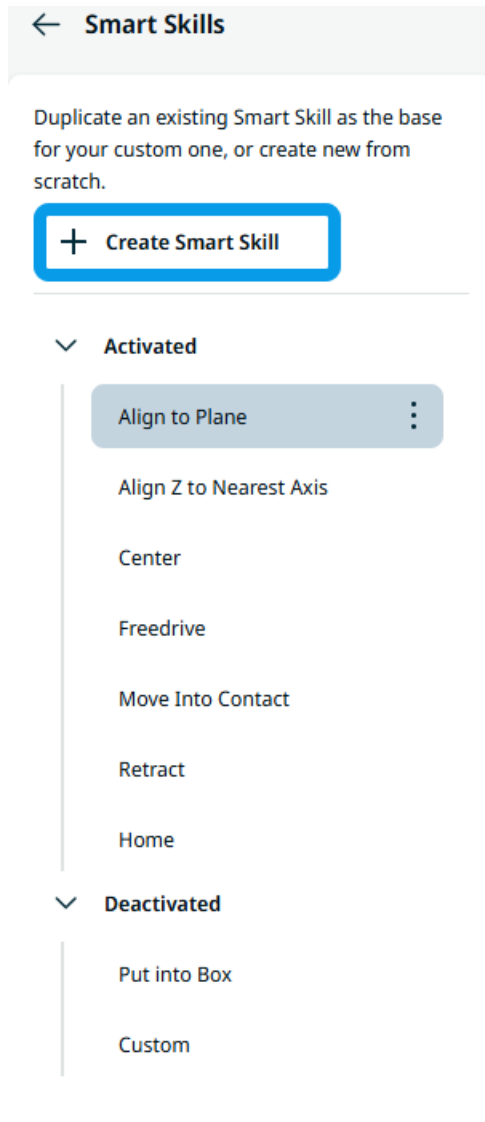
Left panel contains the list of smart skills. Tap the kebab icon of each smart skill to activate or deactivate it. When activated, that smart skill appears in the Smart Skills drawer in the multitasking screen and disappears when deactivated.

Right panel contains the editable fields to adjust the input arguments for each smart skill.





The process of creating custom Smart Skills is streamlined for ease of use. Instead of duplicating existing skills, tap **+ Create Smart Skill** button to start from scratch, found at the left panel before the list of Smart Skills. This makes it easier and more intuitive for users to build tailored Smart Skills, enhancing productivity and reducing setup time.



Smart Skill Input Argument

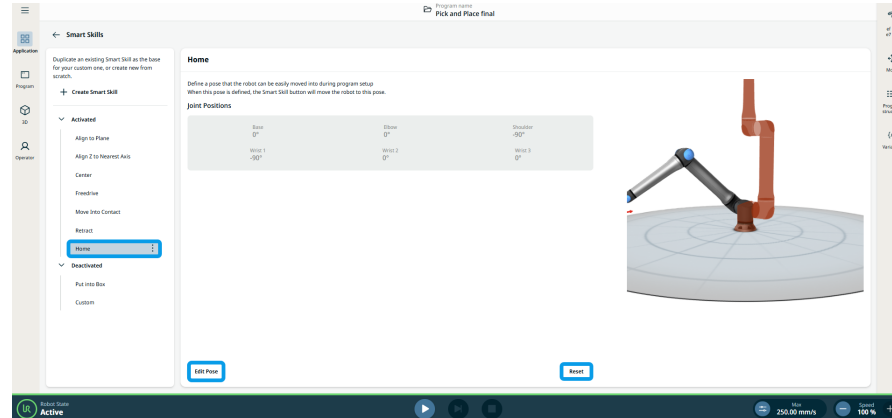
Each Smart Skill has a couple of parameters it uses when it executes. These can be changed to fit the application that you are running, whether or not it fits the default values.

These are the input arguments of the smart skills where you can change the value:

- **Align to Plane**
 - **Radius.** Radius of the circle within the plane that will be touched up.
 - **Push Force.** How hard the robot pushes against the plane.
 - **Number of Plane Points.** Number of points the robot uses to compute the plane.
 - **Maximum Distance.** The maximum distance the robot searches.
 - **Velocity Pressure Downwards.** Velocity when pressing downwards.
 - **Velocity Approaching Touch Up Point.** Velocity used when approaching the touch-up point.
 - **Velocity Move.** Velocity used in freespace.
 - **Acceleration.** Acceleration of the robot.
- **Align Z to Nearest Axis**
Not configurable
- **Center**
 - **Force.** Force the robot uses to determine if a contact has been achieved.
 - **Velocity.** Velocity in freespace.
 - **Acceleration.** Acceleration in freespace.
 - **Maximum Radius.** The maximum search radius.
 - **Number of Fingers.** The number of fingers that the gripper has.
- **Freedrive**
Not configurable
- **Move Into Contact**
 - **Force.** Force the robot uses to determine if a contact has been achieved.
 - **Velocity.** Velocity in freespace.
 - **Acceleration.** Acceleration in freespace.
 - **Maximum Distance.** The maximum distance the robot searches.
 - **Retract Distance.** Retract distance after a contact has been found.
- **Retract**
 - **Distance.** Retraction distance.
 - **Acceleration.** Acceleration used by the robot.
 - **Velocity.** Velocity used by the robot.

- Home

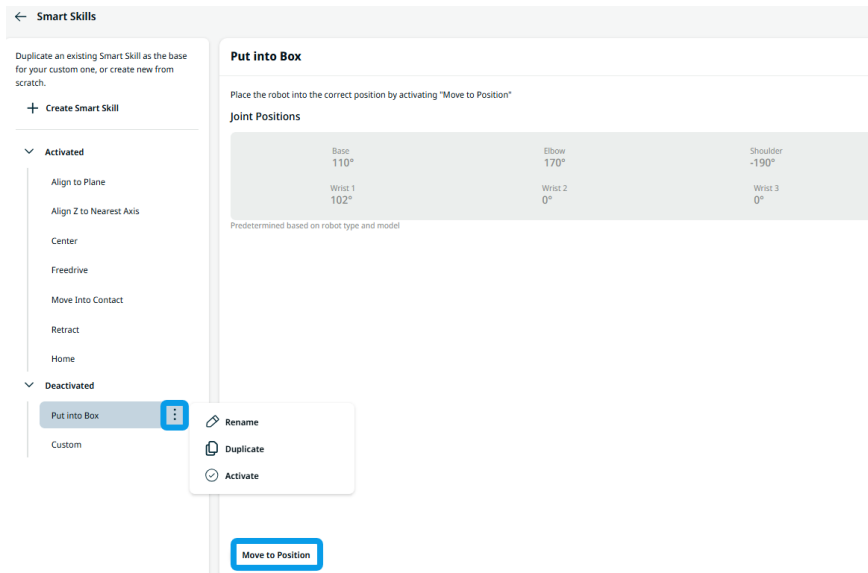
This enables you to define a posture to easily move the robot to its predefined home position. Tap **Edit Posture**, and you will be directed to the 3D viewer main screen. Tap **Reset**, and a dialogue box appears asking you to confirm or cancel. When the position is reset, the previous values will be deleted.



Deactivated Smart Skills by default

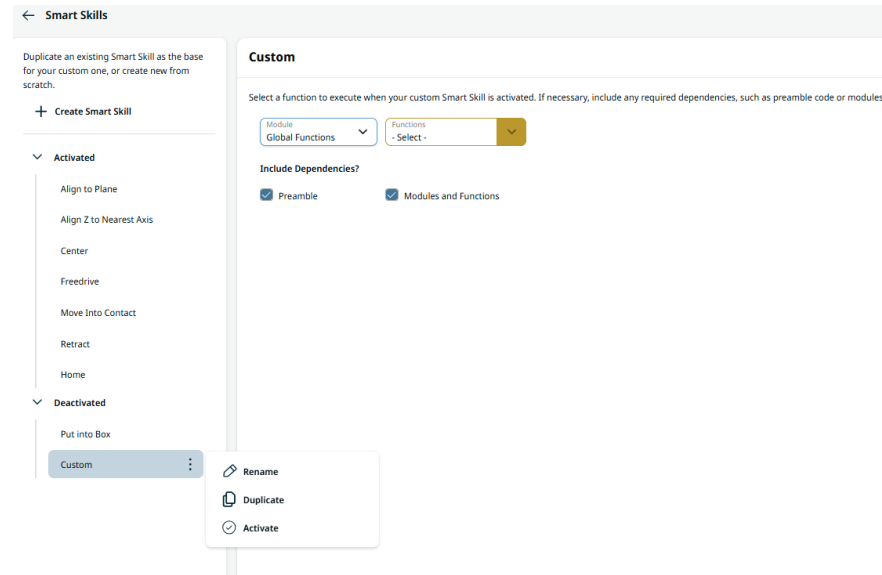
- Put Into Box

This smart skill enables you to put the robot in its default transportation position. Tap **Move to Position** button below to place the robot into the correct position. Its **Preset Joint Position** is deactivated as it is predetermined by the robot type and model.



- Custom

You can duplicate and rename this using the kebab icon. Custom is an easy way to execute a function of a module. It is then shown in the Smart Skills icon.



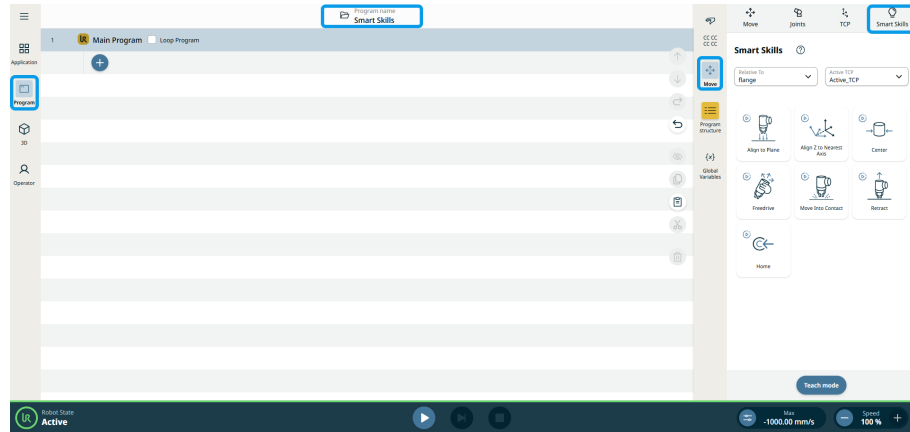
6.4.6. Teach Mode

Description Teach Mode is a new way to program the robot by demonstration through teaching the robot what to do with a combination of freedrive and Smart Skills. Every action that is done in Teach Mode will be recorded and converted to program nodes. To enable Teach Mode, navigate to the Smart Skills pane located in the Move drawer. Apply a Smart Skill or press the physical freedrive button to generate a program node in the tree. Additionally, you can undo the Smart Skill to restore the robot's previous state, for example, by returning it to its position before the action was performed. Read related topics [Program Menu](#) and [Smart Skills](#).



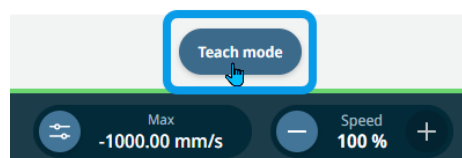
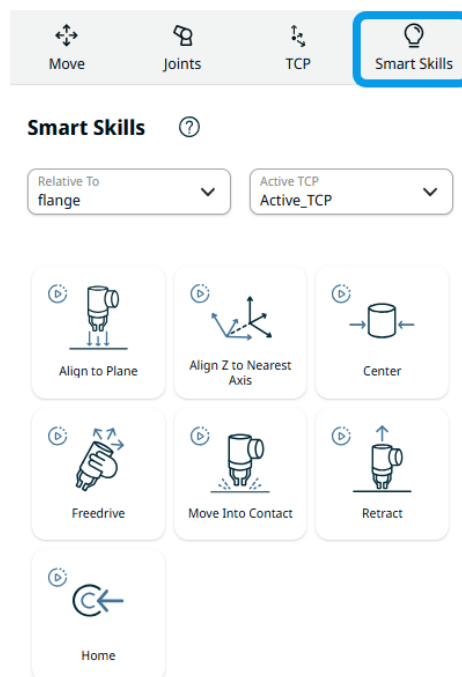
To access Teach Mode

1. Tap **Program** in the main navigation.
2. In the header, tap **Program Name** to use an existing program or create a new one where you want to apply Smart Skills program or freedrive.
3. Tap **Move** in the sidebar to expand the multitask screen, and you can see the **Smart Skills** pane in the top right part.

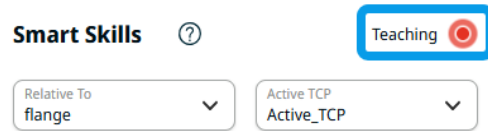


4. Tap Smart Skills , and you can see these components making up the Smart Skills screen:

- Relative to field
- Active TCP field
- Align to Plane
- Align Z to Nearest Axis
- Center
- Freedrive
- Move Into Contact
- Retract
- Home
- Teach Mode button

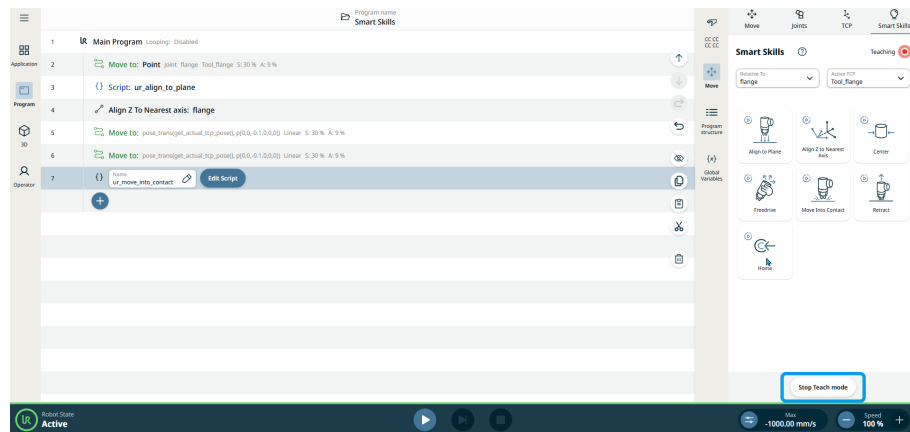


5. Tap the **Teach Mode** button, and you see a blinking red point indicating **Teaching**.



6. Tap a smart skill to start teaching, and tap it a second time to stop teaching. A configurable node is inserted into the program.
7. Continue with other Smart Skills.
8. Tap the **Stop Teach Mode** button when you are done.

This is how the program tree looks like when, for example, you use three smart skills.



6.4.7. Move Menu

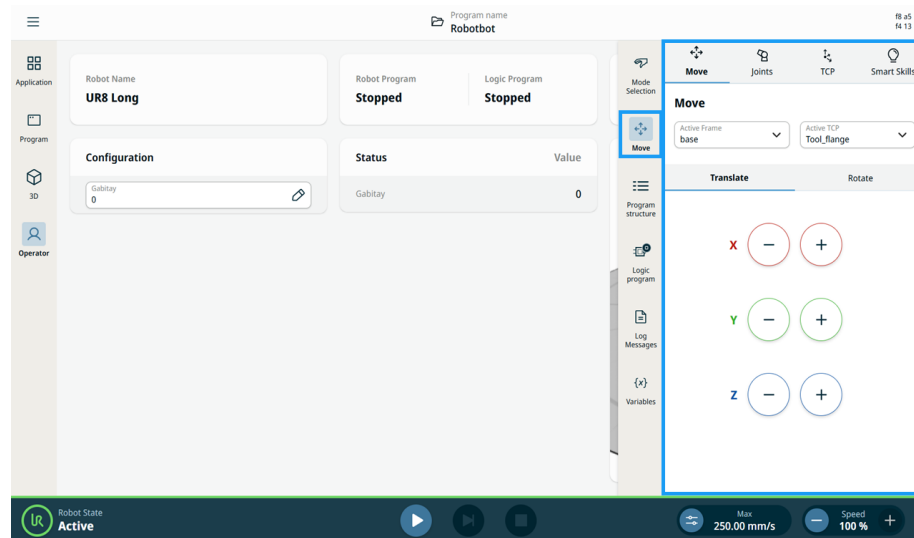
Description

The **Move** menu provides the functions to move the robot arm directly, either by translating and/or rotating the robot tool or by moving robot joints individually.

Using the Move menu

1. Tap the **Move** icon on the sidebar.

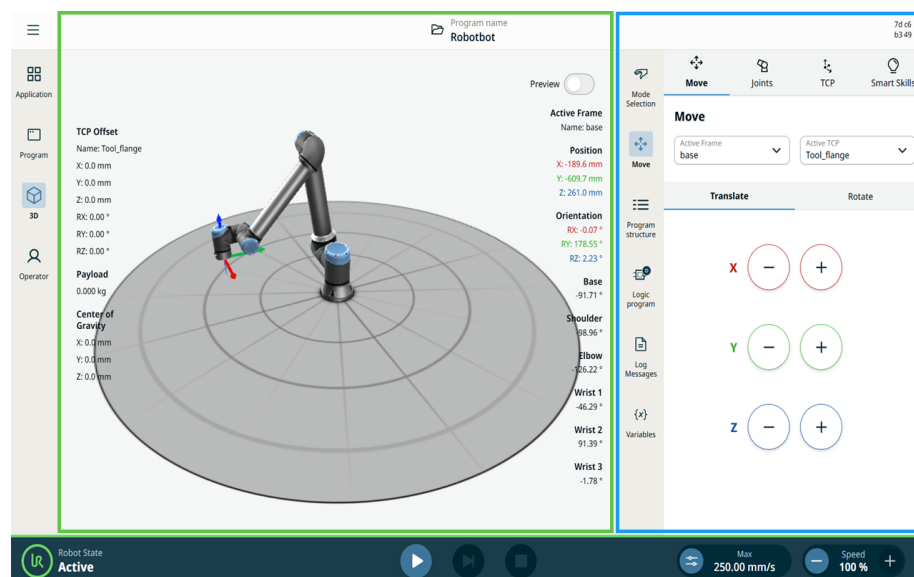
The multitask screen expands to the center. You can see the fields and tabs related to the move menu.



2. Tap the **3D** icon on the main navigation.

The main screen shows the robot arm in X, Y, Z coordinates with the measurement details of each robot arm parts, position, and orientation.

In the main screen, you can see the corresponding movement(s) of the robot with what you input in the multitask screen.



Move Tool

Translate

You use the Translate tab to move the robot arm linearly along a chosen axis (X, Y, Z) to change the position of the tool in space.



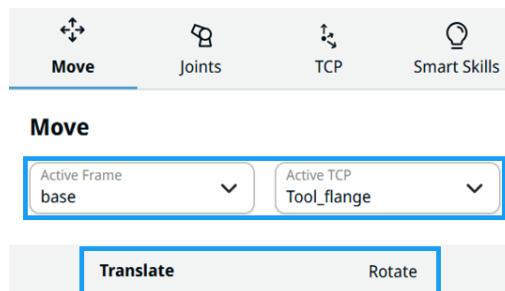
Rotate

You use the Rotate tab to change the orientation of the robot tool about a chosen axis (RX, RY, RZ), without changing its linear position.

Using the Move tool

The Move tool contains two fields and two tabs:

- Relative To
- Active TCP
- Translate
- Rotate



1. Tap the **Relative to** field.

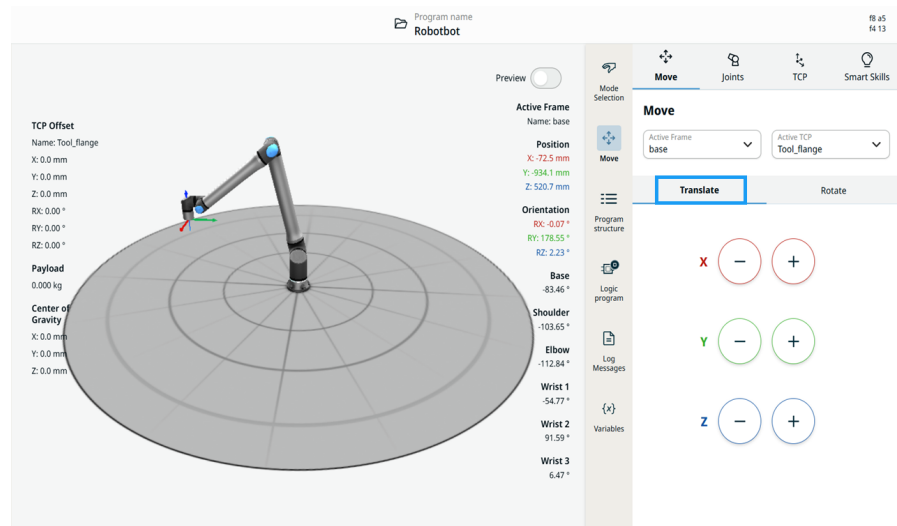
Four choices are given:

- world
- base
- flange
- tcp

2. Select the relative position you want the robot to move.
3. Tap the **Active TCP** field and select your preference.

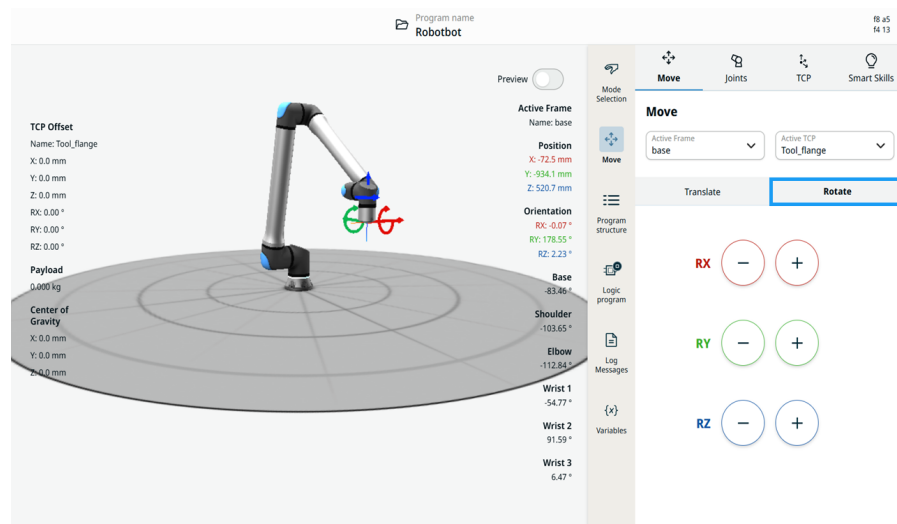
4. Tap the **Translate** tab.
5. Tap the **plus-minus** button of X, Y, Z coordinates to move the robot arm in an indicated direction.

In the main screen, you can see the corresponding movement, as indicated by the red, green, and blue straight arrows of the robot arm.



6. Tap the **Rotate** tab.
7. Tap the **plus-minus** button of RX, RY, RZ coordinates to change the orientation of the robot in the indicated direction.

In the main screen, you can see the corresponding movement, as indicated by the red, green, and blue circular arrows of the robot arm.



Joints Tool



Using the Joints tool

1. Tap Joints.

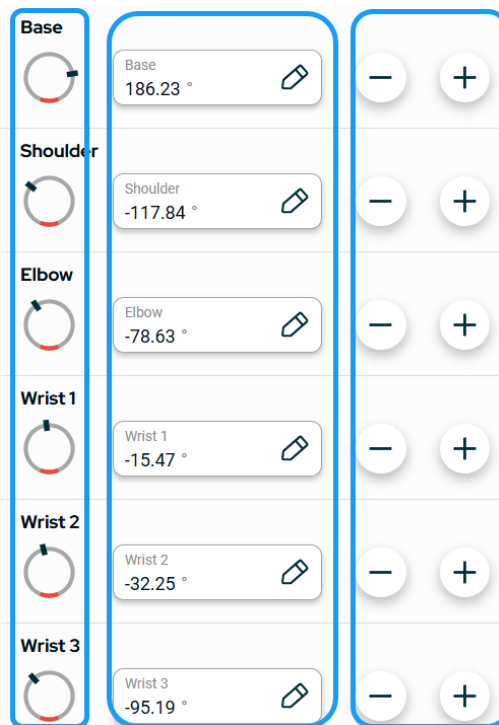


The Joints screen appears, which contains the six parts of the robot arm:

- Base
- Shoulder
- Elbow
- Wrist 1
- Wrist 2
- Wrist 3

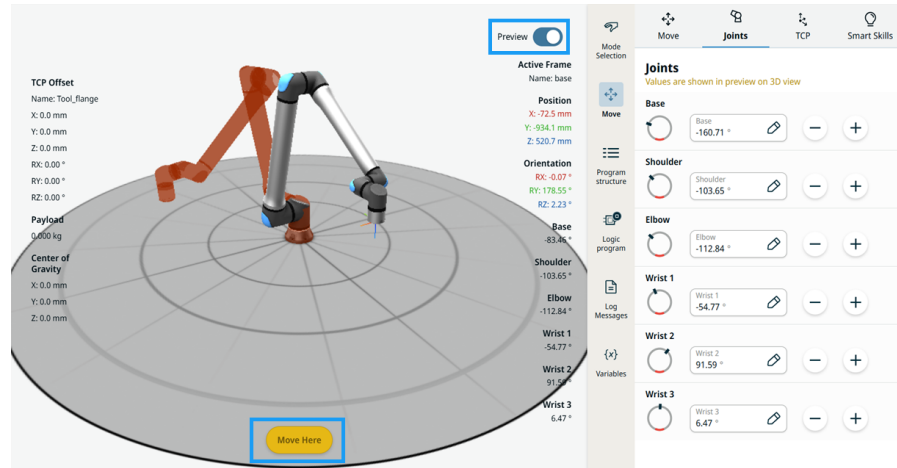
Each robot part gives you three adjustment tools so the position measurement be accurately achieved:

- Rotating ring
- Editable field
- Plus-minus button



2. Rotate the ring clockwise or counterclockwise to reach your preferred value/degree.
3. Tap the edit field and enter the value.
4. Tap **Confirm**.

Note: The rotating ring and the edit field have similar input for preview of the robot arm.



A preview of the robot arm position/orientation appears in brown color. You can disable the preview when you tap the **Preview** button to the left.

5. When you want to preview the whole robot arm movement, tap the brown **Move Here** button below the X, Y, Z coordinate space of the robot arm.
6. Tap the **plus-minus** button to adjust and reach your preferred measurement of the robot arm joints.

TCP Tool



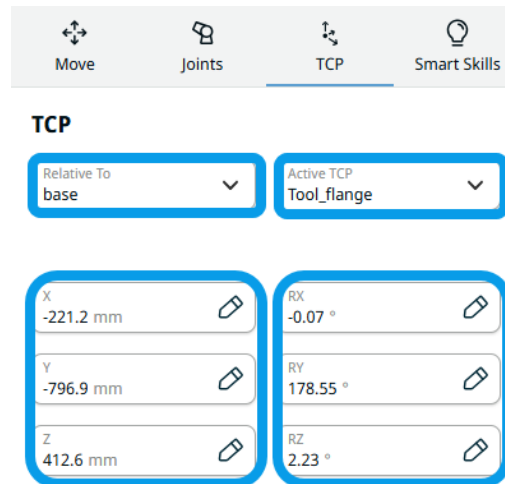
Using the TCP tool

1. Tap TCP.



The TCP screen appears which contains:

- Relative to field
- Active TCP field
- X tab
- Y tab
- Z tab
- RX tab
- RY tab
- RZ tab



2. Tap the **Relative** to field.

Four choices are given:

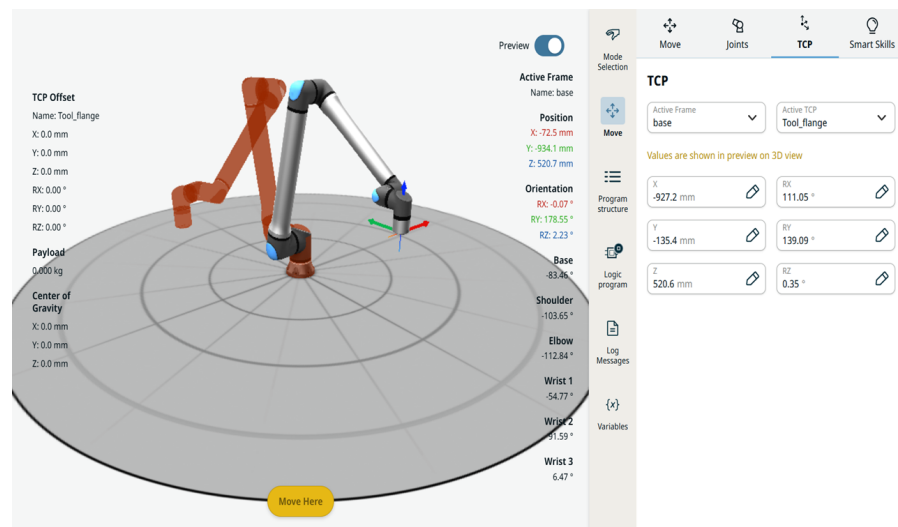
- world
- base
- flange
- tcp

3. Select the relative position you want the robot to move.

4. Tap the **Active TCP** field and select your preference.

5. Tap the **X, Y, Z** tabs and **RX, RY, RZ** tabs and edit the value for each.

A preview of the robot's position is shown in the robot arm with brown color.



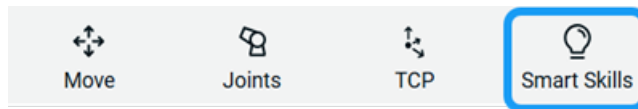
6. To preview the whole robot arm movement, tap the brown **Move Here** button below the X, Y, Z coordinate space of the robot arm.

Smart Skills Tool



Using the Smart Skills tool

1. Tap **Smart Skills**.



The **Smart Skills** screen appears which contains two fields, six large icons, and a button.

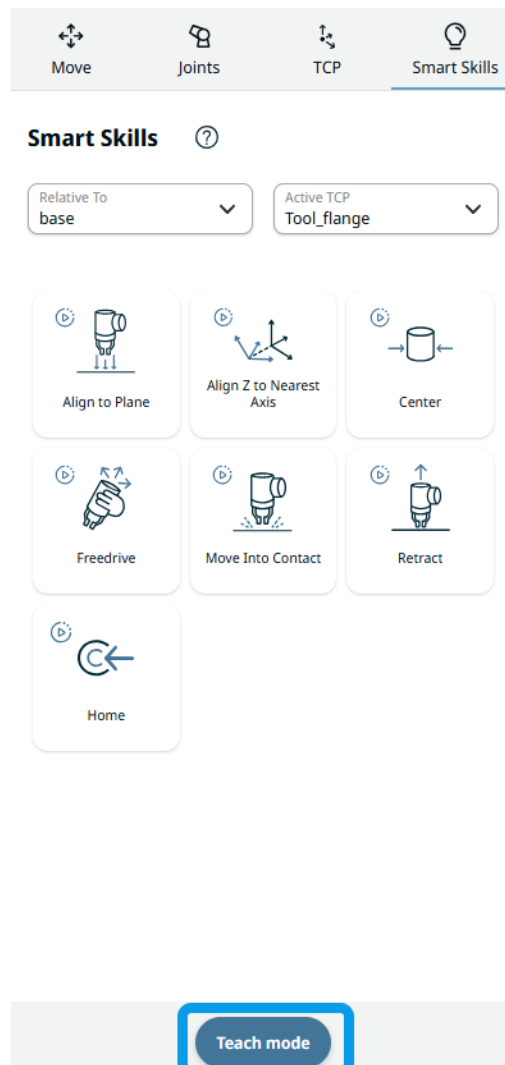
The two fields are:

- Relative to
- Active TCP

The seven Smart Skills icons are:

- Align to Plane
- Align Z to Nearest Axis
- Center
- Freedrive
- Move into Contact
- Retract
- Home

The **Teach mode** button is at the bottom.



2. Tap the **Relative to** field.

Four choices are given:

- world
- base
- flange
- tcp

3. Select the relative position you want the robot to move.

4. Tap the **Active TCP** field and select your preference.

5. Select and tap the smart skill icon of your preference.

In the main screen, you can see the corresponding movement of the robot arm.

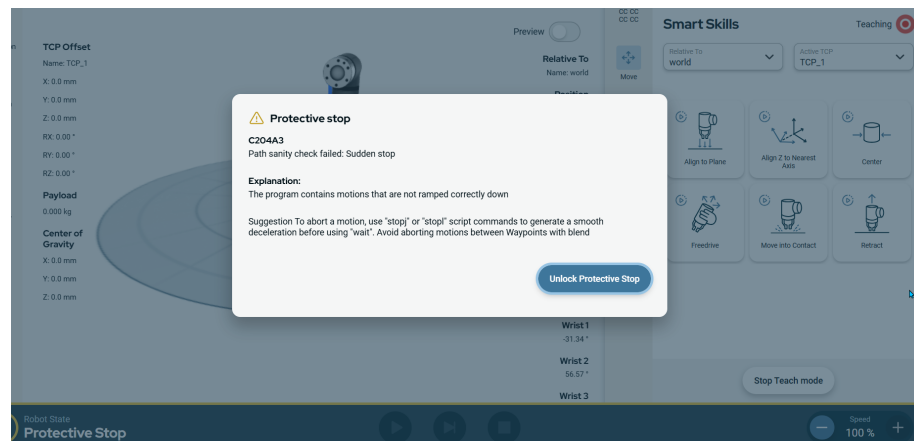
6. Tap the **Teach mode** button to program the robot by demonstration.

You see a red blinking light on the upper right of the multitask screen, signifying teaching is ongoing.

7. Apply a **Smart Skill** and wait for it to stop, or stop it yourself.

A program node is generated in the program tree.

8. If a **Protective stop** caution appears in the main screen, do the necessary commands to properly and safely program the robot.

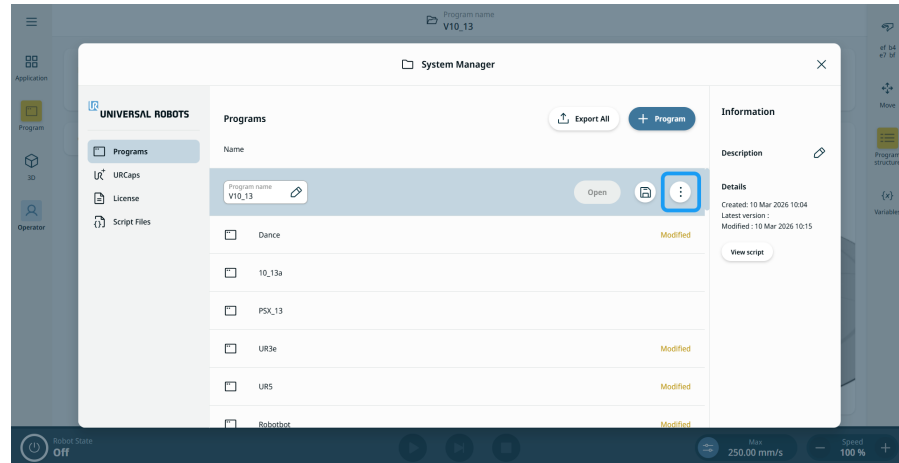


6.4.8. Export Program

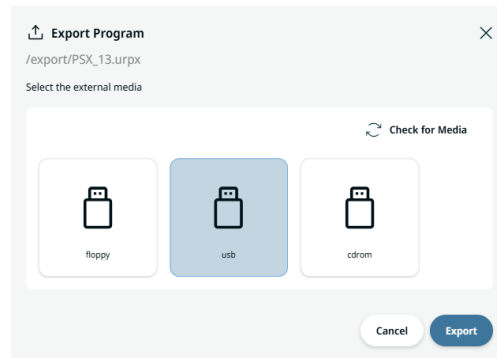
Description The **Export** feature enables you to export programs and/or applications from the robot to external media.

Export a single program

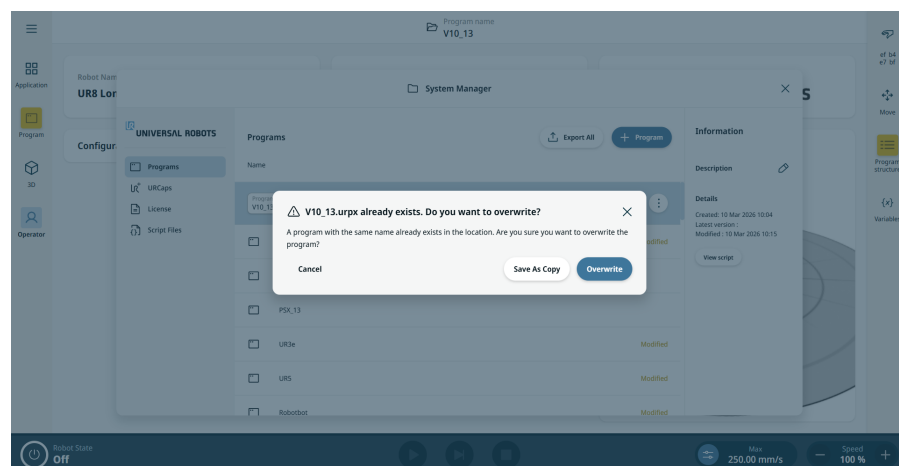
1. Tap the kebab menu (three vertical dots) for the program you want to export, then tap **Export**.



2. If multiple media are connected, select the external media you want to save the program to, then tap **Export**.

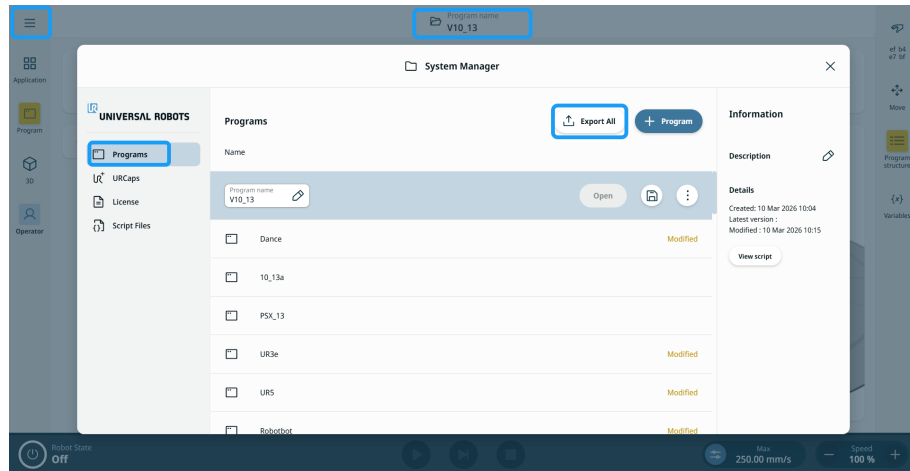


3. If a file name conflict occurs, an **Export Warning** dialog appears. This happens when a program with the same name already exists on the selected media. Tap **Overwrite** to overwrite the existing file or **Save As Copy** to save the exported program as a copy.

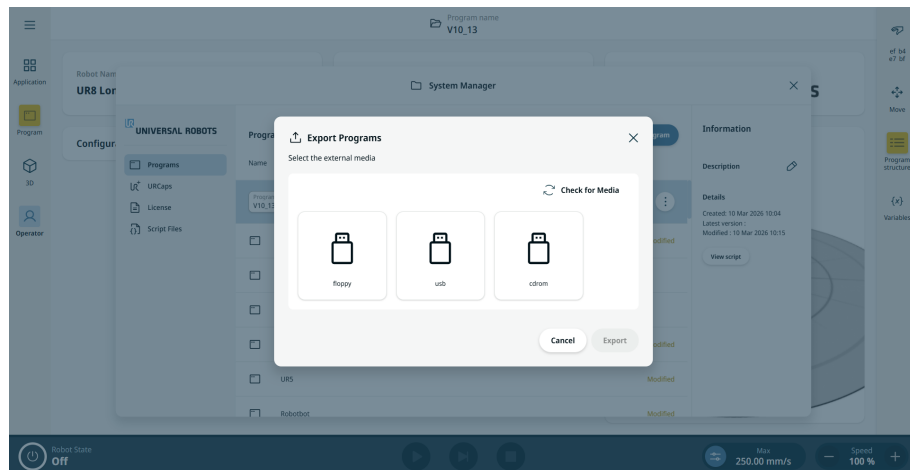


Export All The **Export All** feature enables you to export all programs from the robot to external media.

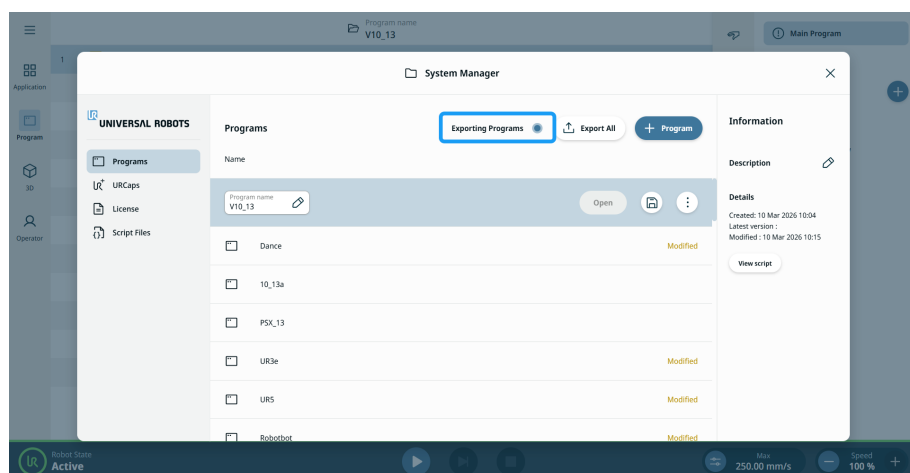
1. Either tap the header or the hamburger menu, then tap **System Manager**.
2. On the Programs screen, tap **Export All**.



3. Select the external media where you want to save the exported file.



4. Tap **Export**. An export progress indicator **Exporting Programs** is shown until the export is complete.





6.4.9. Shared Applications in System Manager

Description An **Application** contains robot configuration and safety settings, such as equipment, end effectors, application variables, and safety parameters. From PolyScope X 10.14, a single Application can be **shared by multiple programs**. When you change settings in a shared Application, every program linked to that Application uses the updated configuration. Manage Applications and their linked programs from the **Programs** tab in [System Manager](#).

Programs tab layout On the **Programs** tab, Applications and programs are shown in a unified accordion structure:

- Each **Application** is a collapsible accordion section listing the programs linked to it.
- The Application used by the currently open program is shown in **bold** with an **Active** tag.
- The Application row contains an input field for renaming the Application and a context menu.
- Programs are listed under their linked Application.

Application context menu Open the Application context menu to access:

- **Duplicate** – see [Duplicate Application](#)
- **Import Application Settings** – see [Import Application Settings](#)
- **Delete** – see [Delete Application](#)

Program actions From a program's context menu you can:

- **Duplicate** – see [Duplicate Program](#)
- **Change Application** – see [Change Application](#)

When [creating a new program](#), use the **Application** dropdown to link the program to an existing Application or create a new one.



NOTICE

Changes to a shared Application affect all programs that reference it. Verify program behavior after linking or changing Applications.

Delete Application

Description You can delete an Application from the **Programs** tab in **System Manager** when it is no longer needed. Deletion rules depend on whether programs are still linked to the Application.

Delete Application with linked programs If you try to delete an Application that still has programs linked to it, a dialog informs you that you must reassign or delete the linked programs first.

Delete Application with no linked programs

1. In **System Manager**, open the Application context menu.
2. Tap **Delete**.
3. When no programs are linked to the Application, a confirmation dialog asks whether you want to delete the Application.
4. Confirm to delete the Application.

Delete last program linked to an Application When you delete the last program linked to an Application, a dialog asks whether you also want to delete the Application. You can delete both the program and the Application in one step, or delete only the program and keep the Application.

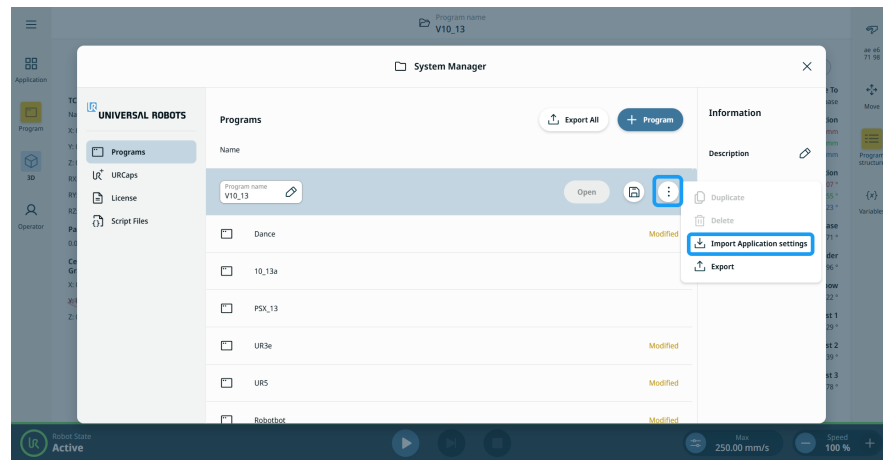
Import Application Settings

Description **Import Application Settings** enables you to reuse existing Application settings by copying them into another Application. See also [Shared Applications in System Manager](#).

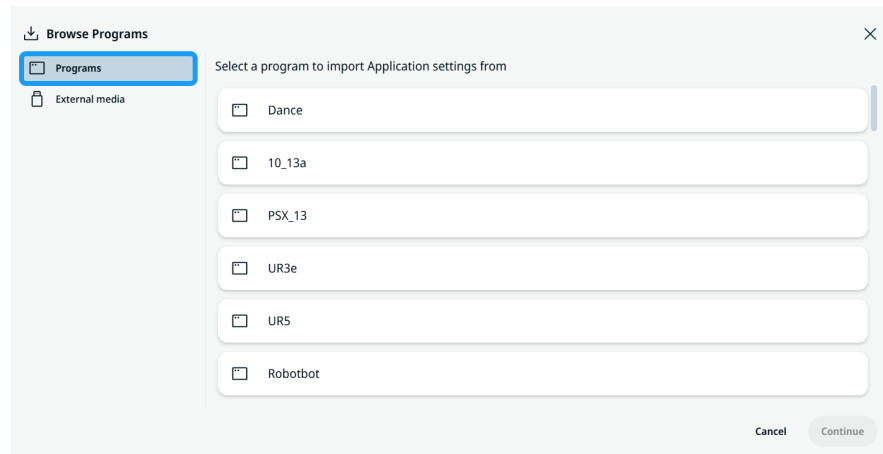


Import Application settings

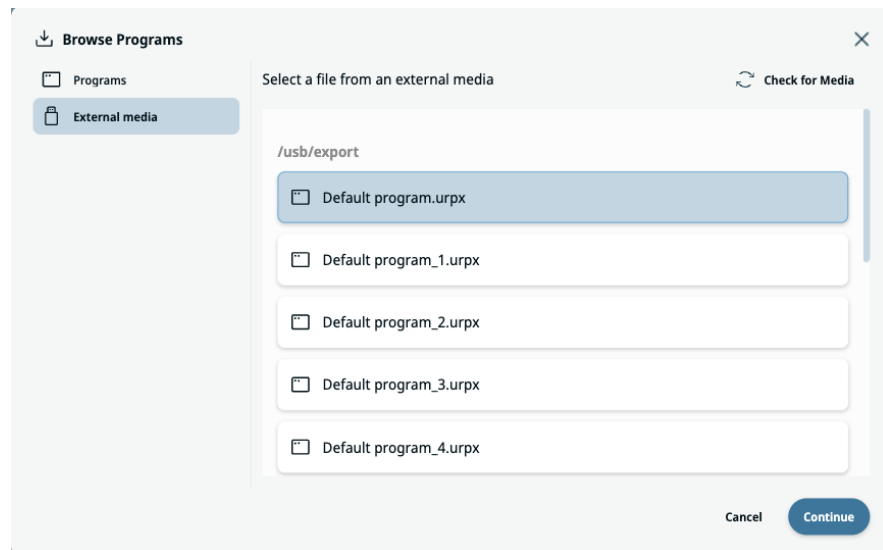
1. In **System Manager**, on the **Programs** tab, open the context menu for the Application that will receive the settings.
2. Tap **Import Application Settings**.



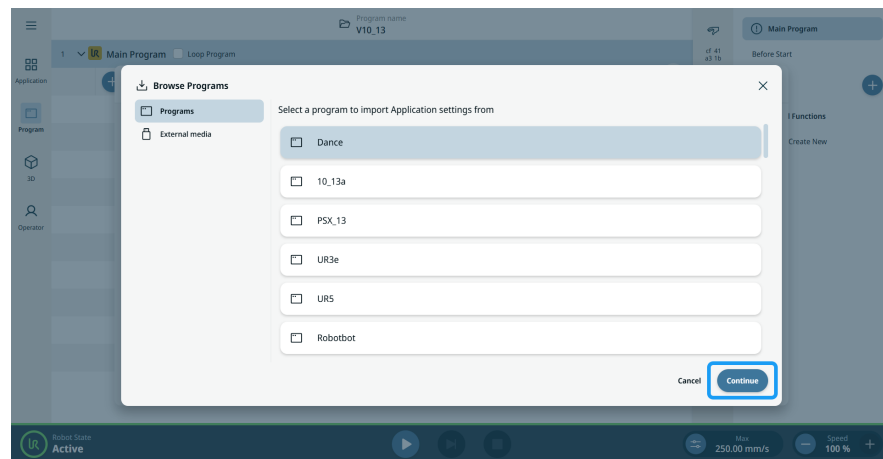
3. The **Browse Applications** screen appears with a list of Applications on the robot. Select the **Applications** tab to import settings from a local Application.



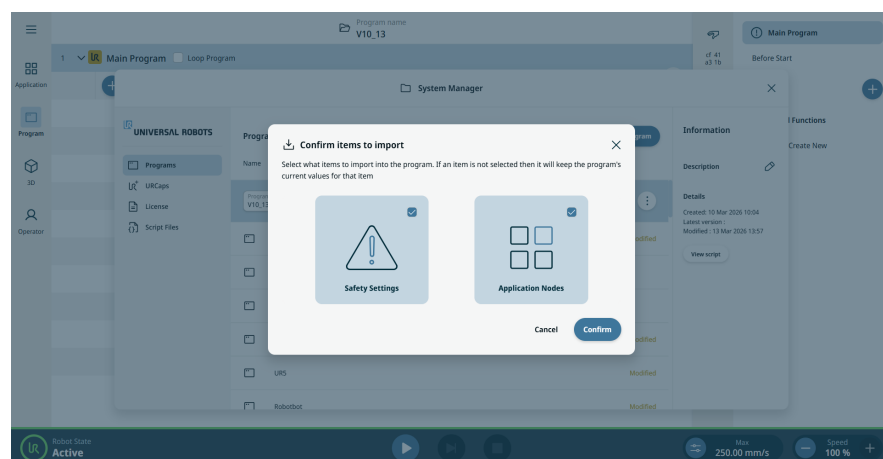
4. Select the **External Media** tab to import application settings from external media.



5. Tap the Application to import settings from, then tap **Continue**.



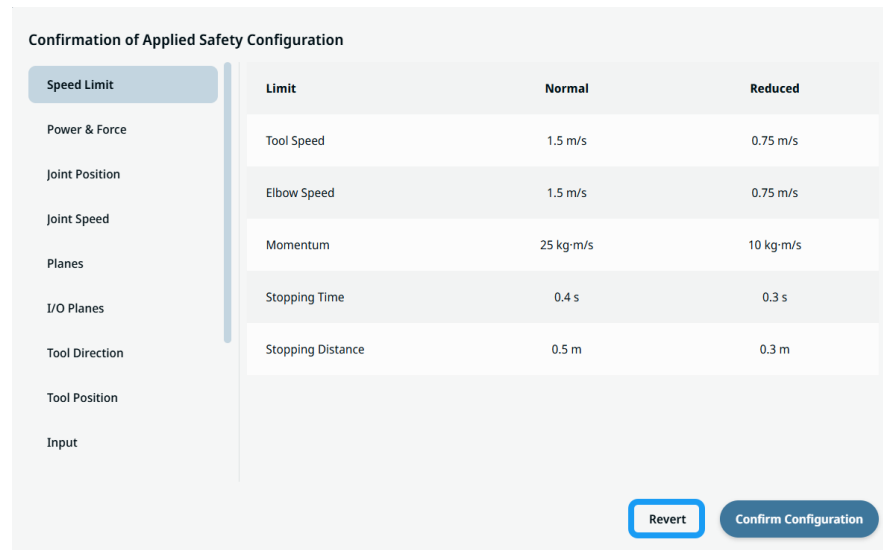
6. You can select either **Safety Settings** or **Application Nodes** or both.



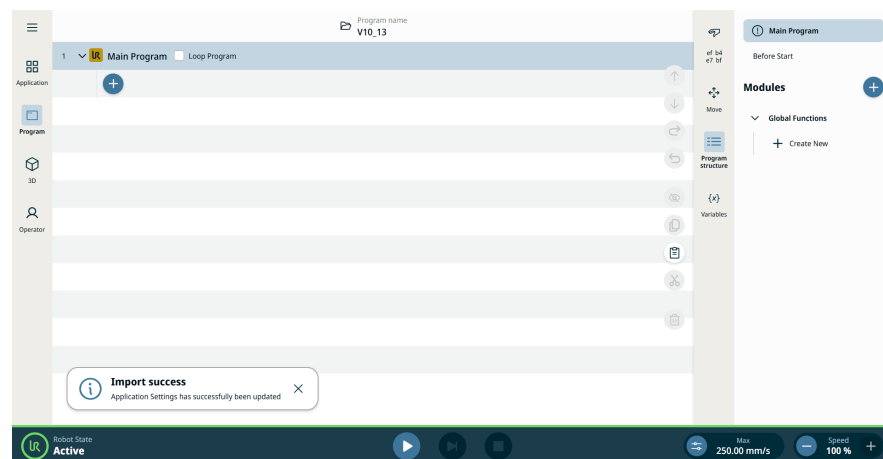
NOTICE

Safety Settings contains the data under **Safety** on the **Application** tab. **Application Nodes** contains all application content on the **Application** tab.

7. If you select **Safety Settings**, you are prompted for the safety password.
8. Tap **Revert** to cancel the entire import process.



9. When you confirm the import, a pop-up in the footer confirms it.

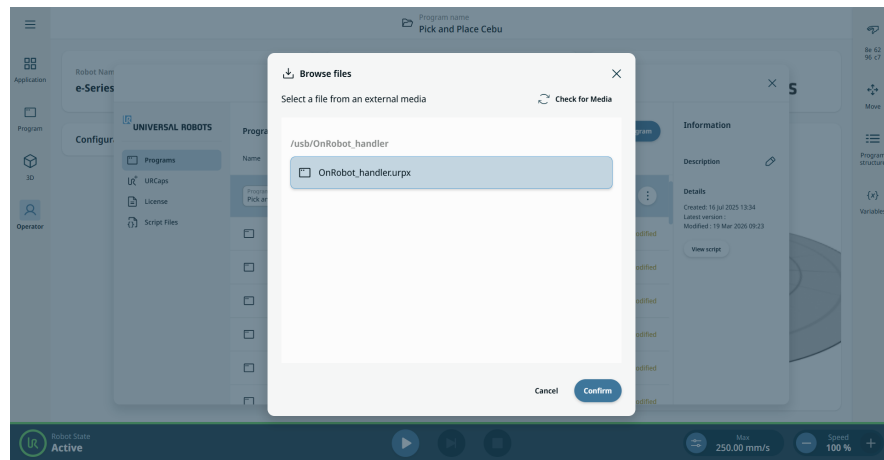


Import Warning Across Robot Types

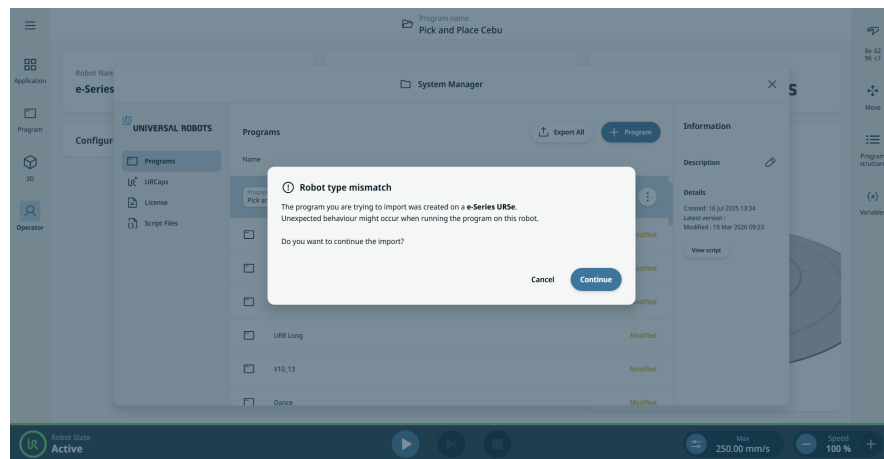
Description An **Import Warning** dialog box appears when you import a program that was exported from a different robot type. For example, when you import a program to a UR10e that was exported from a UR5e robot.

Import Warning (robot type mismatch)

1. In the **System Manager**, tap the **Add Program** icon, then select **Import Program**.
2. Select the program from the external media inserted in your machine, then tap **Confirm**.



3. The **Import Warning** dialog box appears and indicates a robot type mismatch.



4. Tap **Continue** to complete the import and use the program. Tap **Cancel** to abort the import.

Duplicate Application

Description **Duplicate Application** creates a copy of an Application's settings. Only the Application is duplicated – programs linked to the original Application are **not** duplicated.



**To duplicate
an Application**

1. In **System Manager**, on the **Programs** tab, open the context menu for the Application you want to duplicate.
2. Tap **Duplicate**.
3. Enter a name for the duplicated Application.
4. Confirm. The new Application appears in the list with a copy of the source Application settings.

Duplicate Program

Description

Duplicate creates a copy of a program. You can set the name of the new program and choose which Application the duplicated program uses. The dialog is the same as when [creating a new program](#).

**To duplicate
a program**

1. In **System Manager**, on the **Programs** tab, select the program you want to duplicate.
2. Open the program context menu and tap **Duplicate**.
3. Enter a name for the new program.
4. In the **Application** dropdown, select the Application the duplicated program should use, or choose **Create New** to create a new Application.
5. Tap **Confirm**.

Change Application

Description

You can change which Application an existing program uses. The program then references the selected Application's configuration and safety settings.

To change a program's Application

1. In **System Manager**, on the **Programs** tab, select the program.
2. Open the program context menu and tap **Change Application**.
3. In the dialog, select the Application the program should use. The program's current Application is not shown in the list.
4. Tap **Continue**. The program now uses the selected Application.



NOTICE

A program that is valid in one Application is not necessarily valid in another. If the program uses a grid, TCP, or other Application-specific resource from one Application but is changed to use another Application, the program can become invalid. It is your responsibility to ensure that the program behaves as expected after changing Application.

6.4.10. Logic Program

Description

PolyScope X applications can include a **Logic Program** that runs continuously in the background, independently of the main robot program. This is useful for tasks that must keep running when the arm is powered off, the main program is paused or stopped, or the operator has entered a safeguard area.

Typical uses include maintaining fieldbus communication (Modbus, Ethernet/IP, PROFINET, Socket, XmlRPC), reacting to external sensor inputs, reporting robot state to a production PLC, and controlling external equipment when the main program stops.



NOTICE

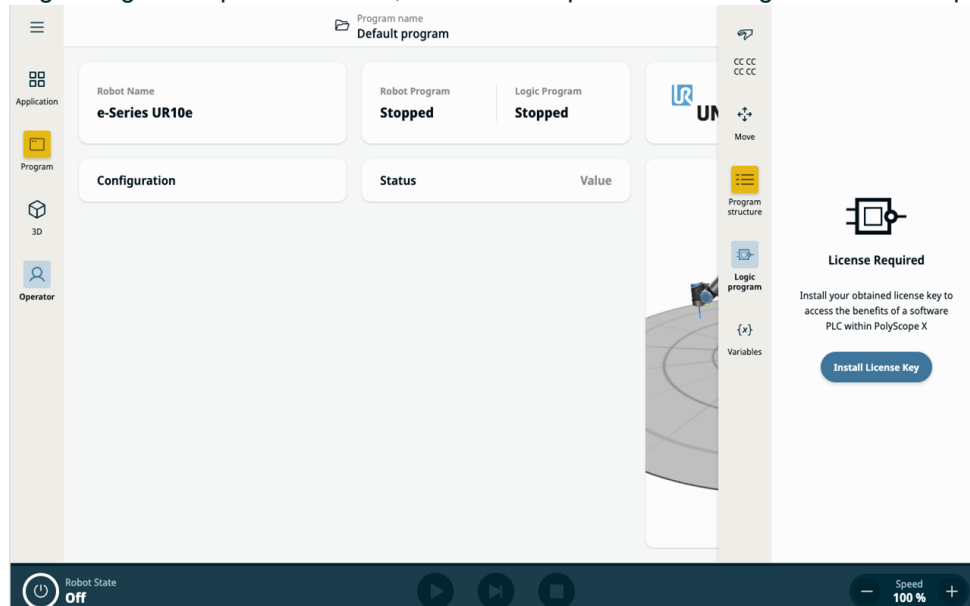
Logic Program cannot use functions that directly move the robot arm, change payload, or modify the TCP parameters.

Examples of applications with Logic Program

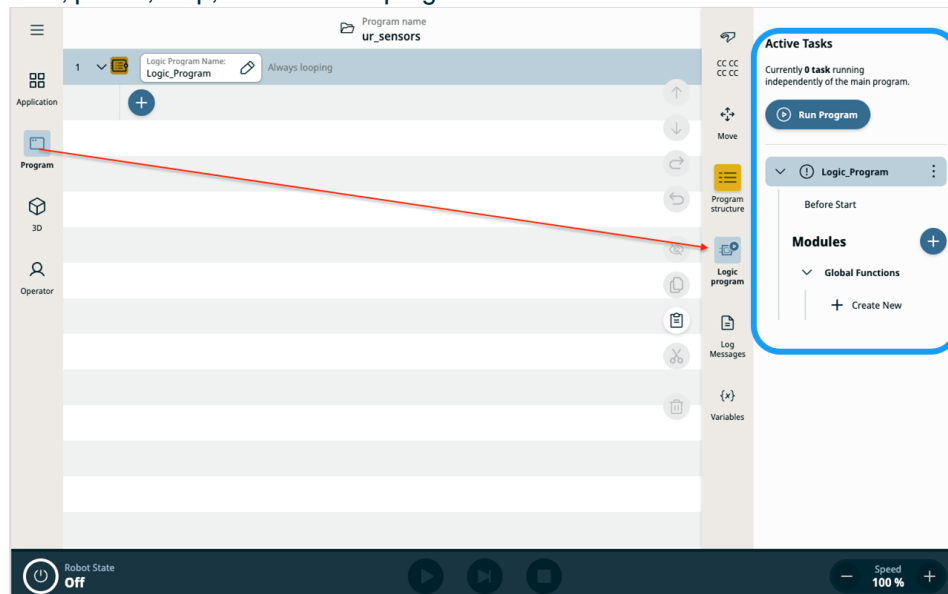
1. Active response to external sensor input while the robot is in a stop mode, such as Safeguard Stop, Protective Stop, or Emergency Stop, and the main program is paused or stopped, or before the robot arm is powered on.
2. Fieldbus communication (Modbus, Ethernet/IP, PROFINET, Socket, XmlRPC) is maintained even if the robot arm is powered off and the program is not running.
3. Continuous cell logic operation while the main program is interrupted by the operator.
4. Robot status report, which is conveyed to the production control software (production PLC).
5. Controlled stop of external equipment, such as welding machine, conveyor, glue dispenser, when the main program stops or pauses.
6. Input from operator is captured before the program is started.
7. Customized recovery due to fault conditions. The operator can decide what to do after resuming the main program.
8. Replacement of external PLC controller in small robot cells.

License

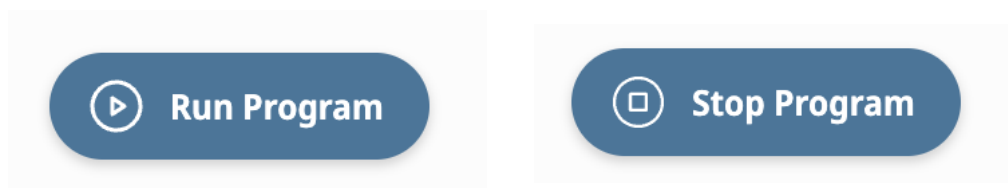
Logic Program requires a license, which can be purchased through the customer portal.



Controlling the Logic Program In the Program page, select **Logic program** on the sidebar. The expanded sidebar shows the controls for navigating the different parts of the Logic Program. It also allows you to start, pause, stop, and reset the program.



The run program button changes to either **Run Program** or **Stop Program**, depending on the state of the Logic Program. It acts as a shortcut for program control actions available in the Logic Program context menu.



The four **Logic Program indicators** show the current state of the program. They also indicate if the program is configured to start automatically when an application is loaded.

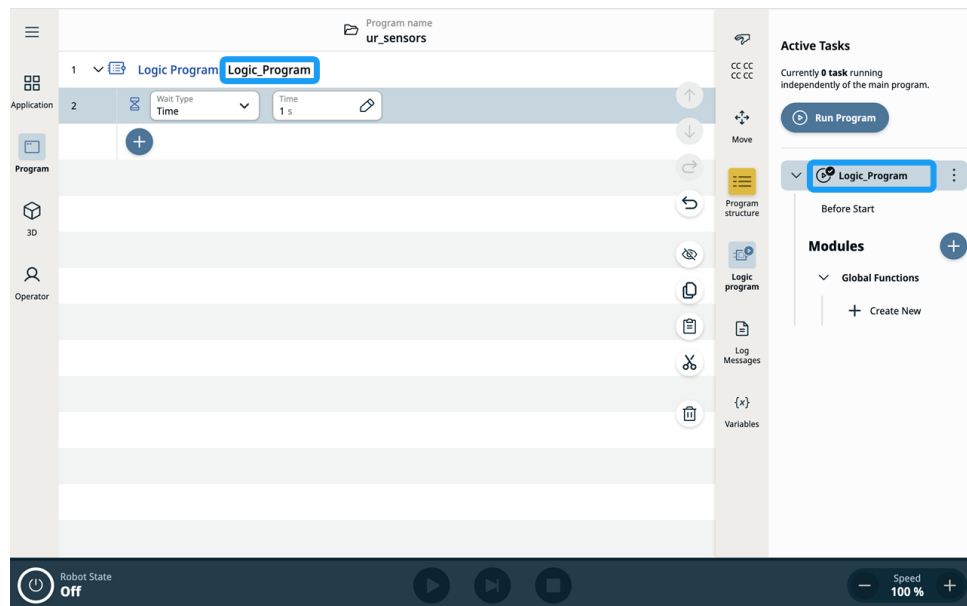


Logic Program structure

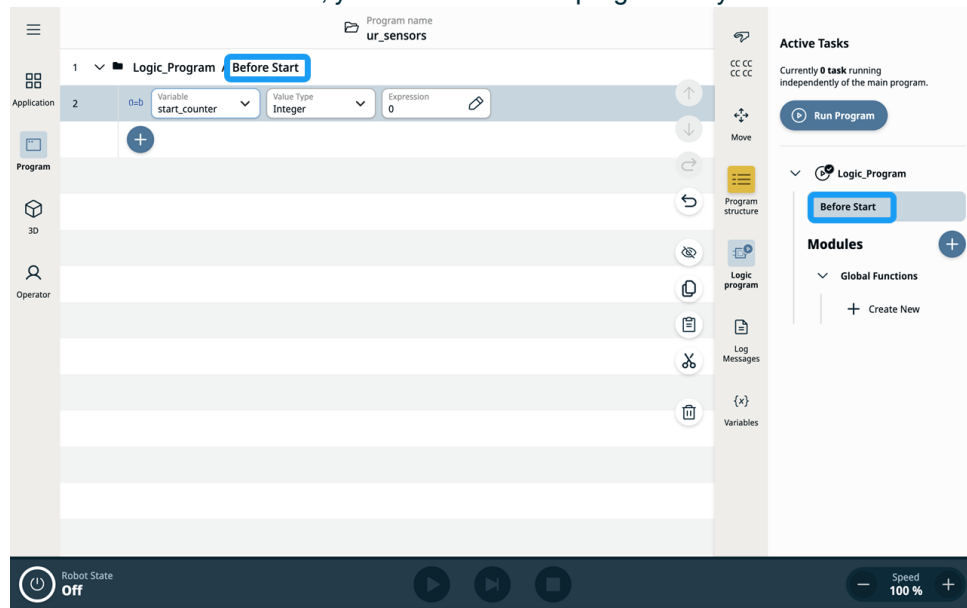
A Logic Program has the same structural sections as a main program:

- **Main program body** – runs continuously until stopped manually, with a `halt` instruction, or by a runtime error.
- **Before Start** – runs once when the Logic Program starts.
- **Modules** – functions and threads, organized the same way as in the main program. Thread functions only execute once by default; enable the loop option to run a thread continuously.

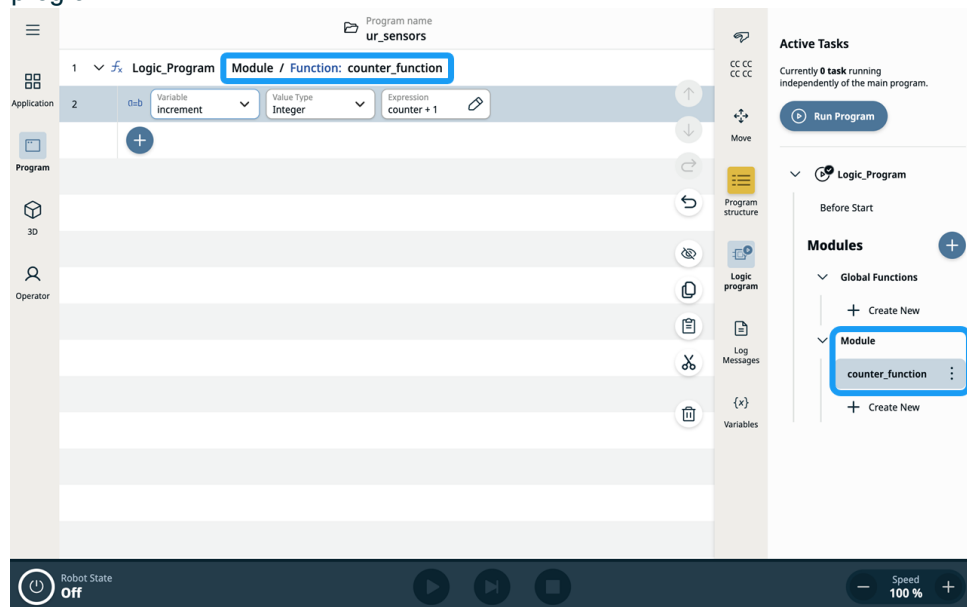
Robot program and Logic program each has its own set of global variables, independent thread pool, and independent execution lifecycle. Starting, stopping, or faulting one program does not stop the other. A `halt` or `pause` instruction affects only the program in which it is executed.



In the **Before Start** section, you can execute the program only once.



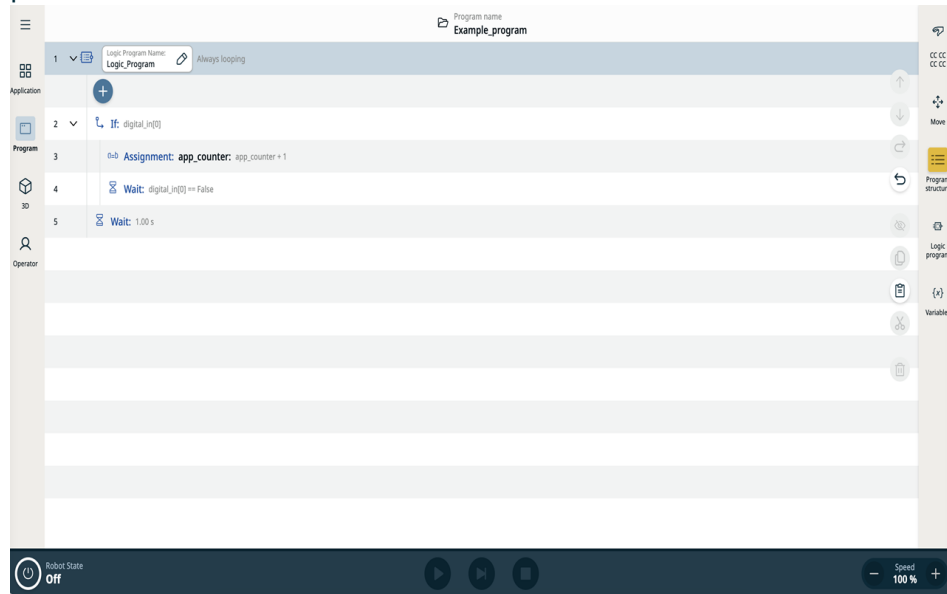
In the **Modules** section, you can see the list of functions and threads available for the program.



Exchanging data with the robot program

The primary way to communicate between the robot program and the Logic Program is through **Application Variables**. Variables can be both set and read from the robot and logic program.

This example counts items entering the robot cell's input buffer and displays the count to the robot program. The counter continues to increment even while the robot arm is powered off.



Reading the robot state

The robot state is available through built-in **RTDE variables**. These variables can be read using the script function `get_rtde_value()`. For a full list of available RTDE fields, see the [RTDE Guide](#).

The following example shows how to check if Safeguard Stop is active:

```
safety_status = get_rtde_value("safety_status")
# check for Safeguard Stop and Automatic Mode Safeguard Stop
is_safeguard_stop_active = (safety_status == 5 or safety_status == 12)
```

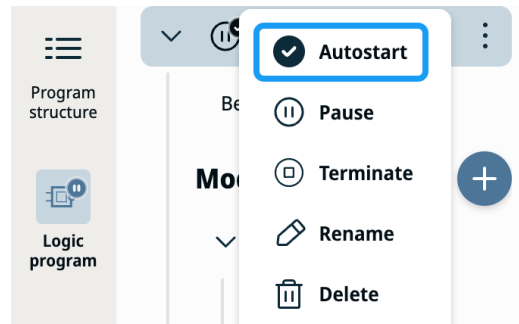
The following example checks if the robot program is paused:

```
is_program_paused = get_rtde_value("runtime_state") == 4
```

The URScript Control API also wraps this functionality.

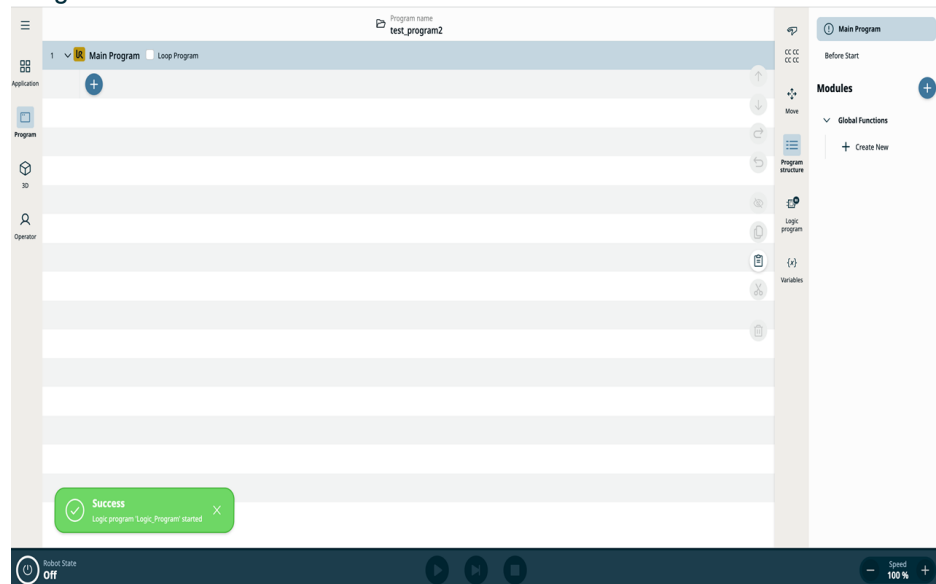
Automatically starting logic program

You can select a program to be started automatically when an application is loaded. When you select the **Autostart** option, the Logic Program starts immediately after the application is loaded.

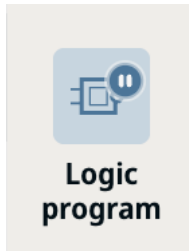


You can also automatically start the Logic Program when the robot controller is powered on and the last application is loaded.

The **Autostart** function works regardless of the operational mode or robot arm status. When Autostart is triggered on load, a snackbar notification confirms that the Logic Program has started.



The sidebar badge on the **Logic Program icon** always indicates the program's current state, whether it is running or stopped.



NOTICE

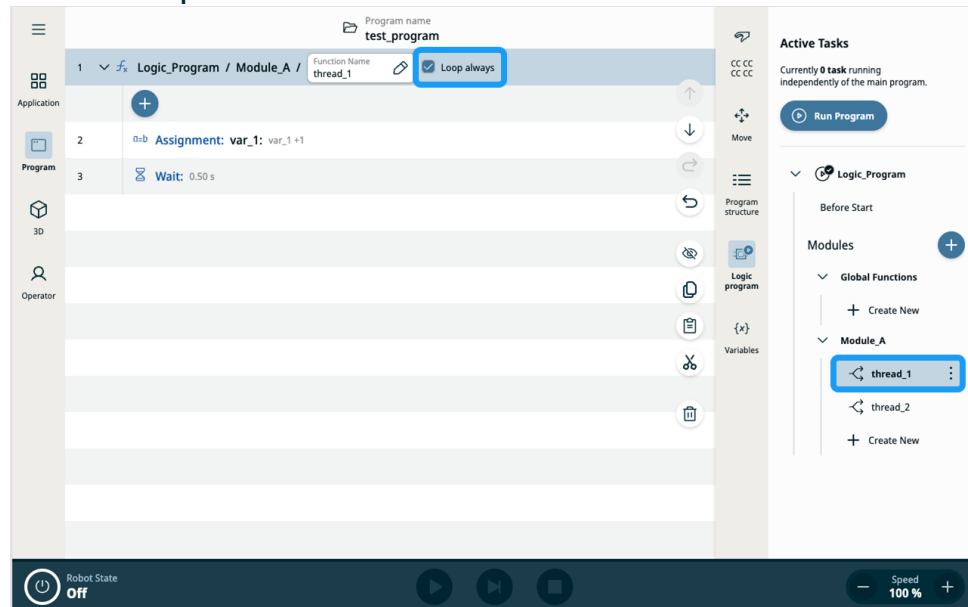
It is your responsibility to ensure that the Logic Program is safe when the robot is powered on and when different applications are loaded.

Logic programs can activate and deactivate external equipment using wired or PLC connections (Modbus, PROFINET, Ethernet/IP, Socket, direct outputs, or other interfaces).

Modules and threads

Logic Program functionality can be organized in functions. This works in the same way as in the robot program.

Each function can be marked as a **Thread**. By default the threads are executed only once. Select the **Loop** check box on the thread function to execute the thread continuously.



Sharing resources

While robot program and Logic program are executed independently, each with their own pool of threads and variables, they all share same hardware resources. This imposes few limitations that programmer should be aware of:

- Logic program threads execute on the same CPU as the robot program. Execution time of all logic program threads will add to execution time of all robot program threads. This can lead to “Runtime too much behind” violation in extreme cases.
- Some communication resources like Modbus signals or general purpose registers are shared resource. Writing to the same register at the same time will have unexpected results. Moreover currently observed behavior can change in the future when underlying implementation is optimized.
- **Modbus** – One Modbus client is shared. Signals created in either program are visible in both.
- **Ethernet/IP flexible adapter** – Each program opens its own reader or writer handle to the same instance. Handles must not write the same data fields.

Limitations

- Only a subset of URScript instructions is available in the Logic Program.
- Program execution indicators and breakpoint-based debugging are not supported for the Logic Program.
- Interpreter mode is not supported in Logic Programs. However, interpreter mode can still be started in the robot program while the Logic Program is running.

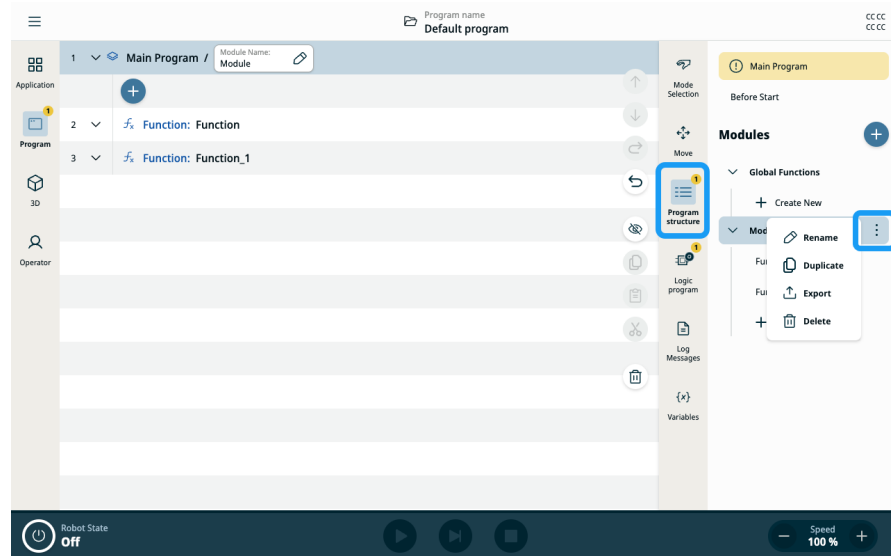
6.4.11. Export and Import of Modules

Description

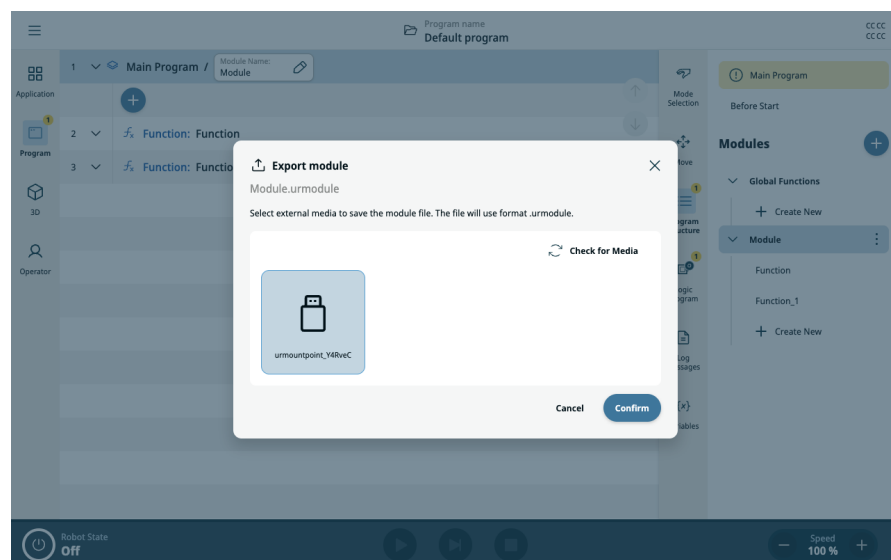
This feature enables you to export and import modules (.urmodule) to and from external storage media, which allows for a portable transfer of reusable program logic across different programs and robot installations.

Exporting a Module

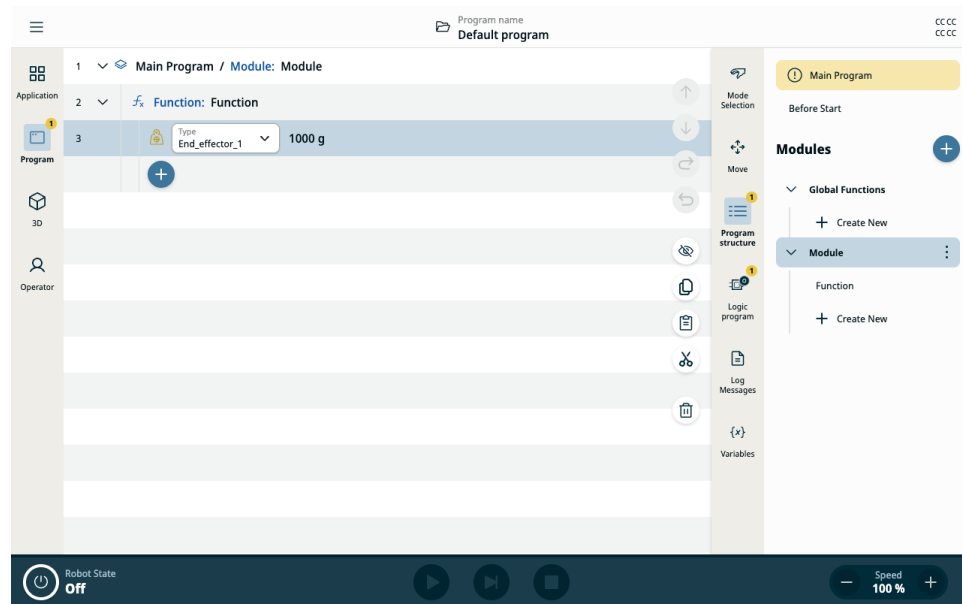
1. Insert a USB device into the Teach Pendant.
2. In the sidebar, open the **Program structure**. The sidebar expands, and you can see the list of modules and functions.
3. Select the specific module to export.
4. Tap the three-vertical dots icon (kebab icon) of the selected module to open the context menu.
5. Select **Export** to complete the process.



6. When you initiate the export, you'll see the standard export dialog. Ensure you've selected the correct external media as the destination. The system packages the module using the .urmodule file extension.
7. Tap **Confirm**, and the export process finalizes and saves the file to your selected external media.

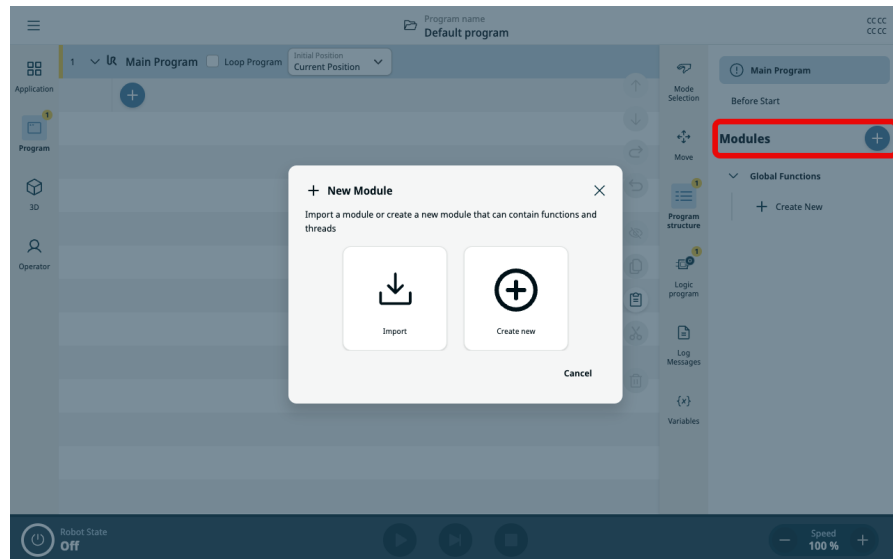


This is how the exported module looks in the interface.

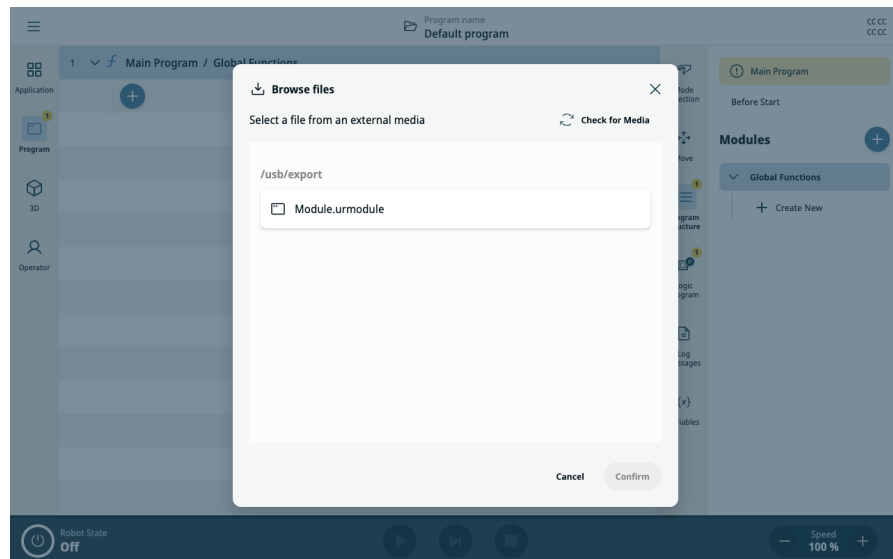


Importing a module

1. In the sidebar, open the **Program structure** and tap the **Add icon** at the right of Modules.
2. A dialog prompts you to either import an existing Module or create a new one. If you cancel the dialog, no Module is added to the program.



3. When you select **Import**, a dialog opens where you can choose the module (.urmodule file) to import.
4. Tap **Confirm** to add it to the program.

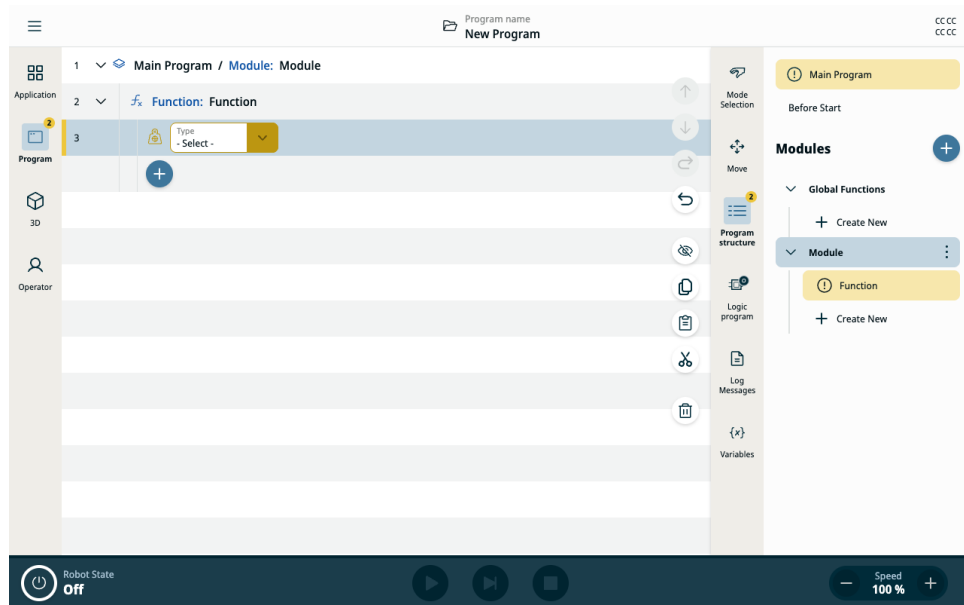




NOTICE

Program nodes that were valid in the module from another program may not remain valid in the new program. It is your responsibility to verify that the imported module works as expected.

This is how the imported module looks in the interface.



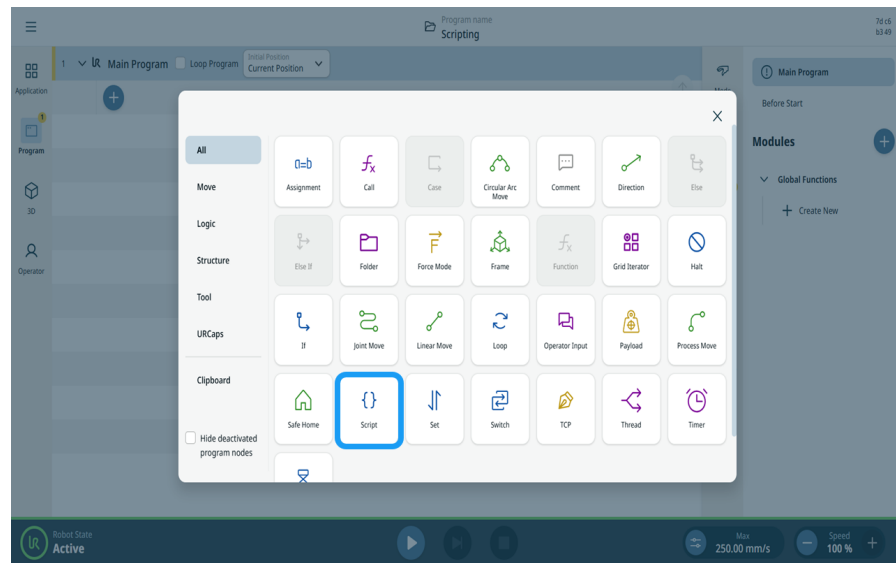
6.4.12. Export to Scripts

Description

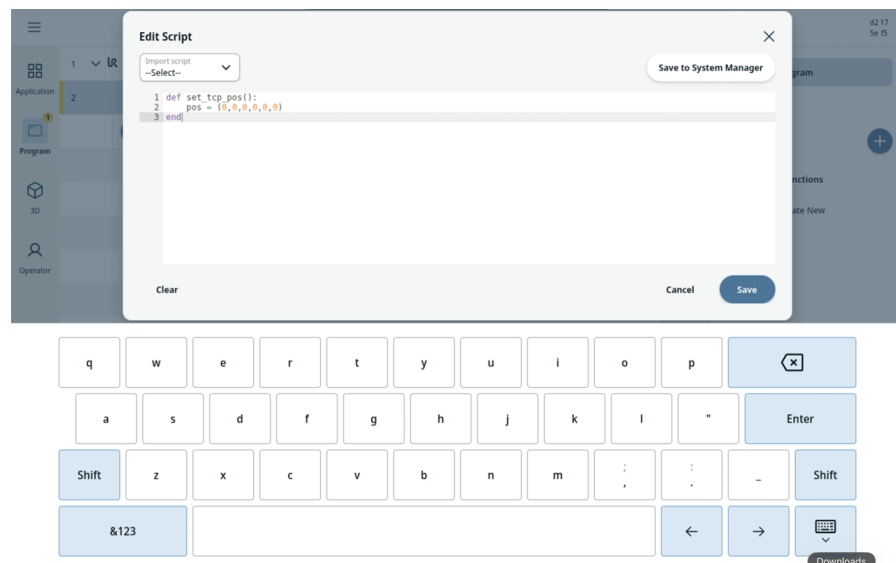
This feature allows you to export script files to a connected USB device. In exporting, the contents of a script node in the program can be saved to the **System Manager** and then exported from the System Manager to the USB device.

Saving to System Manager

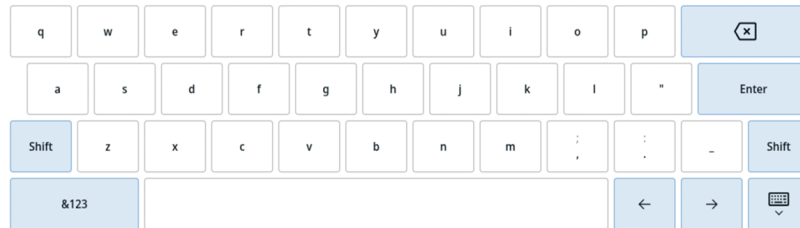
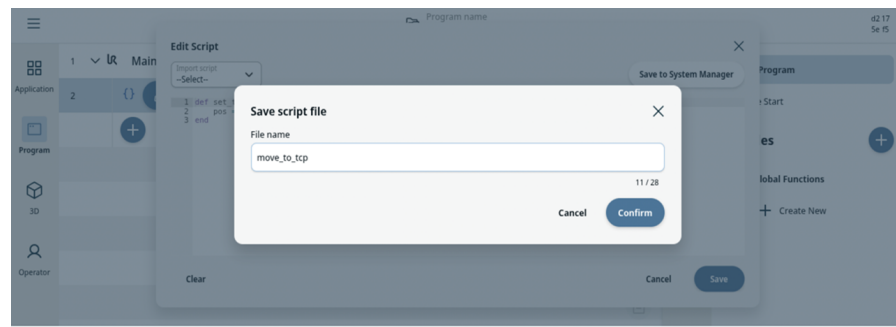
1. Navigate to **Program** in the **Main Navigation**.
2. Tap the **Add icon** and the **Script** command node.



3. Tap **Set Script** and you can make and edit the script.
4. **Save** the script to System Manager.

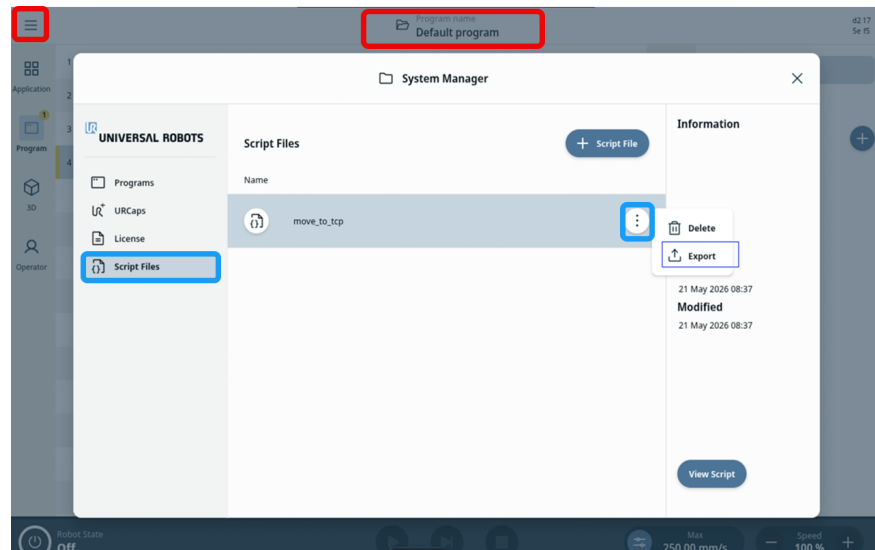


5. Specify the script file name and tap **Confirm**.

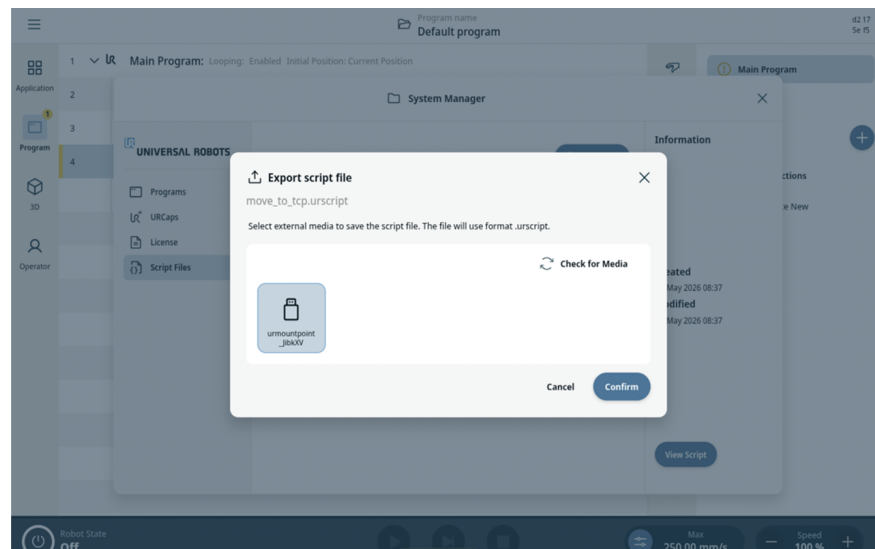


Exporting to USB device

1. Navigate to the **System Manager** by either **Header** or the **Hamburger Menu**.
2. Go to **Script Files** and tap the three-vertical dots (kebab icon) of the script.
3. Select **Export**.



4. Select the USB device and tap **Confirm**.



6.4.13. Move to First Position

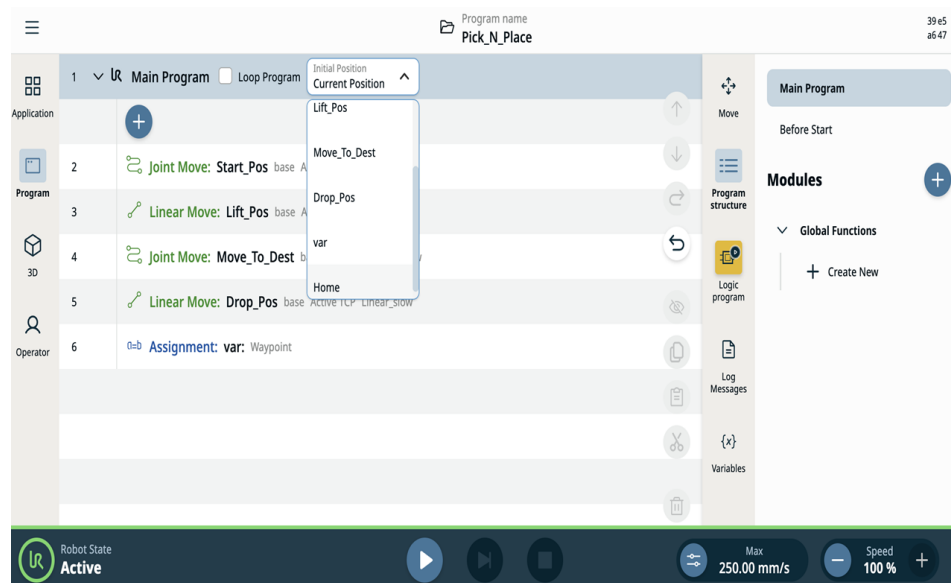
Description

This feature enables you to specify a starting position for a robot program from the drop-down menu at the top of the **Program** page, at the main node level. Before the program execution robot will auto move to this starting position, unless the option of the Current Position is selected.

The current position could be selected from the following options:

- **Waypoints** that are created in the program.
- Any **assignment variables** used in the program that are of type waypoint.
- **Home Position** configured at the application Level.
- **Current Position**, which is the default value for a program if no selection is made.

You can select any one from the drop-down as the starting position for the program to ensure smooth execution of the program without getting into any Protective Stops or other errors.



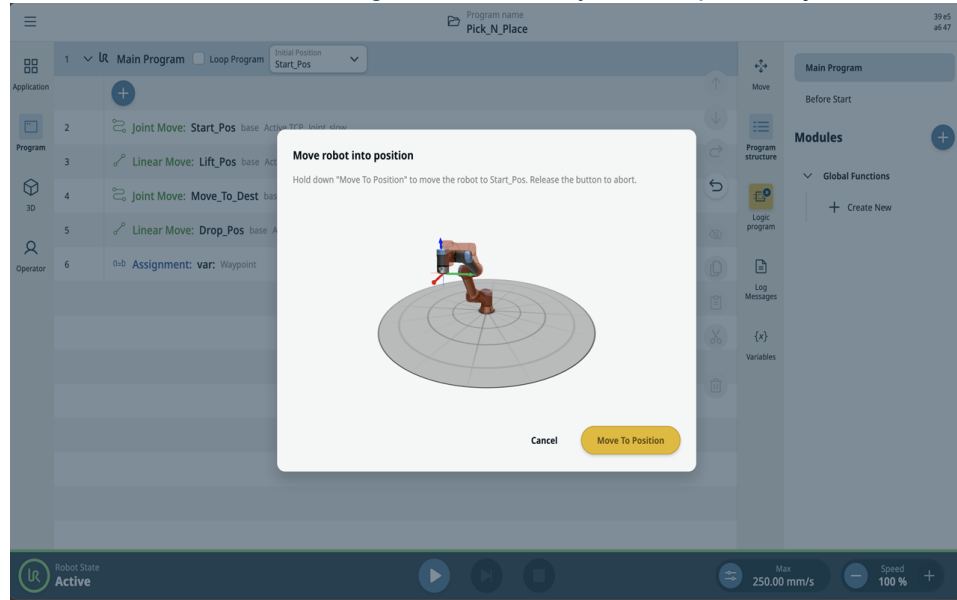
Examples

Action 1: You select **Current Position** as the starting position.

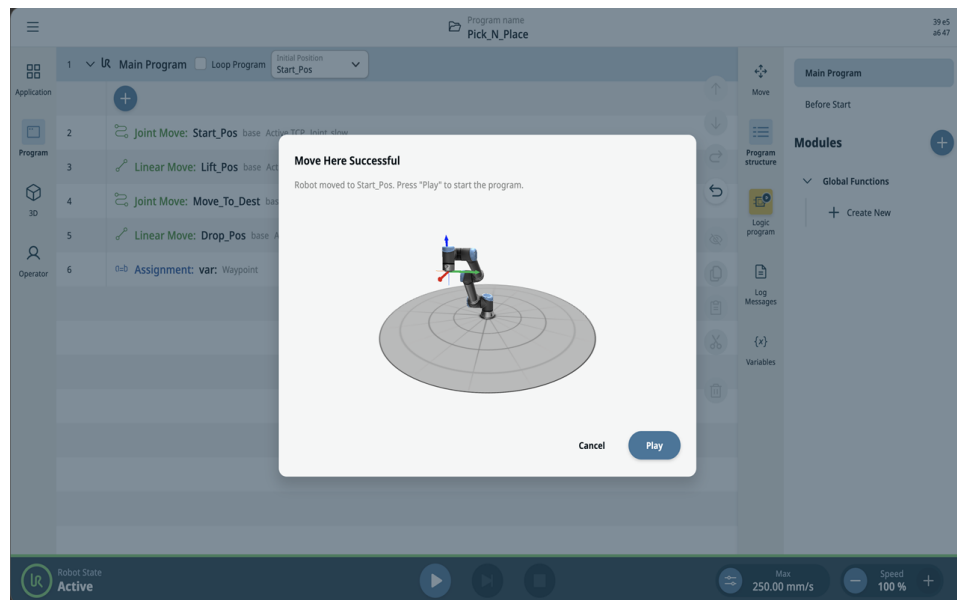
Function 1: When you tap the **Play** button, the program execution will start from the current position of the robot. This is the same as the previous behavior.

Action 2: You specify a starting position as a **value** other than the Current Position.

Function 2a: The Current Position of the robot is different from the starting position, and a **Move robot into Position** dialog is shown where you can tap the **Play** button.



What to do: Tap and hold the **Move to Position** button to move the robot gradually to the starting position. Once the robot reaches the starting position, the program can be played.



Function 2b: The Current Position of the robot is the same as the starting position. When you tap the **Play** button, the program execution starts without displaying the **Move robot into Position** dialog as the robot is already at the starting position. This will also be the case if you do not specifically select a starting position in the drop down menu.

6.4.14. Startup Program

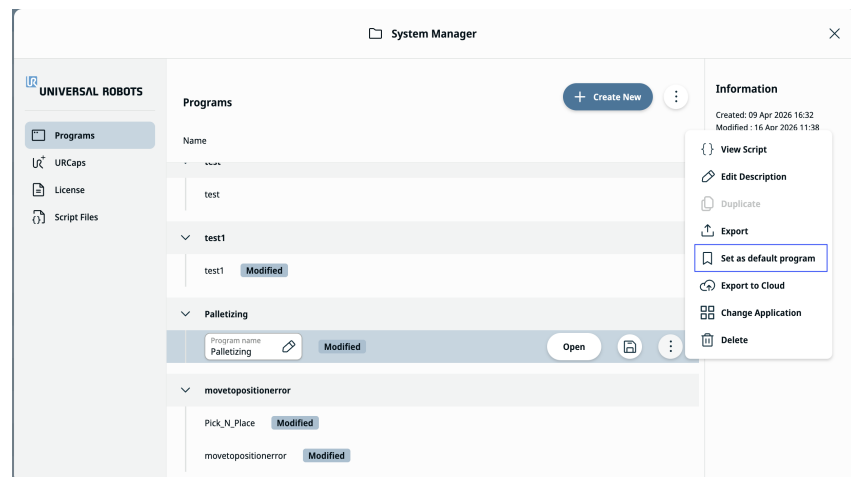
Description Use **Set as Default Program** in **System Manager** to define which program loads automatically on restart or reload.

PolyScope X behavior on Restart or Reload

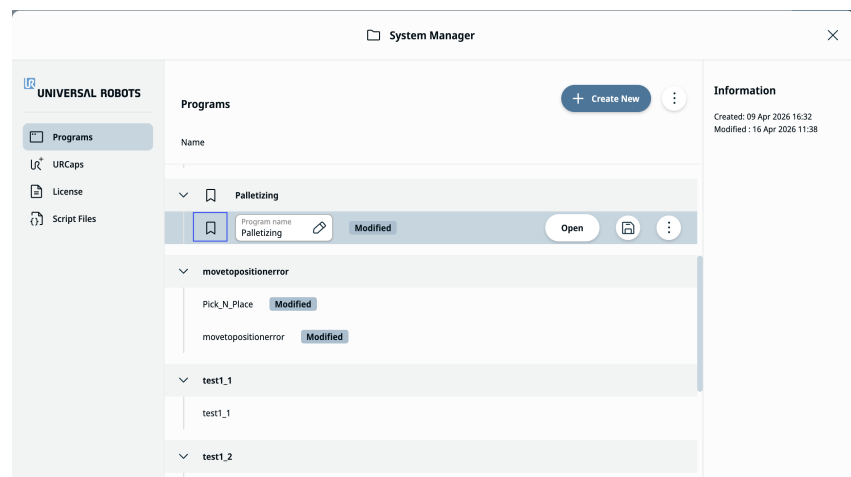
- If you set a default program, the system loads the default program.
- If no default program is set, PolyScope X loads the last opened program.

Set as Default Program

1. In **System Manager**, on the **Programs** tab, select a program.
2. Hit the kebab menu and tap **Set as Default Program**.

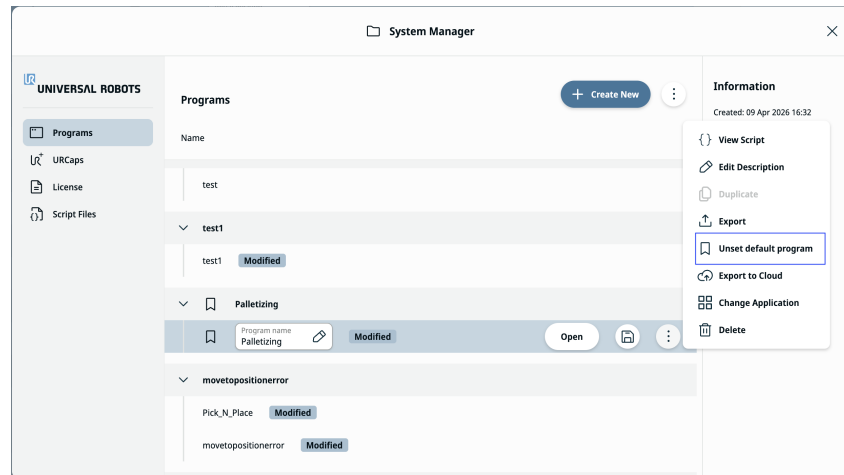


3. Set the selected program as the default.



Unset Default Program

1. Hit the kebab menu for the same program.
2. Tap **Unset Default Program**, as shown in the figure below.



3. Restart or reload PolyScope X. The system loads the last opened program.

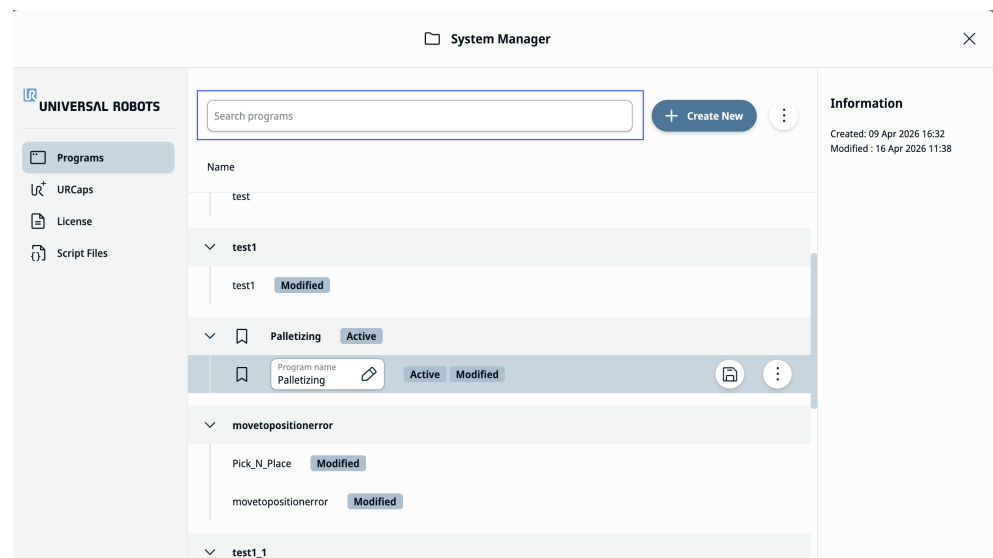
6.4.15. Program Organization

Description

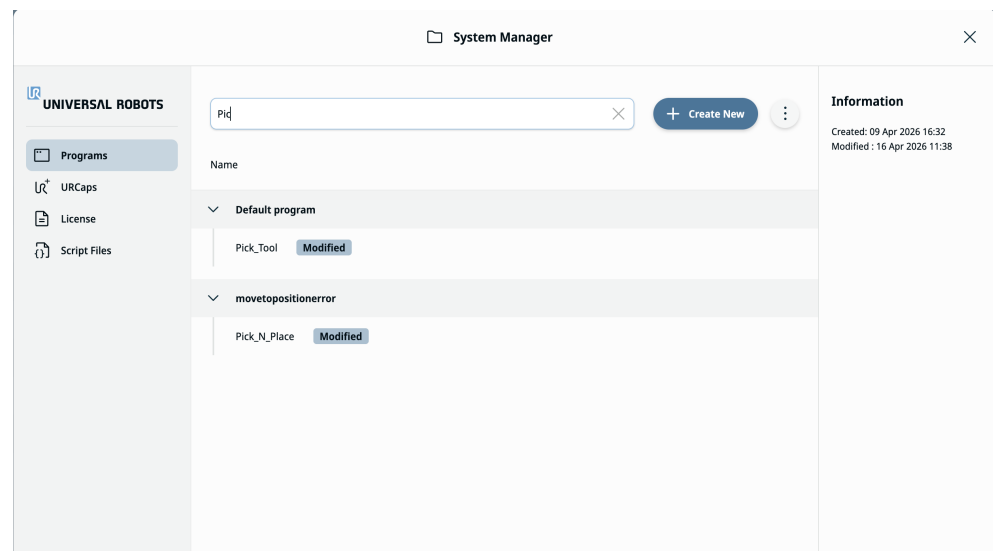
This feature enables you to search for programs in **System Manager** by name. The search matches any part of a program file name and filters the program list to show matching programs.

Predictive Search

1. In **System Manager**, on the **Programs** tab, locate the search field.



2. Type characters in the search field to filter the program list.



3. View the filtered results list. The system displays only programs with file names that contain the characters you entered.
4. Modify or clear the search text to update or reset the results.

6.5. Common Operating Tasks

Description

You perform routine robot operations from the Operator Screen by running programs, adjusting parameters, monitoring status, and managing variables without modifying the program.



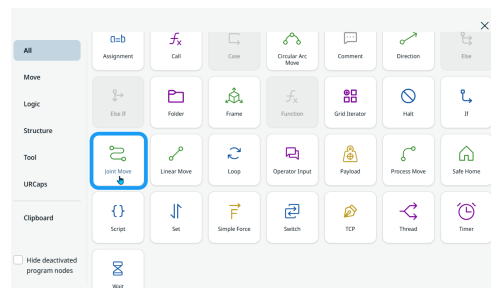
6.5.1. Pick and Place

Description

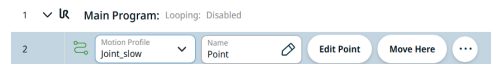
Pick and place is used when a robot picks up a piece from a tray or fixture and places it again in another tray or fixture. In a pick and place program, you create a default payload in the installation. Then you add a Set Payload when picking up an object. You update the payload after the gripper closes but before starting to move. Additionally, you use the Set Payload after the object has been released.

Creating a Pick-and-Place program

1. In the main navigation, tap the program icon. You are directed to the main program screen.
2. Tap the add icon to see the commands toolbox and choose the [7.5.4 Joint Move on page 174](#) icon.



3. A command node with **Motion Profile** and **Name** fields, **Edit Point** and **Move Here** buttons, and ellipsis icon is inserted into the program tree.

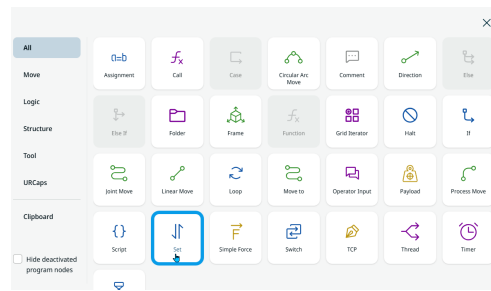


Position the robot above a tray/fixture

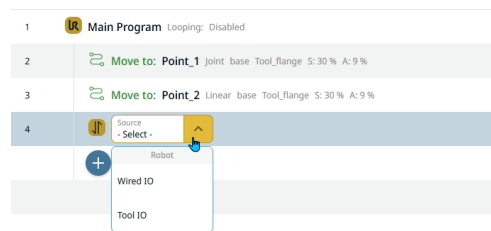
Robot grabs a thing from a tray/fixture

4. Tap **Edit Point** to mark the robot's first position above the corner of the tray or fixture. You are directed to the 3D viewer. Move the robot arm using the **plus-minus button** of the X, Y, Z coordinates and **Save**.
5. Name it as **Point_1** and tap **Confirm**.
6. Tap the add icon and insert [7.5.5 Linear Move on page 180](#) command to grab the piece from the tray in a linear movement.
7. In the Edit Point, move the robot arm using the plus-minus button of the X, Y, Z coordinates, and Save. Jog down using the **Z axis** in blue arrow.
8. Name it as **Point_2**.

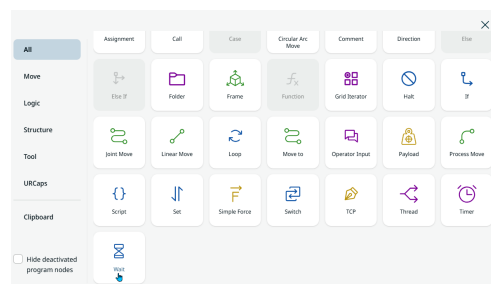
- Tap the add icon. To close the gripper, choose the **Set** icon.



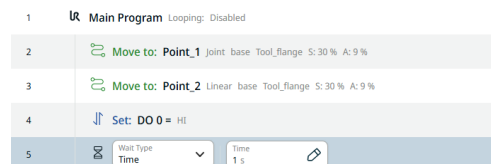
A command node with **Source** field is inserted into the program tree.



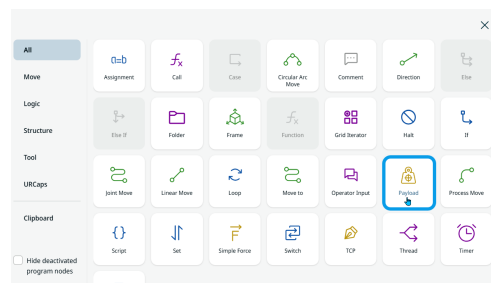
- Choose **Tool IO** in the Source field. A Signal field is inserted to the right and choose **DO 0**. A Value field is further inserted and choose **High**.
- Tap back the add icon and insert a **Wait** command.



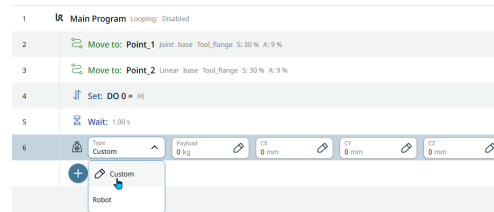
- Choose **Time** in the Wait Type field and enter **1** in the Time field.



- Tap back the add icon and insert a **Payload** command to update while the gripper is holding the piece.

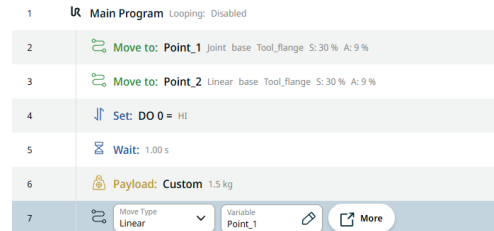


14. Choose Custom in the **Type** field. Four additional fields are added. In the **Payload** field, enter 1.5 kg in its Value tab, and in the **cz** field, enter 75 in its Value tab. Then **Confirm**.



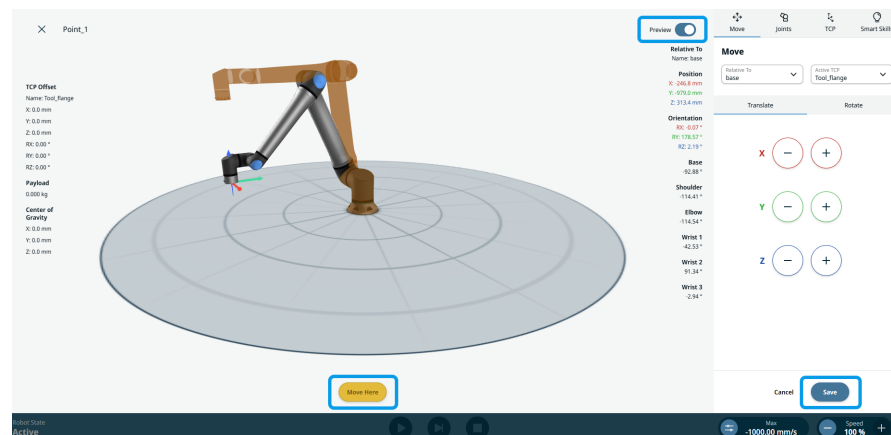
Robot moves
to the point
above a
tray/fixture

15. It is time to move up to the point above the tray again by using a variable.
16. Tap the add icon and insert a [7.5.5 Linear Move on page 180](#) command.
17. Tap the Name field, and in the Variable tab, select Point_1. Then Confirm.
By using a variable position, you only have to reattach once, and not everywhere that position is used in case adjustments are needed.



Robot
reaches the
end of a
tray/fixture

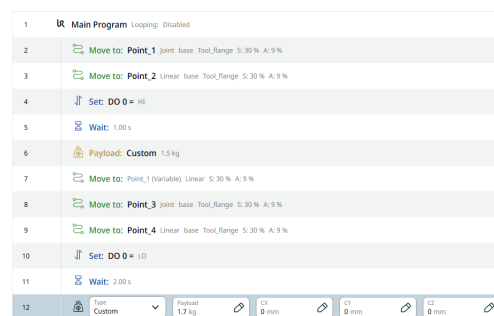
18. To move the robot into position, go back and tap the first [7.5.4 Joint Move on page 174](#) command in the program tree.
19. Tap **Edit Point**, and you are directed to the 3D view of the robot. Press the **Move Here** button until the robot arm reaches the previewed position in brown color. The blue **Preview** button at the right top side turns to white, which indicates that you have reached the desired position. Tap **Save**.



20. For the robot to reach the end of the tray, tap the last created command in the program tree.
21. Tap the add icon and insert another [7.5.4 Joint Move on page 174](#) command in the program tree.
22. In the Edit Point, move the robot arm using the plus-minus button of the X, Y, Z coordinates and Save. Jog sideways using the **X axis** in red arrow.
23. Name it as **Point_3**. Then Confirm.

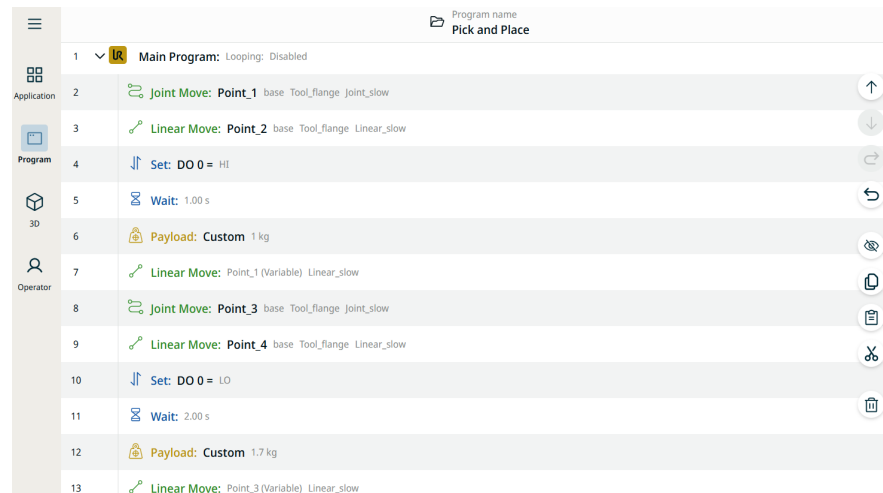
Robot places a picked thing

24. To get to the Place position, tap the add icon and insert another [7.5.5 Linear Move on page 180](#) command.
25. Tap Edit Point, and move the robot arm using the plus-minus button of the X, Y, Z coordinates and Save. Jog down using the **Z axis** in blue arrow.
26. Name it as **Point_4**. Then Confirm.
27. Tap the add icon and insert a **Set** command to open the gripper.
28. Select **Tool IO** in the Source field.
29. Select **DO 0** in the Signal field and **Low** in the Value field.
30. Insert another **Wait** command. In the Time field, enter 2 in the Value tab.
31. Tap the add icon and insert another Payload command while the gripper is empty.
32. Choose Custom in the Type field.
35. In the Payload field, enter 1.7 in the Value tab, and Confirm.



36. Insert another [7.5.5 Linear Move on page 180](#) command in the program tree to move the robot up to a point above the tray again by using a variable.
37. In the Name field, choose **Point_3** in the Variable tab, and Confirm.

This is the final structure of your program tree.



38. Tap the **3D Viewer** in the main navigation and the **Play** button in the footer to view your created pick-and-place program.

6.5.2. Motion Profiles

6.5.3. Global Variables

Description **Global Variables** contain the list of all variables in the robot program with global scope. These encompass ones created and found in:

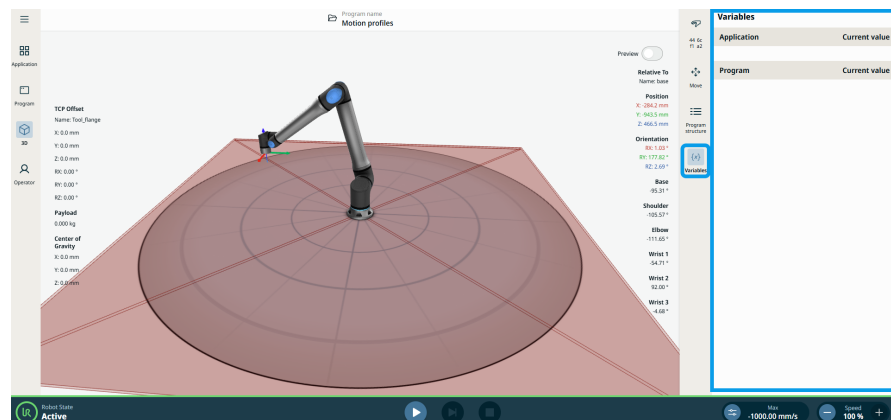
- Assignment nodes
- Operator Screen configuration
- Status
- Frames
- Waypoints

If the values of these variables change while the program is running, their live values is shown.

To access Global Variables

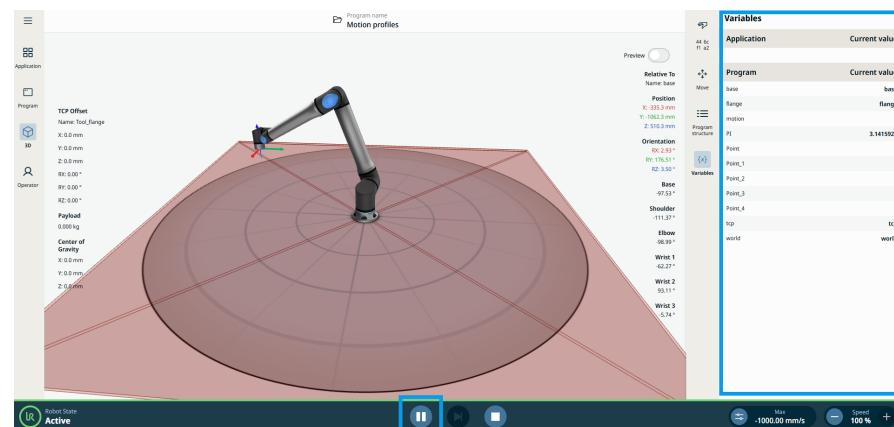
1. You can choose to view the main screen in **3D** view or **Operator** screen.
2. Tap **Global Variables** in the sidebar.

You see that the **Name** and **Value** columns are blank.



3. Tap the **Play** button in the footer and view the variable names and values.

The number of names and values depend on the program you created and how many commands you used.





7. Programming Reference

Description	What's in this chapter
	• Program Structure • Program Menu • Modules and Functions • Command Nodes • Move Command Nodes • Logic Command Nodes • Structure Command Nodes • Tool Command Nodes • Value-Variable-Expression Tabs • Frames • Grids Application • OptiMove

7.1. Program Structure

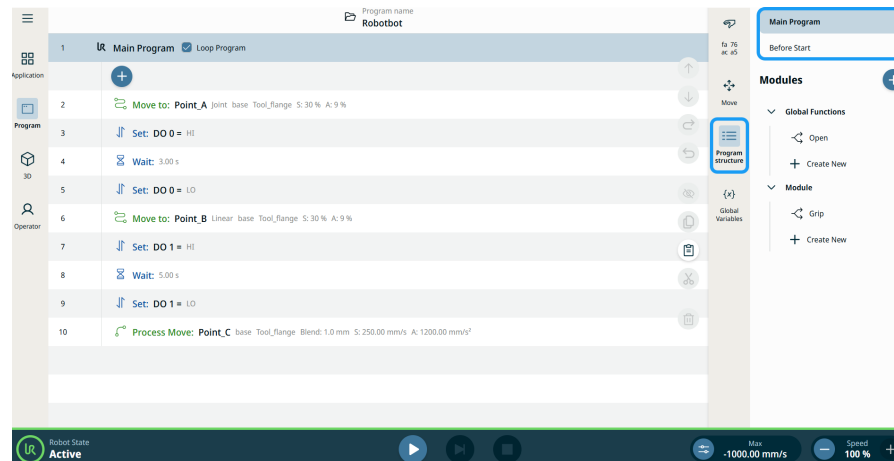
Description	The Program Structure menu lists the order and functions to create robot programs.
-------------	---

To access Program Structure

Tap the **Program Structure** menu.

Access to two tabs related to the program menu is provided:

- Main Program
- Before Start



Tap either Main Program or Before Start, and you can see the corresponding responses in the main screen.

Select **Before Start** to add instructions or program nodes that are executed before the main program starts. Variables created in creating program are always done under **Before Start**.

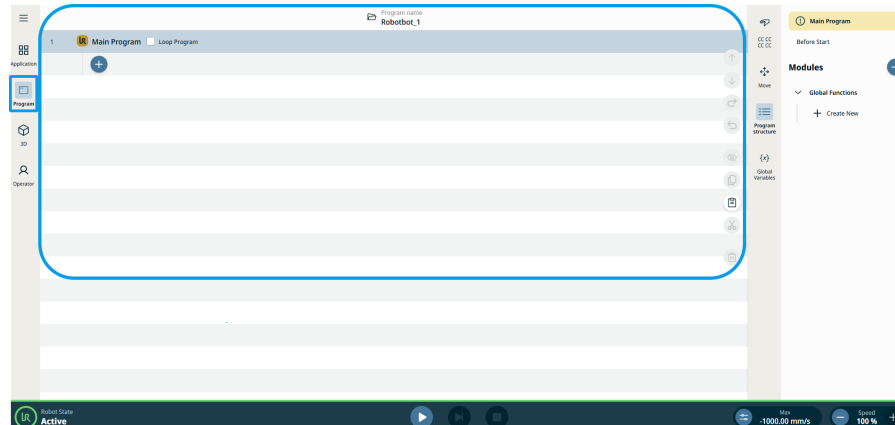
7.2. Program Menu

Description The **Program** menu enables you to create and modify programs for the robot.

To access Main Program screen

When you tap the Program icon, the Main Program screen becomes the main screen. The active program is seen in it, along with the details of the command nodes used in the program tree.

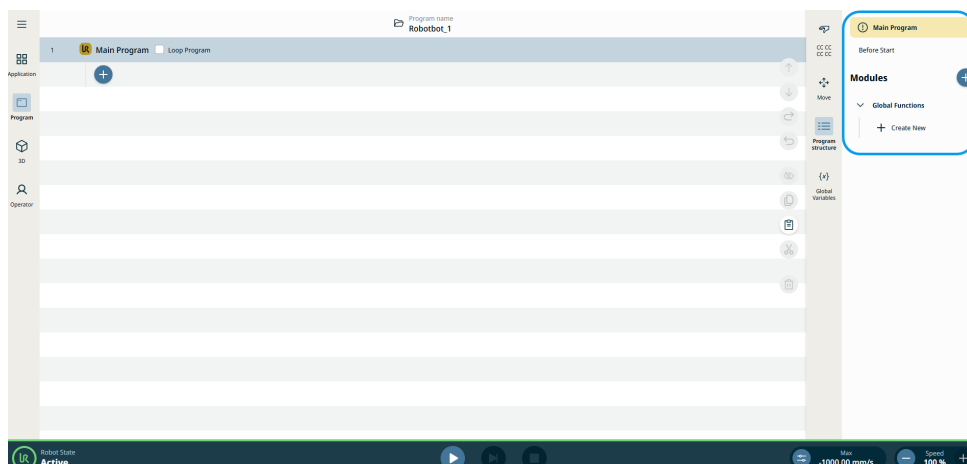
1. Tap the Program icon  on the main navigation.
2. You can see the **Main Program** setting in the main screen.



3. Tap the add icon . The command and clipboard toolboxes appear. The command toolbox is the default setting.
4. In the commands toolbox, tap the command node you want to use.
5. Tap the clipboard toolbox to view your history of program creation.

To access Main Program control tabs

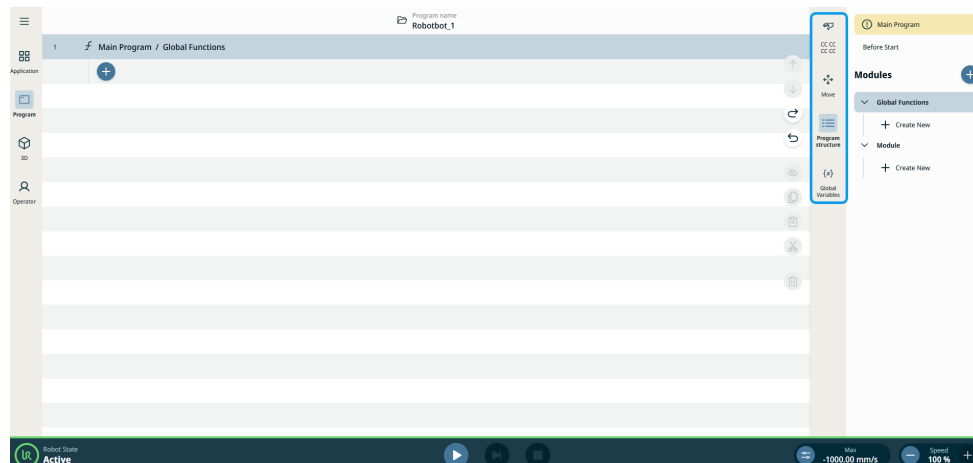
You can see the control tabs of the Main Program at the right side of the screen.



1. Tap the two tabs above **Modules**, and you can see the corresponding responses in the Main Program setting.
2. Tap the Modules **add icon** to create a module.
3. Under Global Functions, tap **+ create new** to add a new function.

To access Sidebar

The Sidebar contains **Safety Checksum**, **Move**, **Program structure**, and **Global variables**.



1. Tap  to access **Safety Checksum**.
2. Tap **Move** to program robot movement, joints, and rotation.
3. Tap **Program structure** to show or hide the control tab of the Main Program setting.
4. Tap **Global Variables** to view variables of global nature.

7.3. Modules and Functions

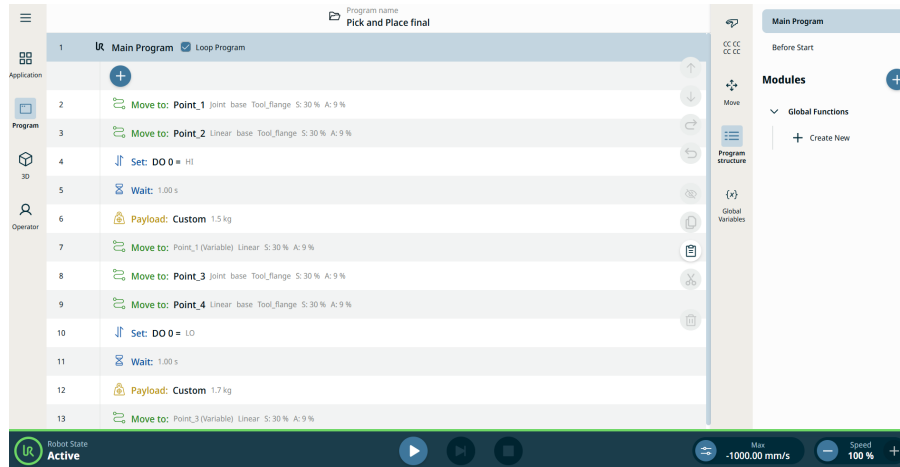
Description Modules and Functions is a feature of PolyScope X that will let you reuse functionality, increase readability, and make the program more maintainable, as you only have to adjust a single function to affect the entire program. Modules and functions are designed to be small and independent program blocks that make it easier for you to debug, reuse, and navigate. The example that follows uses the created [Pick and Place](#) program.

7.3.1. Gripper Module

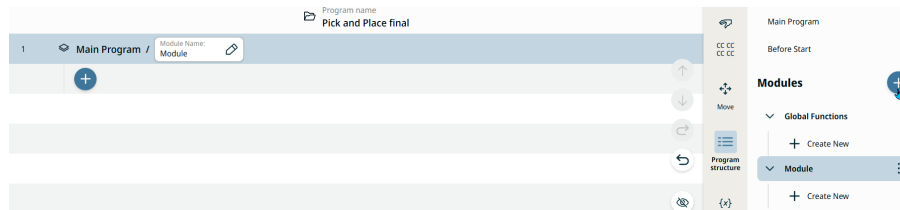
Description Grippers are devices that enable robots to pick up and hold objects. When combined with the robot arm, grippers enable you to automate processes, such as inspection, assembly, pick and place, and machine tending. Similar to the human hand, grippers are positioned at the end of the arm, functioning as end effector, and their abilities allow you to combine the strength of an arm with the dexterity of a hand. When you create a gripper module, you optimize ease, readability, and flexibility in using its appropriate command nodes.

To create Gripper module

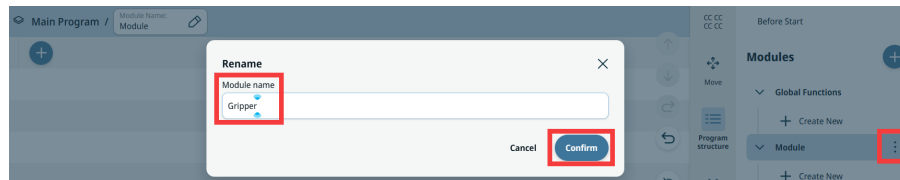
1. Open the created pick-and-place program in the System Manager. In the main screen, you see all the used commands in the program tree.



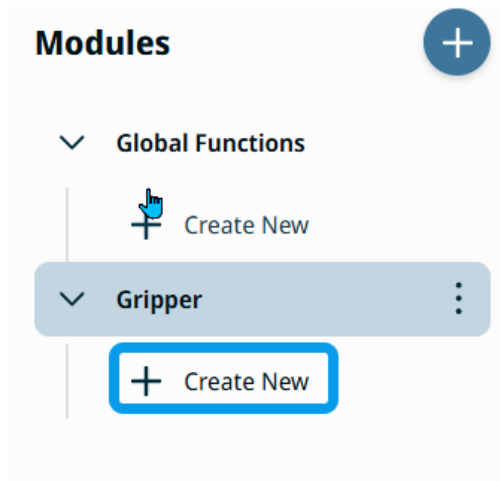
2. Make a new module which will contain the functions for Gripper. Tap the add icon of the modules tree in the multitask screen. A module is inserted below Global Functions, and the main screen is depopulated of the created commands.



3. Tap the Module kebab icon, rename the module name as Gripper, and Confirm.



- Below the created Gripper module, tap + Create New.

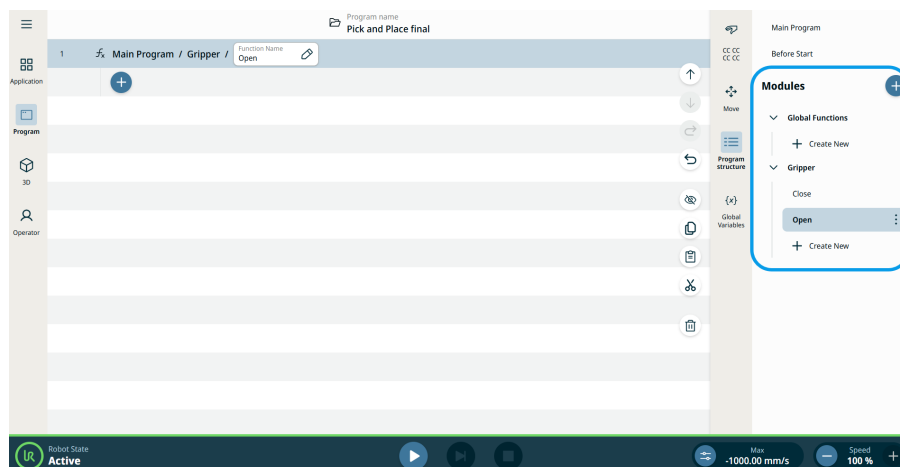


- In the main screen, tap the Function Name field in the Main Program tree, create name as Close, and Confirm.

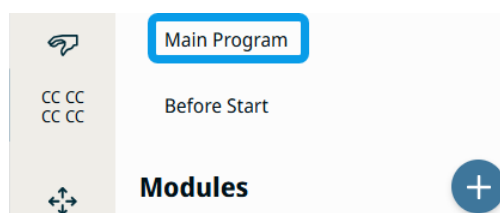


- In the multitask screen, tap + Create New below the newly created Close function of Gripper.
- In the main screen, tap the Function Name field in the Main Program tree, name it as Open, and Confirm.

This is how PolyScope X interface appears in your screen. Take particular notice on the created Modules tree in the multitask screen.



- Tap Main Program in the topmost part of the multitask screen. All used commands become visible again in the main screen.

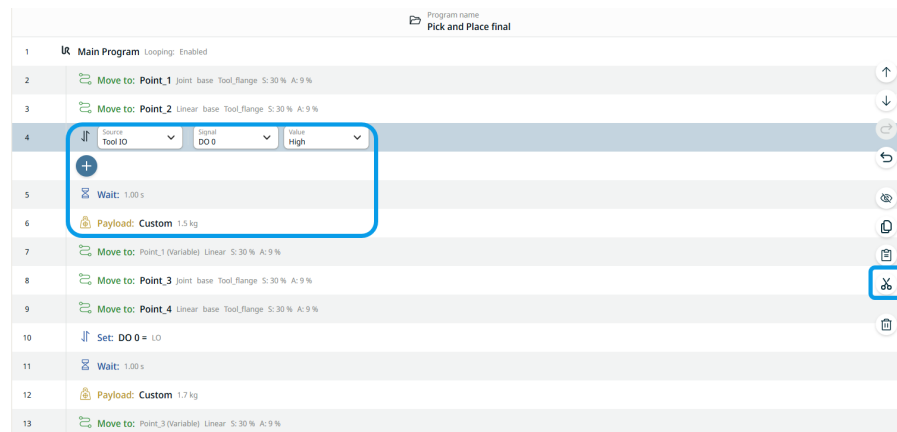


Gripper Close Function

Description A gripper close function ensures that right timing, precise force, and correct payload are applied.

To create Close Gripper Function

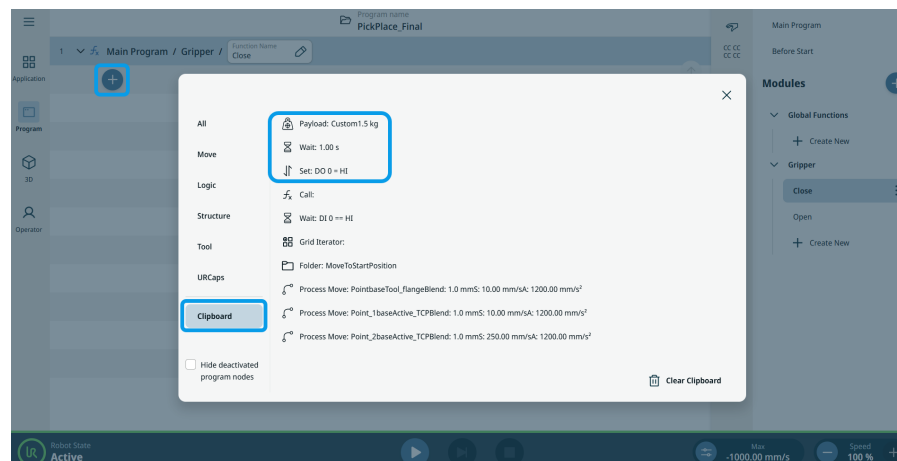
1. Cut out the first set of Set, Wait, Payload commands.



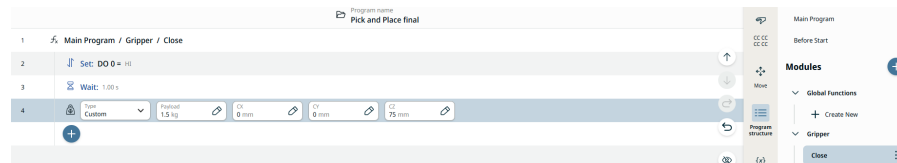
2. Tap the Set command node and tap the cut icon found on the right side of the main screen.
3. Tap the Wait command node and tap the cut icon.
4. Tap the Payload command and tap the cut icon.

These three commands are transferred in the clipboard toolbox.

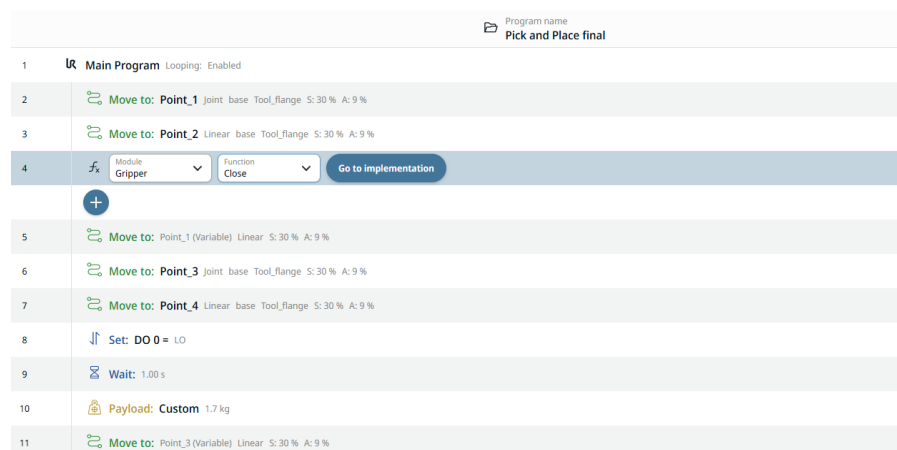
5. In the multitask screen, tap the Close function of Gripper.
6. Tap the add icon in the main screen and choose the Clipboard toolbox.



7. In sequential order, insert the Set, Wait, Payload commands into the Close Gripper function.
 - 7.1. Tap the Set command.
 - 7.2. Tap the add icon below.
 - 7.3. Tap the Wait command.
 - 7.4. Tap the add icon below.
 - 7.5. Tap the Payload command.



8. Tap again the Main Program to insert the Gripper Close Function into the program tree.
9. Tap the command node **Move to: Point_2** and then the add icon.
10. Choose the Commands toolbox. Tap the Call icon, and a node with the Module and Function fields appear.
11. Select Gripper in the Module field and Close in the Function field.



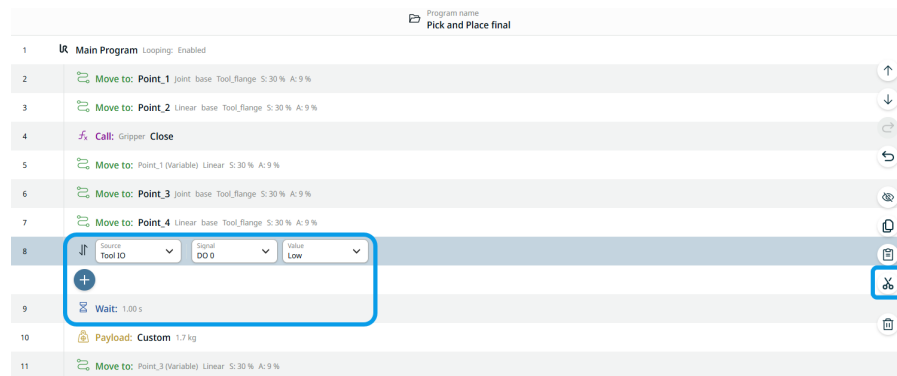
You are able to program a Gripper Close Function once, and whenever you need to close the gripper, just call the function.

Gripper Open Function

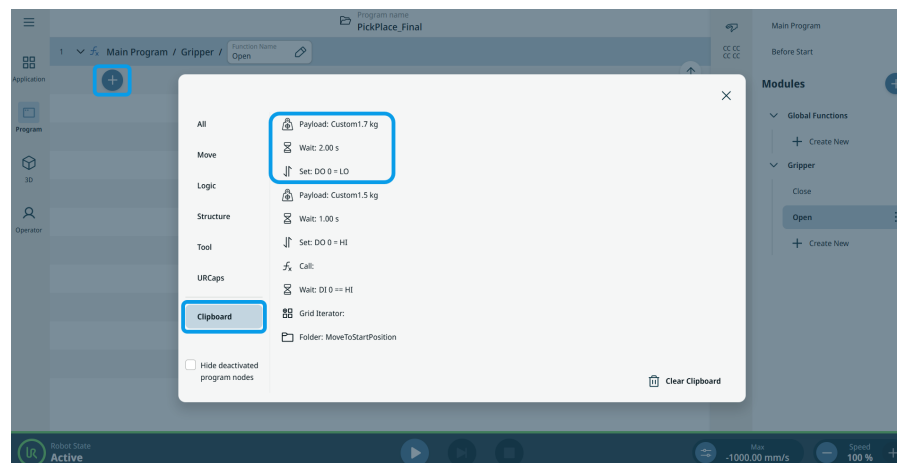
Description A gripper open function ensures that right timing to open the gripper is applied.

To create Open Gripper Function

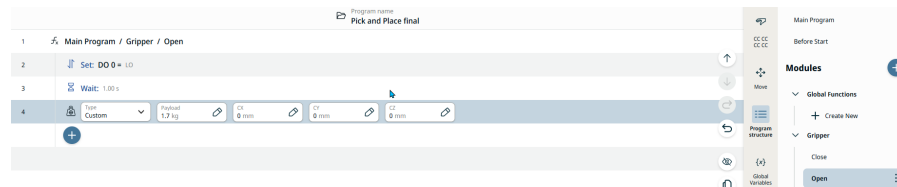
1. Cut out the second set of Set, Wait, Payload commands.



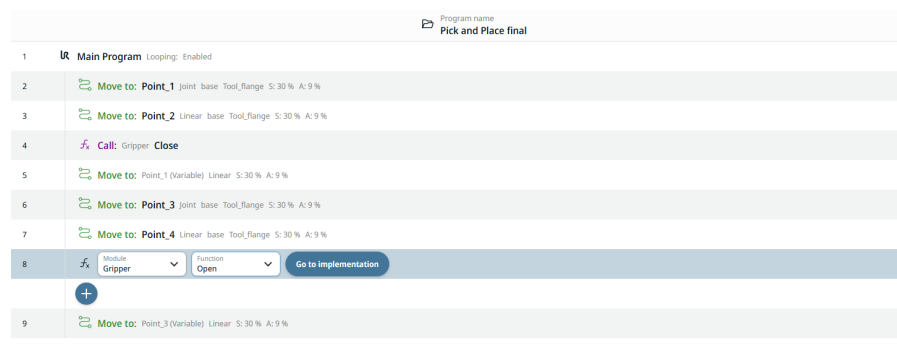
2. Tap the Set command node and tap the cut icon found on the right side of the main screen.
3. Tap the Wait command node and tap the cut icon.
4. Tap the Payload command and tap the cut icon.
These three commands are transferred in the clipboard toolbox.
5. In the multitask screen, tap the Open function of Gripper.
6. Tap the add icon in the main screen and choose the Clipboard toolbox.



7. In sequential order, insert the second Set, Wait, Payload commands into the Open Gripper function.
 - 7.1. Tap the Set command.
 - 7.2. Tap the add icon below.
 - 7.3. Tap the Wait command.
 - 7.4. Tap the add icon below.
 - 7.5. Tap the Payload command.

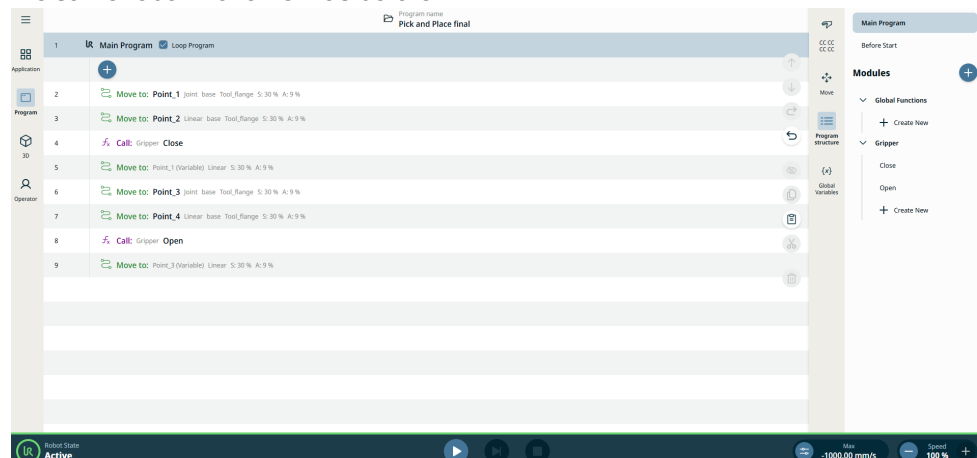


8. Tap again the Main Program to insert the Gripper Open Function into the program tree.
9. Tap the command node **Move to: Point_4** and then the add icon.
10. Choose the Commands toolbox. Tap the Call icon, and a node with the Module and Function fields appear.
11. Select Gripper in the Module field and Open in the Function field.



You are able to program a Gripper Open Function once, and whenever you need to open the gripper, just call the function.

This is now how your program tree appears with the applied module and functions, but with the same robot movement as before:

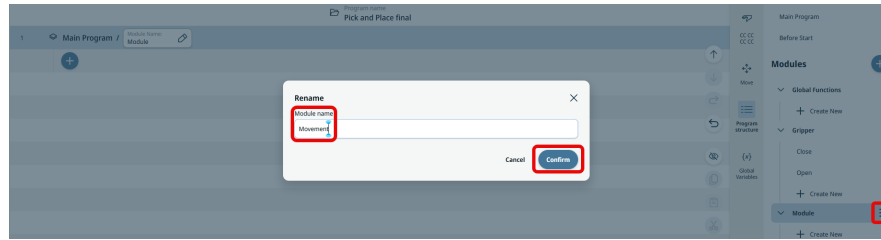


7.3.2. Movement Module

Description A movement module enables you to reuse [7.5 Move Command Nodes on page 155](#) commands to pick and place an object from one place to another using a gripper.

To create Movement Module

1. Tap the add icon of the modules tree in the multitask screen. A module is inserted below Gripper module.
2. Tap the Module kebab icon, rename the module name as Movement, and Confirm.

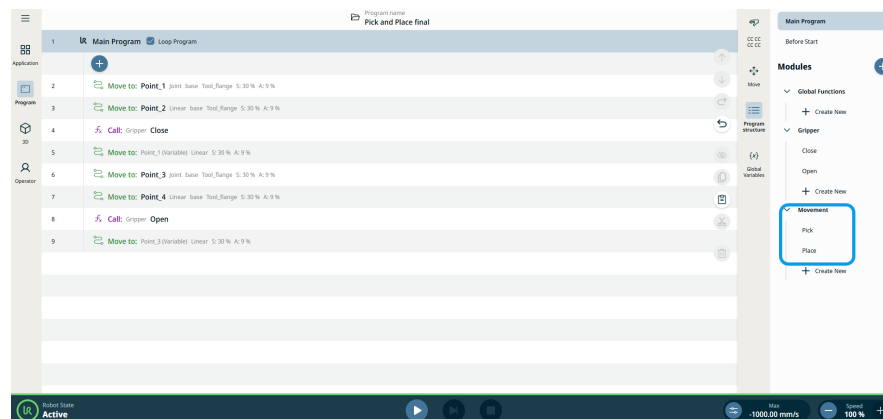


3. Below the created Movement module, tap + Create New.
4. In the main screen, tap the Function Name field in the Main Program tree, create name as Pick, and Confirm.



5. In the multitask screen, tap + Create New below the newly created Pick function of Movement.
6. In the main screen, tap the Function Name field in the Main Program tree, name it as Place, and Confirm.
7. Tap Main Program in the topmost part of the multitask screen. All used commands become visible again in the main screen.

This is how PolyScope X interface appears in your screen.

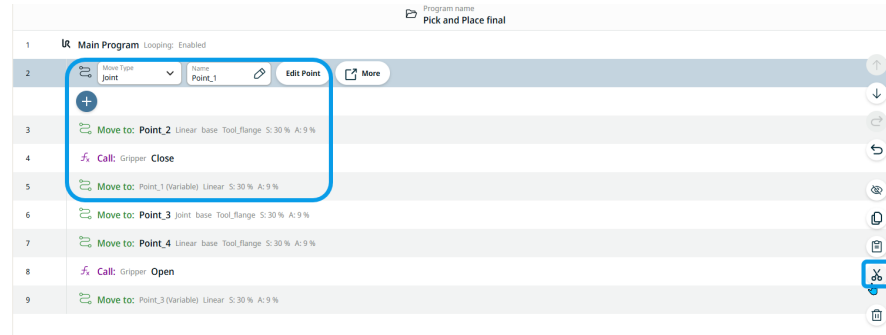


Pick Movement Function

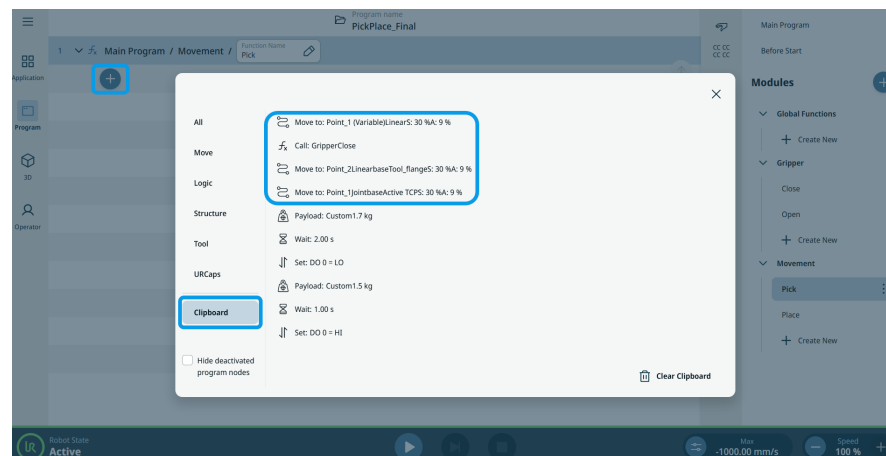
Description A pick function uses the move command in the direction of the object to be picked and the call command to close the gripper.

To create Pick Movement Function

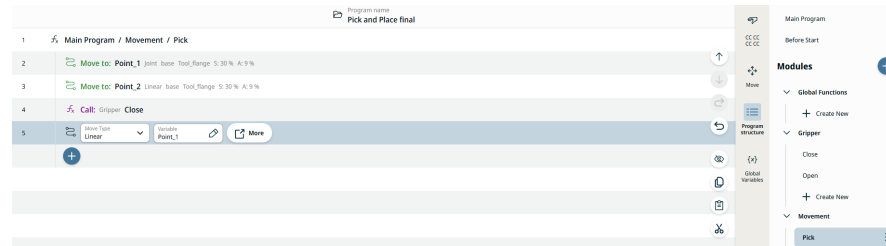
1. Cut out the first set of Move to commands, including the Call command for gripping.



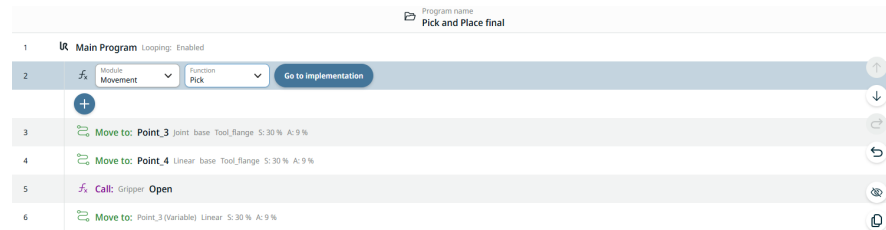
2. In alternate order, tap each command nodes and the cut icon one after the other. The four commands are transferred in the clipboard toolbox.
3. In the multitask screen, tap the Pick function of Movement.
4. Tap the add icon in the main screen and choose the Clipboard toolbox.



5. In sequential order, insert the first set of commands into the Pick Movement function.
 - 5.1. Tap the Move to: Point_1 command.
 - 5.2. Tap the add icon below.
 - 5.3. Tap the Move to: Point_2 command.
 - 5.4. Tap the add icon below.
 - 5.5. Tap the Gripper Close function.
 - 5.6. Tap the add icon below.
 - 5.7. Tap the Move to: Point_1 (Variable) command.



6. Tap again the Main Program to insert the Movement Pick Function into the program tree.
7. Tap the add icon, choose the Commands toolbox, and tap the Call icon. A node with the Module and Function fields appear.
8. Select Movement in the Module field and Pick in the Function field.

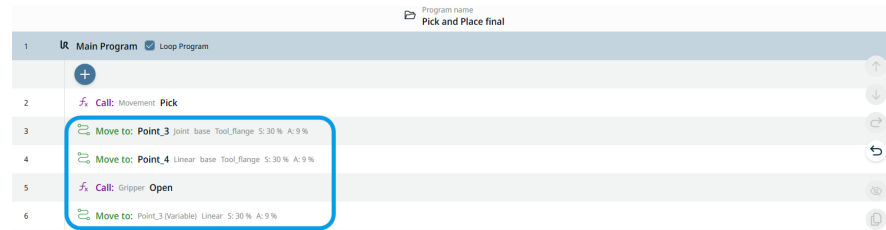


Place Movement Function

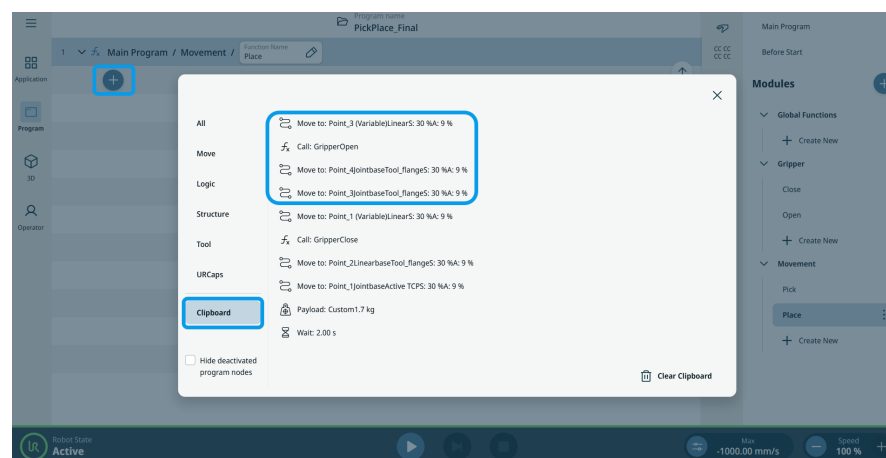
Description A place function uses the move command in the direction of the place where object is to be put and the call command to open the gripper.

To create Place Movement Function

1. Cut out the second set of Move to commands, including the Call command for gripping.

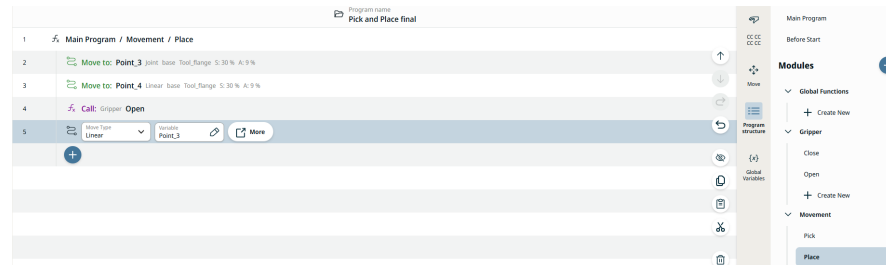


2. In alternate order, tap each command nodes and the cut icon one after the other. These four commands are transferred in the clipboard toolbox.
3. In the multitask screen, tap the Place function of Movement.
4. Tap the add icon in the main screen and choose the Clipboard toolbox.



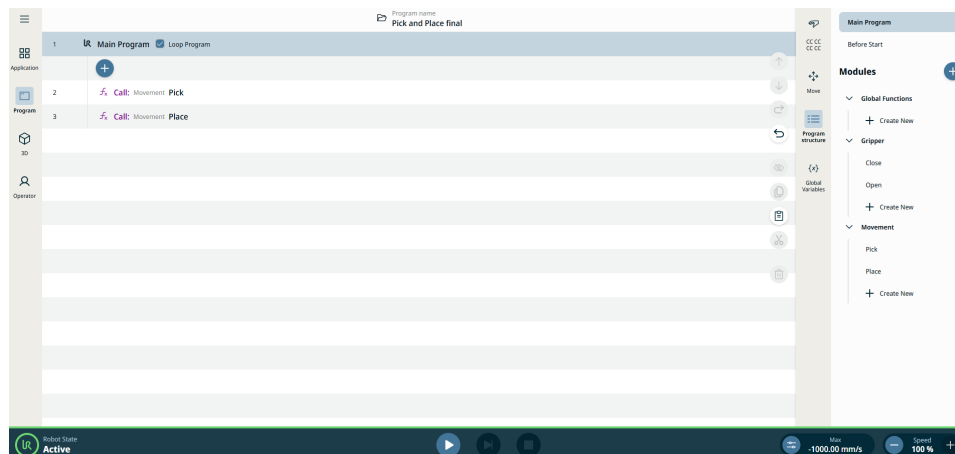
5. In sequential order, insert the first set of commands into the Place Movement function.
 - 5.1. Tap the Move to: Point_3 command.
 - 5.2. Tap the add icon below.
 - 5.3. Tap the Move to: Point_4 command.
 - 5.4. Tap the add icon below.
 - 5.5. Tap the Gripper Open function.
 - 5.5. Tap the add icon below.
 - 5.6. Tap the Move to: Point_3 (Variable) command.

6. Tap again the Main Program to insert the Movement Place Function into the program tree.



7. Tap the command node **Call: Movement Pick** and then the add icon.
8. Choose the Commands toolbox. Tap the Call icon, and a node with the Module and Function fields appear.
9. Select Movement in the Module field and Place in the Function field.

You can see the program tree of the PolyScope X interface when structured in functions and modules:



In structuring your program by functions and modules, it is possible to call functions within functions to make everything even more reusable. Creating a full program mainly comprised of functions will make it easier for you to manage long programs. Furthermore, you make the program more maintainable and avoid repetition in your programming. As you can see on the 3D viewer and Operator screen, the robot movements are still the same as before.

7.4. Command Nodes

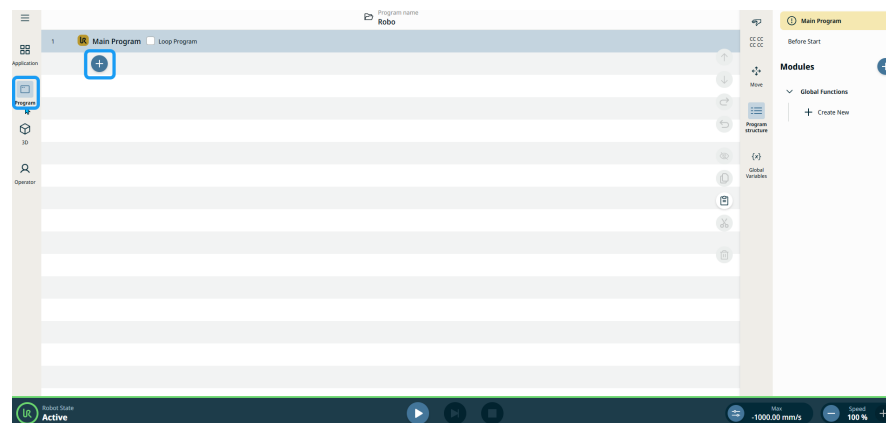
Description The PolyScope X command nodes contain the commands that robot users apply to create a program in the simplest way.

The nodes are categorized and colored to provide the following benefits:

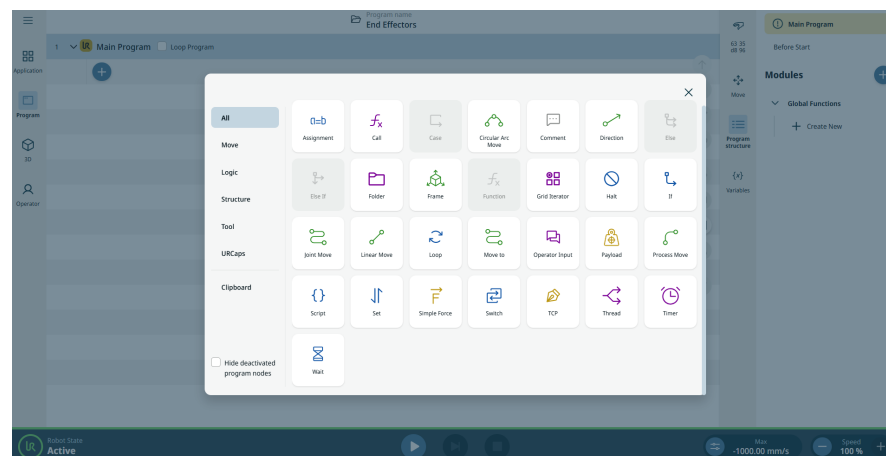
- Provide immediate identification with other nodes of similar properties.
- Highlight relationships and hierarchies in the program structure.
- Track the flow of program execution and identify issues when detecting an error.
- Indicate where the program stops without having to go through the entire program.

To access Command nodes

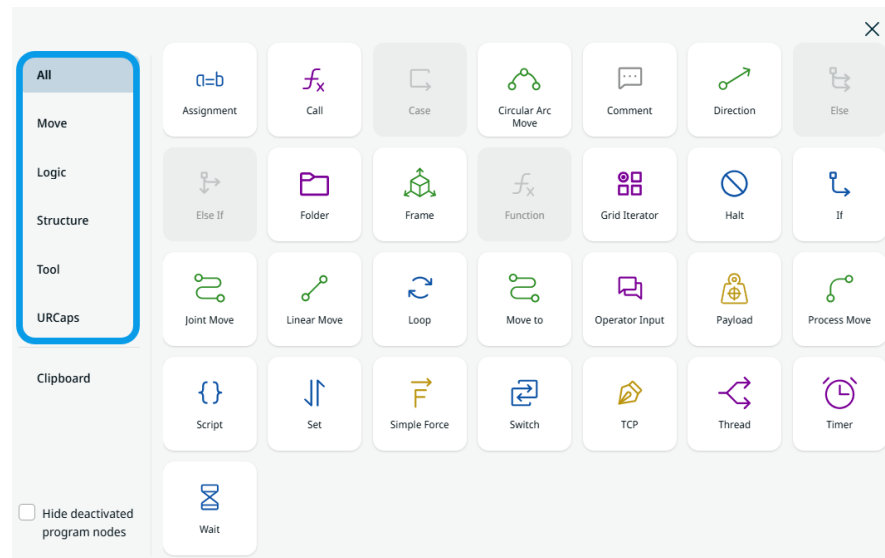
1. Tap **Program** on the main navigation, and then the **Add** icon in the main screen.



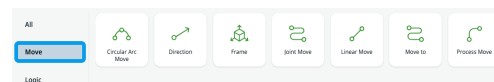
The **Commands** toolbox appears as the default screen, which contains all the command nodes.



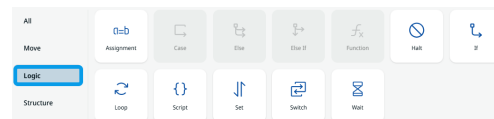
- On the left-hand side of the screen, you have the option to view the commands by category. By default, **All** command nodes are shown.



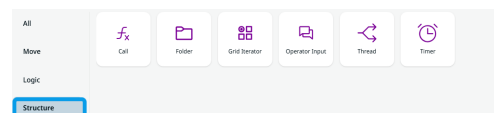
- Tap **Move**, and the seven move command nodes are shown.



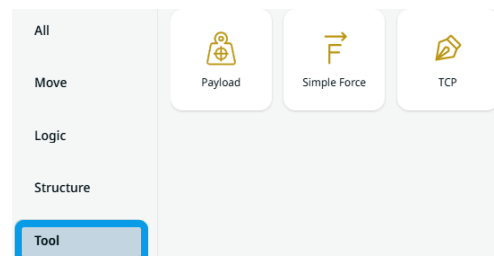
- Tap **Logic**, and the twelve logic command nodes are shown, including the inactive ones.



- Tap **Structure**, and the six structure command nodes are shown.

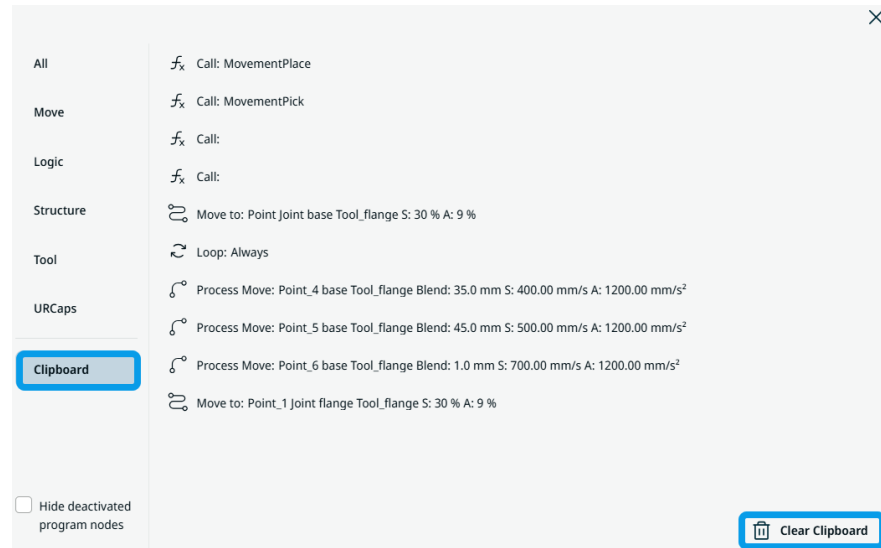


- Tap **Tool**, and the three tool command nodes are shown.



- When you have URCaps added in the System Manager, these become visible when you tap **URCaps** after **Tool**.

8. Tap the **Clipboard** toolbox, and you can see and review the history of cut-and-copied commands used. You also have the option to clear the clipboard at the bottom right side of the screen.



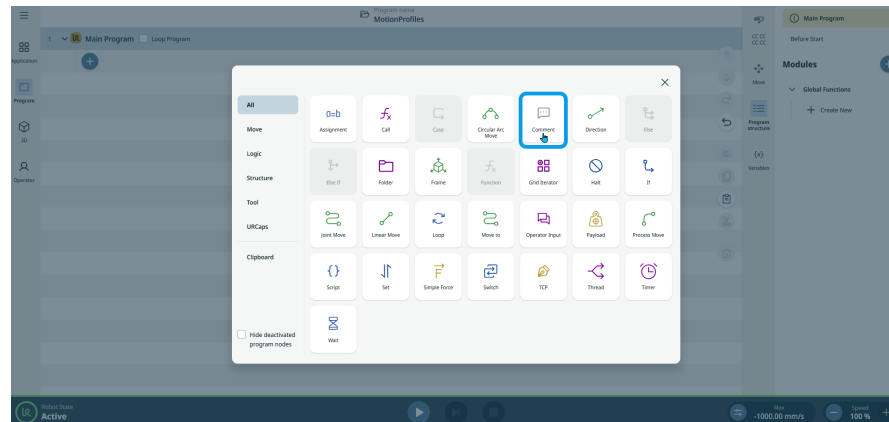
9. If you prefer to view only the active command nodes, tap **Hide deactivated program nodes**.

7.4.1. Comment

Description The **Comment** command allows you to add text descriptions into your program and so enables allows you to keep track of decisions you make while you create or update your robot program.

To access Comment command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Comment** icon in the Commands toolbox.
An editable **Comment** field is inserted into the program tree.



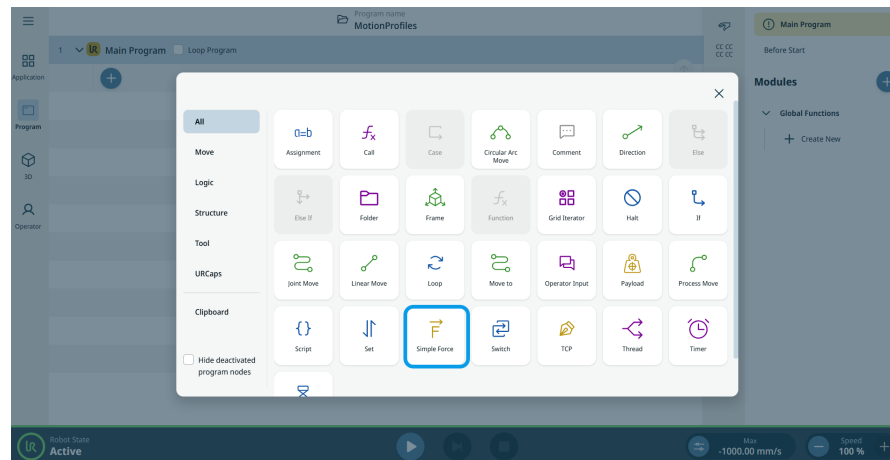
3. Tap the Comment field.
4. Create your comment.
Note: A maximum of 1,000 characters is allowed.
5. Tap **Confirm**.

7.4.2. Simple Force

Description The **Simple Force** command enables you to use only one required axis for a desired force. The force along this axis is adjustable and is always applied along the z-axis of the selected feature.

To access Simple Force command

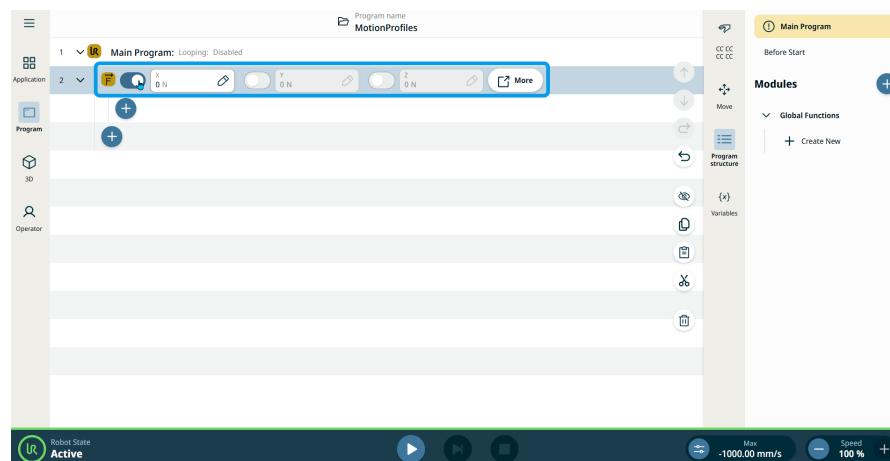
1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Simple Force** icon in the Commands toolbox.



A node is inserted into the program tree. The node has three editable fields and a **More** button for advanced options.

The three fields are **X**, **Y**, and **Z**. These are inactive by default.

Each field has a slider button on the left side to switch activate or deactivate.



3. Tap the slider button to the right to activate the three fields.

Three tabs to input the coordinates are available:

- Value
- Variable
- Expression

Note: The Variable tab is inactive.

4. On the **Value** tab, enter your preferred value and **Confirm**.
5. Tap the **Expression** tab and enter the value and **Confirm**.

See [Value-Variable-Expression Tabs](#).

**To use
More
options**

1. Tap the **More** button.

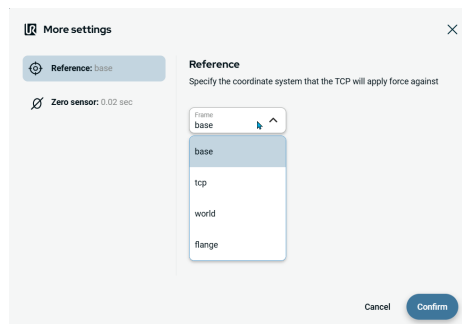
A new screen appears containing two advanced settings:

- Reference
- Zero sensor

2. Tap **Reference** on the left side.

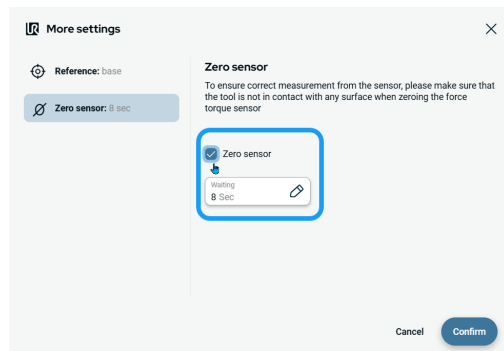
A new **Frame** field appears in the right side, containing four selections:

- base
- tcp
- world
- flange



3. Select your chosen frame of your reference and **Confirm**.

4. Tap **Zero sensor** on the left side.



5. Tap the Zero sensor **check box** to activate or deactivate the setting.
6. When Zero sensor is activated, tap the **Waiting** field below.
7. Enter the waiting **Value** in seconds.
8. Tap **Confirm**.

7.5. Move Command Nodes

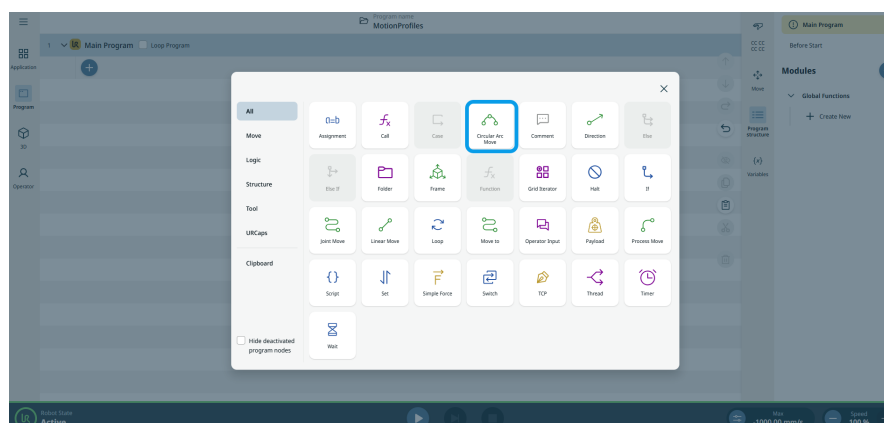
Description You use the different Move nodes in PolyScope X to define robot motion and tool paths, selecting joint motion for efficient point-to-point moves or linear motion when path accuracy and tool orientation are critical.

7.5.1. Circular Arc Move

Description The **Circular Arc Move** enables you to program the robot to move in a circular motion while keeping the tool orientation either fixed or unconstrained.

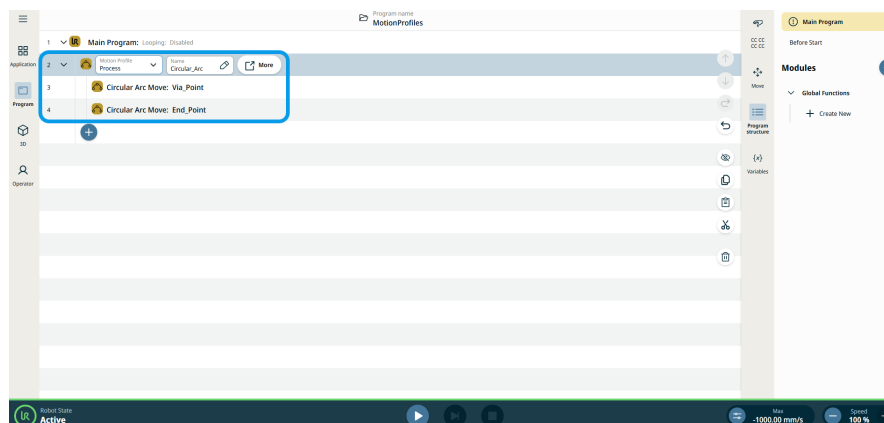
To access Circular Arc Move command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Circular Arc Move** icon in the Commands toolbox.



A parent node and two child nodes are inserted into the program tree.

- First child node = Circular Arc Move: Via_Point
- Second child node = Circular Arc Move: End_point



The node has three editable fields and a **More** button for advanced options.



The three editable fields are:

- **Name**
A name for referencing the node.
- **Speed**
The robot speed during the move.
- **Acceleration**
The robot acceleration during the move.

3. Tap the **Name** field, and three tabs appear:
 - Value
 - Variable
 - Expression
4. Enter the data in each tab and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
5. Tap the **Speed** field, and three tabs appear:
 - Value
 - Variable
 - Expression
6. Repeat step 4.
7. Tap the **Acceleration** field, and three tabs appear:
 - Value
 - Variable
 - Expression
8. Repeat step 4.

Motion Profile labels on Move Nodes

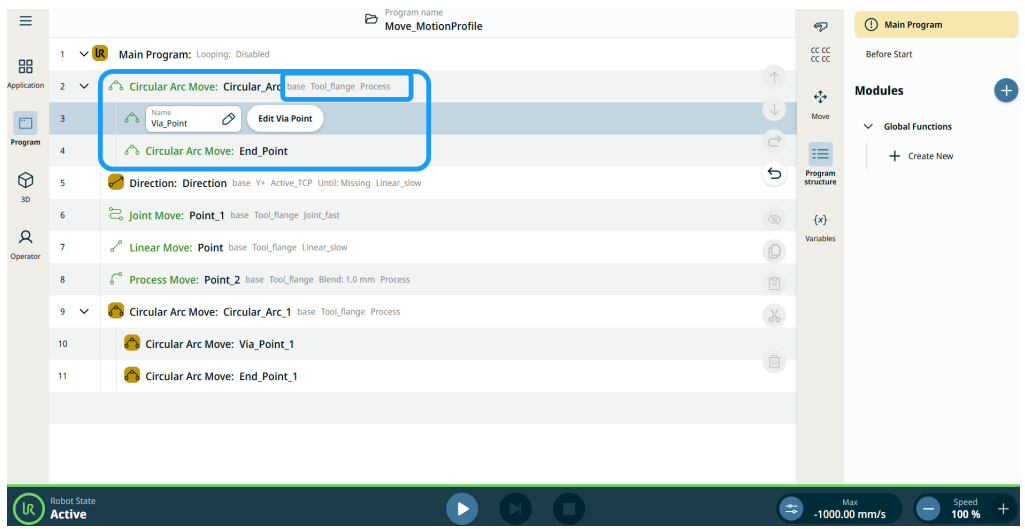
All **Move-related command nodes** in the program tree display the **Motion Profile** associated with each node. This feature enables users to see the selected Motion Profile on a Move node. This feature is implemented on the following command nodes:

- [Linear Move](#)
- [Joint Move](#)
- [Process Move](#)
- [Direction](#)
- [7.5.1 Circular Arc Move on the previous page](#)



NOTICE

Move nodes with Custom motion profile will display the values for the speed and acceleration instead of Motion profile.

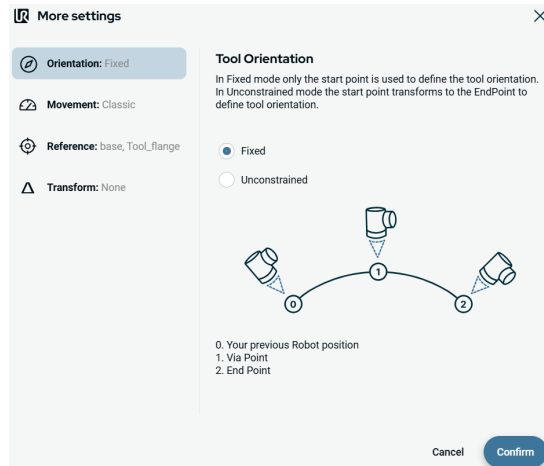




To access More settings

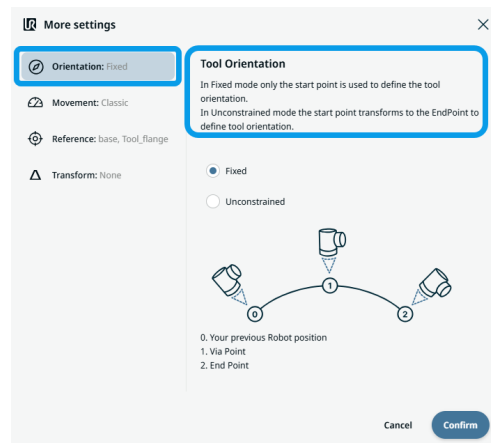
Tap the **More** button, and the **More settings** screen appears.
On the left side, you see four advanced options:

- Orientation
- Movement
- Reference
- Transform



To use Orientation

1. Tap **Orientation**.



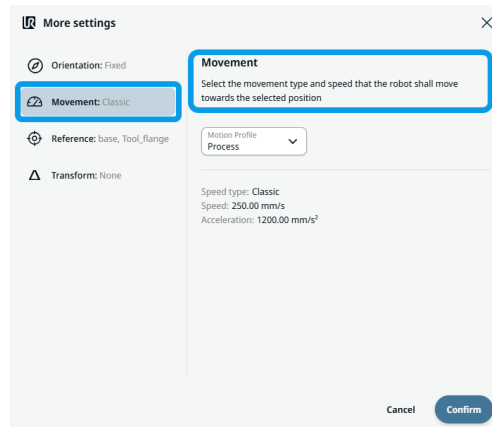
On the right side, two buttons are provided:

- Fixed
 - Unconstrained
2. Tap **Fixed** when you want the start point only to be used to define the tool orientation, and **Confirm**.
 3. Tap **Unconstrained** when you want the start point to transform to the endpoint to define tool orientation and **Confirm**.

Note: A diagram shows the difference between the two endpoints when you choose unconstrained.

To use Movement

1. Tap **Movement**.



Two fields appear:

- Speed
- Acceleration

2. Tap the **Speed** field, and three tabs appear:

- Value
- Variable
- Expression

3. Enter the data in each tab and tap **Confirm**. See [Value-Variable-Expression Tabs](#).

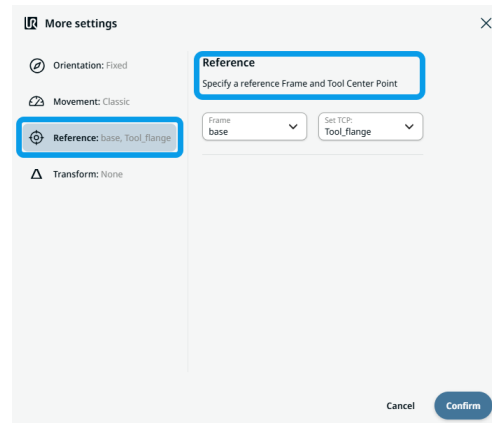
4. Tap the **Acceleration** field, and three tabs appear:

- Value
- Variable
- Expression

5. Enter the data in each tab and tap **Confirm**. See [Value-Variable-Expression Tabs](#).

To use Reference

1. Tap **Reference**, and two editable **Frame** and **TCP** fields appear.



Tap the **Frame** field, and four options are selectable:

- world
- base
- flange
- tcp

2. Tap the **TCP** field, and two options are selectable:

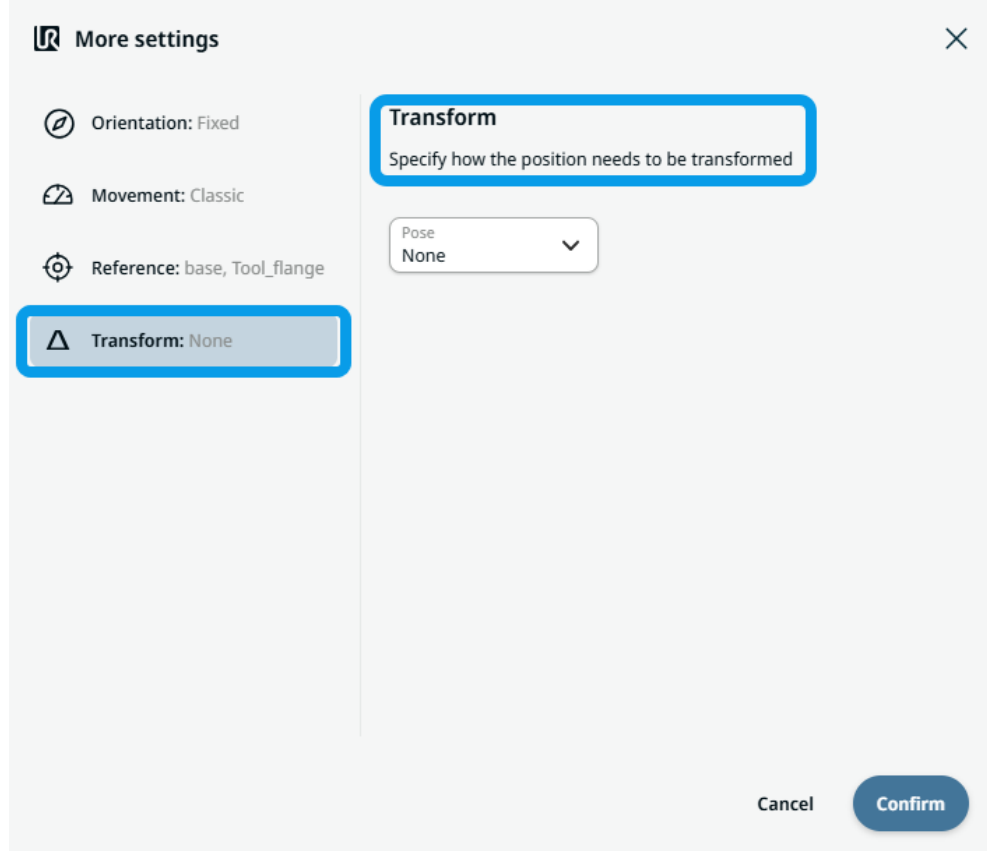
- Tool_flange
- Active_TCP

3. In the **Position** column, tap **X**, **Y**, **Z** fields to enter the value, and **Confirm**.
4. In the **Rotation** column, tap **RX**, **RY**, **RZ** fields to enter the value, and tap **Confirm**.

To use Transform

1. Tap **Transform**.

Tap the **Pose** field to specify how the position needs to be transformed and **Confirm**.

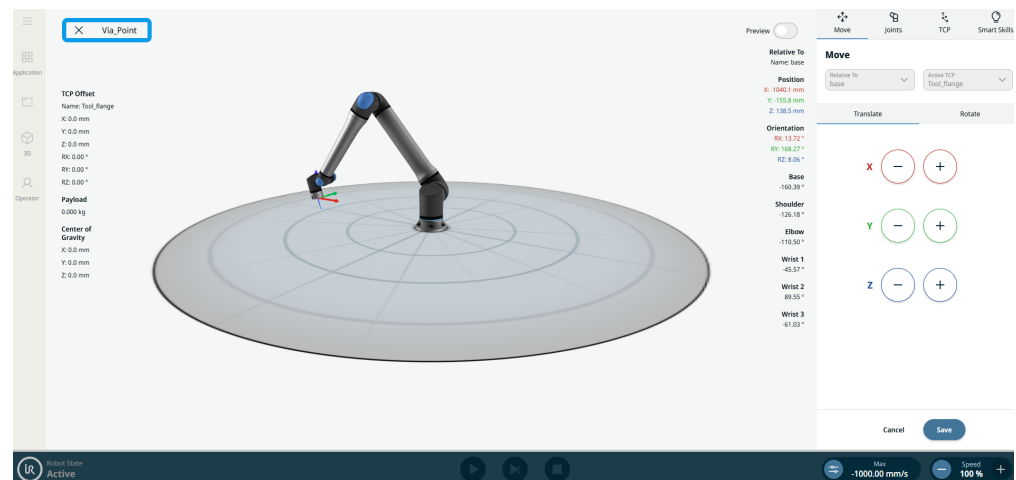


- 2.

To use Circular Arc Move: Via_ point

1. Tap the **Circular Arc Move: Via_Point** node and two fields appear:
 - Name
 - Teach Via Point
2. Tap the **Name** field, and three tabs appear:
 - Value
 - Variable
 - Expression
3. Enter the data in each tab and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
4. Tap **Teach Via Point**.

The 3D viewer screen of the robot appears. (See [Joint Move](#) and [7.5.5 Linear Move](#) on page 180 for details of use.)

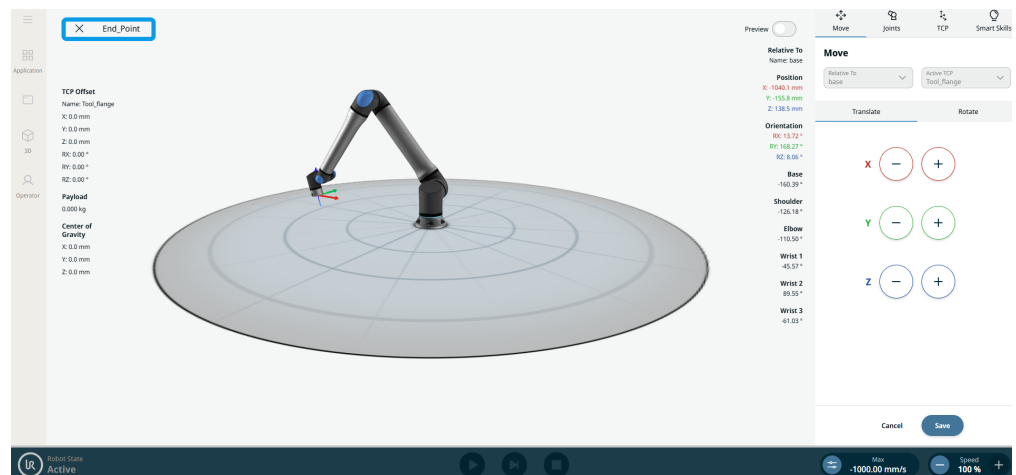




To use Circular Arc Move: End_ Point

1. Tap the **Circular Arc Move: End_Point** node, and three fields appear:
 - Name
 - Blend
 - Teach Via Point
2. Tap the **Name** field, and three tabs appear:
 - Value
 - Variable
 - Expression
3. Enter the data in each tab and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
4. Tap the **Blend** field, and three tabs appear:
 - Value
 - Variable
 - Expression
5. Repeat step 3.
6. Tap **Teach End Point**.

The 3D viewer screen of the robot appears. (See [Joint Move](#) and [7.5.5 Linear Move on page 180](#) for details of use.)

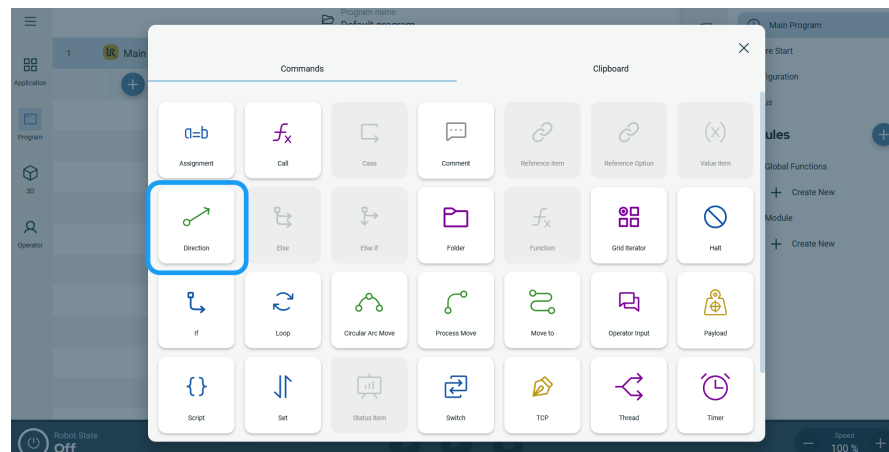


7.5.2. Direction

Description The **Direction** command enables you to move the robot in a specific direction in a linear motion along an axis. A stop condition halts it.

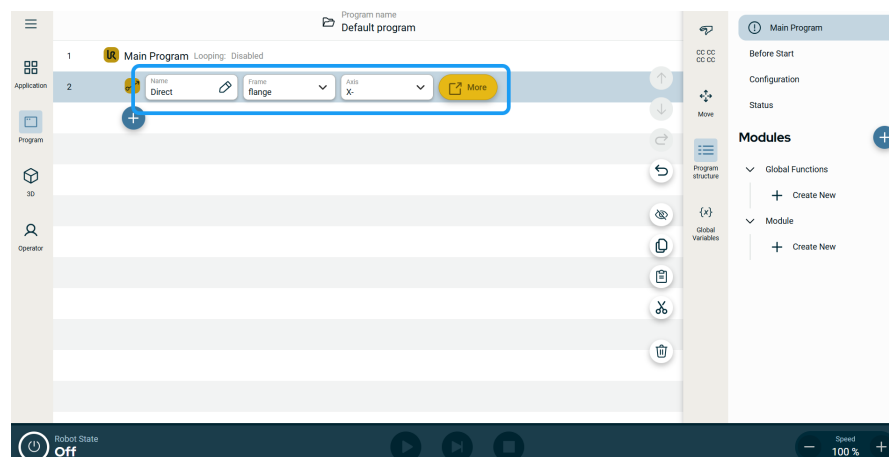
To access Direction command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Direction** icon in the Commands toolbox.



A node with three editable fields and a **More** button is inserted into the program tree. The three fields are:

- Name
- Frame
- Axis



3. Tap the **Name** field.
4. Enter the Value name in the tab and tap **Confirm**.



5. Tap the **Frame** field, and four choices are provided:

- world
- base
- flange
- tcp

6. Select your preferred frame choice.

7. Tap the **Axis** field, and the six axes are provided:

- X+
- X-
- Z+
- Z-
- Y+
- Y-

8. Tap your chosen axis.

Motion Profile labels on Move Nodes

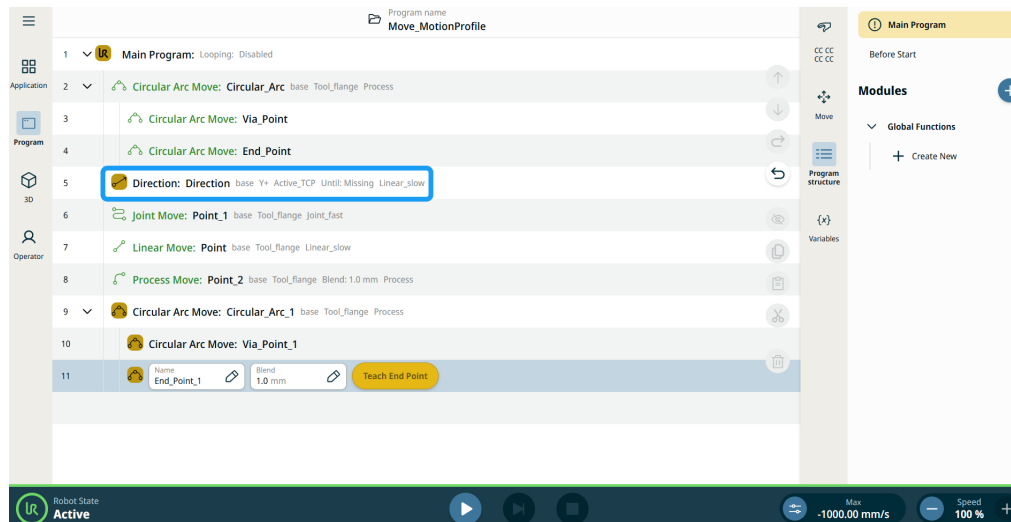
All **Move-related command nodes** in the program tree display the **Motion Profile** associated with each node. This feature enables users to see the selected Motion Profile on a Move node. This feature is implemented on the following command nodes:

- [Linear Move](#)
- [Joint Move](#)
- [Process Move](#)
- [Direction](#)
- [7.5.1 Circular Arc Move on page 155](#)



NOTICE

Move nodes with Custom motion profile will display the values for the speed and acceleration instead of Motion profile.

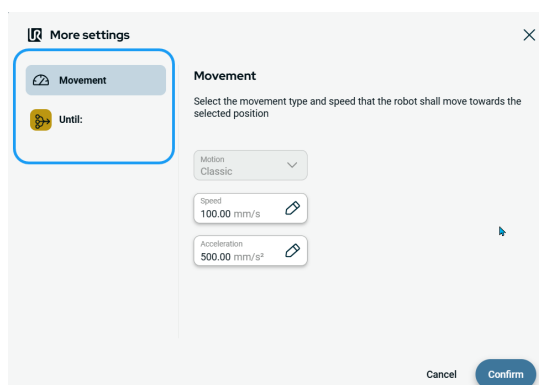


To use the
More
options

Tap the **More** button.

A new screen appears containing two advanced settings:

- Movement
- Until

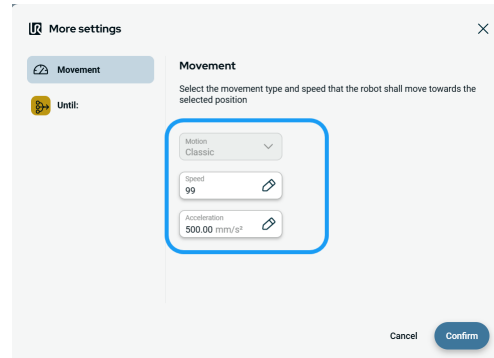


To use Movement setting

1. On the left side, tap **Movement**, and three fields appear:

- Motion
- Speed
- Acceleration

Note: The Motion field is inactive.



2. Tap the **Speed** field, and three tabs are available:

- Value
- Variable
- Expression

Note: The Variable tab is inactive.

3. Enter the data in each tab and tap **Confirm**. See [Value-Variable-Expression Tabs](#).

4. Tap the **Acceleration** field, and three tabs are available:

- Value
- Variable
- Expression

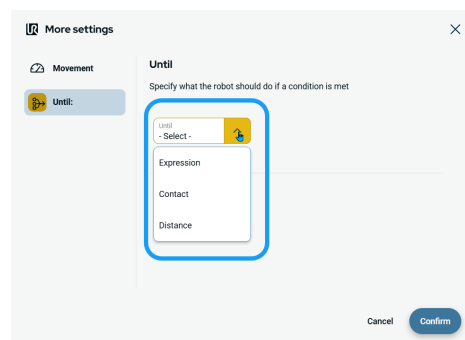
Note: the Variable tab is inactive.

5. Enter the data in each tab and tap **Confirm**. See [Value-Variable-Expression Tabs](#).

To use Until setting

1. On the left side, tap **Until**. An **Until** field appears, giving you three conditions to add:

- Expression
- Contact
- Distance

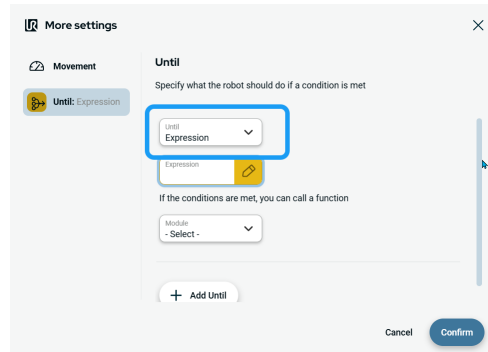


To access Expression

Tap **Expression**, and three functions are added:

- Expression field
- Module Select drop-down menu
- **Add Until** button

1.



Expression. Sets the expression to check against, and you can optionally call a function when the condition is met.

2. Tap the **Expression** field, enter your preferred expression, and **Confirm**.

3. Tap the **Module** field, and two selections are available:

- Global Functions
- Module

4. Select either **Global Functions** or **Module**.

An additional **Function** field appears to the right.

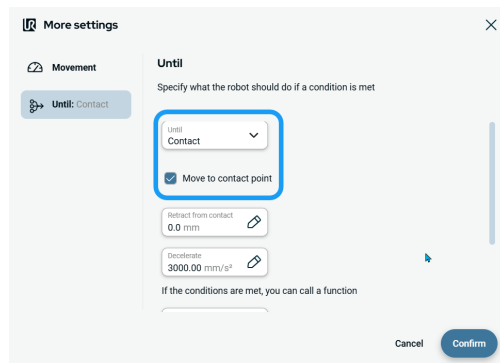
5. Tap **Add Until** button to add further conditions, and **Confirm**.

You can delete an Until condition by clicking the trash can, but the last condition cannot be deleted.

To access Contact

1. Tap **Contact**, and five functions are added:
 - Checkbox for **Move to contact point**
 - **Retract from contact** field
 - **Decelerate** field
 - **Module Select** field
 - **Add Until** button

Until contact checks for tool contact.



2. Check the **Move to contact point** box.
3. Tap the **Retract from contact** field, and three tabs are available:
 - Value
 - Variable
 - Expression

Note: The Variable tab is inactive.

4. Enter the data in each tab and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
Note: When you uncheck the **Move to contact point** box, the **Retract from contact** field becomes inactive.

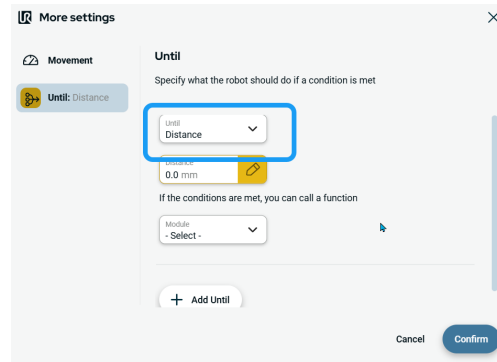
5. Tap the **Decelerate** field, and three tabs are available:
 - Value
 - Variable
 - Expression

Note: The Variable tab is inactive.

6. Enter the data in each tab and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
7. Tap the **Module Select** field, and two selections are available:
 - Global Functions
 - Module
8. Tap either **Global Functions** or **Module**.
An additional **Function** field appears to the right.
9. Tap **Add Until** button to add further conditions, and **Confirm**.

To access Distance

1. Tap **Distance**, and four functions are added:
 - **Distance** field
 - **Module** field
 - **Function Select** field
 - **Add Until** button



2. Tap the **Distance** field, and three tabs are available:
 - Value
 - Variable
 - Expression

Note: The Variable tab is inactive.

3. Enter the data in each tab and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
4. Tap either **Global Functions** or **Module**.
An additional **Function** field appears to the right.
5. Tap **Add Until** button if there is a need for you to use further Until function.
6. Tap **Add Until** button to add further conditions, and **Confirm**.

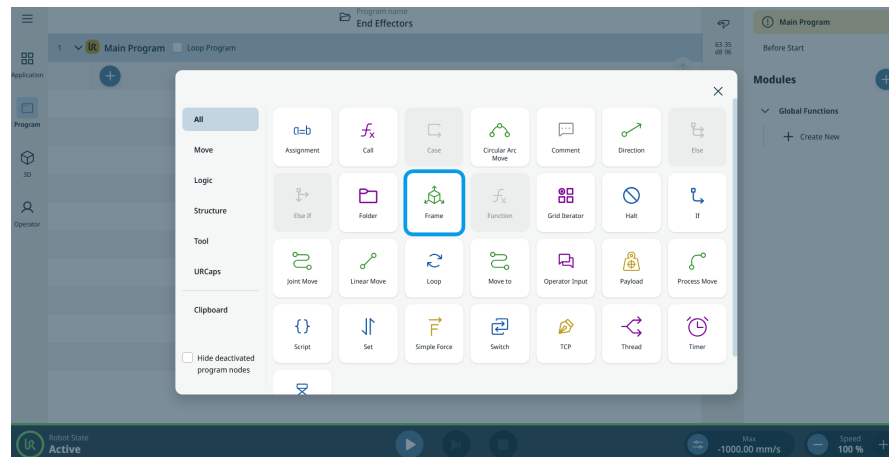
7.5.3. Frame

Description

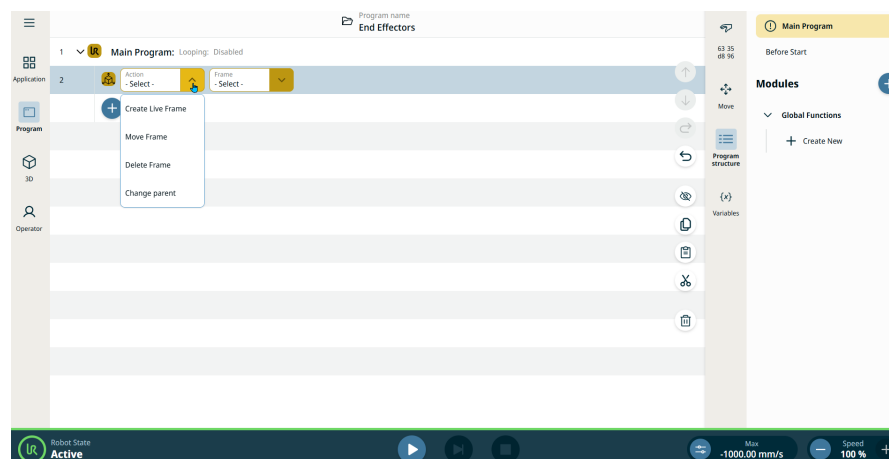
The **Frame** command node allows for the easy creation and modification of frames during the program execution. Frames created with this node are referred to as Live Frames to distinguish them from the frames created from the application section.

To access Frame command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Frame** icon in the Commands toolbox. A node with two editable fields is inserted into the program tree, Action and Frame.



3. Tap the **Action** field, and four choices are provided:
 - Create Live Frame
 - Move Frame
 - Delete Frame
 - Change Parent

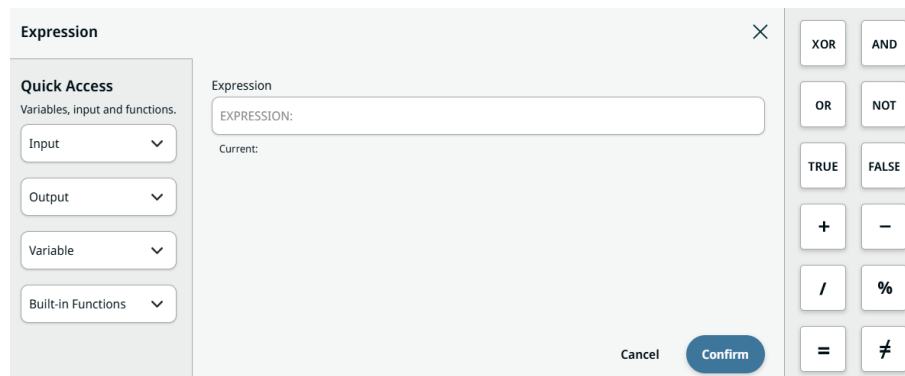


Create Live Frame



This action creates a new frame with the given name. The position of the frame can be specified from the position dialog, as well as the parent of the frame. The position can be given by either an X, Y, Z, RX, RY, RZ or by an expression.

1. Create your preferred frame name in the **Name** field.
2. Tap the **Position** tab, and further field and tabs are provided to choose from.
3. The **Reference** field enables you to select world, base, flange, and TCP.
4. In the default **Pose toolbox** you enter the **Position** in X, Y, Z and **Rotation** in RX, RY, RZ.
5. In the **Expression toolbox** , tap the **Expression field**, and related functions are provided such as the Quick Access panel of IO, Variable, and Built-in Functions.



Move Frame



This action moves an existing frame to a new position. The frame can be either a frame created in the application or a live frame. As with the creating live frame, the position can be specified in either coordinates or an expression.

When you select Position tab, follow the instructions 2-5 provided in “Create Live Frame.”

Delete Frame



This action deletes an existing frame. The frame can be one created in the application or a live frame. Predefined frames such as base and world cannot be deleted.

Change Parent



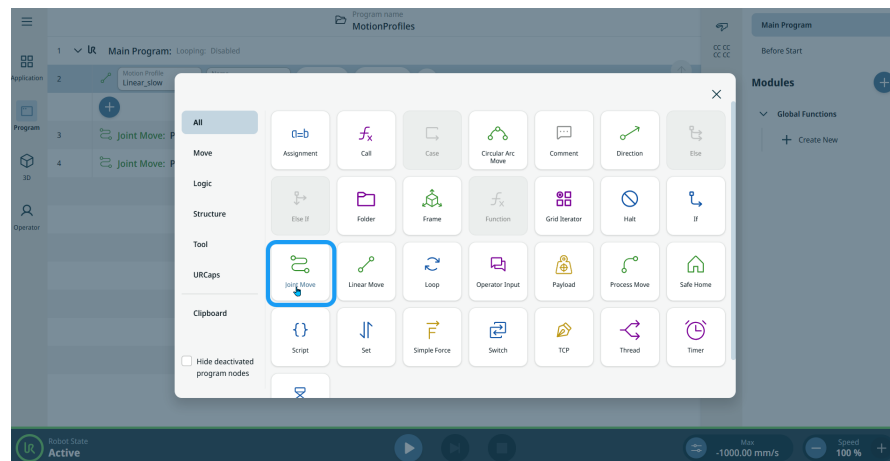
This actions changes the parent of an existing frame to a new one.

7.5.4. Joint Move

Description The **Joint Move** is a move command node that supports Motion Profiles. The command creates a movement from point A to point B that is optimal for the robot. The movement may not be a direct line between A and B, but optimal for the start position of the joints and the end position of the joints. Joint Move makes movements that are calculated in the robot arm joint space. Joints are controlled to finish their movements at the same time. This movement type results in a curved path for the tool to follow.

To access Joint Move command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Joint Move** icon in the Commands toolbox.

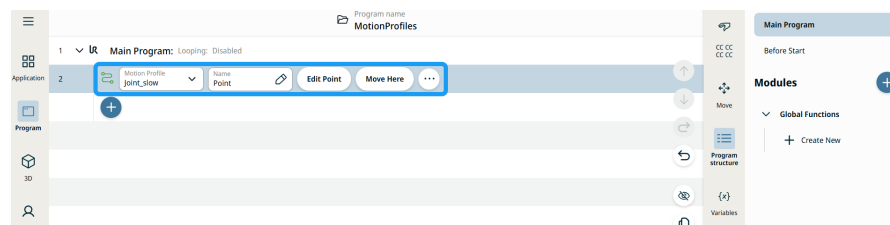


A node with two editable fields, two buttons, and ellipsis icon is inserted into the program tree. These are two editable fields:

- Motion Profile
- Name

These are the two buttons:

- Edit Point
- Move Here



3. Tap the Motion Profile field and select the pertinent from the three choices are available:
 - Custom
 - Joint_fast
 - Joint_slow
4. Tap the Name field, and three tabs appear:
 - Value
 - Variable
 - Expression
5. Enter the data in the **Value**, **Variable**, and **Expression** tabs, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
6. Tap **Edit Point**. A new screen appears with the robot arm interface where you can jog the robot to the desired position.
7. Tap **More** button.

More settings screen appears with four settings:

 - Movement
 - Reference
 - Transform
 - Blend

Motion Profile labels on Move Nodes

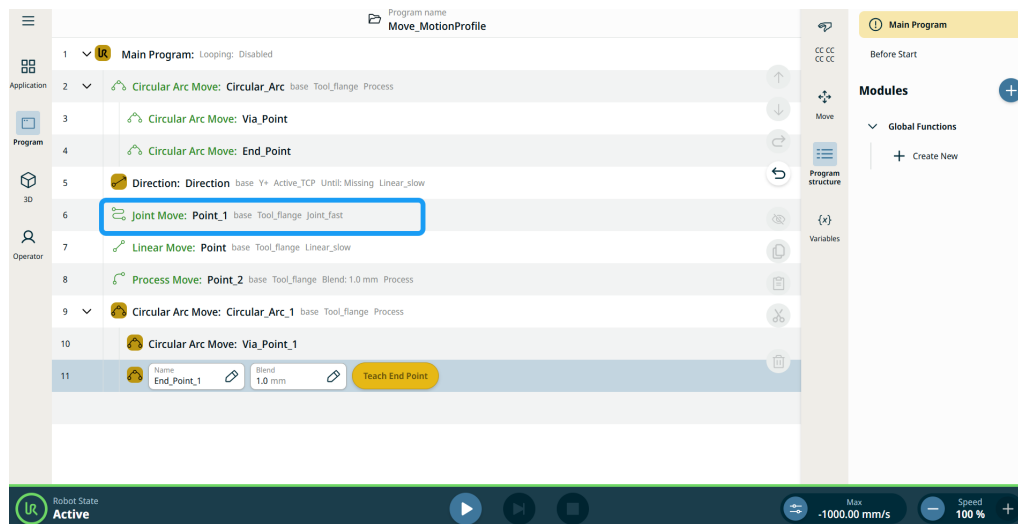
All **Move-related command nodes** in the program tree display the **Motion Profile** associated with each node. This feature enables users to see the selected Motion Profile on a Move node. This feature is implemented on the following command nodes:

- [Linear Move](#)
- [Joint Move](#)
- [Process Move](#)
- [Direction](#)
- [7.5.1 Circular Arc Move on page 155](#)



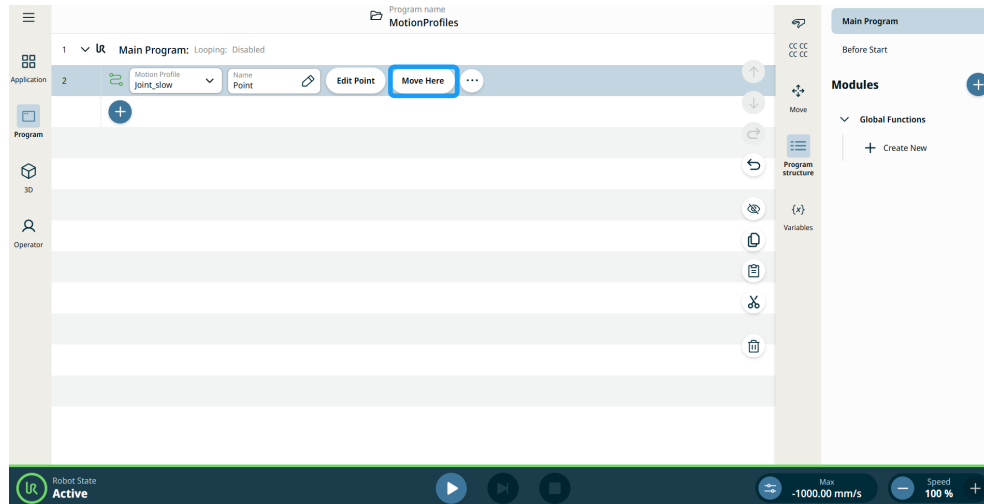
NOTICE

Move nodes with Custom motion profile will display the values for the speed and acceleration instead of Motion profile.

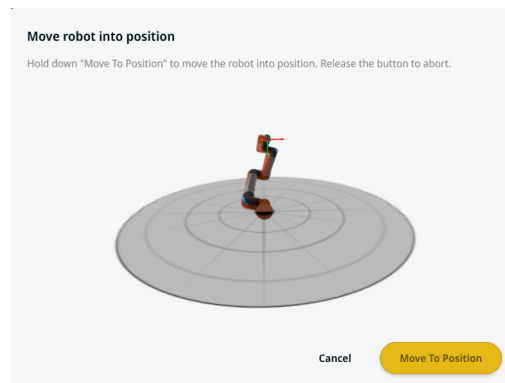


Using the Move Here button

The **Move Here** button is seen in **Linear Move** and **Joint Move** commands. This feature enables you to move to the position of a waypoint directly from the program tree. Pressing the **Move Here** button opens the Move robot into position dialog. The button is visible when the waypoint has been taught and is only enabled when the robot status is in **Normal** or **Reduced** mode.



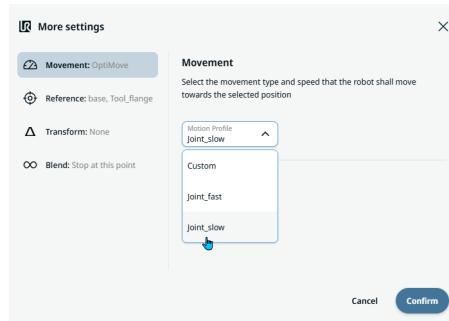
1. Tap the **add icon** to insert the Linear Move and Joint Move in the command tree.
2. Tap the **Edit Point** button of the first move node in the command tree, be it Linear Move or Joint Move nodes.
3. Jog the robot using the plus-minus button in the X, Y, Z coordinates. Tap **Save**.
4. In the second move node in the command tree, tap **Move Here**. A pop-up screen appears to enable **Move robot into position**. Long tap **Move to Position**. Tap **Ok**.





Movement setting

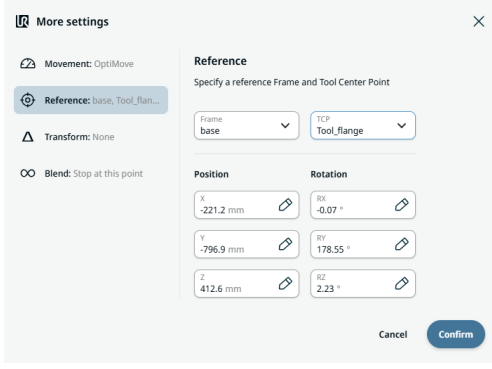
1. Tap the **Movement** setting on the left side. A **Motion Profile** field with the following choices appears on the right side. Joint_slow is the default.
 - Custom
 - Joint_fast
 - Joint_slow



2. If Custom is selected, three more fields appear:
 - Speed Type
 - Speed
 - Acceleration
3. Choose either OptiMove or Classic when you tap Speed Type.
4. Tap the **Speed** field, and the **Value**, **Variable**, and **Expression** tabs pop up.
5. Enter data in each tab, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
6. Tap the **Acceleration** field, and the **Value**, **Variable**, and **Expression** tabs pop up.
7. Enter data in each tab, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
8. When **Joint_fast** and **Joint_slow** are selected, the OptiMove speed and acceleration are shown.

Reference setting

1. Tap the **Reference** setting.



More settings

Movement: OptiMove

Reference: base, Tool_flange

Transform: None

Blend: Stop at this point

Reference
Specify a reference Frame and Tool Center Point

Frame: base TCP: Tool_flange

Position		Rotation	
X	-221.2 mm	RX	-0.07 °
Y	-796.9 mm	RY	178.55 °
Z	412.6 mm	RZ	2.23 °

Cancel Confirm

2. Choose either **Frame** or **TCP** field.

In the **Frame** field, four choices are available:

- world
- base
- flange
- tcp

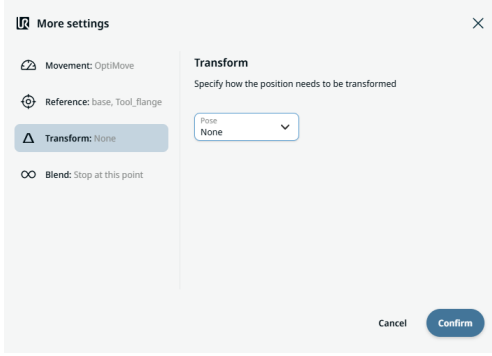
In the **TCP** field, two choices are available:

- Active_TCP
- Tool_flange

3. In the **Position** column, tap the **X**, **Y**, **Z** tabs and enter your chosen value for the robot, and **Confirm**.
4. In the **Rotation** column, tap the **RX**, **RY**, **RZ** tabs and enter your chosen value for the robot, and **Confirm**.

Transform setting

1. Tap the **Transform** setting.



More settings

Movement: OptiMove

Reference: base, Tool_flange

Transform: None

Blend: Stop at this point

Transform
Specify how the position needs to be transformed

Pose: None

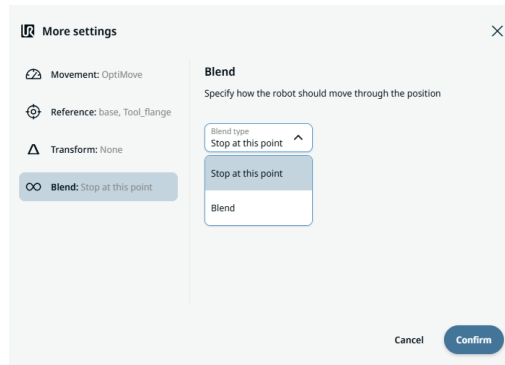
Cancel Confirm

2. Tap the **Pose** field and **Confirm**.



Blend setting

1. Tap the Blend setting.



A **Blend type** field appears with two options:

- Stop at this point
- Blend

2. When you choose **Stop at this point**, no further action is needed but only **Confirm**.
3. Tap **Blend**, and a **Radius** field is inserted.
4. The **Value**, **Variable**, and **Expression** tabs pop up when you tap the Radius field.
5. Enter data in each tab, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
6. To finish, tap **Confirm** in the More settings screen.

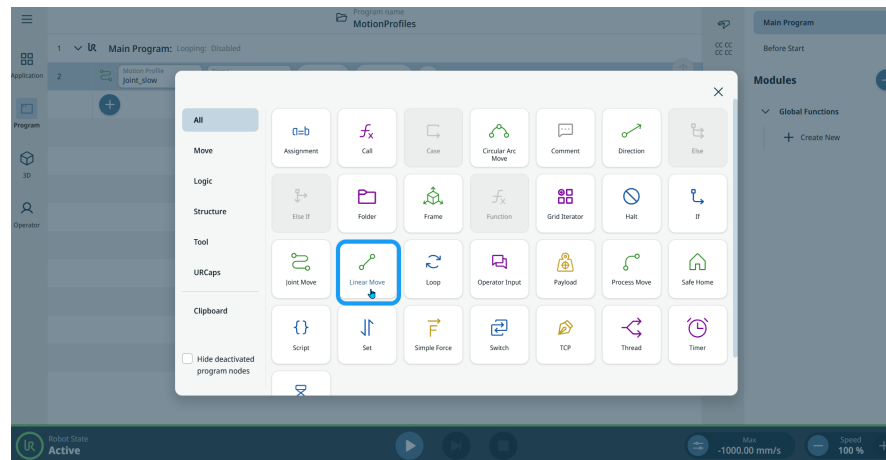
7.5.5. Linear Move

Description

The **Linear Move** command node supports Motions Profiles as Joint Move, but it creates a movement that is a direct line from point A and point B. It moves the Tool Center Point (TCP) linearly between waypoints. This means that each joint performs a more complicated motion to keep the tool on a straight line path.

To access Linear Move command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Linear Move** icon in the Commands toolbox.

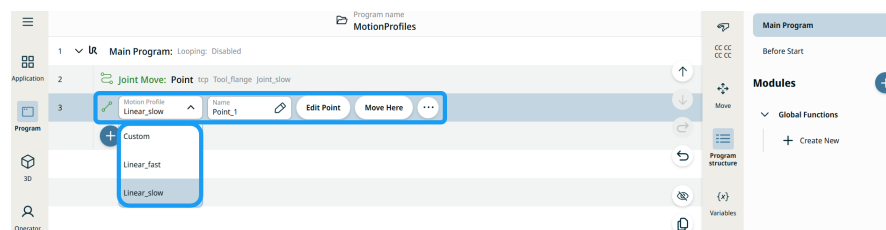


A node with two editable fields, two buttons, and ellipsis icon is inserted into the program tree. These are two editable fields:

- Motion Profile
- Name

These are the three buttons:

- Edit Point
- Move Here



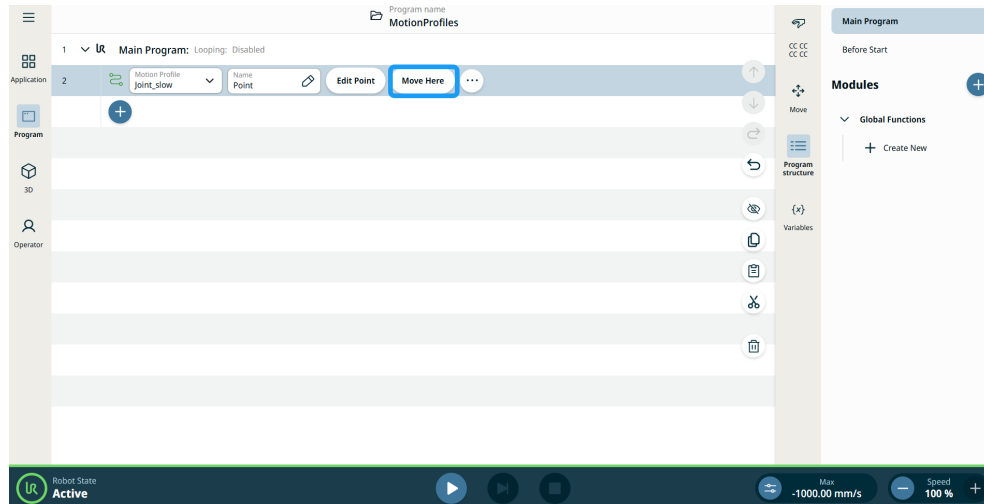
3. Tap the **Motion Profile** field and select the pertinent from the three choices are available:
 - Custom
 - Linear_fast
 - Linear_slow



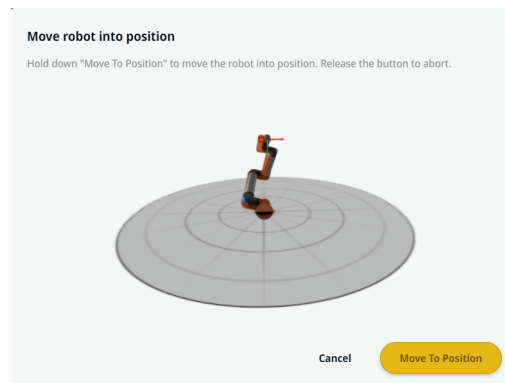
4. Tap the **Name** field, and three tabs appear:
 - Value
 - Variable
 - Expression
 5. Enter the data in the **Value**, **Variable**, and **Expression** tabs, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
 6. Tap **Edit Point**. A new screen appears with the robot arm interface.
 7. Tap **More** button. The **More settings** screen appears with four settings:
 - Movement
 - Reference
 - Transform
 - Blend
-

Using the Move Here button

The **Move Here** button is seen in **Linear Move** and **Joint Move** commands. This feature enables you to move to the position of a waypoint directly from the program tree. Pressing the **Move Here** button opens the Move robot into position dialog. The button is visible when the waypoint has been taught and is only enabled when the robot status is in **Normal** or **Reduced** mode.



1. Tap the **add icon** to insert the Linear Move and Joint Move in the command tree.
2. Tap the **Edit Point** button of the first move node in the command tree, be it Linear Move or Joint Move nodes.
3. Jog the robot using the plus-minus button in the X, Y, Z coordinates. Tap **Save**.
4. In the second move node in the command tree, tap **Move Here**. A pop-up screen appears to enable **Move robot into position**. Long tap **Move to Position**. Tap **Ok**.





Motion Profile labels on Move Nodes

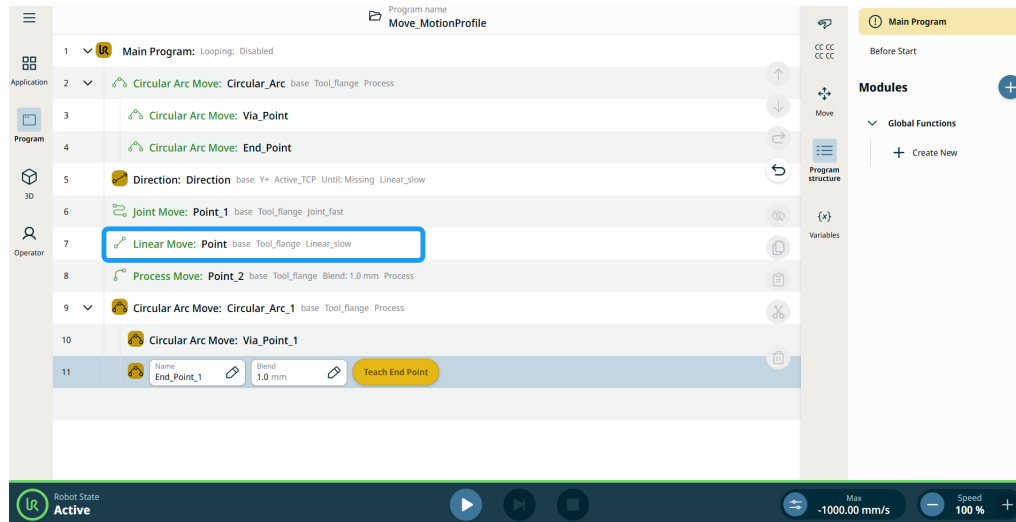
All **Move-related command nodes** in the program tree display the **Motion Profile** associated with each node. This feature enables users to see the selected Motion Profile on a Move node. This feature is implemented on the following command nodes:

- [Linear Move](#)
- [Joint Move](#)
- [Process Move](#)
- [Direction](#)
- [7.5.1 Circular Arc Move on page 155](#)



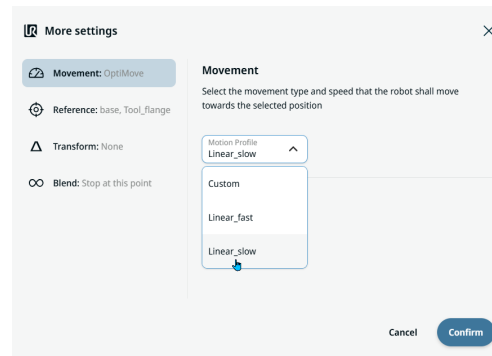
NOTICE

Move nodes with Custom motion profile will display the values for the speed and acceleration instead of Motion profile.



Movement setting

1. Tap the **Movement** setting on the left side. A **Motion Profile** field with the following choices appears on the right side. **Linear_slow** is the default.
 - Custom
 - Linear_fast
 - Linear_slow



2. If **Custom** is selected, three more fields appear:
 - Speed Type
 - Speed
 - Acceleration
3. Choose either OptiMove or Classic when you tap Speed Type.
4. Tap the **Speed** field, and the **Value**, **Variable**, and **Expression** tabs pop up.
5. Enter data in each tab, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
6. Tap the **Acceleration** field, and the **Value**, **Variable**, and **Expression** tabs pop up.
7. Enter data in each tab, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
8. When **Linear_fast** and **Linear_slow** are selected, the OptiMove speed and acceleration are shown.



Reference setting

1. Tap the **Reference** setting.

More settings

Movement: OptiMove

Reference: base, Tool_flan...

Transform: None

Blend: Stop at this point

Reference
Specify a reference Frame and Tool Center Point

Frame: base TCP: Tool_flange

Position

X	Y	Z
465.6 mm	-1401.3 mm	188.7 mm

Rotation

RX	RY	RZ
-12.23 °	148.91 °	-17.22 °

Cancel Confirm

2. Choose either **Frame** or **TCP** field.
3. In the **Frame** field, four choices are available:
 - world
 - base
 - flange
 - tcp
4. In the **TCP** field, two choices are available:
 - Active_TCP
 - Tool_flange
5. In the **Position** column, tap the **X**, **Y**, **Z** tabs and enter your chosen value for the robot, and **Confirm**.
6. In the **Rotation** column, tap the **RX**, **RY**, **RZ** tabs and enter your chosen value for the robot, and **Confirm**.

Transform setting

1. Tap the **Transform** setting.

More settings

Movement: OptiMove

Reference: base, Tool_flange

Transform: None

Blend: Stop at this point

Transform
Specify how the position needs to be transformed

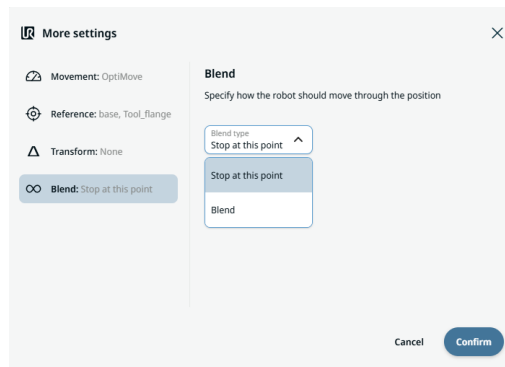
Pose: None

Cancel Confirm

2. Tap the **Pose** field and **Confirm**.

Blend setting

1. Tap the Blend setting.



A **Blend type** field appears with two options:

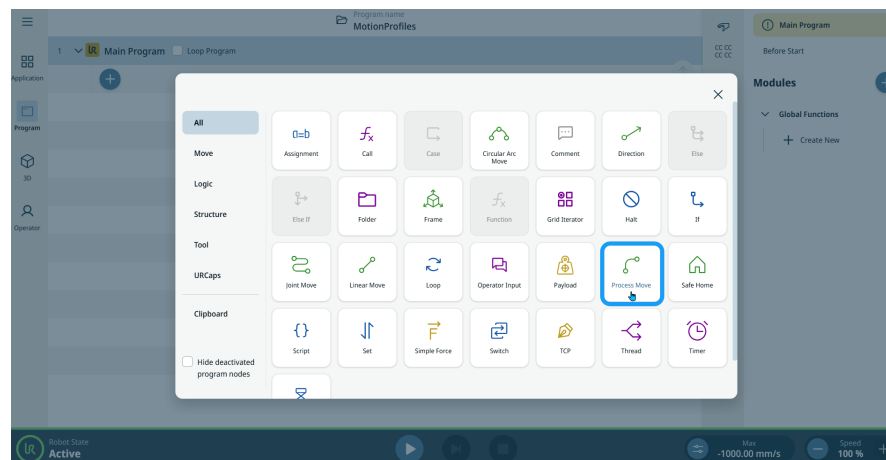
- Stop at this point
 - Blend
2. When you choose **Stop at this point**, no further action is needed but only **Confirm**.
 3. Tap Blend, and a **Radius** field is inserted.
 4. The **Value**, **Variable**, and **Expression** tabs pop up when you tap the Radius field.
 5. Enter data in each tab, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
 6. To finish, tap **Confirm** in the More settings screen.

7.5.6. Process Move

Description The **Process Move** command enables you to program a linear movement with constant speed through a number of points. You must set blend radius to make sure that the robot moves at a constant speed between each move node. You can specify a reference frame and a transform pose.

To access Process Move command

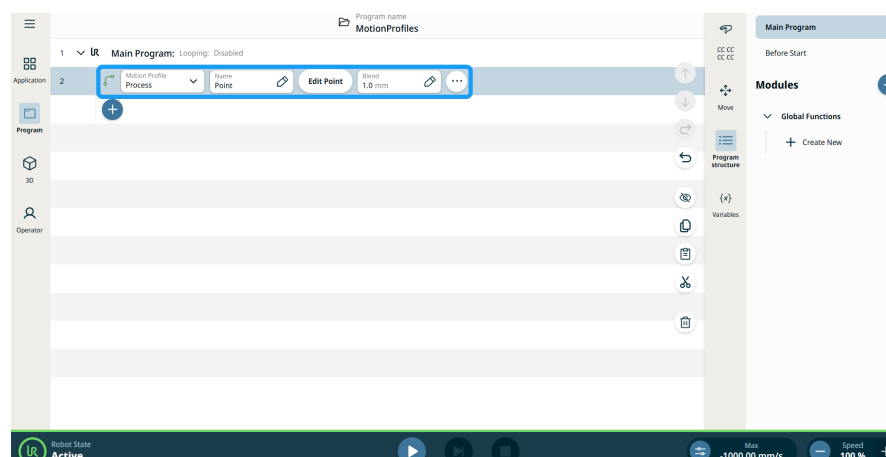
1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Process Move** icon in the Commands toolbox.



A node is inserted into the program tree. The node has three editable fields, Edit Point button, and ellipsis icon.

These are the three fields:

- Motion Profile
- Name
- Blend



3. In the **Motion Profile** field, you can choose either **Custom** or **Process**.
 Custom motion profile enables you to define movement data in **Joint Move** and **Linear Move** nodes.
 Process motion profile is a predefined motion in **Joint Move** and **Linear Move** nodes.
4. Tap the **Name** field.
 Three tabs appear:
 - Value
 - Variable
 - Expression
5. Enter data in the **Value**, **Variable**, and **Expression** tabs, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
6. Tap **Edit Point**.
 A new screen appears with the robot arm interface.
7. Tap the **Blend** field.
 Three tabs appear:
 - Value
 - Variable
 - Expression
8. Enter data in the **Value**, **Variable**, and **Expression** tabs, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).

Motion Profile labels on Move Nodes

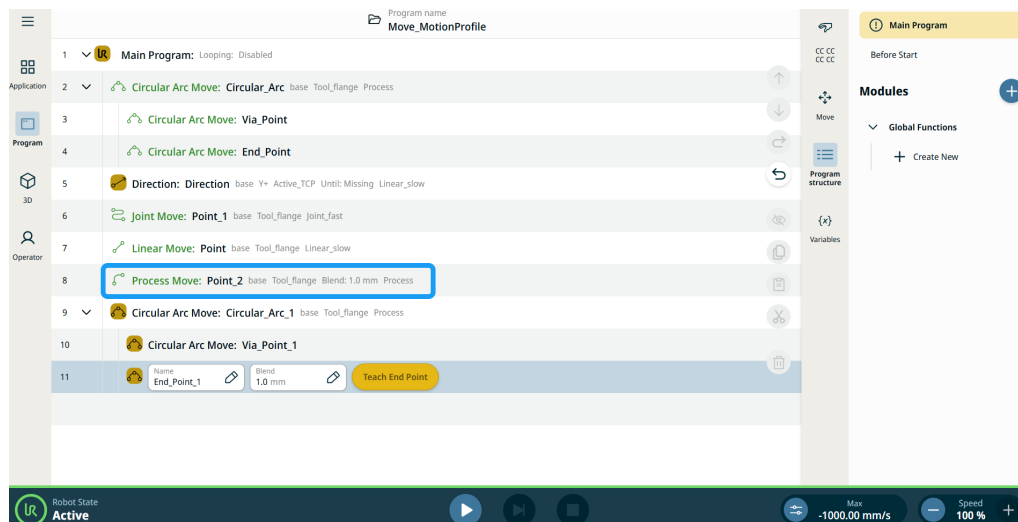
All **Move-related command nodes** in the program tree display the **Motion Profile** associated with each node. This feature enables users to see the selected Motion Profile on a Move node. This feature is implemented on the following command nodes:

- [Linear Move](#)
- [Joint Move](#)
- [Process Move](#)
- [Direction](#)
- [7.5.1 Circular Arc Move on page 155](#)



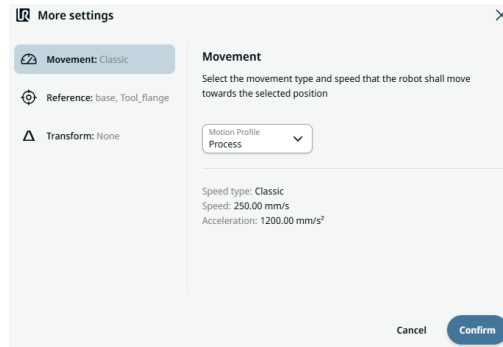
NOTICE

Move nodes with Custom motion profile will display the values for the speed and acceleration instead of Motion profile.



To use More Option

1. Tap the **ellipsis** icon.
More settings screen appears with three settings:
 - Movement
 - Reference
 - Transform



2. Tap the **Movement** setting on the left side.
A **Motion Profile** field with selections **Custom** and **Process** is on the right side.
3. If Custom is selected, a **Speed** field and **Acceleration** field appears.
4. Tap the **Speed** field, and the **Value**, **Variable**, and **Expression** tabs pop up.
5. Enter data in each tab, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
6. Tap the **Acceleration** field, and the **Value**, **Variable**, and **Expression** tabs pop up.
7. Enter data in each tab, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
8. Choose **Process** motion profile and tap **Confirm**.
9. Tap the **Reference** setting.
10. Choose either **Frame** or **Set TCP** field.
11. In the **Position** column, tap the **X**, **Y**, **Z** tab and enter your chosen value for the robot, and **Confirm**.
12. In the **Rotation** column, tap the **RX**, **RY**, **RZ** tab and enter your chosen value for the robot, and **Confirm**.
13. In the **Transform** setting, tap the **Pose** field and **Confirm**.

7.6. Logic Command Nodes

Description

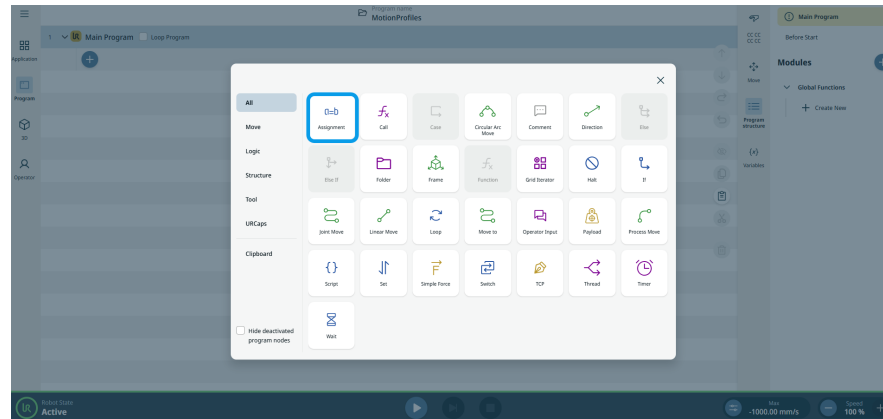
You use Logic nodes to control program flow by branching execution with If/Else, repeating sequences with Loop, and driving and synchronizing I/O through Set and Wait so the robot responds to sensor inputs and variable values.

7.6.1. Assignment

Description The **Assignment** command enables you to assign values to created variables.

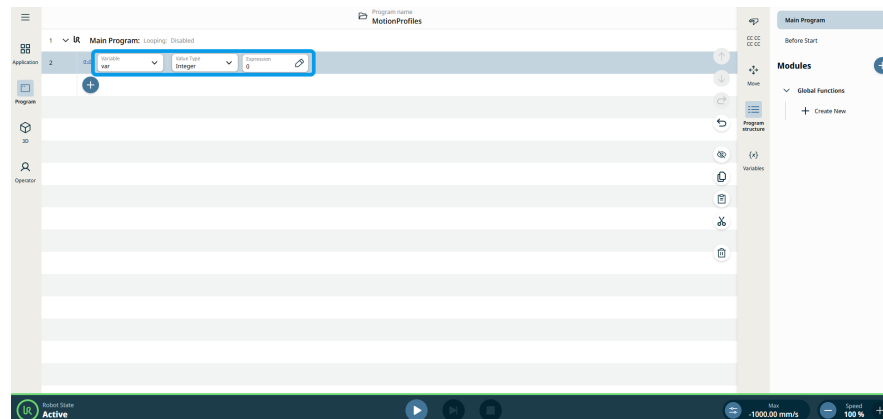
To access Assignment command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Assignment** icon in the Commands toolbox.



A node with three editable fields is inserted into the program tree:

- Variable
- Value Type
- Expression



3. Tap the **Variable** field to choose your variable:
 - grid
 - grid_iterator
 - var

The Variable field contains the name of the variable. By default, the variable is called **var**.

4. Tap the **Value Type** field to and select your choice:

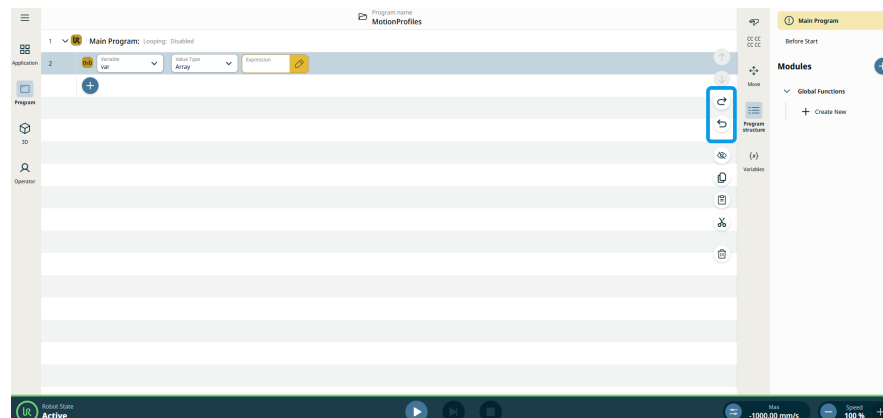
- Array
- Boolean
- Float
- Grid
- Integer
- Matrix
- Pose
- String
- Timer
- Waypoint

The Value Type contains the type of variable you are creating.

Type	Allowed Values/Examples	Examples
Array	A list of values	
Boolean	True/False	True
Float	A decimal point value	1.23
Grid	A grid defined in the application	
Integer	A whole number	5
Matrix		
Pose	A pose for the robot to move	p[0,0,0,0,0,0,]
String	Text	"Hello World"
Waypoint	A waypoint is defined in the program	

5. Tap the **Expression** field, select your variables and expression, and **Confirm**. The Expression is the value to assign the variable. See [Expression Editor](#).

Note: Tap the icon **arrow left** and **arrow right** in the right sidebar of the main screen if you want the assignment to be reverted.

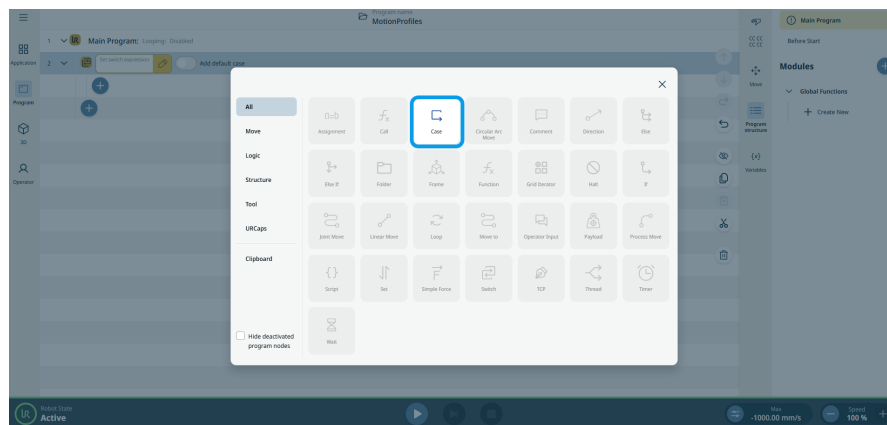


7.6.2. Case

Description The **Case** command enables you to point to a function to be executed, depending on an input. It can only be used with the Switch command. It is usually referred to as "Switch-Case" as they always go together.

To access Case command

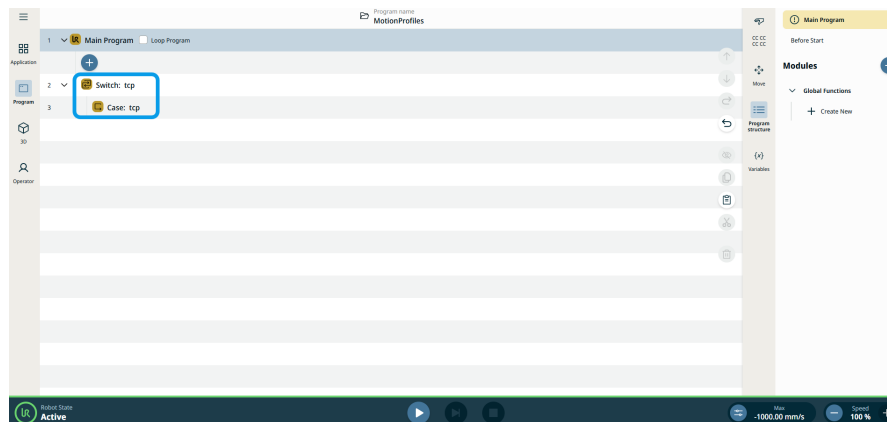
1. Go to the main program screen. See [Command Nodes](#).
2. Access the [Switch](#) command.
3. After the input in **Set switch expression**, tap the first add icon from the child node of the switch command.
4. Tap **Case** icon in the Commands toolbox.



A node with the **Set value** field and **Module** field is inserted into the program tree, under the switch expression.

5. Tap **Set value**.
6. Create the set value and tap **Confirm**.
7. Tap **Module**.
8. Tap **Global Functions**.

A switch-case is created in the program tree.

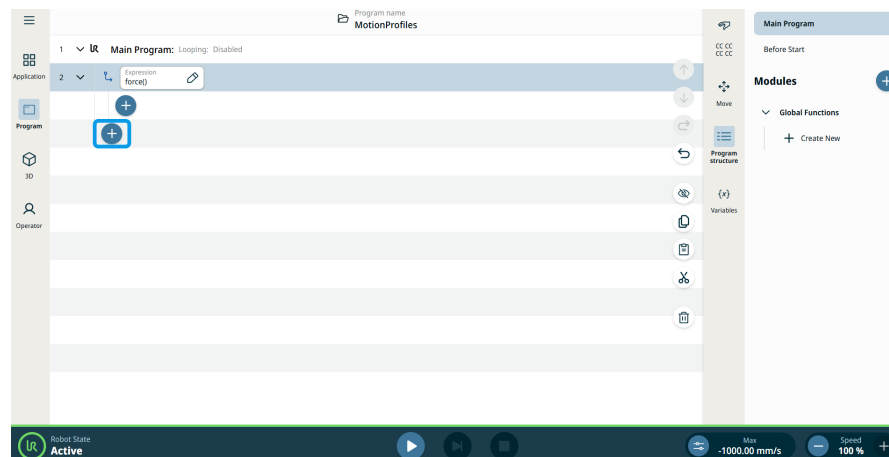


7.6.3. Else

Description The **Else** command, in addition to Else If, is used together with the If command. It enables you to make statements to change the robot's behavior based on sensor inputs or variable values.
It is only insertable directly after an If or Else If node.
The children of this node are only executed if the preceding If or Else If is false.

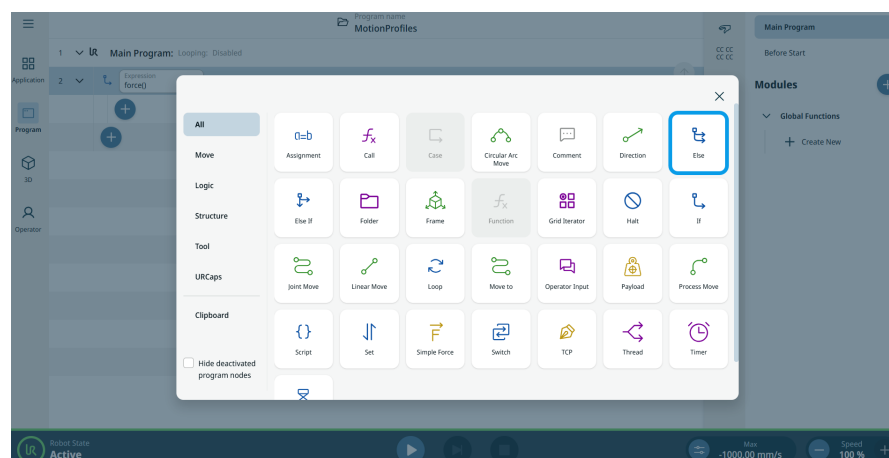
To access Else command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **If** icon in the Commands toolbox.
Follow the process when accessing the If program. See [If](#) for more.
An Expression field and two add icons are inserted into the program tree:
 - add icon in the If child node
 - add icon for another node
3. Tap the **second add icon**.



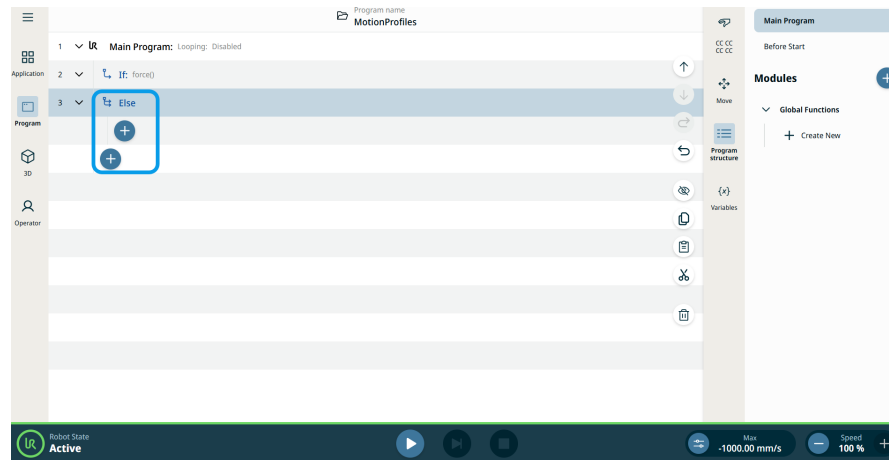
The Commands toolbox appears.

4. Tap the **Else** icon in the Commands toolbox.



The Else node is inserted into the program tree.

5. Choose and tap the appropriate add icon to access other commands for the robot movement.



7.6.4. Else If

Description

The **Else If** command, in addition to Else, is used together with the If command. It enables you to make statements to change the robot's behavior based on sensor inputs or variable values.

The Else If node is only insertable directly after an If or Else node.

The children of this node are only executed if the preceding If or Else If node is False, but its expression evaluates it as True.

To access Else If command

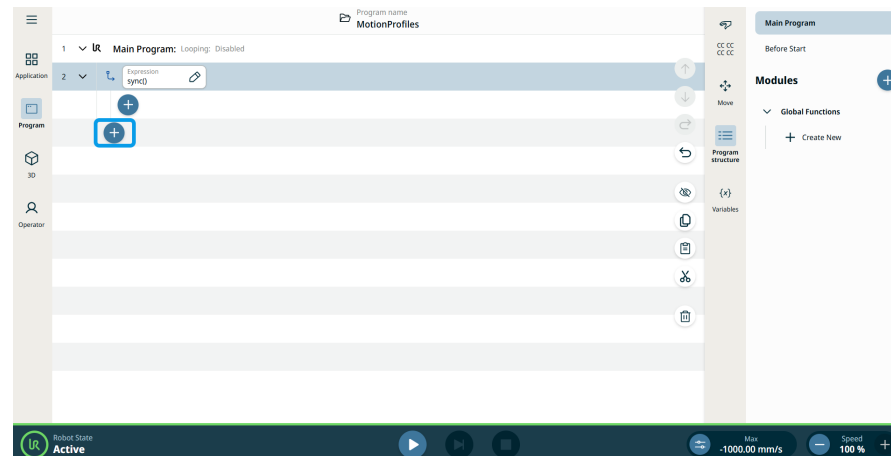
1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **If** icon in the Commands toolbox.

Follow the process when accessing the If program. See [If](#) for more.

An **Expression** field and two add icons are inserted into the program tree:

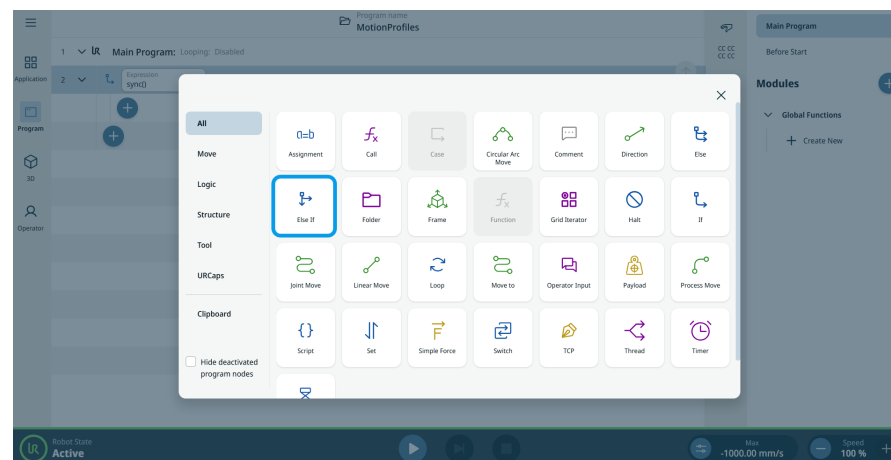
- Add icon in the If child node
- Add icon for another node

3. Tap the **second add icon**.



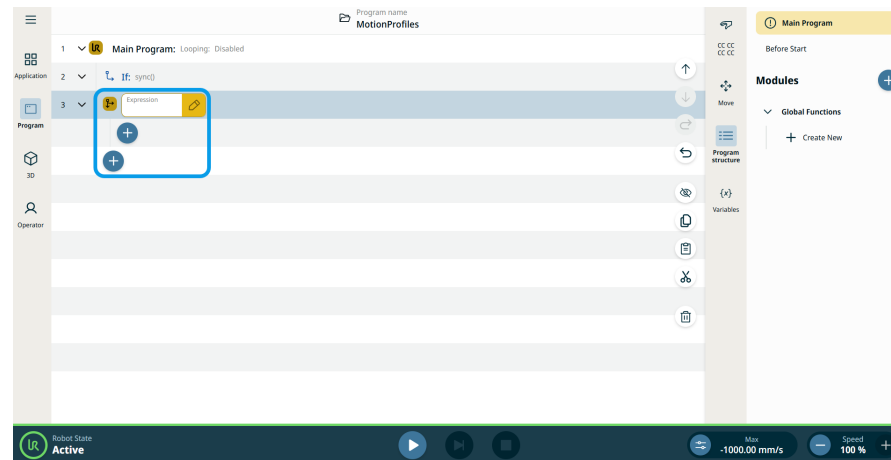
The Commands toolbox appear.

4. Tap the **Else If** icon in the Commands toolbox.



- The Else If node is inserted into the program tree.

The node has an editable **Expression** field and two add icons.



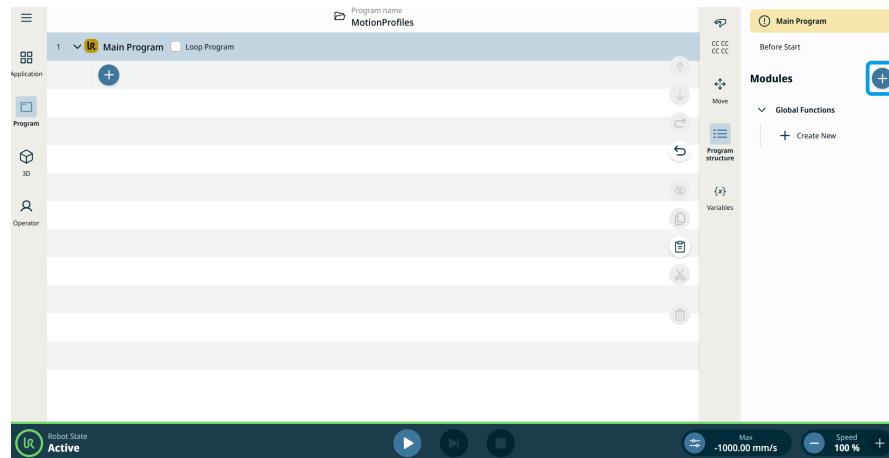
- Tap the **Expression** field.
- Create your expression .
- Tap **Confirm**.
- Choose and tap the appropriate **add icon** to access other commands for the robot movement.

7.6.5. Function

Description The **Function** command allows you to perform a specialized operation, like performing calculation, moving the robot, and other similar movements. The function will not be executed until the call command. It is only insertable inside a module or from the Global Functions.

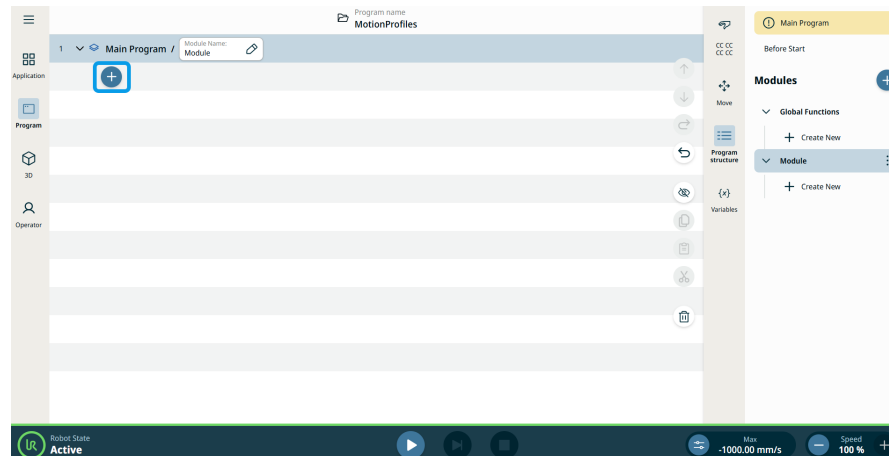
To access Function command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **add icon** beside the **Modules** tree in the multitask screen.

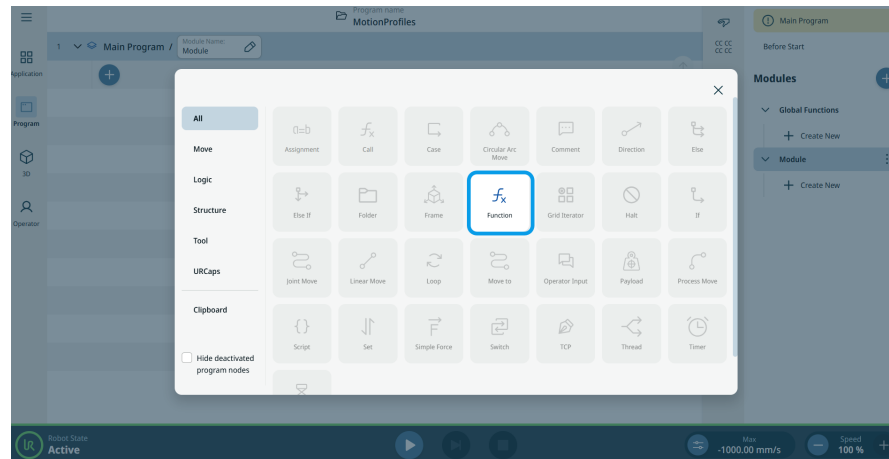


A new module is inserted into the module tree.

3. Tap the **add icon** in the main screen located under the Main Program.



4. Tap the **Function** icon in the Commands toolbox.



A node is inserted into the program tree, which is an editable field called **Function Name**.

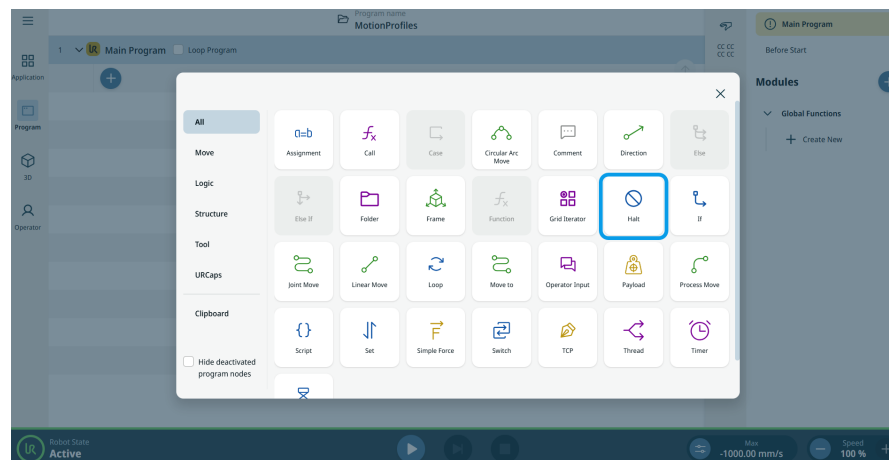
5. Tap the **Function Name** field.
6. Create a function name.
7. Tap **Confirm**.

7.6.6. Halt

Description The **Halt** command allows you to stop the robot at that location.

To access the Halt command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Halt** icon in the Commands toolbox.



You see a notification under the chosen program: "Program execution stops at this point."

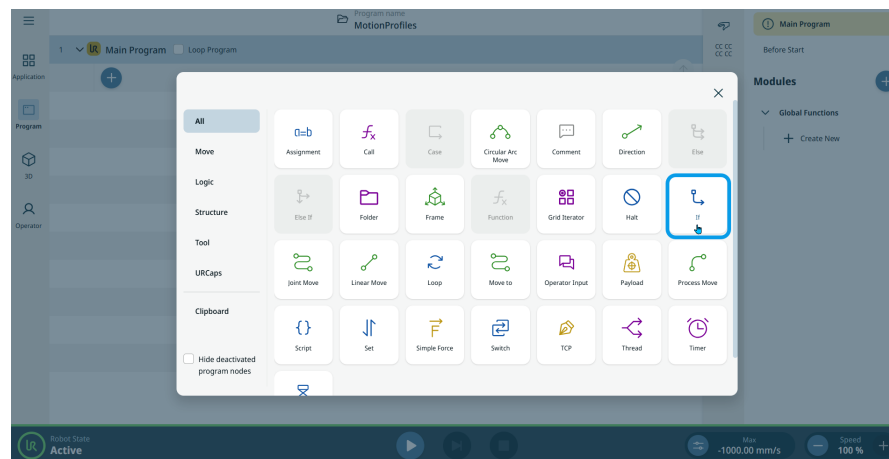
7.6.7. If

Description The **If** command changes the robot's behavior based on sensor inputs or variable values. It allows you to control the flow of your program.

To access If command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **If** icon in the Commands toolbox.

A node is inserted into the program tree, which has an editable field.



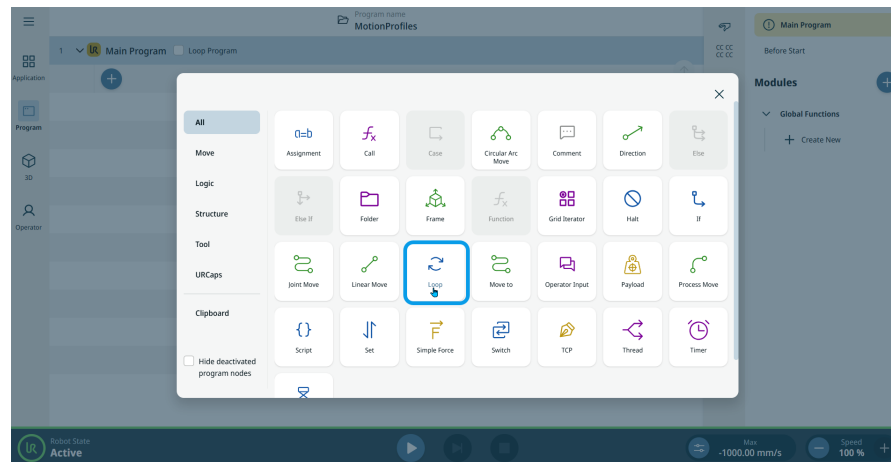
3. Tap the **Expression** field.
The children of the If node will only be executed if the expression evaluates to True.
4. Enter your expression and **Confirm**.

7.6.8. Loop

Description The **Loop** command enables you to loop underlying program commands in PolyScope X.

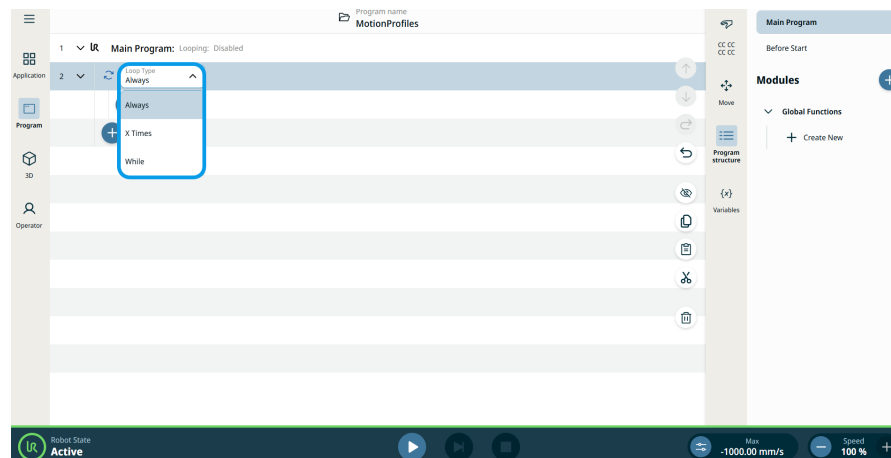
To access Loop command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Loop** icon in the Commands toolbox.



A node with **Loop Type** field is inserted into the program tree.

3. Tap the **Loop Type** field. Three choices are available:
 - **Always**. Any commands under the loop command will be repeated continuously, such as the **Loop Program**, which continually repeats everything under the **Main Program**.
 - **X Times**. Repeats any commands in the loop command the number of times that's entered.
 - **While**. Use some type of expression as the trigger to end the loop cycle.



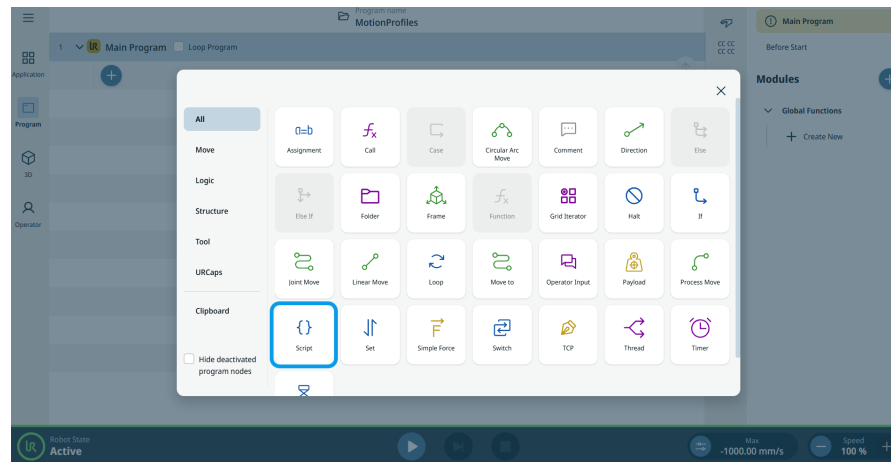
4. Tap the **Always** tab. No further action to do.
5. Tap **X Times**, and two additional tabs appear on the right:
 - X Times
 - Variable Name
6. On the **X Times** tab, edit and **Confirm**.
7. On the **Variable Name** tab, edit and **Confirm**.
8. Tap **While**. An additional **Expression** tab appears on the right.
9. Tap **Expression** to create expression that must be between 1 and 1000 characters, and **Confirm**.

7.6.9. Script

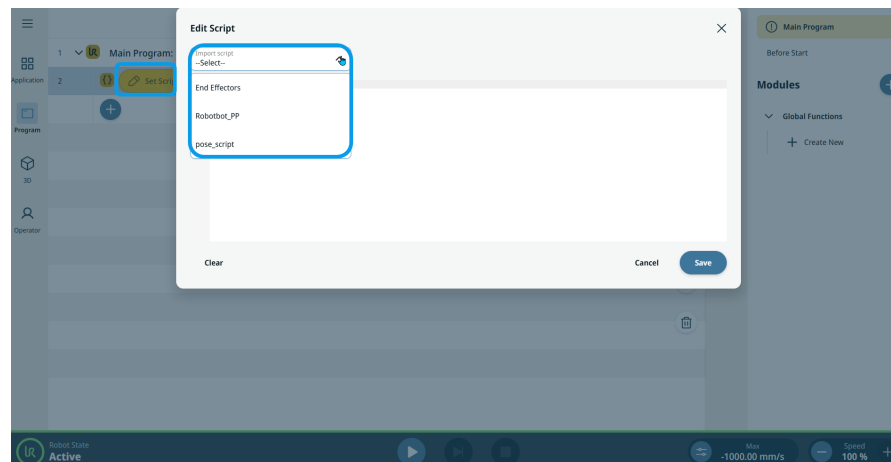
Description The **Script** command enables you to write and insert a URscript code or files directly in your program.

To access Script command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Script** icon in the Commands toolbox. A node is inserted into the program tree with a **Set Script** button.



3. Tap the **Set Script** button. A new screen appears where you can edit the script. Enter the script and tap **Save**.



4. Tap the **Import script** field to import a script from the Script Manager, as well as to clear the contents. See [Script Files](#).



NOTICE

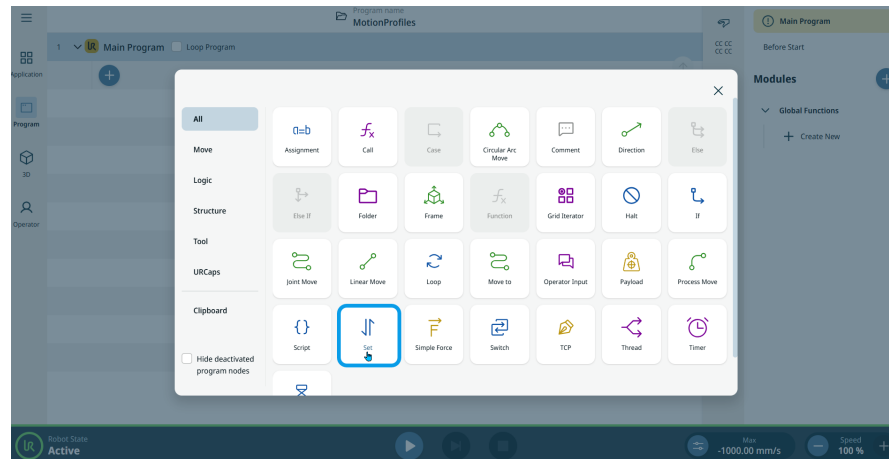
The script file is copied from the script manager into the script node when inserted. Any future imports of the file into the script manager will not change the existing programs. The programs will need to be manually updated, if necessary.

7.6.10. Set

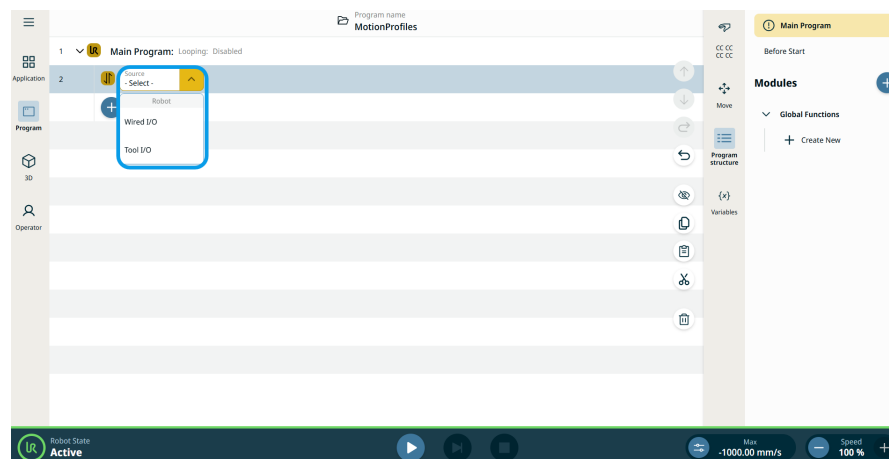
Description The **Set** command enables you to set a given value to either digital or analog outputs.

To access Set command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Set** icon in the Commands toolbox.



3. A node with an editable **Source** field is inserted into the program tree that enables you to select **Wired IO** and **Tool IO**.
4. Tap the **Source** field.



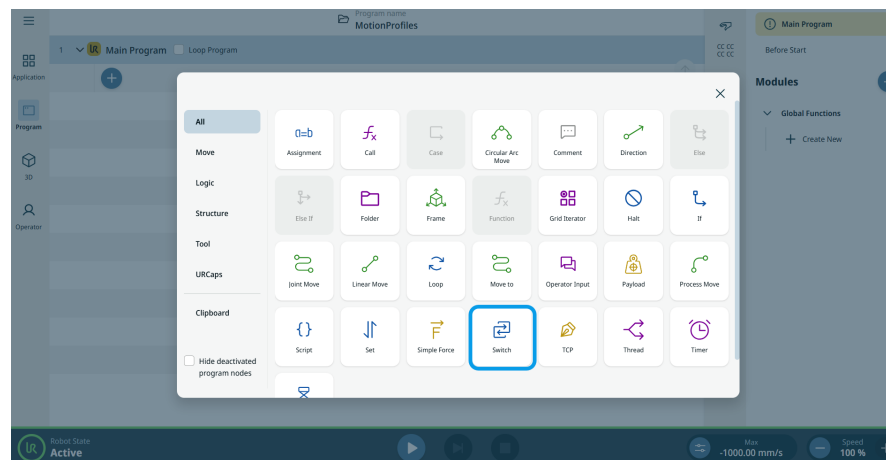
5. When you select **Wired IO**, a **Signal** field is inserted.
6. In the **Signal** field, tap your preferred signal.
 - DO 0-DO 7: Digital signal
 - CO 0-CO 7: Digital signal
 - AO 0-AO 1: Analog signal
7. When a digital signal is selected, a **Value** field is inserted to the right.
8. Choose either **High** or **Low**.
9. When an analog signal is selected, a **Value** field is inserted to the right.
10. Enter data in the **Value**, **Variable**, and **Expression** tabs, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
11. When you select **Tool IO**, a **Signal** field is inserted.
12. Choose either DO 0 or DO 1.
A **Value** field is inserted to the right with three options:
 - Use Variable
 - High
 - Low
13. Tap **Use Variable**, and the **Variable** and **Expression** tabs appear.
14. Enter data in the two tabs, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).
15. You can also choose either **High** or **Low** in the Value field.

7.6.11. Switch

Description	The Switch command enables you to make the robot change behavior based on sensor inputs or variable values.
--------------------	--

To access Switch command

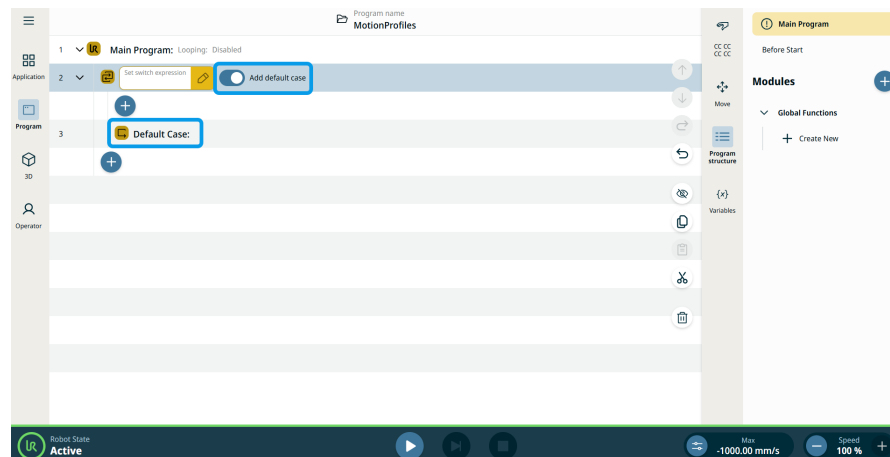
1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Switch** icon in the Commands toolbox.



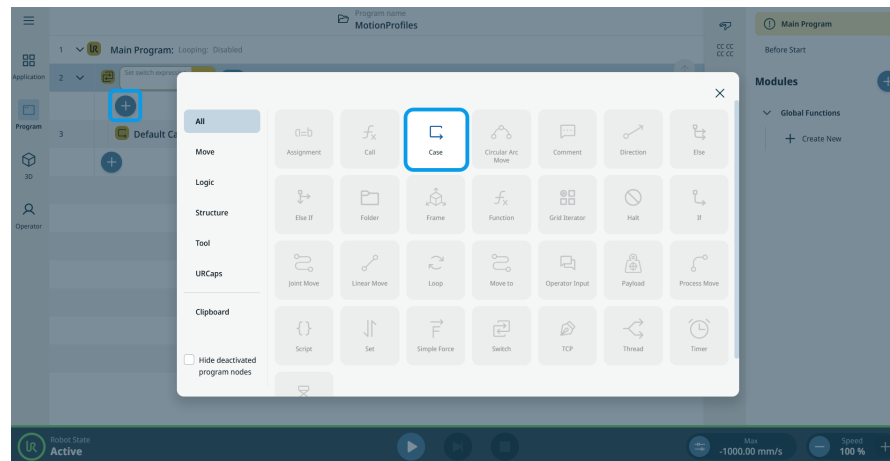
A node with **Set switch expression** field and **Add default case** button is inserted into the program tree.

3. Tap the **Set switch expression** field, and two options are available:
 - Variable
 - Expression
4. Tap **Variable** and select or switch based on an existing variable or an expression.
 Note: Open the Assignment command node to activate the Variable tab in Switch node.
5. Enter the expression in **Expression** tab and **Confirm**.

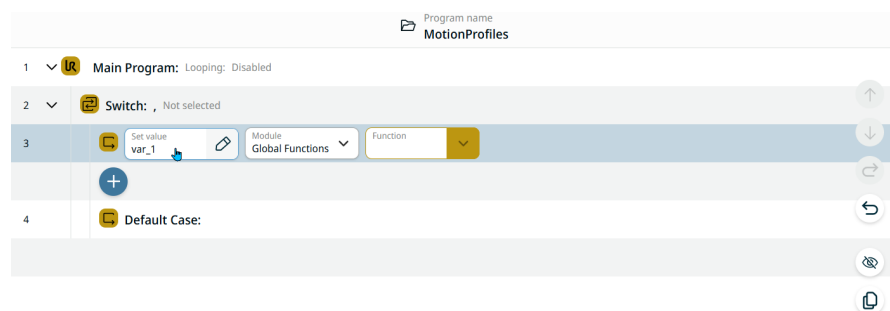
6. Tap **Add default case** button, and **Default Case** is inserted.



7. Tap the **add icon** inside the Switch node to add a case node.
8. Tap the **Case** icon in the Commands toolbox. See [Case](#).



Each case checks the switch variable or expression against a single value.
If a case and switch match, it calls the function you specify in the case node.

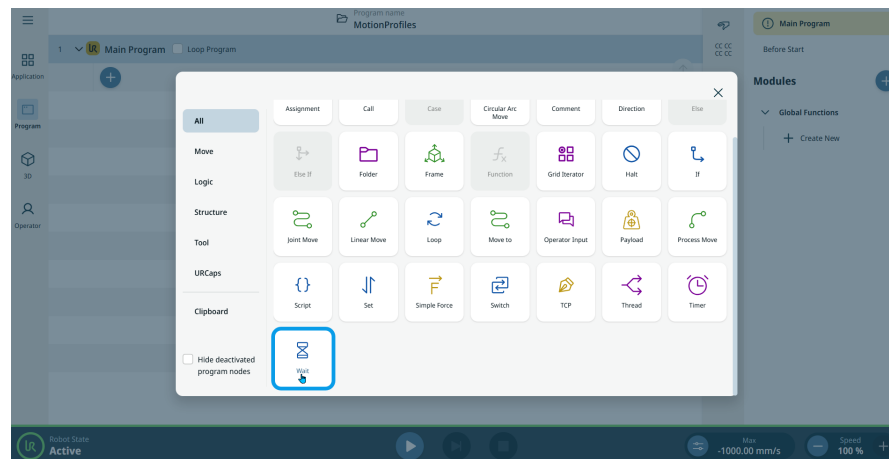


7.6.12. Wait

Description The **Wait** command pauses the robot's movement when new inputs are introduced into the program. You can add a Wait command to a program with external sensors to make the robot wait for one of the sensors to activate before the program continues.

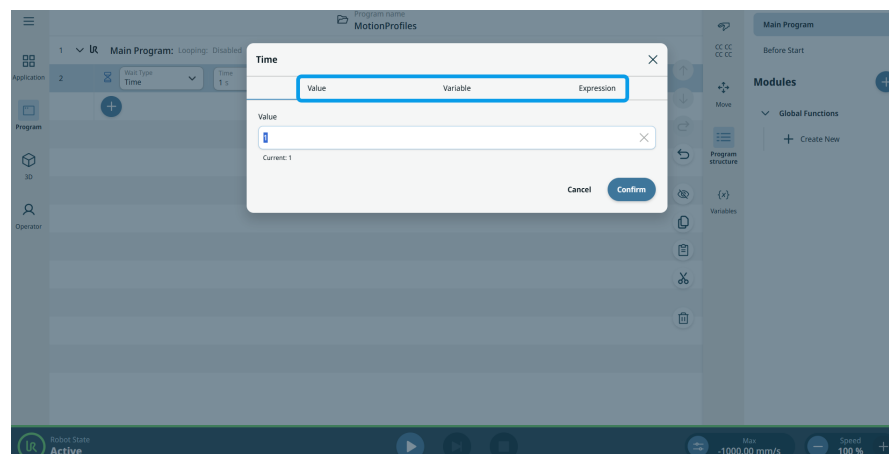
To access Wait command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Wait** icon in the Commands toolbox.



A node with two editable fields is inserted into the program tree:

- Wait Type
 - Time
3. Tap the **Wait Type** field and choose **Time**.
When you select this, the program waits for a number of specified seconds.
 4. On the **Time** field, enter the **Value** and **Expression**, and tap **Confirm**. See [Value-Variable-Expression Tabs](#).

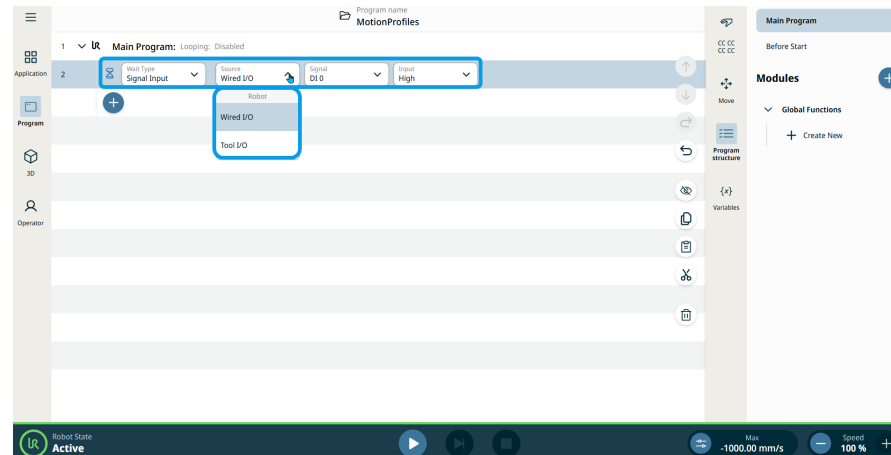


5. Tap the **Wait Type** field and choose **Signal Input**.

When you choose this, additional fields are added to the right for you to specify the type of input.

6. Tap the **Source** field, and two standard inputs are available:

- Wired IO
- Tool IO



7. When you choose either **Wired IO** or **Tool IO**, a **Signal** field is inserted to the right.

8. In the **Signal** field, tap your preferred signal.

- Digital signal: DI 0-DI 7
- Digital signal: CI 0-CI 7
- Analog signal: AI 0-AI 1

When a digital signal is selected, an **Input** field is inserted to the right.

9. Choose either **High** or **Low**.

When an analog signal is selected, **Operator** and **Input** fields are inserted to the right.

10. Tap the **Operator** field.

11. Choose either **<** or **>** to specify the wait compared to a value.

12. Edit your **Input** field and **Confirm**.

7.7. Structure Command Nodes

Description

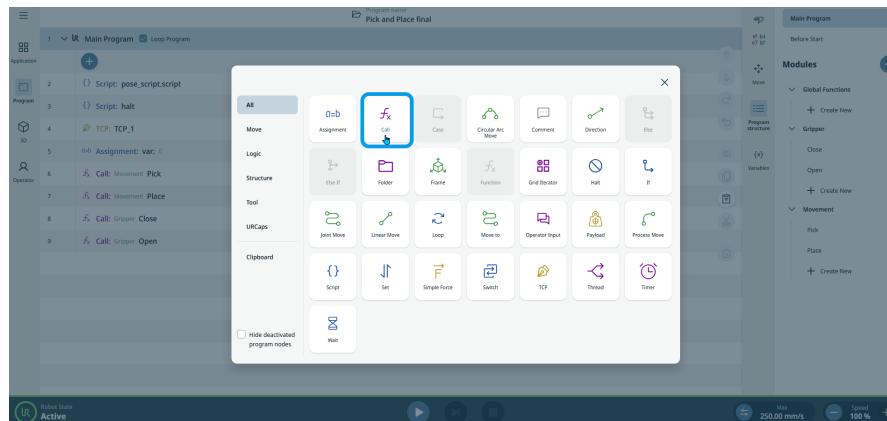
You use Structure nodes to organize and structure into modules your program by grouping steps in folders, building reusable functions that you run with Call, running parallel tasks with Thread, and storing values with Assignment to keep large programs readable and maintainable.

7.7.1. Call

Description The **Call** command enables you to execute and run created global functions and modules, which are seen in the multitask screen. To use this command, a prior program creation using modules and functions is a requisite. See [Modules and Functions](#) on how to set up commands and group them into modules and functions. Read [Gripper Module](#) and [Movement Module](#) for details about modules. Read [Open/Close](#) and [Pick/Place](#) for details about functions.

To access Call command

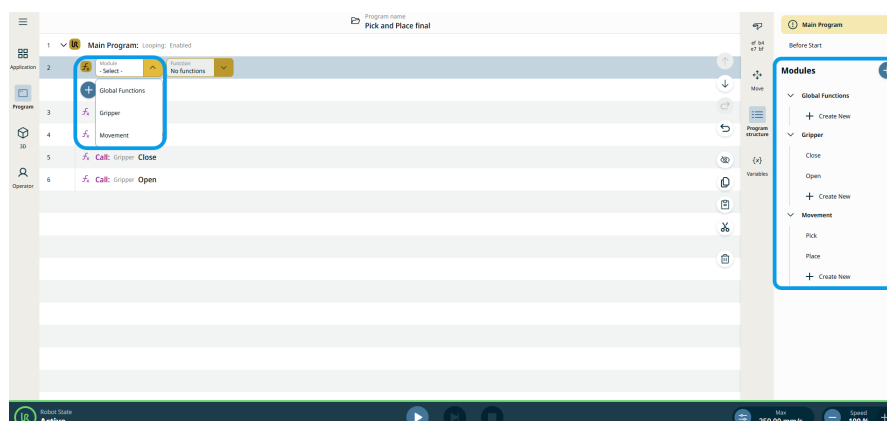
1. Open an existing program where commands are grouped in modules and functions. In this example, open the [Pick and Place program](#).
2. Go to the main program screen. See [Command Nodes](#).
3. Tap the **Call** icon in the Commands toolbox.



An editable **Module** field is inserted into the program tree.

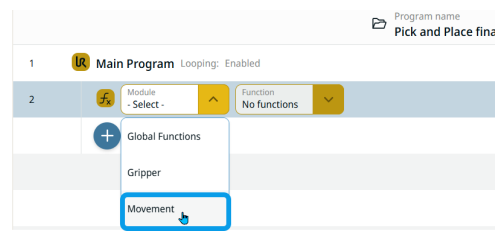
4. Tap the **Module** field.

In addition to **Global Functions**, you can see the created modules: **Gripper** and **Movement**. The list of modules correspond to the ones in the Modules tree in the multitask screen.



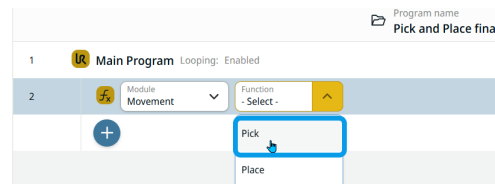
Call Movement module

5. Select **Movement** in the Module field.



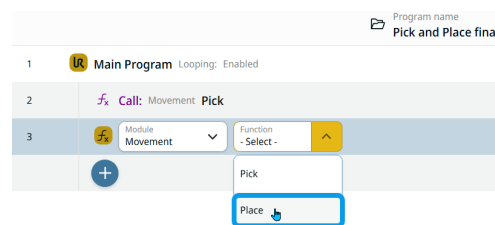
Call a Pick function

6. After the module is selected, the **Function** field is activated. Select **Pick** function.



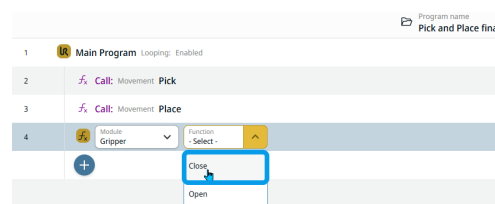
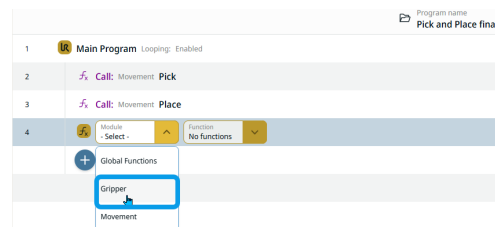
Call a Place function

7. Repeat steps 2-5.
8. In the same **Function** field, select **Place** function.



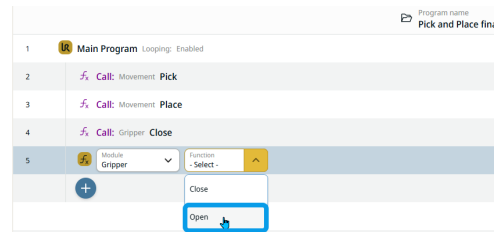
Call Gripper module and Close function

9. Tap the add icon and then the Call icon.
10. Select **Gripper** in the Module field.
11. In the **Function** field, select **Close** function.



Call an Open function

- Repeat steps 2-5.
- In the same **Function** field, select **Open** function.



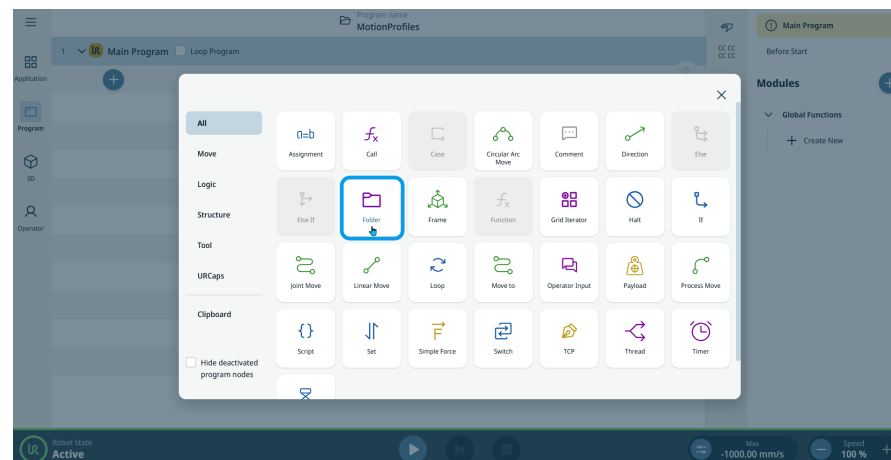
7.7.2. Folder

Description

The **Folder** command allows you group nodes together in a folder to create an easy-to-read overview of your robot program.

To access Folder command

- Go to the main program screen. See [Command Nodes](#).
- Tap the **Folder** icon in the Commands toolbox.

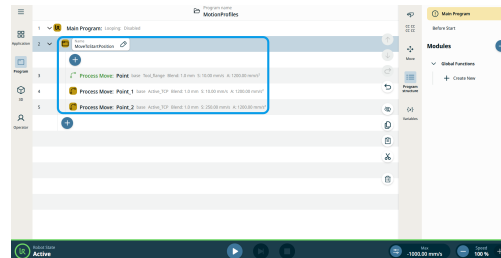


A Folder node, which is the **Name** field, is inserted into the program tree.

- Tap the **Name** field.
- Edit the folder name and tap **Confirm**.
- You can see two add icons below the folder name:
 - You can add a new folder under the chosen program.
 - You can also add a subfolder to your created folder.

Example Create Folder of all Move Commands

1. Tap the **Folder** icon.
2. Tap the **Name** field.
3. Edit folder name to **MoveToStartPosition**.
4. Tap the **first add icon** below the folder node.
5. Add the move-related commands in the program.

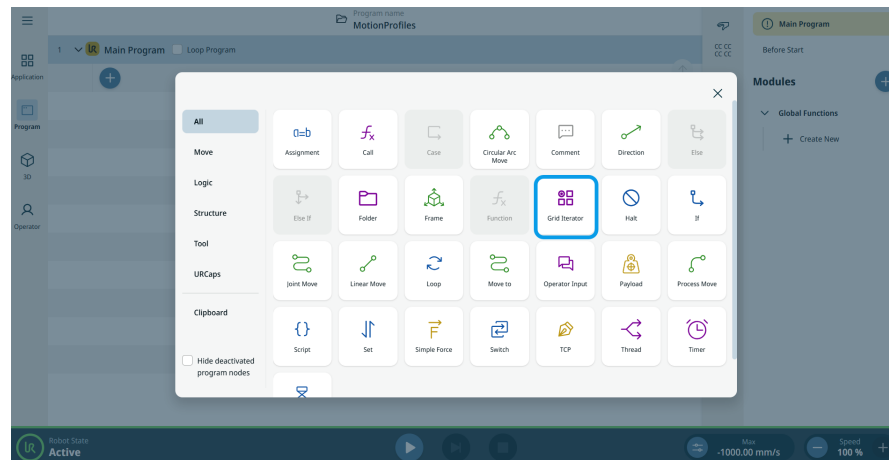


7.7.3. Grid Iterator

Description The **Grid Iterator** command enables you to move a grid from the first position the next to first position. The **Grid Iterator** command is used in combination with a grid created in the Grids submenu under the Application menu, located on the left toolbar. See [Grids Application](#).

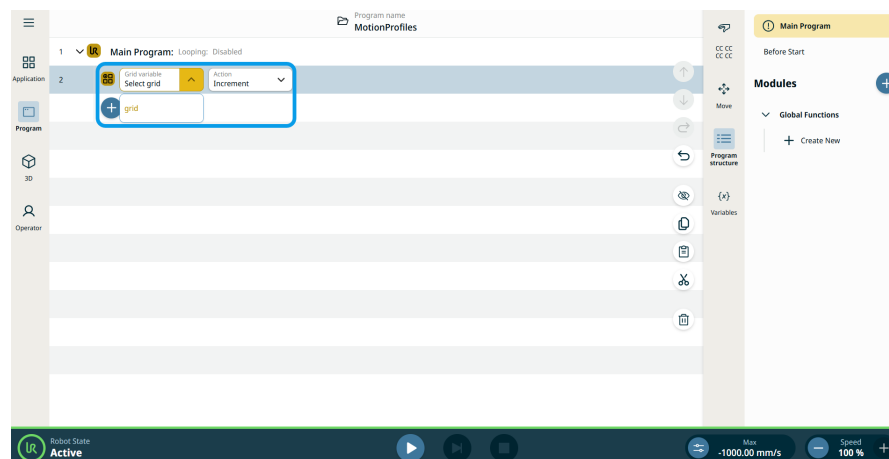
To access Grid Iterator command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Grid Iterator** icon in the Commands toolbox.



A node is inserted into the program tree. The node has an editable field named **Grid variable** and **Action**.

3. Tap the **Grid variable** field.
4. Choose the particular grid you created in the **Grids submenu** of Application.



5. Tap the **Action** field.
6. Select either **Increment** or **Reset**.

Each increment moves the variable to the next grid position. The pose variable, **grid_iterator**, can be used as the target in a Move node instead of a fixed waypoint, enabling sequential motion through the grid layout.

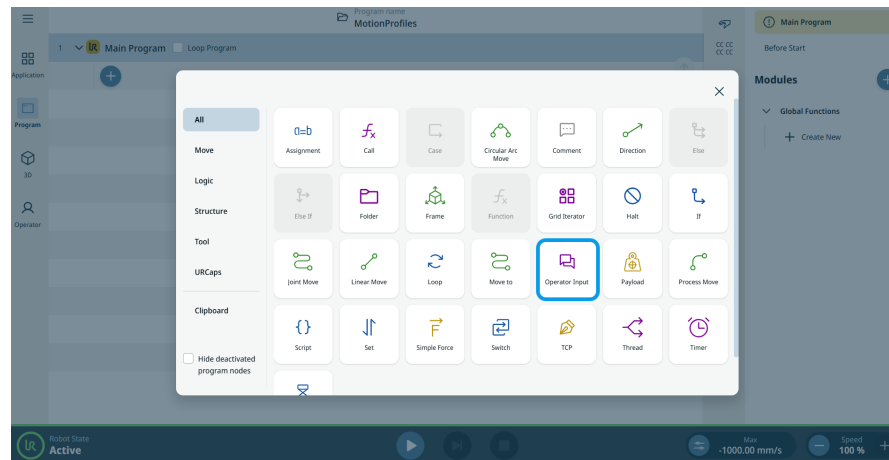
7.7.4. Operator Input

Description

The **Operator Input** command is used to request input from the operator or display a message during program execution. The command allows you to define a question, select the input type, and store the response in a variable.

To access Operator Input command

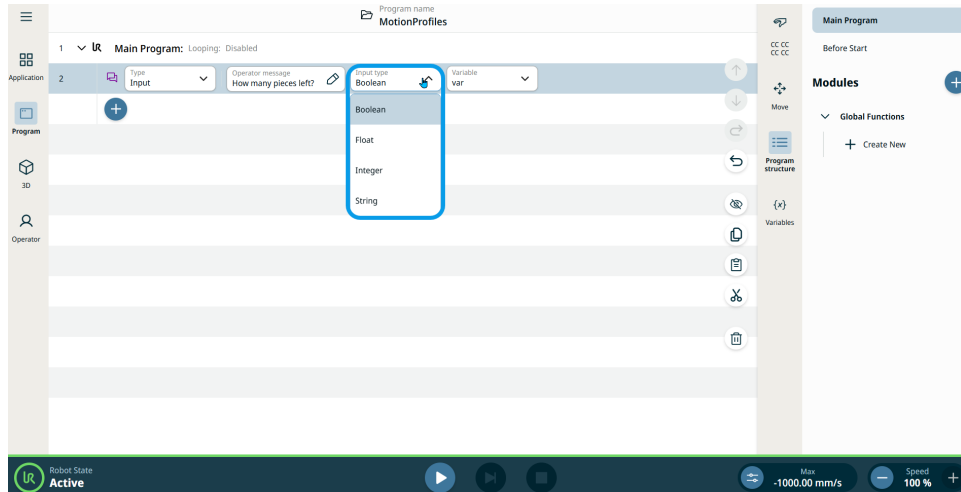
1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Operator Input** icon in the Commands toolbox.



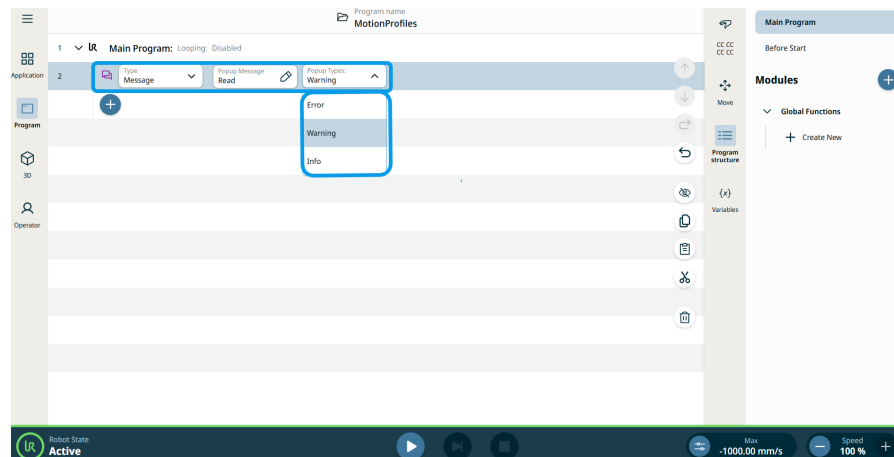
A node with four editable fields is inserted into the program tree.

The editable fields are:

- Type
 - Operator message
 - Input type
 - Variable Name
3. Tap the **Type** field and select either **Input** or **Message**.
 4. When you select **Input**, enter and edit your note or message in the **Operator message** field and tap **Confirm**.
The program stops at this point, displays the message from the Message field, and waits for a response.
The type of response is determined by the Input Type field.
 5. Tap the **Input type** field.



6. Select one of the four input types given:
 - **Boolean** - Select Yes/No buttons.
 - **Float** - Enter a decimal value.
 - **Integer** - Enter a whole number.
 - **String** - Enter text.
7. Tap the **Variable Name** field.
8. Tap the **Edit** tab inside the field to edit the variable name of your preference.
9. When you select **Message** in the **Type** field, the node in the program tree modifies from having four editable fields to two:
 - **Popup Message**
 - **Popup Types**



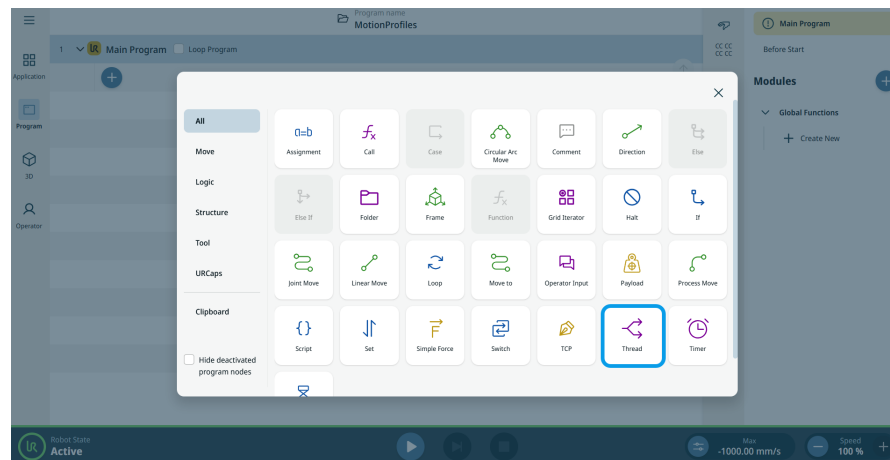
10. Tap **Popup Message** field.
11. Create your message and tap **Confirm**.
12. Tap **Popup Types** field.
13. Select the type from the three choices:
 - **Error**
 - **Warning**
 - **Info**

7.7.5. Thread

Description The **Thread** command enables you to control an external machine independently of the robot arm. A thread can communicate with the robot program with variables and output signals.

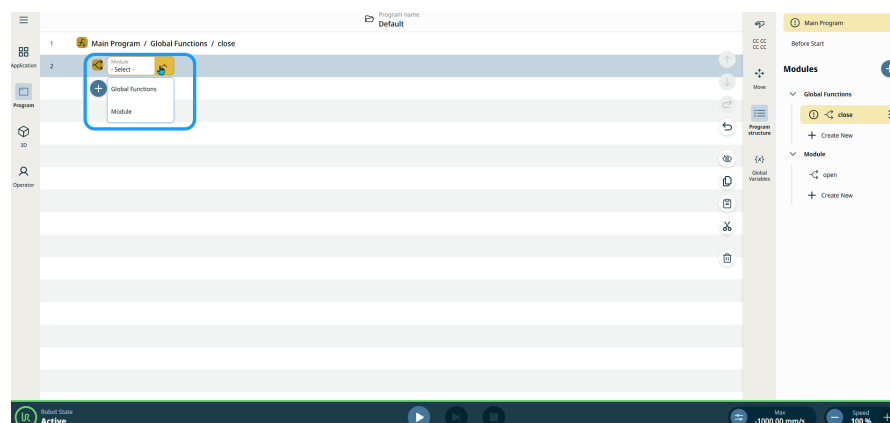
To access Thread command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Thread** icon in the Commands toolbox.



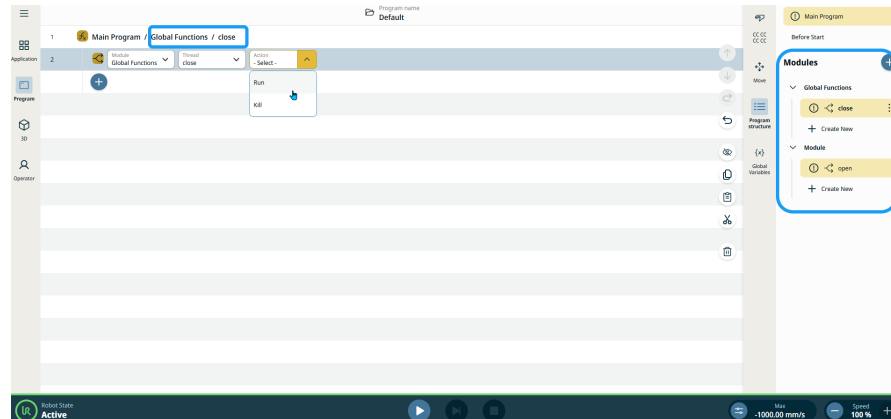
A node with a **Module** field is inserted into the program tree.

3. Tap the **Module** field, and two selections are given:
 - Global Functions
 - Module

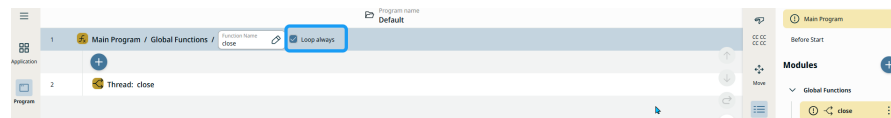


4. Select either **Global Functions** or **Module**, and a **Thread** field appears at the right.

5. Note: For the Thread field to be active and selectable, you first have to create new global functions and modules in the Program Structure and marked as thread. See [Modules and Functions](#).



The function/module is now marked as a thread, and tap the **Loop always** checkbox to make the thread execution loop when done.



6. Tap the **Thread** field and select the function or module you created. An **Action** field is inserted further.
7. Tap the **Action** field and select **Run** or **Kill** to start or stop the thread execution at that point.

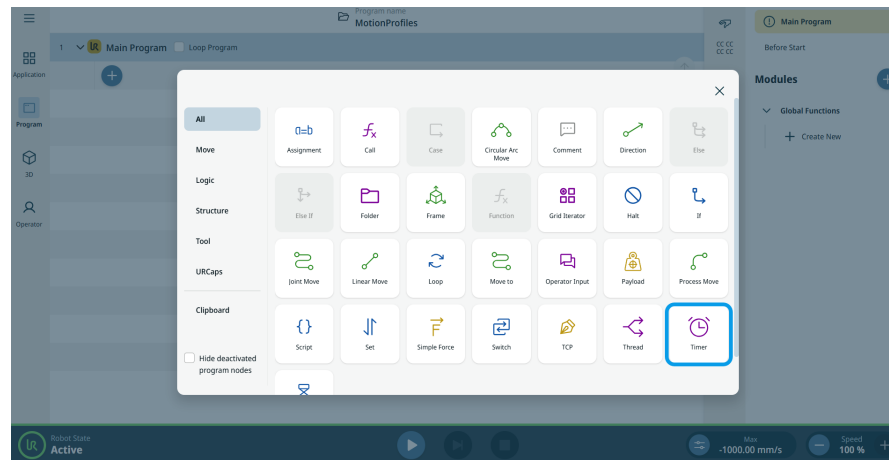
Threads that are meant to run for the entire duration of the program execution can be started from **Before Start** section of the **Main Program**.

7.7.6. Timer

Description The **Timer** command is used to measure real time during program execution. Each timer node adds a timer variable or references an existing timer variable.

To access Timer command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Timer** icon in the Commands toolbox.



A node with two editable fields, **Variable** and **Action**, is inserted into the program tree.

3. Tap the **Variable** field.
4. Tap **Rename** if you choose a new name, and **Confirm**.
5. Tap the **Action** field.

Three options are available:

- Start
- Pause
- Reset

Note: Timer can be started, paused, and reset to 0.

The elapsed time since the timer started can be read from the timer variable.

7.7.7. Control Actions

Description

Main robot program could be controlled by the [6.4.10 Logic Program on page 107](#) through the Control Actions command node. Unlike [7.7.5 Thread on page 217](#) the Logic Program is executed independently of the main robot program and could be used to control robot and main program execution with added logic.

Control Action command node is not available in the main robot program.

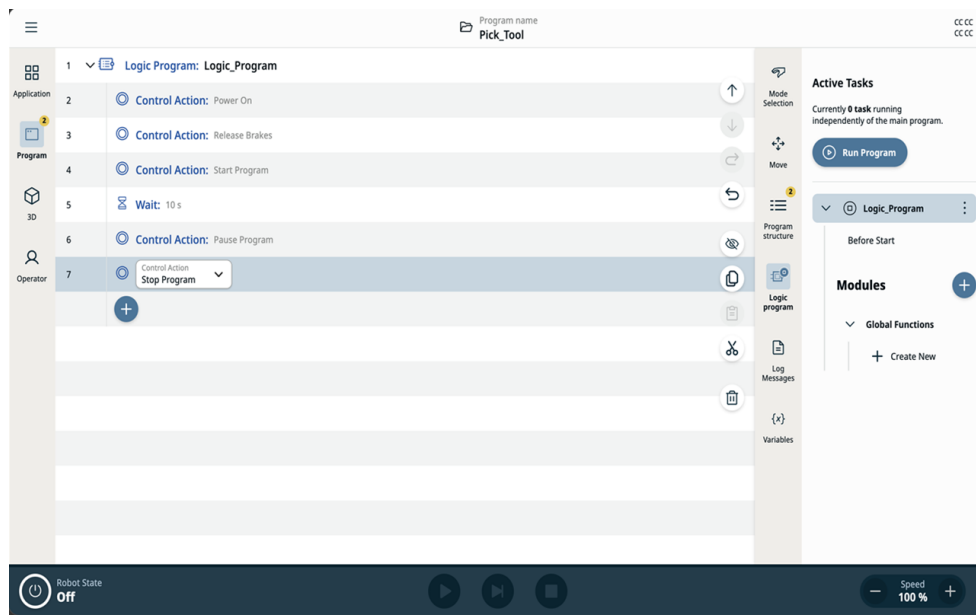


Types of Control Action

Programming of Control Action is done in manual mode, but they can be executed only when the robot is operated in Remote Mode.

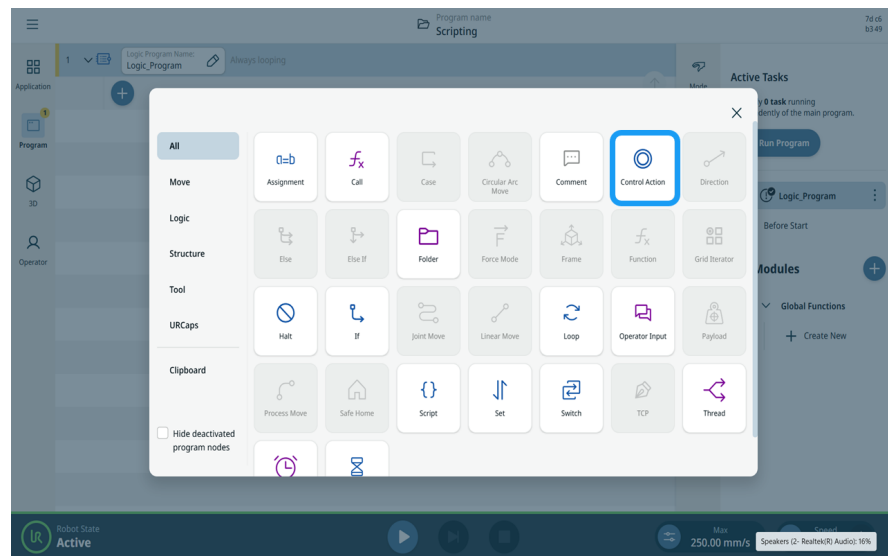
These are the various Control Actions that can be added:

- **Power On** - Power on the robot arm.
- **Power Off** - Power off the robot arm.
- **Release Brakes** - Release the brakes of the robot.
- **Start Program** - Start the currently loaded program.
- **Pause Program** - Pause the currently loaded program.
- **Stop Program** - Stop the currently loaded program.

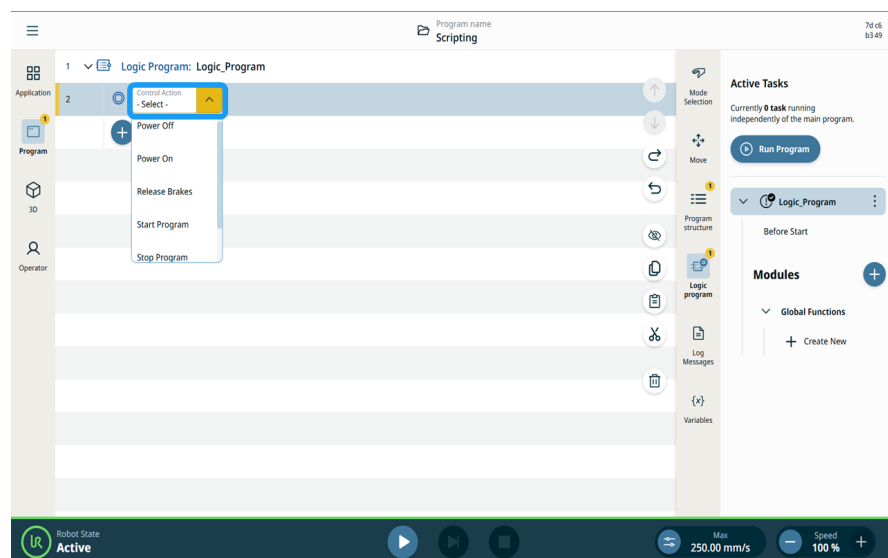


Control Action in the command nodes

1. Tap the **Add icon** and select the **Control Action** node.



2. The Control Action is inserted into the program tree. Tap the **Control Action** field to select the specific action you want to use.



7.8. Tool Command Nodes

Description

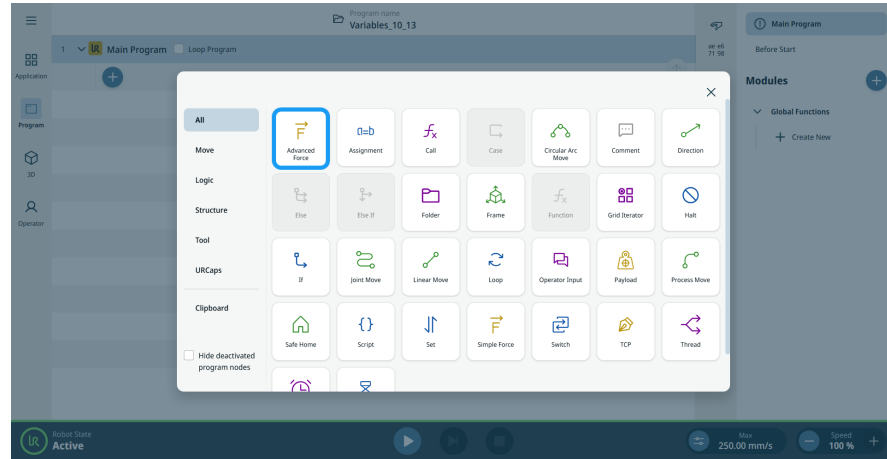
You use Tool nodes to configure end-effector behavior by setting the active TCP and payload at the right points. Any URCaps you install in the System Manager add their nodes after the Tool category.

7.8.1. Advanced Force

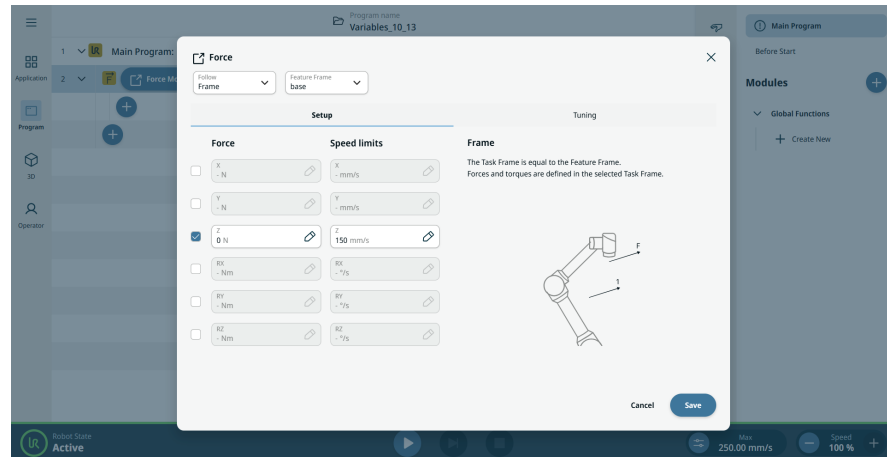
Description The **Advanced Force** command lets you set force and speed limits in the X, Y, Z, RX, RY, and RZ directions.

Access the Advanced Force command

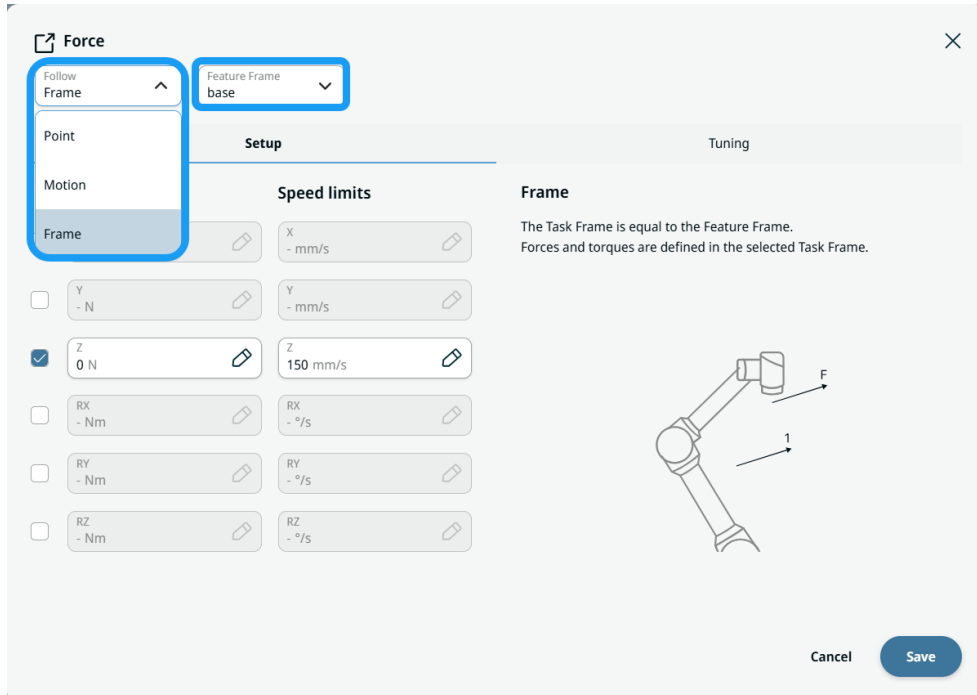
1. Open the main program screen. For more information, see [Command Nodes](#).
2. Tap the **Advanced Force** icon in the Commands toolbox.



3. The system inserts a **Force Mode** node into the program tree.
4. Tap the **Force Mode** node. The **Force** screen opens with tabs and the **Setup** and **Tuning** panels.



Follow and Feature Frame fields



Force

Follow: **Frame** | Feature Frame: **base**

Setup | Tuning

Speed limits

☐ X - mm/s	☐ Y - mm/s
☒ Z 0 N	☐ Z 150 mm/s
☐ RX - Nm	☐ RX - °/s
☐ RY - Nm	☐ RY - °/s
☐ RZ - Nm	☐ RZ - °/s

Frame

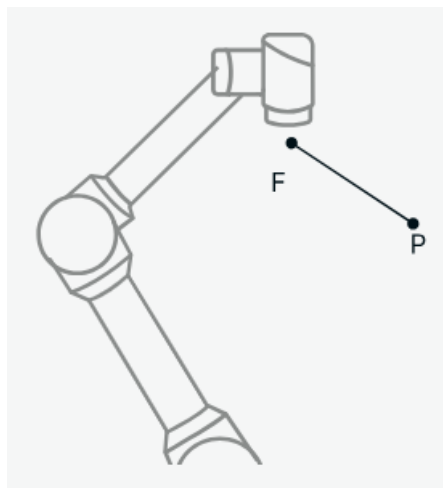
The Task Frame is equal to the Feature Frame.
Forces and torques are defined in the selected Task Frame.

Cancel | Save

In the **Follow** field, you can select one of three modes: **Point**, **Motion**, or **Frame**.

1. Point

- The Feature Frame is a point.
- The task frame Y+ axis points toward the selected point.
- Use this mode to apply force toward the center of a circular path.



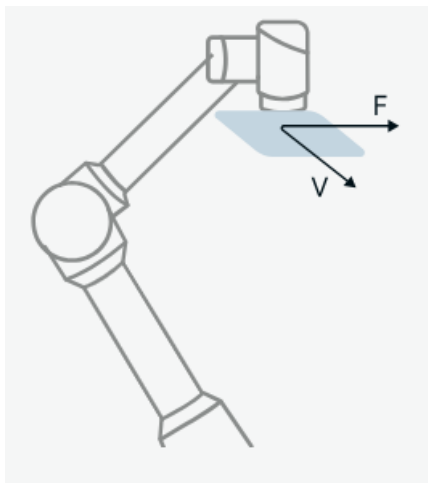
NOTICE

Do not specify forces in the X or Z direction.

2. Motion

The task frame follows the TCP movement direction:

- The X axis aligns with the motion projected in the X, Y plane of the Feature Frame.
- The Z axis is parallel to the Z axis of the Feature Frame.

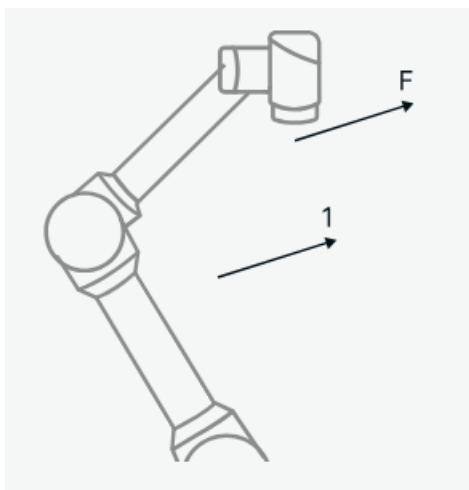


NOTICE

This mode is active only while the robot is moving. The orientation remains from the most recent motion.

3. Frame

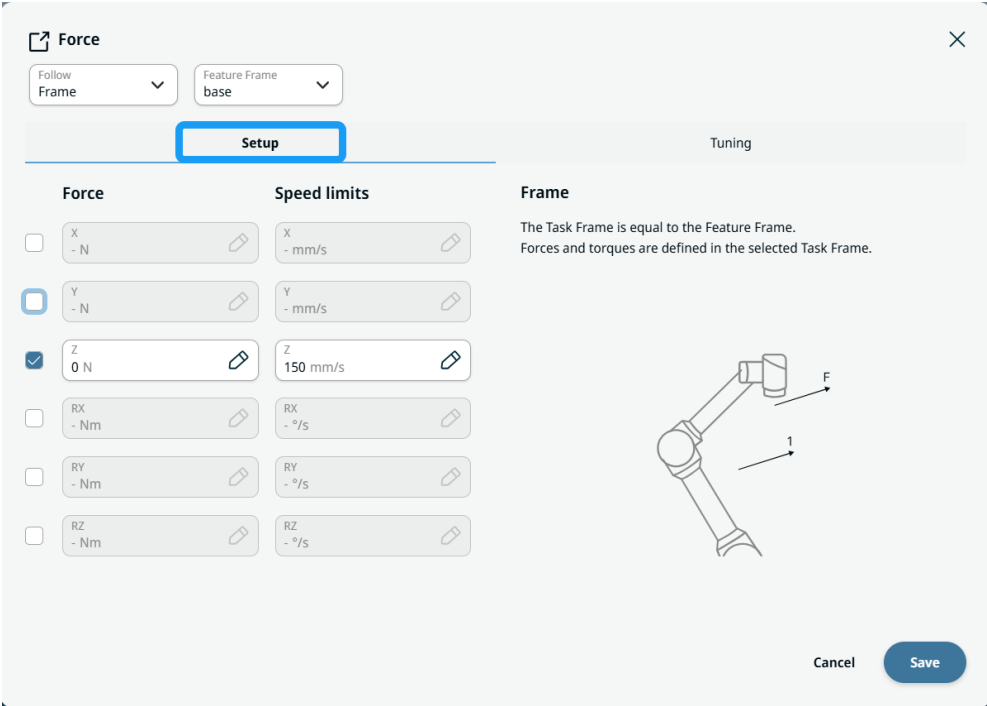
- The task frame matches the Feature Frame.
- Forces and torques are defined in the selected task frame.



In the **Feature Frame** field, you can select:

- **Base**
- **TCP**
- **World**
- **Flange**
- **Frame**

Setup panel



Force

Follow Frame ▼ Feature Frame ▼ base

Setup Tuning

Force

☐ X - N ☐ Y - N ☒ Z - N ☐ RX - Nm ☐ RY - Nm ☐ RZ - Nm

Speed limits

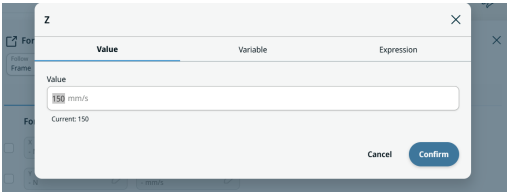
X - mm/s Y - mm/s Z - mm/s RX - °/s RY - °/s RZ - °/s

Frame

The Task Frame is equal to the Feature Frame.
Forces and torques are defined in the selected Task Frame.

Cancel **Save**

1. Tap the checkboxes for X, Y, Z, RX, RY, and RZ to activate them.
2. In the **Force** and **Speed Limits** fields, you can use the [Value, Variable, and Expression tabs](#) and the [Expression Editor](#).



Z

Value Variable Expression

Value: 150 mm/s

Current: 150

Cancel **Confirm**

3. After you set the force and speed limits, tap **Save**.

Tuning
panel

Force

Follow Frame Feature Frame base

Setup Tuning

Gain
Defines how fast the robot moves to reach the desired wrench.

Gain 1.0

High Gain - Quick response, may overshoot or become unstable.
Low Gain - Gentle, stable motion, slower to reach target.

Damping
Defines how quickly the robot 'forgets' the difference between measured and desired wrench.

Damping 0.005

High Damping - Slower convergence, more stable.
Low Damping - Faster convergence, may overshoot or become unstable.

Zero sensor
Sensor gets zeroed when entering force mode

☒ Zero sensor Wait time 0.02 s

Cancel Save

Tap **Tuning**, then use the **Gain**, **Damping**, and **Zero sensor** functions.

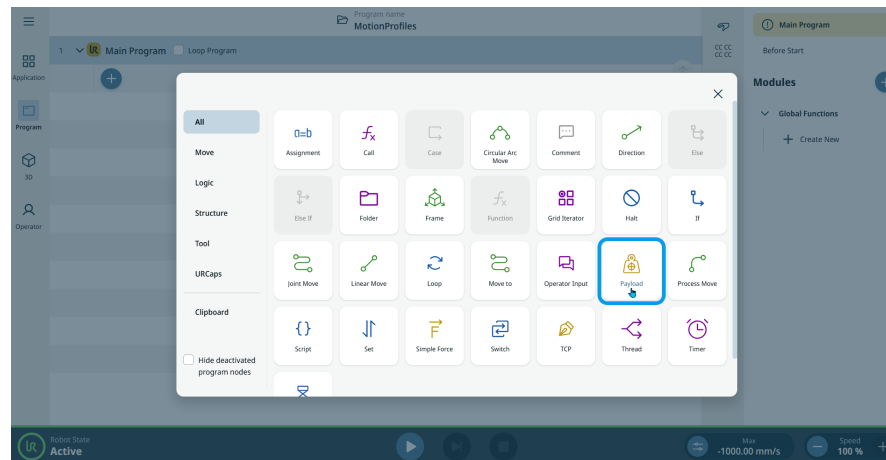
- **Gain** defines how quickly the robot moves to reach the desired wrench.
 - **Damping** controls how quickly the robot “forgets” the difference between the measured and desired wrench.
 - **Zero sensor**: The sensor is zeroed when entering force mode.
1. Use the **Gain** slider or tab to set the gain level.
 2. Use the **Damping** slider or tab to set the robot damping level.
 3. Select the **Zero sensor** check box to enable it and set the wait time.

7.8.2. Payload

Description The **Payload** command allows you to configure the payload of the robot, which is the combined weight of everything attached to the robot tool flange, and the center of gravity to predefined values or custom ones.

To access Payload command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Payload** icon in the Commands toolbox.



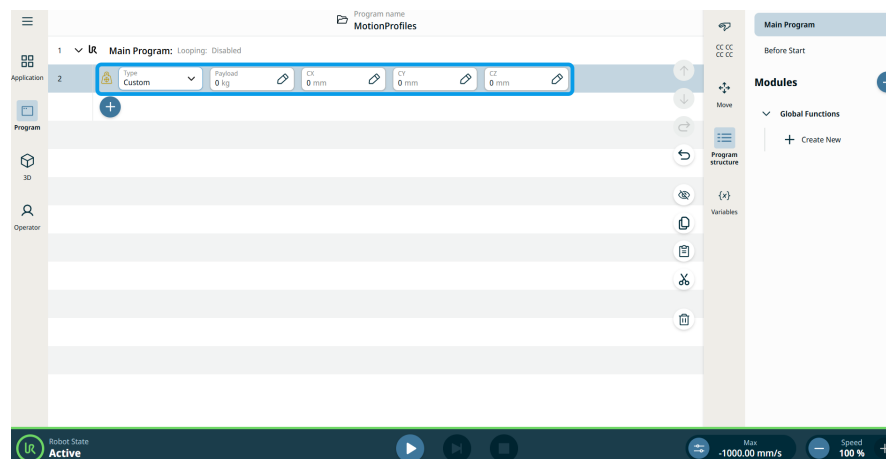
A node with an editable **Type** field is inserted into the program tree. The fields are:

- Custom
 - Robot
3. Tap **Custom**.

When this is selected, the payload and center of gravity can be directly set.

Four additional fields appear on the right side:

- Payload
- CX
- CY
- CZ





4. Tap the **Payload** field.

This contains a list of all end effectors defined in the End Effectors application inside the Application menu, which includes the tool flange predefined end effector. See [End Effectors Application](#).

Three tabs to input the payload are available:

- Value
- Variable
- Expression

Note: The Variable tab is inactive.

5. Make the input in each tab and tap **Confirm**. See [Value-Variable-Expression Tabs](#).

6. Tap the **CX** field and repeat step 5.

7. Tap the **CY** field and repeat step 5.

8. Tap the **CZ** field and repeat step 5.

7.8.3. TCP

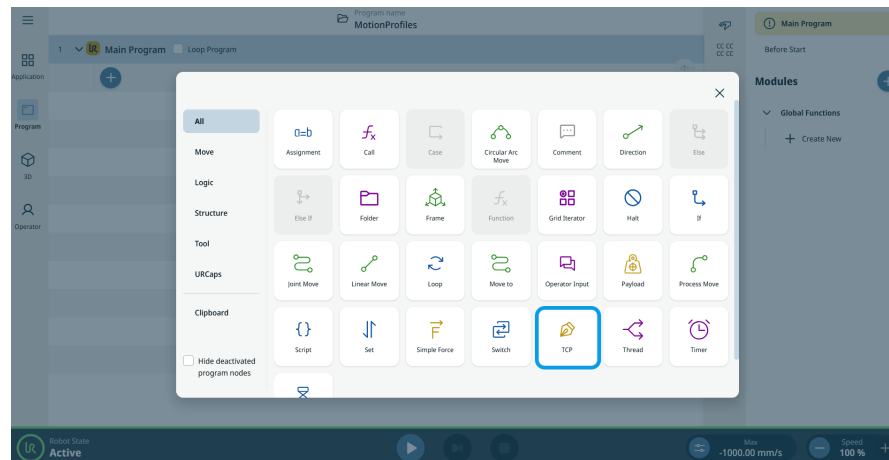
Description

The **TCP** command enables you to change the active Tool Center Point (TCP) during program execution. It uses the TCP already defined in the End Effectors application inside the Application menu. See [End Effectors Application](#).

The TCP command updates the active TCP when changes happen mid-program to ensure the robot's motion planning stays accurate.

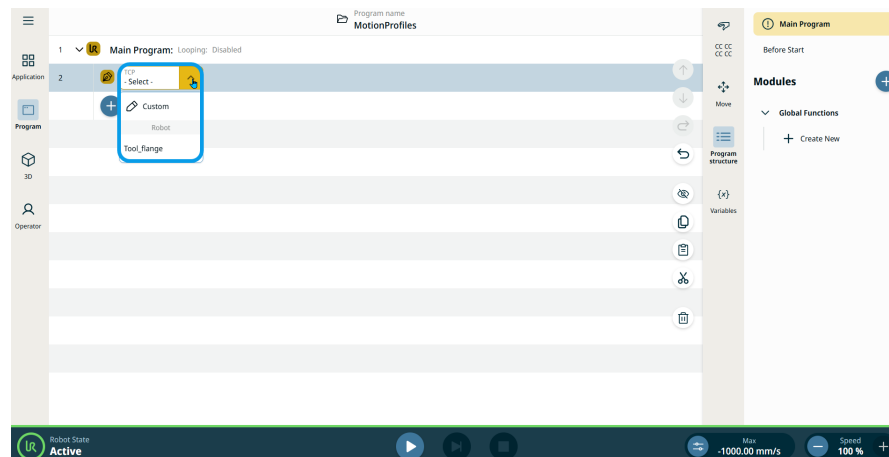
To access TCP command

1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **TCP** icon in the Commands toolbox.

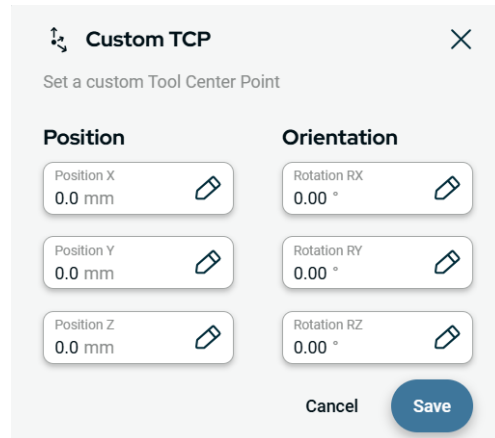


A node with an editable **TCP** field is inserted into the program tree.

3. Tap the **TCP** field and select your preference.



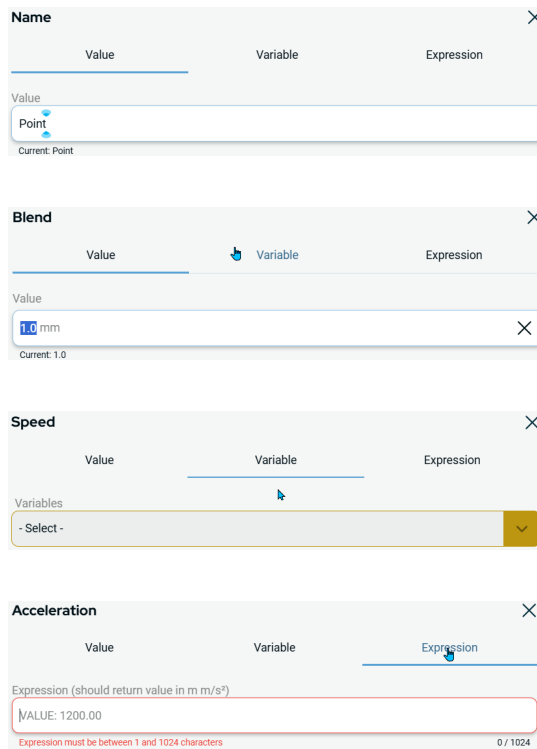
4. Select **Custom**.
A **Setup** tab is inserted to the right.
5. Tap the **Setup** tab.
The **Custom TCP** screen appears.



6. Tap the **Position X**, **Position Y**, **Position Z** tabs in the **Position** column, and enter the Value, Variable, and Expression on their respective tabs, and **Confirm**. See [Value-Variable-Expression Tabs](#).
 7. Tap the **Rotation RX**, **Rotation RY**, **Rotation RZ** tabs in the **Orientation** column, and enter the Value, Variable, and Expression on their respective tabs, and **Confirm**. See [Value-Variable-Expression Tabs](#).
 8. Tap **Save**.
-

7.9. Value-Variable-Expression Tabs

Description The **Value**, **Variable**, **Expression** tabs appear as pop-up dialog in opening commands. They are common components used in almost all commands.



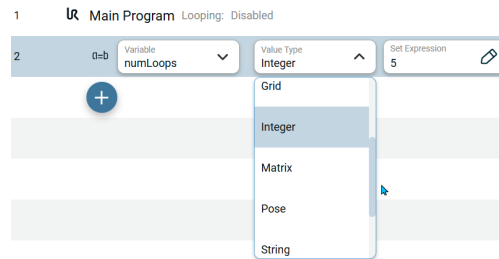
Value tab description The **Value** tab is used when you know the exact fixed value of a program. It restricts you to values appropriate for the given variable. The **Value Type** drop-down tab in the **Assignment** command contains the type of variable you are creating and has one of the following values:

Type	Allowed Values/Examples	Examples
Array	A list of values	
Boolean	True/False	True
Float	A decimal point value	1.23
Grid	A grid defined in the application	
Integer	A whole number	5
Matrix		
Pose	A pose for the robot to move	p[0,0,0,0,0,0,]
String	Text	"Hello World"
Waypoint	A waypoint is defined in the program	



Example use of Value tab Create a restrictive value appropriate for a given variable, which runs five times.

1. In the left toolbar, tap the **Program** icon.
2. Tap the **add** icon.
3. Tap the **Assignment** node command. See [Assignment](#).
4. On the **Variable** field, tap **Rename**.
5. Rename to **numLoops**.
6. Tap **Confirm**.
7. Set **Value Type** to **Integer**.
8. Tap **Set Expression** and enter **5** in the **Value** tab.
9. Tap **Confirm**.



Note: As your variable is integer, the value only allows you to enter whole numbers.

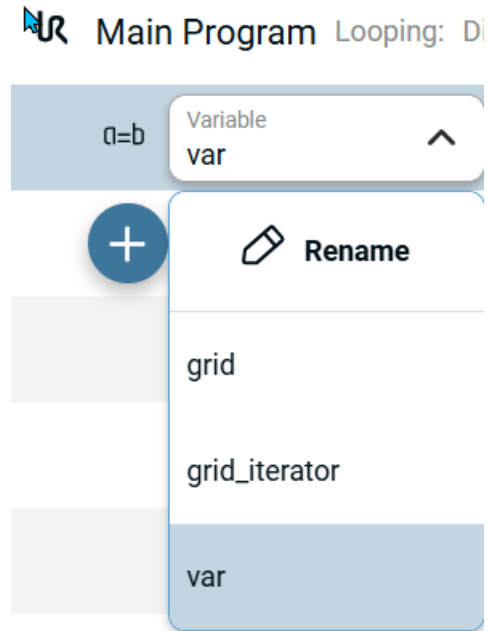
Variable tab description

The **Variable** tab allows you to pick from a list of variables defined in the program or application. Only variables of the appropriate type is shown. When you set the value of a digital output, only Boolean variable type is shown. When you set the value of an analog output, only Integer or Float variable type is shown.

Example use of Variable tab

The **Variable** drop-down menu contains the name of the variable. By default, the variable is **var**.

1. Got to steps 1-3 in the use of Value tab.



The tab contains the list of variables already in the program, such as from another Assignment node, Joint Move or Linear Move nodes, and other application nodes, grid iterators.

2. Tap **Rename** to change the name of the created variable.
3. To iterate and add options, repeat the process.

7.9.1. Expression Editor

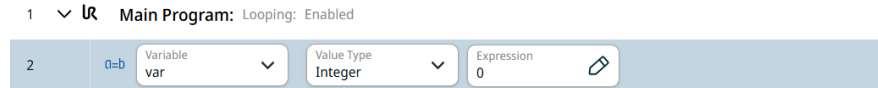
Description

The **Expression** tab enables you to insert an expression, which should be a valid URScript and evaluates to the correct type. An expression in an **If** node evaluates to a Boolean. Whereas a String type requires enclosing quotes to be used. Furthermore, the **Expression** tab allows you to easily insert a variable into the expression.

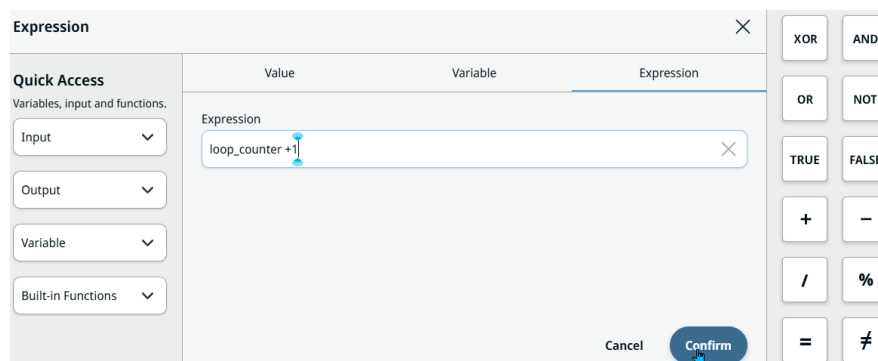
Example use of Expression tab

Create a program loop counter.

1. In the main navigation, tap the **Program** icon.
2. Tap the **add** icon.
3. Tap the **Assignment** node command. See [Assignment](#).
4. Select **Integer** in the **Value Type** field.
5. Tap the **Expression** field.



6. Tap the pop-up **Expression** tab, and the **Expression Editor** appears. Here you can set the expression to **loop_counter +1** and **Confirm**.

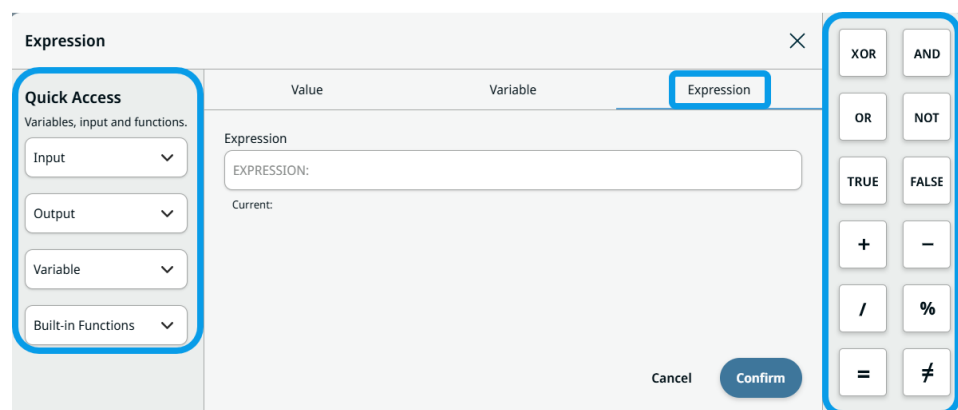


Expression Editor

In the Expression Editor, you can find shortcuts for inserting functions, IOs, and logic operators into expressions.

On the **left column**, you see the Quick Access section that allows you to easily insert an Input, Output, Variable, or Built-in Function at the current cursor position. From PolyScope X 10.14, input and output signals are discovered dynamically from all registered sources and grouped by source in the selectors. See [Expression Editor Signals](#).

On the **right column** are found buttons to insert common logic operators and values, such as AND, OR, NOT, TRUE, FALSE. This functionality has been added to every tabbed into dialog, as well as to few node fields that previously only accepted a string, such as the **If** node.



7.9.2. Expression Editor Signals

Description	From PolyScope X 10.14, the Expression Editor dynamically discovers and displays I/O signals from all registered sources. Signals are no longer limited to a fixed built-in list. When you open the Input or Output selector in the Expression Editor, available signals are grouped by source. Built-in sources include Control Box I/O, standard tool connector I/O, and Smart I/O when a smart panel is connected. Signals from installed URCaps that register I/O sources also appear automatically in the same lists. Each signal is shown with its configured name when one is available. If no name is configured, the signal identifier is shown instead.
Input and Output tabs	In the Expression Editor Quick Access section on the left column: • The Input selector lists input signals from all available sources. • The Output selector lists output signals from all available sources. Signals are grouped under their source name in the dropdown. Select a signal to insert its script reference at the current cursor position in the expression.
Signal sources	The signals shown depend on the connected hardware and installed URCaps. Typical built-in source groups include: • Control Box I/O – wired and configurable control box digital and analog signals. • Tool I/O – signals on the standard tool connector. • Smart I/O – smart panel button, safety, and lamp signals when a smart panel is connected. Additional source groups appear when a URCap or platform extension registers I/O with PolyScope X.
Example	Use a named control box input in an If condition: 1. Add an If node and tap the condition field to open the Expression Editor. 2. In Quick Access, tap Input. 3. Expand the Control Box I/O group and select the desired input signal. 4. Complete the expression and tap Confirm. Named signals appear with the name you configured in the I/O setup screens. This makes it easier to recognize the signal you are selecting compared with generic index-based identifiers.

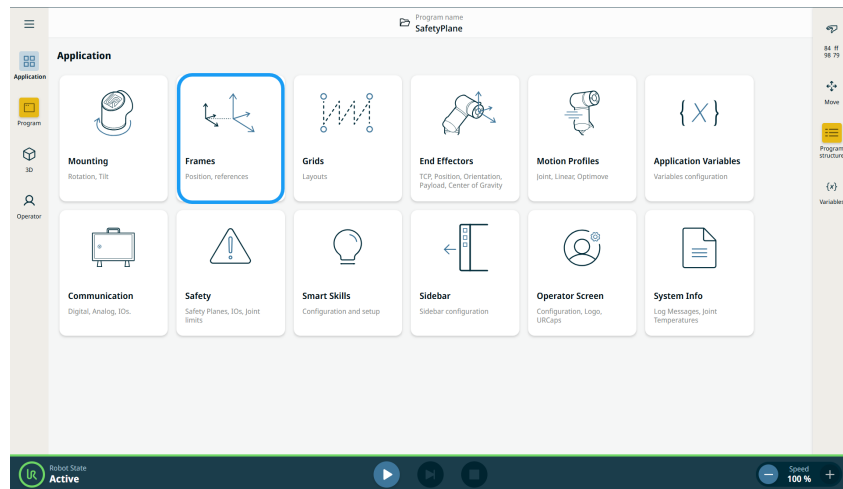


7.10. Frames

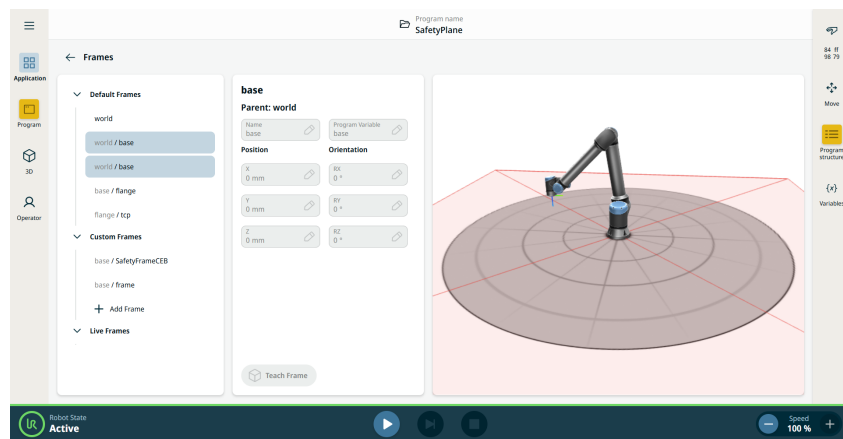
Description The Frames application allows you to define frames of reference for use in a program.

Using the Frames Application

1. Go to the Application tab screen. See [Application Tab](#).
2. Tap the **Frames** icon.



A screen appears, divided into three panels.



The left panel contains the **Default Frames**, **Custom Frames**, and **Live Frames**.

The center panel contains eight fields:

- Name field
- Program Variable field
- X Position field
- Y Position field
- Z Position field
- RX Orientation field
- RY Orientation field
- RZ Orientation field

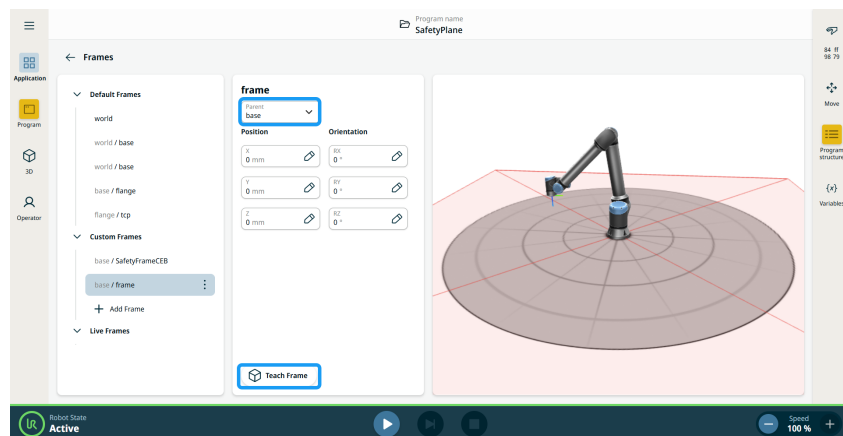
The right panel shows the robot arm in X, Y, Z coordinates.

The **Default Frames** section contains three predefined PolyScope X frames:

- **Base.** The center of the robot base. If the robot is mounted in a fixed location, this frame does not change, and you can define other fixed locations relative to it.
- **TCP.** The position of the active TCP. It updates as the robot moves when jogged or while a program runs.
- **World.** When the robot is mounted in a fixed location, this frame is the same as the Base frame. However, if the robot is mounted on a moveable rail or gantry, you can update the Base frame as the robot moves, while the World frame remains fixed.

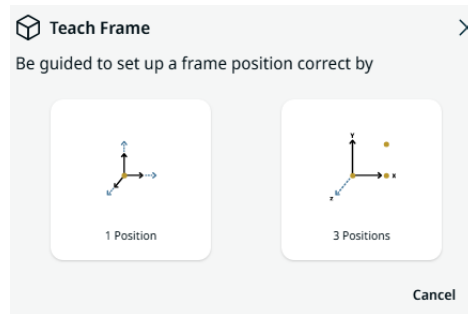
Selecting any of the predefined frames allows you to view their values. However, you cannot edit them.

3. Under **Custom Frames**, tap **+ Add frame**. The eight fields in the center panel are activated.
4. In the **Parent** field, select a frame. Frames are created relative to **Base**.
5. Enter the values in the **X, Y, Z Position** fields and tap **Confirm**.
6. Enter the values in the **RX, RY, RZ Orientation** fields and tap **Confirm**.
7. Tap the **Teach Frame** button at the bottom of the panel.

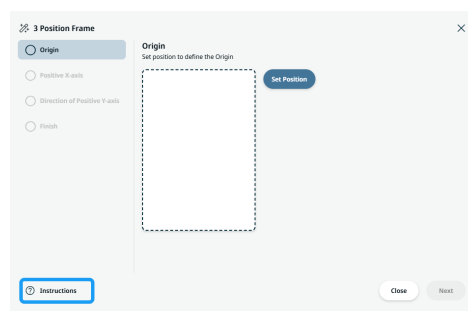


A pop-up appears with two options to set up a frame position.

- **1 Position**
- **3 Positions**

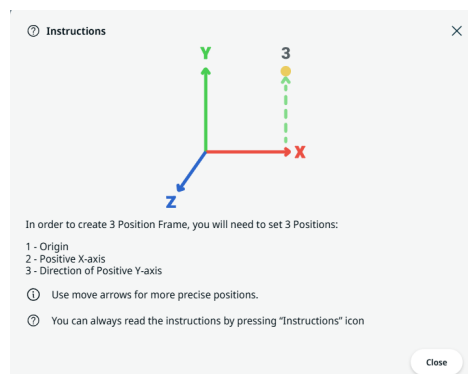


8. When you select **1 Position**, the 3D Viewer becomes the main screen. Use the plus-minus button to translate and rotate the robot arm's three coordinates.
9. Tap **Save**.
10. When you select **3 Positions**, a pop-up appears. Tap the **Instructions** icon on the bottom left.



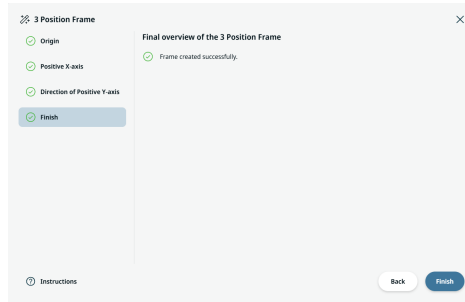
The Instructions box for creating a 3-Position Frame appears.

11. Tap **Set Position** for **Origin**.



12. The 3D Viewer becomes the main screen. Use the plus-minus button to translate and rotate the robot arm's three coordinates.
13. Tap **Save**.
14. An option to edit the position is provided. If no edit is needed, tap **Next**.
15. Tap **Set Position** for **Positive X-axis**.
16. Repeat steps 12-14.

17. Tap **Set Position** for **Direction of Positive Y-axis**.
18. Repeat steps 12-14.
19. Tap **Finish** when the frame is created successfully.



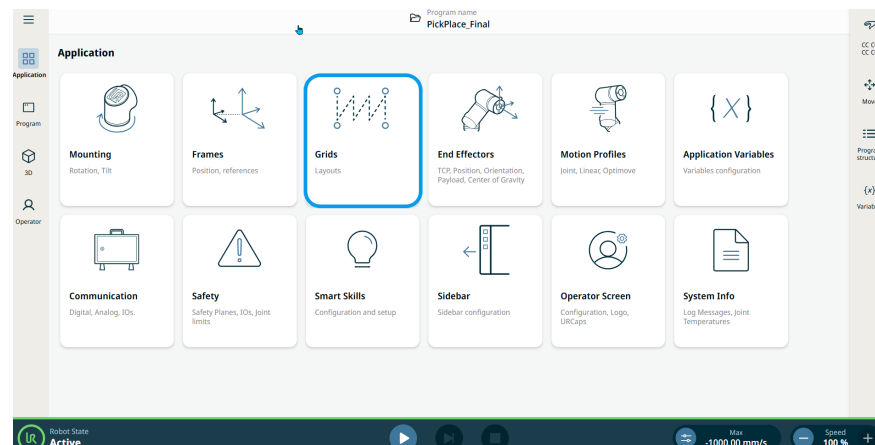
7.11. Grids Application

Description

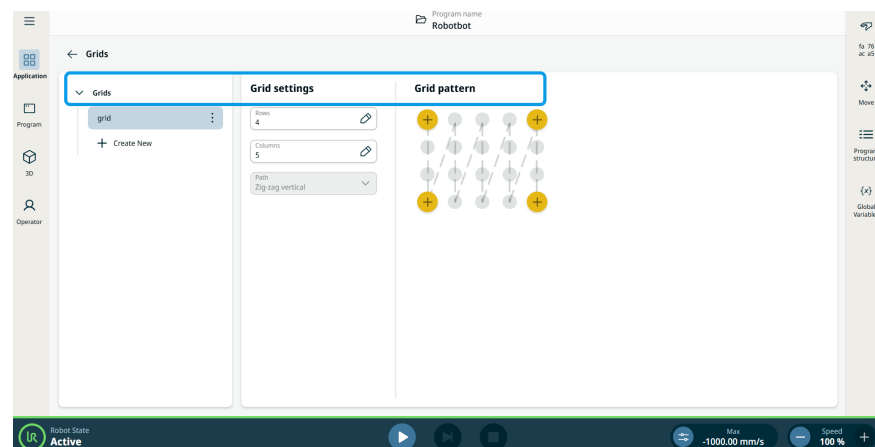
The **Grids** application is used to set up a grid, such as for palletizing, where you define four corners and the number of positions between them. This is like the palletizing wizard in PolyScope 5.

Using the Grids application functionality

1. Go to the Application menu screen. See [Application Tab](#).
2. Tap the **Grids** icon.



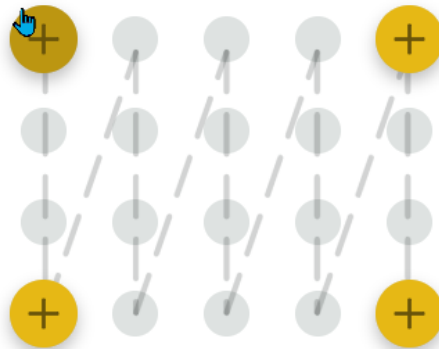
In the main screen, you can see the grids tree on the left panel, and the grid settings and grid pattern on the right panel.



3. Tap **Create New** on the grids tree.
4. Enter the grid name in the **create new grid** field and **Save**.
5. Enter the number of **rows** and **columns** in the fields provided under the grid settings and tap **Confirm**. The grid pattern corresponds with what you input in the grid settings.

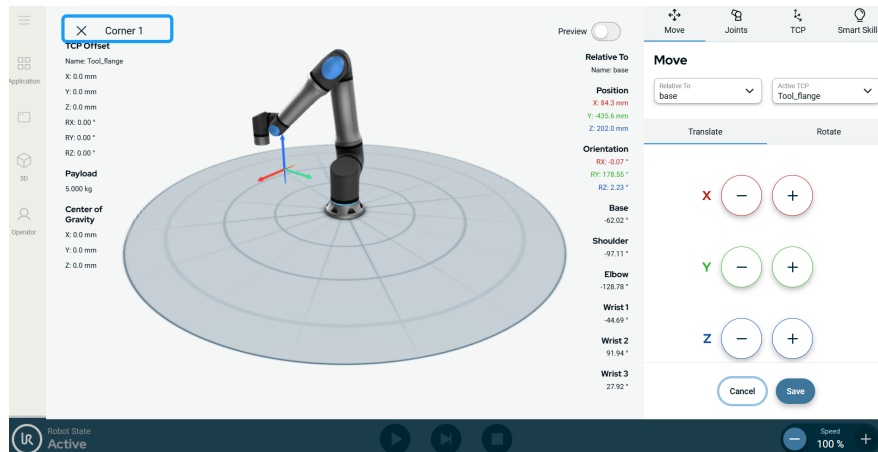
- In the grid pattern, tap the first corner (in yellow circle).

Grid pattern



You will be directed to the 3D viewer of the robot arm in X, Y, Z coordinates.

- On the right side of the screen, tap the **minus-plus** button in each coordinates to translate and rotate. Tap **Save**.



- Press and rotate, pinch and spread the 3D screen to zoom in or zoom out the robot parts, take a close-up view, rotate sideways, and other viewing direction.
- Repeat steps 8-11 thrice to make a grid pattern for the remaining corners 2-4.

When your Grid is set up, you can implement it in your program.

7.12. OptiMove

Description The **OptiMove** motion value is an advanced feature of PolyScope X **Joint and Linear Move** commands. See [7.5.4 Joint Move on page 174](#) and [7.5.5 Linear Move on page 180](#).

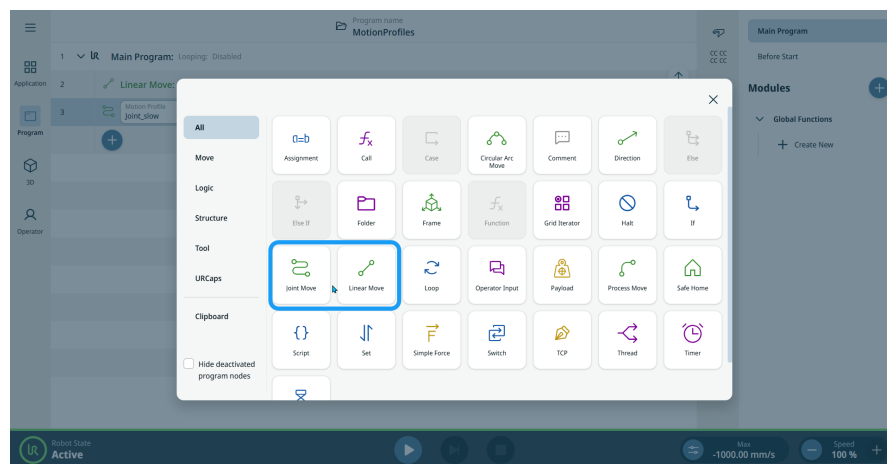
OptiMove allows you to have the following benefits:

- More dynamic robot acceleration and deceleration
- Smoother moves with less jerk
- More fluent acceleration and deceleration
- Higher travel speeds
- Reduced vibrations and loads on the robot and mounted equipment

The combined effect in **OptiMove** generally results in improved cycle time with more fluent motion and reduced strain on the robot.

To access OptiMove

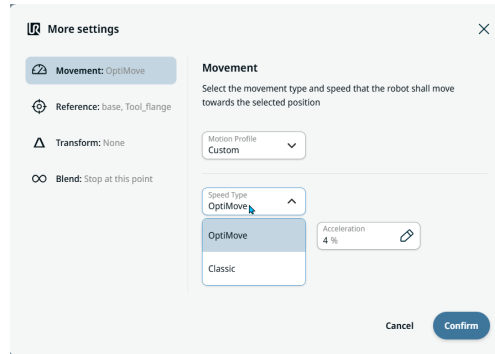
1. Go to the main program screen. See [Command Nodes](#).
2. Tap the **Joint Move** or **Linear Move** icon on the Commands toolbox.



A node is inserted into the program tree. The node has two editable fields and tabs,

and an ellipsis icon .

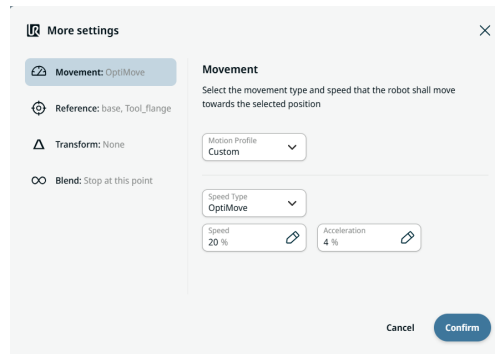
3. Tap the ellipsis icon and the **More settings** screen appears.



4. Tap **Movement** on the left side of the screen.

On the right side, two tabs and fields appear:

- Motion Profile
- Speed Type
- Speed tab
- Acceleration tab



5. In the **Motion Value** field, **OptiMove** is the default option. Choose **Classic** movement if you want to input speed and acceleration in degrees per second or mm per second.
6. Tap **OptiMove**.
OptiMove settings are specified in percent to simplify the usage and setup.
7. In the **Speed** tab, enter the percentage of your preferred speed and tap **Confirm**.
Speed tab sets the target travel speed as a percentage of the robot's maximum speed capability.
8. In the **Acceleration** tab, enter the percentage of your preferred acceleration and tap **Confirm**.

Acceleration tab sets the target torque limits during acceleration and deceleration as a percentage of the robot's maximum power.

8. First Program

Description The first program creation in PolyScope X enables you to teach the robot arm how to move using a series of waypoints to set up a path for it to follow. An End Effector tool should be attached at the end of the arm.
See [9.5 End Effectors Application on page 287](#) on how to configure the tooling at the end of the arm.

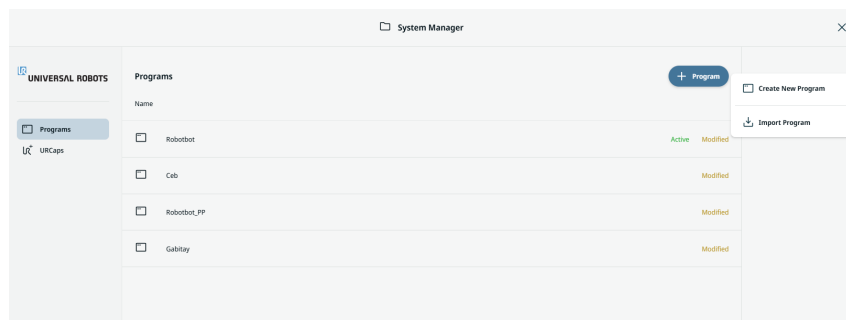
To create a simple program

These are the processes of this first program:

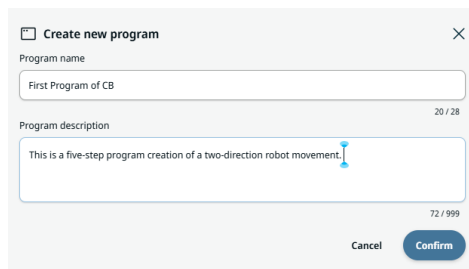
1. Move to position A.
2. Set an IO.
3. Wait for 3 seconds.
4. Unset the IO.
5. Move to position B.

System Manager access

1. In the main screen, tap the **header** to open the System Manager.
2. Tap the **Add Program** icon in the **System Manager** screen and choose **Create New Program**.



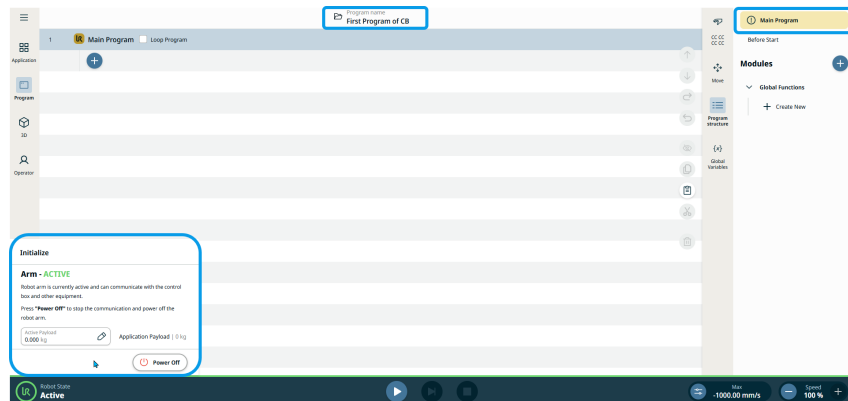
3. Enter the **program name** and **description** in the two fields provided in the **Create new program** box. Tap **Confirm**.



The main screen now shows your program name. At the multitask screen, you see the **Main Program** tab in yellow, indicating that program creation is to be started.

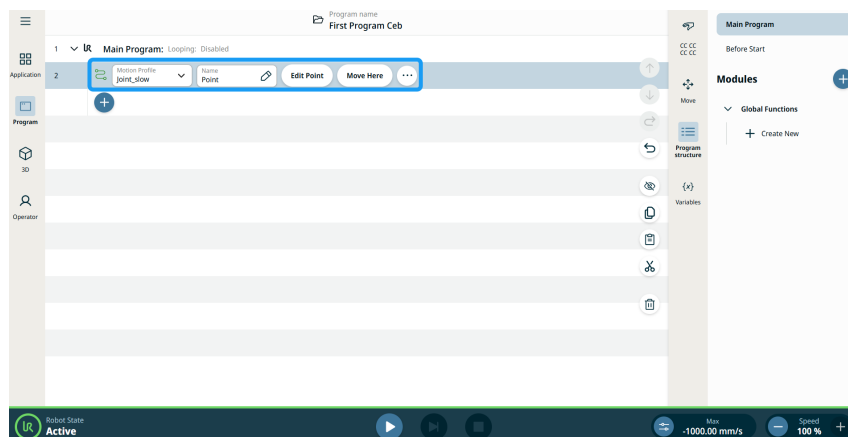
Initialize Robot

- On the left side of the footer, tap **Robot State** to initialize.
- Tap **Power On**, then **Unlock**. Then you see the robot state as **Active**, and the activated button/slider of **Play**, **High Speed Manual Mode**, and **Speed**.



- On the main navigation, tap the **Program** menu icon.
See [Command Nodes](#).
- Tap the **Joint Move** command. Note that in PolyScope X, the Waypoint node is merged to this command.

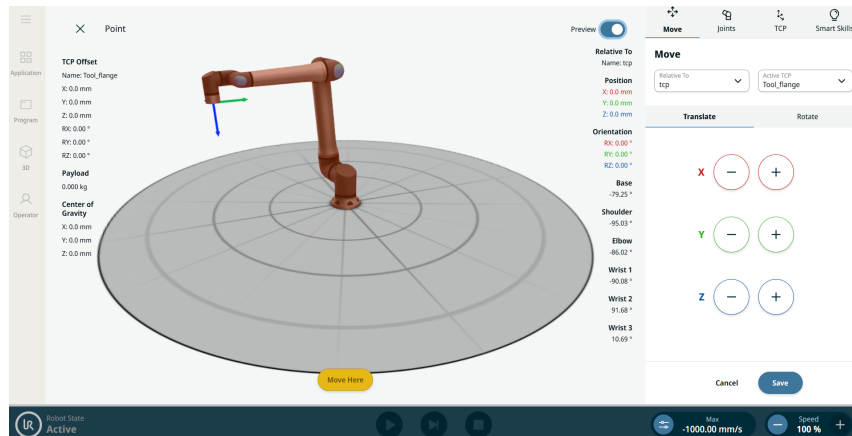
See the [Joint Move](#) command node description.



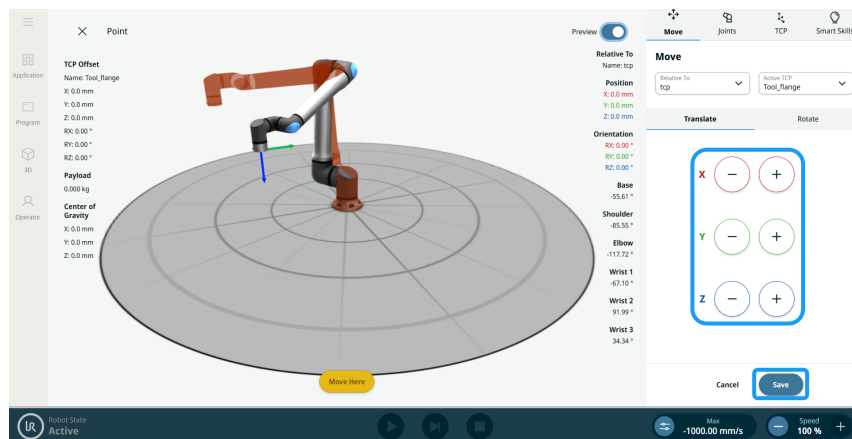
Two editable fields, two tabs, and an ellipsis icon are inserted into the program tree:

- Motion Profile
- Name
- Edit Point
- Move Here

- Tap the **Edit Point** tab. The screen with the robot arm in 3D appears. To the right of the screen, you see the Move menu.

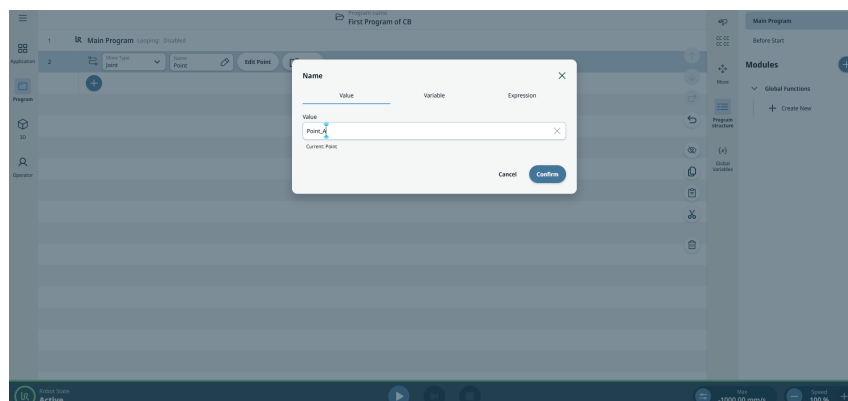


- Use the **plus-minus** button of the X, Y, Z axes to jog the robot and set the position of your waypoint and **Save**.



Name the first waypoint

- In the **Name** field, enter **Point_A** as the name in the **Value** editor tab and **Confirm**.

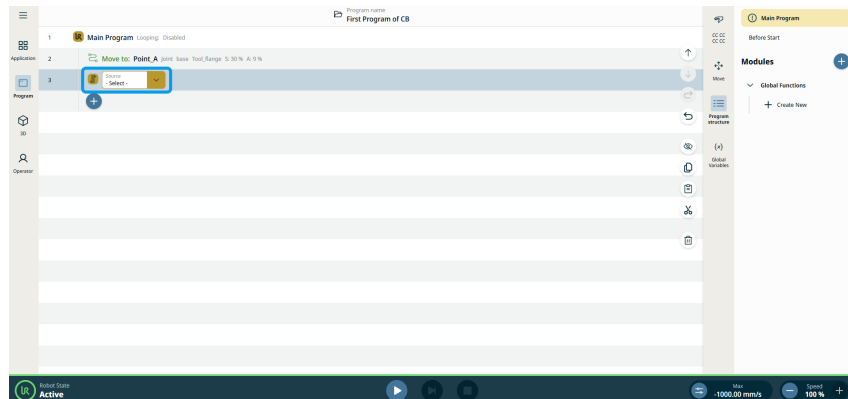




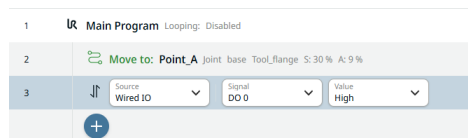
Second command: Set

Action: Set an IO for the end effector

11. Tap the add icon in the program tree and the **Set** command icon.
See [Set](#) command node.
See **Set** command node description in the Software Handbook.



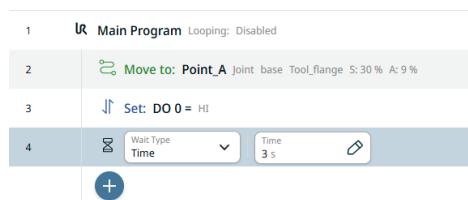
12. Tap the **Source** field and select **Wired IO**.
13. A **Signal** field is added. Select **DO 0**. A **Value** field is added to the right and select **High**.



Third command: Wait

Action: robot waits for a specified number of seconds

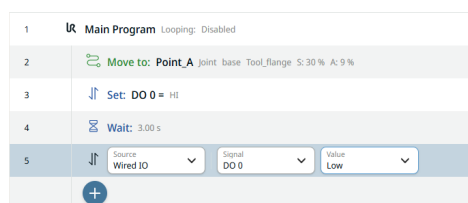
14. Tap the add icon and choose the **Wait** command.
See [Wait](#) command node.
15. In the **Wait Type** field, select **Time**, to make the robot wait for a specified number of seconds.
16. Tap the **Time** field and enter **3** (three seconds) in the **Value** tab and **Confirm**.



Fourth command: Set

Action: Unset an IO for the end effector

17. To create a second Set node, repeat steps 11 to 13.
18. In the **Signal** field, select **DO 0**. A **Value** field is added to the right and select **Low**.



Fifth command:
Joint Move

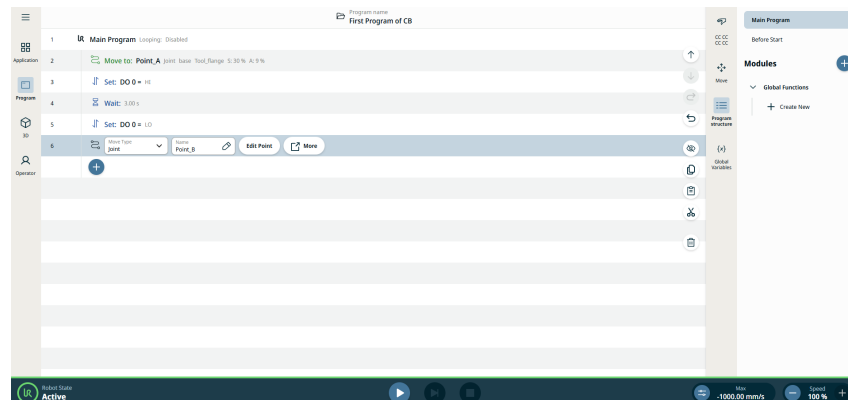
Action: move to
Position B

Name the second
waypoint

19. Tap the add icon in the program tree. Repeat steps 7 to 9 to create a second waypoint. Move the robot arm position in a direction clearly distinguished from Point_A.

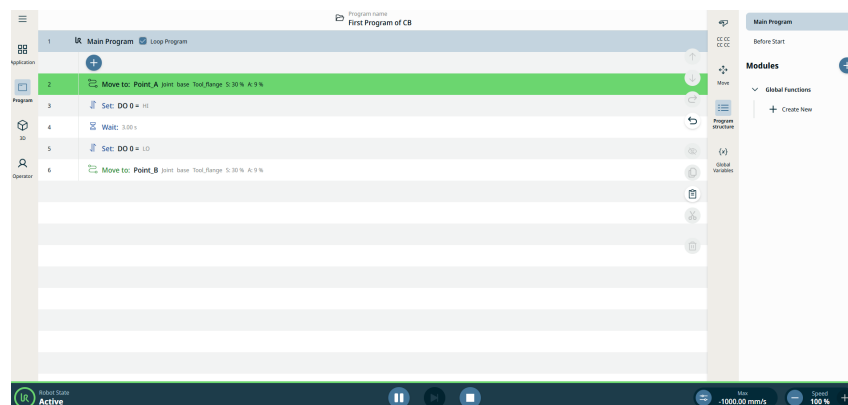
20. In the **Name** field, enter **Point_B** as the name in the **Value** editor tab and **Confirm**.

The main screen shows you the first created program.

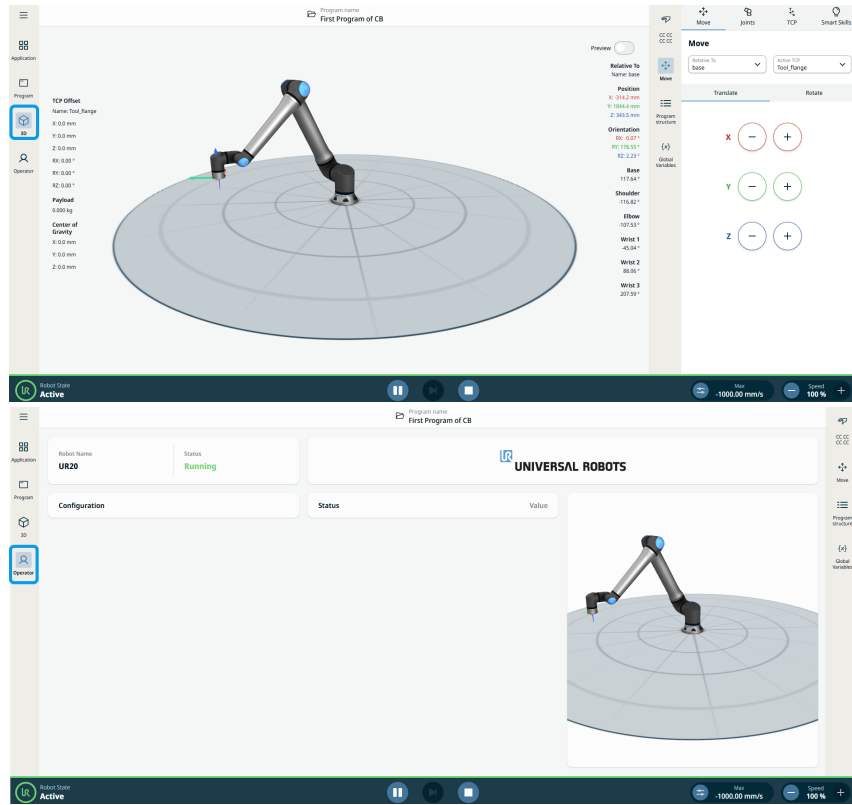


Verify the
program in 3D
viewer

21. Tap the **3D Viewer** icon on the main navigation.
22. On the footer, tap the **Play** button. When you are in the program tree screen, the program in green color signifies that it is the current program execution.

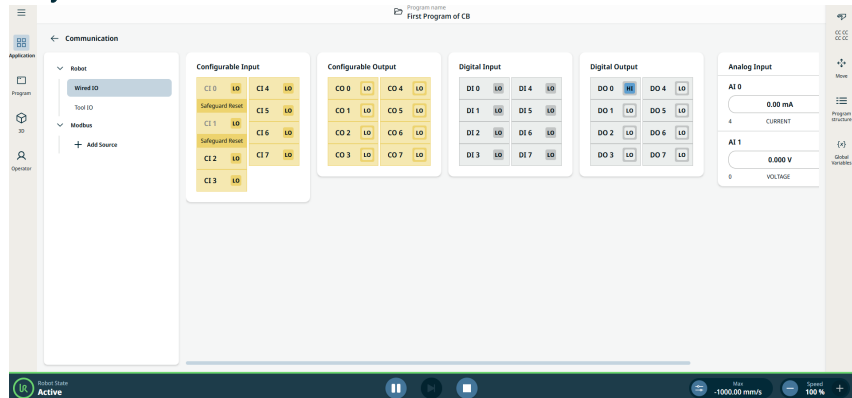


23. When you are in the 3D Viewer and Operator screen, you see the robot move. On the side of the three-dimensional graphics, you can see the numerical details of the robot parts, position, and orientation.



Verify the program in Communication

24. Go to the **Application** tab and tap the **Communication** icon. Observe how the state of **Digital Out 0 (DO 0)** alternates between **Low** and **High** when you tap the **Play** button in the footer.



See [Communication](#) tab for more details.

9. Configuration

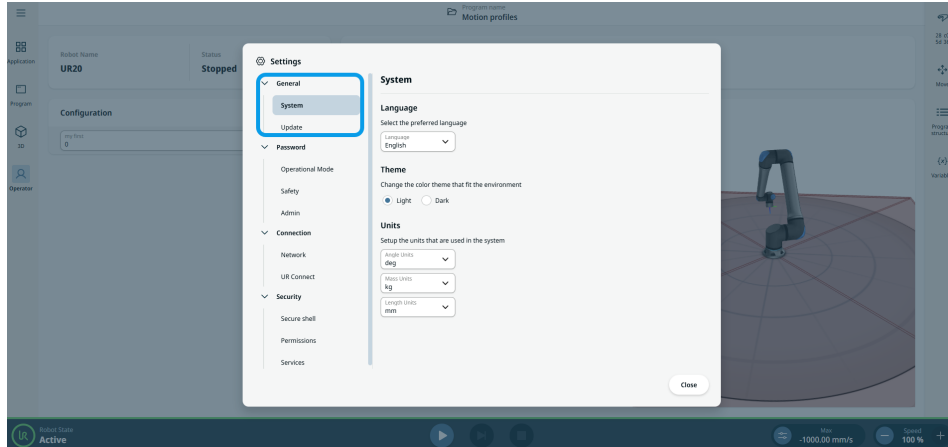
Description	What's in this chapter
	• Configuration Settings • Application Tab • Communication • End Effectors Application • Motion Profiles • Application Variables • Safety Application

9.1. Configuration Settings

Description	The settings in PolyScope X can be accessed via the hamburger menu in the main navigation. You can access the following sections: • General • Password • Connection • Security
General Settings	In the general settings, you can change the preferred language, units of measurements, etc. You also update the software from the general settings.
Password Settings	In the password settings, you can find the default passwords, and how to change them to the preferred and secure passwords.
Connection Settings	In the connection settings, you can set network settings such as IP address, DNS server, etc. Settings related to UR Connect is also found here.
Security Settings	The security settings related to SSH, admin password permissions and enabling/disabling of various services in the software.

9.1.1. General Settings

Description The **General** setting of PolyScope X provides you the function of System and Update.

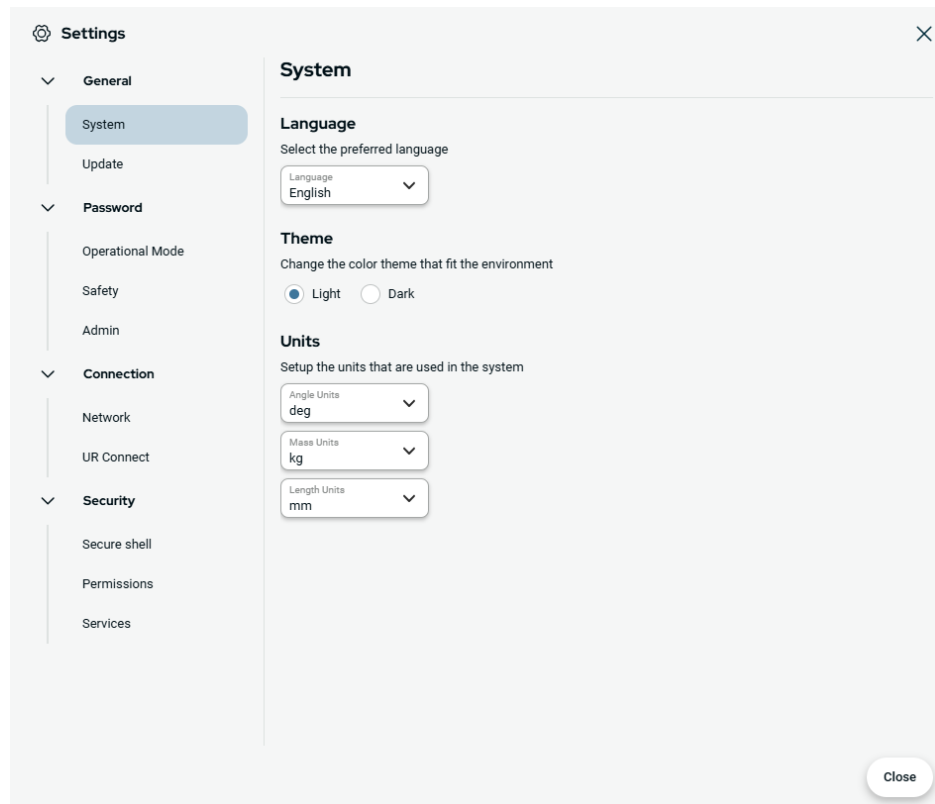


To open General setting

1. Tap the hamburger menu on the main navigation.
2. Tap **Settings** tab.
A left sidebar opens that contains the **General** tab.
3. Under the General tab, tap **System** bar.
You now have access to **Language**, **Theme**, and **Units** menu to suit your preference.
4. Under the General tab, tap **Update** to see your software version and check for updates.

System

Description The system settings contain underlying settings that is set once and very rarely changed.



Languages You can change the preferred language for the UI.
Hit the language selector and find your language.

Theme You can change the preferred language for the UI.
Hit the language selector and find your language.

Units It is possible to change the precision of the software in the settings. Please make sure that you verify this setting, before you start programming the robot program.

Angle Units:
You can change between degrees and radian.

Mass Units:
You can change between kilogram and gram.

Length Units:
You can change between meter and millimeter.

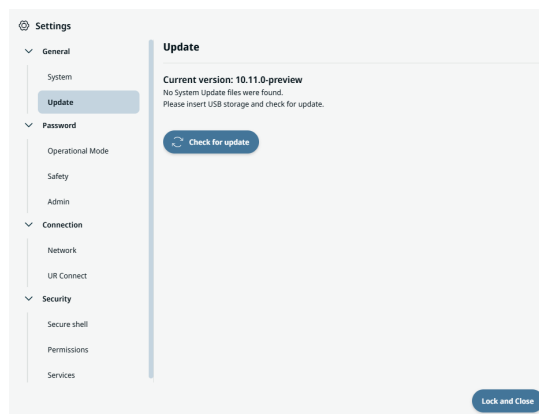


Update

Description When you receive your robot, it is recommended that you update the software to the most recent release from Universal Robots.
Please use this link to find the latest software update: [Download Latest Software](#)

Update SW This is how you update the software:

1. Tap the hamburger menu in the main navigation.
2. Go to Update in the general settings.
3. Insert the USB with the software update.
4. Tap the button "Check for update".

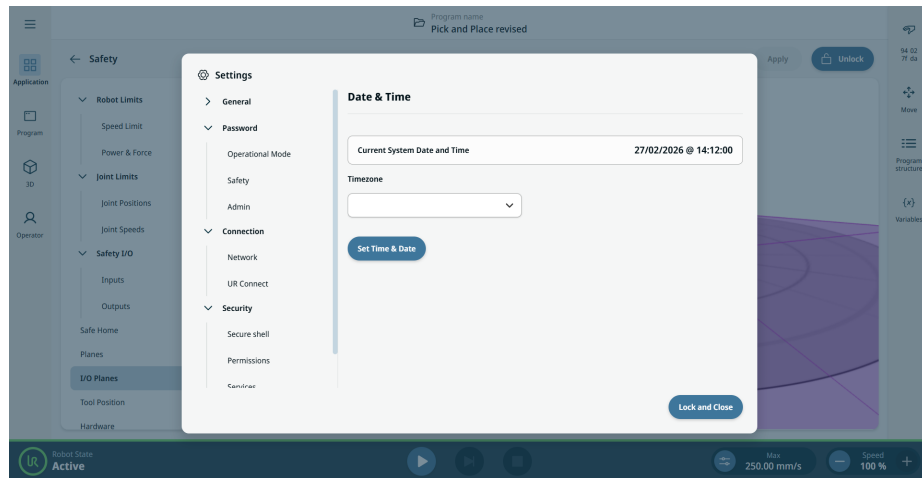


Date and Time

Description You can set the date, time, and time zone on the Teach Pendant.

Set date and time

1. Open the hamburger menu and tap **Settings**.
2. Under **General**, tap **Date & Time**.
3. Unlock and enter your Admin password.



4. To set the time zone, select one from the dropdown. To set the date and time, tap the **Set Time & Date** button and enter the new values.

9.1.2. Password Management

Description In the password settings in PolyScope X, you can find three different types of password.

- Operational Mode
- Safety
- Admin

It is possible to set the same password in all three instances, but it is also possible to set three different password to separate access and options.

Password - Admin



Description All options under Security are protected by an Admin password. The Admin password protected screens are locked by a transparent overlay rendering the settings unavailable. Accessing the Security allows you to configure the settings in the following:

- Secure Shell
- Permissions
- Services

The settings can only be modified by designated administrator/s.
Unlocking any one of the options under Security, also unlocks the other options until you exit the Settings menu.

Default Password

The default password for the admin password is: easybot



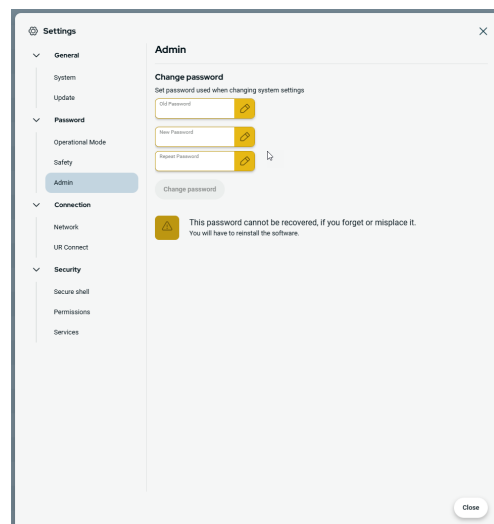
NOTICE

If you forget your admin password, it cannot be replaced or recovered.
You will have to reinstall the software.

To set the Admin password

Before you can use the Admin password to unlock protected screens, you have to change the default password.

1. Access the Hamburger menu and select **Settings**.
2. Under Password, tap **Admin**.
3. Change the current Admin password to a new one.
 - If this is the first time, change the default Admin password from "easybot" to a new password. The new password must be at least 8 characters long.
4. Use the new password to unlock the Settings menu and access the options under Security.



To exit the Settings menu

When one of the Security options is unlocked, the Close button in the bottom right of the Settings menu changes. The Close button is replaced by the Lock and Close button indicating security is unlocked.

1. On the Settings menu locate and tap the **Lock and Close** button.

Password - Operational Mode

Default Password The default password for operational mode: operator



NOTICE

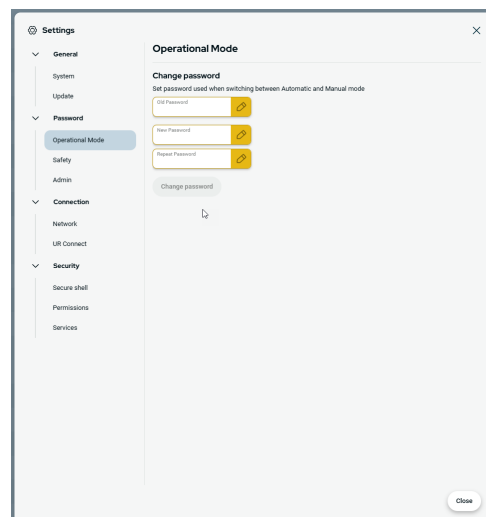
If you forget your password, it cannot be replaced or recovered. You will have to reinstall the software.

You must use the default password, when you change the password for the first time.

Change Operational Mode password

This is how you change the password for operational mode in the PolyScope X setting.

1. Tap the hamburger menu in the main navigation.
2. Tap Settings.
3. Tap Operational Mode in the Password section.
4. Add the default password, if it is the first time.
5. Add your preferred password, at least 8 characters.



Password - Safety

Default Password The default password for safety: **ursafe**



NOTICE

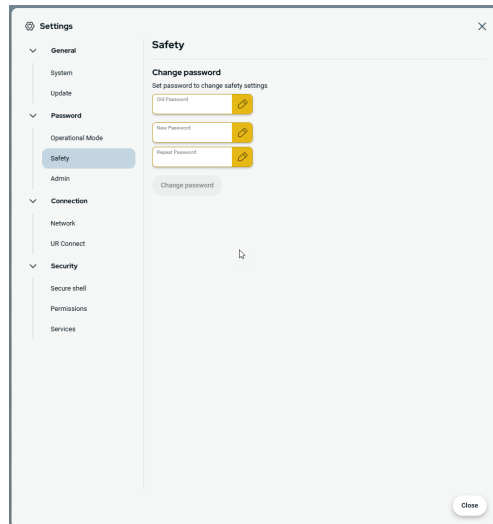
If you forget your password, it cannot be replaced or recovered.
You will have to reinstall the software.

You have to use the default password, when you change the password for the first time.

**Change
Safety
Password**

This is how you change the safety password in the PolyScope X setting.

1. Tap the hamburger menu in the main navigation.
2. Tap Settings.
3. Tap Safety in the Password section.
4. Add the default password, if it is the first time.
5. Add your preferred password, at least 8 characters.

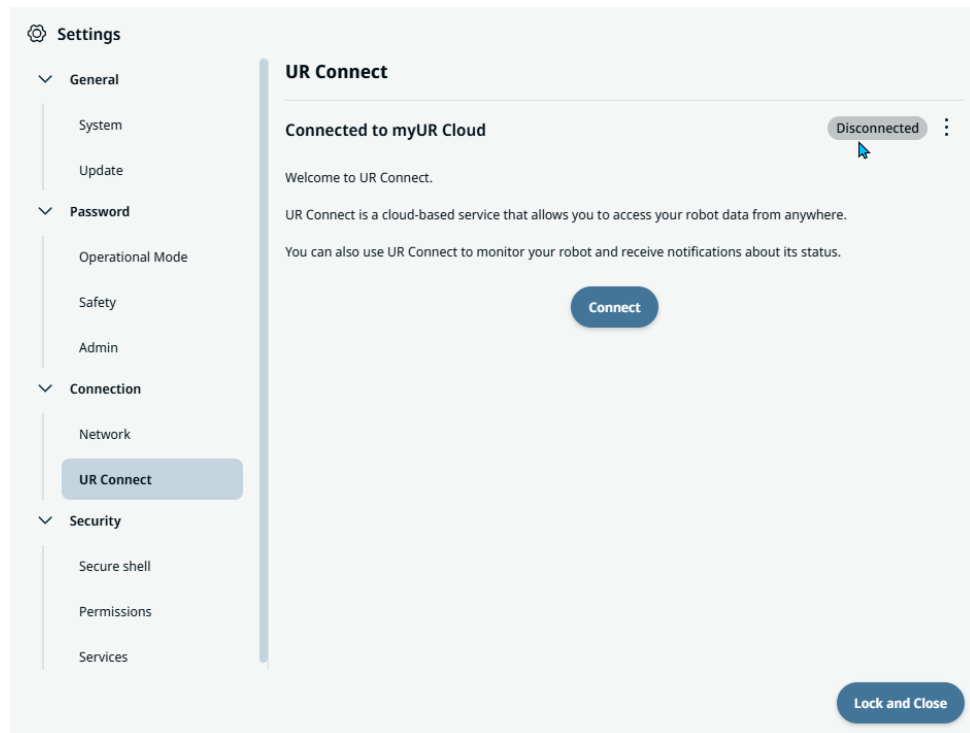


9.1.3. UR Connect

Description The UR Connect menu is used together with the UR Connectivity Kit.

To connect via UR Connect

In UR Connect:

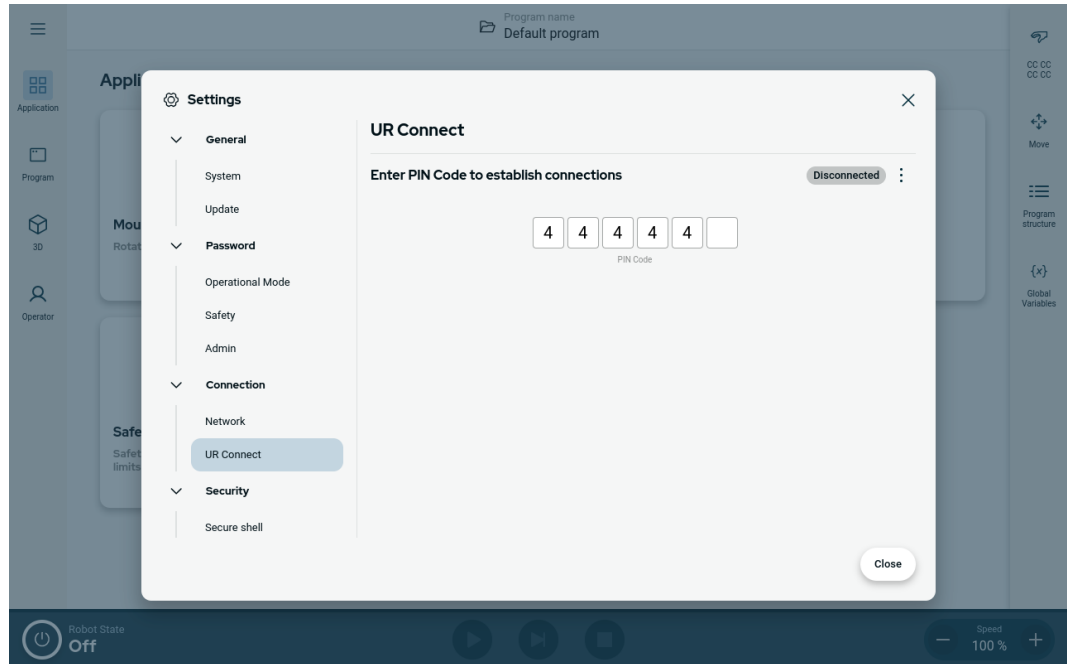


1. Tap Connect.
2. Type your six-digit PIN code.

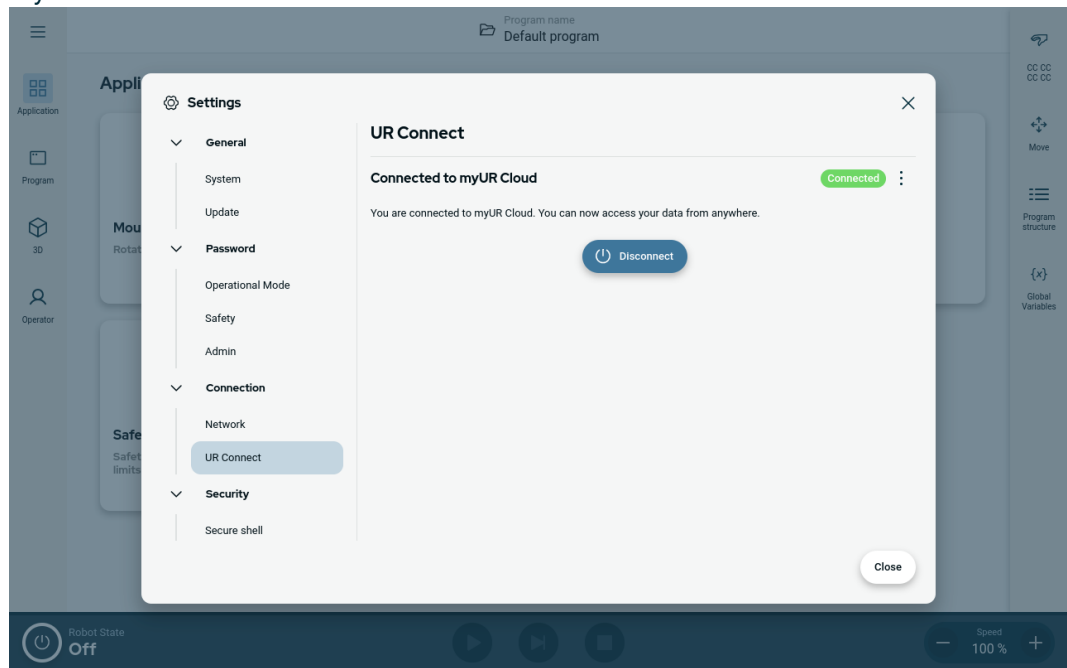
Connect PolyScope X to myUR Cloud

You have to connect your PolyScope X software to the myUR Cloud service. You need to find your PIN code in your myUR account.

1. Go to Settings.
2. Go to UR Connect.
3. Hit the "Connect" button on the main UR Connect page.
4. Add your pin code from myUR.

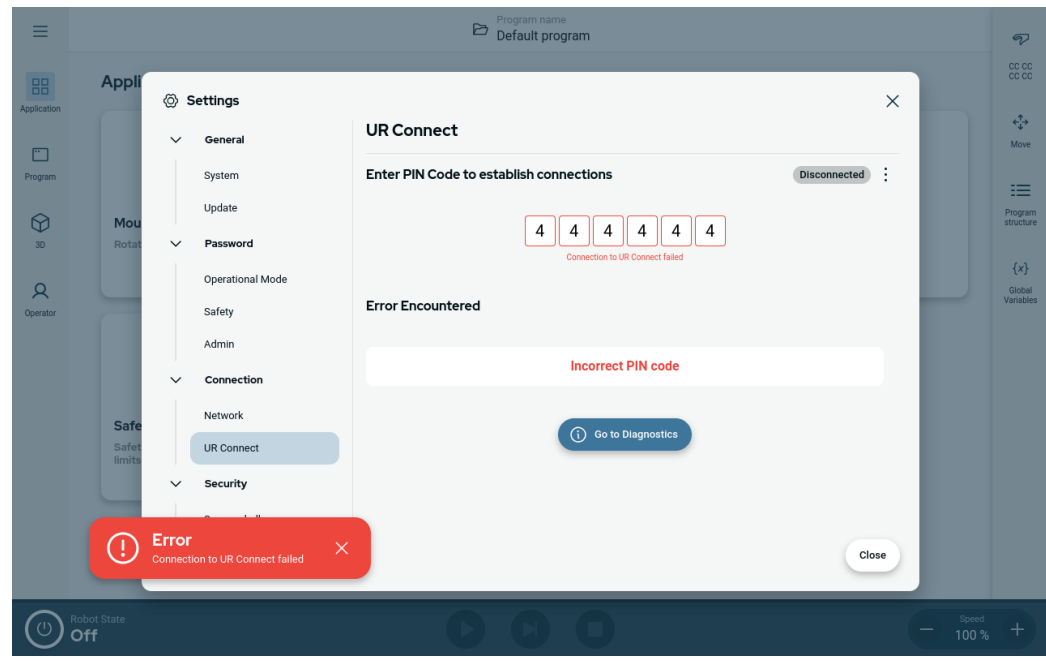


When you see the green icon in the right corner of the window, you are connected to the myUR Cloud.



Unsuccessful connect

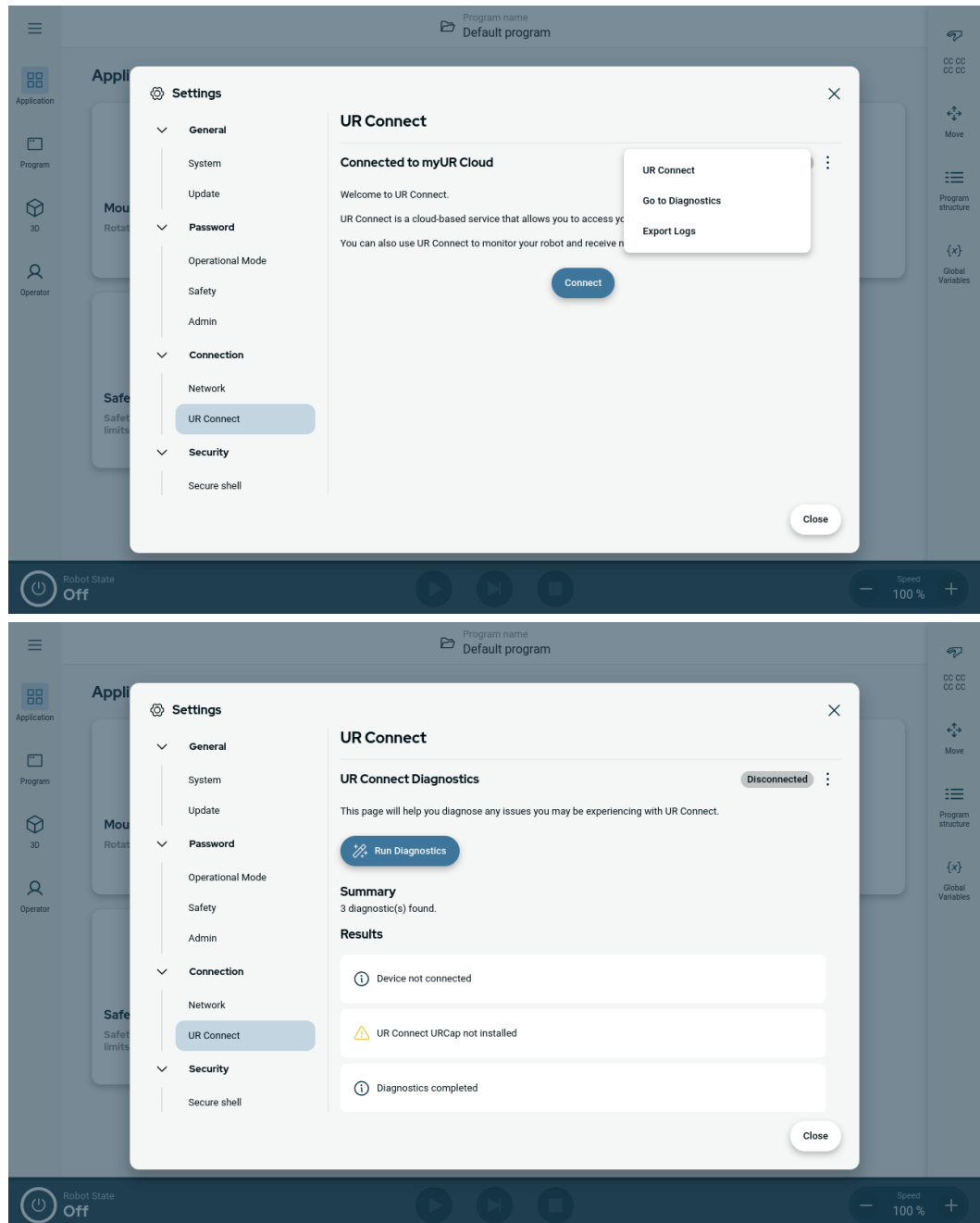
If you see the "Incorrect PIN code", please review your PIN code from myUR.



Diagnostics

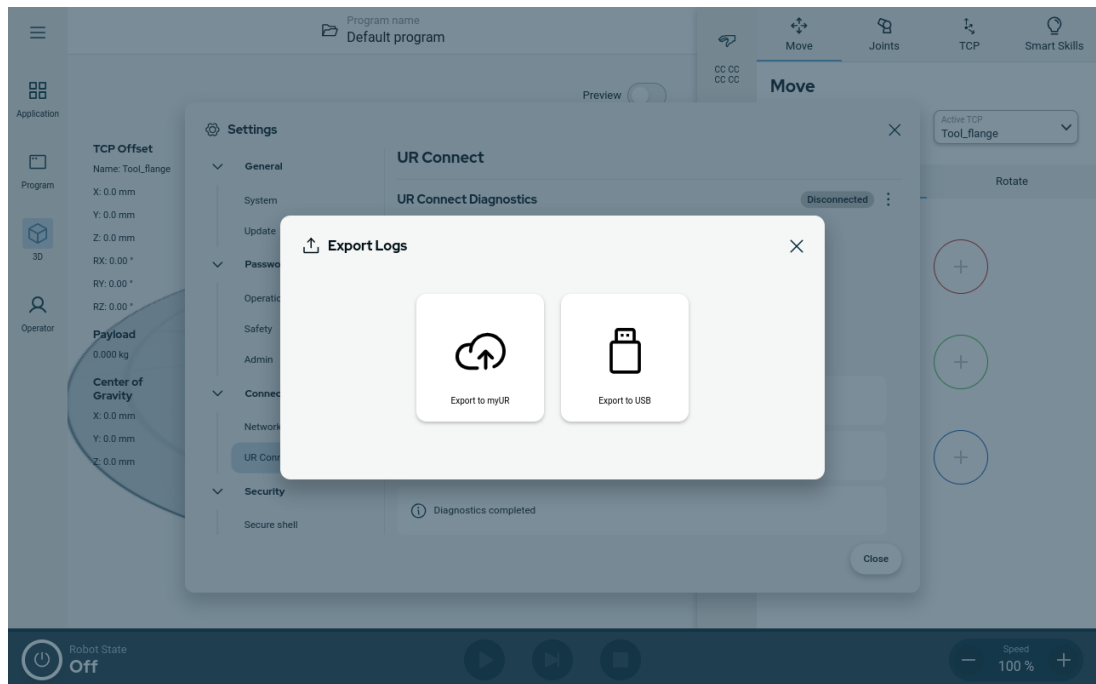
If you experience any unexpected when the UR Connect is active, you can go to the Diagnostics.

1. Go to Settings.
2. Go to UR Connect.
3. Hit the kebab menu in the top right corner.
4. Select the "Diagnostics".



Export logs It is possible to export the UR Connect logs from your PolyScope X software.

1. Go to Settings.
2. Go to UR Connect.
3. Hit the kebab menu in the top right corner.
4. Select the "Export Logs"
5. Select "Export to myUR" or "Export to USB".



9.1.4. Security

Description In the security settings, you can access the following:

- HTTP/HTTPS port access (ports 80 and 443)
- Secure Shell (SSH) Access
- Permissions
- Services

HTTP/HTTPS Port Access



Description Starting in PolyScope X software version 10.14, administrators can control inbound HTTP and HTTPS traffic on ports **80** and **443** directly from the PolyScope X user interface. Use the **HTTP** toggle on the **Services** screen to open or close these ports in the robot firewall. This gives integrators explicit control over web access to the robot without command-line access.

From version 10.14, ports 80 and 443 are **closed by default**. If you close port 80, it **remains closed after the robot restarts**. In earlier versions, port 80 was opened automatically on every boot.



WARNING

After upgrading to PolyScope X 10.14 or later, verify the **HTTP** toggle if external systems, web clients, or externally available URCaps depend on port 80 or 443.

Service enable vs. HTTP toggle

Enabling or disabling a service in the Services list is separate from the **HTTP** toggle:

- The **HTTP** toggle controls whether inbound traffic on ports 80 and 443 is allowed through the firewall.
- Each service toggle controls whether that service is enabled on the robot.

A service may be enabled while its ports are blocked by the firewall. An open port does not enable a disabled service. For secure operation, configure both settings to match your application needs.

URCap connectivity warning

When port 80 is closed while an externally available URCap is active, the Services screen displays a warning that port 80 must be open for the URCap to function. Enable the **HTTP** toggle to restore connectivity.

To control HTTP/HTTPS port access

1. In the main navigation, tap the **hamburger icon**.
2. Tap **Settings**.
3. Under **Security**, select **Services**.
4. Tap the **Unlock icon**, enter the admin password, and **Confirm**.
5. Use the **HTTP** toggle to enable or disable inbound access on ports 80 and 443.
6. Tap **Lock and Close**.

Secure Shell (SSH) Access

Description	You can manage remote access to the robot using Secure shell (SSH). The Secure shell security settings screen allows administrators to enable or disable SSH access to the robot.
To enable/disable SSH	1. Access the Hamburger menu and select Settings. 2. Under Security, tap Secure shell. 3. Enter the password to unlock the settings. 4. Slide the Enable SSH Access to the on position. To the far right of the Enable SSH Access toggle button, the screen shows the port used for SSH communication.
SSH Authentication	Authentication can occur with a password and/or with a pre-shared, authorized key. Security keys can be added by tapping the Add Key button and selecting a security key file. Available keys are listed together. Use the trash icon to remove a selected key from the list.

Permissions

Description	Access to the Networking, URCap Management and Updating PolyScope X screens is restricted by default, to prevent unauthorized changes to the system. You can change the permission settings to allow access to these screens. An Admin password is required to access Permissions.
To access Permissions	1. Access the Hamburger menu and select Settings. 2. Navigate to Security and tap Permissions.
Additional system permissions	You can also lock a few important screens/functionalities with the Admin password. On the Permissions screen in Security section in the Settings menu, it is possible to specify which additional screens are to be protected by the Admin password and which screens are available to all users. The following screens/functionalities can optionally be locked: • Network settings • Update settings • URCaps section in the System Manager • UR Connect



To enable/disable system permissions

1. Access Permission as previously described. The protected screens are listed under Permissions.
2. For the desired screen, slide the On/Off toggle switch to the On position to enable it.
3. To disable the desired screen, slide the On/Off toggle switch to the Off position.

The screen locks again once the toggle is in the Off position.

Services

Description

Services allow administrators to enable or disable remote access to the standard UR services running on the robot, such as Primary/Secondary Client interfaces, PROFINET, Ethernet/IP, ROS2, etc.

Use the Service screen to restrict remote access to the robot by only allowing external access to the services on the robot which the specific robot application is actually using. All services are disabled by default to provide maximum security. The communication ports for each service are to right of the On/Off toggle button in the list of services.

Enabling ROS2

When the ROS2 service is enabled on this screen, you can specify the ROS Domain ID (values 0-9). After changing the Domain ID, the system restarts to apply the change.

Services (Interfaces)

Description

Services lists the standard services running on the robot. You can enable or disable each service.



NOTICE

All services are disabled as a security feature and are password protected. When you start or configure your robot, you have to unlock and enable relevant services.

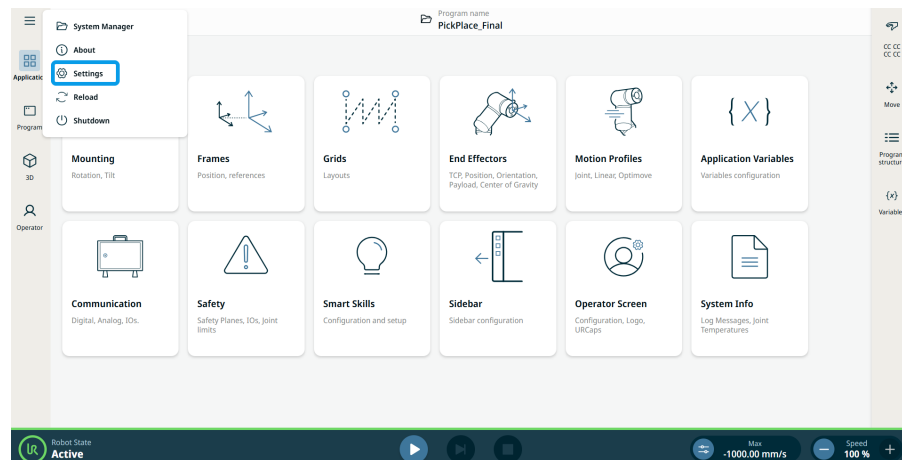
An enabled service remains available even if the ports associated with that service are blocked. So, simply blocking a port is not enough to prevent access to the listed services if they are enabled.

Enabling Services

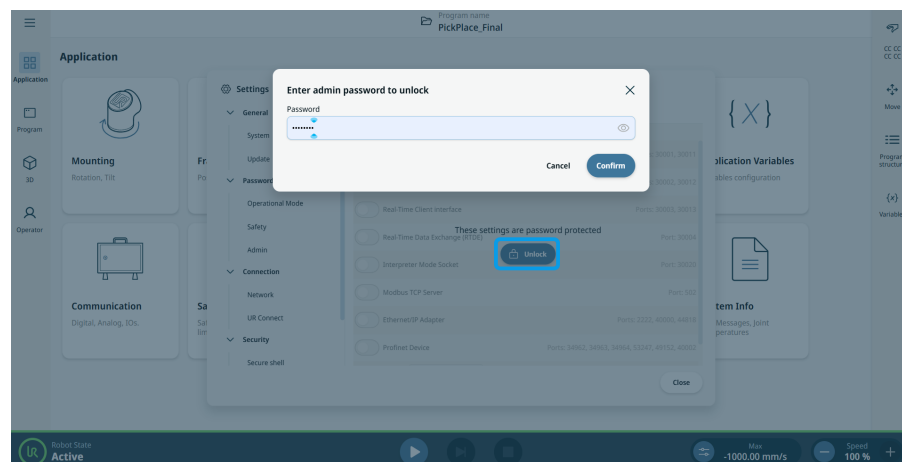
You have to enable the relevant service for the function that you are using.

To enable Services

1. In the main navigation, tap the **hamburger icon**.
2. Tap **Settings**.



3. Under **Security**, select **Services**.
4. Tap the **Unlock icon**, enter the admin password, and **Confirm**.



5. In the list, select an option and tap the slider button to enable (button turns blue) and revert to disable. Then **Lock and Close**.

EtherNet/IP Adapter

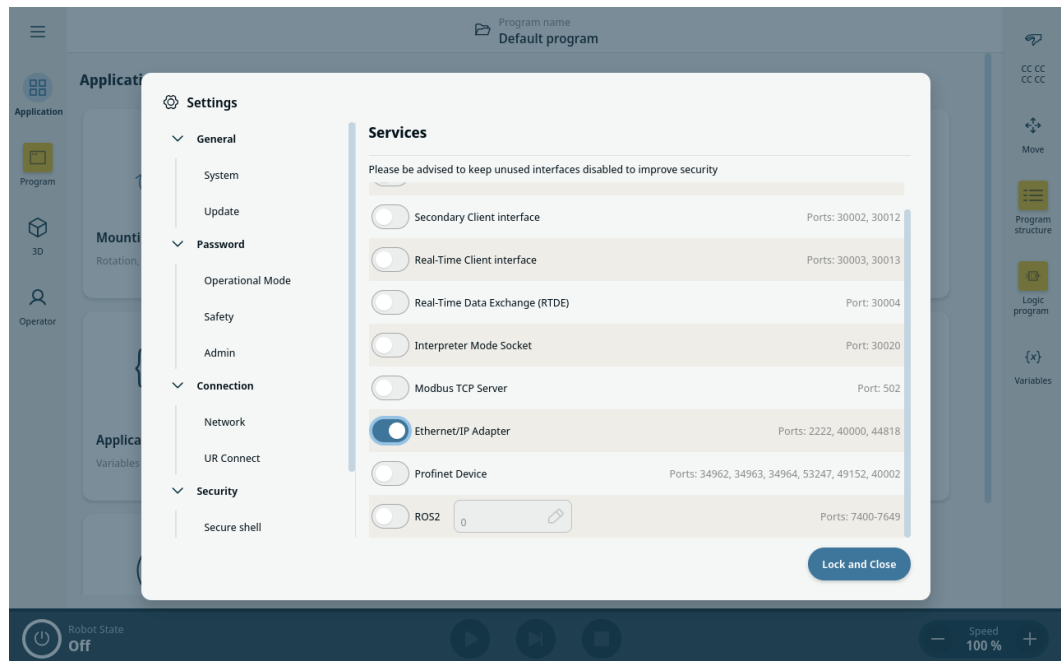
Description

EtherNet/IP is a network protocol that enables the connection of the robot to an industrial EtherNet/IP scanner device. If the connection is enabled, you can select the action that occurs when a program loses EtherNet/IP scanner device connection.

Enable Ethernet/IP

This is how to enable Ethernet/IP function in PolyScope X.

1. In the top left of the screen, tap the Hamburger menu.
2. Tap Settings.
3. In the menu on the left, under Security, tap Services.
4. Type the admin password.
5. Tap the Ethernet/IP Adapter button to switch toggle on.



Using Ethernet/IP

This is how you find the Ethernet/IP functions in PolyScope X:

In the PolyScope X left header.

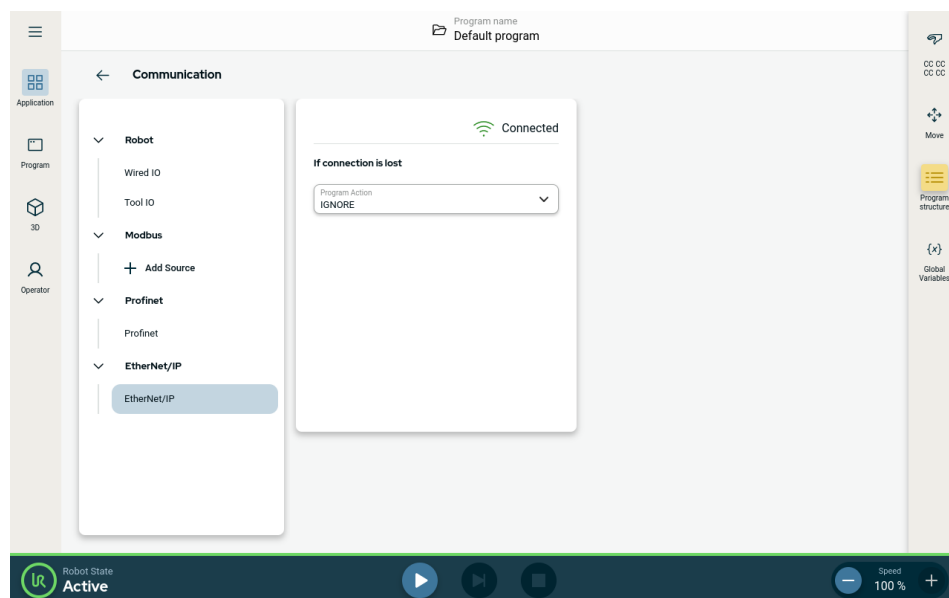
1. Tap the Application icon.
2. Under Communication, select Ethernet/IP from the left menu.

Select the relevant action from the list:

Ignore PolyScope X ignores the loss of EtherNet/IP connection, and the main program continues to run.

Pause PolyScope X pauses the main program. The program resumes from where it stopped.

Stop PolyScope X stops the main program.



In the upper right corner of this screen, you can see the Ethernet/IP status.

Connected The robot is connected to the Ethernet/IP Scanner.

No Scanner Ethernet/IP Device is running, but no Scanner is connected to the robot via Ethernet/IP.

Disabled Ethernet/IP Device is not enabled.



Ethernet diagnostics support

There is an added support to enable Ethernet network diagnostics. When it is enabled, the traffic between robot Ethernet/IP Adapter and Ethernet/IP Scanner will be collected in a pcap file. Collected data is available as a part of Support File.



Robot action upon loss of Ethernet/IP input connection for Robot IO or GP registers.

Program Action
IGNORE

Enable network diagnostics (available in Support File)



PROFINET Device

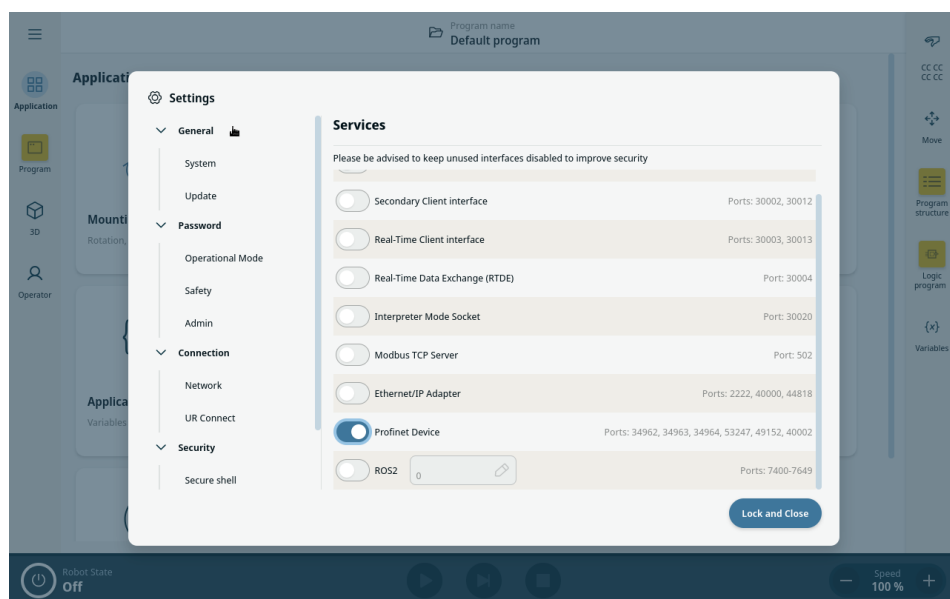
Description

The PROFINET network protocol enables or disables the connection of the robot to an industrial PROFINET IO-Controller. If the connection is enabled, you can select the action that occurs when a program loses PROFINET IO-Controller connection.

Enable PROFINET

This is how you enable PROFINET function in PolyScope X.

1. In the top left of the screen, tap the Hamburger menu and then tap Settings.
2. In the menu on the left, under Security, tap Services.
3. Type the admin password.
4. Tap the PROFINET button to switch PROFINET on.



Using PROFINET

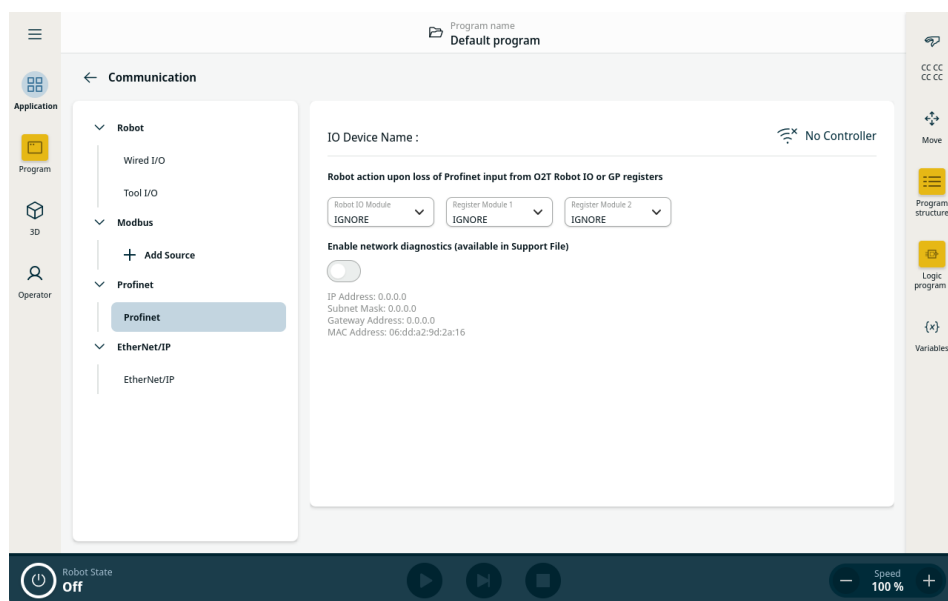
Find the PROFINET functions in PolyScope X:

In the PolyScope X Main Navigation.

1. Tap the Application icon.
2. Under Communication, select PROFINET from the left menu.

Select the relevant action from the list:

Ignore	PolyScope X ignores the loss of PROFINET connection, and the main program continues to run.
Pause	PolyScope X pauses the main program. The program resumes from where it stopped.
Stop	PolyScope X stops the main program.



Diagnostics

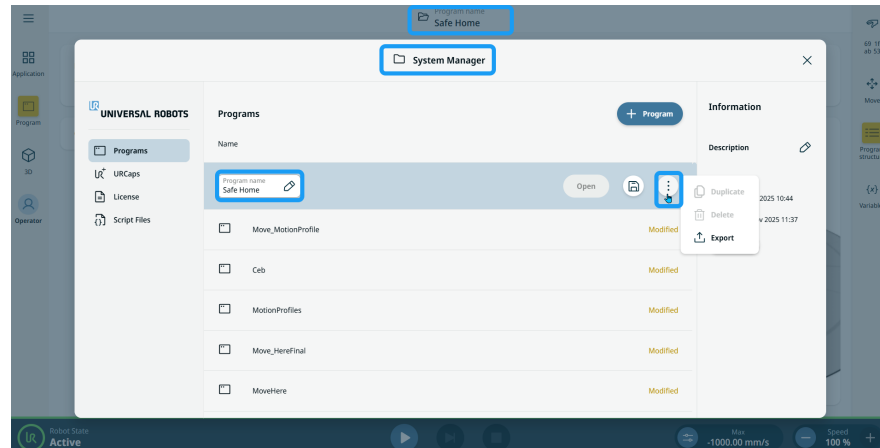
Polyscope X has an option to record network traffic between Robot and PROFINET I/O-Controller. This can be used for diagnostics in case of connectivity issues.

- To enable this option, tap the button "Enable network diagnostics" to toggle on.

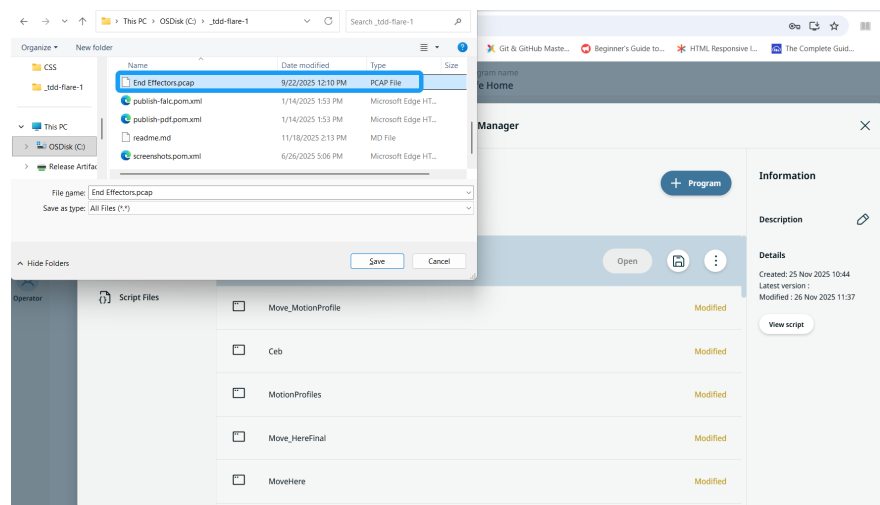
Communication data will be saved in a .pcap file.

The file is saved in Support File in System Manager. Up to 50 MB of data can be recorded in the diagnostics.

- Go to the **Header**, and the **System Manager** appears.
- Choose the program you are working on.
- Tap the three-vertical-dots icon (kebab icon) and choose **Export**.



- Choose the communication data in pcap file and Save.



- A pop-up notification appears in the main screen, stating you have successfully exported the file in the program.

Ethernet port status

When PROFINET Device is enabled, a new virtual ethernet port will be created. Virtual ethernet port configuration shows information about currently configured IP address, subnet mask, gateway and MAC address. Note that this virtual port is different from port configured in Robot network settings.

New PROFINET version

New GSDML file: GSDML-V2.45-UR-PROFIsafe-20251201.xml



NOTICE

Backwards compatibility:

- PLC programs made for previous software versions will run with no changes.
- Internal configuration file for PROFINET is copied, and the copy is migrated to the new PROFINET version.
- New PLC programs should use the recently released and proper GSDML file.



WARNING

It is not possible to downgrade to previous version. All changes made after the upgrade will be lost. Back up all data before upgrading.

Certificate Handling

Description Digital Certificate Handling lets you view, edit, and import or export privacy enhanced mail (PEM) files, generate certificate signing requests (CSRs), and create self-signed certificates.

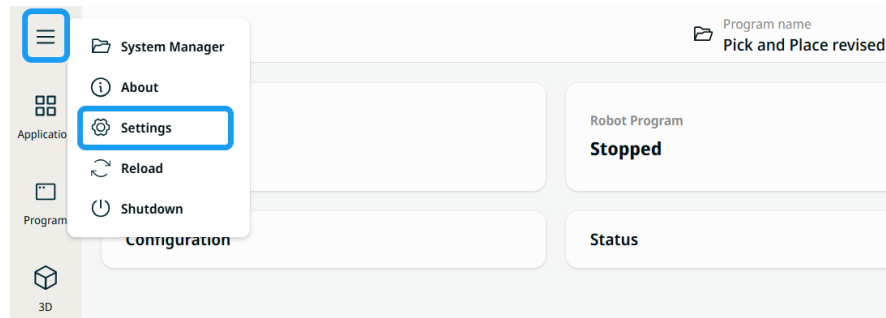
Purpose and benefits This feature ensures secure robot identification and enables encrypted communication. It supports workflows for both self-signed certificates (for quick setup) and CA-signed certificates (for stricter trust requirements). These are the key benefits:

- View and verify certificate details.
- Import and export certificates and CSRs in PEM format.
- Generate self-signed certificates with configurable options, subject fields, and validity period.
- Resolve expired certificate warnings.

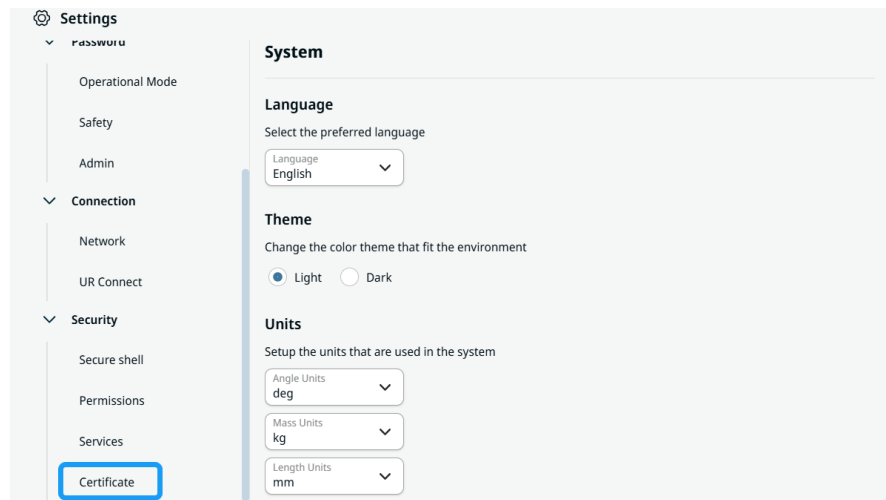


To locate Certificate Handling

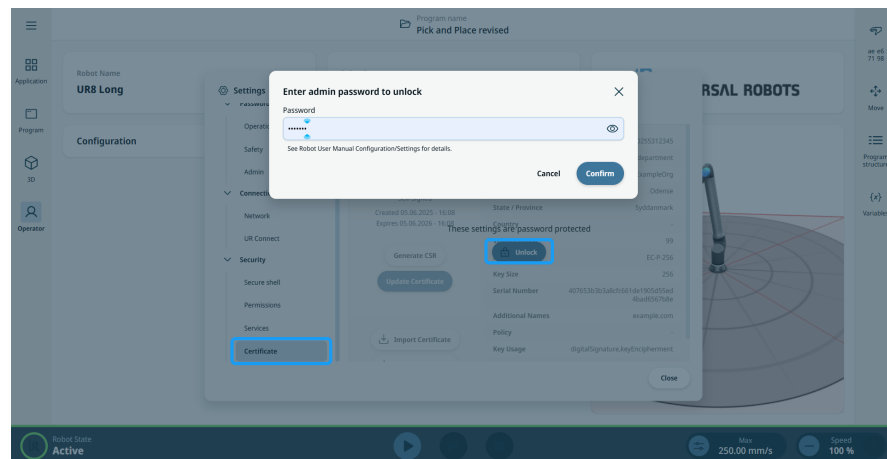
1. Tap the **hamburger menu** in the **Main Navigation** and select **Settings**.



2. Scroll through the **Settings** options to find **Certificate** in **Security**.

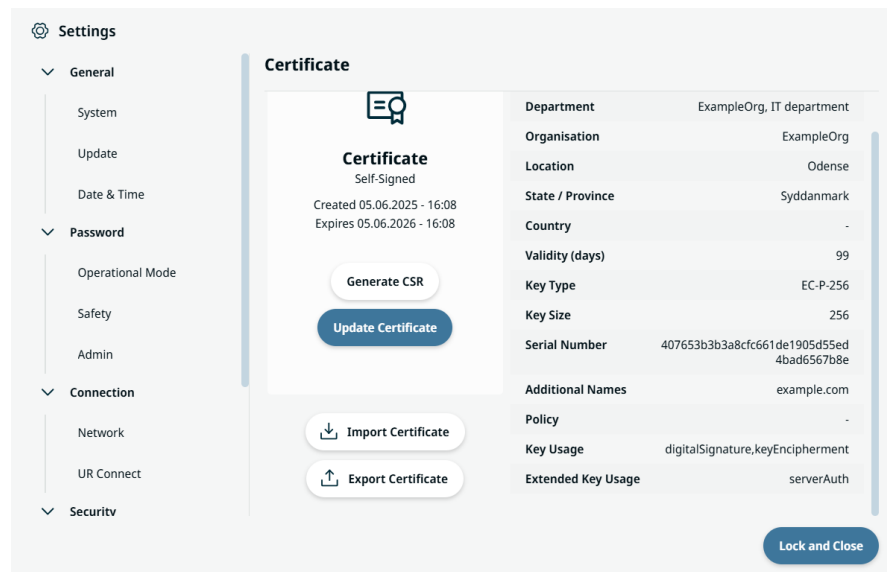


3. Tap **Certificate** and **Unlock** to view the certificate details.



4. The Certificate screen shows the details and the four buttons:

- **Generate CSR**
- **Update Certificate**
- **Import Certificate**
- **Export Certificate**





To update a certificate

1. Tap the **Update Certificate** button.

2. Configure the certificate details in the eight fields.
3. Tap **Generate Self-Signed** to create and install the new certificate.
The current key and certificate are replaced. A success message is displayed, and the details are refreshed.

To generate a CSR



NOTICE

Make sure a certificate is already present. The **Generate CSR** button is disabled if no certificate exists.

1. Tap the **Generate CSR** button.
2. An export dialog opens where you can select external media. The CSR is saved as **csr.pem**.
3. Use the CSR to obtain a CA-signed certificate from your Certificate Authority, which is the company's IT department.
4. Import the CA-signed certificate.

To Import a certificate

1. Tap the **Import Certificate** button.
2. Select a PEM certificate file. Use the file browser to locate the file.
3. After a successful import, the system displays a success message.
The certificate details are updated. If the format is invalid, an error message appears.

- To Export a certificate**
1. Tap the **Export Certificate** button.
 2. An export dialog opens. Tap **Check for Media** to detect external media (SD card or USB).
 3. Select the destination and tap **Export**.
- The certificate is saved as **certificate.pem** on the selected media.

- Handling expired certificates**
- When a certificate expires, the page displays an **Expired** message with the expiration date instead of the usual creation/expiration details.
- You can resolve the issue in two ways:
1. Use **Update Certificate** to generate a new self-signed certificate.
 2. Use **Import Certificate** to install a valid PEM certificate from your CA or backup.

Audit log

- Description**
- In compliance with cyber security requirements, Polyscope X will log all security relevant configuration changes.
- The log can be used to investigate intrusion incidents and anomalies. It is automatically collected and requires admin access to view.
- Logs are kept for 90 days.

- View report**
1. Enable SSH on the robot
 2. Access the robot via ssh
 3. Generate report with the command `aureport`

- View specific entries**
- This will filter and print log messages in human readable format. This feature can be used to inspect the logs.

1. Enable SSH on the robot
2. Access the robot via ssh
3. See specific entries using `ausearch -k <key> --format text`
4. Possible keys are:
 - access_control
 - request_error
 - control_system
 - config_change
 - audit_config
 - audit_rules



Follow the log	This will live print log messages as they happen. This feature can be used to inspect what actions are logged. 1. Enable SSH on the robot 2. Access the robot via ssh 3. <code>tail -f /var/log/audit/audit.log</code>
Additional options	The tools aureport and ausearch are standard tools to access audit logs. More detailed information about their features can be found online.

9.1.5. Unlock Safety Settings

Description	You have to unlock the safety settings to change them.
--------------------	---



NOTICE

Safety Settings are password protected.

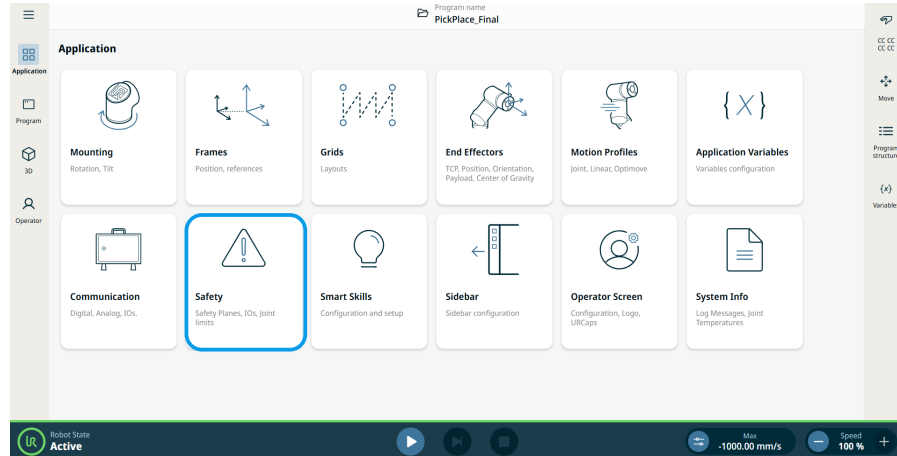
The default password for safety: **ursafe**.

Unlock safety settings	1. In the PolyScope X Main Navigation, tap the Application tab. 2. On the workcell screen tap the Safety icon. 3. Observe that the Robot Limits screen displays, but settings are inaccessible. 4. Enter the safety password and tap UNLOCK to make settings accessible. 5. Tap LOCK or navigate away from the Safety menu to lock all Safety item settings again.
-------------------------------	--

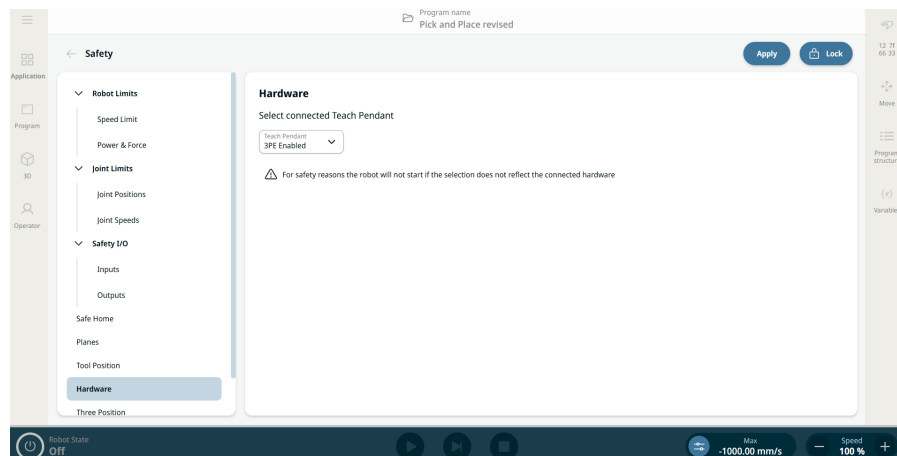
9.2. Configure a 3PE TP in PolyScopeX

To configure the 3PE TP software

1. On PolyScope, in the left menu, tap **Application** and select **Safety**.



2. Tap **Hardware** and the **Unlock** button.



3. Enter the password and tap **Confirm**. Teach Pendant is now enable.
4. Tap **Apply** to restart the system. PolyScope continues to run.
5. Tap **Apply and restart** then **Confirm Configuration** to complete the 3PE Teach Pendant software installation.

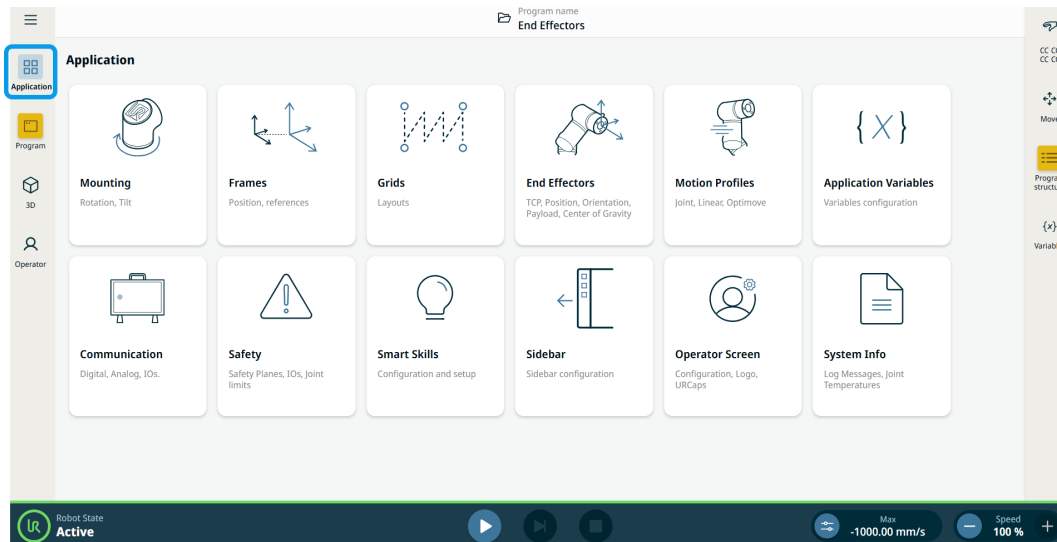
9.3. Application Tab

Description

The **Application** tab allows you to configure the settings which affect the overall performance of the robot and PolyScope X. From PolyScope X 10.14, one Application can be shared by multiple programs. See [Shared Applications in System Manager](#).

To Tap Application.

access
Application



The **Application screen** appears, which shows twelve application submenus to use for robot movement, configuration, and information.

- Mounting
- Frames
- Grids Application
- End Effectors Application
- Motion Profiles
- Application Variables
- Communication
- Safety
- Smart Skills Application
- Sidebar in the Application Tab
- Operator Screen Application
- System Info

9.4. Communication

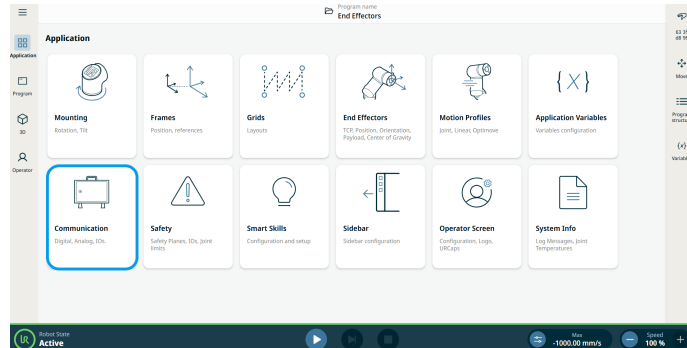
Description The **Communication** tab allows you to monitor and set the state of I/O (input/output) signals on the control box connectors.



Figure 1.1: Communication screen displaying IOs.

Using the Communication tab functionality

1. Go to the Application menu screen.
See [Application Tab](#).
2. Tap the **Communication** icon.



The **Communication** screen appears, which is divided into the left panel and the right panels.



The left panel consists of the communication tree, composed of three functions:

- **Robot Wired IO**
- **Robot Tool IO**
- **Modbus**

The right panels show the editable fields of the chosen function.

9.4.1. Wired IO

Description Wired IO is used to monitor and set the live IO signals from/to the control box.

To access Wired IO

1. Tap **Wired IO** on the left panel.

The Communication screen displays the current state of the IO, including during program execution. If anything is changed during program execution, the program stops. At program stop, all output signals retain their states.

2. Use the scrollbar on the right panel to move sideways, and you can see the editable fields of the six components of the wired IO:
 - Configurable Input
 - Configurable Output
 - Digital Input
 - Digital Output
 - Analog Input
 - Analog Output



Input is used to start, stop, or pause the program, or as an external **Freedrive** button. **Output** shows whether the robot is running or not.

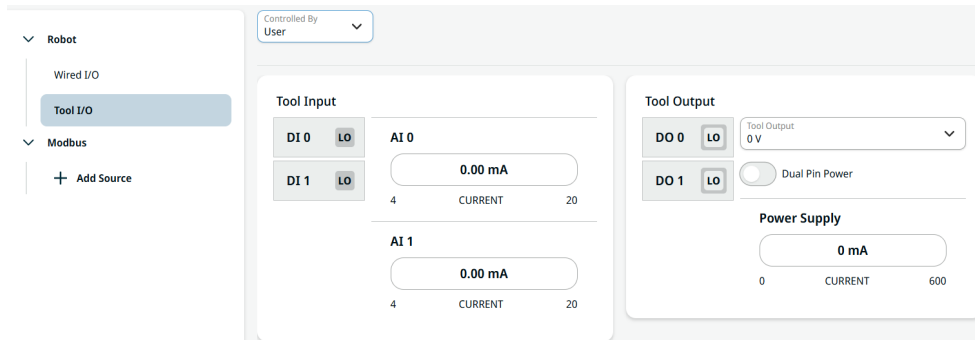
9.4.2. Tool I/O

Description Tool I/O monitors and sets the live IO signals from/to the control box.



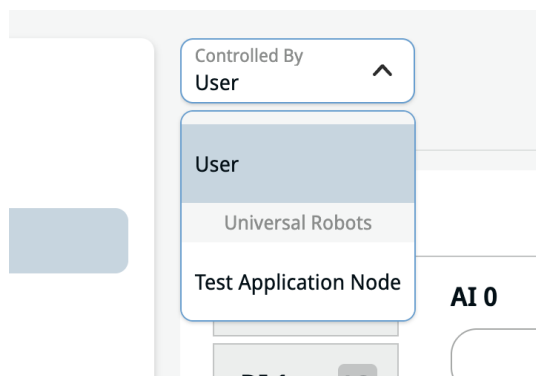
To access Tool IO

Tap **Tool I/O** on the left panel. You can see the editable fields of Tool Input and Tool Output on the right.



URCaps in Control dropdown

The **Controlled By** drop-down menu allows user to select a URCap that can control the tool settings. If a URCap is installed that has this capability, it will be displayed in the list. Follow here on how to install a [URCap](#).



When selected, the URCap will be responsible for the following settings:

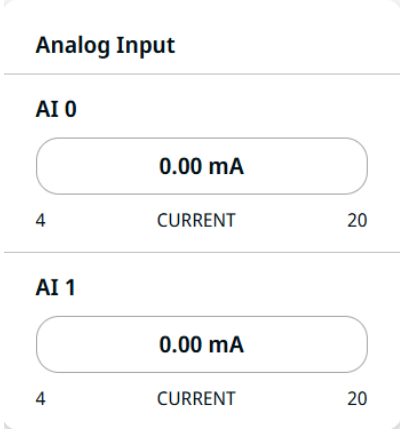
- **Tool Output**
Setting can be 0, 12, or 24 V.
- **Dual Pin Power**
Setting is either true or false.
- **Domain for analog inputs**
Setting current/voltage per analog input, as seen in the pop-up when pressing one.
- **Power Output for digital outputs**
Setting Sinking/Sourcing/Push/Pull, as seen in the pop-up when pressing one.
- **Tool Communication settings**
These settings are not available on the user interface, but are the enabled, baud rate, parity, stop bits, rx idle chars, tx idle chars.

Upon selection of a URCap, any controls related to the above settings will be disabled. The new settings will be applied when an entry is selected in the dropdown, when the program is loaded, and when the program is played.

9.4.3. Analog Input

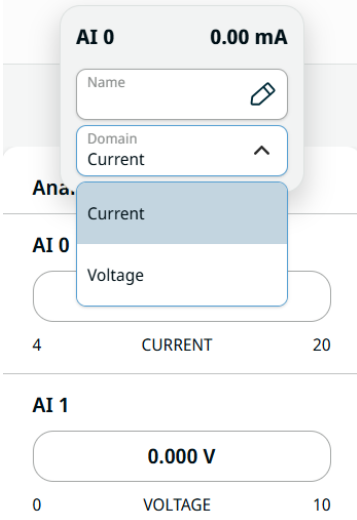
Description Analog input has two fields for AI 0 and AI 1. It can be set to 4-20 mA. These settings are persistent over restarts of the robot controller and saved in the installation.

To access Analog Input



The screenshot shows the 'Analog Input' configuration screen. It has two sections: 'AI 0' and 'AI 1'. Each section has a large rounded rectangle displaying '0.00 mA'. Below each display, the range '4' and '20' is shown, with the word 'CURRENT' centered between them.

1. Tap **AI 0**. The **Name** and **Domain** fields appear.
2. In the **Name** field, create the name you prefer.



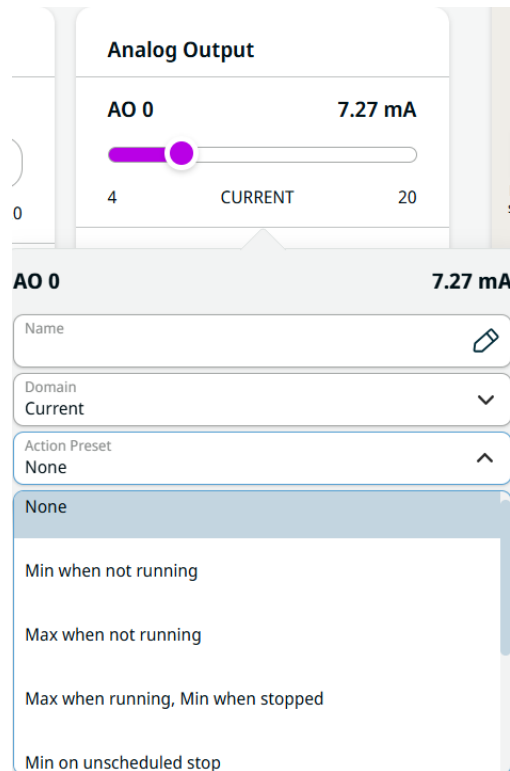
This screenshot shows the 'AI 0' settings menu open over the 'AI 0' section of the configuration screen. The menu has a title bar with 'AI 0' and '0.00 mA'. It contains two fields: 'Name' with a pencil icon and 'Domain' with an upward arrow icon. A dropdown menu is open under 'Domain', showing 'Current' (highlighted) and 'Voltage'. The background shows the 'AI 1' section with a '0.000 V' display and a '0' to '10' 'VOLTAGE' range.

3. Tap the **Domain** field, and you can choose either Current or Voltage.
4. Tap **AI 1**.
5. Repeat steps 2-3.

9.4.4. Analog Output

Description Analog output has two fields for AO 0 and AO 1. As with analog input, it can be set to 4-20 mA.

To access Analog Output



1. Tap the purple slider of each field to adjust the current or voltage settings.
2. Tap **AO 0**.
3. In the **Domain** field, you can choose either Current or Voltage.
4. Tap the **Action Preset** field, and eight choices are available:
 - None
 - Min when not running
 - Max when not running
 - Max when running, Min when stopped
 - Min on unscheduled stop
 - Min on unscheduled stop, otherwise Max
 - Max when drive power is on, otherwise Min
 - Min when drive power is on, otherwise Max

5. Choose the pertinent action preset. The purple slider reflects the action preset you choose.
6. Tap **AO 1** and repeat steps 3-5.

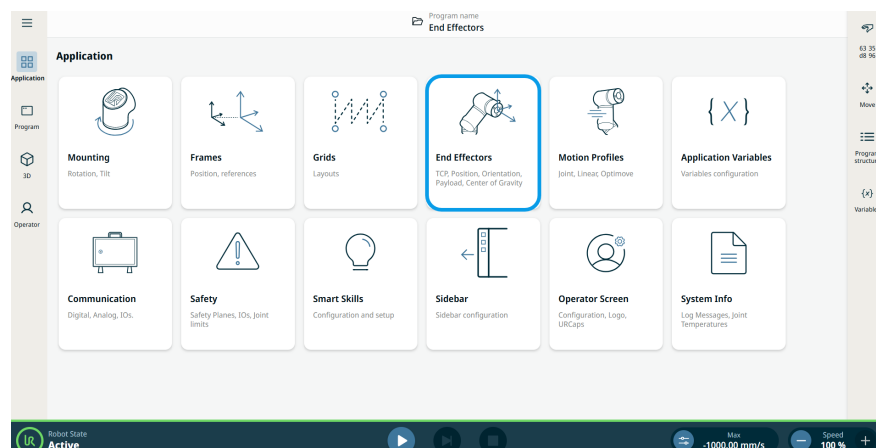
By selecting the last two action preset, the Min/Max value of signals will be changed when power is OFF/ON.

9.5. End Effectors Application

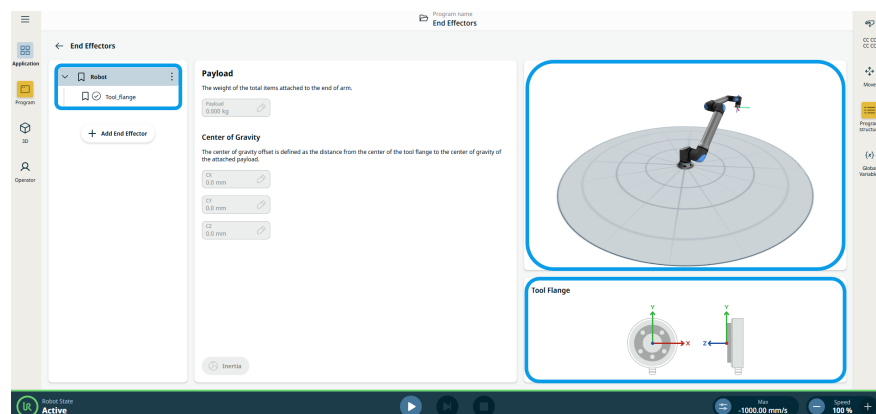
Description The End Effectors application is used for configuring the tooling at the end of the arm, such as the tool center point (TCP) and center of gravity (CoG).

Using the End Effectors application functionality

1. Go to the Application nodes screen. See [Application Tab](#).
2. Tap the **End Effectors** icon.



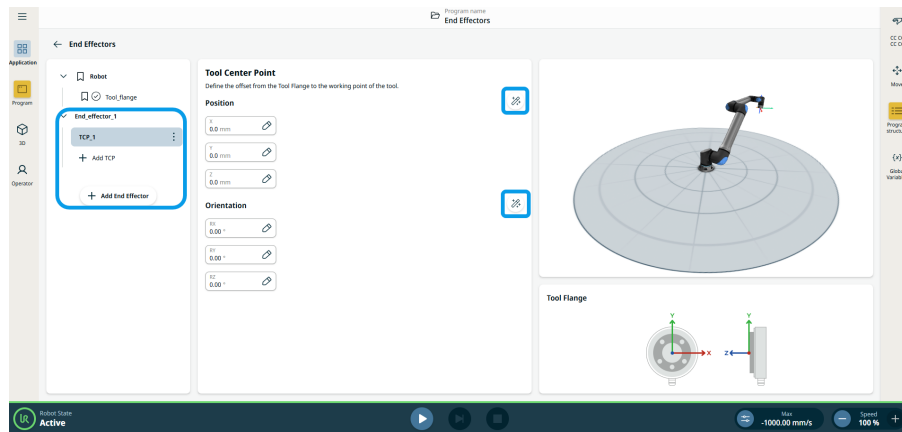
In the main screen, you can see End Effector application screen with a 3D robot view and an illustration of the Tool flange on the right panel. This change has been introduced to help users visualize the frame at the TCP and Tool Flange. On the left panel, the end effectors list now has the **Robot** group. The built-in **Tool_flange** end effector is located under that group.



Tool Center Point (TCP) wizards

The TCP wizards enables you to define the position and orientation of a TCP. The new wizards are accessible from the Tool Center Point application screen by tapping the new wizard icon next to the **Position** header and the **Orientation** header, respectively. In both wizards, you can call up a popup any time with more detailed instructions and animations on how to use the wizard. The instructions can be accessed by pressing the **Instructions** icon in the bottom left corner of the wizards.

1. On the left panel, tap **+ Add End Effector**.
2. Tap **TCP_1** under the newly added **End_effector_1** tree.
3. In the central panel, the Tool Center Point, you see the two wizard wand icons for Position and Orientation.



Adding, Renaming, Duplicating, and Deleting TCP

You can start configuring a new TCP with the following actions:

- Tap **+** to define a new TCP with a unique name. The new payload appears in the End effector tree..
- Tap to rename a TCP.
- Tap to duplicate a TCP.
- Tap to delete a selected TCP. You cannot remove the last TCP.

Active TCP The check-mark in the drop-down indicates which TCP is active

Default TCP The default TCP is set as the active TCP before the program starts. Select the desired TCP and tap to **Set as default** to set a TCP as the default.

9.5.1. TCP Position

Description The TCP position wizard assists user in finding the position (translation) of a TCP. This is useful in situations where the position is unknown and cannot be easily measured.

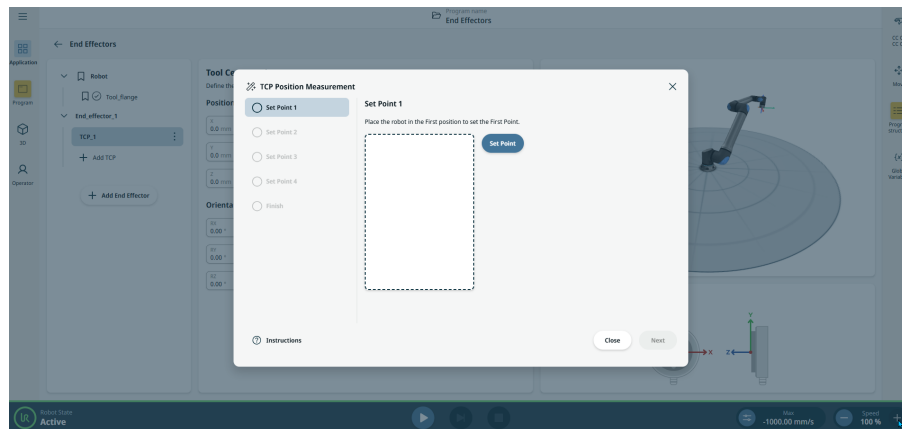
TCP Position wizard

1. On the central panel, tap the TCP Position wizard icon,

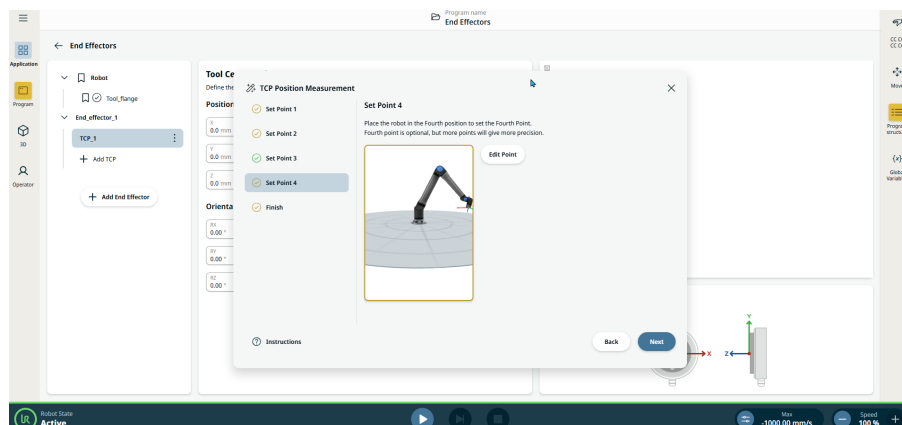
Position



and the pop-up TCP Position Measurement screen appears.



2. Tap the **Set Point** button to set three or four positions by moving the TCP to the same position from different angles. This can be done by freedrive or jogging the robot in the 3D interface. You need at least three points from different angles for successful measurement.

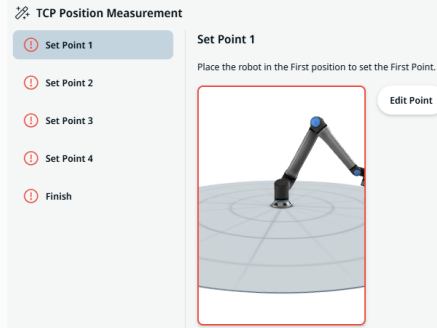


The wizard calculates the TCP position automatically. Though three positions are sufficient to determine the TCP position, a fourth position can be used to further verify that calculations are correct.

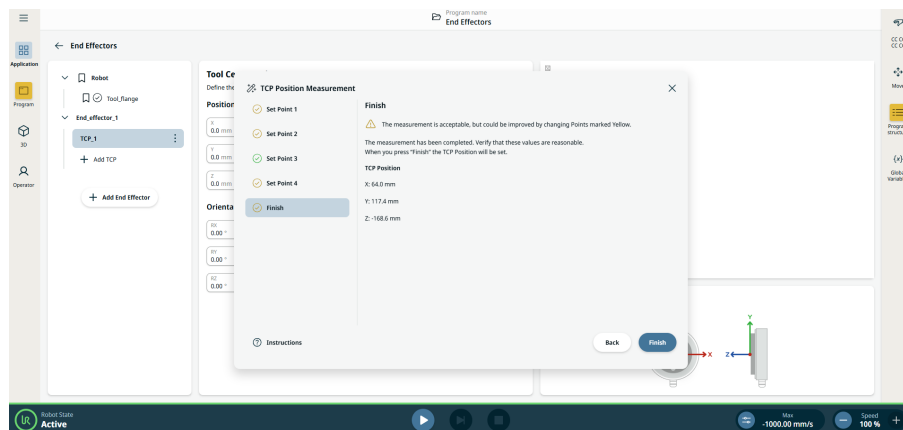


NOTICE

A red warning in Set Points informs user if the taught positions are not sufficiently diverse, in which case it is not possible to compute the TCP position. This means that the positions must be retaught.



- When the measurement is completed, tap **Finish**. You can see the robot arm's position as shown in the X, Y, Z coordinates.



9.5.2. TCP Orientation

Description

The new TCP orientation wizard provides a convenient way of determining the orientation (rotation) of a TCP. By having the user move the robot to a position where the tool's Z-axis aligns with a selected frame, the wizard can compute the TCP orientation automatically.

TCP Orientation wizard

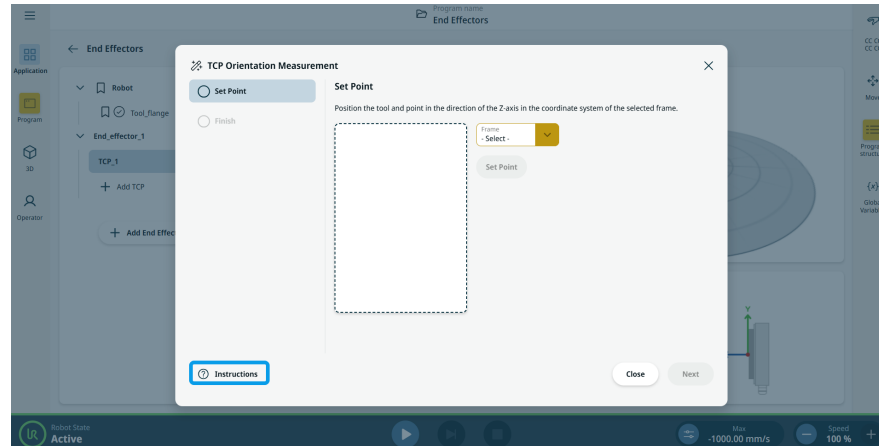
1. On the central panel, tap the TCP Orientation wizard icon,

Orientation

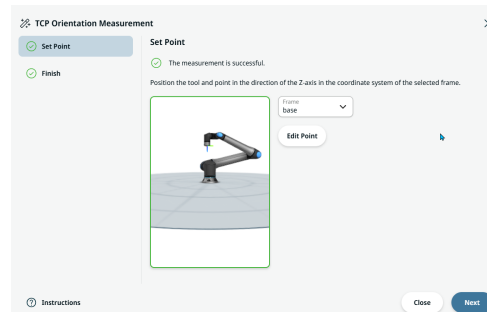


and the pop-up

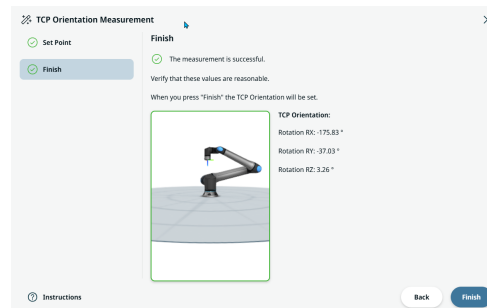
TCP Orientation Measurement screen appears.



2. Tap ? **Instructions** on the bottom left part of the screen.
3. Position the Z-axis of the tool parallel with the Z-axis of the selected frame, for example the base.
4. Tap **Set Point** and jog the robot per prior instruction, then **Save**.
5. When the robot is properly jogged in 3D view, the set point is successful, and tap **Next**.

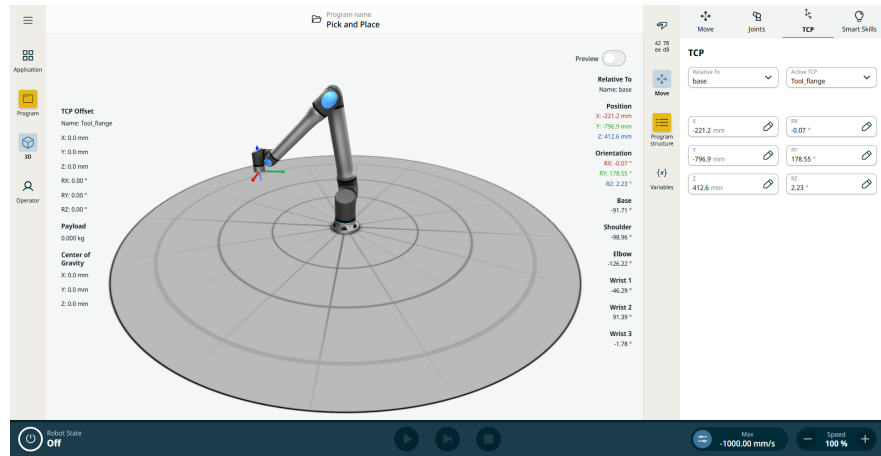


6. To finish, the TCP orientation measurement, verify the RX, RY, and RZ values.



Pose representation

PolyScope X and the ROS 2 interface provide the same TCP pose. The TCP orientation is also the same, but different mathematical representations are used. Therefore, the values are not directly comparable.



9.5.3. Payload and Center of Gravity

Description

The new Payload Estimation wizard allows the robot to help the user set the correct payload and center of gravity (CoG) of a work piece, such as a tool or a combination of attachments to the tool flange.

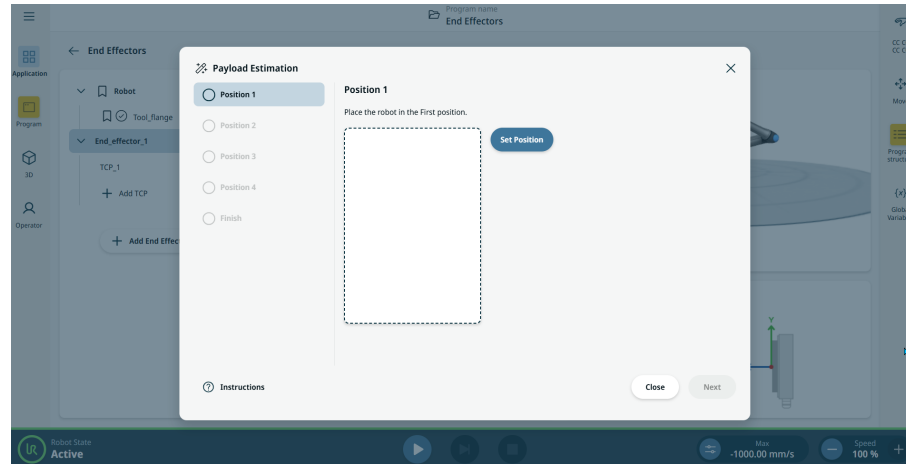
Payload and Center of Gravity (CoG) wizard

1. On the left panel, tap End_effector_1.
2. On the central panel, tap the Payload wizard icon,

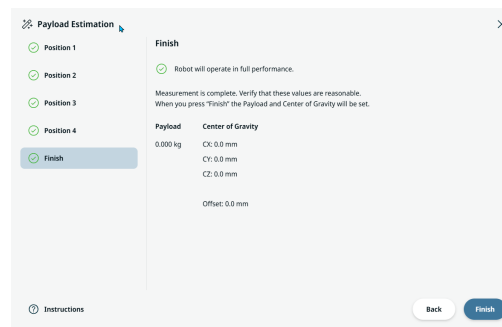


, and the pop-up

Payload Estimation screen appears.



3. By having the user teach four different positions, the robot can estimate the payload mass and CoG automatically. The wizard's final step will indicate, if the computed payload mass or the combination of payload mass and CoG offset may cause reduced robot performance.



As in the TCP position wizard, the user will be informed, if positions need to be re-taught, because the taught positions are not sufficiently diverse. A pop-up with instructions can be opened at any time as in the TCP wizards.



NOTICE

Follow the these guidelines for best Payload Estimation results:

- Ensure the TCP positions are as different as possible from each other
- Perform the measurements within a short time span
- Avoid pulling on the tool and/or attached payload before and during estimation
- Robot mounting and angle must be correctly defined in the installation



Renaming, Duplicating, and Deleting Payloads

You can start configuring a new Payload with the following actions:

- Tap  to rename a Payload.
 - Tap  to duplicate a Payload.
 - Tap  to delete a selected Payload. You cannot remove the last Payload.
-

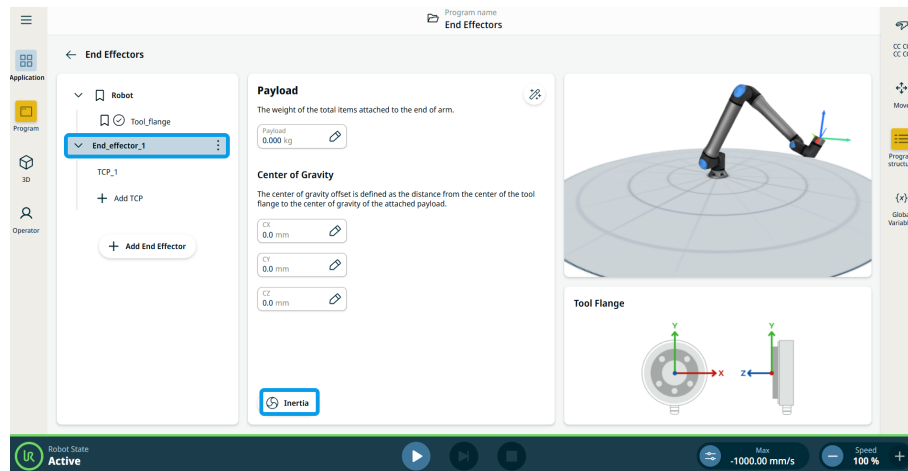
9.5.4. Payload Inertia

Description

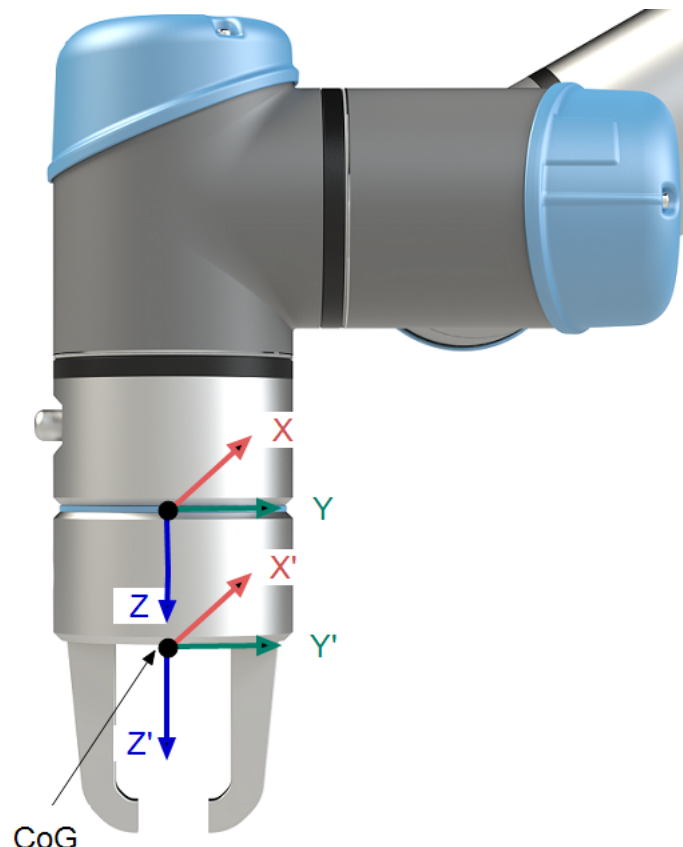
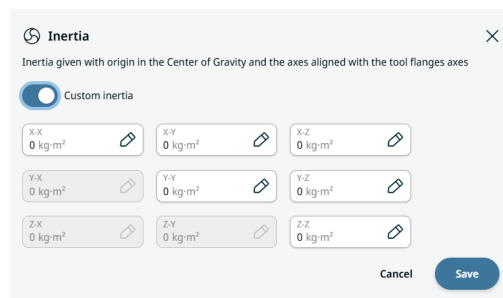
Configuring the inertia of heavy payloads enhance the robot's motion performance and reduces risk of vibrations and protective stops.

Setting Inertia Values

1. Tap **End_effector_1** in the left panel.
2. In the central panel, tap the **Inertia** icon below the Center of Gravity fields.



3. You can select **Custom Inertia** to set inertia values. Slide the button to activate.
4. Tap the fields: **X-X**, **X-Y**, **X-Z**, **Y-Y**, **Y-Z**, and **Z-Z** to set the inertia for the selected payload. **Save**.



9.6. Motion Profiles

Description Motion Profiles application allows you to share move parameters across your program by defining speed and acceleration with a profile name. The profile name can then be referred in any new move node in your program, even if the moves are not grouped in a folder. This application applies to the following commands:

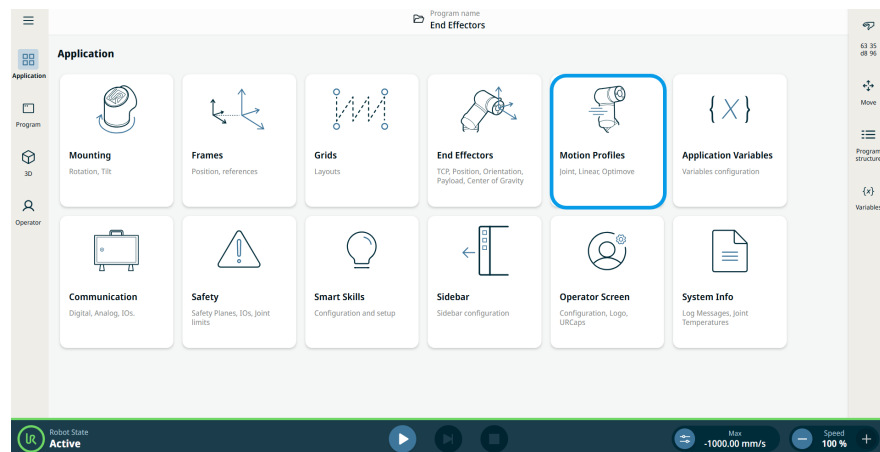
- Joint Move
- Linear/Direction Move
- Process/Circular Move

Intended programming flow

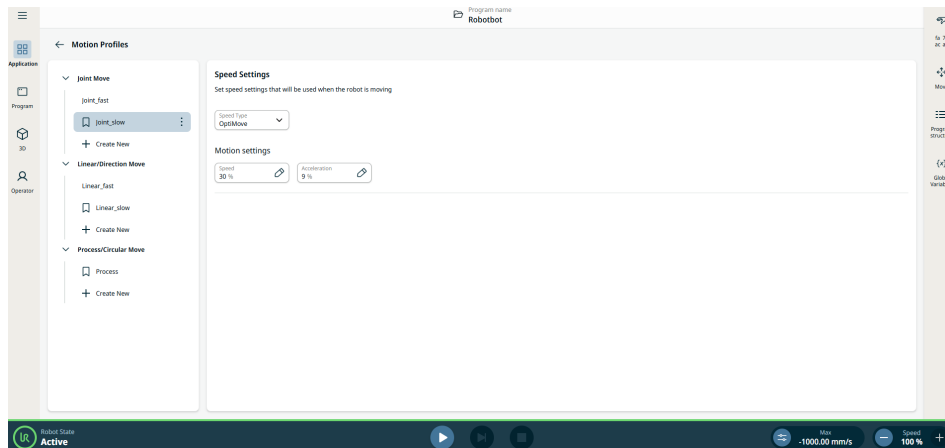
1. User starts by setting up all the Motion Profile needed for the application.
2. Choose a suitable name for easy recognition.
3. Choose the proper Move type (Joint, Linear/Direction, Process Move, Circular arc) from the commands toolbox when the Move node is inserted .
4. Every Move node should use a Motion Profile for easy overview.

Using the Motion Profiles application functionality

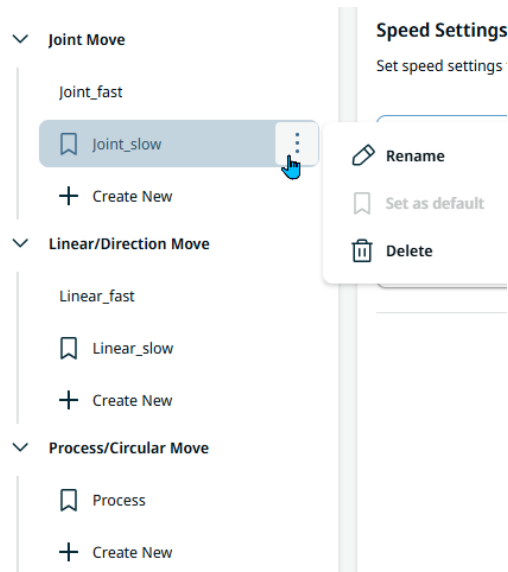
1. Go to the application nodes screen. See [Application Tab](#).
2. Tap the **Motion Profiles** icon.



The Motion Profiles screen appears, which is divided into two panels: left panel contains the three move types, and right panel contains the speed settings.



Left Panel



The left panel contains three move types, which correspond to the three motion profile categories:

- Joint Move
- Linear/Direction Move
- Process/Circular Move

Note: Derived program nodes will use the profile for their underlying move type.

[7.5.4 Joint Move on page 174](#) for joint profile.

[7.5.2 Direction on page 165](#) node for linear profile.

[7.5.1 Circular Arc Move on page 155](#) for process profile.

1. Tap the **kebab icon** and choose **Rename**, **Set as Default**, or **Delete**.
2. Tap the appropriate **+ Create New** action in each profile to add a new motion profile.

Note: The last profile of any move type cannot be deleted.



Right Panel

Speed Settings

Set speed settings that will be used when the robot is moving

Speed Type

OptiMove

▼

Motion settings

Speed

30 %

Acceleration

9 %

The right panel contains three fields of the speed settings:

- Speed Type
- Speed
- Acceleration

1. Tap the **Speed Type** field and choose either **OptiMove** or **Classic**.

OptiMove. This is the recommended option for program safety and reliability, which allows the values to be set as a percentage of the allowed maximum. See [OptiMove](#).

Classic. This speed type enables you to specify values in m/s and m/s².

2. Tap the **Speed** field, and three tabs appear:

- Value
- Variable
- Expression

3. Enter the data in each tab and tap **Confirm**. See [Value-Variable-Expression Tabs](#).

4. Tap the Acceleration field, and three tabs appear:

- Value
- Variable
- Expression

5. Enter the data in each tab and tap **Confirm**. See [Value-Variable-Expression Tabs](#).

Example Program for Motion Profiles

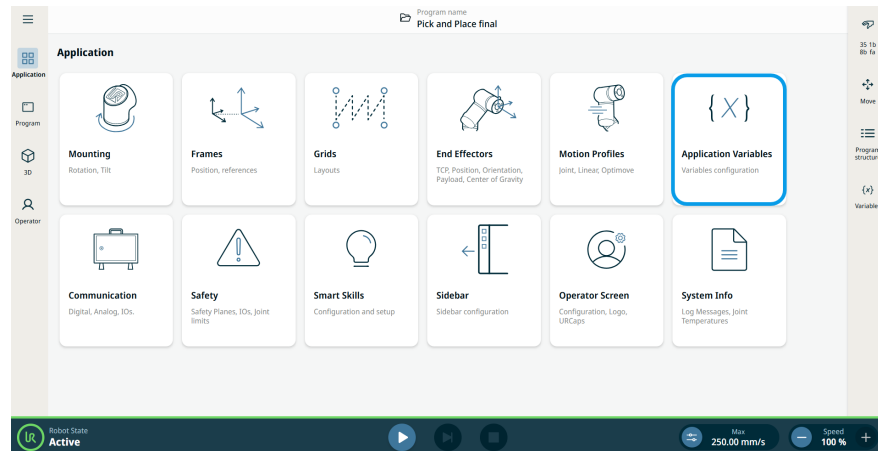
1. Go to steps 1-2 of "Using the Motion Profiles application functionality."
2. On the left panel, tap Process/Circular Move.
3. Tap + Create New, type the motion profile name as "Welding_speed." Save.
4. On the right panel, change the motion settings speed to 50 in the Value tab, and Confirm.
5. Tap the kebab icon of "Welding_speed" and set it as default.
6. Tap the kebab icon of Process, rename it "Medium_speed" and Save.
7. On the right panel, change its motion settings speed to 100 in the Value tab, and Confirm.
8. Tap again + Create New, type the motion profile name "Fast_speed" and Save.
9. On the right panel, change the motion settings speed to 150 in the Value tab, and Confirm.
10. Go to the main program screen. See [Command Nodes](#).
11. Tap the Process Move icon and choose Welding_speed as Point_1.
12. Go to Edit Point button and jog the robot using the X, Y, Z plus-minus button in the multitask screen.
13. Tap the Blend field, enter 5 in the Value tab, and Confirm.
14. Repeat step 10 and tap again the Process Move icon. Choose Medium_speed as Point_2.
15. Repeat step 12.
16. Tap the Blend field, enter 10 in the Value tab, and Confirm.
17. Repeat step 10 and tap again the Process Move icon. Choose Fast_speed as Point_3.
18. Tap the Blend field, enter 15 in the Value tab, and Confirm.
19. In the footer, play the program.

9.7. Application Variables

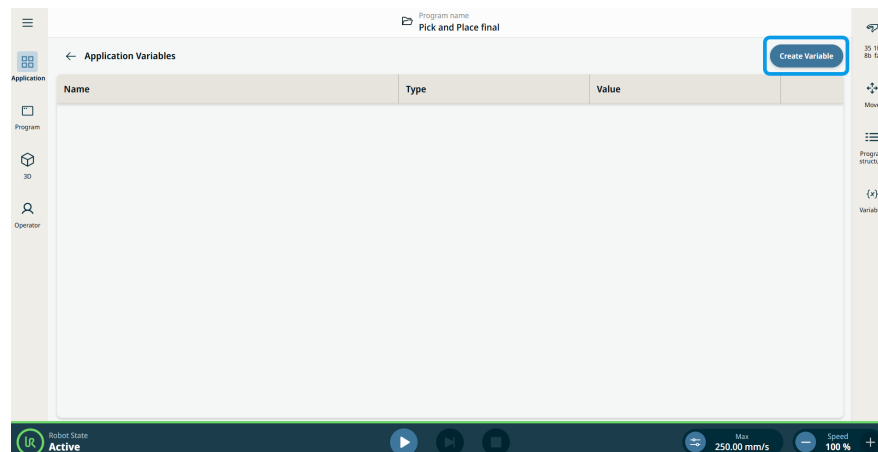
Description **Application Variables** allows you to define a list of variables that are persisting between program runs and robot restarts. It can be used in the same way as any other variable(s) in the program. Along with values, variables can be exported together with program.

Using the Application Variables functionality

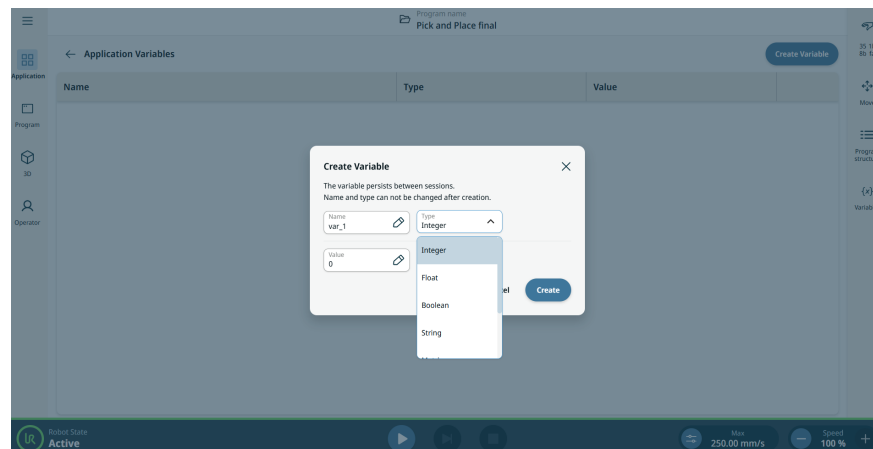
1. Go to the application nodes screen. See [Application Tab](#).
2. Tap the **Application Variables** icon.



3. The Application Variables configuration page displays all created variables for the currently loaded program. Information is sorted in column by name, type, and value.



- On the screen's top right side, tap the **Create Variable** button to add a new variable. A pop-up window appears with the name, type, and value fields for configuration.



- In the **Name** field, enter your variable name. Edit the value in the **Value** field. In the **Type** field, you see the allowed variable types and choose the one pertinent to your program.

9.7.1. Variable Types

Variable Types and examples

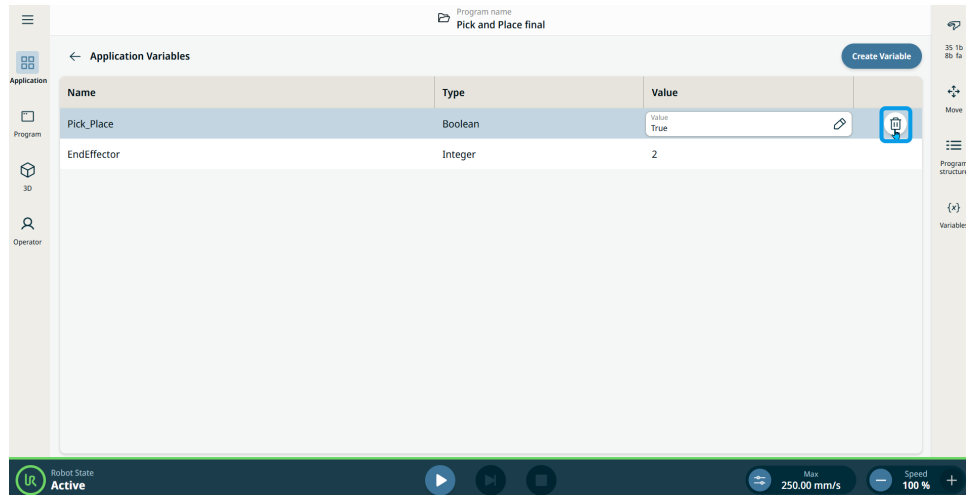
These are the different variable types and their corresponding examples:

- Integer:** numbers [1, 3.14159]
- Float:** numbers [3, 4.24261]
- Boolean:** True, False
- String:** "Next part"
- Matrix:** [[1, 0, 0], [0, 1, 0], [0, 0, 1]]
- Pose:** p[0.1, 0, 0.3, 3.14, 0, 0]
- Array:** array of numbers, strings, poses - [3, 44, 55, 66, 77], ["Start position"], [p[0, 0, 0, 3.14, 0, 0], p[0.3, -0.1, 0.5, 0, 1.57, -0.33]]

9.7.2. Delete Variables

Deleting variables in programs

Tap the trash button to remove a variable. Removing a variable is possible only when a program is stopped. Program will be stopped automatically if it's still running.



NOTICE

Type and name cannot be changed after variable is added. Value can be changed when application is stopped. Value is also updated live when program is running.

9.7.3. Use Variables in Programs

Using variables in programs



NOTICE

Structs, which are complex data types, are not supported in application variables. Communication handler objects (e.g. xmlrpc, ros_subscriber, etc.) are not supported as application variables.

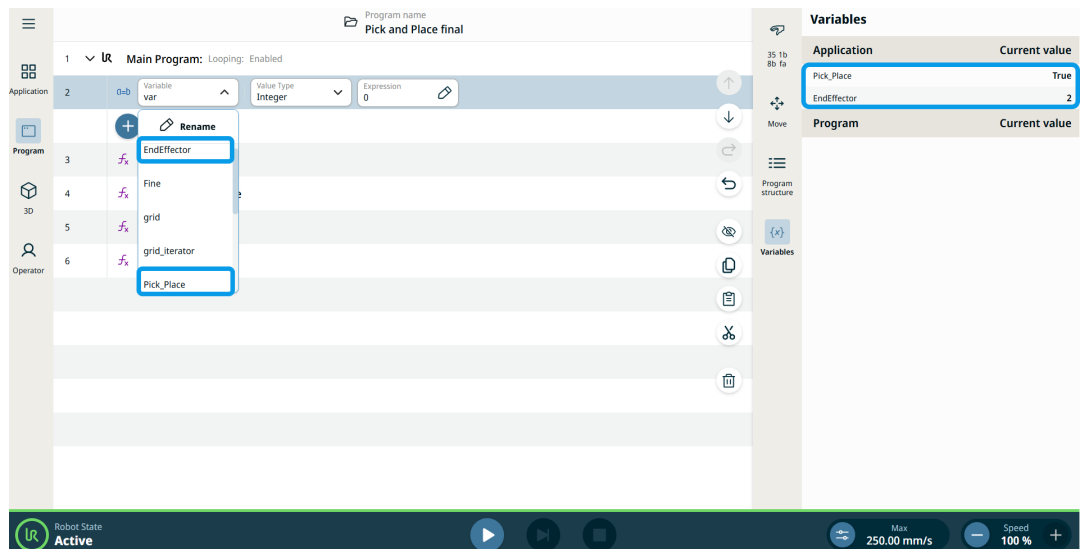
Application variables can be used in the same way as any other variables in the program. They are added to a list of variables available in the dropdown lists.

1. Open the Assignment command node. See [Assignment](#).
2. In the **Variable field**, you can see the variables created in the Application Variables.
3. Tap **Variables** in the sidebar to view the created variables.



NOTICE

See the variables inside the blue box in both the main screen and the multitask screen. Information found in both screens correspond to each other.



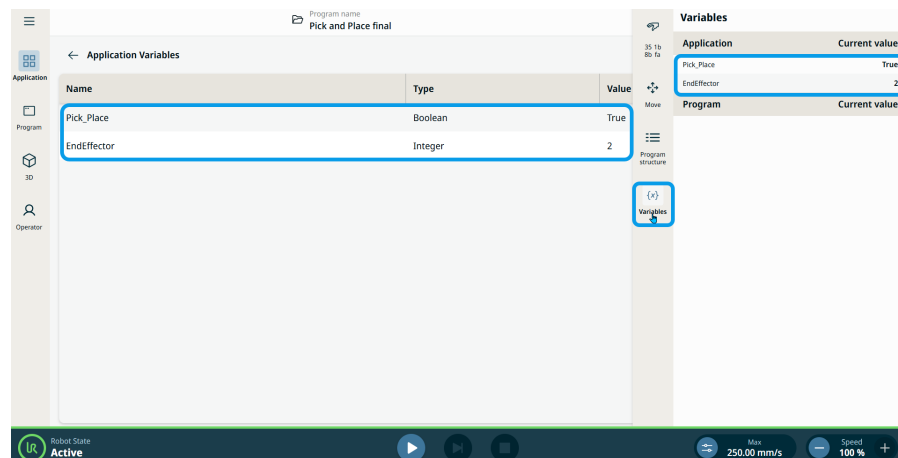
The screenshot displays the PolyScope X software interface. On the left, a sidebar shows the program structure with nodes for Application, Program, and Operator. The main workspace shows a program named 'Pick and Place final' with a 'Main Program' loop. A variable 'var' of type 'Integer' is assigned the value '0'. A dropdown menu is open, showing a list of variables: 'EndEffector', 'Fine', 'grid', 'grid_iterator', and 'Pick_Place'. The 'Pick_Place' variable is highlighted with a blue box. On the right, a 'Variables' panel lists the application variables: 'Pick_Place' with a current value of 'True' and 'EndEffector' with a current value of '2'. The bottom status bar shows the robot state as 'Active' and the speed as '100 %'.

9.7.4. Showing Variables

Showing variable values

Current variable value is shown both on the sidebar in variables section and on the Application Variables screen in Application section.

1. Go to the Application Variables configuration screen, and you see the created variables.
2. Tap **Variables** in the sidebar. The multitask screen expands to the center, and you see the created variables with their current value in the Application list.



9.7.5. Saving Variables

Saving variables

Each program in PolyScope X has its own set of application variables. Variables are saved to the internal program database when program is stopped and automatically every ten minutes.

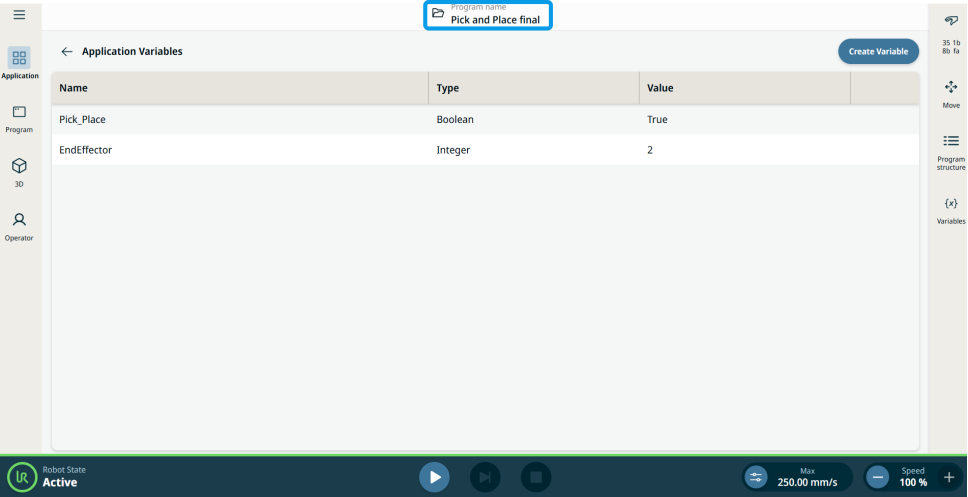


NOTICE

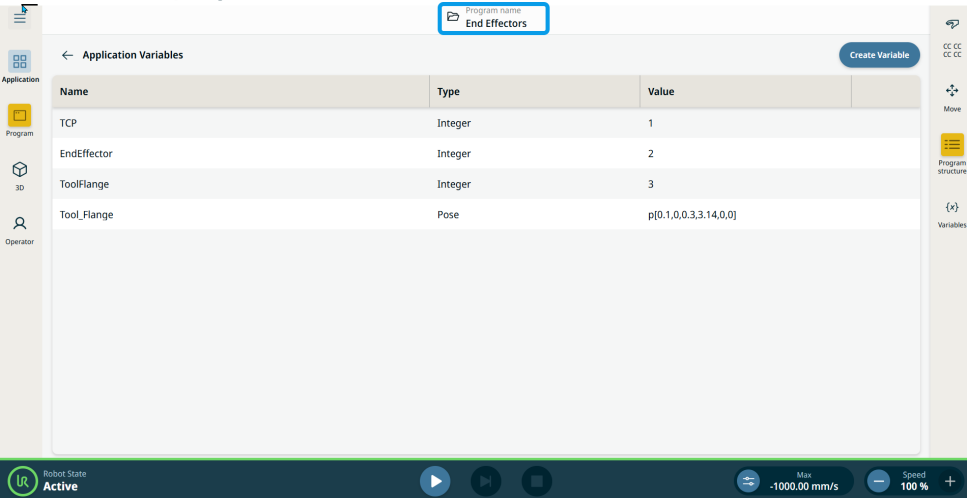
Every time a program is switched, all variables are cleared and replaced with variables defined in the newly loaded program. When a program is loaded again, values of Application Variables are automatically restored to the last state.

See the switch of the application variables between the programs “Pick and Place final” and “End Effectors.”

Pick and Place final program screen



End Effectors program screen





9.7.6. Global Variables in Logic Program

Description

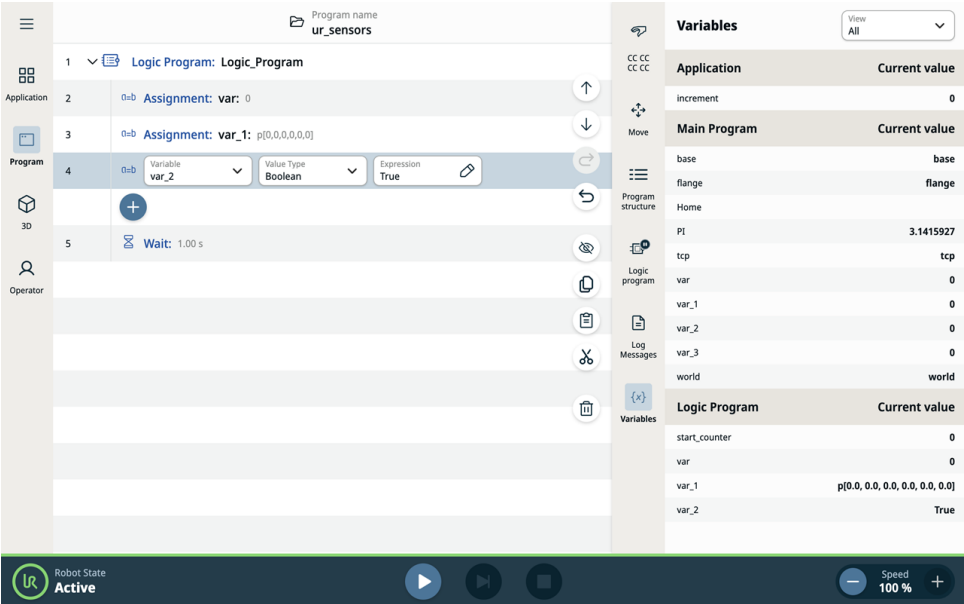
Logic Program variables are shown in the Variables sidebar together with Application and Main Program variables. You can see names and current values for variables that belong to the active Logic Program.

Variables in the sidebar

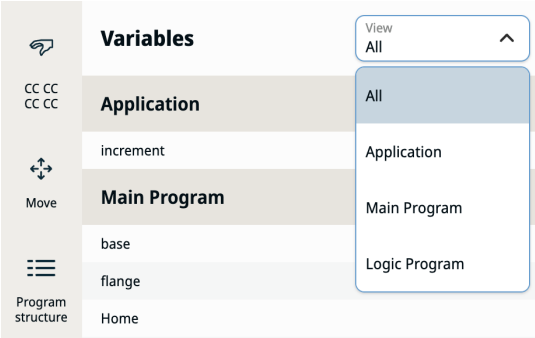


NOTICE

Variable names must not start with the prefix `ur`. This prefix is reserved for internal constants. For compatibility with future improvements, do not use this prefix in robot program variables either.



The filter dropdown in the Variables view lets you choose which set of variables is shown: All, Application, Main Program, or Logic Program.



NOTICE

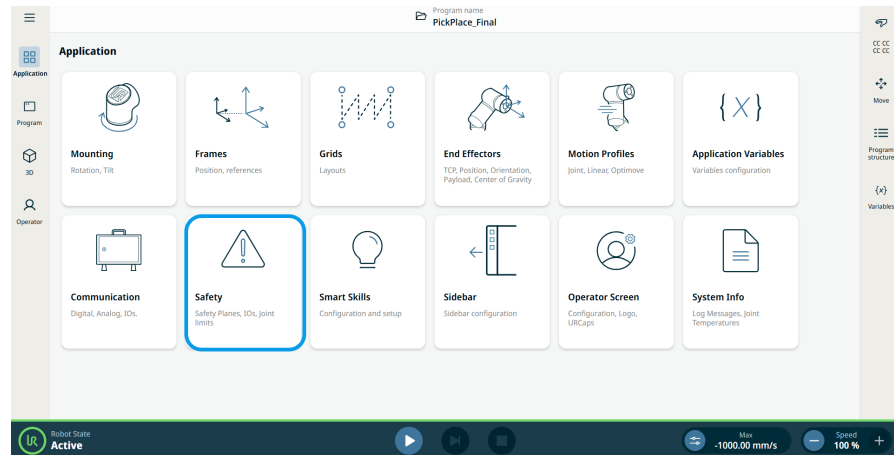
Logic Program supports up to approximately 300 simple variables. Exceeding the limit might stop the running program.

9.8. Safety Application

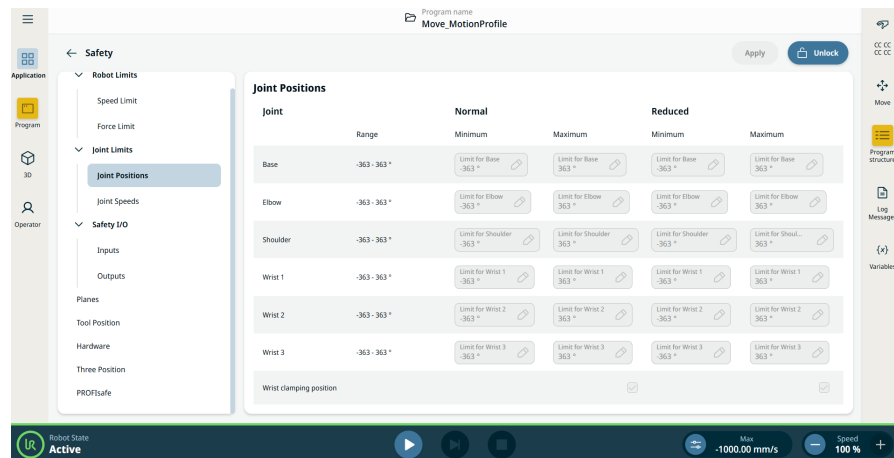
Description The **Safety** application enables you to access the robot safety settings.

Using the Safety application functionality

1. Go to the Application menu screen. See [Application Tab](#).
2. Tap the **Safety** icon.



The safety screen appears, which is divided into the left panel and the right panel.



The left panel consists of the Safety tree, composed of eight functions:

• Robot Limits	• Safety I/O	• Tool Position	• Three Position
• Joint Limits	• Planes	• Hardware	• PROFIsafe

The right panel shows the editable fields of the chosen function, which are inaccessible.

Unlock safety settings

1. To access the safety settings, tap **Unlock** on the top right side of the safety screen.
2. Enter the safety password.
3. Tap **Confirm** to activate all settings.

9.8.1. Safety Menu Settings

Description The safety system limits are defined in the Safety Configuration. The safety system receives values from the input fields and detects any violation if any the values are exceeded. The robot controller prevents violations by making a robot stop or by reducing the speed.



CAUTION

Use of Safety Configuration parameters different from those defined by the risk assessment can result in hazards that are not reasonably eliminated or risks that are not sufficiently reduced.

9.8.2. Robot Limits

Description Robot Limits restrict general robot movements.

- Change the values in robot limits (Normal and Reduced) and apply to see the applied changes in safety checksum.
- Verify the values of reduced should be always less than the values in Normal.



NOTICE

The safety limits restrict forces and motions at the tool flange and the center of the two user-defined tool positions

Robot State
Off

35 7a
c7 de

Program name
Default program

35 7a
c7 de

Program name
Default program

Safety

Apply

Unlock

▼ **Robot Limits**

Speed Limit

Power & Force

▼ **Joint Limits**

Joint Positions

Joint Speeds

▼ **Safety I/O**

Inputs

Outputs

Safe Home

Planes

I/O Planes

Tool Position

Hardware

Speed Limit

Use of Safety Configuration parameters different from those defined by the risk assessment can result in hazards that are not reasonably eliminated or risks that are not sufficient reduced

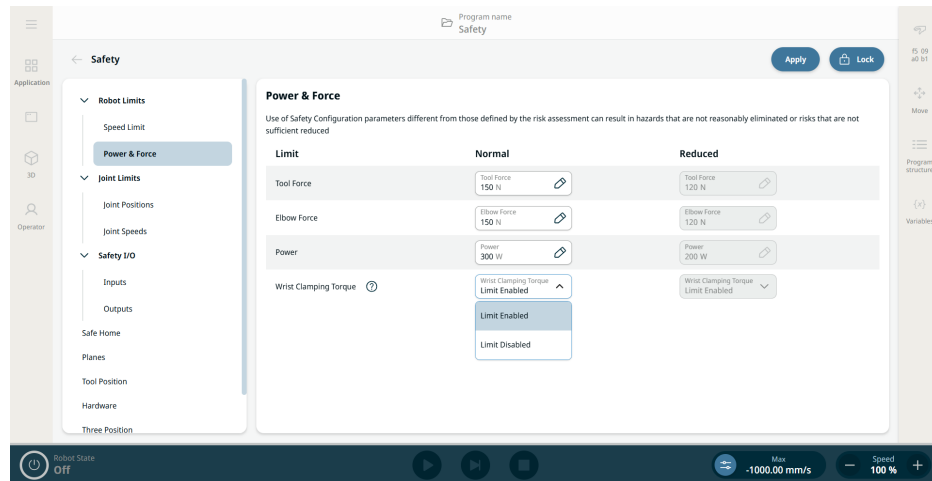
Limit	Normal	Reduced
Tool Speed	Tool Speed 1.5 m/s	Tool Speed 0.75 m/s
Elbow Speed	Elbow Speed 1.5 m/s	Elbow Speed 0.75 m/s
Momentum	Momentum 25 kg·m/s	Momentum 10 kg·m/s
Stopping Time	Stopping Time 0.4 s	Stopping Time 0.3 s
Stopping Distance	Stopping Distance 0.5 m	Stopping Distance 0.3 m

NOTICE

Restricting stopping time and distance affect overall robot speed. For example, if stopping time is set to 300 ms, the maximum robot speed is limited allowing the robot to stop within 300 ms.

Limit	Description
Tool Speed	Limits maximum robot tool speed.
Elbow Speed	Limits maximum robot elbow speed.
Momentum	Limits maximum robot momentum.
Stopping Time	Limits maximum time it takes the robot to stop e.g. when an emergency stop is activated.
Stopping Distance	Limits maximum distance the robot tool or elbow can travel while stopping.

Power and Force



NOTICE

The tool and elbow forces may be exceeded by the three wrist joints if the "wrist clamping torque" safety function is disabled

Limit	Description
Tool Force	Limits the maximum force exerted by the robot tool in clamping situations.
Elbow Force	Limits maximum force that the elbow exerts on the environment.
Power	Limits maximum mechanical work produced by the robot in the environment. This limit considers the payload a part of the robot and not of the environment.
Wrist Clamping Torque	Limits how much torque the wrist can apply for pushing. Enabled by default.

Safety Mode

When no protective stops are active, the safety system operates in a Safety Mode associated with a set of safety limits:

Safety mode	Effect
Normal	This configuration is active by default.
Reduced	This configuration activates when the Tool Center Point (TCP) is positioned beyond a Trigger Reduced mode plane, or when triggered using a configurable input.

Wrist Clamping Torque Limit



Description **Wrist Clamping Torque Limit** is a setting that enables or disables limitation of the maximum clamping torque for the wrist joints of the robot. The limitation is enabled by default. This safety function limits how much torque the wrist can apply when pushing.

Wrist Clamping Torque Limit details

When **enabled**, the robot limits torque in the wrist joints to prevent clamping between the wrist, payload, and the lower arm of the robot.



NOTICE

When the setting is enabled, it also defines a non-safety-rated upper limit for the applied torque windows of the base, shoulder, and elbow joints. In force control applications, the achievable contact forces may be significantly lower than the specified forces in the **Tool Force Limit** and **Elbow Force Limit** safety functions.

When **disabled**, the robot can use higher pushing force with the wrist joints, for example in force-controlled applications. When the function is disabled, risks of clamping between the wrist, payload, and lower arm of the robot must be mitigated by other means.

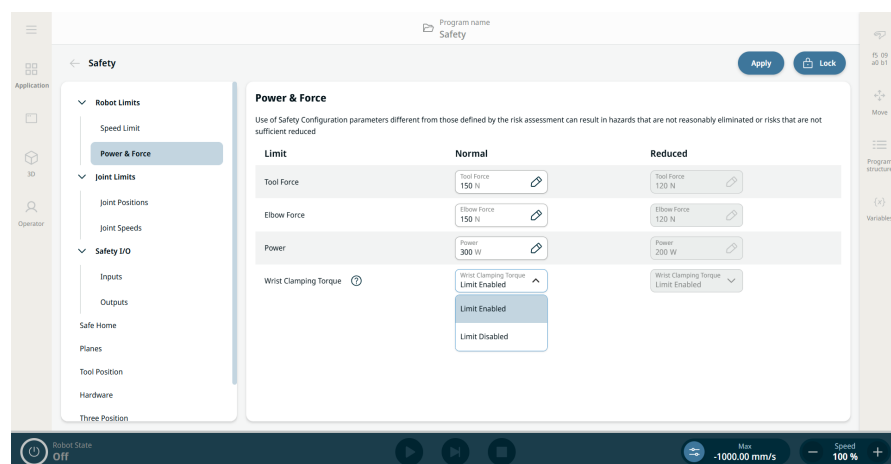


NOTICE

This safety function is enabled on all robots by default. In PolyScope X version 10.12 and newer, it is possible to disable this safety function.

Enable and disable Wrist Clamping Torque Limit

1. Go to the **Application** tab and select **Safety**.
2. Under **Robot Limits**, tap **Power and Force**. Unlock, enter the safety password, and confirm.
3. You can now select **Limit Enabled** or **Limit Disabled** for Normal or Reduced.



4. Tap **Apply** to apply the safety configuration and restart the safety system.
5. A pop-up box appears where you can revert or confirm the configuration.

9.8.3. Joint Limits

Description Joint limits allow you to restrict individual robot joint movements in joint space by joint rotational position and joint rotational speed. Joint limiting can also be called software-based axis limiting. These are the robot joints:

- Base
- Elbow
- Shoulder
- Wrist 1
- Wrist 2
- Wrist 3

About the Normal and Reduced safety modes, read [Robot Limits](#).

Joint Limits options

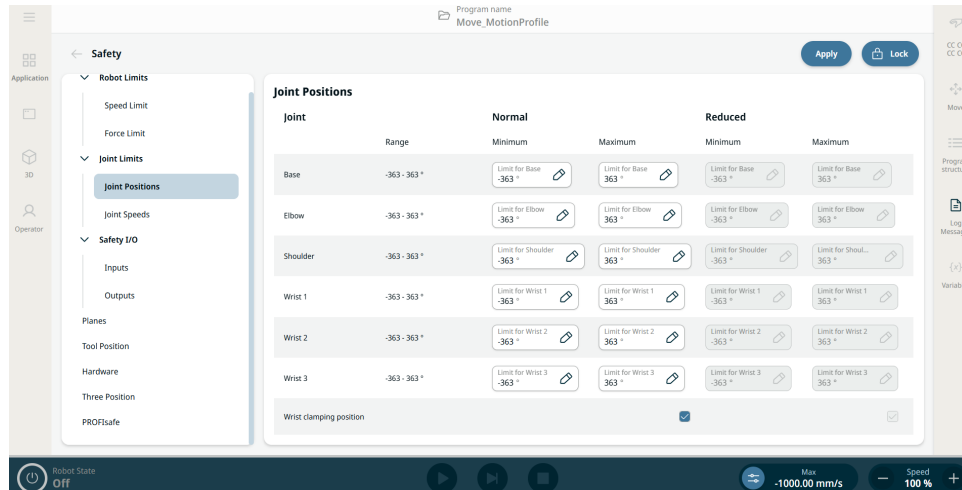
There are two Joint Limits options: Maximum speed and Position range.

Maximum speed: defines the maximum angular velocity for each joint.

Position range: defines the position range for each joint. Again, the input fields for Reduced mode are disabled if there is no safety plane or configurable input set to trigger it. This limit enables safety-rated soft axis limiting of the robot.

Joint Positions

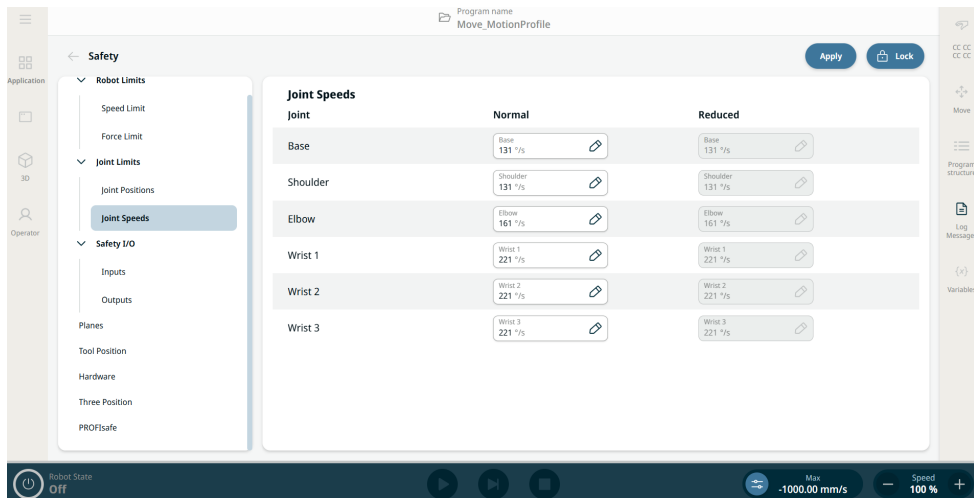
Sets upper and lower limits for the allowed joint positions relative to the base of the robot.



Joint	Range	Normal		Reduced	
		Minimum	Maximum	Minimum	Maximum
Base	-363 - 363 °	Limit for Base -363 °	Limit for Base 363 °	Limit for Base -363 °	Limit for Base 363 °
Elbow	-363 - 363 °	Limit for Elbow -363 °	Limit for Elbow 363 °	Limit for Elbow -363 °	Limit for Elbow 363 °
Shoulder	-363 - 363 °	Limit for Shoulder -363 °	Limit for Shoulder 363 °	Limit for Shoulder -363 °	Limit for Shoulder 363 °
Wrist 1	-363 - 363 °	Limit for Wrist 1 -363 °	Limit for Wrist 1 363 °	Limit for Wrist 1 -363 °	Limit for Wrist 1 363 °
Wrist 2	-363 - 363 °	Limit for Wrist 2 -363 °	Limit for Wrist 2 363 °	Limit for Wrist 2 -363 °	Limit for Wrist 2 363 °
Wrist 3	-363 - 363 °	Limit for Wrist 3 -363 °	Limit for Wrist 3 363 °	Limit for Wrist 3 -363 °	Limit for Wrist 3 363 °
Wrist clamping position		☒			

Joint Speeds

Sets an upper limit for joint speed relative to the base of the robot.



Joint	Normal	Reduced
Base	131 %/s	131 %/s
Shoulder	131 %/s	131 %/s
Elbow	161 %/s	161 %/s
Wrist 1	221 %/s	221 %/s
Wrist 2	221 %/s	221 %/s
Wrist 3	221 %/s	221 %/s

Wrist Clamping Position Limit

Description **Wrist Clamping Position Limit** is a setting that enables or disables the **Wrist Clamping Position Limit** safety function. The safety function is enabled by default and prevents hand or finger clamping between the tool flange and the lower arm of the robot.

Wrist Clamping Position Limit details

When **enabled**, a Robot Stop is issued if the tool flange approaches a distance of 25.0 mm to the lower arm of the robot, reducing the risk of collisions or pinching.



NOTICE

This safety function monitors only the distance between the tool flange and the lower arm of the robot. It does not monitor the distance between mounted equipment and the robot.

When **disabled**, full wrist movement is allowed, and risks of clamping between the tool and the lower arm of the robot must be mitigated by other means.

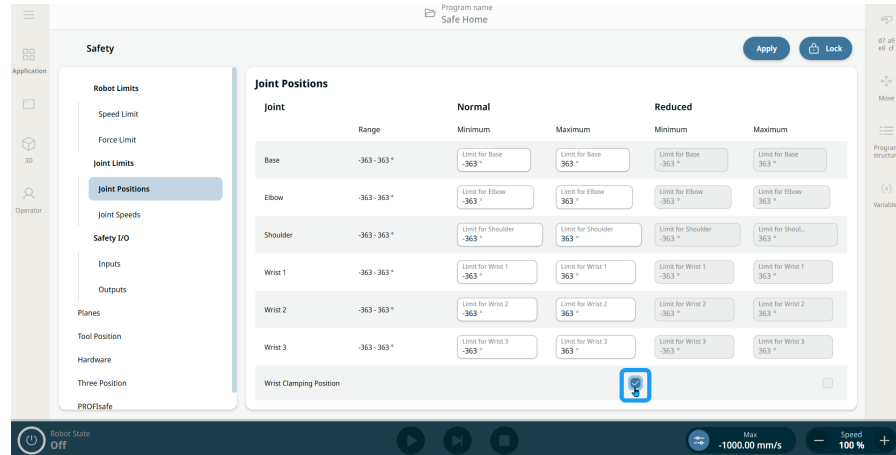


NOTICE

This safety function is enabled on all robots by default. In PolyScope X version 10.12 and newer, it is possible to disable this safety function.

Enable and disable Wrist Clamping Position

1. Go to the **Application** tab and select **Safety**.
2. Under **Joint Limits**, tap **Joint Positions**. Unlock, enter the safety password, and confirm.
3. When the **Joint Positions** screen is activated, select the **Wrist Clamping Position** checkbox.



4. Tap **Apply** to apply the safety configuration and restart the safety system.
5. A pop-up box appears where you can revert or confirm the configuration.

9.8.4. Safety I/O

Description

The I/O are divided between inputs and outputs and are paired up so that each function provides a Category 3 PLd capability.

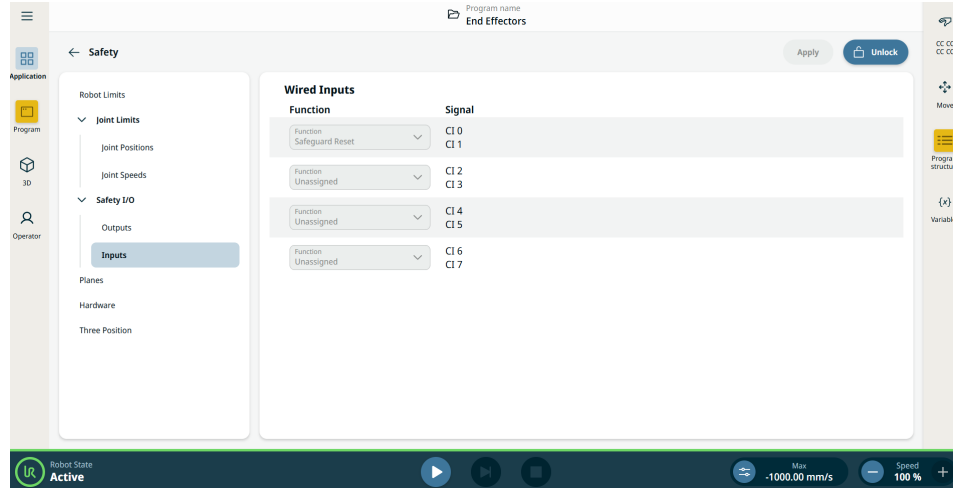
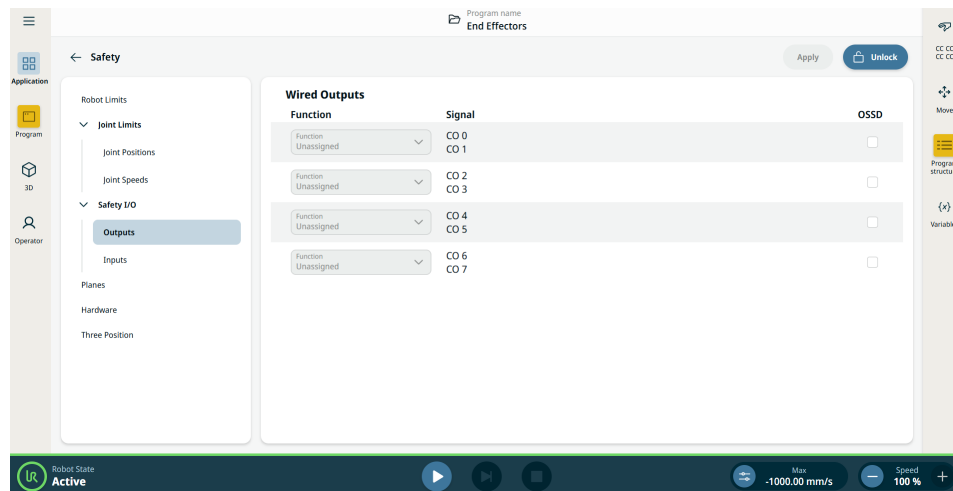


Figure 1.2: PolyScope X screen displaying the Input signals.



NOTICE

When starting programs from an I/O or fieldbus input, the robot can begin movement from the position it has, there will not be any manual movement to the first waypoint via PolyScope required.

Control Box Inputs The inputs are described in the tables below:

Emergency Stop Button	Performs a Stop Category 1 (IEC 60204-1) informing other machines using the System Stop output if that output is defined. A stop is initiated in anything connected to the output.
Robot Emergency Stop	Performs a Stop Category 1 (IEC 60204-1) via Control Box input, informing other machines using the System Emergency Stop Output if that output is defined.
External Emergency Stop	Performs a Stop Category 1 (IEC 60204-1) on robot only.
Reduced	All safety limits can be applied while the robot is using a Normal configuration, or a Reduced configuration. When configured, a low signal sent to the inputs causes the safety system to transition to the reduced configuration. The robot arm decelerates to satisfy the reduced parameters. The safety system guarantees the robot is within reduced limits less than 0.5s after the input is triggered. If the robot arm continues to violate any of the reduced limits, a Stop Category 0 is triggered. Trigger planes can also cause a transition to the reduced configuration. The safety system transitions to the normal configuration in the same way.



The inputs are described in the table below:

Control
Box
Inputs

Operational Mode	When an external mode selection is used it switches between Automatic Mode and Manual Mode . The robot is in Automatic mode when input is <i>low</i> and Manual mode when the input is <i>high</i> .
Safeguard Reset	Returns from the Safeguard Stop state, when a rising edge on the Safeguard Reset input occurs. When a Safeguard Stop occurs, this input ensures that the Safeguard Stop state continues until a reset is triggered.
Safeguard	A stop triggered by a safeguard input. Performs a Stop Category 2 (IEC 60204-1) in all modes, when triggered by a Safeguard.
Automatic Mode Safeguard Stop	Performs a Stop Category 2 (IEC 60204-1) in Automatic mode ONLY. <i>Automatic Mode Safeguard Stop</i> can only be selected when a Three-Position Enabling Device is configured and installed.
Automatic Mode Safeguard Reset	Returns from the Automatic Mode Safeguard Stop state when a rising edge on the Automatic Mode Safeguard Reset input occurs.
Freedrive on robot	You can configure the Freedrive input to enable and use Freedrive without pressing the Freedrive button on a standard TP, or without having to press-and-hold any of the buttons on the 3PE TP in the light-press position.
3-Position Enabling Device	In Manual Mode, an external 3-Position Enabling Device must be pressed and held in the center-on position to move the robot. If you are using a built-in 3-Position Enabling Device, the button must be pressed and held in the mid position to move the robot.



WARNING

When the default Safeguard Reset is disabled, an automatic reset happens when the safeguard no longer triggers a stop.

This can happen if a person passes through the field of the safeguard.

If a person is not detected by the safeguard and the person is exposed to hazards, automatic reset is forbidden by standards.

- Use the external reset to ensure resetting only when a person is not exposed to hazards.



WARNING

When Automatic Mode Safeguard stop is enabled, a safeguard Stop is not triggered in Manual Mode.

Contr All safety outputs go low in the event of a safety system violation or fault. This means the System
ol Box Stop output initiates a stop even when an E-stop is not triggered.
Output You can use the following Safety functions output signals. All signals return to low when the state
ts which triggered the high signal has ended:

¹ System Stop	Signal is <i>Low</i> when the safety system has been triggered into a stopped state including by the Robot Emergency Stop input or the Emergency Stop Button. To avoid deadlocks, if the Emergency Stopped state is triggered by the System Stop input, low signal will not be given.
Robot Moving	Signal is <i>Low</i> if the robot is moving, otherwise high.
Robot Not Stopping	Signal is <i>High</i> when the robot is stopped or in the process of stopping due to an emergency stop or safeguard stop. Otherwise it will be logic low.
Reduced	Signal is <i>Low</i> when reduced parameters are active or if the safety input is configured with a reduced input and the signal is currently low. Otherwise the signal is high.
Not Reduced	This is the inverse of Reduced, defined above.
Safe Home	Signal is <i>High</i> if the robot arm is stopped and is located in the configured Safe Home Position. Otherwise, the signal is <i>Low</i> . This is often used when UR robots are integrated with mobile robots.
3-Position Enabling Stopped	Signal is low when a three position stop is active, high otherwise.
Not 3-Position Enabling Stopped	Signal is low when a three position stop is inactive, high otherwise.



NOTICE

Any external machinery receiving its Emergency Stop state from the robot through the System Stop output must comply with ISO 13850. This is particularly necessary in setups where the Robot Emergency Stop input is connected to an external Emergency Stop device. In such cases, the System Stop output becomes high when the external Emergency Stop device is released. This implies that the emergency stop state at the external machinery will be reset with no manual action needed from the robot's operator. Hence, to comply with safety standards, the external machinery must require manual action in order to resume.

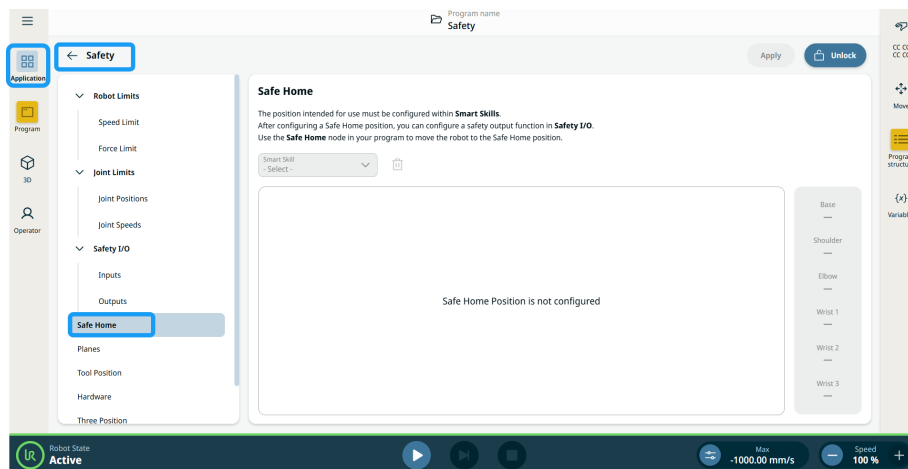
¹System Stop was previously known as "System Emergency Stop" for Universal Robots robots. PolyScope can display "System Emergency Stop".

9.8.5. Safe Home

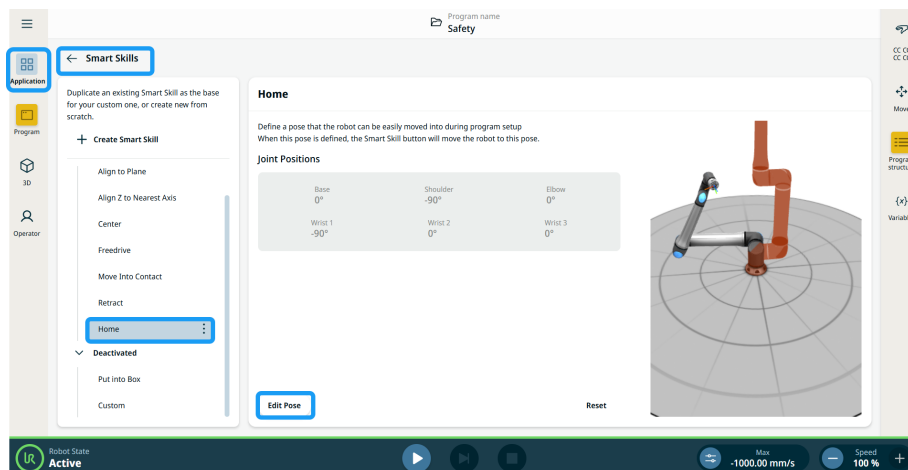
Description **Safe Home** is a PolyScope X feature that enables users to define a safety-related output that activates when the robot reaches the configured safe home position. The user chooses **Home** in the dropdown of available smart skill positions, which can be assigned as the reference for the Safe Home position.

Access Safe Home

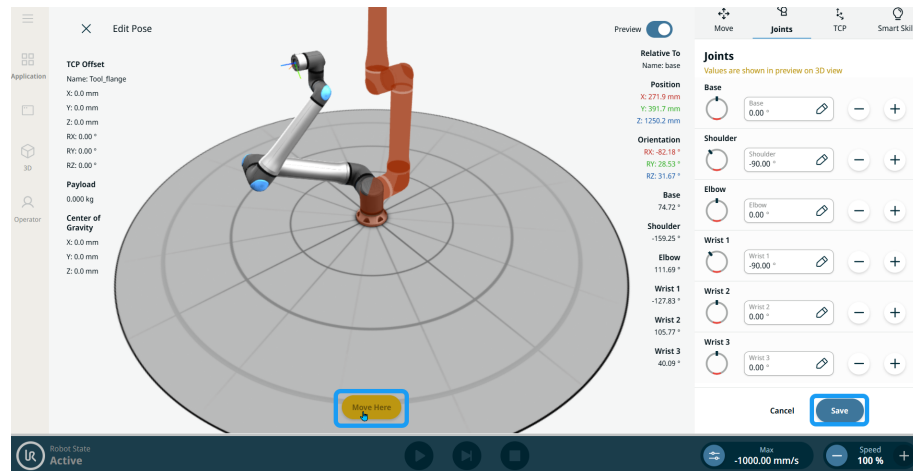
1. Tap the **Application Tab** and choose **Safety Application**. In the left panel, choose **Safe Home**.



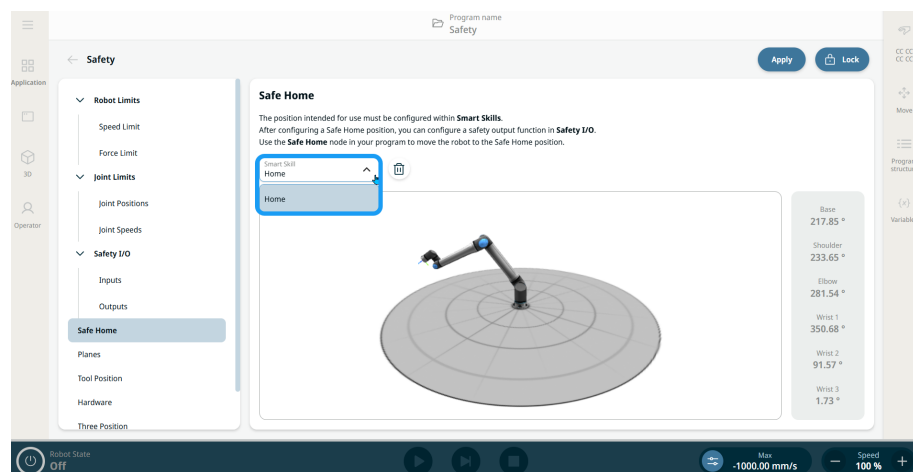
2. To configure the robot position, go to **Smart Skills Application** and choose **Home**.
3. Tap the **Edit Pose** button, and the 3D view of the robot arm appears.



4. Jog the robot to the intended position using the **Move**, **Joints**, **TCP** buttons.
5. Long press the yellow **Move Here** button until the previewed image is positioned to the new configuration.
6. Tap **Save** to finalize the configuration.



7. Go back to **Safe Home** in **Safety Application** and enter the safety password.
8. In the activated **Smart Skill** field, select **Home**, and the configured robot position appears in the main screen.
9. Tap **Apply** and choose **Apply and restart**. Then tap **Confirm Configuration**.



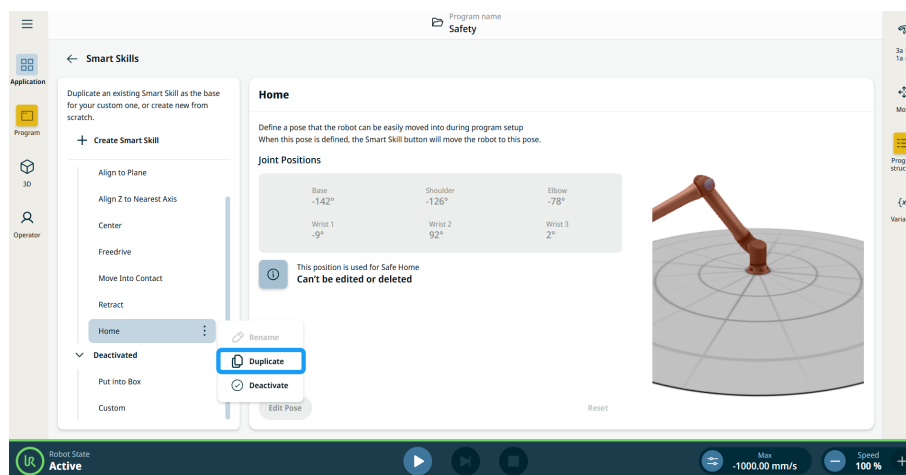


NOTICE

If the Position Smart Skill is selected as the position reference for Safe Home:

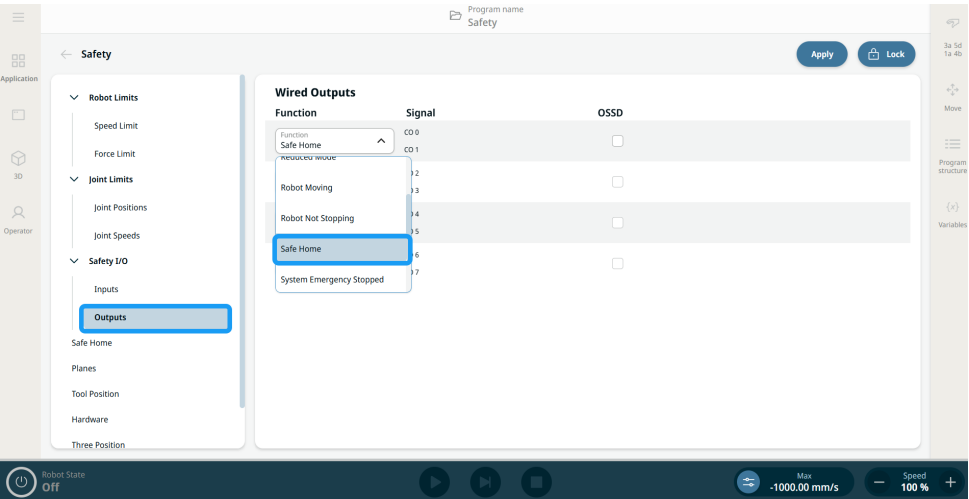
- The message box is displayed on the Position Smart Skill page.
- The Position Smart Skill configured as Safe Home cannot be renamed, edited, or deleted unless the user deletes or unassigns the Safe Home from the **Safety > Safe Home** screen.

10. In the Smart Skills screen, the Home position can be duplicated and edited. All the newly created Smart Skills are also available to be configured as Safe Home.

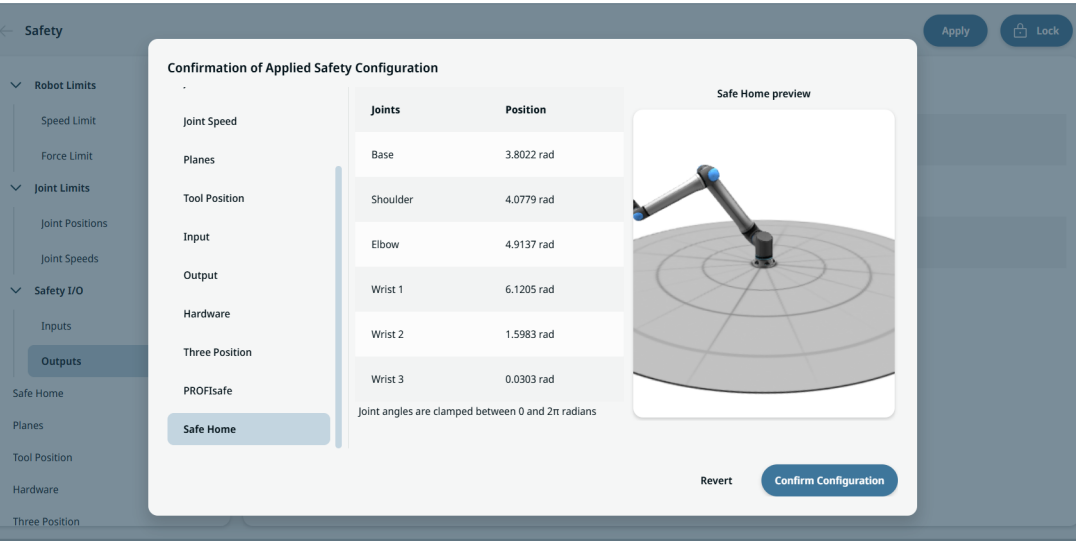


Safe Home
function in
outputs

If the **Home** position is assigned to a position smart skill in **Safety**, the safety output function dropdown contains **Safe Home** as a new function assignment. Choose **Safe Home** in the **Function** fields in **Outputs** of Safety I/O.



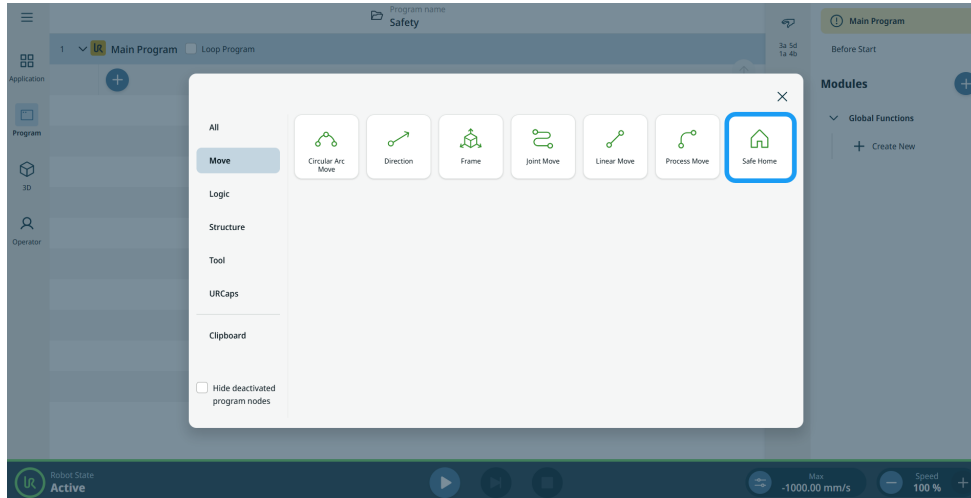
Safe Home can be seen in the table of function assignment on the **Confirmation of Applied Safety Configuration** overview dialog.





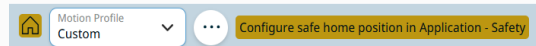
Safe Home as a program node

The **Safe Home** program node is located under the **Move nodes** category within the **Command Nodes**. This node is used to command the robot to move to its predefined safe home position, as configured in the safety application settings for safe home.



NOTICE

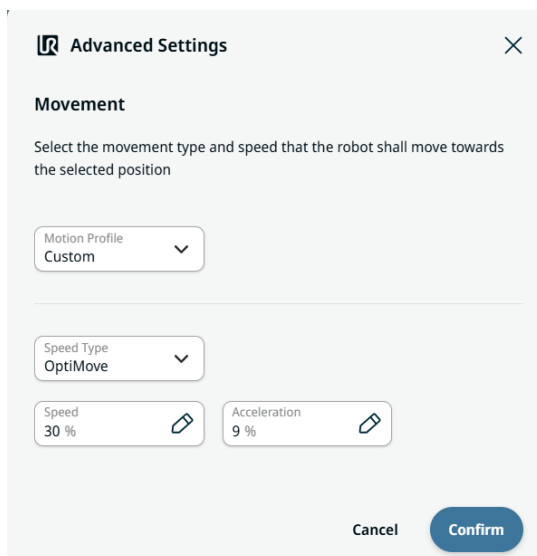
Adding the node without a safe home position configured in safety will mark the node as yellow.



The Safe Home command node has motion profiles options and other advanced settings



when the ellipsis icon is tapped.



Advanced Settings

Users can create movement profiles in the Advanced Settings using the two tabs and two fields:

- Motion Profile
- Speed Type
- Speed tab
- Acceleration tab

Motion Profile is a feature enabling users to set predefined motion profiles in move nodes: Joint Move, Linear Move, Circular Arc Move, Process Move, and Direction. **Custom** Motion Profile enables users to define movement data in move node.

In the **Speed Type** field, **OptiMove** is the default option. Choose **Classic** movement if you want to input speed and acceleration in degrees per second or mm per second. OptiMove settings are specified in percent to simplify the usage and setup.

Users can configure the percentage of preferred speed in the **Speed tab**, which sets the target travel speed as a percentage of the robot's maximum speed capability.

Users can configure the percentage of preferred acceleration in the **Acceleration tab**, which sets the target torque limits during acceleration and deceleration as a percentage of the robot's maximum power.

9.8.6. Tool Position Restriction

Description

The Tool Position screen enables users a more controlled restriction of tools and/or accessories placed on the end of the robot arm by letting you define tool positions with a radius that will interact with the safety planes by either collision detection with Tool Position and plane or enter reduced mode when tool enters plane.

Details

Tool Position has two key benefits:

- Supports two custom configurations to specify where to react to safety planes.
- Visualizes Tool Positions in 3D model.



NOTICE

You can define, configure, and manage up to two tool positions.

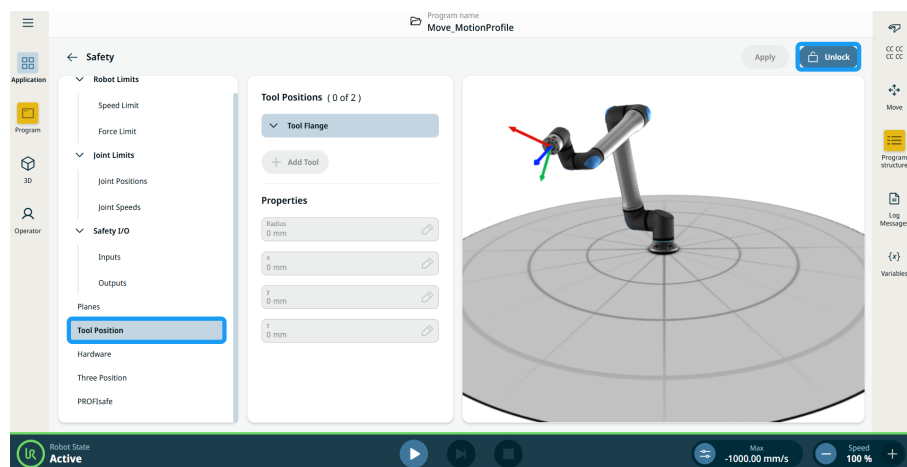
User defined tools

For the user defined tools, the user can change:

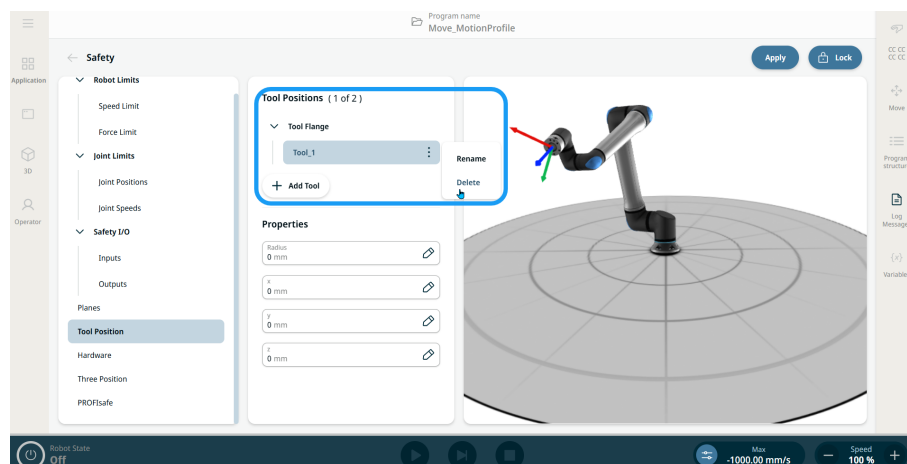
- Radius to change the radius of the tool sphere. The radius is considered when using safety planes.
- X, Y, Z positions to change the position of the tool with respect to the tool flange of the robot. The position is considered for the safety functions for tool speed, tool force, stopping distance, and safety planes.

To access Tool Position

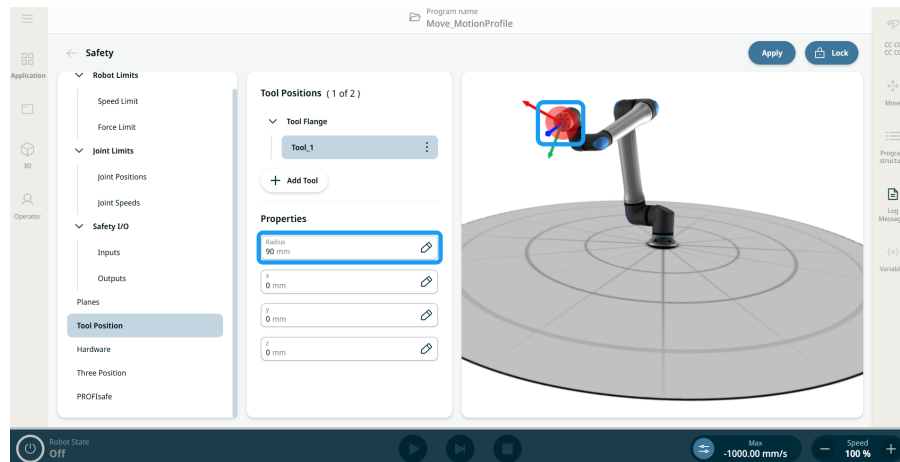
1. Navigate to **Safety Application**.
2. Tap **Tool Position** in the left panel. On the top right side of the main screen, tap **Unlock** to activate adding tools. Enter the safety password and **Confirm**.



3. In the central panel's **Tool Positions** column, tap **+ Add Tool**. The added tool, **Tool_1**, appears under the **Tool Flange** tree.
4. Tap the **kebab icon** of the added tool to rename it to something more identifiable. You can also delete it.



5. In the central panel's **Properties** column, you can find four editable fields for **radius**, **x**, **y**, and **z positions**. Tap the fields to change the radius and x ,y, z coordinates as needed. The sphere in the right panel updates live in the 3D model to assist with accurate placement.
6. Tap **Apply** in the top-right part of the main screen.



7. The robot will now interact with safety planes when the tool position spheres come into contact with them.

9.8.7. Planes

Description

Safety planes restrict robot workspace, the tool, and the elbow.



WARNING

Defining safety planes only limits the defined Tool spheres and elbow, not the overall limit for the robot arm.

Defining safety planes does not guarantee that other parts of the robot arm will obey this type of restriction.

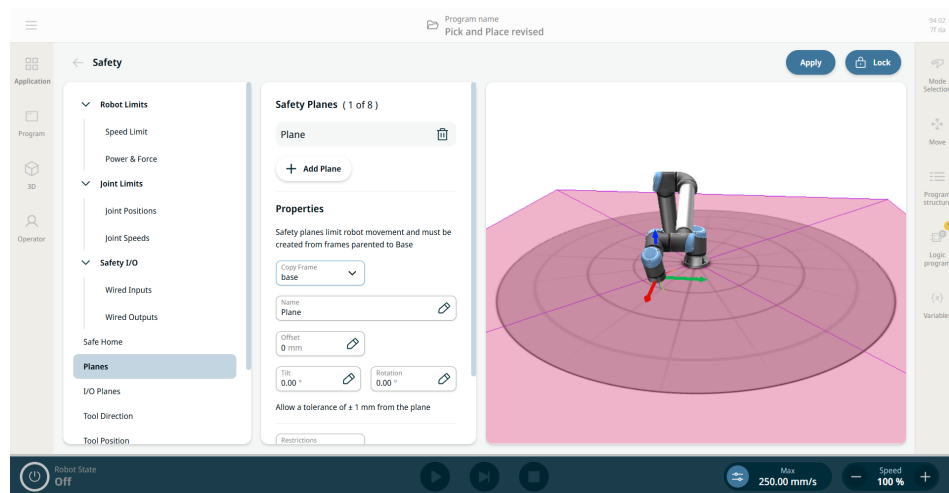


Figure 1.3: PolyScope X screen displaying safety planes.

Configuring a Safety Plane

You can configure safety planes with the properties listed below:

- **Name.** This is the name used to identify the safety plane.
- **Offset from base.** This is the height of the plane from the base, measured in the Y direction.
- **Tilt.** This is the tilt of the plane, measured from the power cord.
- **Rotation.** This is the rotation of the plane, measured clockwise.

You can configure each plane with the restrictions listed below:

- **Normal.** When the safety system is in Normal mode, a normal plane is active and it acts as a strict limit on the position.
- **Reduced.** When the safety system is in Reduced mode, a reduced mode plane is active and it acts as a strict limit on the position.
- **Both.** When the safety system is either in Normal or Reduced mode, a normal and reduced mode plane is active and acts as a strict limit on the position.
- **Trigger Reduced Mode.** The safety plane causes the safety system to switch to Reduced mode if the robot Tool or Elbow is positioned beyond it.

Elbow Joint Restriction

The feature is enabled by default.
 You can use Restrict Elbow to prevent robot elbow joint from passing through any of your defined planes.
 Disable Restrict Elbow for elbow to pass through planes.

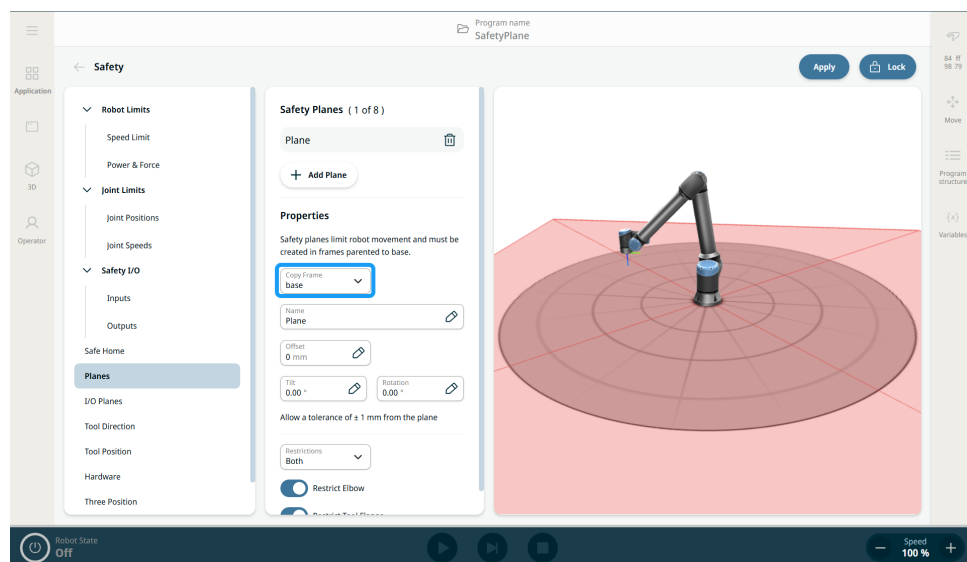
Tool Flange Restriction

Restricting the tool flange prevents the tool flange and the attached tool from crossing a safety plane. When you restrict the tool flange, the unrestricted area is the area inside of the safety plane, where the tool flange can operate normally.
 The tool flange cannot cross the restricted area, outside of the safety plane.
 Removing the restriction allows the tool flange to go beyond the safety plane, to the restricted area, while the attached tool remains inside of the safety plane.
 You can remove the tool flange restriction when working with a large tool off-set. This will allow extra distance for the tool to move.
 Restricting the tool flange requires the creation of a plane feature. The plane feature is used to set up a safety plane later in the safety settings.

Create Safety Plane Using a Frame

Description

You can use an existing frame to define a safety plane. This provides a more intuitive way to specify plane parameters from a predefined position and rotation (X, Y, Z, RX, RY, RZ).
 Select the frame in the **Copy Frame** drop-down menu.



The **Copy Frame** drop-down menu is found in the parameters of a safety plane definition on the **Planes** screen. This list includes frames defined on the **Frames** screen. When you select a frame, it is used as the safety plane definition.
 If you specify offset, tilt, and rotation parameters, they are applied to the selected frame (see "Configuring a Safety Plane"). The resulting safety plane combines all parameters.

See the "Frames" section in Software Handbook.

Creating Safety Planes using Frames

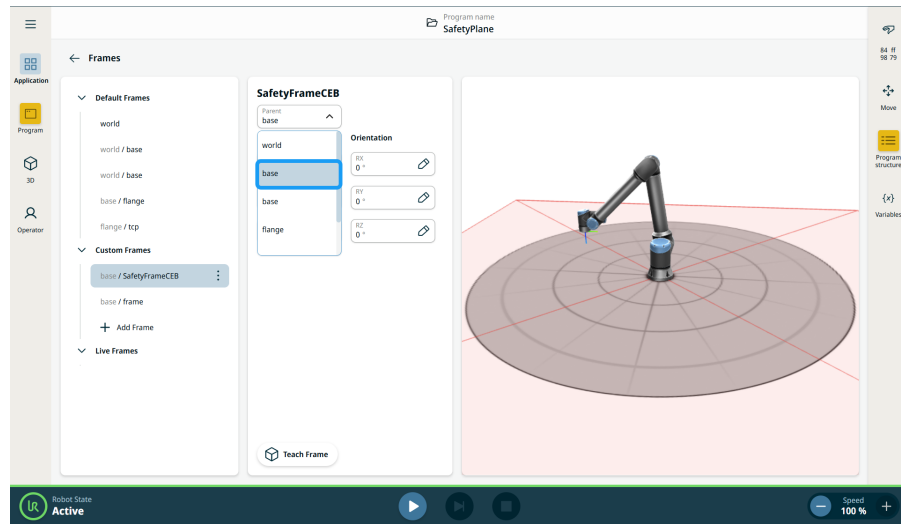
1. In the **Application** tab, select **Frames**.
2. Tap **+ Add Frame** under **Custom Frames**, then rename the frame using the **three vertical dots (kebab icon)**.



NOTICE

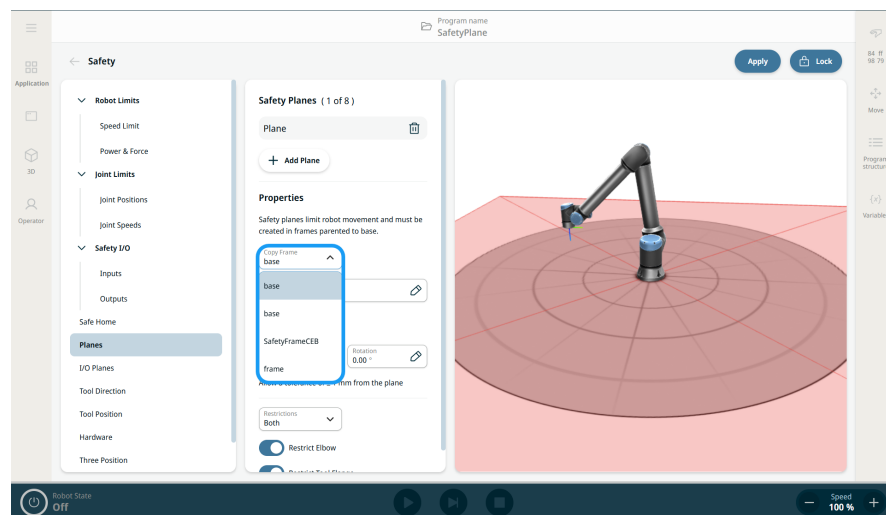
Frames are created relative to **Base**.

3. In the **Parent** field, select **Base**.



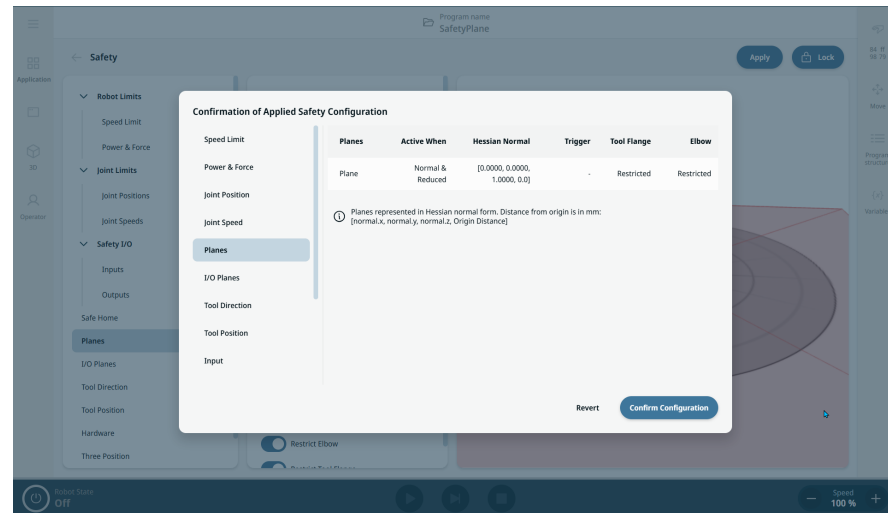
Selecting Reference Frame in Safety Plane

1. In **Application**, go to **Safety**, select **Planes**, and tap **Unlock**.
2. On the **Safety Planes** screen, select a frame in the **Copy Frame** drop-down menu. You can also apply offset, tilt, and rotation.



3. Tap the **Apply** button.

4. Configured planes are listed in the function assignment table on the **Confirmation of Applied Safety Configuration** overview dialog.



NOTICE

A frame used for a safety plane may be inverted from what you expect. If the robot is positioned on the violating side of the plane, add a tilt of 180 degrees.

For restrictions on plane configurations, elbow joint restriction, and tool flange restriction, see "Safety Planes".

9.8.8. I/O Planes

Description

The **I/O Planes** feature provides more flexible use of safety planes compared to the standard **Safety Planes** functionality. It lets you activate and deactivate safety planes using terminal inputs, and/or generate an output signal when the safety spheres cross a safety plane.



Input/Stop Plane

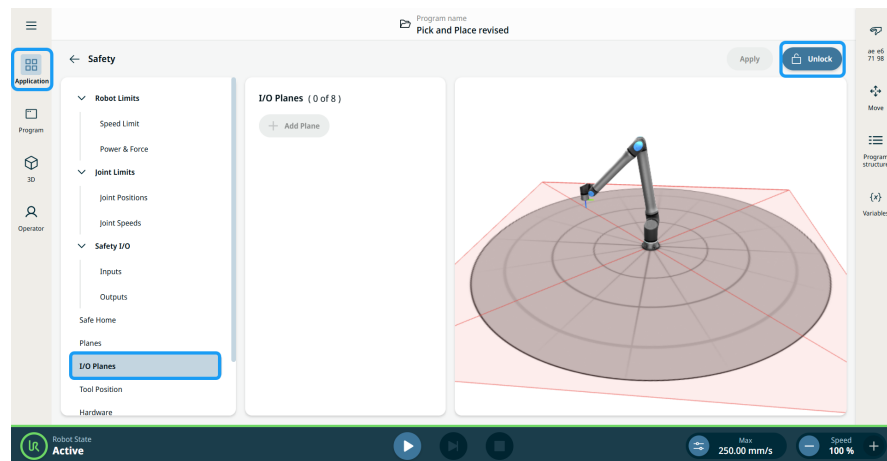
Trigger Stop triggers an I/O plane stop. It can be muted using a safety input. Unless muted, moving toward the plane causes the robot to stop and wait until the safety input mutes the plane.



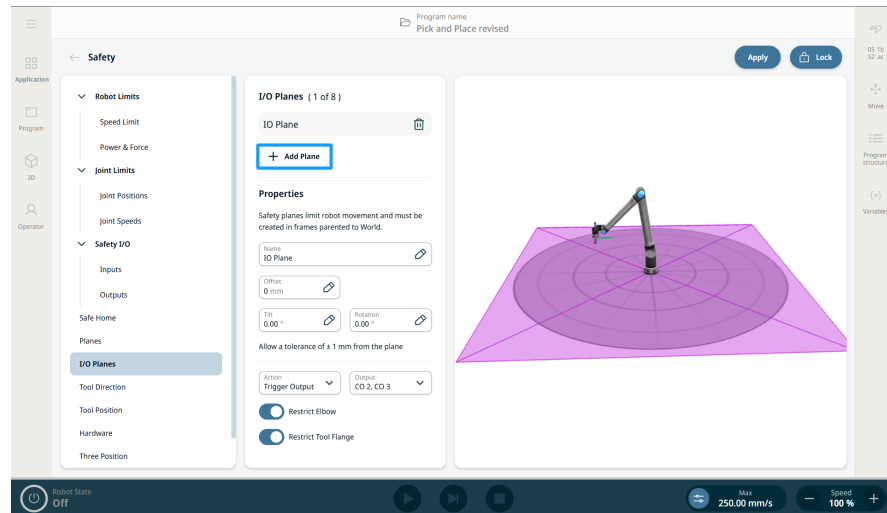
NOTICE

When the plane is inactive, the robot is allowed to move freely. When the plane is active and the robot has not crossed the plane, the robot will stop before crossing the plane. If the plane is activated, while the robot is moving on the edge of the plane, the robot can end up crossing the plane and will in worst case stop according to the configured distance and stopping time limits. When the plane is activated and the robot has crossed the plane, the robot will stop according to the configured stopping distance and stopping time limits.

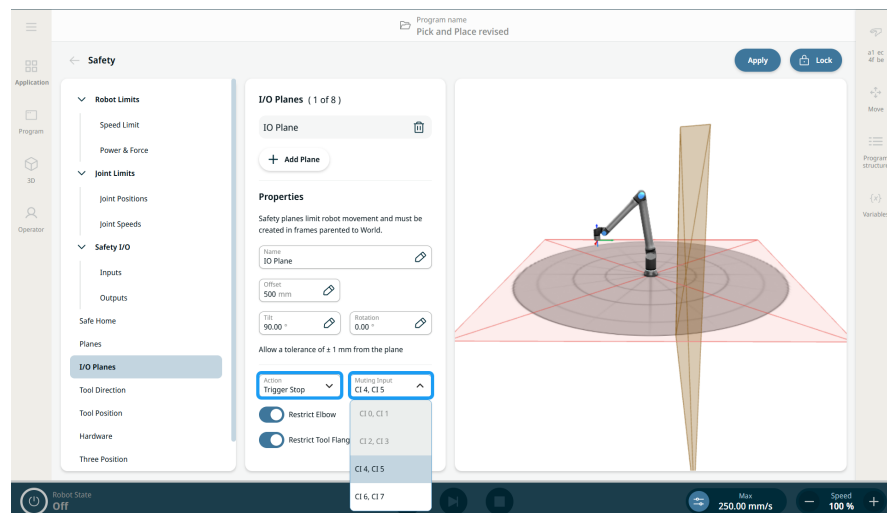
1. Go to [Safety](#) in the **Application Tab** and tap **I/O Planes**.



2. **Unlock** and tap **Add Plane**. Several fields and tabs of the I/O Planes properties appear.



3. Enter the value of **Offset** and **Tilt**.
4. In the **Action** field, select **Trigger Stop**.
5. In the **Muting Input** field, select **CI4** and **CI5**. Tap **Apply**.



6. When you toggle a physical switch connected to the input, the plane disappears from the 3D viewer. Tap the **Play** button, and the robot moves freely from one side of the plane to the other.
7. Toggle the switch again to unmute the plane, and the robot enters I/O stop.

Output Plane

Trigger Output generates a safety output based on whether one or more of the safety spheres have crossed the plane.

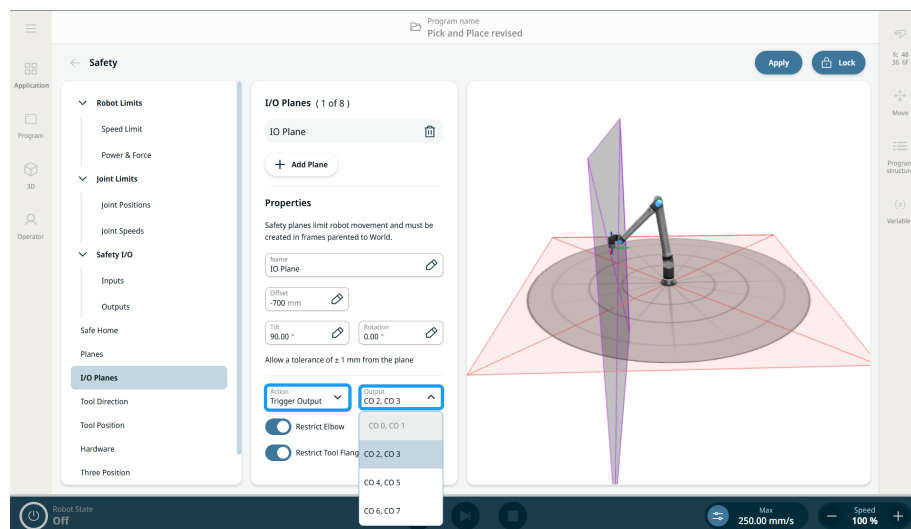
When one or more of the safety spheres have crossed a plane defined as trigger output, the safety output corresponding to that plane is activated. "Activated" means that the safety outputs are electrically set "Low".



NOTICE

When any of the safety spheres has crossed a plane defined as trigger output, the safety output corresponding to that plane is activated.

1. Enter the value of **Offset** and **Tilt**.
2. In the **Action** field, select **Trigger Output**.
3. In the **Output** field, select **CO2, CO3**. Tap **Apply**.



4. When you play the robot program, you can see the "LOW" and "HIGH" outputs in the [Communication](#) tab.

9.8.9. Tool Direction

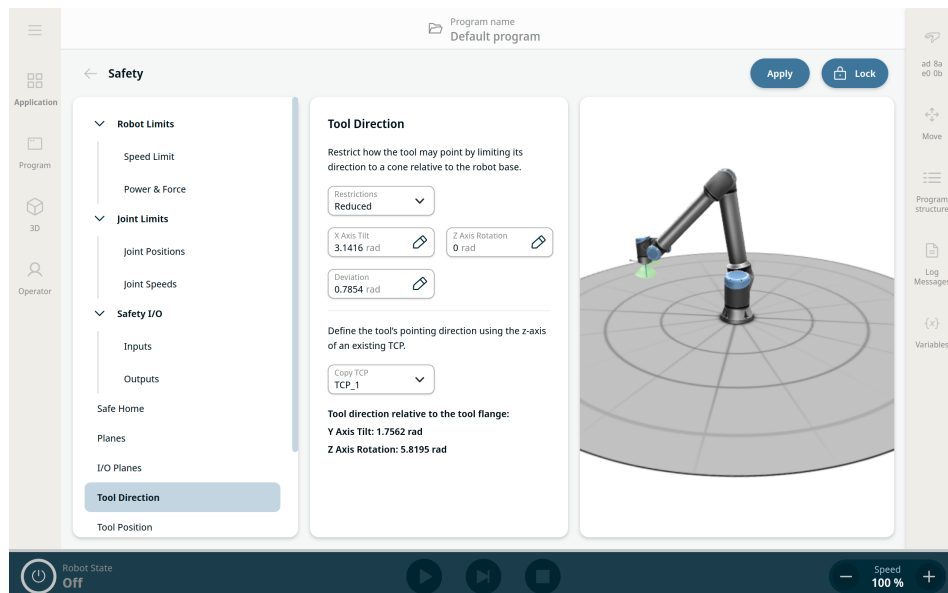
Description

The Tool Direction screen enables you to configure a safety feature that restricts the angle at which the tool is allowed to point. You can see the Tool Direction limit, which is added to the safety settings. The limit is defined by a cone that has a fixed orientation relative to the robot arm's base reference frame. As the robot arm moves around, the tool direction is restricted so it remains within the defined cone.



NOTICE

The restrictions can be disabled or configured to apply in Reduced mode, Normal mode, or both.



NOTICE

The Tool Direction feature enables you to increase the safety of an application when you use a tool that can cause harm or injury when facing in a particular direction.



Limit Properties

The Tool Direction limit has three configurable properties:

1. **Tilt:** You can define the cone's opening angle from the central (reference) direction.
2. **Rotation:** You can define how the cone rotates around its central axis.
3. **Deviation:** You can define the maximum angular offset that is allowed between the desired direction and any direction inside the cone.

Disabled Tool direction limit	Never active
Normal Tool direction limit	Active only when safety system is in Normal mode
Reduced Tool direction limit	Active only when the safety system is in Reduced mode
Both (Normal and Reduced Tool direction limit)	Active when the safety system is in Normal mode as well as when it is in Reduced mode .

The tool's pointing direction is specified using a TCP configured in the **End Effectors application** settings.

Tool Properties

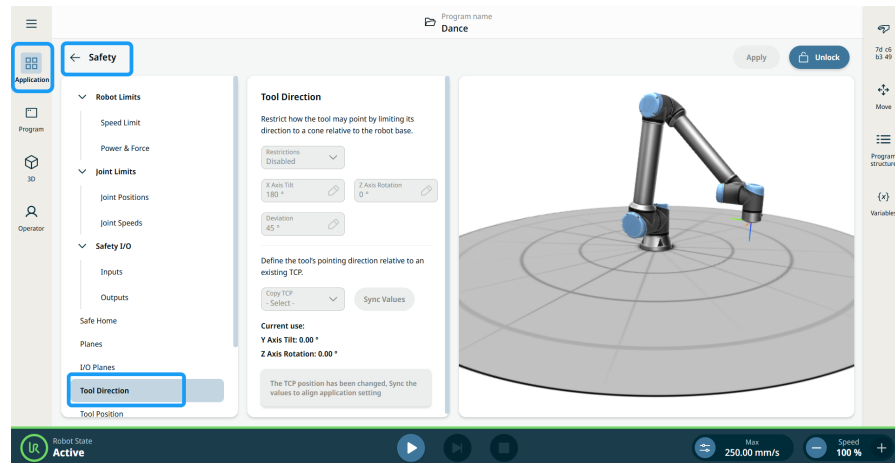
By default, the tool points in the same direction as the Z axis of the tool output flange. You can modify it by specifying two angles:

- **Tilt angle:** How much to tilt the Z axis of the output flange towards the X axis of the output flange.
- **Z-axis rotation:** How much to rotate the tilted Z axis around the original output flange Z axis.

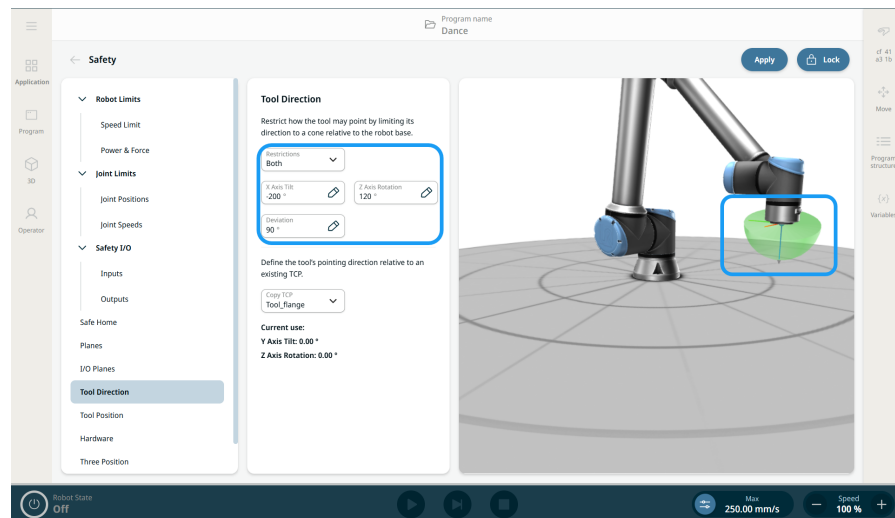
Alternatively, you can copy the Z axis of an existing TCP by selecting that TCP from the drop-down menu.

To access Tool Direction

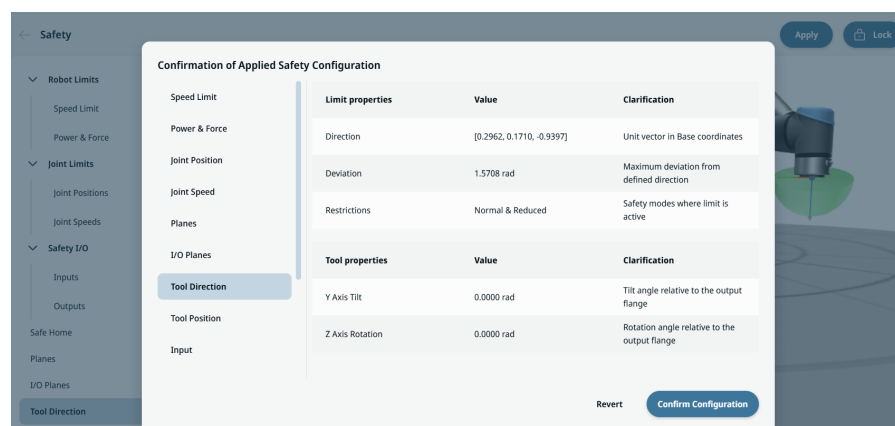
1. Tap the **Application Tab** and select **Safety Application**. In the left panel, select **Tool Direction**.



2. **Unlock** and enter the Safety password.
3. When you enable the **Restrictions** field, the other fields become available. A green cone appears in the TCP, showing the tool's pointing direction.



4. Tap the **Apply** button.
5. You can see **Tool Direction** in the function assignment table on the **Confirmation of Applied Safety Configuration** overview dialog.



9.8.10. Hardware

Description The **Hardware** section allows users to select the type of Teach Pendant (TP) connected to their robot. Available options are:

- **3PE Enabled.** 3-Position Enabling Teach Pendant (3PE TP)
- **None.** This allows the robot to be safely powered up without a Teach Pendant connected to the Control Box.



NOTICE

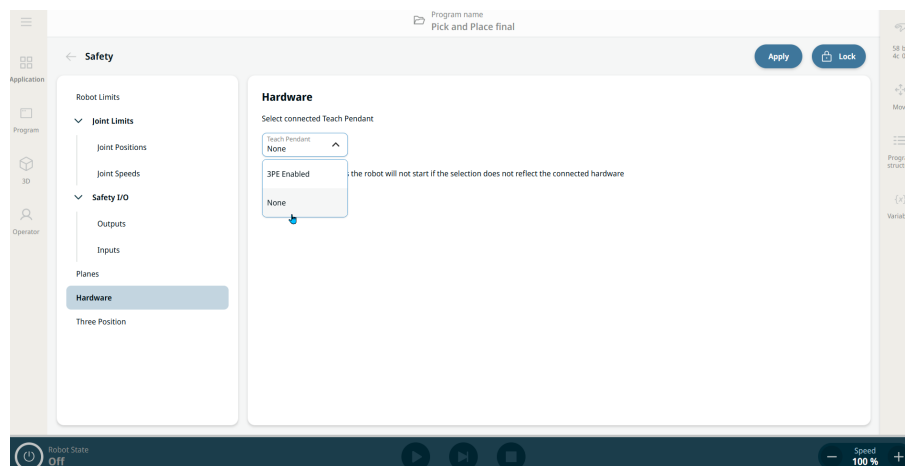
For safety reasons, the robot will not start if the selection does not reflect the connected hardware.

Purpose of None option The **None** option supports setups where a Teach Pendant is not required or physically present, enabling more streamlined and flexible deployments, especially in automated or remote-controlled environments. It gives users three key benefits:

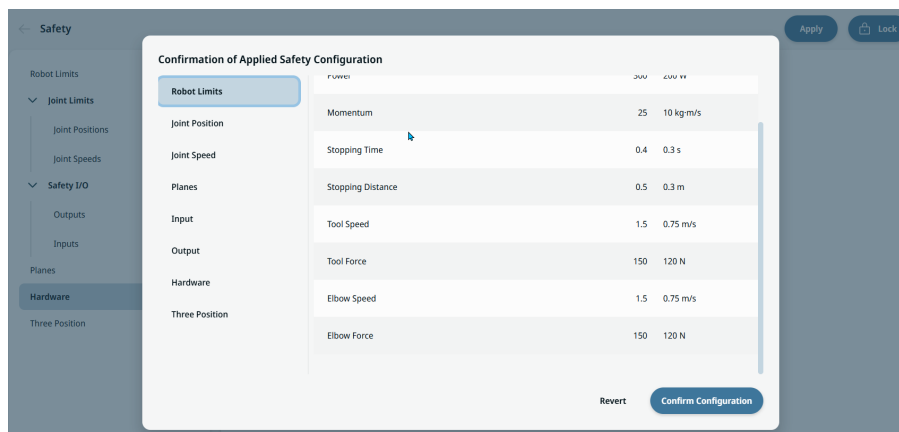
- Supports headless configurations where no Teach Pendant is used.
- Simplifies hardware requirements for minimal setups.
- Improves start-up flexibility by removing dependency on Teach Pendant presence.

To access None

1. Tap **Hardware** on the left panel of the Safety application screen.
2. Unlock the main screen in the topmost right part, enter your safety password, and **Confirm**.
3. On the central panel, tap the **Teach Pendant** field, and choose **None**.



4. Tap **Apply** beside the Lock button in the main screen.
5. Tap **Apply and restart** in the pop-up of Apply Safety Configuration.
6. A Confirmation of Applied Safety Configuration pop-up screen appears, and you can either **Revert** or **Confirm Configuration**.



9.8.11. Three Position

Description Manual high speed allows both tool speed and elbow speed to temporarily exceed 250 mm/s. It is only available when your robot is on manual mode and a three-position enabling device is configured. If there is no interaction with the three-position enabling device in five minutes, increased values will be reset to 250 mm/s.

- To access Three Position**
1. On the left panel, tap **Three Position**.
 2. Tap **Unlock** on the upper right side of the main screen.
 3. Enter the safety password and **Confirm**.
 4. On the central panel, slide on the button to **Allow manual high speed**.
 5. Tap **Apply** on the upper right side of the main screen.



10. Integration Guidelines

Description	What's in this chapter
	• Communication Protocols Installation • API Documentation • Custom Scripting and Automation

10.1. Communication Protocols Installation

Description	Installing the robot can require the configuration and use of input and output signals (I/Os). These different types of I/Os and their uses are described in the following sections.
-------------	--

10.1.1. Ethernet

- Description**
- The Ethernet interface can be used for:
 - MODBUS, EtherNet/IP and PROFINET.
 - Remote access and control.

To connect the Ethernet cable by passing it through the hole at the base of the Control Box, and plugging it into the Ethernet port on the underside of the bracket.

Replace the cap at the base of the Control Box with an appropriate cable gland to connect the cable to the Ethernet port.

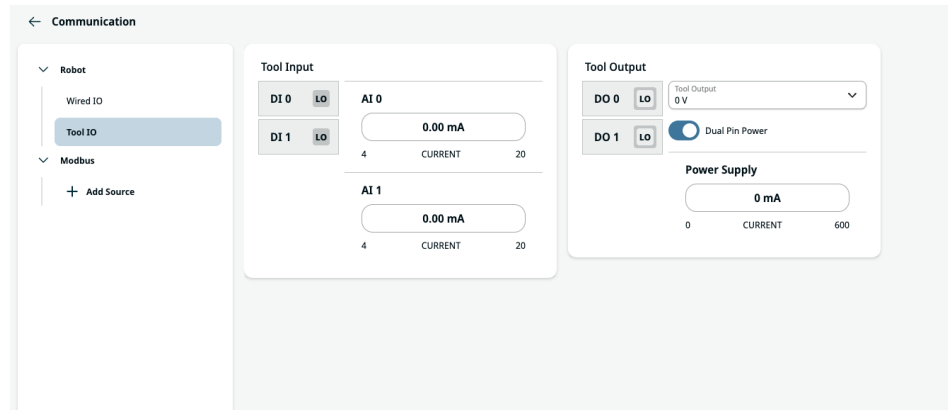


The electrical specifications are shown in the table below.

Parameter	Min	Typ	Max	Unit
Communication speed	10	-	1000	Mb/s

10.1.2. Input and Output

Description The tool IO tab monitors and sets the live IO signals from/to the Control Box.



Tool Output The tool output contains two digital output fields, tool output voltage and power supply current indicator and Dual Pin Power toggle.

- **Digital Output (DO)** - can be independently set to high or low
- **Tool Output Voltage** - selectable 0V, 12V and 24V. This setting is persistent over restarts of the robot controller
- **Power Supply** - current consumption indicator
- **Dual Pin Power** - used to toggle between digital outputs and source of power for the tool. Enabling Dual Pin Power disables the default tool digital outputs (DO)

After selecting a new output configuration, the changes take effect. The currently loaded installation is modified to reflect the new configuration. After verifying the tool outputs are working as intended, make sure to save the installation to prevent losing changes.

Tool Input The tool input contains two digital input (DI) fields and two fields for analog input (AI) currents.

DI fields can be independently configured to action presets:

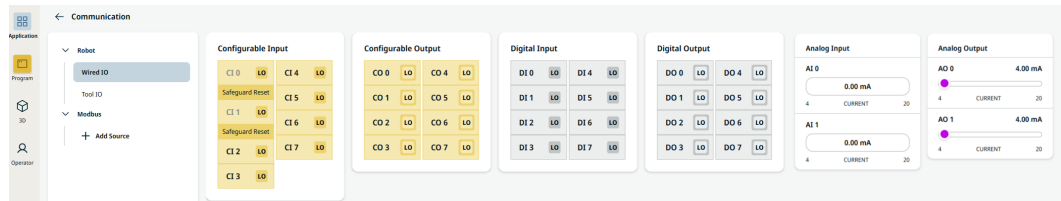
- Start Program
- Stop Program
- Pause Program
- Freedrive

The AI's can be set from 4-20mA. These settings are persistent over restarts of the robot controller and saved in the installation.

Using the Wired I/O Tab

Description Use the Wired I/O Tab screen to monitor and set the live I/O signals from/to the Control Box.

The screen displays the current state of the I/O, including during program execution. The program stops if anything is changed during execution. At program stop, all output signals retain their states. The screen updates at 10Hz, so a very fast signal might not display properly.



Configurable I/Os Configurable I/Os can be reserved for special safety settings defined in the I/O Setup. Under those which are reserved will have the name of the safety function in place of the default or user defined name.

Configurable inputs that are reserved for safety settings are not toggable and will be displayed as LED's only.

For unreserved I/Os, it has the following options:

- Start Program
- Stop Program
- Pause Program
- Freedrive

Digital I/Os DI have the following options:

- Start Program
- Stop Program
- Pause Program
- Freedrive

All DIs are preset to Low.

All DOs are set independently to either high or low.

Analog IOs The analog I/O's can be set to either current [4-20mA] or voltage [0-10V] output. These settings are persistent over restarts of the robot controller and saved in the installation.



Drive Power Indicator

Description The drive power indicator is a light that turns on when the robot arm is powered on, or when there is power to the robot cable. When the robot arm is powered off, the drive power indicator turns off.

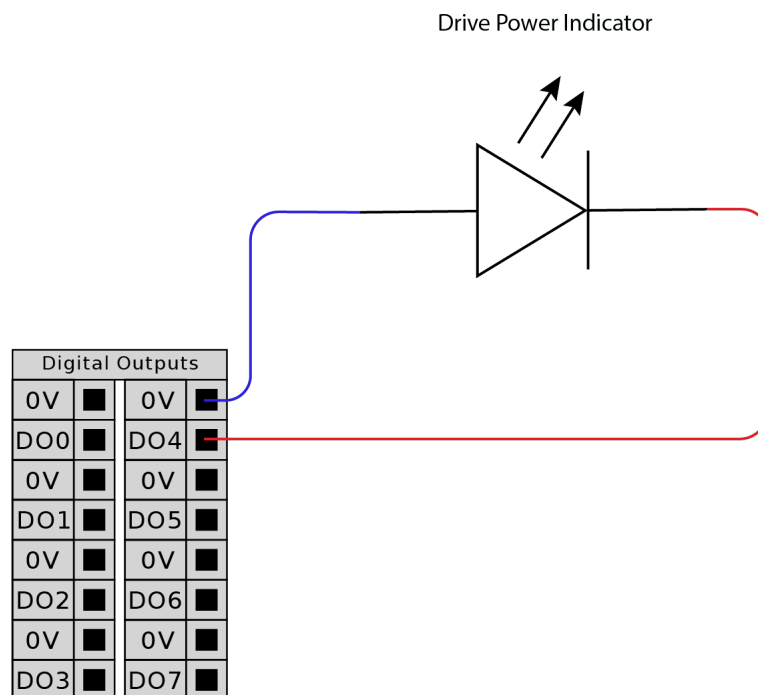
The drive power indicator is connected via the Digital Outputs. It is not a safety feature and does not use safety I/Os.

Indicator The drive power indicator can be a light that can work at 24VDC.

To set up the indicator

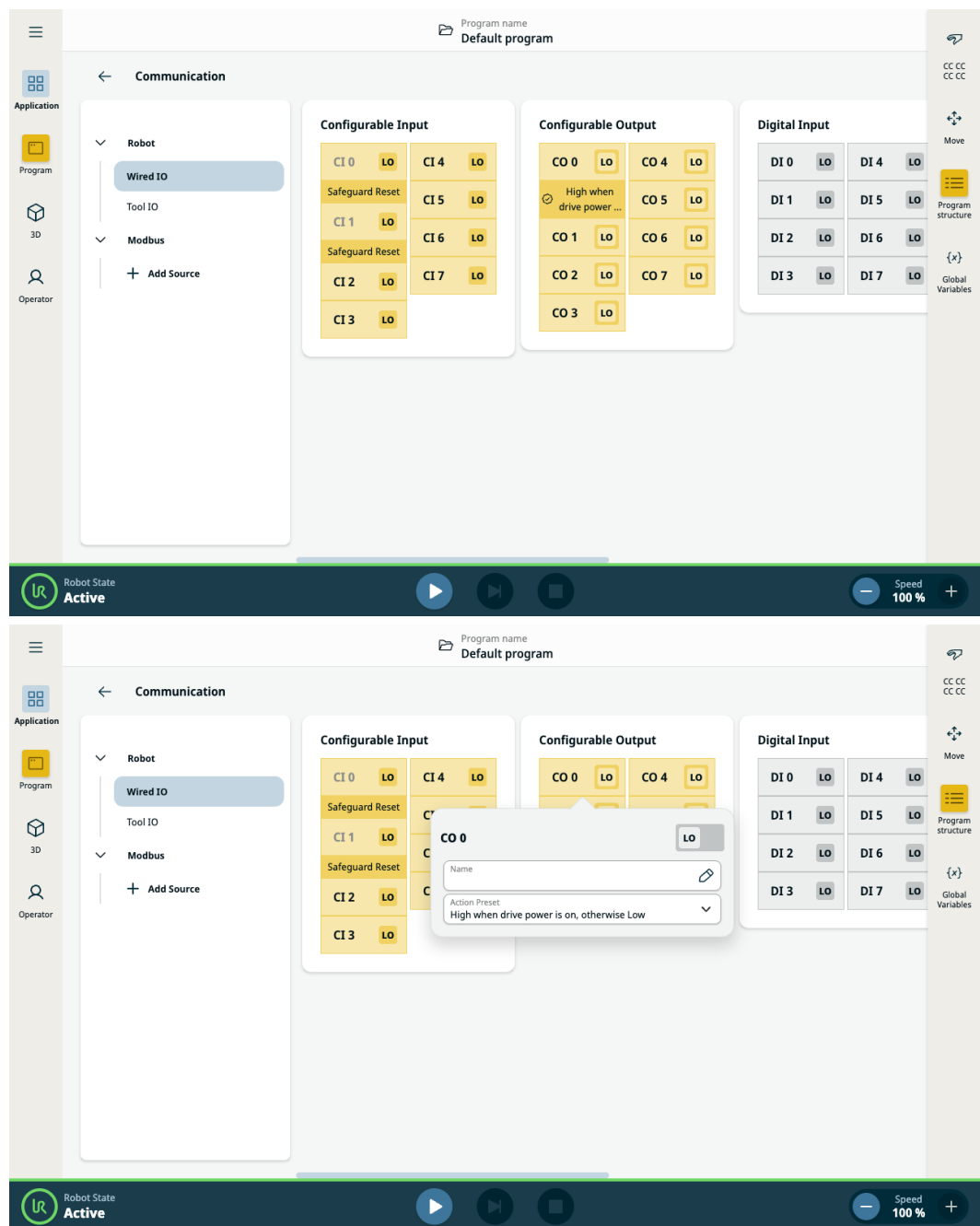
Setting up the indicator requires a light and wiring for the outputs.

1. Connect your drive power indicator to the Digital Outputs as shown in the image below.
2. Verify the drive power indicator is correctly connected.
 - You can power on the robot arm and verify the light turns on.
 - You can power off the robot arm and verify the light turns off.



To configure the indicator

1. In the Navigation menu, tap **Application**.
2. Select **Communication**.
3. In the side menu, select **Wired IO**.
4. Scroll to your desired output type and tap to select one of the following:
 - Configurable Output
 - Digital Output
 - Analog Output
5. Select **Action Preset**
You can name the selected output
6. In the dropdown select **High when drive power is on, otherwise Low**.



Remote Mode in Safety Overview

Description When activated, Remote Mode allows external devices to connect to key services such as the Primary Interface.

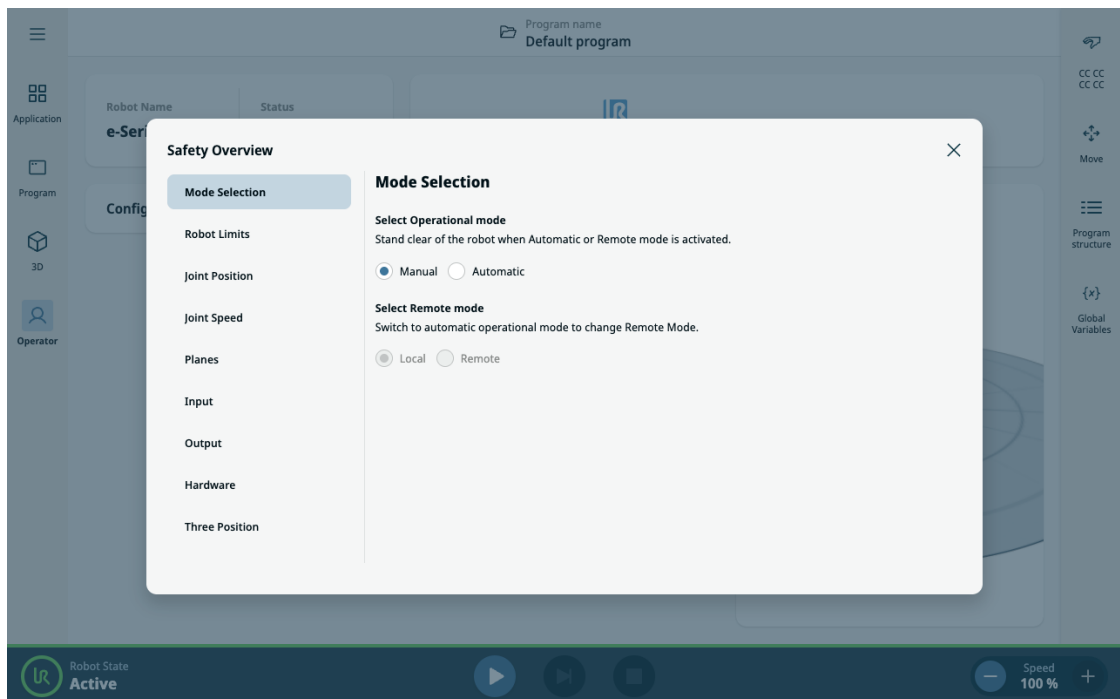
Remote Mode can be toggled via a dedicated switch in the Safety Overview dialog.

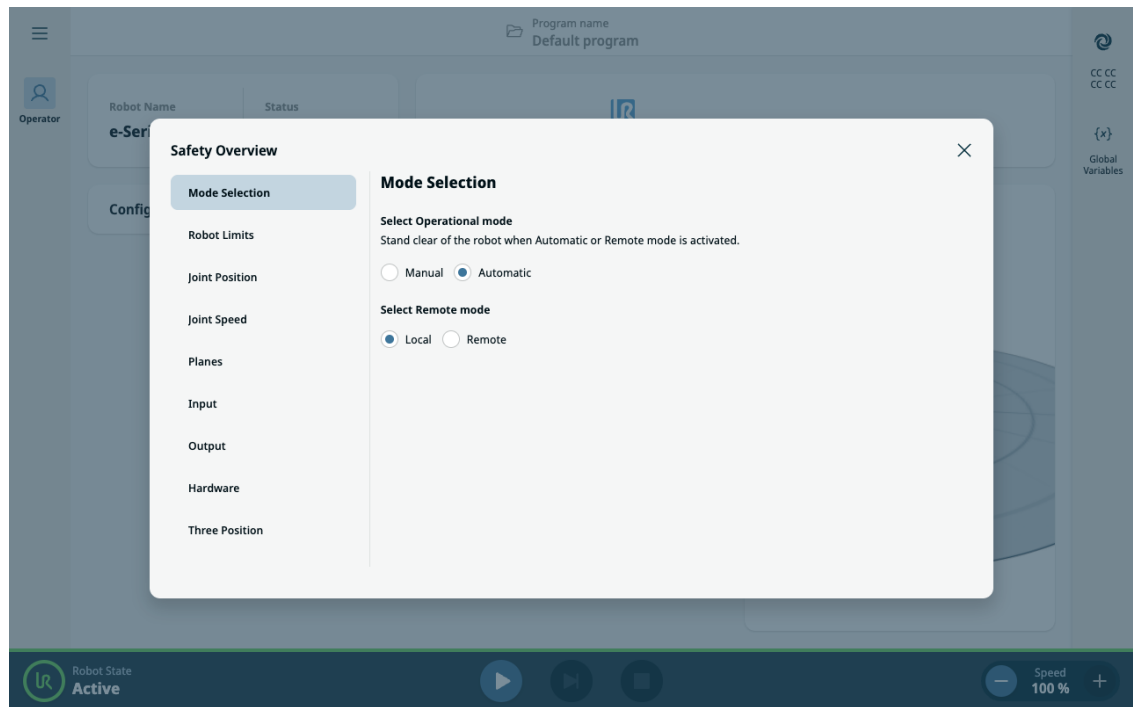
1. Go to the safety overview in the main screen.
2. Click Mode Selection.
3. You can now select Automatic and then Remote.

"Local" is toggled as a default.

"Remote" is only active when the application is in Automatic mode.

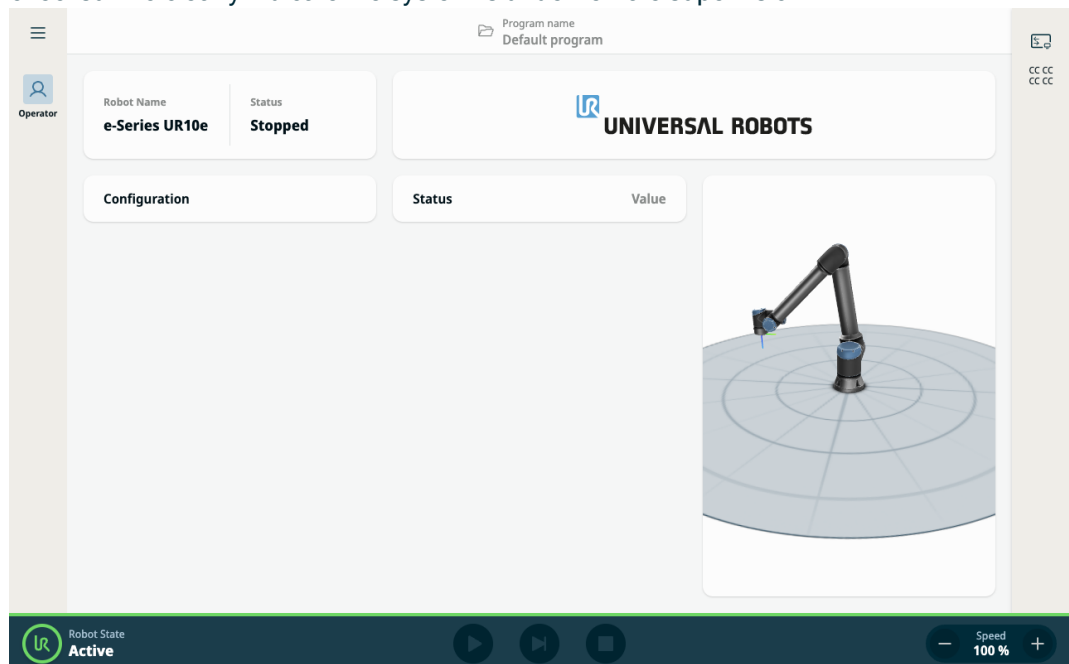
Toggle Access





Secure Lockdown

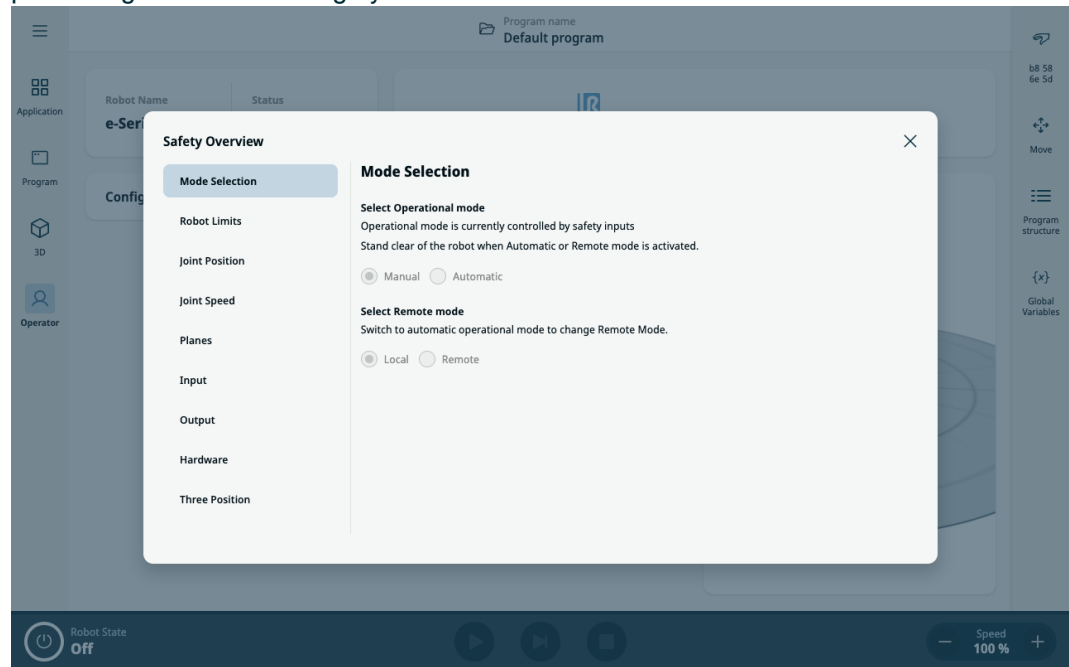
While in Remote Mode, the PolyScope X interface enters a secure, read-only state. All editing and control actions are disabled, and only the Operator Screen remains accessible in view-only mode. Additionally, a Remote Mode icon is displayed above the Safety checksum to clearly indicate the system is under remote supervision.





I/O Controlled Safety

If the robot's operational mode is governed by an I/O signal, switching to Manual mode via I/O will automatically revert Remote Mode back to Local Mode. This feature ensures a safe and structured environment for remote monitoring, while preserving local control integrity when needed.



10.1.3. PROFIsafe

Description

The PROFIsafe network protocol (implemented as version 2.6.1) allows the robot to communicate with a safety PLC according to ISO 13849, Cat 3 PLd requirements. The robot transmits safety state information to a safety PLC and receives commands to reduce safety limits or trigger a safety-related function such as an emergency stop. The PROFIsafe interface provides a safe, network-based alternative to connecting wires to the safety IO pins of the robot control box. PROFIsafe is only available on robots that have an enabling license, which you can obtain by contacting your local sales representative. After you obtain the license, download it from [myUR](#). For information about robot registration and license activation, see [3.7 Robot Registration and License File on page 20](#)

Advanced Options A control message that the safety PLC sends to the robot contains the information shown in the following table.

Signal	Description
E-Stop by system	0: Asserts the system e-stop. 1: Clear system e-stop.
Safeguard stop	0: Asserts the safeguard stop. 1: Normal operation state. Note: Also refer to the "Reset safeguard stop" signal description.
Reset safeguard stop	Resets the safeguard stop state on a 0-to-1 transition when the "safeguard stop" signal is already set to 1.
Safeguard stop auto	0: Asserts safeguard stop if the robot is operating in Automatic mode. 1: Normal operation state. Safeguard stop auto shall only be used when a 3-Position Enabling (3PE) Device is configured. If no 3PE Device is configured, the safeguard stop auto acts as a normal safeguard stop input. Note: Also refer to the "Reset safeguard stop auto" signal description.
Reset safeguard stop auto	Resets the safeguard stop auto state on a 0-to-1 transition when the "safeguard stop auto" signal is already set to 1.
Reduced	0: Activates the Reduced safety limits. 1: Activates the "Normal mode" safety limits. The safety system guarantees the robot is within reduced limits less than 0.5s after the input is activated. If the robot arm continues to violate any of the reduced limits, a Stop Category 0 is triggered.
Operational mode	0: Activates the manual operational mode. 1: Activates the automatic operational mode. If the safety configuration "Operational mode selection via PROFIsafe" is disabled, this field shall be omitted from the PROFIsafe control message.



Advanced Options A status message that the robot sends to the safety PLC contains the information shown in the following table.

Signal	Description
Stop, cat. 0	0: Robot is performing, or has completed, a safety stop of category 0; a hard stop by immediate removal of power to the arm and the motors. 1: Normal operation state.
Stop, cat. 1	0: Robot is performing, or has completed, a safety stop of category 1; a controlled stop after which the motors are left in a power-off state with brakes engaged. 1: Normal operation state.
Stop, cat. 2	0: Robot is performing, or has completed, a safety stop of category 2; a controlled stop after which the motors are left in a power-on state. 1: Normal operation state.
Violation	0: Robot is stopped because the safety system has failed to comply with the active safety limits defined. 1: Normal operation state.
Fault	0: Robot is stopped because of an unexpected exceptional error in the safety system. 1: Robot is not experiencing an unexpected exceptional error in the safety system.
E-stop by system	0: Robot is stopped because of one of the following conditions: • A safety PLC connected via PROFIsafe has asserted a system level e-stop. • An IMMI module connected to the control box has asserted a system level e-stop. • A unit connected to the system e-stop configurable safety input of the control box has asserted a system level e-stop. 1: Robot is not in system e-stop.
E-stop by robot	0: The robot is stopped because of one of the following conditions: • The e-stop button of the teach pendant is pressed. • An e-stop button connected to the robot e-stop non-configurable safety input of the control box is pressed. 1: Robot is not in e-stop by robot.

Advanced Options

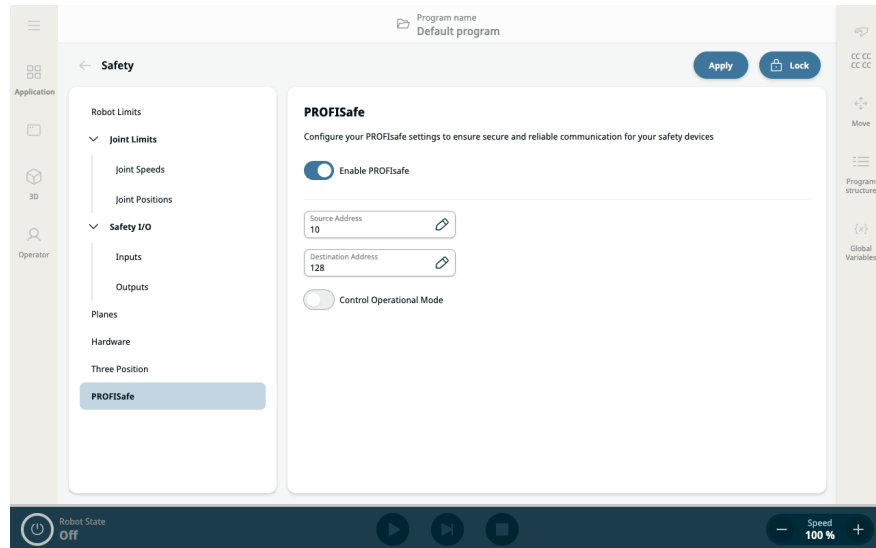
Signal	Description
Active limit set	The active set of safety limits. 0: Normal 1: Reduced 2: Recovery
Robot moving	0: Robot is moving. If any joint moves at a velocity of 0.02 rad/s or higher, the robot is considered in motion. 1: Robot is at standstill.
Safe home position	0: Robot is at rest (robot not moving), and in the position defined as the Safe Home Position. 1: Robot is not at rest, or not in the position defined as the Safe Home Position.

**Advanced Options**

Signal	Description
Safeguard stop	0: The robot is stopped due to one of the following conditions: • A safety PLC connected via PROFIsafe has asserted the safeguard stop. • A unit connected to the safeguard stop nonconfigurable input of the control box has asserted the safeguard stop. • A unit connected to the safeguard stop configurable safety input of the control box has asserted the safeguard stop. 1: The robot is not stopped due to a safeguard stop. Note: Also refer to the “Reset safeguard stop” signal description. PROFIsafe enforces the use of the safeguard reset functionality.
Safeguard stop auto	0: The robot is stopped because it is operating in Automatic mode and one of the following conditions applies: • A safety PLC connected via PROFIsafe has asserted safeguard stop auto. • A unit connected to a safeguard stop auto configurable safety input of the control box has asserted safeguard stop auto. 1: The robot is not stopped due to safeguard stop auto. Note: Also refer to the “Reset safeguard stop auto” signal description. PROFIsafe enforces the use of the safeguard reset functionality.
3PE stop	0: The robot is stopped because it is operating in Manual mode and one of the following conditions applies: • Any 3PE is pressed to the middle position, and Freedrive input is active. • Not all 3PE devices are pressed to the middle position. 1: Robot is not stopped because of a 3-position enabling device.
Operational mode	Indication of the active operational mode of the robot. 0: Disabled 1: Automatic 2: Manual
Reduced	0: Reduced safety limits are active. 1: Normal safety limits are active.

Configuring PROFIsafe

1. On the Safety application screen, tap **PROFIsafe** on the left panel.
2. Tap **Unlock** on the top right main screen to enable PROFIsafe. Enter the safety password and **Confirm**.



The right panel shows two fields and two buttons for configuring PROFIsafe:

- Enable PROFIsafe button
 - Source Address field
 - Destination Address field
 - Control Operational Mode
3. Slide right the **Enable PROFIsafe** button.
 4. Tap the **Source Address** and **Destination Address** fields to specify the addresses to be used by the robot and the safety PLC in identifying each other.
 5. By tapping **Control Operational Mode**, you have the option to enable the PROFIsafe PLC to control the robot operational mode.

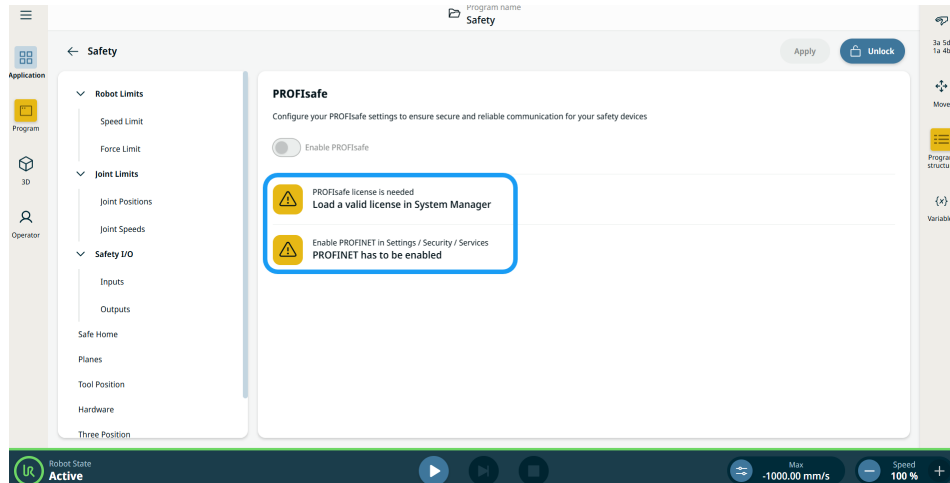


NOTICE

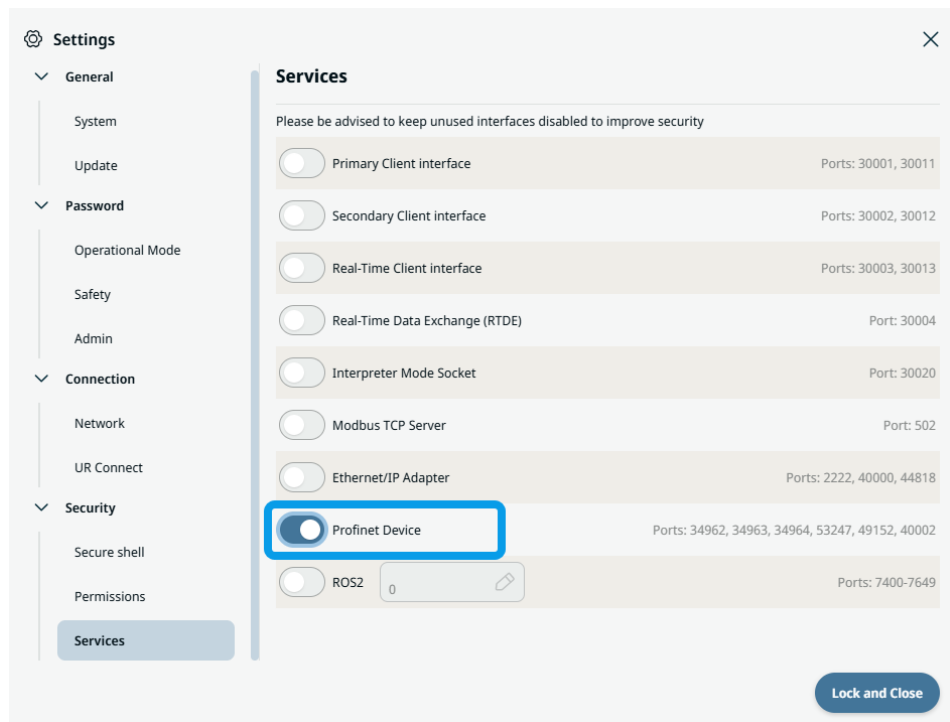
In order to configure and use PROFINET, **PROFINET Device** must be enabled in the security services settings menu.



Enable PROFINET in Settings / Security / Services
PROFINET has to be enabled



Read for details and interface location [PROFINET Device](#).



10.2. API Documentation

Description You reference the API documentation to integrate PolyScope X with external systems through its API architecture, which lets you exchange real-time data, build URCaps, and connect AI modules, sensors, and controllers without OS conflicts.

10.2.1. Robot-API

Description The Robot-API provides a RESTful interface for remotely operating the robot. It follows a predictable, resource-oriented URL structure, accepts form-encoded request bodies, returns JSON-formatted responses, and adheres to standard HTTP response codes for clear and consistent communication.

Network To interact with the Robot-API, the robot must be connected to a network. The API documentation is always accessible directly from the robot itself and includes comprehensive guidance on how to utilize the available endpoints. For more details on network setup, please see [Network](#) in the PolyScope X Software Handbook.

API	Environment	API Base URL	Documentation URL
	Robot	http://[ip]/universal-robots/robot-api	http://[ip]/universal-robots/robot-api/docs
	UR Simulator	http://localhost/universal-robots/robot-api	http://localhost/universal-robots/robot-api/docs

10.3. Custom Scripting and Automation

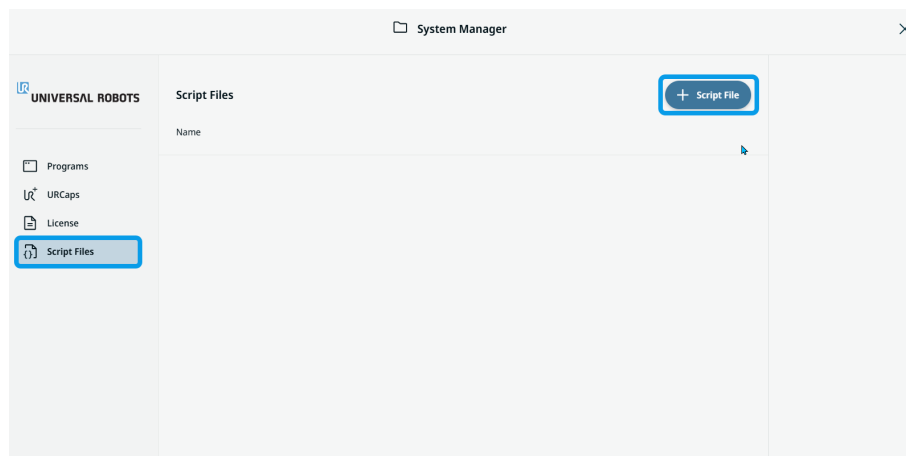
Description You use URScript through Script nodes or runtime interfaces like Interpreter Mode, background programs, and remote procedure calls to add complex flow control, custom force and motion functions, and communication protocols beyond graphical nodes. For advanced integrations and external equipment control, you combine it with PolyScope nodes.



10.3.1. Script Files

Description Script files are used in the program through the Script Node, imported from a disk into PolyScope X. The script files are managed via the System Manager

- To access Script Files**
1. Navigate to the **System Manager** by tapping either the System Manager or header.
 2. Tap **Script Files** in the left column, and add your script file + **Script File**. The file on disk should have the **.script** extension.



Script files are kept according to their file name. So reimporting the script file again from disk will overwrite the old one. Once imported, the script file can be deleted or viewed.

10.3.2. URScript Control API in Logic Program

Description The URScript Control API lets you control the robot arm's state and program lifecycle. You can power the robot arm on and off, release the brakes, start, pause, or stop a robot program, and check the current robot state. Control API methods are available in the Logic Program, the main program, and in threads. The primary use case is calling Control API methods from the Logic Program to control the robot program. For example, you can set up an automatic startup routine that powers on the robot, starts the main program, monitors its state, and recovers from errors.

- Key features**
- **Robot power control:** Power the robot arm on and off from the Logic Program.
 - **Brake control:** Apply power to joint motors and release motor brakes to prepare for program execution.
 - **Program lifecycle control:** Start, pause, stop, and resume the main robot program. Lets you implement custom input and output actions on interface signals (Modbus signals or general-purpose registers).
 - **State queries:** Check remote control mode, arm power, motor power, safety state, and main program state.
 - **Error reporting:** All methods return error codes so the Logic Program can handle failures efficiently.

Most Control API methods require remote control mode, especially those that control the robot arm (power, brakes, and program control). State query methods work in both local and remote mode.

These methods return -1 when the robot is in local mode.

10.3.3. URCaps

Description You can manage your existing **URCaps** or install a new one in your robot.

- To manage URCaps**
1. In the Header, press the Hamburger menu and select **Settings**.
 2. Under System Manager, select **URCaps**.
 3. Tap the **+** button, select the **.urcap** file and press **Open**.
 4. If you wish to proceed with the installation of that URCap, press **Restart**. After that step, the URCaps is installed and ready to be used.
 5. To remove an installed URCaps, select it from Active URCaps, press the **-** button and press **Restart** so changes can take effect.

Active URCaps Details about the new URCap appear in the **Active URCaps** field. A status icon indicates the state of the URCap, as listed below:

-  **URCap ok:** The URCap is installed and running normally.
-  **URCap fault:** The URCap is installed but unable to start. Contact the URCap developer.
-  **URCap restart needed:** The URCap has just been installed and a restart is required.

Example Error messages and information about the URCap appear in the **URCaps Information** field. Different error messages appear depending on the type of error/s detected.

11. Troubleshooting

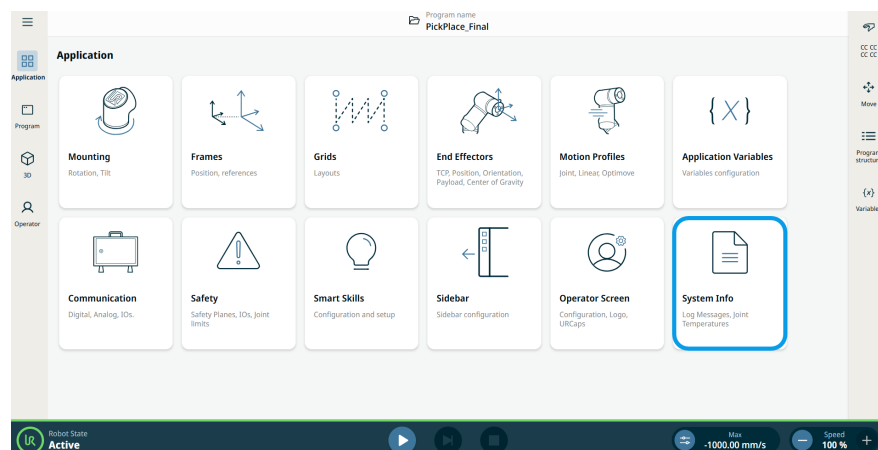
Description	What's in this chapter • System Info • Safety Checksum
-------------	--

11.1. System Info

Description	The System Info screen lists all log messages from the system, enables exporting support files, and checks if the robot joint temperature and current reach an unexpected level.
-------------	---

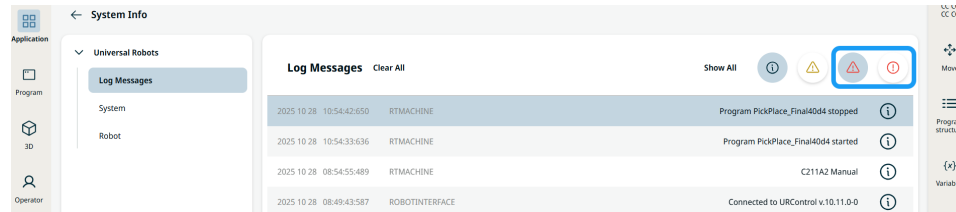
Using the System Info functionality

1. Go to the Application menu screen. See [Application Tab](#).
2. Tap the **System Info** icon.

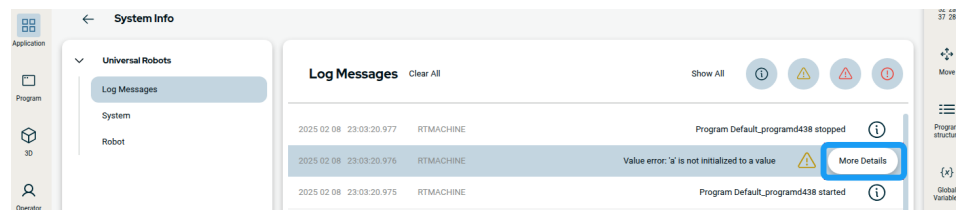


Log Messages

The **Log Messages** page lists all log messages from the system. The page has four buttons at the top of the messages list. The two red buttons allow for filtering on messages for faults and violations independently. The rightmost filter button shows/hides violations, whereas the filter button next to it shows/hides faults.



For some log messages, it is possible to view additional information by pressing the **More Details** button that appears when the message is selected in the list.

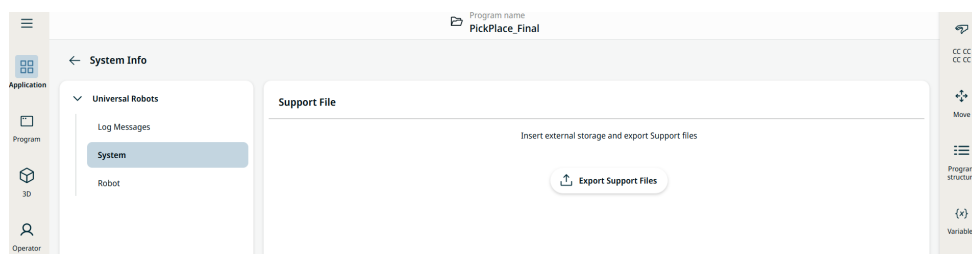


A dialog box displays with more details.



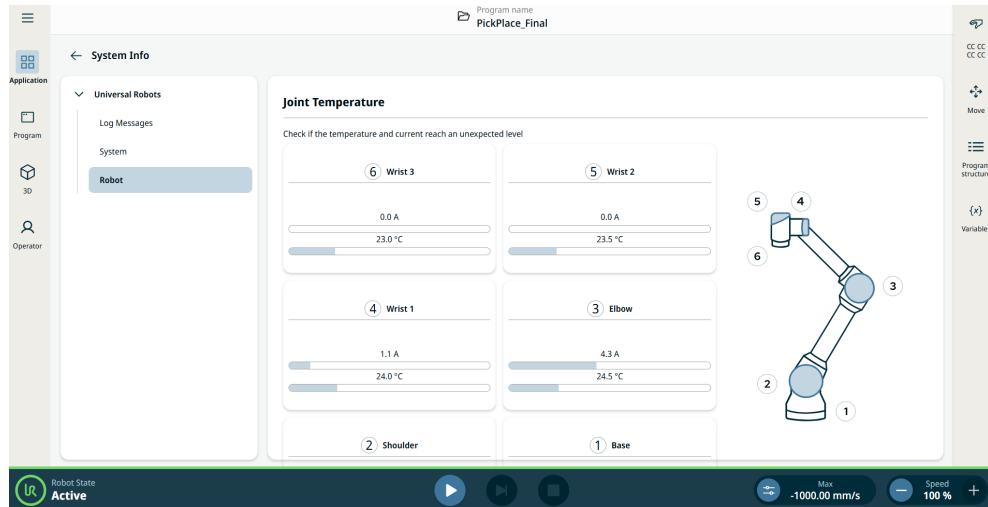
System

The **System** page enables you to insert external storage and export support files.



Robot

The robot **Joint Temperature** checks if the robot joint temperature and current reach an unexpected level.



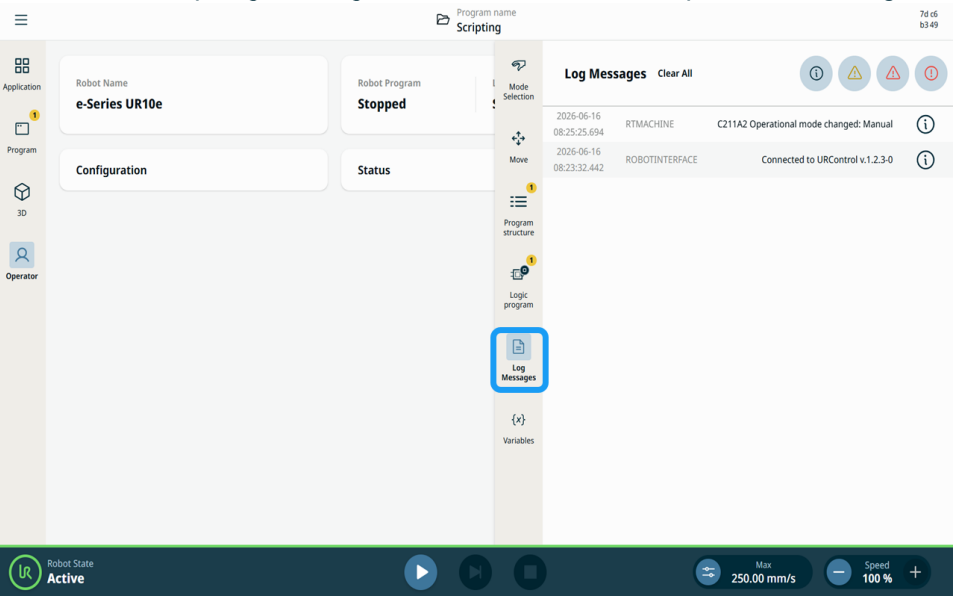
11.1.1. Logic Program Error Messages

Description

Logic Program error messages appear in the same log as robot program messages. The source column shows **Logic Program**.

Viewing Log Messages

In the sidebar, tap **Log Messages**. The multitask screen expands to show log messages.



Robot program and Logic Program log messages appear interleaved in a single list.

Log Messages Clear All			
2026-04-17 12:45:12.404	JOINT_1	*_com_ur_controller_runtime_exceptions.Syntax error on line 12_*	
2026-04-17 12:44:59.661	RTMACHINE	Syntax error on line 545: shared = True	
2026-04-17 12:44:39.099	JOINT_1	Program Logic_Program started	
2026-04-17 12:44:39.099	JOINT_1	Type error: Unsupported operand type(s) for +: Const String and Int	
2026-04-17 12:44:39.099	JOINT_1	Program Logic_Program stopped	
2026-04-17 12:44:24.140	RTMACHINE	Program Default_programd438 stopped	
2026-04-17 12:44:24.138	RTMACHINE	Type error: Unsupported operand type(s) for +: Const String and Int	
2026-04-17 12:44:24.137	RTMACHINE	Program Default_programd438 started	
2026-04-17 12:44:00.848	JOINT_1	Program Logic_Program stopped	
2026-04-17 12:43:57.235	RTMACHINE	Program Default_programd438 stopped	
2026-04-17 12:43:54.018	JOINT_1	Program Logic_Program resumed	
2026-04-17 12:43:51.606	RTMACHINE	Program Default_programd438 resumed	
2026-04-17 12:43:49.597	JOINT_1	Program Logic_Program paused	
2026-04-17 12:43:46.483	RTMACHINE	Program Default_programd438 paused	
2026-04-17 12:43:44.574	JOINT_1	Program Logic_Program started	

11.2. Safety Checksum

Description The Safety Checksum icon displays your applied robot safety configuration. The checksum changes if and only if the safety configuration is changed. The icon has three states/appearances:

-  A **hand** indicates that the control station is in control and the operational mode is set to manual.
-  A **whirling process** indicates that the control station is in control and that the operational mode is set to automatic.
-  A **remotely connected system** indicates that control is external to the station.

Safety Checksum is the only menu with non-labeled icon. The Safety Overview screen is divided in two:

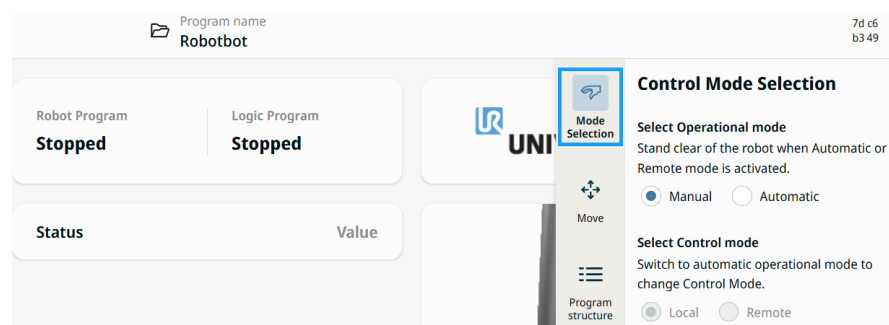
- Left side is smaller and contains the following safety item settings:

Mode Selection	Robot Limits	Joint Position	Joint Speed
Planes	Tool Position	Tool Direction	I/O
Hardware	Safe Home	Three Position	

- Right side contains a tabular data of each safety item setting.

To access Safety Overview

1. Tap the **Safety Checksum** icon.
2. Tap **Mode Selection** on the left side.
3. In **Select Operational mode**, tick either **Manual** or **Automatic**.



12. Maintenance

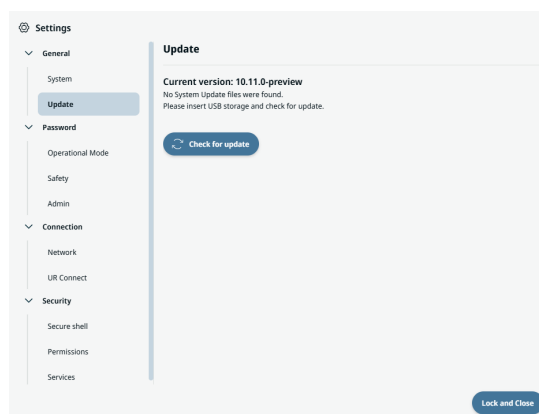
Description For the proper maintenance of PolyScope X, always upgrade to the latest software version.

12.1. Update

Description When you receive your robot, it is recommended that you update the software to the most recent release from Universal Robots. Please use this link to find the latest software update: [Download Latest Software](#)

Update SW This is how you update the software:

1. Tap the hamburger menu in the main navigation.
2. Go to Update in the general settings.
3. Insert the USB with the software update.
4. Tap the button "Check for update".



Software Name: PolyScope X
Software Version: 10.14
Document Version: 10.19.429