User Manual

Contents

Chapter 1. Introduction

Welcome to ForgeOS 5, the universal operating system for the top robot brands and automation hardware.

Chapter 2. Safety Overview

Always follow corporate guidelines and regional safety regulations. For questions about device-specific safety requirements, refer to the device documentation.

Perform a comprehensive risk assessment for each task in each automation cell. **Not identifying, documenting, and addressing all potential risks and hazards can lead to serious injury or death.**

Some robot arms are made for collaborative applications. Never consider them inherently safe. Only use these "collaborative" robots in accordance with **ISO 10218-2:2012** and **ISO/TS 15066:2016**, and according to your comprehensive risk assessment.

Chapter 3. READY pendant Overview

The **READY pendant** is the touch screen interface for ForgeOS. ForgeOS software runs on both the READY pendant and an **industrial PC (IPC)**. Together, they are the brain of your workcell.

Safety Notices

The READY pendant has dual-channel safety features for use with robots and other devices. Safety features include an enabling device, an emergency stop button, and a keyed mode selector switch. All safety functions comply with EN ISO 13849-1 PLe, Category 4.

Follow this manual to avoid malfunctions or damage to the pendant.

CAUTION: HANDLE WITH CARE

- Do not use the READY pendant if defective or damaged.
- Do not use outdoors.
- Do not expose to excessive dust, humidity, or electromagnetic fields.
- Do not crush or damage the cable with any object.
- Do not lay the cable over sharp edges.
- Never clean the housing, touch screen, or operating elements with solvents, scouring agents, or scrubbing sponges. Use only a soft cloth and mild detergent.
- Avoid contact with liquids.
- Make sure that no foreign objects or liquids can penetrate the device.
- Check the cables and protective covers regularly for damage.
- Do not leave near heat sources or in direct sunlight.
- Avoid dropping and don't place on unstable surfaces.
- Do not open the housing. The READY pendant is sensitive to electrostatic discharge.
- To avoid damage to the touch screen, do not place the READY pendant on a surface with the screen facing down.
- Never use sharp objects to operate the touch screen and keypad (e.g., screwdriver, ballpoint pen, etc.). Use only your fingers or a touch-pen.
- Make sure your hands or gloves are clean and free of oils or abrasive debris.

Specifications

READY pendant Specifications	
Dimensions	215(W) x 284(H) x 69(D) mm (without mounting bracket)
Weight	Approx. 1120 g
Display	10.1" TFT, Projected capacitive touch screen, multi-touch 800 x 1280 pixel (16:10) WXGA

READY pendant Specifications	
Safety Elements	Emergency-stop button, 2 channels, IEC 60947-5-5 Compliant Enabling switch, 3 position, 2 channels, IEC 60947-5-8 Compliant Mode selector key switch, 2 positions (two keys included) (Replacement keys: IDEC AS6-SK)
Other Components	Membrane keypad, 18 buttons, 3 status LEDs
Data Transfer	Ethernet 10/100 Mbit/s communication (internal RJ45 plug) USB 2.0 transfer (port)
Cable	10 m, including 200 mm flying leads: RJ45 plug (4 wires) for Pendant communication 15 signal wires with ferrule ends
Power Supply	24V DC, up to 15 W consumption
Environmental Requirements	Operating temp: 0 °C to 45 °C Relative air humidity (non-condensing): 5 % to 95 % Vibration resistance according to IEC 60069-2-6 Shock resistance according to EN 61131-2
Mechanical Properties	Housing made of PPE/PE Dropworthiness: 1 m (on industrial floor), then at least IP54
Protection Rating	IP54

Cable

The READY pendant cable includes an RJ45 Ethernet connector and 15 flying leads. The Ethernet connector is for communication with the IPC. The flying leads with end ferrules are for the pendant power and safety features. See the table below for a description of the leads.

Pendant Cable Lead	Signal
Brown	Three-Position Enabling Switch Circuit 1
Yellow	Three-Position Enabling Switch Circuit 1
Green	Three-Position Enabling Switch Circuit 2
Grey	Three-Position Enabling Switch Circuit 2
Pink	24V DC
Green/Brown	Emergency Stop Circuit 1
White/Green	Emergency Stop Circuit 1
Grey/Pink	Emergency Stop Circuit 2
Red/Blue	Emergency Stop Circuit 2

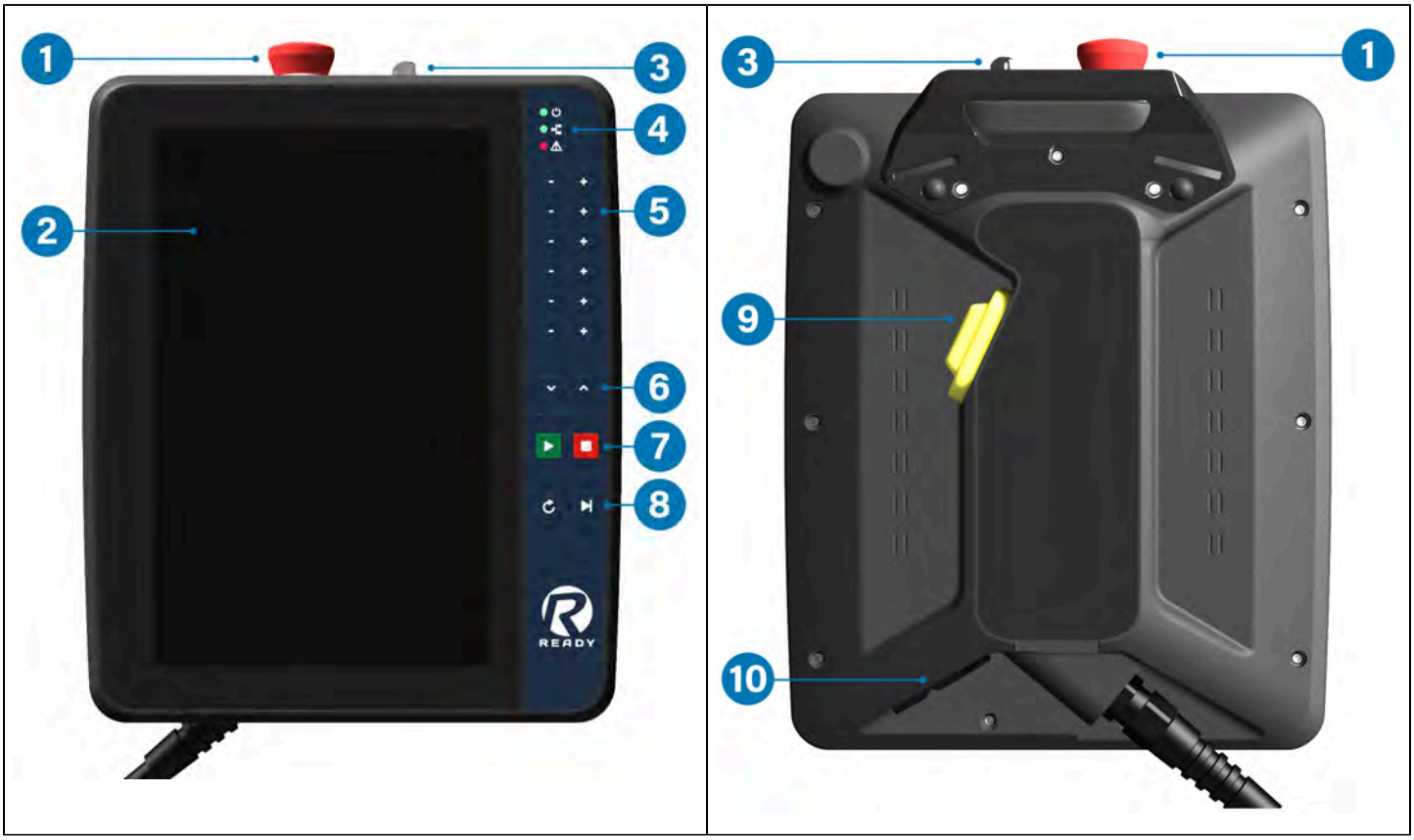
Pendant Cable Lead	Signal
Black	0V DC
Violet	Key Switch Circuit 1
White/Pink	Key Switch Circuit 1
White	Key Switch Circuit 2
Blue	Key Switch Circuit 2
White/Blue	Not Connected
Shielded Communication Wires:	
Blue, Pin 1	TD+ (Ethernet to IPC)
Orange, Pin 2	TD- (Ethernet to IPC)
White, Pin 3	RD+ (Ethernet from IPC)
Red, Pin 6	RD- (Ethernet from IPC)

Functionality

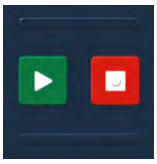
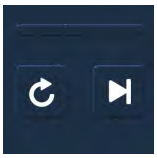
See the images and table below for READY pendant features.

Note:
The hand strap for holding the pendant is not shown. Set up the strap for either left-handed or right-handed use.

Table 1.



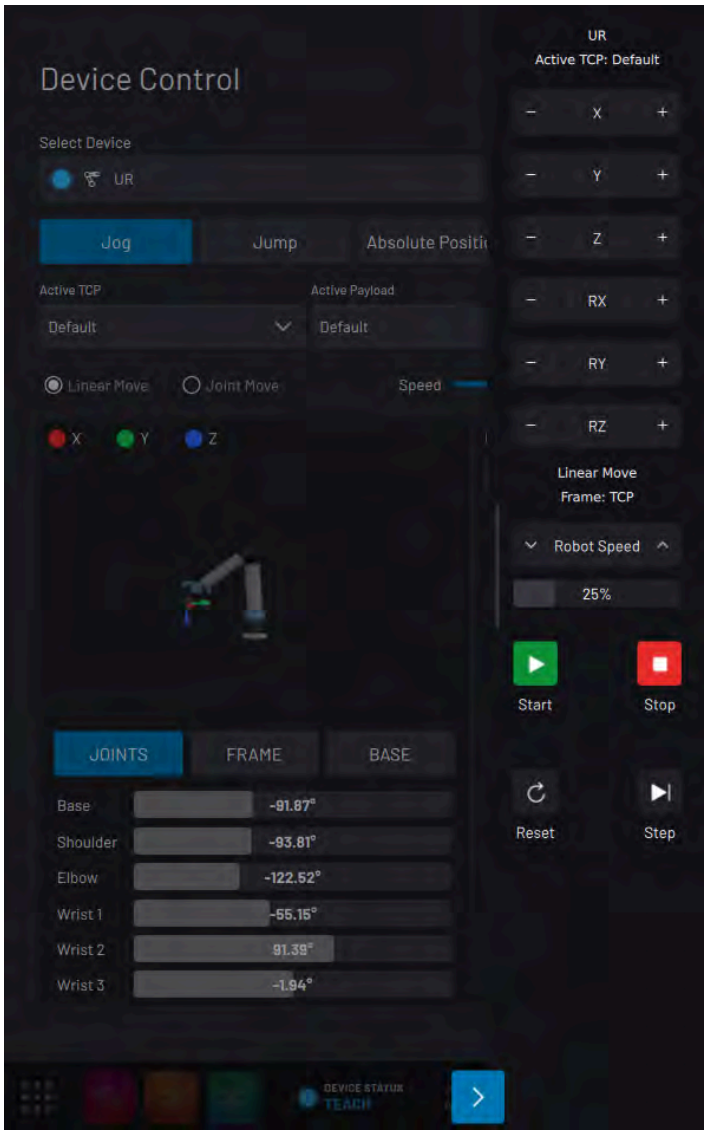
No.	Feature	Description	
1	Emergency Stop Button	Trigger an emergency stop state with this red-yellow button (DPST). Press down to open the safety circuits. This stops the robot and other devices. Twist clockwise to release the button and close the circuits.	
2	Touch Screen	Interact with ForgeOS on this multi-touch display.	
3	Mode Selector Key Switch	Insert the key and turn to select between **Run Mode** (clockwise) and **Teach Mode** (counter-clockwise).	
4	Status LEDs	1. **Green Power LED** - Solid green means the pendant is on. 2. **Green Network LED** - Solid green means a good connection to the IPC. 3. **Red Error LED** - Solid red means there is a device error. Off indicates normal system status.	

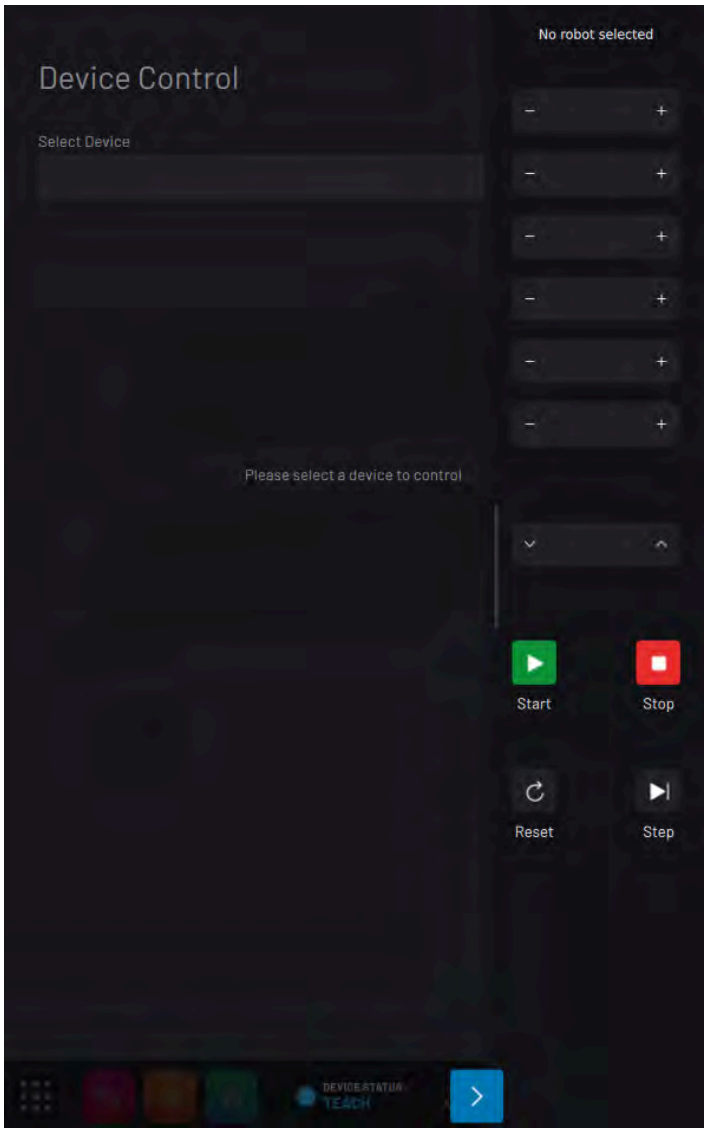
No.	Feature	Description	
5	Jog Buttons	Jog the most recently selected robot in the selected Frame. If you select **Linear** in the Device Control app, these jog buttons move the robot's TCP in X, Y, Z, RX, RY, and RZ. If you select **Joint**, the buttons move the robot joints. Holding two buttons in the same row at one time results in no motion.	
6	Speed Control Buttons	**Decrease Speed** (left) and **Increase Speed** (right). While a task is running, control the runtime speed slider. While a task is not running, control the speed in the Device Control app.	
7	Start/Stop Buttons	**Start** (left): Use in the Task Canvas app to start the task (Run mode). Use in the Device Control app to execute a jump/absolute position command. **Stop** (right): Stop Task Canvas execution and all robot motions.	
8	Reset/Step Buttons	**Reset** (left): Send a reset signal to all devices to fix errors. **Step** (right): Execute the selected block in Task Canvas.	
9	Three-Position Enabling Switch	Enable robot motion control with this three-position switch (DPDT). Most robots require this enabling device when the READY pendant is in Teach Mode. To use, press the switch into the middle position (ON). If you release or squeeze the	

No.	Feature	Description
		switch too tightly in Teach Mode, the robot will not move (OFF).
10	USB Port	Connect USB devices to the pendant. **Tip:** Connect a USB keyboard to type in fields on the screen.

Tip:

If you forget what the keypad buttons do, tap the arrow in the bottom-right corner (or swipe left from the right edge of the screen) to see labels! If you have a robot selected to jog in the Device Control app and it's in Teach/Program mode, this legend lists the currently-selected Active TCP, move type, frame, and speed. If you do not have a robot selected or if it's unable to jog, the -/+ section of this legend appears blank.





Chapter 4. ForgeOS Platform

System Requirements

An industrial computer must meet these minimum requirements to install ForgeOS:

Device Feature	Minimum Required Specification
CPU	i5 8500T / i3 8100T
Memory	8GB DDR4 RAM
Storage	128GB SATA SSD
Media	Two USB 2.0 ports
Display Adapters	One DisplayPort or HDMI
Network Adapters	Two Gigabit Ethernet ports
Temperature Rating	Recommended 0-50ºC

Installing ForgeOS

Follow these steps to install ForgeOS and sign in to the Admin role. Installation takes about 30 minutes, depending on the resources of the IPC.

1. To install ForgeOS, follow these substeps. You need a ForgeOS installation USB flash drive. Contact your READY Robotics distributor for an installation USB drive.

   > **Important:**
   > Installing ForgeOS will erase all data on the target hard drive.

   a. Connect a monitor, keyboard, and mouse to the IPC where you want to install ForgeOS.

   b. Plug the ForgeOS installation USB flash drive into the IPC.

> **Tip:**
> If you need more USB ports, use a USB 3.0 hub.

c. Restart the IPC. While the IPC is powering on, press the keyboard hotkey that takes you to the Boot Menu.

> **Tip:**
> The key that opens the Boot Menu depends on the IPC model. The most common keys that do this are ESC, F10, F11, or F12. Refer to your computer's documentation for boot options.

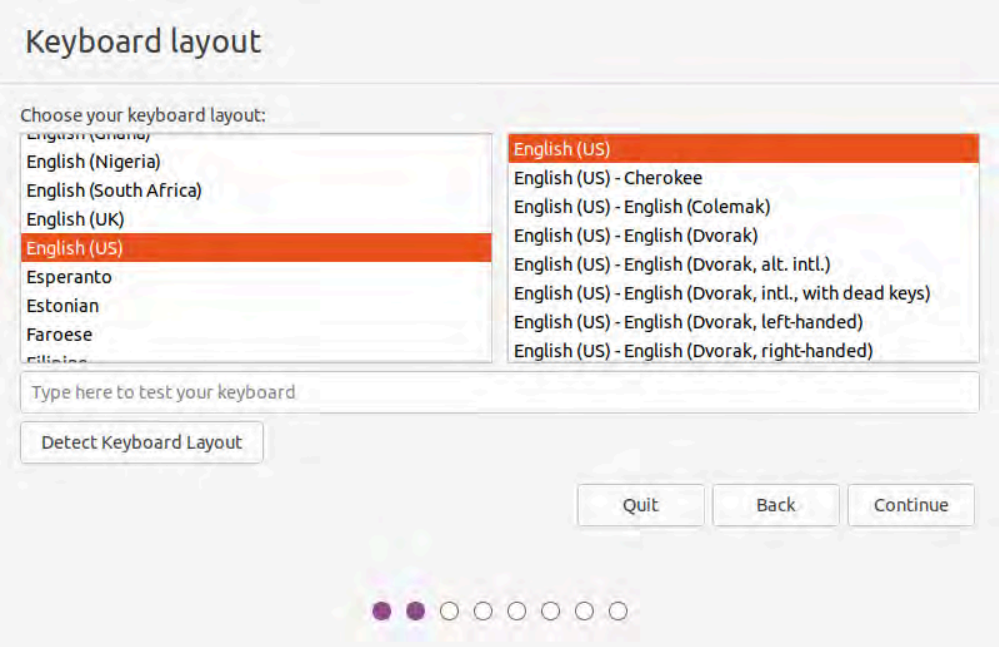
> **Note:**
> If you're installing Forge/OS on a **Forge/Ctrl**, press F11. You may need to enter the **BIOS Admin password**. Contact READY Support if you run into this issue.

d. From the boot options, select **Install ForgeOS** to boot from the installation USB flash drive.

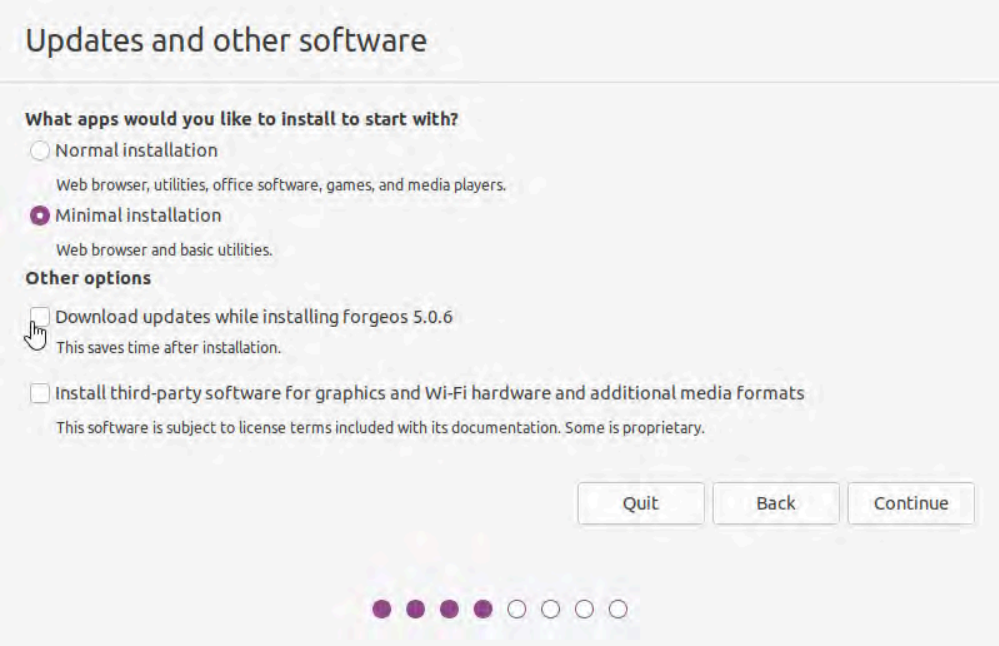
e. The installer may take several minutes to load. Wait until the installation wizard opens.

f. Select your language. Then click **Install Forge**.

g. Choose a keyboard layout. Then click **Continue**.

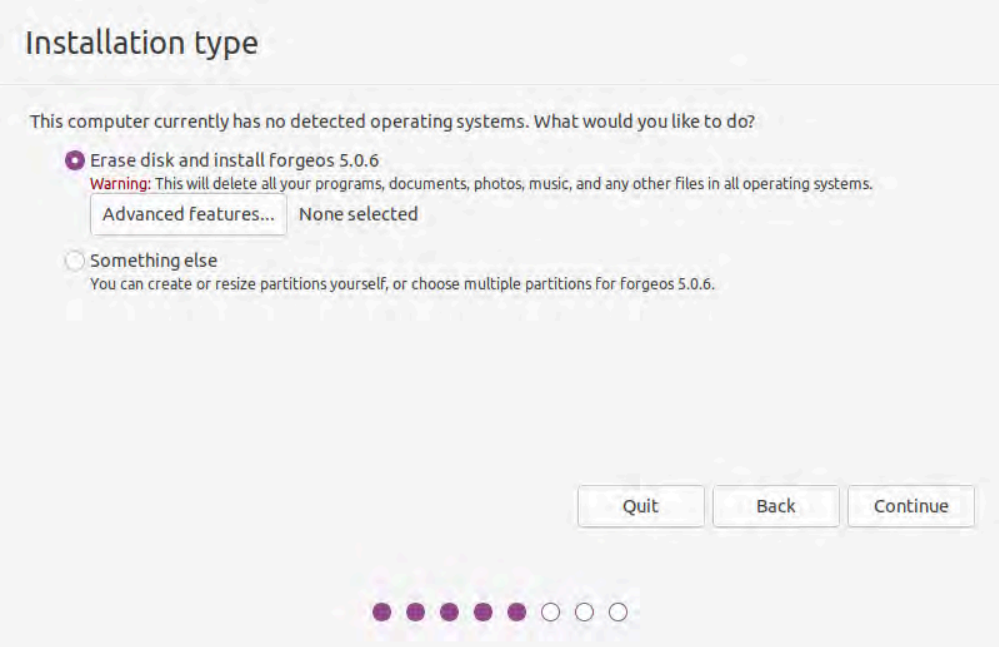


h. Select **Minimal installation**. Uncheck **Download updates while installing forgeos**. Then click **Continue**.

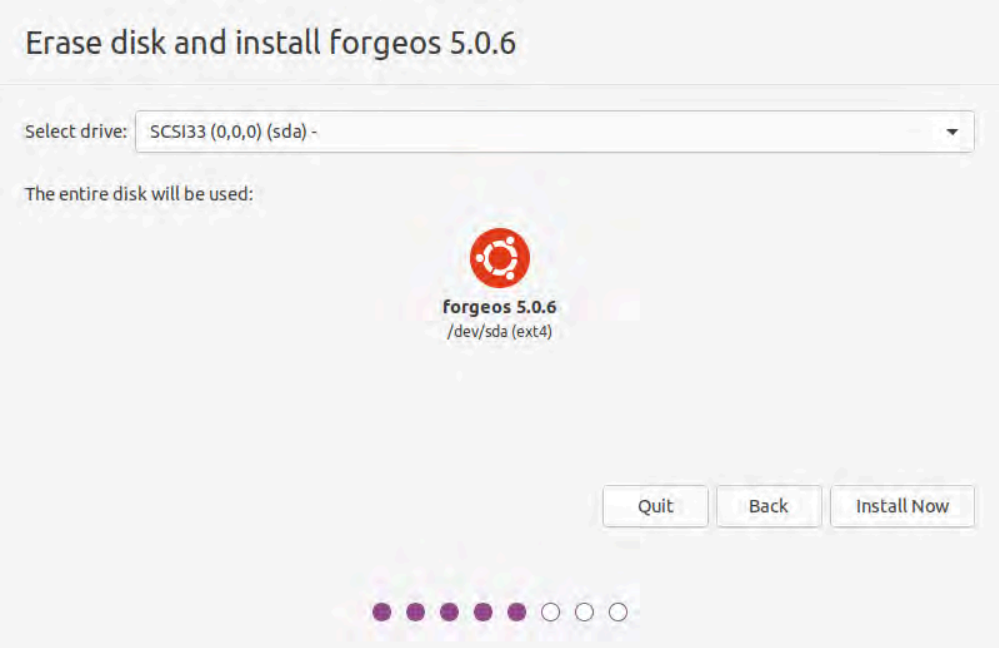


i. Select **Erase disk and install forgeos**. Then click **Continue**.

> **Note:**
> If ForgeOS is already installed, the installation wizard will show additional options. The goal is to erase the entire disk for a brand new installation.

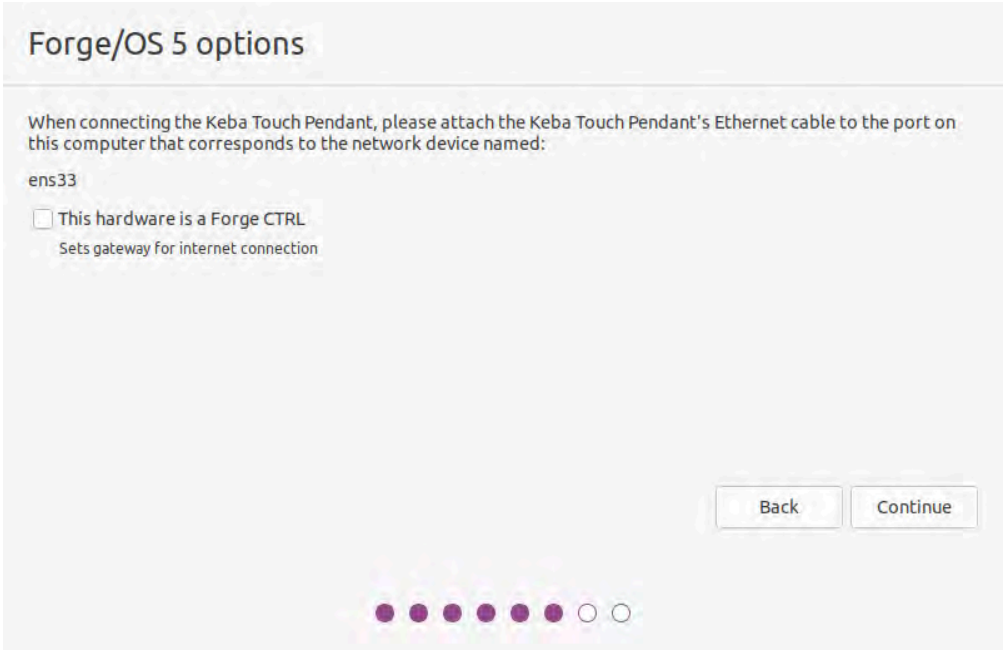


j. Select the IPC hard drive for ForgeOS and click **Install Now**.

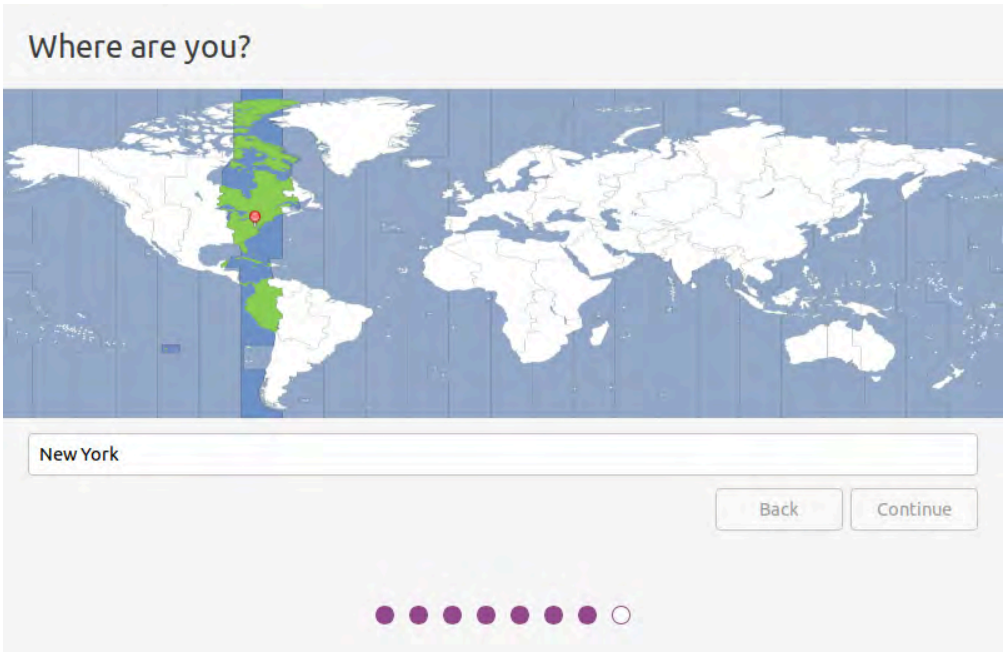


k. Confirm that you want to erase the entire disk by clicking **Continue**.

l. Make a note of the pendant instructions. If you're using a Forge/Ctrl, select the checkbox next to **This hardware is a Forge CTRL**.

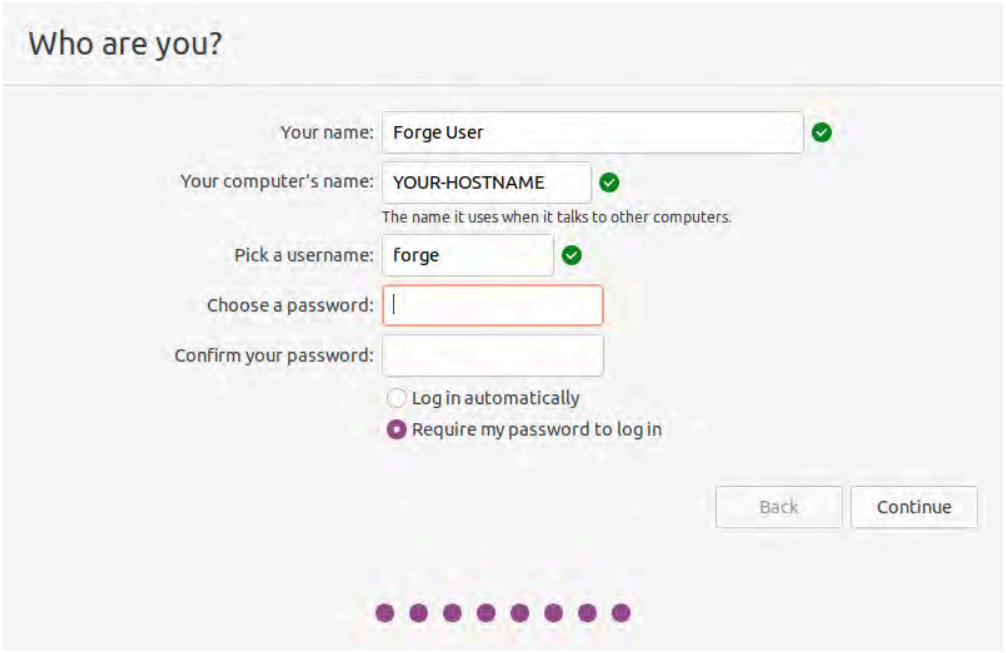


m. Choose your timezone. Then click **Continue**.

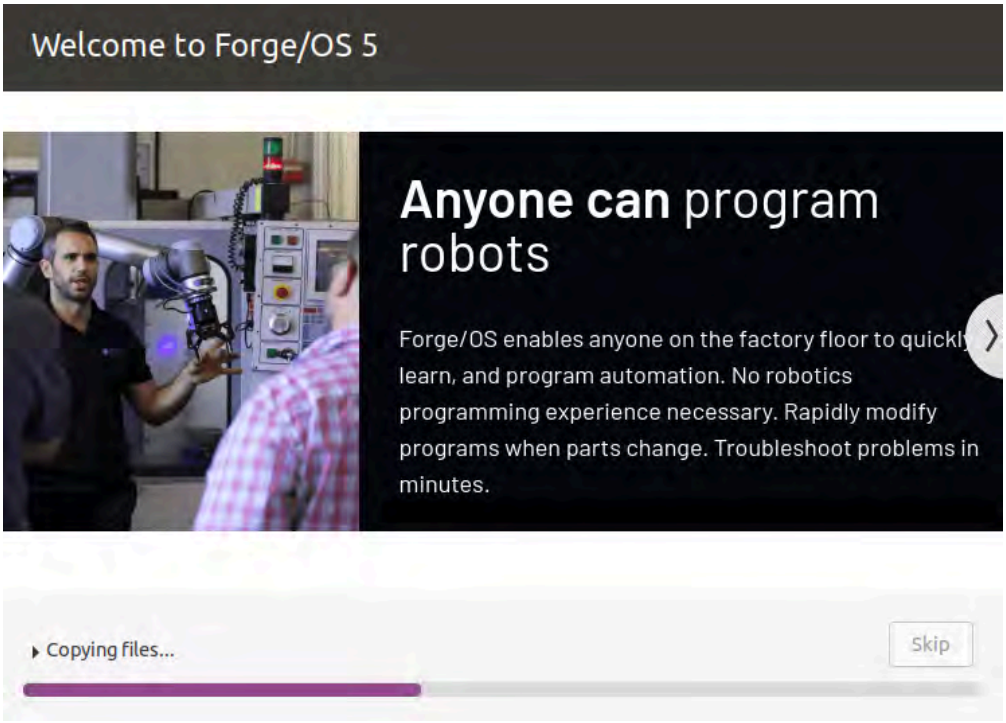


n. Choose your IPC's host name. The host name identifies the IPC on the network. Pick a username and password. Then click **Continue.**

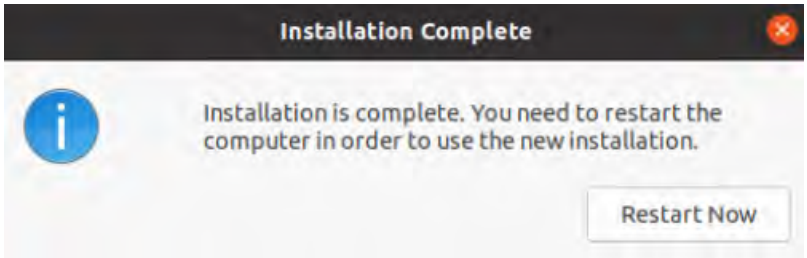
> **Note:**
> The username and password that you create here are for accessing the IPC desktop. They are NOT for signing into ForgeOS on the READY pendant.



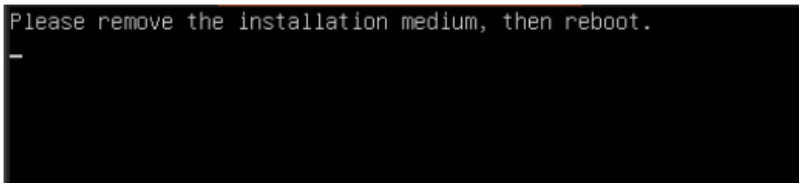
o. Wait for the installer to copy and install ForgeOS.



p. Once the installation completes, click **Restart Now**.

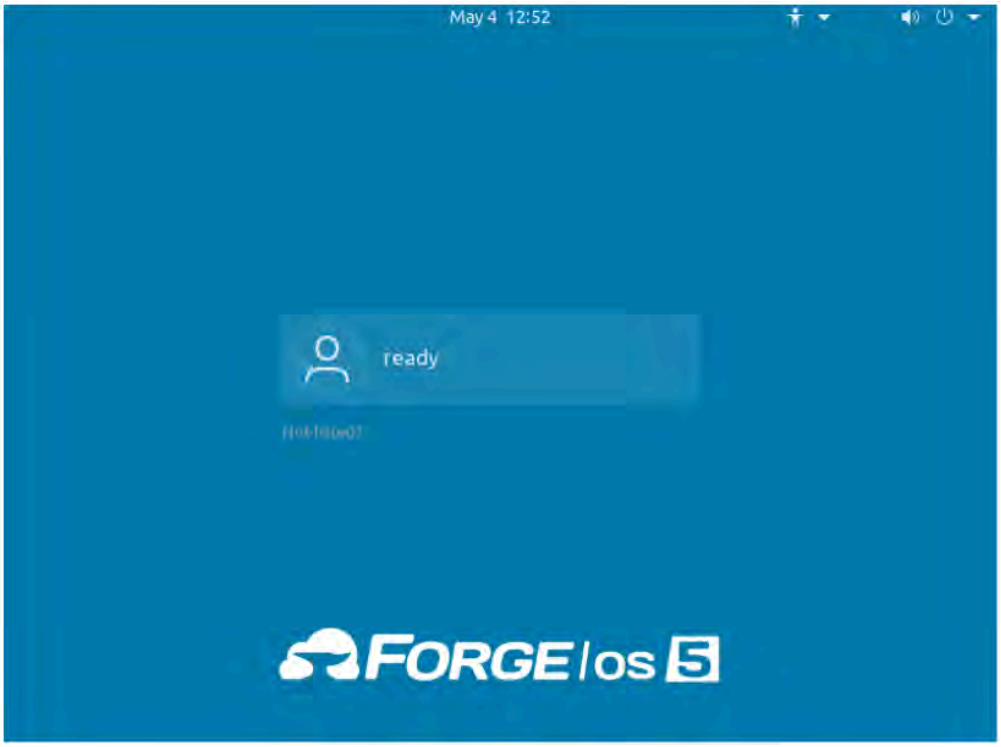


q. When prompted, remove the installation flash drive. Then reboot.



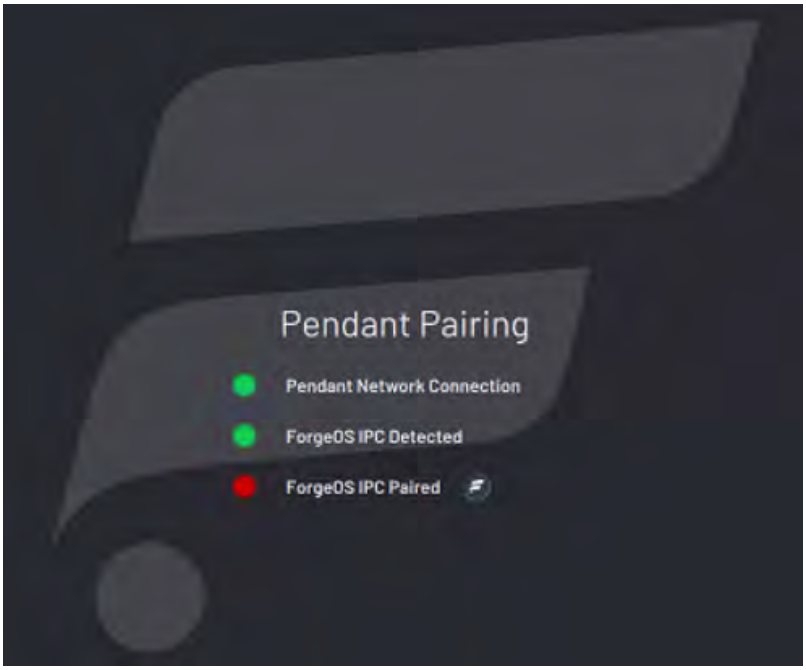
r. Wait for ForgeOS to finish booting.

s. When you see the login screen with the ForgeOS 5 logo, ForgeOS is ready to run on the READY pendant! You don't need to sign in to the desktop. Disconnect the monitor, keyboard, and mouse that you used to install ForgeOS.



2. The READY pendant automatically finds and pairs with the IPC. The three LEDs on the screen help you track the status:
   - **Pendant Network Connection**: This condition is satisfied when the READY pendant has a valid network connection (i.e., the Ethernet cable is plugged in).
   - **ForgeOS IPC Detected**: This condition is satisfied when the READY pendant detects a Forge/OS IPC on the network.
   - **ForgeOS IPC Paired**: This condition is satisfied when the READY pendant successfully pairs with the IPC. If pairing fails, it is automatically retried indefinitely.

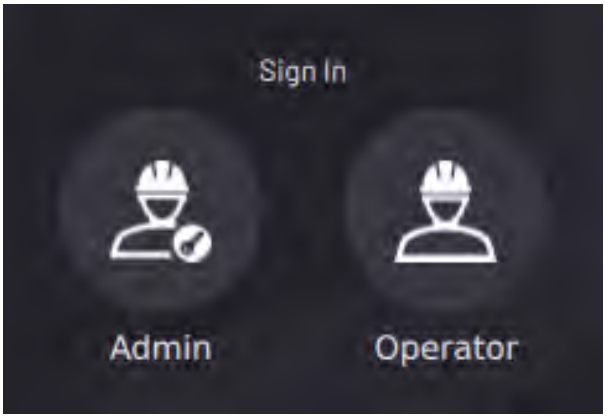
   When a condition is not satisfied, the LED is red. When a condition is in progress of becoming satisfied, a spinner around a READY logo appears to the right of the text. When a condition becomes satisfied, the LED turns green.



The UI shows the real-time state of each step. For example, if the pendant loses its network connection during pairing, all steps become undone. If the READY pendant spends more than 60 seconds on any step, troubleshooting text displays. Common things to check are if the READY pendant network cable is plugged in, if the IPC is powered on, if the READY pendant and IPC are connected to the same network, and if there's only one READY pendant and one IPC on that network.

Note:
The READY pendant IP Address is preset to 172.16.255.253. The network interface that the pendant connects to should use IP Address 172.16.255.250 and Subnet mask 255.255.255.0.

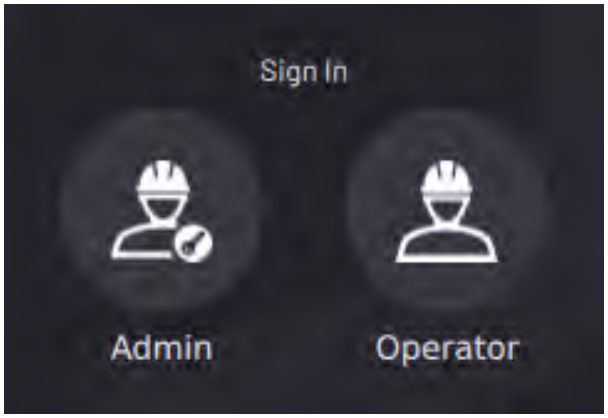
3. Tap **Admin** and sign in. The default Admin password is "forgeadmin".



Note:
After installation, you have limited access to ForgeOS until you activate it with a license code. See Activating ForgeOS with a License Code *(on page 39)*.

User Profiles

After you start up ForgeOS, sign in to a user profile. ForgeOS has two user profiles: Admin and Operator.



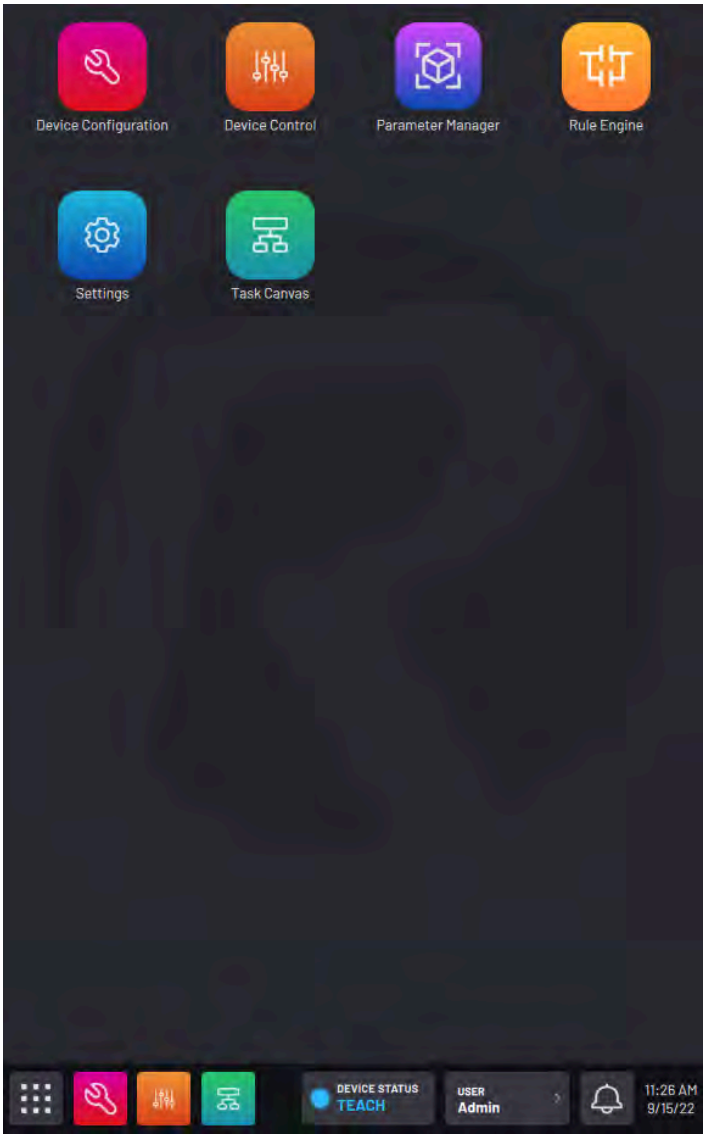
- The **Admin** profile is password-protected and has full access to apps in ForgeOS.
- The **Operator** profile does not have a password and has limited access.

Tip:
The default Admin password is "**forgeadmin**". Change the Admin login password in User Accounts *(on page 33)*.

The Operator profile CANNOT:	The Operator profile CAN:
• Access the System Settings app • Access the Device Configuration app • Create, modify, or remove tasks in Task Canvas	• View Device Status and clear device errors • View system notifications • Control an enabled device from the Device Control app • Load a task in Task Canvas • Execute a task in Task Canvas • Interact with user prompts in a running task

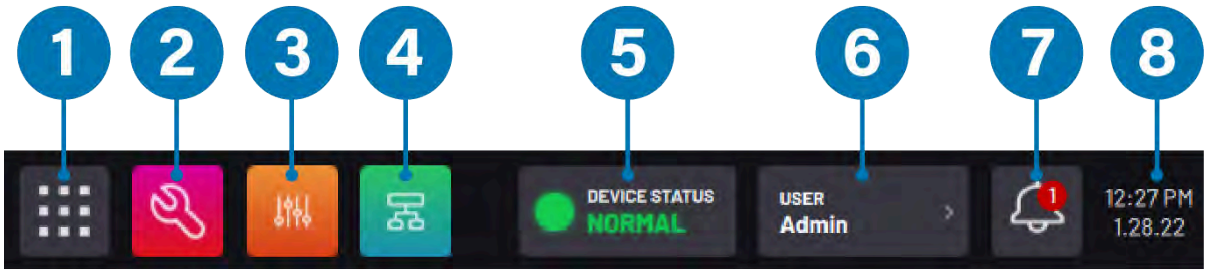
Home Screen

The **Home Screen** shows all apps installed on ForgeOS. Return to the Home Screen at any time by tapping the Home button in the bottom-left corner on the Taskbar.



Taskbar

The **Taskbar** displays app icons, device statuses, and system information. The Taskbar is always available except during task execution.



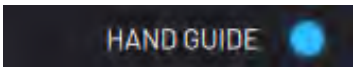
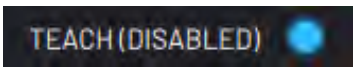
No.	Taskbar Item	Description
1	Home Button	Return to the Home screen. The Home Button switches active apps but does not close them. Apps only close when you log out.

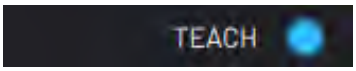
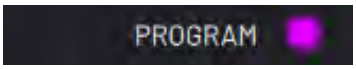
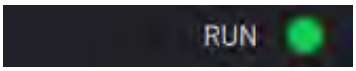
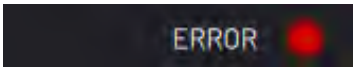
No.	Taskbar Item	Description
2-4	App Icons	Jump between the standard control apps: Device Configuration, Device Control, and Task Canvas. When you are in Remote Control mode, these app icons are replaced with the **Remote Control Status** icon. **Tip:** Change what apps appear here by going to Settings > Applications *(on page 32)*.
5	Device Status Panel	View the operational mode or state of all devices that are **configured** and **enabled**.
6	User Button	View the current user profile name. When you tap the button to expand, you can log out, enter Remote Control mode, or exit Remote Control mode.
7	Notifications	View notifications as they occur in the bottom-right corner. After three seconds, they disappear from the bottom-right corner, but they are still stored in the Notifications Panel. Access the Notifications Panel by tapping the bell icon. Any ongoing status information appears in the Device Status Panel. **Tip:** Also access the Notifications Panel by swiping down from the top of the screen.
8	Time and Date	The system time and date appear on the Taskbar. To update the time settings, see General Settings *(on page 31)*.

Device Status Panel

The **Device Status Panel** displays the operational mode or state of all enabled devices. Expand the panel by tapping the **DEVICE STATUS** button in the Taskbar.

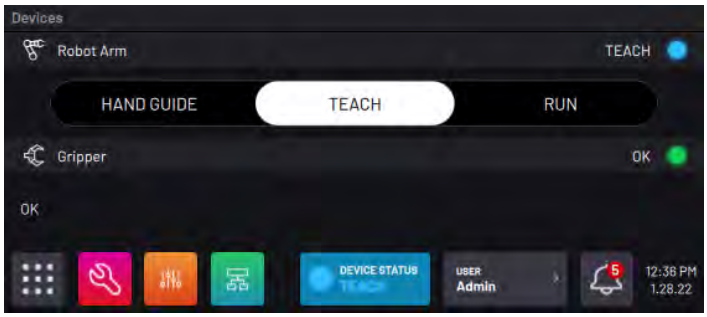
The possible robot states are:

Robot State	Description
	The *collaborative* robot can be moved by hand.
	The robot is in a manual reduced speed mode, the READY pendant enabling switch is not pressed, and the motors are off.

Robot State	Description
	The robot is in a manual reduced speed mode, the READY pendant enabling switch is pressed in the middle position, and the motors are on.
	The robot is in a manual reduced speed mode where speeds are limited by ForgeOS instead of the robot safety system. The READY pendant enabling switch is not required.
	The robot is in an automatic full speed mode. It's ready to run a task.
	The robot requires a reset from ForgeOS.
	The robot requires a reset from ForgeOS or other actions to regain normal operation.

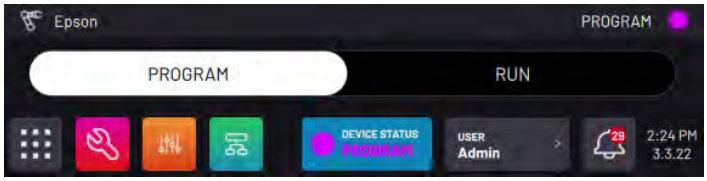
For robots requiring the **Mode Selector Key Switch**, turn the key to select between **Run** (clockwise) and **Teach** (counter-clockwise).

For collaborative robots, open the Device Status Panel and tap to choose a robot mode.



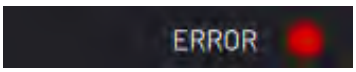
> **Tip:**
> You cannot toggle directly from Hand Guide mode to Run mode. First switch to Teach mode to make Run mode available.

For robots with **PROGRAM** mode, open the Device Status Panel and tap to choose a robot mode. For example, tap **PROGRAM** to change the robot to Program mode.



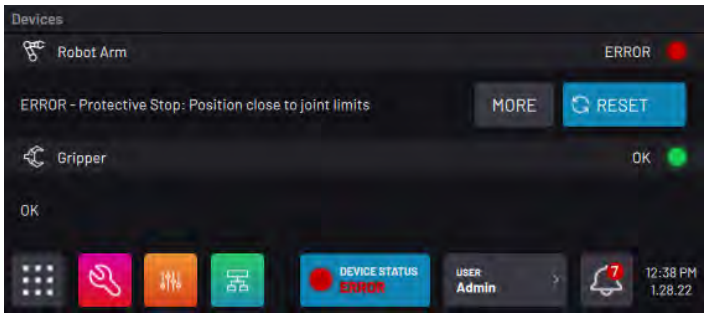
Note:
Robots that do not require hardware switching between modes will always start up in Run mode after a ForgeOS restart. This allows for a smooth integration with Remote Control mode.

The possible states for other devices are:

Device State	Description
	The device is connected and in a normal state.
	The device requires a reset from ForgeOS.
	The device requires a reset from ForgeOS or other actions to regain normal operation.

Expand the panel and tap **MORE** to read possible causes and fixes.

The **RESET** button clears warnings and errors when possible. All safety conditions, such as the pendant enabling switch and key switch, must be met.

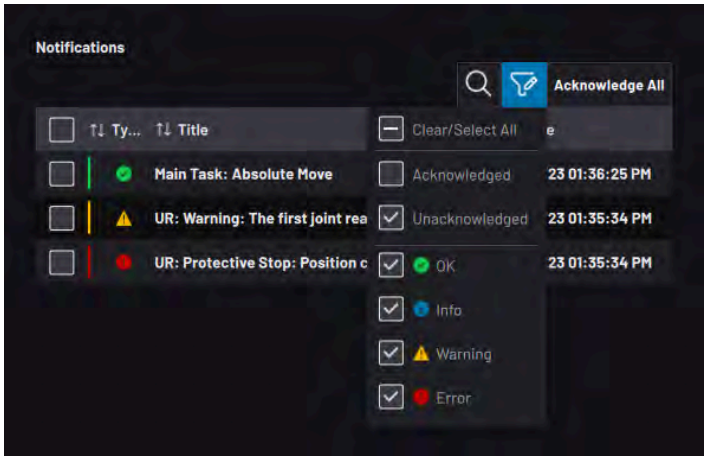


Note:
For hardware errors that will not reset in ForgeOS, follow the manufacturer's error recovery procedures.

Notifications Panel

The Notification Panel displays a table of status history.

Tap the funnel icon to filter the table. Choose one or more filters to only show certain notification types (i.e., OK, info, warning, and error) in the acknowledged list and/or in the unacknowledged list.

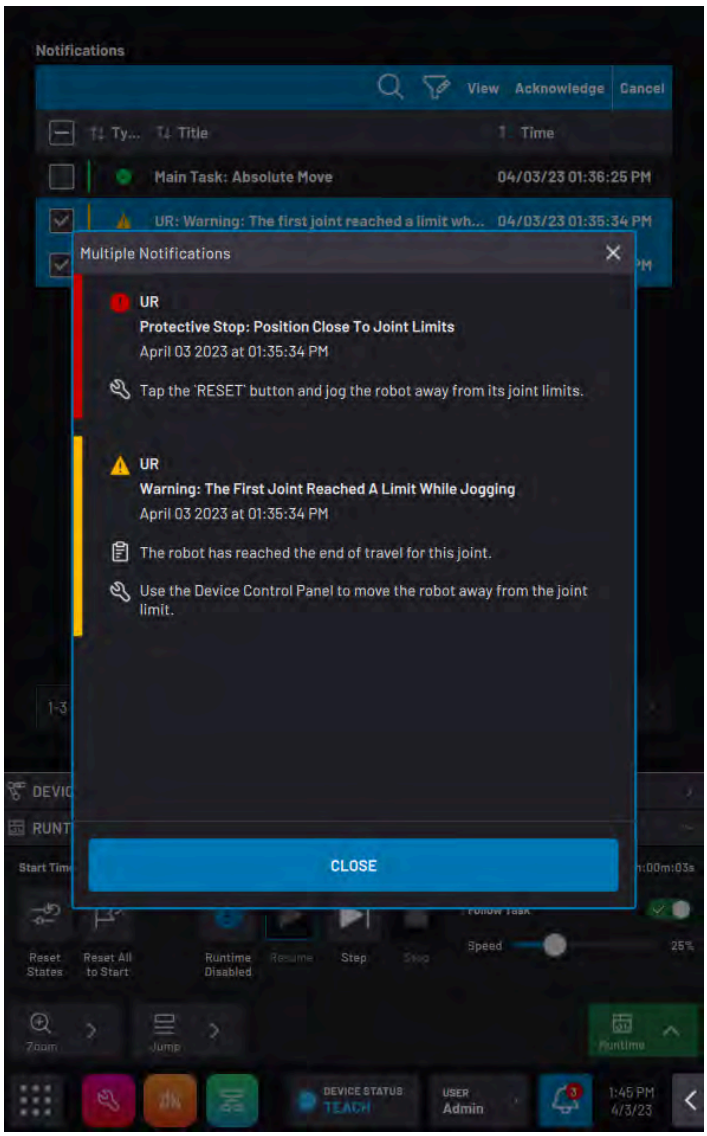


New notifications appear in the unacknowledged list until you acknowledge them.

Select one or more notifications and tap **View** for more information (such as fix instructions).

To clear the selected notification(s) to the acknowledged list, tap **Acknowledge**.

To clear all unacknowledged notifications at once, select no notifications and tap **Acknowledge All**.



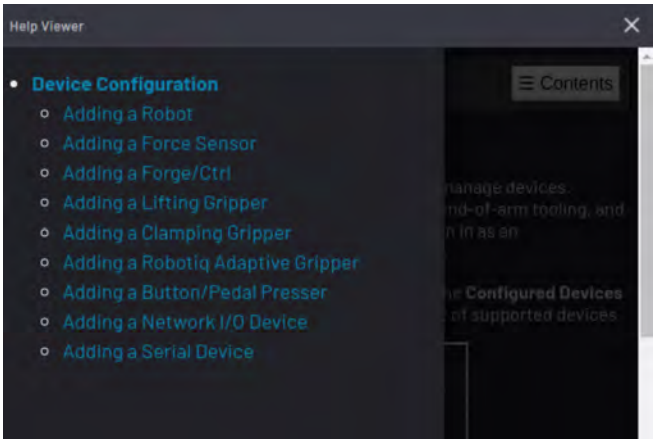
Help Viewer

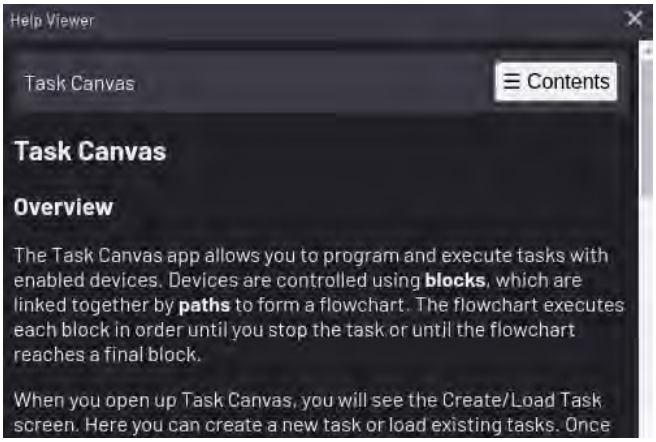
The Help Viewer displays additional information and instructions for using features in ForgeOS.

To open the Help Viewer, tap the **?** icon in the upper-right corner of the screen or pop-up, when available.

Tip:
In the Task Canvas app, select **Help** in the **File** menu.

In some cases, there are links to relevant topics within the help text. To see more contents or subtopics, tap **Contents**.

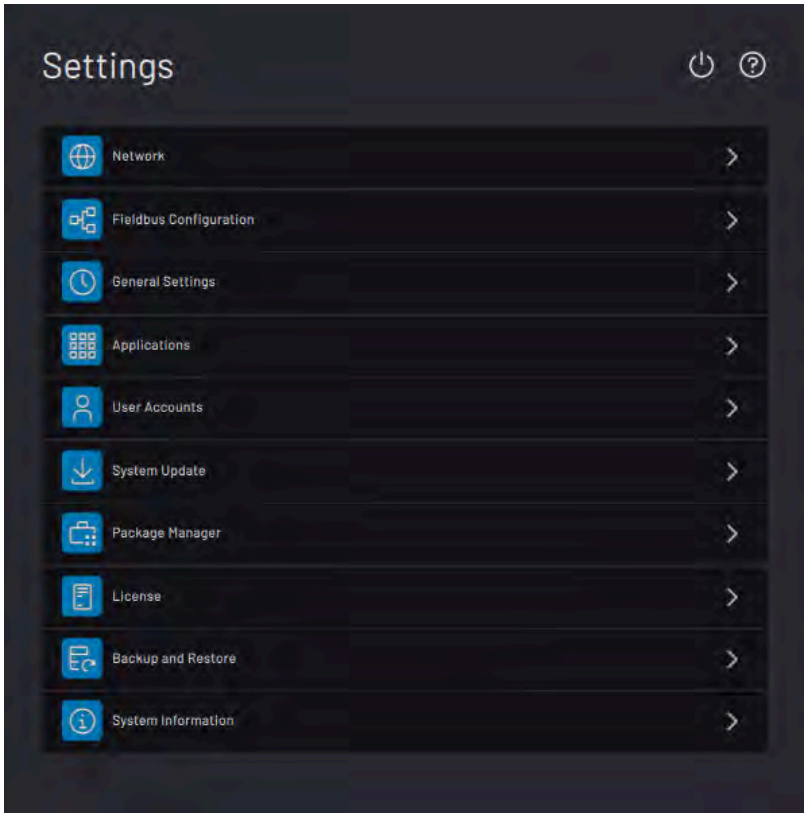

Help Viewer
Device Configuration
Adding a Robot
Adding a Force Sensor
Adding a Forge/Ctrl
Adding a Lifting Gripper
Adding a Clamping Gripper
Adding a Robotiq Adaptive Gripper
Adding a Button/Pedal Presser
Adding a Network I/O Device
Adding a Serial Device
Contents


Help Viewer
Task Canvas
Contents
Task Canvas
Overview
The Task Canvas app allows you to program and execute tasks with enabled devices. Devices are controlled using blocks, which are linked together by paths to form a flowchart. The flowchart executes each block in order until you stop the task or until the flowchart reaches a final block.
When you open up Task Canvas, you will see the Create/Load Task screen. Here you can create a new task or load existing tasks. Once

Chapter 5. Settings

The Settings app is used to view system information and change global settings. Sign in as **Admin** to access it.

The Settings app is organized into several menus. Tap an option to enter that menu. Return to the Settings main screen by tapping the back button at the top of the screen.

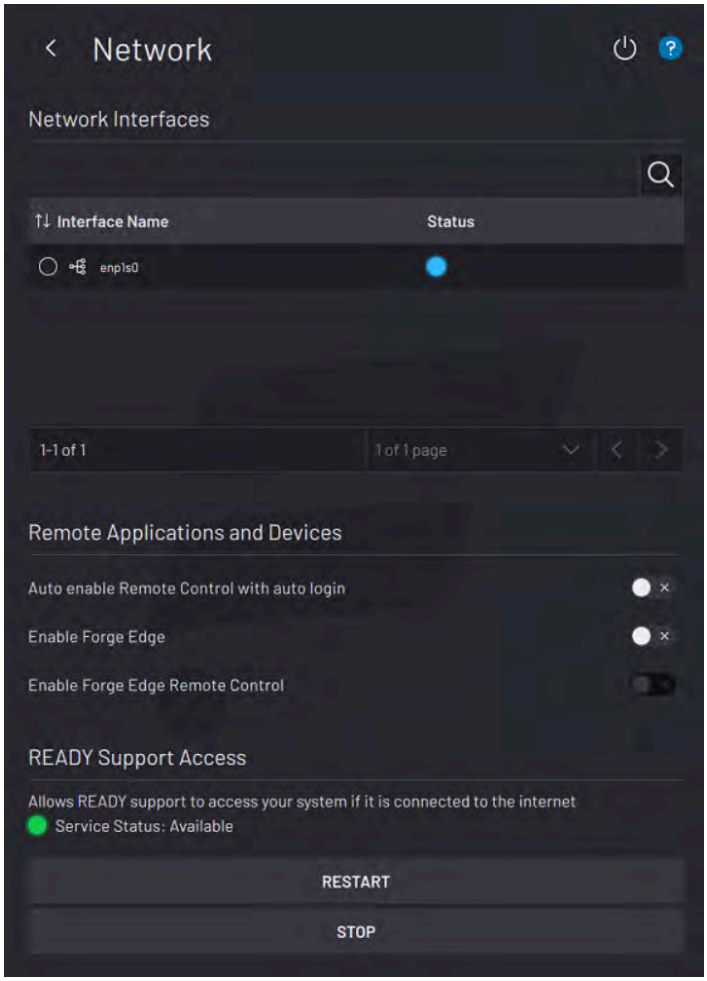


Network

The Network screen is where you control network-related settings.

In the "Network Interfaces" table, view and edit Wi-Fi and Ethernet interfaces on your IPC.

In the "Remote Applications and Devices" section, toggle on **Auto enable Remote Control** to automatically begin in Remote Control mode when ForgeOS restarts.



> **Note:**
> Robots that do not require hardware switching between Teach and Run mode (such as keyless robots and some cobots) will always start up in Run mode after a ForgeOS restart. This allows for a smooth integration with Remote Control mode.

Forge Edge opens the door for creating custom web-based apps for ForgeOS.

If you toggle on **Enable Forge Edge** and set up a network connection to your computer, you can collect real-time data of your ForgeOS system.

> **Important:**
> For security reasons, the Forge Edge toggle is disabled by default. It is your responsibility to ensure that you trust any network that you connect the ForgeOS system to (e.g., deploy within an isolated LAN or a VPN) and that you follow all safety guidelines in your robot OEM manual.

If you additionally toggle on **Enable Forge Edge Remote Control**, you can remotely control ForgeOS from the web interface. This allows you to do things like load and start Task Canvas tasks directly from your computer.

> **Note:**
> For more information about how to set up a network connection to your computer, how to launch Forge Edge, and how to give web-based apps created with Forge Edge access to your ForgeOS instance, refer to the READY Robotics website.

The "READY Support Access" section is where you enable remote access for READY Robotics Support to service or troubleshoot your system. First connect your system to the Internet, then use the controls to stop and restart access for remote connection.

Fieldbus Configuration

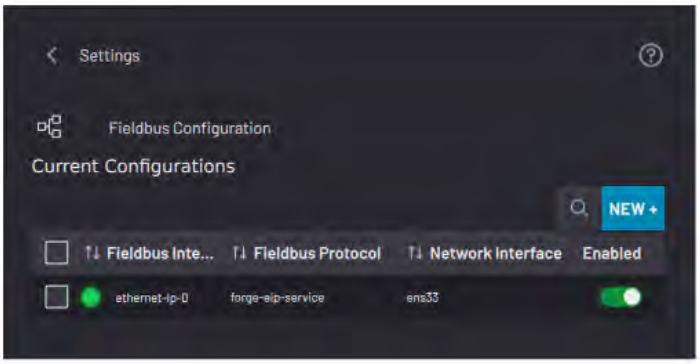
The Fieldbus Configuration screen is where you add, delete, and change the state of Fieldbus interfaces.

Note:
Add fieldbus interfaces in the Settings app before adding remote I/O devices in the Device Configuration app.

There is a Status Indicator to the left of each interface. The indicator is red when the interface is stopped and green when it is running.

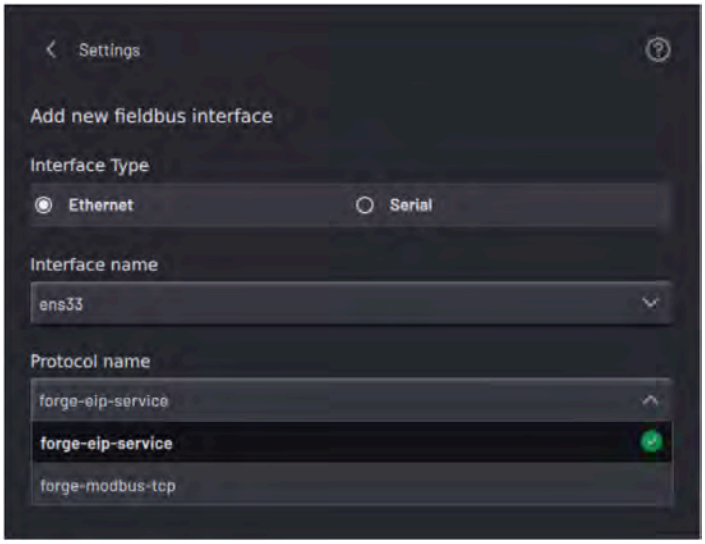
Tap the toggle switch in the **Enabled** column to run or stop the interface.

Tap the blue **NEW +** button to create a new Fieldbus configuration.

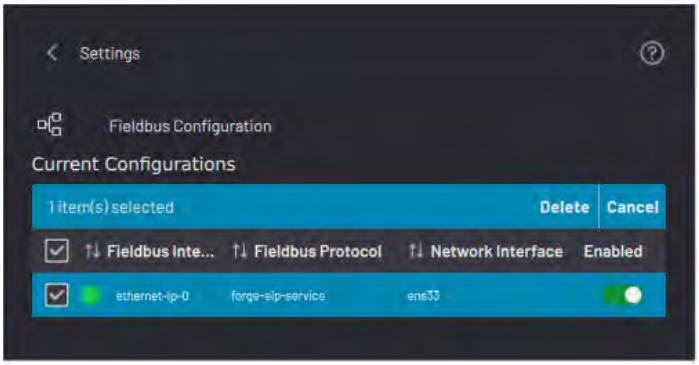


Choose between **Ethernet** and **Serial**, select an interface by name, and choose the protocol.

- For an Ethernet interface, your protocol options are Ethernet/IP and Modbus TCP.
- For a serial interface, ForgeOS supports Modbus RTU.



Select one or more configurations and tap **Delete** to remove them from your Fieldbus configurations list.

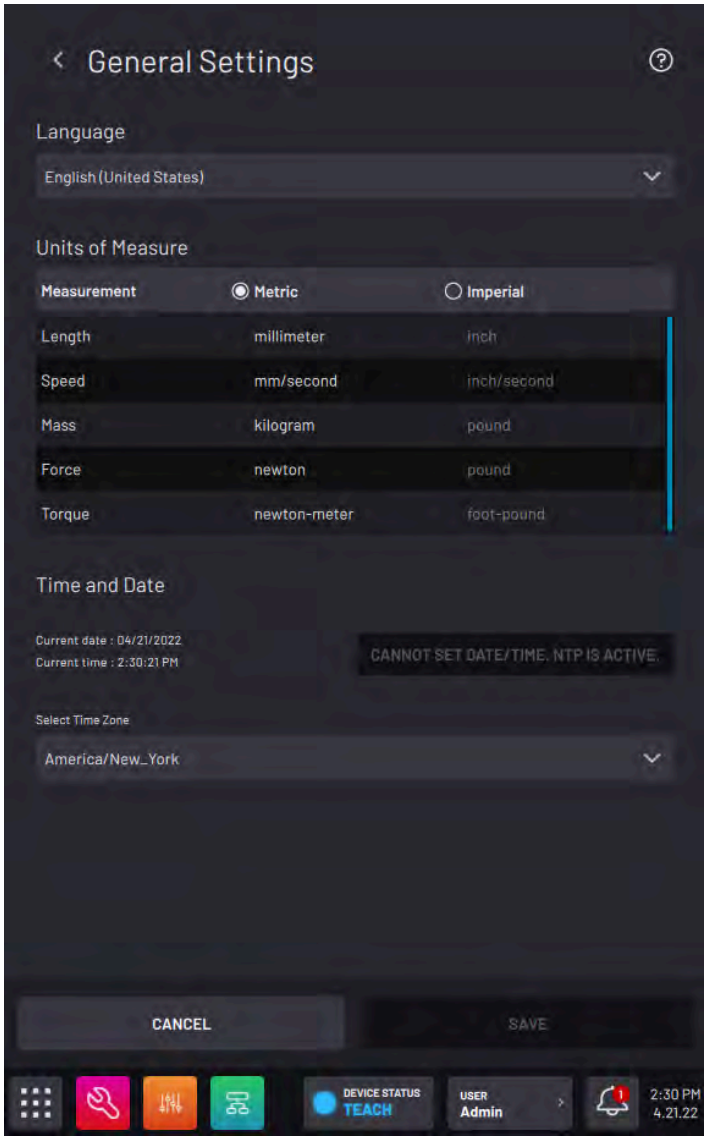


General Settings

The General Settings screen is where you access language, unit, and time settings.

Select a **Language**. If you change the language, restart the system to display all translatable text into the selected language.

Select **Metric** or **Imperial** units of measurement. The default system of measurement is Metric, but you may change units to Imperial at any time. Changing the unit system will automatically convert numerical values of known units.



Important:
Variables are not affected by changing units of measure in General Settings. You must manually update variables that are meant for length, speed, or force parameters. Example:

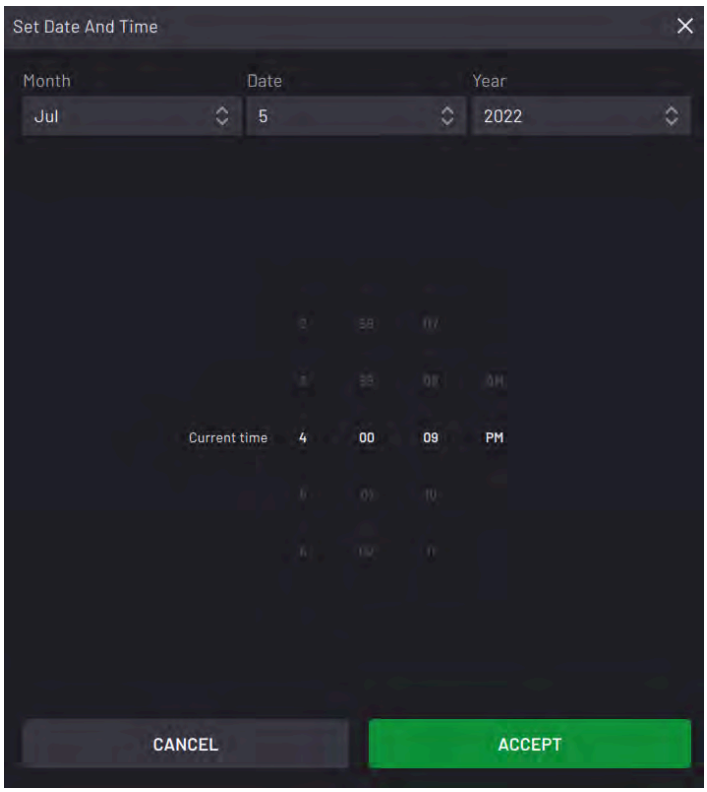
- If you enter 25mm in the Z field of a move block, then you switch to Imperial units, ForgeOS converts that value to 0.984in.
- If you create an integer variable with a value of 25 to use in the Z field of a linear relative move, then you switch to Imperial units, Forge/OS does NOT convert that value.

Tap **SET DATE AND TIME** to change the date and time that appears in the Taskbar.

At the top, use the dropdowns to select the current month, day, and year.

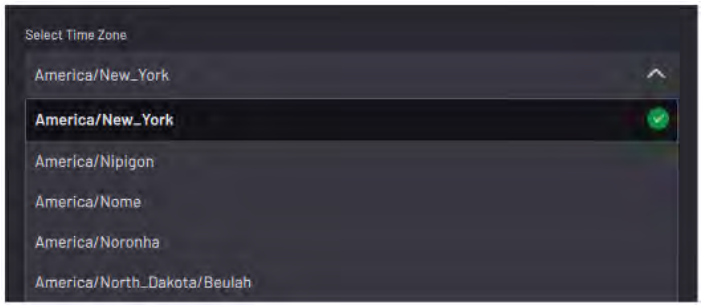
At the bottom, use the number scrolls to select the current time in hours, minutes, and seconds.

Note:
You cannot set the date and time if NTP is active.



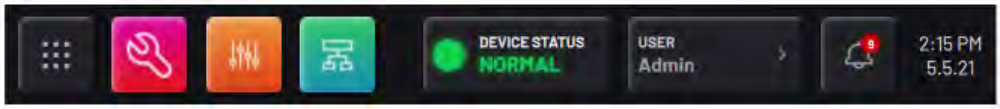
In the **Select Time Zone** dropdown, tap the time zone that you are in.

Note:
When you change the timezone, the time in the Taskbar updates to match the time difference.



Applications

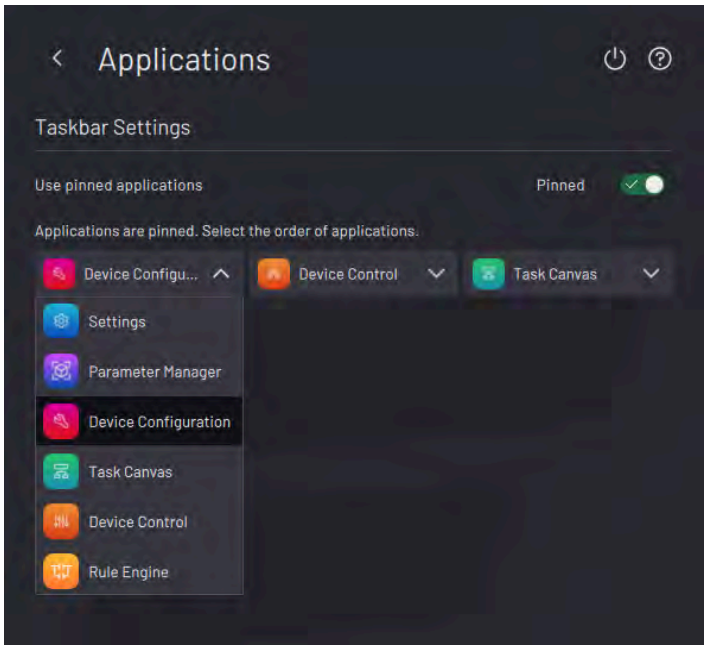
The Applications screen is where you choose what app icons appear in the three spots that are to the right of the Home button.

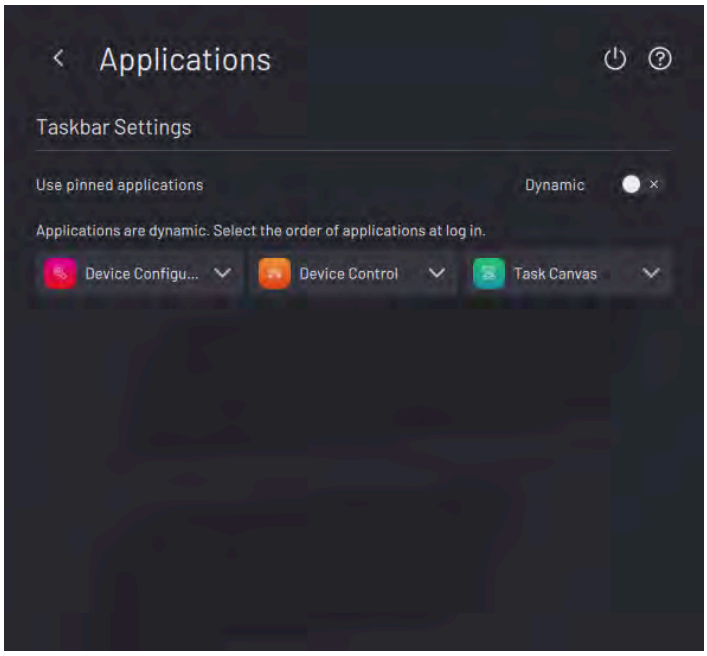


If you enable the “**Pinned**” toggle, your app selections appear in the Taskbar (in the order that you chose) as soon as you tap **SAVE**. The order of these three apps will not change as you navigate between them and other apps.

If you disable the toggle, the label switches to “**Dynamic**”. Your app selections appear in the Taskbar (in the order that you chose) as soon as you log in again to the Admin profile. The order of the apps dynamically updates to show the three most recently used apps:

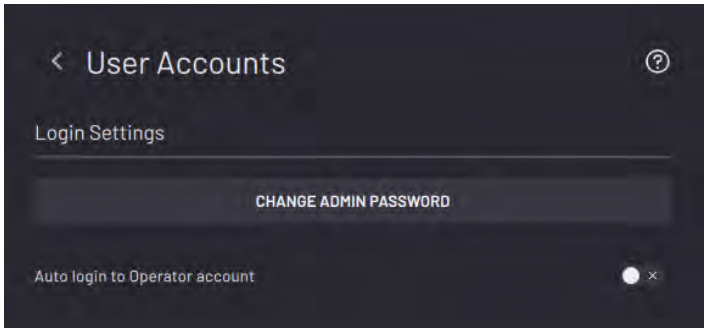
- The active app is on the right.
- The previously-used app is in the middle.
- The app that was used before the previously-used app is on the left.





User Accounts

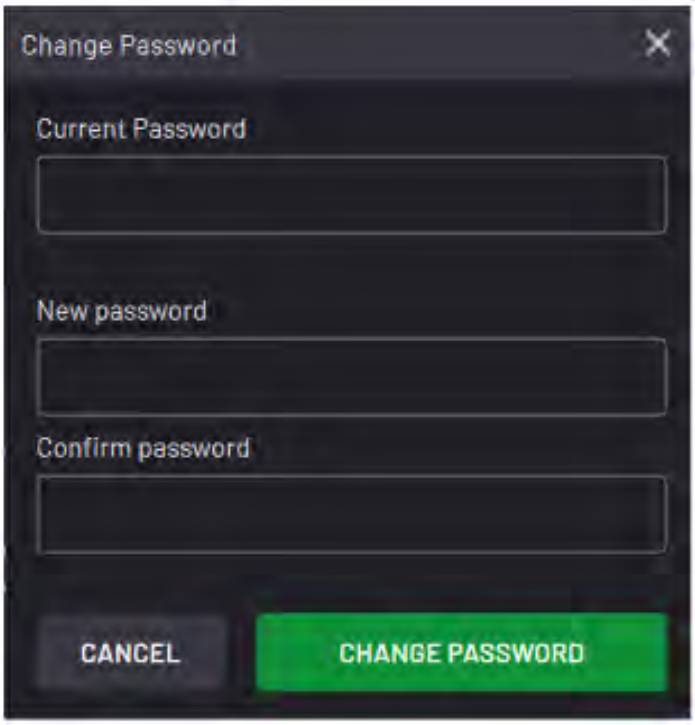
The User Accounts screen is where you set login settings.



The default Admin password is “forgeadmin”. You must change this password when setting up ForgeOS. You can change it again any time later on by tapping **CHANGE ADMIN PASSWORD**.

Enter the current password, enter the new password, then enter the new password again to confirm it.

To save, tap **CHANGE PASSWORD**.



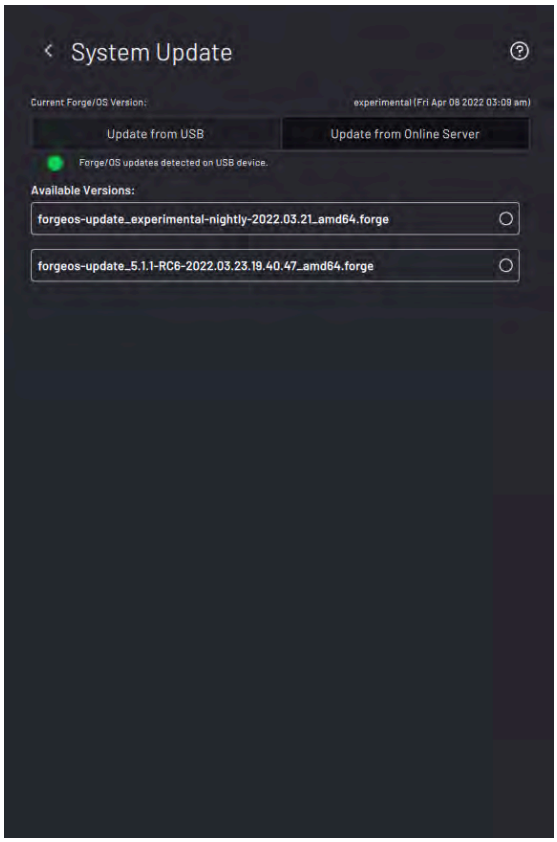
Note:
If you forget your password, contact READY Robotics Support to reset it.

If you enable **Auto login to Operator account**, ForgeOS automatically signs in to the Operator profile when ForgeOS restarts. An Admin can then access the Admin profile by logging out of the Operator profile through the bottom Taskbar.

System Update

The System Update screen is where you update ForgeOS from a USB flash drive or an online server.

Update from USB



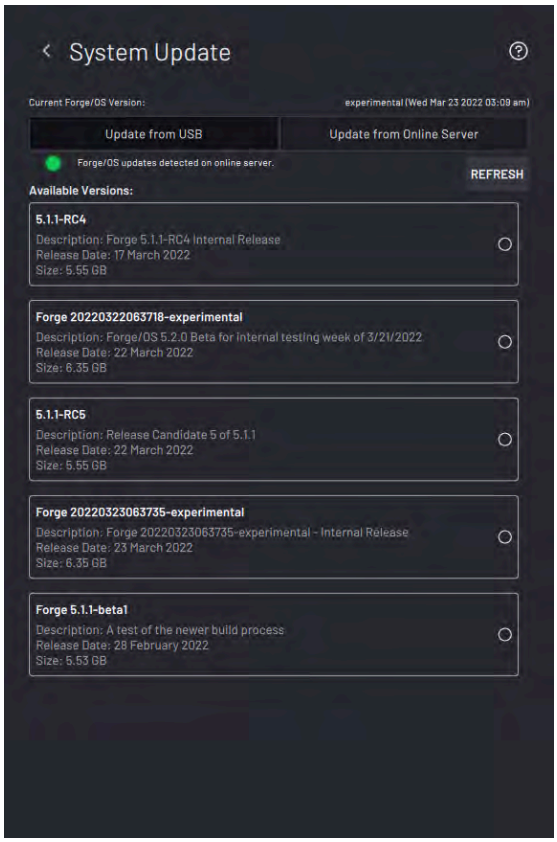
To update with a USB flash drive, download the update file to your USB drive and insert it into your IPC.

In the **Update from USB** tab, tap the available version that you want to update to.

Note:
You must have an eligible Support plan to receive an update file. Update files are too large for the FAT32 file system. **Use an exFAT formatted USB flash drive with at least 8GB of storage**.

Important:
After you transfer an update file to your USB flash drive, make sure you safely eject it from your computer. Early removal of the USB flash drive will corrupt the update file.

Update from Online Server



To update with an internet-connected server, go to the **Update from Online Server** tab. Tap the available version that you want to update to.

Available online updates are based on the system's current version. They include:

- Most recent version of ForgeOS.
- Most recent minor version of the system major version.
- Most recent patch of the system version.

For example, if the system version is 5.3.0 and the latest version is 6.1.1, the options will be something like this:

- 6.1.1 (latest version of ForgeOS)
- 5.6.2 (last version in 5.x)
- 5.3.3 (last version in 5.3.x)

Wait about 20 minutes for the update to finish installing. When it is done, a pop-up asks if you want to restart now or later. Tap **Restart Now** to finish applying the update.

> **Tip:**
> After you log back in, verify that the update completed successfully by checking the **Current ForgeOS Version** in the top-right corner of the **System Update** page or **System Information** page.

> **Important:**
> You may need to update robot configuration files. Refer to READY's "How to Update ForgeOS" guide.

Package Manager

The **Package Manager** is where you install new software components from the Cloud or from a USB flash drive without requiring a new full build of ForgeOS. Components are "packaged" to contain all relevant software and configuration templates. Packages are digitally signed and verified for security and authenticity.

> **Note:**
> If you are a third-party SDK Developer, send your packages to READY Robotics to have them reviewed, tested, and signed before distribution.

Only Admins can manage installed packages, but Operators can still use a package (if the package permits that).

Listed below are the different package types:

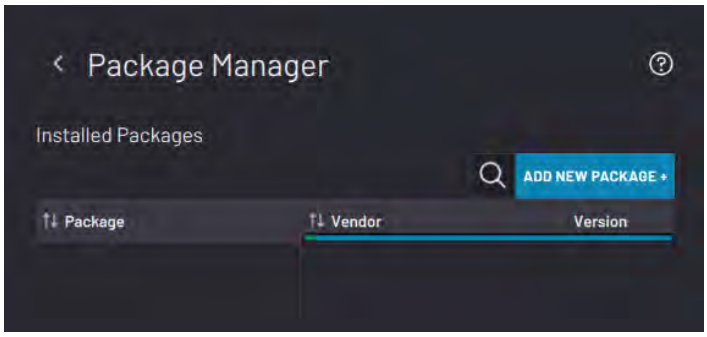
- **Application**: A package that installs a new app into ForgeOS. The app will have one or more icons present on the Home screen.
- **Device**: A package that installs a new device into ForgeOS. The device will appear as a configurable device in the Device Configuration app. Appearance or use in other apps depend on the content of the package. For example, a device may have additional plugin components for the Device Control app and Task Canvas app.
- **Device Configuration**: A package that provides a set of configuration JSON files that customize a device for a specific company and/or product.
- **I/O Driver**: A package that provides access to an IO communication bus (such as an item that appears under "Fieldbus" in Settings).
- **Library**: A package that includes libraries and files that other packages can use.
- **Plug-in**: A package that contains a shared object to extend the functionality of a ForgeOS application (such as new Task Canvas blocks).

In addition to installing packages inside the Forge container, you may install “host packages”. **Host packages** are packages that are installed directly on the host IPC. Decoupled from the Forge platform release process, host packages open the door for many large-scale applications.

Streamline the package installation process with “package bundles”. Instead of having to install several packages individually, **package bundles** allow you to install multiple packages at once with a single install request.

The main Package Manager screen shows a table of installed packages.

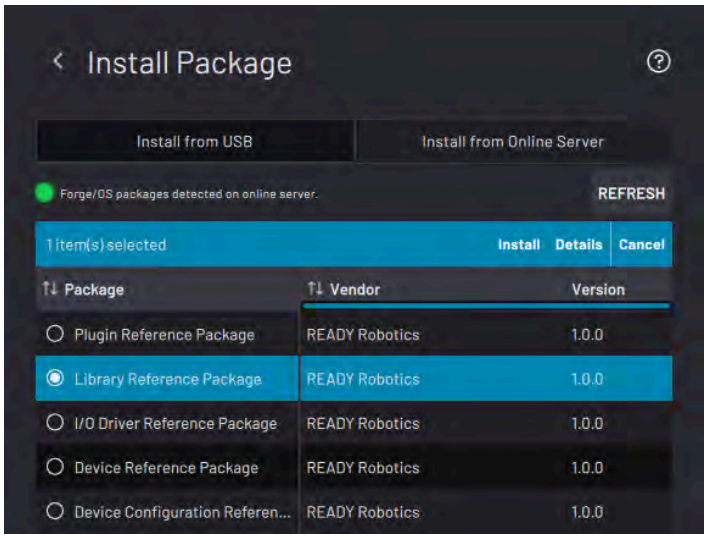
Tap **ADD NEW PACKAGE +** to install a new package.



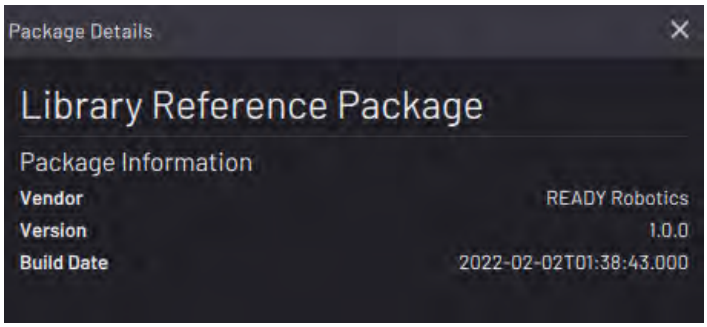
Tap the **Install from USB** tab if you are installing a package from a USB flash drive that you plugged into your IPC.

Tap the **Install from Online Server** tab if ForgeOS is connected to the internet and you are installing a package from the Cloud.

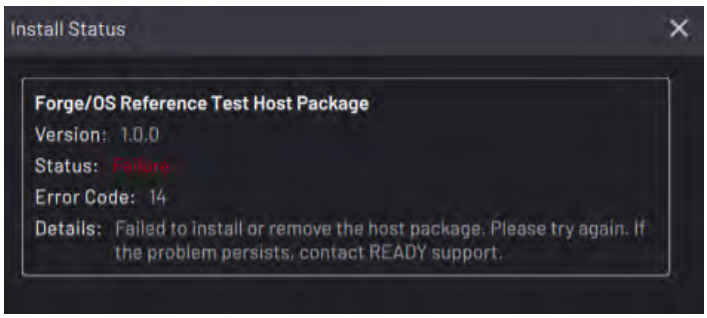
Select a package from the list and tap **Install** in the table header.



Tap **Details** in the table header to view more information about the selected package.



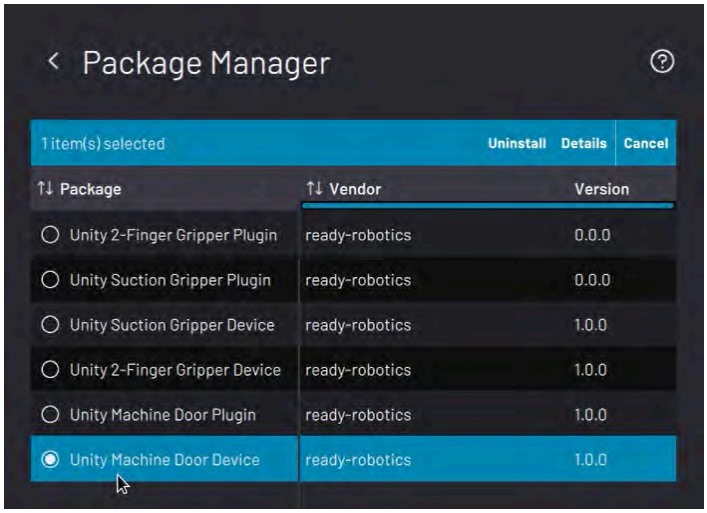
Check the status of the package installation and removal process with the “Install/Uninstall Status” pop-up. In the case of failure, this pop-up displays an error code and details. For example, you will be warned if there's not enough storage space available before installation begins.



> **Tip:**
> If you are installing a package from a USB flash drive, make sure that the USB was safely ejected from the PC it got the file from.

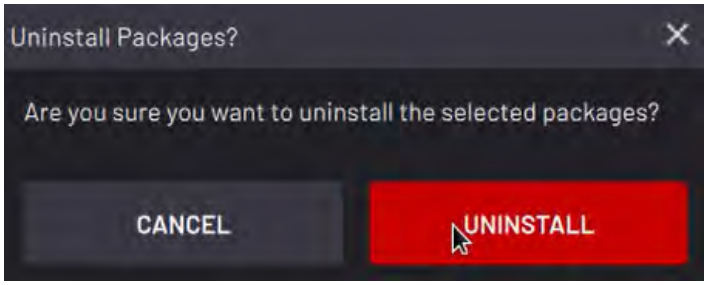
If no errors occur during installation, the new package appears in the Package Manager main screen.

To uninstall a package, select it and tap **Uninstall** in the table header.



> **Note:**
> Previously-existing instances of a package will be automatically removed prior to the installation of the new version.

In the pop-up, tap **UNINSTALL** to confirm the package uninstallation.



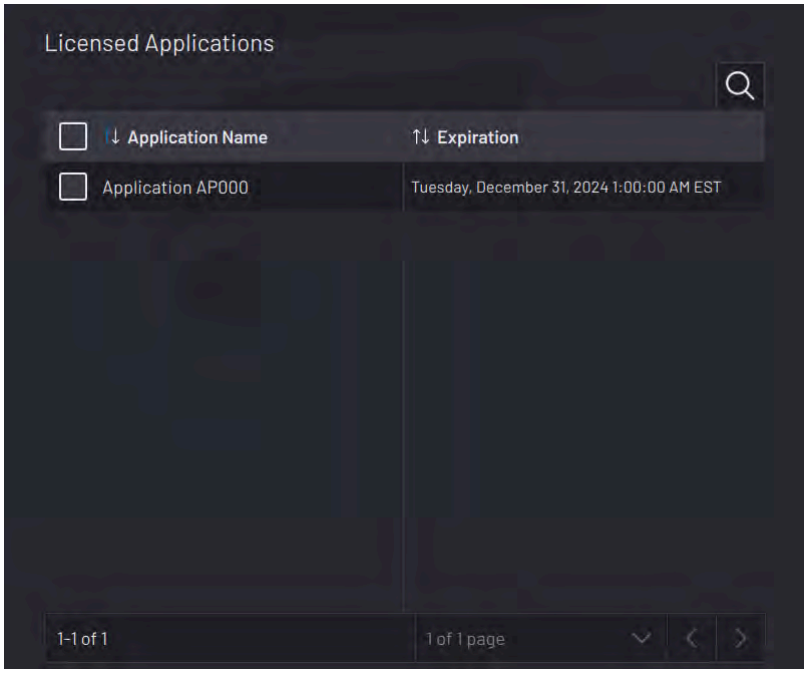
License Information

The License Information screen shows the status of your ForgeOS license. If the license is active, the page shows the license code. If it’s your first time booting ForgeOS, you are taken directly to this page to activate ForgeOS.

See Activating Forge/OS with a License Code *(on page 39)* for license activation steps.

See Deactivating ForgeOS with a License Code *(on page 42)* for license deactivation steps.

Note:
READY can use a web portal to enable a list of applications to appear in the **Licensed Applications** table according to your specific license code. This table will update upon license activation. If a license is already active, the table will update upon license deactivation and reactivation. Once applications appear in the table, sort them by name or expiration date.



Activating ForgeOS with a License Code

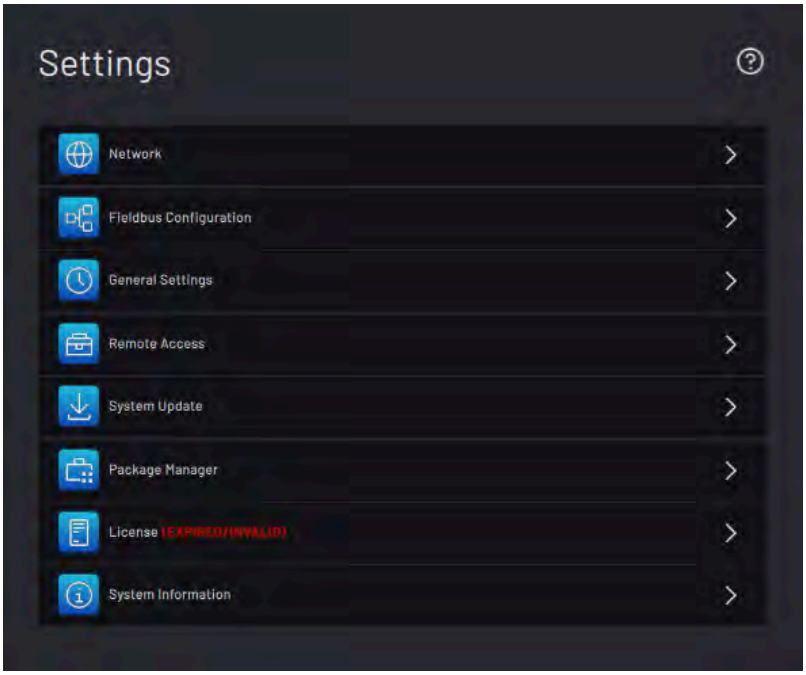
There are two methods to activate ForgeOS: **Online license activation** and **offline license activation**.

The table below lists the requirements for each method.

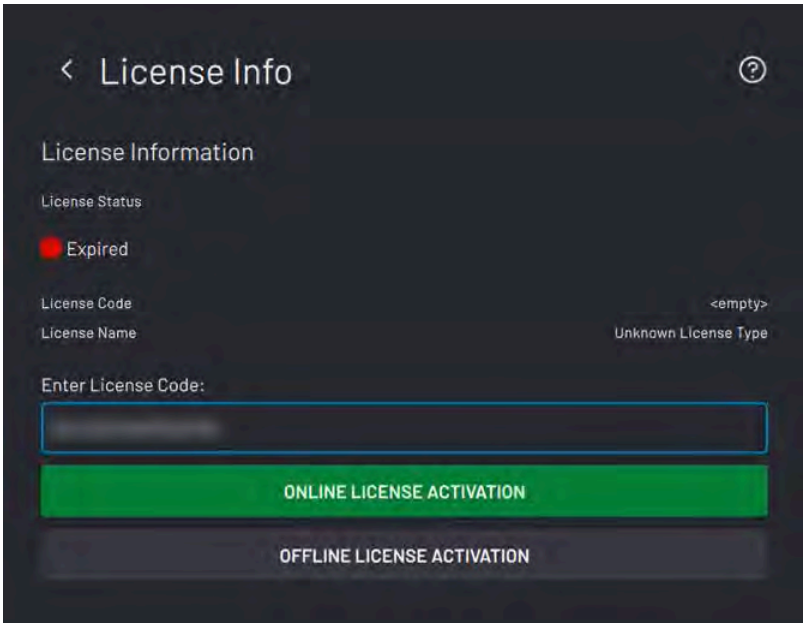
Online License Activation	Offline License Activation
• An internet-connected ForgeOS • A valid ForgeOS license code	• A 2GB or larger USB flash drive • An internet-connected PC • A valid ForgeOS license code

Tip:
Connect a USB keyboard to the port on the bottom of the READY pendant to type in any text field in ForgeOS.

1. On the Settings app main screen, tap **License**.

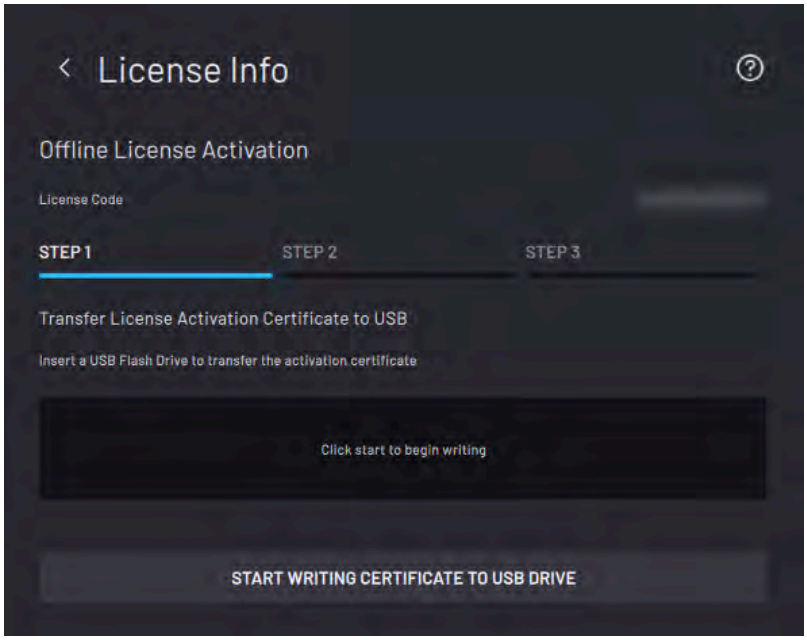


2. Type in your license code.
3. Choose **ONLINE LICENSE ACTIVATION** if ForgeOS is connected to the internet. If not, choose **OFFLINE LICENSE ACTIVATION**.

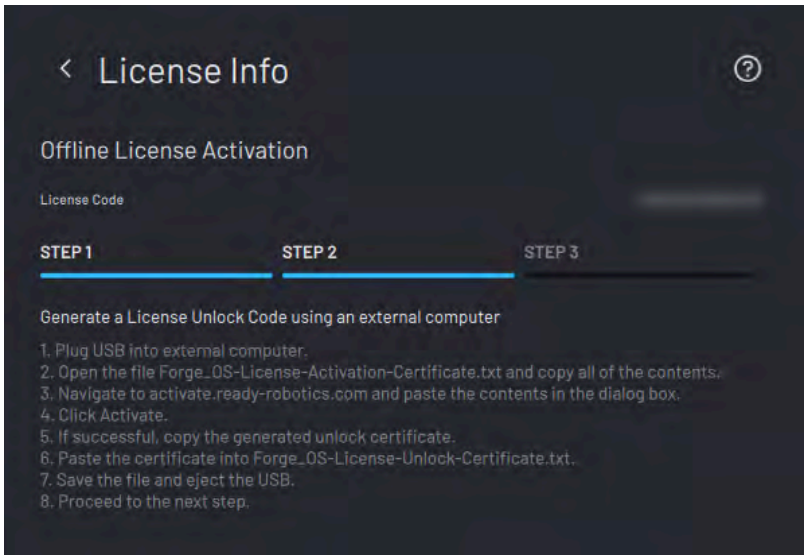


4. If you chose online license activation, you're done!
5. If you chose offline license activation, follow these substeps:

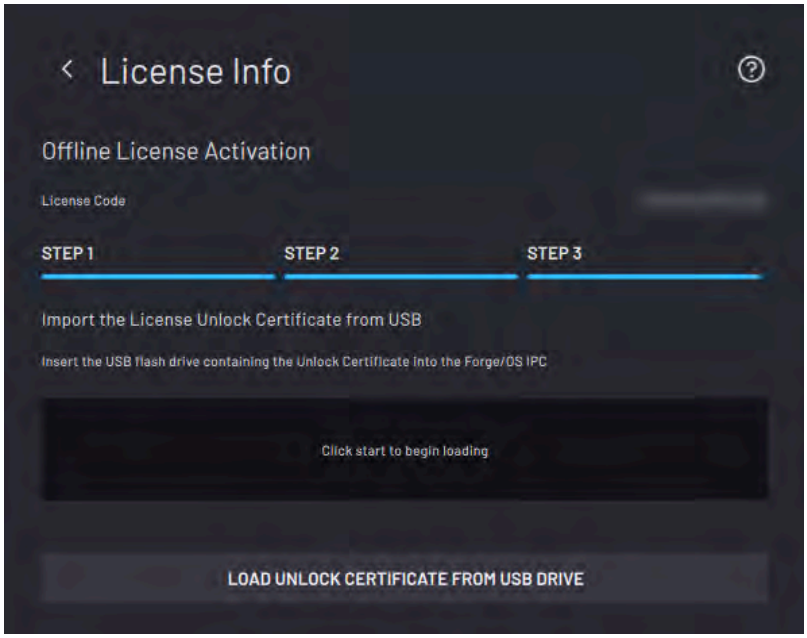
a. Insert the USB flash drive into your IPC. Tap **START WRITING CERTIFICATE TO USB DRIVE**.



b. When the files finish transferring, tap **NEXT**. Follow the instructions on the screen to convert the Activation Certificate to an Unlock Certificate using an internet-connected PC.



c. Insert the USB flash drive back into your IPC. Tap **UNLOAD UNLOCK CERTIFICATE FROM USB DRIVE**.



d. Wait for the file to finish transferring. When the file transfer is complete, remove the USB flash drive and tap **SAVE**.

e. ForgeOS returns to the licensing home screen and shows an active license. If the license status isn't active, restart these license activation steps. Double-check your license code.

Deactivating ForgeOS with a License Code

To transfer a ForgeOS license from one machine to another, first deactivate the license. Then activate it on the new machine.

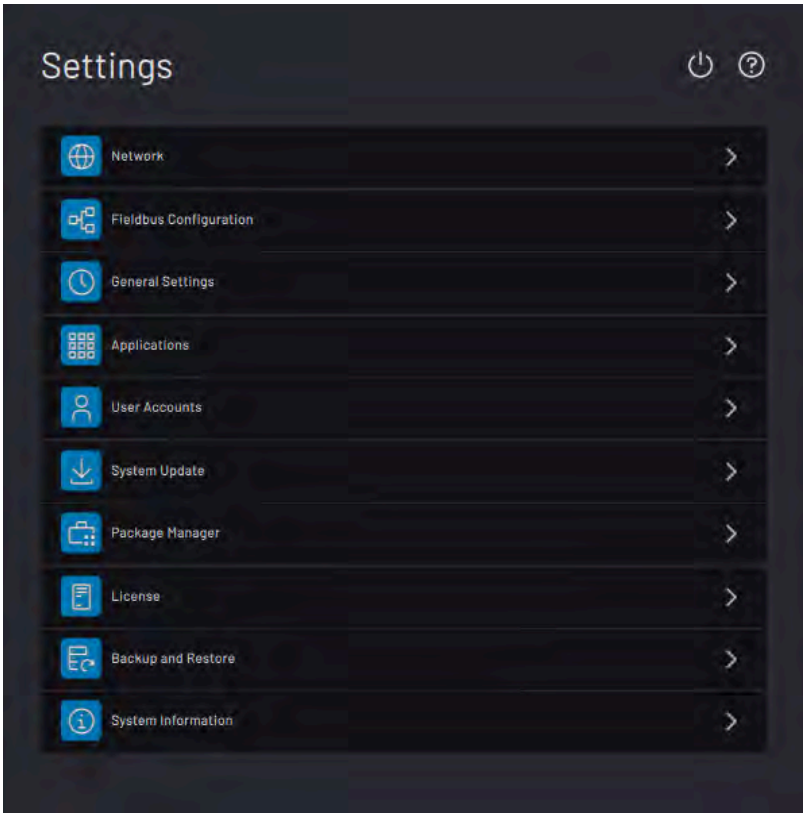
There are two methods to deactivate ForgeOS: **Online license deactivation** and **offline license deactivation**.

The table below lists the requirements for each method.

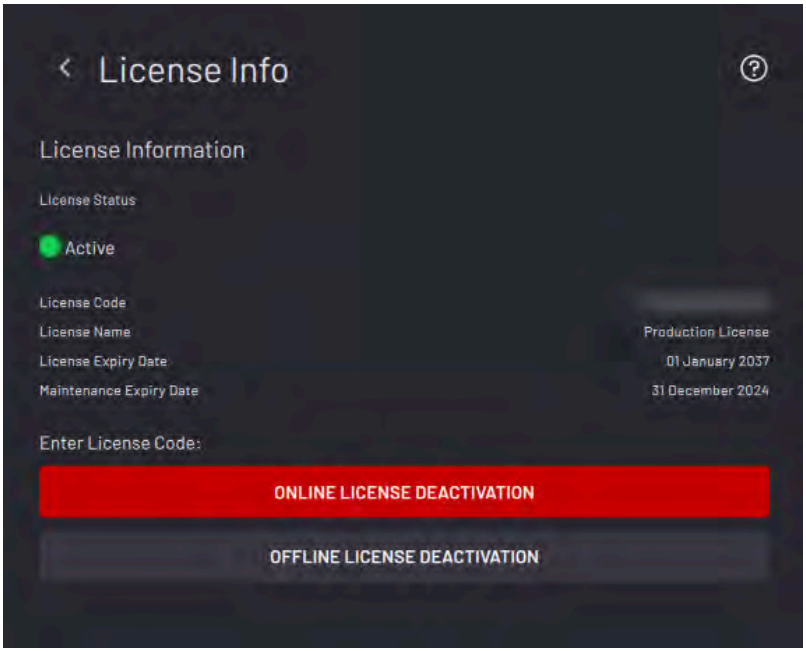
Online License Deactivation	Offline License Dectivation
• An internet-connected ForgeOS	• A 2GB or larger USB flash drive • An internet-connected PC

Note:
If you are unable to deactivate a ForgeOS license, contact READY Robotics Support.

1. On the Settings app main screen, tap **License Info**.

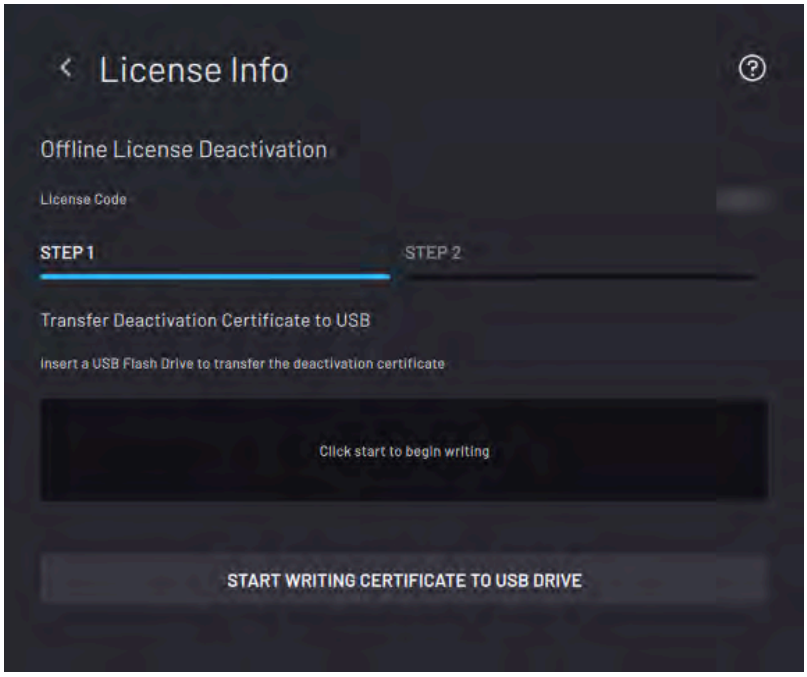


2. Choose **ONLINE LICENSE DEACTIVATION** if ForgeOS is connected to the internet. If not, choose **OFFLINE LICENSE DEACTIVATION**.

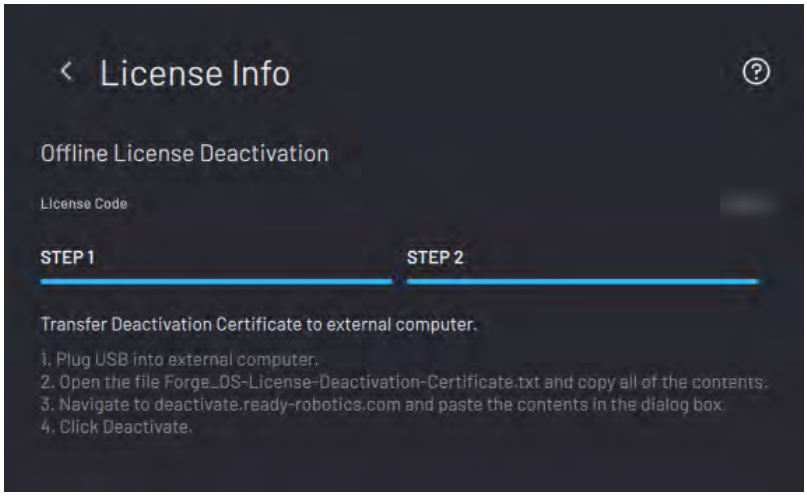


3. In the pop-up, tap **DEACTIVATE** to confirm that you want to deactivate your license.
4. If you chose online license deactivation, you're done!
5. If you chose offline license deactivation, follow these substeps:

a. Insert the USB flash drive into your IPC. Tap **START WRITING CERTIFICATE TO USB DRIVE**.



b. When the files finish transferring, tap **NEXT**. Follow the instructions on the screen to finish deactivating the license using an internet-connected PC.

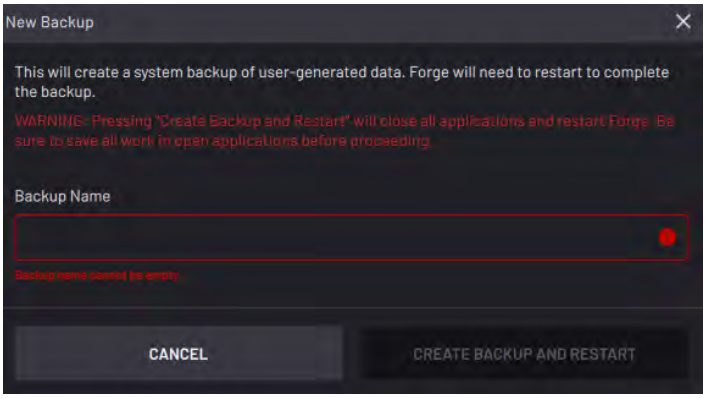


c. Tap **FINISH**.

d. ForgeOS returns to the licensing home screen and shows an inactive license. If the license status isn't inactive, restart these license deactivation steps.

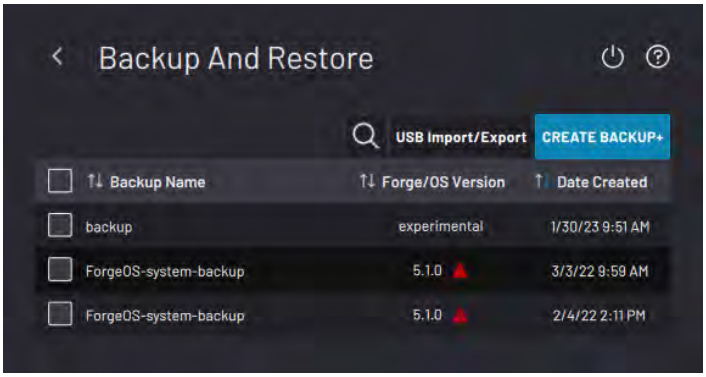
Backup and Restore

To take a backup of all user-generated data (including devices, tasks, rulesets, and system settings), tap **CREATE BACKUP+**. Enter a backup name, then tap **CREATE BACKUP AND RESTART**.

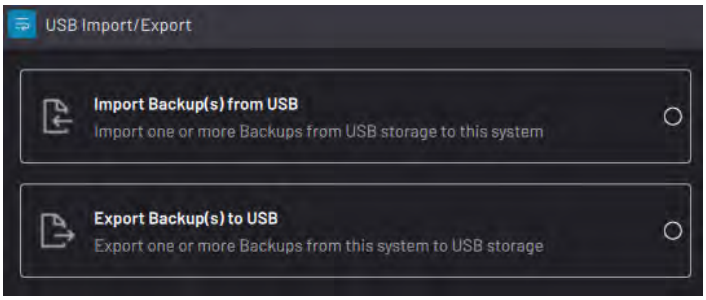


> **Note:**
> Pressing “Create Backup and Restart” will close all apps and restart ForgeOS. Save all your work in all apps before creating a backup and restarting.

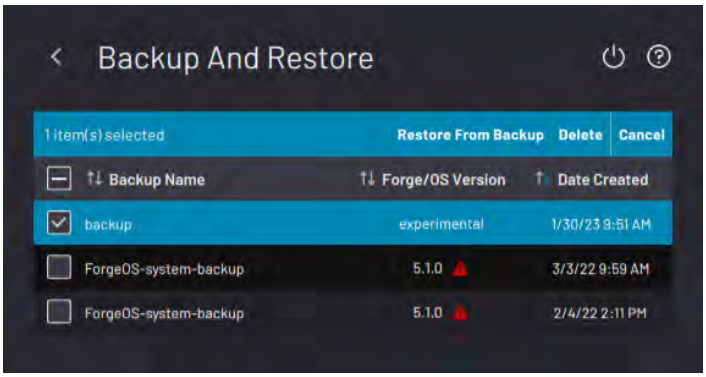
Backups that you create are listed in the table. Sort the table by the backup name, the ForgeOS version, or the date when you created the backup.



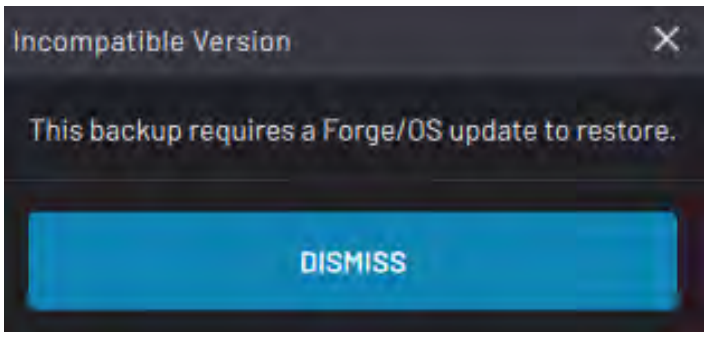
To export or import backup files to or from a USB drive, insert the USB drive into the Forge IPC. Then tap **USB Import/Export** at the top of the table.



To restore your system or another system on the same ForgeOS version, tap **Restore From Backup**. The restore process performs a complete system overwrite with the backed-up data. ForgeOS licenses are not included in the backup and will not be overwritten by a restore.



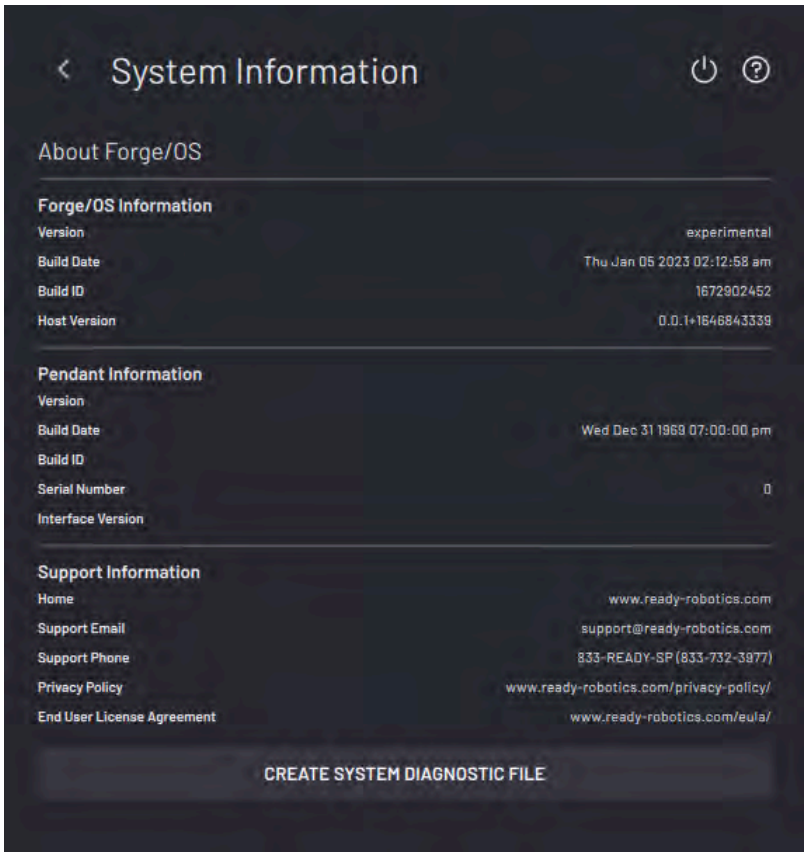
If you see a red exclamation mark next to any of the listed backups, tap it for more information about what's wrong. You may need to update or downgrade ForgeOS to restore the selected backup.



System Information

The System Information screen is where you view the ForgeOS version information, pendant information, and READY Robotics Support information.

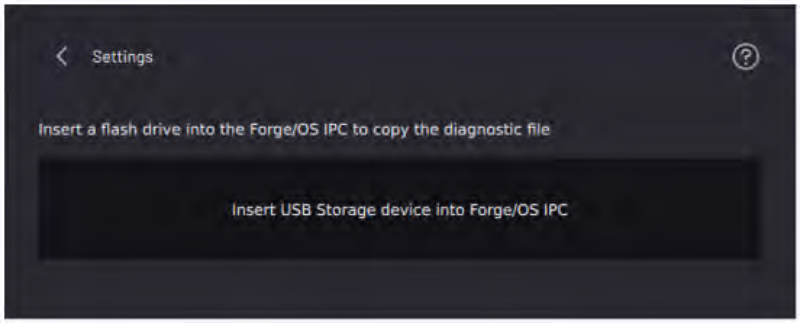
See Creating a System Diagnostic File *(on page 46)* for how to give diagnostic information to READY Support.



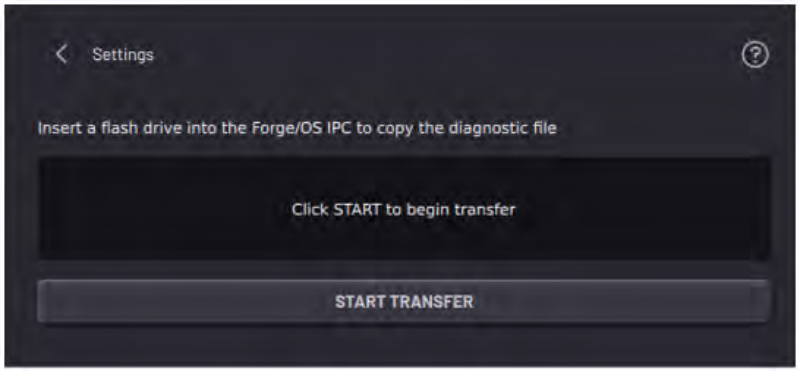
Creating a System Diagnostic File

Follow these steps to create system diagnostic information when you have a problem with your system. Send the files to READY Robotics Support for help.

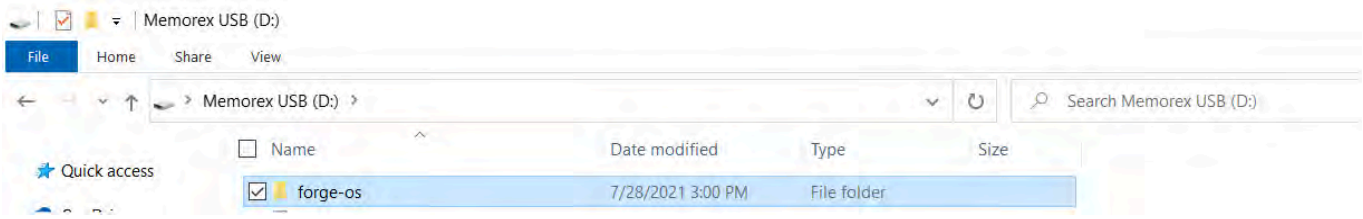
1. In the System Information menu, tap **CREATE SYSTEM DIAGNOSTIC FILE**.
2. Insert a 2GB or greater USB flash drive into your IPC.



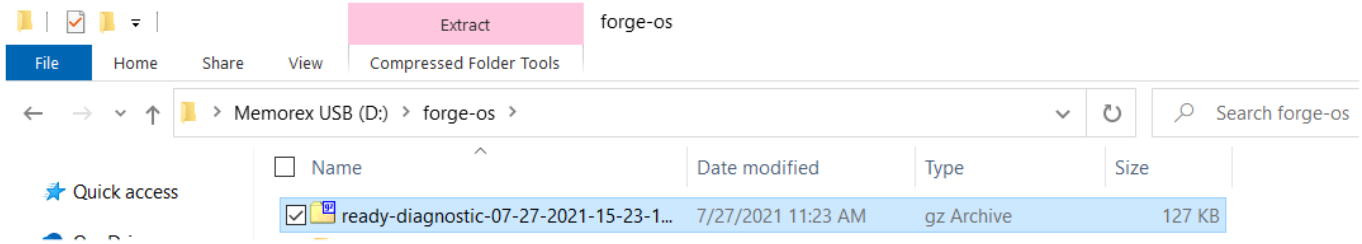
3. Once ForgeOS detects the USB flash drive, tap **START TRANSFER**.



4. Once the file transfer is complete, remove the USB flash drive from your IPC. Insert it into another computer.
5. In the USB storage, go to the newly generated "forge-os" folder.



6. Find the "ready-diagnostic..." archive folder. Send this folder to READY Robotics Support with a description of the issues you are experiencing.



Chapter 6. Device Configuration

The Device Configuration app is used to add and manage devices in ForgeOS. Sign in as an **Admin** to access it. Supported devices include robot arms, PLCs, IO devices, end of arm tooling, and a wide range of fieldbus peripherals.

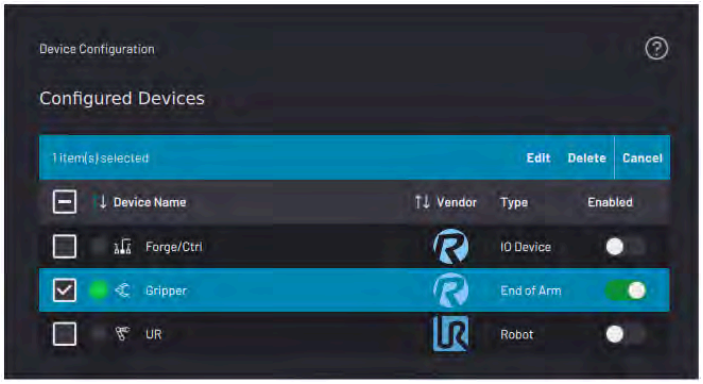
Configured Devices List

The Configured Devices List shows whether added devices are **Enabled** or **Disabled**. Toggle the switches in the table to enable or disable devices. When a device is enabled, the toggle is to the right and the slider is green. Control enabled devices in the Device Control app and Task Canvas app.

To edit a device, select it on the table and tap **Edit** at the top of the table. Only one device may be edited at a time.

To deselect devices, tap the checkbox next to the name or tap **Cancel** in the top right of the table.

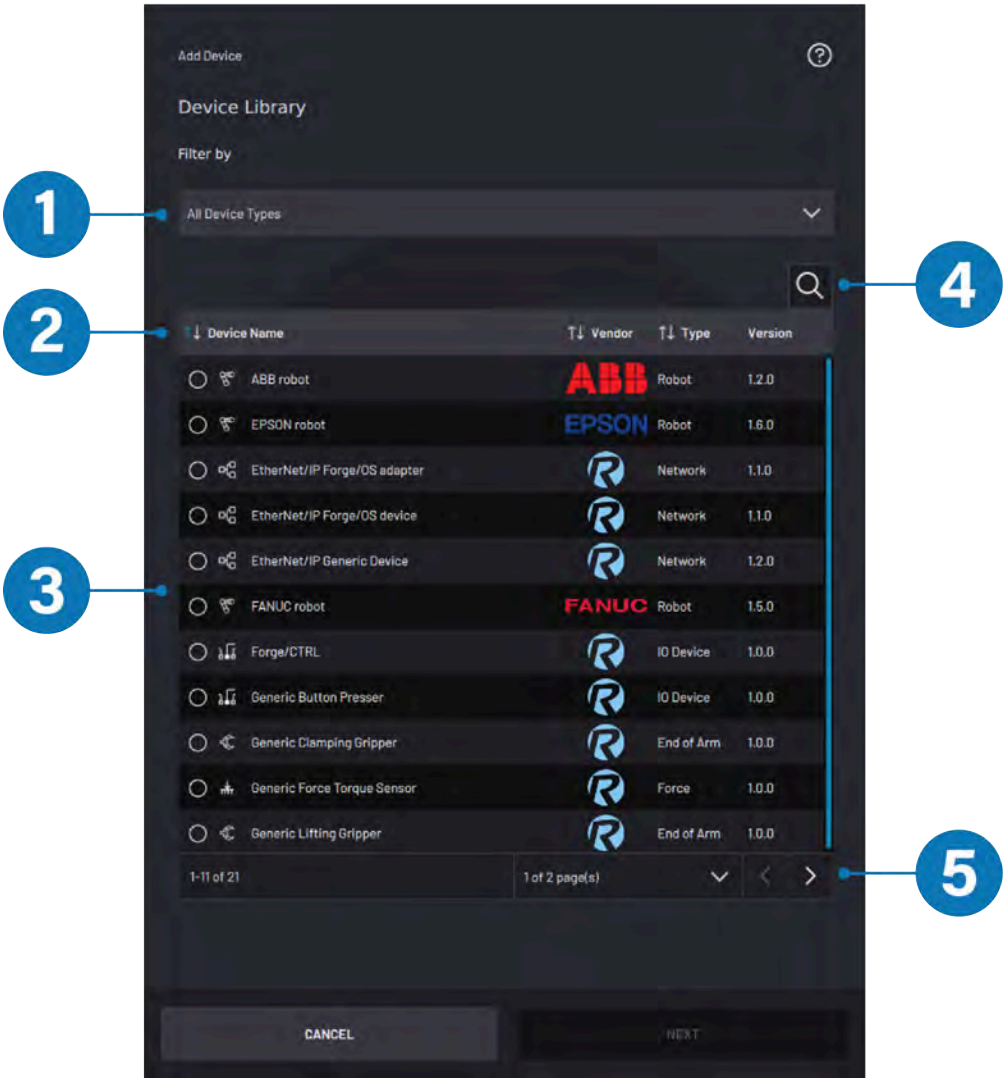
To delete one or more devices, select the devices and tap **Delete** at the top of the table.



Note:
In Task Canvas, device blocks only execute when the device is **Enabled**. If you **Delete** and re-add a device, affected device blocks must be remapped to the new device or removed from the task. See Task Settings *(on page 133)* for more information on device remapping,

Device Library

The Device Library lists all devices supported by ForgeOS. To access the Device Library, tap **New +** in the Device Configuration app. Sort and filter the list to find devices by **Name**, by **Vendor**, or by **Type**.

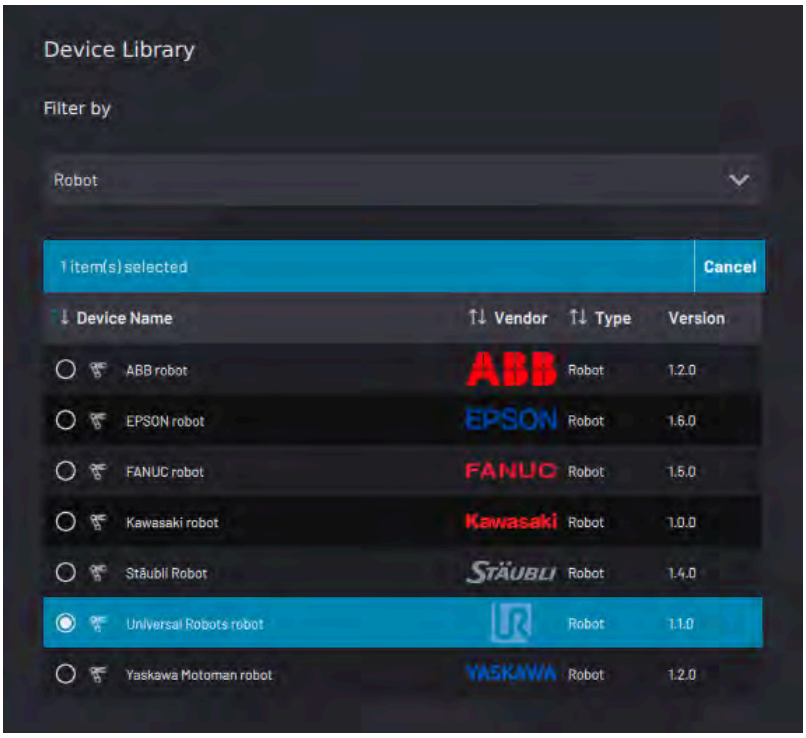


No.	Device Library Feature	Description
1	Filter Selection	Tap the dropdown and select the type of device to add.
2	List Sorting	Tap the sort arrows next to the **Device Name**, **Vendor**, or **Type** columns to sort the list alphanumerically according to that column's category.
3	Device List	Search through the list of supported devices.
4	Search	Tap the search icon (a magnifying glass) and type in the search bar to find a device by name, vendor, or type. Tap the search icon again to close the search bar.
5	List Navigation	Tap the navigation arrows or the page selection dropdown to view more supported devices.

Adding a Robot

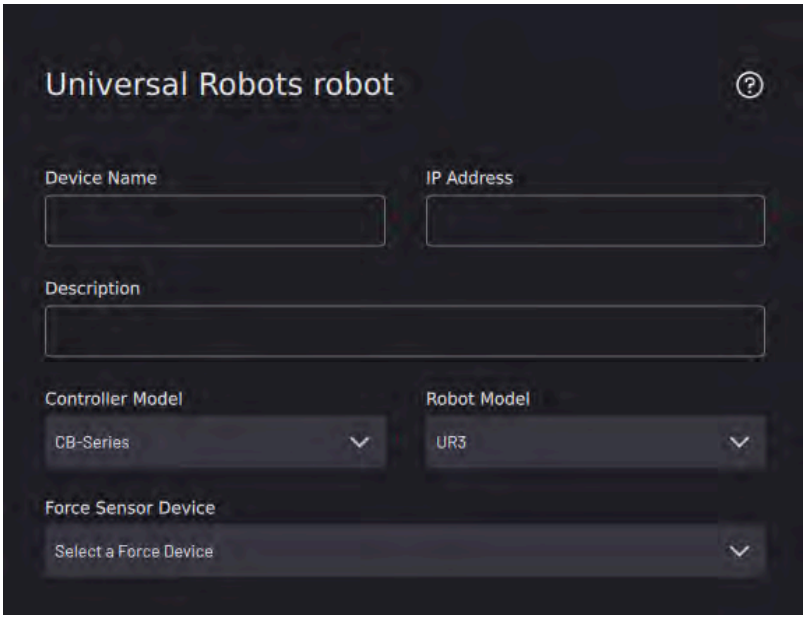
Before adding a robot to the Device Library, install the robot and robot controller following vendor instructions. Go to **ready-robotics.com** to find a startup guide for your robot controller.

1. In the Device Library, select the type of robot that you are using. Then tap **NEXT**.

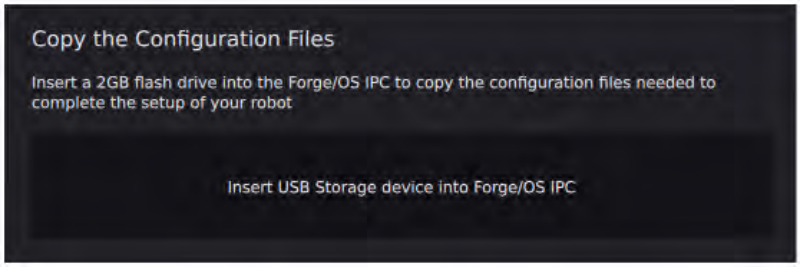


2. Type in a **Device Name** and the device **IP Address**. A **Description** is optional. Then select your **Robot** and **Controller** models. If you are using a force sensor device, select it from the dropdown. The dropdown lists force sensor devices that are already configured on the ForgeOS system.

> **Note:**
> The force sensor selection applies **only** to a force sensor that is attached to the end of the robot arm. If you are using a force sensor device elsewhere in the workcell, set that up as a separate device unrelated to the robot arm.



3. For some robot brands, you may see a prompt to insert a 2GB USB flash drive. If you do, insert a flash drive into your IPC and follow READY startup instructions to transfer files to the robot. Find vendor-specific start up guides on READY's Support page at**ready-robotics.com**.



> **Important:**
> Each time you update ForgeOS, you should update the robot configuration files. For robots that need configuration files, follow update instructions at ready-robotics.com.

4. Tap **SAVE**. Wait for ForgeOS to connect with the robot.

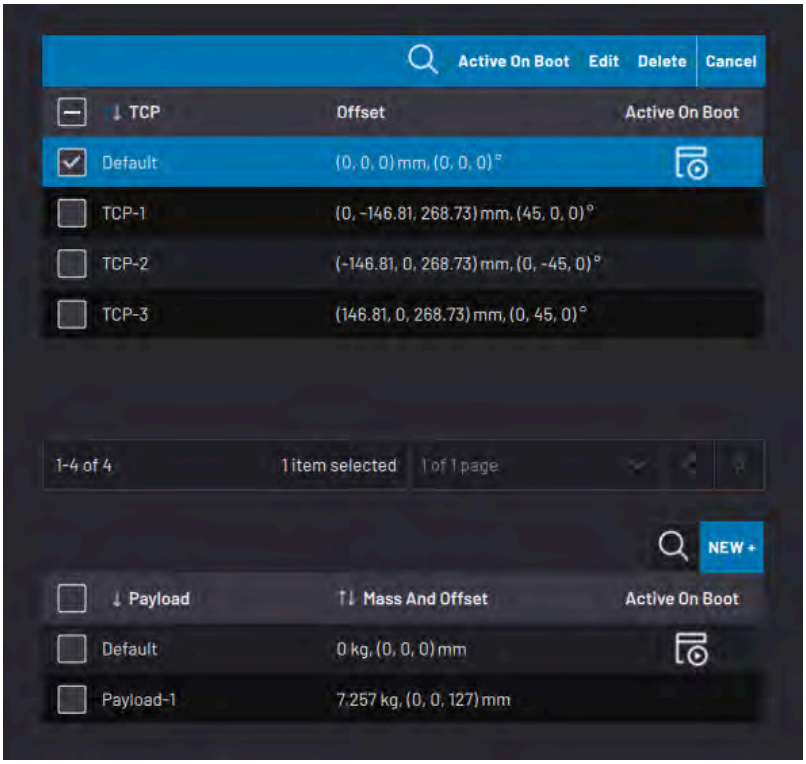
> **Note:**
> The SAVE button is only available when all required fields are filled in.

5. Add all the **Tool Center Points (TCPs)** and **Payloads** that you expect to use in your task. Select one TCP and one Payload to be **Active on Boot** (which means that they will be automatically selected when you log in).

> **Note:**
> Return to this screen at any time after device setup to add, modify, and remove TCPs and Payloads.

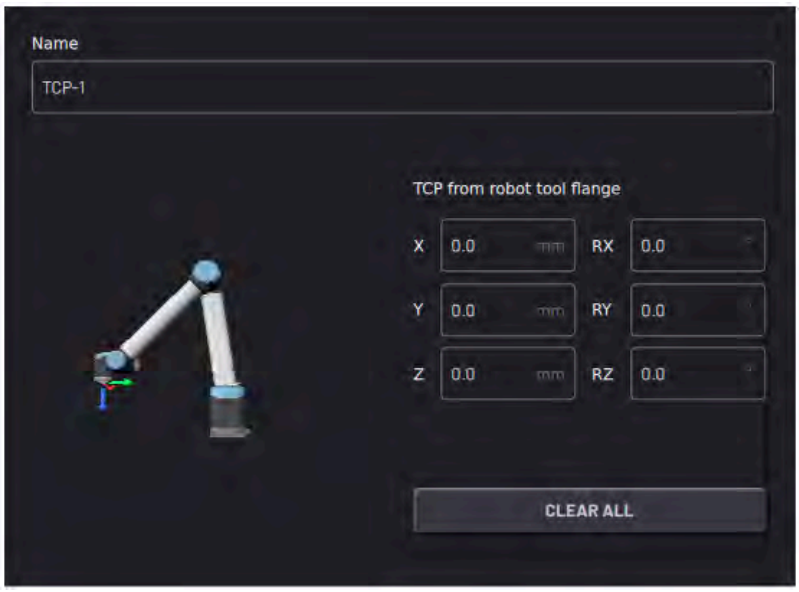
Note:
Some robots require an extra tool loading procedure on the native robot pendant to resolve errors after adding, modifying, and removing TCPs and Payloads on the READY pendant. Refer to the knowledge base at ready-robotics.com.

Note:
You can add a maximum of 100 TCPs and Payloads.



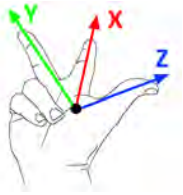
a. The TCP is the exact translational and rotational difference between the robot tool flange (default TCP) and the tip of the end effector. To add a TCP, tap **NEW +** at the top-right corner of the TCP table. Enter a name. Then enter the translational and rotational offset values relative to the default TCP. Look at the robot rendering in the left side panel for reference. Rotate the view by dragging one finger across the window. Zoom in or out by using two fingers on the window in a pinching motion.

Note:
The default TCP is at the robot's tool flange.



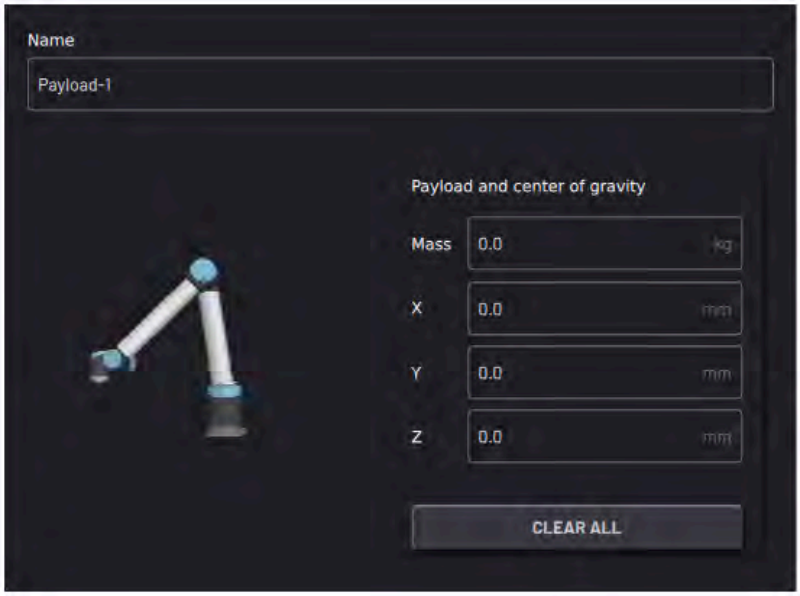
Note:
The values for X, Y, and Z represent the TCP's position with respect to the default TCP in Cartesian coordinates. RX, RY, and RZ represent the TCP's rotational offset from each of the axes on the default TCP. To find the rotational offset values, find the rotation needed around the original X-axis (RX). Then find the rotation needed around the original Y-axis (RY). Last, find the rotation needed around the original Z-axis (RZ).

Tip:
Use your right hand to visualize XYZ coordinates and to find the direction of positive rotations. Point your thumb in the direction of the positive axis (direction of the arrow). The direction that your fingers curl is the positive direction of rotation.



b. The payload is the mass and center of gravity position relative to the robot's default TCP. To add a payload, tap **NEW +** at the top-right corner of the payload table. Enter a name and the mass and offset values relative to the tool flange. Look at the robot rendering in the left side panel for reference. The payload is represented by a sphere at the coordinates you enter. The larger the mass, the larger the sphere.

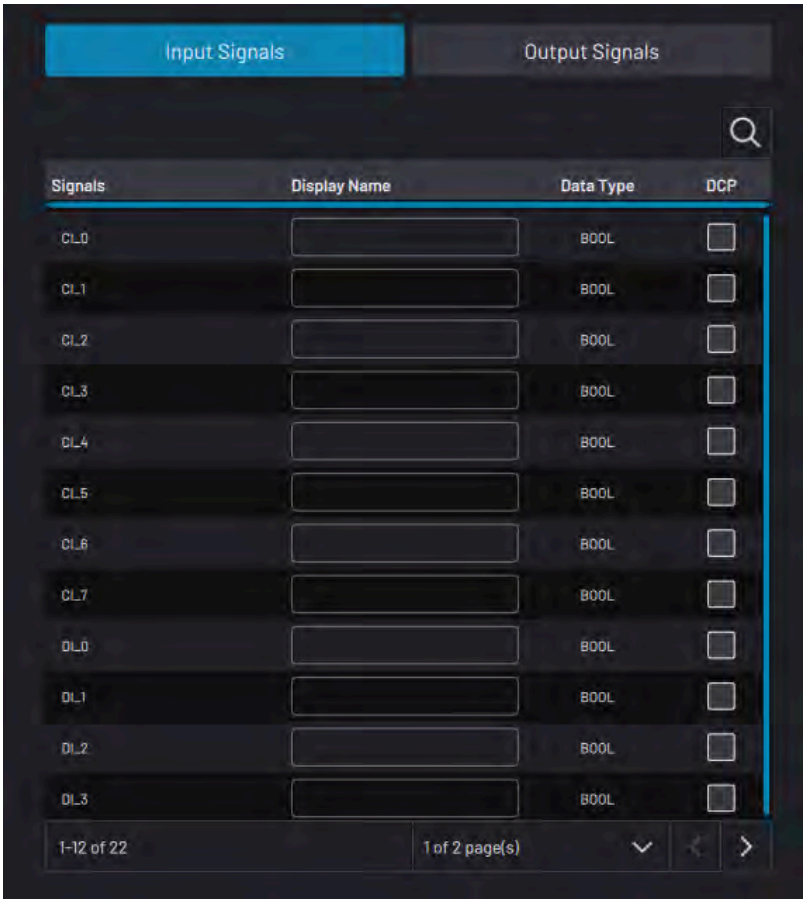
Note:
The default payload is a mass of zero, with the center of mass at the flange.



c. Tap **NEXT**.

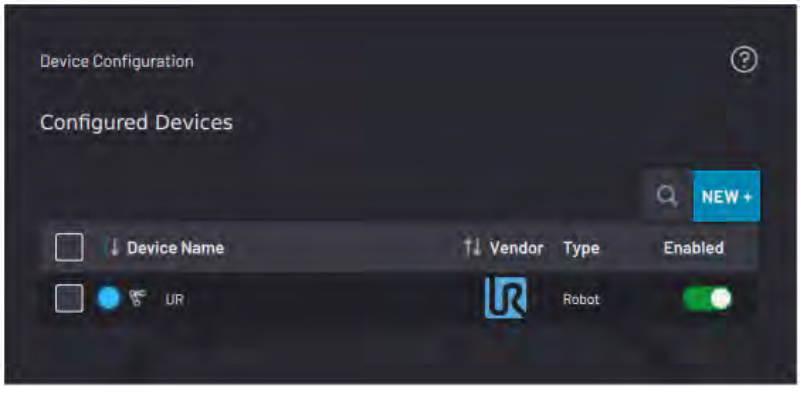
6. If you have devices wired into your robot controller's Input/Output (I/O) signals, follow these substeps:

> **Note:**
> Return to this screen at any time (by selecting the device in Device Configuration and tapping **Edit**) to modify I/O configuration.



a. Enter a **Display Name** (i.e. “Open Machine Door”, “Open Pneumatic Vise”, or “Start Machining Cycle”) to show what each configured signal does.
b. If you want a signal to appear in the device's Device Control page, tap the **DCP** checkbox next to that signal.
c. Tap **SAVE**.

7. ForgeOS returns to the Device Configuration home screen. Make sure that your robot appears on the configured devices list and that it is **enabled**.

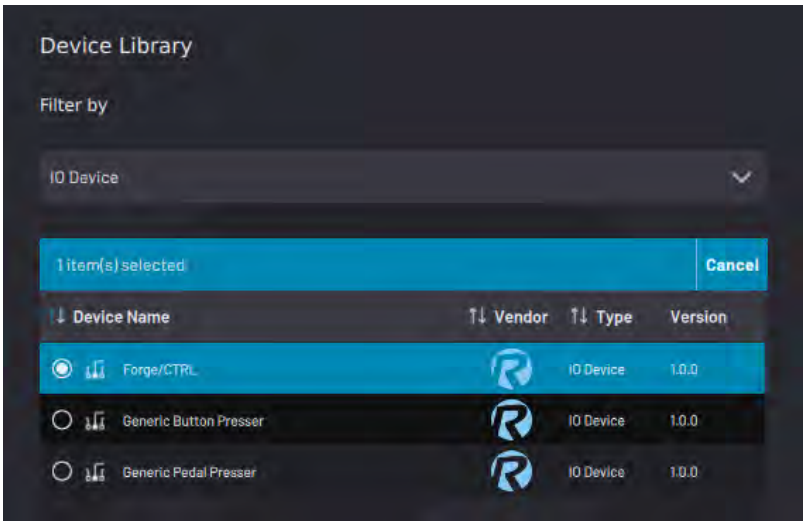


Note:
A device is **enabled** when its switch is green and toggled to the right.

Adding a Forge/Ctrl

The Forge/Ctrl comes with an internal PLC that gives you access to 24V digital I/O, 4mm pneumatic ports, and 6mm pneumatic ports. To access the digital I/O and pneumatic features, first add the Forge/Ctrl from the Device Library.

1. In the Device Library, select **Forge/CTRL** (I/O Device type). Then tap **NEXT**.



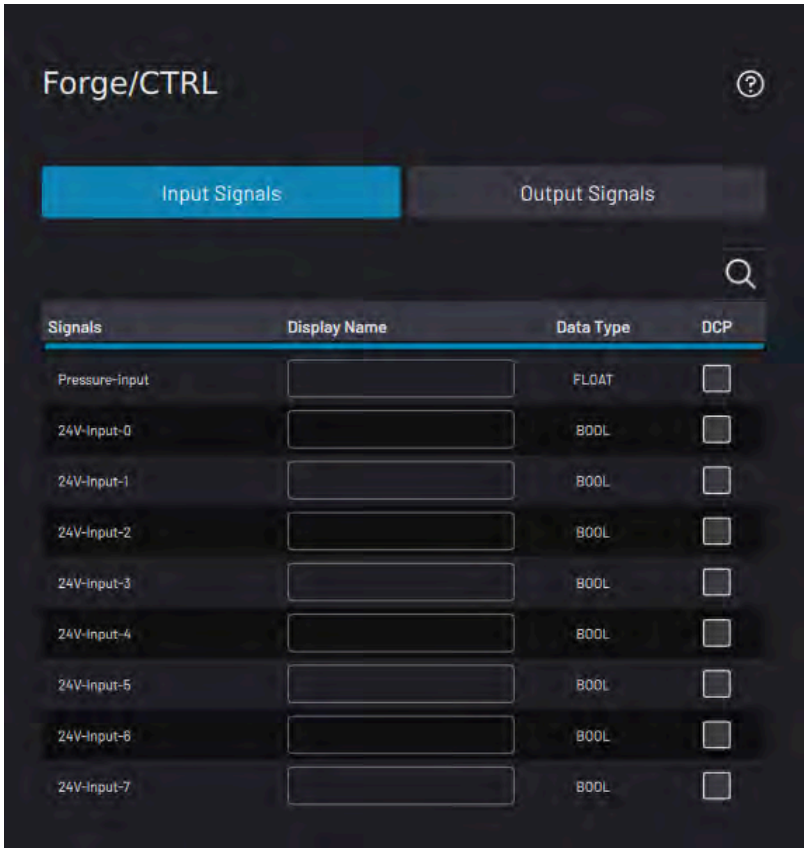
2. Type "Forge/Ctrl" in the **Device Name** field and add a **Description** (optional). The **IP address** is **172.16.255.252**. Then tap **NEXT**.

3. Configure any **Input/Output** (**I/O**) signals you want to control in the Device Control app.

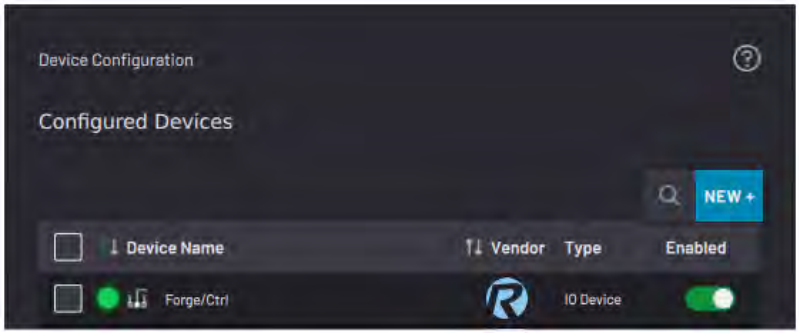
    a. Tap the tabs at the top of the screen to toggle between **Input Signals** and **Output Signals.**

    b. Enter a **Display Name** (i.e. "Suction Gripper Open", "Finger Gripper Close", or "Button Presser Actuate") to signify what each configured signal does.

c. If you want a signal to appear in the device's Device Control page, tap the **DCP** checkbox next to that signal.



4. Tap **SAVE** to return to the Device Configuration home screen. Make sure the Forge/Ctrl appears on the configured devices list and that it is **enabled**.



Note:
A device is **enabled** when its switch is green and toggled to the right.

Adding a Rockwell Automation Logix PAC

In ForgeOS versions prior to 5.3.0, Rockwell Automation Logix PACs connected to the **EtherNet/IP ForgeOS Adapter** network device. This provided a generic set of 132 bytes of input/output for you to define. Even though a usable connection was established, it required effort to map all process data and functions.

Now, the READY Robotics **AOP** (**Add On Profile**) streamlines this process. The AOP establishes a special Ethernet/IP Adapter connection with a defined data map, creating a deeper integration with Logix PACs than was possible with stand-alone EDS files. For an example of how pre-defined signal mapping can ease setup, see Adding a Remote Control Device *(on page 76)*.

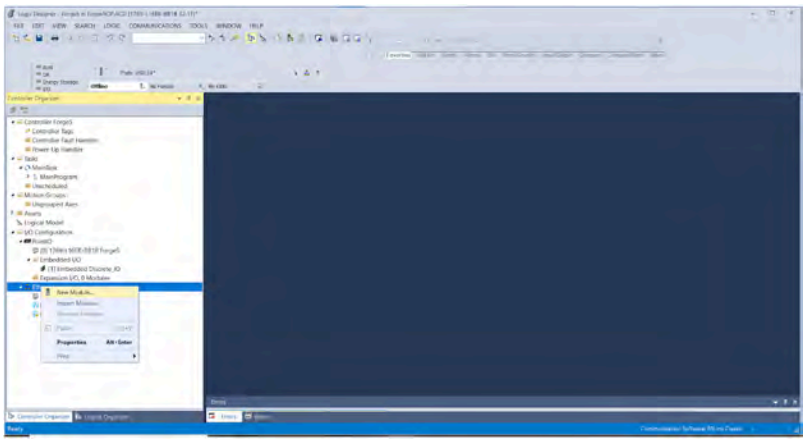
> **Note:**
> The AOP is available for download on the READY Support site.

After you install the READY Robotics AOP in Studio 5000, follow the steps in this section to connect to Forge/OS and create a **Rockwell Automation Logix PAC** device.

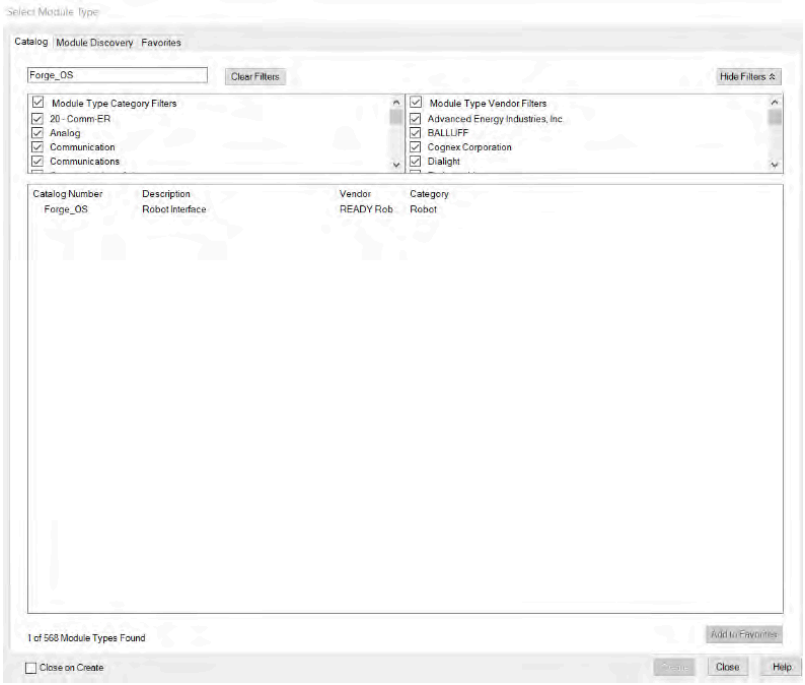
> **Note:**
> Before creating a Rockwell Automation Logix PAC device, first add an Ethernet/IP fieldbus interface in the Settings app. See Fieldbus Configuration *(on page 30)* to learn more.

Once you create the device, you will be able to send and receive data to and from the Logix PAC through Check and Set blocks in Task Canvas.

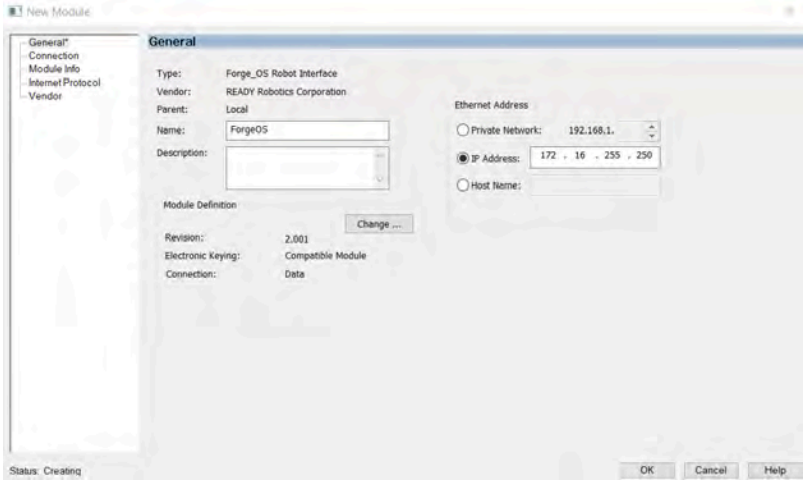
1. In Studio 5000, create a new project.
2. Right-click the **Ethernet** menu and select **New Module**.



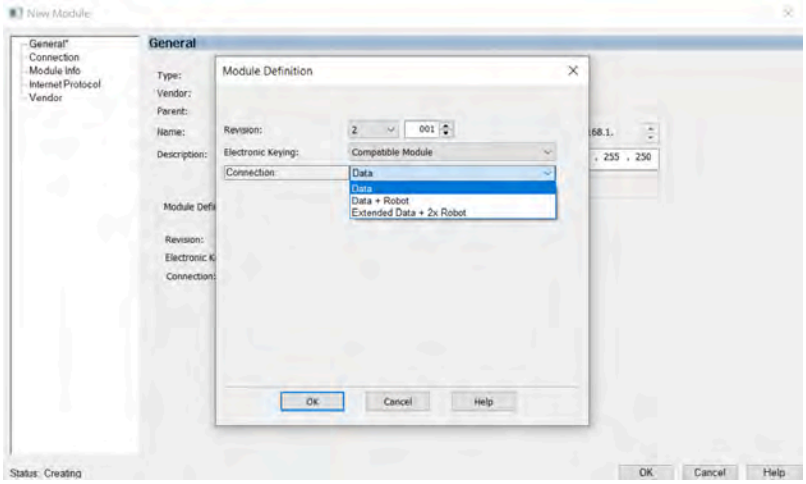
3. Select **Forge_OS** from the module list.



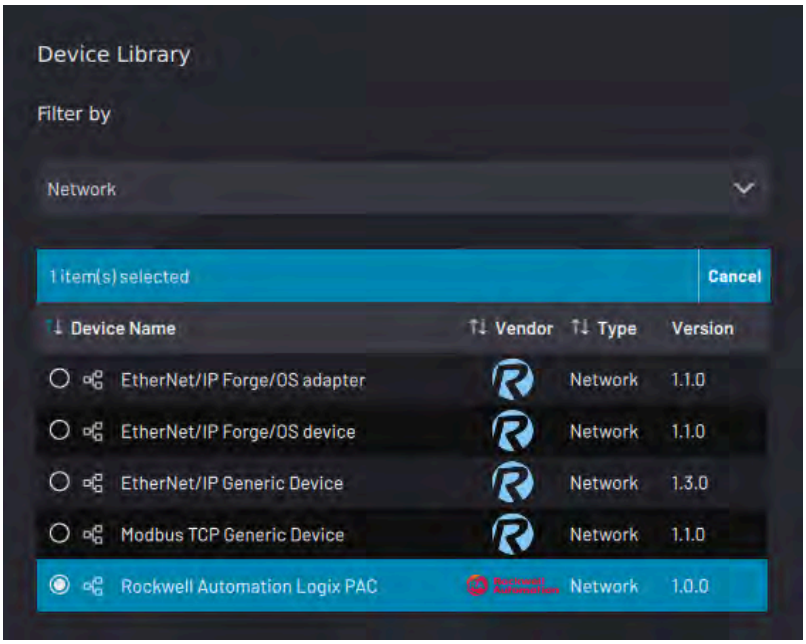
4. Give the Device a **Name**. Then type in the **IP Address** that Forge/OS is running on.



5. Click **Change** under the module definition to select a data model type:
    - **Data**: Transfer integer (INT), unsigned integer (UINT), and float task data.
    - **Data + Robot**: Do everything that the "Data" type can do, plus transfer robot data (such as robot status, current robot position, TCP translational/ rotational velocities, and joint velocities).
    - **Extended Data + 2x Robot**: Do everything that the "Data + Robot" type can do, but for two sets of robot data.



6. Click **OK** to save your module definition change. Click **OK** again to save the module.
7. In the Forge/OS Device Library, select **Rockwell Automation Logix PAC** (Network device type). Then tap **NEXT**.



8. Type in a **Device Name**. A **Description** is optional. Then follow the below substeps before tapping **NEXT**.

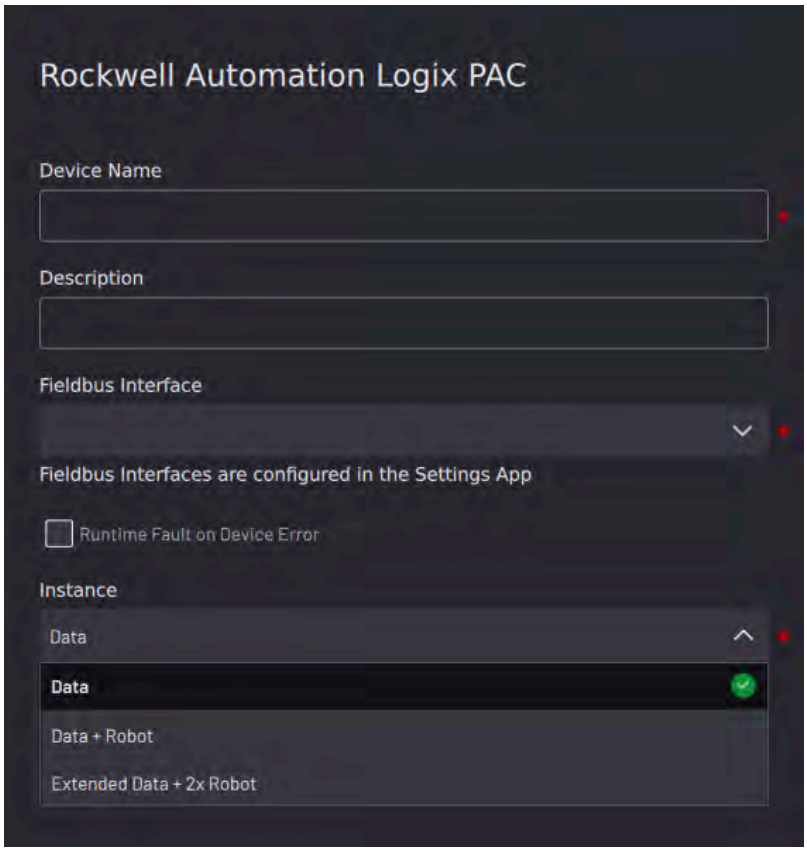
    a. In the **Fieldbus Interface** dropdown, select a configured fieldbus interface.

    b. Select (or leave de-selected) the checkbox labeled "**Runtime error on device disconnection**". This checkbox allows you to configure system behavior if the device disconnects. When this checkbox is active and the PAC is not connected, the device enters a “Device is Disconnected” error state. This error state stops the task if it is running or prevents a task from starting. Currently, as long as the fieldbus driver is functioning properly, the device will never go into an error state.

    c. In the **Instance** dropdown, select the data model type that you chose in Studio 5000.

    > **Note:**
    > Depending on which data model type you choose, you may need to fill in an optional **Robot** dropdown(s). This dropdown includes all robot devices on the system. Selecting a robot here

automatically pulls data from that device and maps it to the specified Ethernet/IP registers. The units specified in the Settings app will be used when sending this data.

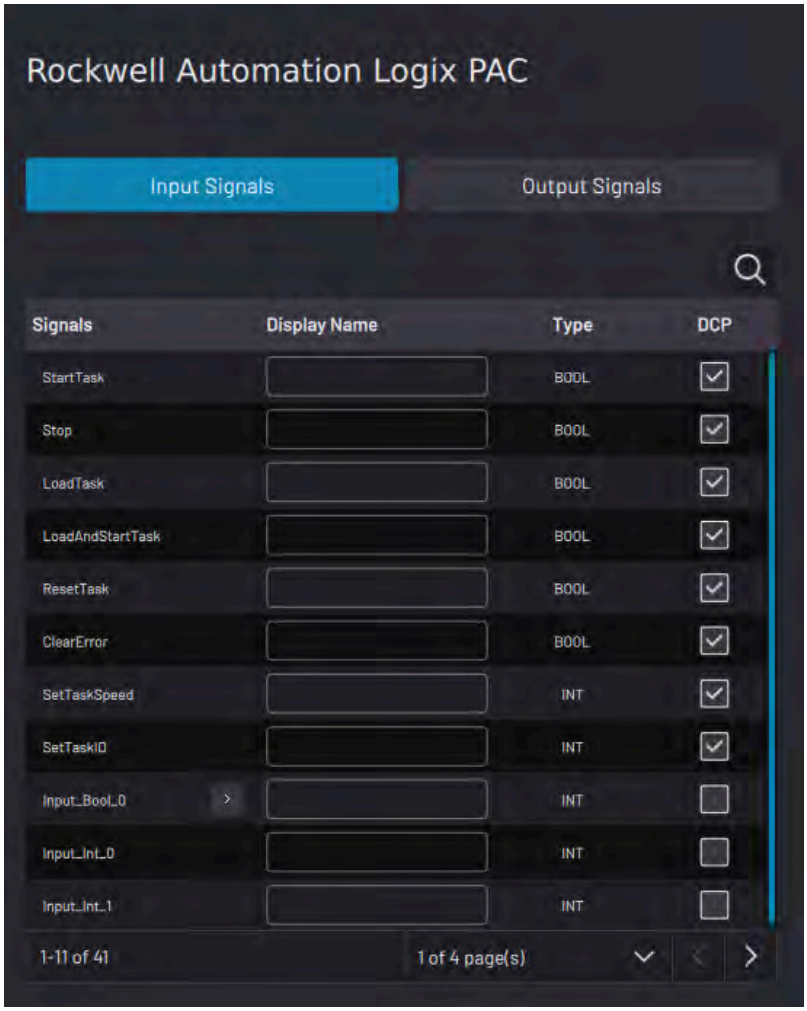


9. Configure any **Input/Output (I/O)** signals that you want to view in the Device Control app.

    a. Tap the tabs at the top of the screen to toggle between **Input Signals** and **Output Signals**.

    b. (Optional): Enter a Display Name to signify what each configured signal does.

c. If you want a signal to appear in the device's Device Control page, tap the **DCP** checkbox next to that signal.



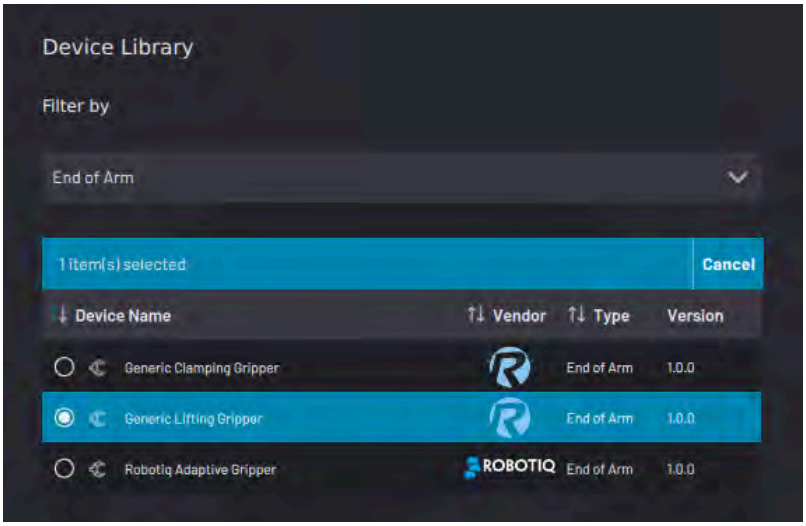
10. Tap **SAVE** to return to the Device Configuration home screen. Make sure the PAC appears on the configured devices list and that it is **enabled**.

Adding a Lifting Gripper

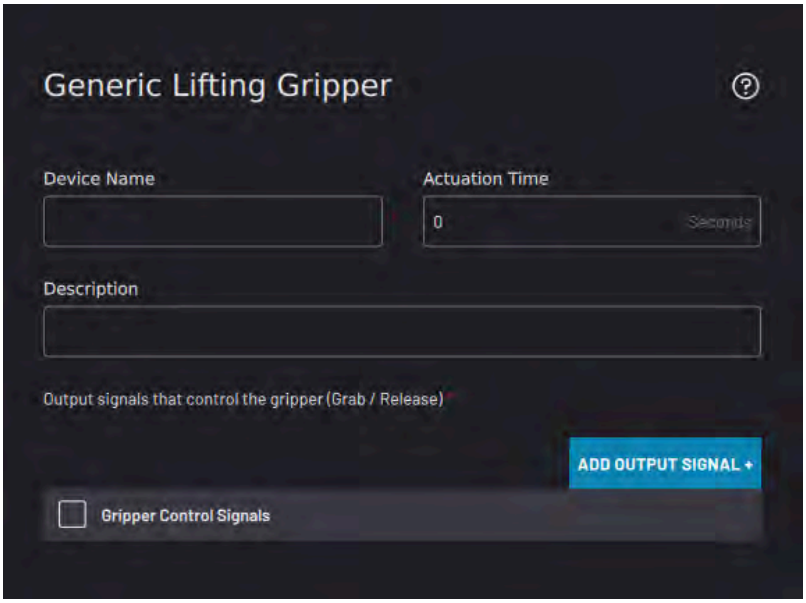
Follow the instructions in this section to configure a generic lifting gripper (such as a suction gripper or magnetic gripper).

Note:
To use a lifting gripper without robot I/O, first configure the IPC and I/O devices. See Fieldbus Configuration *(on page 30)* settings to learn more. Then select the signals for controlling air to the gripper.

1. Select **Generic Lifting Gripper**. Then tap **NEXT**.



2. Enter a **Device Name**. A **Description** is optional.

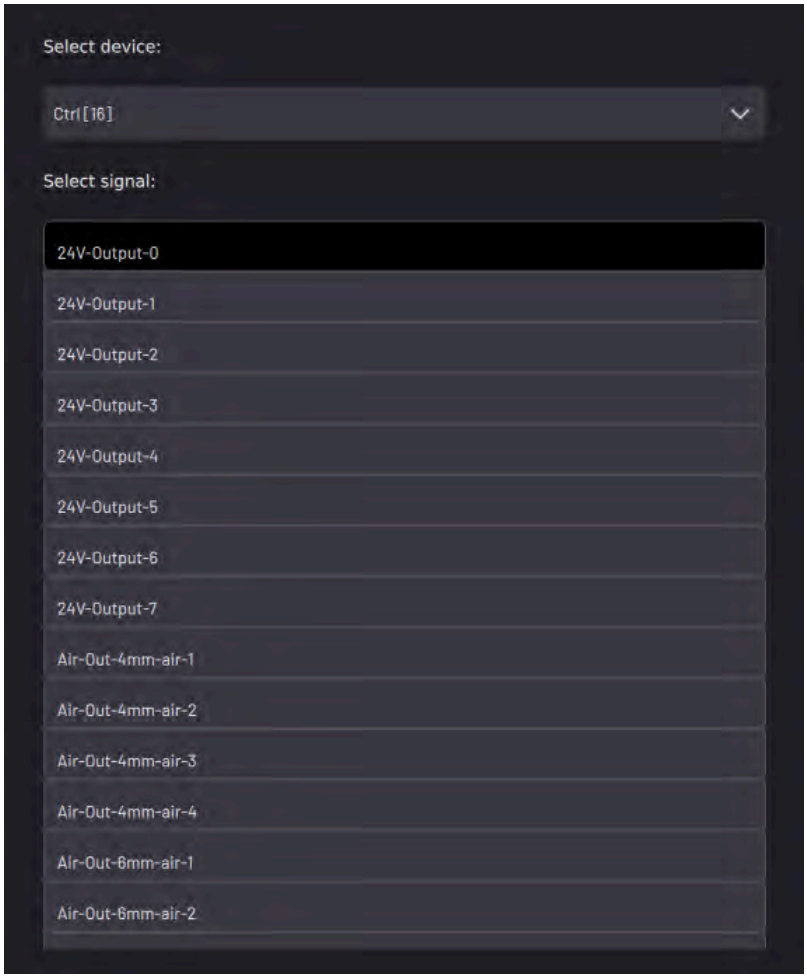


3. To make sure your device has enough time to actuate on and off in a task, enter the device's estimated **Actuation Time**. The default value is zero seconds.

> **Note:**
> In Task Canvas, that device's control blocks prompt you to choose whether or not the task waits for the set Actuation time.

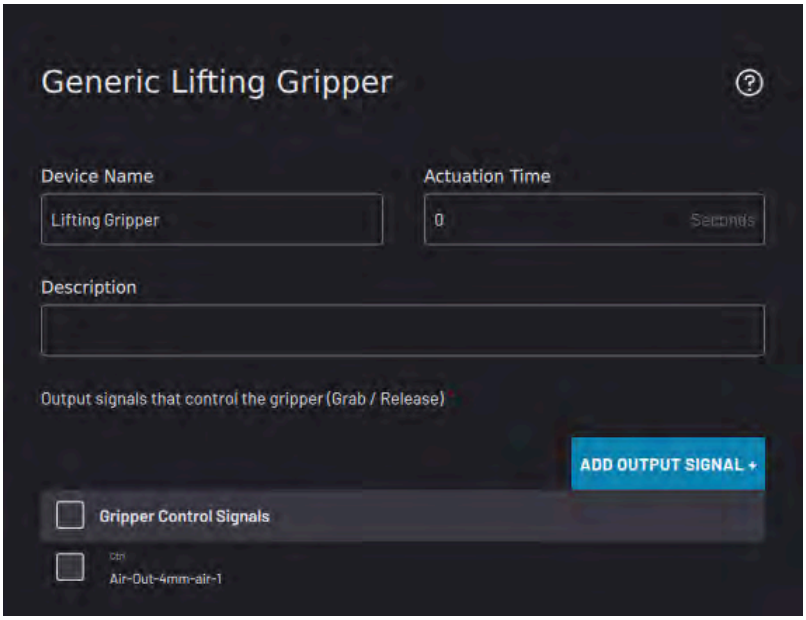
4. Tap **ADD NEW OUTPUT SIGNAL**.

a. From the **Select Device** dropdown menu, select a configured I/O device driving the gripper.

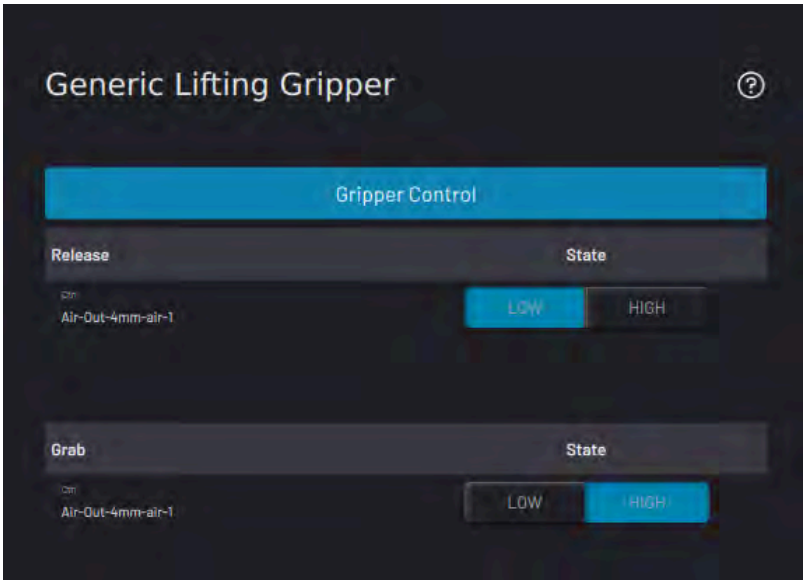


b. Select the air port(s) or I/O signals that your gripper use. Then tap **SAVE**.

5. Once all air ports required for the lifting device are listed in the **Gripper Control Signals** table, tap **NEXT**.

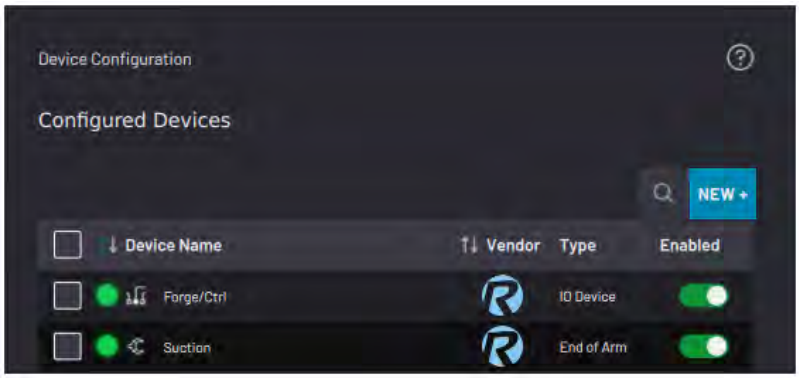


6. Choose which gripper state (Release or Grab) corresponds to a **HIGH** or **LOW** signal.



Note:
You cannot save the gripper configuration if you select the same conditions for multiple gripper states.

7. Tap **SAVE** to return to the Device Configuration home screen. Make sure the Lifting Gripper appears on the configured devices list and that it is **enabled**.



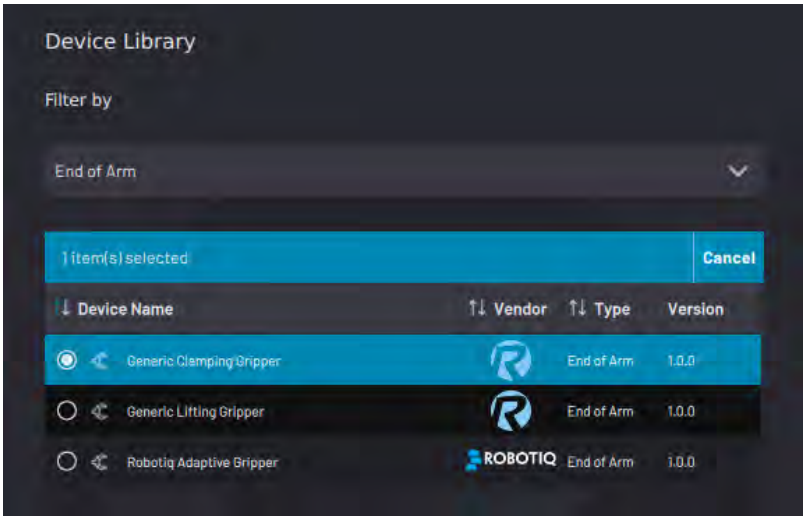
Note:
A device is **enabled** when its switch is green and toggled to the right.

Adding a Clamping Gripper

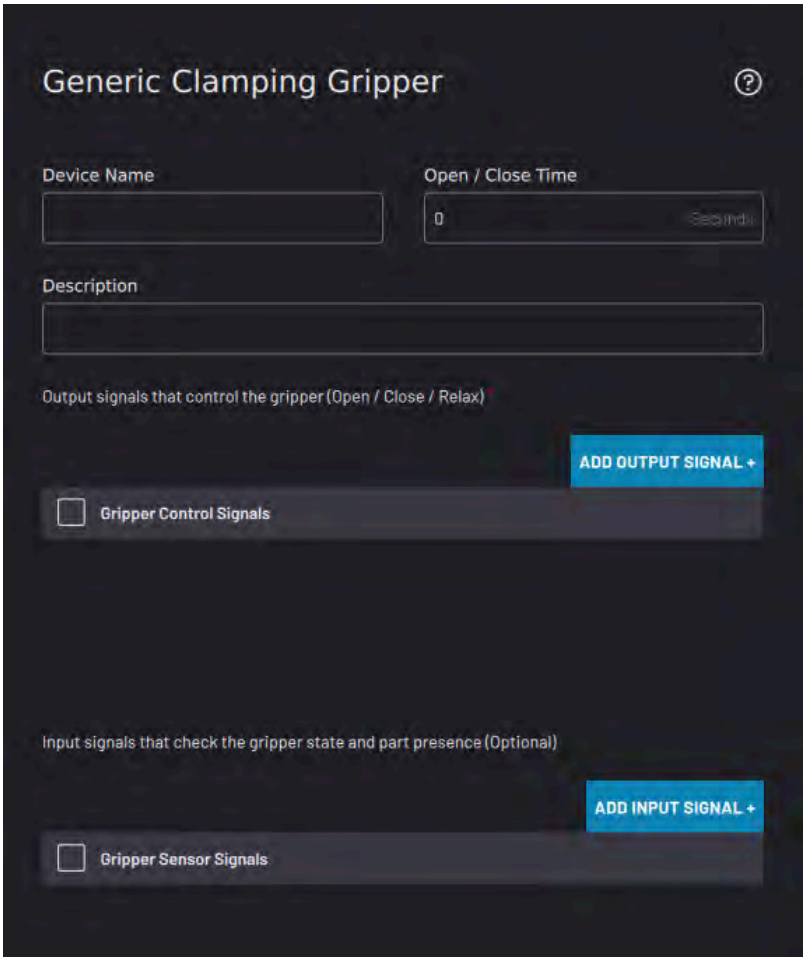
Follow these instructions for configuring a pneumatic or electronic clamping gripper. For an advanced clamping device, such as one with digital force control, check if it is supported in the Device Library.

Note:
To use a clamping gripper without robot I/O, first configure the IPC and I/O devices. See Fieldbus Configuration *(on page 30)* settings to learn more. Then select the signals for controlling air to the gripper.

1. Select **Generic Clamping Gripper**. Then tap **NEXT**.



2. Enter a **Device Name**. A **Description** is optional.

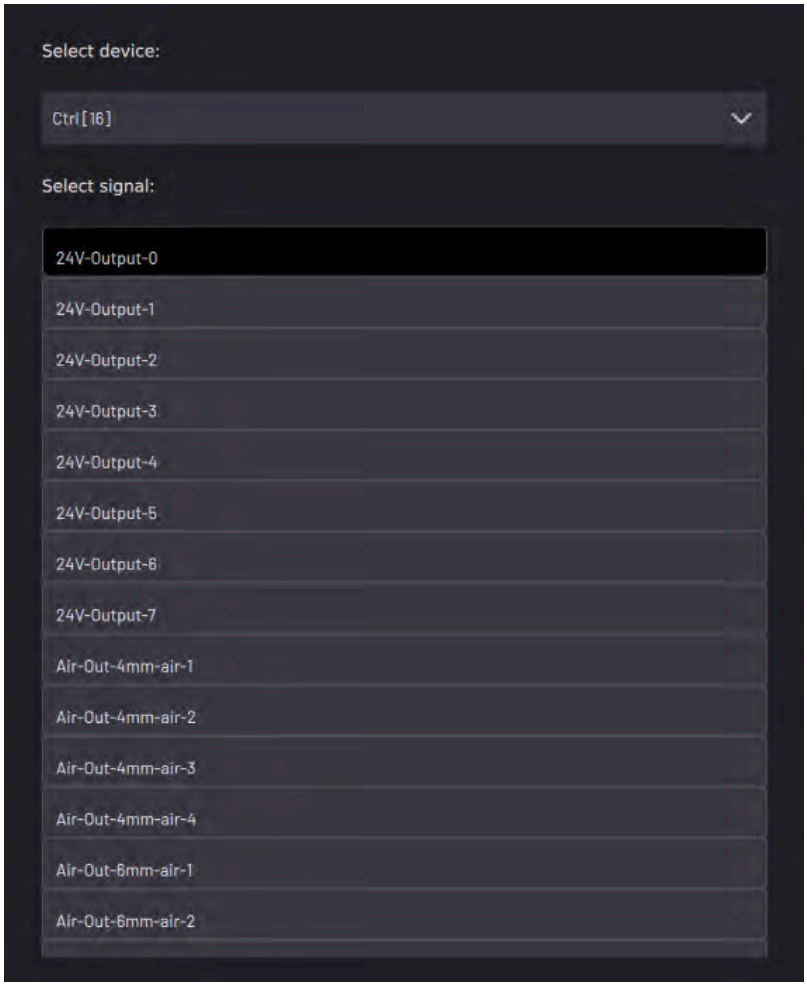


3. To make sure your device has enough time to actuate on and off in a task, enter the device's estimated **Actuation Time**. The default value is zero seconds.

> **Note:**
> In Task Canvas, that device's control blocks prompt you to choose whether or not the task waits for the set Actuation time.

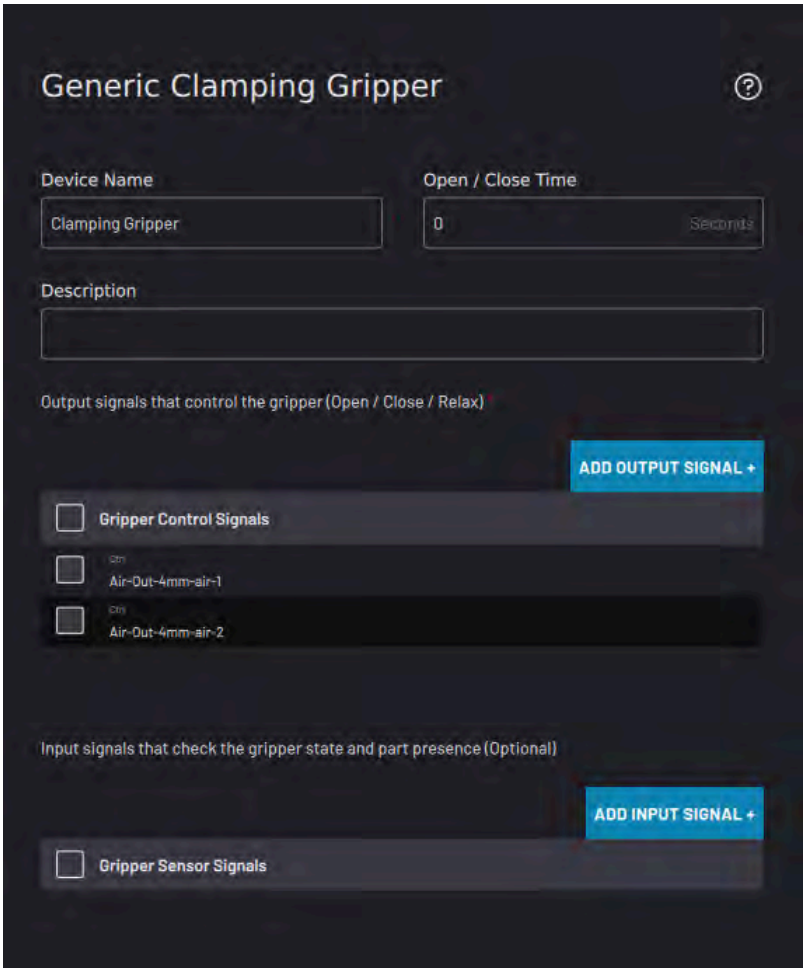
4. Tap **ADD OUTPUT SIGNAL** to select the signals that actuate the gripper.

    a. From the **Select Device** dropdown menu, select a configured I/O device driving the gripper.

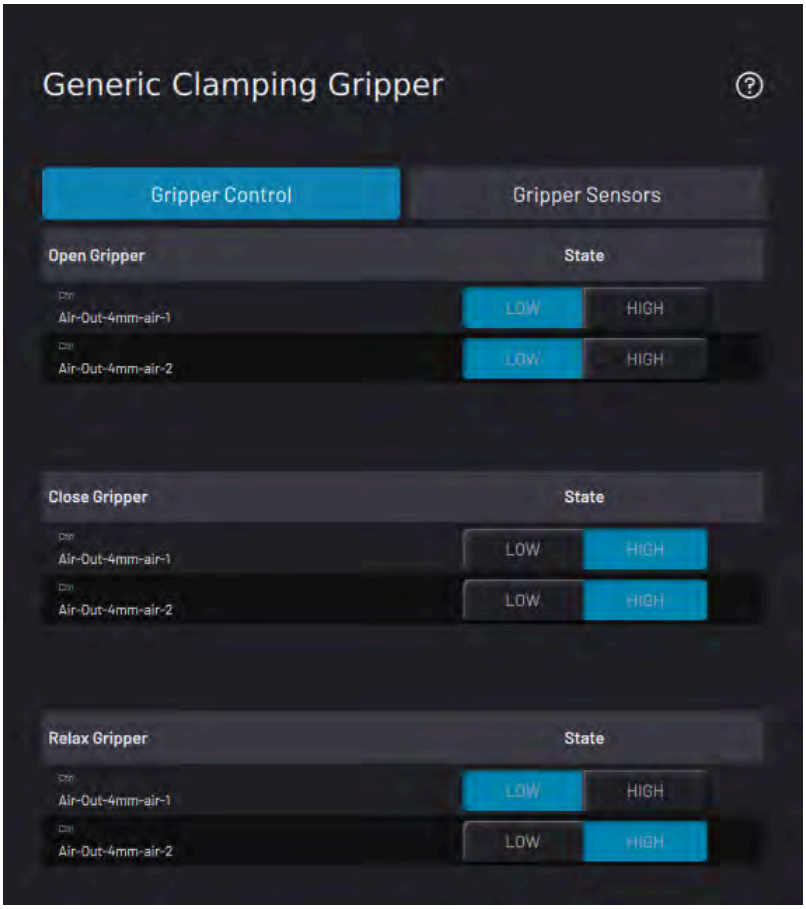


    b. Select the air port(s) or I/O signals that your gripper uses. Then tap **SAVE**.

5. To use gripper sensors to detect the state of the gripper, tap **ADD INPUT SIGNAL**. Repeat the process above to configure the sensor reading signals.
6. Once all air ports required for the clamping device are listed in the **Gripper Control Signals** table, tap **NEXT**.

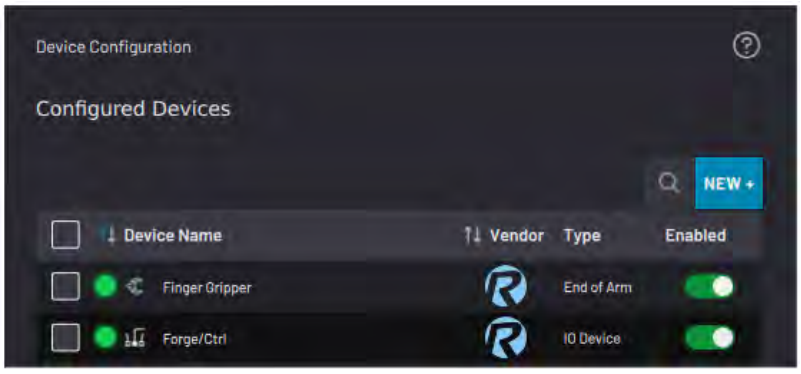


7. Choose which gripper state (Open, Closed, or Relaxed) corresponds to a **HIGH** or **LOW** signal.



Note:
You cannot save the gripper configuration if you select the same conditions for multiple gripper states.

8. Tap **SAVE** to return to the Device Configuration home screen. Make sure the Clamping Gripper appears on the configured devices list and that it is **enabled**.



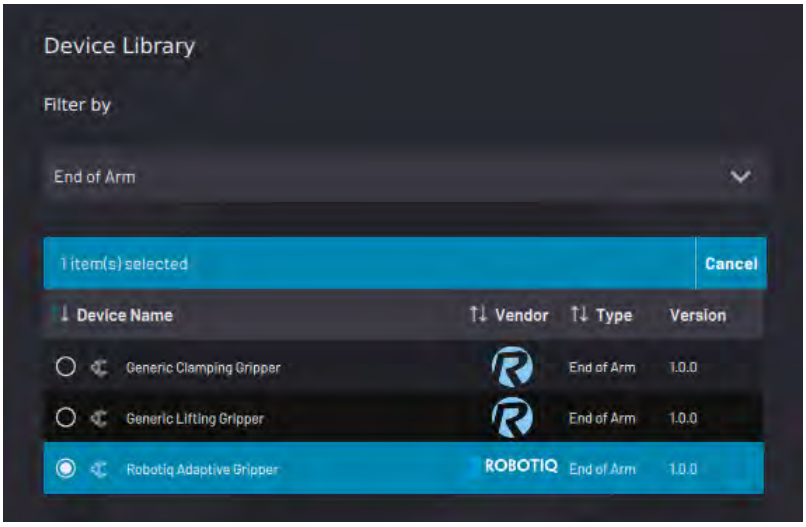
Note:
A device is **enabled** when its switch is green and toggled to the right.

Adding a Robotiq Adaptive Gripper

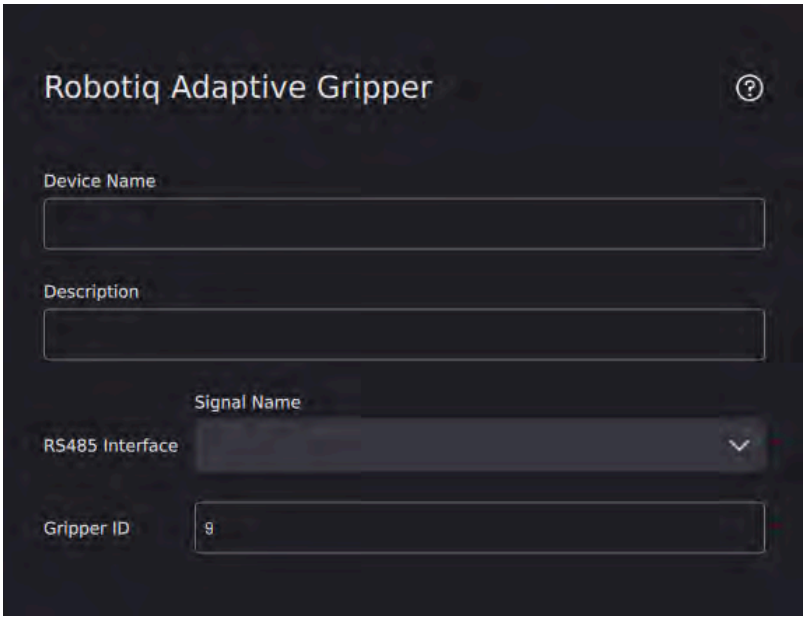
Follow these instructions to add a Robotiq adaptive gripper.

Note:
The Robotiq adaptive gripper requires electronic control through a serial fieldbus interface (Modbus RTU). The simplest way to control the Robotiq gripper is through an RS-485 to USB adapter. First connect the adapter to a USB port on the IPC and add a Modbus RTU serial interface to Forge/OS. See Robotiq instructions and Fieldbus Configuration *(on page 30)* settings to learn more.

1. Select **Robotiq Adaptive Gripper**, then tap **NEXT**.



2. Type in a **Device Name**. Then select the serial **RS485 Interface** (the Modbus RTU interface you added in **Fieldbus Configuration**). A **Description** is optional.



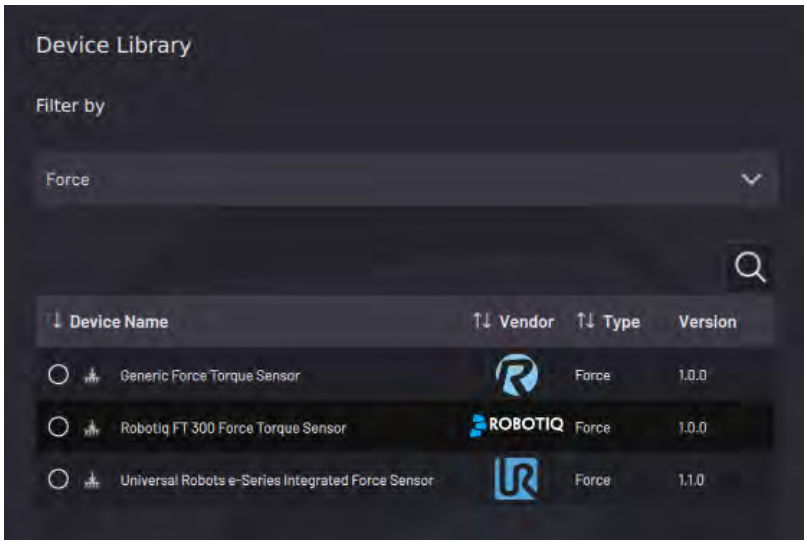
3. Choose a **Gripper ID**. Enter a unique **Gripper ID** if you are using multiple devices over the same serial interface. Leave the default ID value if you are using one serial device.

4. Tap **SAVE** to return to the **Device Configuration** main screen. Make sure the Robotiq Adaptive Gripper appears on the Configured Devices list and make sure it's **enabled**.

> **Note:**
> A device is **enabled** when its switch is green and toggled to the right.

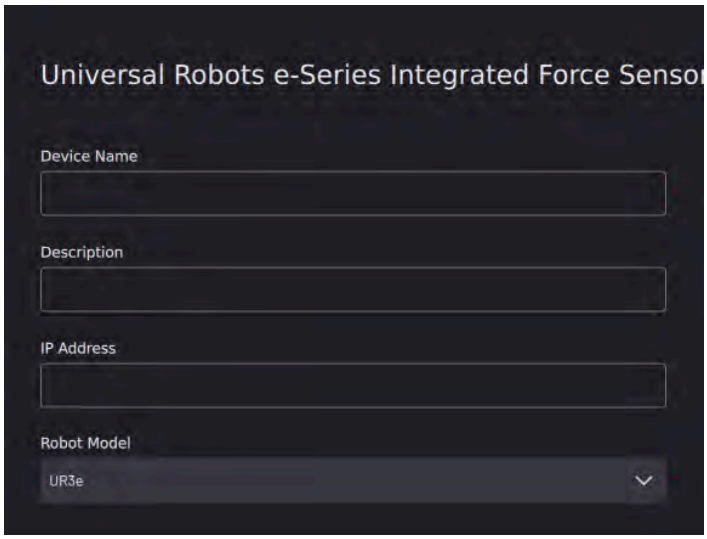
Adding a Force Sensor

ForgeOS 5 supports three force sensor options: **A Universal Robot e-Series Integrated Force Sensor**, a **Robotiq FT 300 Force Torque Sensor**, and a **Generic Force Torque Sensor**.



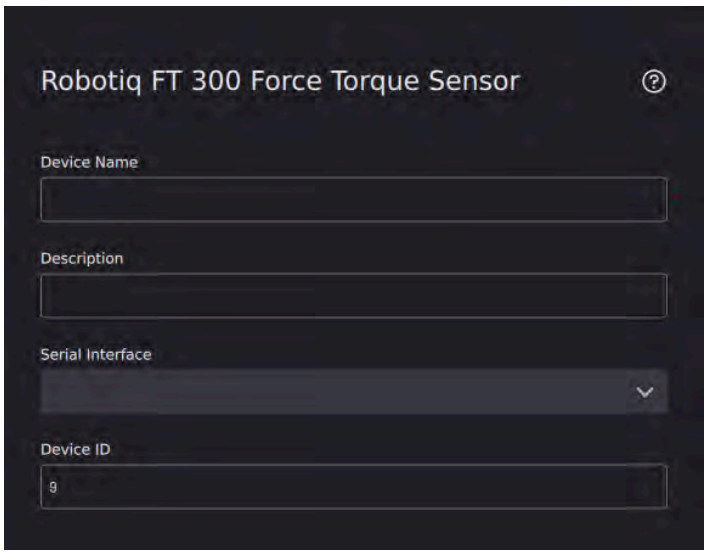
UR e-Series Integrated FT Sensor:

To add a Universal Robots e-Series Integrated Force Sensor, enter a **Device Name**, **IP Address**, and **Robot Model**. A **Description** is optional.



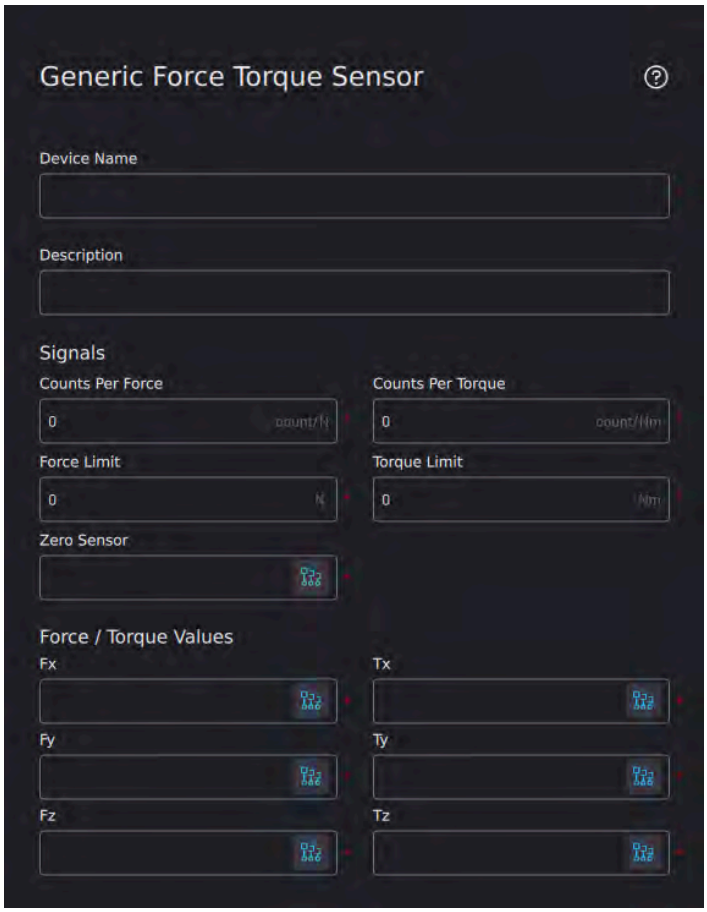
Robotiq FT 300 Sensor:

To add a Robotiq force torque sensor, choose the USB communication port that the sensor is connected to with a serial-USB adapter. Enter a unique **Device ID** if you are using multiple devices over the same serial interface. Leave the default Device ID value if you are using one serial device.

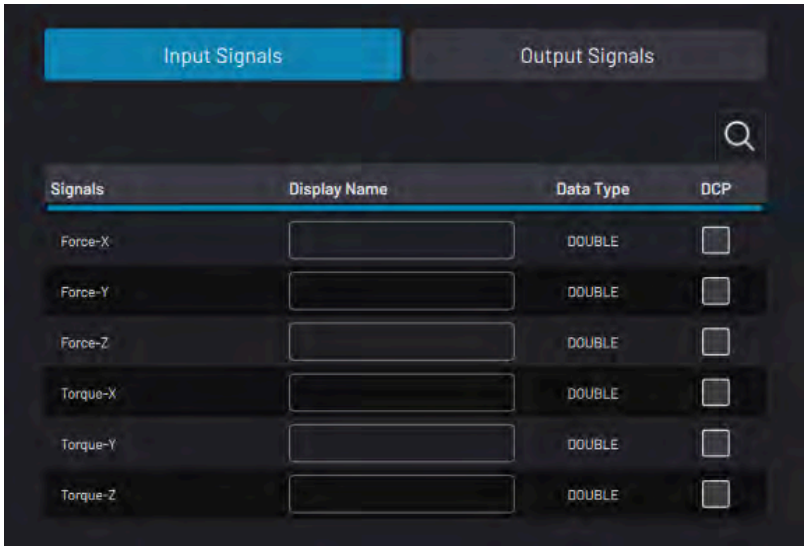


Generic FT Sensor:

If you are not using an e-Series UR robot or a Robotiq Force Torque sensor, add a Generic Force Torque Sensor. Enter the specifications of your sensor, including **counts per force/torque**, **force/torque limits**, and **sampling rate**. Then select the signals for zeroing the sensor and detecting force and torque in each Cartesian direction.



For each of these force torque sensors, select the checkbox next to the signal(s) that you want to see and zero in the device's Device Control page (**DCP**).



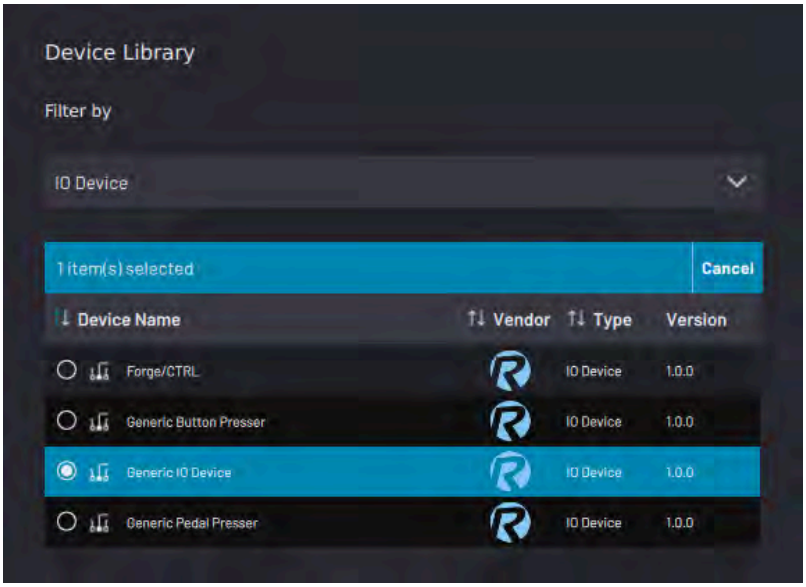
Tap **SAVE** to return to the Device Configuration home screen. Make sure the force sensor appears on the configured devices list and that it is **enabled**.

Note:
A device is **enabled** when its switch is green and toggled to the right.

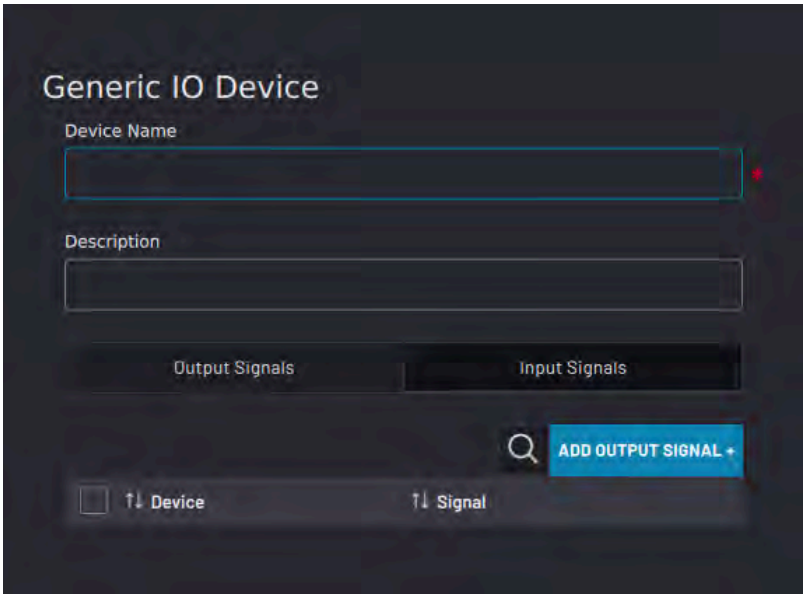
Adding a Generic IO Device

A Generic IO Device allows you to define a custom device by creating functions that can be manually controlled through the Device Control app or programmed as blocks in Task Canvas.

1. Select **Generic IO Device**, then tap **NEXT**.



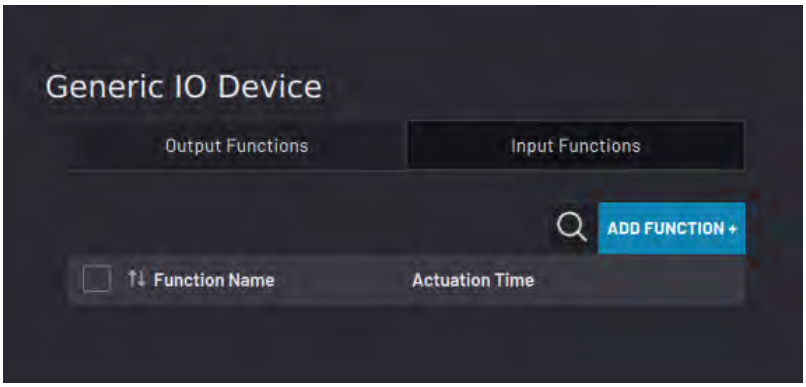
2. Type in a **Device Name**. A **Description** is optional.



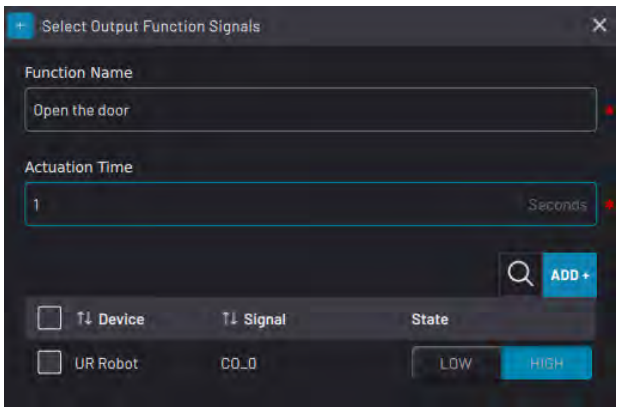
3. Tap the **Output Signals** and **Input Signals** tabs to link existing signals to the device. Tap **ADD OUTPUT SIGNAL+** or **ADD INPUT SIGNAL+** to open the parameter selector and select the signals. These are the signals that you will select from in the next few steps when creating the device functions. Click **NEXT** when you are finished.

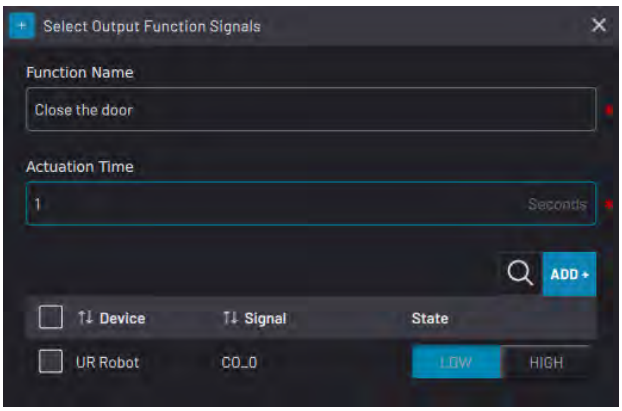
> **Tip:**
> You can always return to this screen later to add or remove signals.

4. Follow these substeps to add your output and input functions.



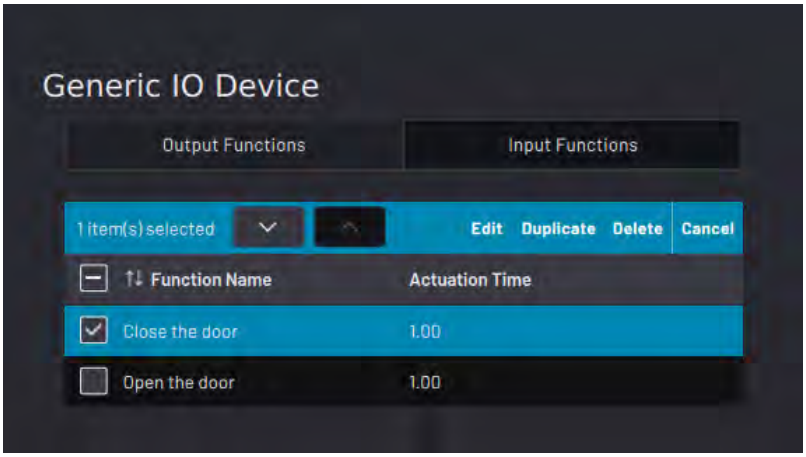
   a. In the **Output Functions** tab, tap **ADD FUNCTION+** to set the LOW/HIGH conditions and actuation times of the selected boolean output signals. Select a signal and tap **Delete** to remove it from the function. Tap **ADD +** to add a signal back. The **Actuation Time** is how long Forge/OS will wait for the function to execute. Tap **SAVE** for the function to appear in the table.





b. In the **Input Functions** tab, tap **ADD FUNCTION+** to set the LOW/HIGH conditions of the selected boolean input signals. Select a signal and tap **Delete** to remove it from the function. Tap **ADD+** to add a signal back. Tap **SAVE** for the function to appear in the table.

5. In the table of saved functions, select a function to change its order in the table, edit it, duplicate it, or delete it.



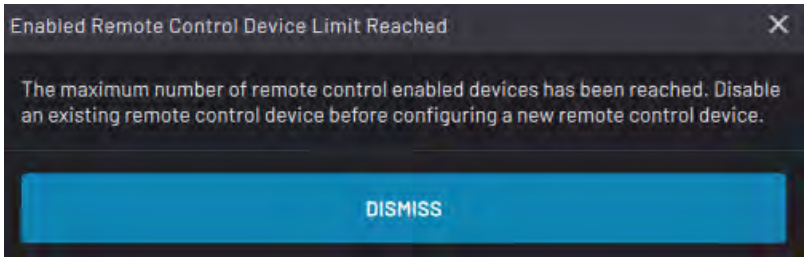
6. Tap **SAVE** to return to the Device Configuration home screen. Make sure the Generic IO Device appears on the configured devices list and that it is **enabled**.

Note:
A device is **enabled** when its switch is green and toggled to the right.

Adding a Remote Control Device

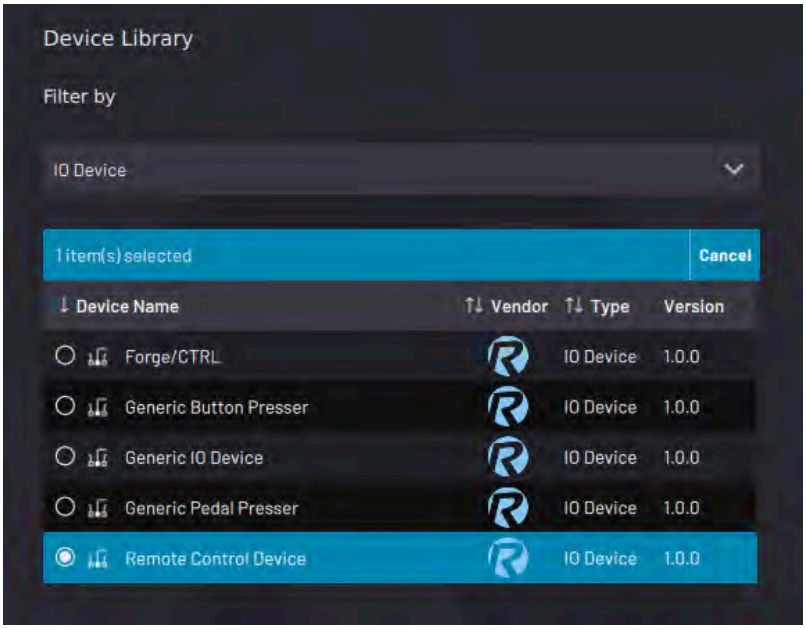
A Remote Control Device allows you to communicate with ForgeOS from an external HMI.

You can only have one Remote Control Device enabled at a time. If you try to add another one, a pop-up reminds you to disable the existing one first.

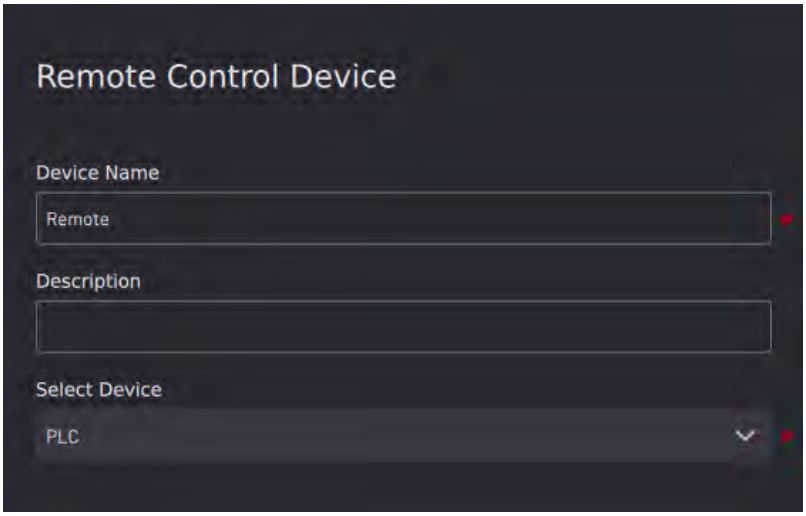


Note:
The Remote Control Device requires you to map signals. First configure a PLC or other relevant devices. Any device with IO is available to send and receive commands. Devices with pre-configured signal mapping (such as the **Rockwell Automation Logix PAC**) will have commands and outputs pre-mapped, allowing setup in just a few steps.

1. Select **Remote Control Device**, then tap **NEXT**.



2. Type in a **Device Name**. A **Description** is optional. In the **Select Device** dropdown, choose a device to use for remote control. Then tap **NEXT**.



3. In the **Command Input Signals** tab, tap **ADD COMMAND +** to add an input command. To remove one or more input commands, select the checkbox(es) and tap **Remove**. For the input commands in the table, add device signals by tapping the variable selector to the right of the ‘Select a value” field.

Input signals (that are sent from the PLC to Task Canvas) include:

Input Command	Description
Clear Error	Send a “Clear Error” signal to all devices (as if pressing the Reset button on the READY pendant).

Input Command	Description
	Note: This command is ignored if a task is executing.
Load Task	Load the current Task ID task and select the Start block.
Load and Start Task	Load the associated task and immediately start it at 100% speed from the Start block. **Note:** If another task is currently open and executing, it stops the task and loads/starts the called task. If the called task is currently open and NOT executing, it starts the task.
Reset Task	Reset all block states. Place the view and block selection on the Start block of the Main Task. **Note:** This command is ignored if a task is executing.
Set Speed	Set the Task Canvas speed slider speed. This can be received when a task is open regardless of whether it is executing. **Note:** Accepted values are 1-100. Any value outside the accepted range is ignored.
Start Task	Start the loaded task from the selected block. **Note:** This command is ignored if a task is not open or if a task is executing. If a block was stopped mid-execution, it is resumed and not restarted. If there are multiple paused subtasks executing, this command assumes the “Resume Simultaneous Execution” option.

Input Command	Description
Stop Task	Perform a controlled stop on an executing task (as if pressing the Stop button on the READY pendant). **Note:** This command does NOT execute a reset, but rather leaves blocks in a paused state. This command is ignored if a task is not executing.
Task ID	Specify the task to be loaded on a **Load Task** or **Load and Start Task** command. A Task's ID can be set in the Remote Task ID field in Task Settings. **Note:** A change in value does NOT immediately load the specified task, but rather just places that task "on deck" for a Load Task or Load and Start Task command.

4. In the **Output Signals** tab, tap **ADD OUTPUT +** to add an output. To remove one or more outputs, select the checkbox(es) and tap **Remove**. For the outputs in the table, add device signals by tapping the variable selector to the right of the 'Select a value" field.

Output signals (that are sent from Task Canvas or ForgeOS to the PLC) include:

Output Command	Description
Last Task Failed	The last task failed upon completion. This value is reset when a task is started or resumed.
Last Task Passed	The last task passed upon completion. This value is reset when a task is started or resumed.
Ready to Run	A task is open. All devices needed to execute the task are in RUN or OK mode. There are no safety errors. A block is selected to Start or Resume the task.
Remote Mode	The Remote Control toggle in the Taskbar's User button flyout is enabled (green and toggled to the right).
Reset	A task is open and not executing. All blocks are at their initial states. No block has the “Last Executed” icon, and the Start block of the Main Task is selected and visible.
Task Executing	A task is currently running.
Task ID	This is the integer value of the current task (as defined by the Remote Task ID field in Task Settings). This field

Output Command	Description
	is 0 when no task is loaded (even if the command “Task ID” has been set).
Task Speed	This is the current value of the Task Canvas speed slider. The range of possible values is 1-100. This field is null when a task is not open.

5. Tap **SAVE** to return to the Device Configuration home screen. Make sure the Remote Control Device appears on the configured devices list and that it is **enabled**.

> **Note:**
> The SAVE button is only available when all Device Signal fields are filled in.

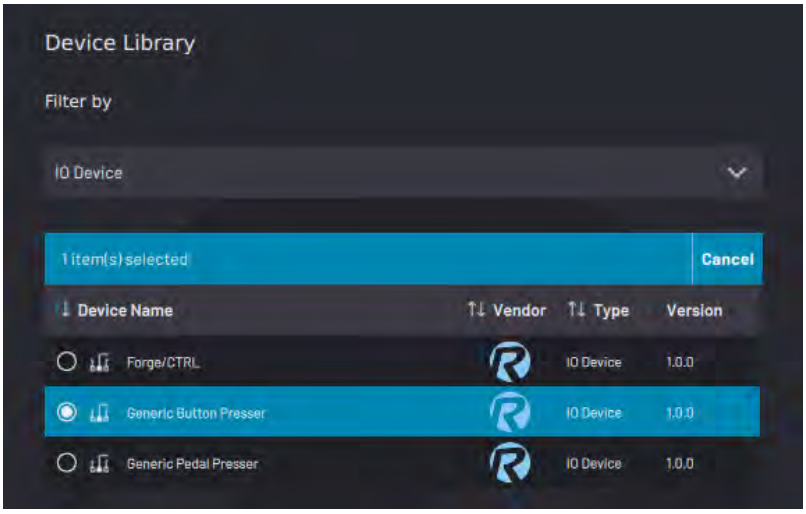
> **Note:**
> A device is **enabled** when its switch is green and toggled to the right.

Adding a Button/Pedal Presser

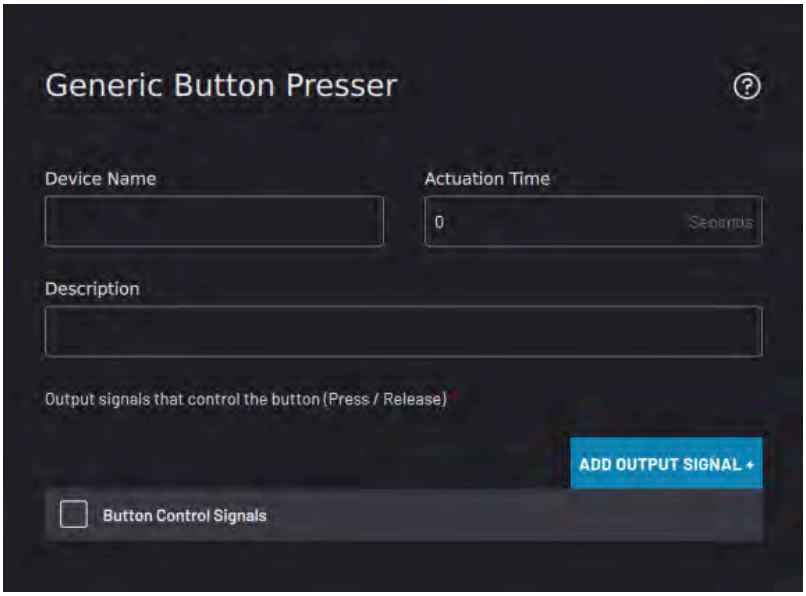
Follow the instructions in this section to configure a generic button or pedal presser. These steps are for a button presser, but adding a pedal presser follows the same steps.

> **Note:**
> The button/pedal presser requires pneumatic or electronic control through I/O. First configure the IPC and I/O devices. See Fieldbus Configuration *(on page 30)* settings to learn more. Then select the signals for controlling air or current to the pressing device.

1. Select **Generic Button Presser** or **Generic Pedal Presser**. Then tap **NEXT**.



2. Type in a **Device Name**. A **Description** is optional.

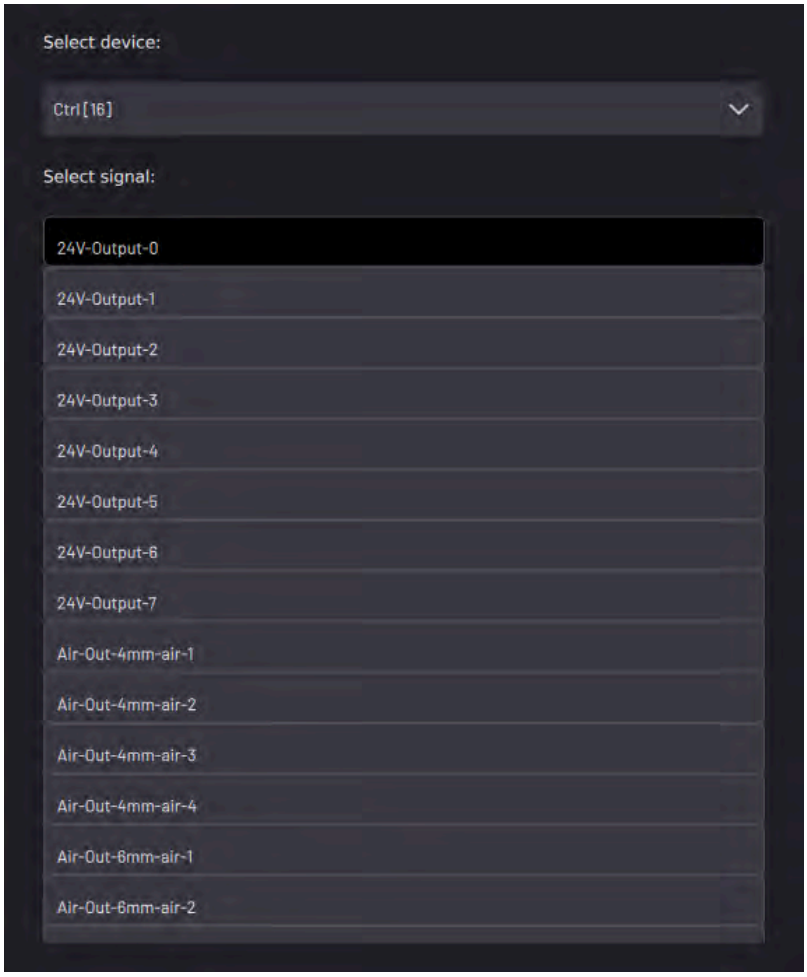


3. To make sure your device has enough time to actuate on and off in a task, enter the device's estimated **Actuation Time**. The default value is zero seconds.

> **Note:**
> In Task Canvas, that device's control blocks prompt you to choose whether or not the task waits for the set Actuation time.

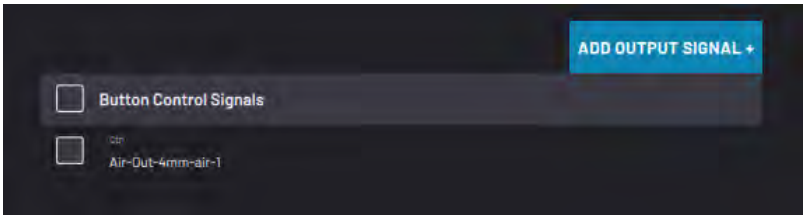
4. Tap **ADD NEW OUTPUT SIGNAL**.

a. From the **Select Device** dropdown menu, select a configured I/O device driving the presser.

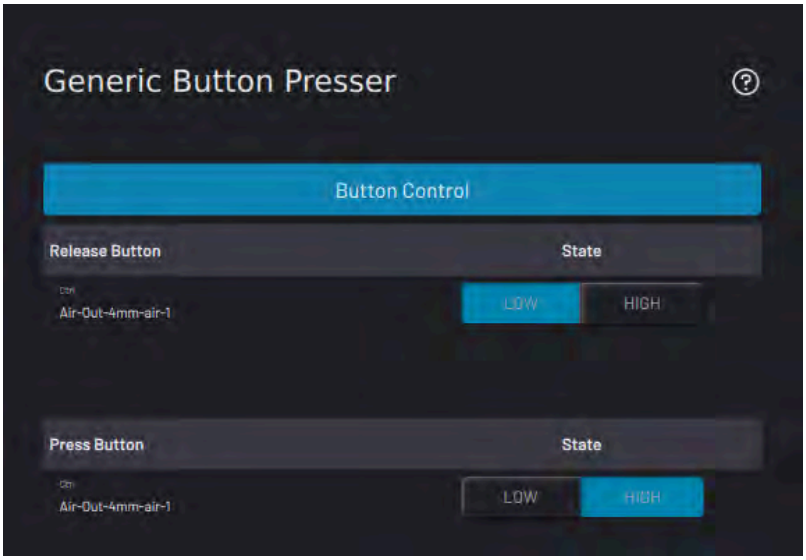


b. Select the air port(s) or I/O signals that your device uses. Tap **SAVE** for each selection.

5. Once all air port(s) required for the device are listed in the **Gripper Control Signals** table, tap **NEXT**.

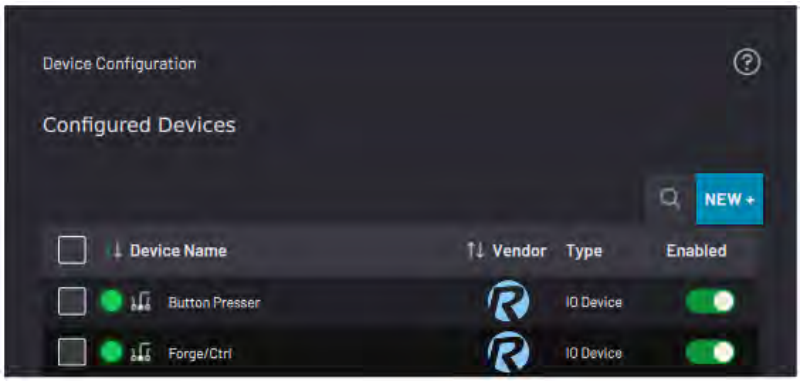


6. Choose which state (Release or Press) corresponds to a **HIGH** or **LOW** signal.



Note:
You cannot save the configuration if you select the same conditions for multiple states.

7. Tap **SAVE** to return to the Device Configuration home screen. Make sure the Button Presser appears on the configured devices list and that it is **enabled**.

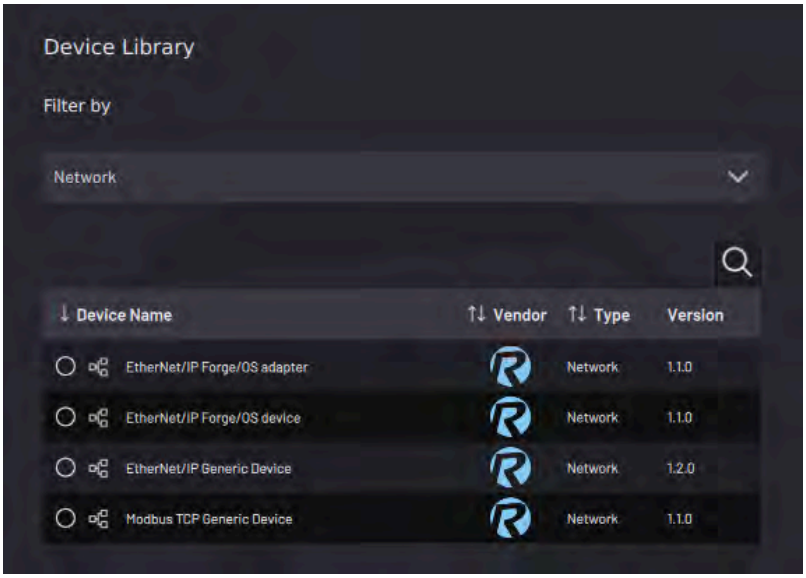


Note:
A device is **enabled** when its switch is green and toggled to the right.

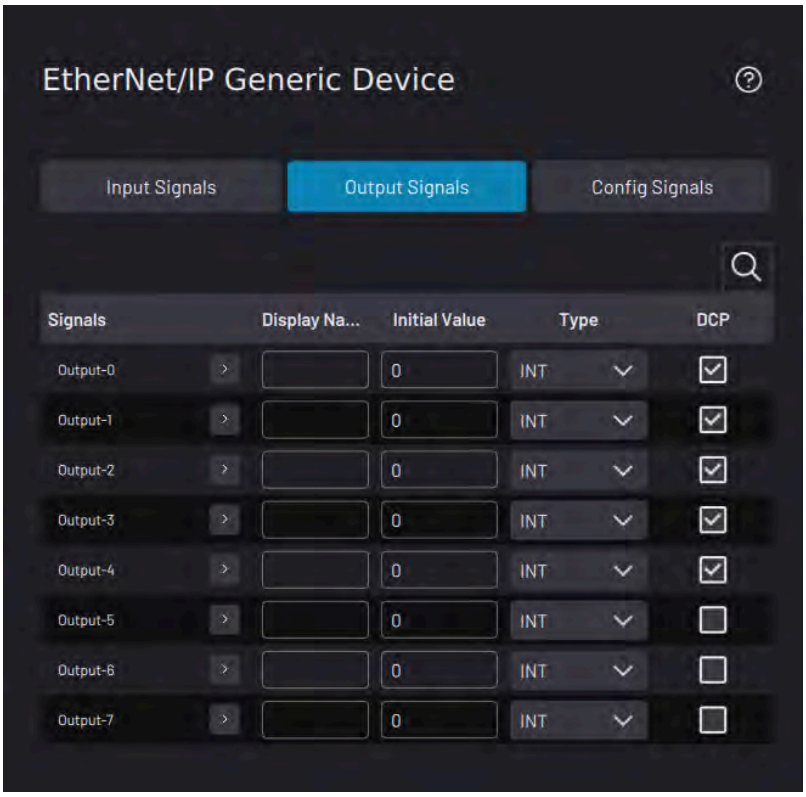
Adding a Network I/O Device

Note:
To add a network-based fieldbus device, such as an EtherNet/IP or Modbus TCP device, first add the relevant fieldbus interface in the Settings app. See Fieldbus Configuration *(on page 30)* settings to learn more.

1. Select the name of your fieldbus device. Then tap **NEXT**.



2. Type in a **Device Name** and any other required information for your I/O device. Refer to the device manufacturer's guides for device setup parameters.
3. **(EtherNet/IP devices only)** Tap **NEXT** to set up the Input, Output, and Config signals.



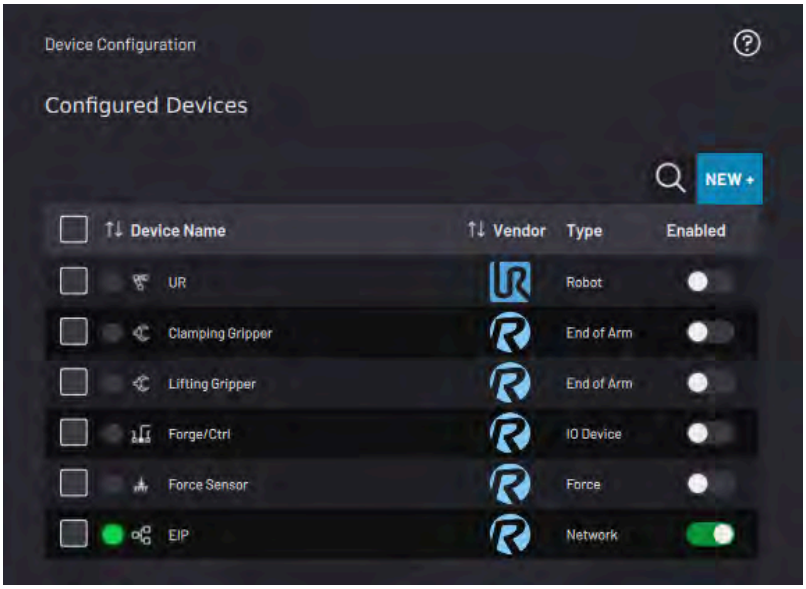
Note:
The **Config Signals** tab is available for the **EtherNet/IP Generic Device** only.

a. (Optional): Type in a **Display Name** for each signal to show what each signal does.
b. For each Output and Config signal, type in an **Initial Value** if you want the signal to be set as soon as Forge/OS connects to the device.

c. Choose the data **Type** for each signal in the drop down

d. If you want a signal to appear in the device's Device Control page, tap the **DCP** checkbox next to that signal.

4. Tap **SAVE** to return to the Device Configuration home screen. Make sure the Network I/O device appears on the configured devices list and that it is **enabled**.

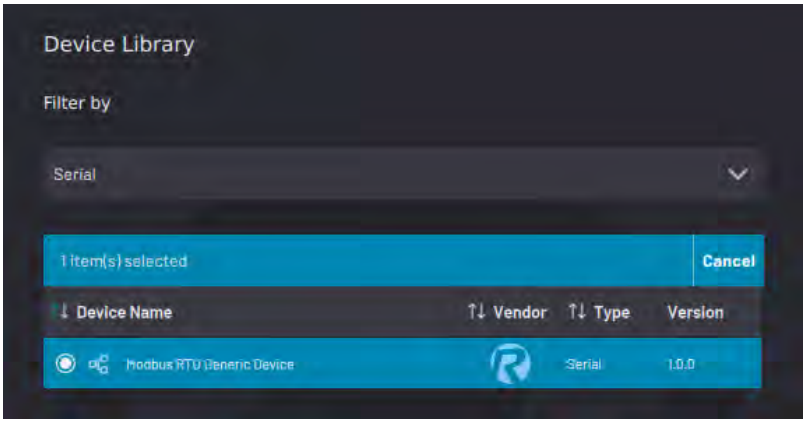


> **Note:**
> A device is **enabled** when its switch is green and toggled to the right.

Adding a Serial Device

> **Note:**
> To add a Serial-based fieldbus device, such as a Modbus RTU device, first add a serial interface to ForgeOS in the Settings app. See Fieldbus Configuration *(on page 30)* settings to learn more.

1. Select a Serial device. Then tap **NEXT**.



2. Type in a **Device Name**, select the **Fieldbus Interface**, and enter other required information. Refer to the device manufacturer's guides for device setup parameters.

3. Tap **SAVE** to return to the Device Configuration home screen. Make sure the serial device appears on the configured devices list and that it is **enabled**.

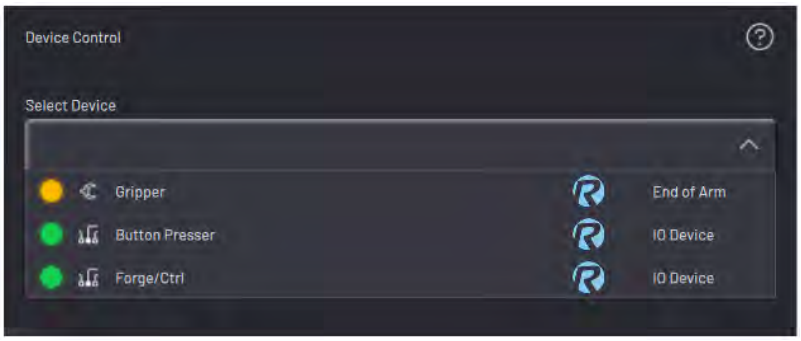
> **Note:**
> A device is **enabled** when its switch is green and toggled to the right.

Chapter 7. Device Control

Use the Device Control app to manually control devices while programming a task or to recover from errors.

Selecting a Device

Tap the Select Device dropdown and choose the device to control. Only devices added and enabled in the Device Configuration app may be controlled. The screen shows information and controls for the selected device.



Only one device at a time may be controlled in the Device Control app. For example, a robot arm cannot be moved at the same time that a gripper is actuated. Switch between the devices using the dropdown.

Switch between the Device Control app and any other app as needed. When you leave the Device Control app and return, it still displays the controls for the most recently selected device.

Controlling a Robot

When you select a robot, the control panel depends on the control mode you select. The available modes are **Jog, Jump, Absolute Position,** and **Signals**.

Note:
The robot must be in **TEACH** mode or **PROGRAM** mode to control it from the Device Control app. See Device Status Panel *(on page 23)* for more on robot states.

When you select one of the three motion modes, the panel shows a 3D visualization of the robot arm, robot position data, and motion controls.

No.	Robot Control Feature	Description
1	Mode Selection	Enter the **Jog**, **Jump**, **Absolute Position**, or **Signals** mode.
2	TCP/Payload Selection	Tap the dropdowns to choose the **Active Tool Center Point (TCP)** and/or **Active Payload**. Tap **SET** to update the active TCP and Payloads.
3	Linear/Joint Move Selection	Move the robot linearly in a Cartesian frame or in Joint space. The Jog/Jump Controls panel updates for the type of move you choose.
4	3D Visualization Window	Interact with a 3D rendering of the selected robot arm. Rotate the view by dragging one finger across the window. Zoom in or out by using two fingers on the window in a pinching motion.

No.	Robot Control Feature	Description
		Note: The selected Frame is displayed over the 3D rendering for reference.
5	Position Data Selection	Select a type of robot position data to view. • **Joints** displays absolute joint position data. • **Frame** displays the TCP pose relative to the origin of the selected frame. • **Base** displays the TCP pose relative to the Base frame.
6	Position Data	View the robot's position for each Position Data mode selection. **Note:** You may notice a mismatch between the linear rotational coordinates that you specify and the linear rotational coordinates that the robot reports. See Absolute Positioning *(on page 88)* or Check a Waypoint or Frame *(on page 170)* for more info.
7	Speed Slider	Drag the slider to scale the speed of robot moves that you execute from the Device Control app. This slider does **not** affect the Task Canvas speed slider. **Tip:** Also control speed by pressing the keypad speed buttons on the right side of the READY pendant.
8	Frame Selection	Choose the Frame for Jog/Jump motions. The 3D Visualization Window displays the selected Frame. Available Frames are TCP, Base, and global Frames. To learn more about Frames, see Create and Manage Frames *(on page 156)*.
9	Jog Controls (Jog mode)	Press and hold the Jog buttons to move the robot in the selected Frame at the speed set by the Speed Slider. When you choose **Linear Move**, the Jog buttons correspond to linear and rotational motion (+/-) relative to each axis in the selected Frame. When you choose **Joint Move**, the Jog buttons correspond to +/- rotation for each joint on the robot arm.

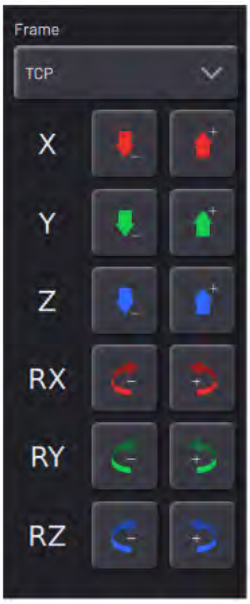
No.	Robot Control Feature	Description
	Jump Controls (Jump mode)	Enter **Jump By** values and tap **EXECUTE JUMP** to move the robot arm relative to the position it is in. When you choose **Linear Move**, specify a linear or rotational distance (+/-) relative to each axis in the selected Frame. When you choose **Joint Move**, specify the rotational distance (+/-) for each joint on the robot arm.
	Absolute Position Controls (Absolute Position mode)	Enter **Jump To** values and tap **EXECUTE JUMP** to move the robot arm to that absolute position relative to the selected Frame, usually Base. When you choose **Linear Move**, specify the Cartesian coordinates of the TCP relative to the selected Frame. When you choose **Joint Move**, specify the absolute positions to which each joint moves.
	"Snap to" Controls	Access in the **Jump** and **Absolute Position** modes. Align the TCP Frame with the selected Frame. Use this feature to align the tool and visualize where the robot will move. You must select a Frame other than TCP. **Snap +Z axis** aligns the TCP's +Z axis to the selected Frame's closest axis. **Snap all axes** aligns all of the TCP's axes with those of the selected Frame by moving the shortest distance possible. **Tip:** Also execute the Snap To command by pressing the keypad Start button on the READY pendant.

Jogging

Jogging is when you manually move the robot by holding a directional button. There are two modes of jogging: Linear and Joint.

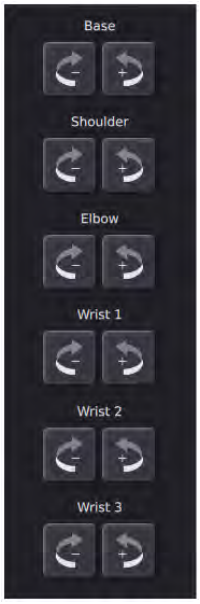
Linear Jog

Jog the robot's Tool Center Point (TCP) in the selected Frame along each of that Frame's coordinate axes (X, Y, Z). Rotate the TCP around each of the selected Frame's axes (RX, RY, RZ).



Joint Jog

Rotate the robot's joints in that joint axis's positive or negative direction.



Note:
In Jog mode, the READY pendant's keypad Jog buttons also move the robot. If the robot is in Teach Mode, jog the robot with the keypad Jog buttons no matter which app is open. The robot jogs with the most recently selected Frame, speed, and mode (Linear/Joint).

Tip:
Jog at a low speed if you're not sure which direction is positive.

Jumping

Jumping is when you move the robot by a set distance in at least one direction. There are two modes of jumping: Linear and Joint.

Linear Jump

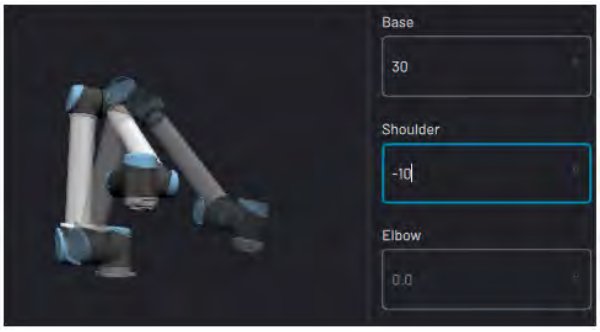
Move the TCP by a set distance relative to its position in the selected Frame. Rotate the TCP by a set number of degrees around the Frame's axes (RX, RY, RZ).

Joint Jump

Rotate one of more joints by a set number of degrees.

Refer to the 3D Visualization Window to see how the robot will move.

Perform the jump by pressing **EXECUTE** or by pressing the keypad **Start** button on the READY pendant.



Absolute Positioning

Absolute positioning is when you move the robot to an exact position using coordinates or joint positions. There are two modes of absolute positioning: Linear and Joint.

Linear Absolute Positioning

The coordinate boxes show the TCP's position in light gray text.

Move the TCP to a set position (X, Y, and Z) in the selected Frame. Rotate the TCP around the Frame's axes (RX, RY, RZ).

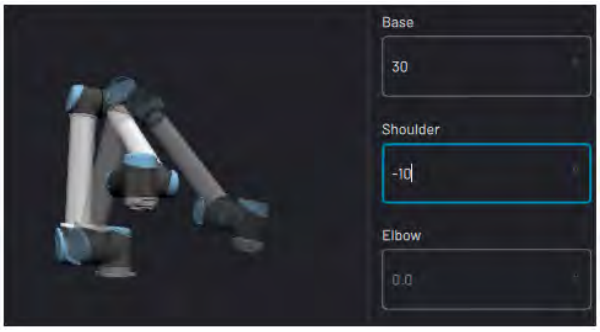
Joint Absolute Positioning

The coordinate boxes show each joint's position in light gray text.

Move one or more joints to a set position (in degrees).

Refer to the 3D Visualization Window to see how the robot will move.

Perform the jump by pressing **EXECUTE** or by pressing the keypad **Start** button on the READY pendant.



Note:
You may notice a mismatch between the linear rotational coordinates that you specify and the linear rotational coordinates that the robot reports. For example, if you Absolute Position the robot to the (Rx, Ry, Rz) values of (-150deg, 110deg, -165deg) in the Base Frame, the reported values in the “Base” position data tab could be (30deg, 70deg, 15deg).

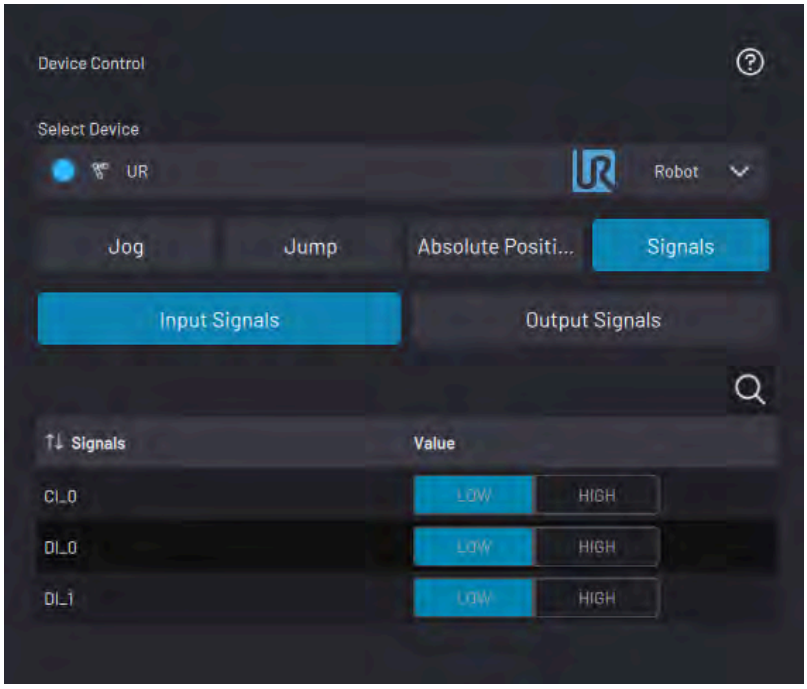
Even though robots accept Ry values between -180deg and 180deg, they only report Ry values between -90deg and 90deg. If your robot adjusts a specified Ry value to fit inside the latter range, it adjusts Rx and Rz values as well. **The specified values and adjusted values represent the same position, so the robot will move as expected. However, the different notation may cause Check blocks to fail.**

Signals

When you select the **Signals** mode, the panel shows a list of Read Only (Input) signals or a list of Writeable (Output) signals.

Toggle between input and output signals by tapping **Input Signals** and **Output Signals**.

For a signal to appear in the Device Control app, select its **DCP checkbox** in Device Configuration.



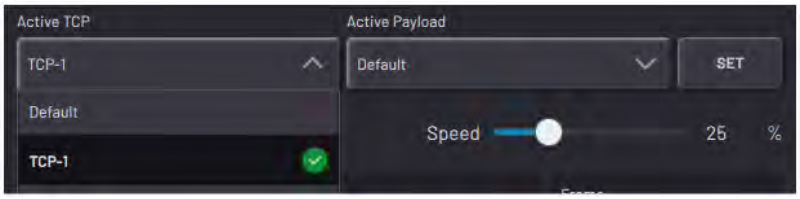
- The **Input Signals** list shows the robot's configured Input signals in real-time. Analog signals display as float values. Digital signals display as LOW or HIGH.

- In the **Output Signals** list, set the values of the robot's configured Output signals. Set the value of analog signals by typing a number in the field. Set digital signals using the **LOW** and **HIGH** toggle buttons. The active LOW/HIGH toggle button appears blue.

See Controlling a Network I/O Device *(on page 102)* to learn more about Input/Output signal control.

Active TCP/Payload Selection

To change the **Active TCP** and/or the **Active Payload**, choose the options you want in the dropdowns. Then tap **SET**. The TCP/Payload options are the ones you add in the robot's Device Configuration (see Adding a Robot *(on page 50)*).



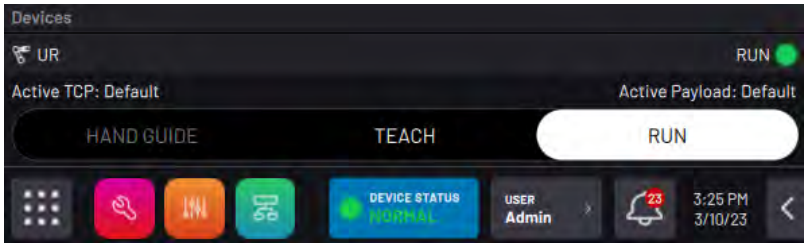
Tip:
Don't forget to tap **SET** to change the Active TCP and Payload to the selected values.

Active TCP	Active Payload
The Active Tool Center Point is where the end effector interacts with objects. It's the point around which the end ef-	The Active Payload tells the robot to account for the amount of weight at the end of its arm.

Active TCP	Active Payload
fector moves and rotates when the robot executes linear moves. When you select the TCP Frame in the Linear Move mode, the Active TCP appears in the 3D Visualization for reference.	Update the Active Payload whenever the actual payload changes, such as after picking up or putting down a part or tool.
Important: Forge/OS uses the Active TCP to define Waypoints and Frames. Setting the correct TCP on the robot is crucial for precise motion control in the Device Control app and in Task Canvas.	**Important:** A mismatch between the expected payload (Active Payload) and actual payload can cause unexpected motion and safety-related robot errors.

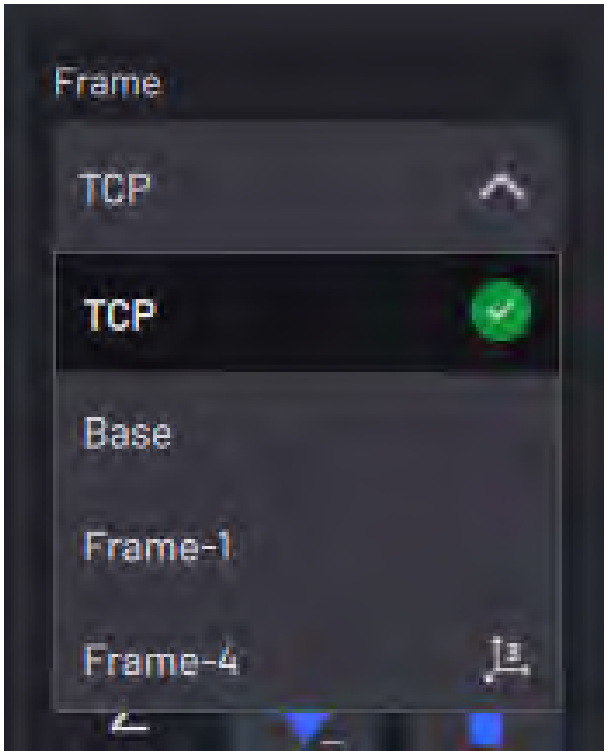
Note:
See Set TCP/Payload Block *(on page 198)* to learn more about setting the Active TCP/Payload in a task.

The Active TCP and Active Payload are listed in the Device Status Panel (via the “DEVICE STATUS” button) and the side legend (via the arrow in the bottom-right corner).



Frame Selection

All linear moves occur at the end effector (Active TCP) relative to the selected **Frame**. Choose the Frame to control the direction of motion from the Frame dropdown. When you choose Linear Move, the 3D Visualization window shows the selected Frame for reference.



Every robot has a TCP and Base frame by default.

- The **TCP** Frame is defined by the Active TCP's orientation. Its position and orientation move whenever the robot and attached tooling move. Select the TCP Frame to move and rotate the end effector relative to the TCP.
- The **Base** Frame is at the base of the robot. It does not change orientation when the robot moves. Select the Base Frame to move and rotate the end effector relative to the Base coordinates of the robot.

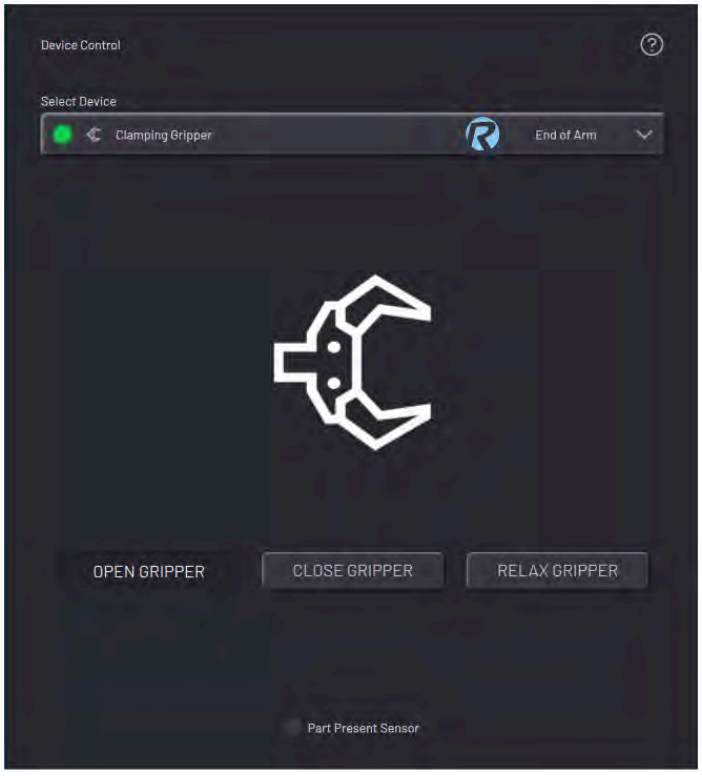
Note:
For information on how to view a custom Frame in the Device Control app, see Create and Manage Frames *(on page 156)*

Controlling a Lifting Gripper

When you select a lifting gripper, such as an end-of-arm vacuum suction gripper, the app shows controls to release or grab the part. For a suction gripper, these controls turn the suction off and on.

Controlling a Clamping Gripper

The clamping gripper panel lets you open the gripper, close the gripper, or relax the gripper. The relaxed state does not apply any force to close or open. The controls are set by the device's configuration.



If you add gripper sensors to the gripper's configuration, then the indicators below the control buttons show the state of those sensor signals. If the sensors are installed and configured correctly, a green circle indicates when one of the sensors is activated.

Controlling a Robotiq Adaptive Gripper

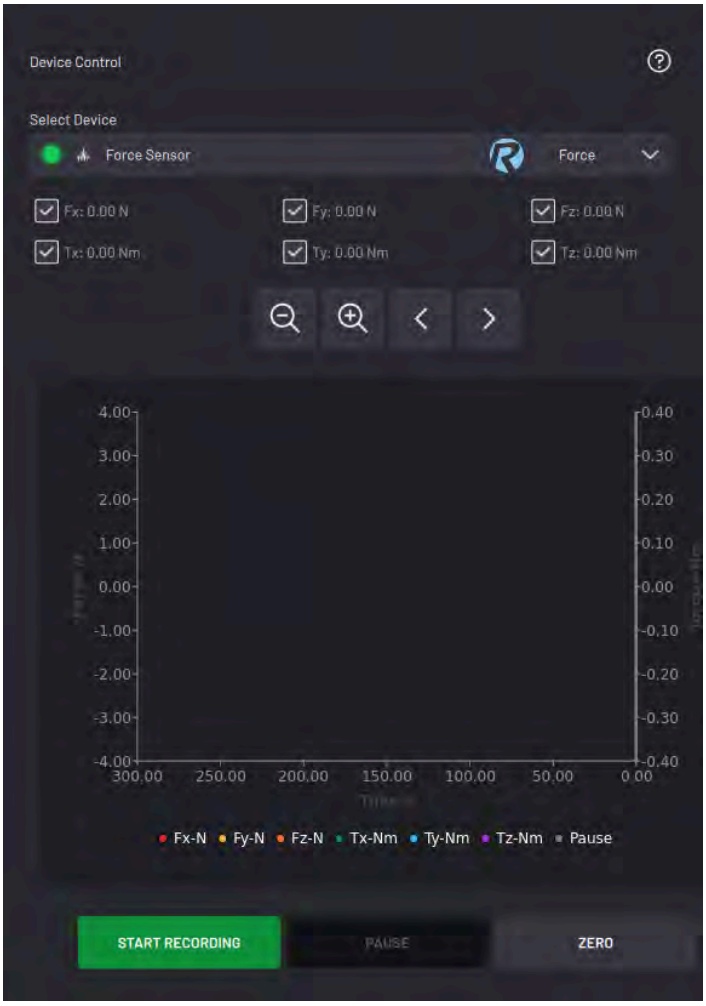
When you select a Robotiq adaptive gripper to control, the app shows the gripper position status and controls.

To fully open the gripper, tap **OPEN GRIPPER**. To fully close the gripper, tap **CLOSE GRIPPER**.

Choose how far the gripper opens/closes, how fast it opens/closes, and how much force it uses to open/close. Enter values into the **Position**, **Velocity**, and **Force** boxes. Then tap **SET POSITION** to move the gripper with the set parameters.

Controlling a Force Sensor

When you select a force torque sensor, the app displays an active graph of force and torque over time.



By default, all components of force and torque (**Fx, Fy, Fz, Tx, Ty,** and **Tz**) are shown on the graph. Un-check any of the items that you don't want to see.

- Tap **START RECORDING** to begin reading force and torque values.
- Tap **PAUSE** to stop collecting force and torque values.
- Tap **ZERO** to zero or bias the sensor.

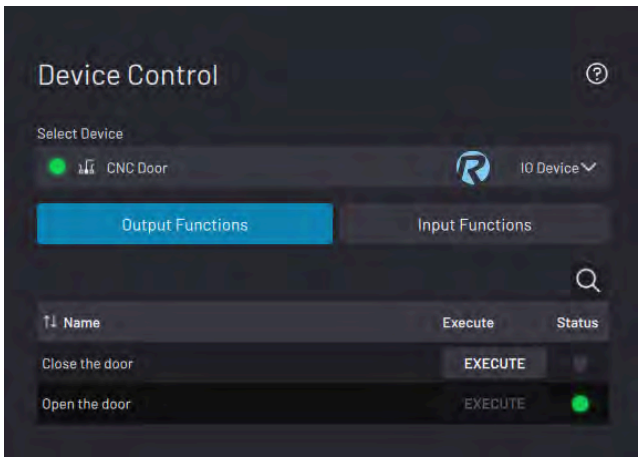
Look through the graph's data with the navigation tools above it. You can zoom in, zoom out, or scroll left or right. The zoom controls change the scaling of the Time axis.

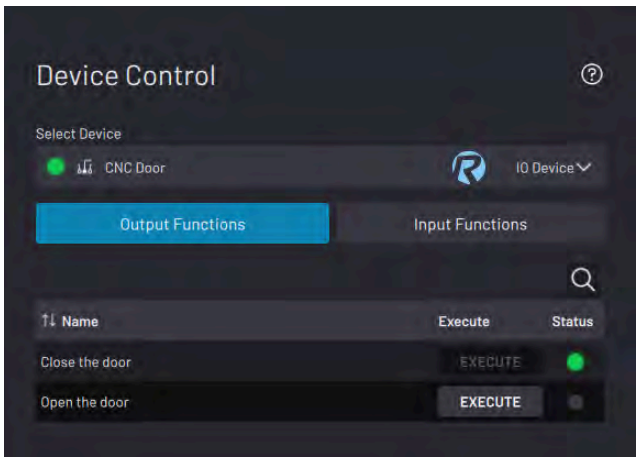
Controlling a Generic IO Device

When you select a generic IO device, the app shows the status of its output and input functions.

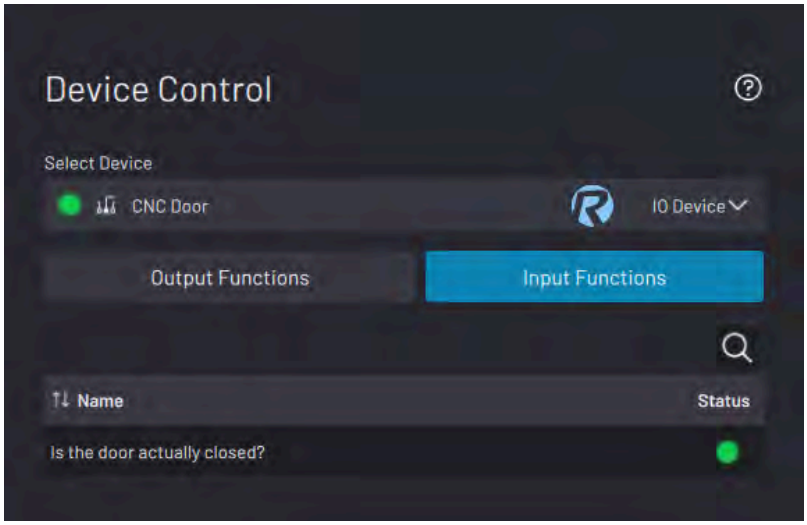
In the **Output Functions** tab, tap **EXECUTE** next to a function that you want to run. When a function's conditions are satisfied, its EXECUTE button is greyed out and its status light is green.

For example, suppose you have a CNC mill. Two actions that you can do with its door are close it or open it. Each of these actions use the same signal, except one is set to LOW while the other is set to HIGH. When you execute the "Open the door" function, the "Close the door" function's status light turns from green to grey.





In the **Input Functions** tab, see the status of read-only input functions. Functions with satisfied conditions have a green status light.

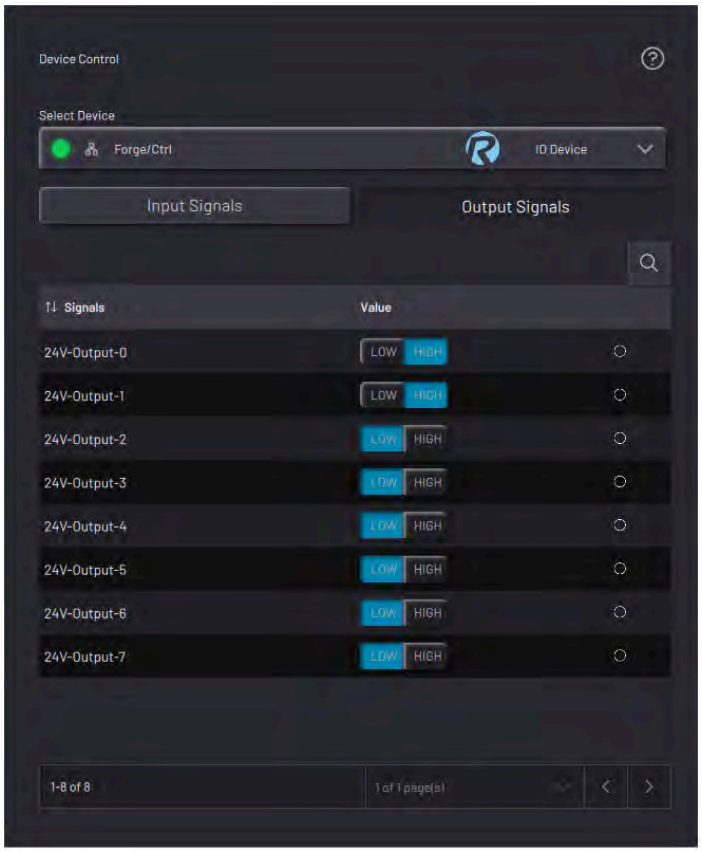


Controlling a Button/Pedal Presser

When you select a button presser or pedal presser, the app shows controls to press or release with force. You can also relax the pedal presser (apply no force to press or release). The controls are set by the device's configuration.

Controlling a Network I/O Device

When you select an I/O device, the app shows selection tabs for **Input Signals**, **Output Signals**, and **Config Signals**. The Config Signals tab is available for the EtherNet/IP Generic Device only.



Feature	Description
Input Signal List	View a sortable list of Read Only (Inputs) signals, including the display and signal names, real-time values, and units. The value of LOW/HIGH signals appears blue.
Output Signal List	View a sortable list of Writeable (Output) signals, including the display and signal names, value controls, and an active indicator. Set digital signals using the LOW and HIGH toggle buttons. The active LOW/HIGH toggle button appears blue.
Config Signal List	(Only available for the **EtherNet/IP Generic Device**.) View a sortable list of READ ONLY Config signals, including the display and signal names and real-time values. The value of LOW/HIGH signals appears blue.

Chapter 8. Parameter Manager

The Parameter Manager app is where you view and modify system-wide data. Depending on what type of data you want to view, tap one of the three tabs at the top: **Global User Variables**, **Applications (Apps)**, or **Devices**.

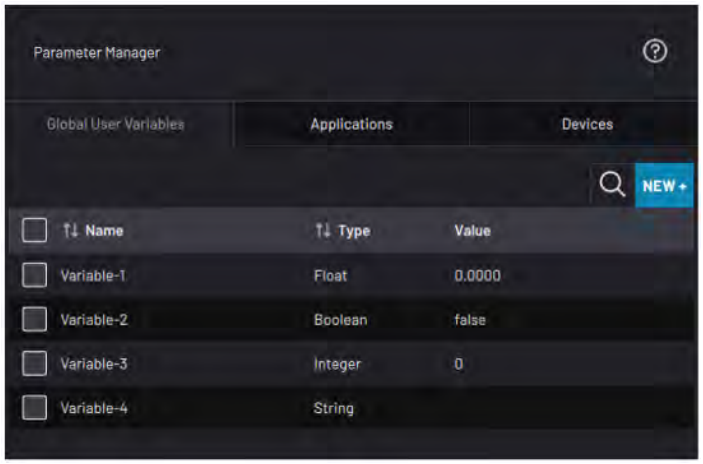
Global User Variables

The Global User Variables tab displays data for each global variable. If a variable is **global**, you can access it in the Parameter Manager and in all Task Canvas tasks.

> **Note:**
> Variables that you create in the Parameter Manager are *always* global. Variables that you create in a task (Variable Manager *(on page 161)*) are local to the task, but they can be made global.

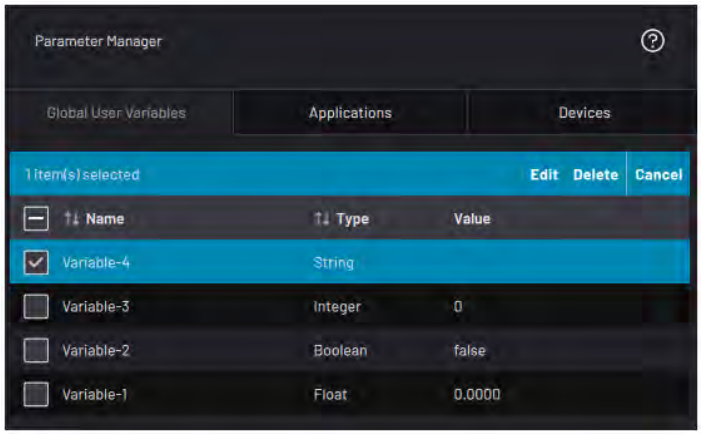
Possible variable types are:

- Float
- Boolean
- Integer
- String



To edit a global variable in the Parameter Manager, select it and tap **Edit** at the top of the table.

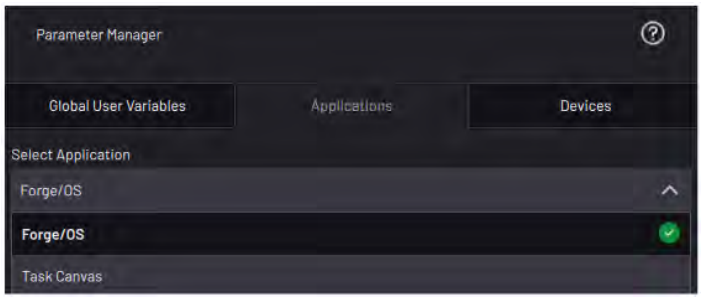
To delete a global variable, select it and tap **Delete** at the top of the table.



Applications

The Applications tab displays float, integer, date/time, boolean, and string data relevant to the selected app.

In the **Select Application** dropdown, select whether you want to view **ForgeOS** or **Task Canvas** data.



App	Parameter	Type	Description
Forge/OS	Date-Time	Date/Time	The system date and time, updated every second. Displays as [month/day/year] and [hours/minutes/seconds].
	Weekday	String	The current weekday (not abbreviated).
	Admin Logged In	Boolean	• **True** when logged in as an Admin. • **False** when logged in as another account or on the Sign In screen.
	Operator Logged In	Boolean	• **True** when you are logged in as an Operator. • **False** when you are logged in as another account or on the Sign In screen.
	Hard Drive Free Space	Float	The amount of free space on the ForgeOS hard drive, in megabytes (MB).

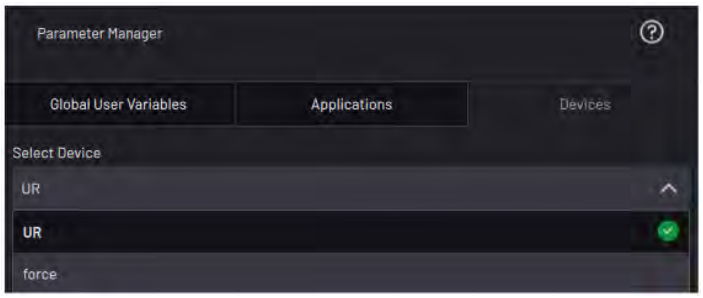
App	Parameter	Type	Description
Task Canvas	Task Executing	Boolean	• **True** when a task is executing, either from the Start Task or Start From Selected options in the Runtime Menu. • **False** when a task is stopped or a block is being Stepped.
	Current Task Name	String	The name of the task that is currently open in Task Canvas. Blank when no task is open.
	Last Task Passed	Boolean	• **True** when the last executed task finished on a "Finish" block set to "Passed". • **False** when a task begins executing. **Note:** This entry is not True when a task ends by running out of blocks or when you stop the task.
	Last Task Failed	Boolean	• **True** when the last executed task finished on a "Finish" block set to "Failed". • **False** when a task begins executing.

App	Parameter	Type	Description
			Note: This entry is not True when a task ends by running out of blocks or when you stop the task.

Devices

The Devices tab displays frame, float, double, integer, boolean, and string data relevant to the selected device.

In the **Select Device** dropdown, select which device's data you want to view. All devices added in the Device Configuration app show in the dropdown.



Device	Parameter(s)	Type	Description
(All Devices)	Device Status	String	The current device status as reported to the Device Status Panel. A device reports **Fault** when it's in an error state or it's disabled in Device Configuration.
6 Axis Robot	Global Speed Setting	Double	A percentage of max speed, as last confirmed by the robot. **Note:** This parameter automatically updates when you adjust the speed in the Device Control app in Teach Mode. This parameter does not automatically update when you adjust the speed in the Task Canvas app in Run Mode until you execute a move at the new speed. For example, if you execute a move in Run Mode at 100% speed and then reduce the Task Canvas speed, the Global Speed Setting parameter still reports “100%” until you execute another move. Because of this, you cannot adjust the speed and

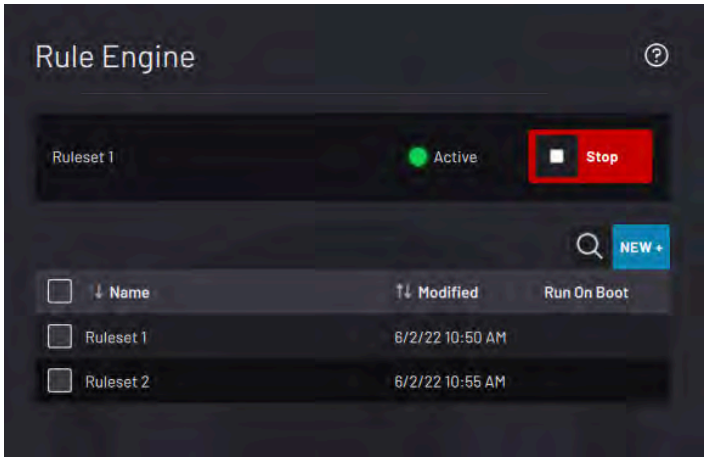
Device	Parameter(s)	Type	Description
			then start a task that immediately checks the Global Speed Parameter.
	Active TCP	String	The name of the active TCP.
	Active Payload	String	The name of the active payload.
	Robot Motion Possible	Boolean	Defined by the motor state, device status, and safety inputs (e.g., enabling switch). • **True** if robot motion is possible. • **False** if robot motion is not possible.
	Robot in Motion	Boolean	• **True** if the robot is moving. • **False** if the robot is not moving.
	TCP X, Y, Z Velocities	Double	The current velocity of the TCP in the X, Y, and Z directions of the TCP coordinate frame.
	TCP RX, RY, RZ Velocities	Double	The current rotational velocity of the TCP in the RX, RY, and RZ directions of the TCP coordinate frame.
	TCP X, Y, Z Velocities in Base Frame	Double	The current velocity of the TCP in the X, Y, and Z directions of the Base frame.
	TCP RX, RY, RZ Velocities in Base Frame	Double	The current rotational velocity of the TCP in the RX, RY, and RZ directions of the Base frame.
	TCP Speed	Double	The current speed of the TCP, independent of direction.
	J1-J6 Velocities	Double	The current rotational velocity of joints 1-6.
	Frame(s)	1-Point Frame or 3-Point Frame	The frame that the selected frame references.
Generic Clamping/Lifting Gripper	Gripper State	String	The current state of the gripper as defined by its control signals. Possible values are **grasped**, **released**, and **relaxed**.
	Sensor State	String	The current state of the gripper sensor. Possible values are **open**, **closed**, and **part present**. Blank if sensor signals are not configured.

Device	Parameter(s)	Type	Description
Smart Clamping Gripper	Gripper Position	Double	The current position of the gripper fingers. • **0**: Fully open • **100**: Fully closed
Generic Button/Pedal Presser	Presser State	String	The current state of the device as defined by its control signals. Possible values are **pressed** and **released**.

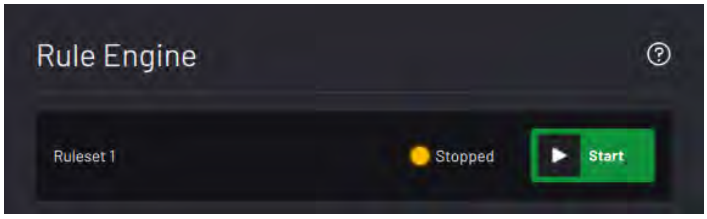
Chapter 9. Rule Engine

The Rule Engine is where you link device signals, system variables, and "if…then" statements. Run these processes in the background of Forge/OS, even when a Task Canvas task isn't executing! This app is great for instructing stack lights to be one color when a task is running and another when the task is idle (and for many other uses).

When you first open the Rule Engine, the Load Ruleset screen appears. Once you create and save rulesets, they show up in the table.



At the top of the screen, view the **Active** or **Stopped** status of the ruleset that most recently ran. **Stop** an Active ruleset and **Start** a Stopped ruleset here or in that ruleset's manager.



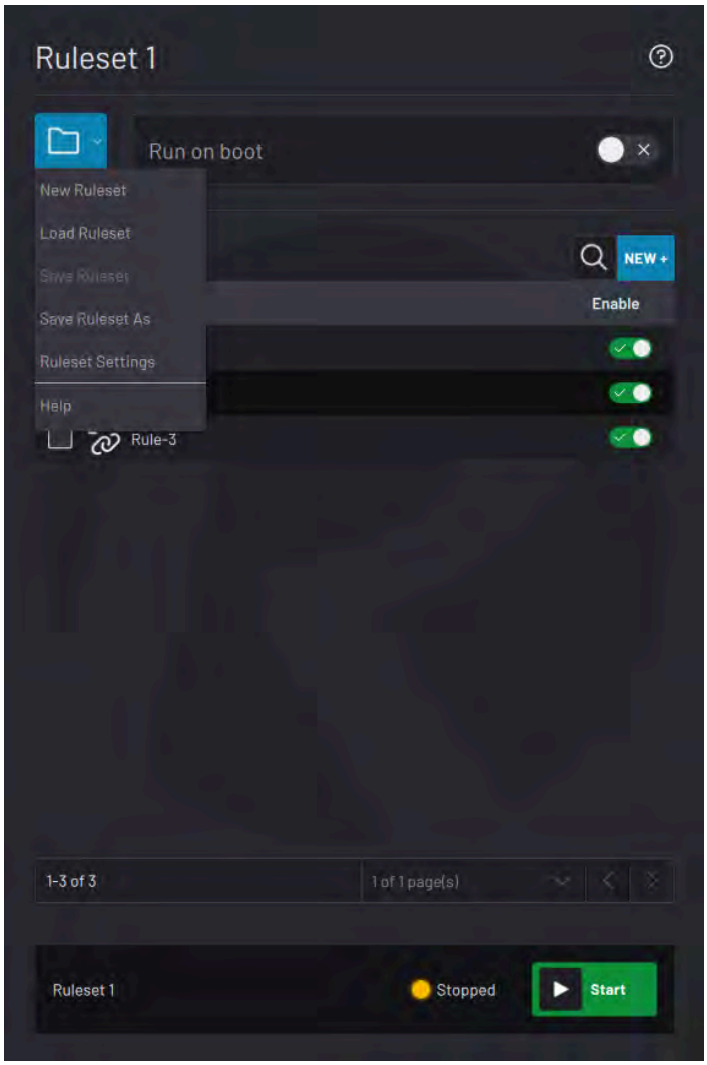
Ruleset Management

Inside a ruleset's manager, there is a **File menu** on the left side. The File menu lets you create or load a ruleset, save the ruleset, open ruleset settings, and access Help.

Enable the **Run on boot** toggle switch to automatically start the ruleset when ForgeOS launches.

In the table, manage the ruleset's rules. See Rule Management *(on page 115)*.

At the bottom of the screen, tap **Start** to begin running the ruleset's rules. Tap **Stop** to stop running the ruleset's rules.



New Ruleset

To create a ruleset, select **NEW +** from the Rule Engine main screen or **New Ruleset** from the File Menu.

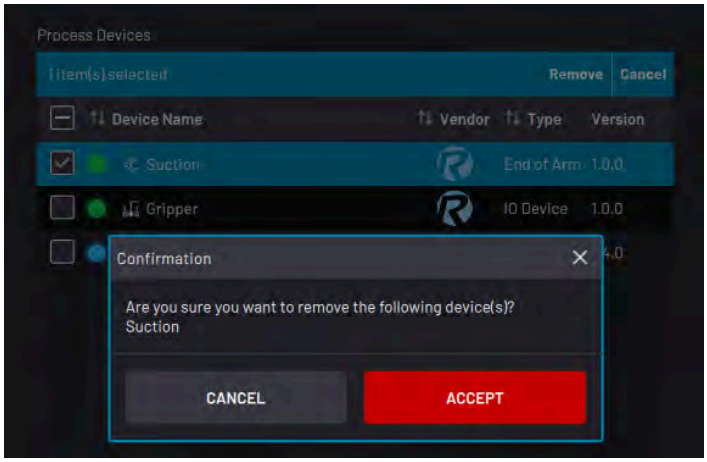
Give the ruleset a unique name.

The ruleset will access all devices that are listed in the table. By default, the device table includes all devices that are enabled in Device Configuration.

> **Note:**
> At the bottom, tap **Add Tag** to make the ruleset easier to find later. When searching for the ruleset, enter a tag to filter the search results.

Create a ruleset with no devices or with many.

Remove devices by selecting each unwanted device and tapping **Remove**.

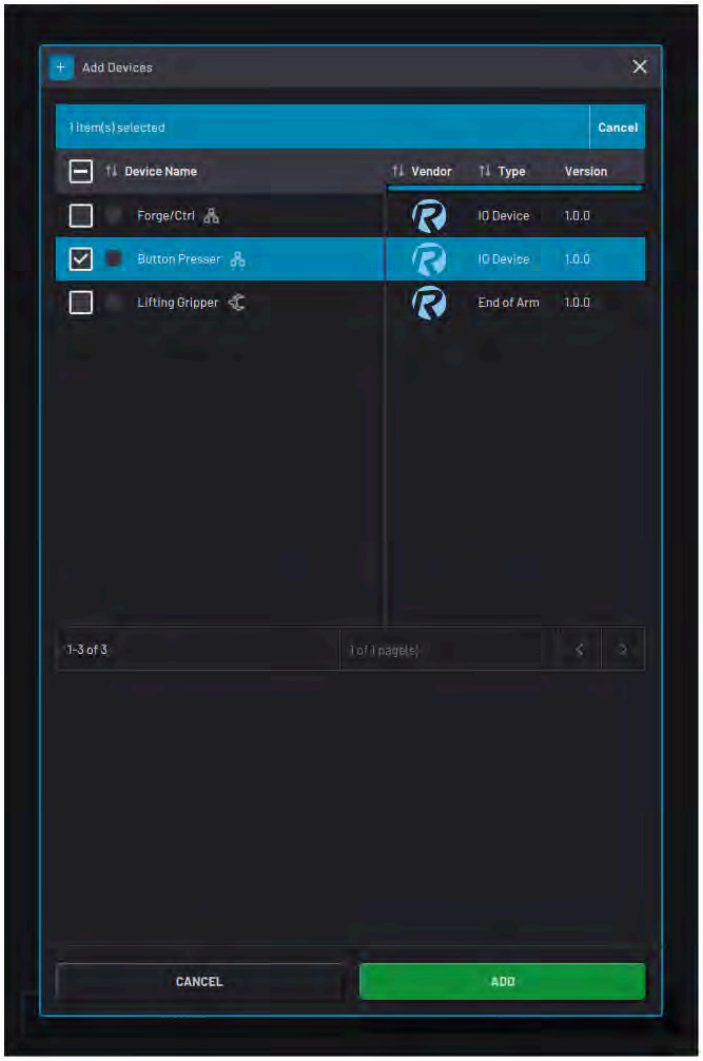


> **Note:**
> Tapping Remove here hides the selected device(s) from this ruleset's list of devices. It does NOT remove the device(s) from the Device Configuration app.

To add a device to the ruleset, tap **ADD +**. Select the checkbox next to each device that you want to add and tap **ADD**.

> **Tip:**
> You can add or remove more devices later from Ruleset Settings *(on page 114)*.

Tap **CREATE RULESET**. ForgeOS directs you that ruleset's manager.



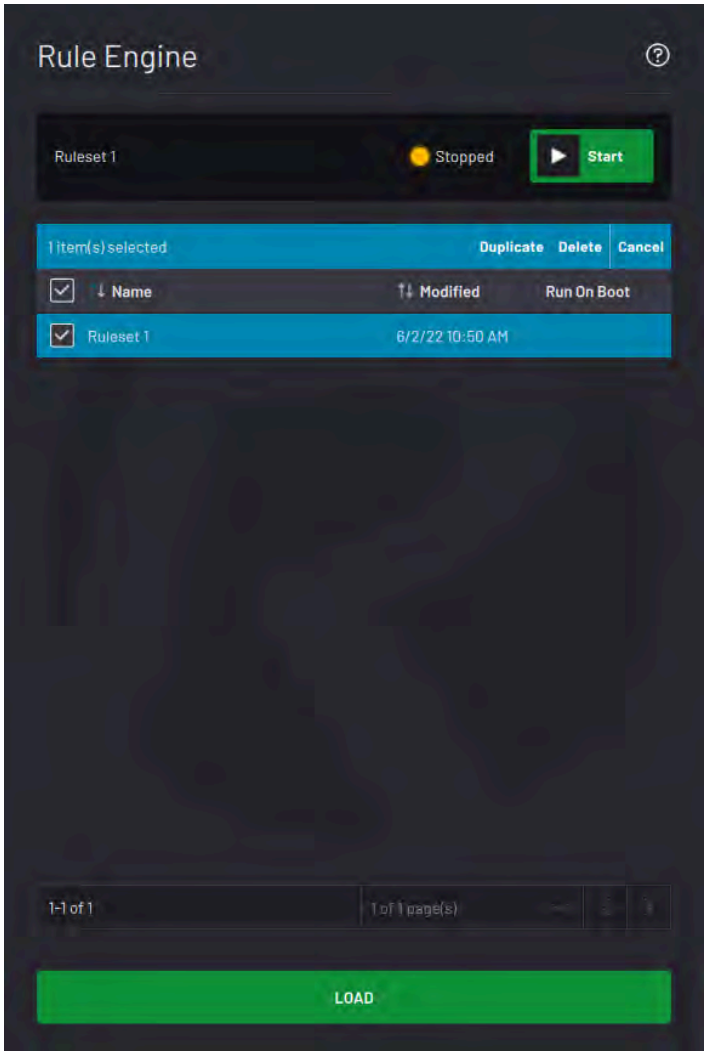
Load Ruleset

Load a ruleset from the Rule Engine main screen or from the File Menu. The Load Ruleset screen displays the list of saved rulesets.

Select a ruleset in the table and tap **LOAD** to open it.

Tap **Duplicate** in the table header to create a copy of the selected ruleset(s).

Tap **Delete** in the table header to remove the selected ruleset(s).



Save Ruleset

From the File menu, tap **Save Ruleset** to save the changes you made to the open ruleset. You can then load another ruleset without losing your work.

Tap **Save Ruleset As** to save the ruleset with a new name. After saving a ruleset with a new name, you are editing the new ruleset.

Ruleset Settings

From the File Menu, tap **Ruleset Settings** to change the settings of your open ruleset.

In the Name field, type in a name to rename the ruleset.

To add a device to the ruleset, tap **ADD +**. Select the checkbox next to each device that you want to add and tap **ADD**.

In the Ruleset Devices table, select the checkboxes of devices that you want to remove. In the table header, tap **Remove**.

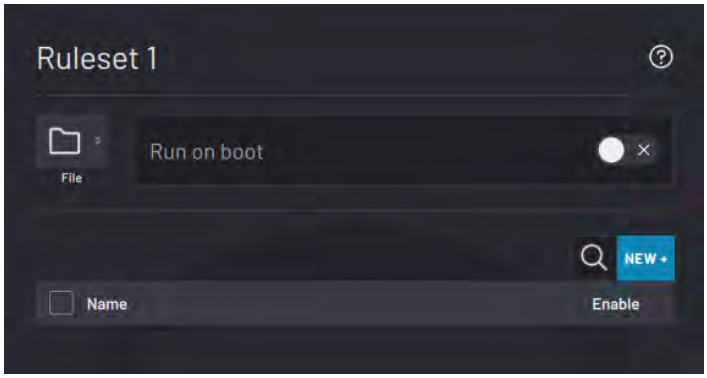
> **Note:**
> Removing a device here does NOT remove it from other rulesets or from the Device Configuration app.

Tap **APPLY AND SAVE** to save the changes and return to the ruleset's manager.

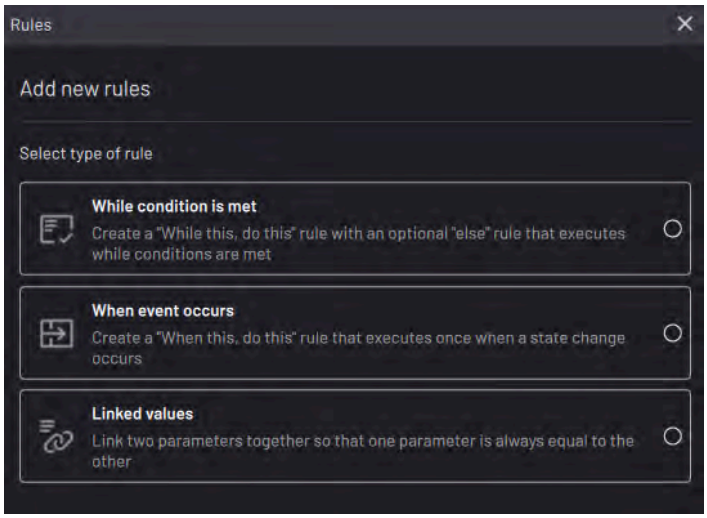
Rule Management

Inside a ruleset's manager, create, enable, edit, and delete that ruleset's rule(s).

Tap **NEW +** to create a new rule.



Decide if you want to add a "While condition is met", "When event occurs", or "Linked values" rule.



Select "**While condition is met**" to create a "While this... do this" rule with an optional "else" rule that executes while conditions are met.

In the **When this** tab, tap **ADD +** to add conditions. In the **Conditions met when** dropdown, select:

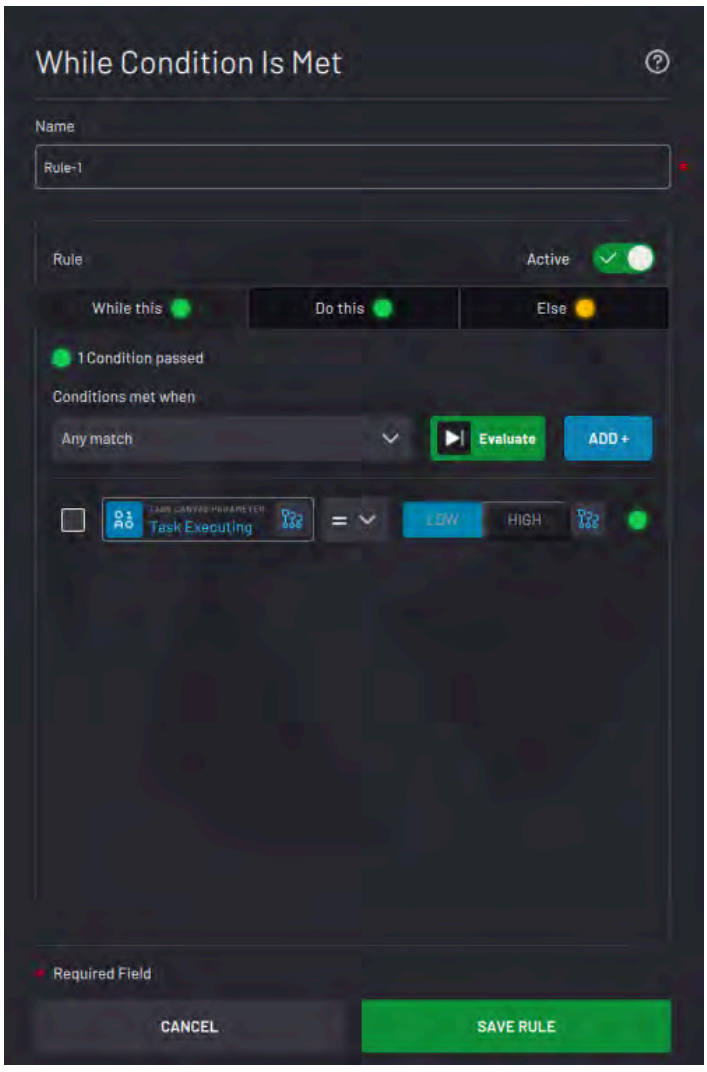
- **All match**: All lines must pass to perform the "Do this" assignment.
- **Any match**: Any of the lines must pass to perform the "Do this" assignment.

In the **Do this** tab, tap **ADD +** to assign actions for while the conditions are met. For example, turn a stack light one color while a task is running.

In the **Else** tab, tap **ADD +** to assign alternative actions if the conditions are not met. For example, turn a stack light another color while a task is not running.

To see which conditions are satisfied and which ones are not, tap **Evaluate**. Conditions that pass have a green light next to them. Conditions that fail have a yellow light next to them.

To delete a condition, select it in the table. The ADD + button turns into a red trash can. Tap the trash can and confirm **DELETE**.



Select "**When event occurs**" to create a "When this... do this" rule that executes once when a state change occurs.

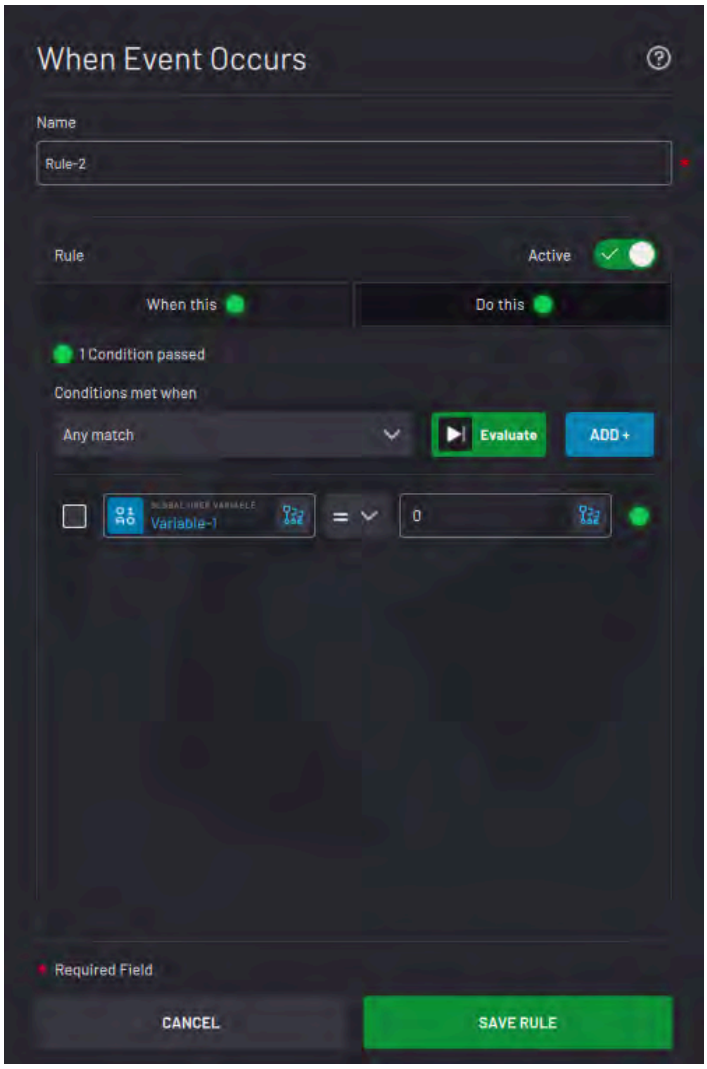
In the **When this** tab, tap **ADD +** to add conditions. In the **Conditions met when** dropdown, select:

- **All match**: All lines must pass to perform the "Do this" assignment.
- **Any match**: Any of the lines must pass to perform the "Do this" assignment.

In the **Do this** tab, tap **ADD +** to assign actions for as soon as the conditions are met. For example, set the value of Variable 2 as soon as Variable 1 reaches a certain value.

To see which conditions are satisfied and which ones are not, tap **Evaluate**. Conditions that pass have a green light next to them. Conditions that fail have a yellow light next to them.

To delete a condition, select it in the table. The ADD + button turns into a red trash can. Tap the trash can and confirm **DELETE**.



Select "**Linked values**" to link two parameters together so that one parameter is always equal to the other.

Tap **ADD +** to add assignments. Set the **Target** and **Source** fields. For example, set the value of Variable 1 to equal Variable 2.

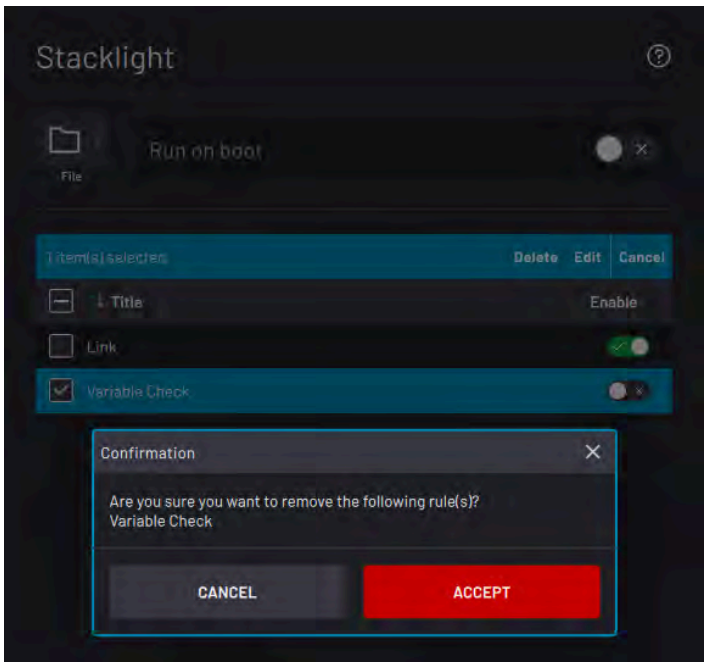
To delete an assignment, select it in the table. The ADD + button turns into a red trash can. Tap the trash can and confirm **DELETE**.

Tap **SAVE RULE** for the rule to appear in that ruleset's manager.

To make a rule active when the ruleset is run, toggle the **Enable** switch next to that rule in the ruleset's manager. Or toggle the **Active** switch in the rule's editor.

To edit a rule, select it in the table and tap **Edit** in the table header.

To remove one or more rules, tap **Delete** in the table header.

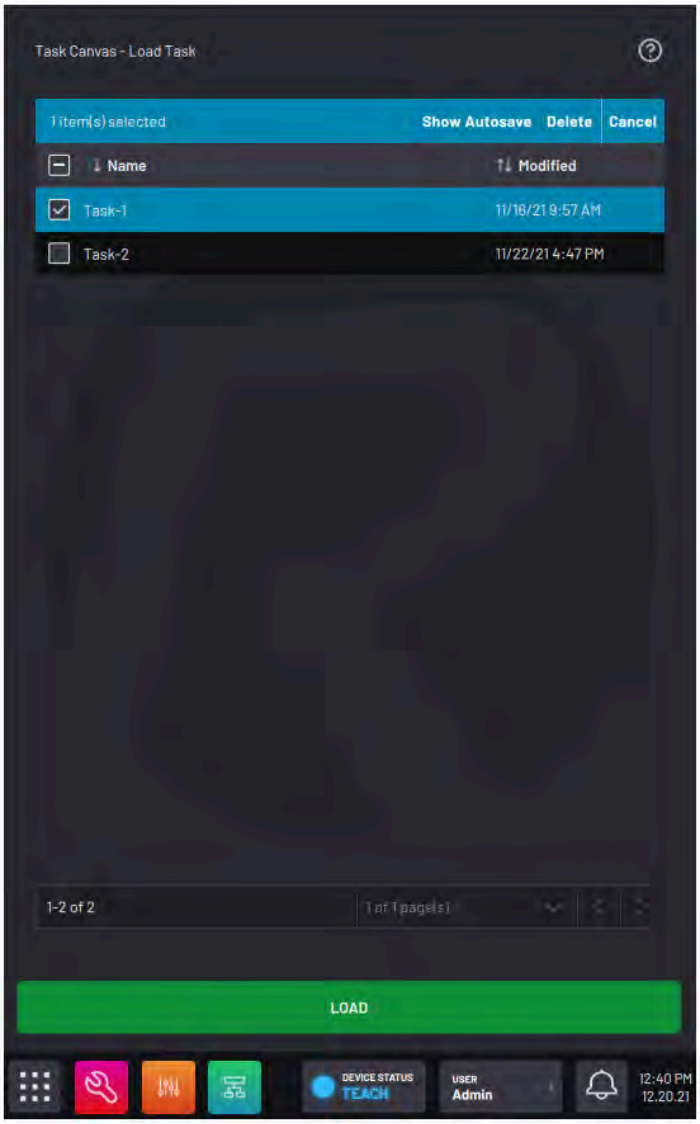


Chapter 10. Task Canvas

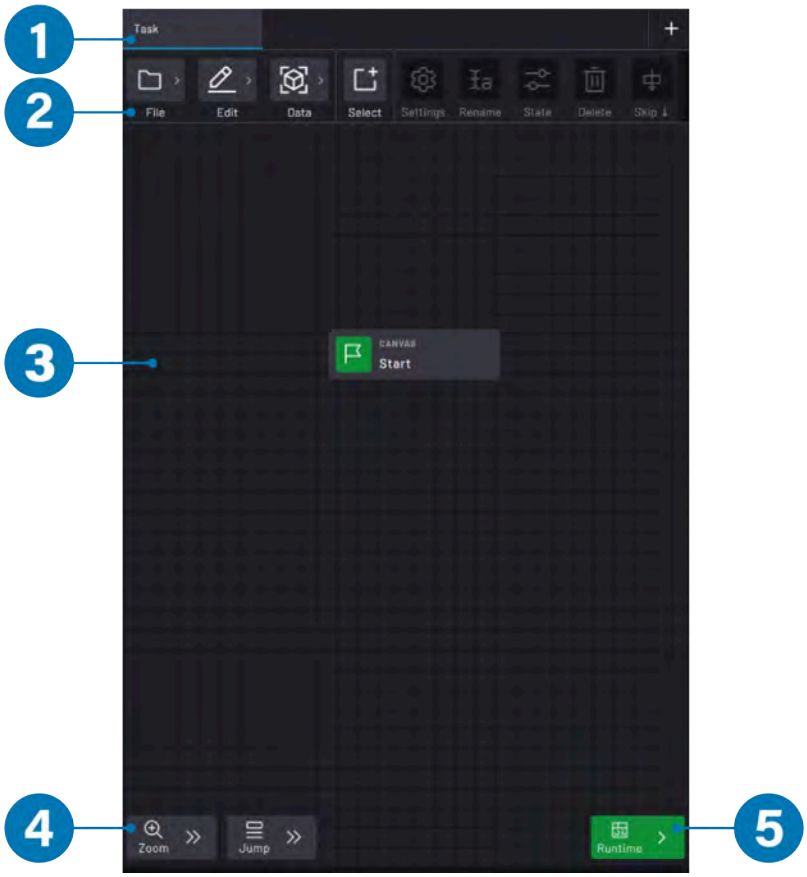
Overview

The Task Canvas app is where you program and execute tasks using your configured devices. Task Canvas controls devices using **blocks**. **Paths** link blocks together to form a **flowchart**. The flowchart executes each block until you stop the task or until the flowchart reaches a final block.

When you open Task Canvas, the Load Task screen appears. This is where you create or load a task. Once you create and save tasks, they show up in the Load Task table.



While creating a task, you'll see the **Subtask Tab Bar**, the **Canvas Menu**, the **Canvas**, the **Runtime Controls Menu**, and the **Navigation Tools**.



No.	Section	Description
1	Subtask Tab Bar	See the name of the open task and subtasks. Create subtasks by tapping the **+** sign in the right-hand corner. As you add subtasks, this bar fills with tabs that allow you to access those subtasks.
2	Canvas Menu	Access the **File Menu**, **Data Menu**, **Select Mode Toggle**, and **Block Editor**.
3	Canvas	Add and organize blocks in a flowchart to create the task.
4	Navigation Tools	Navigate the Canvas like a map. Reset the view, zoom in or out, go back to the Start block or last executed block, and Search existing blocks.
5	Runtime Controls Menu	Execute the task.

Canvas Menu

The Canvas Menu includes the **File Menu**, **Data Menu**, **Select Mode Toggle**, and **Block Editor**.

No.	Name	Description
1	File Menu	• Tap **New Task** to create a new task. • Tap **Load Task** to open the Load Task screen. • Tap **Save Task** to save the open task. • Tap **Save Task As** to save the open task as a new file and open the new task. • Tap **Task Settings** to change the task information (name, devices, and tags). Save the task to commit the updated settings. • Tap **Help** to browse information about Task Canvas. • Tap **New Subtask** to create a new subtask in the open task. The subtask appears in a tab across the top of the menus. • Tap **Rename Subtask** to rename the open subtask. To rename the Main task, use Task Settings. • Tap **Duplicate Subtask** to duplicate the open sub-task, including its entire flowchart. • Tap **Delete Subtask** to delete the open subtask. This does not delete parameters used in the sub-task, such as waypoints and variables. • Tap **Export Subtask** to export the open subtask as a new task file. This exports the flowchart and any parameters used in it. • Tap **Import Subtask** to import an existing task or subtask into the open task as a new subtask.

No.	Name	Description
2	Data Menu	• Tap **Waypoint Manager** to view, create, and modify waypoints local to the open task. • Tap **Frame Manager** to view, create, and modify reference frames local to the open task. • Tap **Variable Manager** to view, create, and modify variables local to the task. It also displays variables owned by devices (like fieldbus devices).
3	Select Mode Toggle	Toggle to tap/select multiple blocks. Delete, skip, or unskip multiple blocks at one time.
4	Block Editor	Change a selected block's settings, rename it, change its execution state, add comments, or delete it. On the right, tap the arrow labeled **More**/**Close** to expand or collapse the menu. In the expanded section: • Tap **Undo** to undo an action (such as block creation, block deletion, or path change). **Note:** Undo stores up to 32 steps that can be undone. • Tap **Redo** to redo an action that was previously undone. • **Cut/Copy** the highlighted block. Tap the location where you want the new block to be and tap **Paste Here**. Tap the green check mark labeled **Paste** to confirm. • **Skip** a block. Choose if the task moves on to that block's bottom or right path. Tap **Unskip** to restore the block.

Canvas Navigation

Drag a finger or stylus across the touch screen to pan and scroll. You can also use the Canvas Navigation Tools at the bottom of the screen. There are two Navigation Tool menus. **Zoom** provides canvas view settings. **Jump** centers the view on a specific block.

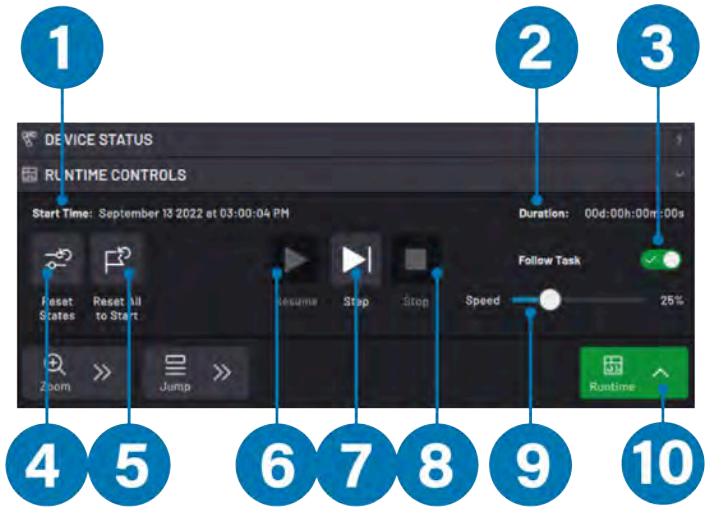
Zoom Menu	Description
Reset	Reset the view to the default zoom level.
Zoom Out	Zoom out to view more of the flowchart.
Zoom In	Zoom in to get a closer view of the flowchart.

Jump Menu	Description
Search	Search for blocks on the canvas. Type an entry to search through: • Block types (e.g. Count or Absolute Move) • Custom block names (from the "Rename" button) • Block devices • Internal parameters (e.g. variables or waypoints)
Back to Start	Jump to the Start block.
Last Executed	Jump to the Last Executed block. The Last Executed block is tagged with a magenta "Step" icon.

Runtime Controls

The Runtime Controls menu allows you to execute the task. Tap the **Runtime Controls** button to open or close the menu. The menu remains expanded when a task is executing.

For more on executing a task, see Running a Task *(on page 165)*.

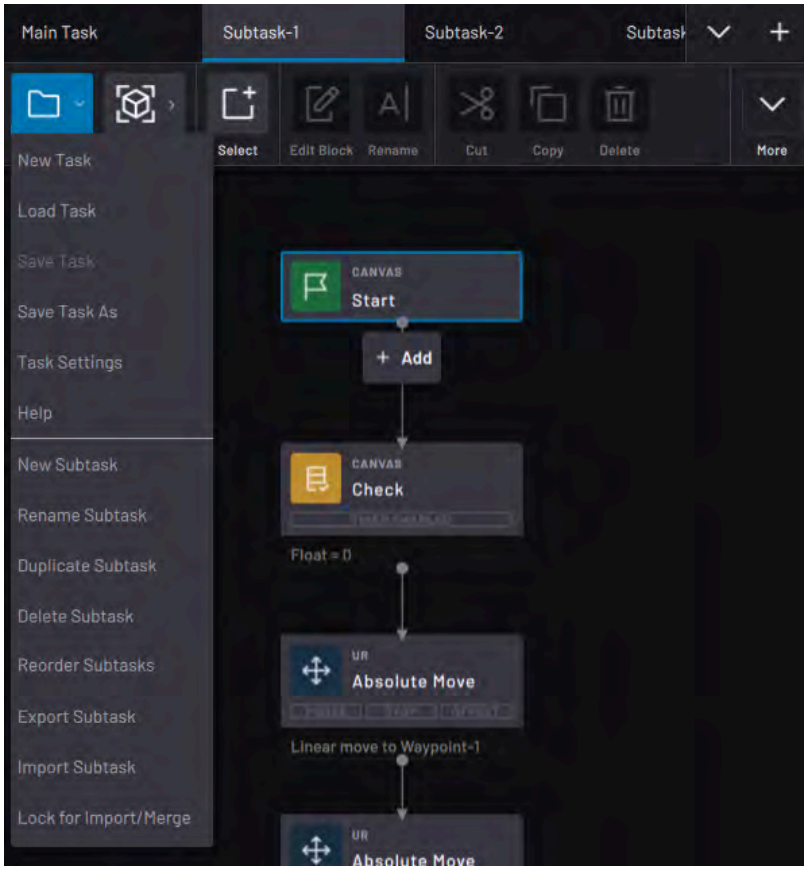


No.	Feature	Description
1	Start Time	View the time that one of the execution buttons was last pressed (**Start Task**, **Start Subtask**, or **Step**).
2	Duration	View the total execution time of the task. Like a stopwatch, it resets to 0 when you tap **Start Task** or **Start Subtask** and pauses when the task stops.
3	Follow Task	Tap the **Follow Task** toggle to center the view on the executing block during Task execution. Follow Task toggles ON whenever a task stops, but you can toggle it OFF before execution.
4	Reset States	Tap **Reset States** to set all incremental blocks (e.g., Wait, Count, Timer, Grid, and Array blocks) to their initial states. This button also removes any paused states of blocks (such as paused Continuous Move blocks) and resets the values of task variables.
5	Reset All to Start	Tap **Reset All to Start** to do everything that Reset States can do plus clear the queue of paused sub-tasks and jump to the Main Task's Start block.
6	Start Task OR Start Subtask OR Resume	Tap Start Task to start the task from the selected block with the current block states. For this button to be available, you must have a block selected. If there are any blocks with errors, you must **Skip** them. All referenced devices must be enabled. If you are using a robot, you must put it in **Run** mode. If you have a subtask open, this button reads **Start Subtask**. In this case, only the visible flowchart will run. If you've executed blocks and have not tapped **Reset All to Start**, the button reads **Resume**. If you are working with subtasks, a pop-up may ask which subtask(s) to resume.
7	Step	Tap **Step** to execute only the selected block.

No.	Feature	Description
8	Stop	Tap **Stop** to stop the execution of a block or the Main Task and all subtasks. Stopping a task or block maintains the iterative state of any blocks. The Stop button executes a controlled stop, which is the easiest stop method on mechanical devices in the task. **Important:** Never use the Stop button in an emergency or when an operator's safety is at risk. In such cases, use the emergency stop (E-Stop) button.
9	Speed Slider	Use the **Speed Slider** to scale the speed of all robot motion in the task for all robots. If you program an Absolute Move block with 60% acceleration and 80% speed, then you set the Speed Slider to 50%, the block executes with 30% acceleration and 40% speed. When devices in the task are in **Teach** mode, the speed slider defaults to 25% speed. When devices are set to **Run**, the speed slider defaults to 100% speed.
10	Runtime Controls Button	Tap the **Runtime** button to expand or hide the Runtime Controls.

Task Management

On the left side of the Canvas Menu is the **File menu**. The File menu lets you create or load a task, save the task, open task settings, access Help, and manage subtasks.



New Task

To create a task, select **NEW +** from the Task Canvas Load Task screen or **New Task** from the File Menu.

Give the task a unique name.

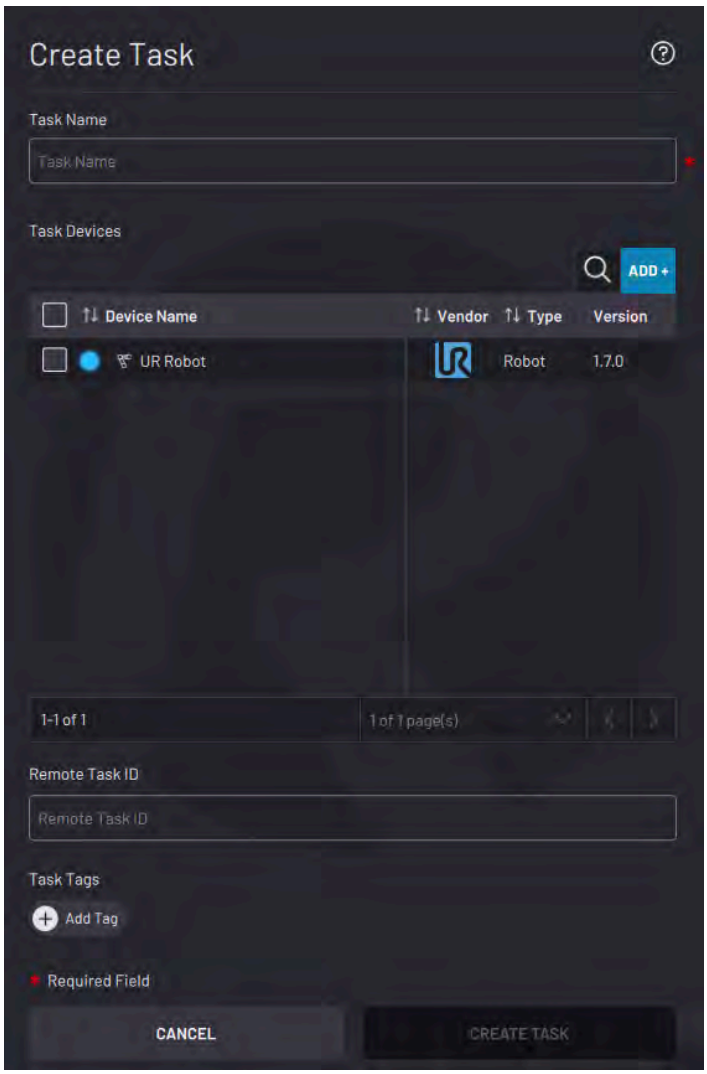
> **Note:**
> Your task name cannot include one or more of the following characters: ` ~ ! # %^ & * \ | ' " : ; ? / < >

The task will access all devices that are listed in the table.

By default, the device table includes all devices that are enabled in Device Configuration.

If you are using Remote Control mode, type in a **Remote Task ID** to allow the Remote Control Device to identify which task to load. The ID must be a unique, positive integer.

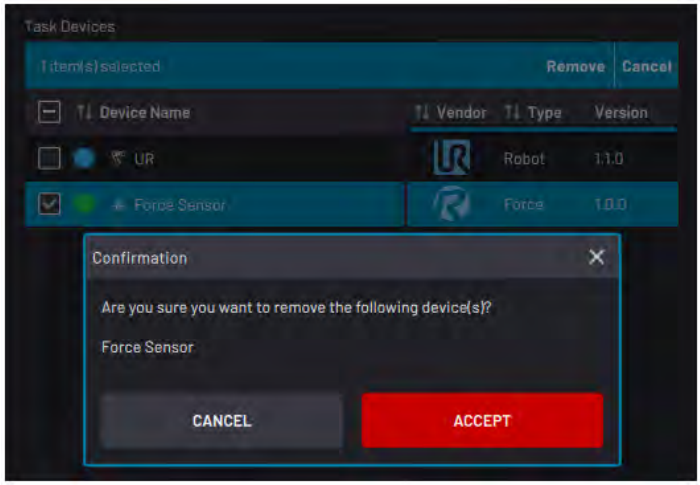
> **Tip:**
> At the bottom, tap **Add Tag** to make the task easier to find later. When searching for the task, enter a tag to filter the search results.



Create a task with no devices or with many.

Remove devices by selecting each unwanted device and tapping **Remove**.

> **Note:**
> Tapping **Remove** here hides the selected device(s) from this task's list of devices. It does NOT remove the device(s) from the Device Configuration app.

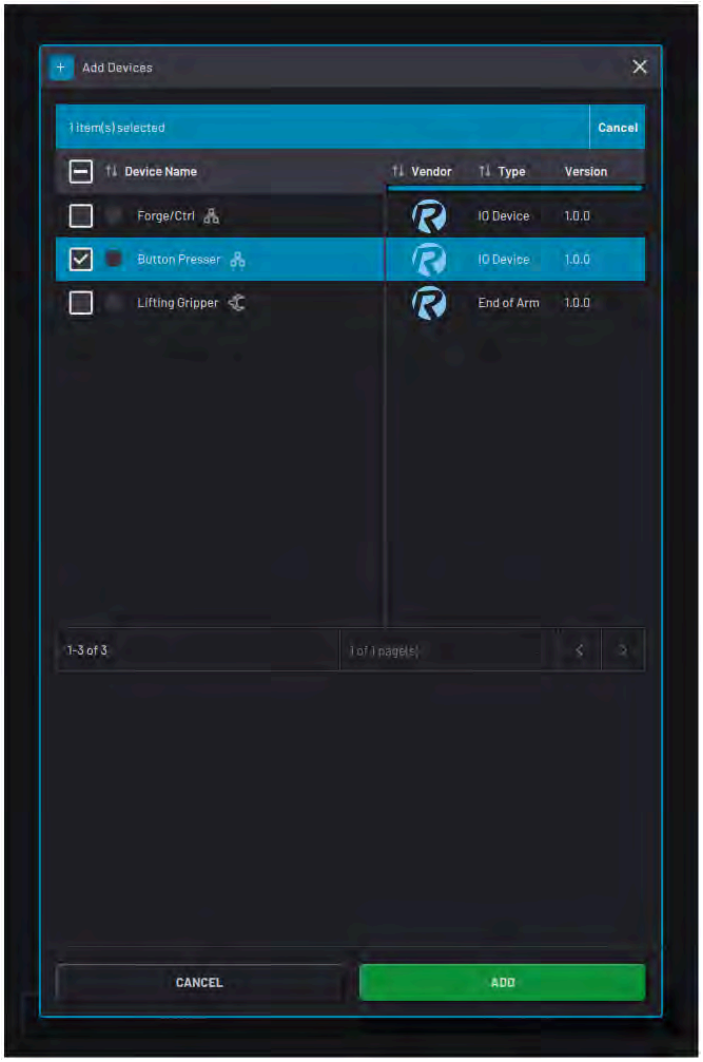


To add a device to the task, tap **ADD +**.

Select the checkbox next to each device that you want to add and tap **ADD**.

> **Note:**
> You can add or remove more devices later from Task Settings *(on page 133)*.

Tap **CREATE TASK**. ForgeOS directs you to a blank Canvas.



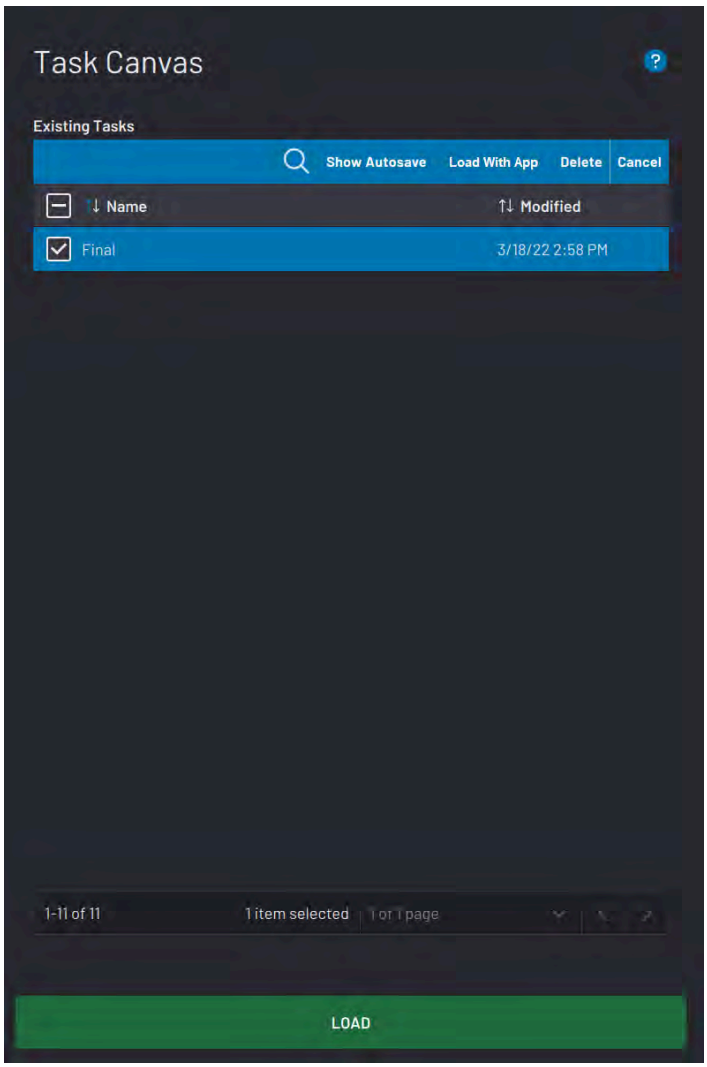
Load Task

Load a task from the Task Canvas main screen or from the File Menu.

> **Note:**
> If you do not want operators to have the ability to choose which task to load, you can instead use the **Load with App** feature. This feature loads a task when Task Canvas launches.

The Load Task screen displays the list of saved tasks. Select a task in the table and tap **LOAD** to open it.

View an automatic backup of the selected task by tapping **Show Autosave** at the top of the table. If you load an autosave file and tap **Save** from the File Menu, it saves over the main file for that task.



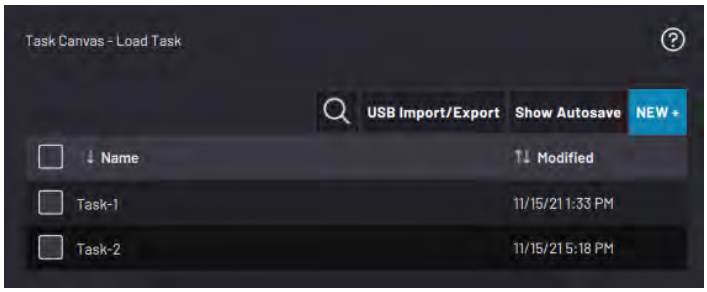
Note:
Every task has an autosave version that updates after every change. If you lose power to your system without saving your task, the autosave version contains the most recent changes.

Remove the selected task by tapping **Delete** at the top of the table.

Important:
Deleting a task cannot be undone. The only way to recover a deleted task is to load the autosave within 14 days and save it as a new file.

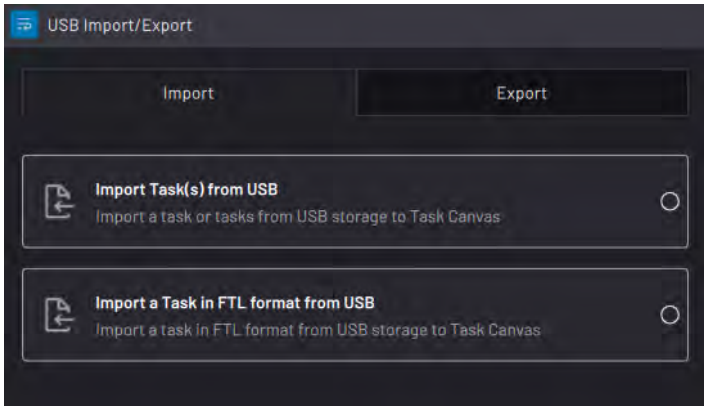
To import or export a task, use a USB flash drive that has at least 1 GB of storage and is formatted to FAT32, NTFS, or ext4.

Tap **USB Import/Export** at the top of the table.



Choose if you want to import or export a task.

Then tap **NEXT**.



Importing Tasks

Imported tasks include all local and global parameters used in the task. The imported file is a copy, so the original file stays on the USB flash drive.

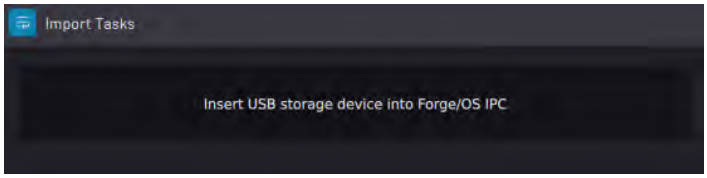
There are two types of formats that you can import:

- Task files (that end in “.task”)
- FTL files (that end in “.ftl”)

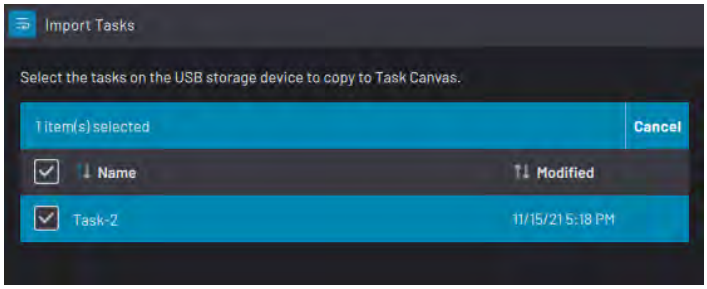
Importing Task Files

Select **Import Task(s) from USB** to import tasks that were normally programmed on a READY pendant and exported to a USB drive.

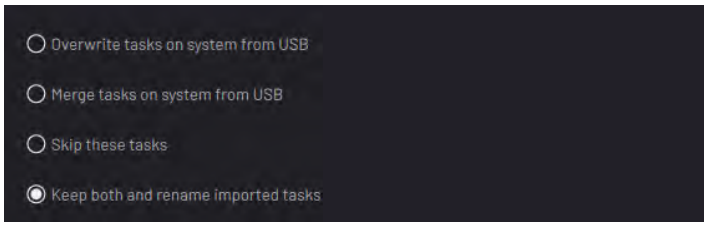
Plug the USB flash drive with your ".task" file(s) into your IPC. Once the system detects your USB flash drive, tap **NEXT**.



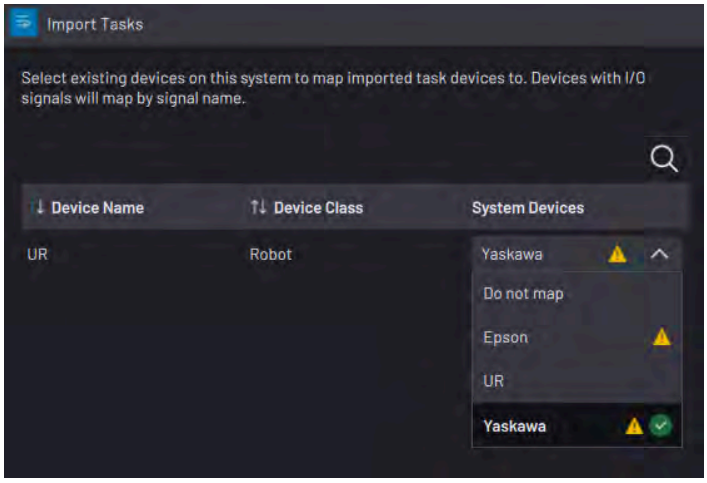
Select the task(s) you want to import. Then tap **NEXT**.



If Task Canvas already has tasks or autosaved tasks with the same names as files that you are trying to import, choose if you want to overwrite the tasks on the system, merge the tasks, skip those tasks, or import the new tasks with “-copy” added to the end of the names.

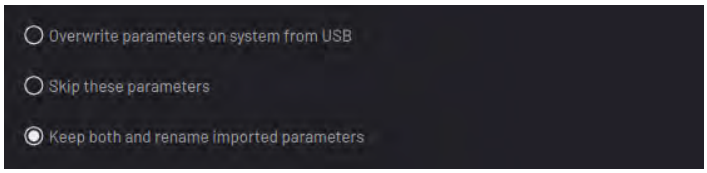


Use the **System Devices** dropdowns to map the task's devices to your system's configured devices. The remapped device must be of the same device class (e.g., robot, gripper, sensor, etc.) as the original device. Then tap **NEXT**.



Important:
If you are remapping to a different type of robot, update waypoints in the task's Waypoint Manager to avoid collisions, singularities, or joint limits. Some parameters (such as Set blocks for robot I/O) may require manual reassignment in that block's settings.

If the ForgeOS system already has parameters with the same names as parameters that you are trying to import, choose if you want to overwrite the parameters on the system, skip those parameters, or import the new parameters with “-copy” added to the end of the names.

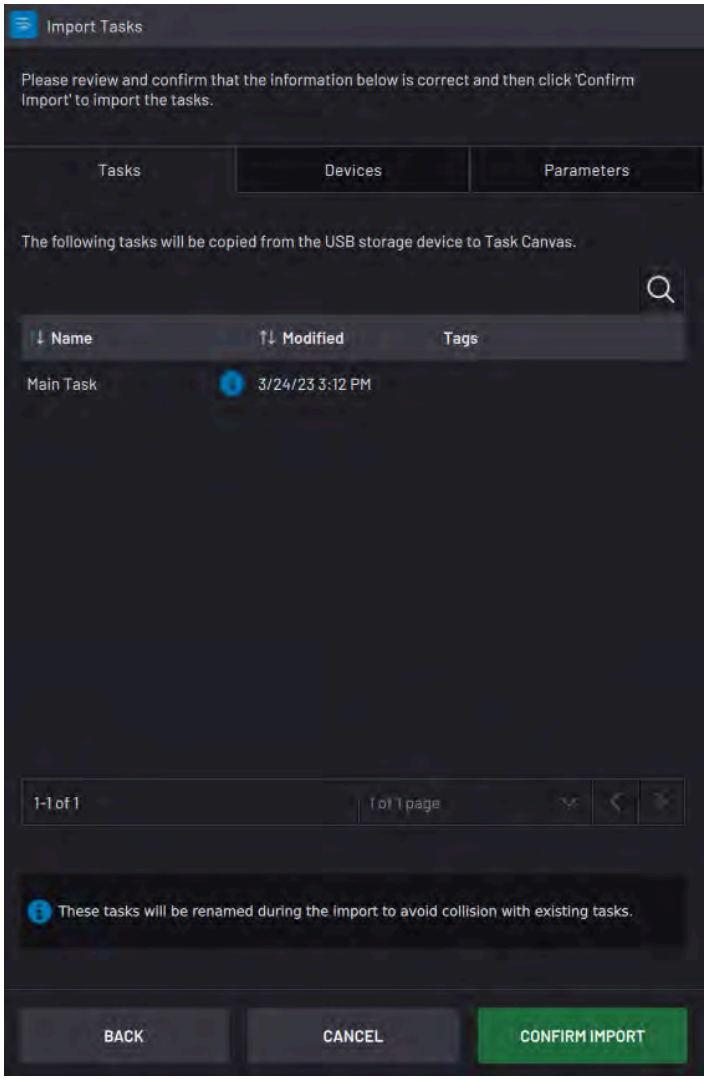


Review the information in the **Tasks**, **Devices**, and **Parameters** tabs.

Tap **BACK** if you need to make any changes.

If everything looks correct, tap **CONFIRM IMPORT**.

Once the import is complete, tap **FINISH**. Then remove the USB flash drive.



Importing FTL Files

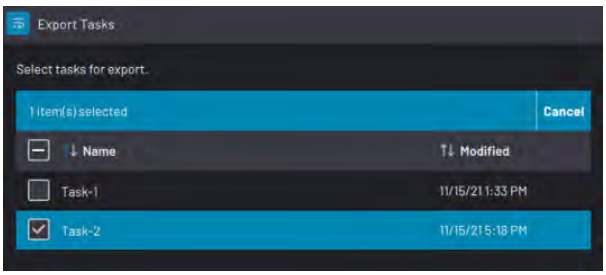
FTL stands for "**Forge Task Language**". FTL is a JSON specification that provides a common API for creating Task Canvas tasks. Select **Import Task(s) in FTL format from USB** to import tasks that were programmed by READY or by third-party developers from any programming language and exported to a USB drive.

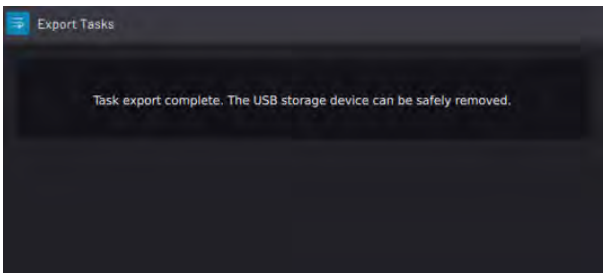
FTL merges imported tasks into existing tasks by overwriting locked subtasks and ignoring unlocked subtasks. For more info on locked and unlocked subtasks, see "Multitasking with Subtasks *(on page 139)*".

Exporting Tasks

Exported tasks include all local and global parameters used in the task. The exported file is a copy, so the original file stays on the READY pendant.

Plug the USB flash drive into your IPC.Once the system detects your USB flash drive, tap **NEXT**. Select the task(s) you want to export. Then tap **CONFIRM EXPORT**. Once the export is complete, tap **FINISH**. Remove the USB flash drive.





Save Task

From the File menu, tap **Save Task** to save the changes you made to the open task. You can then load another task without losing your work.

Tap **Save Task As** to save the task with a new name. After saving a task with a new name, you are editing the new task.

Task Settings

From the File Menu, tap **Task Settings** to change the settings of your open task.

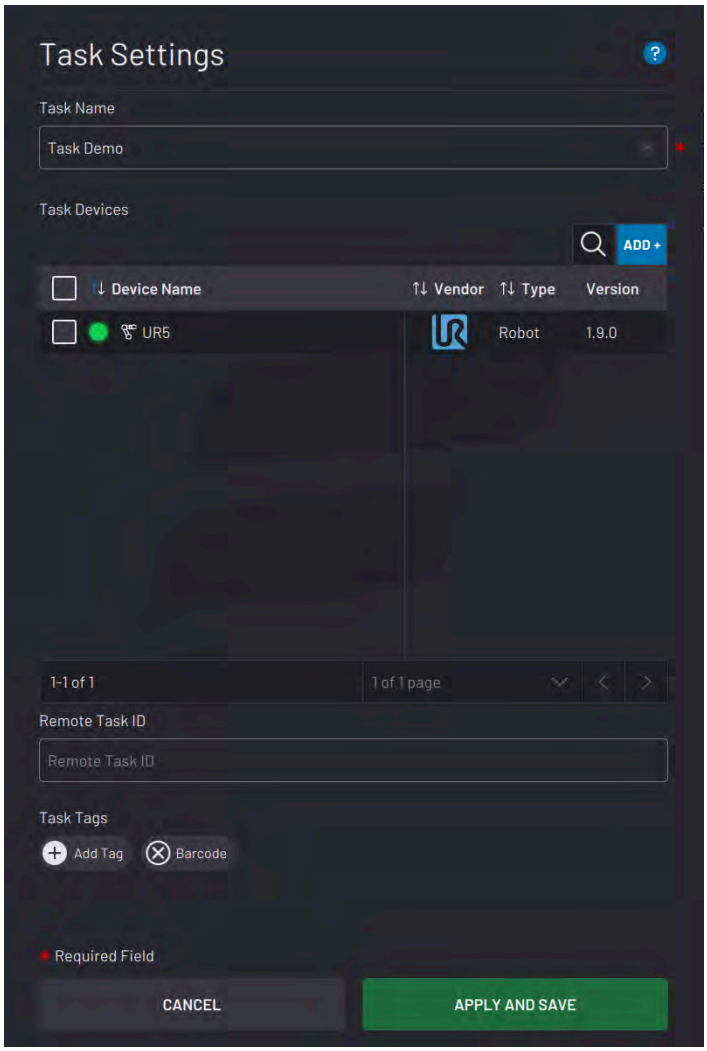
In the Task Name field, type in a name to rename the task.

> **Note:**
> Your task name cannot include one or more of the following characters: ` ~ ! # %^ & * \ | ' " : ; ? / < >

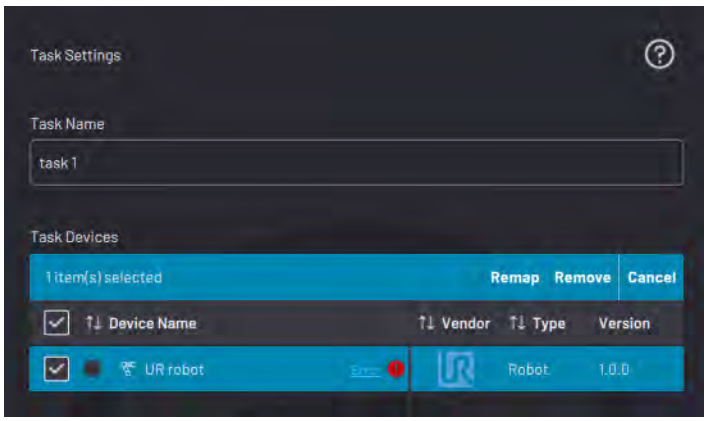
To add a device to the task, tap **ADD +**. Select the check-box next to each device that you want to add and tap **ADD**.

If you are using Remote Control mode, type in a **Remote Task ID** to allow the Remote Control Device to identify which task to load. The ID must be a unique, positive integer.

> **Tip:**
> At the bottom, tap **Add Tag** to make the task easier to find later. When searching for the task, enter a tag to filter the search results.



In the Task Devices table, select the checkboxes of devices that you want to remap or remove.

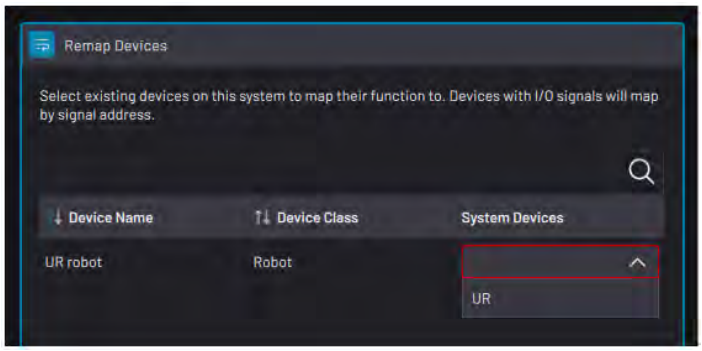


> **Note:**
> Removing a device here does NOT remove it from other tasks or from the Device Configuration app.

Tap **Remap** to reassign one or more devices to another device. Select the new device from the **System Devices** dropdown.

The remapped device must be of the same device class (e.g., robot, gripper, sensor, etc.) as the original device.

To save the device remap, tap **CONFIRM REMAP**.

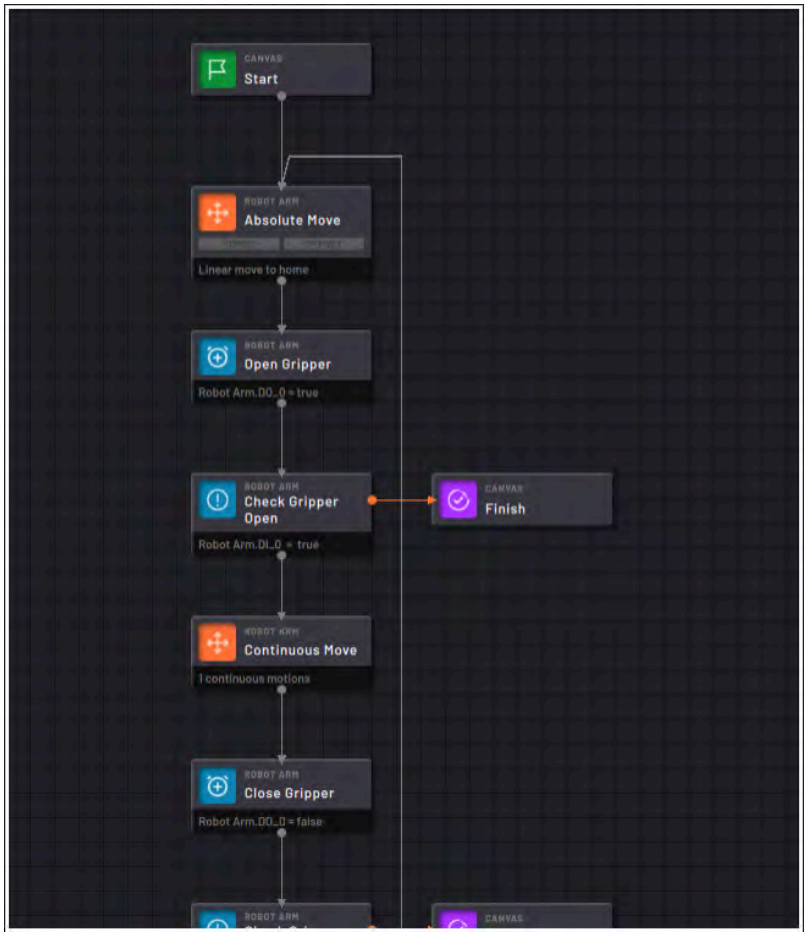


> **Important:**
> If you are remapping to a different type of robot, update waypoints in the task's Waypoint Manager. Some parameters (such as Set blocks for robot I/O) may require manual reassignment in that block's settings.

Tap **APPLY AND SAVE** to save the changes and return to the Canvas.

Features of a Task

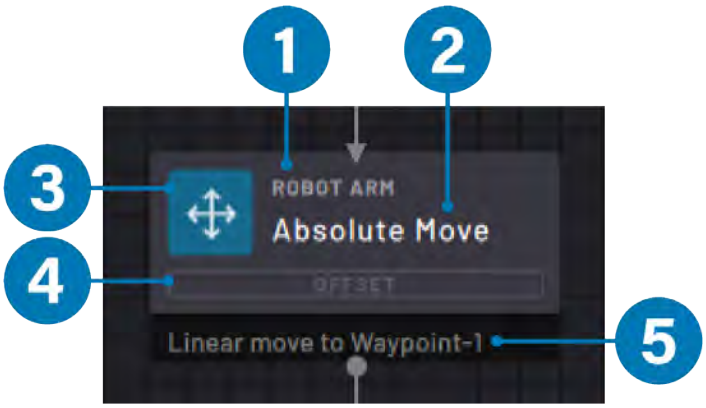
Create a **task** by combining blocks into a **flowchart**. Create additional flowcharts as **subtasks** to execute functions inline or at the same time as the main task. For more information about subtasks, see Multitasking with Subtasks *(on page 139)*.



Every flowchart begins with a **Start** block and follows the path of execution. Task Canvas performs each block's function before moving on to the next block. A flowchart ends with a **Finish** block or when there are no more blocks to execute.

Blocks

Each block is a function that Task Canvas executes. On the Canvas, a block displays several pieces of important information:



No.	Block Information	Description
1	Device	The device that owns the block and receives the command when the block executes.

No.	Block Information	Description
2	Block Name	The name given to the block. By default, each block is named after its block type or function. Rename blocks for reference while programming a task.
3	Block Type (Icon)	Shows the type of block for quick reference. Icons and colors depend on the block type and device type.
4	Modifiers	Any internal settings that change the way the block executes. Highlighted white if the block uses the modifier.
5	Execution State or Parameters	If the block is state-based (e.g., Timer, Count, Grid Move), this shows the execution state of the block. If the block isn't state-based (e.g., Absolute Move or Set), this shows the parameters that the block uses to execute.

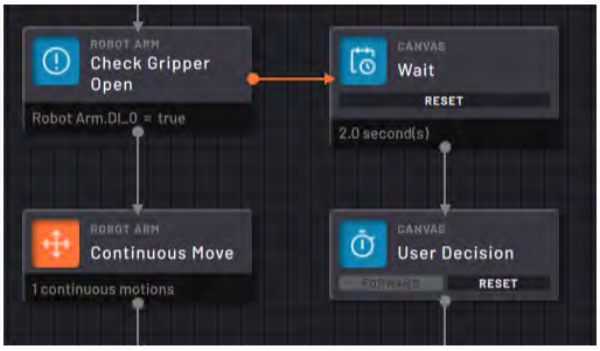
In the image above, the block device is "ROBOT ARM". The block name and function is "Absolute Move". There is no Force modifier available because there is no force sensor configured for that robot. The Offset modifier isn't highlighted because there are no offset values entered. The parameter defining where the robot moves to is "Waypoint-1".

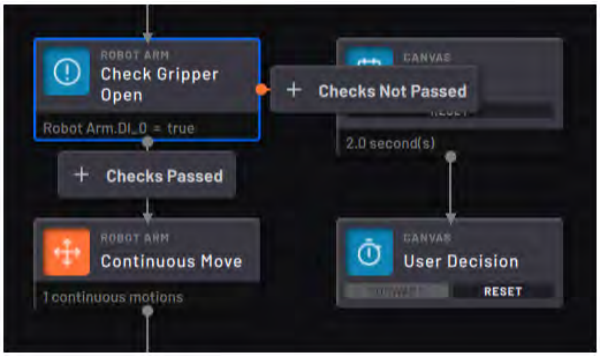
Program Paths

Paths connect blocks together to form the flowchart. Each block may have one or more paths leading to it. The number of possible exit paths from a block depends on the type of block.

Some blocks have two possible outcomes: a bottom gray path or a right orange path. Other blocks have one possible outcome: the bottom gray path.

See the images below for an example. The block named "Check Gripper Open" connects to the Continuous Move block through the bottom gray path and to the Wait block through the right orange path. In this case, the Check Gripper Open block is checking the state of gripper sensors. It follows the gray path if the gripper is open (Checks Passed) or it follows the right orange path if the gripper is closed (Checks Not Passed).

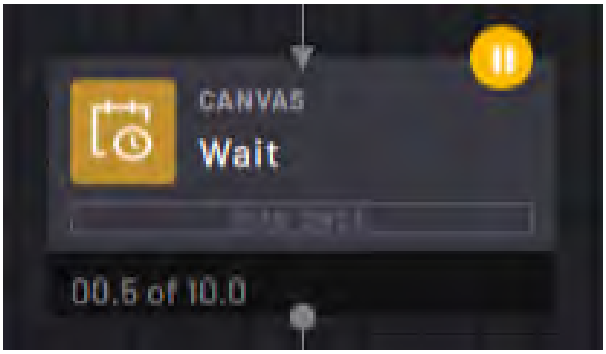




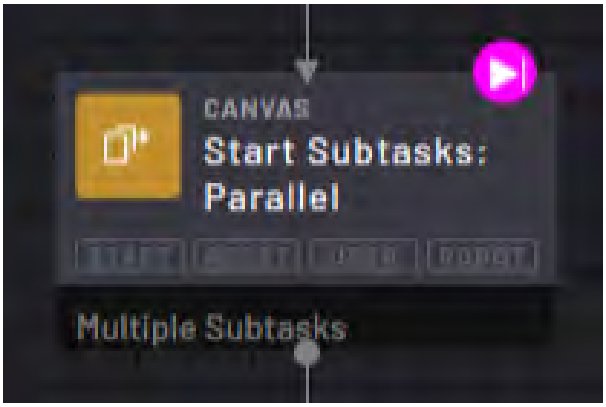
When you place a new block on the exit path of an existing block, Task Canvas creates the path connecting them. For information on how to create loops, disconnect paths, or reconnect paths, see Block Programming *(on page 142)*.

Special Block States

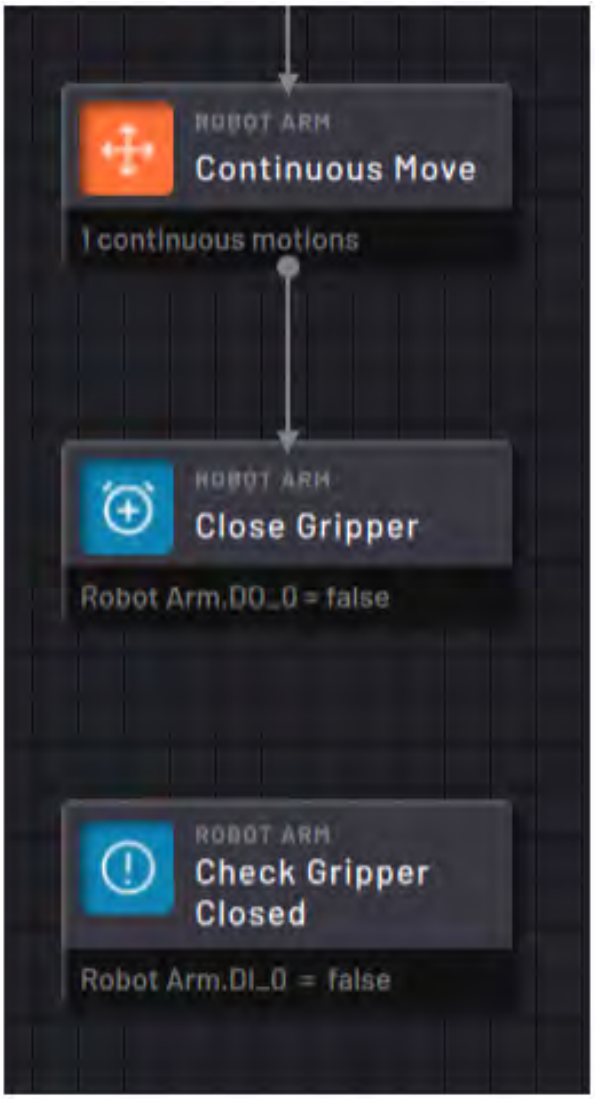
An incremental or robot motion block that was stopped mid-way through its execution displays a yellow **paused** icon. Press the **Last Executed** button in the Canvas Navigation menu to select and center the screen on the paused block.



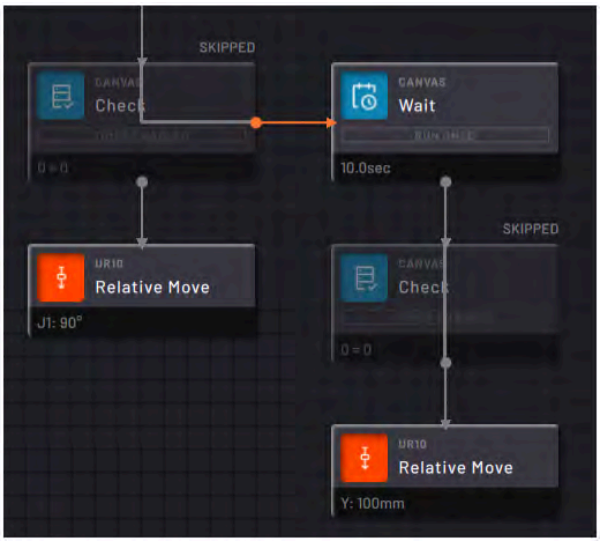
If the last block that was executed finished its execution when stopping (or stepping through) a task, it displays a magenta **Last Executed** icon. Press the **Last Executed** button in the Canvas Execution menu to select and center the screen on the last executed block.



A block that can't be traced back to the Start block through paths is a **Detached block**. You can modify and step Detached blocks, but they don't execute as part of a task since there is no path to them on the flowchart. Add a Detached block back to the task by connecting a path to it.

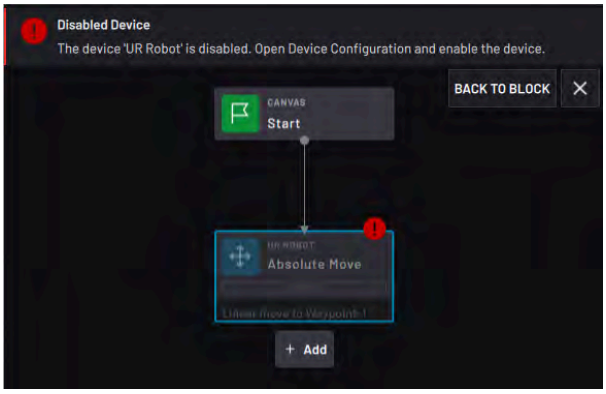


If you don't want your task to execute a block, but you're not ready to delete it, skip the block. A **Skipped block** stays connected to the flowchart, but it is passed over during task execution. Choose to skip to an available bottom or right path. Restore a block by tapping Unskip in the Block Editor.



A block that is missing a key parameter or device needed to execute is a **Template block**. A template block appears transparent with a red exclamation point. Template blocks can prevent the task from executing until you **Skip** them.

Tap the red exclamation point to view the block's error message. Tap **BACK TO BLOCK** to center the screen on that template block. Tap **X** to close the error message.



To restore a Template block:

- Fill in the missing parameters in that block settings.
- Enable the referenced device in Device Configuration.
- Remap the missing device to another device in Task Settings *(on page 133)*.

Multitasking with Subtasks

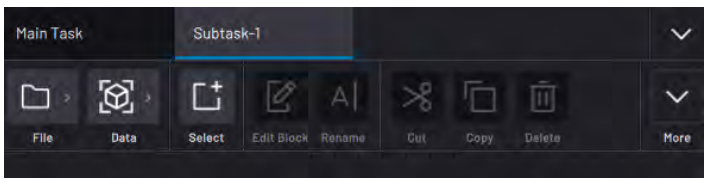
Task Canvas lets you create more than one canvas within a single task. Each of these canvases, or **subtasks**, has its own flowchart that can run inline or parallel to your **Main Task**. Use subtasks to multitask or to organize blocks into a functional group.

- A **parallel subtask** starts when a **Start: Parallel** block executes. The parallel subtask executes as the Main Task continues to run.
- An **inline subtask** starts when a **Start: Inline** block executes. The inline subtask executes its entire flowchart before the Main Task continues to run. An inline subtask returns control to the calling task through an **Exit** block. The Exit block decides if the calling subtask moves on to the bottom path or the right path from the Start: Inline block.

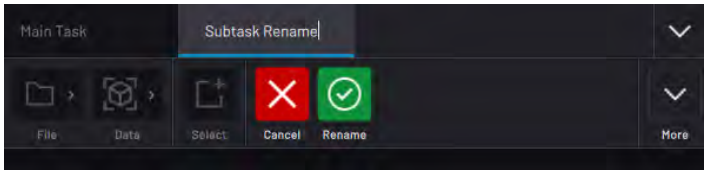
Some devices can execute commands across parallel subtasks, such as setting a digital output or variable value.

Some devices, including robots, can only receive a command from one flowchart at a time. For those devices, the block in one subtask queues and waits for the executing block in another to finish.

To add a subtask, tap **New Subtask** in the File menu. A new canvas appears for you to create the subtask flowchart.



To rename a subtask, open the subtask you want to rename and select **Rename Subtask** in the File Menu. Type

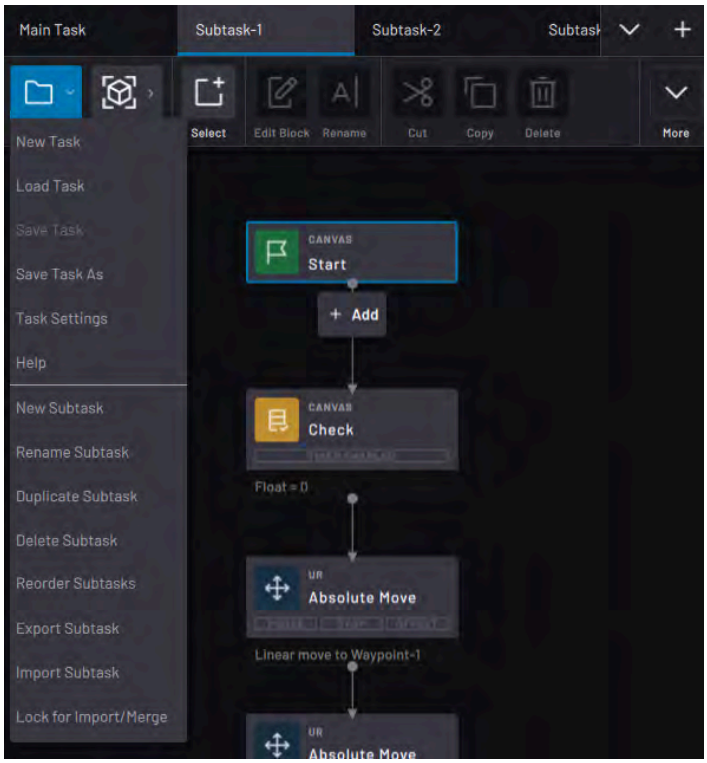


in the new name and then tap the green Rename check-mark in the top menu.

To duplicate a subtask, open the subtask you want to copy and select **Duplicate Subtask**.

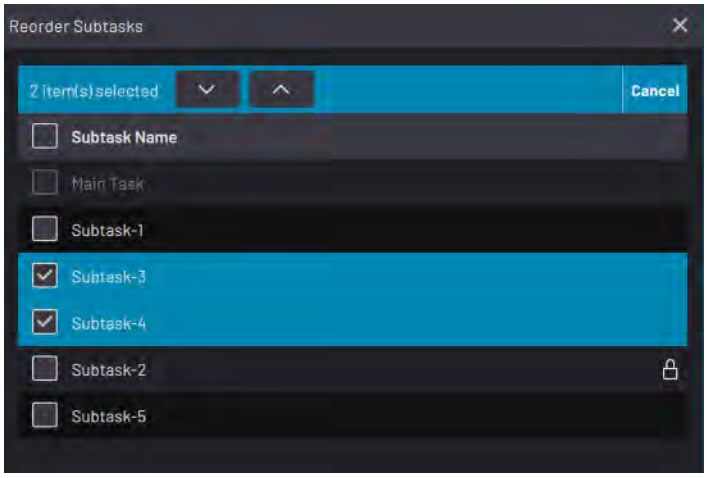
To delete a subtask, open the subtask you want to delete and select **Delete Subtask**.

> **Note:**
> You cannot delete the Main Task.

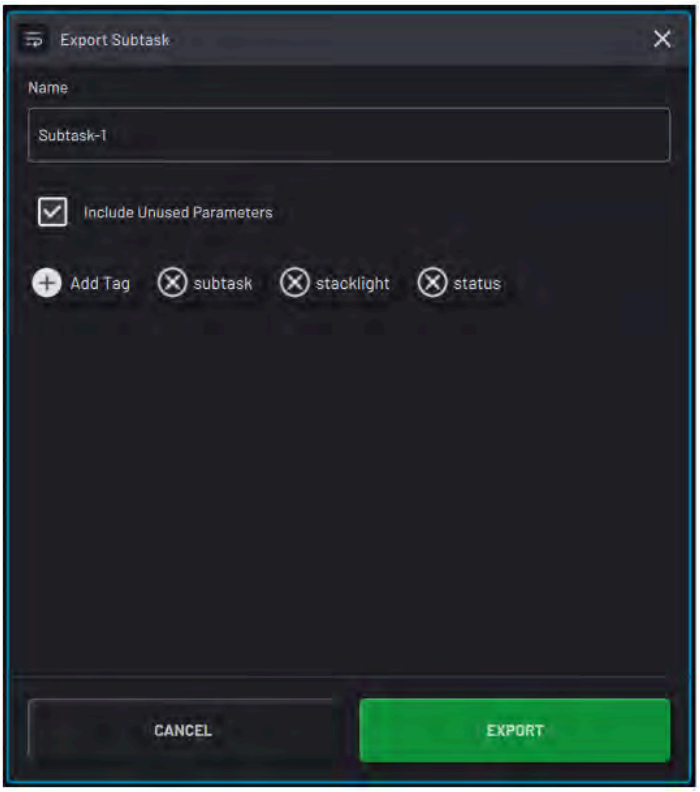


To change the order that the subtasks appear in the taskbar, tap **Reorder Subtasks**. In the pop-up, select one or more subtasks and tap the up/down arrows.

> **Note:**
> You cannot change the position of the Main Task.

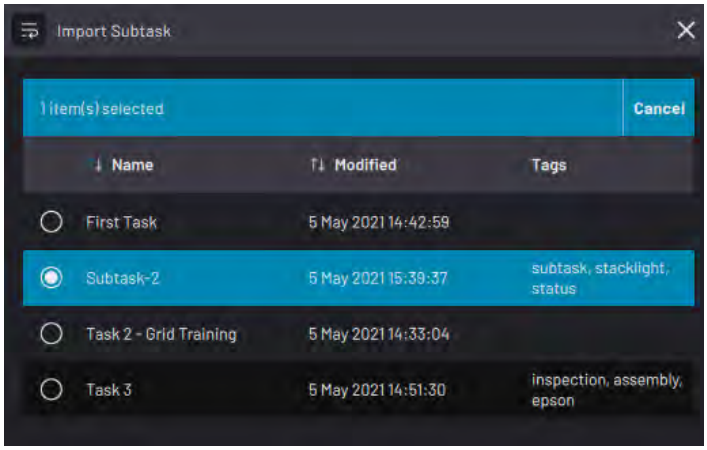


To export a subtask to the Load Task screen, select **Export Subtask** in the File Menu. Name the subtask and enter tags for easier finding later. Select **Include Unused Parameters** to export all waypoints, frames, and variables with the subtask.



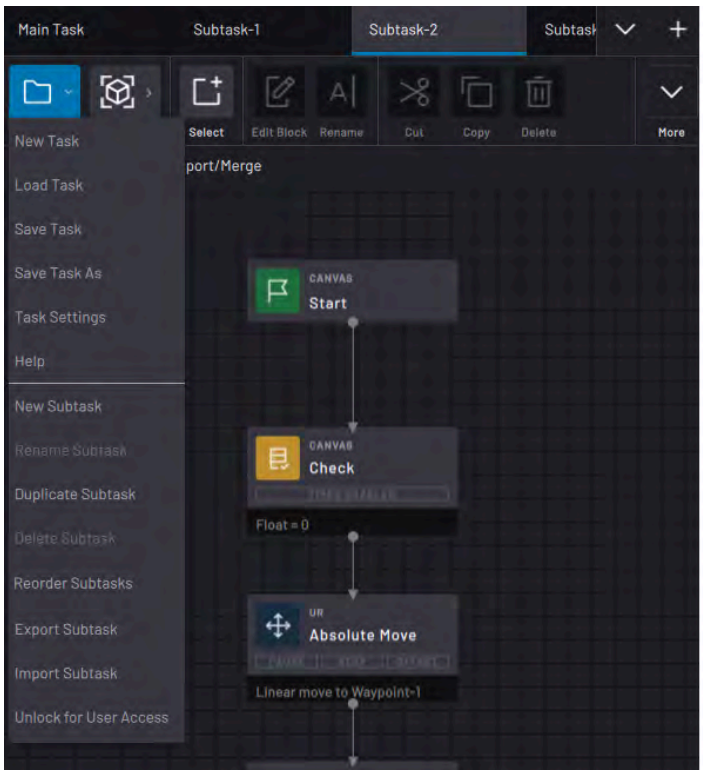
Note:
If you don't select the **Include Unused Parameters** box, the exported file only contains waypoints, frames, and variables that are in use on the canvas.

To import a subtask into another task, open the new task and select **Import Subtask** in the File Menu. Find the subtask by name, select it, and tap **IMPORT**.



Note:
Changes that you make to the imported subtask do not affect the original file that you exported from.

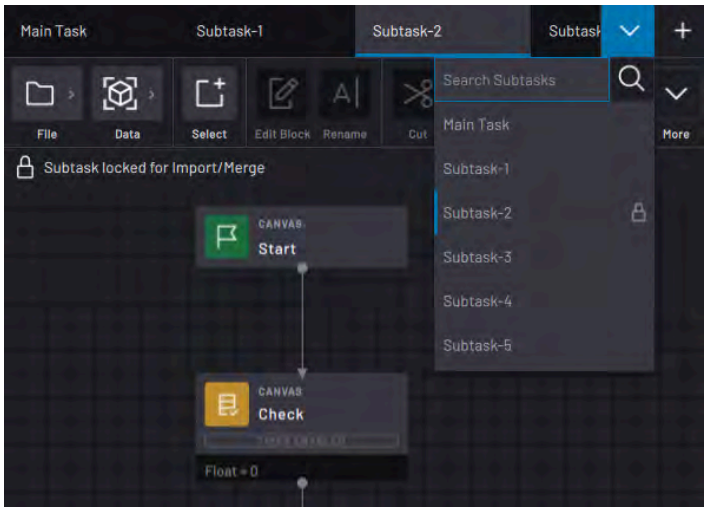
To prevent any subtask from being modified and to allow it to be overwritten by FTL imports/merges, tap **Lock for Import/ Merge**. A lock icon appears in the top-left corner of locked subtasks. Tap **Unlock for User Access** to undo this change.



> **Note:**
> Subtasks are unlocked by default. Unlocked subtasks cannot be overwritten by FTL. For more info about FTL, see Importing FTL Files *(on page 130)*.

To navigate between subtasks, tap the down arrow in the top-right corner. From the dropdown, tap a subtask's name to jump to that subtask's flowchart.

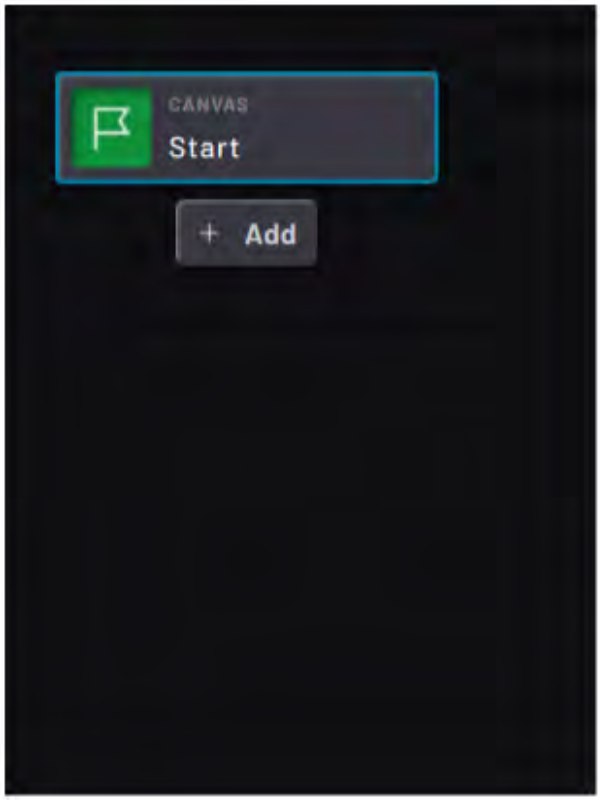
Locked subtasks have a lock icon in the dropdown.

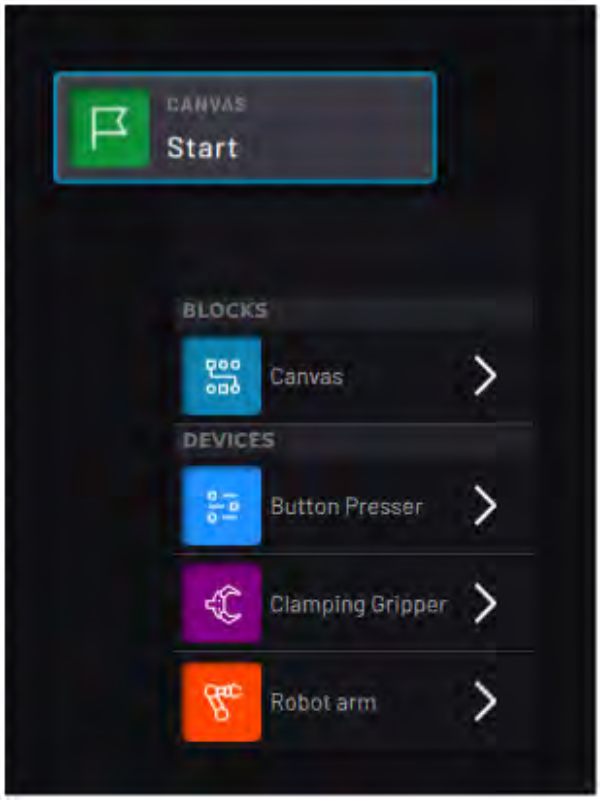


Block Programming

Add Blocks

To add a block, tap an existing block and then **+ Add**. This reveals the **Block Menu.** Task Canvas organizes the block menu by the devices that a block can control.





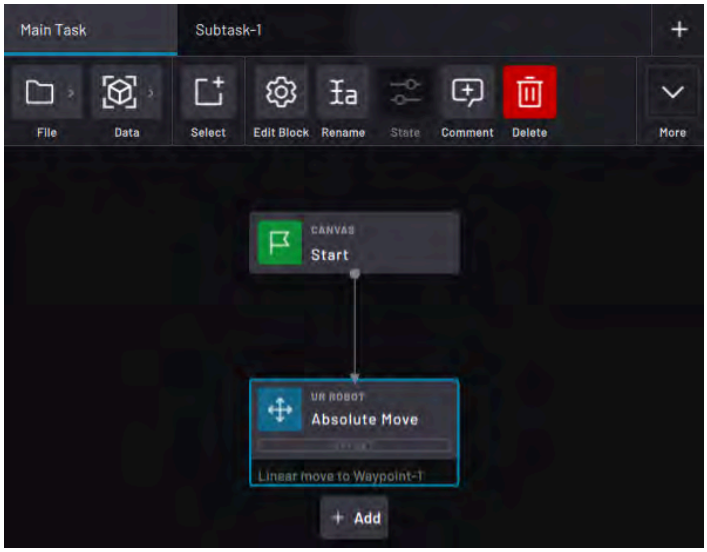
Tap any block on the canvas to display the available paths. When you create a new task, tap the Start block to add your first block.

Blocks in the **Canvas** category are not specific to any device. Some of them can control multiple devices and others control flowchart logic or variables.

Add blocks through the bottom or right paths of existing blocks. You can add blocks on the path between existing blocks. Task Canvas inserts the new block and connects all the blocks.

Edit Blocks

To edit a block, tap the block to select it and tap the **Edit Block** icon in the Block Editor Menu.

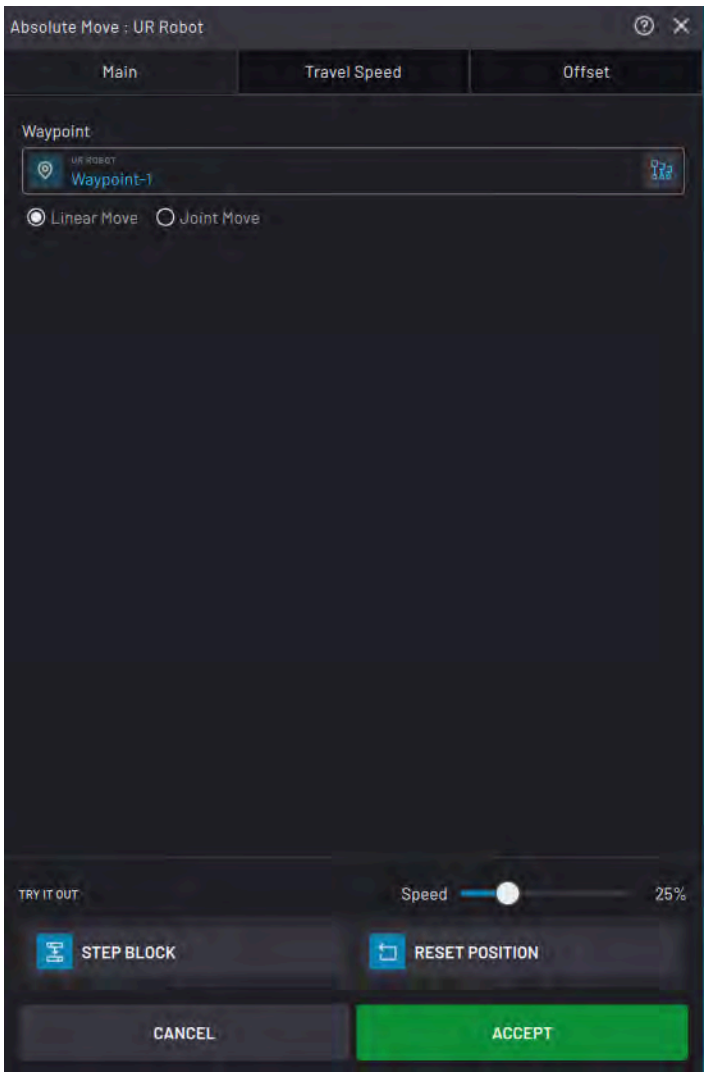


In the pop-up, modify the block's settings and parameters.

For example, in the Absolute Move block, change the waypoint, speed, and offset parameters.

In robot move blocks, use the **TRY IT OUT** section to preview the move. Tap **STEP BLOCK** to move the robot to the set position. Drag the **Speed** slider to determine how fast the robot moves. Tap **RESET POSITION** to move the robot to the position it was in when you opened the block.

Tap **CANCEL** to close the block without saving the changes or **ACCEPT** to close the block and save the changes.

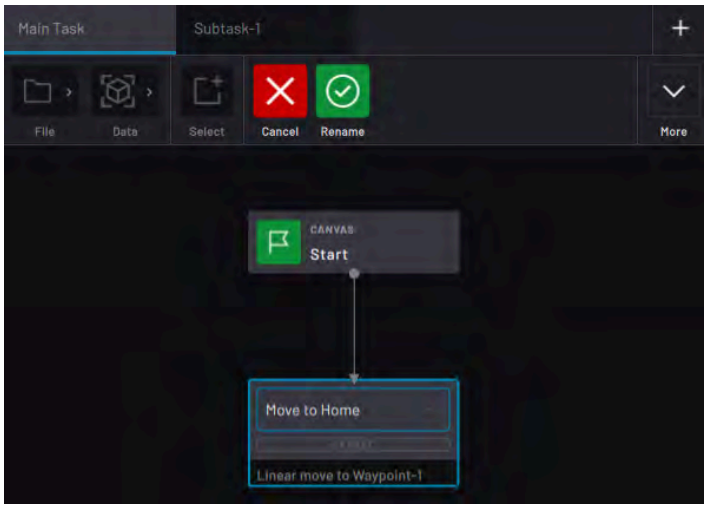


Rename Blocks

Naming blocks can be useful to identify the function that a block performs. For example, rename a Set block to "Turn on machine" or a Continuous Move block to "Motion exiting machine".

To rename a block, tap the block to select it and tap the **Rename** icon on the Block Editor Menu.

Type in the name and tap the green check mark to confirm.

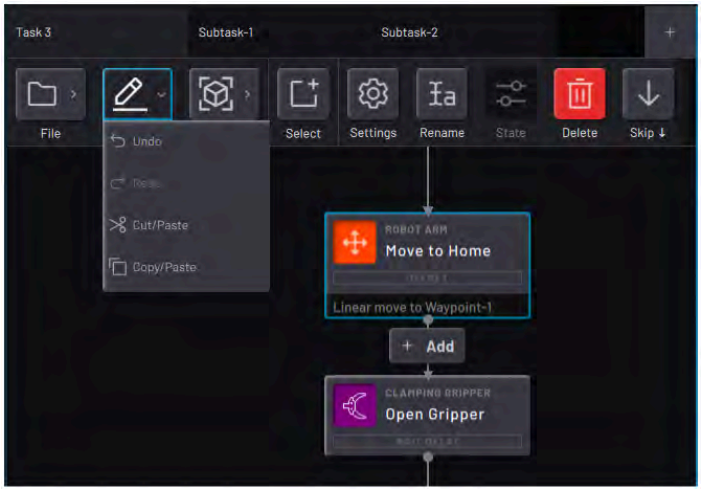


Cut/Copy and Paste Blocks

Move or duplicate a block using the **Cut/Paste** and **Copy/Paste** features in the expanded Block Editor Menu. Cut and Copy do not affect the parameters or custom text used by a block. You can cut or copy only one selected block at a time.

Select the block on the canvas you want to move or duplicate, then tap **Cut** or **Copy**. Select the target block after which you want to place the cut or copied block, then select the path on which to place the block.

Finish by tapping **Paste** on the top bar.

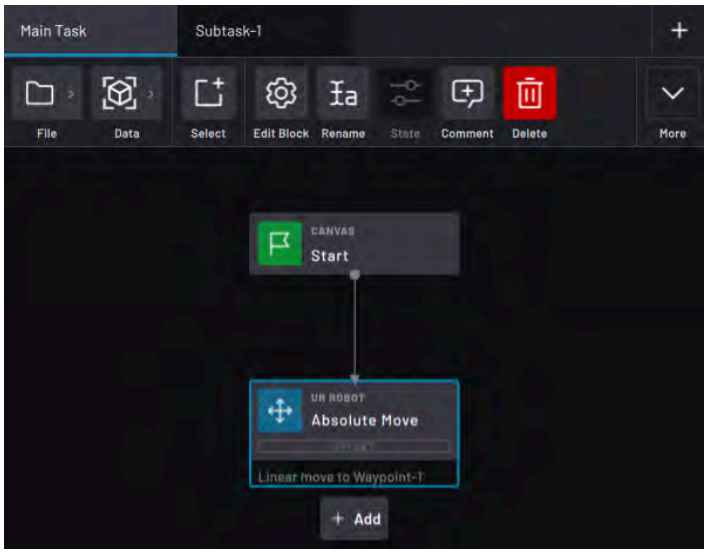


> **Note:**
> Cut removes the block from its previous location when pasting it to the new one. This may break existing paths at the previous location in a way that requires you to reconnect them.

Delete Blocks

To remove a block from the Canvas, tap the block to select it and tap the **Delete** icon on the Block Editor Menu.

Task Canvas reconnects paths that were going to and from that block. The app notifies you if you need to manually reconnect some paths.



> **Note:**
> Deleting a block from the canvas does not delete the parameters used in the block, such as waypoints or variables. To delete task parameters, use the correct data manager in the Data Menu.

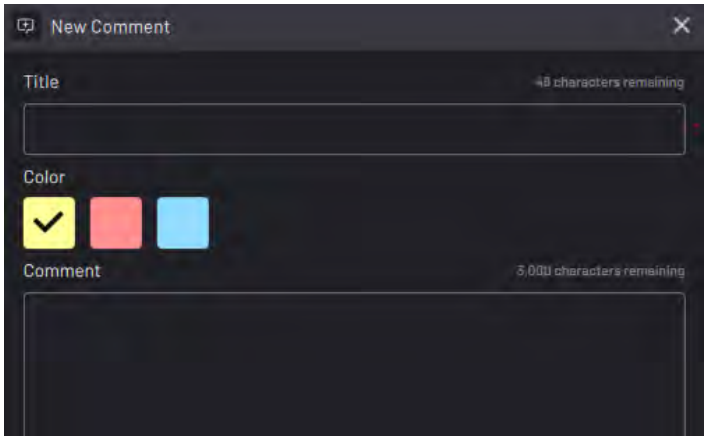
Add Comments to Blocks

Add comments to blocks to keep track of what each block does.

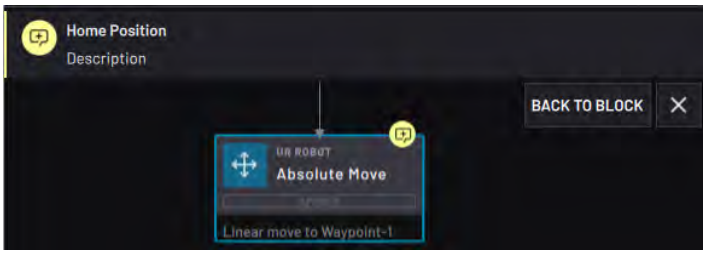
Tap a block to select, then tap **Comment**.

In the pop-up, type in a Title and choose a comment color (yellow, red, or blue). In the Comment box, type additional details. Tap **ACCEPT** to save.

To edit or delete the comment, select the block and tap Comment again. Make changes and **ACCEPT**, or tap **DELETE COMMENT**.



To view a comment, tap the colored icon at the top-right corner of the block. Tap **BACK TO BLOCK** to center the screen on the commented block. Tap **X** to close the comment.



Add Block Notifications

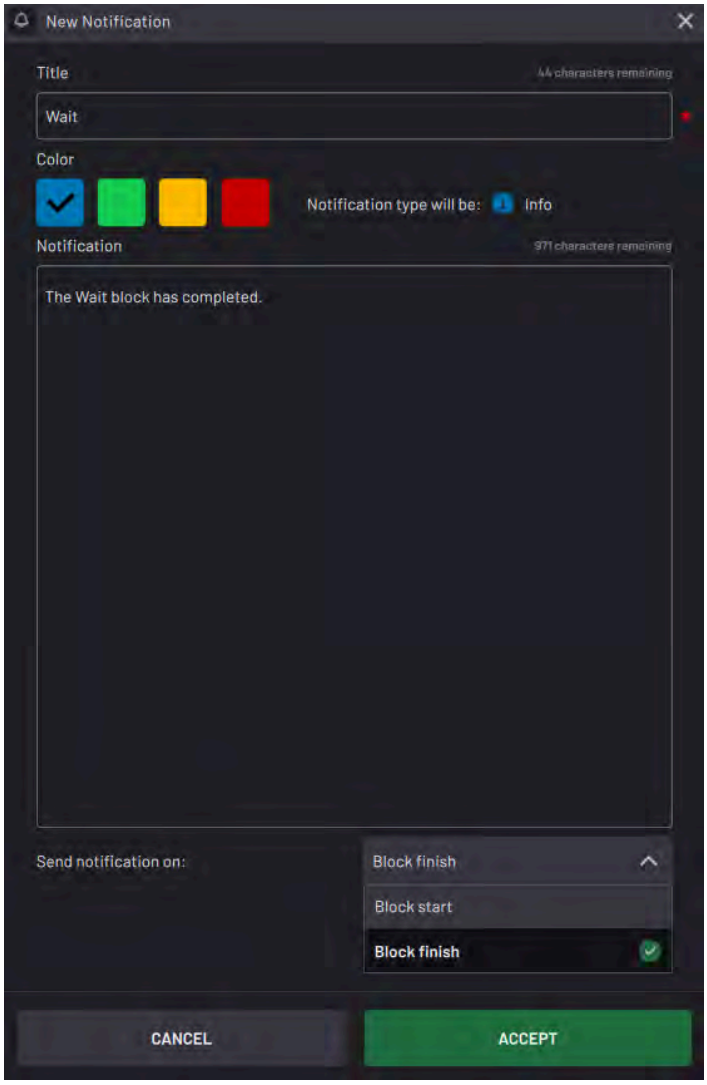
Blocks can publish a notification when it either starts or finishes.

Tap a block to select, then tap **Notify**.

In the Title field, type in a name. This is the name that will display in the Notifications Panel table. By default, this field fills in with the name of the block.

Choose one of four notification types. The corresponding color and symbol will appear as an icon on the block and in the “Type” column in the Notification Panel.

- Blue: Info
- Green: OK
- Yellow: Warning
- Red: Error

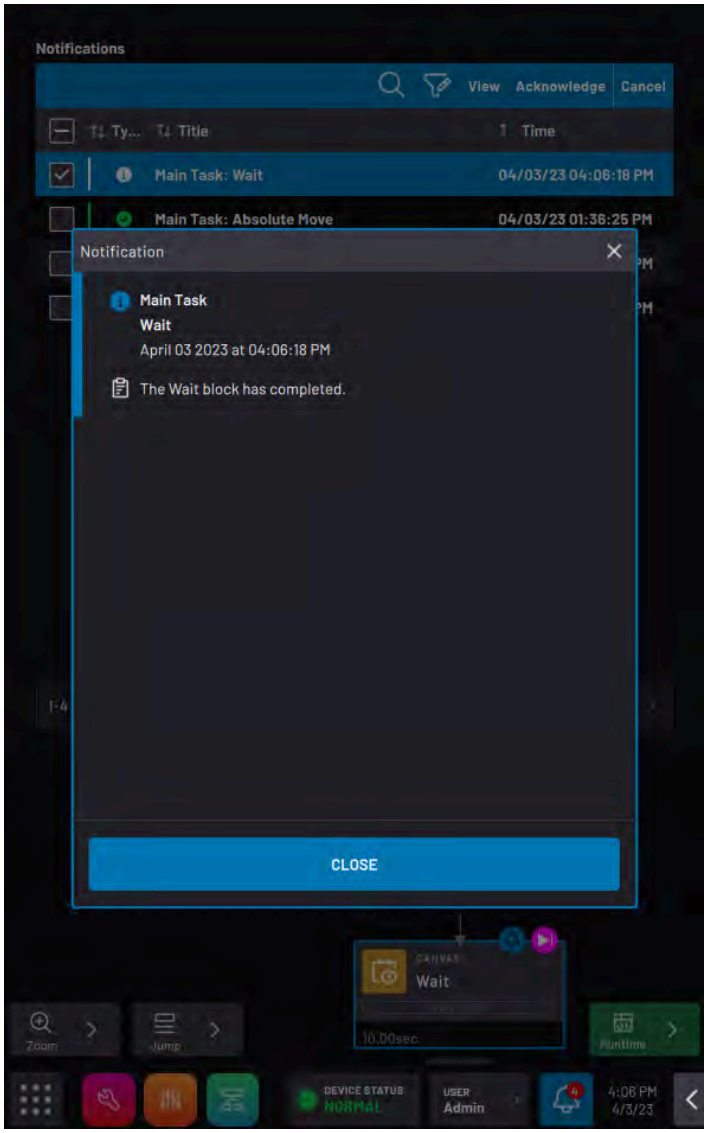


In the Notification box, type additional details. This info will display in a pop-up when viewing the notification.

Finally, choose whether the notification is sent when the block starts or finishes executing.

To edit or delete the notification settings, select the block and tap **Notify** again. Make the changes and **ACCEPT**, or tap **DELETE NOTIFICATION**.

> **Tip:**
> You can add this notification feature to multiple blocks at once in Select mode. When doing this, blocks will give default info notifications.



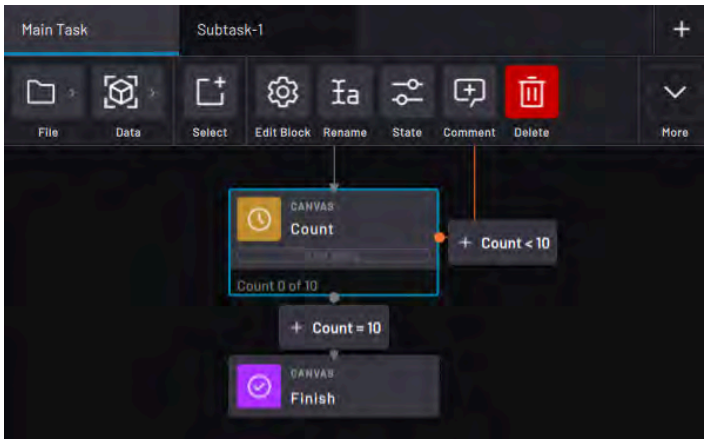
Change Block Execution State

Some blocks take on different states while the task is running. These states are called **Execution States**.

For example, the execution state of a Count block that has counted to 4 is "4".

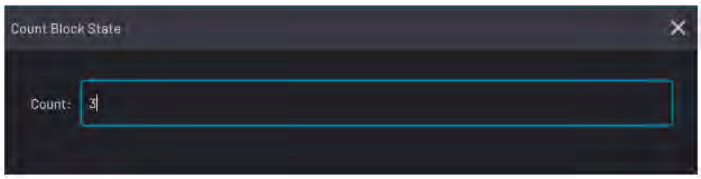
The execution state of a 30 second Wait block that is stopped at 20 seconds is "20 seconds".

When you **Resume** or **Step**, these blocks continue from where they left off. Press **Reset States** or **Reset All to Start** to quickly return them to their starting values.



Note:
A *Count block* with the execution state "Count 4" resumes by counting to "Count 5". A *Grid block* with an execution state "Move to 14" resumes by moving to the 14th position in the grid. A *Wait block* with an execution state of 20 seconds resumes by counting down from 20 seconds.

To set the execution state of a block, select the block and tap **State**. In the pop-up, type in the state and tap **UPDATE**.

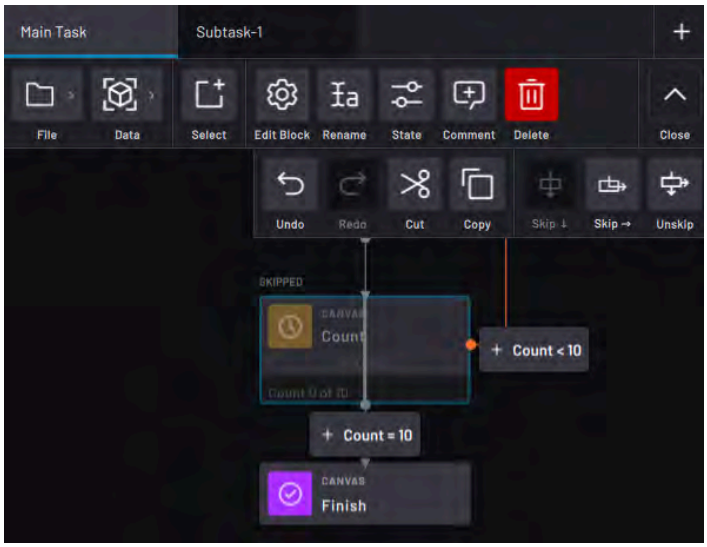


Skip Blocks

A "skipped" block means that it doesn't execute at runtime.

Select a block and choose **Skip Down** or **Skip Right** to skip the block and instruct Task Canvas which path to follow instead.

Choose **Unskip** on a skipped block to restore it to execution.



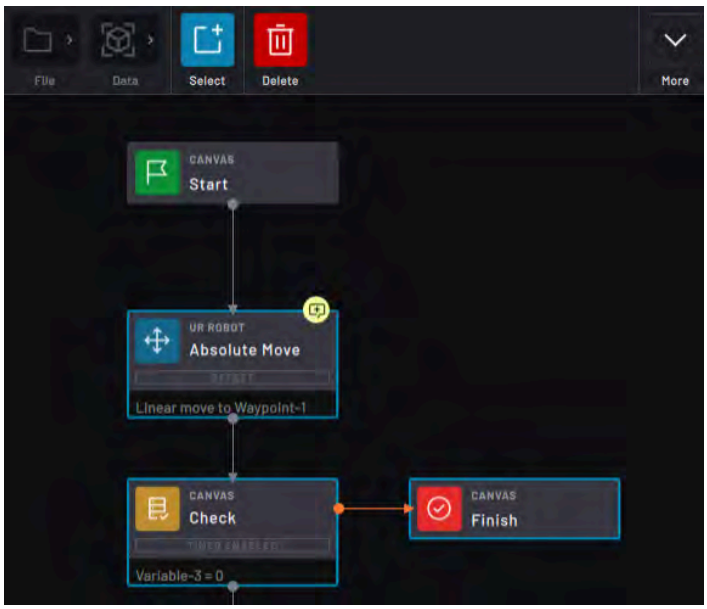
Important:
Be careful not to skip a function necessary for later actions to execute. For example, don't skip a block that opens a machine door before the robot moves into the machine.

Select Multiple Blocks

To select multiple blocks, toggle the multi-select mode. Tap the **Select** icon on the Block Editor Menu to toggle it on (blue). Tap Select again to toggle multi-select off (gray).

While the multi-select mode is on, select one or multiple blocks by tapping them. Task Canvas highlights selected blocks with a blue outline.

To deselect a block, tap it again.

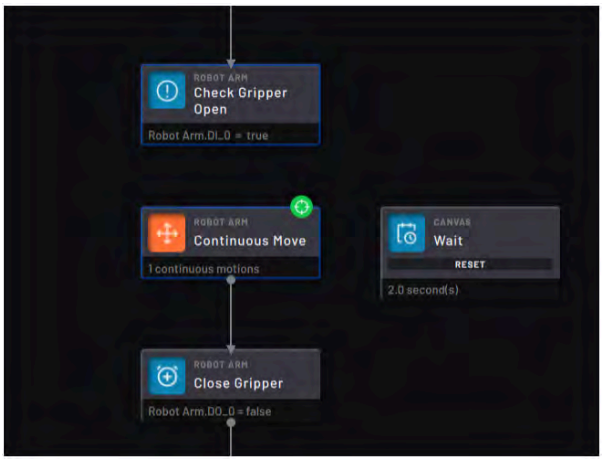


> **Note:**
> You cannot access block settings, rename blocks, or change block states in the multi-select mode. You can only Delete, Skip, or Unskip selected blocks in the multi-select mode.

Connect Blocks

To create a path between existing blocks, select an exit path of the first block. Then tap **Connect To** from the Block Menu.

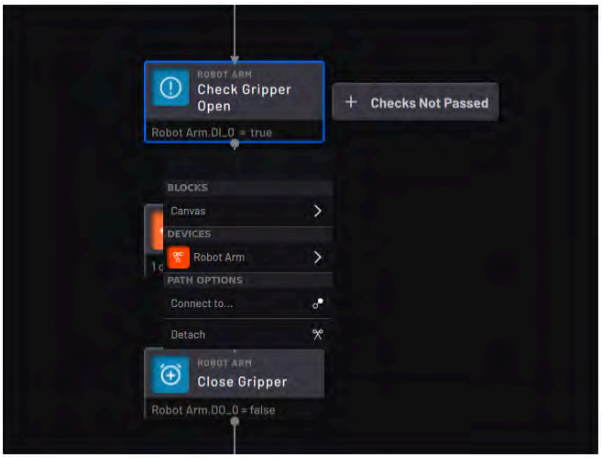
Select the second block where path goes, then tap **Connect** on the top menu.

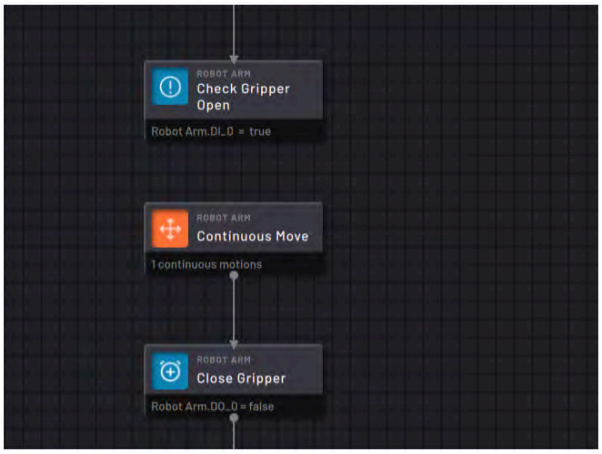


Detach Blocks

To remove a path between two existing blocks, select the exit path from the first block. Then select **Detach** from the Block Menu.

> **Note:**
> Detached blocks do not execute with the task. Be sure to reconnect the blocks where you want them before executing the task.





Undo/Redo

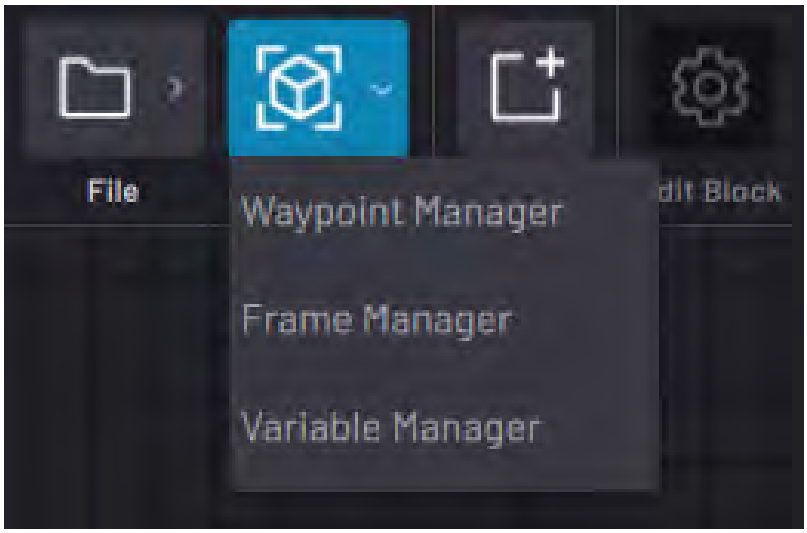
Use the Undo and Redo buttons in the expanded Block Editor Menu to undo or restore actions like creating a block, deleting a block, or modifying paths.

Tap **Undo** to undo an action in the Canvas. Tap **Redo** to re-do an action that was previously undone.

Note:
A maximum of 32 modifications are saved in the Undo stack.

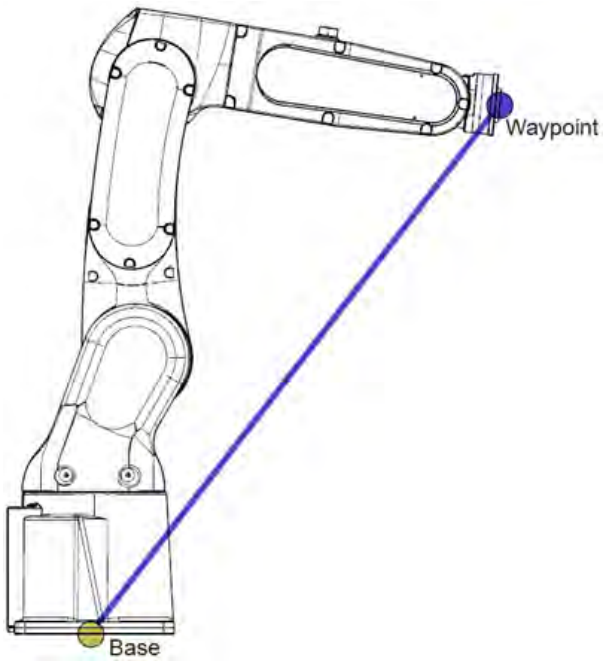
Data Management

The Canvas Data menu is where you modify data used by blocks in the task, including robot positioning data and variables.

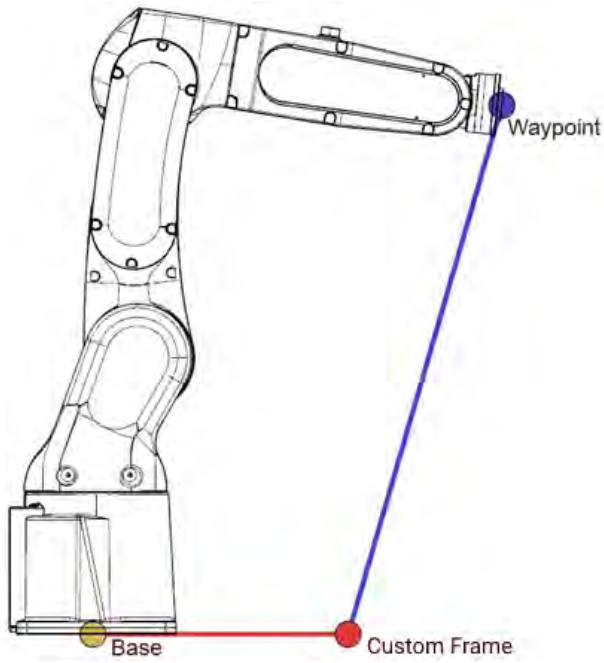


Create and Manage Waypoints

A **Waypoint** defines the **pose** (position and orientation) of the robot relative to the origin of a **Frame**. Define waypoints using the **Base Frame** (at the base of the robot) or a **custom Frame**.



Waypoint defined by the Base Frame



Waypoint defined by a Custom Frame

Tap **NEW** in any robot command block waypoint field to create a waypoint.

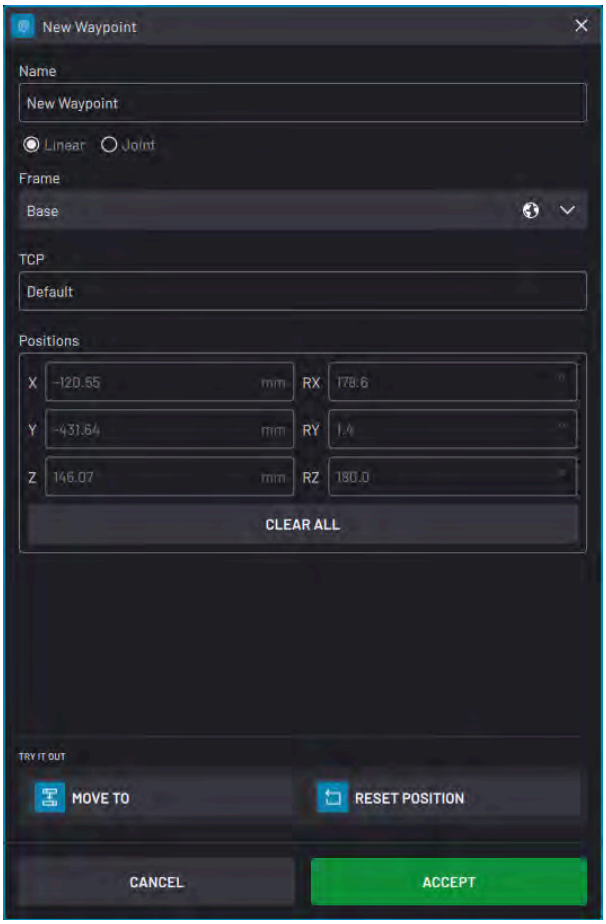
In the New Waypoint pop-up, choose how to define the waypoint (**Linear** or **Joint**).

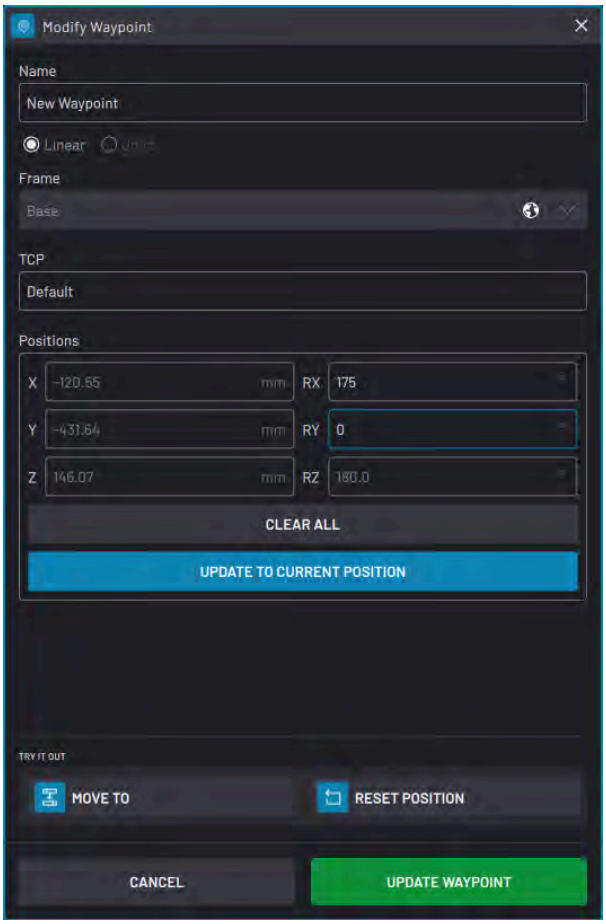
	Linear Waypoint	Joint Waypoint
What type of data does it save?	The robot saves the pose and orientation of the Active **TCP** with respect to the chosen **Frame**. If you choose a custom Frame and that Frame moves, the waypoint moves based on the new Frame origin. **Important:** If the Active TCP is different from the one used to create	The robot saves the positions of each of its joints. It does not reference any Frame or TCP.

	Linear Waypoint	Joint Waypoint
	a waypoint, the end effector may not move to the expected position.	
When should you choose this type of waypoint?	When you need the robot to travel the shortest, straight-line path to the waypoint. Depending on where the robot is moving from, several joint positions may be possible for the same TCP pose.	When a straight-line travel to the waypoint would cause the robot to collide with itself. Or when you need to go to the same joint positions every time (i.e., to avoid tangling cables).

When creating a new waypoint, the **Positions** fields fill in with the robot's current position. Move the robot to the position you want to save or enter your own values in the Positions fields. For each field you don't type a value, the field tracks the current position of the robot until you save the waypoint.

To save the waypoint, tap **ACCEPT**.





After you save a waypoint, tap the **EDIT** button to make changes to it. There are two ways that you can edit a waypoint:

1. In the **Positions** fields, type in new values. Then tap **UPDATE WAYPOINT** to save.

   > **Note:**
   > Do not tap **UPDATE TO CURRENT POSITION**. That button changes the Positions fields to the current robot position.

2. Move the robot in the Device Control app. Overwrite the old position values by tapping **UPDATE TO CURRENT POSITION**. Then tap **UPDATE WAYPOINT** to save.

In the New Waypoint or Modify Waypoint windows, tap **CLEAR ALL** to erase your input in all Positions fields. The fields then track the current position of the robot until you save the waypoint.

Waypoint Manager

Use the **Waypoint Manager** to create and manage Waypoints in Task Canvas. The Waypoint Manager lists saved waypoints for the selected robot device in the open task. A waypoint created with one robot can't be used for a different robot.

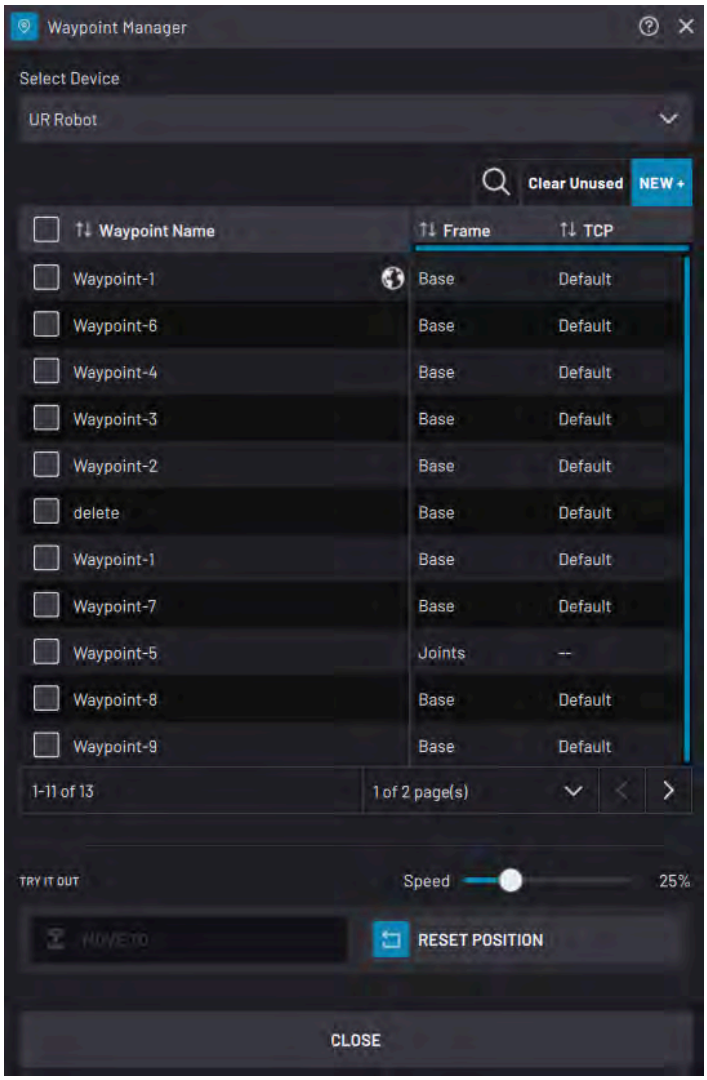
Listed for each waypoint is the **Waypoint Name**, the **Frame**, and the **Active TCP** at the time you saved the waypoint.

To find waypoints, use the search button, sort them by name, or navigate using the page navigation buttons.

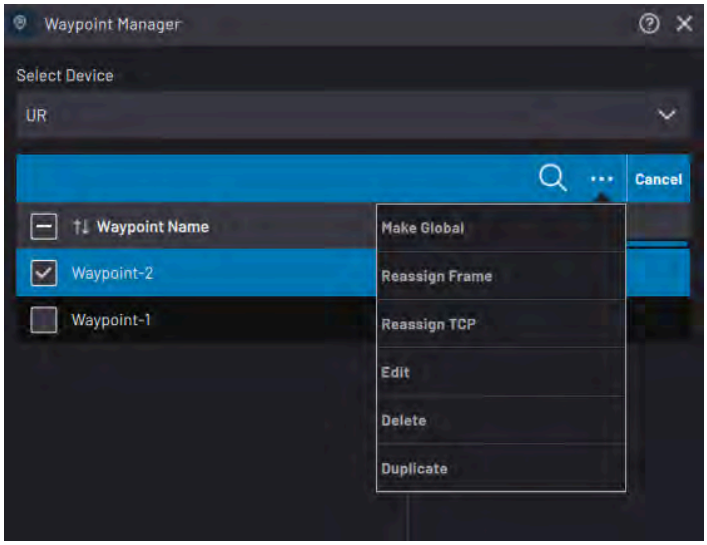
To remove all unused waypoints from the task, select none of the waypoints and tap **Clear Unused**. A prompt lists all the unused local variables and asks you to confirm before deleting them. **Unused** waypoints are not referenced by any blocks on the canvas.

To create a new waypoint, tap the **NEW +** button at the top of the table.

At the bottom of the manager, use the **TRY IT OUT** section to preview a selected waypoint. Tap **MOVE TO** to move the TCP to the waypoint. Drag the **Speed** slider to determine how fast the robot moves. Tap **RESET POSITION** to move the robot to the position it was in when you opened the Waypoint Manager.



To make waypoints global, select them on the table and tap **Make Global**. Global waypoints are accessible to other tasks and apps. If you change a global waypoint, it changes everywhere it is used. Global waypoints have a globe icon next to them in the table.

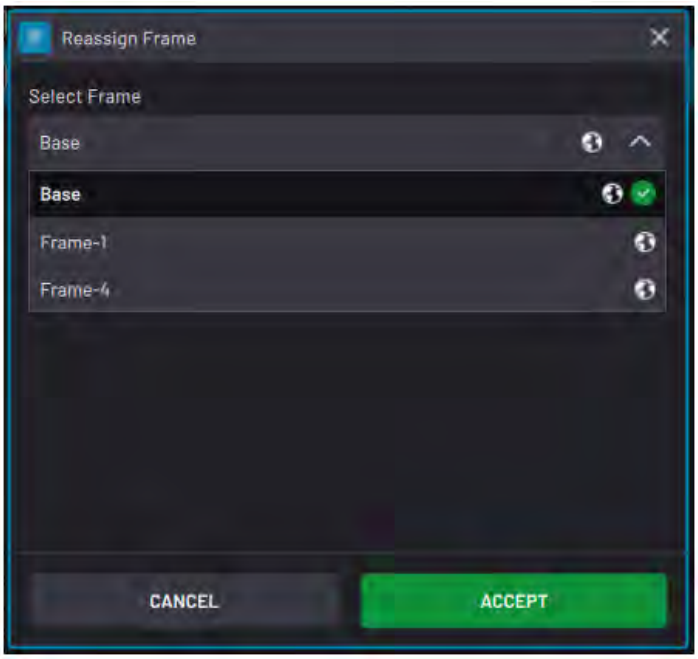


Important:
Once you make a waypoint global, you cannot make it local again. However, you can create a local duplicate of a global waypoint.

To reassign the Frame and TCP for selected waypoints, tap **Reassign Frame** or **Reassign TCP**. A dropdown appears to let you choose the new frame or TCP.

Re-assigning the TCP or Frame for a waypoint does NOT change the position of the waypoint. For example:

- If you change Waypoint-2's defining Frame to Frame-1 instead of Base, Waypoint-2's position won't change. Waypoint-2's position will change if you modify the coordinates of Frame-1.
- If some waypoints use a 100mm long tool, but you want to switch to a 300mm long tool, add a new TCP for that tool. Then use Set TCP to reassign the TCP for each waypoint.

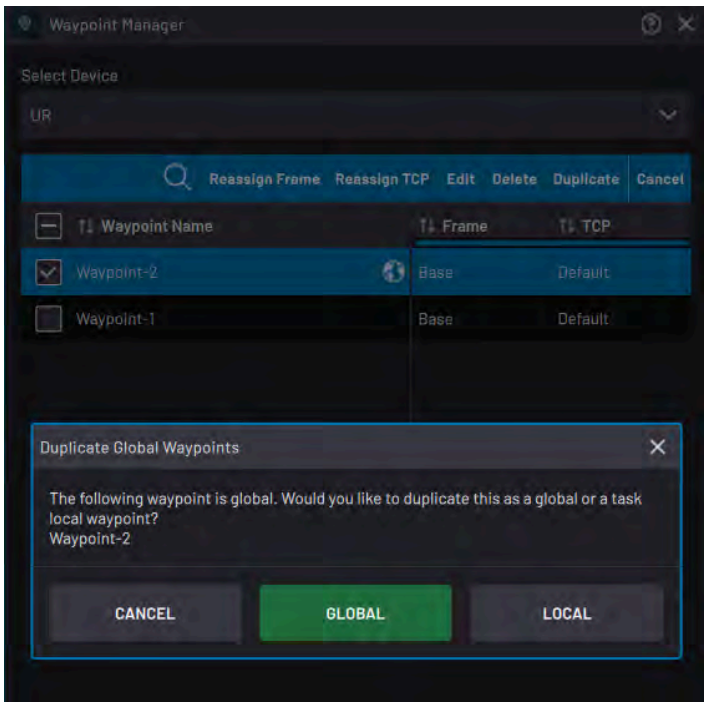


Important:
You may set the Active TCP in the Device Control app or with a Set TCP/Payload block in Task Canvas. If the Active TCP is different from the TCP used to create a waypoint, the end effector may not move to the expected position.

To edit a waypoint (change its name or position), select it on the table and tap **Edit**. You can only edit one waypoint at a time.

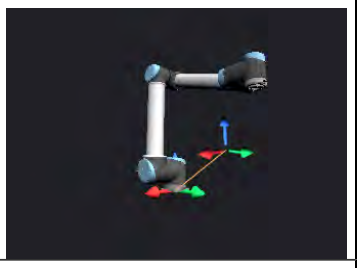
To remove one or more waypoints from the task, select them on the table and tap **Delete**.

To make a copy of one or more waypoints, select them on the table and tap **Duplicate**. If you are duplicating a global waypoint, choose if you want the copy to be local or global. The copy's name will end with "-copy" to avoid name collision.



Create and Manage Frames

Frames are Cartesian reference frames for moving the robot and defining **Waypoints**. Every robot has a default Base frame and TCP frame. You can also create custom 1-point or 3-point frames.

	Base Frame	TCP Frame	Custom Frame
Image			
Location	The base of the robot	The active Tool Center Point (TCP)	Wherever you define, relative to the Base frame or another custom frame
Can you use it to define Waypoints?	Yes	No	Yes
Can you use it to define Frames?	Yes	No	Yes
Can you view it in the Device Control app?	Yes	Yes	If you make the frame "Global" in the Frame Manager

Why use custom frames? Custom frames allow you to define a new coordinate system along an inclined tray, workbench, grid, or more! Here are the differences between 1-point frames and 3-point frames:

	1-Point Frames	3-Point Frames
Description	Define a 3D coordinate system based on the position and orientation of the TCP.	Define a 3D coordinate system based on three positions of the TCP.
How do you create one?	Define with values or position the TCP at the "Origin" you want with each of the TCP axes pointing along the desired X, Y, and Z axes.	Position the TCP at the "Origin", the "X point", then the "Y point" (OR) Choose existing waypoints to define the "Origin", the "X point", then the "Y point".
What are the advantages?	Because 1-point frames only reference one position, they are slightly faster to create.	Because 3-point frames reference three positions, they are much more accurate.

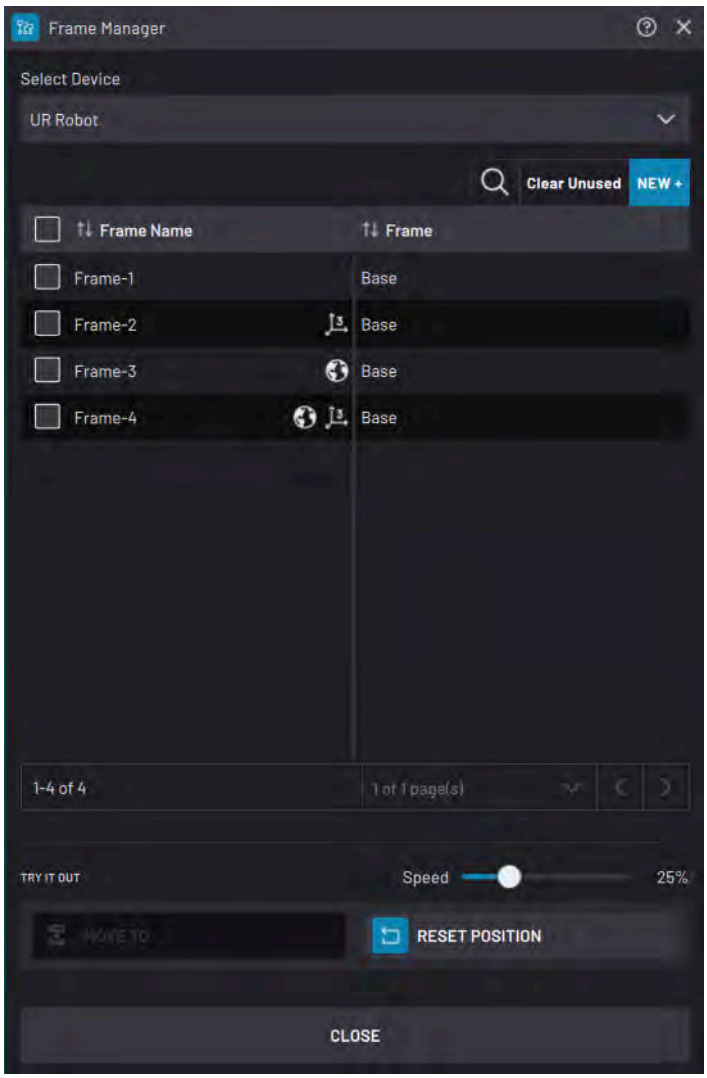
Frame Manager

Use the **Frame Manager** to create and manage custom frames for the selected robot.

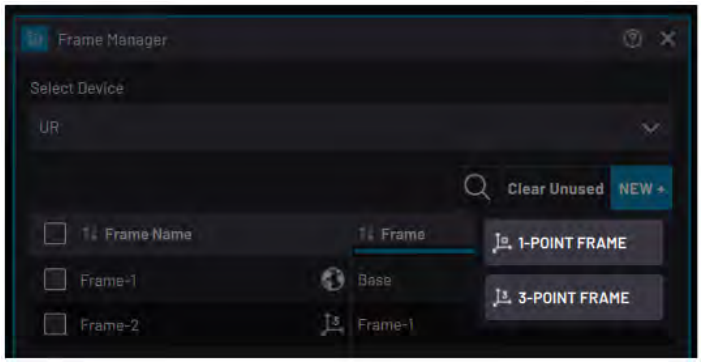
Note:
The Base and TCP frames are not in the Frame Manager. You cannot modify or remove them.

Some frames have icons to the right of their name to show what type of frame it is:

- **1-Point Frame**: No icon
- **3-Point Frame**: Coordinate system icon with the number "3"
- **Global 1-Point Frame**: Globe icon
- **Global 3-Point Frame**: Globe icon and the coordinate system icon with the number "3"



To create a new frame, tap **NEW +** at the top of the table. Decide if you want to create a **1-POINT FRAME** or a **3-POINT FRAME**.

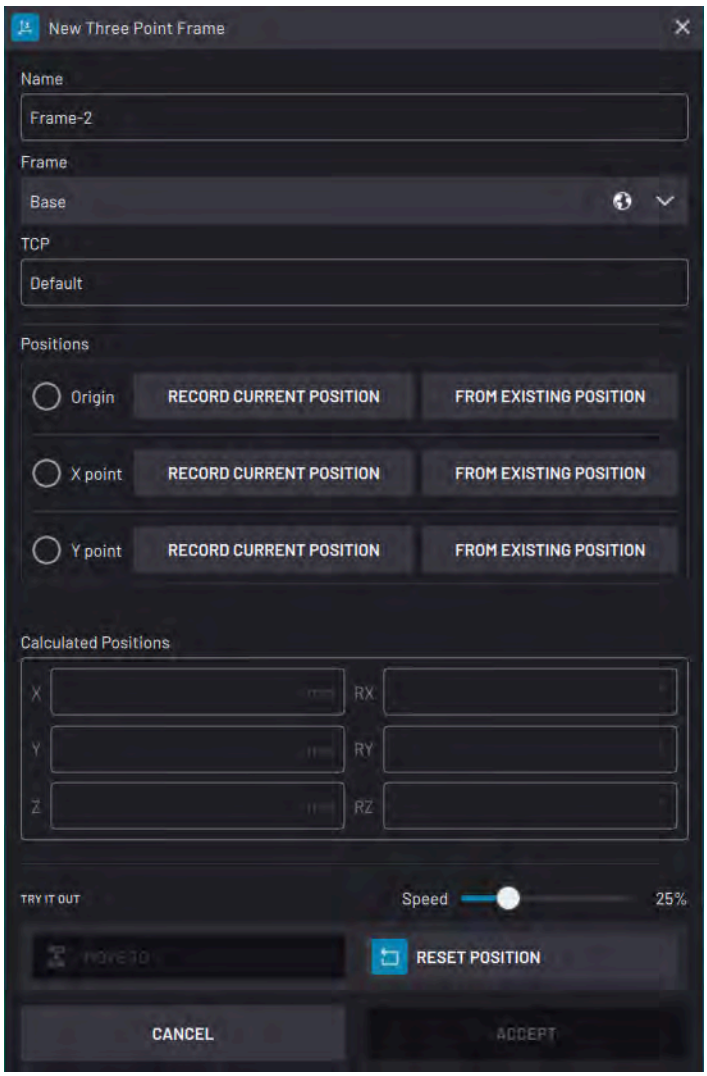


To create a 1-point frame, follow these steps:

1. Enter a **Name**.
2. Choose a **Frame** from which to define the new frame. Except for special cases, select the **Base** frame.
3. If the active TCP is not the TCP that you want to use, switch to a different one in the Device Control app.
4. In the Device Control app, move the TCP to the desired location of the new frame. In the New Frame pop-up, the **Positions** fields automatically fill in with the TCP's current position. Alternatively, directly type values into the **Positions** fields.
5. Tap **ACCEPT** to save the frame.

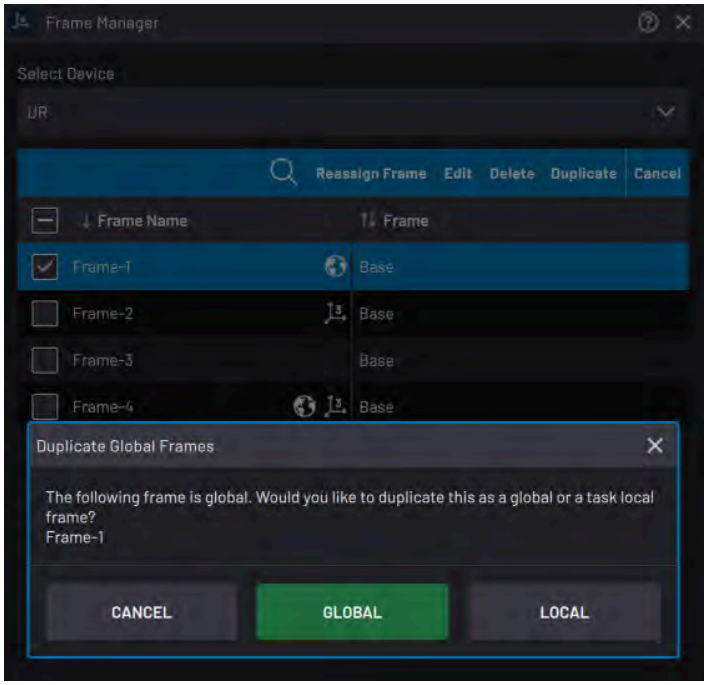
To create a 3-point frame, follow these steps:

1. Enter a **Name**.
2. Choose a **Frame** from which to define the new frame. Except for special cases, select the **Base** frame.
3. If the active TCP is not the TCP that you want to use, switch to a different one in the Device Control app.
4. In the Device Control app, move the TCP to the desired **Origin** of the new frame and tap **RECORD CURRENT POSITION**. Alternatively, tap **FROM EXISTING POSITION** to use an existing waypoint for the Origin position.
5. Repeat the above step for the **X point** and **Y point**.
6. Once there is a checkmark next to the Origin, X point, and Y point fields, tap **ACCEPT** to save the frame.



To make one or more frames global, select them and tap **Make Global** at the top of the table. A global frame is accessible to other tasks and apps (including the Device Control app). If you change a global frame, it changes everywhere.

> **Important:**
> Once you make a frame global, you cannot make it local again. However, you can create a local duplicate of a global frame.



To reassign the defining frame for selected frames, tap **Reassign Frame**.

> **Note:**
> Re-assigning the frame changes the defining frame WITHOUT changing the frame's position. For example, if you change Frame-2's defining frame to be Frame-1 instead of Base, Frame-2's position will not change. Frame-2's position will change if you modify the coordinates of Frame-1.

To edit a frame (change its name or orientation), select it on the table and tap **Edit**. You can only edit one frame at a time.

To remove one or more frames from the task, select them on the table and tap **Delete**. A prompt asks you to confirm before deleting them.

To make a copy of one or more frames, select them on the table and tap **Duplicate**. If you are duplicating a global frame, choose if you want the copy to be local or global. The copy's name will end with "-copy" to avoid name collision.

To delete all unused frames, select none of the frames and tap **Clear Unused** at the top of the table. A prompt lists all the unused local variables and asks you to confirm before deleting them. Unused frames are frames with zero waypoint references.

At the bottom of the manager, use the **TRY IT OUT** section to preview a selected frame. Tap **MOVE TO** to move the TCP to the selected frame. Drag the **Speed** slider to determine how fast the robot moves. Tap **RESET POSITION** to move the robot to the position it was in when you opened the Frame Manager.

Create and Manage Variables

Variables store and transmit values between blocks and devices. Use variables for decision-making with **Check** blocks. Use **Set** and **User Input** blocks to change variable values. Four types of variables are possible:

Type	Description
Boolean	"True" or "False"
Integer	Positive and negative whole numbers, including zero (i.e. 7) Min/max values are: -2147483648 and +2147483647
Float	Positive and negative numbers up to 5 decimal places (i.e. -108.64) Min/max values are: -3.4028E+38 and +3.4028E+38
String	Alphanumeric text value (i.e. 'Hello world') Maximum character length is 255 characters

Enter a variable into compatible block parameters so the block uses the variable when it executes. For example, use a variable in the distance field in a robot Relative Move block or the seconds field in a Wait block. If the variable changes somewhere in the task, these blocks may behave differently each time they execute.

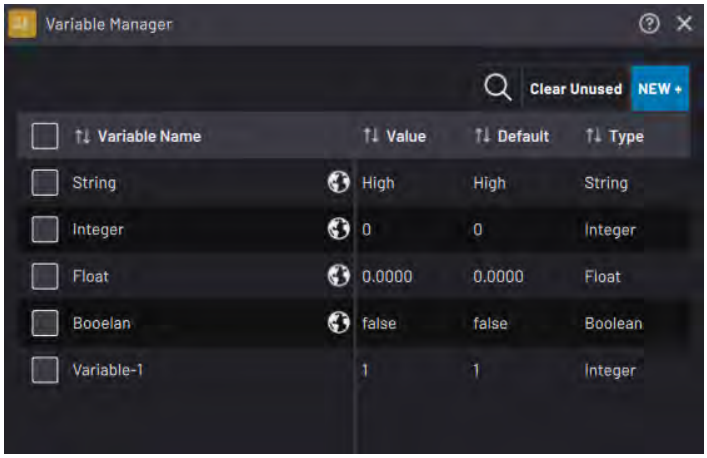
Note:
Variables are unitless in Forge/OS. The parameter field determines the unit.

Variable Manager

Use the **Variable Manager** to create and manage variables. Listed for each existing variable are its **Name**, current **Value**, **Default Value**, and **Type**.

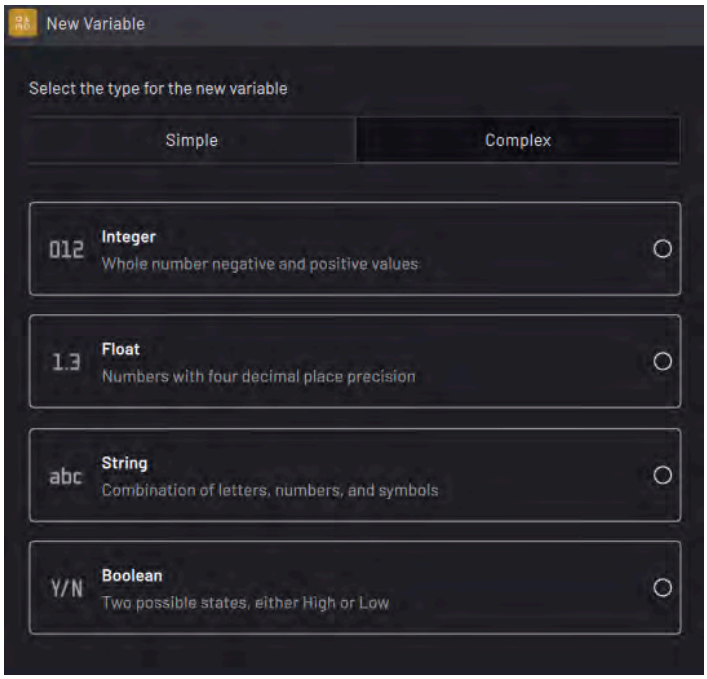
Note:
The **Value** is what you or the task last set the variables to. The **Default** value is what task variables reset to when you tap **Reset States** or **Reset All to Start**.

To delete all unused variables, select none of the variables and tap **Clear Unused** at the top of the table. A prompt lists all the unused local variables and asks you to confirm before deleting them. Unused variables are variables with zero block references.

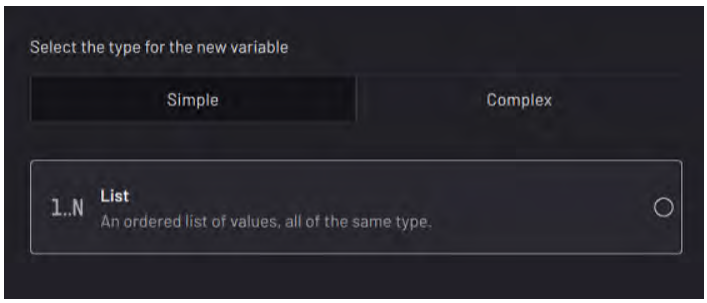


To add a new variable, tap **NEW+** at the top of the table. Select which type of variable you want to create, then enter that variable's information.

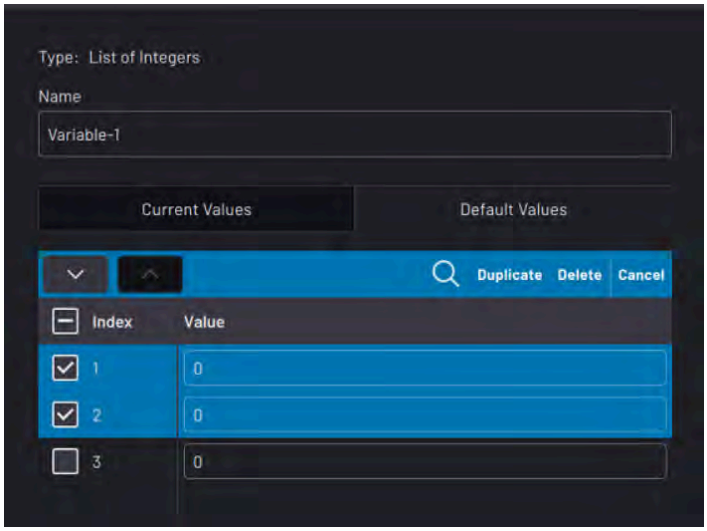
Important:
Once you create a variable, you can't change the type of variable it is.



In the **Complex** tab, add a list variable for advanced applications. A **list variable** is a series of one or more values of a chosen sub-type (i.e., integer, float, string, or boolean).

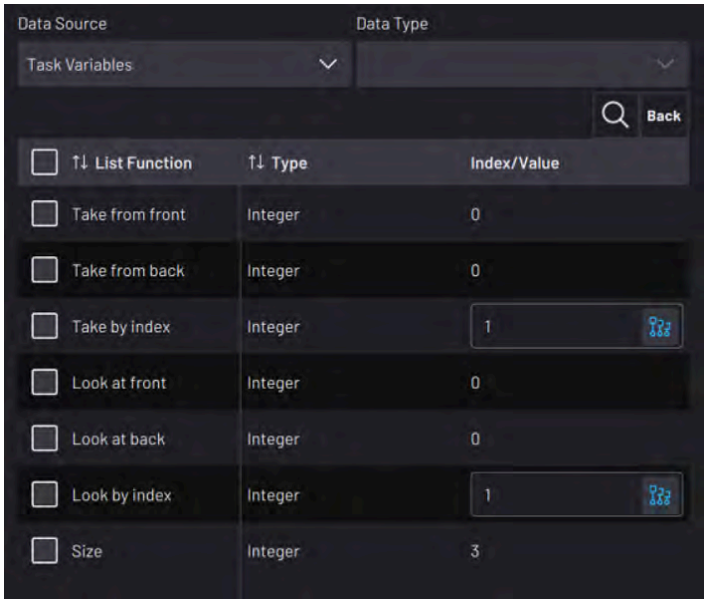


Each sub-type item has a current Value and a Default value. The first item in the list has an index of “1”, the second item has an index of “2”, etc. Change the order by selecting one or more items and pressing the up or down arrows. In the **Default Values** tab, copy the list from the **Current Values** tab by selecting no entries and then tapping **Copy From Current** in the table header.



Using the variable selector in various pop-ups, you may call a list variable in its entirety or any index from it. Depending on the context of the variable selector, you will either be presented with "Use" functions or "Set" functions.

- **Use functions** are available in places like the Check block. These functions include "take" and "look".
- **Set functions** are available in places like the Set block. These functions include "add", "insert", "overwrite", "merge", and "truncate".

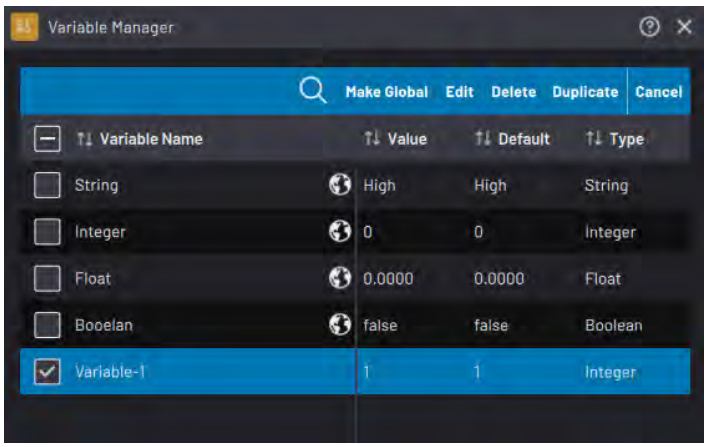


"Use" Function	Description
Take from front	The current value of the first item in the list is evaluated. Immediately afterwards, that item is removed from the list. The items below it move up to fill the space, and the list size reduces by 1.

"Use" Function	Description
Take from back	The current value of the last item is evaluated. Immediately afterwards, that item is removed from the list. Other items in the list do not move position, and the list size reduces by 1.
Take by index	The current value of the identified item in the list is evaluated. Immediately afterwards, that item is removed from the list. The items below it move up to fill the space, and the list size reduces by 1.
Look at front	The current value of the first item in the list is evaluated. No items are removed, and the list size does not change.
Look at back	The current value of the last item in the list is evaluated. No items are removed, and the list size does not change.
Look by index	The current value of the identified item in the list is evaluated. No items are removed, and the list size does not change.
Size	This integer value represents the number of items in the list.

"Set" Function	Description
Add to front	The item being set gets added to the first postion. The other items move down.
Add to back	The item being set gets added to the "N+1" position. The other items do not move.
Insert by index	The item being set gets added to the selected index position. The other items below it move down.
Overwrite by index	The item being set gets added over the selected index position. The other items do not move.
Merge from list	The items in one list get added to the end of another list.
Truncate	This integer value represents the desired number of items in the list to cut down to. If the entered value is smaller than the list's "Size", items at the bottom of the list are removed. If the entered value is equal to or greater than the list's "Size", no action is taken.

To access a variable in other tasks and in the Parameter Manager *(on page 104)*, select it on the table and tap **Make Global**. A pop-up asks you to confirm your decision.

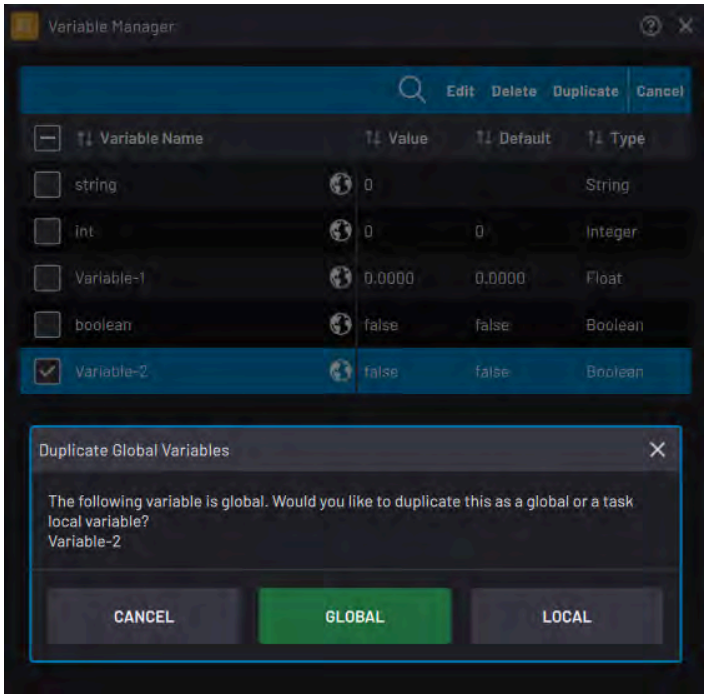


> **Important:**
> Once you make a variable global, you cannot make it only local again. However, you can create a local duplicate of a global variable.

To edit a variable, select it on the table and tap **Edit**. You can rename the selected variable, change its current value, or change its default value. You cannot change the variable's type.

To remove one or more variables, select them and tap **Delete**. A prompt asks you to confirm before deleting them.

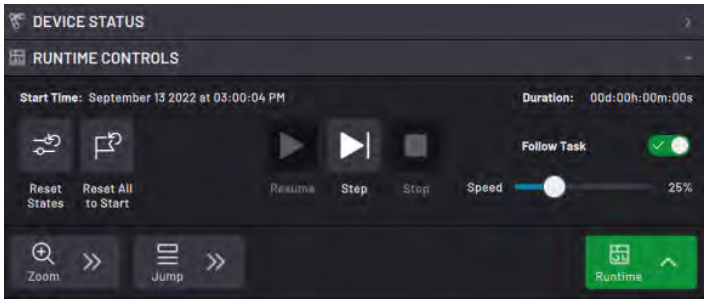
To make a copy of one or more variables, select them on the table and tap **Duplicate**. If you are duplicating a global variable, choose if you want the copy to be local or global. The copy's name will end with "-copy" to avoid name collision.



Running a Task

The Runtime Controls menu is where you execute the task.

Tap the **Runtime** button in the bottom-right corner to open or close the menu. See Runtime Controls *(on page 123)* for more info on each button.



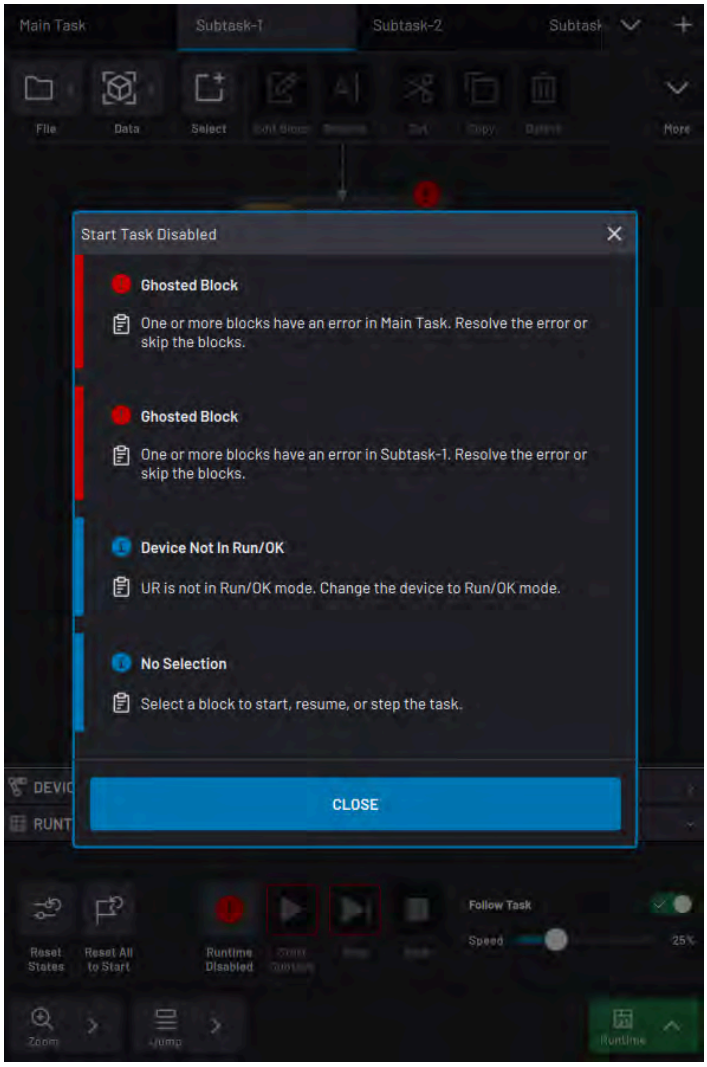
> **Tip:**
> You can only access some runtime controls under certain conditions:

- **Start Task/ Start Subtask/ Resume** is available when a block is highlighted and all devices in Task Settings *(on page 133)* are enabled and in *OK* or *RUN* modes. All Template blocks must be skipped.
- **Step** is available when the highlighted block is not a Template block and the referenced device (if any) is in *OK*, *RUN*, *PROGRAM* or *TEACH* mode.

For more info on Template blocks, see "Special Block States" in Features of a Task *(on page 134)*.

If the **Start Task/ Start Subtask/ Resume** button is unavailable, ForgeOS gives pointers on what to fix.

- Blue info icons indicate minor recovery actions (such as selecting a block to start from or switching the robot operational mode).
- Red exclamation points indicate actions that are more major (such as filling in a missing parameter in a template block).

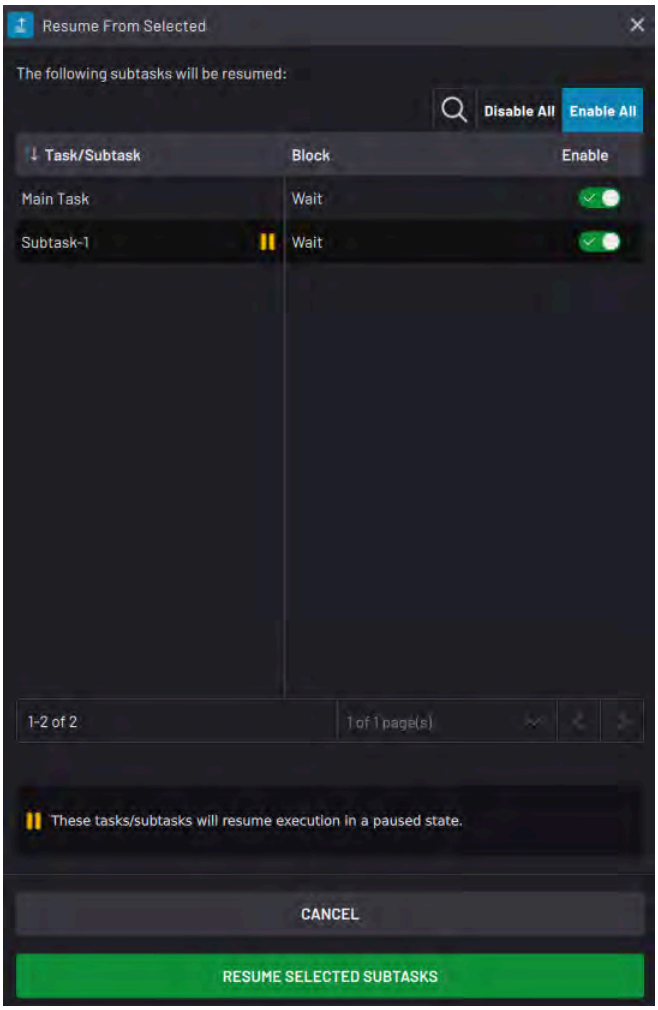


When you have the Main Task tab open (and either have not executed blocks yet or have tapped **Reset All to Start**), the **Start Task** button executes the task from the selected block with the current block states.

When you have a subtask tab open, the Start Task button changes to **Start Subtask**, which executes only that subtask.

If you've executed blocks and have not tapped **Reset All to Start**, the Start Task/ Subtask button reads **Resume**. If you are working with subtasks, a pop-up may ask which subtask(s) to resume. In the **Block** column, see where in the flowchart the subtasks will resume from. Toggle the **Enable** switch to specify which subtasks to run (toggle to the right) and which subtasks not to run (toggle to the left). A subtask that was paused from a Pause Subtask block has a yellow paused icon, and if you enable that subtask, it will resume in a paused state until it gets called by a Resume Subtask

block. In the table header, tap **Disable All** to quickly disable all subtasks or **Enable All** to quickly enable all subtasks. You must have at least one subtask enabled to tap **RESUME SELECTED SUBTASKS**.



When you start or resume a task, the task executes until one of these events:

- The task reaches a Finish block.
- The task reaches the last block in the flowchart.
- You stop the task with the Stop button.
- One of the devices enters an error state, including emergency stop.

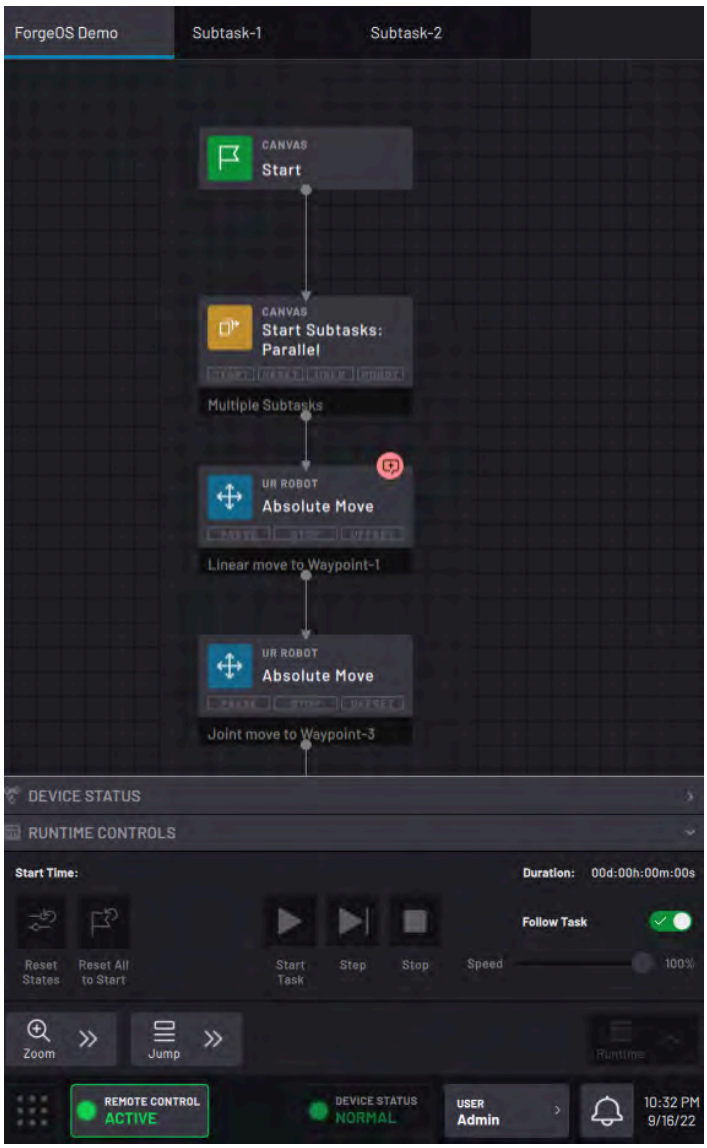
Using Remote Control Mode

Remote Control mode allows you to communicate with ForgeOS from an external HMI.

After you configured and enabled this HMI as a **Remote Control Device** in Device Configuration, you can use it in Task Canvas.

Enter and exit Remote Control mode in the Taskbar's **USER** button flyout. When you are in Remote Control mode, the **Remote Control Status** button replaces the Taskbar's app icons.

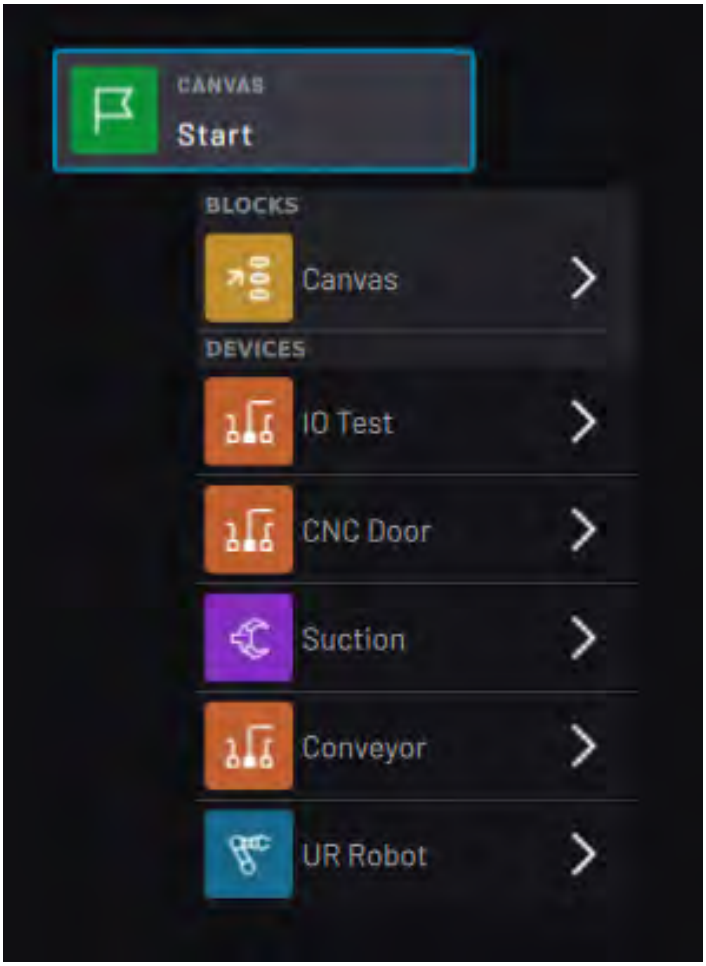
While in Remote Control mode, there is limited on-screen interaction with the READY pendant. The Remote Control device sends and receives data, based on what signals you configured for its inputs and outputs.



Block Glossary

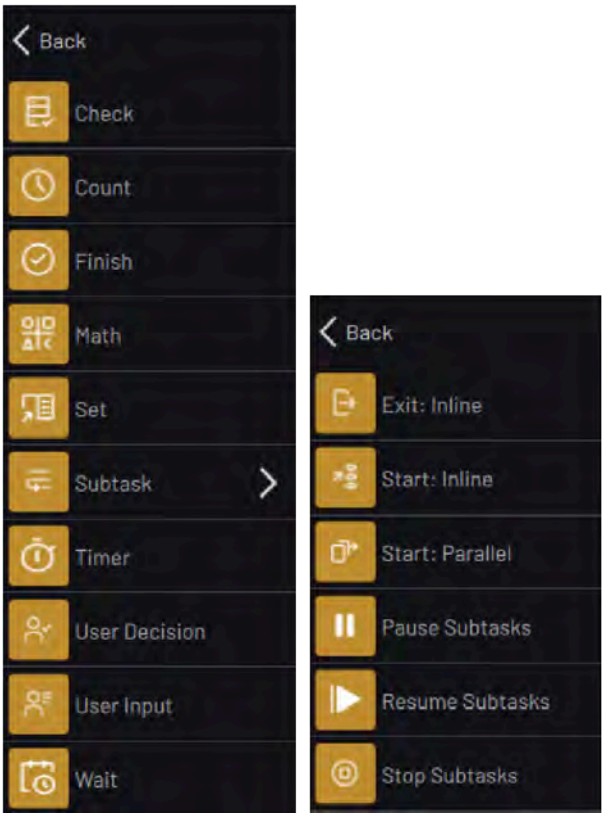
The types of **blocks** available in Task Canvas depend on the devices you added in Task Settings.

Add blocks to a task from the **Block Menu**. The Block Menu organizes block types by device, except for the Canvas blocks. **Canvas** blocks are not specific to any device. Some of them can control multiple devices and others control flowchart logic or variables.



Canvas Blocks

The Canvas blocks include Check, Count, Finish, Math, Set, Timer, User Decision, User Input, Wait, and Subtask blocks. The Subtask blocks allow for multitasking with subtasks. They include, Exit, Start, Pause, Resume, and Stop.

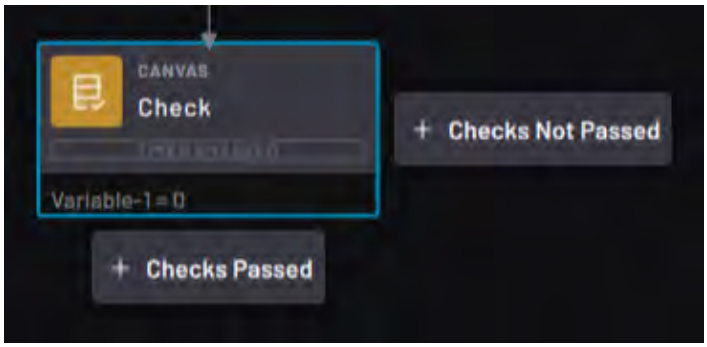


Check Block

The Check block compares chosen parameters or inputs to expected values.

If the inputs meet the expected conditions, the task moves on through the bottom path (**Checks Passed**).

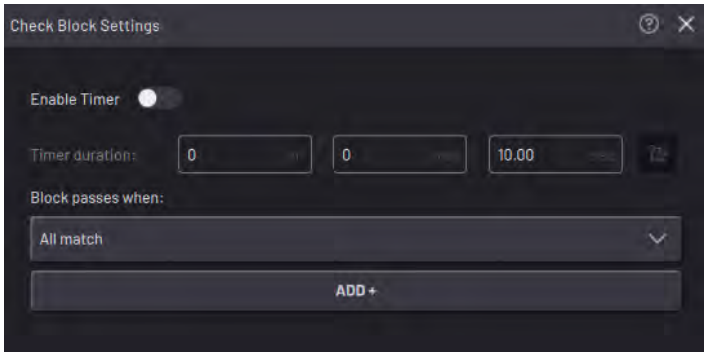
If the inputs don't meet the expected conditions, the task moves on through the right path (**Checks Not Passed**).

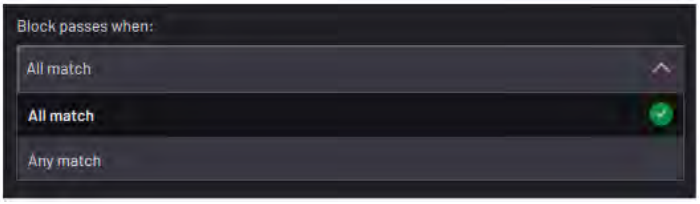


In the Check block settings, choose the check conditions.

Tap the **Block passes when** dropdown to choose:

- **All match:** All comparisons must be true for the block to follow the bottom path.
- **Any match:** At least one of the conditions must be true for the block to follow the bottom path.





By default, a Check block performs the check right away. To make the Check block check continuously for a set period of time, toggle the **Enable Timer** switch and enter a timer duration. When the conditions are met, the block passes to the bottom path and the timer resets for the next execution. If the conditions are not met before the time expires, the block moves to the right path and the timer resets for the next execution.

Tap **ADD+** to add a condition to the block.

Conditions display as:

[Parameter or Signal] [Operator] [Value].

To remove a line, tap the red Delete button.

The Check block can perform these checks:

Parameter or Signal	Operator	Checked Value
Digital Input, digital output	=, ≠	High, low, Boolean variable, digital signal
Analog input, analog output	=, ≠, <, ≤, >, ≥, ≈, ≉	Numerical value, numerical variable, analog signal
Integer or Float variable	=, ≠, <, ≤, >, ≥, ≈, ≉	Numerical value, numerical variable, analog signal, Waypoint or Frame component
String variable	=, ≠	String variable
Boolean variable	=, ≠	High, low, Boolean value, digital signal
Waypoint or Frame	≈, ≉	Waypoint or Frame (on the same robot)

Check a Waypoint or Frame

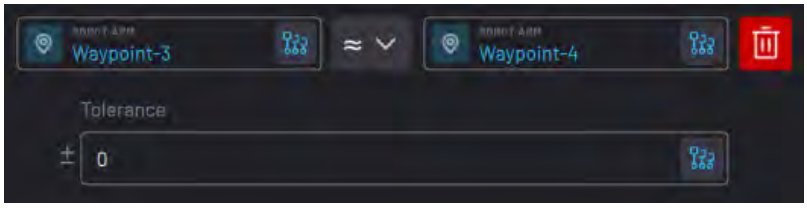
Use the Check block to check if a Waypoint or Frame has changed or if the current position is at a Waypoint or Frame.

The approximate comparisons, ≈ and ≉, check if a Waypoint or Frame is almost equal to another one position. Use them to see if one position is close enough to another position within your defined tolerance. You can compare a linear position to another linear position or a joint position to another joint position.

For example, in the image below we check if the Current Robot Position is approximately at the same position as Waypoint-2, within +/-0.5 mm and +/-1 degree.

When you compare *linear positions*, you enter a Linear Tolerance and Rotation Tolerance. The **Linear Tolerance** defines the allowed linear distance from the checked position. The **Rotation Tolerance** defines the allowed rotation for each axis (RX, RY, RZ) from the checked position. The Check block fails if you use ≈ and any one of the axis rotations is off by more than the tolerance.

When you compare **joint positions**, you enter a Joint Tolerance. The Joint Tolerance defines the allowed rotation for each joint from the checked position's joint values. The Check block fails if you use ≈ and any one of the joint values is off by more than the tolerance.



Note:
You may notice a mismatch between the linear rotational coordinates that you specify and the linear rotational coordinates that the robot reports. For example, if you create a waypoint or frame with the (Rx, Ry, Rz) values of (-150deg, 110deg, -165deg) with respect to the Base Frame, Task Canvas may change these values to (30deg, 70deg, 15deg).

Even though robots accept Ry values between -180deg and 180deg, they only report Ry values between -90deg and 90deg. If your robot adjusts a specified Ry value to fit inside the latter range, it adjusts Rx and Rz values as well. **The specified values and adjusted values represent the same position, so the robot will move as expected. However, the different notation may cause Check blocks to fail.**

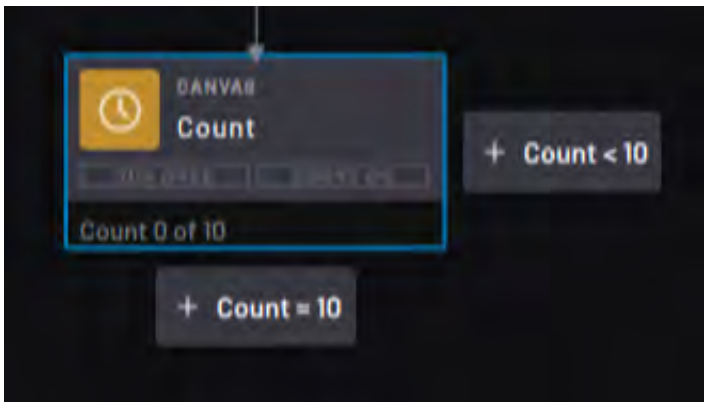
Count Block

The Count block counts up from zero to a chosen number.

Each time the block executes, the count increases by one, then the path taken depends on the count value.

The task follows the right **Count <** path when the count is less than the chosen value.

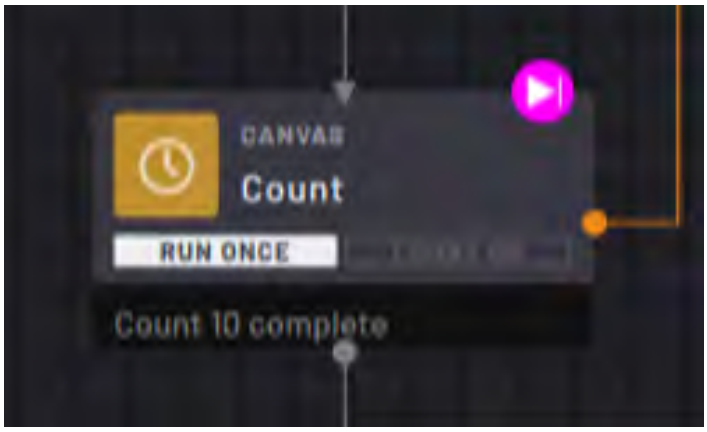
The task follows the bottom **Count =** path after the count reaches the chosen value.



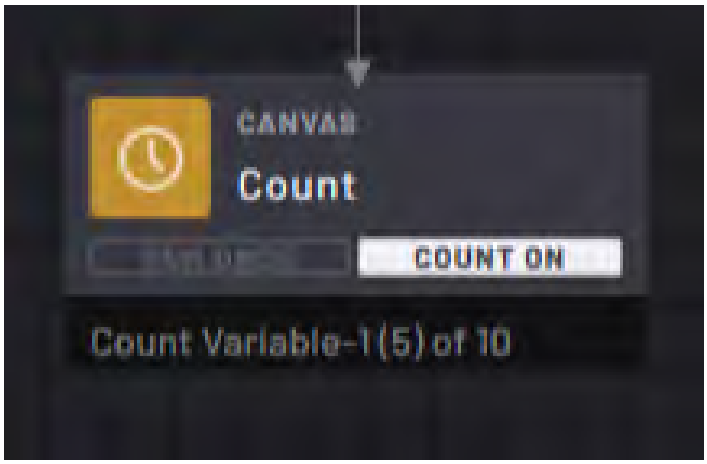
By default, the count resets to "0" after reaching the chosen value. Then the next time the task executes the block, it counts to "1" and follows the right path.

In the Count block settings, choose the **Count to** value.

When the **Run Once** toggle is enabled, the Count does not reset when the Count to value is reached. After it reaches its maximum value, it reads "Count X complete". If the task executes the block again, it follows the bottom path.



When the **Count on Variable** toggle is enabled, the Count is set by the value of an integer variable. This variable can be referenced and used elsewhere in the task. Stepping the block or changing its state will change all other instances of the variable, and changes from any other instances of the variable will also change the Count.



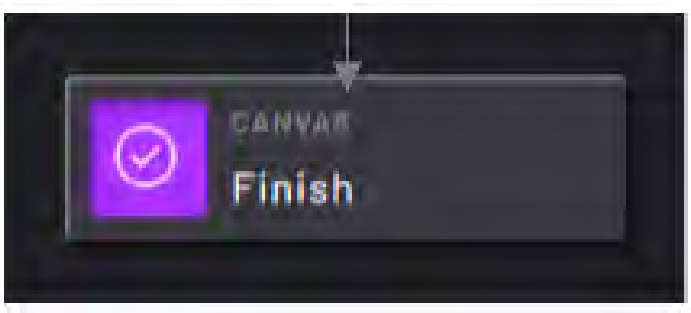
To repeat a section of blocks with a Count block, add the Count block **at the end of the section** and **Connect** the right path of the Count block to the **start of the section.** Use Count blocks to create repeating loops in a task. Use the defined value to set how many times the loop repeats before moving on in the task.

A Count block must have a right path. If there is no right path, the task ends as soon as the Count block is executed.

Finish Block

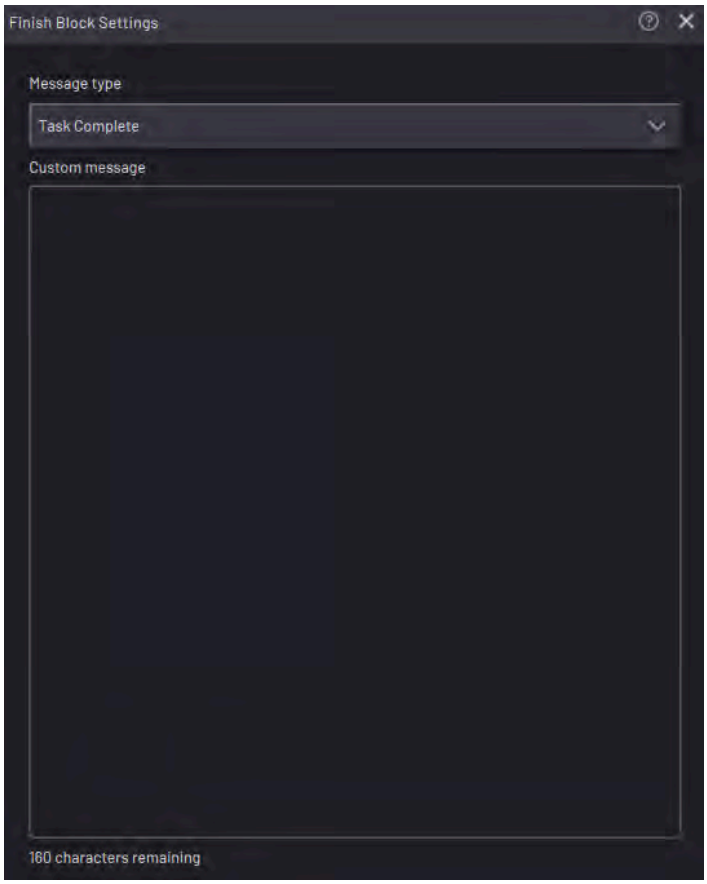
The Finish block marks the end of the task.

A task doesn't need a Finish block to end, but use a Finish block to mark where you expect the task to end. Depending on the complexity of your task, it may use multiple Finish blocks.

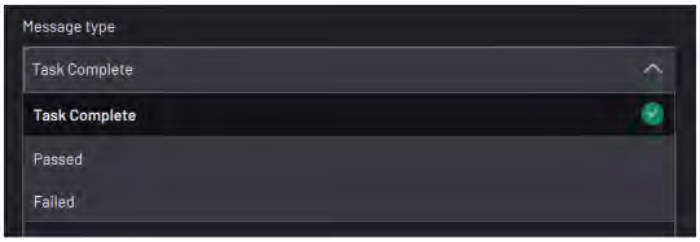


Note:
A Finish block in any subtask stops the entire task, including the Main Task and all subtasks. Only use the Finish block when you want to stop the entire task.

In the Finish block settings, type in a custom message that appears when the Finish block executes.



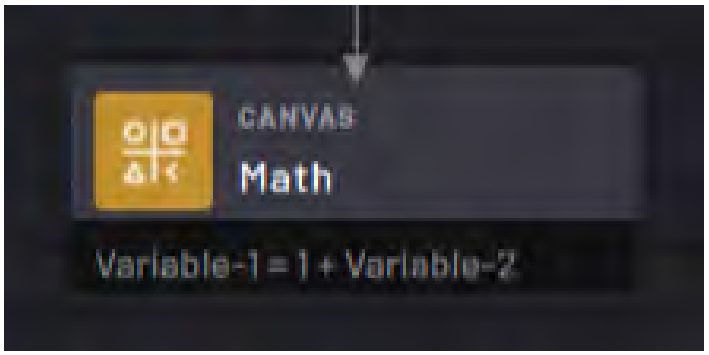
Expand the **Message Type** dropdown to choose the pop-up that appears when the task completes, passes, or fails.



Math Block

The Math block executes mathematical expressions to set the values of variables.

For example, use the Math block to increment a variable as a task runs.



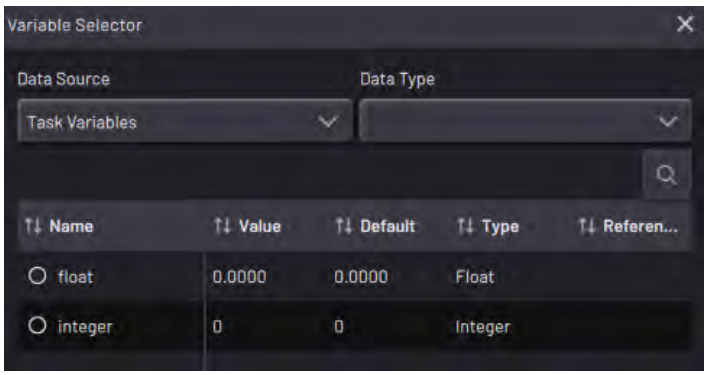
In the Math block settings, create mathematical expressions.

Tap **ADD+** to add an operation to the block.

You may add multiple operations to execute. The Math block executes each operation in order from top to bottom.

Tap the red **Delete** button to remove a line.

Tap the **Selection** button in any field to select an existing variable or signal to change.

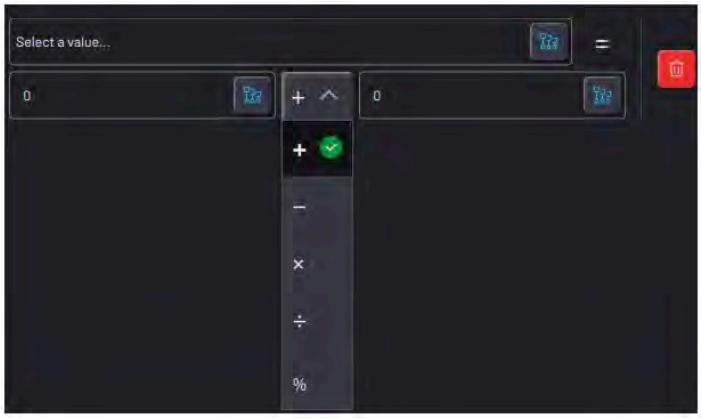


Note:
You can choose **Float** and **Integer** variables in the Math block. You cannot choose non-numerical variables.

The mathematical expressions are defined in the form:

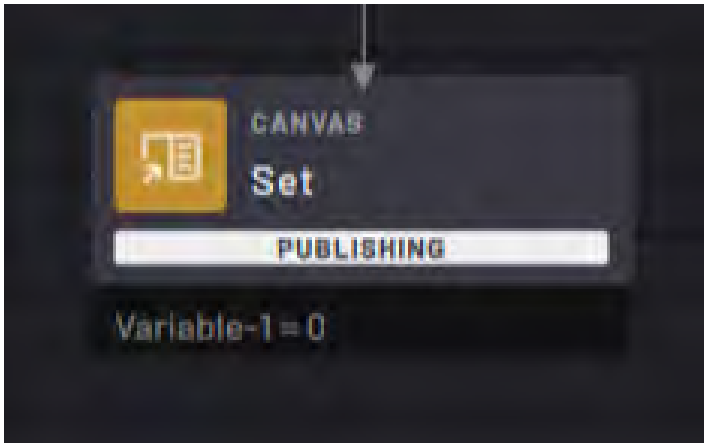
[Variable] = [# or Variable] [Operator] [# or Variable].

The types of operations you can choose from are addition, subtraction, multiplication, division, and modulo (mod).



Set Block

The Set block sets the values or states for chosen variables and parameters.



In the Set block settings, add Set expressions for the block to execute.

Tap **ADD+** to add a Set expression to the block.

Tap the **Selection** button in the left field to select an existing variable or signal.

After you select a variable or signal, enter the value to set it to or tap the **Selection** button to select another variable or signal for the first one to match.

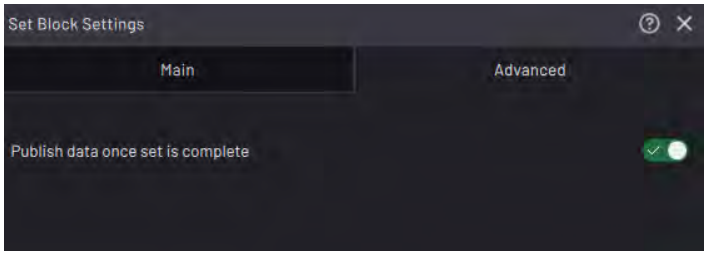
To add multiple variables and/or signals to set, tap the **ADD +** button. The Set block executes operations in order from top to bottom.

Tap the red **Delete** button to remove a line.

The block can set these values:

Type of Variable or Output	Set Value
Digital output	Low, High, Invert, variable value, or digital signal value
Analog output	Rational numbers, variable value, or analog signal value
Float variable	Rational numbers, variable value, or analog signal value
Integer variable	Whole numbers (positive and negative) or variable value
String variable	Alphanumeric string with special characters
Boolean variable	Low, High, Invert, variable value, or digital signal value
Waypoint/Frame	Another Waypoint/Frame (from the same device)

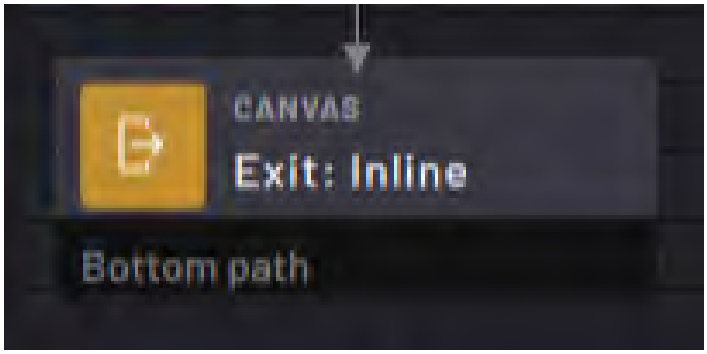
In the "**Advanced**" tab, enable the "**Publish data once set is complete**" to bypass synchronization delays and immediately broadcast the changed values from the block. Enabling this toggle is not necessary to guarantee that value changes are synchronized within Task Canvas.



Subtask > Exit: Inline Subtask Block

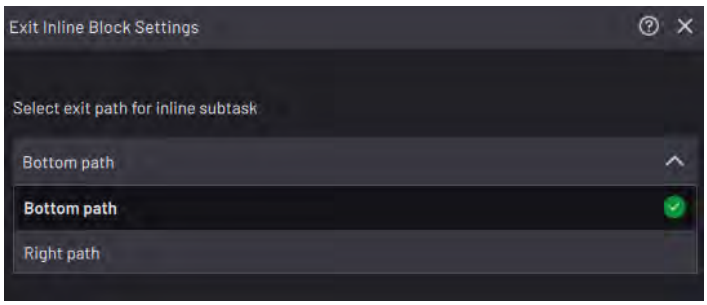
The Exit block completes an inline subtask and exits to the canvas where the subtask was called. Without an Exit block in an inline subtask, the subtask and the calling task finish after executing the last available block.

When an Exit block is executed, the task returns to the Start Subtask: Inline block that called the subtask and follows the chosen path (bottom or right).



Note:
The Exit block is not the same as the Finish block. The Finish block stops all subtasks, including the main task. Only use the Finish block to stop all activity in your task.

In the Exit block settings, choose whether the subtask exits to the Bottom or Right path of the **Start Subtask: Inline** block it returns to.



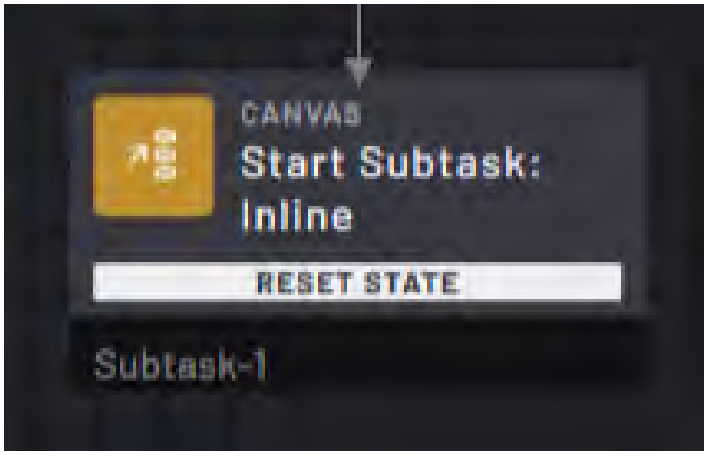
Subtask > Start Subtask: Inline Block

The Start Subtask: Inline block starts an inline subtask of your choice.

When the block is executed, execution enters and starts the inline subtask.

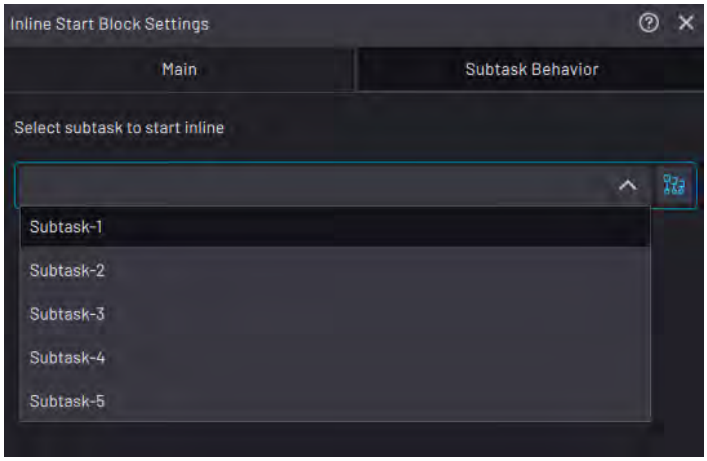
If the inline subtask executes an Exit block, the task returns to the Start Subtask: Inline block and follows the path set in the Exit block.

If you don't include an Exit block in the inline subtask, the subtask and the calling task finish after executing the last available block.

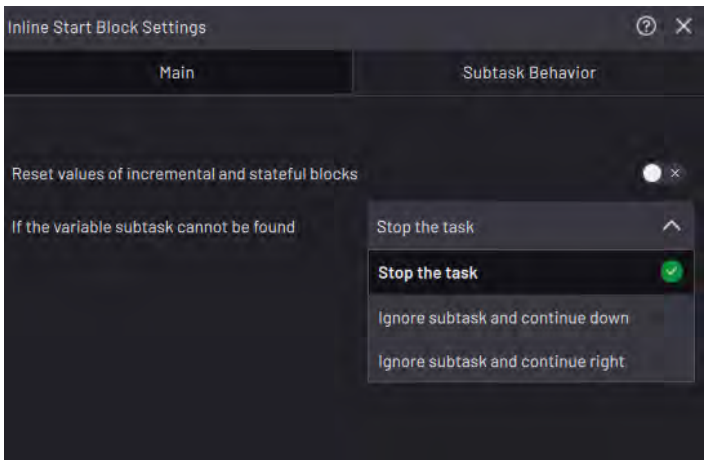


In the block settings, use the dropdown selector or variable selector in the **Main** tab to select which inline subtask to start.

Calling a subtask from a string or integer variable provides flexibility for having this block call whichever subtask has a name that corresponds to the current variable value. For example, if this block calls a string variable named “Variable-1” and a Set block before it sets Variable-1 to “Subtask-1”, this block will call Subtask-1.



In the **Subtask Behavior** tab, enable the toggle next to **Reset values of incremental and stateful blocks** to reset all blocks in the subtask before it starts to execute. This resets Count, Timer, Grid, and other state-based blocks before the subtask executes again.



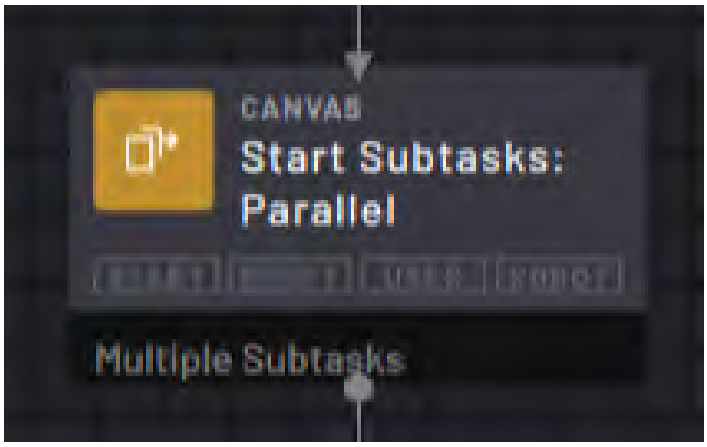
Note:
If you are calling a subtask from a string or integer variable, select what happens when the variable subtask cannot be found. You can **Stop the task** (to stop the task as soon as the flowchart encounters the invalid subtask) or **Ignore subtask and continue down/right** (to continue the task down the bottom or right paths).

Subtask > Start Subtasks: Parallel Block

The Start Subtasks: Parallel block starts one or more parallel subtasks of your choice.

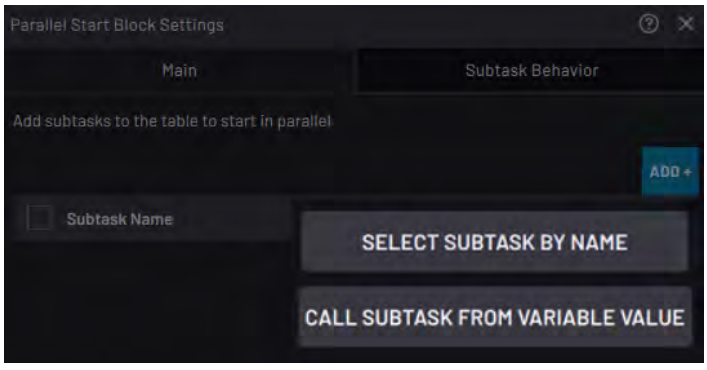
When the block is executed, the parallel subtasks start execution while the task continues along the bottom path.

The subtasks you start execute at the same time as the Main Task until you stop it or until the subtasks reach final blocks.



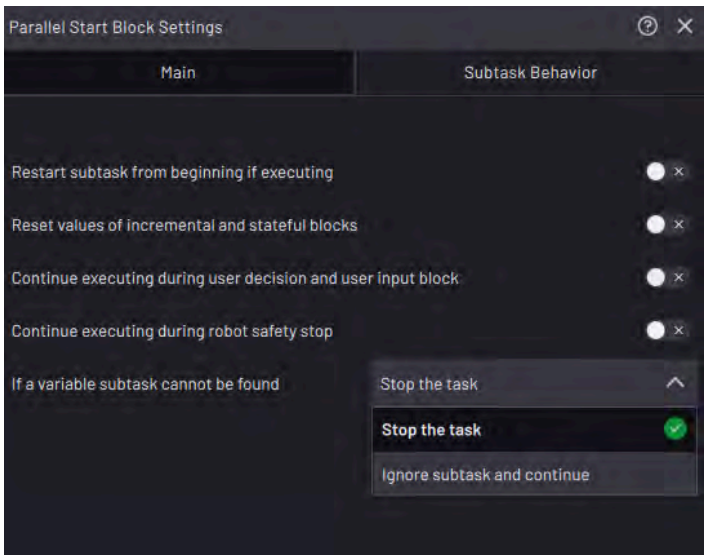
In the **Main** tab of the block settings, tap **ADD +** to select which parallel subtasks to start.

Calling a subtask from a string or integer variable provides flexibility for having this block call whichever subtask has a name that corresponds to the current variable value. For example, if this block calls a string variable named “Variable-1” and a Set block before it sets Variable-1 to “Subtask-1”, this block will call Subtask-1.



In the **Subtask Behavior** tab, enable the toggles to:

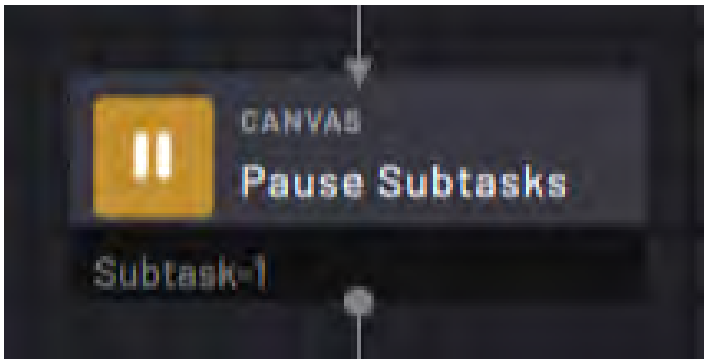
- Restart the subtasks if they were running when this block executes.
- Reset the state of counting blocks (like Count and Grid) before the subtask executes again.
- Ignore pauses during User Decision and User Input blocks (or continue the subtask even when a pause takes place in another subtask).
- Ignore pauses during robot safety stops (or continue the subtask even when a robot safety stop occurs).



Note:
If you are calling a subtask from a string or integer variable, select what happens when a variable subtask cannot be found. You can **Stop the task** (to stop the task as soon as the flowchart encounters an invalid subtask) or **Ignore subtask and continue** (to only ignore the invalid subtasks but still execute any valid subtasks).

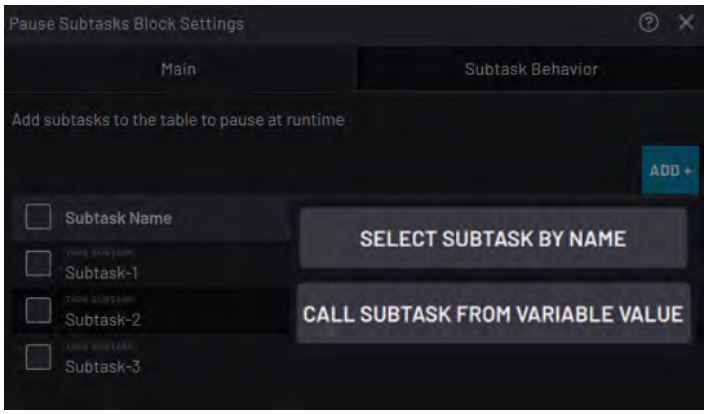
Subtask > Pause Subtasks Block

The Pause Subtasks block pauses one or more subtasks until you resume them with a Resume Subtasks block.

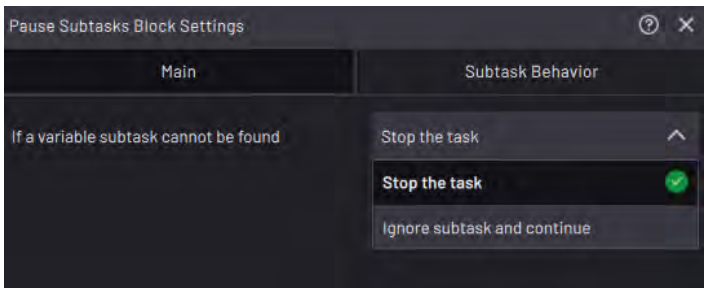


In the block settings, tap **ADD +** to select one or more subtasks to pause at the block's execution.

Calling a subtask from a string or integer variable provides flexibility for having this block call whichever subtask has a name that corresponds to the current variable value. For example, if this block calls a string variable named "Variable-1" and a Set block before it sets Variable-1 to "Subtask-1", this block will call Subtask-1.

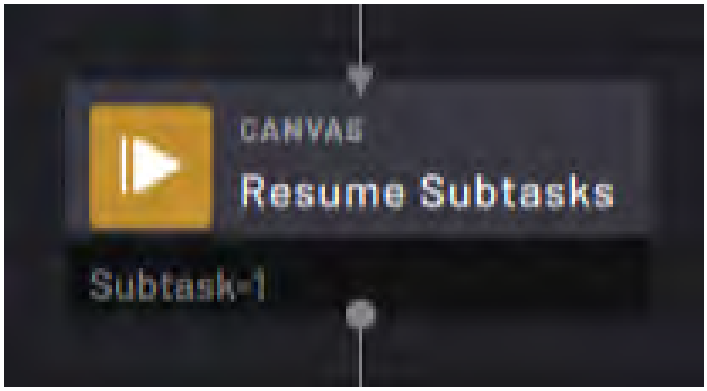


Note:
If you are calling a subtask from a string or integer variable, select what happens when a variable subtask cannot be found. You can **Stop the task** (to stop the task as soon as the flowchart encounters an invalid subtask) or **Ignore subtask and continue** (to only ignore the invalid subtasks but still execute any valid subtasks).



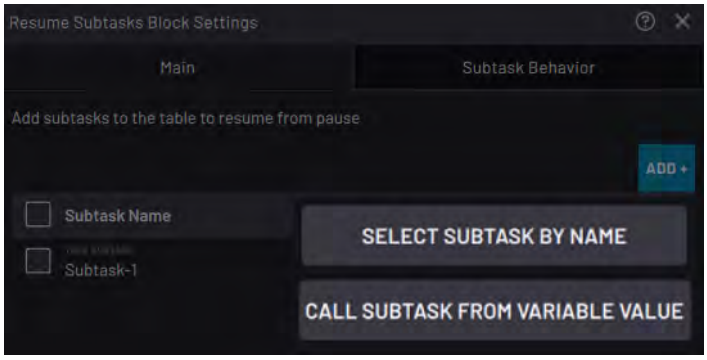
Subtask > Resume Subtasks Block

The Resume Subtasks block resumes one or more subtasks that you paused with a Pause Subtasks block.

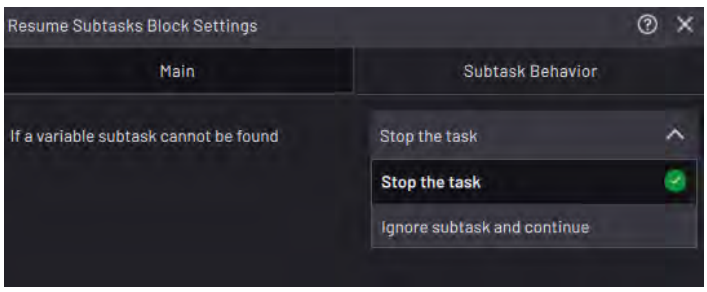


In the block settings, tap **ADD +** to select one or more subtasks to resume at the block's execution.

Calling a subtask from a string or integer variable provides flexibility for having this block call whichever subtask has a name that corresponds to the current variable value. For example, if this block calls a string variable named "Variable-1" and a Set block before it sets Variable-1 to "Subtask-1", this block will call Subtask-1.



Note:
If you are calling a subtask from a string or integer variable, select what happens when a variable subtask cannot be found. You can **Stop the task** (to stop the task as soon as the flowchart encounters an invalid subtask) or **Ignore subtask and contin-**



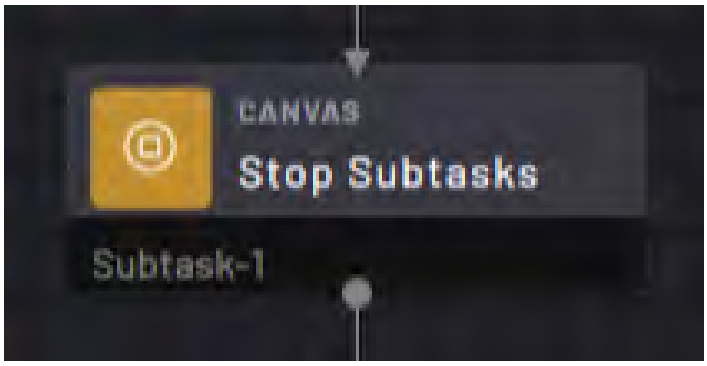
ue (to only ignore the invalid subtasks but still execute any valid subtasks).

Subtask > Stop Subtasks Block

The Stop Subtasks block stops one or more selected subtasks and immediately continues along the bottom path.

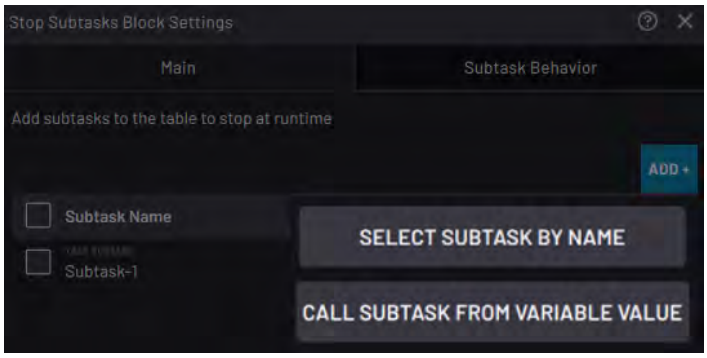
Use this block to stop subtasks from another subtask when they no longer need to run.

For example, you may stop a subtask that runs a conveyor if you run out of parts.

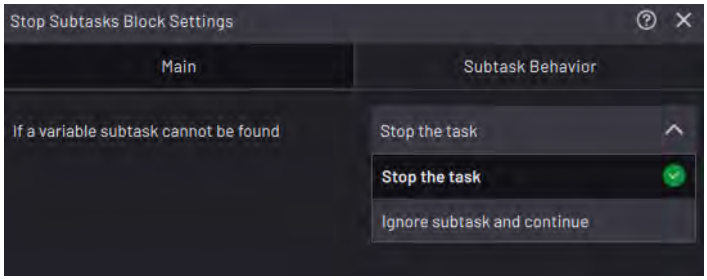


In the block settings, tap **ADD +** to select the subtasks to stop.

Calling a subtask from a string or integer variable provides flexibility for having this block call whichever subtask has a name that corresponds to the current variable value. For example, if this block calls a string variable named “Variable-1” and a Set block before it sets Variable-1 to “Subtask-1”, this block will call Subtask-1.



Note:
If you are calling a subtask from a string or integer variable, select what happens when a variable subtask cannot be found. You can **Stop the task** (to stop the task as soon as the flowchart encounters an invalid subtask) or **Ignore subtask and continue** (to only ignore the invalid subtasks but still execute any valid subtasks).



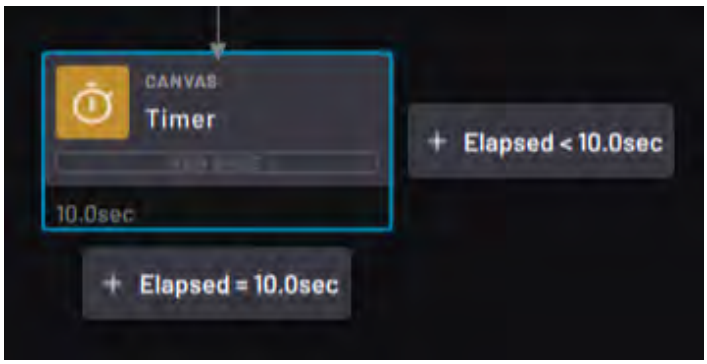
Timer Block

The Timer block starts a timer that counts down while the task moves on.

The timer begins when the Timer block is executed, and it continues until the time reaches zero.

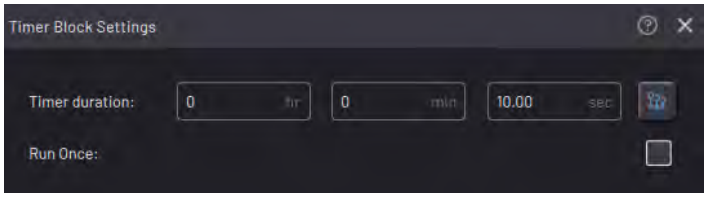
When the block is executed again, the path taken depends on the timer value.

The task follows the right **Elapsed <** path while the timer is running. When the timer reaches zero, the task follows the bottom **Elapsed =** path.

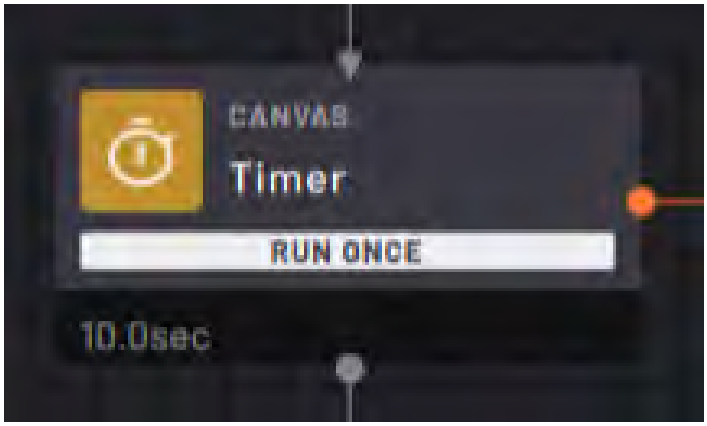


By default, the countdown resets to the defined length of time after it reaches zero. The next time you execute the Timer, it starts the countdown over.

In the Timer Block settings, choose the **Timer duration**. Leave the Run Once checkbox checked or unchecked.



When the **Run Once** checkbox is checked, the Timer countdown runs only the first time you execute the block. Then next time the block is executed, the task follows the bottom path.

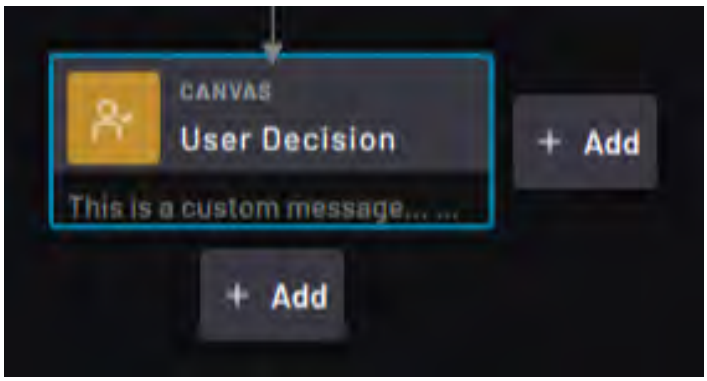


To create a loop of blocks that executes while the timer is running, set the looped blocks on the Timer block's **Elapsed <** path. Then connect the end of the loop back to the Timer block. A Timer block must have a right path. If there is no right path, the task ends as soon as the timer starts.

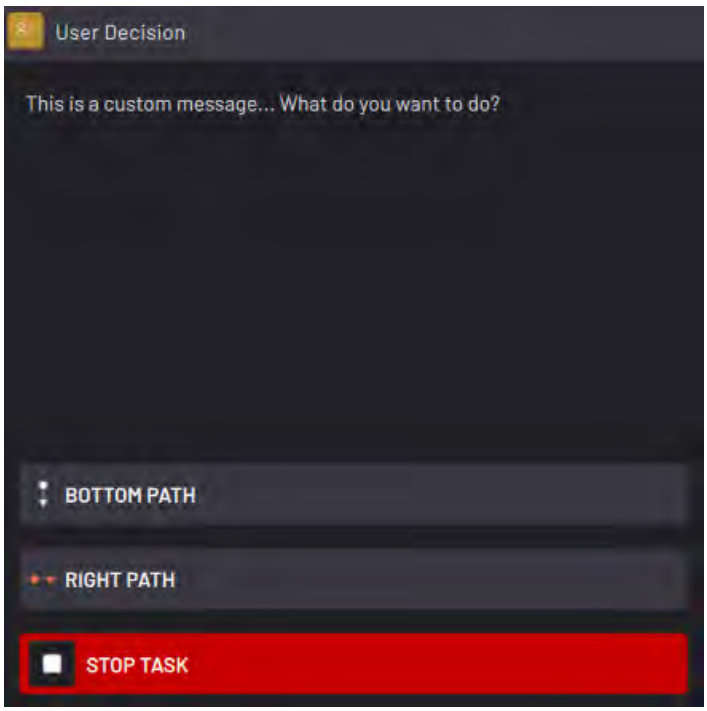
User Decision Block

The User Decision block pauses execution until you instruct the task to resume or stop.

When executed, the User Decision block displays custom text and prompts you to choose which path to follow (the bottom path or the right path).



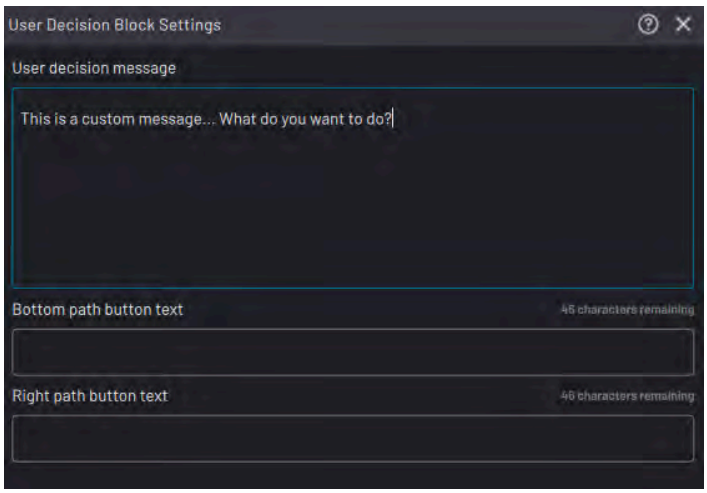
When the block is executed, a pop-up pauses execution and prompts you with the custom message and path choices. Tap one of the buttons to do that action.



In the User Decision block settings, you enter a message and path button texts.

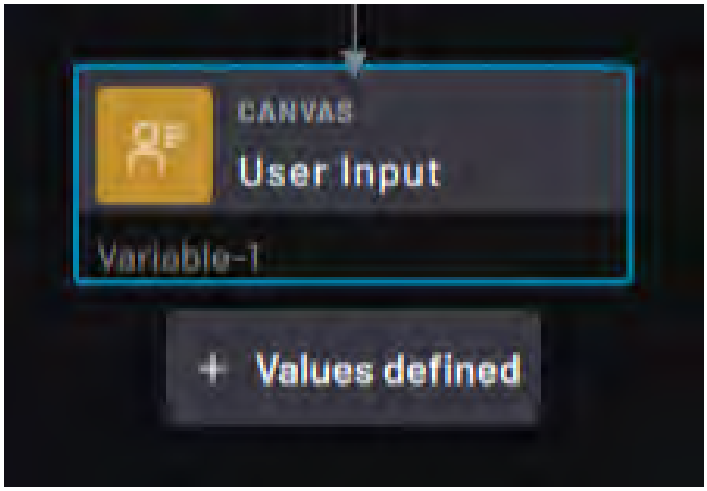
Add text to the **User decision message** box to display when the block executes at runtime.

By default, the buttons are labeled **BOTTOM PATH**, **RIGHT PATH**, and **STOP TASK**. Tap the Bottom Path and Right Path text boxes to customize the runtime button text.



User Input Block

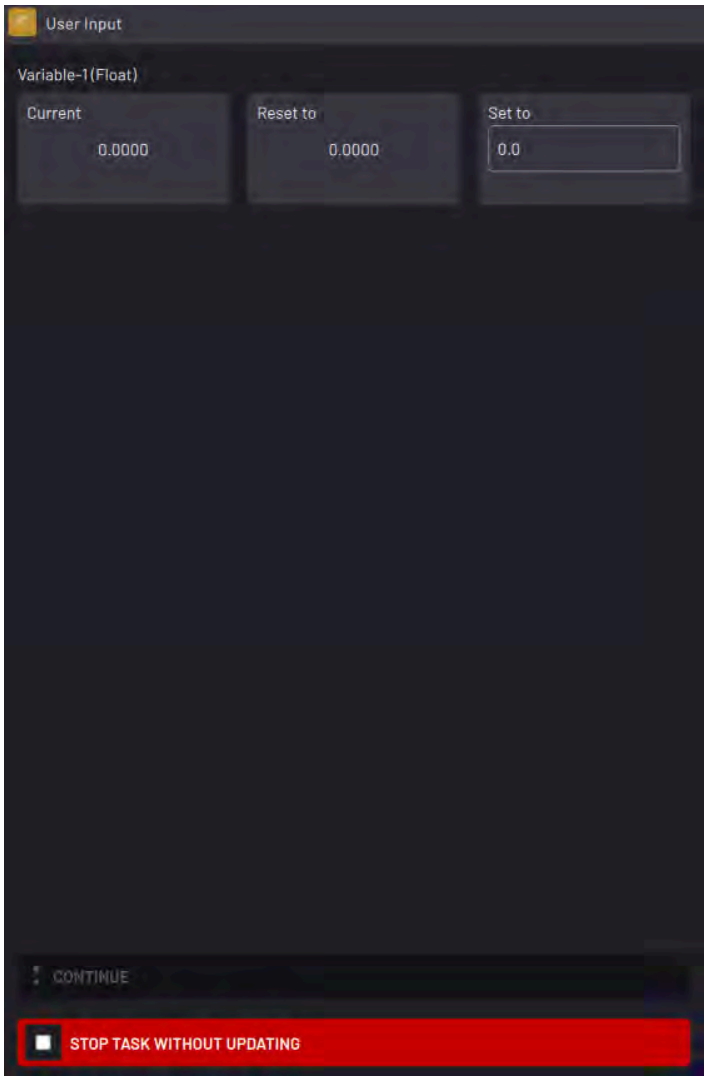
The User Input block pauses execution and prompts you to choose the values of task variables before moving on.



Note:
You can choose **Float, Integer,** and **String** variables in the User Input block. You cannot choose a **Boolean** variable.

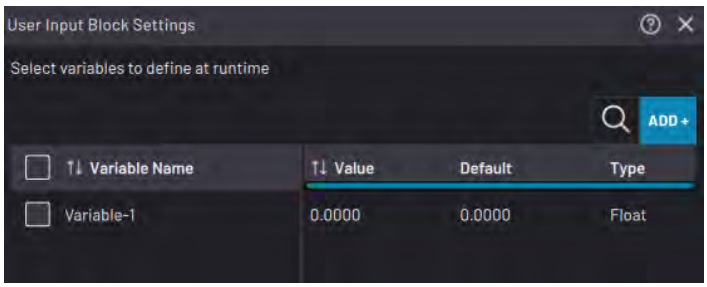
When the User Input block is executed, a pop-up prompts you to provide a value for each selected variable.

For each listed variable, choose to keep its **Current** value, **Reset** it to its default value, or **Set** it to another value. Or tap **STOP TASK**.



In the User Input block settings, add variables for you to set when the block is executed.

To add a variable to the table, click **ADD +** and select it from the Variable Selector.

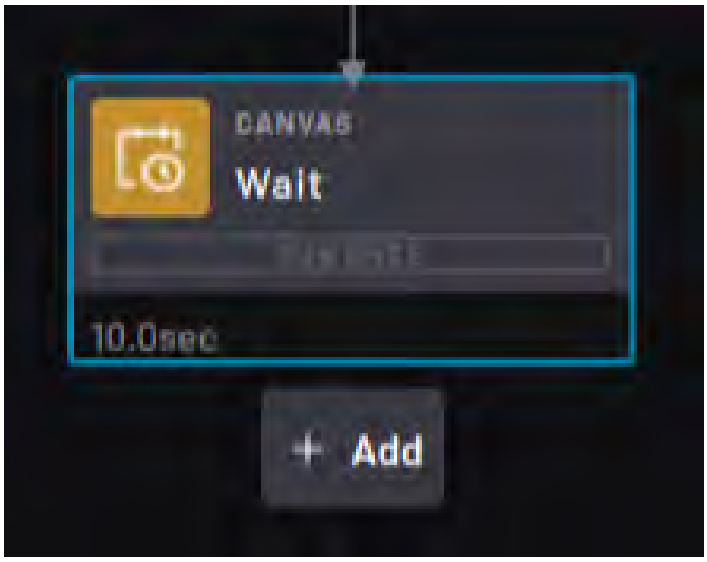


Wait Block

The Wait block pauses execution for a defined length of time.

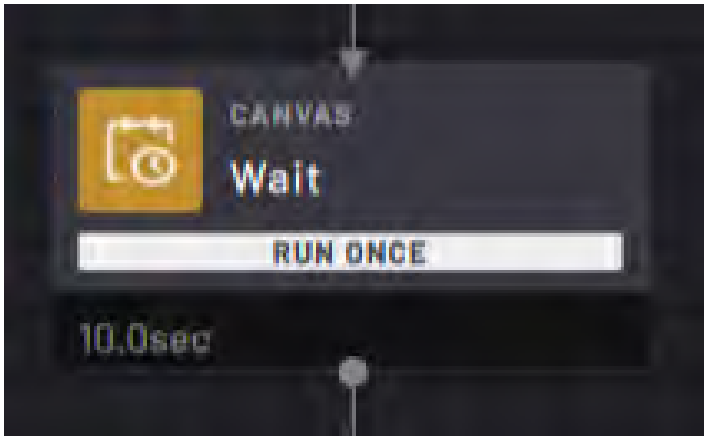
When the Wait block is executed, the count down begins. When the Wait block ends, the task continues through the bottom path.

By default, the countdown resets to the defined length of time after it reaches zero. The next time you execute the Wait, it starts the countdown over.



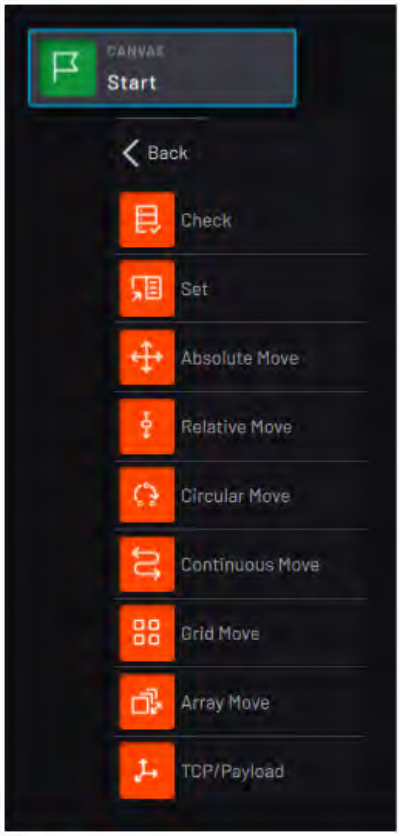
In the Wait Block Settings, choose the Wait duration. Leave the Run Once checkbox checked or unchecked.

When the Run Once checkbox is checked, the Wait count-down runs only the first time you execute the block.



Robot Command Blocks

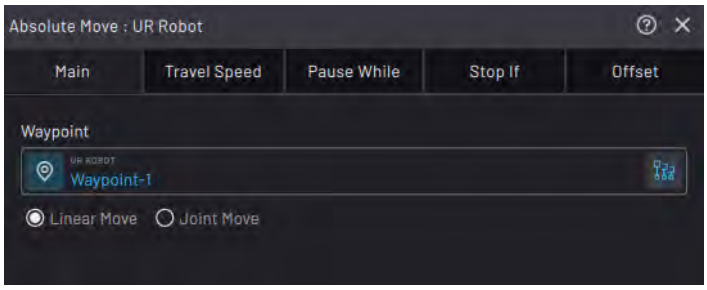
The Robot Command blocks are available in the block menu for active robot devices. They include robot move blocks, robot IO blocks, and other robot command blocks.



All robot move blocks share a few common features:

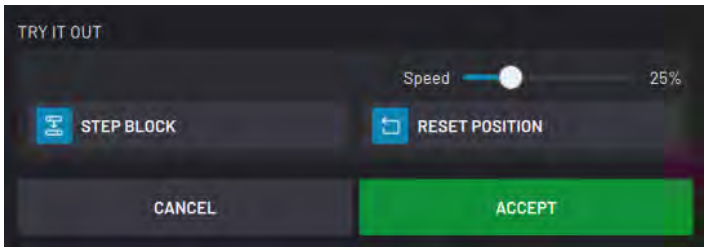
The Main Tab

Set waypoints and other robot motion parameters. Requirements vary based on the block.



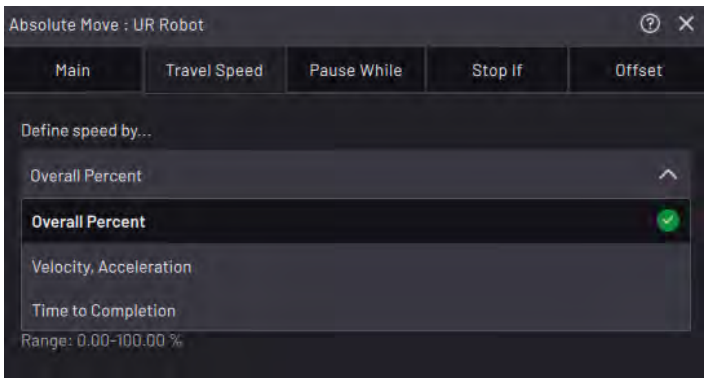
The TRY IT OUT Section

Preview the move. Tap **STEP BLOCK** to move the robot to the set position. Drag the **Speed** slider to determine how fast the robot moves. Tap **RESET POSITION** to move the robot to the position it was in when you opened the block.



The Travel Speed Tab

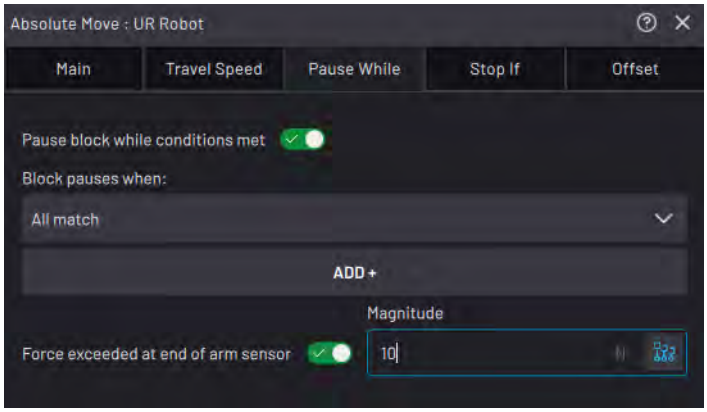
Set the speed of the motion. Select in the dropdown how you want to define the speed. All robots have the option of defining speed by Overall Percent (0%-100%) of the robot's maximum speed. Some robots also have the options of defining speed by precise parameters (i.e., Time to Completion or some combination of Velocity, Acceleration, and Deceleration).



Note:
If you define motion by velocity, acceleration, deceleration, or time to completion and then import the task to another robot that doesn't support those parameters, the block may show an error until you correct and save the block.

The Pause While Tab

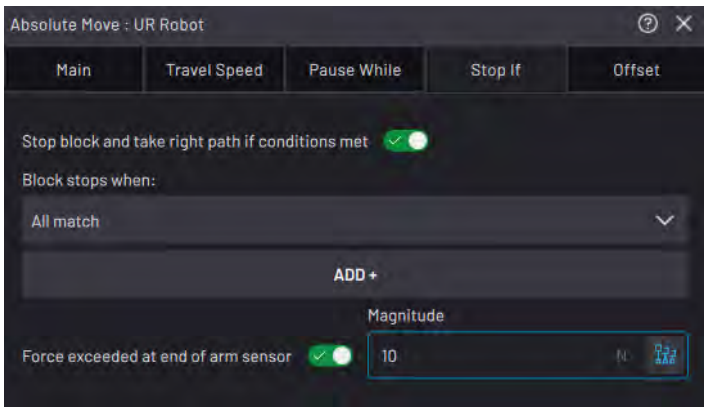
Set conditions that pause the block while all conditions are met or while any conditions are met. Enable the "**Pause block while conditions met**" toggle to use this feature. If this toggle is enabled, you must have conditions listed for the **ACCEPT** button to be available. Add conditions with the **ADD +** dropdown.



Note:
If you have a force sensor listed in the robot's configuration in Device Configuration, enable the "**Force exceeded at end of arm sensor**" toggle. In the **Magnitude** field, set the force threshold that would pause the block.

The Stop If Tab

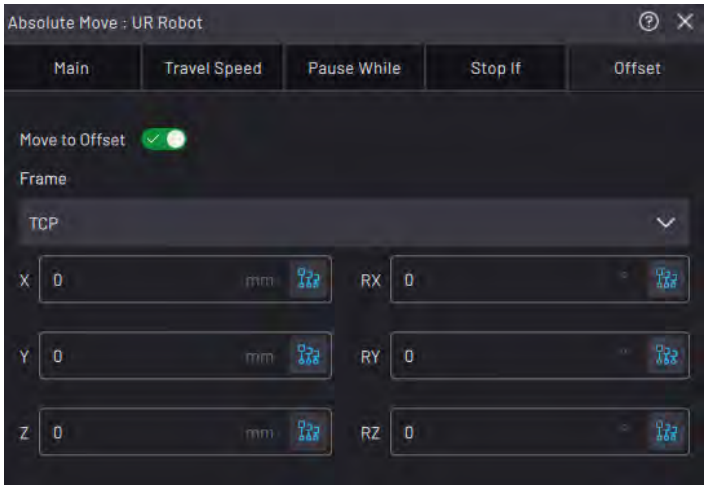
Set conditions that stop the block while all conditions are met or while any conditions are met. When this occurs, the task proceeds down the block's right path. Enable the "**Stop block and take right path if conditions met**" toggle to use this feature. If this toggle is enabled, you must have conditions listed for the **ACCEPT** button to be available. Add conditions with the **ADD +** dropdown.



Note:
If you have a force sensor listed in the robot's configuration in Device Configuration, enable the "**Force exceeded at end of arm sensor**" toggle. In the **Magnitude** field, set the force threshold that would stop the block.

The Offset Tab

Enable the Move to Offset toggle to offset the motion. An offset defines a relative position from the waypoint(s) to which the robot moves. Use the Frame dropdown to set the Frame in which the offset applies. Use the value fields to set the offset from the waypoint along the axes of that Frame.



Robot I/O Blocks

The Check and Set blocks are the same as the Canvas Check and Set blocks, but the device is filtered to the robot device.

Check

See *Canvas Blocks > Check (on page 170)*.

Set

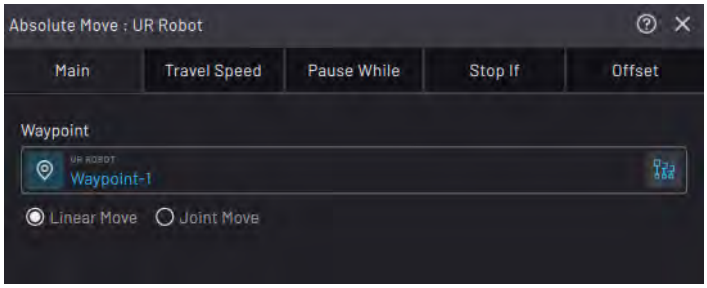
See *Canvas Blocks > Set (on page 176)*.

Absolute Move Block

The Absolute Move block performs a motion to a single position defined by a waypoint.

In the **Main** tab, tap the **Waypoint selector** to create a new waypoint or choose an existing waypoint.

Tap **Linear Move** or **Joint Move** to select the motion that the robot performs to the waypoint.



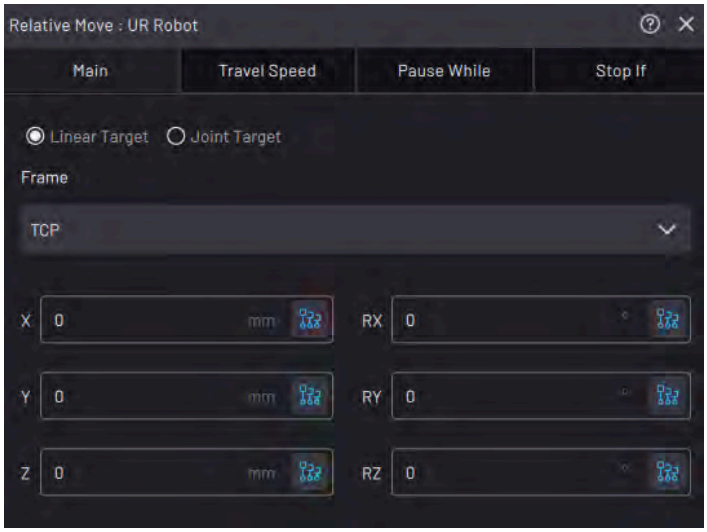
- A **Linear** move tells the robot to take the shortest path to the programmed TCP pose. Depending on where the robot is moving from, several joint positions may be possible for the same TCP pose.
- To guarantee that the robot moves to the same joint positions every time, choose **Joint**.

See Robot Command Blocks *(on page 187)* for more info on the other block features.

Relative Move Block

The Relative Move block performs a motion relative to the robot's position when the block executes. Because the motion is relative, the ending position depends on the starting position.

In the **Main** tab, tap **Linear Target** or **Joint Target**.



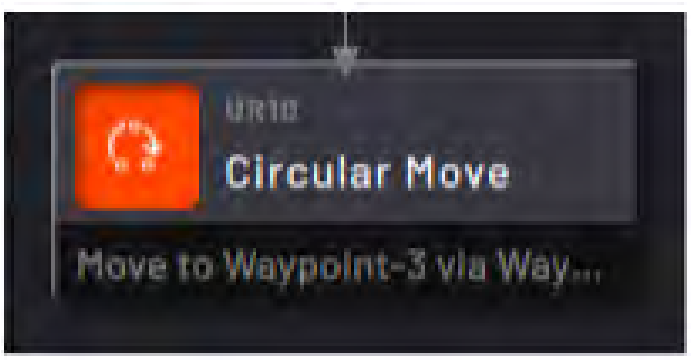
- Linear lets you select a Frame and program a motion along and around the Frame's axes.
- Joint lets you program how far one or more joints rotate.

See Robot Command Blocks *(on page 187)* for more info on the other block features.

Circular Move Block

The Circular Move block performs an arc motion (with a constant radius) relative to the robot's position when the block executes.

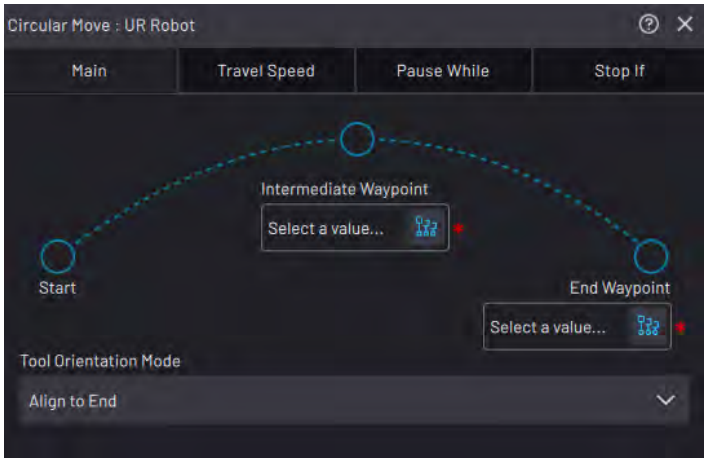
In the **Main** tab, tap the **Intermediate Waypoint** and **End Waypoint** selectors to choose the waypoints that define those points on the arc.



Note:
The block does not define a starting position, so the robot's position defines the arc when the block executes. The same block can execute different motions from different start positions.

The Intermediate Waypoint cannot be at the same location as the Start or the End Waypoint.

Note:
Forge/OS calculates circular moves using the TCP. You can only select Linear waypoints. You cannot select Joint waypoints.



Use the **Tool Orientation Mode** dropdown to choose how the tool moves through the arc. **Align to End** moves the Tool Center Point (TCP) through the arc, so it finishes the motion in the exact orientation of the End Waypoint. **Maintain Tangency** keeps the TCP orientation relative to the arc center point the same as the start position. If you select Maintain Tangency, the TCP's orientation at the end may be different than the End Waypoint orientation.

See Robot Command Blocks *(on page 187)* for more info on the other block features.

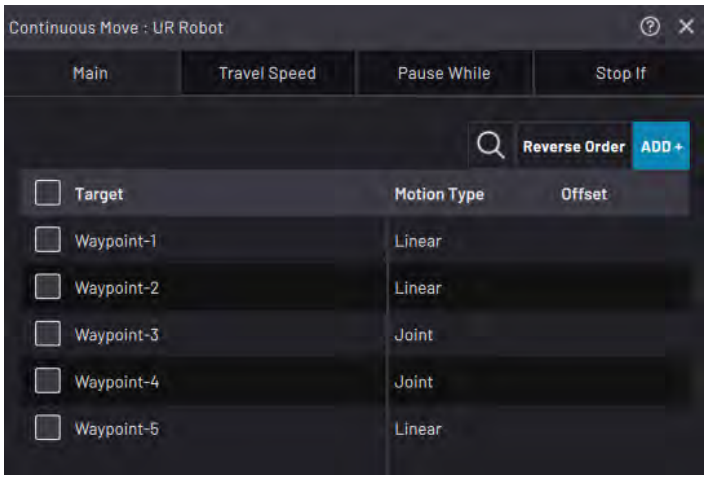
Continuous Move Block

The Continuous Move block performs a fluid and unbroken motion through all the programmed positions and relative motions.

Use a Continuous Move to program a motion that requires more than one move type or multiple points to define the motion.

The **advantages** of using a Continuous Move are:

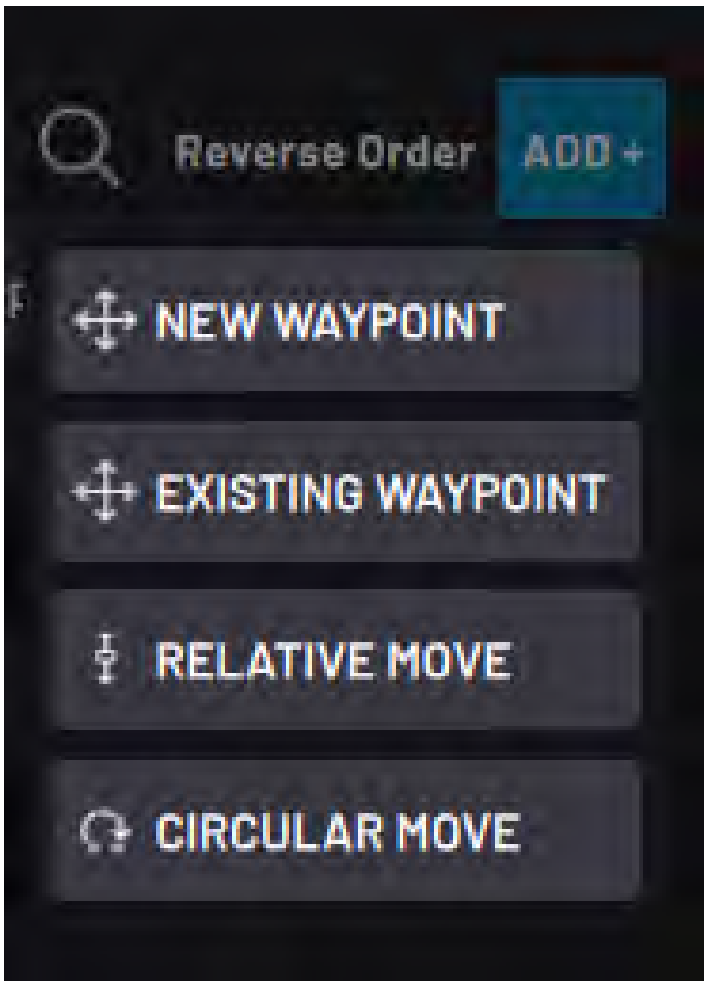
- A cleaner Canvas: Instead of a complex motion with 10 waypoints requiring 10 Absolute Move blocks, a single Continuous Move block can contain all 10 waypoints.
- Smoother and faster motion: The robot executes changes in direction more fluidly and doesn't need to come to a complete stop between targets.



In the **Main** tab, the **Target** table shows the list of positions and motions that the robot executes.

Tap **ADD +** to select a motion to add to the table.

From the dropdown, select an option.

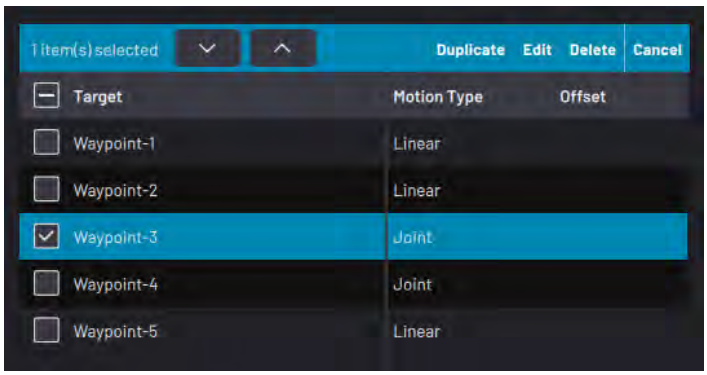


> **Tip:**
> To program Grid or Array Moves into a Continuous Move, look for the **Resolve Position to Waypoint** feature in those block descriptions.

Motion Type	Description
Absolute Move with New Waypoint	Create a new waypoint and add it to the table as an Absolute Move. Choose the motion type and set an offset.
Absolute Move with Existing Waypoint	Choose an existing waypoint and add it to the table as an Absolute Move. Choose the motion type and set an offset.
Relative Move	Program a relative motion using linear or joint values. The starting point of the relative motion is the previous position in the Target table.
Circular Move	Program a circular motion using an intermediate and end waypoint. The starting point of the circular motion is the position before it in the Target table.

After programming a motion, it appears in the table on the Main tab. Select a single target in the table to edit it or change its order using the **Up and Down arrows**.

To reverse the order of all targets, deselect all targets and tap **Reverse Order**. Use this when you want to duplicate and reverse a Continuous Move you programmed earlier.



To edit a target, select it on the table and tap **Edit**. You may edit one target at a time.

To remove one or more targets from the Continuous Move, select them and tap **Delete**.

To duplicate a target in the Continuous Move, select that target in the table and tap **Duplicate**.

In the **Travel Speed** tab, **Motion Smoothing** defines how much the robot attempts to smooth the transition between the motions. A large blend radius creates a more fluid motion, but it increases the curving near targets. A small blend radius creates longer, more precise motions to each target before changing directions.

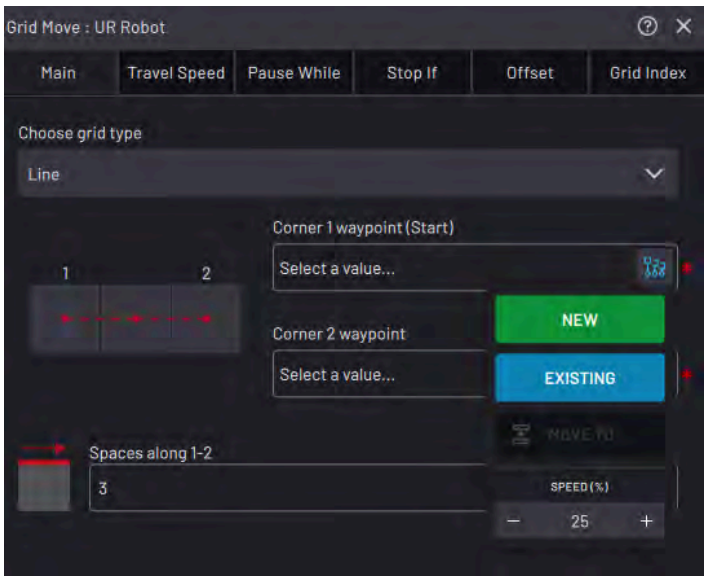
See Robot Command Blocks *(on page 187)* for more info on the other block features.

Grid Move Block

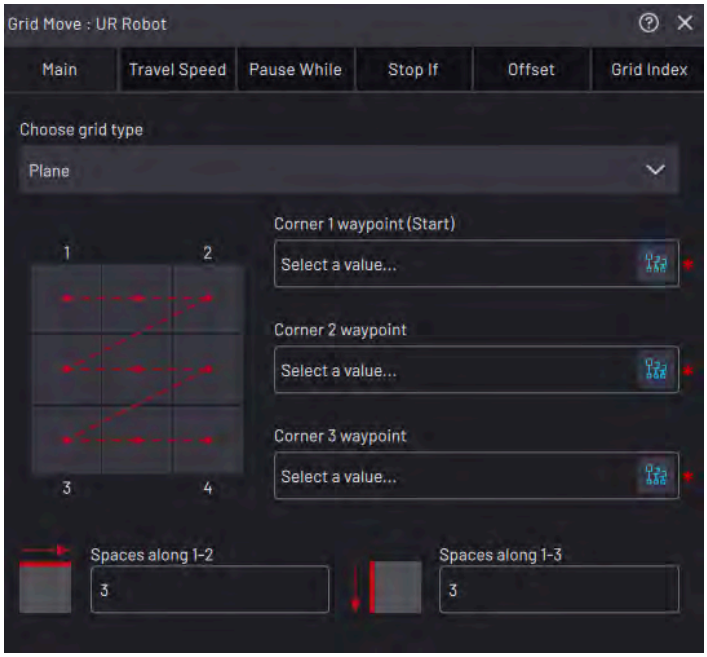
The Grid Move block creates a repeating pattern of positions that the robot moves to, one at a time. Each time the block executes, the robot moves to the next position in the grid and then continues to the next block. This continues until the robot has moved to all the positions (which prompts it to restart at the beginning of the grid).

In the **Main** tab, choose the grid type from the dropdown, either **Line** (one-dimensional) or **Plane** (two-dimensional).

For a linear grid, choose the start and end waypoints and the number of positions along the line. The block fills in intermediate positions with equal spacing.

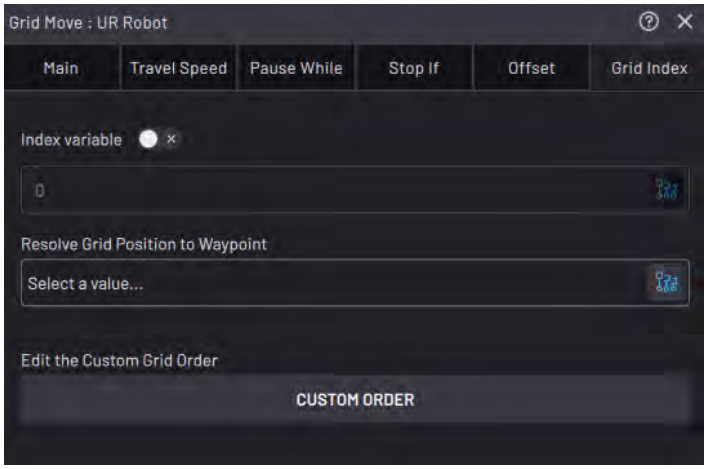


For a planar grid, choose the three corner waypoints and the number of positions between corners 1 and 2, and corners 1 and 3. Use the grid visualization to make sure your corners are in the correct positions. By default, the robot moves first along the row from corner 1 to corner 2, then starts the next row.



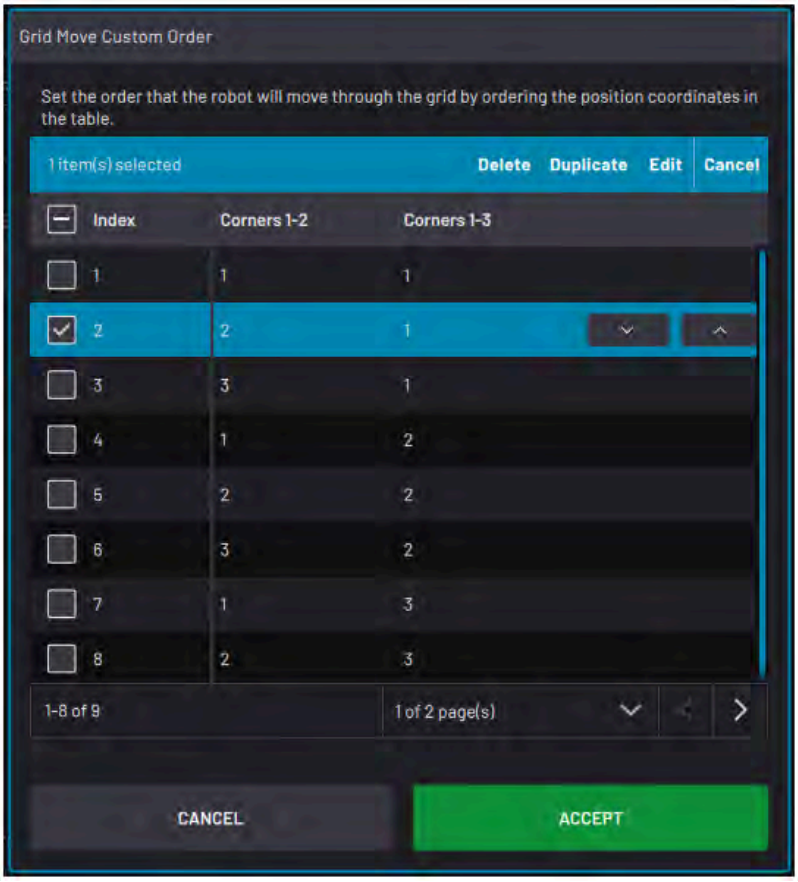
In the **Grid Index** tab, customize the indexing behavior of the Grid Move. The **Index Variable** field is where you choose an integer variable to decide which index the Grid Move moves to when it executes. When you use an index variable, your task must update the variable with other blocks.

Choose a waypoint in the **Resolve Grid Position to Waypoint** field to update that waypoint with the next grid position each time the block executes.



> **Note:**
> When there is a waypoint in the Resolve field, the grid block does **not** move the robot. It saves the chosen waypoint at that grid position. Then you may use this waypoint in other move blocks, like the Continuous Move. Execute the Grid Move block to update the waypoint with the next position in the grid.

Tap **CUSTOM ORDER** to set the order that the robot moves through the grid. Select a position coordinate on the table and use the up and down arrows to move it earlier or later in the sequence.



Tap **Delete** at the top of the table to remove the selected position from the sequence.

Tap **Duplicate** to copy the selected position.

Tap **Edit** to enter a new row or column coordinate.

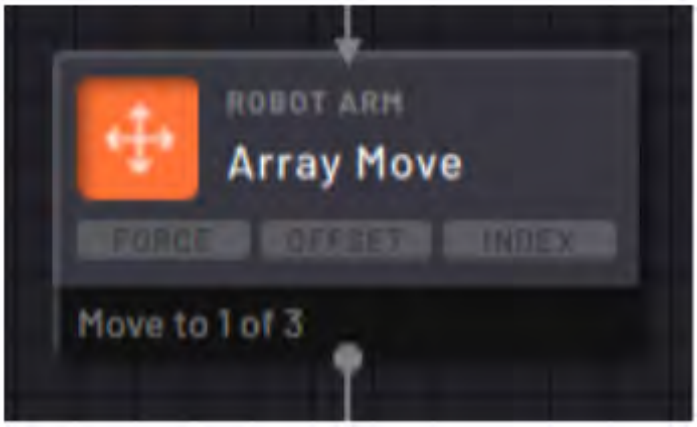
To add a new index, tap **ADD +**. Enter the Index and the row and column coordinates. By default, the Index field fills in as one integer value larger than the last table entry. If you type in a smaller value, the new entry is added at that order of the sequence. The existing position that used to have that index (and everything after it) are pushed down.

To undo your re-ordering, deletion, duplication, and coordinate changes, select none of the positions and tap **Reset** at the top of the table.

See Robot Command Blocks *(on page 187)* for more info on the other block features.

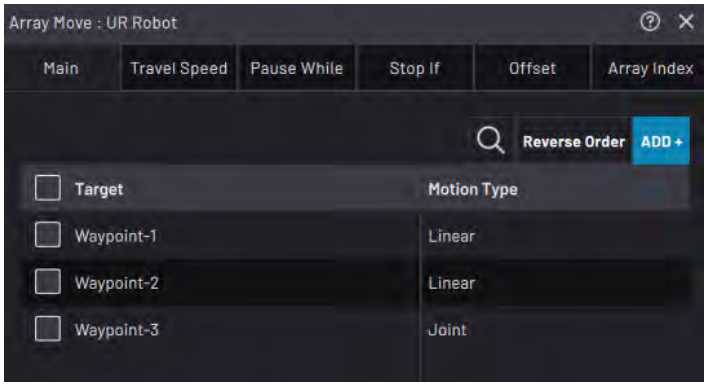
Array Move Block

The Array Move specifies a custom list of waypoints that the robot moves to, one at a time, each time the block executes.



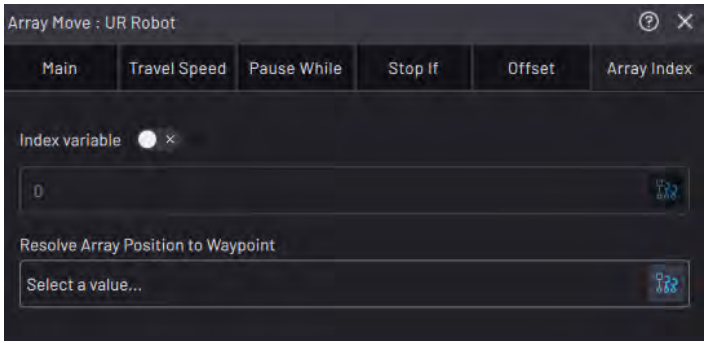
In the **Main** tab, the Target table shows the list of targets that the robot moves to during the Array Move. The order that the robot visits these targets is the order they appear in the table. After the robot has moved through all the positions, it starts over from the top.

Tap **ADD +** to select a waypoint to add to the table. From the dropdown, select a new or existing waypoint.



In the **Array Index** tab, customize the indexing behavior. The **Index Variable** field is where you choose an integer variable to decide which index the Array Move moves to when it executes. When you use an index variable, your task must update the variable with other blocks.

Choose a waypoint in the **Resolve Array Position to Waypoint** field to update that waypoint with the next array position each time the block executes.



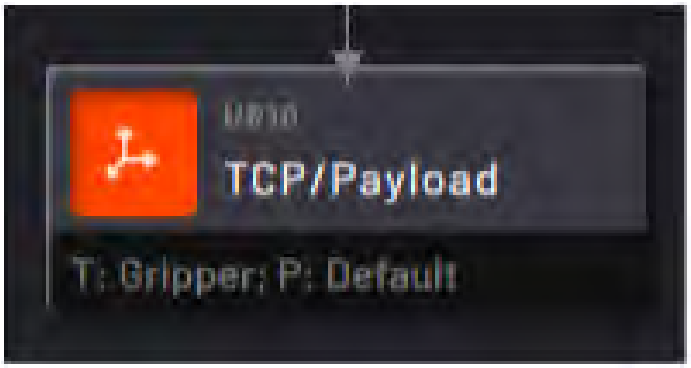
Note:
When there is a waypoint in the Resolve field, the Array block does **not** move the robot. It saves the chosen waypoint at the grid position. Then you may use this waypoint in other move blocks, like the Continuous Move. Execute the Array Move block to update the waypoint with the next position in the list.

See Robot Command Blocks *(on page 187)* for more info on the other block features.

Set TCP/Payload Block

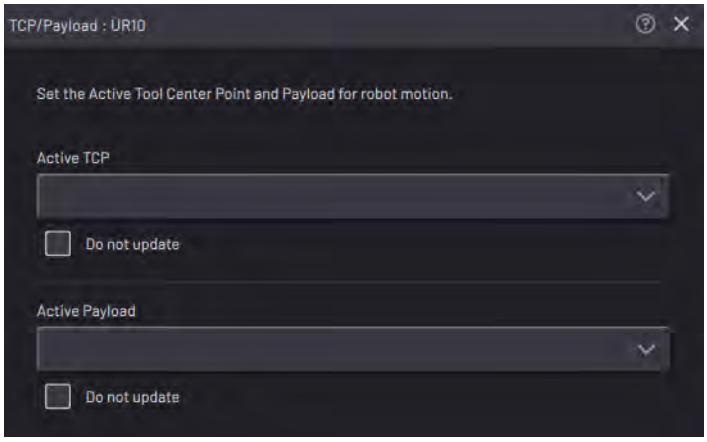
The TCP/Payload block sets the robot's Active Tool Center Point (TCP) and Active Payload. While the task is running, the Active TCP and Payload stay as the values set until another TCP/Payload block is executed.

Note:
First add the TCPs and Payloads to a robot's configuration in Device Configuration.

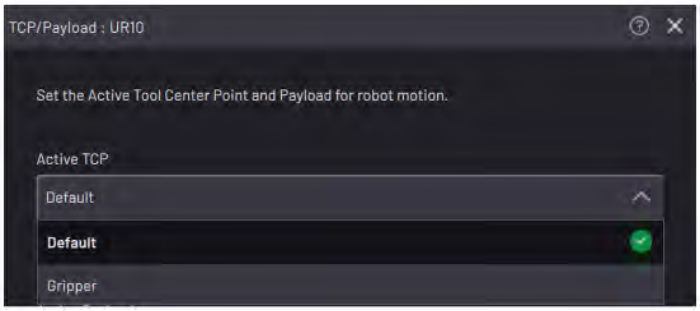


When you add a TCP/Payload block, choose whether or not to change the Active TCP and/or Active Payload.

Check the box labeled "**Do not update**" to maintain the active TCP or Payload at the time when the block executes.



Use the dropdown menus to select the configured TCP and/or Payload you want the block to make active.



Important:
When executing waypoint motion blocks, the **Active TCP** on the robot should match the TCP defined for that waypoint. If not, the position the robot moves to won't match the expected position. Use the TCP/Payload block in a task to update the Active TCP before moving to that waypoint.

Important:
The **Active Payload** instructs the robot to account for the amount of inertia at the end-of-arm when it is moving. A mismatch between the expected payload (Active Payload) and actual payload at the end-of-arm can cause errors.

Use the TCP/Payload block in a task to update the Active Payload when the actual payload at the end-of-arm changes. For example, change the payload after picking up or putting down a part or a tool.

Forge/Ctrl Blocks

The Forge/Ctrl includes an internal PLC for Digital I/O control. The Check and Set blocks are the same as the Canvas Check and Set blocks, but the device is filtered to the Forge/Ctrl.

Check

See *Canvas Blocks > Check (on page 170)*.

Set

See *Canvas Blocks > Set (on page 176)*.

Network I/O Device Blocks

Add network-based Fieldbus devices to check and set I/O device states. These blocks work the same as Canvas Check and Set blocks, but the device is filtered to the Fieldbus device.

Check

See *Canvas Blocks > Check (on page 170)*.

Set

See *Canvas Blocks > Set (on page 176)*.

Serial Device Blocks

Add serial Fieldbus devices to check and set I/O device states and sensors. These blocks work the same as Canvas Check and Set blocks, but the device is filtered to the serial device.

Check

See *Canvas Blocks > Check (on page 170)*.

Set

See *Canvas Blocks > Set (on page 176)*.

Smart Gripper Blocks

The Smart Gripper blocks include **Close Gripper**, **Open Gripper**, **Check Part Present**, and **Granular Control**.

The **Close Gripper** block closes the smart gripper fully. The **Open Gripper** block opens the gripper fully. In each block, the **Wait for gripper to finish** checkbox adds a delay so the gripper can fully open or close before moving on.

Note:
Make sure you add the proper delay to gripper in Device Configuration and check the wait box in the gripper block. Without the added delay, the task could move on or finish before the gripper completes its action.

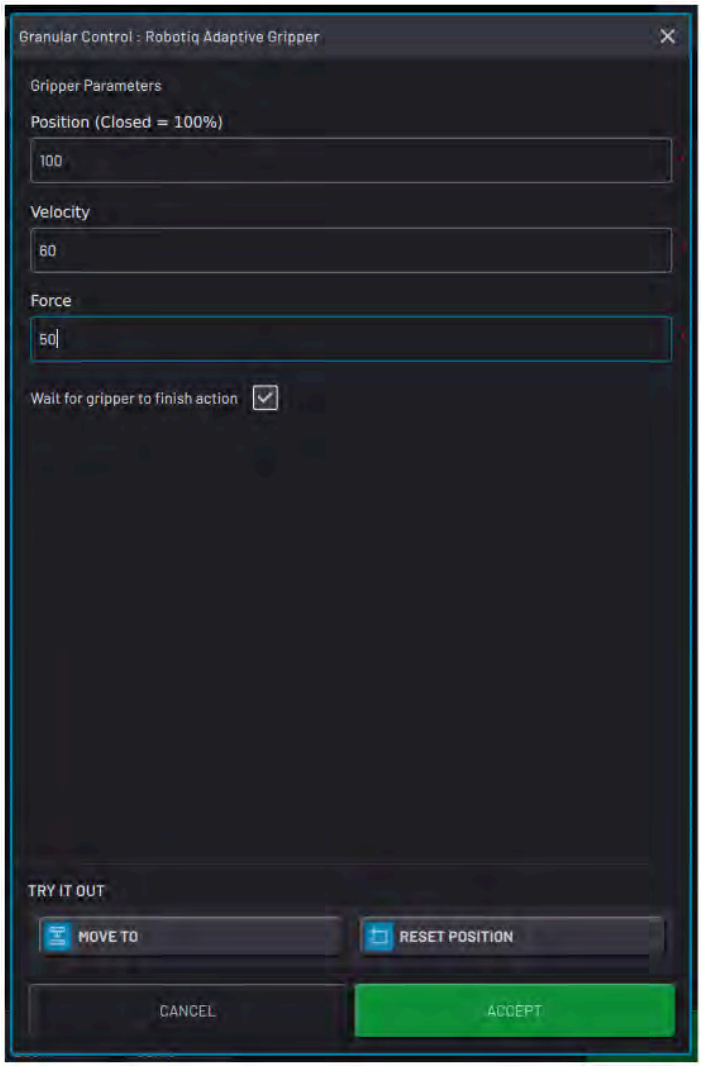
The **Check Part Present** block acts as a Check block, but you don't have to edit any parameters. It checks the force exerted by the gripper to see if it is holding a part.

- If a part is present, the task continues on the bottom path.
- If a part is not present, the task continues on the right path.

The **Granular Control** block sets the gripper to a specified position with your chosen velocity and force. To create a Granular Control block, enter a number between 0-100 (%) in the **Position** field. Then enter numbers between 0-100 (%) in the **Velocity** and **Force** fields.

Select the **Wait for gripper to finish action** checkbox to make sure the task doesn't move on until after the gripper moves to the position you chose.

Use the **TRY IT OUT** section to preview the action. Tap **MOVE TO** to move the gripper to the chosen position. Tap **RESET POSITION** to move the gripper to the position it was in when you opened the block.



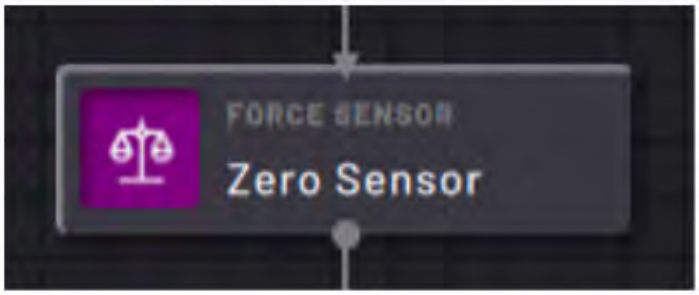
Force Sensor Blocks

The Force Sensor blocks include Check and Zero Sensor. The Check block is the same as the Canvas Check block, but the device is filtered to the Force Sensor. See *Canvas Blocks > Check (on page 170)*.

Note:
To use force in a move block, add a force sensor to the robot in Device Configuration, then find the **Force** tab in a move block (see Robot Command Blocks *(on page 187)*).

Zero Sensor Block

The Zero Sensor block zeroes or biases the sensor when it executes. There are no editable settings.



Clamping Gripper Command Blocks

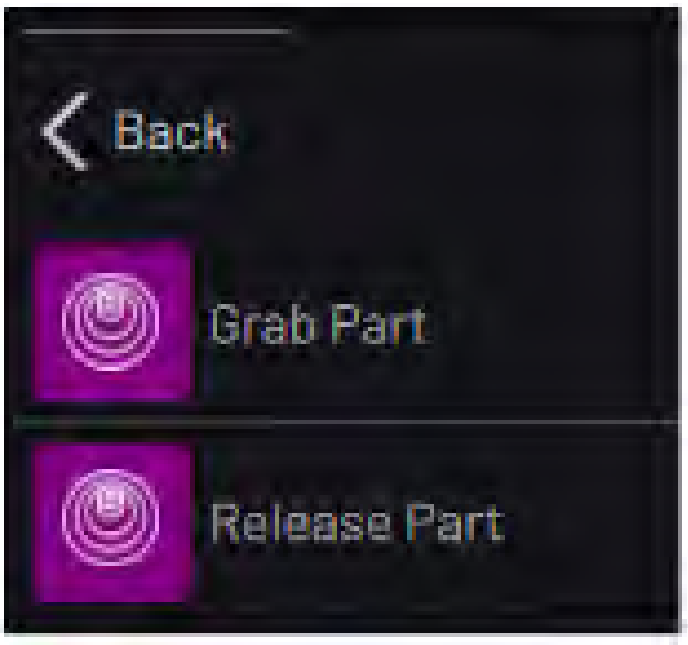
The Clamping Gripper blocks include **Close Gripper**, **Open Gripper**, and **Relax Gripper**.

Enable delay to wait for the time specified in the device configuration before the task moves on.

Note:
Make sure you add the proper delay to gripper in Device Configuration and check the wait box in the gripper block. Without the added delay, the task could move on or finish before the gripper completes its action.

Lifting Gripper Command Blocks

The Lifting Gripper blocks include **Grab Part** and **Release Part**.

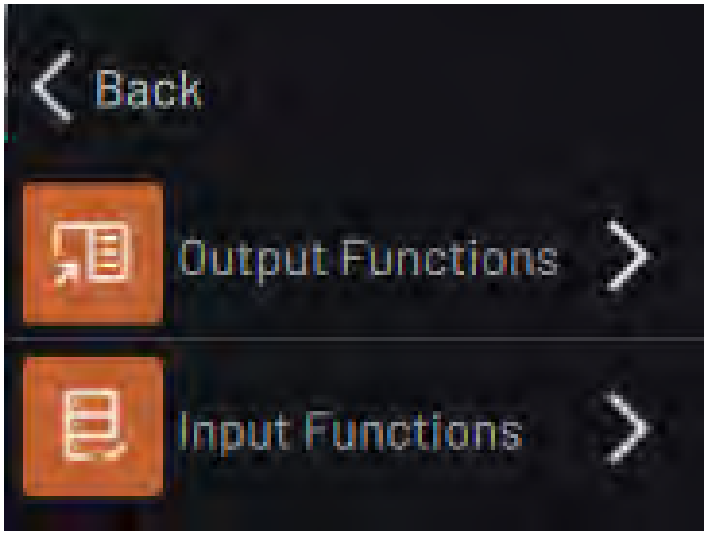


Enable delay to wait for the time specified in the device configuration before the task moves on.

Note:
Make sure you add the proper delay to gripper in Device Configuration and check the wait box in the gripper block. Without the added delay, the task could move on or finish before the gripper completes its action.

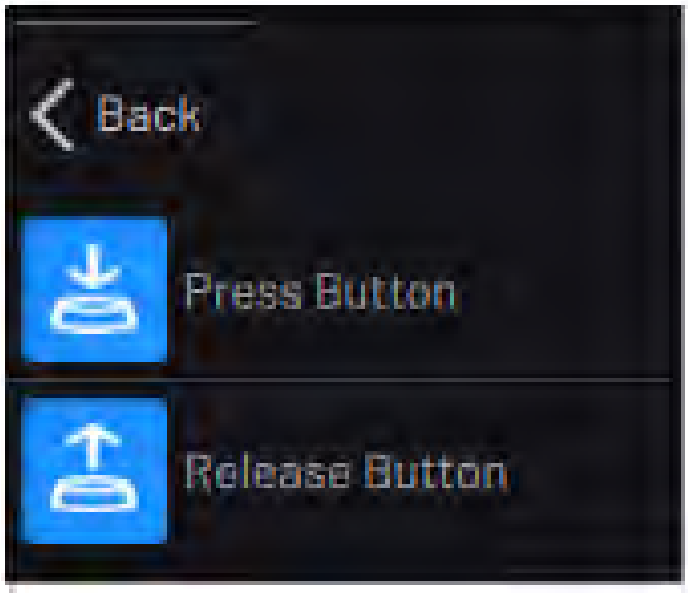
Generic IO Device Blocks

The Generic IO Device blocks include menus for calling that device's **Output Functions** and **Input Functions**.



Button Presser Blocks

The Button Presser blocks include **Press Button** and **Release Button**.

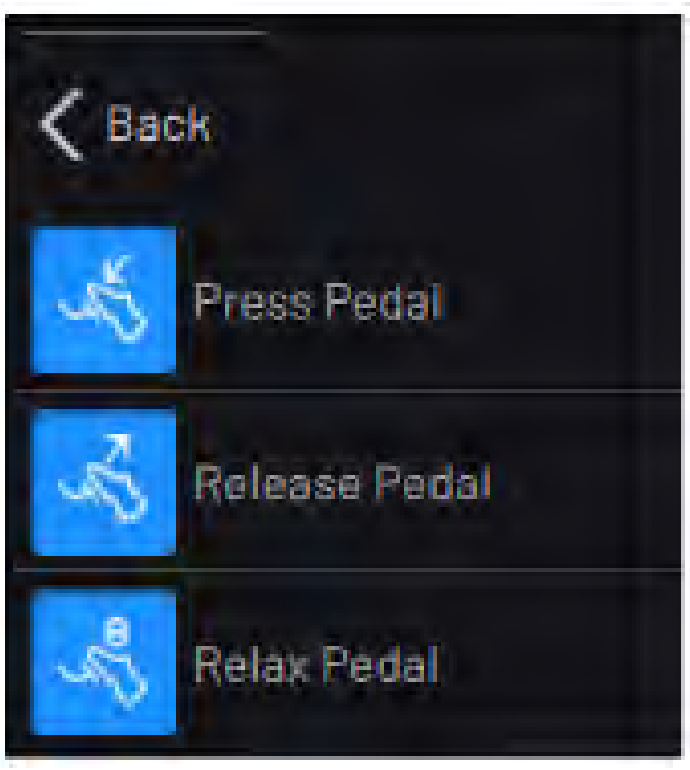


Enable delay to wait for the time specified in the device configuration before the task moves on.

Note:
Make sure you add the proper delay to gripper in Device Configuration and check the wait box in the gripper block. Without the added delay, the task could move on or finish before the gripper completes its action.

Pedal Presser Blocks

The Pedal Presser blocks include **Press Pedal, Release Pedal**, and **Relax Pedal**.



Enable delay to wait for the time specified in the device configuration before the task moves on.

Note:
Make sure you add the proper delay to gripper in Device Configuration and check the wait box in the gripper block. Without the added delay, the task could move on or finish before the gripper completes its action.

Chapter 11. Contacting READY

Visit READY.academy for *FREE* hands-on courses to help you deploy a robotic system.

Visit our **Support** site for robot startup guides, FAQs, and more.

If you encounter a problem and need to talk to someone, reach out to us.

- Email READY Robotics: support@ready-robotics.com
- Call READY Robotics: +1-833-732-3977