KINOVA

TOGETHER IN ROBOTICS

MEET LINK6

Contents

Link 6 overview

The Link 6 is an *arm* with six *degrees of freedom (DoF)*, a powerful *controller*, and a Kinova *teach pendant*. It works under the Kortex software framework and *Kortex Web App*. Kortex Web App works on the teach pendant and from a web browser on a remote computer.

Each axis is a joint and the part of the arm between each joint is a link. The robot parts from base to wrist are referred to as the arm. The arm has six parts. The remaining parts of the robot are the controller and the teach pendant.

Figure 1: Link 6

1 Teach pendant
2 Wrist
3 Forearm
4 Elbow
5 Arm
6 Shoulder
7 Base
8 Controller

The robot can be manipulated by one hand. The wrist has a set of buttons on it that are used for toggling between *Cartesian* and Joint Hand Guiding, waypoint capturing, moving the arm, and *end effector* actions. With these buttons on the wrist, it is easy to teach the robot the movements it needs to learn.

Thanks to the modular design of the robot, a variety of optional interfaces are available to connect end effectors. Each of these optional interfaces has no visible cables. Also, Kinova offers a variety of optional parts.

- Kinova wall mount kit attaches the controller to walls.
- Graphics card kit is required, in some cases, for cameras and the vision module.
- Robotiq Adapter attaches directly to the wrist and is compatible with multiple tools by Robotiq.
- Gigabit Ethernet Adapter *flange* attaches directly to the wrist and is compatible with third-party Vision adapters.
- The flange attaches to the OnRobot ISO adapter, as well as the OnRobot RG2 Gripper.
- *Emergency stop* attaches to the controller.
- One or more *enabling devices* attach to the controller.

The end of the arm has a force torque sensor built into it. With the force torque sensor, hand guiding Link 6 is smooth.

At the end of the arm is an end effector. The end effector may be a gripper, polishing disk, nail gun, screwdriver, tool changer, and so on. It is the only part of the robot that is in direct contact with the piece being manipulated.

The controller can be considered the brain of the robot. It is stored on an accessible location on the floor, table, wall, or in a cabinet; store it wherever it makes it easy to power on or off the robot. Some installations of the controller require the optional Kinova wall mount kit.

Related topics

Components on page 12

Accessories on page 333

Intended use

Link 6 is intended for typical industrial applications to manipulate objects and tools in a non-explosive environment.

To ensure safe operation for all intended uses, perform a risk assessment and mitigate all the risks to a safe, acceptable level. Use the robot in industrial applications only and within the stated specifications.

There are many different types of industrial applications in which Link 6 can be used.

- Material handling
- Small parts assembly
- Machine tending
- Packaging
- Palletizing
- Bin picking
- Screw driving

Important: The list of industrial applications is not exhaustive.

When deciding how you can use the robot, keep in mind that the arm, controller, and teach pendant of Link 6 is designed to have an ingress protection rating of 54 (IP54). The first number indicates the level of protection from objects, such as dust; the second number indicates the level of protection from water. Therefore, an IP rating of 54 indicates two things about Link 6.

- The robot is designed to have limited protection against dust and other small objects. It is limited because some small particles may still enter the robot. However, the robot can still operate with some particles in it.
- The robot is designed to be protected from splashes of water from all directions.

Related topics
Safety on page 47

Unintended uses

Link 6 is not intended to work in certain applications because it can damage the robot system, or lead to serious injury to, or death of, the user.

Kinova is not responsible for any damage caused to the robot system when the robot is not used as intended.

There are many different types of industrial applications in which Link 6 cannot be used.

- Flammable environments
- Radioactive environments
- Liquid environments, including water
- Outdoor climate conditions
- Medical applications
- Life critical applications

There also are a number of uses that can damage the robot and cause injury or death.

- Using the robot without first performing a risk assessment
- Climbing on the robot while it is in its box or on the robot when it is not in its box
- Standing on the controller
- Placing liquids, such as cups of coffee or tea, directly on the controller

DANGER: Do not use the robot in conditions that do not conform with the ingress protection rating of IP54.

Important: The lists of unintended uses is not exhaustive. There are many more unintended uses that are not cited.

Important: Link 6 does not support Power and Force Limiting (PFL) collaborative mode as described by ISO TS 15066.

Notice: Kinova assumes no liability of any kind if safety measures are not followed. Please read all the information concerning this product before using it.

Warranty information

Subject to the terms of this clause, Kinova warrants to the End User that the Products are free from defects in materials and workmanship that materially affect their performance for a period of twelve (12) months from the date Kinova ships the Products to the End User, known as the "Delivery Date".

Kinova agrees to repair or replace, at Kinova's discretion, all Products that fail to conform to the relevant warranty provided that:

1. notification of the defect is received by Kinova within the specified warranty period;
2. the allegedly defective Products are returned to Kinova, at the End User's expense, with prior authorization from Kinova within thirty (30) days of the defect becoming apparent;
3. the Products have not been altered, modified, or subject to misuse, incorrect installation or maintenance, neglect, accident, or damage caused by excessive current or having been used with incompatible parts;
4. the End User is not in default under any of its obligations under this Agreement;
5. replaced Products must have the benefit of the applicable warranty for the remainder of the applicable warranty period.

If Kinova diligently repairs or replaces the Products in accordance with this section, it will be deemed to have no further liability for a breach of the relevant warranty.

Allegedly defective Products returned to Kinova in accordance with this contract will be returned to the End User if found by Kinova on examination not to be defective. Kinova may charge a fee for examination and testing. The warranty cannot be assigned or transferred, and is to the sole benefit of the End User. Where the Products have been manufactured and supplied to Kinova by a third party, any warranty granted to Kinova in respect of the Products may be passed on to the End User. Kinova is entitled in its absolute discretion to refund the price of the defective Products in the event that such price has already been paid.

Applicable firmware

The current version of the guide reflects a specific version of the firmware used in Link 6.

Whenever there is a change in the firmware, there usually is a change in the user guide. Always make sure the guide is matching the installed firmware.

If you are using an earlier firmware or software version, some of the features discussed in the document may not apply to your installation. To upgrade firmware and software, download the packages from the product technical resource page on the Kinova website and install them from the Kortex Web App **Upgrade** page.

Table 1: Firmware associated with the guide r2.0

Software	Version
Firmware	3.2.0

Technical specifications

The Link 6 is a modern, robust, collaborative-capable robot for all industries that want to benefit from automation.

Table 2: General

Degrees of freedom (DoF)	6
Continuous payload **(full range, at 200 mm from wrist, for optimal performance)**	6.0 kg
Maximum semi-dextrous reach	1000 mm
Maximum Cartesian translation speed	1.85 m/s
Repeatability according to ISO 9283 specification	up to 0.1 mm
Power supply voltage and frequency	110-240 VAC, 50-60 Hz
Recommended overcurrent protection device rating	15 A at 120 VAC
	7.5 A at 240 VAC
	SCCR 5 kA
	opens in less than 400 ms
Ingress protection rating (arm and controller)	IP54
Ambient operating temperature	0 °C to 40 °C
Maximum storage and operating humidity levels (non-condensing)	15% to 90%
Arm weight (with 5 m cable)	23.45 kg
Arm materials	Aluminum, ABS plastic
Installation orientation	All
Acoustic noise level	70 dBA
Robot base footprint diameter	149 mm
Lifetime	35000 hrs

Table 3: Integrated force torque sensor

Range force/torque Fx, Fy, Fz, Tx, Ty, Tz	200 N / 20 N·m
Repeatability force/torque Fx, Fy, Fz, Tx, Ty, Tz	0.5 N / 0.05 N·m

Table 4: Actuators

Joint 1, 2, 3 range	± 357°, 200 °/s
Joint 4, 5, 6 range	± 357°, 320 °/s
Actuators sensors	Redundant position, current, voltage and estimated torque

Table 5: Interfaces

Controller interfaces	USB 3.0, Ethernet
Controller I/O	Digital: 16 inputs (8 redundant safe), 16 outputs (8 redundant safe)
	General purpose analog: 4 programmable input/output
	General purpose digital: 8 inputs, 8 outputs
Wrist I/O	Digital: 8 programmable input/output
	Analog: 2 programmable input/output
	Modbus RTU, 24 V supply @ 4 A
Wrist - Mechanical	Flange ISO 9409-1-50-4-M6

Table 6: Controller

Controller dimensions (without covers, feet, and protrusions)	454 mm x 208 mm x 395 mm
Controller dimensions (with covers; without feet and protrusions)	
Controller weight (approx.)	17.6 kg
Processor	Intel® Core™ i7 processor[1]
Optional GPU	Up to NVIDIA RTX™ 3090
Memory	RAM: 8 GB DDR4
	Storage: 128 GB SSD

Table 7: Teach pendant

General	12.1”, touch capacitive, IP54
Weight (approx.)	2 kg

1 Intel, the Intel logo, and Intel Core are trademarks of Intel Corporation or its subsidiaries.

Table 8: Safety

Notice: All safety specifications are currently in progress.

Certification	Developed according to UL1740, as well as ISO 10218-1:2011, and EU directives
Safety functions	All safety functions *PLd* Cat3 (developed according to ISO 13849-1/-2:2015)

Table 9: Main accessories (optional parts)

Tool adapters with mounting hardware	Robotiq, OnRobot
	0.2 kg
E-stop button	0.2 kg

Related topics

Accessories on page 333

Schematics and dimensions

The dimensions of the *arm* and the *controller* are important for integration purposes. The arm must be placed such that it can perform the desired task safely.

Remember: Add the height of the *end effector* and its required workspace to Link 6, as well as the height of the surface on which the base is installed, for an accurate assessment of the amount of space required to install a robot.

Each joint can turn as much as +360° and -360°. Trying to turn more than that when manually *backdriving* the robot can break the internal wiring. The maximum amount each joint can turn can be modified on the **Joint Limits** in *Kortex Web App*.

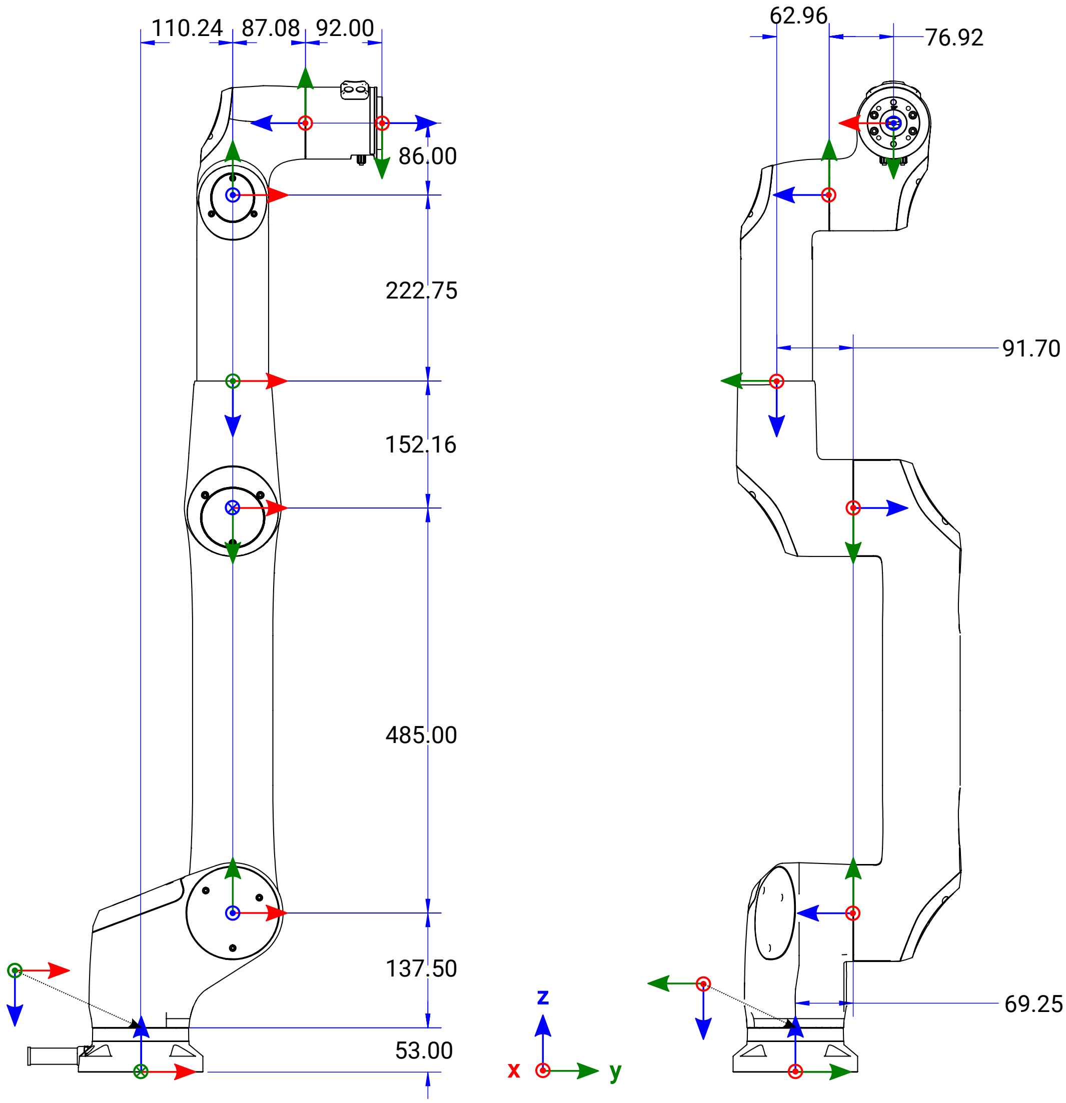


Figure 2: Arm dimensions in millimeters

The controller is a box that sits on a table, floor, wall, or in a cabinet near the arm.

Note: To install the controller in a cabinet or on a wall, use the Kinova wall mount. The wall mount is an optional accessory.

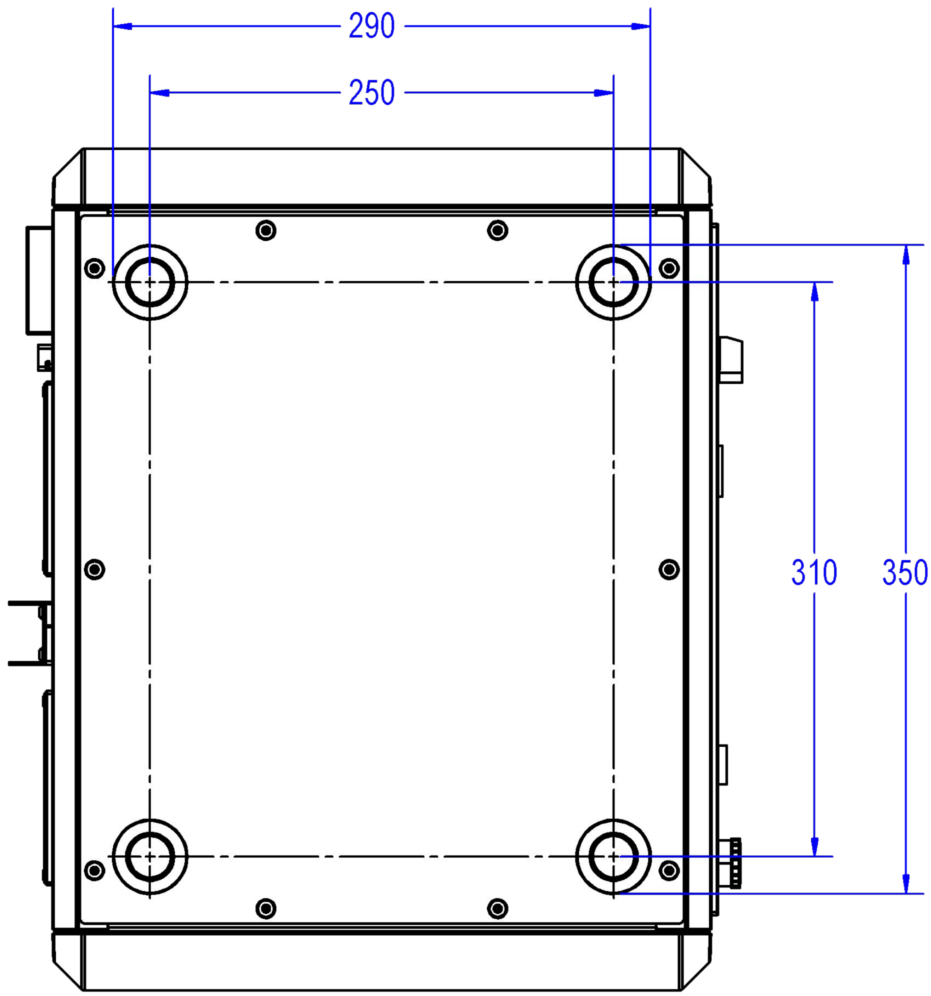


Figure 3: Controller dimensions in millimeters: Bottom view

Related topics

Safety directives and warnings on page 48

Move robot arm without power on page 54

Joint limits on page 73

Controller wall mount

Effective workspace

Link 6 requires enough space to work optimally without interference from other equipment and people when it is being integrated. The workspace is limited when the robot is used in a restricted environment by using protection zones in the *Kortex Web App*.

The robot uses an three-dimensional (3D) oval space. Take the capsule-shaped aspect of the recommended reach and maximum working area into account when integrating the robot so that Link 6 is used for optimal performance.

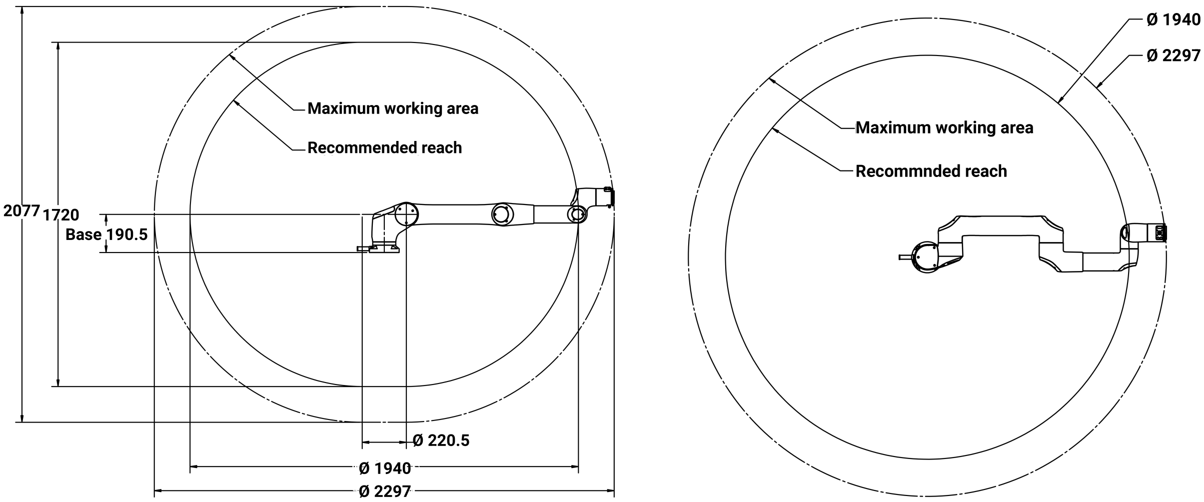


Figure 4: Recommended reach and maximum working area, vertical and horizontal

Related topics

Protection zones on page 77

Components

Each component of Link 6 can be considered as modular parts of the robot; each one has its specific role to play in the overall functionality.

There are several main components that make the robotic arm.

- *Base*
- *Actuators*
- *Wrist*
- *Controller*
- *Teach pendant*
- *Industrial I/O panel*
- *Wrist I/O*

Related topics
Link 6 overview on page 1

Base

The *base* is mainly a mounting interface for the *arm*. It also is an electrical and mechanical interface for the first *actuator*.

The base has a cable that comes out of it; the cable is used to connect it to the *controller*. The minimum *bending radius* of the cable is 70 mm.

Attention: Never disconnect the cable between the controller and the arm when the controller is switched on.

Figure 5: The base component

Specifications and dimensions of the base of the arm

The *base* of the *arm* is designed to take into account the most common base sizes in industries, which facilitates swapping robotic arms.

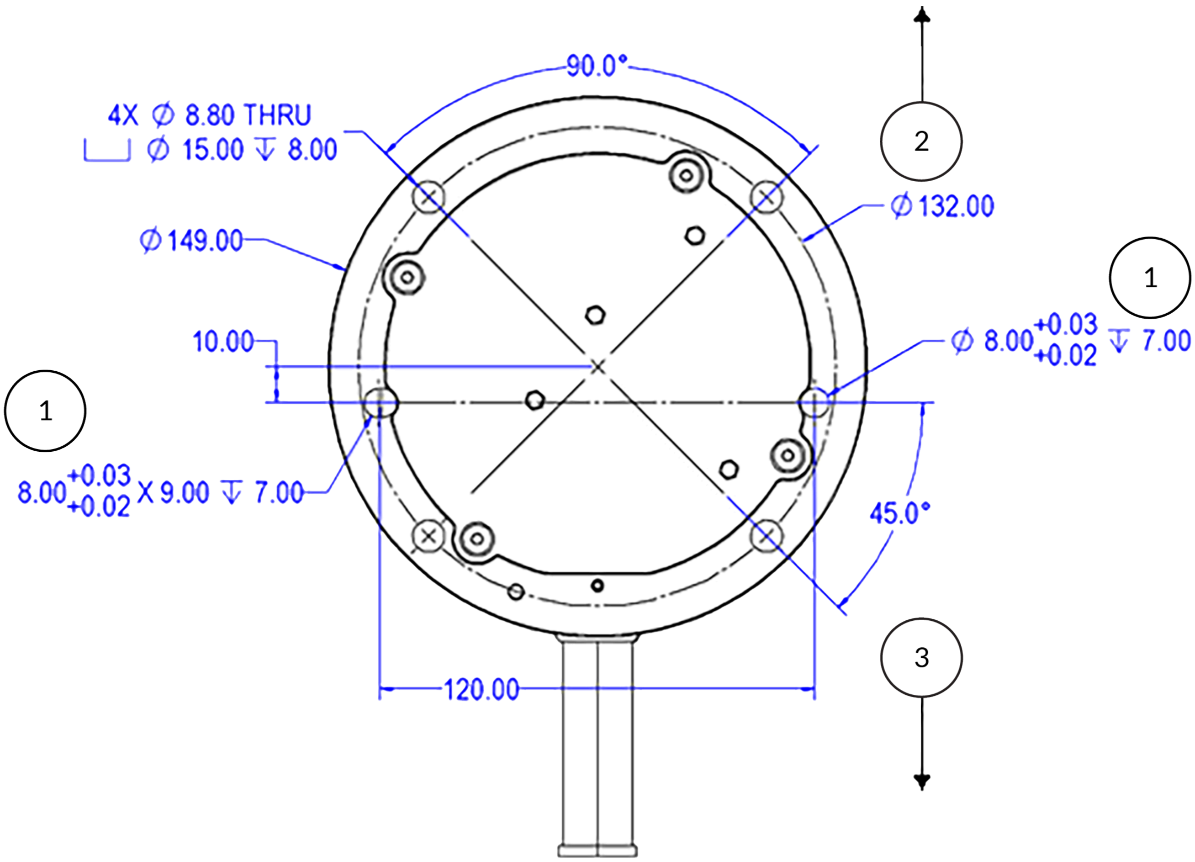


Figure 6: Top view of base dimensions in millimeters

1	Dowel hole feature for optional dowel pins
2	Front
3	Back

Related topics

Installing the arm directly to a surface on page 131

Installing the arm on a third-party adapter plate on page 133

Actuators

Link 6 has six *actuators*. The actuators come in two sizes. Three are large; three are small.

The Size 110 actuators have more load capacity than the Size 80 actuators. The Size 110 actuators are in the first three joints, when counting joints from the *base* upward. The Size 80 actuators have more speed capacity than the Size 110 actuators. All actuators include dual *microcontrollers* and various sensors to satisfy the Safety architecture category 3, *performance level D* monitoring architecture.

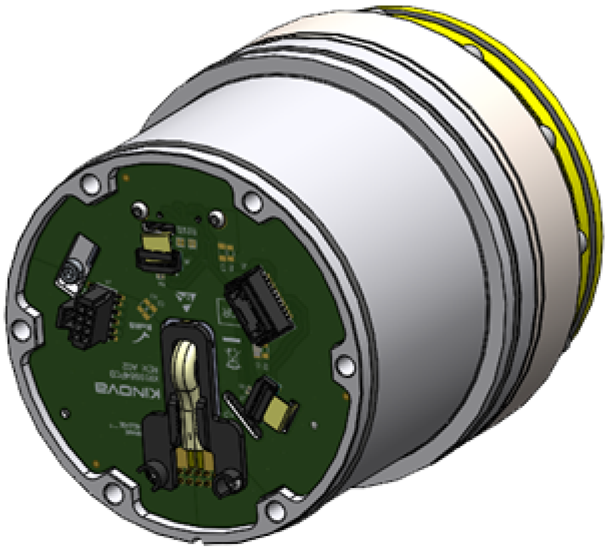


Figure 7: Size 110 actuator

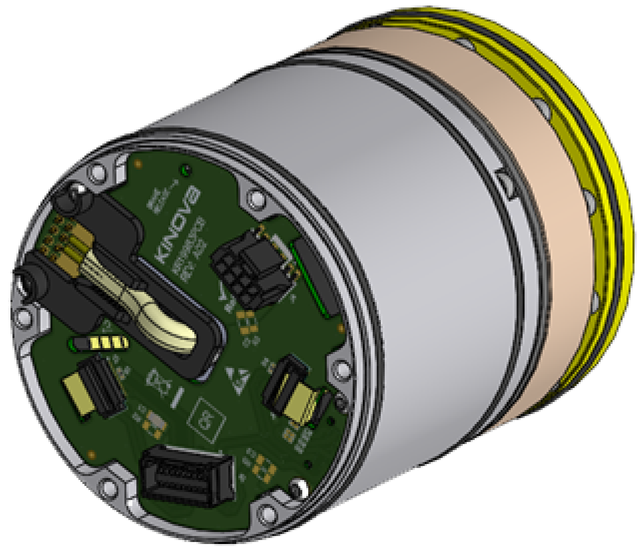


Figure 8: Size 80 actuator

Each actuator also has a mechanical brake that is activated from the *firmware* or when the power is removed from the arm.

Wrist interface overview

The *wrist* is a modular component to which a variety of *end effectors* can be attached.

The wrist has buttons for the user to control the robot manually, buttons for the user to interact with *Kortex Web App*, and LED status indicators on the *wrist ring* to indicate the current state of the robot. The wrist also has a built-in force torque sensor. The sensor is used in different applications, such as *Cartesian* Hand Guiding. The sensor ensures the correct amount of force and torque is used with the tool.

The wrist interface is made of eight main parts, one of which is not visible.

hidden	Built-in force torque sensor
2	Buttons
3	*Wrist ring*
4	*Flange* interface
	ISO 9409-1-50-4-M6
5	0° and 180° dowel pin positions

Figure 9: Top of wrist

6	Connector A
	M8, 8 pin male connector
	1 *RS-485* for Modbus RTU and 1 digital *I/O*
	24 V
7	Connector B
	M8, 8 pin male connector
	2 analog I/O and digital I/O configuration
	24 V

Figure 10: Bottom of wrist, M8 connectors

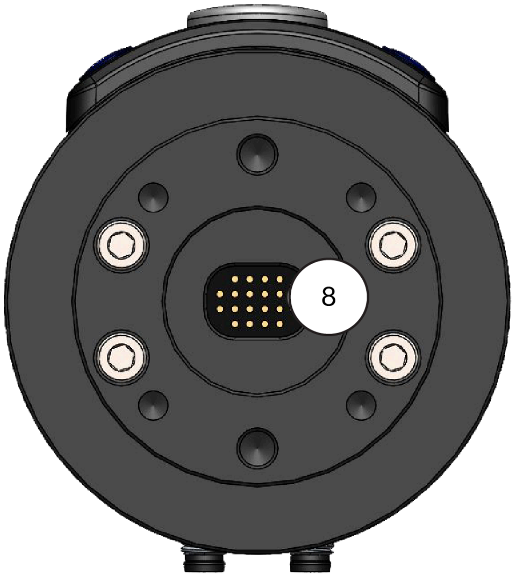


Figure 11: Bottom of wrist, pogo pin connector

8	Central pogo pin connector Gigabit Ethernet RS-485 for Modbus *RTU* 24 V power

Use the buttons on the top of the wrist to control the arm manually.

Figure 12: Buttons on the wrist

1	Hand Guiding mode Toggle button	Press to toggle between *Cartesian* and *angular* Hand Guiding modes.
2	Waypoint capture (Snapshot button)	Press to capture a snapshot of the current location of the arm when you are editing a waypoint block during visual programming.
3	*Wrist enabling device*	Press and hold while moving the arm to teach it the trajectory.
4	*End effector* actions	Press + and - as a way of using the actions of the end effectors as defined by the plugin of the end effector. For example, + can close grippers and - can open grippers.

Related topics

Modes of operation of Link 6 on page 106

Robot lights and what they mean: a reference on page 122

Force torque sensor

The force torque sensor detects the force and torque applied to the *end effector*.

The six-axis force torque sensor is used in Hand Guiding mode to provide smooth and effortless experience.

Important: Applying forces above 1000 N and torques above 100 N·m can permanently damaged the sensor.

Mechanical interface

The *wrist* is designed to connect to many different types of *end effectors*. Its interface complies to ISO 9409-1-50-4-M6.

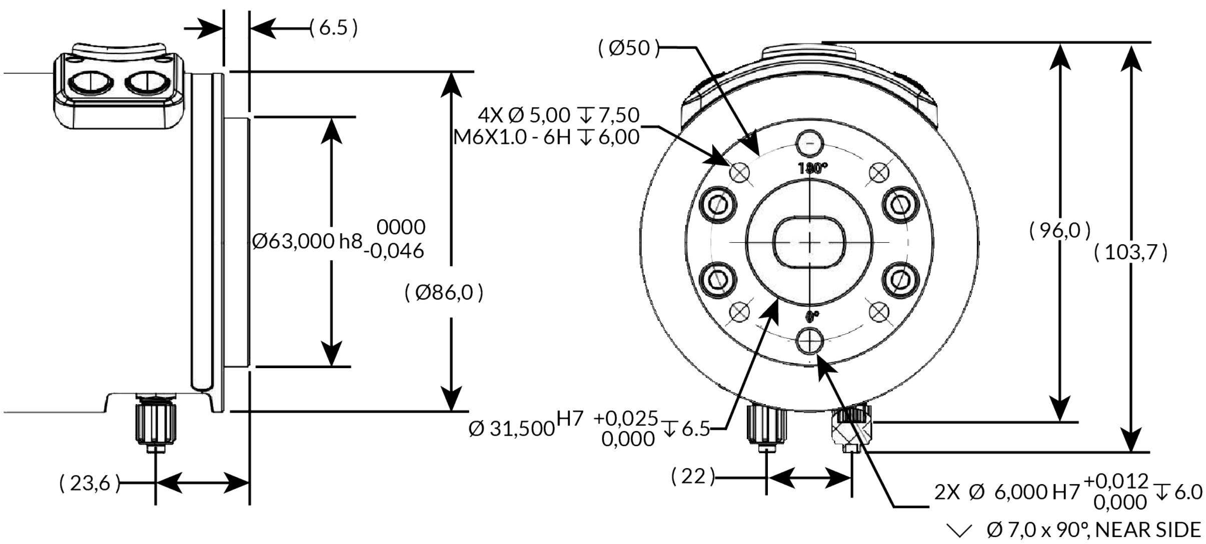


Figure 13: Mechanical interface of the wrist

Electrical interface

The *wrist* can connect electrically with *end effectors* through any combination of M8 8-pin connectors, central pogo pin, or all central pogo pins. Each of these methods is considered an interface.

Important: Accessories that connect to the wrist may or may not use the central connection of pogo pins. When the accessory does not use the central connection, keep the cap on the central pins to avoid potential damage to the pads on the *printed circuit board (PCB)*.

Figure 14: Central connector on wrist with cap on and with cap off

M8 connectors

Each of the two M8 connectors, which are compliant to IEC61131-2, have 8 pins. They provide 24 V power to whatever is attached to them. Connector A has two analog *I/Os* and four digital I/Os. Connector B has an *RS-485* interface and four digital I/Os. The digital I/Os of connector A and connector B can be configured independently so that you can have as many as eight different configurations.

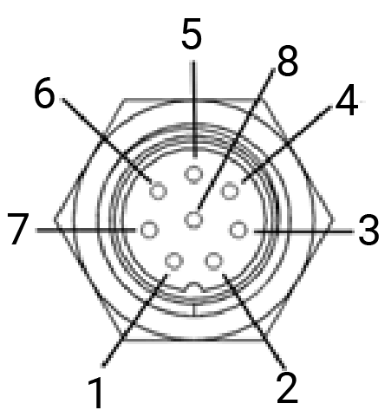


Figure 15: M8 connectors when facing the connector

Table 10: M8 Connectors

	Connector A	Connector B
Pin #	**Signal**	**Signal**
1	AIO_1	RS_485_P
2	AIO_2	RS_485_N
3	DIGITAL_IO1	DIGITAL_IO1
4	DIGITAL_IO2	DIGITAL_IO2
5	+24v	+24v
6	DIGITIAL_IO3	DIGITIAL_IO3
7	DIGITIAL_IO4	DIGITIAL_IO4
8	Ground	Ground

Remember: Always leave the caps on the M8 connectors when they are not being used.

Each analog I/O can be configured in four different ways.

Table 11: Analog interface configuration

Input or Output	Bottom Range	Upper Range	Notes
Input	0 V	10 V	
Input	4 mA	20 mA	
Output	0 V	10 V	for high impedance loads
Output	4 mA	20 mA	for high impedance loads

Each digital I/O can be configured in three different ways.

Table 12: Digital interface configurations

Input or Output	Bottom Range	Upper Range	Maximum
Input - low signal	-3 VDC	15 VDC	15 mA
Input - high signal	15 VDC	30 VDC	15 mA
Output 0	24 V	24V	600 mA

Pogo pins

In the center of the front of the wrist is an interface for pogo pins. It is supplied with 24 VDC, a dedicated Gigabit Ethernet, and *RS-485*. Each pin has an associated connector pin assignment, which can be seen in the pogo pin distribution table and figure.

Table 13: Pogo pin distribution

Pin Number	1	2	3	4	5	6
1						
2		Ground	BI_DD+	BI_DD-	Ground	
3	485_P	BI_DB-RX0-	24 V	24 V	BI_DA-TX0+	485_N
4	485_N	BI_DB-RX0+	24 V	24 V	BI_DA-TX0-	485_P
5		Ground	BI_DC-	BI_DC+	Ground	
6						

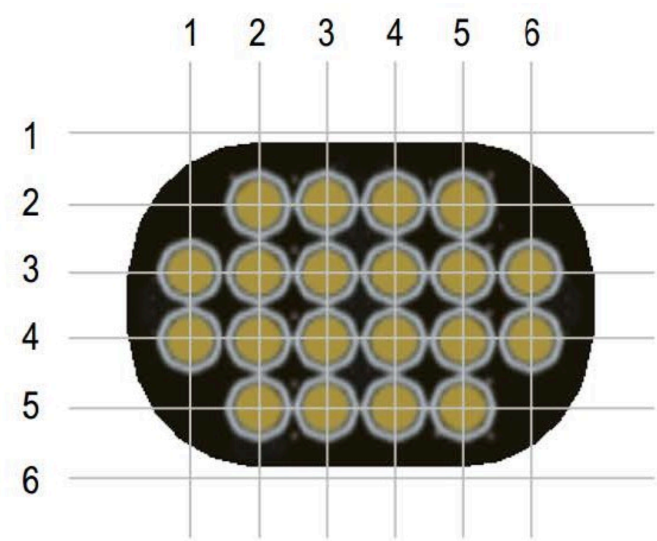


Figure 16: Pogo pin distribution

Available power in the wrist

The wrist can handle a maximum of 5 A among all M8 connectors plus central pogo pin. If one M8 connector is using 1 A, there is only 4 A left for the other interfaces.

Table 14: Power limitations

Electrical interface	Maximum continuous current
M8-8 24 V pin	1.4 A
Single pogo pin in center	2 A
All pins in center	5 A

Related topics

Wrist I/O on page 43

Wrist analog on the Industrial I/O plugin page on page 328

Wrist interface compatibility

The *wrist* must be compatible with the *end effector*, and other peripherals, attached to it.

Link 6 is compatible with a variety end effectors.

- Kinova accessories
  - Gigabit Ethernet adapter is an optional Kinova accessory that exposes an M12 X-Coded Gigabit Ethernet connector to connect easily to Ethernet devices.
  - Robotiq adapter is an optional Kinova accessory that interfaces both mechanically and electrically with Robotiq products. It has central pogo pins, which means no cables are required when attaching Robotiq products.
- ISO flange, which follows the standard ISO 9409-1-50-4-M6, interfaces with end effectors.

Related topics

Gigabit Ethernet adapter

Controller overview

The *controller* powers the *arm*. It stores all safety and non-safety parameters, the *firmware*, the trajectories, arm calibration, and so on.

Typically, the controller is connected to the *teach pendant* or an external computer that runs *Kortex Web App*. The controller sends commands to *actuators* that translate the commands into movements in the joints of the arm.

The controller can be placed on a table or floor. It also can be mounted on a wall.

The front of the controller has switches, buttons, and *I/O* ports.

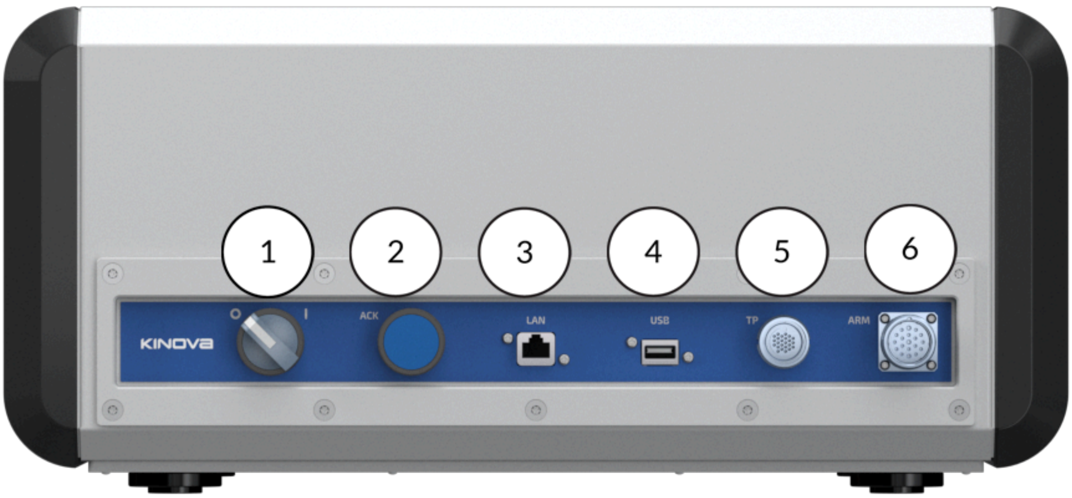


Figure 17: Front of controller and its parts

1 Power knob

A light illuminates behind the power knob when electrical power is coming through the controller. The light acts as a visual cue of the state the robot is currently in.

2 ACK button

A light illuminates in the center of the button and acts as a visual cue of the state the robot is currently in and what action must be taken.

The button can be configured in **Systems** > **Robot** in Kortex Web App such that it is mandatory to be pressed when operating in Automatic mode.

3 Ethernet connector

Use the Ethernet connector in the front of the controller to connect to a laptop or computer.

Note: Alternatively, use the Ethernet connector on the back of the controller.

4 *USB* port

Use the port to insert a *USB* key that contains data that must be pushed onto the controller, such as a new software unit package.

Use the port to insert a USB key to store files that must be retrieved from the controller, such as an exported program.

5 TP connector

Plug the teach pendant into the connector marked TP to connect to the controller.

6 ARM connector

Plug the arm into the connector marked ARM to connect it to the controller.

It also has a panel on the side to access *I/O* wiring to connect to the arm of the robot. Inside the panel, there is a map of the location of each type of input and output.

Figure 18: Location of input and output connectors, which on the side panel

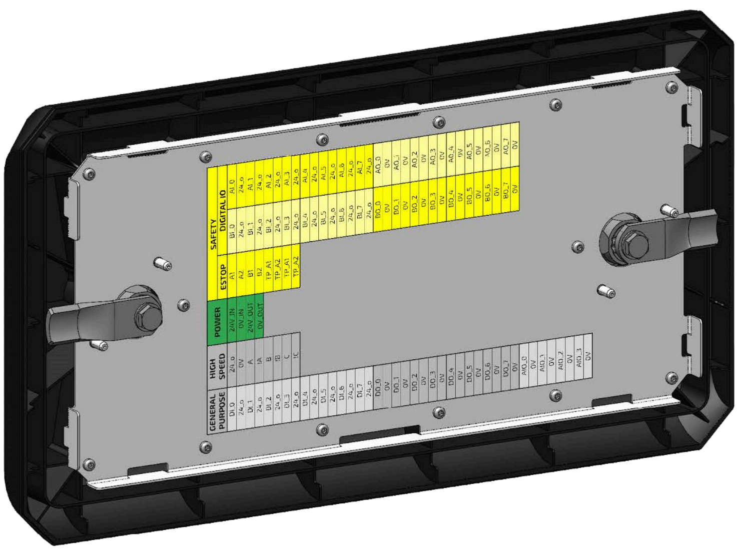


Figure 19: Map of input and output connectors is on the inside of the side panel

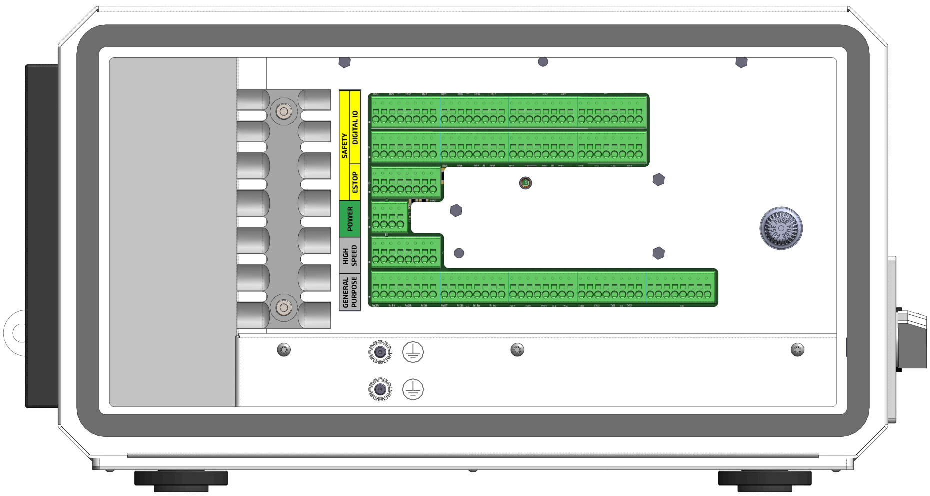


Figure 20: Input and output connectors are on the controller behind the side panel

The controller uses 12 VDC to power the internal computer units, which include *Safety Control Unit (SCU)*, *Main Processing Unit (MPU)*, *I/O* module, and optional *GPU*. It uses 48 VDC to power the arm.

Connect the controller to a power supply from its back.

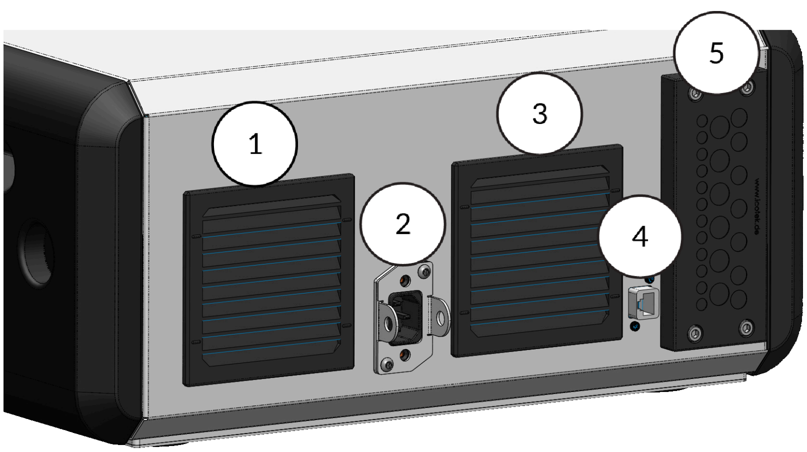


Figure 21: Back of controller

1	Fan-powered air exhaust
2	Electrical input for *alternating current (AC)* power Unplug the *AC* power cord from the controller and use the brackets around the input to lock access to the power input whenever you perform maintenance on the robot.
3	Fanless air intake
4	*RJ45* Ethernet connector

Use the Ethernet connector at the back of the controller to connect to the network.

Note: Alternatively, use the Ethernet connector on the front of the controller.

5 Entry plate for I/O connectors

Puncture holes in the entry plate for the wires that need to connect to the controller. The entry plate protects the wires from dust and liquids.

Attention: Never disconnect any cable from the controller when it is switched on.

Controller mounting interface

The *controller* can be mounted on the floor or on the wall using either the feet mounting locations or an optional wall bracket.

An appropriate location to install the controller satisfies certain conditions.

- The underside of the installation surface is accessible.
- There is enough space to hold the controller.
- There is ample space around the controller for the airflow and access.

Figure 22: Mounting points of the controller

Related topics

Considerations before installing the controller on page 134

Installing the controller on a flat surface on page 136

Surface mount of controller

Dimensions of the *controller*, along with the location of its mounting points, is important to know before the robot is installed.

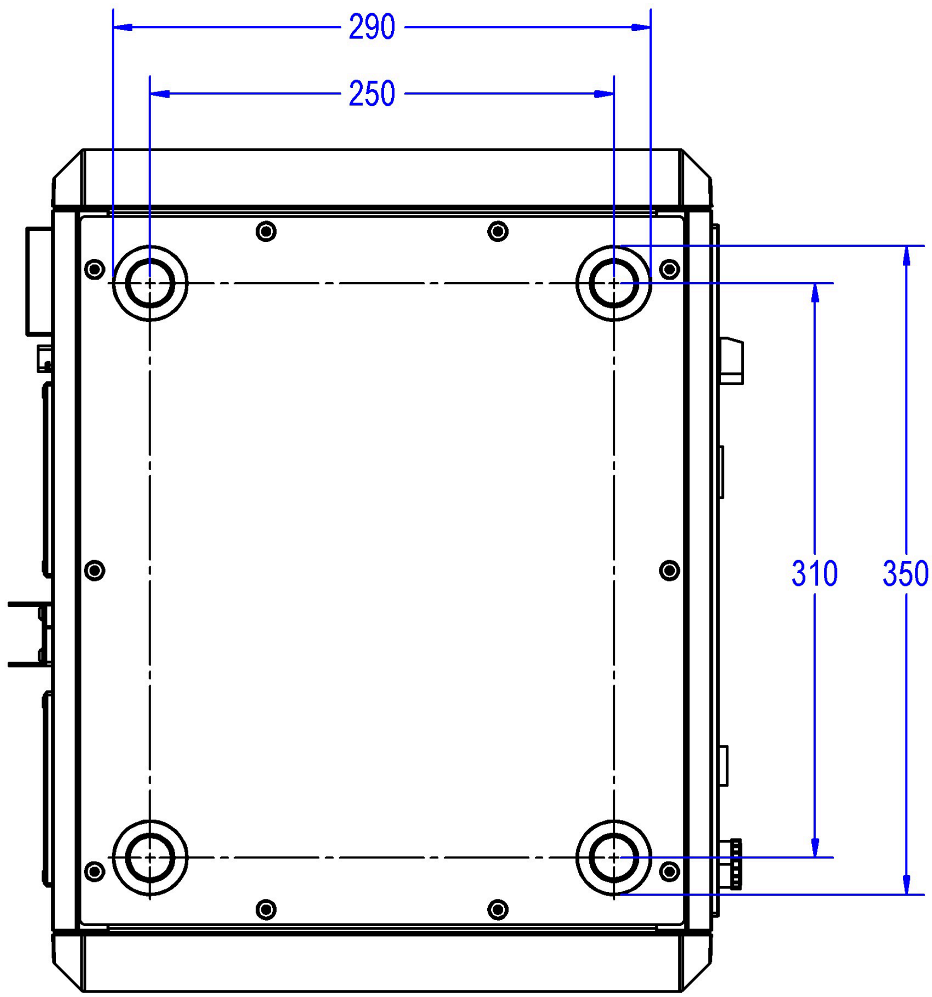


Figure 23: Mounting point measurements in millimeters

Related topics

Installing the controller on a flat surface on page 136

Controller electrical and communications interface

There is an access panel on the *controller* with a mapping of all the inputs and outputs listed directly on the back.

The access panel is on the side of the controller. It is held closed by two *camlocks*. Lock and unlock directions are indicated on the front of the side panel. A bent lip on the door aids in guiding the door on and off the rest of the box.

Figure 24: Controller side panel

1	Camlocks; Use 8 mm hex key to unlock and lock

On the inside of the panel is a mapping of the location of each input and output. See Figure 19: Map of input and output connectors is on the inside of the side panel on page 24.

GENERAL PURPOSE	HIGH SPEED	POWER	SAFETY		
			ESTOP	DIGITAL IO	
DI_0	24_o	24V_IN	A1	BI_0	AI_0
24_o	0V	0V_IN	A2	24_o	24_o
DI_1	A	24V_OUT	B1	BI_1	AI_1
24_o	!A	0V_OUT	B2	24_o	24_o
DI_2	B		TP_A1	BI_2	AI_2
24_o	!B		TP_A2	24_o	24_o
DI_3	C		TP_B1	BI_3	AI_3
24_o	!C		TP_B2	24_o	24_o
DI_4				BI_4	AI_4
24_o				24_o	24_o
DI_5				BI_5	AI_5
24_o				24_o	24_o
DI_6				BI_6	AI_6
24_o				24_o	24_o
DI_7				BI_7	AI_7
24_o				24_o	24_o
DO_0				BO_0	AO_0
0V				0V	0V
DO_1				BO_1	AO_1
0V				0V	0V
DO_2				BO_2	AO_2
0V				0V	0V
DO_3				BO_3	AO_3
0V				0V	0V
DO_4				BO_4	AO_4
0V				0V	0V
DO_5				BO_5	AO_5
0V				0V	0V
DO_6				BO_6	AO_6
0V				0V	0V
DO_7				BO_7	AO_7
0V				0V	0V
AIO_0					
0V					
AIO_1					
0V					
AIO_2					
0V					
AIO_3					
0V					

Figure 25: Inputs and outputs map on the back of the panel

I/O connections are connected to the controller through the side panel. The *protective earth* mount points and fuse are also inside the side panel of the controller.

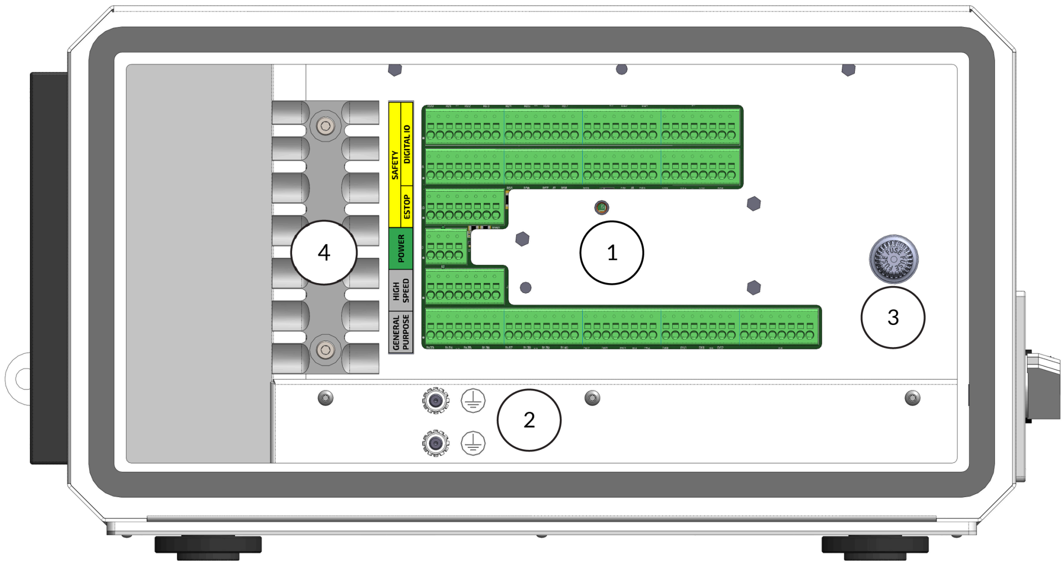


Figure 26: Inside the side panel of the controller

1 *I/Os*

From top to bottom, they are Safety Digital I/O, Safety Digital I/O, Safety ESTOP, Power, High Speed, and General Purpose

2 Protective earth stud

3 Fuse holder with a fuse held in it

4 Strain relief mounting plate, used to tie-wrap and secure cables; ensures not too much strain is applied to the terminal

Related topics

Controller digital outputs on the Industrial I/O plugin page on page 326

Controller digital inputs on the Industrial I/O plugin page on page 325

Teach pendant overview

The *teach pendant* is a hand-held device used by *integrators* to configure and program the motions of a robot and by *operators* to select a program to run.

A cable connects the teach pendant to the *controller*. The controller saves the information coming from the teach pendant.

The teach pendant is a touchscreen device with three buttons.

- *Emergency stop*
- Power
- *Pendant enabling device*

To configure trajectories and to operate the robot, use *Kortex Web App*.

Figure 27: Teach pendant front

1 *E-stop* button

Use the E-stop button to stop the robot in emergency situations.

2 Power button

Use the power button to turn on the teach pendant and access Kortex Web App.

Figure 28: Teach pendant back

3 *Pendant enabling device*

Press and hold the pendant enabling device in the middle position when you are using the Jog interface to manually move the robot or when you are running a program in Hold-to-Run mode.

4 Mounting interface

Align the mounting interface with the mounting hooks and slide the pendant on the mount.

5 Strap holding interface

Slide your hand underneath the strap such that your thumb rests on top of the strap.

CAUTION: Store all disconnected teach pendants in a safe location out of sight to prevent confusion between active and inactive emergency stops.

Attention: Never disconnect the cable from the teach pendant when the controller is switched on.

Related topics

Installing the teach pendant mount on page 138

Industrial I/O panel overview

The *controller* provides general-purpose electrical inputs and outputs from the Industrial *I/O* panel. It also provides safety-related electrical inputs and outputs from the I/O panel.

Note: Unless otherwise specified, all voltages and currents are *direct current (DC)*.

The *I/O* panel is made of two parts.

The safety *I/O*s are always powered by the internal power supply. The safety *I/O*s are divided into two banks, bank A and bank B, for redundancy. Each bank consists of eight (8) digital inputs and eight (8) digital outputs. The safety *I/O*s also contain the connections for an Emergency stop button.

The general-purpose *I/O*s are powered by 24 V IN or 0 V IN terminals. The terminals can receive their power from 24 V OUT or 0 V OUT terminals, or from an external power supply. The general-purpose I/Os contains eight (8) digital inputs, eight (8) digital outputs, four (4) analog I/Os, and provisions for high-speed encoder inputs.

The *industrial I/O panel* is designed to comply with specific standards.

- Immunity requirements for safety I/O are designed for accordance with IEC 61000-6-7.
- Safety digital inputs and general-purpose digital inputs are designed for accordance with IEC 61131-2 Type 1 and Type 3.
- Safety digital outputs and general-purpose digital outputs are designed for accordance with IEC 61131-2 Type 1 and Type 3.
- *electrostatic discharge (ESD)*, surge, and *Electrical Fast Transient(EFT)* are designed for accordance with IEC 62326-3-1.

Positive logic, which is current sinking input and current sourcing output, is supported on the panel. *Negative logic (NPN)* is not supported on the panel.

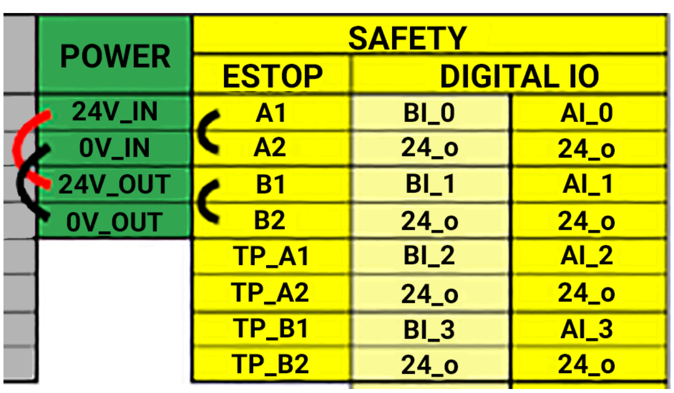


Figure 29: Default factory setting with internal supply used and external E-stop bypassed

Table 15: Industrial I/O terminals

Mechanical	Min	Typical	Max	Unit
Conductor size (*UL*/*cUL*)	16		24	*American Wire Gauge (AWG)*
Conductor size (*International Electrotechnical Commission (IEC)*)	0.2		1.5	mm^2
Stripping Length	8		10	mm
Terminal pitch		3.5		mm
Ferrule		*Deutsches Institut für Normung (DIN)* 46228-4		

Related topics

Installing the external 3-position enabling device on page 335

Industrial I/O power supply

Negative connections, labeled as 0V, are only for power and signaling. *protective earth (PE)* provides a low impedance path to ensure proper functioning in case of faults.

Be careful to connect power to the supplies correctly. When connected incorrectly, you can have undesirable side effects, such as reverse polarity, improper voltage levels, and improper frequencies. Link 6 is designed such that the 24V_IN and 0V_IN terminals are protected against reverse polarity.

Important: Use only *DC* signals.

Important: Do not exceed 28 V under any conditions.

During the power up sequence, all outputs are low and all inputs are not read.

During the power down sequence, all outputs shut down to the low state. The output from the *(E-stop)* enters a low state; it generates and *E-stop* event on all external devices connected to this terminal. All inputs are not read.

Table 16: Internal power

Internal 24 V power supply	Min	Typical	Max	Unit
Voltage	23.5	24	24.5	V

Internal 24 V power supply	Min	Typical	Max	Unit
Current (total)			2	A

Table 17: External power

External 24 V input requirements	Min	Typical	Max	Unit	Note
Voltage	20	24	28	V	Designed for IEC 61131-2
Current (total)			5	A	Fused internally at 10 A. The external power supply should be fused to a lesser value. It is recommended to fuse it at 5 A with an opening time that is less than 5 seconds.

The terminal connections belong to the POWER terminal group.

Table 18: Power terminal connections

Label	Purpose
24V_IN	24 V input to supply the general purpose *I/O*s.
0V_IN	0 V input to supply the general purpose *I/O*s.
24V_OUT	24 V output of the internal *I/O* power supply. Use only to jump with the terminal 24V_IN. If an external power supply is used to connect on the 24V_IN terminal, leave the terminal unconnected.
0V_OUT	0V output of the internal *I/O* power supply. Use only to jump with the terminal 0V_IN. If an external power supply is used to connect on the 0V_IN terminal, leave the terminal unconnected.

There are specific times when you can use the external power supply for general purpose *I/O*s.

- The total output current of the I/O panel exceeds 2 A.
- The general purpose *I/O*s must be supplied from the same power supply as the rest of the industry floor.

- The general purpose I/Os are tied to an external reference point, with up to 2000 V electrical isolation from the rest of the controller.

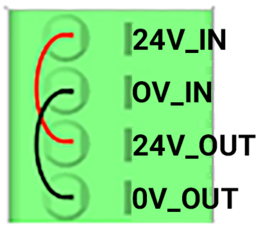


Figure 30: Internal power supply terminal connections

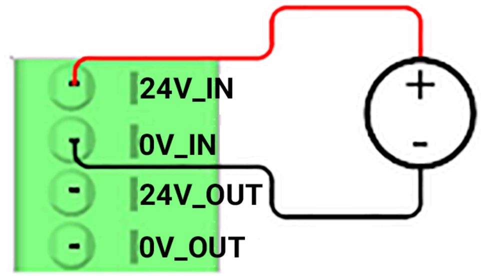


Figure 31: External power supply terminal connections

General-purpose digital inputs

There are no common points between channels. Each input has its own associated 24 V wetting port.

The panel may not work properly when there is a miswired condition coming from any other location on the panel; this type of miswired condition does not lead to any permanent damage. When the miswired condition is fixed, the panel starts working.

Important: The panel may be damaged if the miswiring comes from an external device.

Important: Miswiring issues does not apply to the 24 V power supply output.

There are isolation potentials between a channel and other circuits, including earth, as well as between channels under normal operation.

There is no hardware state indicator for general-purpose digital points.

There are side effects when you withdraw or insert an input module that is under power.

There is an additional external load when interconnecting inputs and outputs.

Table 19: Digital inputs

Digital inputs	Min	Typical	Max	Unit	Note
Low state voltage	-3	0	5	V DC	
High state voltage	11	24	30	V DC	
Current		2.0	2.6	mA	

Digital inputs	Min	Typical	Max	Unit	Note
Voltage drop					
Function					Current sinking
IEC 61131-2 Type		1, 3			Digital input type 1 and 3 designed for accordance with IEC 61131-2

Each wetting port is designed to provide the power needed for a single input point. Do not use it as a generic power source.

The terminal connections belong to the GENERAL PURPOSE terminal group.

Table 20: Digital input terminal connections

Label	Purpose
DI_n (n=[0-7]	Terminal points for digital input
24_o	Wetting port for each digital input Each input has its own wetting port.

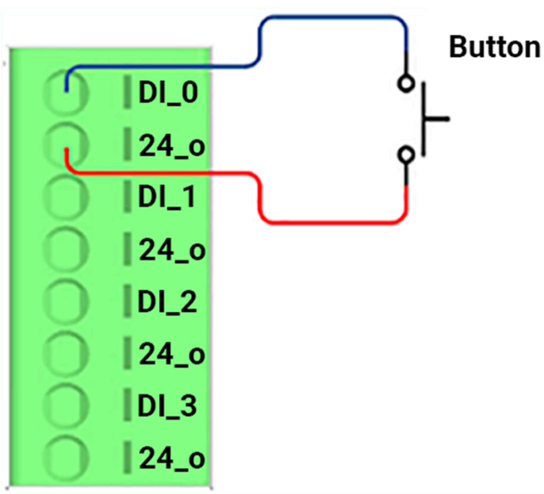


Figure 32: Digital input terminal connections

Related topics
Controller digital inputs on the Industrial I/O plugin page on page 325

General-purpose digital outputs

All 0 V ports are common, although each output has its own 0 V terminal.

All the digital outputs are short-circuit protected. An incorrect terminal connection does not lead to any permanent damage.

There are isolation potentials between a channel and other circuits, including earth, as well as between channels under normal operation.

Under normal operation, channels are not isolated from each other.

There is a monitoring point and a binary state of a visual indicator although there is no actual hardware visual indicator.

The delay time for transitions between 0 and 1 and 1 and 0 depends on the load and on software delays.

Table 21: Digital outputs

Digital outputs	Min	Typical	Max	Unit	Note
Rated current			high side mode: 500	mA	
			push-pull mode: 300	mA	
Voltage drop				V	At maximum current
Leakage current				mA	For low state
Type					Positive logic, current sourcing

Each output can drain up to a maximum of 500 mA. If many outputs are connected and are high, the maximum current available from the internal power supply could be exceeded; an external power supply must be used.

The terminal connections belong to the GENERAL PURPOSE terminal group.

Table 22: Digital output terminal connections

Label	Purpose
DO_n (n=[0-7])	Terminal points for digital output
0V	0 V input to supply the digital I/Os

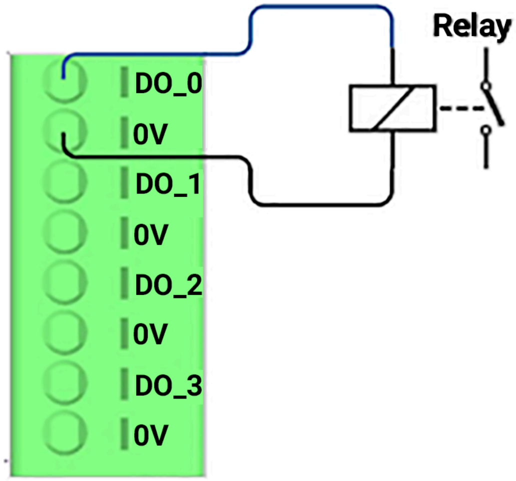


Figure 33: Digital output terminal connections

Related topics
Controller digital outputs on the Industrial I/O plugin page on page 326

Analog inputs

There are four (4) analog inputs.

Analog input types are current and voltage. The input impedance in the signal range is in the on or off state; the state is specified by the manufacturer.

Table 23: Analog inputs in current mode

Analog inputs	Min	Typical	Max	Unit	Note
Current	4		20	mA	
Voltage		24		V	
Resolution		16		Bits	

Table 24: Analog inputs in voltage mode

Analog inputs	Min	Typical	Max	Unit	Note
Voltage	0		10	V	
Current				mA	
Resolution		16		Bits	

The terminal connections belong to the GENERAL PURPOSE terminal group.

Table 25: Analog input general purpose terminal connections

Label	Purpose
AIO_n (n=[0-7]	Terminal points for analog input.
0V	0 V input to supply the analog I/Os.

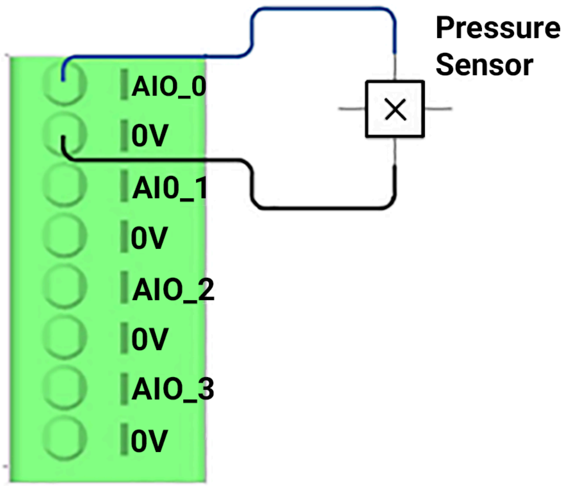


Figure 34: Analog terminal connections

Analog outputs

There are four (4) analog outputs.

Analog output types are current and voltage. The output impedance in the signal range is in the on or off state; the state is specified by the manufacturer.

The analog output has a maximum error at 25 °C ± % of full scale, with a temperature coefficient ± % of full scale/K. The maximum error over the full temperature range is ± % of the full scale.

The digital resolution is measured in number of bits.

The data format returned from the application program is in binary, *Binary Coded Decimal (BCD)*, and so on. The value of the data is in the *Least Significant Bit (LSB)*.

Table 26: Analog outputs in current mode

Analog outputs	Min	Typical	Max	Unit	Note
Current	0		10	mA	
Resistance			760	ohm	
Resolution		13		Bits	

Table 27: Analog outputs in voltage mode

Analog outputs	Min	Typical	Max	Unit	Note
Voltage	0		10	V	
Current			29	mA	
Resolution		13		Bits	

The terminal connections belong to the GENERAL PURPOSE terminal group.

Table 28: Analog output general purpose terminal connections

Label	Purpose
AIO_n (n=[0-7]	Terminal points for analog output.
0V	0 V input to supply the analog I/Os.

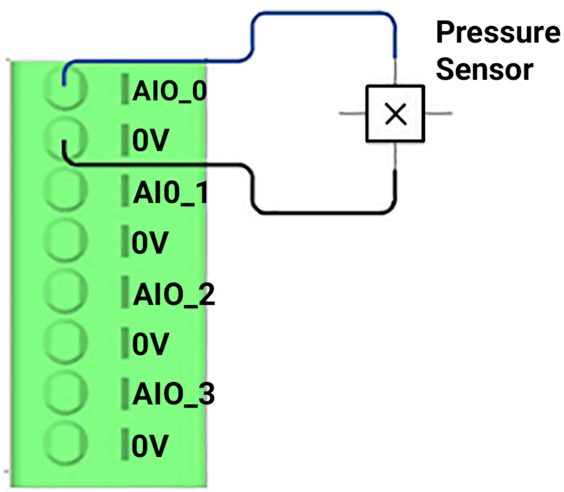


Figure 35: Analog terminal connections

Emergency stop inputs

Emergency stop devices must comply with ISO 13850:2015.

Link 6 requires an emergency stop device with two (2) *Normally Closed (NC)* redundant contacts.

The terminal connections belong to the SAFETY - ESTOP terminal group.

Table 29: Emergency stop terminal connections

Label	Purpose
AI	24 V wetting port

Label	Purpose
A2	Channel A - digital input
B1	24 V output
B2	Channel B - digital input

A typical installation involves a teach pendant that is connected on the front panel. The emergency stop is an integrated part of the teach pendant. The only thing to do is to install a jumper wire on A1, A2, B1, and B2.

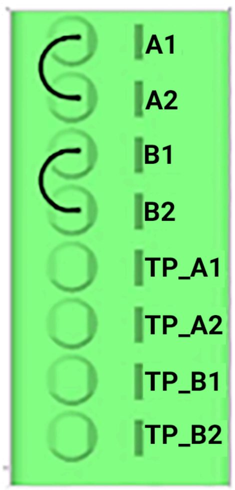


Figure 36: Terminal connections with jumper wire on A1, A2, B1, and B2 when the teach pendant is connected

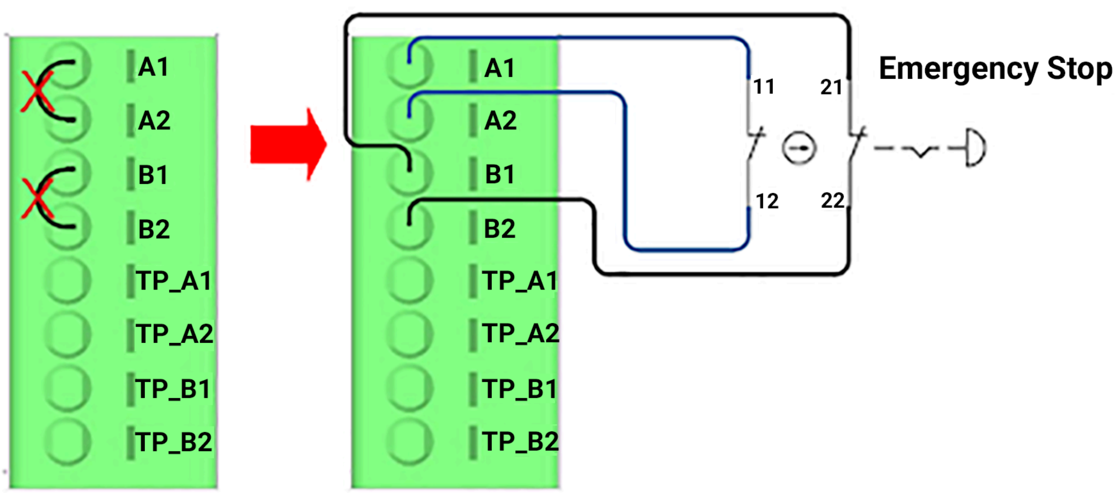


Figure 37: Terminal connections with external emergency stop connected

Teach pendant bypass inputs

There are only two ways to bypass teach pendant inputs. All other configurations prevent the robot arm from receiving power.

Important: Make sure the teach pendant is not installed when you bypass teach pendant inputs.

If a teach pendant is connected on the front panel, leave TP_A1, TP_B1, and TP_B2 open. Remove any jumper wires that are installed.

If a teach pendant is not connected on the front panel, connect TP_A1 to TP_A1 and TP_B1 to TP_B2.

The terminal connections belong to the SAFETY - ESTOP terminal group.

Table 30: Teach pendant bypass inputs terminal connections

Label	Purpose
TP_A1	24V wetting port
TP_A2	
TP_B1	24V wetting port
TP_B2	

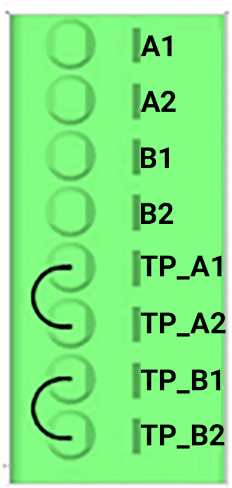


Figure 38: Teach pendant bypass inputs terminal connections

Safety-rated digital inputs

Safety-rated digital inputs can be used as redundant signals.

To use the inputs as redundant signals, connect Bank A to one of the contacts of the safety device and connect Bank B to the redundant, secondary contact of the same safety device. For example, A0 and B0 can be connected to an enabling device.

All safety-rated inputs are expected to be *NC* when both redundant contacts are in a fault state. A fault is triggered when there is a disagreement between the status of the tied redundant signals.

The terminal connections belong to the SAFETY - DIGITAL IO terminal group.

Table 31: Safety-rated digital inputs terminal connections

Label	Purpose
AI_n (n=[0-7])	Safety rated Bank A digital inputs
BI_n (n=[0-7])	Safety rated Bank B digital inputs
24_o	24V wetting port for safety rated Bank A and safety rated Bank B digital inputs

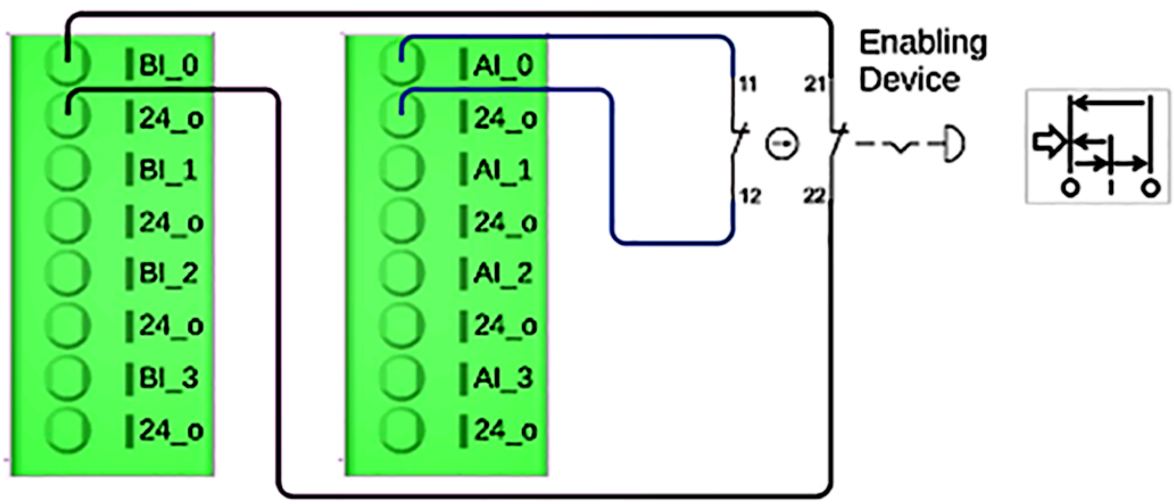


Figure 39: Safety-rated digital inputs terminal connections

Wrist I/O

Each of the two M8 connectors has eight (8) pins. The *I/O* levels associated with the connectors are designed to comply to1/ IEC 61131-2.

Connector A has two analog I/Os and four digital I/Os. Connector B has a Modbus *remote terminal unit (RTU)* with an *RS-485* interface and four (4) digital I/Os. The digital I/Os of connector A and of connector B can be configured independently.

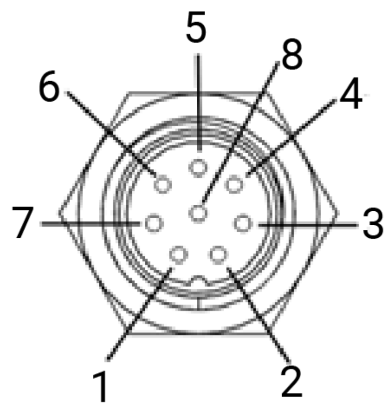


Figure 40: M8 connectors when facing the connector

Table 32: M8 Connectors

	Connector A		Connector B	
Pin #	**Label**	**Purpose**	**Label**	**Purpose**
1	AOL_1	Analog In/Out	RS_485_P	RS-485 D+ (B)
2	AOL_2	Analog In/Out	RS_485_N	RS-485 D- (A
3	DIGITAL_IO1	Digital In/Out	DIGITAL_IO5	Digital In/Out
4	DIGITAL_IO2	Digital In/Out	DIGITAL_IO6	Digital In/Out
5	+24v	Power	+24v	Power
6	DIGITAL_IO3	Digital In/Out	DIGITIAL_IO7	Digital In/Out
7	DIGITAL_IO4	Digital In/Out	DIGITIAL_IO8	Digital In/Out
8	GND	Ground	GND	Ground

Related topics

Wrist digital on the Industrial I/O plugin page on page 330

Wrist power supply

The maximum current is shared among all power outputs of the wrist, including the A and B connectors and the pogo pad.

The wrist can handle a maximum of 5 A among all connection points. If one M8 connector is using 1 A, there is only 4 A left for the other interfaces.

Important: The maximum current that can be used by any one M8 connector is 1.4 A.

General-purpose inputs and outputs for the wrist

Unless otherwise specified, all general-purpose digital and analog inputs and outputs have the same specifications as the *Industrial I/Os*.

The currents are the same as that of the digital IOs of the wrist. The currents of the push-pull are more like those of the controller IOs, but with small differences.

Table 33: Wrist IO power and current limitation

Digital outputs	Min	Typical	Max	Unit	Note
Rated current			high side mode: 500	mA	
			push-pull mode, sink, low: 100	mA	
			push-pull mode, sink, high: 300	mA	

Related topics

General-purpose digital inputs on page 35

General-purpose digital outputs on page 36

Analog inputs on page 38

Analog outputs on page 39

Wrist RS-485

RS-485 on the wrist is used only for Modbus *RTU* when it is exposed through EIA/TIA-485-A, commonly referred to as RS-485.

Important: There is no *RS-485* communication available from *Kortex Web App*.

Table 34: RS-485 pins on the wrist

Label	Purpose
RS_485_P	RS-485 D+ (B)
RS_485_N	RS-485 D- (A)

Label	Purpose
GND	RS-485 SC (G)

Table 35: RS-485 power inputs

RS-485	Min	Typical	Max	Unit
Input voltage at A/B terminal	-7	5	12	V
Current			2	A

Safety

Link 6 is a powerful robot that can perform heavy duty industrial tasks. Read and understand all safety considerations before installing and using the robot.

Safety includes general warnings about the robot, risk assessments, what to do in an emergency, how to work the robot when it loses power, and so on

It also includes all safety functions of Link 6.

Important: Any and all topics that discuss safety refer specifically to Link 6 and do not apply to the robot system or interactions with other systems or devices.

Related topics

Installing the arm directly to a surface on page 131

Installing the arm on a third-party adapter plate on page 133

Precautions

Always observe precautions to avoid serious injury to users, damage to other equipment, and damage to Link 6.

Read all safety information available in this guide, as well as diagnostic, maintenance, servicing, support, and troubleshooting topics, to perform a proper risk assessment and to know how to train users.

Use the robot as it is intended to be used.

Perform a risk assessment not only on Link 6, but on the entire *robot system*. Make sure in the risk assessment that the international safety standard for robotic systems, ISO 10218-1/-2, is respected, as well as all territory-specific standards for your area when applicable.

Perform a *Lockout Tagout (LOTO)* procedure so that no cable can be connected to the robot when it is under maintenance or any other situation where the power needs to be removed; it prevents the robot from being switched on accidentally.

Dispose of the robot in accordance with the laws and regulations of the territory in which the robot is installed.

Related topics

Intended use on page 2

Risk assessment on page 51

Maintenance on page 337

Servicing on page 343

Disposal on page 344

Safety directives and warnings

Follow basic safety guidelines when working with Link 6 to avoid injury to the user or damage of the equipment.

Familiarize yourself with what must be done to avoid the various dangers, warning, and cautions when working with the robot.

Directives specific to the integrator

The *integrator* is responsible for the safe use of Link 6. For the robot to be used safely, there are certain directives the integrator must follow.

Important: Link 6 does not support Power and Force Limiting (PFL) collaborative mode as described by ISO TS 15066. Keep this in mind during analysis and assessment phases.

- Perform a hazard analysis for the robot and the robot cell.

  Important: Hazard analysis is a mandatory activity to have a complete risk assessment.
- Perform a risk assessment based on the system, the environment, and the user before installing Link 6.
- Include any and all external equipment, objects, and any other parts of the complete robot system added by the integrator in the risk assessment.
- Include specific precautions with respect to earthquakes when the robot is installed in an earthquake zone.
- Design the robot cell so that it is optimally ergonomic.
- Install the *arm* and the *controller* such that they are easily accessible for maintenance purposes.
- Install the arm and the controller such that they do not impede on the ergonomics of the entire robotic environment.

Danger

DANGER: A danger is an impending hazardous situation that can result in serious injury or even death, or can severely damage property. Avoid impending hazardous situations that are associated with dangers.

Keep objects and hands out of *pinch point* zones. Pinch points directly on the robot are located at the shoulder, elbow, and wrist.

1	shoulder
2	elbow
3	wrist

Disconnect the power supply and perform *Lockout Tagout* procedures when maintenance tasks must be performed.

Stop the robot when a person is trapped by the robot or some other similar emergency.

1. Push the robot out of the way, using approximately 60 N of force.

   Important: Applying too much force may damage the arm.

   Important: The *backdriving* range is +/- 180°.

2. Make sure nothing is interfering with the robot and that it is safe to proceed,
3. Clear the fault.
4. Power the arm.

Stop the robot immediately when a part of the robot begins to vibrate. Contact support at support@kinova.ca

---

Stop the robot immediately when a joint on arm suddenly becomes loose. Contact support at support@kinova.ca

---

Stop the robot immediately when the arm suddenly loosens from the base. Secure the arm tightly to the surface.

Warnings

Warning: A warning is a potentially hazardous situation that can cause minor to moderate injury to workers, might result in serious injury or even death, and can damage the property significantly. Avoid potentially hazardous situations that are associated with warnings.

The personnel working with Link 6 should always be trained.

Note: Training requirements are determined by the integrator's *risk analysis*.

Either warm or cool the environment before storing the robot.
Do not store the robot in an environment that is below 0 °C.
Do not store the robot in an environment that is above +65 °C.
Do not power the robot in an environment that is below 0 °C; the robot may malfunction.
Do not power the robot in an environment that is above 40 °C; the robot may malfunction.
Do not use the robot in an environment that is below 0 °C; the robot may malfunction.
Do not use the robot in an environment that is above 40 °C; the robot may malfunction.
Do not touch the robot if the temperature of the environment is above 40 °C. The robot may be hot to the touch.
Do not use the robot when there are external signs of significant damage on the arm, controller, or the teach pendant.
Do not use the robot when the I/O panel is open or damaged.
Do not open the controller panel when the controller is in a wet environment.
Do not open the controller panel when the controller is in an EMI-sensitive environment.
Do not use the robot near flames or sources of heat.
Do not use the robot to take objects under water.
Do not use in heavy rain.
Lock out the device during maintenance; not locking the device could result in serious injury.

Cautions

CAUTION: A caution is a hazardous situation that may cause minor to moderate injury to workers and can damage the property. Avoid hazardous situations that are associated with cautions.

Do not touch the robot when the surface of the robot or the controller become hot.

Remove objects blocking the enclosure fans.

Note: Obstructed fans can lead to loss of functionality in internal components and can cause the controller to become hot to the touch.

If there are no objects blocking the enclosure fans and the controller becomes hot to the touch, the fans may be clogged by dust or debris. Perform basic maintenance and fan cleaning.

Related topics

Intended use on page 2

Unintended uses on page 3

Weekly maintenance on page 339

Installing the arm directly to a surface on page 131

Installing the arm on a third-party adapter plate on page 133

Risk assessment

Before installing Link 6, determine all potential risks in using the robot according to applicable standards ISO 10218, ISO 12100 for robot and machine safety.

There are several different steps in determining all potential risks.

Hazard identification	Identify all hazard and risk factors that can potentially cause harm
Risk analysis	Analyze and evaluate the risk associated with each hazard.
Risk control	Eliminate, or reduce the risk to an acceptable level, each potential hazard.

Take into account the entire *robot system* and its environment when performing a risk assessment, not just Link 6. It is impossible to cite every potential hazard because of the myriad applications into which Link 6 can be integrated. The potential hazards listed is not exhaustive.

Examples of what to take into account when performing a risk assessment

- Other robots used in the robot system
- Anticipated tasks of the robot system
- Anticipated workpieces
- Anticipated tools
- Anticipated uses
- Anticipated tasks
- Anticipated environments
- Types of applications, whether *collaborative* or *non-collaborative*

  Note: For more details, please refer to ISO 10218 and ISO TS 15066.
- Robot parameters including, but not limited to, speeds and trajectories
- Safety function parameters including, but not limited to, restricted areas and speed limited by areas

- Protective measures including, but not limited to, interlocks, presence detection, fencing, and guards
- User training

Perform all risk assessments in compliance with the applicable standards of the territory where the robot is installed.

Important: Make sure to make the risk assessment accessible to all robot users.

Safety equipment

Applicable standards and the results of the *risk analysis* determine the requirements for safety equipment.

The equipment includes items that protect users against injury and death. Some equipment is mandatory.

Examples of mandatory equipment

- One or more emergency stops

  The exact number of required emergency stop buttons is determined by the layout of the system and the application of the robot.
- One *enabling device*

Some equipment may be useful, depending on the environment, standards, and risk analysis.

Examples of optional, yet useful, equipment

- Fencing
- Dividers
- One or more light curtains to detect when something is entering the working area of the robot
- One or more scanners to detect when something is entering the working area of the robot

Assessments before using the robot

Every time before using Link 6, perform system checks to prevent potential damage to the robot, the robot system, and the user that could lead to a dangerous situation.

Make sure the risk assessment is available to all users.

Make sure all users receive training regarding safety concerns.

Verify that all tasks that are intended to be performed with the robot , including installation, programming, maintenance, and decommissioning, are listed in the risk assessment and that the risk assessment has been performed.

Related topics

Preventive maintenance on page 338

Stop categories

The different categories of stopping the motion of the robot safely when an emergency arises is designed to reduce existing hazards. The stop categories are aligned with IEC 60204-1.

Table 36: Stop categories

Stop category	Description
0	Removes electrical power from the robot immediately.
1	Initiated by pressing the *E-stop* button and by faults detected by the *Safety control unit* (SCU).
	Initiated by faults detected by the *SCU*.
	Stops the movement of the robot and, when the robot has stopped moving, removes electrical power from the robot.
2	Brings the robot to a Monitored Stop.
	Power remains available to the robot.

Related topics

List of safety functions on page 54

State: Monitored stop on page 112

Operating modes on page 115

Emergency stop

An emergency is an unexpected situation that can lead to injury, or can damage the robot system or the objects it manipulates. An *E-stop* helps to stop the robot quickly in case of emergency.

When an emergency arises during the operation of Link 6 or any other part of the robot system, press the *E-stop* button. Pressing the button triggers a stop category 1, which stops the movement of the robot before removing power from the arm.

Whenever the *E-stop* is pressed, the light behind the ACK button on the controller and the light around the wrist wring switches off. Re-arm the *E-stop* button and reset the robot in *Kortex Web App*. After recovering from the stop, inspect the area for any obstructions that can interfere with the operation of the robot. Also, inspect the robot to ensure it has not been damaged.

If there is no damage and the emergency situation has been handled, restart the robot using Kortex Web App.

Note: The *E-stop* is found on the *teach pendant*. An *E-stop* may also be connected to the controller. When Link 6 is being used in Hand Guiding mode, make sure the *E-stop* is within reach.

Related topics

State: Fault on page 113

Move robot arm without power

It is possible to move the *arm* when it has no power to prevent injuring a person who is trapped by the robot, for instance. The action is known as *backdriving*.

Move the arm to a safe position even without a powered drive system by backdriving the brakes manually in each joint. Moving the arm without power requires a substantial amount of force.

Warning: Backdriving the actuators more than 360 ° may damage the robot and internal cables. Depending on the position of the robot, damage may occur when the movement is less than 360 °.

Warning: Backdriving the actuators may require the arm to be reprogrammed.

CAUTION: Backdriving the actuators manually may damage the brakes.

Related topics

Safety directives and warnings on page 48

Safety functions

Link 6 has built-in tools designed to help the *integrator* to reduce risk when performing the risk analysis.

The tools are also known as safety functions. They rely on the hardware, firmware, the *SCU*, electronics, sensors, and other elements of the robot.

List of safety functions

Safety functions are built into Link 6 to minimize hazards in the robot system.

Table 37: List of safety functions

Number	Name	Triggers stop category	Level of performance
SF01	*Emergency stop*	SC1	PLd

Number	Name	Triggers stop category	Level of performance
SF02	*Protective stop*	SC2	PLd
SF03	*Joint position*	SC2	PLd
SF04	*Joint speed*	SC2	PLd
SF05	*Monitored stop*	SC0	PLd
SF06	*Protection zone*	SC2	PLd
SF07	*Tool Center Point (TCP)* & *Elbow* speed	SC2	PLd

Related topics

Stop categories on page 53

Safety modes on page 61

Robot safety protections and limits on page 72

Protection zones on page 77

Status of safety functions on page 298

SF01 Emergency stop

What does SF01 monitor?

SF01 monitors external emergency signals from the Safety I/Os to the robot. It initiates a *stop category* 1, which stops the movement of the robot before removing power from the arm.

Cause of the trigger

- The *E-stop* button is pressed on the *teach pendant*.
- A signal is detected on the input reserved for emergency stops.

Remedy

1. Make sure that it is safe to resume operation of the robot.
2. Tap **CLEAR FAULT** in the Robot control panel to remove the signal.
3. Reset the E-stop button by turning the button.
4. Switch the arm on.

Related topics

State: Fault on page 113

List of safety functions on page 54

Status of safety functions on page 298

SF02 Protective stop

What does SF02 monitor?

SF02 monitors *protective stop* signals sent to the robot. Protective stops interrupt the robot motion, but it does not remove power from the robot. It is a *stop category* 2.

Cause of the trigger #1

Internal processes of the *main processing unit* (*MPU*) are avoiding a category 0 stop before safety thresholds are exceeded.

Remedy for trigger #1

Tap **Diagnostics** > **Notifications** or tap > **VIEW ALL** to identify the cause of the protective stop.

Cause of the trigger #2

External devices assigned as protective stops on the **Safety I/Os** page.

Remedy for trigger #2

Ensure the signal of the specific I/O channel is off by sending a Protective Stop Reset signal. For example, switch off the light curtain signal, close a door linked with this specific signal, and so on.

Related topics

Safety I/O on page 87

List of safety functions on page 54

Status of safety functions on page 298

SF03 Joint position monitoring

What does SF03 monitor?

SF03 monitors the positions of each joint and compares the positions with the limits set in the **Joint Limits Position Limits** page.

Cause of the trigger

Certain trajectories may inadvertently make a single joint rotate constantly in the same direction without unwrapping. It is most common with the last joint. The unidirectional rotation may also happen when operating in Automatic and Recovery modes. It results in a *stop category* 2 protective stop.

Remedy

1. Clear the associated fault.
2. Identify which joint is nearing its position limit.
3. Move that joint in the position opposite to its limit.
4. Launch the **Jog Angular** panel and check the current joint positions.
5. Adjust the trajectory that caused the fault so that the joint unwraps before it reaches the limit.

Related topics

Joint limits on page 73

Operating modes on page 115

Mode: Manual Jog on page 115

List of safety functions on page 54

Status of safety functions on page 298

SF04 Joint speed monitoring

What does SF04 monitor?

SF04 monitors the speed of each joint and compares the speeds with the limits set in the **Joint Limits Speed Limits** page.

Cause of the trigger

- Unforeseen environmental forces could cause parts of the robot to exceed the speed planned for its trajectory. The unforeseen forces, such as a collision with a wall, may cause the *actuators* to *backdrive*.
- It can also be caused when the robot is in Monitored Stop when an external force is applied to it.

It results in a *stop category* 2 protective stop.

Remedy

Resume the trajectory. Link 6 has built-in functions to saturate its speed limits automatically.

If the error is triggered consistently, consider modifying the trajectory.

Related topics

Joint limits on page 73

Elbow limits on page 76

Operating modes on page 115

Mode: Manual Jog on page 115

List of safety functions on page 54

Status of safety functions on page 298

SF05 Monitored stop

What does SF05 monitor?

SF05, or *Monitored stop*, monitors the velocity of each joint and makes sure the velocity is zero when the robot is not moving. It makes sure the arm remains in a standstill position.

Cause of the trigger

Motion in the *actuators* is detected and results in a stop category 0. For example, the robot stops moving when it is vibrating a lot.

Remedy

1. Remove the external source of motion.
2. Add damping elements to each source of vibration that is in contact with the robot.

Related topics

List of safety functions on page 54

Status of safety functions on page 298

SF06 Protection zone monitoring

What does SF06 monitor?

SF06 monitors the robot to make sure no part of the robot, including the user-defined tool sphere, comes in contact with any of the user-defined *protection zones* that are enabled.

Cause of the trigger

The robot touches an enabled protection zone when operating in Manual or running a program. It results in a *stop category* 2.

Remedy

1. Power the robot.
2. Clear the fault.
3. Use Recovery mode to leave the protection zone.
4. Reconfigure protection zones if necessary and safe from **Safety** > **Protection Zones** in *Kortex Web App*.
5. Disable protection zones if necessary and safe from **Safety** > **Protection Zones** in Kortex Web App.

Related topics

Operating modes on page 115

Protection zones on page 77

List of safety functions on page 54

Status of safety functions on page 298

SF07 TCP and elbow speed monitoring

What does SF07 monitor?

SF07 monitors the *Cartesian* speeds of the *TCP* and the *elbow*, which is the third joint,so that the speeds do not exceed the values set in the **TCP & Elbow Limits** page.

Cause of the trigger

- A large force is exerted on the robot during Hand Guiding mode, causing the speed of the *TCP* or elbow to increase.
- Joint commands, especially on the first joint, reach large angular speeds.

It results in a *stop category* 2.

Remedy

- Apply constraints on local motion to limit the joint speeds and to keep the joint speeds within safe values.
- Modify the trajectory to include more waypoints such that the end effector and the elbow cannot reach high velocities.

Related topics

Mode: Hand Guiding on page 118

Tool center point limits on page 74

Elbow limits on page 76

List of safety functions on page 54

Status of safety functions on page 298

Stopping times and distances

The stopping time, measured in milliseconds, and distance, measured in degrees, are detailed for the first three joints of the *arm*. It follows ISO 10218-1:2011.

Table 38: Stopping time and distance in accordance with ISO 10218-1:2011

Percentage of maximum payload (%)	Extension (%)	Speed (%)	First joint (Base)		Second joint (Shoulder)		Third joint (Elbow)	
			Stop time (ms)	Stop distance (°)	Stop time (ms)	Stop distance (°)	Stop time (ms)	Stop distance (°)
33%	33%	33%	290.13	7.76	238.42	6.87	532.08	8.50
		66%	331.05	19.02	611.09	20.11	756.18	30.78
		100%	409.52	38.57	591.21	40.43	705.37	41.67
	66%	33%	205.13	6.20	582.20	7.79	786.88	8.43
		66%	320.01	18.35	568.73	19.44	718.73	21.26
		100%	324.87	32.39	553.50	39.02	681.18	35.97
	100%	33%	343.08	4.80	333.61	6.26	704.49	8.95
		66%	182.39	9.12	440.58	12.00	601.87	22.07
		100%	262.86	17.04	677.68	22.66	717.89	41.58
66%	33%	33%	197.77	7.39	506.05	16.22	499.89	8.45
		66%	286.76	20.12	648.17	19.14	709.07	28.28
		100%	476.89	39.13	726.71	41.24	632.92	41.79
	66%	33%	267.15	6.06	430.09	10.97	588.40	8.41
		66%	254.99	16.61	501.77	23.12	604.31	21.10
		100%	331.13	32.28	525.56	36.19	608.16	33.33
	100%	33%	136.35	3.96	482.65	6.25	794.85	8.10
		66%	178.68	8.85	467.53	13.46	694.57	21.52
		100%	272.87	17.40	651.80	24.57	761.62	41.16
100%	33%	33%	198.08	7.54	997.89	7.06	649.97	8.54
		66%	286.74	19.16	614.52	20.16	671.20	28.75
		100%	478.41	40.18	541.15	40.65	589.54	41.46
	66%	33%	162.72	6.27	506.89	11.63	668.56	8.68
		66%	275.33	16.77	484.38	21.69	726.22	22.62

Percentage of maximum payload (%)	Extension (%)	Speed (%)	First joint (Base)		Second joint (Shoulder)		Third joint (Elbow)	
			Stop time (ms)	Stop distance (°)	Stop time (ms)	Stop distance (°)	Stop time (ms)	Stop distance (°)
		100%	323.83	32.35	699.15	37.72	649.07	31.39
	100%	33%	174.20	5.94	390.97	6.59	698.42	9.25
		66%	180.34	9.05	463.63	14.23	742.03	23.35
		100%	344.00	37.41	509.47	23.16	892.70	42.64

Diagnostic coverage

Link 6 continually monitors itself and all diagnostics are displayed in *Kortex Web App*.

All safety functions are monitored. The state of each part that makes the *arm* work are also monitored. Problems encountered, with respect to safety functions of the arm, are displayed on the **Status** pages. The problems must be resolved before the robot can move again.

General and detailed information about the *base*, *actuators*, *end effector*, and *flange* positions are displayed on the **Monitoring** pages.

Related topics

Diagnostics on page 297

Status of safety functions on page 298

Safety modes

Configure speeds both as normal or reduced to define safe operation of the robot.

Normal *safety mode* determines the maximum speeds at which the robot can move in *Cartesian* and *angular* directions.

Reduced safety mode also determines the maximum speeds at which the robot can move in Cartesian and Angular directions. However, the maximum speeds in reduced mode are lower than those of the normal safety mode.

Reduced mode can be triggered by safety inputs. For instance, when the robot is learning its trajectories through Hand Guiding, the speed is dictated by the reduced mode to prevent injury.

Remember: It is the responsibility of the *integrator* to set all normal and reduced speeds with respect to the risk assessment. Also, all reduced speeds must be configured to be less than the corresponding normal speed.

Reduced mode can be selected from *Kortex Web App*.

Related topics

Mode: Manual Jog on page 115

Mode: Hand Guiding on page 118

Mode: Hold-to-Run on page 119

List of safety functions on page 54

Robot safety protections and limits on page 72

State timelines of safety functions

Each safety function enters one of two states during operations.

A change in the state of the arm during different modes of operation dictate the state of each of the safety functions.

Table 39: Existing states and modes of operation

Safety function states	• active • inactive
Arm states	• Initialization • Idle • Maintenance • Brake Release • Operational • Fault power off • Recovery • Fault power on
Operating modes	• Manual Jog • Hand Guiding • Hold-to-Run • Automatic • Monitored Stop • Recovery

Related topics

State timelines of safety outputs on page 94

SF01 Emergency stop timeline

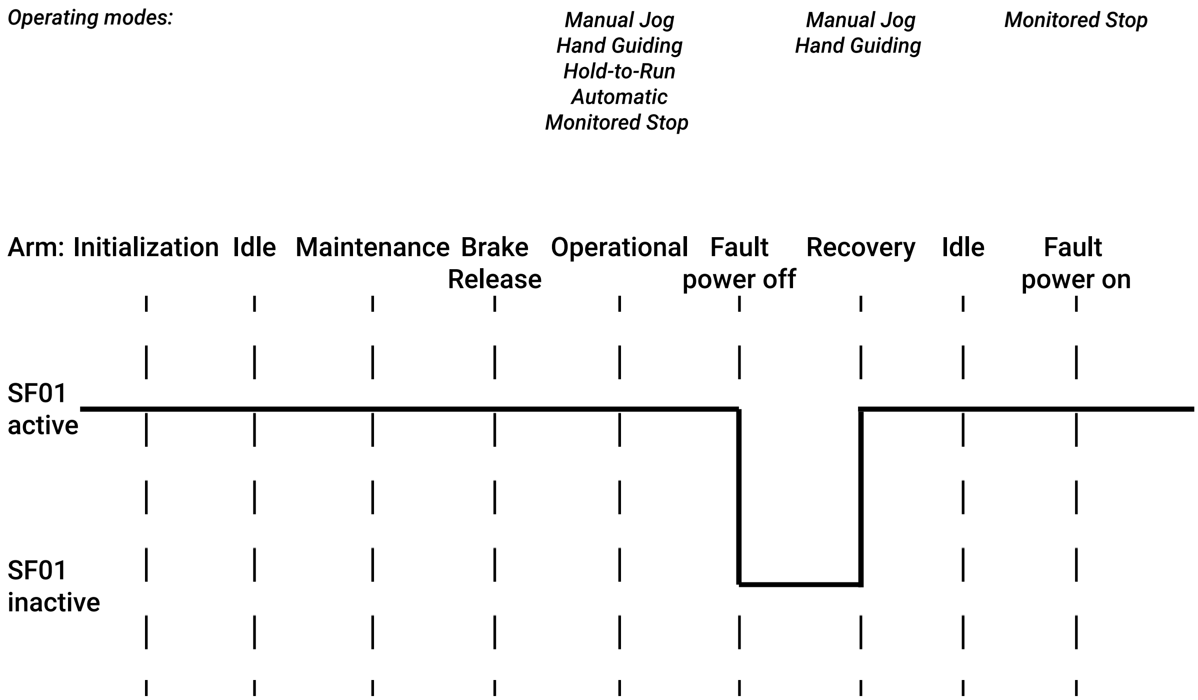
The triggers for and the recovery from the Emergency stop function changes the state of the function.

Equally important in knowing the timeline of a safety function is knowing where to find the current safety function status. For the status of SF01 Emergency stop function, there are a few places in Kortex Web App.

- **Diagnostics** > **Status** > **Safety Functions**
- **Diagnostics** > **Status** > **SCU** > **Emergency stop button signals disagreement**
- **Diagnostics** > **Status** > **SCU** > **Teach pendant emergency stop button signals disagreement**

Table 40: SF01 trigger and action timeline

Monitor frequency of trigger	0.01 s
Trigger	• One of two channels of the Emergency stop safety input sends the signal to the controller . • The *E-stop* button on the teach pendant is pressed.
Trigger results	• Stop category 1 is deployed. • The arm is in Fault power off.
Reset	• Release E-stop attached to the controller. • Release E-stop button on the teach pendant. • Clear the fault from Kortex Web App .
Reset result	• State of robot is Idle. • No power is feeding the arm.
Failsafe	Not applicable



Note: SF01 is inactive when the arm is in the state Fault power off only when it is initiated by stop category 0 or stop category 1 only.

Figure 41: Timeline for SF01 before and after being triggered

SF02 Protective stop timeline

The triggers for and the recovery from the Protective stop function changes the state of the function.

Equally important in knowing the timeline of a safety function is knowing where to find the current safety function status. For the status of SF02 Protective stop function, there is one place to look in Kortex Web App: **Diagnostics** > **Status** > **Safety Functions**.

Table 41: SF02 trigger and action timeline

Monitor frequency of trigger	0.01 s
Trigger	One of two channels of the Protective stop safety input sends the signal to the controller .
Trigger results	• *Stop category* 2 is deployed. **Remember:** The operating mode is Monitored Stop. • The operating mode cannot be changed. • The arm is Operational.

Reset conditions - Automatic	The function resets automatically when all the channels of the safety input that are configured for Protective Stop with Automatic Reset signals the controller.
Reset conditions - Manual	All conditions must be met. • All the channels of the safety input that are configured for Protective Stop with Manual Reset. • All the channels of the safety input that are configured for Protective Stop Reset signals the controller.
Reset result	The program resumes operation.
Failsafe	SF02 can experience processing errors. In this case, stop category 0 is deployed.

Operating modes: *Hold-to-Run* *Manual Jog* *Automatic* *Hand Guiding* *Monitored Stop*

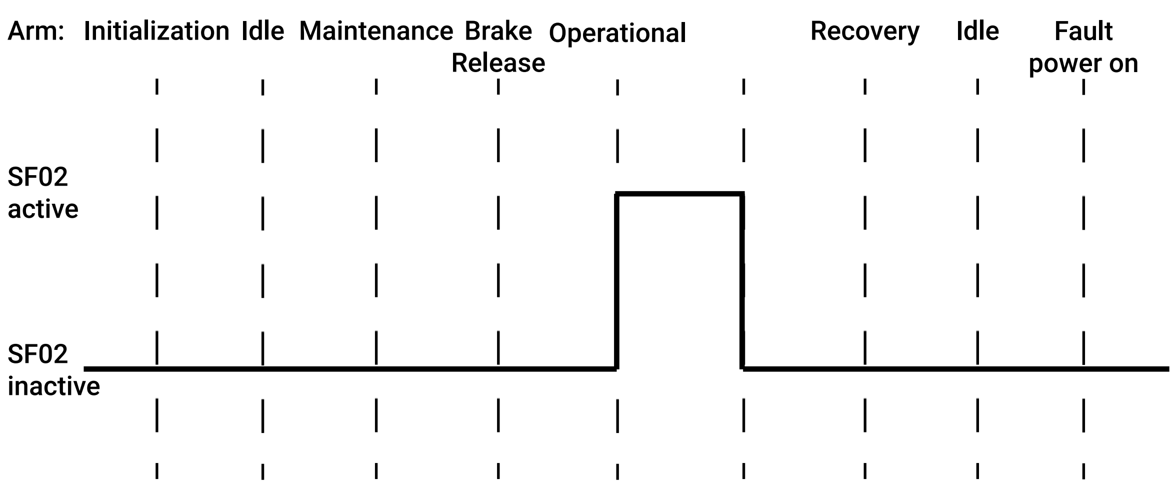


Figure 42: Timeline for SF02 before and after being triggered

SF03 Joint position monitoring timeline

Each safety function enters one of two states during operations; it depends on the current state of the arm and the current mode of operation.

Equally important in knowing the timeline of a safety function is knowing where to find the current safety function status. For the status of SF03 Joint position monitoring function, there are a few places in Kortex Web App.

- **Diagnostics** > **Status** > **Safety Functions**

- **Diagnostics** > **Safety** > **Joint Limits** > **Position Limits**

Table 42: SF03 trigger and action timeline

Monitor frequency of trigger	0.01 s
Trigger	One of the joint positions is outside the configured limit.
Trigger results	• Stop category 2 is deployed. • The arm is in the state Fault power on.
Reset conditions	All conditions must be met. • Each joint position is within operational limits. • The arm is Operational and is in the operating mode Monitored Stop.
Reset	Tap **CLEAR FAULT**.
Reset result	Normal conditions lead to two results. • The operating mode is Monitored Stop. • The arm does not move. Abnormal conditions lead to two results. • The arm is in Recovery. • The position of at least one joint is between the recovery and the operational limit.
Failsafe	SF03 can experience processing errors. In this case, stop category 0 is deployed.

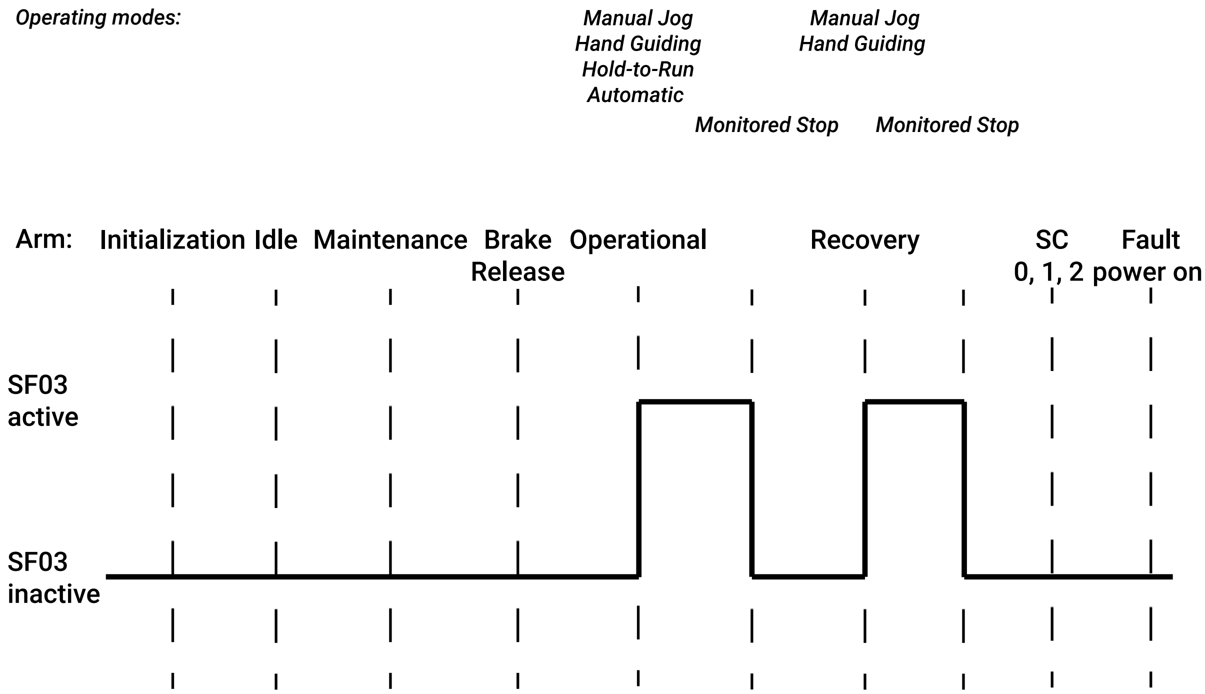


Figure 43: Timeline for SF03 before and after being triggered

SF04 Joint speed monitoring timeline

Each safety function enters one of two states during operations; it depends on the current state of the arm and the current mode of operation.

Equally important in knowing the timeline of a safety function is knowing where to find the current safety function status. For the status of SF04 Joint speed monitoring function, there are a few places in Kortex Web App.

- **Diagnostics** > **Status** > **Safety Functions**
- **Diagnostics** > **Safety** > **Joint Limits** > **Speed Limits**

Table 43: SF04 trigger and action timeline

Monitor frequency of trigger	0.01 s
Trigger	The speed of at least one joint is faster than the configured limit.
Trigger results	• Stop category 2 is deployed. • The operating mode is Monitored Stop. • The arm is in the state Fault power on.
Reset conditions	The speed of the joints are corrected to be within operational limits.
Reset	Tap **CLEAR FAULT**.

Reset result	Normal conditions lead to two results. • The operating mode is Monitored Stop. • The arm does not move.
Failsafe	SF04 can experience processing errors. In this case, stop category 0 is deployed.

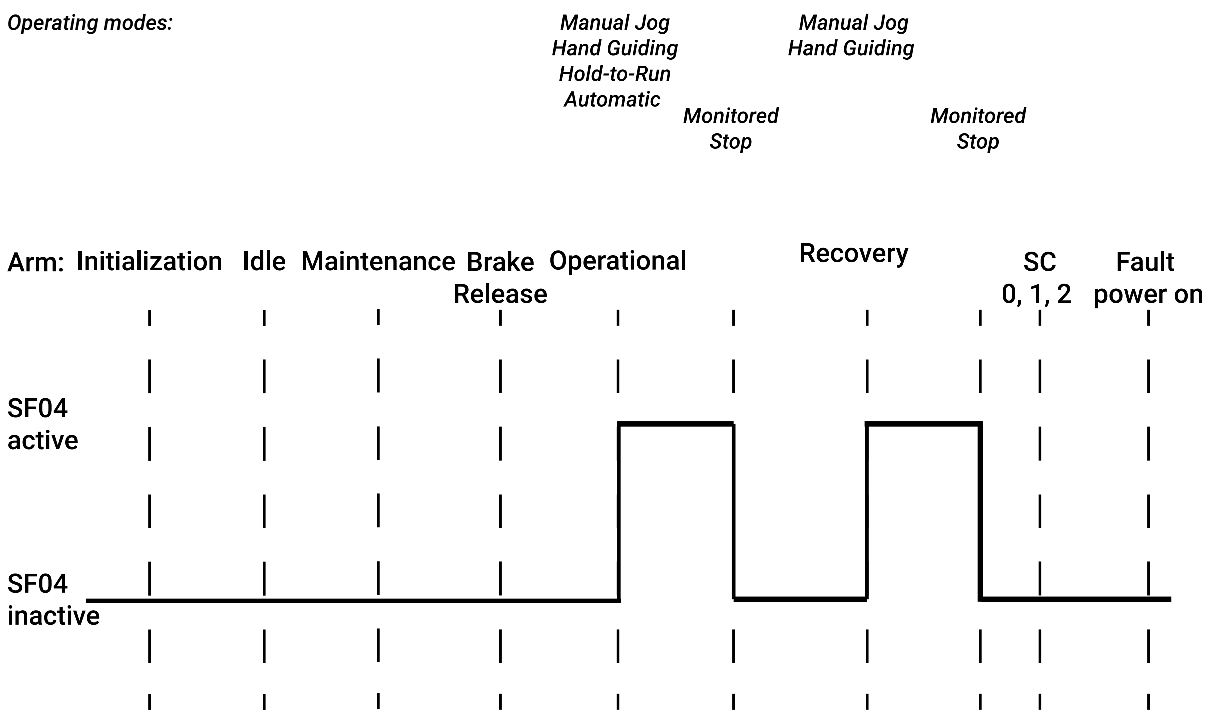


Figure 44: Timeline for SF04 before and after being triggered

Related topics

State timeline for the safety output Normal Speed Mode on page 96

SF05 Monitored stop timeline

Each safety function enters one of two states during operations; it depends on the current state of the arm and the current mode of operation.

Equally important in knowing the timeline of a safety function is knowing where to find the current safety function status. For the status of SF05 Monitored stop0 function, there is one place to look in Kortex Web App: **Diagnostics** > **Status** > **Safety Functions** .

Table 44: SF05 trigger and action timeline

Monitor frequency of trigger	0.01 s

Trigger	• The arm is in the process of stopping and has taken more than 1000 ms to stop. • The difference between the actual position of at least one joint and its position when the robot entered the stop state is more than 0.5°.
Trigger results	• Stop category 0 is deployed. • The arm is in the state Fault power off.
Reset	Tap **CLEAR FAULT**.
Reset result	The operating mode is Monitored Stop.
Failsafe	When SF05 does not receive joint positions for more than 60 ms, stop category 0 is deployed.

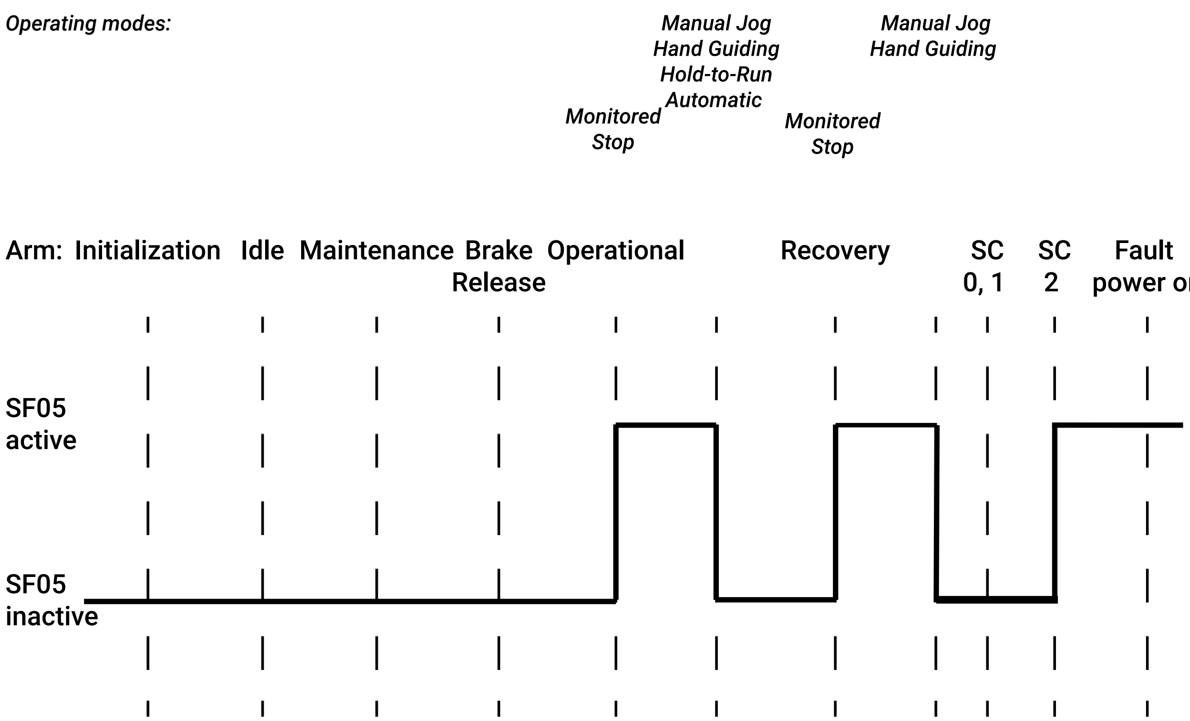


Figure 45: Timeline for SF05 before and after being triggered

SF06 Protection zone monitoring timeline

Each safety function enters one of two states during operations; it depends on the current state of the arm and the current mode of operation.

Equally important in knowing the timeline of a safety function is knowing where to find the current safety function status. For the status of SF05 Monitored stop0 function, there is one place to look in Kortex Web App: **Diagnostics** > **Status** > **Safety Functions** .

Table 45: SF06 trigger and action timeline

Monitor frequency of trigger	0.01 s
Trigger	• Any part of the flange enters at least one protection zone. • Any part of the tool enters at least one protection zone. • Any checkpoint enters at least one protection zone.
Trigger results	• Stop category 2 is deployed. • The arm is in the state Fault power on.
Reset conditions	All conditions must be met. • Tool sphere is outside all configured protection zones. • All checkpoints are outside all configured protection zones.
Reset	Tap **CLEAR FAULT**.
Reset result	• The operating mode is Monitored Stop. • The arm does not move.
Failsafe	SF06 can experience processing errors. In this case, stop category 0 is deployed.

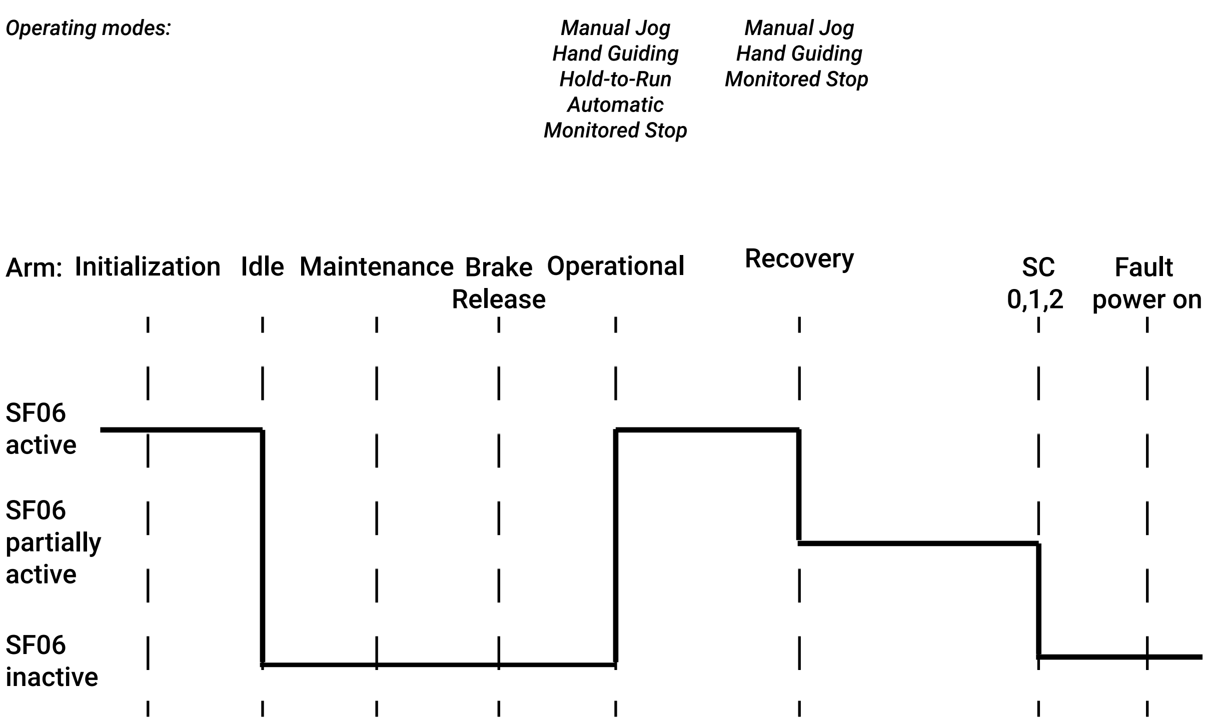


Figure 46: Timeline for SF06 before and after being triggered

Related topics

State timeline for the safety output Normal Speed Mode on page 96

SF07 TCP and Elbow speed monitoring timeline

Each safety function enters one of two states during operations; it depends on the current state of the arm and the current mode of operation.

Equally important in knowing the timeline of a safety function is knowing where to find the current safety function status. For the status of SF07 TCP and Elbow speed monitoring function, there are a few places in Kortex Web App.

- **Diagnostics** > **Status** > **Safety Functions**
- **Safety** > **TCP & Elbow Limits** > **TCP Limits**
- **Safety** > **TCP & Elbow Limits** > **Elbow Limits**

Table 46: SF07 trigger and action timeline

Monitor frequency of trigger	0.01 s
Trigger	At least one trigger is needed. • TCP translation speed is higher than the configured limit. • TCP rotation speed is higher than the configured limit. • Elbow translation speed is higher than the configured limit.
Trigger results	• Stop category 2 is deployed. • The operating mode becomes Monitored Stop. • The arm is in the state Fault power on.
Reset conditions	• The TCP translation speed is corrected to be lower than the configured limit. • The TCP rotation speed is corrected to be lower than the configured limit. • The Elbow translation speed is corrected to be lower than the configured limit.
Reset	Tap **CLEAR FAULT**.
Reset result	• The operating mode is Monitored Stop. • The arm does not move.
Failsafe	SF07 can experience processing errors. In this case, stop category 0 is deployed.

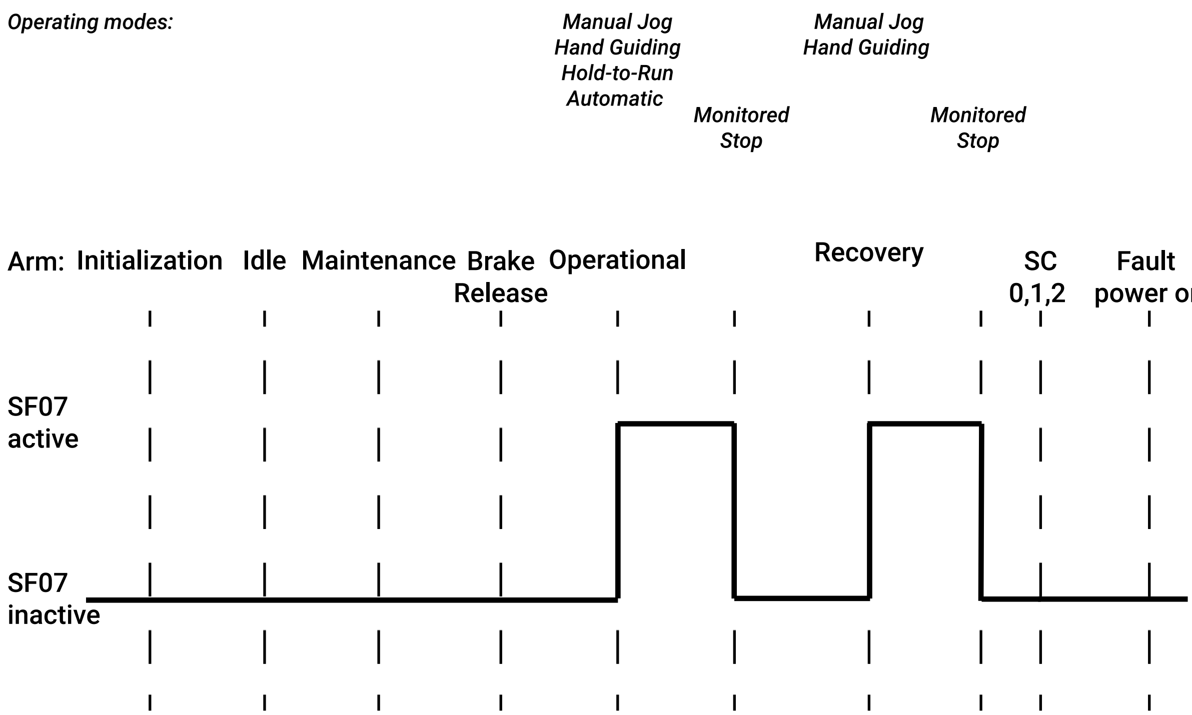


Figure 47: Timeline for SF07 before and after being triggered

Related topics

State timeline for the safety output Normal Speed Mode on page 96

Robot safety protections and limits

To make sure the robot operates safely in its working environment, configure joint limits, *TCP* and *elbow* limits, *protection zones*, and safety I/Os.

All tasks related to configuring the robot with respect to safety are performed by the *integrator*.

Important: Safety parameters can be modified only when there is no power in the arm.

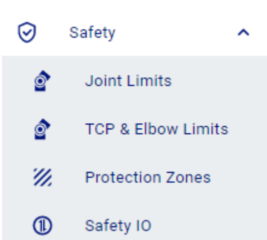


Figure 48: Accessing the safety pages

The *controller* prevents movement, speed, and force outside of the configured limits. The configured limits are monitored by certain safety functions.

- *Emergency stop*

- *Protective stop*
- *Joint position*
- *Joint speed*
- Protection zone
- *TCP* and elbow speed

Related topics

List of safety functions on page 54

Safety modes on page 61

Joint limits

Each joint can be configured with specific minimum and maximum *angular* positions.

Additionally, the speed of each joint can be configured in both normal and reduced safety modes.

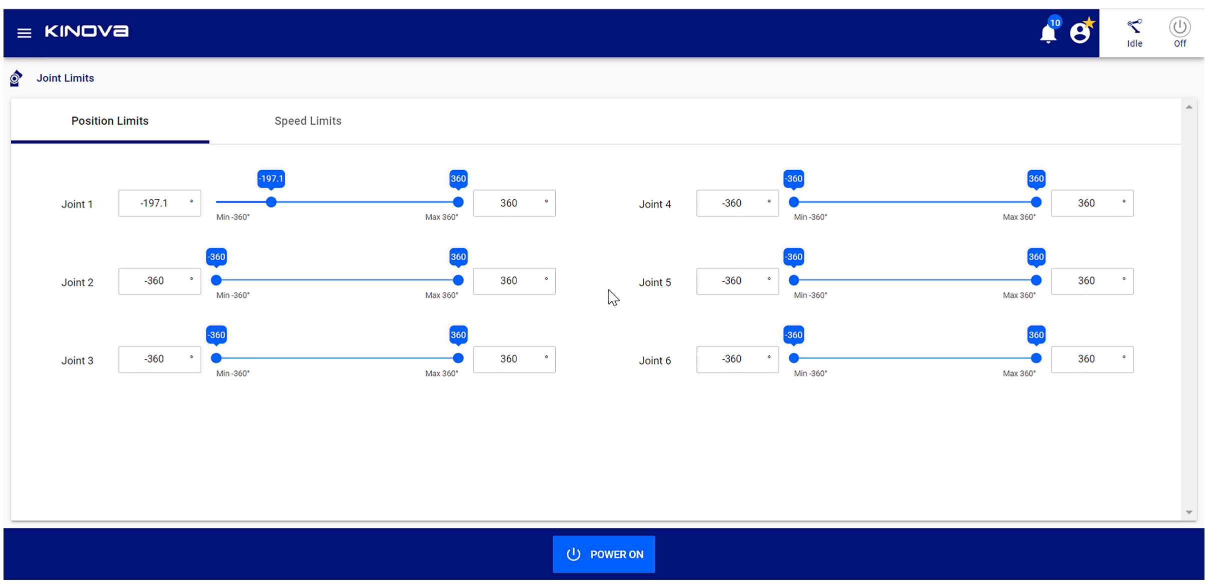


Figure 49: Position Limits page of Joint Limits page

When configuring the speed limits of the joints, select either **Normal** or **Reduced** from the **Safety Mode** menu. The limits apply to the safety mode that is selected.

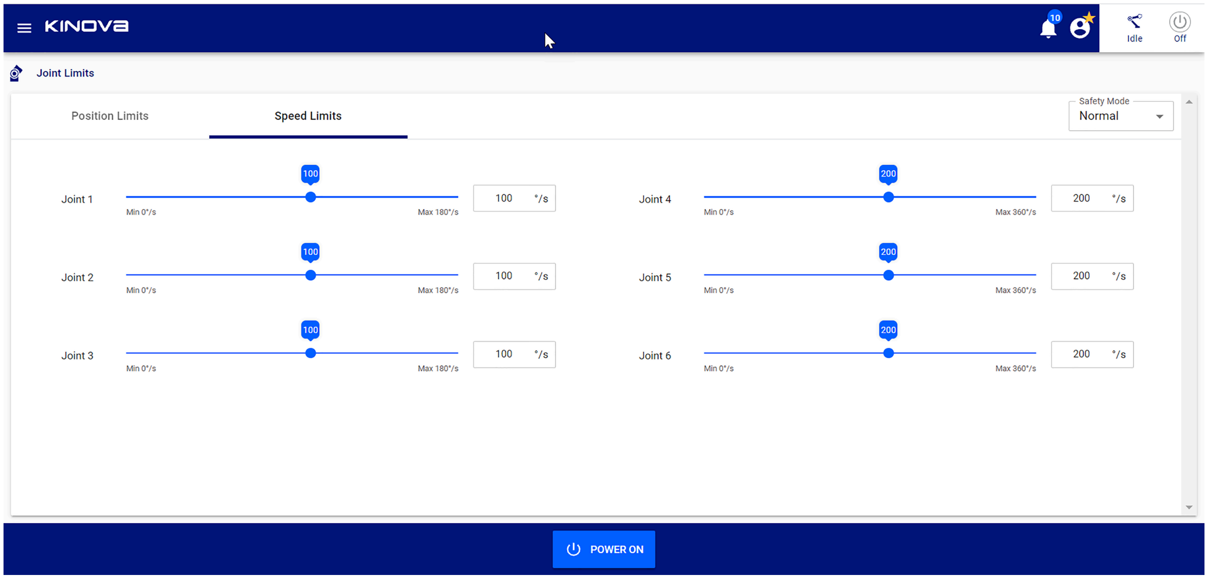


Figure 50: Speed Limits page of Joint Limits page

Table 47: Ranges of speed limits for joints

Safety mode	Joint size	Speed range
Normal	Size 110	0 °/s to 180 °/s
	Size 80	0 °/s to 360 °/s
Reduced	Size 110	0 °/s to 100 °/s
	Size 80	0 °/s to 200 °/s

Tool center point limits

The position and rotation of the *TCP* being used must be configured.

The *TCP* has translation and rotational speed, force, and *angular* positions. The configured *TCP* speeds in the reduced safety mode must be less than the configured *TCP* speeds in the corresponding normal safety mode.

Important: There must be no power in the arm when the robot safeties are being configured.

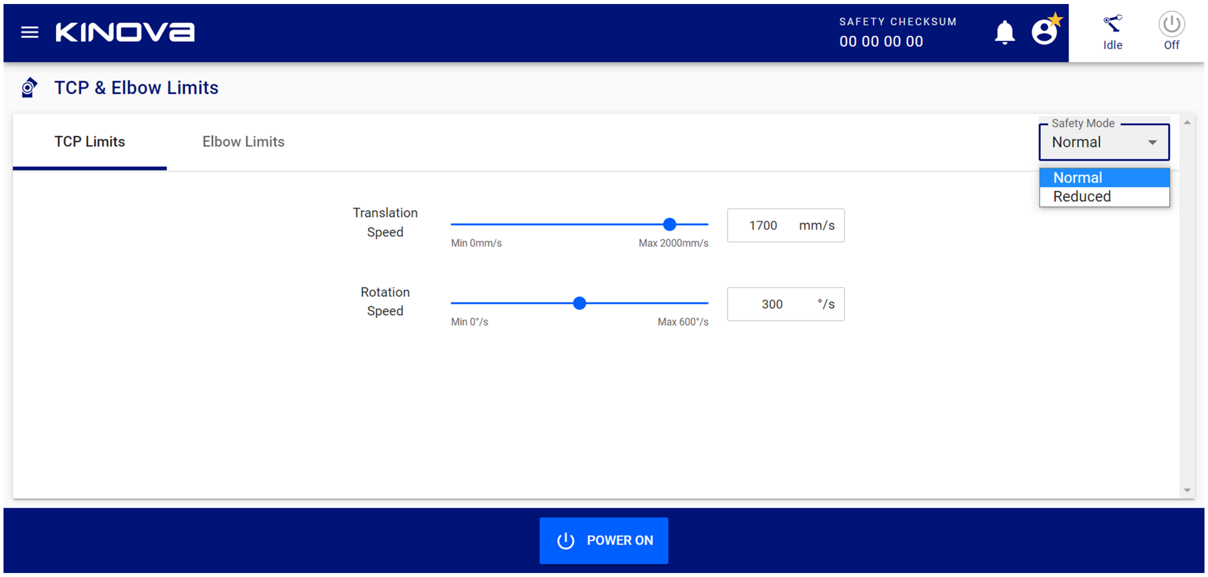


Figure 51: TCP limits

Table 48: TCP configuration limits

TCP limit	Safety mode	Description
Note: The range represents the limits, or thresholds, that trigger faults. They are not in any way desired velocities.		
Translation Speed	**Normal**	Use the slider or enter the value in the text box. Range: 0 mm/s to 2000 mm/s Default: 1700 mm/s
	Reduced	Use the slider or enter the value in the text box. Range: 0 mm/s to 250 mm/s Default: 250 mm/s
Rotation Speed	**Normal**	Use the slider or enter the value in the text box. Range:: 0 °/s to 600 °/s Default: 300 °/s

TCP limit	Safety mode	Description
	Reduced	Use the slider or enter the value in the text box. Range: 0 °/s to 150 °/s Default: 150 °/s

Elbow limits

The speed and orientation of the *elbow* must be configured.

The position and orientation of an *end effector* being used must be configured.

Important: There must be no power in the arm when the robot safeties are being configured.

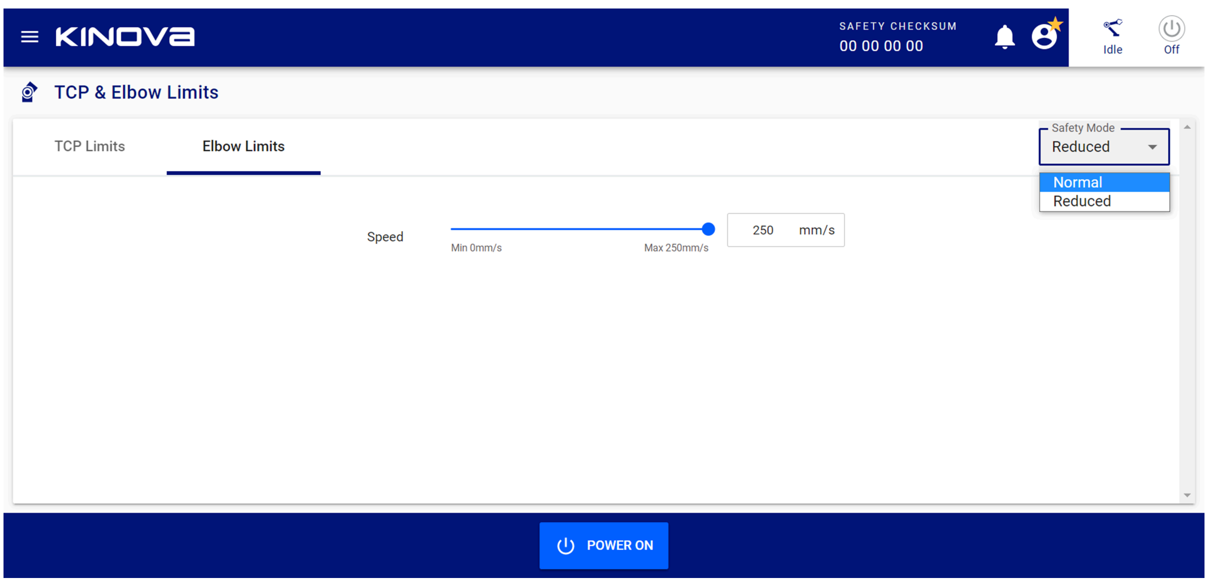


Figure 52: Elbow limits

Table 49: Elbow configuration limits

Elbow limit	Safety mode	Description
Speed	**Normal**	Use the slider or enter the value in the text box. Range: 0 mm/s to 2000 mm/s Default: 1700 mm/s

Elbow limit	Safety mode	Description
	Reduced	Use the slider or enter the value in the text box. Range: 0 mm/s to 250 mm/s Default: 250 mm/s

Protection zones

Prevent or limit where the robot can move by creating *protection zones*.

The zones are three-dimensional geometric volumes intended to limit the possibility of the robot running into users or objects near it. Protection zones define such things as *end effectors*, walls, tables, and objects. Up to eight zones, excluding the tool zone, can be active simultaneously.

Important: The zone defined for the tool is the only zone that cannot be disabled. However, like the other zones, it can be modified. The tool zone must exist so that the robot takes the size of the end effector into account.

Important: There must be no power in the arm when the zones are being defined.

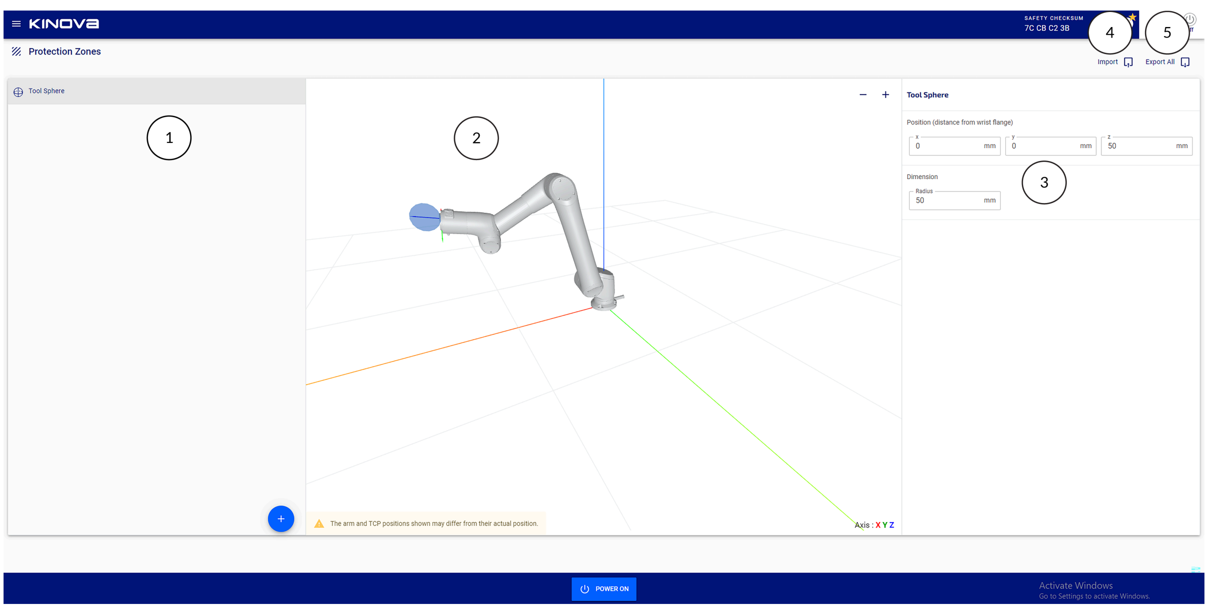


Figure 53: The Protection Zones page

1	Zones pane
2	Visualization pane
3	Zones configuration pane

4 **Import**

Important: Import is available on the teach pendant only when a *USB* is connected in the USB port on the controller. It is available from a web browser when there is no USB stick.

5 **Export All**

Important: Export is available on the teach pendant only when a *USB* is connected in the USB port on the controller. It is available from a web browser when there is no USB stick.

When you first access the **Protection Zones** page, only the **Zones** and **Visualization** panes are visible. Tap on any of the zones in the Zones pane to view its corresponding configuration pane.

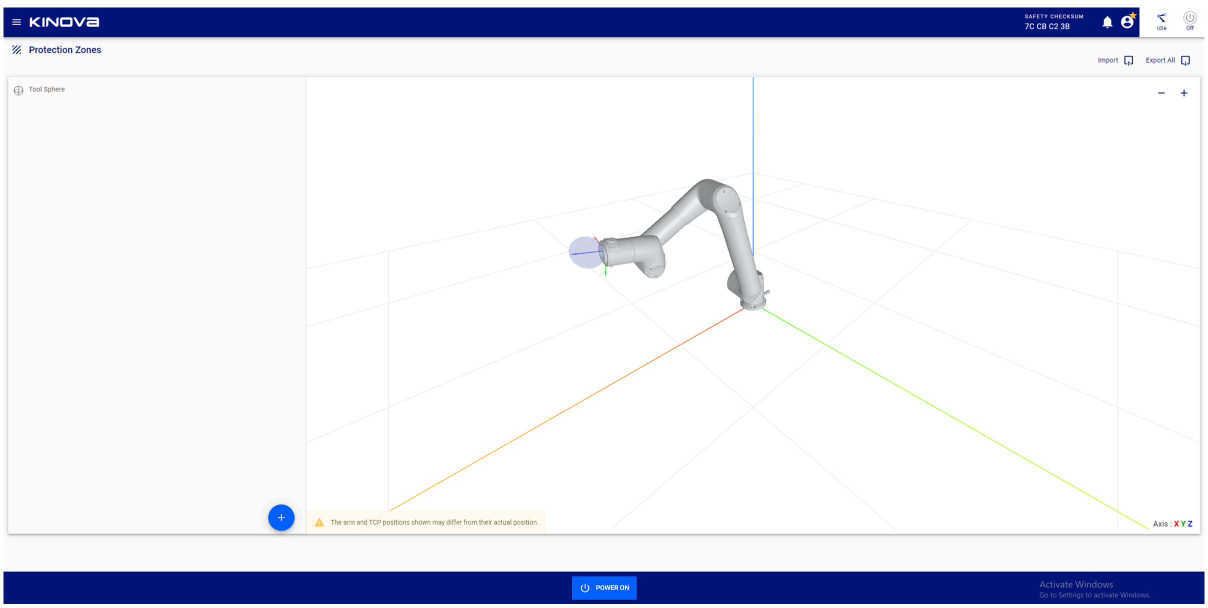


Figure 54: Initial launch of Protection Zones page

Related topics

Effective workspace on page 10

List of safety functions on page 54

Tool sphere

The tool sphere, displayed in the **Protection Zones** page, is a *protection zone* for the *end effector* on the *arm*. The tool sphere is the only protection zone that can move.

The end effector and the arm must never collide with each other. Therefore, the tool sphere is always displayed in the **Protection Zones** page. It can never be deleted and it can never be deactivated. However, the zone of the tool pythosphere can be modified.

Important: Always define the tool sphere with respect to the flange.

Important: Always make sure the tool sphere size and offset reflect reality.

Zones pane

Create, rename, duplicate, and delete zones from the **Zones pane**. Export individual protection zones as XML or JSON from a web browser.

The zone for the *end effector* is always present; in the pane, it is listed as the **Tool Sphere**.

Important: The zone defined for the tool is the only zone that cannot be deleted. It must exist so that the robot takes the shape and offset of the end effector into account.

Protection zones that are already defined are listed in the **Protection Zones** pane.

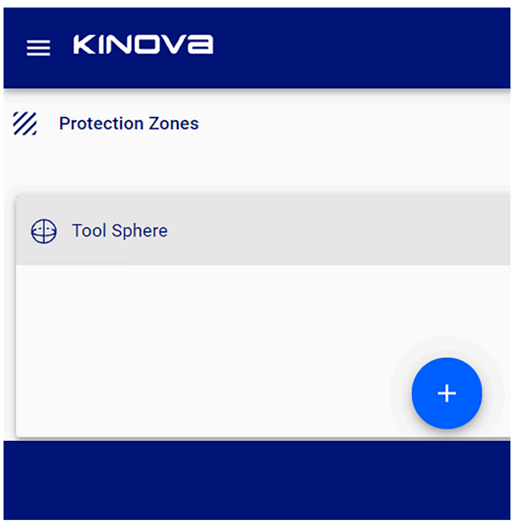


Figure 55: Example when no zones, except for the tool, are listed in the Zones pane

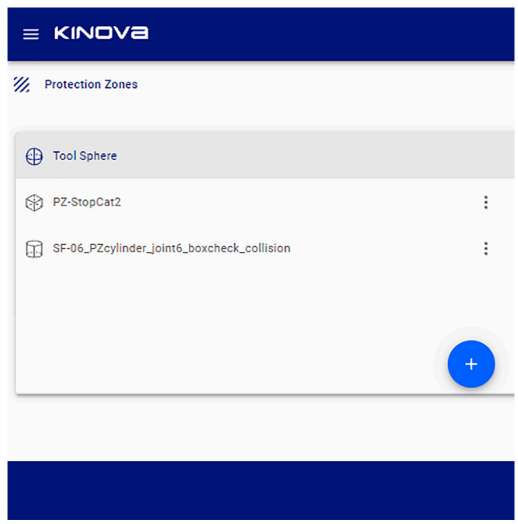
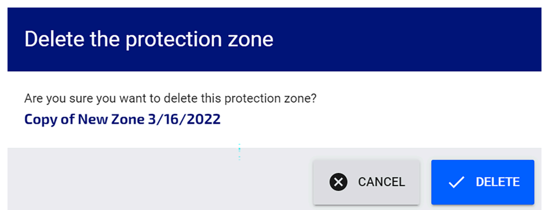


Figure 56: Example of multiple zones listed in the Zones pane

Table 50: Actions that can be performed on each protection zone from the *kebab* (⋮) menu

Selected zone actions	Description
Duplicate	Create a new zone that is a duplicate of the selected protection zone. **Tip:** When the maximum number of zones are created, a message is displayed on the page indicating that no more zones can be added. Also, the option to duplicate from the menu is not selectable.
Delete	Delete the selected zone. A confirmation dialog launches. Tap **Delete** to delete the zone. 
Export XML	Export the currently saved configuration of the selected zone as an XML file. **Important:** Export is available on the teach pendant only when a *USB* is connected in the USB port on the controller. It is available from a web browser when there is no USB stick.
Export JSON	Export the currently save configuration of the selected zone as a JSON file. **Important:** Export is available on the teach pendant only when a *USB* is connected in the USB port on the controller. It is available from a web browser when there is no USB stick.

Adding a safety zone

Safety zones are areas where the robot cannot enter.

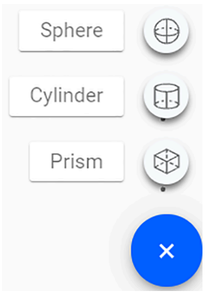
About this task

Create one safety zone for each object or area that the robot must avoid. Up to eight zones can be active and monitored by the robot at one time. However, more than eight zones can be created; zones can be left inactive.

Procedure

1. Tap the hamburger menu (☰) to expand the menu.
2. Tap **Safety** > **Protection Zones**.
3. Tap ⊕ in the Zones pane of the **Protection Zones** page.

   A menu of zone shapes launches.

4. Tap one of the three shapes.

   The new zone with its shape is highlighted in the Visualization pane.

   The default values of the zone is launched in the Configuration pane.
5. Enter the desired values in the Configuration pane.
6. Tap **Activate Zone** to make the zone active.

Manipulating existing safety zones

Deleting, duplicating, and renaming zones are all ways of manipulating existing zones.

About this task

The entire operation is performed within the Zones pane.

Procedure

1. Tap the hamburger menu (☰) to expand the menu.
2. Tap **Safety** > **Protection Zones**.
3. Tap the zone in the list that needs to be manipulated.

   A single tap selects the zone.
4. Tap the associated *kebab* (⋮) menu.

   A menu for the selected zone launches.

5. Tap the action needed for that zone.
   - Tap **Delete** to delete a zone.
   - Tap **Duplicate** to duplicate a zone. Modify the zone properties after duplicating it.
   - Tap **Export JSON** to export the zone to a JSON file.

     Important: Export is available on the teach pendant only when a *USB* is connected in the USB port on the controller. It is available from a web browser when there is no USB stick.
   - Tap **Export XML** to export the zone to an XML file.

     Important: Export is available on the teach pendant only when a *USB* is connected in the USB port on the controller. It is available from a web browser when there is no USB stick.

Zone shapes

Each zone is created using a specific shape.

There are three shapes that are available. Some properties are shared among all shape. Some properties are specific to the shape. The properties of the zones must be defined or configured. Configure the zones in the Configuration pane.

Table 51: Properties specific to the tool sphere zone

Properties	Description
Position	Distance is measured from the x,y,z coordinates of the wrist flange and is measured in millimeters.
	Default coordinates: (0,0,0)
Dimension	The dimension of the radius of the end effector measured in millimeters.
	Default radial size: 100 mm

Table 52: Properties common to all zone shapes that can be added

Properties	Description
Name	When the shape is first created, its name is automatically set to the current date appended to the string "New Zone".
	Change the name to something meaningful.
Activate Zone	By default, the zone is active.

Properties	Description
	Deactivate the zone when the environment around the robot is going to change. For example, when a single arm is used in multiple setups, it is convenient to be able to activate the desired zone for the current environment.

Table 53: Properties specific to the cylinder zone 

Properties	Description
Orientation - Euler	Enter the orientation of the end effector using the extrinsic ZYX Euler angle convention.
	Orientation is measured in degrees.
Position	Distance is measured from the x,y,z coordinates of the base and is measured in millimeters.
	Default coordinates: (0,0,0)
Dimension	Enter the height (z) of the cylinder and the radius of the zone.
	Height and radius are measured in millimeters.
	Default height: 300 mm
	Default radial size: 200 mm

Table 54: Properties specific to the prism zone 

Properties	Description
Orientation - Euler	Enter the orientation of the end effector using the extrinsic ZYX Euler angle convention.
Position	Distance is measured from the x,y,z coordinates of the base and is measured in millimeters.
	Default coordinates: (0,0,0)
Dimension	Enter the length (x), width (y), and height (z) of the zone.
	Length, width, and height are measured in millimeters.
	Default length: 250 mm
	Default width: 250 mm

Properties	Description
	Default height: 250 mm

Table 55: Properties specific to the sphere zone 

Properties	Description
Dimension	Enter the dimension of the zone as a radius.
	The radius is measured in millimeters.
	Default radial size: 200 mm
Position	Distance is measured from the x,y,z coordinates of the base and is measured in millimeters.
	Default radial size: 200 mm

Related topics
Euler angle convention used on page 84

Euler angle convention used

Euler angles are used to represent the orientation of an object, such as a robot tool. The convention used is Tait-Bryan, extrinsic ZYX.

There are three ways to rotate from the reference coordinate from (x,y,z) to the considered rigid body.

1. Rotation about fixed x axis of an angle rx (γ).
2. Rotation about fixed y axis of an angle ry (β).
3. Rotation about fixed z axis of an angle rz (α).

With this angle convention, we can define the associated rotation matrix.

$$R = Z(\alpha)Y(\beta)X(\gamma)$$

Exporting or importing all zones into one file

All zones can be exported into one XML or JSON file after they are defined. All zones can be imported at once.

About this task

Exporting zones is a fast way to integrate the robot, especially when multiple robots are in the working environment. Define zones for one of the robots, import the file for the next robot, and modify the values for it. Again, these new zones can be exported.

Important: Export is available on the teach pendant only when a *USB* is connected in the USB port on the controller. It is available from a web browser when there is no USB stick.

Important: Import is available on the teach pendant only when a *USB* is connected in the USB port on the controller. It is available from a web browser when there is no USB stick.

Important: The exported and imported zones do not include the zone definition for the tool.

Figure 57: Location of Protection zone Export All button

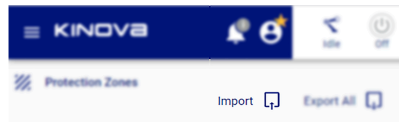


Figure 58: Location of Protection zone Import button

Procedure

Export all zones to one file.

1. Tap **Export All** > **Export JSON** or **Export All** > **Export XML** to export all the zones to a JSON or XML file.
   All zones are saved in the zone definition file.

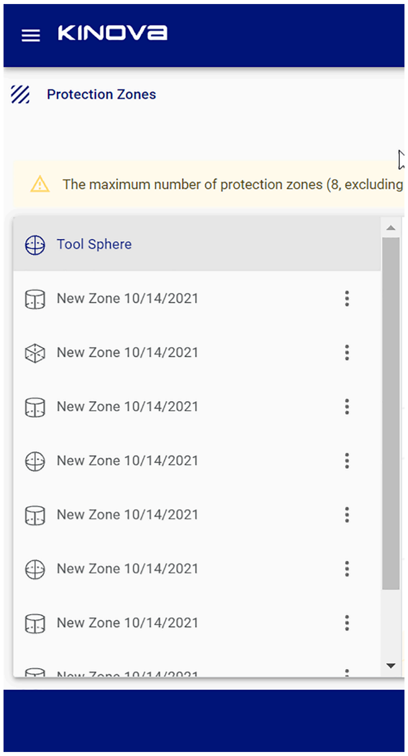
Import all zone from one file

2. Tap **Import**.
3. Browse to the location where the protection zone XML or JSON files are stored.
4. Select the desired file.

Results

The Zones pane populates with all the zones defined in the file.

Example:



Visualization pane

A visualization of the defined protection zones, as well as the arm of Link 6 in its current position, is in the Visualization pane. The view helps to determine the protection zones in a 3-dimensional space with respect to the robot.

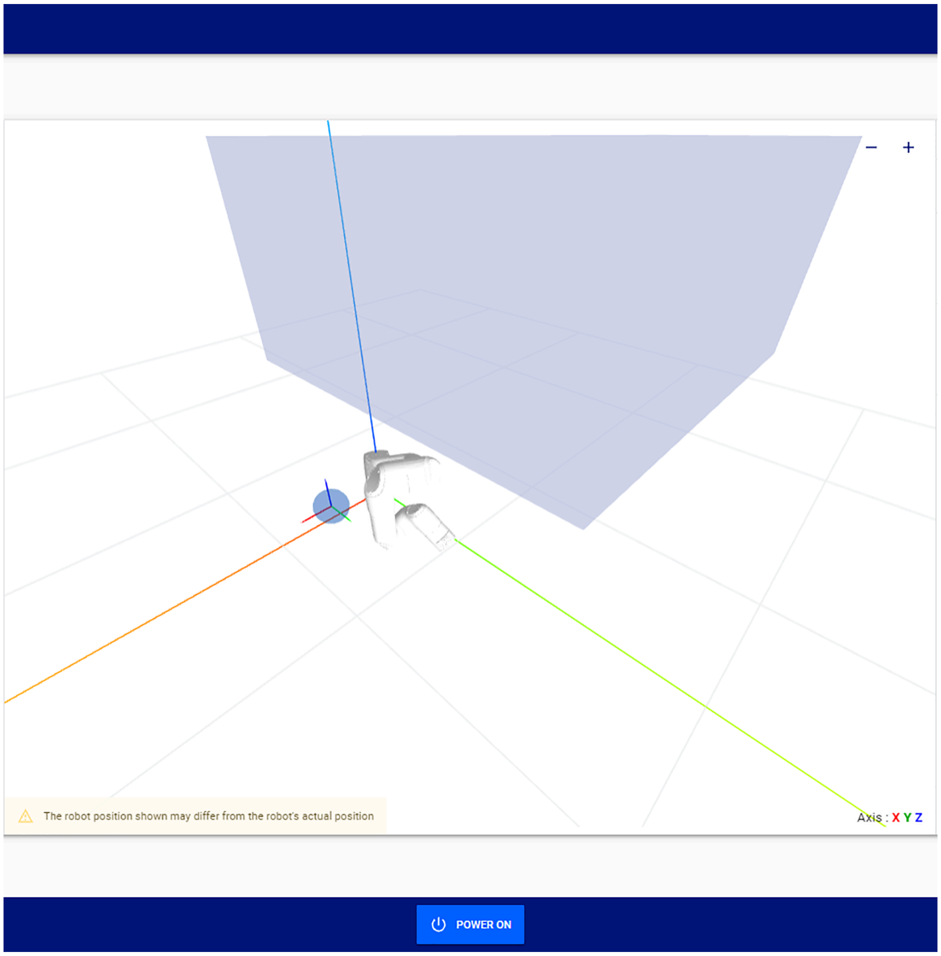


Figure 59: Visualization pane with tool sphere, robot, and other protection zones

Zoom in, zoom out, and view the zones from different angles.

Safety I/O

Each output safety function can be assigned to one or more of the eight channels.

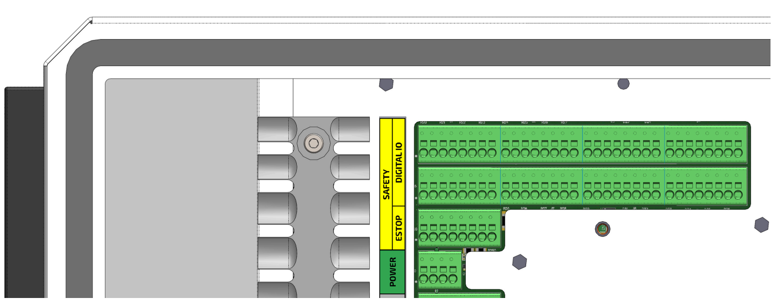


Figure 60: Location of connectors for the Safety I/O equipment

When the physical equipment is connected to the *controller* through the green slots, it can then be assigned a specific behavior on the corresponding channel in Kortex Web App. The channels are numbered 0 through 7.

Important: Make sure the arm is in the Idle *state* before you configure safety inputs. The arm is in the Idle state when the arm is switched off, but the controller is switched on.

Each input safety function is connected to two electrical inputs. When a safety function is triggered, the state changes on both electrical inputs simultaneously.

Likewise, each output safety function is connected to two electrical outputs. When a safety function is triggered, the state changes on both electrical outputs simultaneously.

A discrepancy in states between the two channels implies there is an electrical system fault or there is a device fault.

Safety inputs

Each input safety function can be assigned through *Kortex Web App* to one of eight channels. When a channel is assigned to a safety input, that safety input is no longer available on the other channels.

Access the **Safety I/Os** page by tapping **Safety** > **Safety I/Os**. The default pane is **Inputs**.

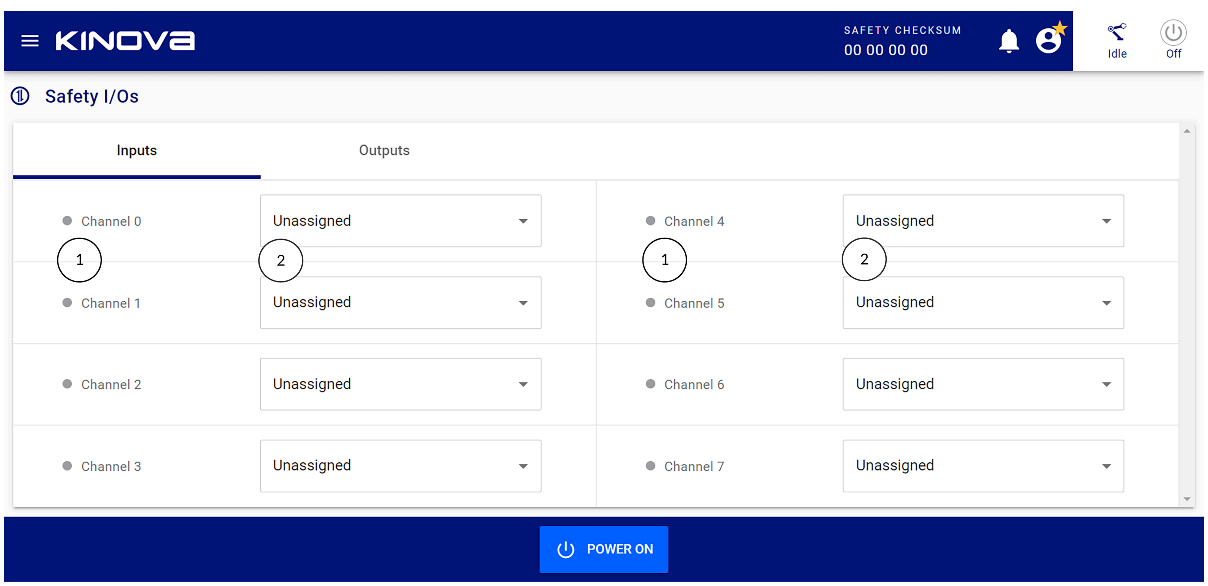


Figure 61: Safety inputs pane

1	Input channel	2	Safety behavior

The safety input behavior drop-down has several choices. Each safety input monitors a specific condition. The different conditions are listed in the safety input drop-down.

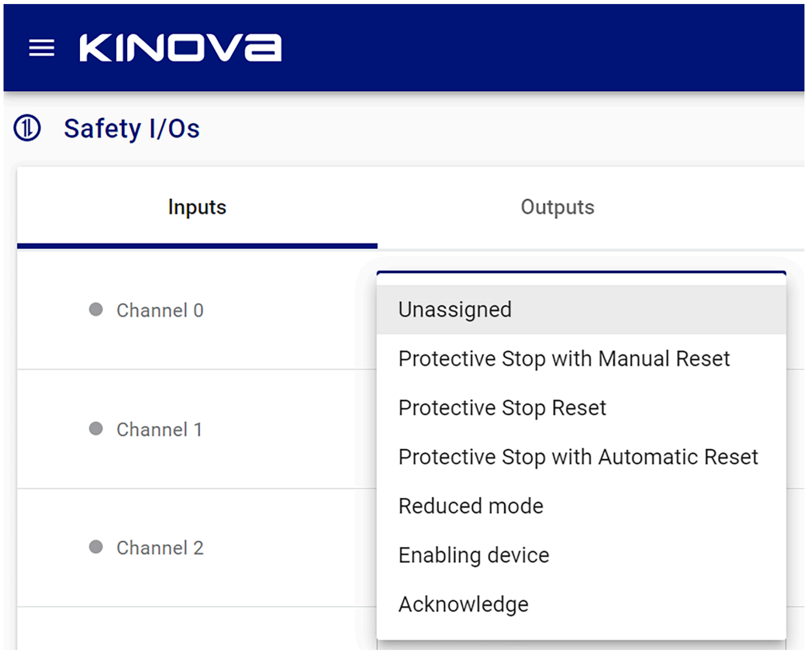


Figure 62: Safety input functions

All safety digital inputs are *normally closed*. The behavior of *NC* changes depending on what configuration is assigned to the channel.

Table 56: Input safety conditions available for each channel

Safety input function name	Description
Unassigned	Channel behavior is not altered for the connected equipment.
Protective Stop with Manual Reset	The robot automatically comes to a *stop category* 2. When the input safety function is triggered, the program automatically pauses running. After the signal is released, the user must reset the robot before it can resume operations. The robot must be in one of two operating modes. • Automatic mode • Hold-to-Run mode

Safety input function name	Description
Protective Stop Reset	It resets a *protective stop*. The robot is in *Monitored stop* when it is in a protective stop. Use the reset input of the protective stop to resume operation. The program resumes where it left off provided the program paused because of a **Protective Stop with Manual Reset** and provided no fault has occurred. If an electrical system fault or a device fault occurs, the robot comes to a category 0 stop with no chance of resuming from the original paused location of the program.
Protective Stop with Automatic Reset	The robot pauses its task automatically and comes to a stop category 2. After the signal is released, the robot resumes operation immediately. When the input safety function is triggered, the program automatically pauses running. After the fault is cleared or the intrusion is gone, the robot resumes operations. The robot must be in one of two operating modes. • Automatic mode • Hold-to-Run mode
Reduced Mode	The reduced speed limits are enforced in all operating modes as long as the signal is active. If the robot is moving at a speed higher than the reduced speed limit when reduced mode is triggered, the robot slows down and complies with the reduced speed. The robot does not start operating in normal mode when the signal is released. When there is an electrical system or device fault, the robot comes to a category 0 stop.

Safety input function name	Description
Enabling device	Connect an external *enabling device* to the controller and assign the corresponding channel the enabling device behavior. When one enabling device is pressed, regardless of where it is, the robot is enabled. **CAUTION:** It is possible that more than one enabling device is connected. However, only one enabling device should be used at a time. When there is an electrical system or device fault, the robot comes to a category 0 stop.
Acknowledge Automatic Mode	The channel overrides the ACK button on the controller when Acknowledge Automatic Mode is not set on the **Controller** pane of the **Robot** page. When the robot is operating in Automatic mode, the acknowledge safety input overrides the need to press the ACK button on the controller; the **Acknowledge Automatic Mode** dialog never launches at the start of running a program in Automatic mode.

Input signals are compatible with devices that implement *OSSD* behavior. The pulsed *Output Signal Switching Device (OSSD)* signals are filtered with the period is less than 3 ms.

Safety outputs

Each output safety function can be assigned through *Kortex Web App* to one or more channels. When a channel is assigned to a safety output, that safety output is still available on the other channels.

Access safety outputs by tapping **Safety** > **Safety I/Os** > **Outputs**.

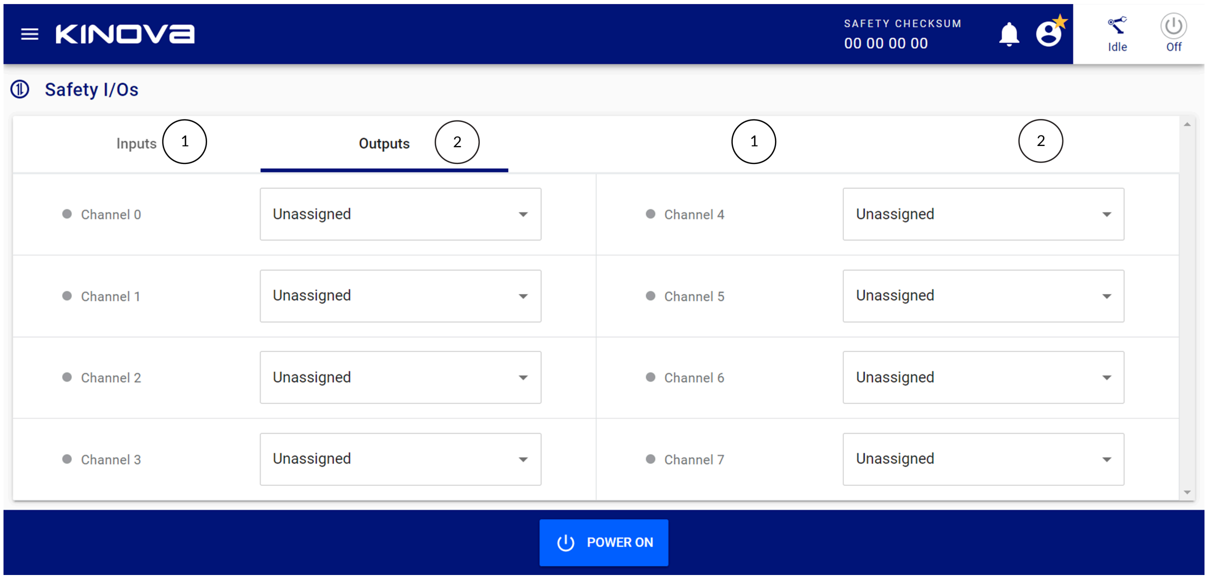


Figure 63: Safety outputs pane

1	Output channel	2	Safety behavior

Each safety output monitors a specific condition. The different conditions are listed in the safety output menu. Connect Safety Output channels to external Safety monitoring devices, such as a PLC, to obtain live feedback on the state of safety functions.

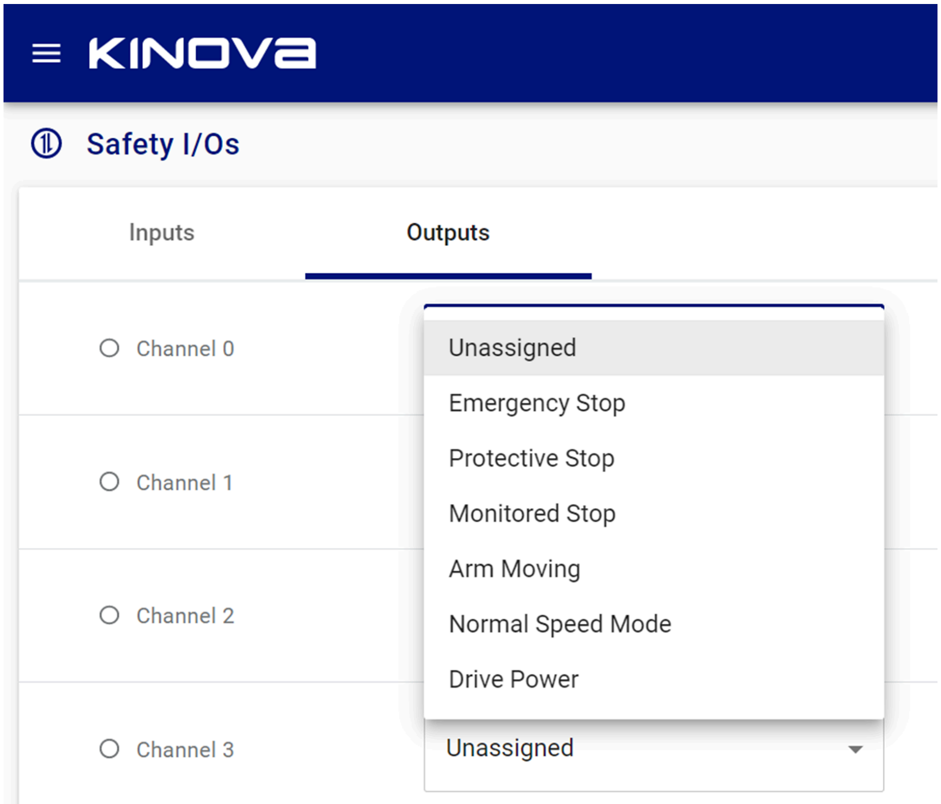


Figure 64: Safety output functions

Table 57: Output safety conditions available for each channel

Safety output function name	Description
Unassigned	No output safety function is assigned for the connected equipment. **Important:** When a factory reset is performed, any channel assigned to a safety output channel becomes unassigned. The output channels need to be reassigned.
Emergency Stop	The output condition is set only when it is triggered by pressing an *E-stop* button; the safety function remains active even when the robot arm is switched off. After the *E-stop* button is released and the fault is cleared, the safety function is inactive.
Protective Stop	The output condition is set when the robot is currently in a Protective Stop.

Safety output function name	Description
Monitored Stop	The channel signals the robot is waiting for user input; it makes sure there is no movement in any joint of the arm. The output condition is set only when the robot in the Monitored Stop operating mode.
Arm Moving	The channel signals the robot that the arm is currently moving, as is the case when the robot is in Hand Guiding, Manual Jog, Hold-to-Run, or Automatic mode.
Normal Speed Mode	The channel signal is raised when the robot is operating using the **Normal Speed Safety Mode** limits. It is triggered whenever the Normal Speed Mode is set and the Reduced Mode Safety function is not set.
Drive Power	The channel signal is raised when the actuator drives receive power.

Safety outputs set redundant hardware channels. For example, when the safety output function Emergency stop is assigned to channel 3, it actually is assigned to both AO_3 and BO_3. By using redundant hardware channels, load capacity, and the speed of switching between supplying and absorbing electricity is maximized.

Important: Although you can configure the channels to use the safety output functions, you cannot configure the hardware channels independently. For example, you cannot configure AO_3 and BO_3 to do different things.

Transitioning between giving and taking current is performed within 20 ms of triggering an output function. Any discrepancy between the channels during transitions does not exceed 5 ms.

Related topics

Installing the drive power indicator on page 97

State timelines of safety outputs

The majority of the state timelines of safety outputs are described through the different safety functions. However, there are two safety outputs that are not described by the safety functions.

Each safety output enters one of two states during operations.

A change in the state of the arm during different modes of operation dictate the state of each of the safety outputs. However, the Normal Speed Mode safety output may also depend on the selected Safety Mode.

Table 58: Safety output states that are described as safety function

Safety output state	Corresponding safety function
Unassigned	n/a
Emergency Stop	SF01 Emergency Stop
Protective Stop	SF02 Protective Stop
Monitored Stop	SF05 Monitored Stop
Arm Moving	n/a
Normal Speed Mode	n/a

Note: The safety output state **Unassigned** indicates that no safety output is assigned to the channel.

Related topics

SF01 Emergency stop timeline on page 63

SF02 Protective stop timeline on page 64

SF05 Monitored stop timeline on page 68

State timelines of safety functions on page 62

State timeline for the safety output Arm Moving

Each safety output enters one of two states during operations; it depends on the current state of the arm and the current mode of operation.

Equally important in knowing the timeline of a safety output is knowing where to find the current safety output status. To know the status of the safety output **Arm Moving**, look at what is highlighted in the Robot control panel or the robot status button in the robot status area. The current operating mode reflects the current status of Arm Moving.

Figure 65: Robot control panel

Figure 66: Robot status button

Table 59: Arm Moving timeline

Monitor frequency of trigger	0.02 s

Trigger off	The robot must be in one of the arm states to exit the Arm Moving state. • Initialization • Idle • Maintenance • Brake Release • Fault power off • Fault power on Alternatively, the robot must be in the operating mode Monitored Stop.
Trigger on	The arm must be operational and the robot must be in one of the operating modes to enter the Arm Moving state. • Automatic • Hand Guiding • Hold-to-Run • Manual Jog

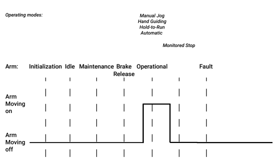


Figure 67: Timeline for the safety output Arm Moving

State timeline for the safety output Normal Speed Mode

Each safety output enters one of two states during operations.

Safety Mode is defined as either **Normal** speed or **Reduced** speed. To know the status of the **Normal Speed Mode**, look at the **Safety Mode** drop-down in different areas of Kortex Web App.

- **Safety** > **Joint Limits** > **Speed Limits**
- **Safety** > **TCP & Elbow Limits** > **TCP Limits**
- **Safety** > **TCP & Elbow Limits** > **Elbow Limits**
- **Programs** > **<select a program>**: the Program speed control area of the Robot control panel

Table 60: Normal Speed Mode timeline

Delay of trigger	0.02 s
Trigger off	• **Reduced** is selected from the **Safety Mode** drop-down or in the Program speed control of the Robot control panel. • The safety system is set to **Reduced**.
Trigger on	• **Normal** is selected from the **Safety Mode** drop-down or in the Program speed control of the Robot control panel. • The safety system is set to **Normal**. • The safety system is transitioning from **Reduced** to **Normal**.

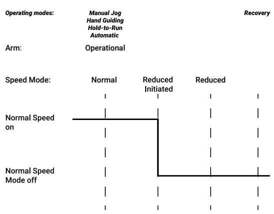


Figure 68: Timeline for the safety output Normal Speed Mode

Related topics

SF04 Joint speed monitoring timeline on page 67

SF06 Protection zone monitoring timeline on page 69

SF07 TCP and Elbow speed monitoring timeline on page 71

Installing the drive power indicator

The drive power indicator is a light that indicates that the drive power in the actuator is on.

About this task

The drive power indicator connects through the green channel slots on the *controller*. To conform to UL1740 requirements, the indicator must illuminate yellow or amber.

Procedure

1. Choose either a yellow or amber light to conform to UL1740 requirements.
2. Connect the indicator to the controller through the green slots on the *controller*.

   Tip: Pay attention to which green slot is used physically to connect the indicator. You have to use the same numbered channel in Kortex Web App.
3. Tap **Safety**.

4. Tap **Drive Power** from the list of outputs to the channel on which the indicator is physically connected to on the controller.

Results

The indicator illuminates when the drive power is on.
Related topics
Safety outputs on page 91

Safety checksums

The *administrator* needs a quick way to look at the robot and walk away with peace of mind knowing the safety configurations he put in place are working as expected.

The safety checksum is a human readable hexadecimal value displayed beside the Notification icon. It is the value calculated from the *SCU* and the *MPU*, using *Cyclic redundancy check (CRC)*, on all safety parameters.

Whenever the arm is switched on, the *MPU* safety parameter checksum is compared with the *SCU* safety parameter checksum. When a safety parameter is changed, a new checksum is calculated by the *MPU* and compared against the *SCU* checksum.

The administrator can look at the safety checksum and know whether someone has modified any of the safety configurations. They can know from a glance whether any safety configurations are failing. They can write configuration build instructions for their environment and list the checksum for other administrators to know the valid safety checksum.

If the safety parameter checksums do not match, the robot enters a fault state in which the administrator must manually correct all safety parameter values. This task is not time-consuming provided there is a configuration build instruction available that includes the valid safety checksum value.

Parameters that are the basis for the safety checksum

The calculation of the safety checksum is based on the configuration of the safety functions, the state of each of the safety functions, and calibration files.

Table 61: Checksum parameters

Safety parameter	Description
Configuration of safety inputs	Binding between safety input channels and the configured function Example: Channel 1 is assigned to the function **Enabling device**.
Configuration of safety outputs	Binding between safety output channels and the configured function Example: Channel 4 is assigned to the function **Monitored stop**
SF03	Safety joint position limit monitoring for joints 1 to 7
SF04	Safety joint speed monitoring for reduced speed limits for joints 1 to 7 Safety joint speed monitoring for normal speed limits for joints 1 to 7
SF05	Monitored stop has no parameters
SF06	Protection zone monitoring • Position of all configured protection zones • Shape type (sphere, cylinder, rectangular prism) • Dimensions of all configured protection zones • Orientation of all cylindrical and prismatic protection zones • Activation status: enabled or disabled Tool Sphere • Position (X,Y,Z) • Dimension

Safety parameter	Description
SF07	Safety TCP monitoring • Translation speed normal speed mode limit • Orientation speed normal speed mode limit • Translation speed reduced speed mode limit • Orientation speed reduced speed mode limit Safety Elbow monitoring • Translation speed normal speed mode limit • Translation speed reduced speed mode limit

Arm calibration

Microscopic assembly defects are compensated by the firmware of the robot when the arm is calibrated.

The calibration package contains the default calibration file associated with the serial number found on the arm, its checksum, and the serial number of the arm.

The calibration file contains a few vital pieces of information.

- Calibration date
- Serial number of the arm
- Geometric calibration parameters
- Elasto-static calibration parameters

Access the status of the arm at **Systems** > **Robot** > **Arm** > **Calibration.**

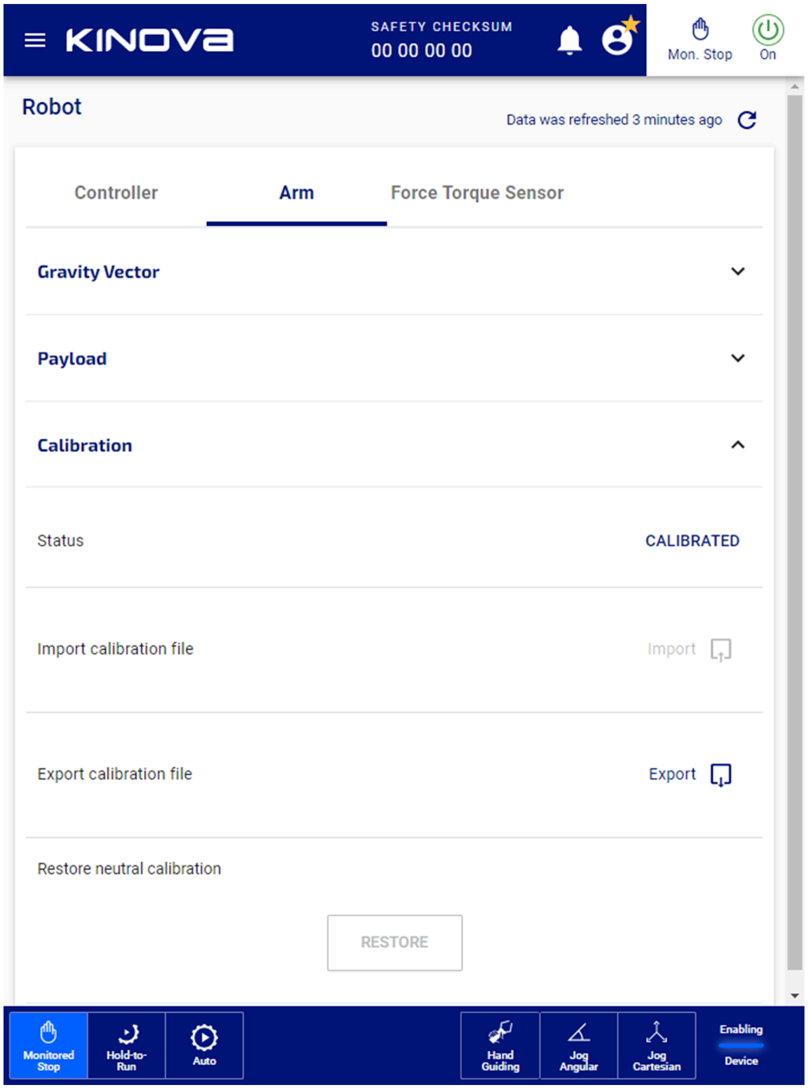


Figure 69: Arm calibration status

The arm calibration can be in one of three states.

Table 62: Arm calibration states

State	Description
Not Calibrated	The controller is switched on and validation of the calibration file is not yet started. The neutral calibration package is selected and used. The neutral calibration package is used because the validation of the selected calibration package fails.

State	Description
Waiting Arm Power On	The controller is switched on and validation of the calibration file is successful. The serial number does not match the serial number in the calibration file; the arm cannot power on. **Tip:** To recover from this situation, restore the neutral calibration file and reboot the controller.
Calibrated	The controller is switched on, the calibration file is valid, and the arm serial number is valid.

Exporting calibration files

When you first switch on your Link 6, save the default arm calibration package. Use the exported arm calibration package to communicate with Kinova support if anything goes wrong when you first start your robot.

About this task

If you have multiple Link 6 units, you may want tot save a backup calibration package for each unit. Each backup calibration package can be re-uploaded on controllers if you end up inadvertently matching an arm with a controller that is not its original controller.

The default arm calibration file is also known as the neutral arm calibration file. It can be used to back up the default calibration for a specific arm.

Important: Only an administrator can export calibration files.

Procedure

Procedure

1. Insert a *USB* into the controller.
2. Tap **Systems** > **Robot** > **Arm** > **Calibration.**.
3. Tap **Export** in the**Export calibration file** pane.
   A **Select mount to export** dialog launches with the name of the USB drive.
4. Tap **EXPORT**.

   Note: Tap **CANCEL** when you do not want to export the current calibration package.

Results

The current arm calibration is saved as a zip file on the USB drive.

What to do next

Remember to remove the USB from the controller and store the USB in a safe place.

Importing arm calibration files

You may want to calibrate the arm with the default neutral file, the last saved file, or a completely different calibration file.

Before you begin

Kinova keeps archives of the calibration package of each unit. The calibration package contains the data obtained during calibration in our factory. If you misplace the calibration file of your arm, contact support@kinova.ca to provide you with the original calibration package that matches your arm.

About this task

All imported calibration packages are verified before storing it in the controller.

Important: Calibration packages can be imported only when the controller is idle and the arm is switched off.

Important: Only an administrator can import calibration files.

Procedure

Procedure

1. Insert a *USB* into the controller.
2. Tap **Systems** > **Robot** > **Arm** > **Calibration.**.
3. Tap **Import** in the **Import calibration file**.
   A dialog launches with a list of all arm calibration packages on the USB drive.
4. Tap the name of the calibration package that you want to import.
5. Tap **PROCEED**.

   Note: Tap **CANCEL** when you do not want to export the current calibration package.

Results

The selected arm calibration is saved in the controller.

Note: If, for any reason, the imported calibration file cannot validate, the neutral calibration file is installed and the arm is listed as being Not Calibrated.

What to do next

Verify whether the arm calibration package imported successfully.

- Tap **Systems** > **Robot** > **Arm** > **Calibration**.
  - Success: the arm is listed as **Calibrated**
  - Failure: the arm is listed as **Not calibrated**
- **Diagnostics** > **Status** > **Controller** and check the status of **Serial number mismatch between arm and calibration file**.
  - Success: the status icon is green with a check mark
  - Failure: the status icon is red

Note: When the arm is listed as **Not Calibrated** and the status is green, the neutral arm calibration package is running. All movements are less accurate.

Preventing an unrecoverable fault

If you exchange a controller or arm, the robot may enter a fault state.

About this task

Link 6 has a serial number. The calibration file is associated with the serial number. A change in parts means a change in serial number. A change in serial number means the calibration file cannot function on the new parts and the system enters a fault state.

Unrecoverable faults due to exchanges of parts are preventable.

Procedure

1. Make sure you have a copy of the current configuration file for the old arm and controller.

   If you cannot find the calibration file for the robot before exchanging parts, export the calibration file.

   Tip: If ever you put the old arm back in place, you already have its specific calibration ready to go.
2. Switch off the arm and the controller of Link 6.
3. Exchange the arm or controller.
4. Switch on the controller and keep the arm switched off.
5. Perform one of two actions.
   - Restore the calibration file for the arm by tapping **Systems** > **Robot** > **Controller** > **RESTORE**.
   - Import the correct calibration file by tapping **Systems** > **Robot** > **Controller** > **Import** and selecting the correct file.

6. Switch on the arm.

Results

The robot is operational.

Related topics

Configuring the arm for robot controls on page 187

Importing arm calibration files on page 103

Exporting calibration files on page 102

Recovering from an unrecoverable fault on page 105

Recovering from an unrecoverable fault

If you exchange a controller or arm, and the calibration file is associated with the old controller or arm, you can still recover from the fault.

About this task

It is assumed that the steps in preventing the fault from occurring were accidentally omitted when the parts were exchanged.

It is assumed Link 6 is running.

Procedure

1. Switch off the arm.
2. Switch off the controller.
3. Switch on the controller.
4. Perform one of two actions.
    - Restore the calibration file for the arm by tapping **Systems** > **Robot** > **Controller** > **RESTORE**.
    - Import the correct calibration file by tapping **Systems** > **Robot** > **Controller** > **Import** and selecting the correct file.
5. Switch on the arm.

Results

The robot is operational.

Related topics

Configuring the arm for robot controls on page 187

Exporting calibration files on page 102

Preventing an unrecoverable fault on page 104

Modes of operation of Link 6

The robot enters different *states* and *modes* during operation. The states often have a visual component for the user to recognize. The modes, which is how the user interacts with the robot, have a visual component for the user to recognize.

The basic modes of operation require user intervention.

- Startup
- Operation
- Recovery

Within each of the basic modes are internal modes and states.

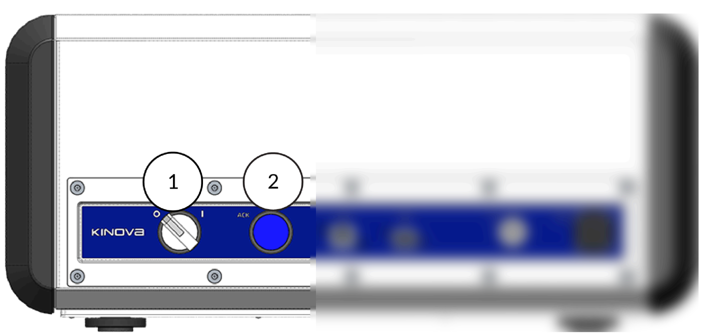
There are three places to look for cues to determine what state the robot is in.

- *Controller*
- *Wrist*
- *Kortex Web App*

Location of lights to watch on the controller and the wrist

Communication between the robot and the user is achieved by lights found on the controller and around the *wrist ring*.

Table 63: Controller and arm when Link 6 is off



1	Power knob	3	Wrist ring
2	ACK button		

Convention for the representation of the states

In an attempt to help visualize what is happening, the graphics follow a convention for flashing lights. In every case, the number of lines represent the number of flashes of light for each second.

- A single line coming out the sides of what is supposed to be the lit portion of equipment indicates one flash for each second. The color of the lines represents the color of the lights.

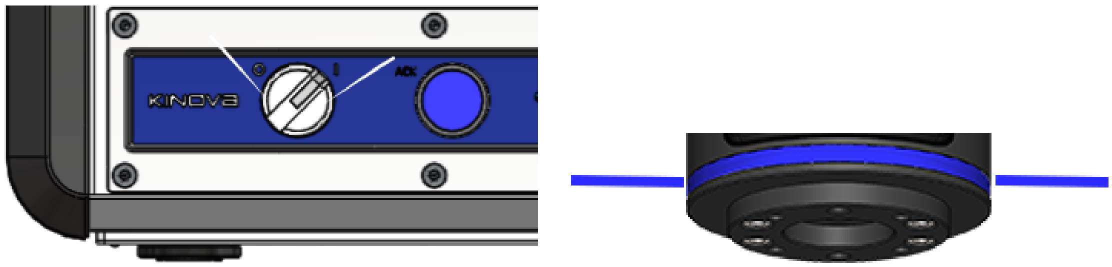


Figure 70: Example of equipment flashing once per second

- Two lines coming out the sides of what is supposed to be the lit portion of equipment indicates two flashes for each second. The color of the lines represents the color of the lights.

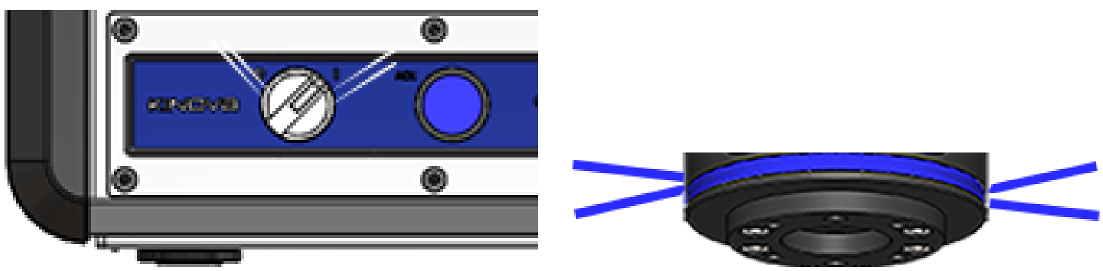


Figure 71: Example of equipment flashing twice per second

There also is a table that explains what each portion of a graphic is attempting to represent.

Complete start-up sequence

It is a two-part process to start up Link 6: first turn on the controller and then the arm. This two-part process means the robot goes through six different *states*.

Off — Controller initialization — Controller idle — Arm initialization — Arm maintenance — *Monitored stop*

State: Off

Link 6 is off and is not energized when the power button on the controller is switched off.

Before you start working with the arm, make sure it is off. When it is off, all lights are extinguished.

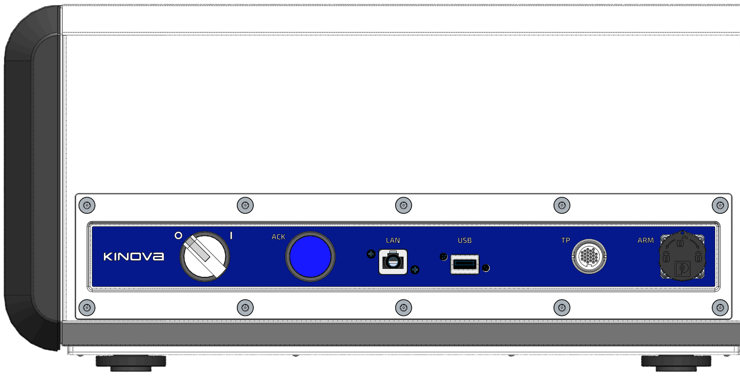


Figure 72: Controller when it is off

Figure 73: Wrist ring when it is off

Important: If there is a light on the controller power knob, turn off Link 6 by turning the power knob to the off position.

Controller lights	Wrist ring	Description
Power knob points to the O, or off, position.	No light illuminates from the wrist.	Arm does not receive power.
No light illuminates on the controller power knob.		
ACK button on the controller does not illuminate.		

The startup workflow begins with switching on the controller and switching on the arm.

State: Controller initialization

When the knob on the *controller* is turned to the I, or on, position the controller is in its initialization state.

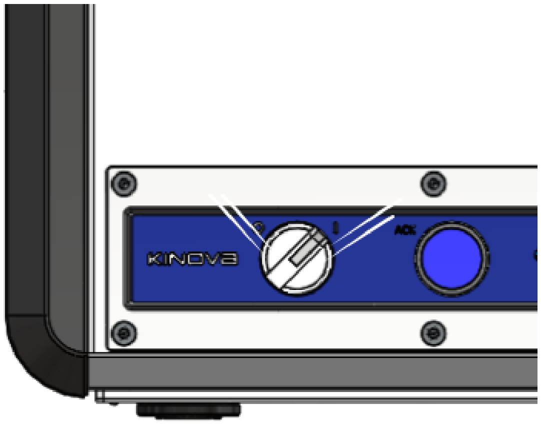


Figure 74: Visual cues for controller initialization

Controller lights	Wrist ring	Description
Power knob points to the I, or on, position.	No light illuminates from the wrist.	Controller receives power.
Power knob on the controller flashes white twice (2) each second.		Processors in controller *boot*.
ACK button on the controller does not illuminate.		Application software for the I/O boards and teach pendant, if installed, start.
		Calibration, configuration, and safety checks are validated.

When the light behind the power button is a steady white, the initialization of the controller is complete. The controller enters a new state: Idle.

State: Controller idle

When Link 6 is in the idle state, there is power in the *controller*, but not in the *arm*.

You can recognize the Idle state by looking at the power button on the controller; its light is a steady white.

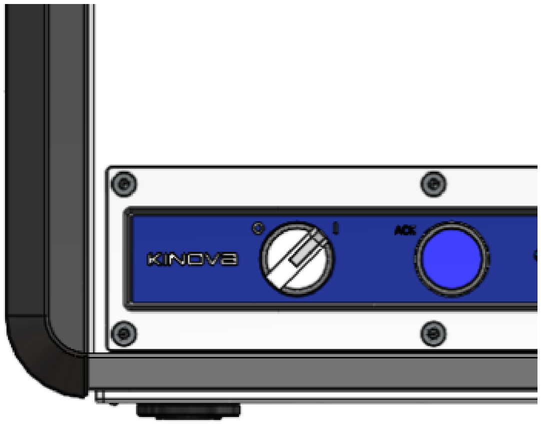


Figure 75: Controller when it is idle

Controller lights	Wrist ring	Description
Power knob on controller illuminates white steadily.	No light illuminates from the wrist.	Controller is fully operational and idle.
ACK button on the controller does not illuminate.		Arm is not on.

State: Arm initialization

After you turn on the *arm* from *Kortex Web App*, the arm needs to initialize itself, much like the *controller*.

Arm initialization is recognized by looking at the white light of the power knob on the controller and at the light around the wrist.

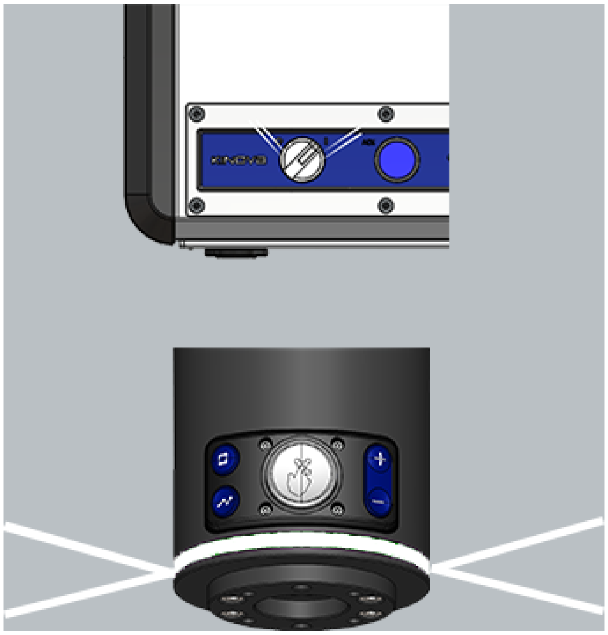


Figure 76: Visual cues of arm initialization

Controller lights	Wrist ring	Description
Power knob on the controller flashes white twice (2) each second.	Wrist ring flashes white once (1) each second.	Arm is receiving power.
		Arm is *booting* up the *actuators* and is checking the *firmware* and software are compatible with the arm.
Power knob on the controller flashes white twice (2) each second.		Kortex Web App displays the state as **Initialized**.

Important: The wrist ring flashes white when the arm is switched on to ensure the functionality of all components of the wrist ring light. White uses all components of the wrist ring lights; it can be used to validate the functionality of the hardware, in accordance with IEC60204. If the wrist ring is not white, report the situation immediately to support@kinova.ca.

If the hardware needs a firmware update, the arm enters the Arm maintenance state and returns to the Arm initialization state after maintenance is complete. Firmware updates take approximately 5 minutes.

After the arm is initialized and the firmware is up-to-date, the brakes in the actuators release.

The brakes in each actuator for each joint are released one at a time in a specific sequence. The sequential release keeps the robot steady when it is powering up.

State: Arm maintenance

Maintenance only happens when the *firmware* must be updated.

After the firmware is updated, the robot restarts the Arm initialization process to make sure there are no other issues.

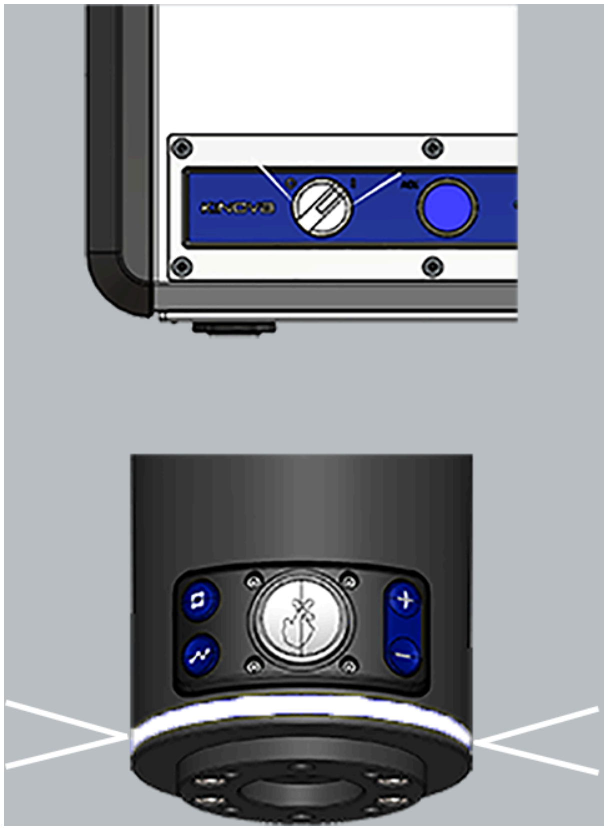


Figure 77: Visual cues of arm maintenance in progress

Controller lights	Wrist ring	Description
Power knob on the controller flashes white once (1) each second.	Wrist ring flashes white twice (2) each second.	*Microcontrollers* on the *actuators* receive firmware updates.
ACK button on the controller does not illuminate.		*Kortex Web App* displays the state as **Initialized**.

After firmware is updated, the Arm initialization state restarts.

State: Monitored stop

Link 6 is ready to be used and is in a standstill position; it is waiting for user input.

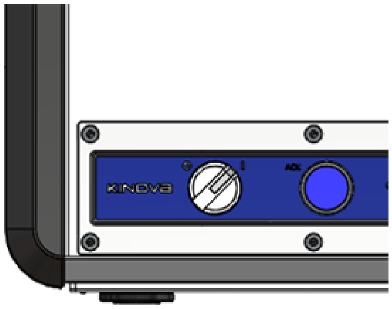


Figure 78: Visual cues of robot in *Monitored stop*

Controller lights	Wrist ring	Description
Power knob on controller illuminates white steadily.	Wrist ring illuminates blue steadily.	Mechanical brakes in the *arm* are released.
ACK button on the controller does not illuminate.		The arm is fully operational and is waiting for commands.

Related topics
Stop categories on page 53

State: Fault

Link 6 can stop unexpectedly or may need to stop suddenly.

There are a few main ways the robot can stop suddenly.

- The *arm* exceeds the configured safety limits for position; a *protective stop* has been issued to the arm.
- Someone presses the *emergency stop* button.
- A fault is detected. Faults are failures in internal, built-in tests and safety functions.

The *SCU* is responsible for triggering a *stop category* 0 fault; no power is in the arm

The *Main Control Unit (MCU)* is responsible for triggering the fault; the *wrist ring* illuminates red.

Figure 79: Visual cue of fault without power

Figure 80: Visual cue of fault with power

Controller lights	Wrist ring	Description
Power knob on the controller flashes white four (4) times each second.	No light illuminates from the wrist.	An internal fault occurred.
ACK button on the controller does not illuminate.		There is no power in the arm.
Power knob on the controller flashes white four (4) times each second.	Wrist ring flashes red twice (2) each second.	An internal fault occurred.
ACK button on the controller does not illuminate.		There is power in the arm.

Related topics

Recovery mode on page 122

Why can the robot not exit Recovery mode? on page 352

Operating modes

When the *controller* and *arm* are on, Link 6 is in the Monitored Stop state and can be used in any of the modes of operation available to it.

Each mode of operation is a way in interacting with the robot to make it move.

- Manual Jog
- Hand Guiding
- Hold-to-Run
- Automatic

Related topics
Stop categories on page 53

Mode: Manual Jog

Use the mode Manual Jog to move the *arm* with *Kortex Web App*. There are two jog interfaces: *Cartesian* control and joint, or *angular*, control.

There are a few requirements before you start moving the arm.

- Use the *teach pendant enabling device*. If the *pendant* is not available, use an external *enabling device* connected to the *controller*. The enabling device must be pressed down and held in its center position.
- There must be a request for movement coming from Kortex Web App.

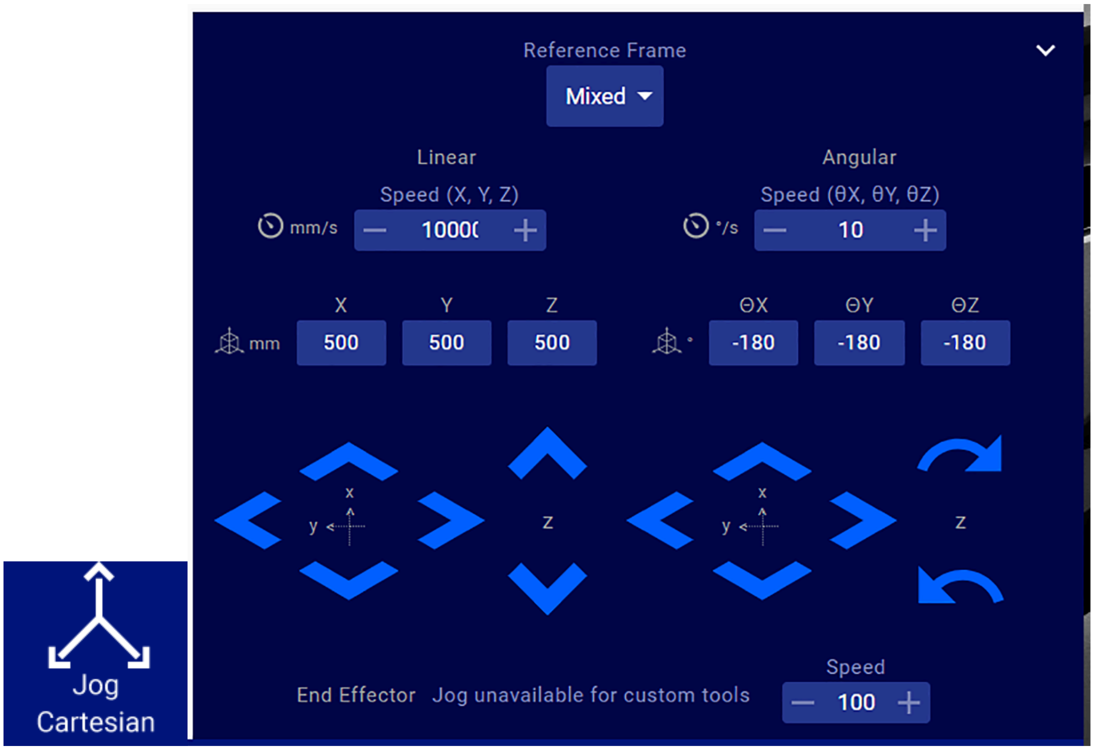


Figure 81: Jog Cartesian icon and control panel

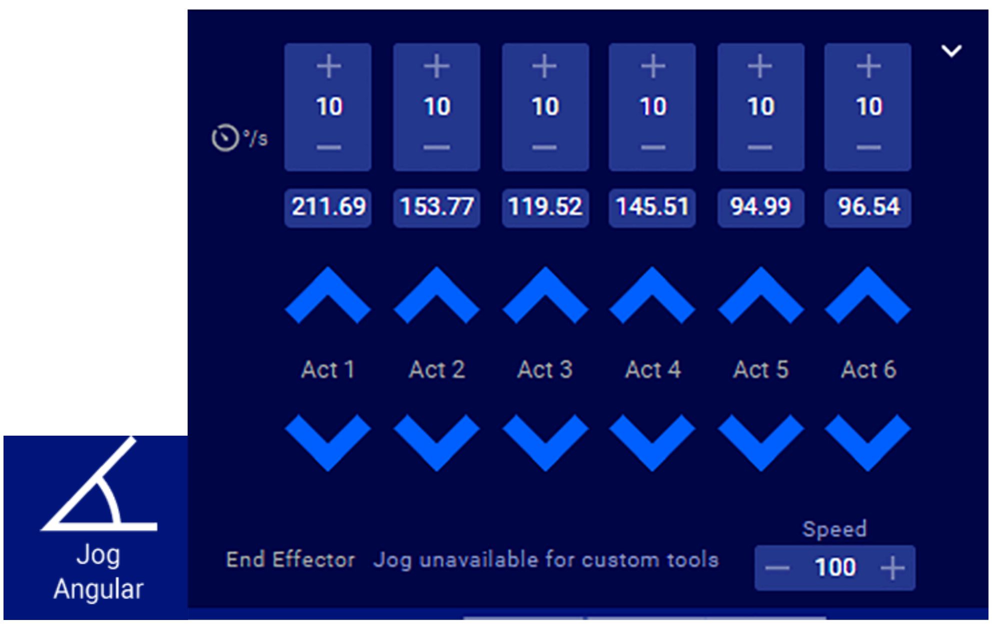


Figure 82: Jog angular icon and control panel

1 Enabling device

[2]

Figure 83: Location of enabling device on the teach pendant

When the robot is in the mode Manual Jog, there is a steady, white light behind the power knob on the controller and a steady, blue light on the wrist.

Figure 84: Visual cues of Manual mode

Controller lights	Wrist ring	Description
Power knob on controller illuminates white steadily.	Wrist ring illuminates white steadily.	Move the arm in any direction with Kortex Web App while holding the pendant enabling device in the pressed position.
ACK button on the controller does not illuminate.		

When the robot is in Manual mode, Link 6 operates strictly in Reduced safety speeds to reduce the possibility of injury.

Exit Manual jog mode by releasing the enabling device. Link 6 returns to the *Monitored stop* state.

Related topics

Safety modes on page 61

Jog Angular panel on page 289

2 Disclaimer: the color of the hardware in the image may differ from your installation

Jog Cartesian panel on page 287

Mode: Hand Guiding

Use Hand Guiding mode to move the robot to a specific location by using one hand.

While keeping the *wrist enabling device* pressed in the middle position, move the *arm* up or down, left or right, forward or backwards, as well as rotate the arm or the wrist around any of the joints.

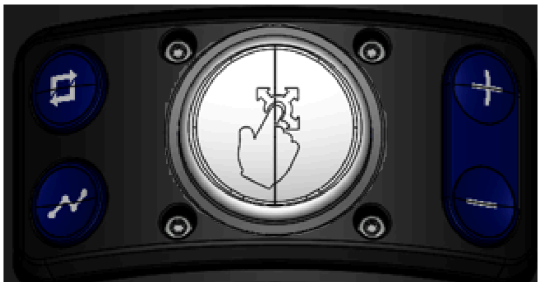


Figure 85: Center button on the wrist is the wrist enabling device button

The arm stops moving when the wrist enabling device is not pressed or when it is pressed completely.

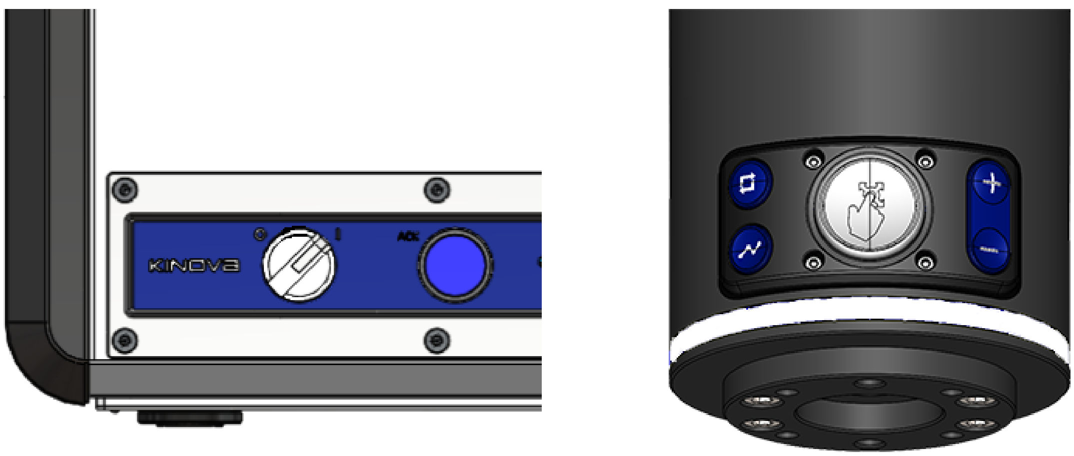


Figure 86: Visual cues of the Hand Guiding mode

Controller lights	Wrist ring	Description
Power knob on controller illuminates white steadily.	Wrist ring illuminates white steadily.	Move the arm in any direction while holding the wrist enabling device button in the pressed position.
ACK button on the controller does not illuminate.		

Within the Hand Guiding mode are two different Guidance modes.

- *Cartesian*

  Warning: Cartesian Hand Guiding uses sensors situated internally to the wrist. To avoid unintentional movement, it is recommended to manipulate the arm from the wrist only during Cartesian Hand Guiding.

- *Angular*

Enable Hand Guiding mode by pressing the *wrist enabling device*. Press the Hand Guiding mode Toggle button to enable one of the two Hand Guiding modes.

Figure 87: Location of Hand Guiding mode Toggle on the wrist

When the robot is in Hand Guiding mode, Link 6 operates strictly in Reduced safety speeds to reduce the possibility of injury.

Related topics

Safety modes on page 61

Hand Guiding panel on page 285

Mode: Hold-to-Run

Hold-to-Run is a special mode available only to users with *admin* privileges.

The Hold-to-Run mode is used to develop and verify a programmed sequence works as expected before the program is release to other users.

Hold-to-Run is also used for troubleshooting a situation with the robot. Robotic movement can be slowed during this mode to ease observing the actual actions in the programmed sequence.

To be able to move the *arm*, an *enabling device* must be on.

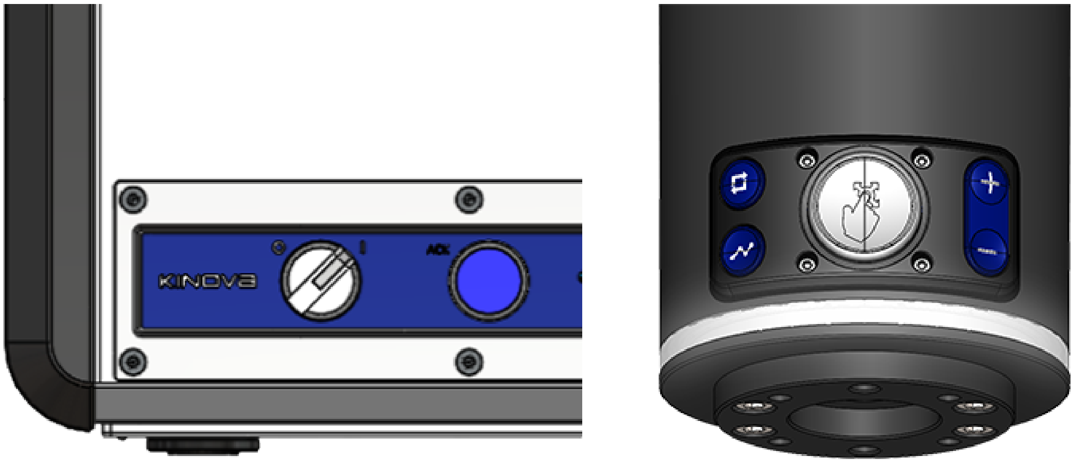


Figure 88: Visual cues of the mode Hold-to-Run when there is no issue

Controller lights	Wrist ring	Description
Power knob on controller illuminates white steadily.	Wrist ring illuminates white steadily.	The selected program is ready to be verified.
ACK button on the controller does not illuminate.		

When the robot is in Hold-to-Run mode, Link 6 can operate in either Reduced or Normal safety speeds. By default, the robot operates in Reduced safety speeds to reduce the possibility of injury.

DANGER: Do not attempt to use the wrist buttons when a program is running and the robot is in Hold-to-Run mode; it can result in serious injury.

Related topics

User management on page 161

Running a program in Automatic mode on page 295

Safety modes on page 61

Running a program in Hold-to-Run mode on page 296

Mode: Automatic

Known as the normal mode of operation, it is used to play a *program sequence* that has been validated.

When the robot is in Automatic mode, Link 6 can operate in either Reduced or Normal safety speeds. By default, the robot operates in Normal safety speeds.

One of the first things the robot does when Automatic mode is selected is to activate all safeties, even those that are suspended due to an event or user interaction.

Note: Automatic mode is intended for the *operator*.

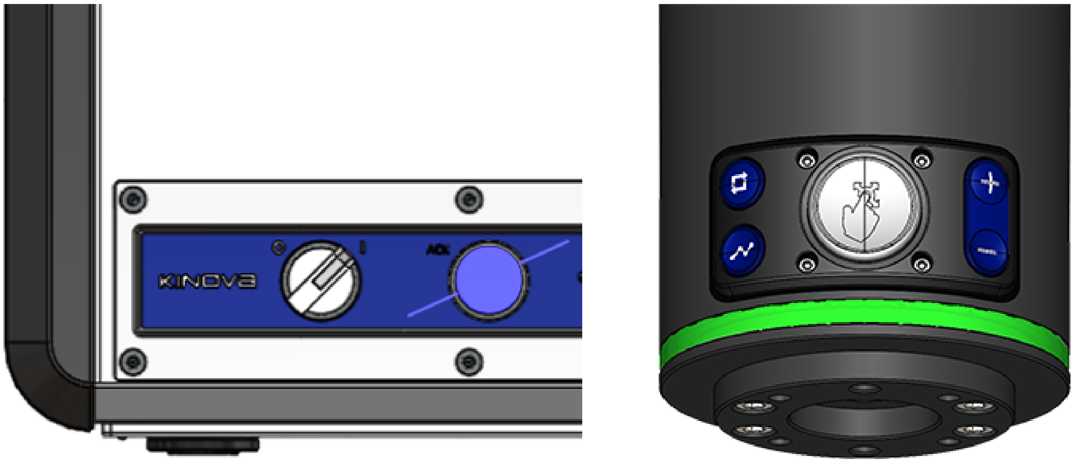


Figure 89: Visual cues for the Automatic mode

Controller lights	Wrist ring	Description
Power knob on controller illuminates white steadily.	Wrist ring illuminates green steadily.	The selected program is ready to be run.
ACK button on the controller flashes blue once (1) each second.		

When the robot is in Automatic mode and the *E-stop* is pressed, the robot stops automatically and no power feeds the robot.

When the robot is in Automatic mode and the robot enters a protective stop, the robot stops. However, once the issue is corrected and the ACK button is pressed, the robot can resume the program.

DANGER: When the robot is operating in Automatic mode, never attempt to control the robot by grabbing the arm or touching the controls on the wrist. Attempting to control the robot when it is moving in Automatic mode can result in injury.

Tip: Control the robot safely when it is not moving.

Related topics

Safety modes on page 61

Running a program in Automatic mode on page 295

Running a program in Automatic mode with acknowledgment
Canceling a program running in Automatic mode

Recovery mode

Some faults stop the robot in a state that triggers the same fault after it has been cleared. These are the faults that require a recovery strategy for the robot.

There are several ways a fault can be cleared and leave the robot in a fault state. One example is a robot enters a *protection zone* and stops. The fault is cleared using *Kortex Web App*, but the *arm* is still within the protection zone.

There is only one way to get out of recovery mode. Move the robot to a safe position using either Hand Guiding or Manual Jog. All position-related safeties are ignored during recovery. All joint speed safety limits are limited to a maximum speed of 30 °/s.

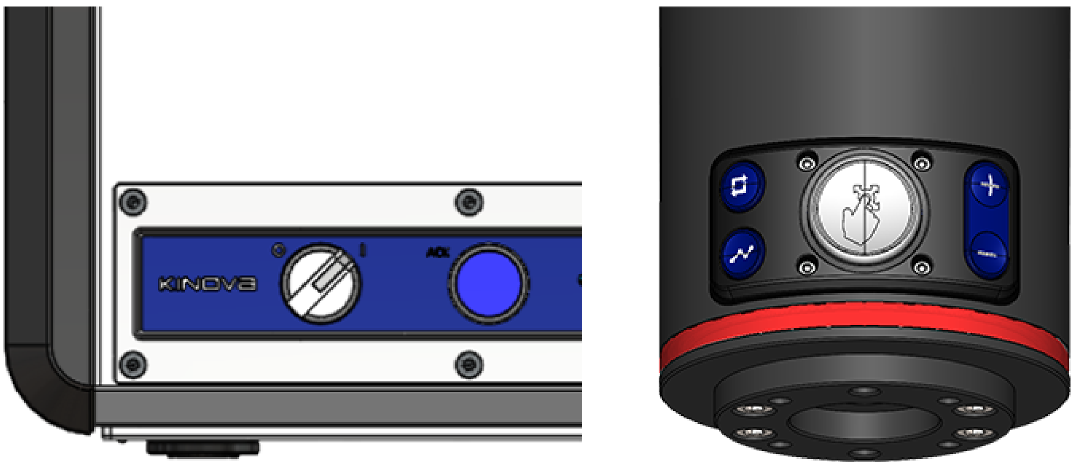


Figure 90: Visual cues of robot in Recovery

Controller lights	Wrist ring	Description
Power knob on controller illuminates white steadily.	Wrist ring illuminates red steadily.	The robot requires user intervention that involves using *Kortex Web App* and the *wrist enabling device*.
ACK button on the controller does not illuminate.		

Related topics

State: Fault on page 113

Why can the robot not exit Recovery mode? on page 352

Robot lights and what they mean: a reference

You can tell which mode or state the robot is currently in by looking at the lights on the *controller* and on the *wrist ring* on the *arm*.

Table 64: What the lights on the controller and the wrist ring on the arm mean

Description	Meaning
Power knob points to the O, or off, position.	Controller is off.
ACK button on the controller does not illuminate.	
Power knob on the controller flashes white twice (2) each second.	Controller is powering on.
	Processors in controller *boot*.
ACK button on the controller does not illuminate.	Application software for the I/O boards and *teach pendant*, if installed, start.
	Calibration, configuration, and safety checks are validated.
Power knob on controller illuminates white steadily.	Controller is fully operational and idle.
ACK button on the controller does not illuminate.	Arm is not on.
Power knob on the controller flashes white twice (2) each second.	Arm is powering on.
Power knob on the controller flashes white twice (2) each second.	
ACK button on the controller does not illuminate.	Controller checks whether the *firmware* in the arm matches the firmware in the software unit in the controller.
Wrist ring illuminates white steadily.	*Kortex Web App* displays the state first as **Initialized** and then as **Brakes Released** during this stage unless a firmware updates is required. If firmware updates are required, Kortex Web App displays first **Initialized**, then **Maintenance**, and then **Brakes Released**.
Power knob on the controller flashes white once (1) each second.	Firmware updates. **Important:** Firmware in the arm updates right after powering on the arm only when an update occurs in the software unit in the controller.
ACK button on the controller does not illuminate.	
Power knob on the controller flashes white twice (2) each second.	Kortex Web App displays the state as **Maintenance**.

Description	Meaning
Power knob on controller illuminates white steadily.	The arm is fully operational and is in *Monitored stop*.
ACK button on the controller does not illuminate.	Kortex Web App displays the state first as **Brakes Released** and then as **Operational**.
Wrist ring illuminates blue steadily.	
Power knob on controller illuminates white steadily.	Link 6 is operating in Hold-to-Run mode or is waiting for confirmation to operate in Automatic mode.
ACK button on the controller flashes blue once (1) each second.	
Wrist ring illuminates white steadily.	
Power knob on controller illuminates white steadily.	The robot is operating either in Hold-to-Run mode or in Automatic mode.
ACK button on the controller flashes blue once (1) each second.	The program is running and the *enabling device* has not been pressed for over five (5) minutes.
Wrist ring illuminates green steadily.	It is also possible that the configured speed is greater than the reduced mode speed.
Power knob on controller illuminates white steadily.	The robot is operating in Manual Jog mode.
ACK button on the controller does not illuminate.	Move the arm in any direction with Kortex Web App and with the enabling device pressed in the middle position.
Wrist ring illuminates white steadily.	
Power knob on controller illuminates white steadily.	The robot is operating in Hand Guiding mode.
ACK button on the controller does not illuminate.	Move the arm in any direction while holding the wrist enabling device button in the middle position.
Wrist ring illuminates white steadily.	
Power knob on controller illuminates white steadily.	The robot is waiting for confirmation to proceed operations in Automatic mode.
ACK button on the controller flashes blue once (1) each second.	
Power knob on controller illuminates white steadily.	The robot is in Recovery mode.

Description	Meaning
Wrist ring illuminates red steadily.	The arm of the Link 6 is outside the area in which it is allowed to be in, such as within a *protection zone*, or one or more of the joints is outside its safety limits.
Wrist ring illuminates red steadily.	
Power knob on controller illuminates white steadily.	The robot is recovering from an emergency stop caused by pressing the *E-stop* button.
ACK button on the controller illuminates blue steadily.	
No light illuminates from the wrist.	
Power knob on the controller flashes white four (4) times each second.	An internal fault occurred.
ACK button on the controller does not illuminate.	There is no power in the arm.
No light illuminates from the wrist.	
Wrist ring flashes white four (4) times each second.	An internal fault occurred.
ACK button on the controller does not illuminate.	There is power in the arm.
Wrist ring illuminates red steadily.	The joint or zone limit is reached.

Related topics

Indicators, user profile, and power on page 153

Hardware installation

Link 6 requires installation in the working environment.

Installation involves placing the *arm* where it is going to be performing tasks, placing the *controller* within the constraints of the length of the cable to the arm in an accessible location, and making sure the robot is installed according to safety guidelines.

Before unpacking, the *integrator* should already know the size of the robot from the technical specifications, have performed a risk assessment, and be familiar with the safety considerations.

Important: Always place the boxes so that the arrows on the boxes point up to avoid possible damage to the equipment.

Related topics

Safety directives and warnings on page 48

Inside the Link 6 boxes

Link 6 comes in two cardboard boxes. Both boxes are heavy. It is advisable to have two people carry each box.

The smaller box contains the *controller*, a box with the accessories, and a box with the *teach pendant*. The box with accessories contains the power cord for the controller, as well as all optional accessories bought with Link 6.

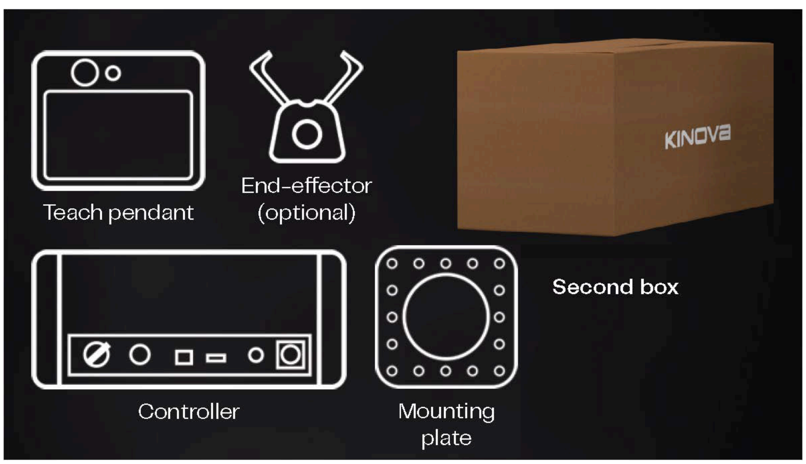


Figure 91: Box with controller, mounting plate, accessories box, and teach pendant box

The larger box contains the *arm*, along with its cable attached to the *base* of the arm.

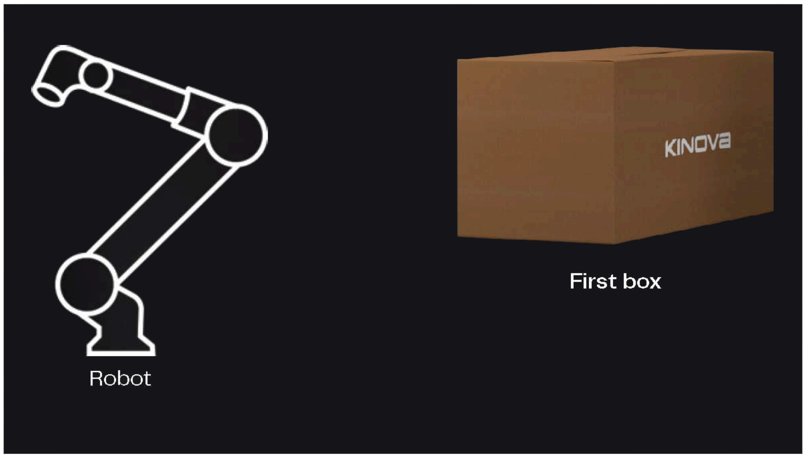


Figure 92: Box with arm and attached cable

Related topics

Accessories on page 333

Components on page 12

Unpacking the arm

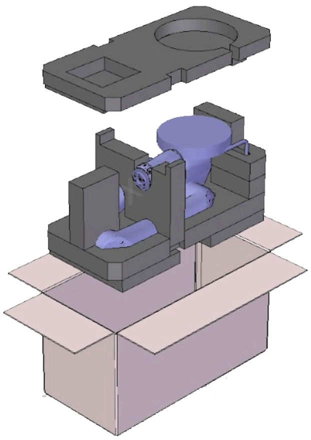
The box containing the *arm* is the larger of the two Link 6 boxes.

About this task

It is recommended to unpack the contents of the arm box at the installation site for the arm.

Procedure

1. Make sure the arrows on the box are pointing upward.
2. Open the larger box.
3. Lift the layer of foam off and set it aside.



4. Remove the arm from the box.

Considerations before installing the arm

The *arm* is heavy and large. It is advisable for two people to be involved in unpacking and installing the arm.

Make sure the box that contains the arm is near the area where the arm is to be installed. Have all the tools and screws necessary for installing the arm easily accessible at the mounting location.

Wherever the arm is installed, there is a set of guidelines that should be followed.

- Make sure all the tools and screws necessary to install the arm are easily accessible at the mounting location.
- Make sure the surface on which the arm is to be mounted is flat within 0.1 mm.
- Make sure the surface on which the arm is to be mounted is stiff enough so that it does not degrade the task to be performed by the robot.

  DANGER: When the mounting surface where the arm is not sufficiently sturdy, the arm could become a hazard to people and equipment around it.
- Make sure you know which installation approach is required: directly on a surface or on a mounting plate.
- Have a minimum of two (2) people for safe installation for any orientation installation that is not on a flat, horizontal surface.
- Make sure to adjust the gravity vector for any orientation installation that is not on a flat, horizontal surface.
- Use the mounting dowel holes when your installation requires the robot to be repositioned accurately at the same place.

  DANGER: The arm could be unstable until is fixed securely.

The arm must be kept within specific temperatures ranges during storage and use.

Table 65: Safe temperatures for the arm

Action	Minimum temperature	Maximum temperature
Storage	-40 °C	65 °C
Powering up	0 °C	40 °C
Operating	-20 °C	40 °C

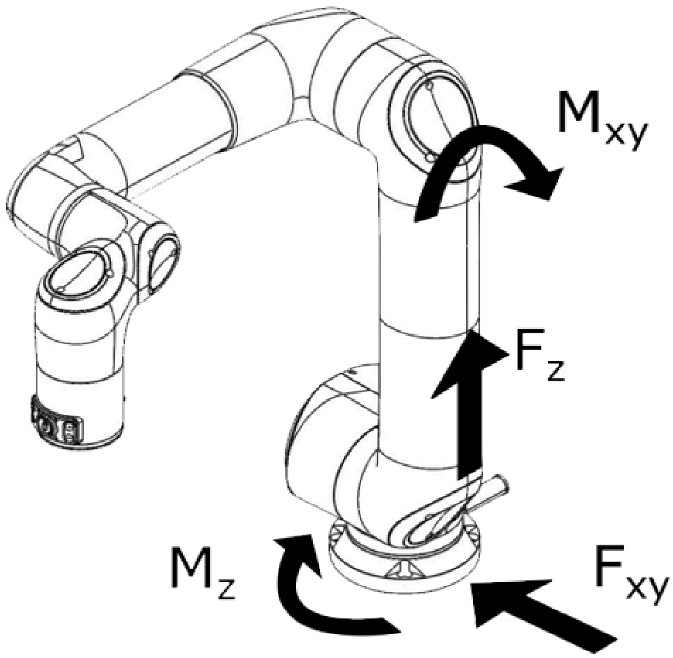
Related topics
Safety directives and warnings on page 48

Configurations for the robot settings on page 185

Base force reactions

The *base* feels the forces whenever any part of the *arm* moves.

The calculation includes the forces and moments created from the weight of the robot and its dynamic properties. The calculation does not include any of the equipment fitted on the robot.



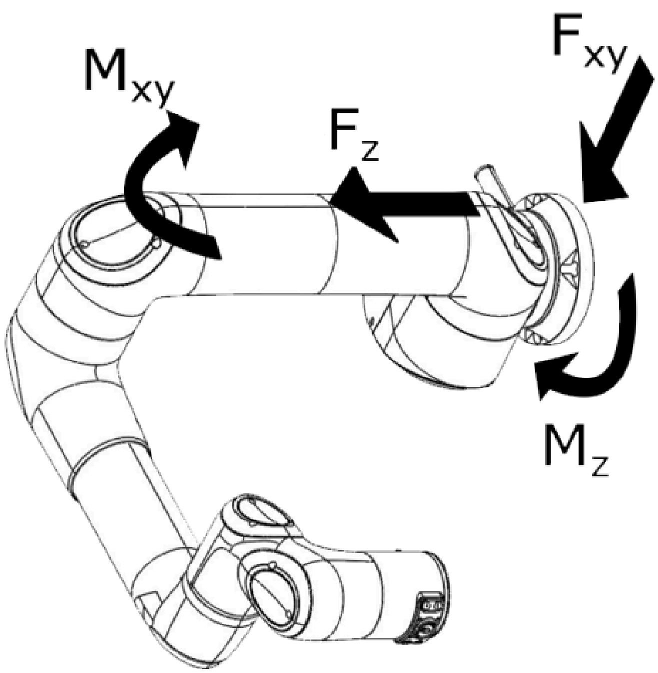
F_{xy}	Force in any direction along the XY plane
F_z	Force along the Z plane
M_{xy}	Moment that acts on the robot base along the XY plane
M_z	Moment that acts on the robot base along the Z plane

Figure 93: Floor-mounted safety forces

Table 66: Safety factor applied on load cases when the base is mounted on the floor

Load case details	Vertical Moment Mx N·m Max	Min	Vertical Force Fz N Max	Min	Horizontal Moment Mz N·m Max	Min	Horizontal Force Fxy Max	Min
At rest	352	-352	-602	-602	0	0	0	0

Load case details	Vertical Moment Mx N·m Max	Min	Vertical Force Fz N Max	Min	Horizontal Moment Mz N·m Max	Min	Horizontal Force Fxy Max	Min
At maximum speed No acceleration	210	210	57	-1262	0	0	899	-899
At maximum servoing acceleration Maximum speed	529	-529	188	-1003	500	-500	773	-773
At maximum braking torque Emergency stop maximum speed	634	-634	334	-1421	600	-600	896	-896



F_{xy}	Force in any direction along the XY plane
F_z	Force along the Z plane
M_{xy}	Moment that acts on the robot base along the XY plane
M_z	Moment that acts on the robot base along the Z plane

Figure 94: Wall-mounted safety forces

Table 67: Safety factor applied on load cases when the base is mounted on the wall

Load case details	Vertical Moment Mx N·m Max	Min	Vertical Force Fz N Max	Min	Horizontal Moment Mz N·m Max	Min	Horizontal Force Fxy Max	Min
At rest	352	-352	0	0	0	0	-602	-602
At maximum speed No acceleration	210	210	659	-659	0	0	297	-1502
At maximum servoing acceleration Maximum speed	529	-529	802	-802	500	-500	80	-1285
At maximum braking torque Emergency stop maximum speed	634	-634	937	-937	600	-600	-105	-1421

Installing the arm directly to a surface

One way to install the *arm* is directly to a surface, without a mounting plate.

Before you begin

Personnel

- **Number of Workers:** 2 when the orientation installation is not on a flat, horizontal surface

Supplies

- 4 x M8 socket head cap screws, grade 12.9 steel

  Note: Torque: 20 N·m

  Note: An equivalent imperial 5/16 in screw with equal or better mechanical properties and underhead contact patch can be used.

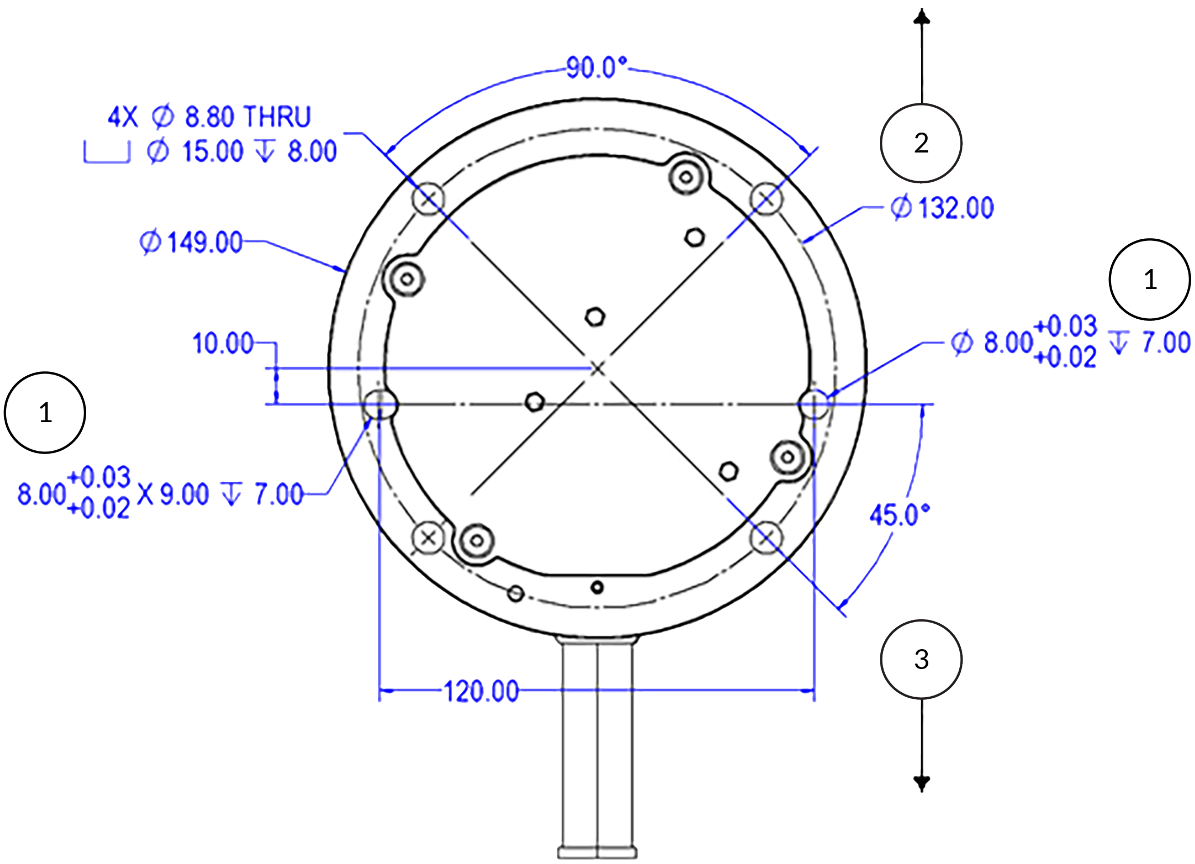
- 2 x 8 mm dowels

About this task

The arm is in a flat packaging pose to minimize the amount of space it takes in its packaging. This packaging pose also eases installation.

Procedure

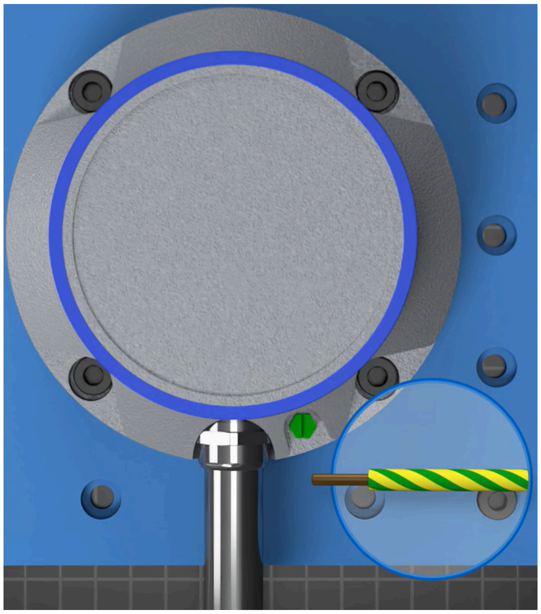
1. Define the location and prepare the surface according to the mounting pattern drawing.



1 - Dowel hole feature for optional dowel pins; 2 - Front; 3 - Back

2. Place the arm on the surface with the holes of the *base* lining up with the holes of the surface.
3. Secure the base with 5 mm hex key according to the hardware requirement.

4. Secure the green ground cable to the *protective earth* at the back of the base using the green screw.



5. Remove the protective covering from the arm.

Related topics

Specifications and dimensions of the base of the arm on page 13

Safety on page 47

Safety directives and warnings on page 48

Installing the arm on a third-party adapter plate

One way to install the *arm* is to install it on a mounting plate.

About this task

Follow the installation procedure outlined in the third-party documentation.

Related topics

Specifications and dimensions of the base of the arm on page 13

Safety on page 47

Safety directives and warnings on page 48

Unpacking the controller

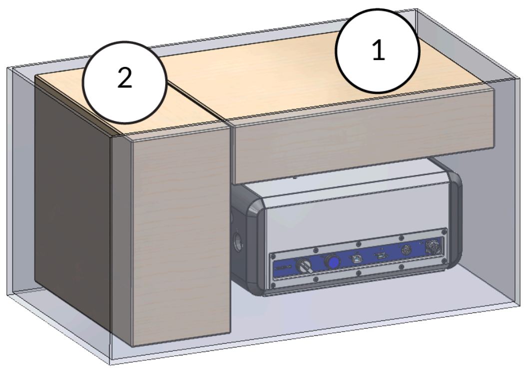
The box containing the *controller* is the smaller box, but it contains all the accessories bought with Link 6, as well as the power cord for the controller.

About this task

Unpack on a stable surface clear of other objects to make unpacking easier. It is recommended to unpack the contents of the controller box near the installation site for the controller.

Procedure

1. Make sure the arrows on the box are pointing upward.
2. Open the smaller box.
   Two cardboard boxes become accessible.



1 Accessories box

2 *Teach pendant* box

3. Remove the accessories and teach pendant boxes and place them on a stable surface.
4. Open the boxes and make sure everything you bought is there.
5. Remove the items from their foam wrapping one at a time and set them aside in a safe place.
6. Take out the Quick start guide and place it in an accessible location.
7. Remove the controller from the box and place it on a table.

Considerations before installing the controller

The *controller* must be installed on a stable surface. The surface must be able to support the weight of the controller without becoming deformed or unstable.

There are several ways in which the controller can be installed.

- Placed on a stable surface, such as a table.
- Secured to a stable surface, such as a table.
- Secured to a wall.
- Secured it in a stable cabinet, either on a shelf or on the wall of the cabinet.

Note: When the controller must be secured to a wall or to the inside wall of a cabinet, it is recommended to use the optional Kinova wall bracket.

Wherever the controller is installed, there is a set of guidelines that should be followed.

- Do not put anything on top of the controller.
- Make sure the controller has ample space around it.
- Make sure the controller is not in a position where it can be bumped.
- Make sure the controller is installed less than the attaching cable length away from the arm; the cable that connects the arm to the controller has a finite length and the route the cable must take must be taken into account.
- Make sure there is at least a 50 cm gap in the back where the fans are installed.
- Make sure 10 cm on the side from a wall to maintain proper air flow around the controller.
- Make sure the I/O panel and the controls on the front of the controller are easy to access.
- Make sure the LED on the power knob and on the ACK button are visible.

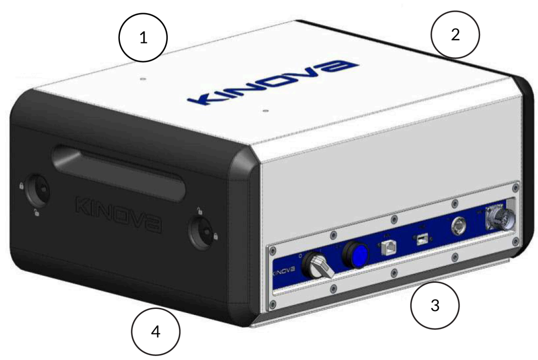


Figure 95: Linear perspective of the controller and its airflow requirements

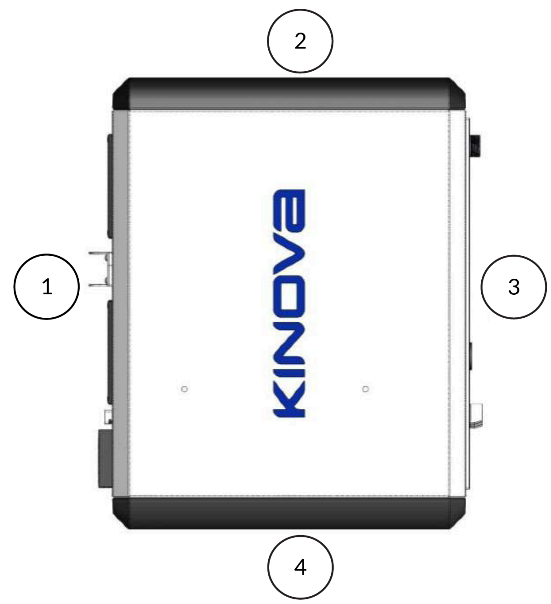


Figure 96: Top view of the controller and its airflow requirements

1	Minimum 50 cm from the wall for fan airflow
2	Minimum 10 cm from the wall
3	Easy access to connectors, switches, and LED status
4	Easy access to I/O panel

If there is no access to the underside of the installation surface and there is no need to fix it, leave the controller on its four rubber feet. However, if there is a need to fix it and it cannot be fixed to the installation surface, use the optional Kinova wall bracket.

The controller must be kept within specific temperatures ranges during storage and use.

Table 68: Safe temperatures for the controller

Action	Minimum temperature	Maximum temperature
Storage	-40 °C	65 °C
Powering up	0 °C	40 °C
Operating	-20 °C	40 °C

Related topics

Controller mounting interface on page 26

Installing the controller on a flat surface on page 136

Maximum ambient temperature reached on page 304

Maximum core temperature reached on page 304

Installing the controller on a flat surface

The *controller* must never be in a position where it can slip or fall.

Before you begin

Personnel

- **Number of Workers:** 1

Supplies

- 4 x M6-1.0-6h screws, closed end nut inserts

  Note: Maximum thread engagement depth: 6 mm

  Note: Torque: 7-9 N·m

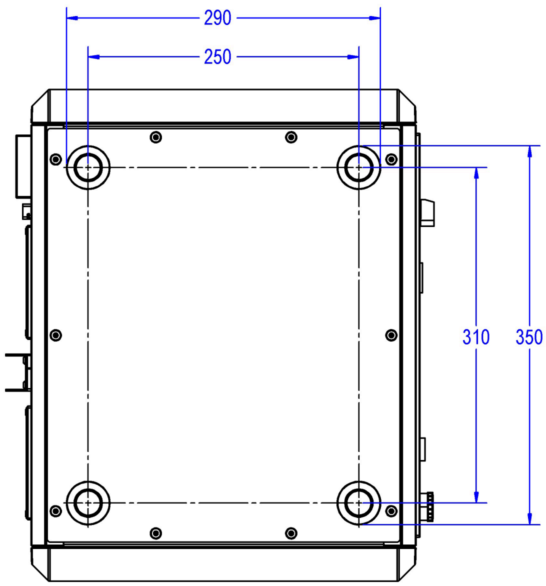


Figure 97: Mounting points of the controller, measured in millimeters

About this task

Verify the technical specifications of the controller to ensure there is maximum airflow around the controller, as well as enough space to hold the controller without putting it in a position in which it can be bumped.

The controller does come with anti-slip feet. Therefore, it is possible that placing it on a suitable flat surface suffices.

Alternatively, the controller can be secured to a flat surface.

Procedure

1. Align the foot mounts on the controller with the holes in the stable surface.
2. Place each of the M6 screws from under the surface into the foot mount location on the controller.
3. Tighten each screw until the controller cannot move.

Related topics

Considerations before installing the controller on page 134

Controller mounting interface on page 26

Surface mount of controller on page 27

Connecting the arm to the controller

The *arm* receives commands from the *controller* through a cable.

Procedure

1. Take the cable that comes out of the *base* of the arm.
2. Connect the end of the cable into the connector marked ARM on the controller.

3. Screw the cable in place.

Installing the teach pendant mount

The *teach pendant* comes with a flat display mounting interface (FDMI) that can be attached to a wall or other vertical surface. When the teach pendant is not in use, store it on the mount.

Before you begin

Personnel

- **Number of Workers:** 1

Supplies

For the MIS-D installation

- 4 x MIS-D compatible screws (M4 with 0.7 mm thread)
- Washers

For the 90 x 45 mm

- 4 x M8 screws

About this task

The teach pendant mount is compatible with VESA MIS-D, 75, and is compatible with the VESA Flat Display Mounting Interface (FDMI™) Standard. It has a 75 x 75 mm screw mounting pattern.

It also has a 90 x 45 mm pattern, which is not FDMI compatible. However, that pattern can be used to anchor the mount with M8 screws. The 90 x 45 mm alternative approach is excellent for such installations as 45 mm extrusion systems.

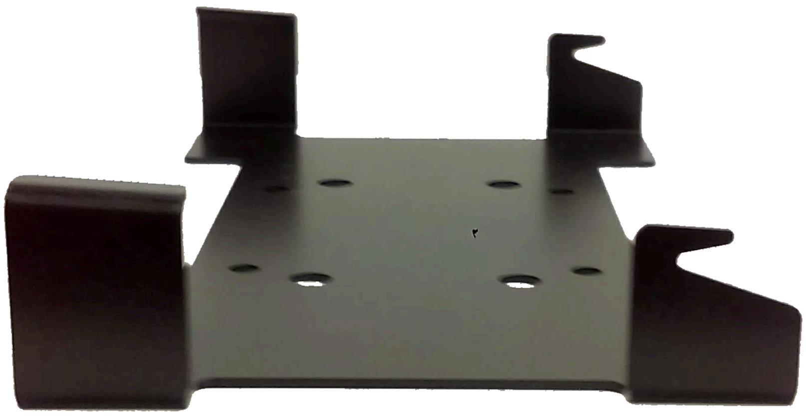


Figure 98: Teach pendant mount

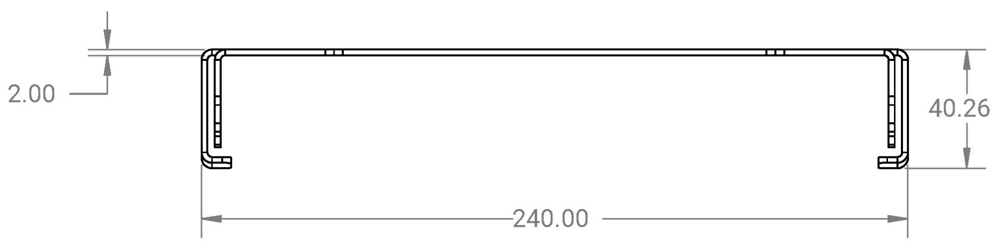


Figure 99: Specifications of the top view of the teach pendant mount

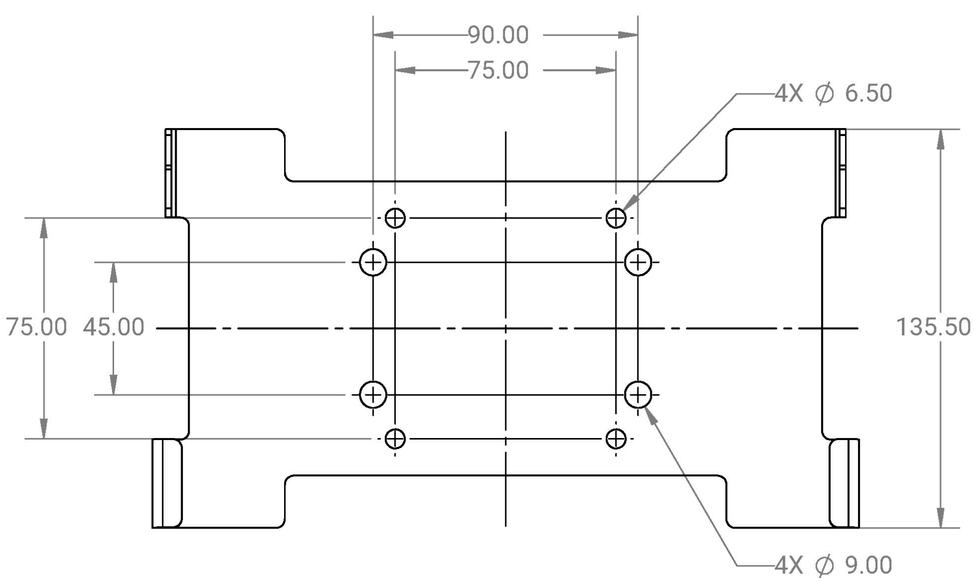


Figure 100: Specification of the front view of the teach pendant mount

Procedure

1. Choose whether you are installing using the FDMI standard or the 90 x 45 mm pattern; this determines which equipment and tools you need for installation.

2. Use the specifications in the figures to secure the teach pendant mount to the surface.

Related topics

Teach pendant overview on page 30

Connecting the teach pendant to the controller

The *controller* sends and receives commands from *Kortex Web App* on the *teach pendant*.

About this task

A cable is attached permanently to the teach pendant.

Procedure

1. Take the cable that comes out of the teach pendant.
2. Connect the end of the cable into the connector marked TP on the controller.

3. Screw the cable in place.

Connecting power to the controller

The *controller* requires power to power the rest of the robot.

Before you begin

The *AC* power line must be protected with a 15 A circuit breaker, or equivalent overcurrent protective device, at 120 VAC, and with a 7.5 A circuit breaker, or equivalent overcurrent protective device at 240 VAC.

The controller must be switched off.

About this task

The *arm* must be connected to the controller before the initial start-up of the controller.

Procedure

1. Plug one end the *AC* cord into the back of the controller.

2. Plug the other end of the *AC* cord into an electric socket.

Connecting a computer to the controller

Kortex Web App may be run on a computer, but it works only if the computer is connected to the *controller*.

Before you begin

You need to have an Ethernet CAT5 cable.

About this task

You can use a computer to program Link 6, particularly when you want to program the robot with minimal downtime.

Procedure

1. Connect one end of the cable to the Gigabit Ethernet port on the front or back of the controller.

2. Connect the other end of the cable to the computer.

Connecting a network to the controller

There are times when you need a permanent set-up in which the *controller* connects directly to a network.

Before you begin

You need to have an Ethernet CAT5 cable.

About this task

You can use a computer to program Link 6, particularly when you want to program the robot with minimal downtime.

Procedure

1. Connect one end of the cable to the Ethernet port on the front or the back of the controller.
2. Connect the other end of the cable to the network computer.

Initial software setup

Before you can start using Link 6, you have to configure the robot using *Kortex Web App*.

Kortex Web App is a Web-based *GUI* used for configuring, controlling, and monitoring the robot.

Supported client devices

Kortex Web App runs on the *teach pendant* that is connected to the *controller* with a cable. It also runs on a web browser on the computer that is connected to the controller over a wired Ethernet.

Kortex Web App is a responsive web application. It is designed to adapt itself to various aspect ratios and resolutions so that it can run on multiple platforms.

There are two types of users who use the teach pendant.

- The integrator can use the teach pendant when they configure or program the robot.
- The operator can use the teach pendant to run a validated program, provided they use it within an area defined by the integrator. The area should be outside of the *robot cell*.

Figure 101: Kortex Web App on a teach pendant

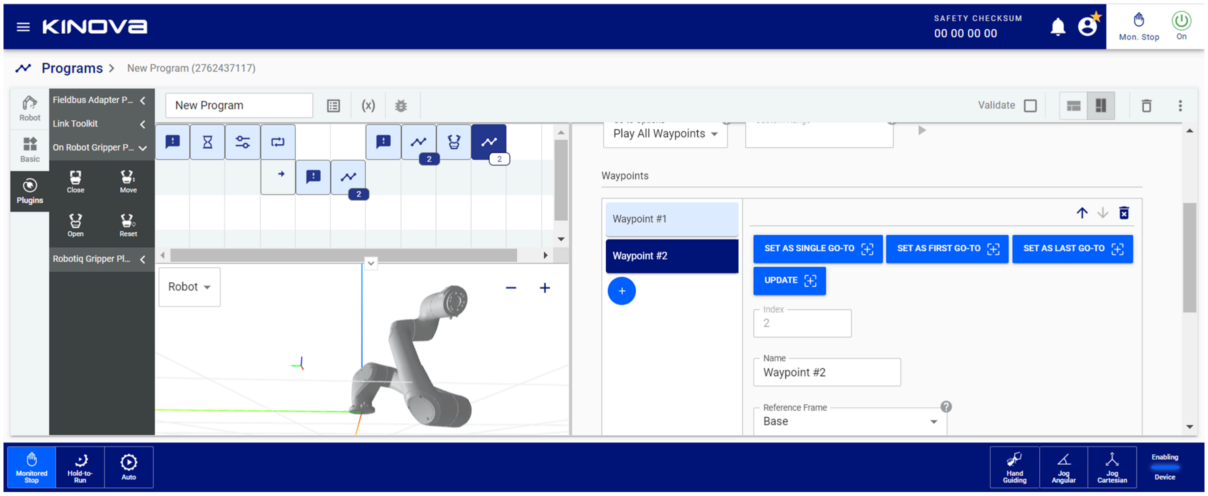


Figure 102: Kortex Web App in a web browser

Logging in

The first step in using Link 6 is to log into *Kortex Web App*. You can use either a computer or a *teach pendant*.

Before you begin

When you use Kortex Web App in a web browser, make sure the *controller* and the computer are connected with an Ethernet cable so that there is an established network connection.

When you use Kortex Web App in a teach pendant, make sure it is connected to the controller. There is no need to establish a network connection in this case.

Link 6 may still be in its package position. It also is possible that it is in its home position.

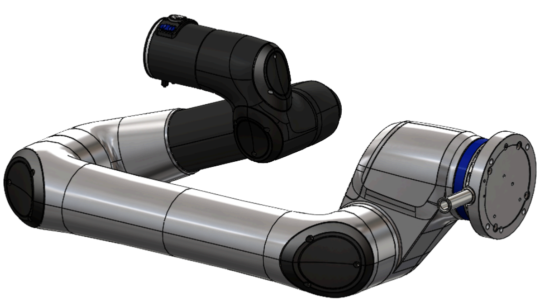


Figure 103: Link 6 in its package position

About this task

Kortex Web App is a Web-based *Graphical User Interface (GUI)* used for configuring, controlling, and monitoring the robot.

Procedure

1. Switch on the controller.

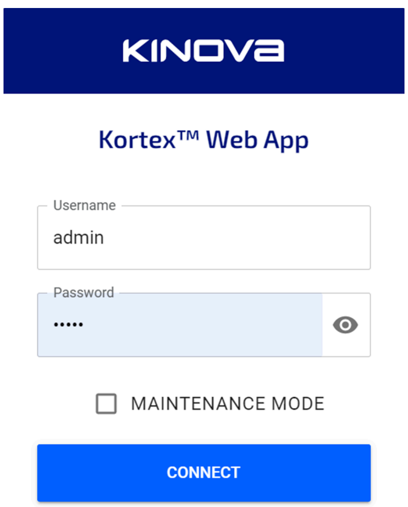
   

   The power knob flashes a white light while the controller initializes the software and *firmware*. When the controller is ready to be used, the power knob illuminates a steady white light.

   The teach pendant switches on automatically if it is connected to the controller.

   Kortex Web App launches automatically. It uses the default IP address `192.168.1.10` if it is connected to an external device using an Ethernet cable.

   Kortex Web App launches its login dialog.

2. Enter the user name and password of the admin user.

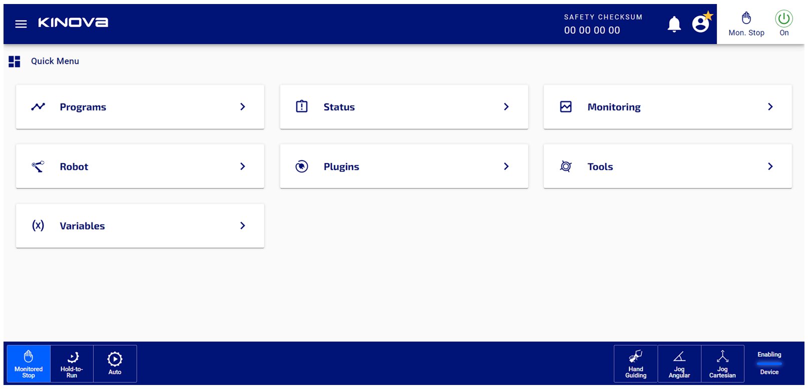
   Username is `admin`.
   Password is `admin`.

   Important: Change the admin password when this is the first log in.

   Important: Usernames and passwords are case sensitive.

3. Tap **CONNECT**.

   The **Home** page of Kortex Web App launches. The **Home** page displays the Quick Menu.



Repositioning the arm to use it for the first time

After the *arm* is installed, it probably will not be in an optimal position for starting to use the robot.

About this task

The robot must be on and the *admin user* must be logged on. The robot is in its factory settings, which means there are no programs yet. Use *Kortex Web App* in close proximity to the robot arm to make it easier to reposition the arm for the first time.

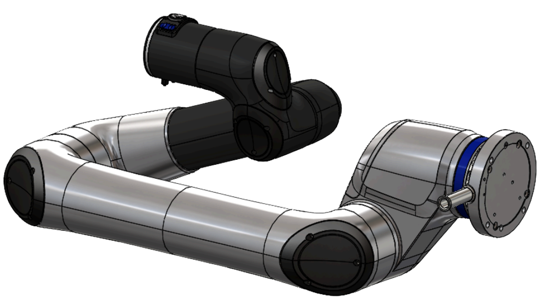


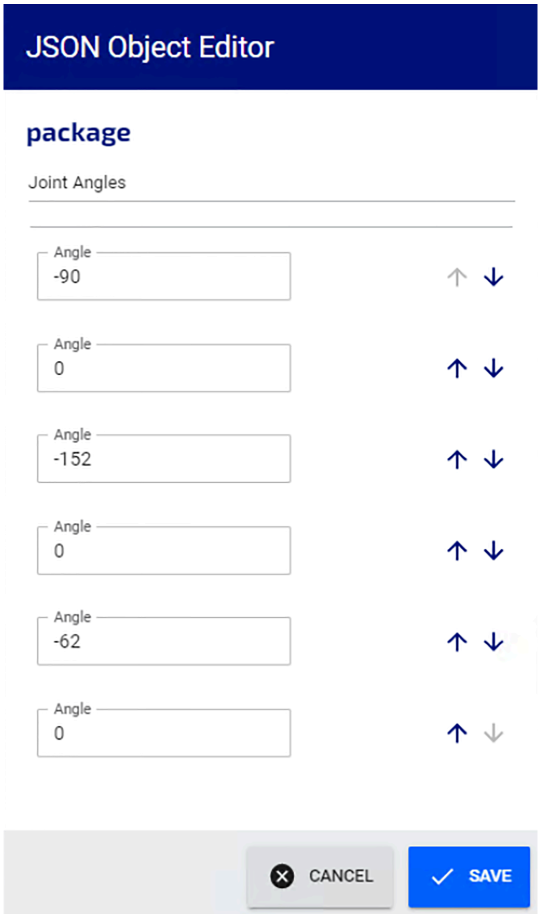
Figure 104: Link 6 in its package position

Table 69: Angles of each joint, from joint 1 to joint 6, of the packaging pose

Joint	Packaging pose angle
1	-90°
2	0°
3	-152°
4	0°
5	-62°

Joint	Packaging pose angle
6	0°

Figure 105: Angles of each joint in the Packaging pose



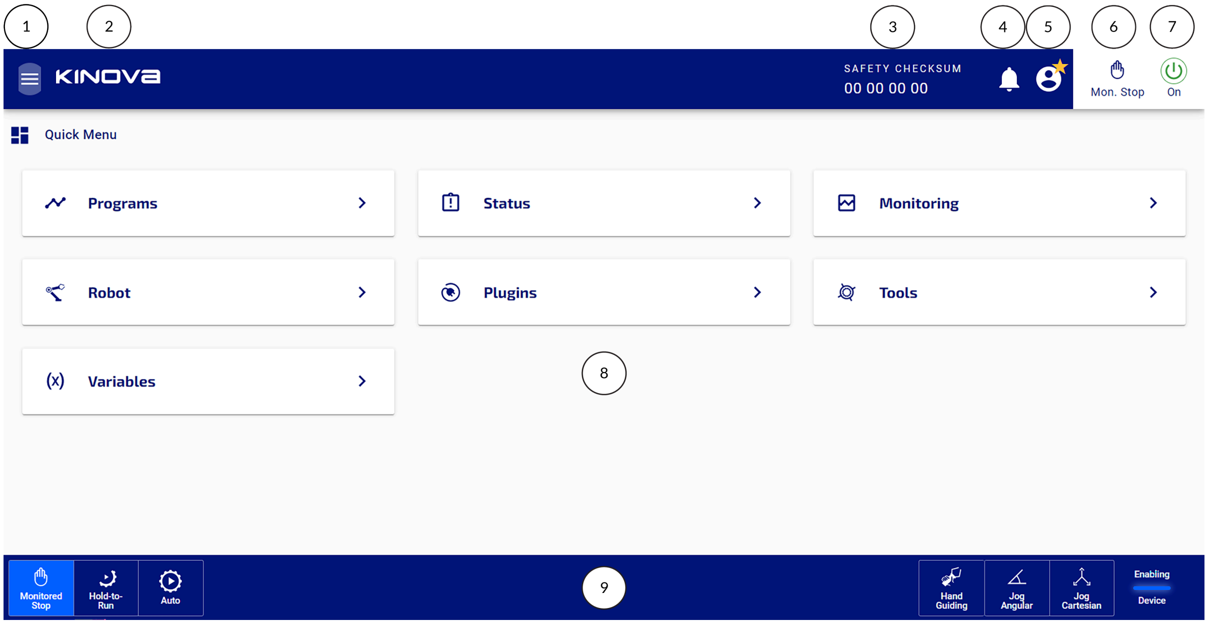
Procedure

1. Press and hold the *wrist enabling device* on the arm.
2. Use joint, or *angular*, Hand Guiding mode to move the robot out of its packaging pose and into a stable starting pose.

   Alternative: Press and hold the teach pendant enabling device , and use **Jog Angular**.
3. Release the wrist enabling device.
   The arm is now in the correct position to begin programming.

Kortex Web App layout

Kortex Web App is made of several sections.



Header

1 **Hamburger** to access pages

2 **Quick Menu** to return to the Home page

3 SAFETY CHECKSUM

4 Notification indicator

5 User profile

6 Mode indicator

7 On power button

Important: Tapping the power button switches the arm on or off.

Footer

8 Main information panel

9 Robot control panel

Menu pages

From either the Hamburger and a Quick Menu, you can access the different *Kortex Web App* functionalities.

When you first log in to Kortex Web App, you are brought automatically to the Quick Menu page. All of the most frequently accessed pages are there. Tapping on any of the items in the Main information panel brings you to the selected function.

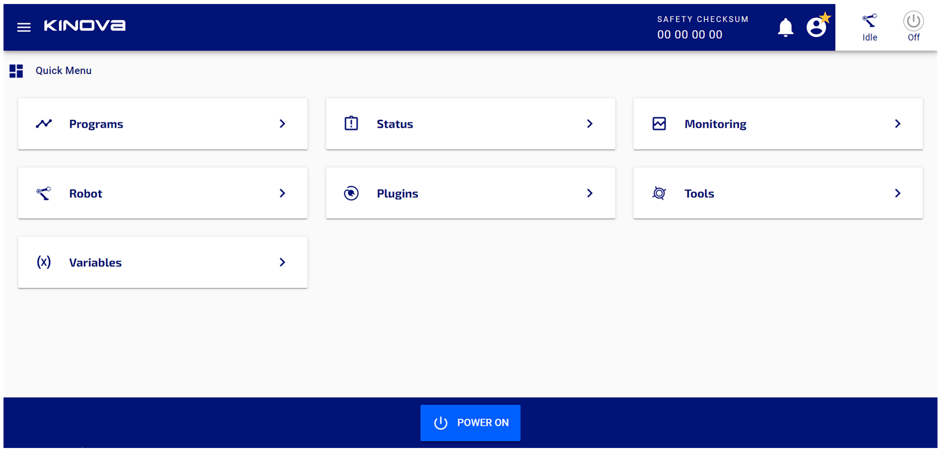


Figure 106: Home page is also the Quick menu

When you are not on the home page, tap on the Kinova logo to access the **Quick Menu**.

Alternatively, you can tap on the **Hamburger** icon (☰) to display the full menu. The different functions are organized into logical groups.

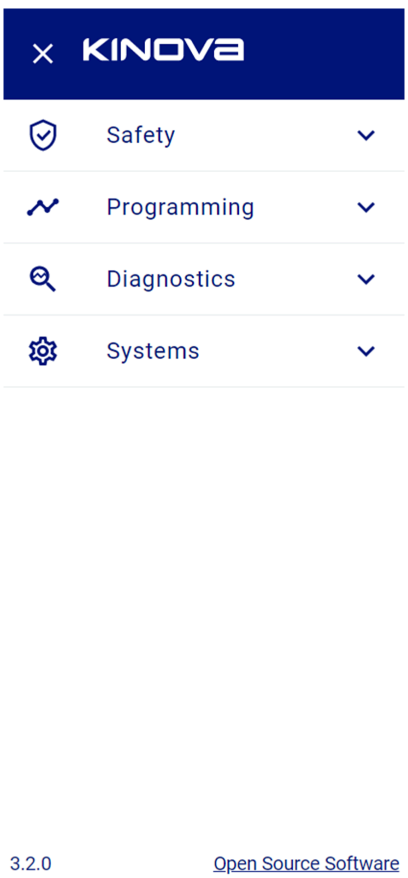


Figure 107: Full menu from the Hamburger icon

The main menu is made of four main options, each of which is subdivided further.

Access all safety-related actions you can take on the robot from the **Safety** menu.

- **Joint Limits**: Define the maximum position of each joint from -360° to +360°, as well as the maximum speeds at which the joints move in reduced safety mode and normal safety mode.
- **TCP and Elbow Limits**: Define the maximum translation and rotation speeds, as well as the maximum force and torque that can be used on the *TCP* and the elbow.
- **Protection zones**: Define the zones where the robot is not to enter.
- **Safety I/Os: Define the behavior of the equipment connected to the safety I/Os on the controller on the corresponding input and output channels.**

Access everything you need to create a program from the **Programming** menu.

- **Variables**: Define variables, along with their scopes, for modifying plugins and creating programs.
- **Programs**: Create, modify, and select programs, using actions defined by plugins and variables in the system.

Access monitoring and status information from the **Diagnostics** menu.

- **Status**: View the status of the main parts of the robot
  - **Safety Functions**
  - **Controller**
  - **Actuators**
  - **SCU**
  - **Wrist**
- **Monitoring**: View the live feedback from every sensor reading and the values derived form them.

  Example: The position of the *TCP*.

From the **Systems** menu, you can access information about the system in general, as well as perform general purpose tasks, such as upgrading the system and adding new users.

- **Information**: View information specific to Link 6 as a product, its controller, and its arm.
- **Robot**: Configure controls for the robot.
- **Tools**: View, modify, and add tools to reflect what is installed on the robot.
- **Plugins**: View, modify. and add *plugins* to use with the robot.
- **Networks**: View and modify the existing network communication system.
- **Remote Access**: Select which external sources can communicate with the robot.
- **Users**: View, modify, and add users who can use the robot. All admin users and operators are listed on this page.

- **Upgrade**: View the current version for Kortex Web App, the *firmware* bundle, and firmware versions. Upgrade the robot to the latest software and firmware updates. Reset the robot to its factory settings.
- **Maintenance**: Export logs of information related to the functionality of the robot, as well as save and load system configurations.

At the bottom of the menu, there are two pieces of information.

- The version of Kortex Web App
- A link to the **Open Source Software** that are used to build Kortex Web App

Open Source software packages

Link 6 is built using some Open Source software components. The licenses are listed on this **Open Source Software** page.

Access the **Open Source Software** page by tapping **Hamburger** > **Open Source Software**.

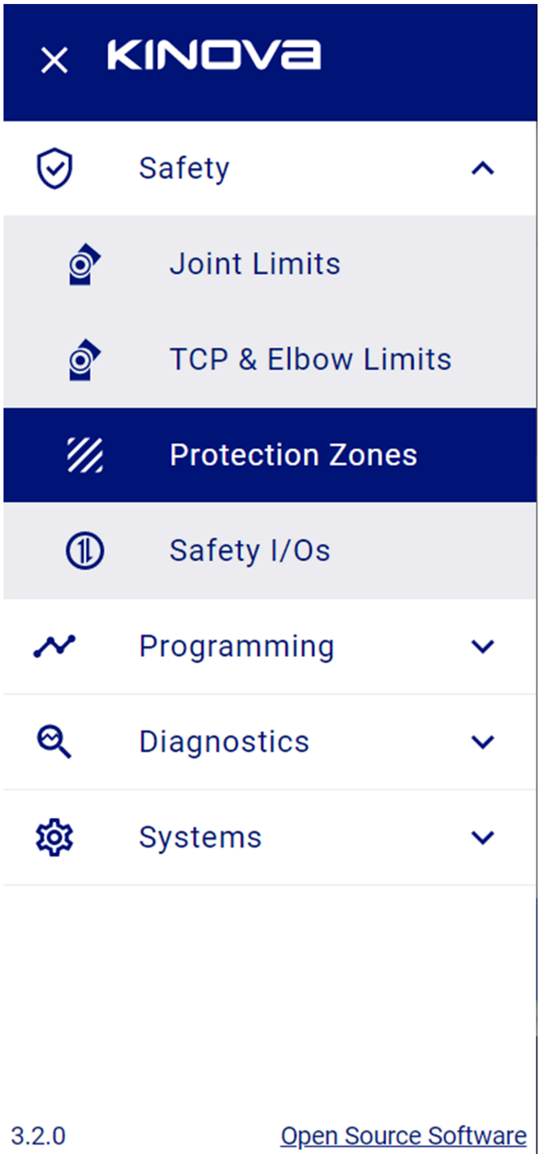


Figure 108: Accessing the Open Source Software page

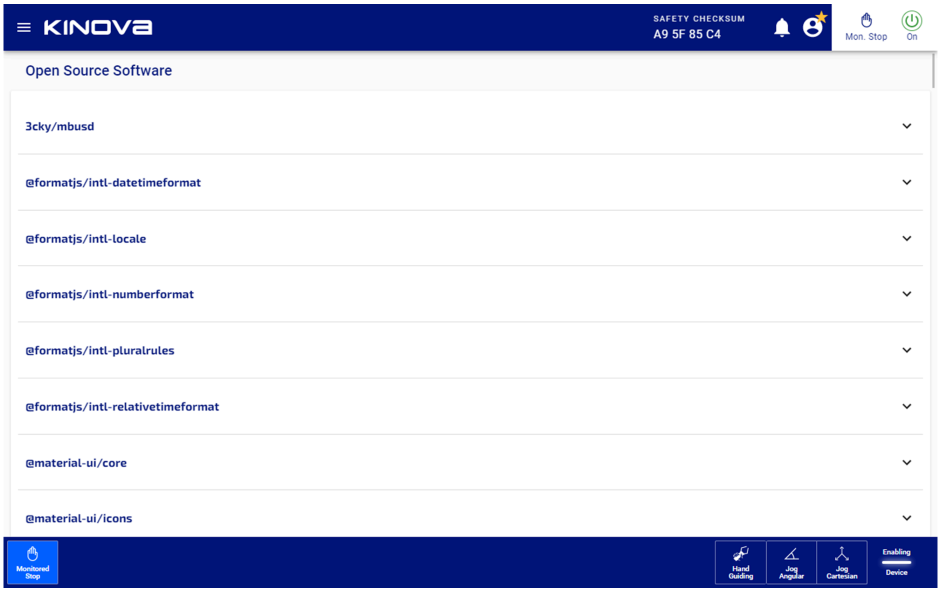


Figure 109: Open Source Software page

Tap on any of the packages listed to view its corresponding license.

Indicators, user profile, and power

Sometimes, you need a quick way to access notifications, your profile, the current mode of the robot, and a way to power on or off the *arm* of Link 6.

In the same header as the Hamburger and the Kinova logo are four more icons.

Figure 110: Indicators, modes, and arm power status

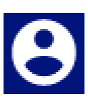
The Safety checksum icon is a unique number that represents the current configuration of several safety elements derived from the *MPU* and the *SCU*.

- SF03 Safety position limits
- SF04 Safety joint speed monitoring reduced and normal speed limits for all joints
- SF-06 Protection zone positions, shape type, dimensions, orientations, and activation status
- Tool sphere position and dimensions
- Safety TCP orientation speed monitoring normal and reduced speed mode limits
- Safety TCP translation speed monitoring reduced speed mode limits
- Safety elbow translation speed monitoring normal and reduced speed mode limits

The administrator is responsible for writing down the checksum when they are satisfied with the safety configuration. Any change in the value displayed in the Safety checksum icon indicates a change in the safety configuration.

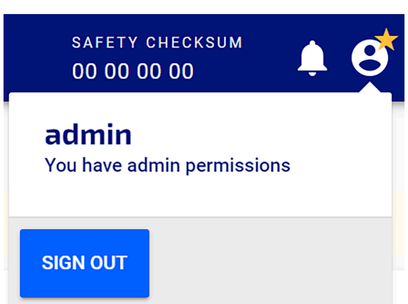
Tap the Notification icon to launch the **Events** page.

The admin user is logged into Kortex Web App.



An *operator* is logged into *Kortex Web App*.

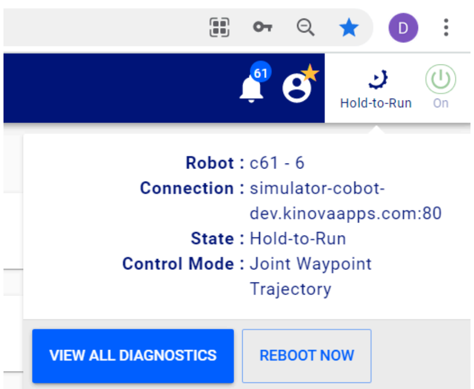
Tap the User profile icon to launch the **User profile** .



The **User profile** displays the role and permissions of the currently logged-in user. It is from this panel that you can sign out.

The Robot status icon displays the current mode that Link 6 currently is in. The arm is not operational during the Monitored Stop mode.

- Automatic
- Monitored Stop
- Hold-to-Run
- Maintenance mode



Tap the Robot status icon to launch a summary of the diagnostics of the robot.

Tap **VIEW ALL DIAGNOSTICS** to view information on the **Status** page.

Tap **REBOOT NOW** to *reboot* the robot.

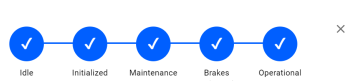
The Arm Power Off icon indicates the arm of Link 6 is off. The icon is gray.

Tap to switch the arm of the robot on.

When the arm is switched off, the state of the arm shows that it is idle; the arm is waiting to be used.

The Arm Power On icon indicates the arm of Link 6 is on. The icon is green. The states above the Robot control panel indicate the arm is operational.

Tap to switch the arm off.



When the arm is switched on, the state of the arm shows that it is on and is operational.

The Robot status icon changes to Monitored Stop.

Related topics

Event notifications

Robot lights and what they mean: a reference on page 122

User management on page 161

Robot control panel

The Robot control panel is both a status panel when operating the *arm* manually and another form of access to controls of the arm.

The current operating mode is displayed on the left of the panel . Tap it to access the other available operating modes.

Figure 111: Example of the Robot control panel

Figure 112: Example of the Robot control panel in Manual Jog mode

The current mode is a lighter blue than the rest of the Robot control panel.

At the opposite end of the Robot control panel are the manual control modes, as well as the *enabling device* status.

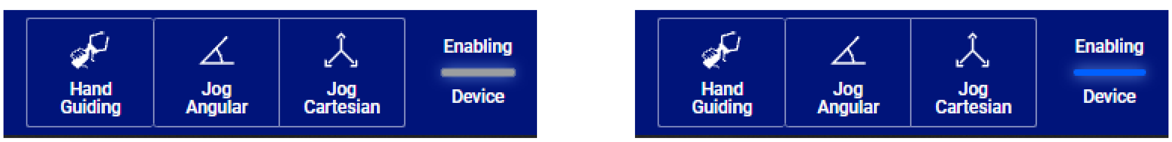
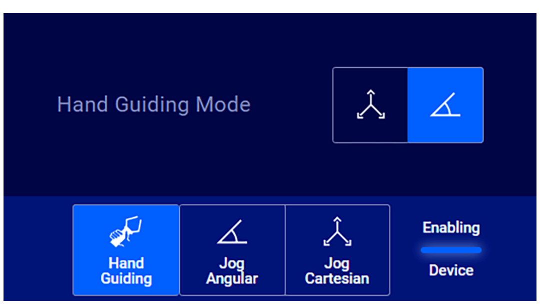
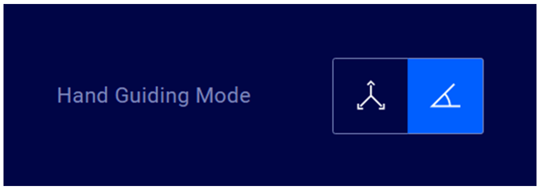


Figure 113: Virtual controls and mode toggles with enabling device off and then on

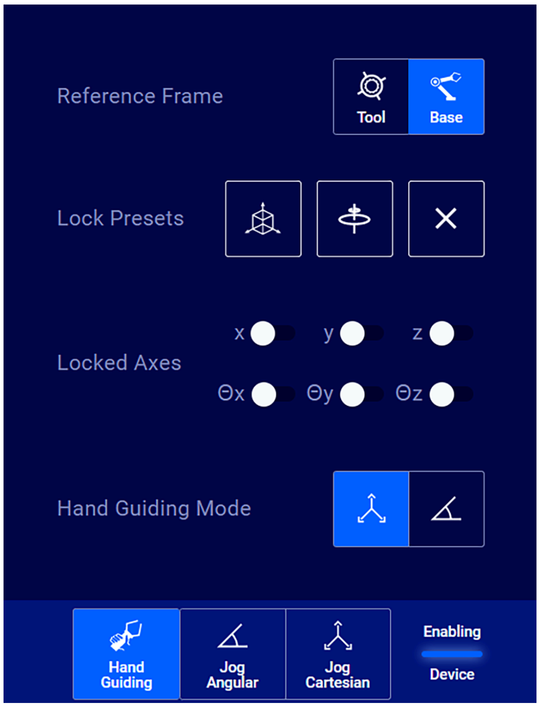
Tapping any one of these three buttons launches a window with controls. The controls are available only when the enabling device is held in the middle position.



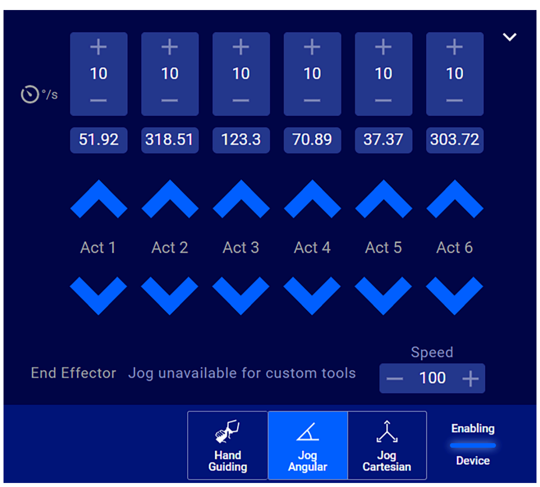
Modes available in Hand Guiding mode



Virtual controls in *angular* Hand Guiding mode



Virtual controls in *Cartesian* Hand Guiding mode

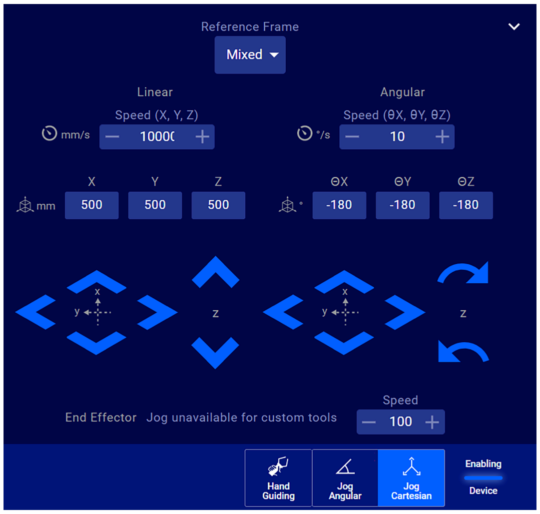


Virtual controls in *angular* Manual Jog mode

Tip: The controls are available only when the enabling device is held in the center position.

Tip: Tap the up arrow (^) in the corner to expand the view of the controls.

Tip: Tap the down arrow (v) in the corner to collapse the view of the controls.



Virtual controls in *Cartesian* Manual Jog mode

Tip: The controls are available only when the enabling device Is held in the center position.

Tip: Tap the up arrow (⌃) in the corner to expand the view of the controls.

Tip: Tap the down arrow (⌄) in the corner to collapse the view of the controls.

Note: When you tap Hand Guiding in the Robot control panel, the current mode is automatically Monitored Stop.

Note: When you tap Jog Angular or Jog Cartesian in the Robot control panel, the current mode is automatically Manual.

Related topics

Additional settings on page 290

System information

Whenever you need to know anything related to the product and the versions associated with it, whether to know whether it is eligible for an upgrade or Support requests specific information from you about the robot, check the **Information** page.

Tap **Systems** > **Information** to access the **Information** page.

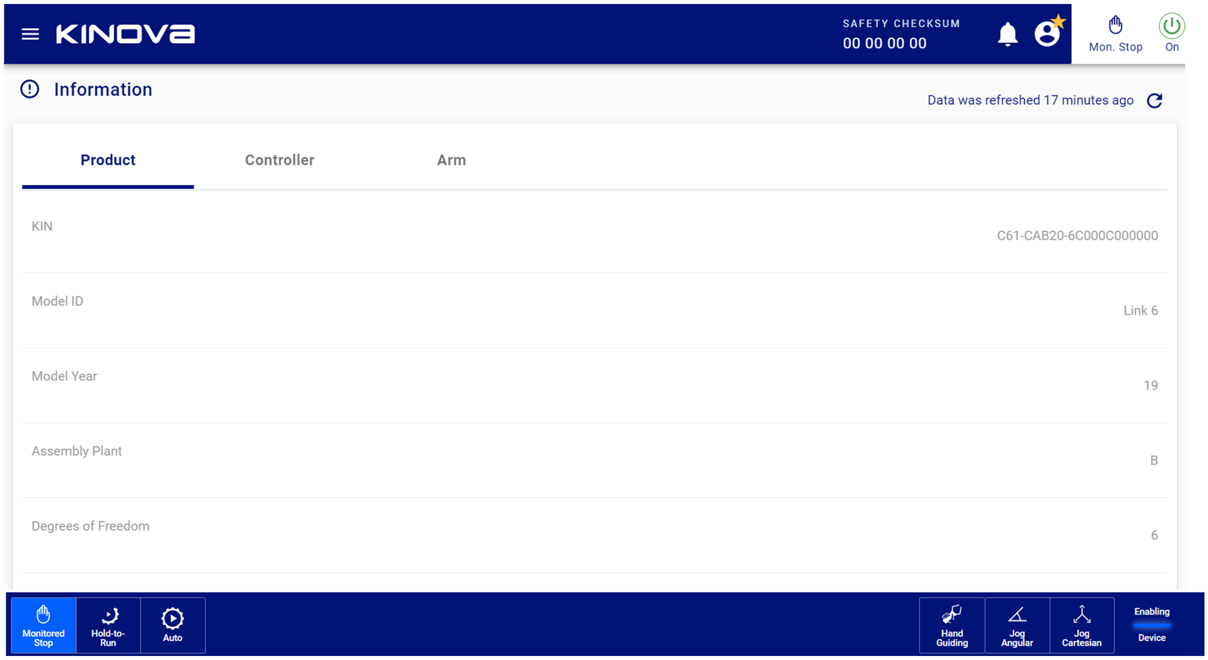


Figure 114: Information page when first launched

Tap on **Controller**, **Product**, or **Arm** to access the associated system information.

Table 70: Information available on each system pane

System part	Associated information
Controller	Serial Number
	Part Number
	Part Number Revision
	Main Processing Unit **(MPU)**
	MAC Address
	Safety Control Unit **(SCU)**
	MAC Address
	Bootloader Version
Product	KIN
	Model ID
	Model Year
	Assembly Plant
	Degrees of Freedom

System part	Associated information
Arm	***Actuator* 1** through **Actuator 6**
	Bootloader Version
	Device Type
	MAC Address
	Part Number
	Part Number Revision
	Serial Number
	Actuator Safety *MCU*1 through **Actuator Safety *MCU*6**
	Bootloader Version
	MAC Address

Robot networking and communication

An Ethernet cable connects the controller of Link 6 to the rest of the network. Link 6 can be connected to an IPv4 network.

All local networking and communication configurations are accessed by tapping **Systems** > **Networks**.

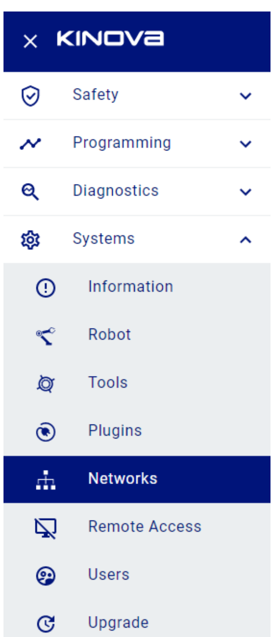
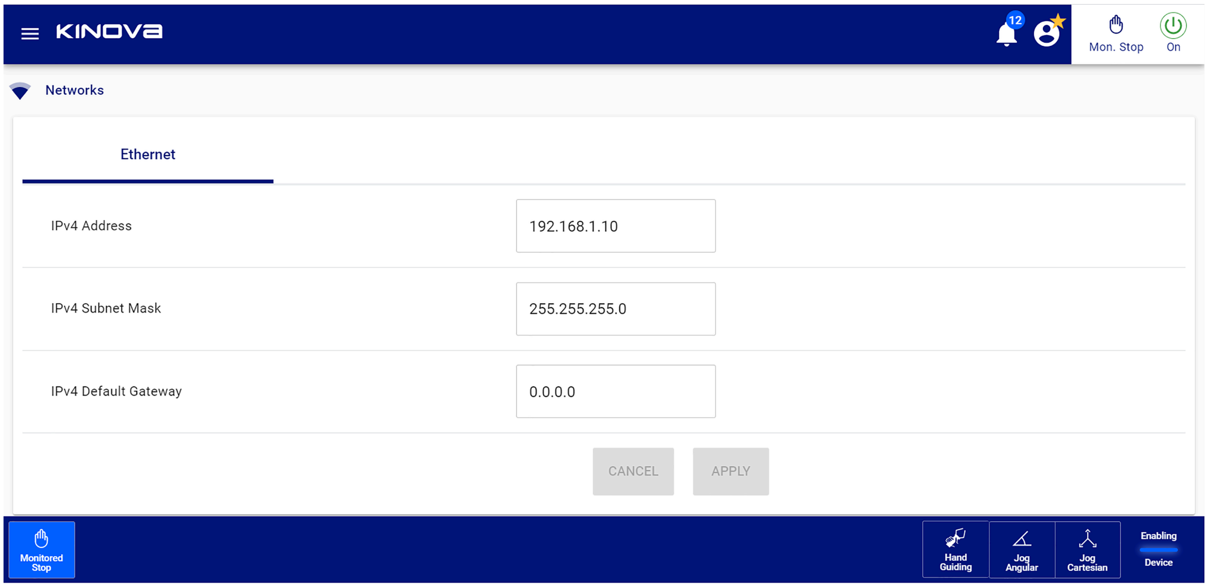


Figure 115: Accessing robot networking and communication configurations

The factory settings of the Controller Ethernet port IPv4 network can be modified in the **Network** page typing in the appropriate values.

- `IPv4 Address`
- `IPv4 Subnet Mask`
- `IPv4 Default Gateway`



When you change the values on the **Network** page, tap **APPLY** for the values to take effect.

Tip: The IPv4 address can be entered in Kortex Web App from a web browser on a computer, instead of the teach pendant, as long as the computer is connected to Link 6 using the Ethernet port.

If accessing the robot through the IPv4 address does not work, verify that the IPv4 address of the Ethernet port on the computer is on a compatible IP address; the address must be `192.168.1.XX`, where XX represents a set of two digits.

User management

There are two different types of users: *admin* users, and *operators*. Each of these user types are associated with *roles*, which determine what they can and cannot do in an application. It is common to refer to a user by the role they have.

The Administrator role carries more permissions than the Operator role. For instance, an admin is able to create users, but an operator cannot. However, both are able to run a validated program.

When there is a star beside the User profile icon, the user is an admin user. When there is no star, the user is an operator user.

Table 71: Access to pages depending on role

Admin user ()	Operator user ()
Login	**Login**
Quick Menu , Home page • **Programs** • **Status** • **Monitoring** • **Notifications** • **Robot** • **Plugins** • **Tools** • **(x) Variables**	**Quick Menu**, Home page • **Programs** • **Status** • **Monitoring**
Safety • **Joint Limits** • **TCP & Elbow Limits** • **Protection Zones** • **Safety I/O** Configure	**Safety** • **Joint Limits** • **TCP & Elbow Limits** • **Protection Zones** • **Safety I/O** View only
Programming • **(x) Variables** • **Programs** Create, debug, validate for use, run program	**Programming** • **Programs** Select and run validated program

Admin user ( )

Diagnostics

- **Status**: View only
- **Monitoring**:
  - View only.
  - Download snapshot of current data if a USB is inserted in the USB port of the controller .
- **Events**

  Tip: Access the **Events** page by tapping on the Notification bell icon.

  - Search for an event.
  - Filter for specific events by type and by date range.
  - Control what is displayed on the **Events** page.
  - Select an event and delete it.

view only

Systems

- **Information**: View only
- **Robot**: Configure
- **Networks**: Configure
- **Plugins**: Install and configure
- **Tools**: Add and configure
- **Upgrade**: Upgrade software and firmware
- **Users**: Manage users
- **Remote Access**: Enable and disable Modbus protocol and Kortex API communication
- **Maintenance**: Perform basic software maintenance of the robot, such as exporting current data, saving configurations of the robot, and restoring older configurations of the robot

Operator user ( )

Diagnostics

- **Status**: View only
- **Monitoring**:
  - View only.
  - Download snapshot of current data if a USB is inserted in the USB port of the controller .
- **Events**

  Tip: Access the **Events** page by tapping on the Notification bell icon.

  - Search for an event.
  - Filter for specific events by type and by date range.
  - Control what is displayed on the **Events** page.
  - Select an event and delete it.

view only

Systems

- **Information**: View only
- **Maintenance**: Perform basic software maintenance of the robot, such as exporting current data and saving current configurations of the robot

All user management tasks are accessed by tapping **Systems** > **Users**.

Related topics
Indicators, user profile, and power on page 153

Adding a user

New *operators* can be added to the system.

About this task

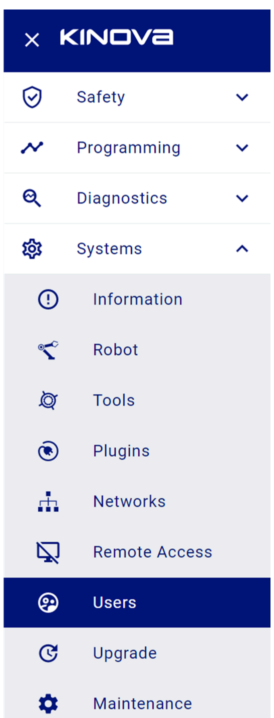
The *admin* user adds users to the system. Additional admin users cannot be added because there should be only one device, or person, controlling the robot at any given time. All users added to the system are operators.

Procedure

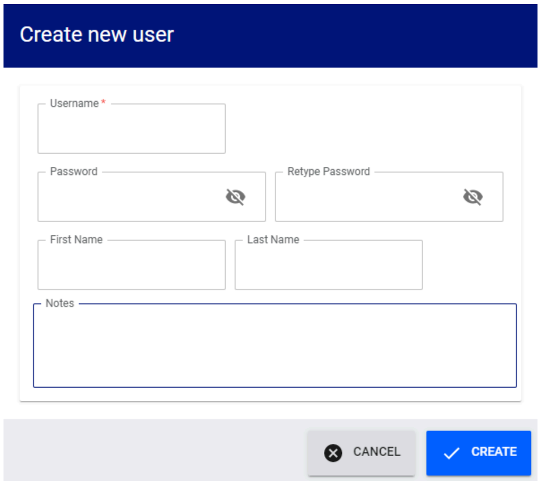
1. Tap **Systems** > **Users.**

   All users are displayed on the **Users** page.

   Important: The first time the **Users** page is accessed, one user is already listed. That is the Administrator user account used to log into Web App.



2. Tap + in the bottom right corner to add more users.
   The **Add User** dialog launches.



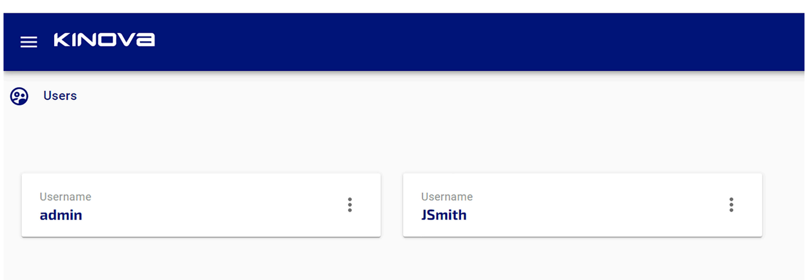
3. Enter information in the fields.
   - `Username`
   - `Password`
   - `Retype Password`
   - `First Name`
   - `Last Name`
   - `Notes`

   Important: The username and password are case sensitive.

4. Tap **CREATE**.

Results

A user is added to the list of current users.



Editing a user

When a user is created in the system, it is possible that the user's name was misspelled, a different username must be used, the password must be updated, or the user wants to work in a different language. In all these cases, the admin must edit the user information.

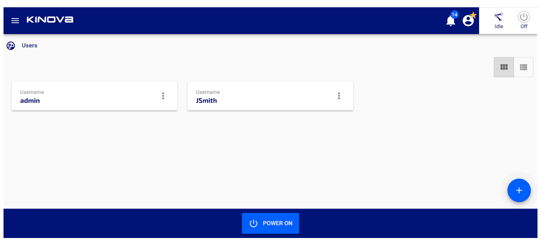
About this task

The *admin* edits users in the system.

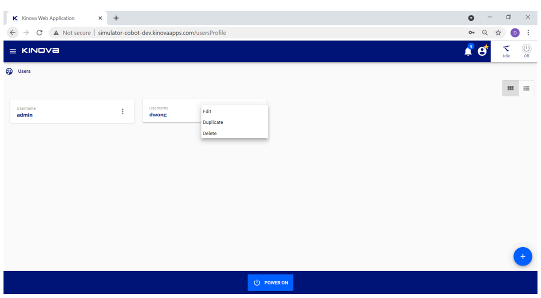
Procedure

1. Tap **Systems** > **Users.**

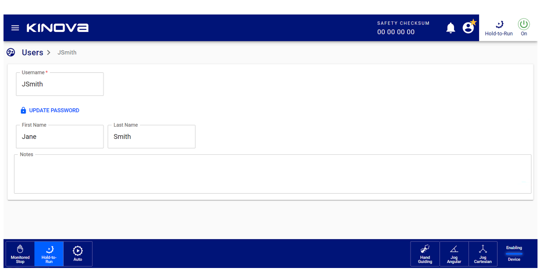
   All users are displayed on the **Users** page.

2. Tap on a user's name.
3. Tap the *kebab menu* (⋮).
4. Tap **Edit**.

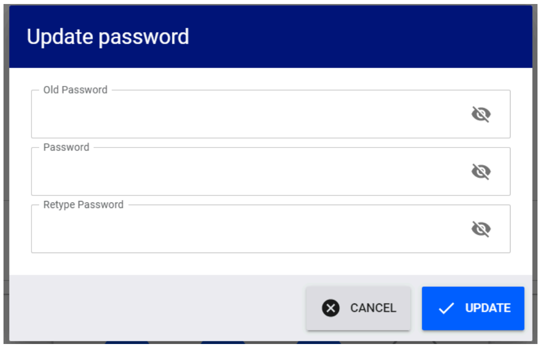
   

   The **Edit User Profile** page is displayed.

5. Update the information.
   - `Username`
   - `First Name`
   - `Last Name`
   - `Notes`
6. Tap **UPDATE PASSWORD** to change the user's password.

   The **Update password** dialog launches.

7. Enter the `Old Password`, new `Password`, and `Retype Password`.
8. Tap **UPDATE**.

Results

All edited information is saved automatically.

Deleting a user

Sometimes a user must be removed from the system.

About this task

The *admin* removes users from the system.

Procedure

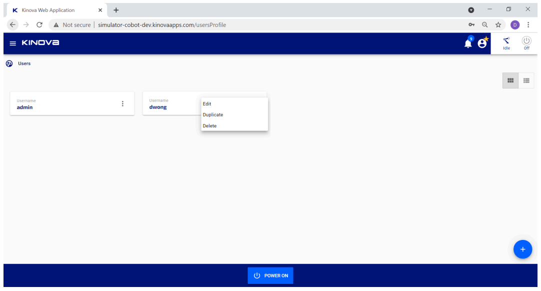
1. Tap **Systems** > **Users.**

   All users are displayed on the **Users** page.

2. Tap the *kebab menu* of the user to be deleted.

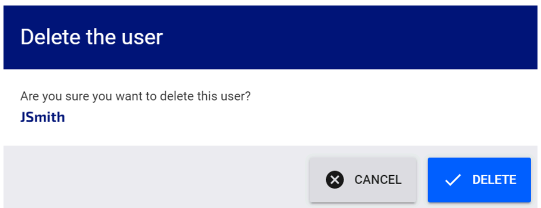
   Note: Administrator accounts cannot be modified or deleted. The content of the kebab menu is not selectable.

   A contextual menu launches, one of the choices being to delete the user.

3. Tap **Delete**.

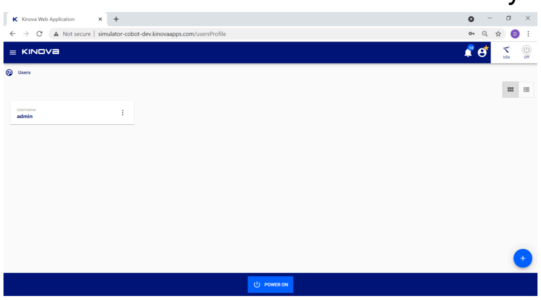
   A **Delete the user** confirmation dialog launches.

4. Tap **Delete** to delete the selected user.

Results

The selected user is deleted from the system.



Elevating permissions

The *role* of a user may need to change from *operator* to *admin user*.

Before you begin

Log in *Kortex Web App* as an operator.

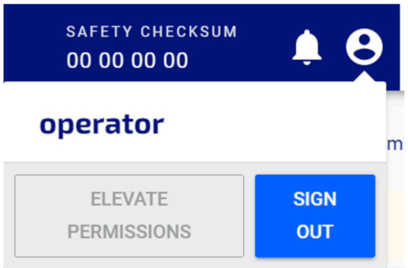
About this task

An operator can elevate their permissions to the administrator level only when the operator knows the admin password.

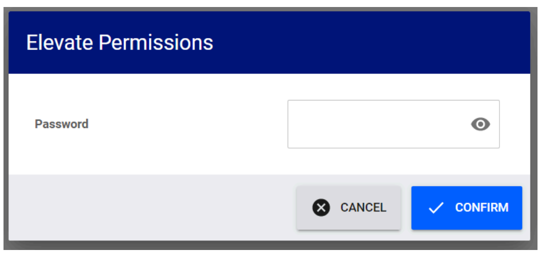
Tip: An operator can elevate their permissions only if they know the admin password.

Procedure

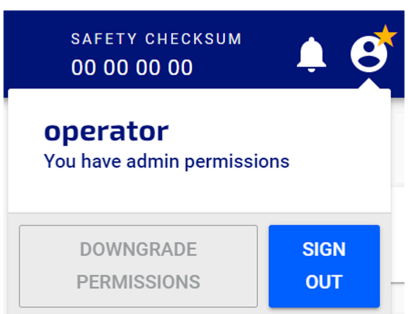
1. Tap **User Profile** .
   A **User profile** launches.



2. Tap **ELEVATE PERMISSIONS**.
   The **Elevate Permissions** dialog launches.



3. Enter the admin user `password`.
4. Tap **CONFIRM** to elevate permissions.
   The current user can now do the same tasks as an admin user.



Tip: A quick way to make sure the current user is an admin user or an operator is to look at the **User Profile** icon.

- An administrator user has steady star on the **User Profile** icon. An administrator user cannot downgrade their permissions.
- An operator with elevated permissions has a flashing star on the **User Profile** icon. An operator with elevated permissions can downgrade their permission level.

- An operator with operator level permissions has no star on the **User Profile**. An operator can elevate their permissions if they know the admin password.

Remote access

You do not have to use the teach pendant to communicate with Link 6. You can access it remotely.

All remote communication configurations are accessed by tapping **Systems** > **Remote Access**.

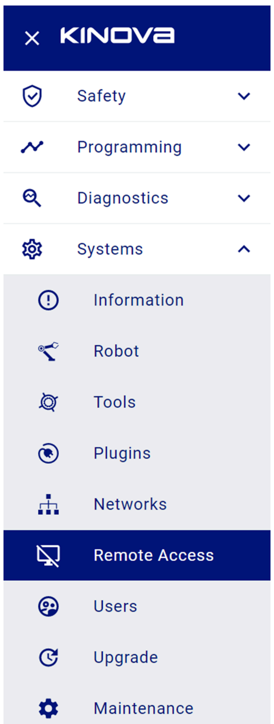


Figure 116: Accessing remote access configurations

A warning is displayed directly on the **Remote Access** page that states that active sessions may close when the communication is enabled or disabled.

The only communication systems that can be enabled or disabled directly with the arm are the Modbus Slave protocol and the Kortex *API*. The enabled state of the protocol and the API are maintained, or persist, so that they are always there after *booting* and *rebooting*.

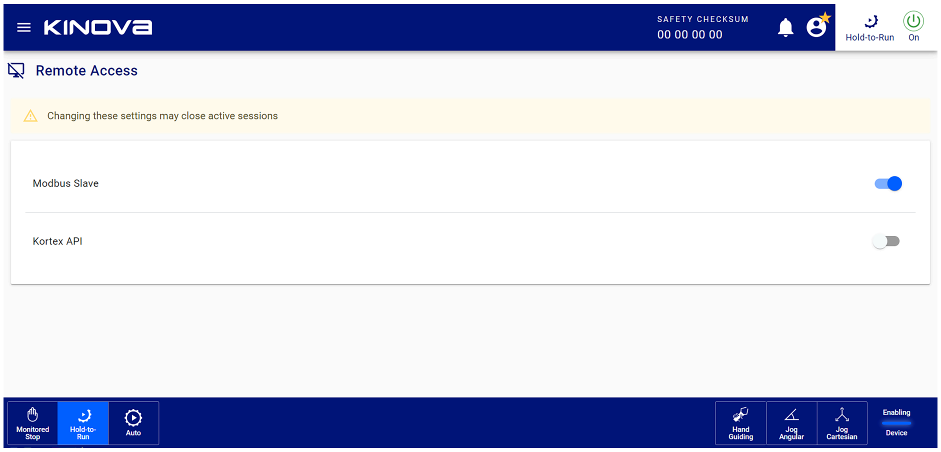


Figure 117: Remote Access page

Important: When **Kortex API** is enabled, you can use the software interface to program the interactions of your system with Link 6. The correct set of permissions is required to work with the Kortex API.

Upgrading software

There are continuous improvements made to Link 6. Make sure that your operations take advantage of these improvements by upgrading the software driving the robot.

About this task

Only the *admin* can update the software.

All software upgrades are accessed by tapping **Systems** > **Upgrade**.

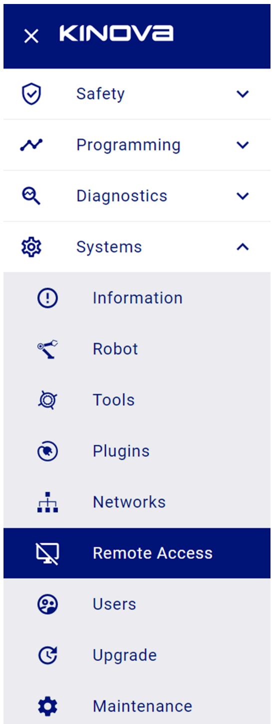
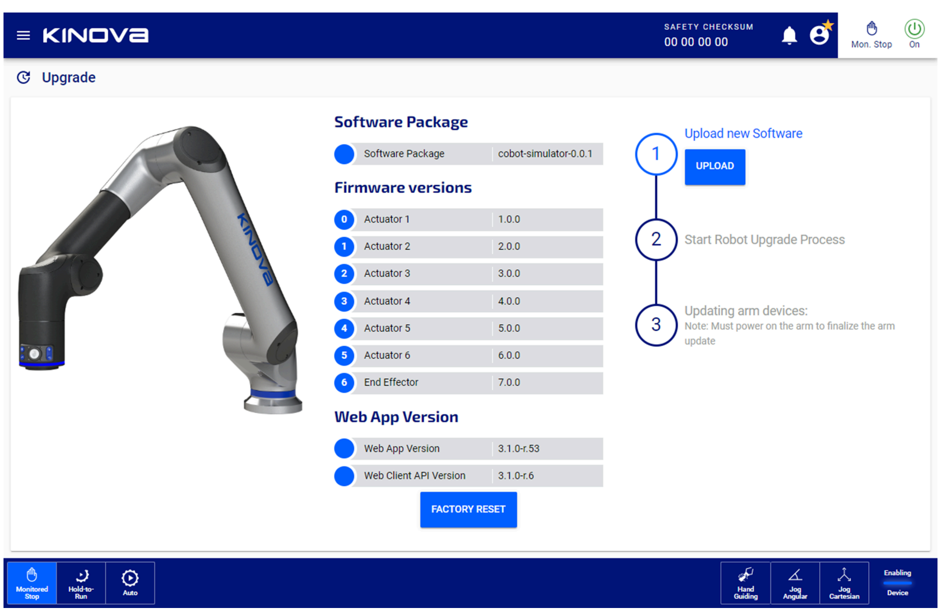


Figure 118: Accessing the software upgrade operation

Procedure

1. Tap **Off** in the On section of *Kortex Web App* to turn off the *arm*.
2. Tap **Systems** > **Upgrade.**

   The **Upgrade** page launches. It shows the current version of Kortex Web App, Web Client API, *firmware* bundle, and firmware versions.



3. Download the new software package from a computer onto a *USB*.

4. Place the USB into the drive on the controller.
5. Tap **UPLOAD**.
   A file explorer window launches.
6. Navigate to the new software package on the drive, select it, and tap **Open**.
   a. The *controller* receives its upgrades. Controller upgrades take little time.
   b. Each *actuator* receives their upgrades. This takes more time than the controller. There are six (6) actuators and each actuator receives two different firmware updates.
      - Safety *MCU* updates
      - Normal operation updates
   c. The user cannot access Kortex Web App until after the controller *reboots*.

Results

The robot reboots with the upgraded firmware and software.

Consequences of downgrading from v3.2 to v3.1

It is entirely possible that you may want to work with an older version of Link 6. Reverting to an older version comes with restrictions.

Version 3.2 is not backwards compatible with version 3.1.

Any configurations done using version 3.2 are lost with a downgrade to 3.1. If you do want to downgrade to 3.1, take note of the current configuration of your system, downgrade to 3.1.x, and perform a factory reset. Re-enter the configuration of your system by referring to your notes.

Plugins

In addition to the pre-installed *plugins*, the *administrator* can extend the functionality of the robot through additional plugins to the *Kortex Web App*.

Pre-installed plugins are related directly to Link 6 and cannot be uninstalled.

Arm plugin	The plugin handles all actions needed to control the *arm*. The configuration and settings of the arm are not visible because they are not modifiable.
Industrial I/O plugin	The plugin handles all general *I/O* between the *controller* and on the *wrist*. That is, the plugin exchanges information using pure electrical

signals through cables connected to some other piece of equipment.

All pre-installed plugins start to run when the controller is switched on. However, they can be stopped so the settings can be modified. All plugins that are running are, by default, active.

Related topics

Plugins on page 323

Getting the plugins for the robot

Upload and installation must be done from a computer and a *USB* storage device.

Before you begin

All plugins have dependencies on different aspects of the robot.

Make sure to verify all dependencies before installing the plugin.

Table 72: Plugin dependencies and expected issues when dependencies not met

Dependency	What to expect when dependencies are not met
Hardware resources • *RAM* • Storage • *CPU* core • Video camera stream, if applicable • *GPU*, if applicable	Installation failure

About this task

The plugin installation file is named `<plugin_name_><version>.kp`, where <version> is the version of the plugin. Contact Kinova to get the appropriate plugin file if you are unsure which files you need.

A computer or *USB* device is needed to upload and install the plugin installation file.

Note: Plugins are updated from time to time. Always make sure you have the latest plugin for your installation.

Important: If the plugin is not compatible with the currently running version of Kortex Web App, an error dialog launches.

Procedure

1. Download the plugin installation file `<plugin_name_><version>.kp`.

2. Save the plugin installation file in the root directory of the USB storage device.
3. Eject the USB after the transfer of the plugin installation file is complete.
4. Plug the USB storage device on the *controller*.
5. Switch on the *teach pendant*.
6. Enter the administrator credentials.
7. Tap **CONNECT**.
8. Tap **POWER ON**.
   The arm is switched on.

What to do next

Install the required plugins from the **Plugins** page in *Kortex Web App*.

Installing plugins

Some *end effectors* and robot systems require special plugins that need to be installed before they can be use as intended. Other plugins may operate in a standalone fashion to add new features to your robot.

Before you begin

The arm must be turned off before plugins are installed.

The plugin must not be installed.

About this task

Plugin files have a `.kp` extension.

All installed plugins are accessed by tapping **Systems** > **Plugins**.

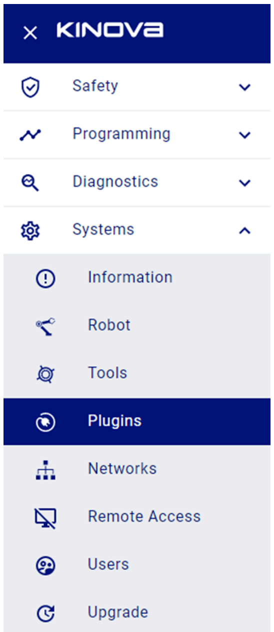
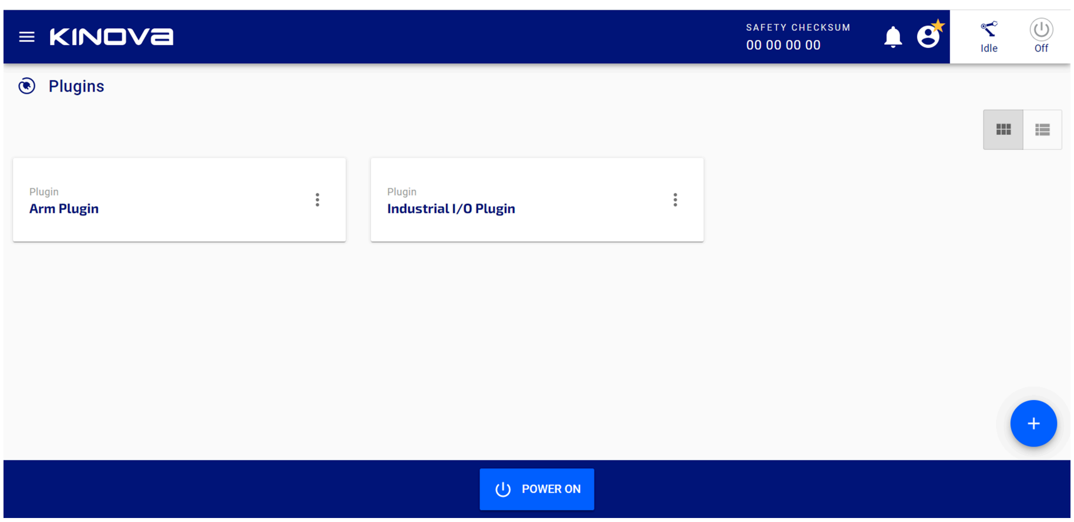


Figure 119: Accessing the plugins

Procedure

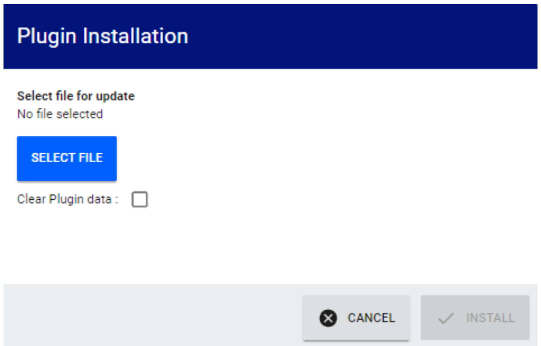
1. Tap **Systems** > **Plugins**.

   The **Plugins** page launches.



2. Tap **+**

   A **Plugin Installation** dialog launches.

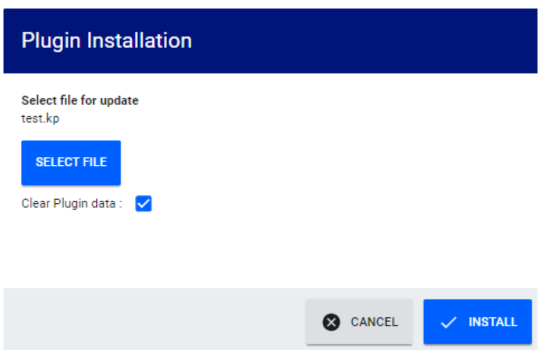


3. Tap **SELECT FILE**.

   A window launches.

4. Browse and select the file with `.kp` extension.
5. Tap **Open**.

   The **Plugin Installation** dialog shows the name of the selected file.



6. Tap **Clear Plugin data**.

   When you clear the plugin data, you are making sure you have a clean installation of the plugin.

7. Tap **INSTALL**.

Results

The installed plugin appears on the **Plugins** page and is running.

Any tools associated with the uninstalled plugin are automatically removed from the **Active** and **Inactive (drag & drop)** panes of the **Tools** page.

Related topics

Uninstalling plugins on page 178

Uninstalling plugins

Some *end effectors* and robot systems require special plugins that need to be installed before they can be use as intended. It is recommended to uninstall the existing plugin before installing the new plugin because there may be conflicts in their structures.

Before you begin

The arm must be turned off before plugins are uninstalled.

About this task

All installed plugins are accessed by tapping **Systems** > **Plugins**.

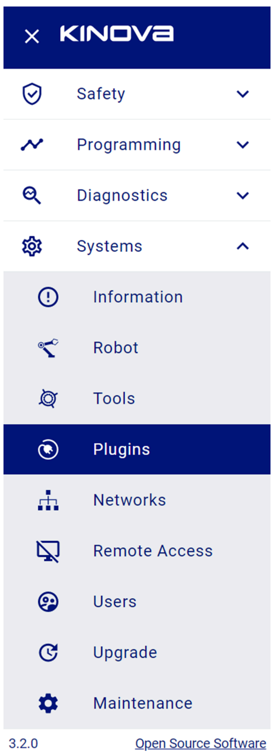
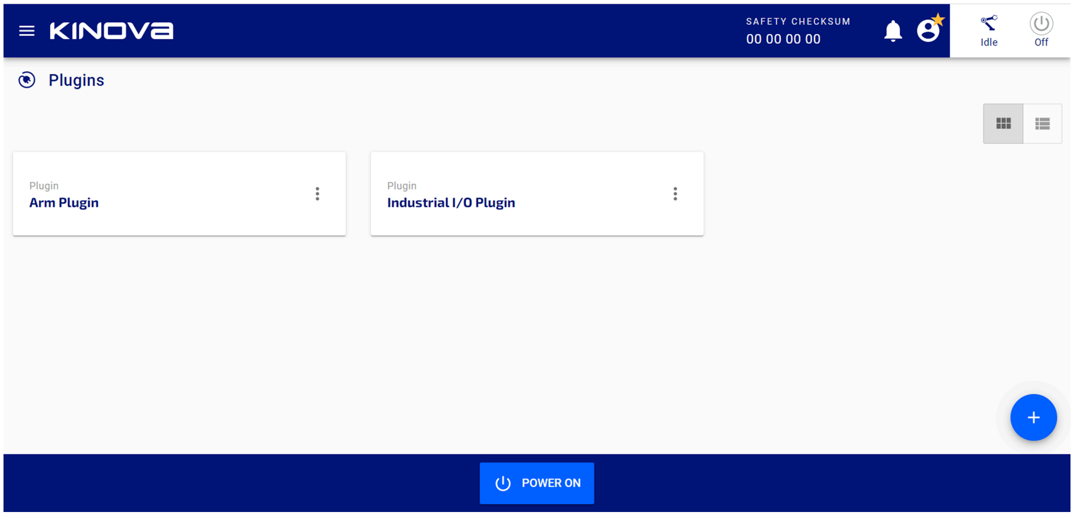


Figure 120: Accessing the plugins

Procedure

1. Tap **Systems** > **Plugins**.
   The **Plugins** page launches.



2. Tap the plugin that you plan on upgrading.
   The **Configurations** page of the plugin launches.
3. Switch off **Activate Plugin**.
   The plugin stops running.
4. Tap **Plugins** to return to the main **Plugins** page.
5. Tap ⋮ > **Delete**.

Results

The plugin is uninstalled and is not on the **Plugins** page.

Any tools associated with the uninstalled plugin are automatically removed from the **Active** and **Inactive (drag & drop)** panes of the **Tools** page.

Related topics
Installing plugins on page 175

Overview of working with plugins

Tapping a plugin from the **Plugins** page launches the page of that plugin.

Figure 121: Example of a plugin page of a plugin that is stopped

1	Tap **Plugins** to return to the list of plugins.
2	Name of the selected plugin with the name of the namespace associated with the selected plugin in parentheses.
3	Tap **Configurations** to view and modify the current configurations of the selected plugin. **Important:** Tap **Stop** (Stop ■) before you attempt to modify the current configurations. **Important:** Tap **Run** (Run ▸) after you modify the configurations. The plugin runs with the latest configuration values. When there are no modifiable configurations in the plugin, a message is displayed. This plugin does not expose any configuration.
4	Tap **About** to view important information about the selected plugin. • Name • Version • Installation Type • Licenses • Description • Email
5	Toggle **Activate Plugin** to deactive plugins that need to be installed but are not being used. When the toggle is white, it is off. Tap **Activate Plugin** to activate a deactivated plugin. When the toggle is blue, it is on.
6	The current operating state of the plugin. Tap **Stop** (Stop ■) to stop the plugin from running. The button changes to **Run** (Run ▸). Tap **Run** to restart the plugin after modifying its settings. The button changes to **Stop**.

Configuring tools

All tools need to be configured before they are used.

About this task

All tool configuration tasks are accessed by tapping **Systems** > **Tools**.

Tip: Before you can edit the active tools, the robot must be switched on and it must be in Monitored Stop mode.

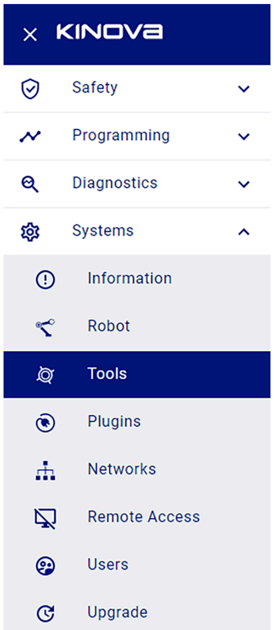
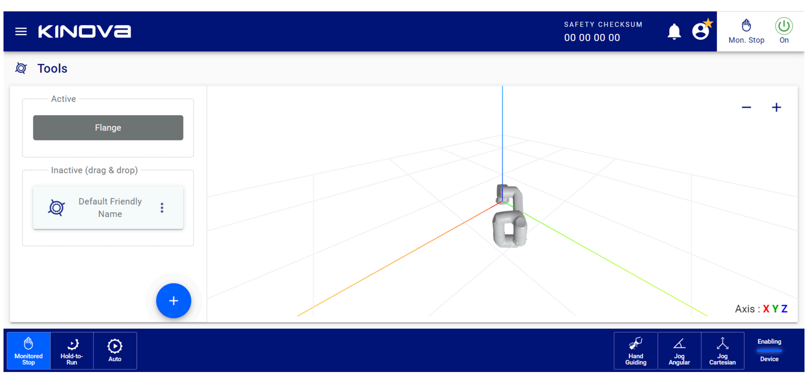


Figure 122: Accessing tool configuration operations

Procedure

1. Tap **Systems** > **Tools**.

   The **Tools** page launches.

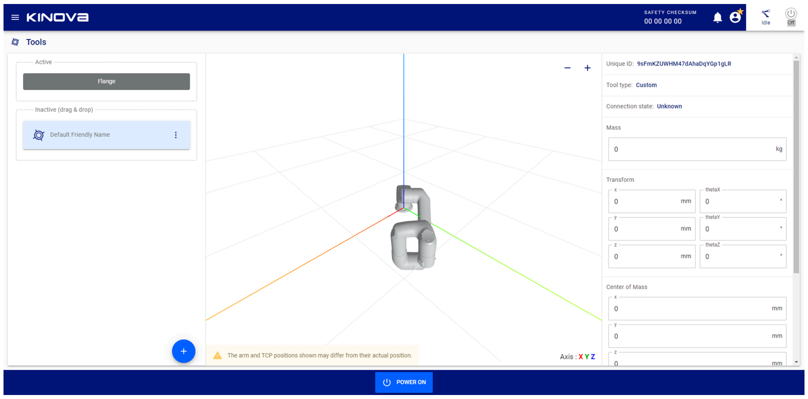
   The tools are listed on the left side in two groupings: **Active** and **Inactive (drag & drop)**. The *flange* is always listed in the **Active** tools list because all tools are attached to it.

   The center is a 3D view of the arm that can be moved around to help you visualize the tool reference frame. Tap **+** or **-** to zoom in or out. Hold and move the view around to have different viewpoints of the robot.

2. Select a tool in the **Inactive (drag & drop)** pane.

Tip: Kinova accessories and tools supported by plugins that are installed are available in the **Inactive (drag & drop)** pane.

The properties of the selected tool are displayed on the right.



3. Modify the values in the **Properties** panel.

Note: Configure the tools only when they are still in the **Inactive (drag & drop)** pane.

4. Drag each tool in the order they are assembled onto the flange of the *wrist*.

Adding tools

Custom tools do not come with their own *plugins*. They still can be added to the robot so that the robot can take the tool into consideration when following position commands.

Before you begin

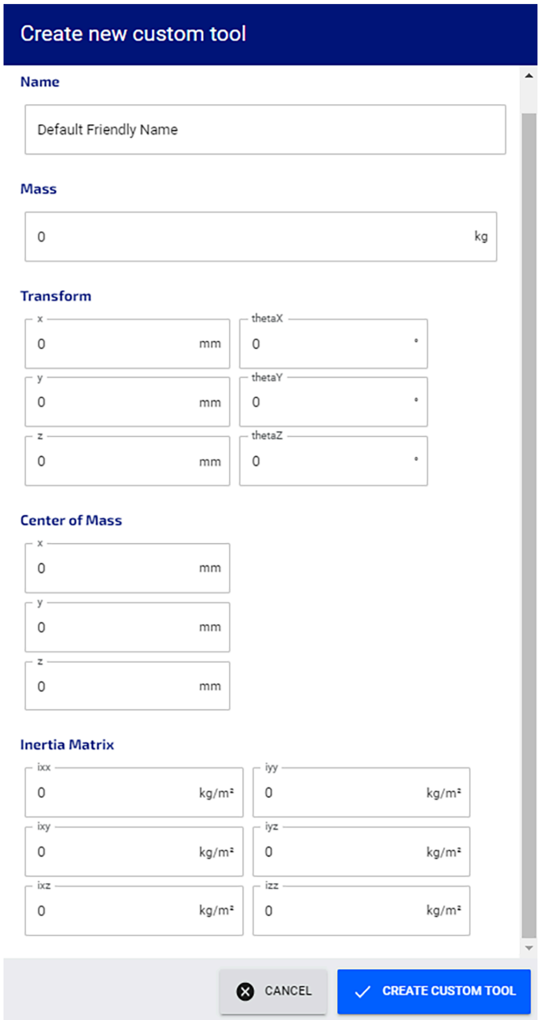
Know the mass, transformation, center of mass, and inertia matrix of your tool.

About this task

The properties of the custom tool affect the *TCP* of the entire robot.

Procedure

1. Tap .
   The **Create new custom tool** dialog launches.



2. Enter the information about the custom tool.
   - `Name`
   - `Mass`
   - `Transform: x, y, z` in mm
   - `Transform: θx, θy, θz` in °
   - `Center of Mass: x, y, z` in mm
   - `Inertia Matrix: ixx, ixy, ixz, iyy, iyz, izz` in kg/m^2
3. **DANGER:** Make sure the **Mass** and **Center of Mass** input for the payload correspond to the physical mass and center of mass of the payload installed on the robot.

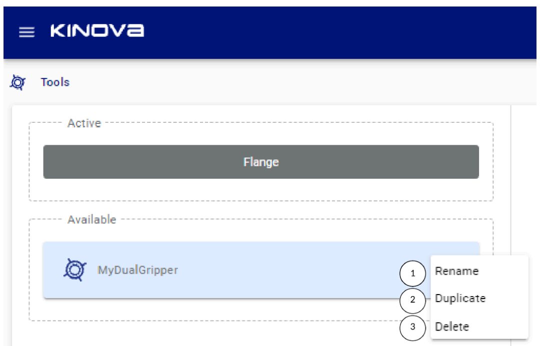
If the physical mass and physical center of mass of the payload does not match what is entered in the Create new custom tool form, it could lead to erratic behavior of the robot when it is used in Hand Guiding mode.

Always validate the mass and center of mass that is entered in Create new custom tool against the physical inertial parameters.

Tap **CREATE CUSTOM TOOL**.
The new tool is added to the list of available tools.

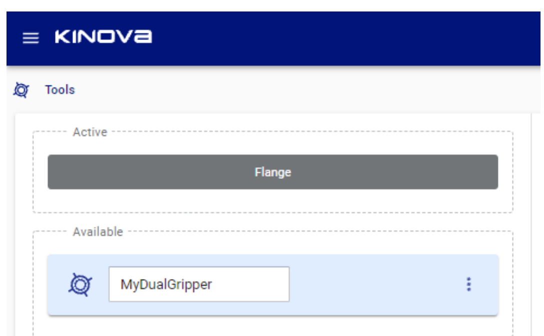
Other actions that can be taken on tools

There are three actions you can take on available custom tools.

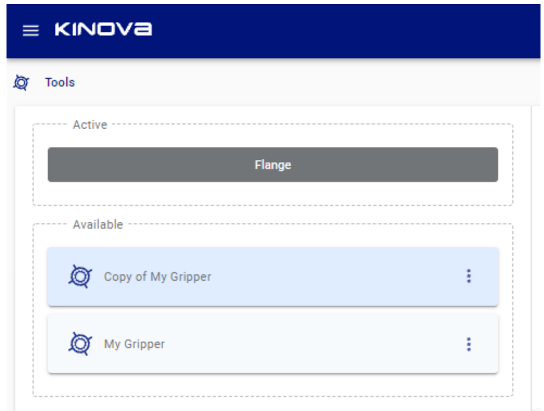


1 **Rename** Enter a new name and press Enter.

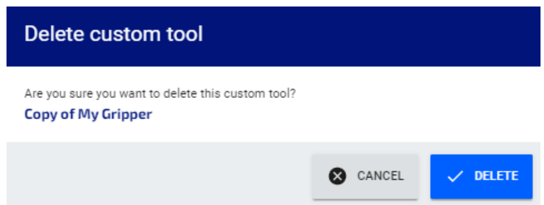
The name of the tool changes.



2	**Duplicate**	The tool is duplicated as "Copy of" followed by the name of the copied tool.



3	**Delete**	A confirmation dialog launches.



Tap **Delete** and the selected tool is no longer available.

Configurations for the robot settings

Some of the robot settings in the controller, the arm, and the force torque sensor in the wrist are configurable.

All configurations specific to the robot are accessed by tapping **Systems** > **Robot**.

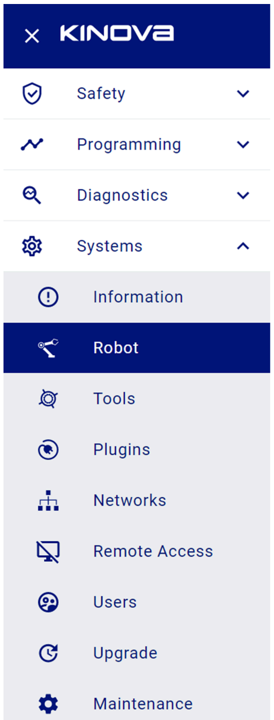


Figure 123: Access the robot settings

Related topics
Restoring factory defaults on page 194

Configuring the controller for robot controls

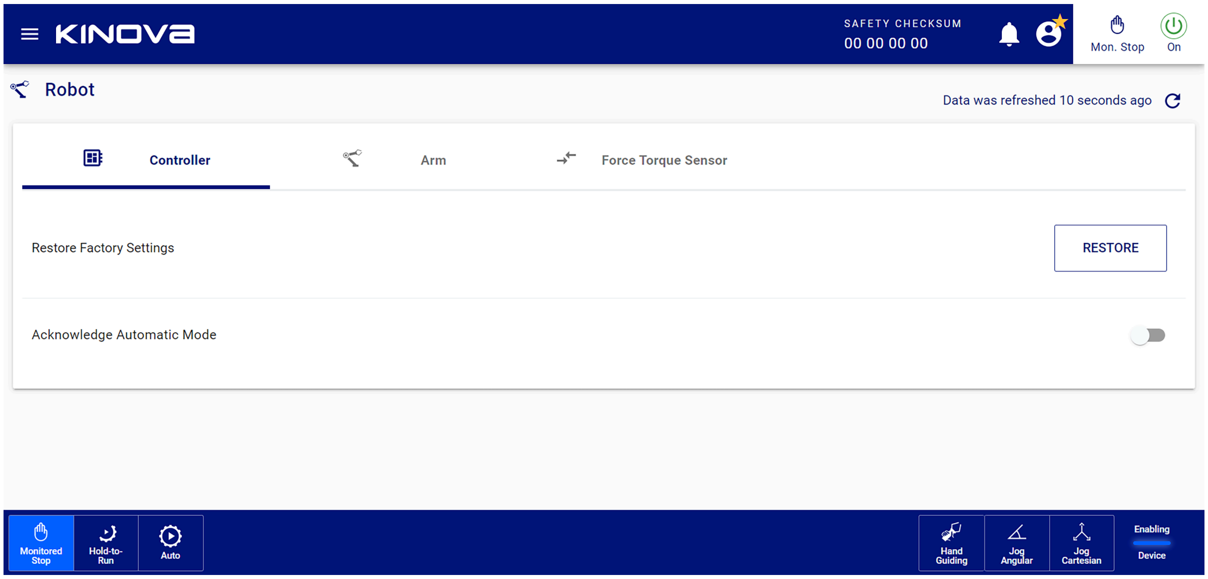
The **Robot Controller** page displays the configurations that can be changed for the controller.

About this task

All controller configurations for the robot must be performed when the robot is in Monitored Stop.

The only changes that can be made to the robot with respect to the controller involve using the settings from the factory, as well as using or overriding the Acknowledge button on the controller.

Access the **Controller** page by tapping **Systems** > **Robot**. The **Controller** page is the default page.



Procedure

1. Tap **RESTORE** in the **Restore Factory Settings** pane to reinstate factory settings in the *base*.

   Remember: When you restore factory settings, all configurations, *plugins*, and programs are deleted.

2. Tap the **Acknowledge Automatic Mode** slider to enable or disable the use the ACK button on the controller to confirm running a program in Automatic mode.

   Note: By default, the **Acknowledge Automatic Mode** is enabled.

Configuring the arm for robot controls

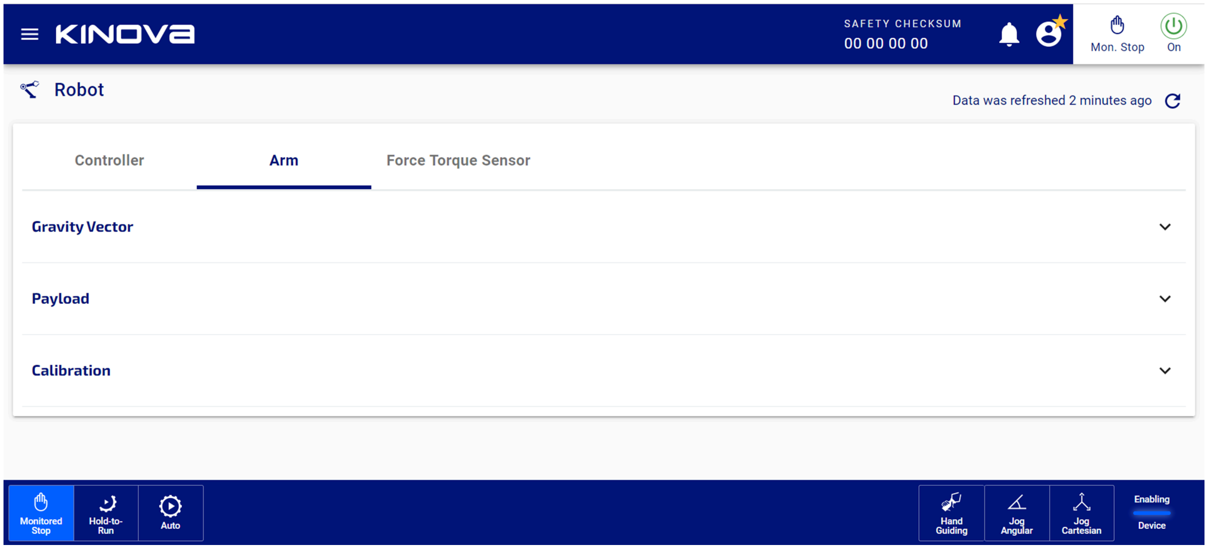
The **Robot Arm** page displays the configurations that can be changed for the arm.

About this task

All *arm* configurations for the robot must be performed when the robot is in Monitored Stop.

The only changes that can be made to the robot with respect to the arm involve its gravity vector, payload, calibration.

Access the **Arm** page by tapping **Systems** > **Robot** > **Arm**.

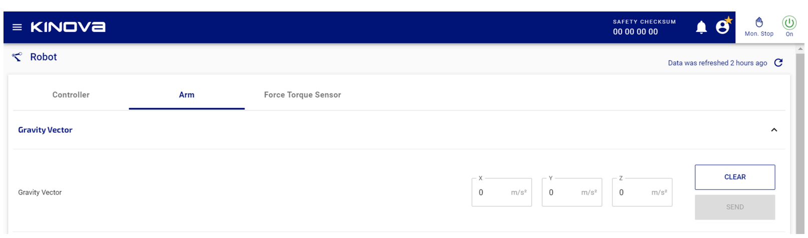


Procedure

Configuring the gravity vector of the arm

When the robot is unboxed, the robot is oriented sideways. The *controller* expects the robot to be oriented up with gravity pointing down. After the robot is installed, the values for **Gravity Vector** must be verified, especially when the robot is installed on a wall, ceiling, or other location that is not on a table or floor, because it is not oriented up.

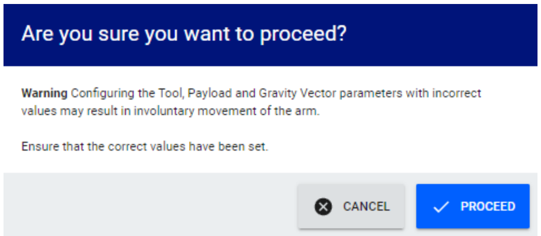
1. Tap **Gravity Vector** to access configurable parameters of the gravity vector of the arm.
   The **Gravity Vector** pane launches.

2. Enter values for x, y, and z in m/s^2, expressed in the Base reference system.

   Important: The gravity vector is not normalized and is not scaled. Make sure to keep the standard acceleration of gravity at 9.81 m/s^2; any other value affects the feeling of the robot when operating in Hand Guiding mode.

3. Tap **SEND**.
   A confirmation dialog launches when errors are suspected.



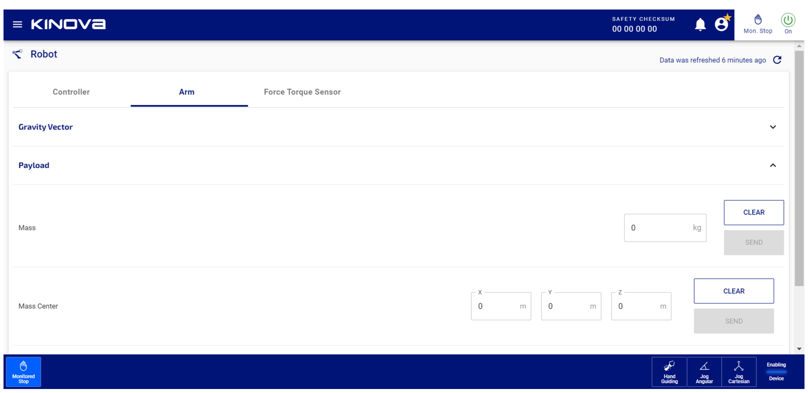
4. Tap **PROCEED** to send the gravity vector values to the controller or tap **CANCEL** to cancel the operation.
5. Tap **CLEAR** to erase the values entered that have not been sent.

Configuring the payload of the arm

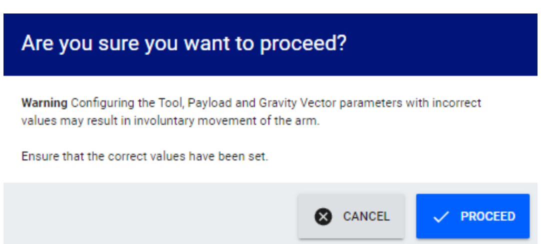
Important: *Payload* does not refer to the *end effector*. It refers to the mass of the object that the robot is picking.

The mass and its center of the object being picked affect the positioning and speed of the robot.

6. Tap **Payload** to access configurable parameters of the payload attached to the arm.
   The **Payload** pane launches.



7. Enter the `Mass` in kg of the object being picked.
8. Tap **SEND**
   A confirmation dialog launches.



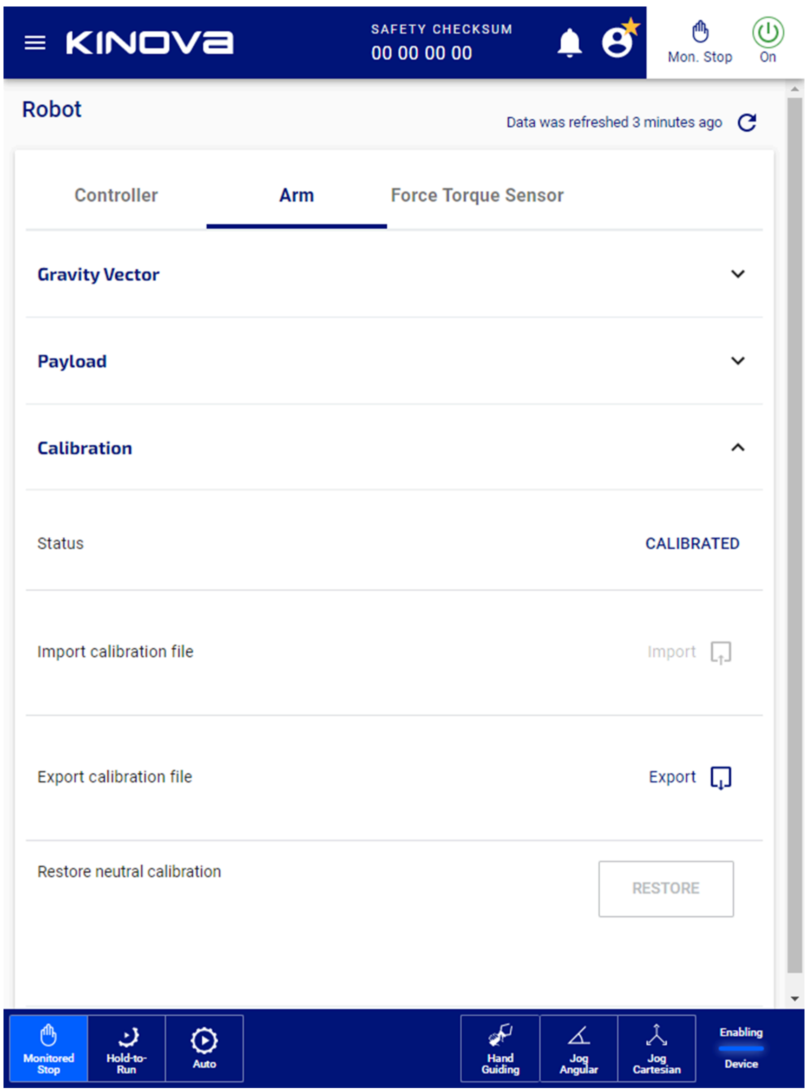
9. Tap **PROCEED** to send the mass of the object that is to be picked to the controller or tap **CANCEL** to cancel the values entered.

10. Tap **CLEAR** to erase the values from the page.
11. Enter the `Mass Center` in meters of the object being picked.
12. Tap **SEND**.
13. Tap **CLEAR** to erase the values from the page.

Configuring the calibration parameters of the arm

Calibrating the arm leads to increased accuracy in the movement. Calibration files contain the date and time of the calibration file, the serial number of the arm, and parametric values. There must be an active calibration file for the robot to operate.

14. Tap **Calibration** to access configurable parameters to calibrate the arm.
    The **Calibration** pane launches.



15. Look at **Status** for the current calibration status of the arm.

    The status of the calibration indicates whether there are issues before you start using the robot.

    - Calibrated
    - Not Calibrated
    - Waiting Arm Power On

16. Tap **Import** from the **Import calibration file** pane to import a calibration file when the controller is replaced.

    Calibration files can be imported only when the robot is idling with the arm switched off. The arm is recalibrated with the content of the file only after a reboot of the robot.

    Tip: The arm must not be switched on to import a calibration file.

17. Tap **Export** from the **Export calibration file** to export the current calibration file before changing the controller.

18. Tap **RESTORE** in the **Restore neutral calibration** pane.

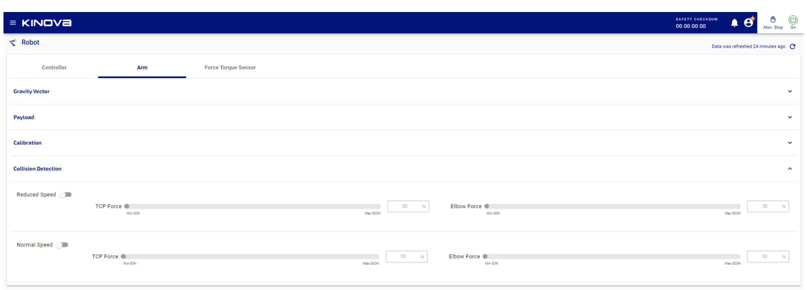
    Neutral calibration implies the arm is not calibrated.

    Tip: The arm must not be switched on to restore the calibration.

Configuring collision detection on the arm

19. Tap **Collision Detection** to access configurable parameters for detecting collisions with the arm.

    The **Collision Detection** pane launches.



20. Tap the **Reduced Speed** slider to enable or disable reduced speeds in collision detection.

21. Drag the **TCP Force** slider or enter an amount of force to be used as the amount of external force on the TCP that is needed for a collision to be detected when **Reduced Speed** is enabled.

    The range of the force on the TCP is from 50 N to 500 N.

22. Drag the **Elbow Force** slider or enter an amount of force to be used as the amount of external force on the elbow that is needed for a collision to be detected when **Reduced Speed** is enabled.

The range of the force on the elbow is from 50 N to 500 N.

23. Tap the **Normal Speed** slider to enable or disable normal speeds in collision detection.

24. Drag the **TCP Force** slider or enter an amount of force to be used as the amount of external force on the TCP that is needed for a collision to be detected when **Normal Speed** is enabled.

The range of the force on the TCP is from 50 N to 500 N.

25. Drag the **Elbow Force** slider or enter an amount of force to be used as the amount of external force on the elbow that is needed for a collision to be detected when **Normal Speed** is enabled.

The range of the force on the elbow is from 50 N to 500 N.

Configuring the force torque

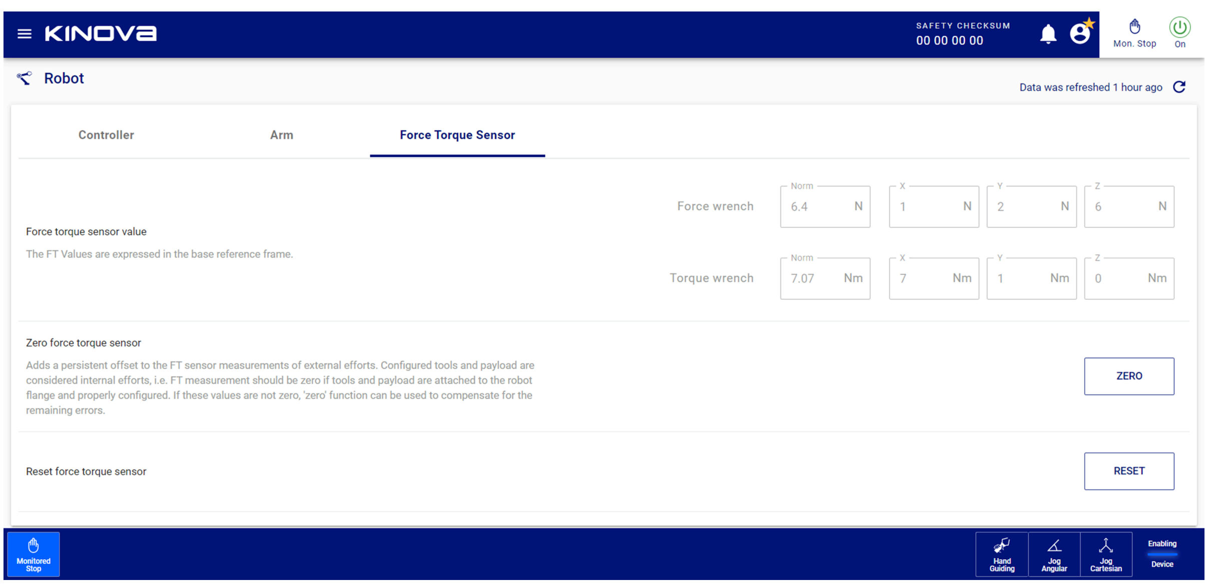
The **Robot Force Torque Sensor** page displays the configurations that can be changed for the force torque sensor in the wrist.

About this task

All configurations for force torque on the robot must be performed when the robot is in Monitored Stop.

Force torque is measured with respect to the base of the robot.

Access the **Force Torque Sensor** page by tapping **Systems** > **Robot** > **Force Torque**.



Procedure

1. Look at the force and torque values in the **Force torque sensor value** pane.

   The values are relative to the reference frame at the tool mounting flange of the robot. It displays the current **Force wrench**, measured in N, and **Torque wrench**, measured in N·m, of the robot.

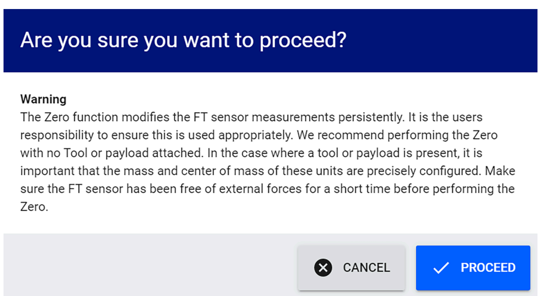
Performing a Zero function

2. Tap **ZERO** in the **Zero force torque sensor** pane.

   A zero function is necessary when the values listed in **Force wrench** are not 0 N or the values listed in **Torque wrench** are not 0 N·m when no external forces are applied. Shock and temperature can cause non-zero values.

   A confirmation dialog that recommends two situations for the robot to be in before proceeding with the Zero function.

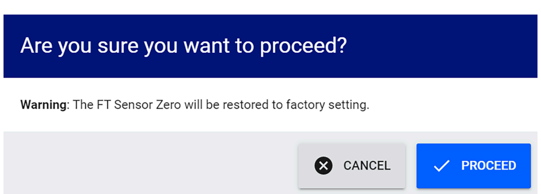
   - No tool or payload should be attached to the robot.
   - If a tool or payload is already attached to the robot, make sure the correct values for the mass and center of mass are entered accurately.



3. Tap **PROCEED** to zero the force torque sensor or tap **CANCEL** to cancel the Zero operation.

Resetting

4. Tap **RESET** in the **Reset force torque sensor** to restore the default offsets of the force torque sensor.
   A confirmation dialog launches with the warning that the force torque sensor is going to be set to its factory settings.



5. Tap **PROCEED** to return to factory settings of the force torque or **CANCEL** to cancel the reset operation.

Restoring factory defaults

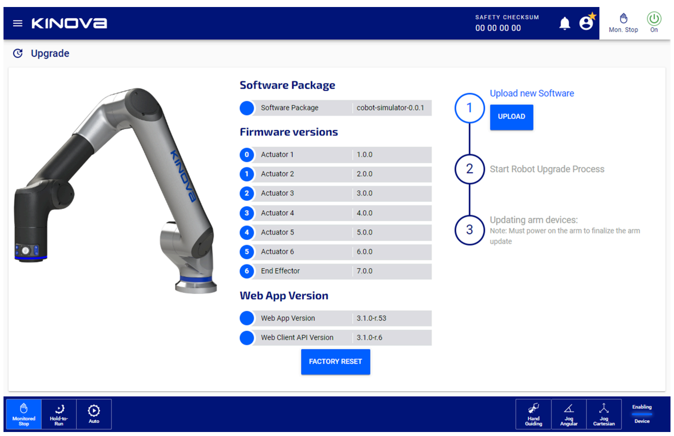
If ever you do not like an upgrade, restore the robot to its factory settings.

About this task

Only the *admin user* can update the software.

CAUTION: When you restore factory defaults, you lose all your configurations, settings, and programs. You also lose all the user profiles.

Procedure

1. Tap **Off** in the On section of *Kortex Web App* to turn off the *arm*.
2. Tap **Systems** > **Upgrade**.
   The **Upgrade** page launches.
   

3. Tap **FACTORY RESET**.
   All settings and configurations revert to the factory settings. All programs, *plugins*, and users, except for the admin user account, are deleted. All safety I/Os revert to their default values. The username and password of the admin user reverts to the `admin` and `admin`, respectively.

Related topics

Configurations for the robot settings on page 185

Maintenance mode

Kinova recommends backing up the Link 6 firmware operations and data from the internal processes on a regular basis to make troubleshooting issues easier.

Performing software maintenance is made of two actions.

- Back up the software configuration of your Link 6 system.
- Save log files that contain vital information about the *MPU*, *SCU*, and *MCU*, as well as the database and information when the system suddenly stops working the way it should.

Select **MAINTENANCE MODE** on the login dialog or tap **Systems** > **Maintenance** to perform maintenance when Link 6 is operating or when the arm is off

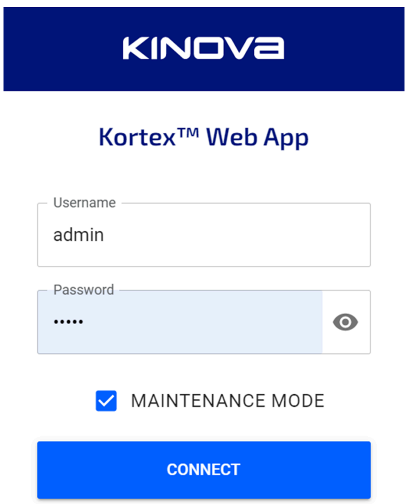


Figure 124: Login dialog

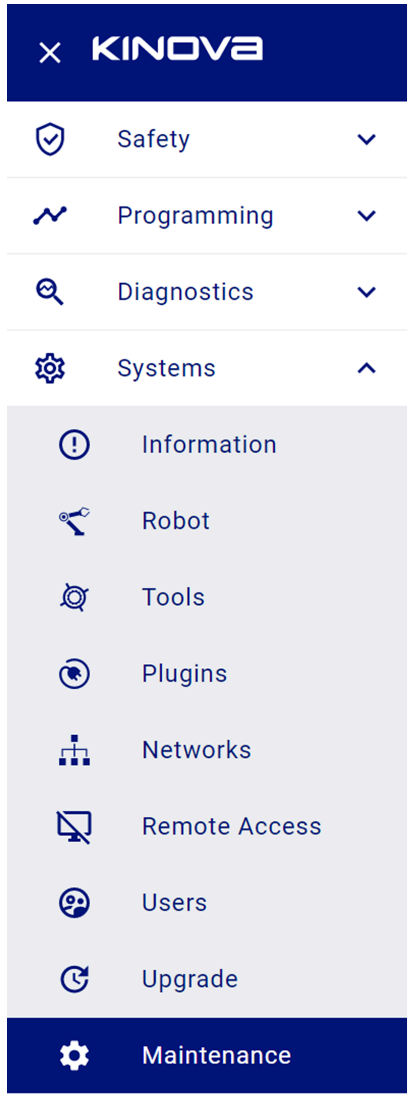


Figure 125: Accessing maintenance tasks

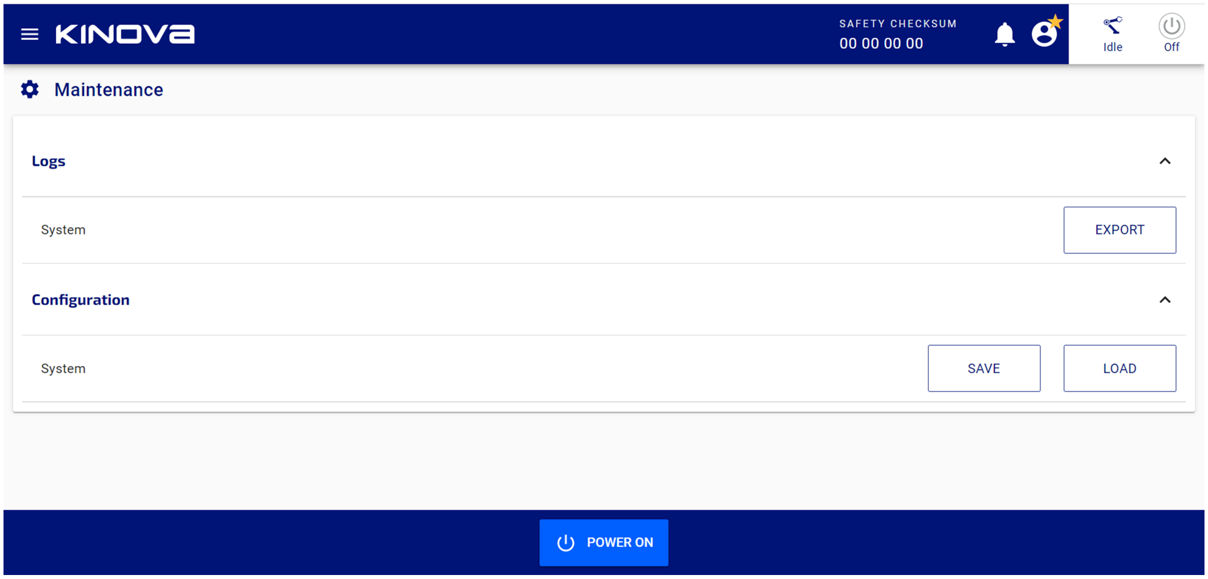


Figure 126: Maintenance page

The header of the **Maintenance** page has a slightly different look when it is accessed from the login dialog with **MAINTENANCE MODE** selected. There is no **Hamburger**,no **Quick Menu**, and no SAFETY CHECKSUM. The header has only the Kinova logo, User profile icon, and Mode indicator.

The footer of the **Maintenance** also has a slightly different look when it is accessed from the login dialog with **MAINTENANCE MODE** selected. The Robot control panel is not present.

Figure 127: Kortex Web App header in Maintenance mode

Figure 128: Regular Kortex Web App header

The **Maintenance** page has functionality differences that depend whether you are an *administrator* or an *operator*. The difference is that the administrator is the only user who can **LOAD** different configurations.

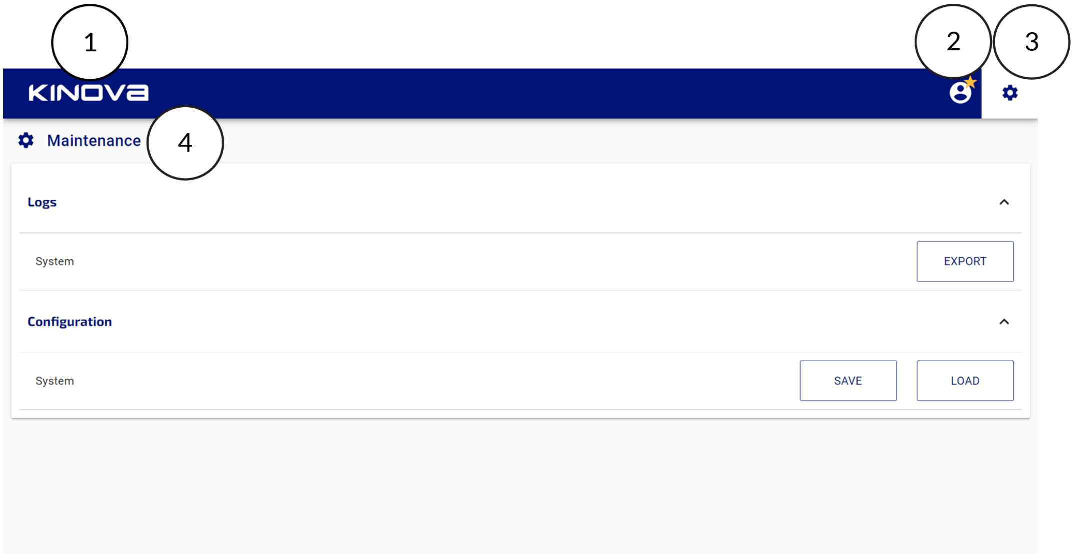


Figure 129: Maintenance page when log in with Maintenance Mode

1	The **Quick Menu** keeps the user on the **Maintenance** page.
2	The User profile icon is available so that you can log out.
3	The Mode indicator shows the user is in Maintenance mode.
4	The text indicates the name of the page.

The **Maintenance** page is divided into two panes.

- **Logs**
- **Configuration**

Logs

Collecting logs on a regular basis helps to troubleshoot issues with the Kinova support team.

Log files are `.ZIP` files that contain vital information about the robot. The log files are saved automatically with the file name `logs_[YYYY]_[MM]_[DD]_-_[hh]_[mm]_[ss]Z.zip`.

YYYY	four-digit year **Example:** 2023
MM	month, from 1 to 12
DD	day, from 1 to 31
hh	hour of the day, from 00 to 23
mm	minutes, from 00 to 59

ss	seconds, from 00 to 59
Z	timestamp is within the UTC time zone indicator

Log files do not exceed 2 GB. The exported log include logs from the current system *boot* and, if there is still space, it includes logs from up to the three previous system starts.

The information comes from the *MPU*, *SCU*, and *MCU*. The log also includes information about resource usage, the total amount of time the robot has been operating, memory statistics, version of the *SWU* that is currently installed, system and software interrupts, network adapter statistics, disk space statistics, mount points of the system, a snapshot of the database, a snapshot of the dump files generated by the controller.

All log files are password protected.

When a request to export the logs is made, no other request to export can be performed until the previous export is completed.

Exporting logs

The Kinova support team may ask you to export logs to help troubleshoot an issue while using Link 6.

Before you begin

Make sure you have a *USB* connected to the *controller* when you are exporting logs from the teach pendant. When you are using a computer instead, the log file exports to a location on your computer.

About this task

Log files are protected with passwords.

Procedure

1. Access the **Maintenance** page by performing one of two actions.
   - Log into Kortex Web App in Maintenance mode.
   - Tap **Systems** > **Maintenance**.
2. Tap **Logs**.
   The **Logs** pane expands.
3. Tap **EXPORT**.
   A file with all the system information is saved automatically to the *USB*.

Generated events

Events are generated whenever the system encounters an issue with an operation.

Each event is structured with the same format.

Table 73: Structure of generated events

Event heading	Description
Issue	What generated the event
Title	The title given to the issue
Severity	One of four levels of severity • CRITICAL • ERROR • INFO • WARNING
Timestamp	The time when the event is generated
Code	The error code associated with what generated the event
Explanation	An explanation of the problem if there is one
Suggestion	Zero or more suggestions on how to proceed

Configuration files

Backup configuration files protect you by saving the amount of time needed to return to known working configurations of a system that may have been altered for a minor change in a previous task.

Configuration files explicitly list information about how the robot is set up.

Table 74: Information saved in configuration files

Configurations	
Safety functions and their limits	Safety I/Os
Protection zones	State of acknowledge automatic mode
Gravity vector	Payload
Remote access	Collision detection
Energy limitations	Programs (names, properties, actions)

Persistent variables	Users and their encrypted passwords
Tool configurations	Plugin configurations
Plugin binary files	Plugin docker volumes
Other information	
Date and time of the backup file	Factory KIN
Serial number of the controller	Serial number of the arm **Tip:** The arm must be switched on to have its serial number saved.
Version of the *SWU*	List of all plugins that are installed
Version of each plugin that is installed	Checksum

Configuration files are `.ZIP` files that contain vital information about the robot. The configuration files are saved automatically with the file name `Link_Configuration_[XXXXX]_[YYYY]_[MM]_[DD][Thh]_[mm]_[ss][Z]<_#>.zip`.

XXXXX	Controller serial number
YYYY	Four-digit year the file is created **Example:** 2023
MM	Month the file is created, from 1 to 12
DD	Day the file is created, from 1 to 31
Thh	Hour the file is created, from 00 to 23
mm	Minutes the file is created, from 00 to 59
ss	Seconds the file is created, from 00 to 59
Z	Timestamp is within the UTC time zone indicator
_#	Optional sequential number added when the name of the configuration backup file matches another configuration file that already exists on the *USB* **Examples:** `Link_Configuration_XXXXX_YYYY-MM-DDThh-mm-ssZ.zip`,

`Link_Configuration_XXXXX_YYYY-MM-DDThh-mm-ssZ_1.zip`, `Link_Configuration_XXXXX_YYYY-MM-DDThh-mm-ssZ_2.zip`

Backing up the current configuration and setup

When you have determined the best configuration needed for Link 6 to operate a task, save the configuration.

Before you begin

Make sure to connect a *USB* to the controller before backing up your configuration.

About this task

Operators and Administrators can create the backup configuration files.

Remember: Only one backup configuration file can be created at a time.

Procedure

1. Access the **Maintenance** page by performing one of two actions.
   - Log into Kortex Web App in Maintenance mode.
   - Tap **Systems** > **Maintenance**.
2. Tap **Configuration**.
   The **Configuration** expands.
3. Tap **SAVE**.

Option	Description
To external device	File saves automatically to the device
To teach pendant	File saves to the user-selected destination on the USB on the controller.

Common errors when backing up configuration files

As with all things software, there may be times when error messages are displayed when backing up configuration files. When there is an error, the configuration files are not saved.

Issue	Error message
There is a request from the teach pendant to save a configuration file, but there is no *USB* in the *controller*.	`No USB storage device detected on the controller Front Panel USB port.`
There is a *Universal Serial Bus (USB)* inserted in the port on the controller, but the USB has less than 2 GB of space left on it.	`Not enough space in "<pathname>" to save the configuration backup file.` **Note:** <pathname> is the path where the backup configuration file is to be saved.

Issue:

Error code	9000
Severity	Error
Title	Logs Export Error
Explanation	The specified USB storage device is not mounted.
Suggestion	Try the following actions to resolve the issue:

a. Make sure the USB storage device is correctly plugged in the controller.

b. Unplug/plug the USB storage device.

c. Reboot the system.

d. Contact Kinova Support.

Issue: The USB has less than 2 GB of free space available for the log file.

Error code	9001
Severity	Error
Title	Logs Export Error
Explanation	The specified USB storage device has insufficient space to save the logs. The USB storage device needs at least 2 GB of available space.
Suggestion	Try the following actions to resolve the issue:

a. Remove USB storage device and free space.

b. Use a USB storage device with sufficient available space.

Issue: A log is in the process of being exported and there is another request to export a log.

Error code	9004
Severity	Error
Title	Logs Export Error
Explanation	A request to export logs is already being processed. A single request can be processed at a time.
Suggestion	Try the following actions to resolve the issue: a. Retry later. b. Reboot the system.

Loading configuration files

Load configuration files when you need to recover from a change or when you want to use a specific configuration for a specific task.

About this task

For safety reasons, only the *administrator* can load configuration files; the correct file must be restored to the robot given the tasks the robot needs to perform.

Note: Only one backup configuration file can be restored at a time.

Important: The arm must be switched off.

Procedure

1. Access the **Maintenance** page by performing one of two actions.
   - Log into Kortex Web App in Maintenance mode.
   - Tap **Systems** > **Maintenance**.
2. Tap **Configuration**.
   The **Configuration** expands.
3. Tap **LOAD**.
   The **File Upload** dialog launches with the message `The system will automatically reboot at the end of the configuration restoration.`

4. Select the configuration file to restore.

Option	Description
From external device	Tap **SELECT FILE** to select the configuration file from the device.
From teach pendant	Select the file from the list on the **File Upload** dialog.

Results

Success: The system reboots. The login dialog displays the message `The configuration was restored successfully.`

Partial success: The system reboots. The login dialog displays a list of parameters and plugins that could not be restored after the message `The configuration was partially restored. The following parameters/plugins were not restored:`

Failure: The system cancels the restoring of a configuration file when the current configurations of the network, force torque sensor offset, and arm calibration cannot be maintained. All the parameters that caused the cancellation are listed on Kortex Web App.

Collision detection

Link 6 has a feature in which collisions between the *arm*, and at least one object or person, is detected. When a collision is detected, the robot transitions to Monitored Stop.

When the *arm* is switched off or the robot is in Monitored Stop, configure thresholds for the amount of force detected before stopping the robot. Collision detection is triggered when the force values exceed the thresholds. After the fault is cleared, the robot resumes operation from where it was during its program.

Important: Although the thresholds triggers a fault, the thresholds do not prevent the robot from applying more force than the defined threshold.

Small thresholds decrease the amount of force that needs to be felt for a collision to be detected. Large thresholds increases the amount of force that needs to be felt for a collision to be detected.

Access the **Arm** by tapping **Systems** > **Robot** > **Arm**.

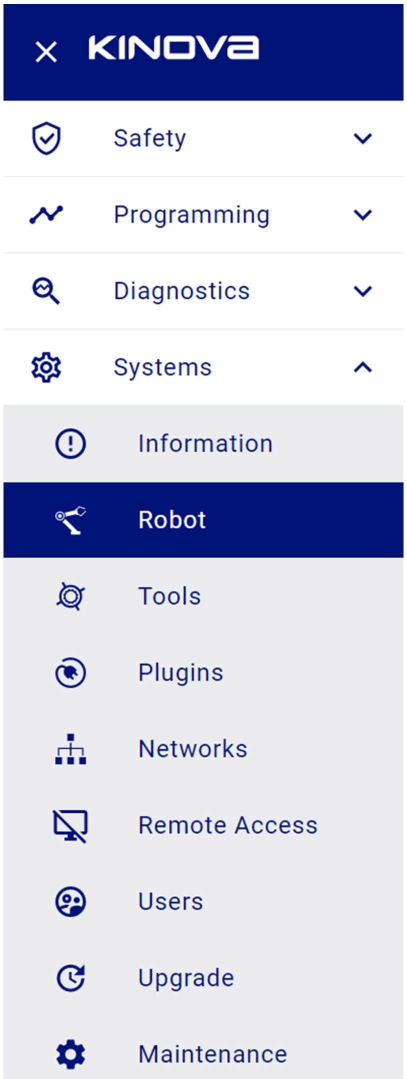


Figure 130: Accessing the Robot page

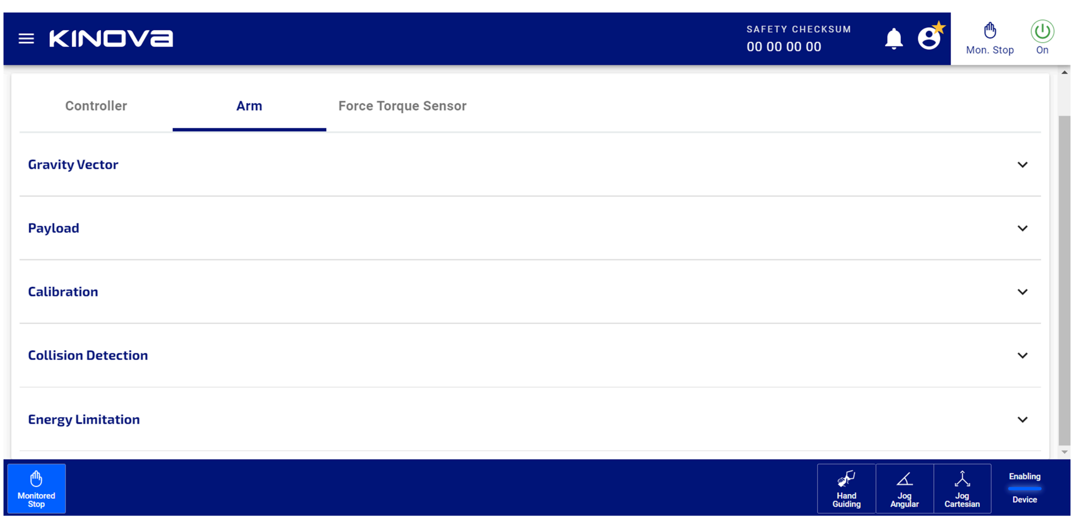


Figure 131: The Arm page

Tap **Collision Detection** to launch the **Collision Detection** pane.

Figure 132: Collision Detection pane on the Robot Arm page

Configure thresholds for collision detection in both reduced and normal speeds.

Table 75: Parameters to configure collision detection

Configuration	Description
Reduced Speed	Tap to switch between activating and deactivating collision detection at reduced speeds. Default setting: on
TCP Force Threshold	Use the slider or enter the threshold value in the text box. Range: 120 N to 500 N Default setting: 180 N
Elbow Force Threshold	Use the slider or enter the threshold value in the text box. Range: 120 N to 500 N Default setting: 180 N
Normal Speed	Tap to switch between activating and deactivating collision detection at normal speeds. Default setting: off
TCP Force Threshold	Use the slider or enter the threshold value in the text box. Range: 120 N to 500 N Default setting: 180 N

Configuration	Description
Elbow Force Threshold	Use the slider or enter the threshold value in the text box. Range: 120 N to 500 N Default setting: 180 N

Energy limitation

When the energy that can be used for the TCP and the elbow is limited, the speed that the robot travels along the trajectory is controlled.

Kinetic energy of the TCP and the elbow is monitored at all times. It is defined by the equation.

$$\text{kinetic energy} = \frac{1}{2} * m * vv^2$$

Small limits of energy means the robot travels slower along a trajectory. Large limits of energy means the robot travels faster along a trajectory.

When the *arm* is switched off or the robot is in Monitored Stop, configure the limits of energy that can be used to control the maximum speed of the *TCP* or *elbow* in real time.

Access the **Arm** by tapping **Systems** > **Robot** > **Arm**.

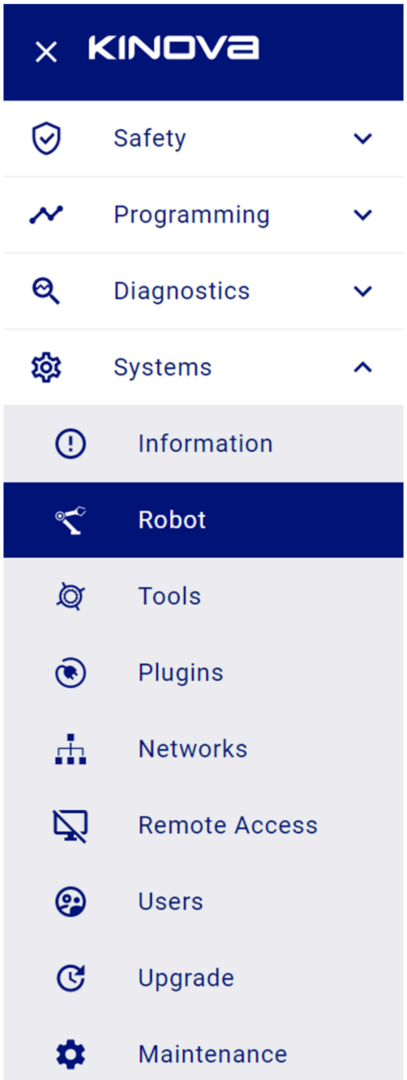


Figure 133: Accessing the Robot page

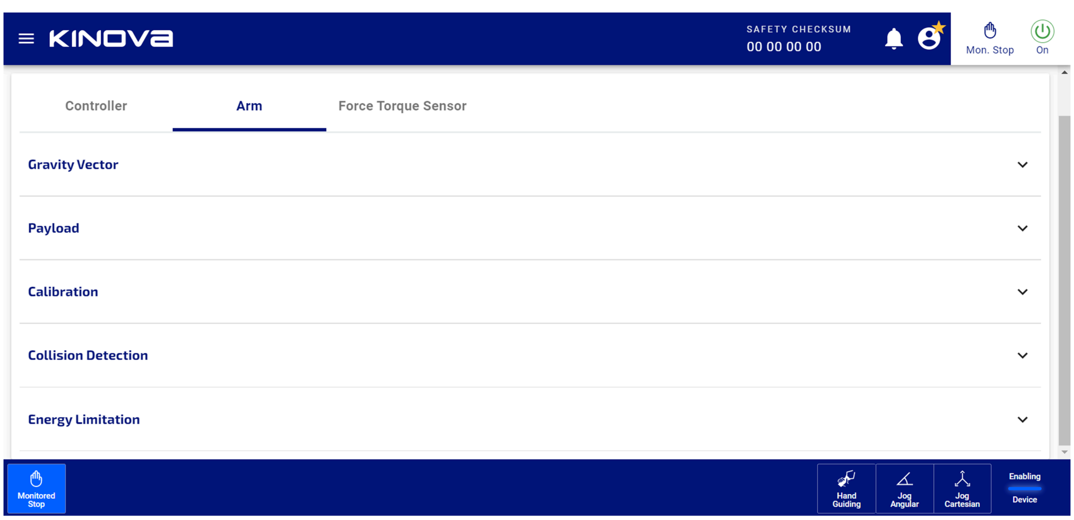


Figure 134: The Arm page

Tap **Energy Limitation** to launch the **Energy Limitation** pane.

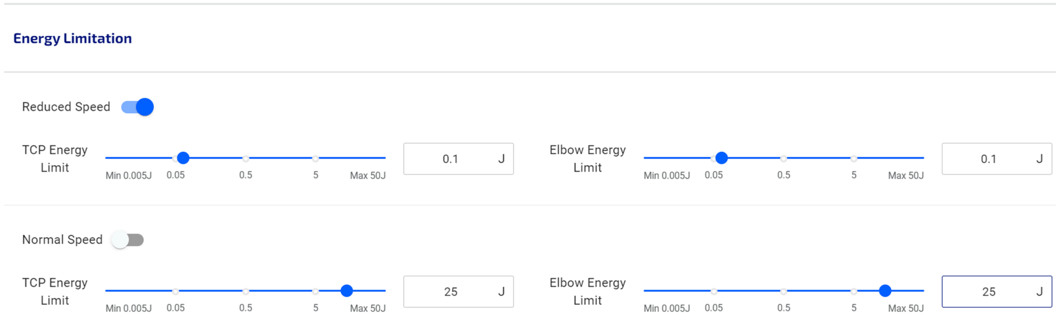


Figure 135: Energy Limitation pane on the Robot Arm page

Configure the energy limits for both reduced and normal speeds.

Table 76: Parameters to configure collision detection

Configuration	Description
Reduced Speed	Tap to switch between activating and deactivating collision detection at reduced speeds. Default setting: on
TCP Energy Limit	Use the slider or enter the limit in the text box. Range: 0.005 to 50 J Default setting: 0.1 N
Elbow Energy Limit	Use the slider or enter the limit in the text box. Range: 0.005 to 50 J Default setting: 0.1 J
Normal Speed	Tap to switch between activating and deactivating collision detection at normal speeds. Default setting: off
TCP Energy Limit	Use the slider or enter the threshold value in the text box. Range: 0.005 to 50 J Default setting: 25 N

Configuration	Description
Elbow Energy Limit	Use the slider or enter the threshold value in the text box. Range: 0.005 to 50 J Default setting: 25 N

Contact force reduction

It is important to minimize the amount of force used whenever two objects make contact with each other, whether that is between the robot and another inanimate object, or between the robot and a human.

To achieve a minimal amount of force in case of contact, configure several collision detection aspects of the robot and configure the amount of kinetic energy of the *arm* relative to other parts of the robot.

To reduce the total amount of force that is felt at the time of the robot coming into contact with another object you must reduce the thresholds of the amount of force that is felt during collision detection at the *TCP* and at the *elbow*, as well as the amount of kinetic energy that the arm can accumulate.

Expect higher collision forces under specific conditions; the performance of contact force reduction changes over the workspace of the robot.

- Higher collision forces can be expected because of the physical properties of the manipulator.
- Higher collision forces can be expected when there is a collision with an arm that is nearly fully extended in the radial direction because it takes longer to detect this type of collision. Use protection zones and limit the range of the joints to avoid motion of the TCP in areas where the arm is nearly fully extended.
- Reduced performance can be expected when the *end effector*, or tool, moves closer to the base.

A collision that is detected can be a false positive.

- The posture of the arm is close to a singularity.
- A payload is added and it is not configured correctly.
- The gravity vector is not defined correctly.
- Force threshold configuration values are smaller than the configuration values used in factory setting in Reduced speed mode.
- Force threshold configuration values are reduced, but the energy limits are not reduced.
- Energy limitation functionality is disabled.

- High acceleration constraints in waypoints.

There are many other ways for a false positive collision detection to occur.

Tip: Have the robot pick up objects as close as possible to the object's center of mass to avoid in-hand movement of the object when it is lifted.

Visual programming

To create programs for your Link 6, use the intuitive and easy-to-use *GUI*, Kortex Web App, and its tile-based, code-free visual programming interface.

By using the tiles, you can teach the robot its tasks, manipulate variables, and interact with external devices.

The result is accurate trajectories with reduced downtime for the robot.

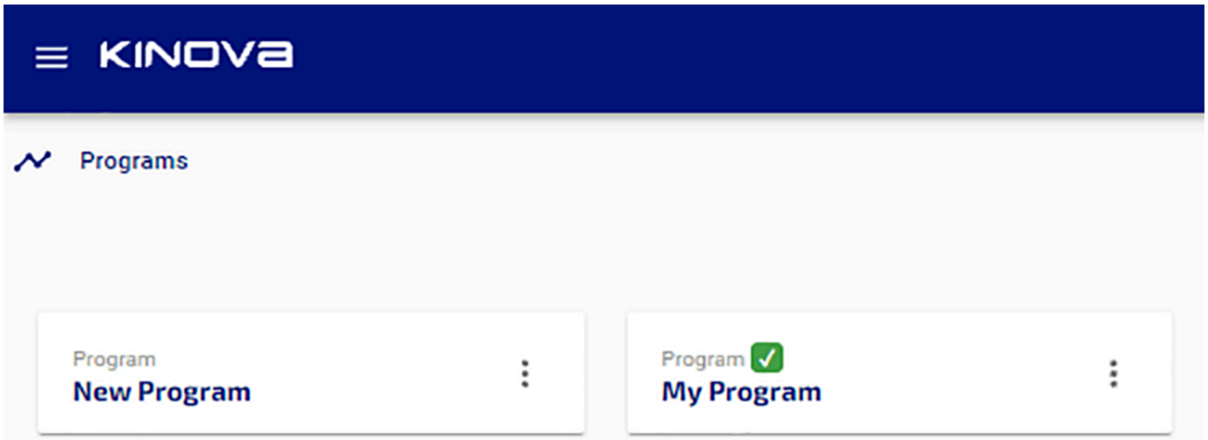


Figure 136: Programs page after programs are created

Overview of the Program page

The **Program** page is where programs are created and debugged.

Tap **Programming** > **Program** to access the visual programming interface.

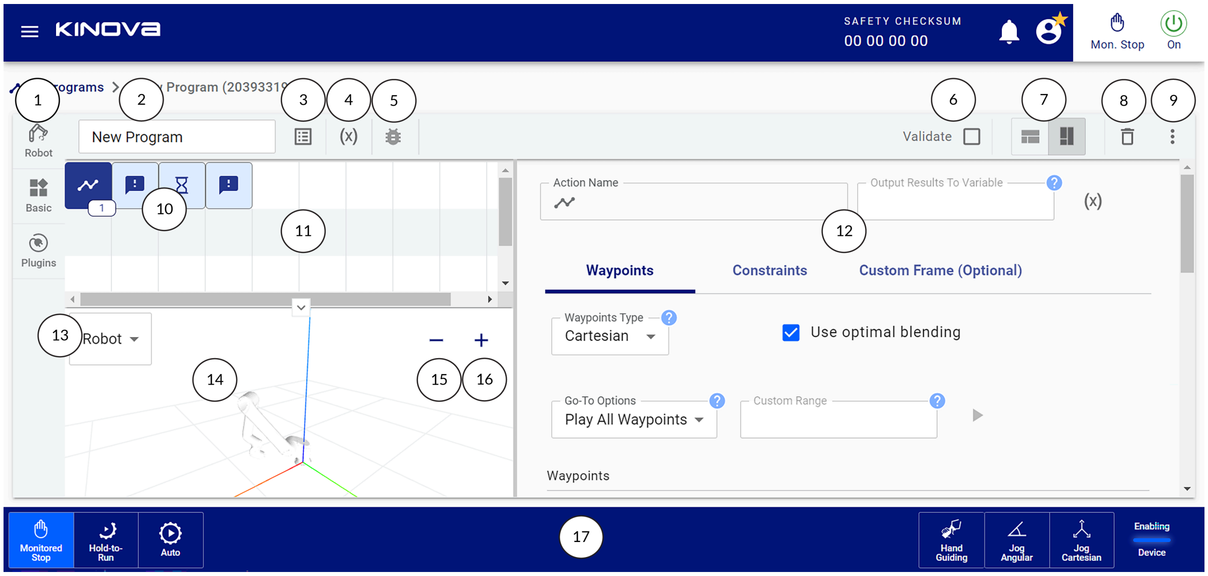


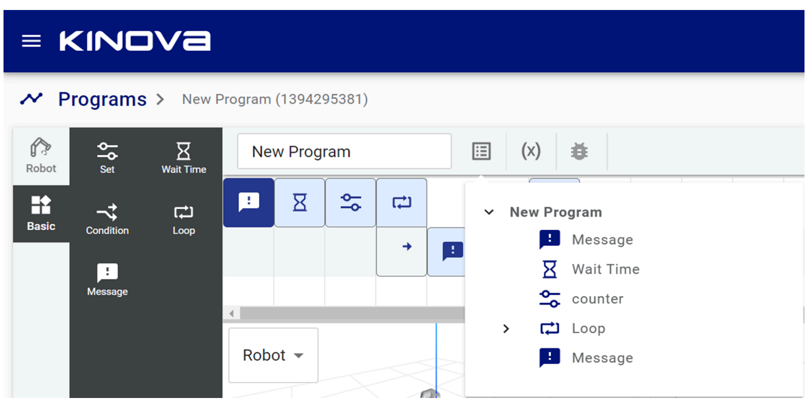
Figure 137: Parts of the Program page

1 Tap to access the **Action** menu.

2 Enter a `meaningful name` for the program.

Tip: Program names can have a maximum of 80 characters.

3 Tap **Tree layout** to view the tiles of the program in tree layout.



4 Tap **Variable Manager (X)** to launch the **Variable Manager** page and define variables within the scope of the program, as well as within the scope of persistent variables.

5 Tap **Debugger** to select the debugger tool and gather information about the program and its events while it runs. Tap **Debugger** to exit the debugger.

Tip: The **Debugger** button acts as a toggle button. When it is a dark blue, it is on. When it is gray, it is off.

6 Tap **Validate** to release the program to the operator.

7 Tap one of the two **Layout** buttons.

is the default layout where then entire **Sequence editor** is to the right of the **Sequence editor**.

has the **Sequence editor** across the top half of the **Program** page.

8 Tap **Delete** to delete the currently selected tile.

9 Tap the *kebab menu* (⋮) to access additional tasks that can be performed on the program in the **Sequence editor**. The additional tasks are **Duplicate**, **Delete**, and **DeleteExport JSON**.

10 Drag available **Tiles** from the Action into the **Sequence editor** to create a program.

11 **Sequence editor** is a drag-and-drop timeline editor of tiles that are performed by the program when it runs.

12 Tap on any tile in the program to populate the **Tile configuration** with the fields to create the correct parameters for the tile.

13 Select the visualization object.

Note: Currently, the only visualization object that is available is the robot.

14 The **Visualization** pane displays the robot in its current position.

15 Zoom out in the **Visualization** pane.

16 Zoom in the **Visualization** pane.

17 Use the icons in the Robot control panel to change modes of operation and to manipulate the playing of a program.

When *Kortex Web App* is operating in Hold-to-Run mode and in Automatic mode, the Robot control panel displays two more parts.

- Program play area
- Program speed control

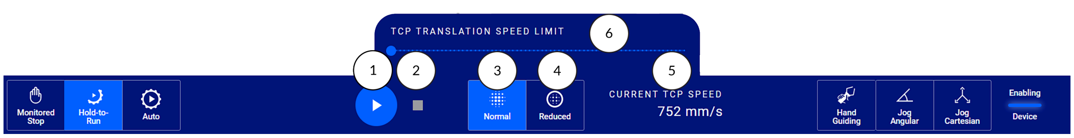


Figure 138: The parts of the Robot control panel

Program play area

1 Tap to play the program.

2 Tap to stop the program.

Program speed control

3 Tap to run the program at normal speed.

Important: It is recommended to run at reduced speed at least once before running at normal speeds when you are operating in Hold-to-Run; injury is possible.

4 Tap to run the program at reduced speed.

Important: It is recommended to run at the reduced speed when you are operating in Hold-to-Run.

5 The current translation speed set for the *TCP*.

6 Slide the **TCP Translation Speed Limit** slider gradually to scale the speed of your program from the Reduced speed values to the maximum speed.

Note: The **TCP Translation Speed Limit** slider is not available in Reduced speed in Hold-to-Run mode.

Variables

Define variables for the all programs and for specific *plugins* with a global *namespace*. Variables in the program affect how the robot functions.

To access the **Variables** page, tap **Programming** > **Variables**.

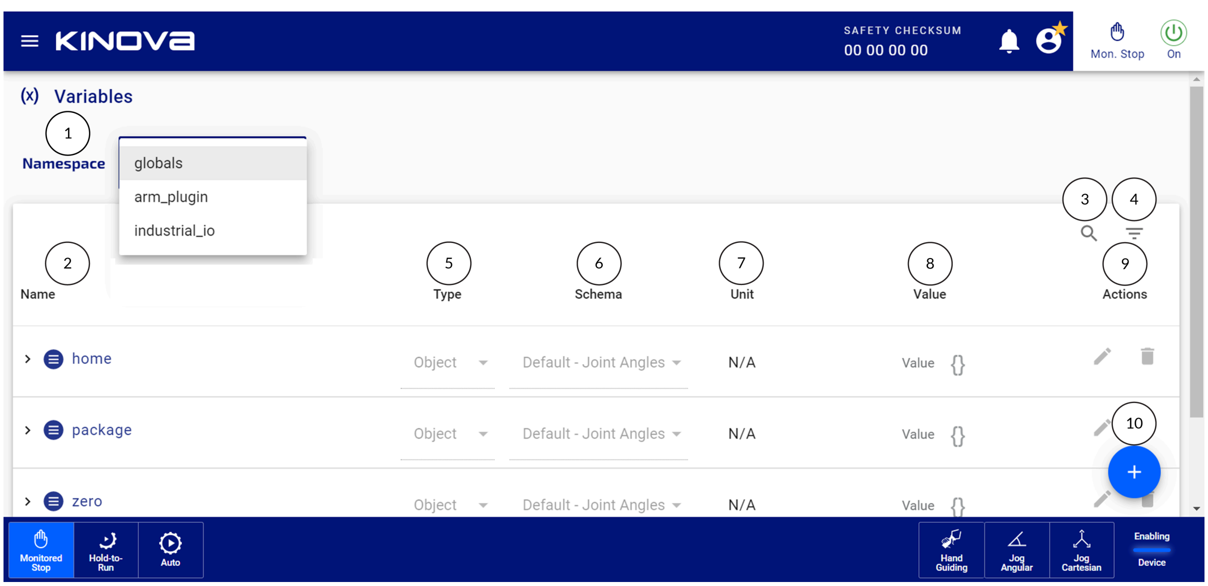


Figure 139: The Variables page

1 **Namespace** helps define the scope of the variable. Variables defined from this page are defined for all programs.

There are default namespaces.

- **globals** are global variables that are global for all programs. They include the pre-made global variables *home*, *package*, and *zero*.
- **arm_plugin** are global variables that are global for all programs and are directly related to the arm.
- **industrial_io** are global variables that are global for all programs and are directly related to the industrial I/Os of the robot.

In addition to the default namespaces, each plugin that is installed on Link 6 has their namespace available in the same list.

2 Name of the variable that is defined in the currently selected namespace.

3 Tap to search variable names, types, or schemas. The result is displayed on the page. For example, search *package* results in the page displaying all global variables named **package** if it exists as a global variable in the currently selected global namespace.

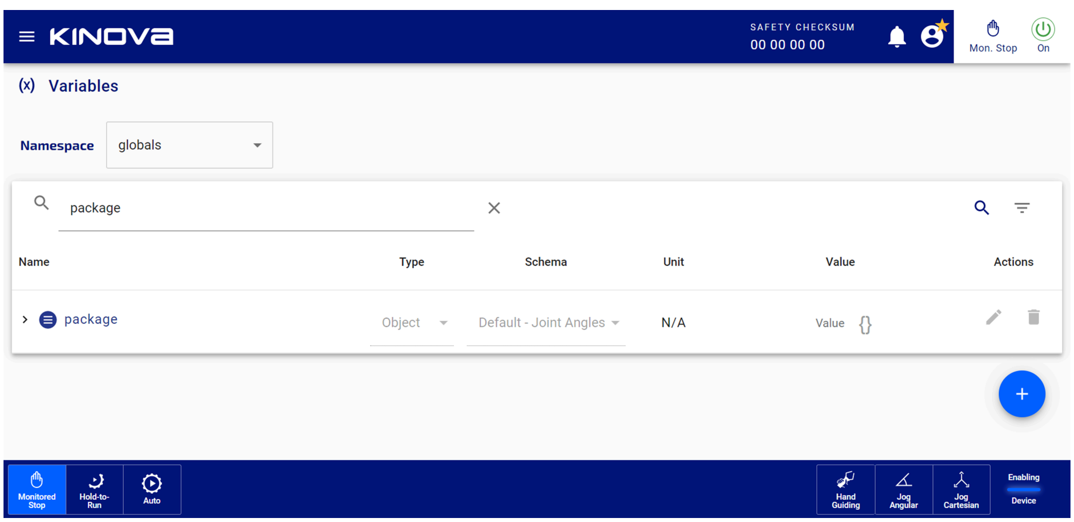


Figure 140: Result of a search for variable *package* in global namespace

4 Tap the filter icon (☰) to filter the list of variables by Type and by Schema on the **Variables** page.

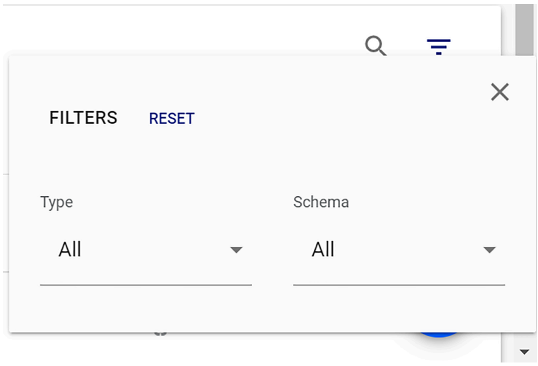


Figure 141: Filter panel

Select a filter type from the **Type** menu.

- **All**: Boolean, Number, Object, and String
- **Boolean**: True, False
- **Number**: Integer, float
- **Object**: Complex structure
- **String**: Text

Select a Schema type from the **Schema** menu list when the **Type** is **Object**. The menu lists all schemas that are available.

5 Define the type of the user-defined variable by selecting a type from the **Type** menu.

- **String**: Text
- **Boolean**: True, False
- **Number**: Integer, float
- **Object**: Complex structure

Tip: You must create a variable to define a type.

6 Define the schema of the user-defined variable of type **Object** by selecting a schema from the **Schema** menu.

- **Unspecified**: custom structure
- **Default - Pose**: Cartesian transformation
- **Default - Joint Angles**: List of angles, one for each joint
- **Default - Payload**
- **Arm Plugin - Matrix - Output**: List of poses
- **Arm Plugin - Waypoints - Output**: Cartesian and angular position of the waypoint
- **Industrial I/O Plugin - Read Input - Output**: Read input and its corresponding defined output

Remember: There are four types of read input.

- Controller digital: **DI_0** to **DI_7**
- Controller analog: **AIO_0** to **AIO_3**
- Wrist digital: **WRIST_DIGITAL_IO1** to **WRIST_DIGITAL_IO8**
- Wrist analog: **WRIST_AOI_1** and **WRIST_AOI_2**

Note: Read Input Output is used in conjunction with the Industrial I/O plugin.

Tip: You must create a variable to define a type.

Each plugin that is installed has its own set of schemas that you can use for new variables.

7 Define the value associated with the variable. Tap the {} icon to invoke the **JSON Object Editor** dialog and change the key-pair values as necessary.

8 When the modifiable global variable is of type **Number**, select the unit to be associated with the variable.

- unspecified
- s
- mm
- kg
- mm/s
- mm/s^2
- °/s
- °/s^2
- N
- °
- mA
- V
- °C
- N·m

Note: Units are used in context of selecting a number to validate its compatibility with the selected field.

9 There are at most two actions that can be taken for each modifiable global namespace variable.

- Edit ()
- Delete ()

Tap the edit icon () to modify the name of the user-defined global variable.

Note: The user-defined global variables that come with Link 6 cannot be edited.

Tap the delete icon () to delete the user-defined global variable.

Note: The global variables that come with Link 6 cannot be deleted.

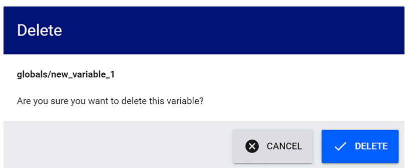


Figure 142: Delete user-defined global variable confirmation dialog

Important: Care must be taken when deleting a variable that is global to all programs. Always make sure there are no programs using the global variable before deleting it.

10 Tap to create a variable in the currently selected namespace.

Global variables created on the **Variables** page are accessible in the **Persistent Variables** pane of the **Variable Manager** page, as well as on the **Persistent Variables** pane of the **Select Variable** page of the currently selected program.

Related topics

Variable Management page on page 224

Industrial I/O Plugin on page 324

Examples of JSON statements

JavaScript Object Notation (JSON) is a key-value format for representing structured data. It is text-based, which makes it readable. There are a few things to know about its syntax.

Defining strings

Use double quotation marks around strings.

Examples:

```
{"name":"John"}
{"age":30}
{"sale":true}
```

```
meaningful_variable_name = {
"employee":{"name":"John", "age":30, "city":"Montreal"}
}
```

Access the value John by writing `${meaningful_variable_name.employee.name}`

Define a variable named *var* as an array of values.

```
{"n":[0,1,2,3,4,5]}
```

The result of `${var.n[0]}` is 0 and of `${var.n[3]}` is 3.

Defining iterators

When you define a variable as an iterator, it must be defined as type Number. The first index is always 0. `${var.n[${i}]}` outputs the value of each iteration over the array.

Adding a variable global to all programs from the Variables page

Variables can be created either for all programs a robot uses or for only a specific program. Variables created for all programs are made from the **Variables** page.

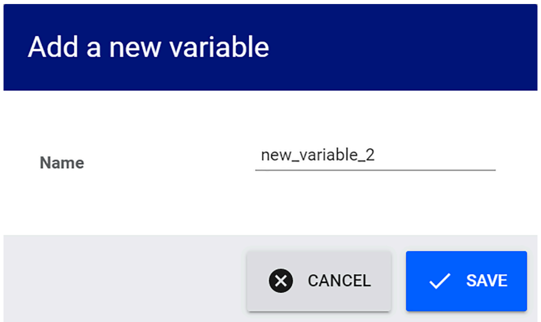
About this task

Variables that belong to the global namespace are accessible as **Persistent Variables** in the **Select Variable** and **Variable Manager** pages. Variables defined here are available to all programs.

Procedure

1. Tap **Programming** > **Variables**.
   The **Variables** page launches.
2. Select the appropriate *namespace* for the variable that is about to be created.

3. Tap .
   The **Create Variable** dialog launches.



4. Enter a meaningful name for the global variable.
   Underscores are allowed. Spaces and dashes are not allowed in the name.
5. Tap **SAVE**.
   The new variable is displayed on the **Variables** page. The variable can also be seen on the **Select Variable** and **Variable Manager** pages.
6. Select one **Type** for the variable.

Option	Description
String	A word, phrase, or sentence
Boolean	**False** or **True**
Number	Integer or floating point number
Object	A **Schema** • **Unspecified** • **Default - Pose** • **Default - Joint Angles** • **Default - Payload** • **Arm Plugin - Matrix - Output** • **Arm Plugin - Waypoints - Output** • **Industrial I/O Plugin - Read Input - Output** **Note:** The schemas that are listed are the default schemas. When additional plugins are installed, their associated schema are available from the **Object** menu.

7. (Only when the variable is of **Type Number**. Enter a number in the **Value** box and associate a **Unit** of measurement from the menu.

   Important: When you leave the **Unit** as **unspecified**, the variable has no unit associated with it.

8. (Only when the variable is of **Type Object**.) Tap the **Schema** for the type of object that variable represents.

9. (Only when the variable is of **Type Object**.) Tap the **Value** icon ({}) next and enter the values in the **JSON Object Editor** and tap **SAVE**.

   The information entered can then be available in entities of various tiles.

Figure 143: Editor for object type Unspecified

Figure 144: Editor for object type Default - Pose

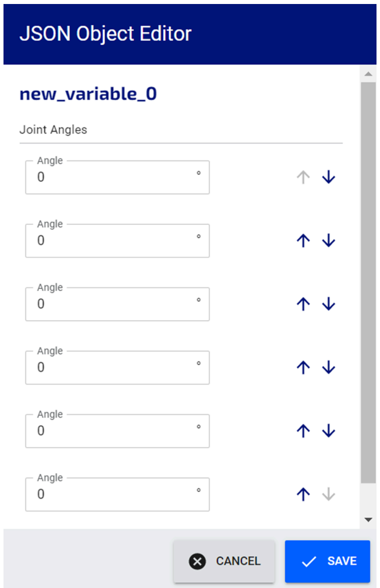


Figure 145: Editor for object type Default - Joint Angles

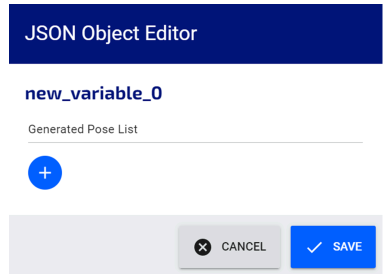


Figure 146: Editor object type Default - Payload

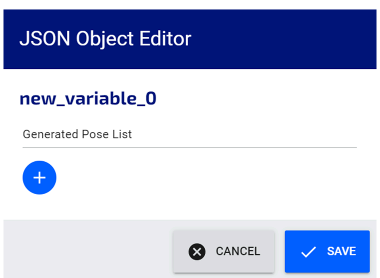


Figure 147: Editor object type Arm Plugin - Matrix - Output

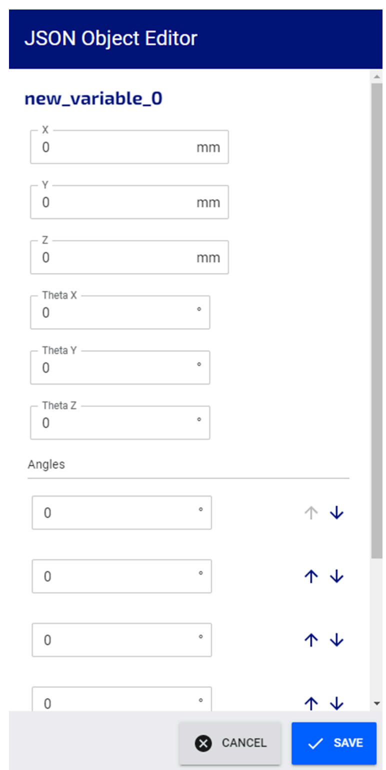


Figure 148: Editor object typeArm Plugin - Waypoints - Output when first invoked

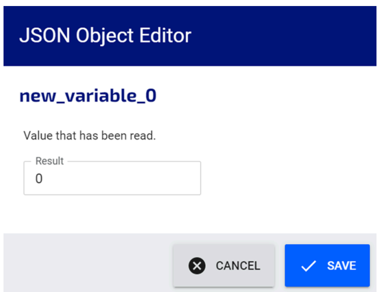


Figure 149: Editor object typeIndustrial I/O Plugin - Read Input - Output

Variable Management page

Variables in the program, either as specific to a program or as global to all programs, affect how the robot functions.

To access the **Variable Manager** page, select a program from the **Programs** page and tap the **Variable Manager** (x).

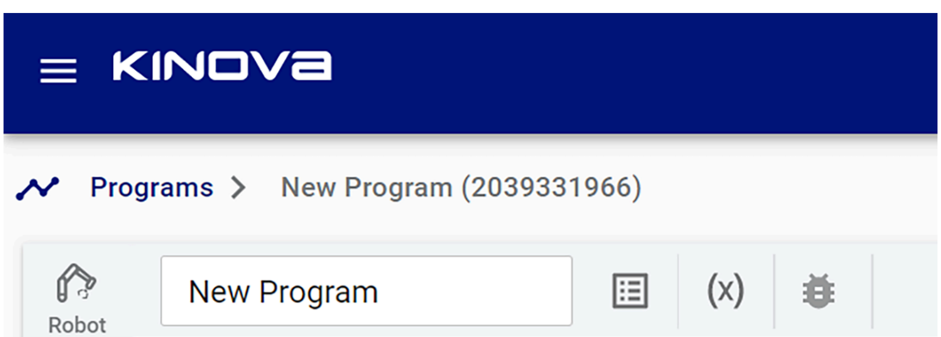


Figure 150: Variable Manager (x) button location on the Program page

The **Variable Manager** page launches. All the variables that currently defined for the current program are under the **Program Variables** pane. All the variables that are currently defined globally for all programs are in the **Persistent Variables** pane.

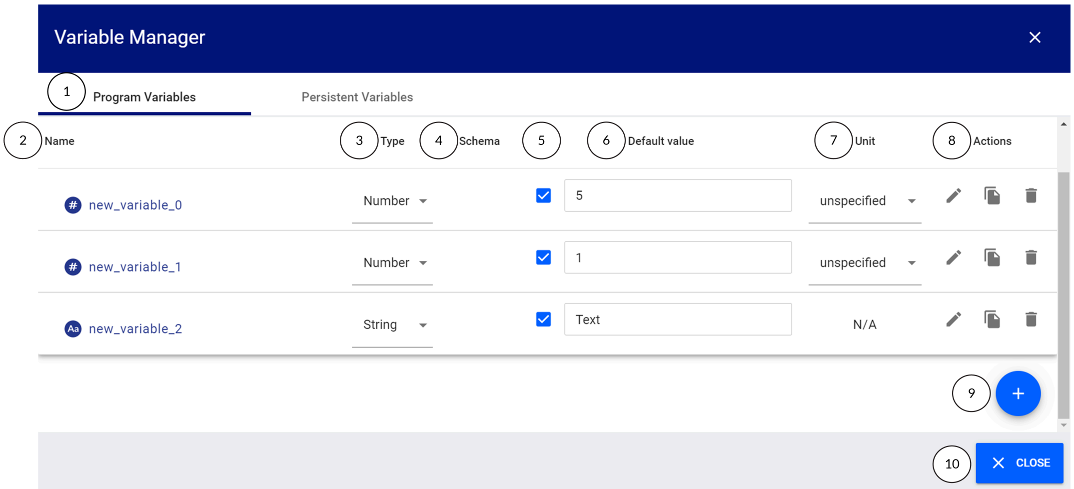


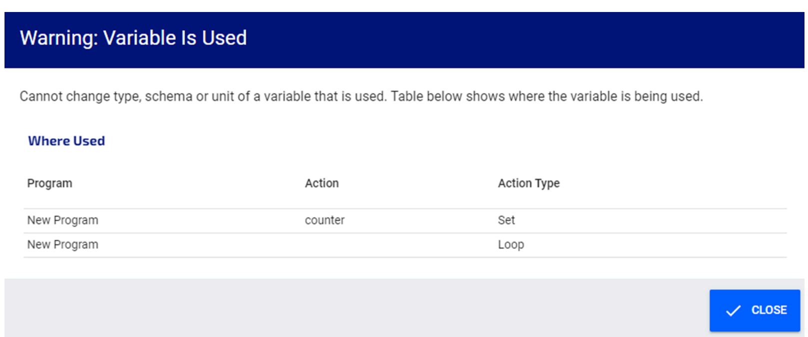
Figure 151: The Variable Manager page

1 Work with the variables that are defined only for the currently selected program by making sure **Program Variables** is selected.

2 Name of variable.

3 The **Type** of the variable is displayed. Tap it to change the type.

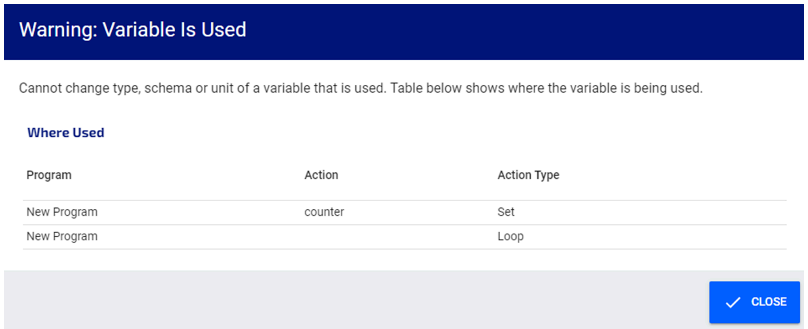
Note: You cannot change the type of the variable when the variable is being used in a program. If you attempt to change the type, a warning dialog launches.



Tap **CLOSE** to close the dialog. Nothing changes on the **Variable Manager** page.

4 The **Schema** of the variable is available only for **Object** types. Tap it to change the **Schema**.

Note: You cannot change the type of the variable when the variable is being used in a program. If you attempt to change the type, a warning dialog launches.



Tap **CLOSE** to close the dialog. Nothing changes on the **Variable Manager** page.

5 A check mark in a box indicates the variable has a default value. Tap to deselect the box when you do not want the variable to have a default value.

6 The **Default** value of the variable is displayed when the checkbox is selected. Enter a different default value for the variable.

7 The **Unit** associated with a variable of **Type Number** is displayed. Tap to change the unit, especially when the **Unit** is listed as **unspecified**.

8 Edit ( ), Duplicate ( ), and Delete ( ) are **Actions** that you can take on the variables.

Tap to edit the name of the variable. The **Edit variable** dialog launches and shows where the variable is already in use.

Important: When you change the name of a variable that is already in use, make sure to select **Rename all references**.

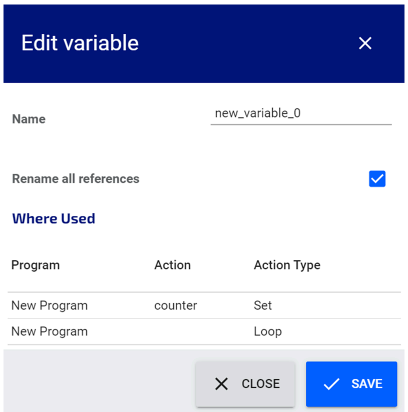


Figure 152: Edit variable dialog when the variable is in use

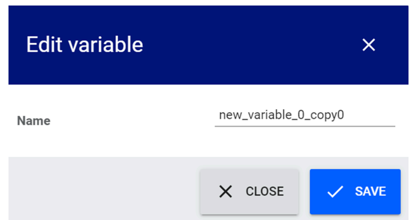


Figure 153: Edit variable dialog when the variable is not in use

Tap to duplicate the current variable. The duplicated variable has appends to the name of the copied variable the string "_copy" and the number of times the same variable has been copied.

Example: Suppose the variable name is *new_variable_0*. The first time it is duplicated, the new variable is named *new_variable_0_copy0*. The second time it is duplicated, the new variable is named *new_variable_0_copy1*.

Tap to delete the variable.

When the variable is being used, a **Delete** warning or confirmation dialog launches.

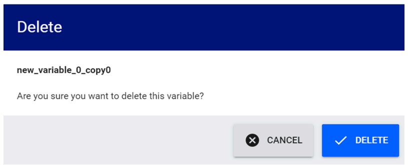


Figure 154: Delete confirmation dialog

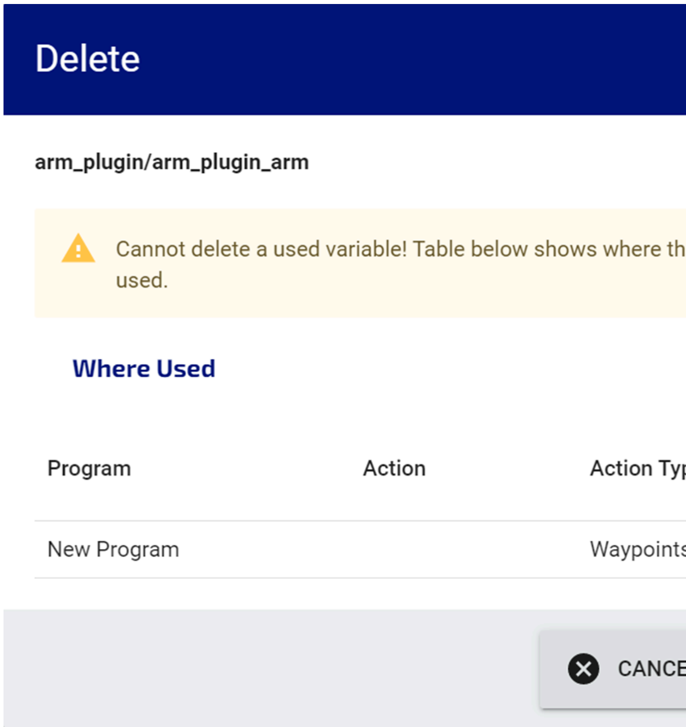


Figure 155: Delete warning dialog

9 Tap to create a variable for the program.

The **Create Variable** dialog launches. Enter a meaningful name for the new variable.

10 Tap **CLOSE** to close the **Variable Manager** page and return to the **Programs** page.

Related topics

Variables on page 215

Industrial I/O Plugin on page 324

Adding a variable global to all programs from the Variable Manager page

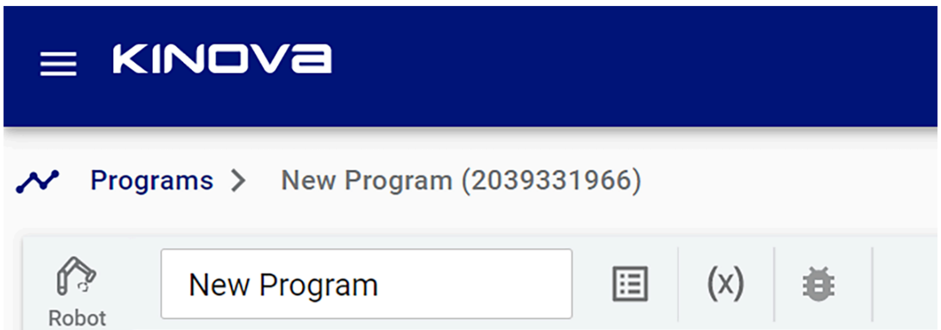
Variables can be created either for all programs a robot uses or for only a specific program. Variables created for all programs can also be made from the **Variable Manager** page.

About this task

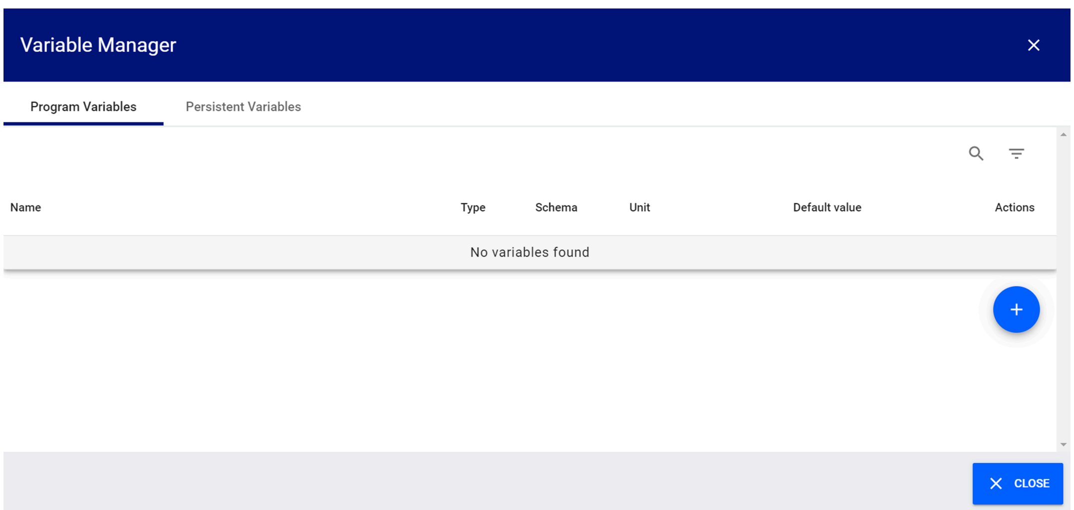
Variables that belong to the global namespace are accessible as **Persistent Variables** in the **Select Variable** and **Variable Manager** pages. Variables defined here are available to all programs.

Procedure

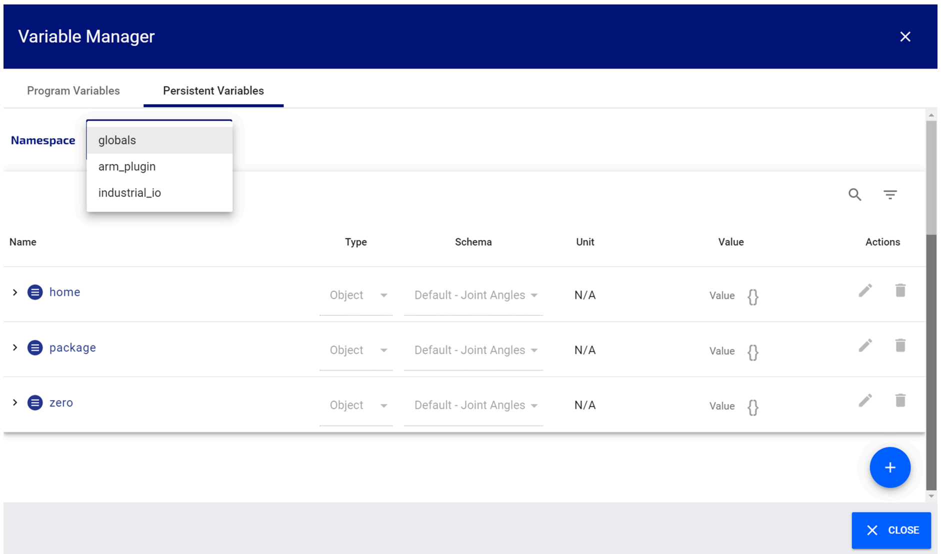
1. Select a program from the **Programs** page.
2. Tap **Variable Manager** (x).



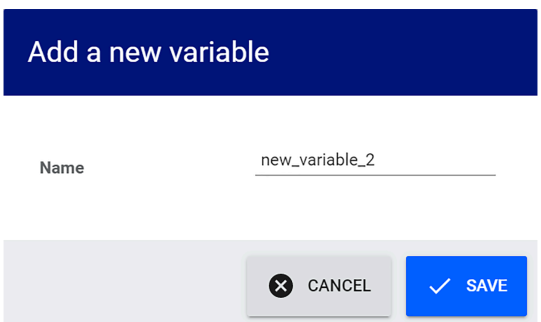
The **Variable Manager** page launches.



3. Tap the **Persistent Variables** pane.



4. Select the appropriate *namespace* for the variable that is about to be created.
5. Tap .
   The **Create Variable** dialog launches.



6. Enter a meaningful name for the global variable.
   Underscores are allowed. Spaces and dashes are not allowed in the name.
7. Tap **SAVE**.
   The new variable is displayed on the **Variable Manager** page. The variable can also be seen on the **Select Variable** and **Variables** pages.
8. Select one **Type** for the variable.

Option	Description
String	A word, phrase, or sentence
Boolean	**False** or **True**
Number	Integer or floating point number
Object	A **Schema**

Option	Description
	• **Unspecified** • **Default - Pose** • **Default - Joint Angles** • **Default - Payload** • **Arm Plugin - Matrix - Output** • **Arm Plugin - Waypoints - Output** • **Industrial I/O Plugin - Read Input - Output** **Note:** The schemas that are listed are the default schemas. When additional plugins are installed, their associated schema are available from the **Object** menu.

9. (Only when the variable is of **Type Number**. Enter a number in the **Value** box and associate a **Unit** of measurement from the menu.

   Important: When you leave the **Unit** as **unspecified**, the variable has no unit associated with it.

10. (Only when the variable is of **Type Object**.) Tap the **Schema** for the type of object that variable represents.

11. Tap the **Value** icon ({}) to enter the values in the **JSON Object Editor** and tap **SAVE**.

    The information entered can then be available in entities of various tiles.

Figure 156: Editor for object type Unspecified

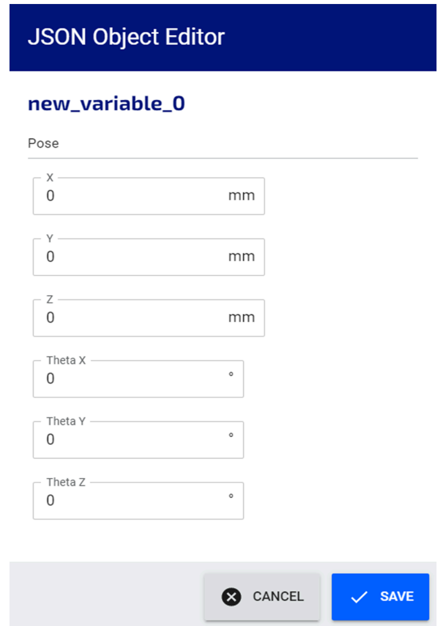


Figure 157: Editor for object type Default - Pose

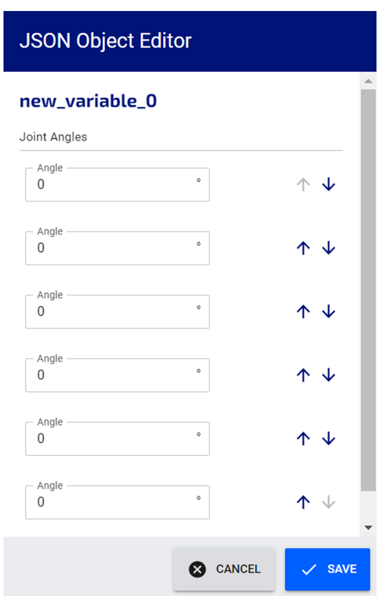


Figure 158: Editor for object type Default - Joint Angles

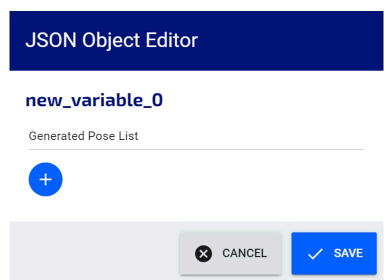


Figure 159: Editor object type Default - Payload

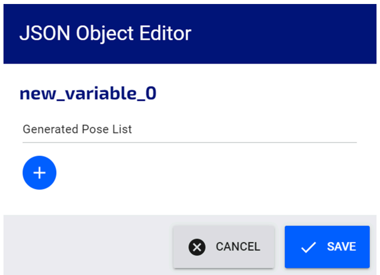


Figure 160: Editor object type Arm Plugin - Matrix - Output

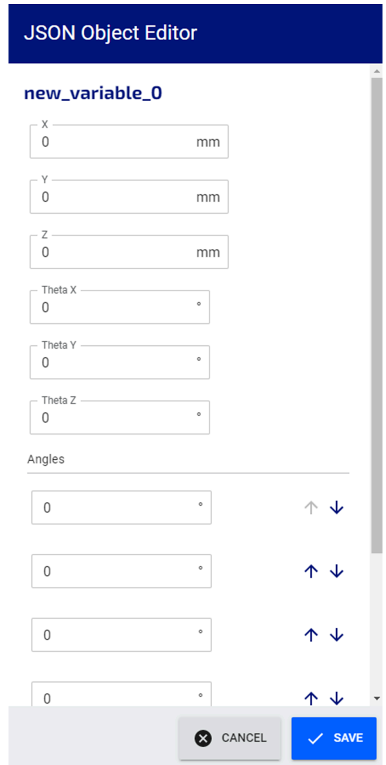


Figure 161: Editor object typeArm Plugin - Waypoints - Output when first invoked

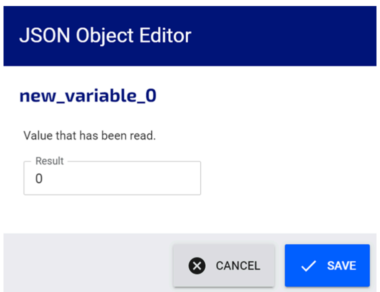


Figure 162: Editor object typeIndustrial I/O Plugin - Read Input - Output

Adding a variable global only to a specific program

A program can benefit from global variables that are specific only to that program. Variables that are global only within a program are created from the **Variable Manager** page.

About this task

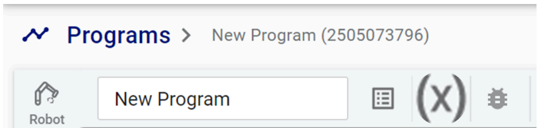
Variables that are added to the program namespace are available as **Program Variables** on the **Variable Manager** and **Select Variable** pages.

Procedure

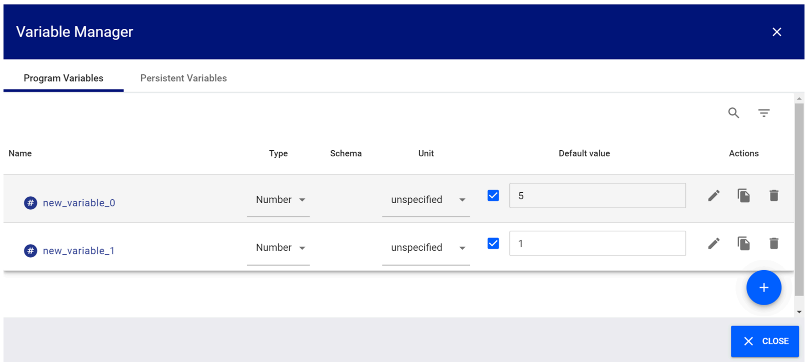
1. Tap **Programming** > **Programs** and select a program.

   The **Program** page launches.

2. Tap **(X)** in the banner beside the name of the program.

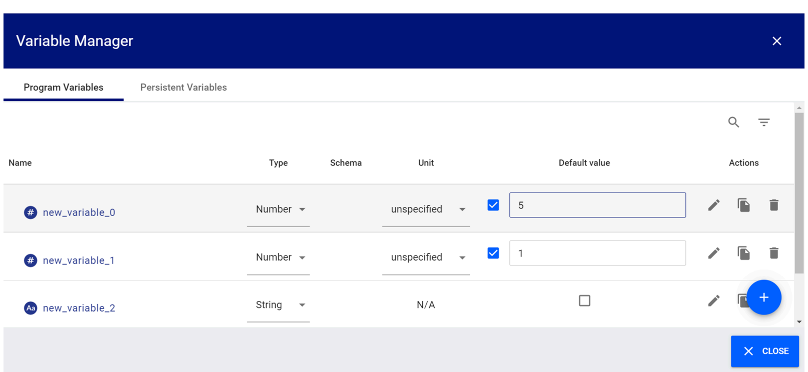


The **Variable Manager** page launches.



3. Tap **Program Variables**.
4. Tap +.
   The **Create Variable** dialog launches.

5. Enter a meaningful name for the variable.
   Underscores are allowed. Spaces and dashes are not allowed in the name.
6. Tap **SAVE**.
   The new variable is displayed on the **Program Variables** pane of the **Variable Manager** page.



7. Select a **Type** for the variable.

Option	Description
String	A word, phrase, or sentence
Boolean	**False** or **True**
Number	Enter an integer or floating point number. Select a **Unit** for the number.
Object	A **Schema** • **Unspecified** • **Default - Pose** • **Default - Joint Angles** • **Default - Payload** • **Arm Plugin - Matrix - Output** • **Arm Plugin - Waypoints - Output** • **Industrial I/O Plugin - Read Input - Output** **Note:** The schemas that are listed are the default schemas.

8. (Only when the variable is of **Type Object**.) Tap the **Schema** for the type of object that variable represents.

9. (Only when the variable is of **Type Object**.) Tap the **Value** icon ({}) next and enter the values in the **JSON Object Editor** and tap **SAVE**.

   The variable is available in entities of various tiles.

Figure 163: Editor for object type Unspecified

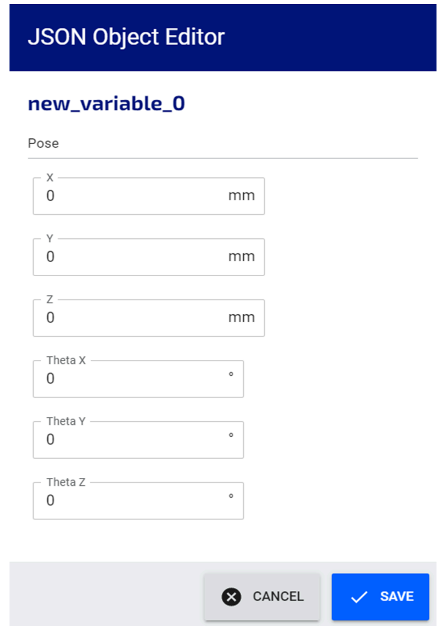


Figure 164: Editor for object type Default - Pose

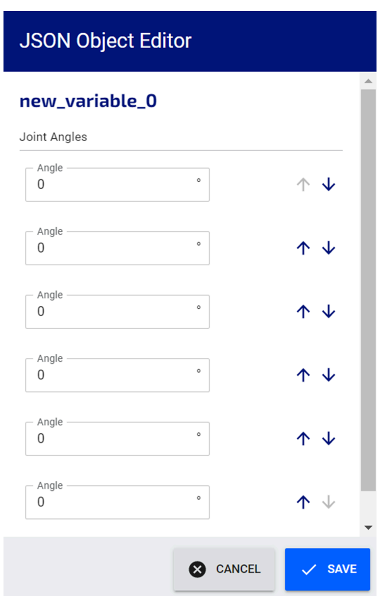


Figure 165: Editor for object type Default - Joint Angles

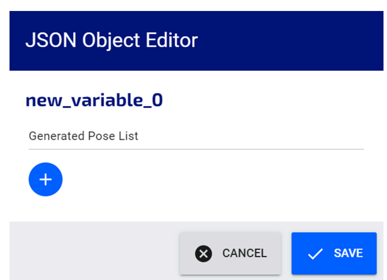


Figure 166: Editor object type Default - Payload

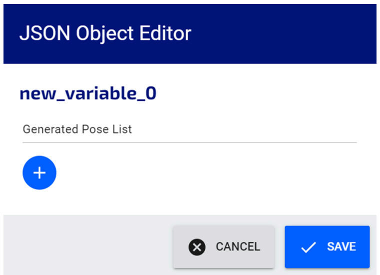


Figure 167: Editor object type Arm Plugin - Matrix - Output

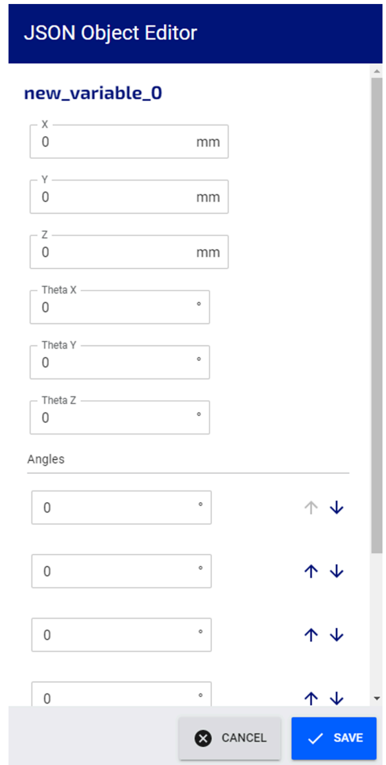


Figure 168: Editor object type Arm Plugin - Waypoints - Output

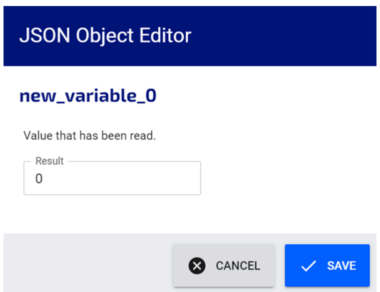
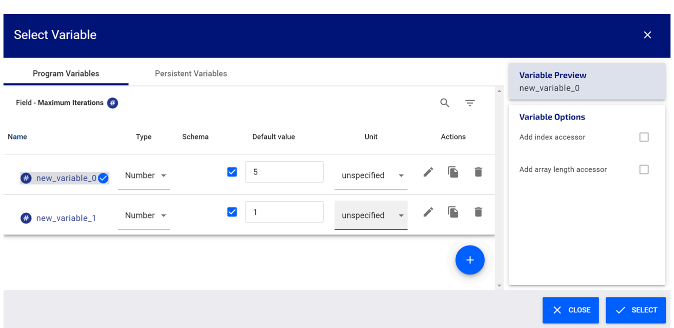


Figure 169: Editor object typeIndustrial I/O Plugin - Read Input - Output

10. Tap **CLOSE**.
    The **Select Variable** page closes and the **Program** page is displayed.

Tip: You can also create variables in context from the **Select Variable** page.

11. Tap a tile in the **Sequence editor** pane.
12. Tap **(X)** beside **Output Results To Variable**.
    The **Select Variable** page launches with the name of the field to which the variable is to be associated.



Selecting a variable

Sometimes you need to select a variable that is already defined and use it in a program.

About this task

It assumes that all program variables and global variables are already defined.

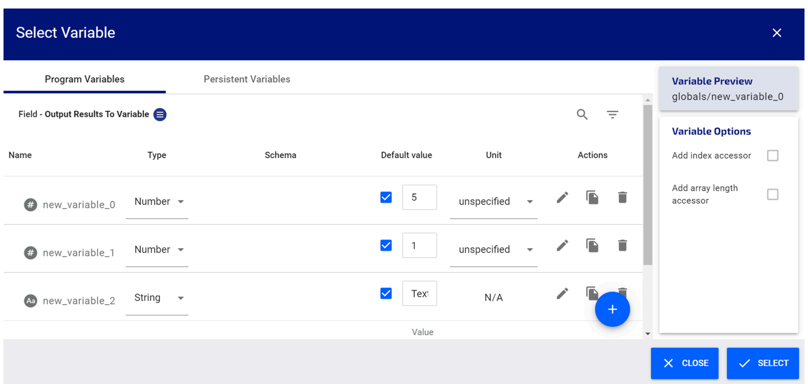
Procedure

1. Drag and drop an action tile in the **Sequence editor**.
2. Tap **(X)** beside any field in the **Tile configuration** pane.

   Note: The **(X)** that launches the **Select Variable** page is displayed in different locations in the **Tile configuration**, depending on which tile is selected in the program.

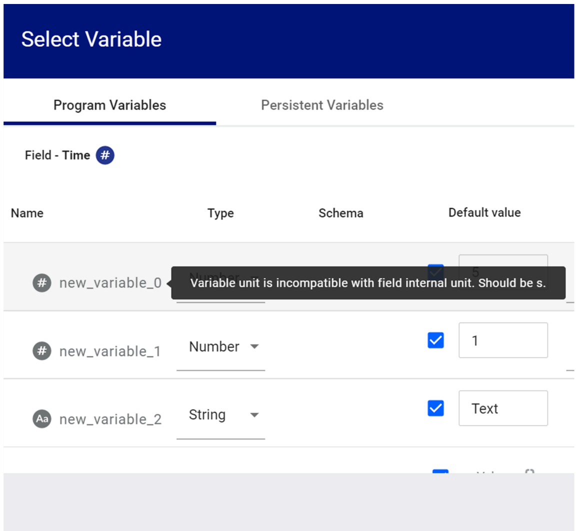
   The selectable variables are filtered, based on the type, schema, and units associated with the field.

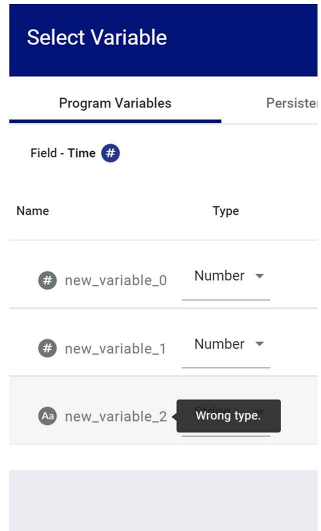
   The **Select Variable** page launches.



3. Tap the variable that you want.

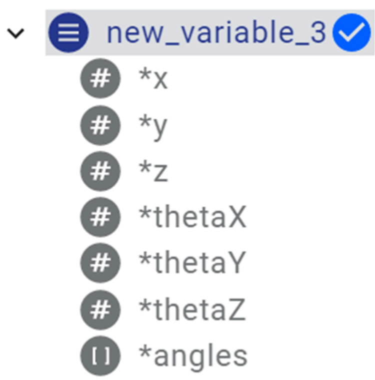
   Note: You can select from **Program Variables** or **Persistent Variables**. If the variable you select is not of the same type as the field, a tip displays an error message.





Note: If the variable that you need is not defined as a variable global to the program and is not defined as a persistent variable, you can create the variable by tapping + and creating the desired variable.

The name of the variable is highlighted and a check mark displays beside the name.



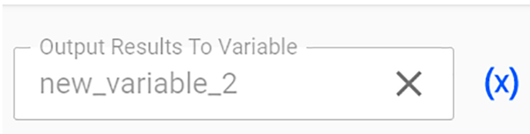
Tapping on the small arrow beside the variable name reveals the content of that variable if it is a complex variable.

4. Tap **SELECT**.

The **Select Variable** page closes.

The name of the selected variable displays in the **Output Results To Variable** text box.

Remember: The name of the selected variable is in the text box where the **Select Variable** is invoked.



Tip: Clear any text field that is filled with a selected variable or selected variable element by tapping the X in the text field.

Deleting a variable that is being used in a program

When you duplicate a program to speed up program creation, you may find that some variables are not needed for your new program. You need to delete the variable, but only from the program.

About this task

Variables that are used in programs are on the **Program Variables** pane of the **Variable Manager** page. When you attempt to delete a program variable that is used by the program from the **Variable Manager** page, a **Delete** warning dialog launches.

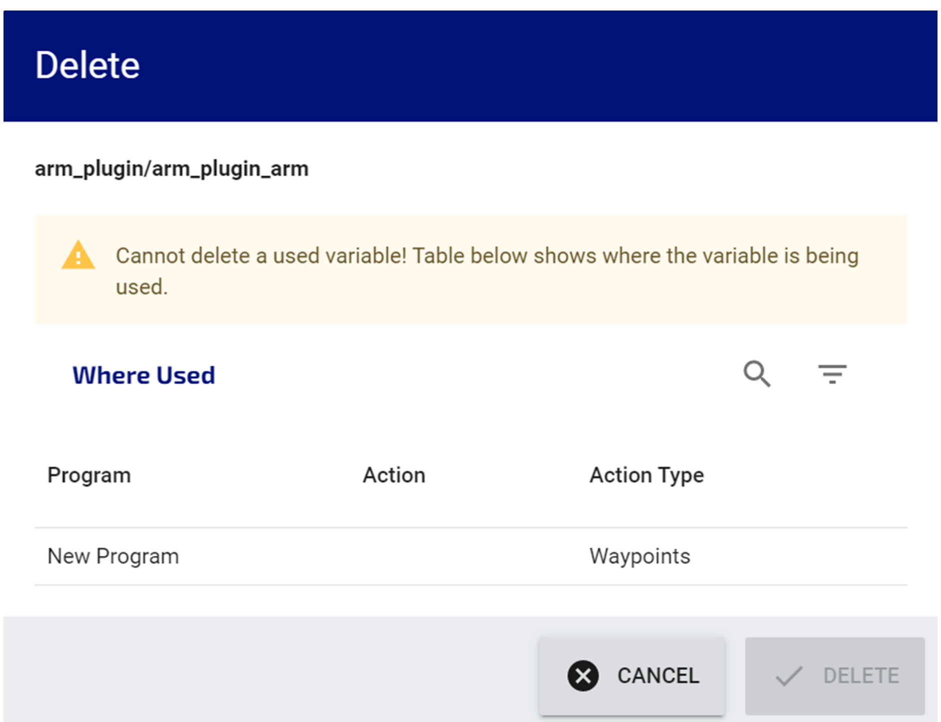


Figure 170: Delete program variable warning dialog

From the **Delete** warning dialog, you can see which program is using the variable and where it is being used.

Important: Do not delete any variable defined as a global variable from the **Persistent Variables** pane of the **Variable Manager** and **Select Variable** pages unless you know for a fact that the variable in question is not used in any program used by the robot.

Procedure

1. Take note of all the places the variable is used in the program.
2. Tap **CLOSE** on the **Delete** warning dialog.
3. Tap **CLOSE** on the **Variable Manager** page.
4. Tap on the tile corresponding to the location of the variable.
5. Tap the **X** beside the variable in the **Tile configuration**.
   The field clears.
6. Launch the **Variable Manager** page .
7. Select the variable and tap .
   The **Delete** confirmation dialog launches.
8. Tap **Delete**.

Results

The variable is no longer available.

Core tiles

The **Programs** page has a set two sets of core tiles that are used to drag and drop into the **Sequence editor** to help create programs.

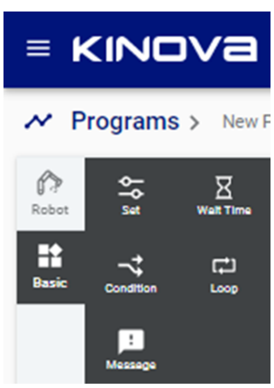
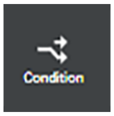
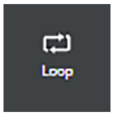
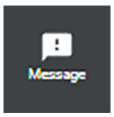
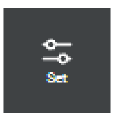


Figure 171: Basic action tiles

Table 77: Basic action tiles

Tile	Tile name	Description
	**Condition**	Create a condition in the program.
	**Loop**	Create a loop in the program.
	**Message**	Create a message in the program. **Note:** Runtime values of variables can be printed in messages by using the syntax `${variable_name}` .
	**Set**	Set the value of a variable. If there is a default value, **Set** overrides the default value.
	**Wait Time**	Introduce a period of time to wait before proceeding to the next step in the program.

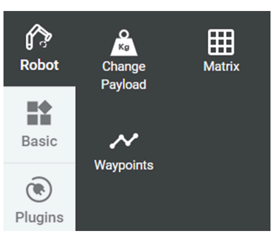
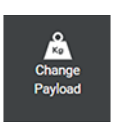
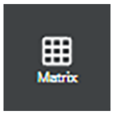


Figure 172: Robot tiles

Table 78: Tiles

Tile	Tile name	Description
	**Change Payload**	Define a new payload during the course of a program.

Tile	Tile name	Description
	**Matrix**	Define a matrix of positions.
	**Waypoints**	Create a list of waypoints in the program. **Remember:** Each list must be all *angular* or all *Cartesian*.

Important: Aside from the core tiles, each installed *plugin* may have its own set of tiles. These additional tiles are displayed in the same area as the core actions to ease programming.

Each core tile has its own set of variables that must be defined in the **Tile configuration**.

Common core parameters in the Tile Configuration pane

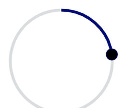
The **Tile configuration** changes dynamically, depending on which tile is dragged into the **Sequence editor**.

The content of the **Tile configuration** changes dynamically with the selected tile. However, there are a few parameters that remain the same for all tiles.

Figure 173: Common parameters in the Tile Configuration pane

Table 79: Entities common to all core tiles

Item	Description
Action Name	Enter a meaningful name for the currently selected tile.
Output Results To Variable	Output results of a variable that is selected for the currently selected tile. **Note:** Not all tiles have any information for this field.

Item	Description
(X)	Tap to launch the **Select Variable** page. Select and modify, or create a variable. **Note:** Not all tiles can select a variable. It is used for the Industrial I/O tiles.
	Play the selected tile when you are in Hold-to-Run mode. It launches **Variables use by action**, where you can modify values, if necessary, before playing tapping **PLAY**.
	Tap to launch a circle help visualize the angle. Use the circle to change the angle of the joint. 

Basic: Set tile parameters

The **Set** tile is a core basic tile. When you drag it into an available tile in the **Sequence editor**, the **Tile configuration** changes dynamically to expose the entities of **Set** that can be configured.

Use **Set** when you need to set the value of one or more variables.

Tip: Whenever you see **(X)** beside an exposed entity, tap it to launch the **Select Variable** page and store the value you entered in a variable.

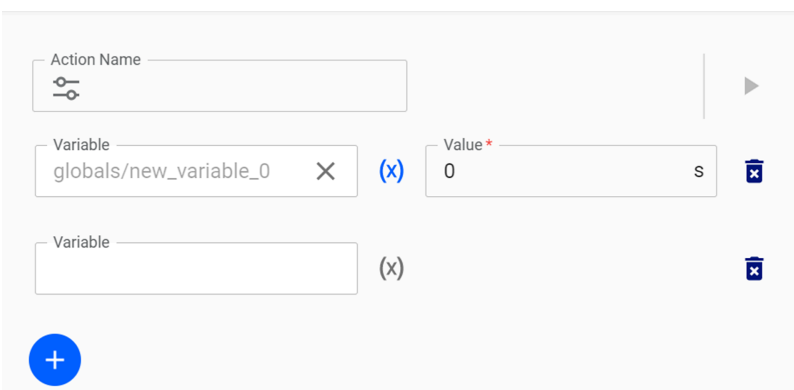


Figure 174: Tile configuration when Set is the current tile

Table 80: Core basic Set entities

Item	Description
Action Name	Select an action from the list that launches when you tap it or enter a meaningful action name for the selected tile.
Variable	Enter a meaningful name for the variable. **Note:** Tapping the text field launches the **Select Variable** window where you can select an existing variable, in which case you are assigning the variable a value, or you can create a variable.
Value	Enter the value of the variable.
	Tap to delete the variable from the list.
	Tap to add another variable.

Basic: Wait tile parameters

The **Wait Time** tile is a core basic tile. When you drag it into an available tile in the **Sequence editor**, the **Tile configuration** changes dynamically to expose the entities of **Wait Time** that can be configured.

Use **Wait Time** when you need to introduce a specified amount of time for the robot to stop moving before proceeding in the program.

Remember: Whenever you see **(X)** beside an exposed entity, tap it to launch the **Select Variable** page and store the value you entered in a variable.

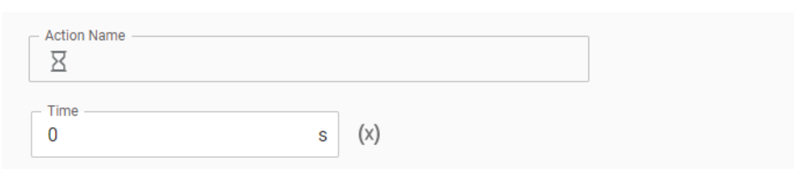


Figure 175: The Tile configuration when Wait Time is the current tile

Table 81: Core basic Wait Time entities

Item	Description
Time	Enter either an integer or a floating point value. Time is measured in seconds.

Basic: Condition tile parameters

The **Condition** tile is a core basic tile. When you drag it into an available tile in the **Sequence editor**, the **Tile configuration** changes dynamically to expose the entities of **Condition** that can be configured.

Use **Condition** when you need to introduce one or more conditions in the program before proceeding in the program.

Tip: Whenever you see **(X)** beside an exposed entity, tap it to launch the **Select Variable** page and store the value you entered in a variable.

When the **Condition** tile is dragged into the **Sequence editor**, it expands.

Figure 176: The Condition tile expands in the Sequence editor

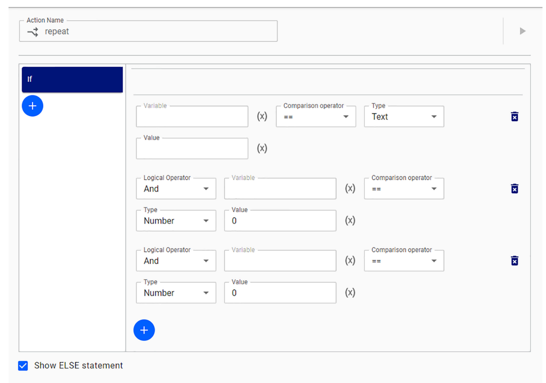


Figure 177: The Tile configuration when Condition is the current tile

A condition starts with **If** and requires variables and values to set the condition.

Table 82: Core basic Condition entities

Item	Description
Variable	Tap to launch the **Select Variable** window where you can select or create a variable to use in the condition.
Comparison operator	Select from the drop-down. • **==** • **>** • **>=** • **<** • **<=**

Item	Description
Type	Select from the drop-down. • **Boolean**: **true**, **false** • **Number**: integer or floating point • **Text**: string of letters
Value	Enter or select the value associated with the selected type.
	Tap to introduce an alternative condition as an **Else If** statement. The **Else If** also requires a **Variable**, **Comparison operator**, **Type**, **Value** for the selected Type. When **Else If** is selected, another tile is displayed in the **Sequence editor**. 
Show ELSE statement	Select to show the program going through the else condition. The tile in the **Sequence editor** expands. 

Basic: Loop tile parameters

The **Loop** tile is a core basic tile. When you drag it into an available tile in the **Sequence editor**, the **Tile configuration** changes dynamically to expose the entities of **Loop** that can be configured.

Use **Loop** when you need actions to repeat a certain number of times.

Tip: Whenever you see **(X)** beside an exposed entity, tap it to launch the **Select Variable** page and store the value you entered in a variable.

When the **Loop** tile is dragged into the **Sequence editor**, it expands.

Figure 178: The Loop tile expanded in the Sequence editor

The parameters of the **Loop** tile are divided into the global aspects of an action an the specific conditions of the variable, known as the **Looping Conditions**.

Table 83: Core basic Loop entities

Item	Description
Action Name	Select an action from the list that launches when you tap it or enter a meaningful action name for the selected tile.
Do at least once	Select if you need to perform the actions in the loop at least once.
Iterator	Tap to select or create a variable that increments automatically by 1 on each repetition of the actions in the loop. • The variable used as an iterator must be of type `integer`. • The variable is used for iterating, or counting, over a loop. • The value of the iterator can be modified during run-time; its modified value is taken into consideration. • The default value is used when the iterator has no value defined. **Example:** When the iterator variable has a default of 0 and has not be used yet, the iterator uses `0` as the first number. When the iterator has a default value of 5 and has not be used yet, the iterator starts counting from 5.

Item	Description
Maximum Iterations	Enter the maximum number of times the actions within the loop are needed. **Warning:** When **Maximum Iterations** is set to 0, the actions in the loop repeat as long as the **Looping Conditions** are true. **Note:** The default maximum number of iterations is 1.
Looping Conditions	
Variable	Tap to launch the **Select Variable** window where you can select or create a variable to use for the loop. **Tip:** The variable used in the **Looping Conditions** can be any variable, including the **Iterator**. The value and the variable that are being compared must be of the same type.
Comparison operator.	Select from the drop-down. • **==**: equals • **>**: greater than • **>=**: greater than or equal • **<**: less than • **<=**: less than or equal • **!=**: not equal
Value	Enter the value associated with the selected type.
	Alternatively, tap **(X)** to select a variable as the value.
	The value entered, or the variable selected, must be of the same type as the **Variable** in the **Looping Conditions** pane.

Item	Description
Logical Operator	Select from the drop-down. • **And** • **Or** You need the logical operator to combine two or more conditions for the loop to continue.
+	Tap to add more conditions.

Examples of maximum iterations

Example 1: The default value of the iterator variable is 0. The maximum number of iterations is 1. The loop is performed one time; the value of the iterator is 0.

Example 2: The default value of the iterator variable is 1. The maximum number of iterations is 1. The loop is performed one time; the value of the iterator is 1.

Example 3: The default value of the iterator variable is 5. The maximum number of iterations is 1. The loop is performed one time; the value of the iterator is 5.

Example 4: The default value of the iterator variable is 0. The maximum number of iterations is 5. The loop is performed five times; the value of the iterator is 4.

Example 5: The default value of the iterator variable is 1. The maximum number of iterations is 5. The loop is performed five times; the value of the iterator is 5.

Basic: Message tile parameters

The **Message** tile is a core basic tile. When you drag it into an available tile in the **Sequence editor**, the **Tile configuration** changes dynamically to expose the entities of **Message** that can be configured.

Use **Message** when you want to print messages when a program runs.

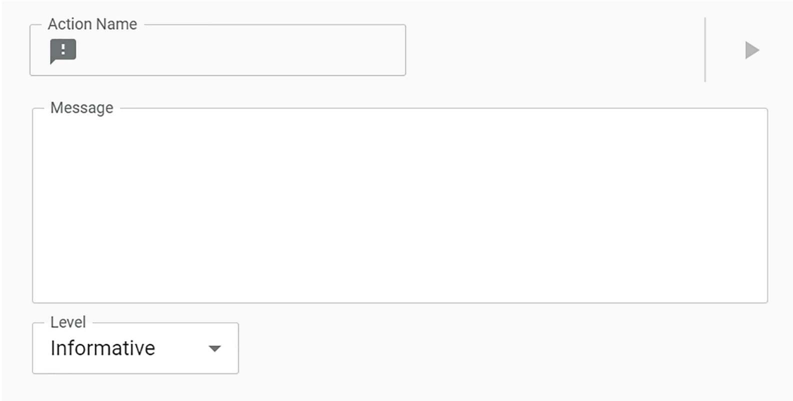


Figure 179: The Tile configuration when Message is the current tile

Although messages are primarily string based, you can print out the values of variables.

Table 84: Core basic Message entities

Item	Description
Message	Enter a message. Messages can contain variables. **Example** A variable can be defined as `meaningful_variable_name = { "employee": {"name":"John", "age":30, "city":"Montreal"}` Enter in the message field: `Hello, ${meaningful_variable_name.employee.name}.` When the program reaches the message, it prints out `Hello, John.`
Level	Select the level of severity for the message. • **Error** • **Informative** • **Warning**

Robot: Matrix tile parameters

The **Matrix** tile is a core robot tile. When you drag it into an available tile in the **Sequence editor**, the **Tile configuration** changes dynamically to expose the entities of the matrix that can be configured.

The **Matrix** tile is used to define a large number of Cartesian poses that are spaced evenly on a 3D grid. Only a few parameters must be entered. Store the list of poses in a variable so that it can be used in a program.

Important: The tile does not make the robot move; it only performs the computations needed to make the robot move.

The parameters of the matrix are grouped into three parts. Each part is a sub-pane in the **Tile configuration**. Each sub-pane is accessible by tapping on its name.

- **Parameters**
- **Coordinates**

Tip: Whenever you see **(X)** beside an exposed entity, tap it to launch the **Select Variable** page and store the value you entered in a variable.

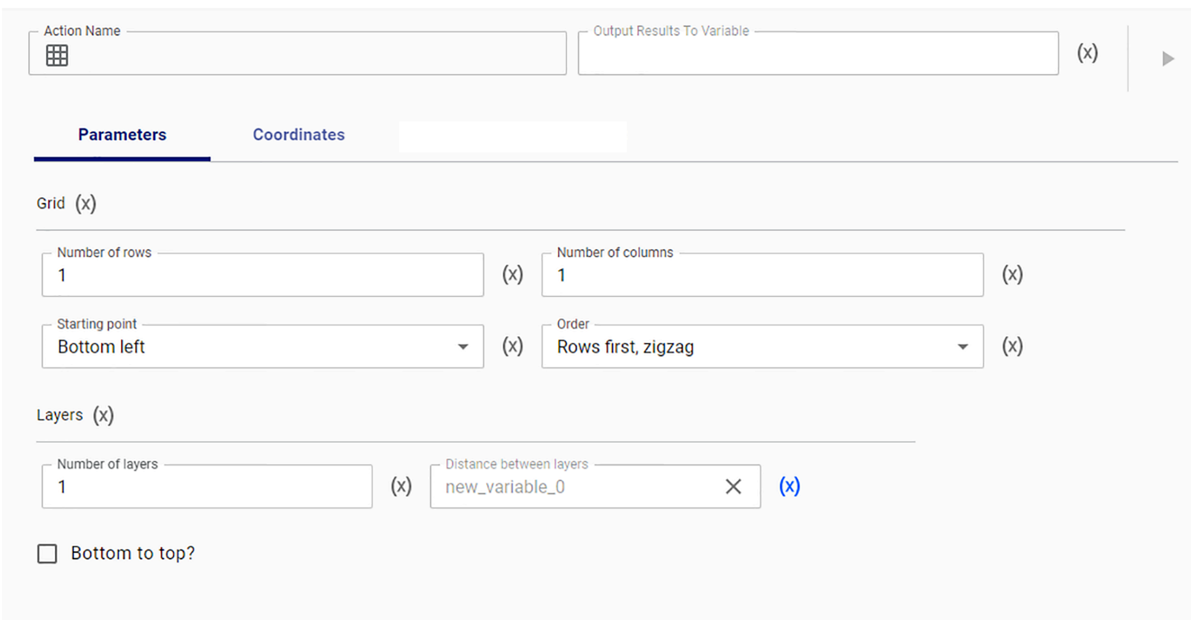


Figure 180: Entities of the Matrix tile on the Parameter sub-pane in the Tile configuration

Table 85: Core robot Matrix entities on the Parameters sub-pane

Item	Description
Grid	
Number of rows	Enter the number of rows in the matrix.
Number of columns	Enter the number of columns in the matrix.

Item	Description
Starting point	Select from the drop-down. • **Bottom left** • **Bottom right** • **Top left** • **Top right**
Order	Select from the drop-down. • **Rows first, zigzag** • **Rows first, back and forth** • **Columns first, zigzag** • **Columns first, back and forth** • **Randomized**
Layers	
Number of layers	Enter the number of layers in the matrix.
Distance between layers	Tap in the text field or tap **(X)** to select, or create, a variable that defines the distance between layers.
Bottom to top?	Select to run the matrix from the bottom layer to the top layer. Deselect to play from the top layer to the bottom.

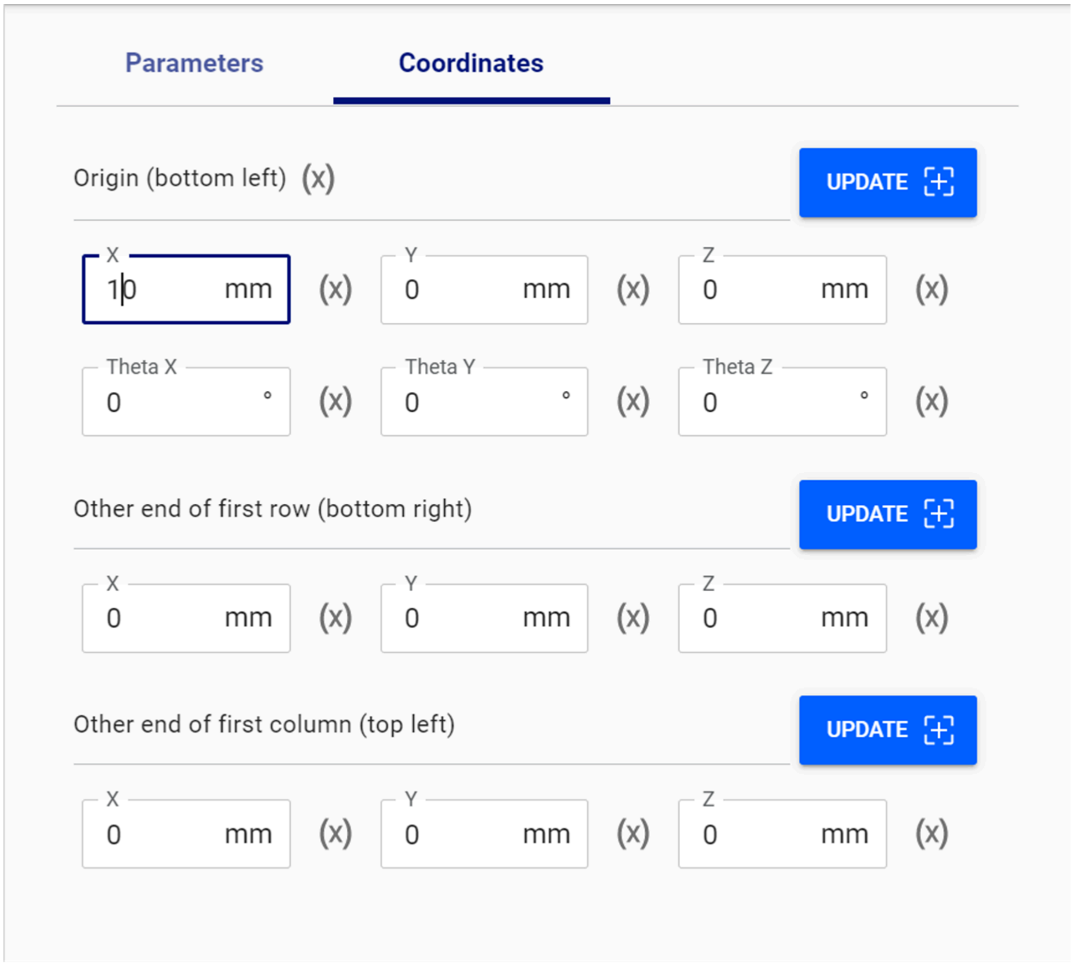


Figure 181: Entities of the Matrix tile on the Coordinates sub-pane in the Tile configuration

Table 86: Core robot Matrix entities on the Coordinates sub-pane

Item	Description
Origin	The origin is the selected **Starting point** on the **Parameters** sub-pane. Tap the **X**, **Y**, **Z**, **Theta X**, **Theta Y**, **Theta Z** text boxes, or the **(X)** beside them, to select or create a variable that has the associated values. **Remember:** Tap **UPDATE** when you change the values in the text boxes. Tap **CLEAR** to undo changes you made. **Note:** You can also use the snapshot button on the wrist of the robot to set the origin.

Item	Description
Other end of first row	Enter values for **X**, **Y**, and **Z**. **Remember:** Tap **UPDATE** when you change the values in the text boxes. Tap **CLEAR** to undo changes you made. **Note:** You can also use the snapshot button on the wrist of the robot to set the end of the first row.
Other end of the first column	Enter values for **X**, **Y**, and **Z**. **Remember:** Tap **UPDATE** when you change the values in the text boxes. Tap **CLEAR** to undo changes you made. **Note:** You can also use the snapshot button on the wrist of the robot to set the end of the first column.

Robot: Change Payload tile parameters

The **Change Payload** tile is a core robot tile. When you drag it into an available tile in the **Sequence editor**, the **Tile configuration** changes dynamically to expose the entities of the payload that can be configured.

The **Change Payload** tile is used to configure payloads while a program is running. No other action can run while the **Change Payload** is running. If the program is stopped at the time the **Change Payload** tile is active, the **Change Payload** action remains active. One or more **Change Payload** actions can be dragged into the **Sequence editor** to create a program.

The parameters of a payload are grouped into three parts.

- **Payload**
- **Center of mass**
- **Inertia Matrix**

The parameters are modifiable when the robot is in one of three states.

- The arm is switched off.
- The arm is switched on and is in Monitored Stop mode.
- The arm is switched on and is in Hold-to-Run mode.

Tip: Whenever you see **(X)** beside an exposed entity, tap it to launch the **Select Variable** page and store the value you entered in a variable.

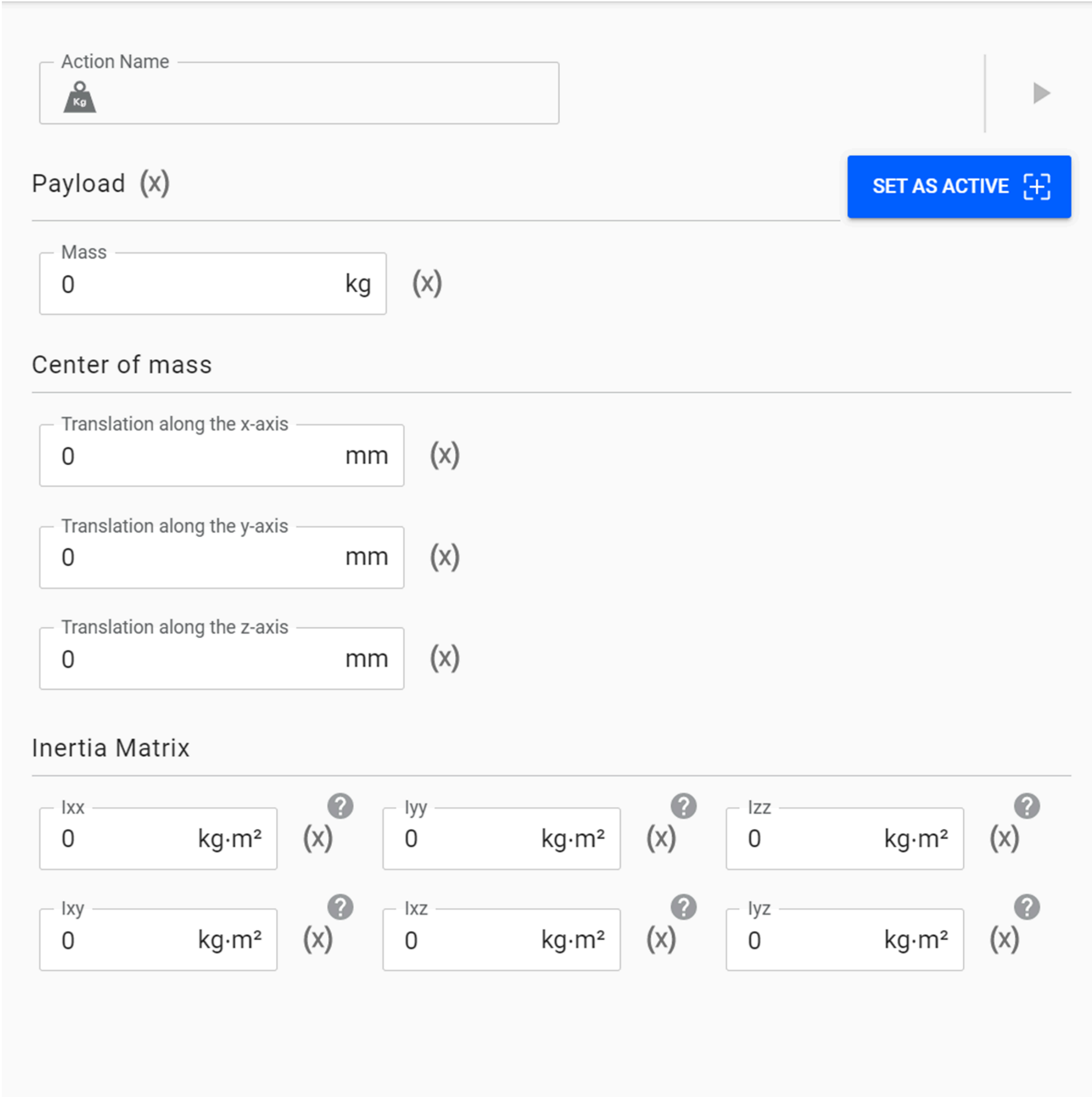


Figure 182: Entities of the Matrix tile on the Parameter sub-pane in the Tile configuration

Table 87: Core robot Change Payload entities

Item	Description
Payload	

Item	Description
SET AS ACTIVE	Tap to set the payload configuration defined in the parameter configuration pane as active. The currently active payload is displayed in the **Payload** pane on **Systems** > **Robot** > **Arm**.
Mass	Enter the mass of the new payload. Valid range for mass: 0.0000 to 9.0000 kg
Center of mass	
Translation along the x-axis	Enter the translation along the x-axis with respect to the flange Valid range: -1000 to 1000 mm
Translation along the y-axis	Enter the translation along the y-axis with respect to the flange Valid range: -1000 to 1000 mm
Translation along the z-axis	Enter the translation along the z-axis with respect to the flange Valid range: -1000 to 1000 mm
Inertia Matrix	
Ixx	Enter the moment of inertia with respect to the x-axis of the flange Valid range: 0 to 1 kg m^2
Iyy	Enter the moment of inertia with respect to the y-axis of the flange Valid range: 0 to 1 kg m^2
Izz	Enter the moment of inertia with respect to the z-axis of the flange Valid range: 0 to 1 kg m^2
Ixy	Enter the product of inertia Ixy Valid range: -1 to 1 kg m^2

Item	Description
Ixz	Enter the product of inertia Ixz Valid range: -1 to 1 kg m^2
Iyz	Enter the product of inertia Iyz Valid range: -1 to 1 kg m^2

Robot: Waypoints tile parameters

The **Waypoints** tile is a core robot tile. When you drag it into an available tile in the **Sequence editor**, the **Tile configuration** changes dynamically to expose the entities of the waypoints that can be configured.

The parameters of the list of waypoints are grouped into three parts. Each part is a sub-pane in the **Tile configuration**. Each sub-pane is accessible by tapping on its name.

- **Waypoints**
- **Constraints**
- **Custom Frame**

Tip: Whenever you see **(X)** beside an exposed entity, tap it to launch the **Select Variable** page and store the value you entered in a variable.

Waypoint parameters

Each **Waypoints** tile in a program must define one or more waypoints for the robot to move.

The **Waypoints** tile holds a configurable list of waypoints the robot uses to move to complete a task.

Important: The arm must be switched on before you can define waypoints for the robot.

Tip: Configure the waypoints in the waypoint list when you are in Monitored Stop mode or Hold-to-Run mode. Select the mode from the Robot control panel.

Some of the parameters are the same regardless of the type of waypoint; others are specific to the type of waypoint.

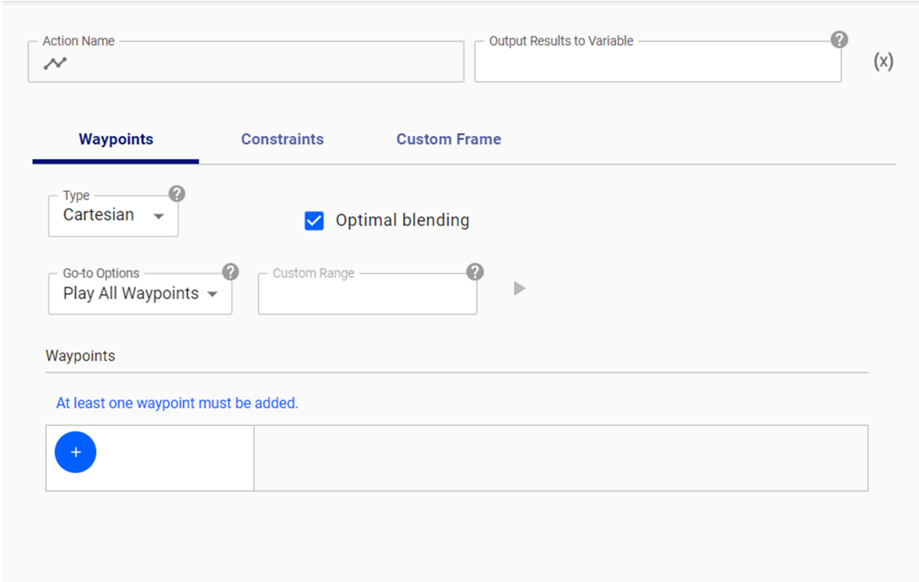


Figure 183: Parameters common to all types of waypoints

Table 88: Parameters common to all types of waypoints

Item	Description
Type	Select **Cartesian**, which is moving the arm, or **Angular**, which is moving the joints.
Optimal bleeding	When selected, the robot takes an time-optimal path from one end of the trajectory to another without actually reaching the intermediary points.
	Tap **+** in the **Waypoints** sub-pane to add a waypoint to the list of waypoints for the currently selected waypoint tile. The waypoint is added to the list of waypoints. Parameters for the new waypoint launches in the **Waypoints** sub-pane. The parameters that are displayed depend on what is selected from **Type**.

Item	Description
Index	Each waypoint in the list of waypoints in the waypoints tile has an associated index number. The first waypoint is 1, the second is 2, and so on. **Note:** The field is read-only. It can be used as a reference when you want to use **Custom Range**.
Name	Enter a meaningful name for the currently selected waypoint in the list of waypoints.

Cartesian waypoint parameters

When you select **Cartesian** as the waypoint type, the parameters related to Cartesian poses are displayed.

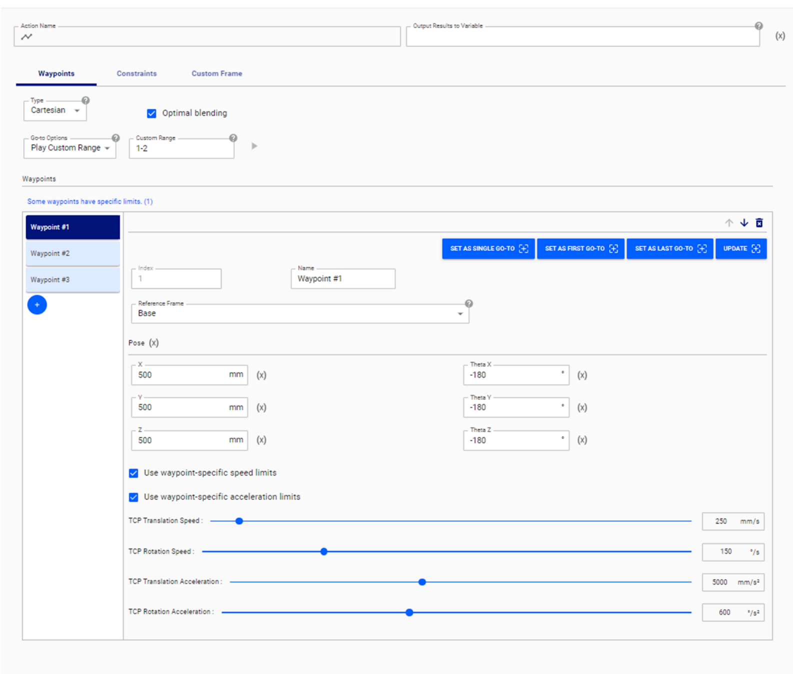


Figure 184: Parameters of Cartesian waypoints

Table 89: Parameters of Cartesian waypoints

Item	Description
Reference Frame	Select a frame of reference. The waypoint pose converts automatically with respect to the frame of reference. • **Base** • **Tool** • **Custom**
Pose	Pose is expressed in Cartesian coordinates.
X, **Y**, **Z**	The current Cartesian position of the waypoint selected in the list is displayed. Change the values if you know your values are more accurate than what is currently listed. Measurements are in millimeters.
Theta X, **Theta Y**, **Theta Z**	The current angular position of the waypoint selected in the list. Change the values if you know your values are more accurate than what is currently listed. Measurements are in degrees.
Use waypoint-specific speed limits	Select when you want to set a specific speed limit on the robot at the selected waypoint instead of using the parametric values shared among all waypoints in the list. The **Speed Limits** sub-pane launches.
Speed Limits sub-pane	
TCP Translation Speed	Drag the slider to, or enter, the desired translation speed for the *TCP* at the selected waypoint. Minimum: 10 mm/s Maximum: 2000 mm/s
TCP Rotation Speed	Drag the slider to, or enter, the desired rotational speed for the *TCP* at the selected waypoint. Minimum: 1 °/s Maximum: 600 °/s

Item	Description
Use waypoint-specific acceleration limits	Select when you want to use specific acceleration limits on the robot at the selected waypoint instead of using the parametric values shared among all waypoints in the list. The **Acceleration Limits** sub-pane launches.
Acceleration Limits sub-pane	
TCP Translation Acceleration	Drag the slider to, or enter, the desired translation acceleration for the *TCP* at the selected waypoint. Minimum: 10 mm/s^2 Maximum: 12,000 mm/s^2
TCP Rotation Acceleration	Drag the slider to, or enter, the desired orientation acceleration for the *TCP* at the selected waypoint. Minimum: 1 °/s^2 Maximum: 1500 °/s^2.

Angular waypoint parameters

When you select **Angular** as the waypoint type, the parameters related to the orientation of the robot at that location are displayed

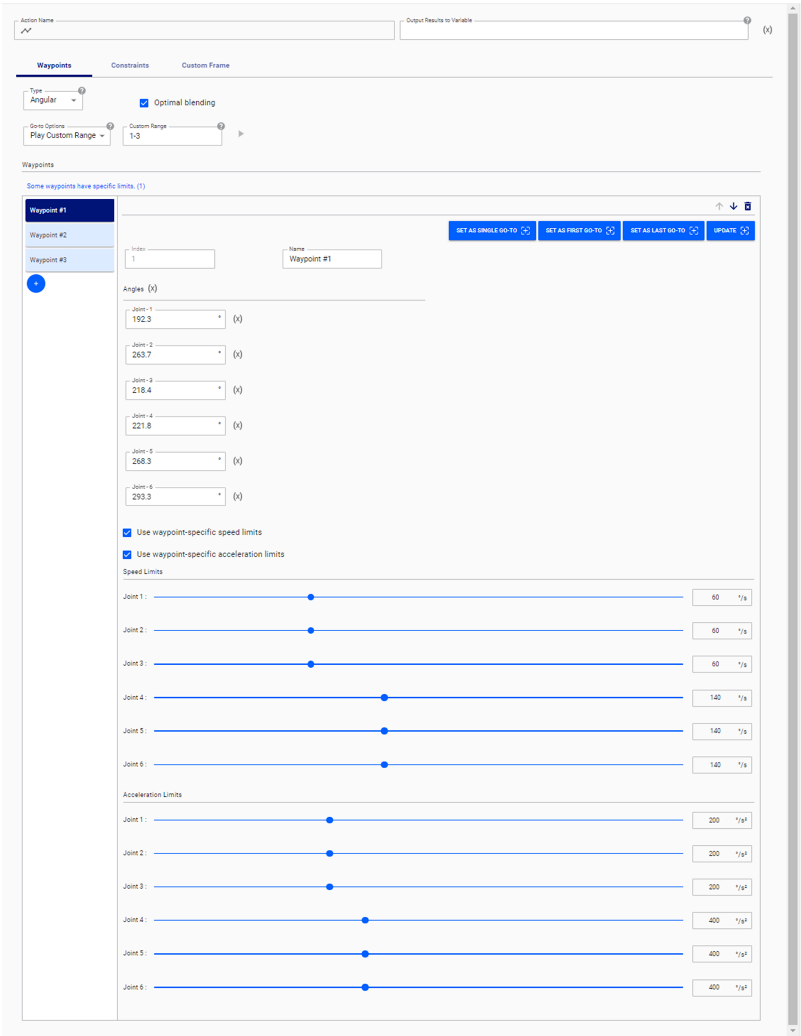


Figure 185: Parameters of angular waypoints

Table 90: Parameters of Angular waypoints

Item	Description
Angles	The pose is expressed in degrees for each of the joints.
Joint 1, **Joint 2**,**Joint 3**, **Joint 4**, **Joint 5**, **Joint 6**	The current angular position of each joint at the waypoint selected in the list is displayed. Change the values if you know your values are more accurate than what is currently listed. Minimum of each joint: -357° Maximum of each joint: +357°

Item	Description
Use waypoint-specific speed limits	Select when you want to set a specific speed limit on the robot at the selected waypoint instead of using the parametric values shared among all waypoints in the list. The **Speed Limits** sub-pane launches.
Speed Limits	
Joint 1, **Joint 2**,**Joint 3**	Drag the slider to, or enter, the desired speed at the joint at the selected waypoint. Minimum: 1 °/s Maximum: 200 °/s. **Important:** The desired speed cannot exceed the Safety Joint Speed Limit defined on **Safety** > **Joint Limits**.
Joint 4, **Joint 5**, **Joint 6**	Drag the slider to, or enter, the desired speed at the joint at the selected waypoint. Minimum: 1 °/s Maximum: 320 °/s **Important:** The desired speed cannot exceed the Safety Joint Speed Limit defined on **Safety** > **Joint Limits**.
Use waypoint-specific acceleration limits	Select when you want to set a specific speed limit on the robot at the selected waypoint instead of using the parametric values shared among all waypoints in the list. The **Acceleration Limits** sub-pane launches.
Acceleration Limits	
Acceleration Limits for **Joint 1**, **Joint 2**,**Joint 3**, **Joint 4**, **Joint 5**, **Joint 6**	Drag the slider to, or enter, the desired acceleration at each joint for the selected waypoint. Minimum: 1 $°/s^2$ Maximum: 600$°/s^2$

Constraint parameters

Each **Waypoints** tile can have its own set of global speed and acceleration limits.

The global speed and acceleration limits do not apply to the waypoints that have waypoint-specific speed and acceleration limits defined on the **Constraints** pane.

The constraints available depend on the type of waypoint: Cartesian or angular.

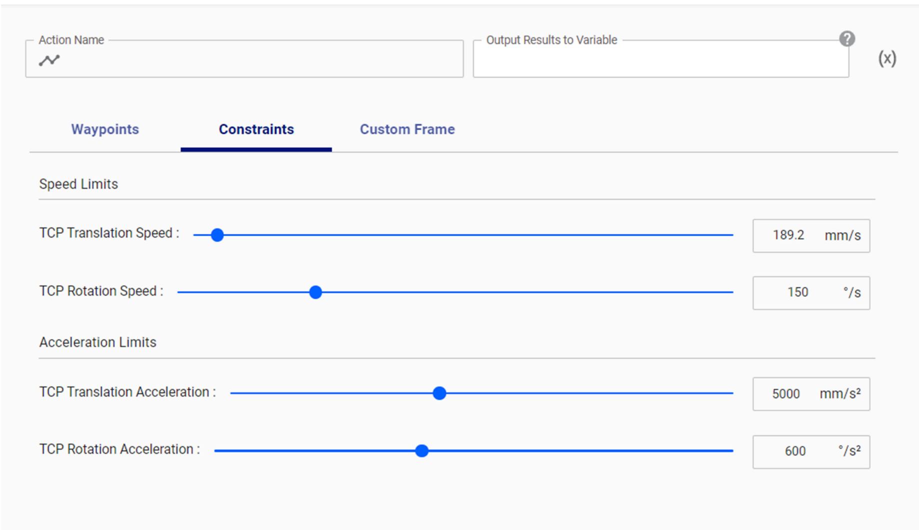


Figure 186: Constraints for Cartesian waypoints

Table 91: Constraints for Cartesian waypoints

Item	Description
Speed Limits sub-pane	
TCP Translation Speed	Drag the slider to, or enter, the desired translation speed for the *TCP* at the selected waypoint. Minimum: 10 mm/s Maximum: 2000 mm/s
TCP Rotation Speed	Drag the slider to, or enter, the desired rotational speed for the *TCP* at the selected waypoint. Minimum: 1 °/s Maximum: 600 °/s
Acceleration Limits sub-pane	

Item	Description
TCP Translation Acceleration	Drag the slider to, or enter, the desired translation acceleration for the *TCP* at the selected waypoint. Minimum: 10 mm/s^2 Maximum: 12,000 mm/s^2
TCP Rotation Acceleration	Drag the slider to, or enter, the desired orientation acceleration for the *TCP* at the selected waypoint. Minimum: 1 $°/s^2$ Maximum: 1500 $°/s^2$.

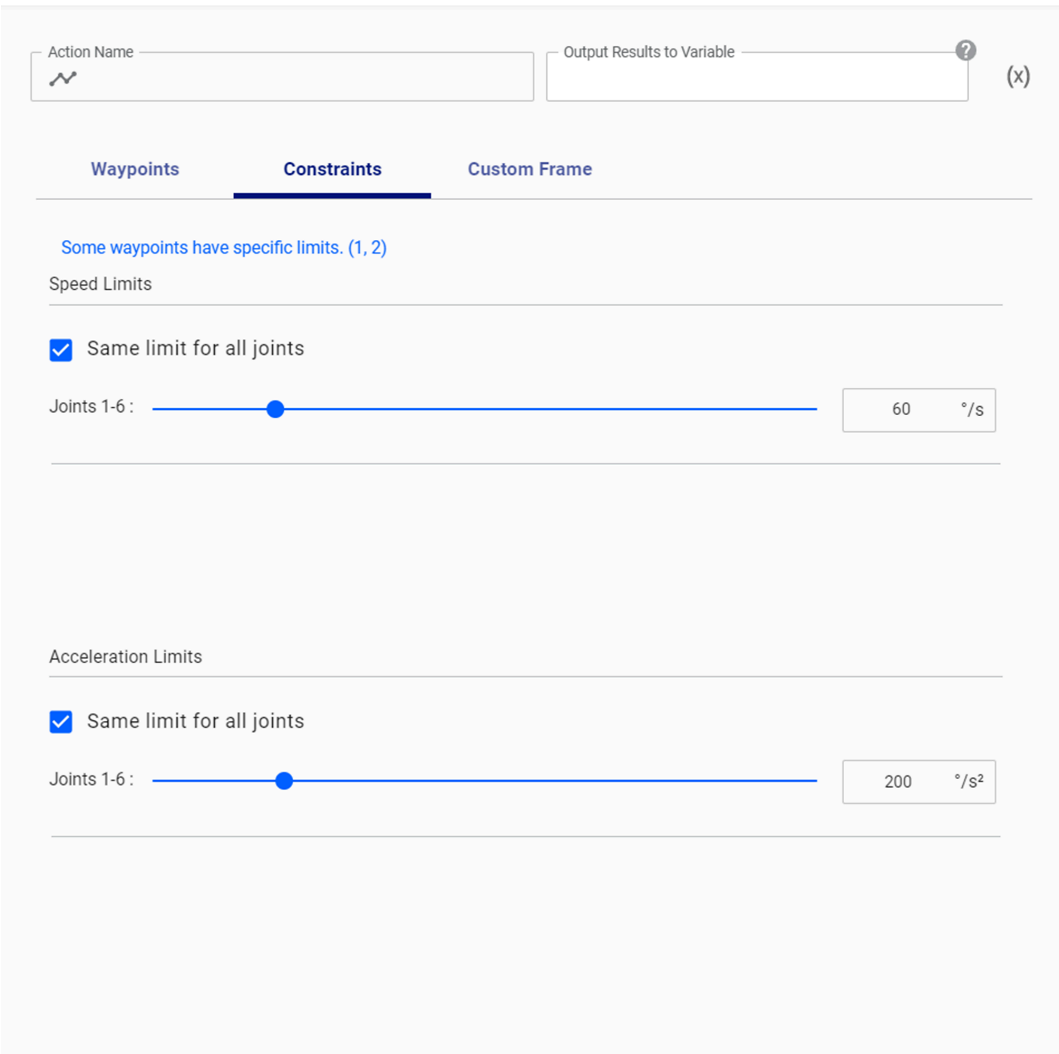
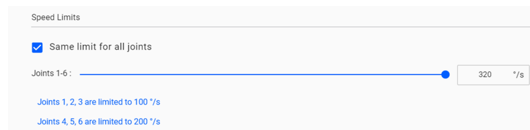
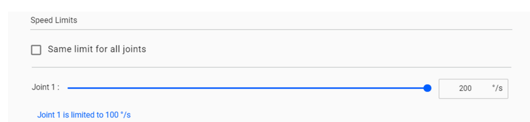


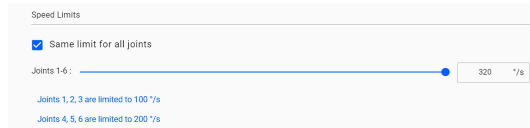
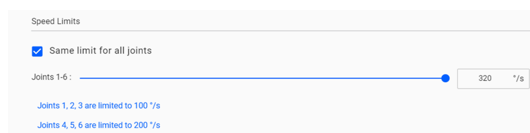
Figure 187: Constraints applied to all joints of an angular waypoint

Figure 188: Constraints applied to specific joints of an angular waypoint

Table 92: Constraints for angular waypoints

Item	Description
Speed Limits sub-pane	

Item	Description
Same limit for all joints	Select when you want to constrain all six joints by the same speed.
Joints 1-6	Drag the slider to, or enter, the desired speed for all joints at the selected waypoint. **Tip:** A message launches when the speed you select exceeds the safety speed defined on **Safety** > **Joint Limits** > **Speed Limits** for one or more joints.  **Remember:** Safety speed limits override all other speed limits. Minimum: 1 °/s Maximum: 320 °/s
Same limit for all joints	Deselect when you want to constrain one or more specific joints by a specific speed. **Tip:** A message launches under each joint where the speed you select exceeds the safety speed defined on **Safety** > **Joint Limits** > **Speed Limits**. 

Item	Description
Joint 1Joint 2Joint 3	Drag the slider to, or enter, the desired speed for the specific joint at the selected waypoint. **Tip:** A message launches when the speed you select exceeds the safety speed defined on **Safety** > **Joint Limits** > **Speed Limits** for the specified joint.  **Remember:** Safety speed limits override all other speed limits. Minimum: 1 °/s Maximum: 200 °/s
Joint 4Joint 5Joint 6	Drag the slider to, or enter, the desired speed for the specific joint at the selected waypoint. **Tip:** A message launches when the speed you select exceeds the safety speed defined on **Safety** > **Joint Limits** > **Speed Limits** for the specified joint.  **Remember:** Safety speed limits override all other speed limits. Minimum: 1 °/s Maximum: 320 °/s
Acceleration Limits sub-pane	
Same limit for all joints	Select when you want to constrain all six joints by the same acceleration.

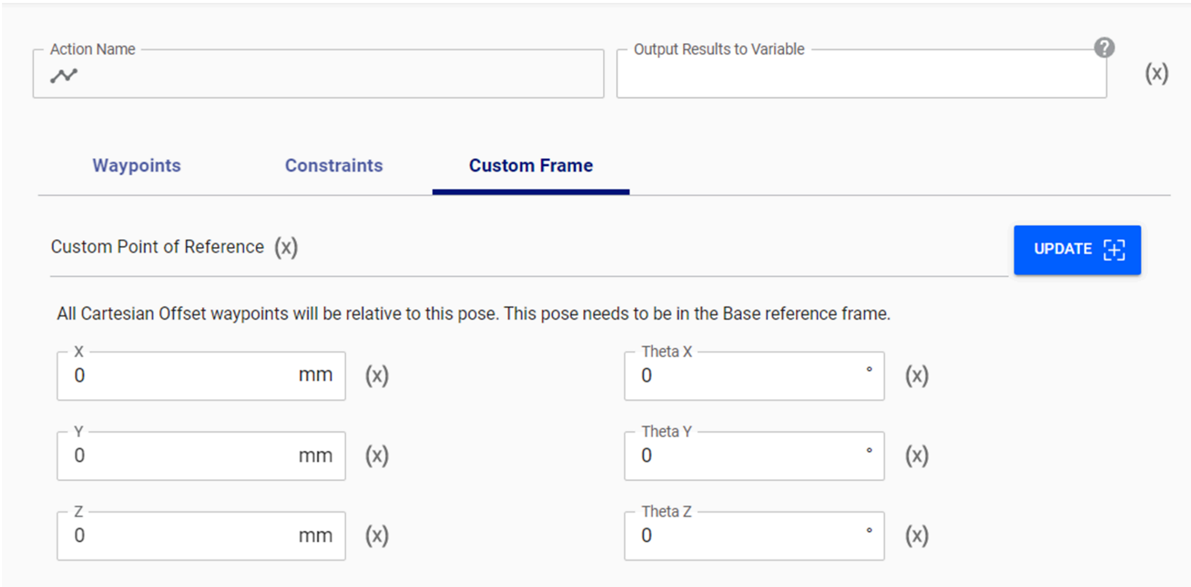
Item	Description
Joints 1-6	Drag the slider to, or enter, the desired speed for all joints at the selected waypoint. Minimum: 1 °/s Maximum: 1000 °/s
Same limit for all joints	Deselect when you want to constrain one or more specific joints by a specific acceleration.
Joint 1Joint 2Joint 3	Drag the slider to, or enter, the desired speed for the specific joint at the selected waypoint. Minimum: 1 °/s Maximum: 600 °/s
Joint 4Joint 5Joint 6	Drag the slider to, or enter, the desired speed for the specific joint at the selected waypoint. Minimum: 1 °/s Maximum: 1000 °/s

Custom Frame parameters

In certain scenarios, it is easier to program waypoints in relation to a specific pose.

The specific pose can be used as a frame of reference for any waypoint in the list. The frame of reference for the custom frame definition must be **Base**.

Tip: The only debugging tool available for a custom frame is **UPDATE**.



Debug tools for waypoints

To simplify creating the perfect waypoints in a program, a few debug tools are available.

Any settings made on individual waypoints with the debug tools are not valid during an actual run of the full program.

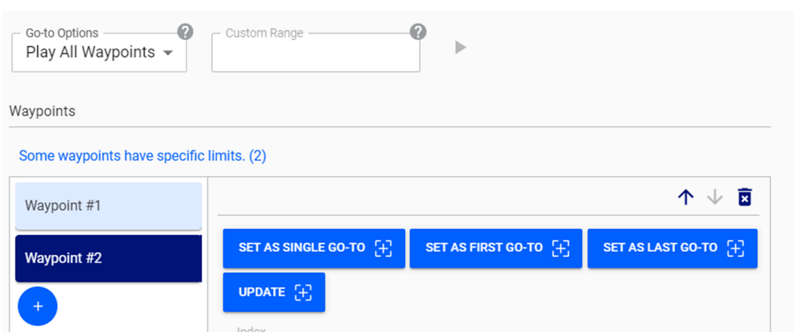


Figure 189: Tools for debugging waypoints

Table 93: Tools for debugging waypoints

Item	Description
Go-To Options	**Play All Waypoints** Play all waypoints listed in the **Waypoints** tile. **Play Custom Range** Play only the waypoints within the range defined in **Custom Range**

Item	Description
Custom Range	Enter the start and end waypoints. For example, if 6 waypoints are defined in a tile and you want to investigate only from waypoint #2 and want to end at waypoint #5, you would enter the range as `2-5`. **Note:** The **Custom Range** field can be used only when **Play Custom Range** is selected from **Go-To Options**.
SET AS A SINGLE GO-TO	Tap to set the correct indices in the **Go-To Options**. Click **Play** ( ) near the **Go-To Options** to make the arm move. **Go-to Options** changes automatically to **Play Custom Range**. **Custom Range** displays the index of the waypoint in the list to which the arm automatically moves. **Warning:** Tapping **Play** ( ) in the Program play area runs the entire program, disregarding the **Go-To Options**.

Item	Description
SET AS FIRST GO-TO	Tap to set the correct indices in the **Go-To Options**. Click **Play** () near the **Go-To Options** to make the arm move. **Go-to Options** changes automatically to **Play Custom Range**. **Custom Range** displays the index of the waypoint selected as the first waypoint to which the arm automatically moves. When the last go-to waypoint is not set explicitly, the range ends with the index of the last waypoint in the list. When the last go-to waypoint is set explicitly, the range ends with the index of the waypoint selected as the last go-to. The index of the waypoint defined as the first go-to must be smaller than the index of any waypoint that is defined as the last go-to. **Warning:** Tapping **Play** () in the Program play area runs the entire program, disregarding the **Go-To Options**.

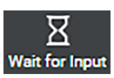
Item	Description
SET AS LAST GO-TO	Tap to set the correct indices in the **Go-To Options**. Click **Play (▶)** near the **Go-To Options** to make the arm move. **Go-to Options** changes automatically to **Play Custom Range**. When the first go-to waypoint is not set explicitly, **Custom Range** begins with the first index of waypoints in the list. When the first go-to waypoint is set explicitly, the range begins with the index of the waypoint selected as the first go-to. **Custom Range** ends with the index of the waypoint selected as the last waypoint to which the arm automatically moves. The index of the waypoint defined as the last go-to must be smaller than the index of any waypoint that is defined as the first go-to. **Note:** It is possible to have waypoints listed after the last go-to waypoint in the **Waypoints** list, but those waypoints do not end up in the range unless the range is altered to include the new last waypoint. **Warning:** Tapping **Play (▶)** in the Program play area runs the entire program, disregarding the **Go-To Options**.
UPDATE	Tap to update the orientation or pose of the arm automatically with the actual orientation or pose of the arm at the currently selected waypoint.

Item	Description
	Tap **PLAY** from the **Waypoints** sub-pane to check the Cartesian poses and orientation of the arm at each of the locations in the **Waypoints** list. **Tip:** Available only in Hold-to-Run mode when you are testing the waypoints for the program you are creating.

Industrial I/O: tiles

The Industrial I/O plugin has a few tiles that can be used in programs.

Table 94: Tiles that belong to Industrial I/O

Tile	Tile name	Description
	**Read Input**	Reads a signal from an Industrial input channel and stores it in a variable for later use.
	**Set Output**	Sends a signal on a desired industrial output channel.
	**Wait for Input**	Pauses a program until a signal is received on a specific input channel or until a timeout is triggered.

Industrial I/O: Read Input tile parameters

The **Read Input** tile is an Industrial I/O tile. When you drag it into an available tile in the **Sequence editor**, the **Tile configuration** changes dynamically to expose the entities of **Read Input** that can be configured.

Use **Read Input** to configure the type of input that needs to be used at that point in the program.

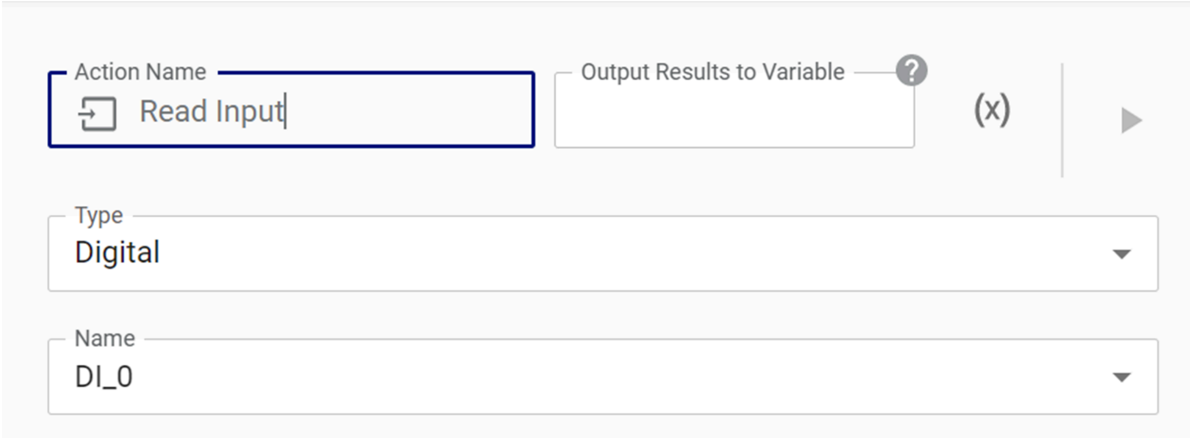


Figure 190: The Tile configuration when Read Input is the current tile

The **Type**

Table 95: Industrial I/O Read Input parameters

Item	Description
Type	Select one of the available read input types. • **Digital** • **Analog** • **Wrist Digital** • **Wrist Analog**
Name	Select the name of the available input of the chosen type as defined by the Industrial I/O plugin configuration. The content depends on the selected read input type.
Type	***Corresponding Name choices***

Item	Description
Digital	Select one of the user-defined names from the list of digital inputs. • **DI_0** • **DI_1** • **DI_2** • **DI_3** • **DI_4** • **DI_5** • **DI_6** • **DI_7**
Analog	Select one of the user-defined names from the list of analog inputs. • **AIO_0** • **AIO_1** • **AIO_2** • **AIO_3**
Wrist Digital	Select one of the user-defined names from the list of wrist digital inputs. • **WRIST_DIGITAL_IO1** • **WRIST_DIGITAL_IO2** • **WRIST_DIGITAL_IO3** • **WRIST_DIGITAL_IO4**
Wrist Analog	Select one of the user-defined names from the list of wrist analog inputs. • **WRIST_AOI_1** • **WRIST_AOI_2**

If the name of the pin that is selected is not configured to be an input, an error message displays directly in the **Tile configuration** pane.

Example:

> This pin is not configured as input.

Industrial I/O: Set Output tile parameters

The **Set Output** tile is an Industrial I/O tile. When you drag it into an available tile in the **Sequence editor**, the **Tile configuration** changes dynamically to expose the entities of **Set Output** that can be configured.

Use **Set Output** to configure the type of input that needs to be used at that point in the program.

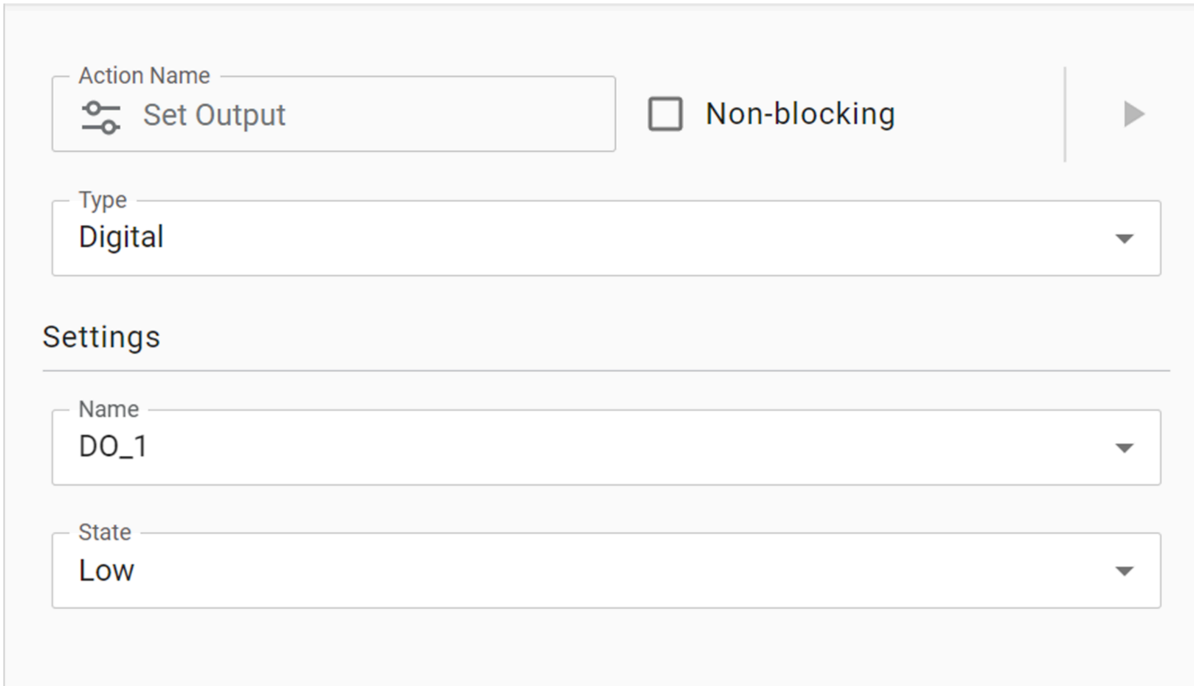


Figure 191: The Tile configuration for when Set Output is the current tile and Digital is selected

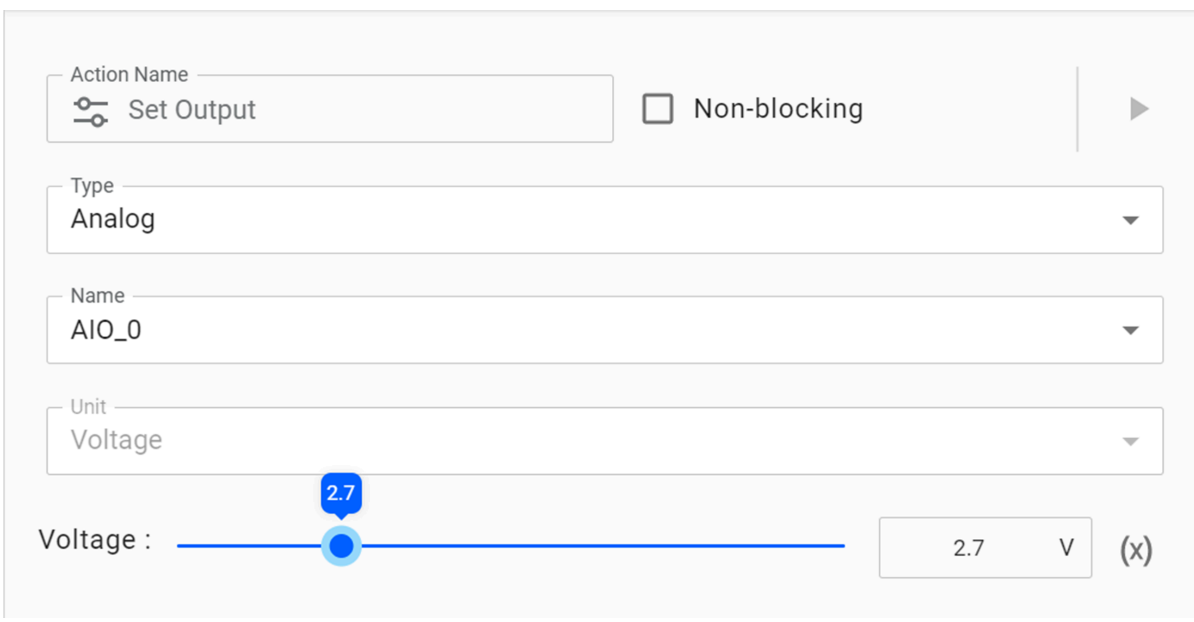


Figure 192: The Tile configuration when Set Output is the current tile and Analog is selected

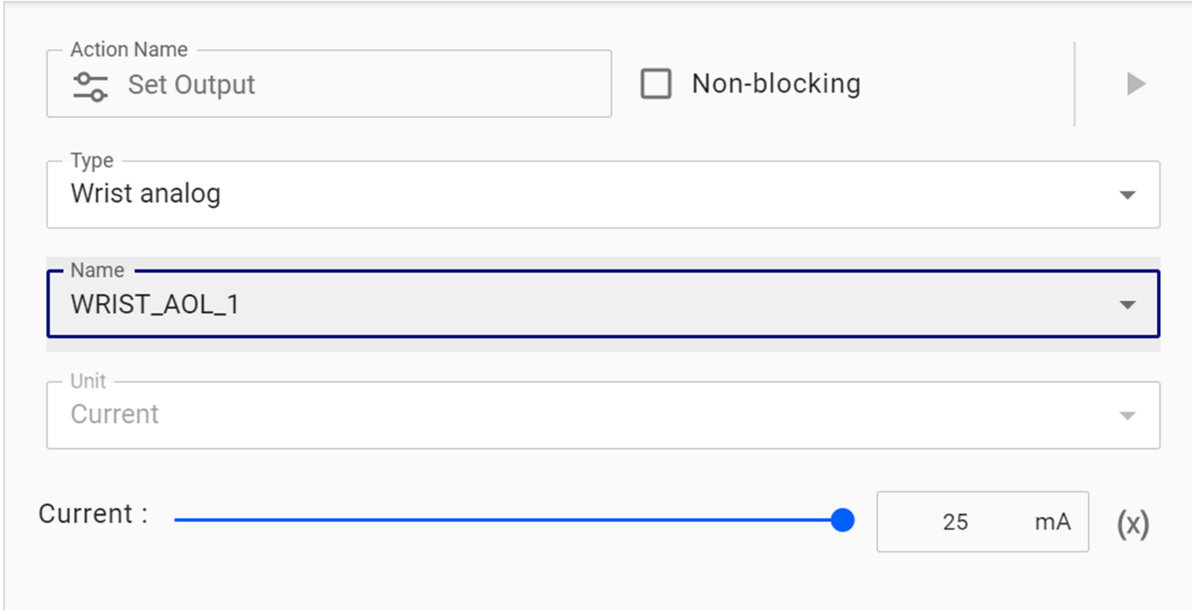


Figure 193: The Tile configuration when Set Output is the current tile and Wrist Analog is selected

Table 96: Industrial I/O Set Output parameters

Item	Description
Type	Select one of the available set output types. • **Digital** • **Analog** • **Wrist Digital** • **Wrist Analog**
Name	Select the name of the available input of the chosen type as defined by the Industrial I/O plugin configuration. The content depends on the selected read input type.
Parameters for the output **Type Digital**and the **Type Wrist Digital**	
State	When the selected output channel is in the selected state, your program resumes. • **High** • **Low**

Item	Description
Parameters for the output **Type Analog** and the **Type Wrist Analog**	
Unit	Displays the unit that is configured for the pin. **Note:** The unit can only be changed from the **Industrial I/O** plugin **Configurations** pane.
Current	Use the slider or enter the maximum current that the output can reach. Range: Range: 0 to 25 mA
Voltage	Use the slider or enter the maximum voltage that the output can reach. Range: Range: 0 - 11 V

If the name of the pin that is selected is not configured to be an input, an error message displays directly in the **Tile configuration** pane.

Example:

This pin is not configured as output.

Industrial I/O: Wait for Input tile parameters

The **Wait for Input** tile is an Industrial I/O tile. When you drag it into an available tile in the **Sequence editor**, the **Tile configuration** changes dynamically to expose the entities of **Wait for Input** that can be configured.

Use **Wait for Input** to pause a program until a signal is sent on an industrial input channel.

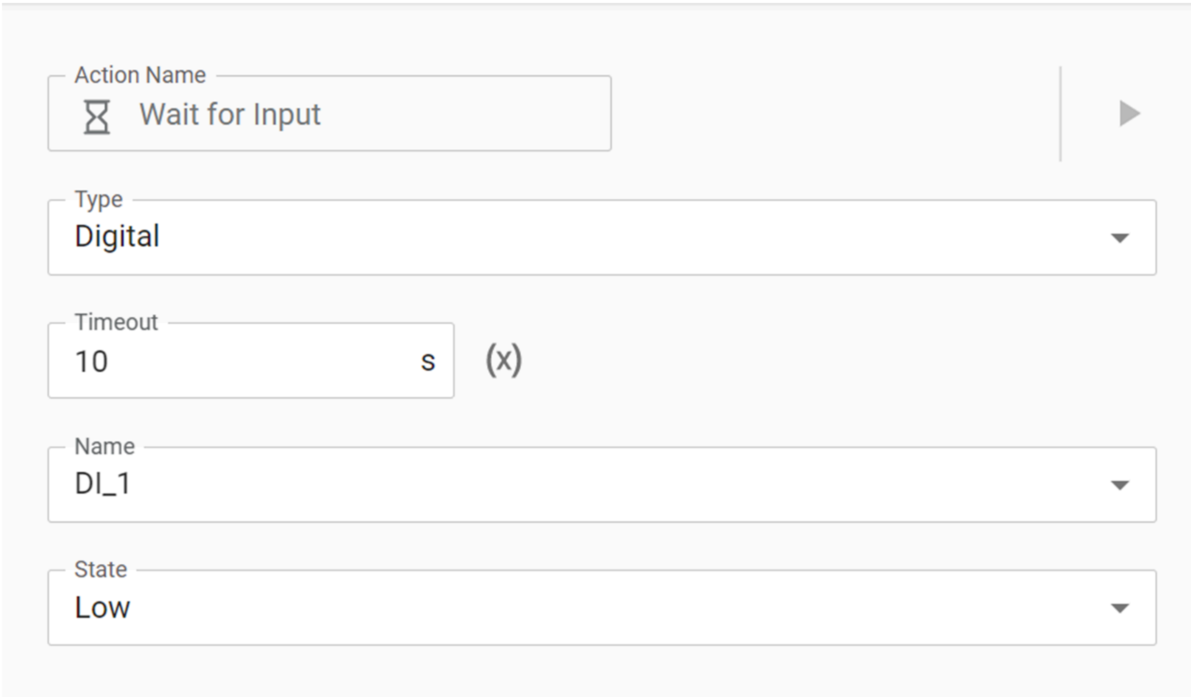


Figure 194: The Tile configuration when Wait for Input is the current tile and the Input Type is either Digital or Wrist Digital

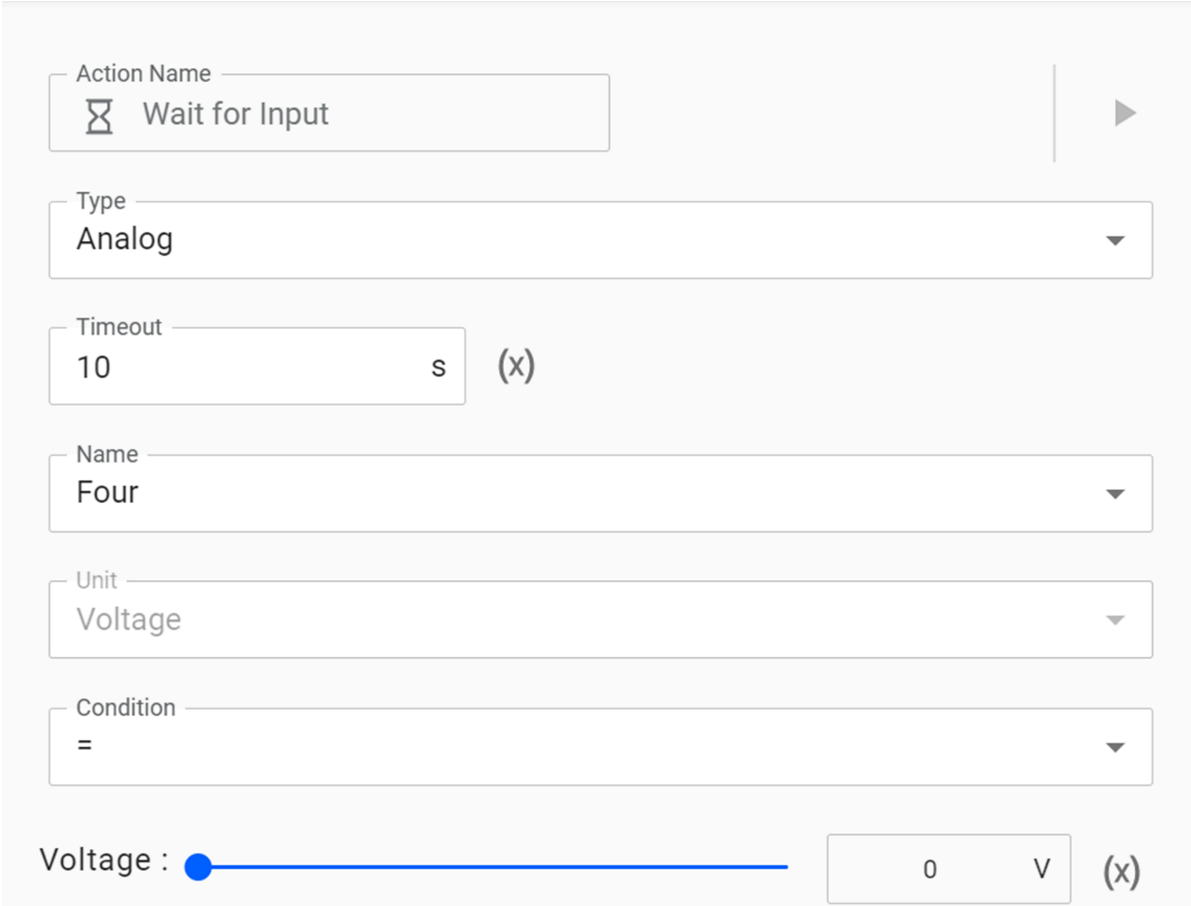


Figure 195: The Tile configuration when Wait for Input is the current tile and the Input Type is either Analog or Wrist Analog

Table 97: Industrial I/O Wait For Input parameters

Item	Description
Type	Select one of the available set output types. • **Digital** • **Analog** • **Wrist Digital** • **Wrist Analog**
Timeout (s)	Force the program to stop running if it must wait the number of seconds entered.

Item	Description
Name	Select the name of the available input of the chosen type as defined by the Industrial I/O plugin configuration. The content depends on the selected read input type.
Parameters specific to **Digital** and **Wrist Digital**	
State	When the selected input channel is in the selected state, your program resumes. • **High** • **Low**
Parameters specific to **Analog** and **Wrist Analog**	
Condition	When the input timeout meets the selected condition, the program resumes. • **==** • **>** • **>=** • **<** • **<=**
Current	Use the slider or enter the maximum current that the output can reach. Range: Range: 0 to 25 mA
Voltage	Use the slider or enter the maximum voltage that the output can reach. Range: Range: 0 - 10 V

Hand Guiding panel

Hand Guiding takes into account external forces and torque feedback from the environment of the robot.

When you press and hold the Wrist enabling device in the middle position, the **Hand Guiding** in the Robot control panel is selected and the **Enabling Device** icon illuminates blue.

Figure 196: Hand Guiding button in the panel

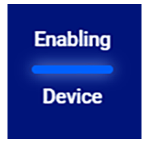


Figure 197: Enabling Device icon

There are two types of motion in Hand Guiding mode.

- **Cartesian**: The tool moves according to the force and torque applied on the wrist, as measured by the built-in force torque sensor.
- **Joint**: The joints of the robot rotate according to the external torques applied at the joint.

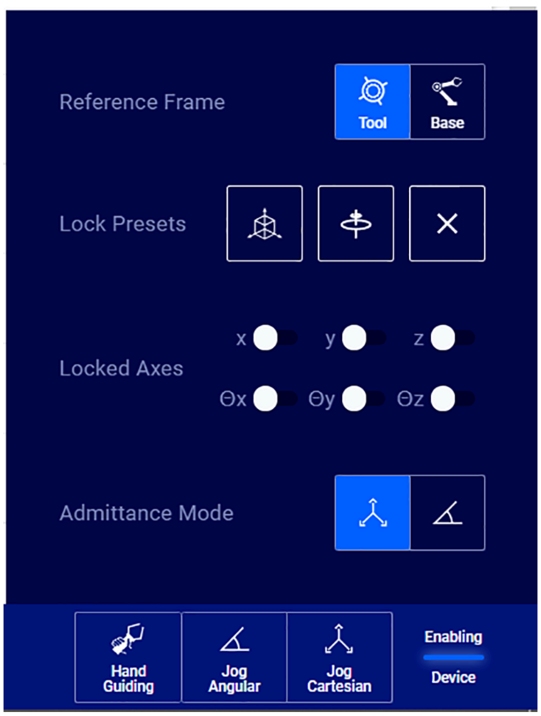


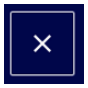
Figure 198: Hand Guiding panel

In this mode, usually you are interacting directly with the arm. However, some actions can be performed from the panel.

- Select the frame of reference as either **Tool** or **Base** when locking axes.
- Use **Lock Presets** to control the Hand Guiding constraints in *Cartesian* translation and rotation admittance mode.

Locks all x, y, and z axes. Cartesian movement is constrained. Only rotations are possible.

Locks all θx, θy, and θz coordinates. Rotations are constrained. Only movement in Cartesian translation are possible.



Cancels the selected lock preset. There are no constraints on the admittance mode.

- Any combination of zero or more axes, and zero or more angles, can be selected from **Locked Axes** to control the Hand Guiding movement with greater precision.

Related topics
Mode: Hand Guiding on page 118

Jog Cartesian panel

To control the position and orientation of the *end effector* of the robot, use the **Jog Cartesian** panel in Kortex Web App.

Important: Jog Cartesian is accessible only when the *teach pendant enabling device* or an external *enabling device* is held in the middle position.

Tap **Jog Cartesian** in the Robot control panel to access the **Jog Cartesian** panel.

Figure 199: Jog Cartesian button in the panel

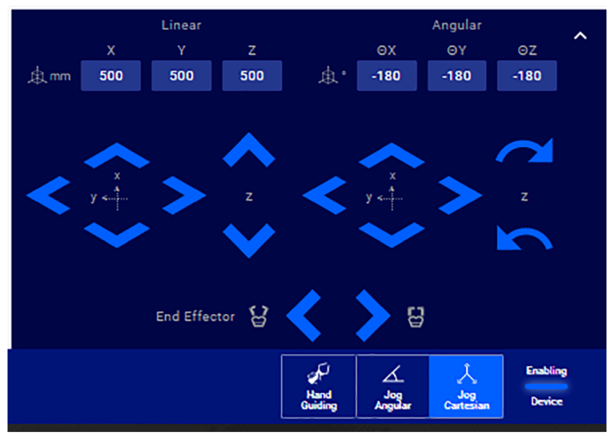


Figure 200: Jog Cartesian panel

Related topics
Mode: Manual Jog on page 115

Translation and rotation from the Jog Cartesian panel

Control linear and angular motion of the tool by using the **Jog Cartesian** panel.

There are two sets of virtual joysticks.

- The linear joystick applies a translational motion to the *TCP*.

- The angular joystick applies a rotational motion to the *TCP* with respect to the current position.

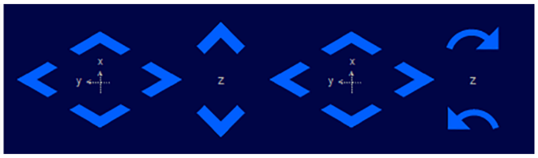


Figure 201: Controls for the Cartesian movement of the arm

Each set of virtual joysticks features a 2-axis joystick for controlling the x and y axes, and a 1-axis joystick to control in the z-axis.

When you tap on the controls, the values are displayed for the current position (x, y, z) and orientation (θx, θy, θz) of the end effector.

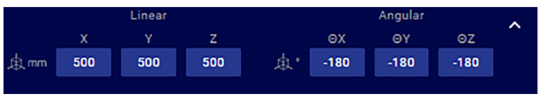


Figure 202: Values of the position of the arm

Note: The orientation representation uses a z-y-x Tait-Bryan intrinsic convention.

Additional settings

Reference frames, linear speeds, and *angular* speeds are not modified as often as other *Cartesian* settings, but can be modified in the **Jog Cartesian** panel.

There is an up arrow in the upper right corner.

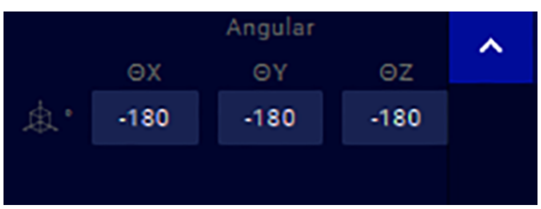


Figure 203: Jog Cartesian panel up arrow

Tapping it unveils additional settings available in Jog Cartesian mode.

- speed control
- reference frame selection

Figure 204: Additional settings in the Jog Cartesian panel

The linear speeds are measured in mm/s and the angular speeds in °/s. Configure the linear and angular speeds for the motion.

The Cartesian velocity of the *TCP* is specified in one of three reference frame conventions.

- **Mixed**: linear with the *base* being the frame of reference; angular with the tool being the frame of reference
- **Tool**: linear and angular with the base being the frame of reference
- **Base**: linear and angular with the base being the frame of reference

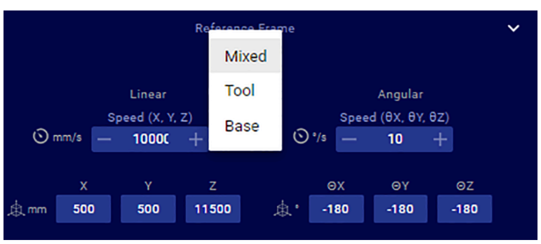


Figure 205: Available frames of reference in the Jog Cartesian panel

Jog Angular panel

To control the joint by joint movement of the robot, use the **Jog Angular** panel in *Kortex Web App*.

With the **Jog Angular** panel, control the robot joint angles and the *end effector* using a mouse or *teach pendant*.

Tap **Jog Angular** in the Robot control panel to access the **Jog Angular** panel.

Figure 206: Jog Angular button in the panel

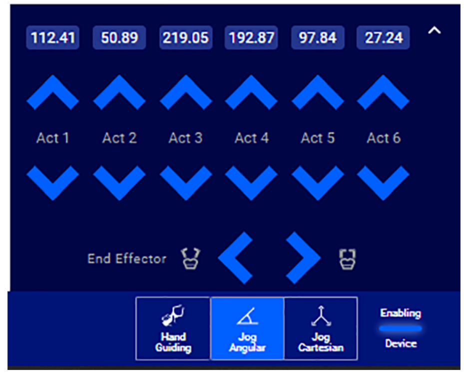


Figure 207: Jog Angular panel

The joint angles are controlled through angular velocity. Control the angle of each *actuator*, as well as end effectors. The robot *arm* responds to the virtual *angular* manipulation.

Note: For joints with joint rotation limits, the robot enforces software joint angle limits to prevent these joints from reaching the physical limits. When you control these joints, the software causes the arm joints to stop responding when the limits are reached.

Each angle is measured in degrees. The value displayed is restricted to minus or plus a full rotation; that is, -360° to +360°.

Related topics

Mode: Manual Jog on page 115

Additional settings

Maximum speeds for actuators and fingers are not modified as often as other *angular* settings, but can be modified in the **Jog Angular** panel.

There is an up arrow in the upper right corner.

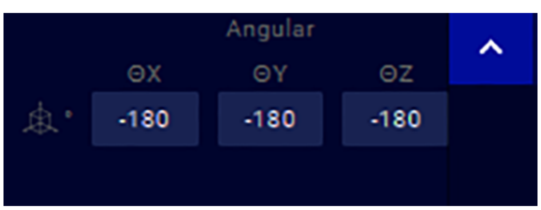


Figure 208: Jog Angular panel up arrow

Tapping it unveils additional settings available in Jog Angular mode.

- Maximum speed for each actuator
- Maximum speed for fingers when a gripper is installed

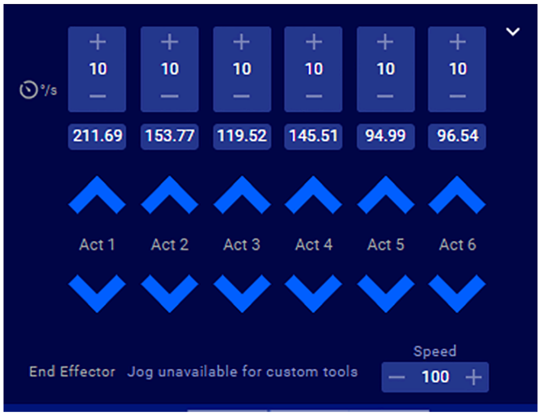


Figure 209: Additional settings in the Jog Angular panel

To control the angle of each *actuator*, use the virtual controls to apply a velocity in the given direction. Increase the angle by tapping the up arrow; decrease the angle by tapping the down arrow. The speed does not exceed the set limit. The angle continues to change as long as the arrows are being used.

Note: The desired joint speeds must be less than the default joint speeds.

Related topics

Robot control panel on page 156

Program creation

The *admin* is responsible for creating programs that can be run safely by *operators*.

It is common practice to create a program using a combination of Hand Guiding, Manual Jog, and *Kortex Web App*. The combination used is a matter of preference. However, there are times when using Hand Guiding and Kortex Web App is the preferred approach, such as when the robot *arm* is approaching a surface.

After the program is debugged and is known to be working as expected, it must be validated. Tap **Validate**.

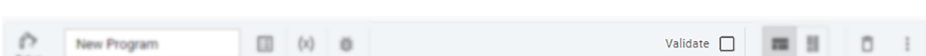


Figure 210: Validate button on the same line as the name of the program on the Program page

Validated programs on the **Programs** page have a green check mark.

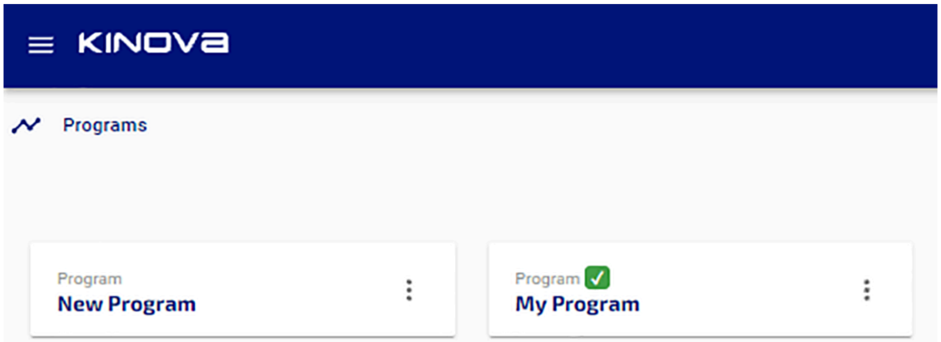


Figure 211: Example of a validated program

Only validated programs are available to operators.

Creating a program

Every program is unique to the work environment of the robot.

About this task

Teaching trajectories to the robot in a program can be made easier using Hand Guiding mode. Use the Cartesian Hand Guiding mode to leverage the force torque sensor for more sensitive and accurate motion. Use angular Hand Guiding to guide any joint.

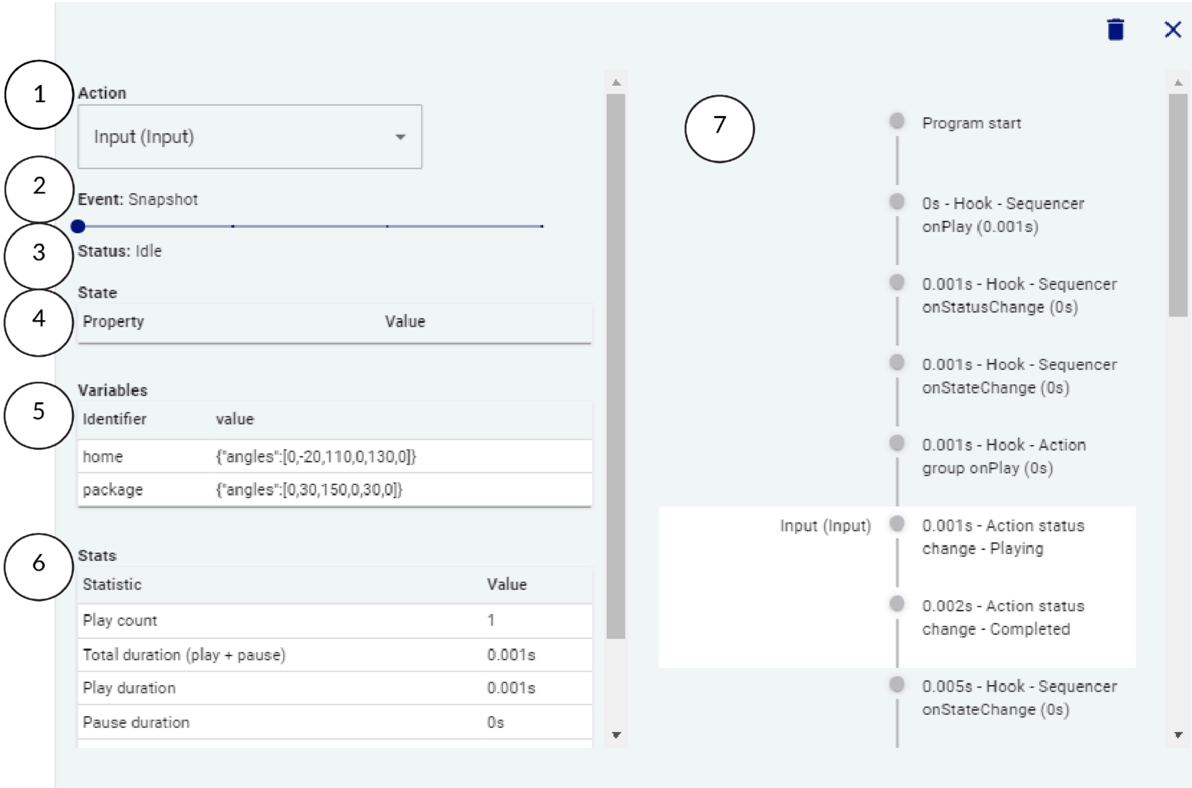
Alternatively, use the Visual Programming interface.

Procedure

1. Tap **Programming** > **Programs**.
2. Tap **Add** (+).
3. Drag and drop tiles from the **Action** menu to the **Sequence editor**.
4. Tap an action tile in the **Sequence editor** to configures the properties of the action.
5. Test the program.
6. Tap **Validate**.

Debug panel overview

It is important to know whether the program is running optimally or suboptimally before it is released to the users. The **Debug** panel launches at the end of the run of a debugging session.



1	Action	Select one of the actions in the drop-down to highlight them in the timeline. Alternatively, tap on any of the events in the timeline to view the action in the drop-down list.
2	Event	Event is the action selected from which the debugger is to run.
3	Status	Status is the mode of operation the robot is in.
4	State	State lists the different property states and their values during the run of the program.
5	Variables	Variables are used by the program. • Identifier is the name of the variable. • Value is the value of the named variable.
6	Stats (Statistics)	Statistics are information about the program. • Play count is the number of times the program is run. • Total duration (play + pause) is the total of time the program runs, before pauses, before the program ends. • Play duration is the amount of time the program plays before it ends. • Pause duration is the amount of time the program is paused. • Pause count is the number of times the program is paused. • Failure count is the number of times the program fails.

7	Timeline of events	The visual timeline of events that occur while the program runs. Tap on any of the events in the timeline to view information related to that event.

Debugging a program

Determine whether the program is running optimally or suboptimally by using the debugger and the play controls in the Robot control panel.

About this task

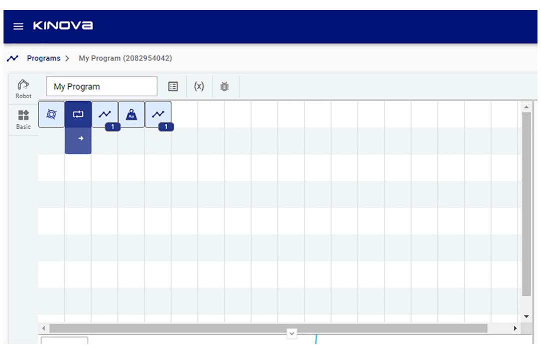
There must be a program in the **Program** page.

Procedure

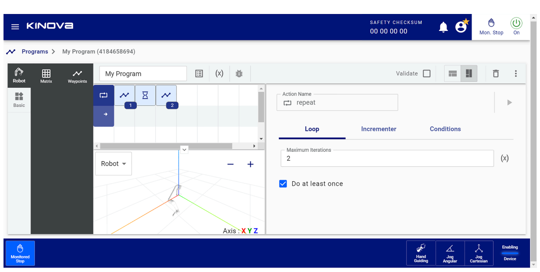
1. Tap Hold-to-Run ( ).
2. Tap **Debug** ( ).

3. Tap a tile in the **Sequence editor** from where you want to start debugging.

Tip: It is easier to debug small portions of a program. However, remember to debug from the very beginning of the program as well to make sure the program runs smoothly between sections.



The **Tile configuration** populates with details about the selected action.



Program is ready to use

The robot is ready to be used in its environment and to perform its tasks.

All users can run a program that is validated. There are different ways to run a program.

- Run the program in Automatic mode. This is the most common way of running a program for operators.
- (Admin users only) Run a program in Hold-to-Run mode.

Running a program in Automatic mode

Work in Automatic mode to run a program with little to no intervention.

Before you begin

The program is fully debugged and validated.

About this task

The robot must be on and operational. All users can run a validated program.

Important: To run in Automatic mode, you usually use the ACK button on the *controller* or an external ACK button, and **Acknowledge Automatic Mode** is enabled from **Systems** > **Robot**. **Acknowledge Automatic Mode** is enabled by default.

It is possible to avoid using the ACK button on the controller.

- Disable the **Acknowledge Automatic Mode** button on **Systems** > **Robot** > **Acknowledge Automatic Mode**.
- Assign **Acknowledgment** to one channel on **Safety I/Os** > **Inputs**.

Procedure

1. Tap **Programming** > **Programs**.
2. Tap the program to select a program to run.
3. Tap Automatic ().
4. Press the ACK button on the controller.
5. Tap  in the Robot control panel ().
   The program runs at the programmed speed.

   Note: At any time in Automatic mode, tap the Program speed control buttons to slow down the speed of the program and to resume its normal speed.

Related topics

Mode: Automatic on page 120

Running a program in Hold-to-Run mode

Work in Hold-to-Run mode to run a program knowing that intervention may be required.

Before you begin

The enabling device must be pressed and held in the middle position to run the program.

About this task

The robot must be on and operational. When you run a program in Hold-to-Run mode, you may want to determine whether it needs further debugging or they may want to see how it works either at reduced or normal speeds.

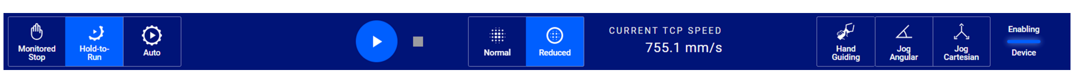
Procedure

1. Tap **Programming** > **Programs**.
2. Tap the program to select a program to run.
3. Tap Hold-to-Run ( ).
4. Tap in the Robot control panel.

   Note: If the playing speed of the program is more than 250 mm/s, a warning dialog launches. Follow safety measures and confirm before proceeding.

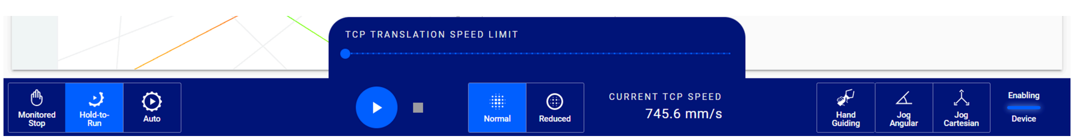
   The program runs at the programmed speed.

   Note: At any time in Hold-to-Run mode, tap the Program speed control buttons to slow down the speed of the program and to resume its normal speed.
5. Tap **Reduced** ( ) or **Normal** ( ) speed.



6. Tap in the Robot control panel.

   When you operate Hold-to-Run mode at **Normal** safety speeds, you can change the speed limit of the *TCP* translation speed while the program is running.



Related topics

Mode: Hold-to-Run on page 119

Diagnostics

You can view potential issues in any part of the robot from the diagnostic pages.

The pages are found under the **Diagnostics** menu.

- **Status**: displays the results of all the built-in tests
- **Monitoring**: displays live information from the sensors about the robot, divided into four groups

Faults identified by the diagnostics tool, as well as normal operating events, generate messages that are displayed as notifications and can be found in the **Events** page. Access all events through Kortex Web App by tapping the **Notifications** icon ( ).

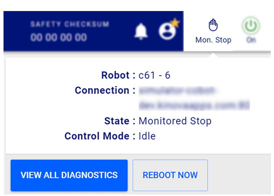
Built-in self tests

Kortex Web App displays notifications, such as a robot component not functioning properly, on the **Status** page.

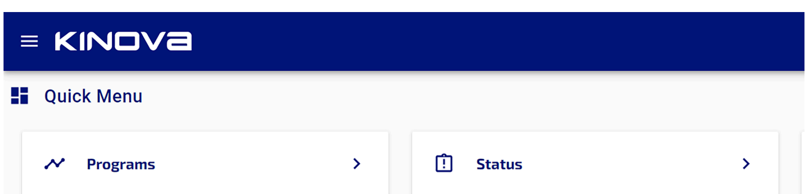
The **Status** page displays an overview of the status of each component in the system.

There are several ways to access the **Status** page.

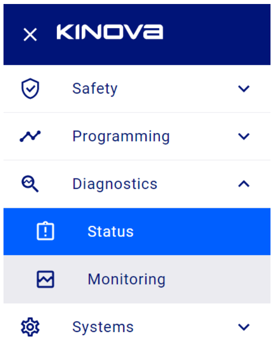
- Tap the Robot status icon ( ) and tap **VIEW ALL DIAGNOSTICS**.



- Tap **Status** from Quick Menu.



- Tap *hamburger menu* > **Diagnostics** > **Status**.



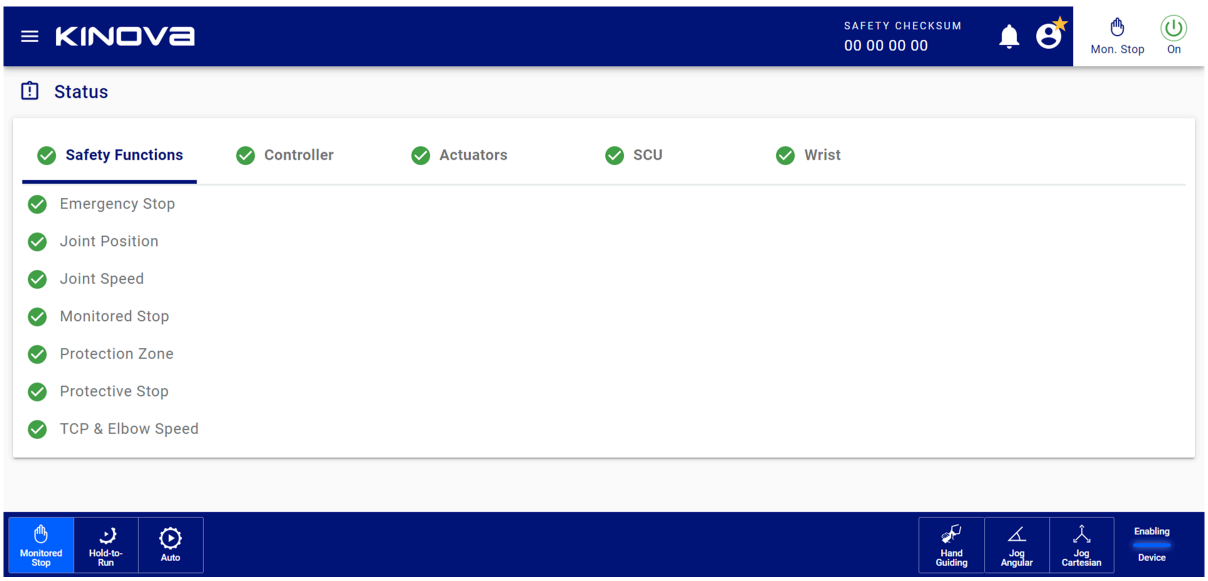


Figure 212: Status page

The page is divided into several panes.

- **Safety Functions**
- **Controller**
- **Actuators**
- **SCU**
- **Wrist**

Status of safety functions

The safety functions that are listed on the **Safety Functions** pane on the **Status** page are the same safety functions listed with their current state.

Safety functions are monitored by the *SCU*. The state of each safety function is displayed.

- A green circle with a check mark () beside the name of the safety function indicates that safety function has no events.
- A red octagon () beside the name of the safety function indicates that safety function has errors, or events.

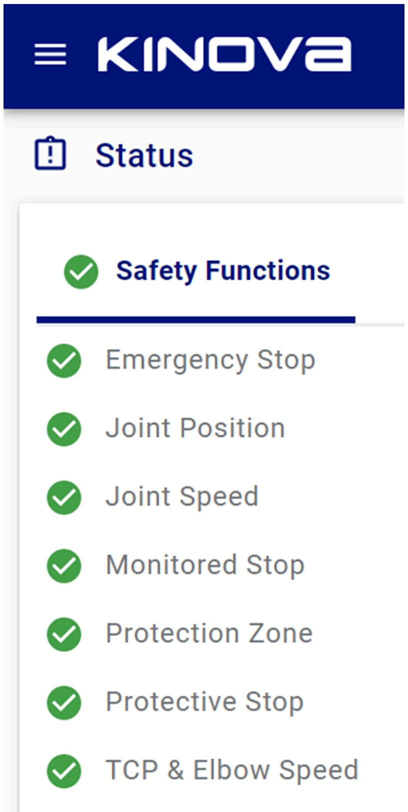


Figure 213: Status of safety functions

Related topics

List of safety functions on page 54

SF01 Emergency stop on page 55
SF02 Protective stop on page 56
SF03 Joint position monitoring on page 56
SF04 Joint speed monitoring on page 57
SF05 Monitored stop on page 58
SF06 Protection zone monitoring on page 58
SF07 TCP and elbow speed monitoring on page 59

Status of the controller

The status of each part of the *controller* is displayed on the **Controller** pane on the **Status** page.

The status of the controller summarizes the state of the entire robot.

On the **Controller** pane, you can search for different issues or view only what is failing.

Figure 214: Search and filter on Controller pane

Some status have an tooltip ( ) beside them. Tap them to view the thresholds associated with the status.

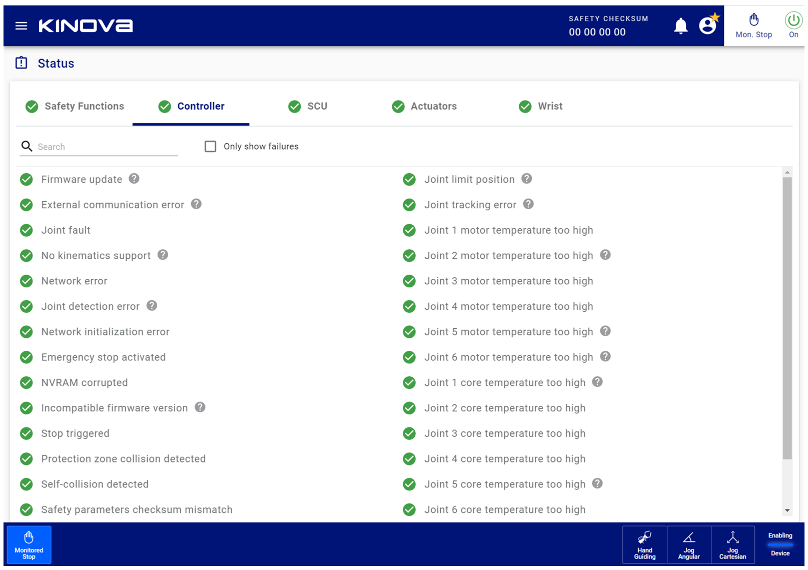


Figure 215: Status page of the controller

Each controller status displays one of its two states.

- A green circle with a check mark ( ) beside the status indicates that controller status is fine.
- A red octagon ( ) beside the name of the controller status indicates that status has errors; more information may be available by tapping the Notifications icon ( ).

The **Controller** pane displays all the status that are being monitored, including internal variables. Some of the internal variables cannot be accessed by users; the variables may be associated with hardware problems.Contact support at support@kinova.ca

Table 98: Listing of controller status

Status	Not addressable by users	Addressable by users
Firmware update failure		x
External communication error		x

Status	Not addressable by users	Addressable by users
Joint fault		x
No kinematics support	x	
Network error	x	
Joint detection error	x	
Network initialization error	x	
Emergency stop activated		x
NVRAM corrupted	x	
Incompatible firmware version		x
Stop triggered		x
Protection zone collision detected		x
Self-collision detected		x
Safety parameters checksum mismatch		x
Hard Stop command failed		x
Singularity error detected		x
Joint limit position		x
Joint tracking error		x
Joint 1-6 motor temperature too high		x
Joint 1-6 core temperature too high		x
Serial number mismatch between arm and calibration file		x
Actuator model invalid	x	

When you are not able to intervene with a status, you can attempt to recover on your own before contacting support.

1. Take note of the exact error along with the conditions that seem to have triggered it.

   Tip: When a fault triggers, a message launches in Kortex Web App that provides information.

- Name of the fault
- Error code
- Explanation of what can trigger the fault
- What can be done to fix the problem

2. Reboot the controller.
3. Re-install the *firmware update package (SWU)* if the error persists.
4. If the problem persists, contact support at support@kinova.ca.

Related topics
Protection zones on page 77

Firmware update

What does *Firmware update* monitor?

Firmware update failure monitors whether the latest *firmware* uploaded to the robot is installed properly.

Causes of the trigger

- Communication is faulty when the *SWU* package is uploaded to the robot.
- Firmware is corrupted during its initial download from the Kinova Artifactory server.

Procedure

1. Download the latest version of the firmware from our Artifactory server.
2. Restart the firmware installation.

Incompatible firmware version

What does *Incompatible firmware version* monitor?

Incompatible firmware version monitors whether the *firmware* installed on the controller is compatible with the robot.

Cause of the trigger

A *SWU* package meant for another model of the robot may have been installed on the controller.

Procedure

1. Download the latest version of the firmware from our Artifactory server.
2. Restart the firmware installation.

Joint detection error

What does *Joint detection error* monitor?

Joint detection error monitors whether the number of *actuators* detected by the *controller* is the same as its expected configuration.

Cause of the trigger

Communication packets may be lost between the internal devices of the robot.

Procedure

1. Take note of the exact error along with the conditions that seem to have triggered it.
2. Reboot the controller.
3. Re-install the *SWU* package if the problem persists.
4. If the problem persists, contact support at support@kinova.ca.

Joint fault

What does *Joint fault* monitor?

Joint fault monitors all actuators to determine whether any of them are currently in a fault state.

Cause of the trigger

There are multiple reasons for actuators being in fault.

Procedure

1. Tap **Actuators**.
2. Search `Joint`.
3. Identify the *actuator* in fault and its trigger.

Related topics

Status of the actuators on page 306

Status of the safety actuators

No kinematics support

What does *No kinematics support* monitor?

No kinematics support monitors whether the *controller* can find the kinematic information to control the *arm* properly.

Cause of the trigger

The files that contain the information in the file system of the controller may be corrupted.

Procedure

1. Take note of the exact error along with the conditions that seem to have triggered it.
2. Reboot the controller.
3. Re-install the *SWU* package if the problem persists.
4. If the problem persists, contact support at support@kinova.ca.

Maximum ambient temperature reached

What does *Maximum SCU Board temperature* monitor?

Maximum SCU Board temperature reached monitors the temperature of the *SCU* board.

Cause of the trigger

- The room where the *controller* is installed may be too warm.
- There is not enough space behind the controller for its fan to disperse its internal heat properly; the temperature close to the controller is warmer than the actual temperature of the room.

Remedy

Make sure the controller has enough room to dissipate heat properly and that it operates under the recommended conditions.

Related topics

Considerations before installing the controller on page 134

Maximum core temperature reached

What does *Maximum core temperature reached* monitor?

Maximum core temperature reached monitors the temperature within the core of the *controller*. It must not exceed 75 °C.

Cause of the trigger

The fan within the controller is not able to dissipate heat efficiently enough.

Procedure

- Make sure the air filters on the controller are clean.
- Make sure the fan is spinning freely.

- Make sure the controller has enough room to dissipate heat properly.

Related topics
Considerations before installing the controller on page 134

Maximum current reached

What does *Maximum current reached* monitor?

Maximum current reached monitors whether the electrical current pulled by the *controller* stays below a certain amount.

Cause of the trigger

The instantaneous power pulled by the robot for its motion in addition to the power required by the components of the controller may exceed the limits of the hardware.

Procedure

- Avoid having computationally expensive algorithms running on the controller when the robot has to apply high torques if possible.
- Reduce the force or velocity output by the robot during its trajectory.

Network error

What does *Network error* monitor?

Network error monitors whether the controller can find all the expected devices on its network.

Cause of the trigger

Communication packets may be lost between the internal devices of the robot.

Procedure

1. Take note of the exact error along with the conditions that seem to have triggered it.
2. Reboot the controller.
3. Re-install the *SWU* package if the problem persists.
4. If the problem persists, contact support at support@kinova.ca.

Procedure

1. Reboot the controller.
2. Re-install the *SWU* package if the problem persists.
3. Contact support at support@kinova.ca

Self-collision detected

What does *Self-collision detected* monitor?

Self-collision detected monitors whether non-consecutive robot links are at risk of entering in contact with each other.

Cause

Manual control and trajectories can drive the robot in configurations where self-collisions can occur.

Procedure

- Modify your trajectory accordingly when it is clear from observing the pose of the robot which links are at risk of entering in contact.
- Validate the size of your Tool Sphere in the **Protection Zones** page.

Related topics

Protection zones on page 77

Status of the actuators

The status of each *actuator* is displayed on the **Actuators** pane of the **Status** page.

The status of the actuators summarizes the state of the actuators.

On the **Actuators** pane, you can search for different issues or view only what is failing.

Figure 216: Search and filter on Actuators pane

Some status have an information icon beside them. Tap them to view the thresholds associated with the status.

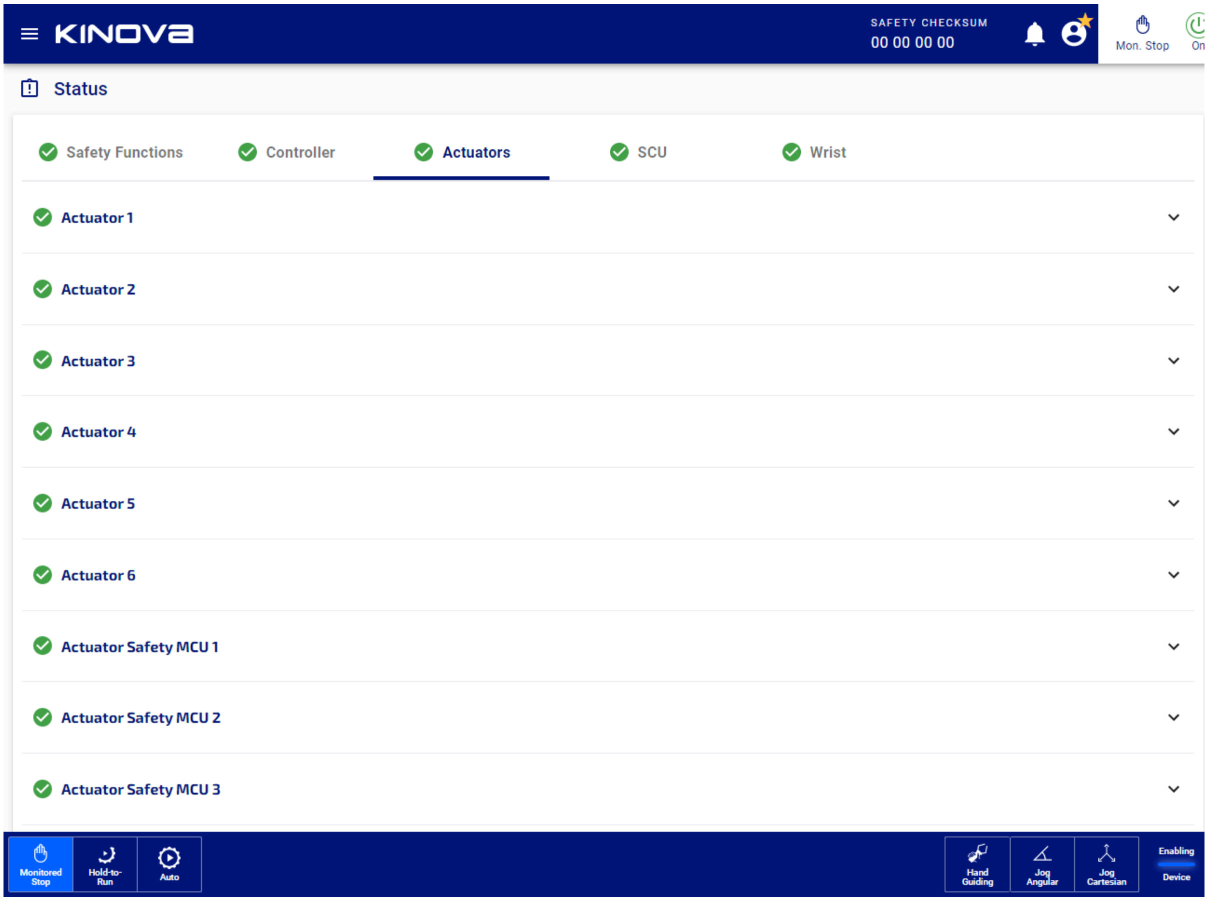


Figure 217: Status page of the actuators when all actuator status are good

Each actuator pane displays one of two states, depending on the overall state of all its status.

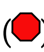
- A green circle with a check mark () beside the actuator indicates that all status for that actuator is fine.
- A red octagon () beside the actuator indicates that at least one status of the actuator has errors. More information is available by expanding the pane and possible recovery steps may be available by tapping the Notifications icon ().

Figure 218: Status page of the actuators with an actuator pane expanded

The **Actuators** pane displays all the status that are being monitored, including internal variables. Some of the internal variables cannot be accessed by users; the variables may be associated with hardware problems.

Each safety MCU actuator status is listed separately for each actuator.

Table 99: Listing of status of each actuator safety MCU

Status	Status
Startup self-test timed out	VMON 48V disagreement
VMON 48V overvoltage	Motor housing temperature disagreement
VMON 48V undervoltage	Joint position disagreement
VREF 2.5V overvoltage	Motor position disagreement
VREF 2.5V undervoltage	Joint velocity disagreement
Motor current sense disagreement	Motor velocity disagreement
Motor encoder error pin asserted	Motor current disagreement
Joint encoder diagnostic pin asserted	Safe communication
Encoders position disagreement	Independent or window watchdog reset occurred
Motor housing temp sensor I2C error	MCU core general purpose registers
Motor housing temperature too high	MCU core arithmetic functions
Motor housing temperature too low	MCU core multiplier functions
Core temperature too high	MCU core control flow instructions
Core temperature too low	MCU core load store unit
Emulated EEPROM corrupted	MCU core shift rotate functions
XCOM SPI communication failure	MCU core stack pointer
Motor driver enable pin disagreement	MCU core pipeline functions
Brake driver enable pin disagreement	MCU RAM BIST fault

When you are not able to intervene with a status, you can attempt to recover on your own before contacting support.

1. Take note of the exact error along with the conditions that seem to have triggered it.

   Tip: When a fault triggers, a message launches in Kortex Web App that provides information.

- Name of the fault
- Error code
- Explanation of what can trigger the fault
- What can be done to fix the problem

2. Reboot the controller.
3. Re-install the *SWU* if the error persists.
4. If the problem persists, contact support at support@kinova.ca.

Related topics

Joint fault on page 303

Remedy for MCU firmware corruption

Many different errors of the status of the *actuators* can be caused by corrupted *firmware* on the *MCU*.

About this task

Take the same remedial steps for each of the actuator status.

- Startup self-tests timed out
- *MCU* core general purpose registers self-test failure
- *MCU* core arithmetic functions self-test failure
- *MCU* core multiplier functions self-test failure
- *MCU* core control flow instructions self-test failure
- *MCU* core load store unit self-test failure
- *MCU* core stack pointer self-test failure
- *MCU* core pipeline functions self-test failure
- *MCU* core FPU registers bank self-test failure
- *MCU* core FPU single precision functions self-test failure

Procedure

1. Take note of the exact error along with the conditions that seem to have triggered it.
2. Reboot the *controller*.
3. Re-install the *SWU* if the error persists.
4. If the problem persists, contact support at support@kinova.ca.

Emergency stop button signals disagreement

What does *Emergency stop button signals disagreement* monitor?

Emergency stop button signals disagreement monitors whether the signals coming from the emergency stop button are in alignment with the state of the robot.

Cause

The emergency stop button may not be fully engaged or fully released.

Remedy

Procedure

1. Make sure the button is either fully engaged or fully released.
2. Take note of the exact error along with the conditions that seem to have triggered it.
3. Reboot the controller.
4. Re-install the *SWU* package if the problem persists.
5. If the problem persists, contact support at support@kinova.ca.

Safety input signals disagreement

What does *Safety input signals disagreement* monitor?

Safety input signals disagreement monitors whether each of the safety I/O signals are in alignment with the state of the robot.

Cause

The connections on the *I/O* panel may be loose.

Remedy

Procedure

1. Make sure the connections on the I/O panel are secure.
2. Take note of the exact error along with the conditions that seem to have triggered it.
3. Reboot the controller.
4. Re-install the *SWU* package if the problem persists.
5. If the problem persists, contact support at support@kinova.ca.

Core temperature too high

What does *Core temperature too high* monitor?

Core temperature too high monitors whether the core temperature is below a specific temperature.

Cause of the trigger

Trajectories with sustained high torques generate heat at a faster pace than it is dissipated.

Remedy

Leave the robot at rest and let it cool down. If a continuous program is triggering this fault, consider modifying the trajectory to reduce the duration of high applied torques.

Core temperature too low

What does *Core temperature too low* monitor?

Core temperature too low monitors whether the core temperature is above a specific temperature.

Cause of the trigger

Low internal temperature is a consequence of low ambient temperature.

Remedy

Make sure the robot is operating in the recommended ambient temperature.

Motor housing temperature too high

What does *Motor housing temperature too high* monitor?

Motor housing temperature too high monitors whether the temperature of the motor housing is below a specific temperature.

Cause of the trigger

Trajectories with sustained high torques generate heat at a faster pace than it is dissipated.

Remedy

Leave the robot at rest and let it cool down. If a continuous program is triggering this fault, consider modifying the trajectory to reduce the duration of high applied torques.

Motor housing temperature too low

What does *Motor housing temperature too low* monitor?

Motor housing temperature too low monitors whether the temperature of the motor housing is over a specific temperature.

Cause of the trigger

Low internal temperature is a consequence of low ambient temperature.

Remedy

Make sure the robot is operating in the recommended ambient temperature.

Status of the wrist

The status of each part of the *wrist* is displayed on the **Wrist** pane on the **Status** page.

The status of the wristsummarizes the state of the wrist.

On the **Wrist** pane, you can search for different issues or view only what is failing.

Figure 219: Search and filter on

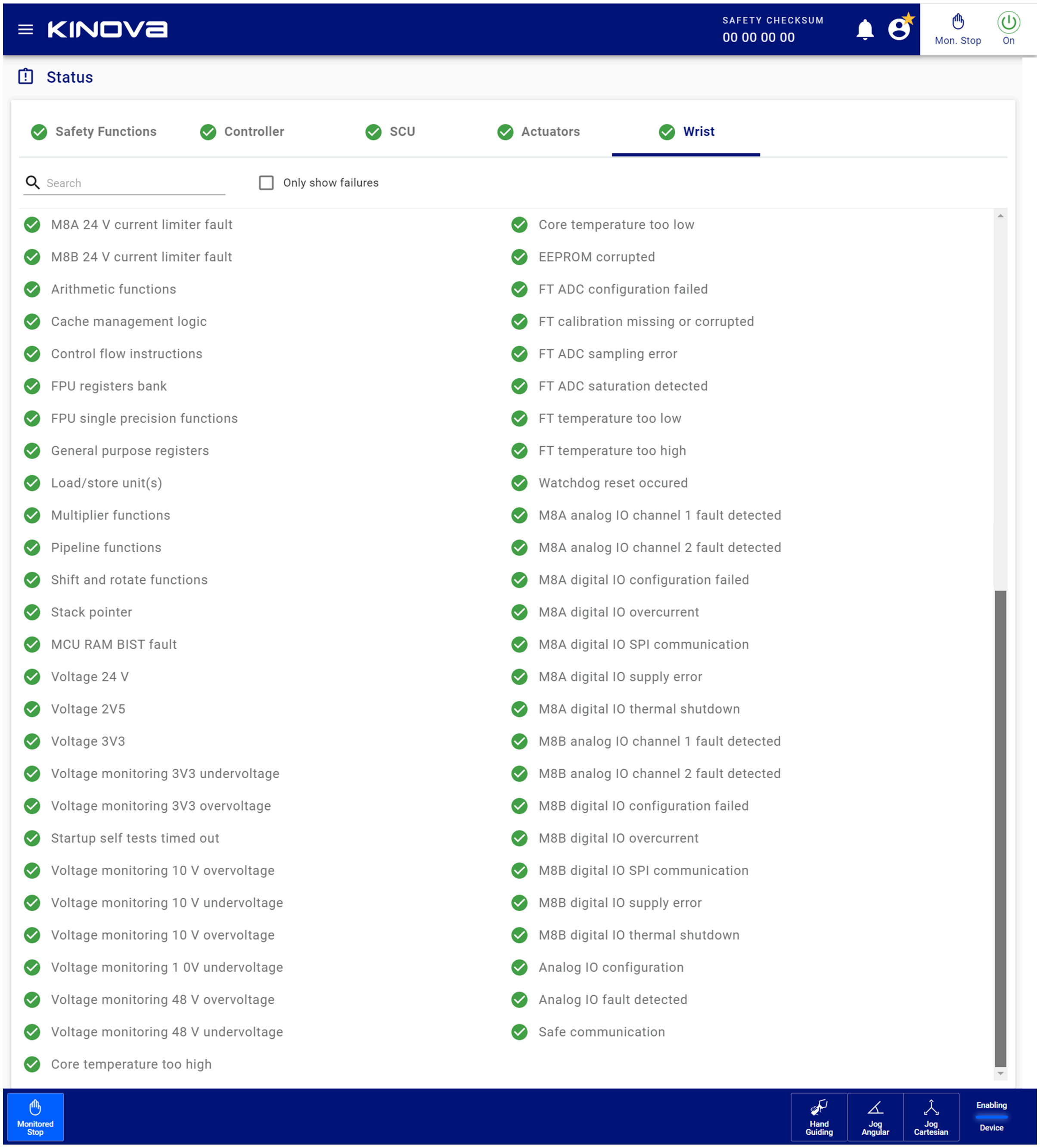


Figure 220: Status page of the wrist

Each wrist status displays one of its two states.

- A green circle with a check mark ( ) beside the status indicates that controller status is fine.
- A red octagon ( ) beside the name of the controller status indicates that status has errors; more information may be available by tapping the Notifications icon ( ).

The **Wrist** pane displays all the status that are being monitored, including internal variables. Some of the internal variables cannot be accessed by users; the variables may be associated with hardware problems. Contact support at support@kinova.ca

When you are not able to intervene with a status, you can attempt to recover on your own before contacting support.

1. Take note of the exact error along with the conditions that seem to have triggered it.

   Tip: When a fault triggers, a message launches in Kortex Web App that provides information.

   - Name of the fault
   - Error code
   - Explanation of what can trigger the fault
   - What can be done to fix the problem
2. Reboot the controller.
3. Re-install the *SWU* if the error persists.
4. If the problem persists, contact support at support@kinova.ca.

Robot monitoring

At any time, you can view current information about the *base*, *actuators*, *end effector* that is in use, and the flange pose.

The **Monitoring** pages display the live feedback from the sensors on the robot, as well as values computed from that feedback. Computed values include TCP position and velocity, among others.

To access the robot **Monitoring** page, tap > **Diagnostics** > **Monitoring**.

Figure 221: Overview pane of the Monitoring page

The page is divided into two panes.

- **Overview**
- **Detailed**

Both panes have a **Data Snapshot** button. To use the snapshot button, you must either have a *USB* in the port on the controller or you must be working from a computer. Tapping **Data Snapshot** saves a JSON file that contains all the information displayed on the pane you are viewing in the **Monitoring** page.

Besides saving a snapshot of the robot and viewing the robot, you can change the Actuators table to invert the top row with the first column.

Overview pane of the Monitoring page

The **Overview** pane displays an overview of the position and state the robot is currently in.

It is subdivided into four main parts.

- **Controller**
- **Actuators**
- **End Effector**
- **Flange Pose**

Table 100: Content of the Overview pane

Part	Information displayed
Controller	Operating mode
	Control mode
	Servoing mode
Actuators	Position of each actuator; measured in °
	Torque of each actuator; measured in N·m
	Velocity of each actuator; measured in °/s
End Effector	X, Y, and Z; measured in mm
	Theta X, theta Y, and theta Z; measured in °
	Tool twist linear X, Y, and Z; measured in m/s
	Tool twist angular X, Y, and Z; measured in °/s
	Commanded tool pose X, Y, and Z; measured in mm
	Commanded tool pose Theta X, theta Y, and theta Z; measured in °
	External tool wrench torque x, y, z measured in N·m
	External tool wrench force x, y, z measured in N
Flange Pose	X, Y, and Z; measured in mm
	Theta X, theta Y, and theta Z; measured in °

Detailed pane of the Monitoring page

The **Detailed** pane displays an more details about the current details of the *base* and the *actuators*. The information about the *end effector* and the *flange* are the same as that in the **Overview** pane.

Figure 222: Detailed pane of the Monitoring page

Table 101: Content of the Detailed pane

Part	Information displayed
Controller	Operating mode
	Control mode
	Servoing mode
	Arm voltage
	Arm current
	CPU temperature
	Acceleration X, Y, and Z
	Angular velocity X, Y, and Z
Actuators	Position of each actuator, measured in °
	Torque of each actuator, measured in N·m
	Velocity of each actuator, measured in °/s
	Motor current of each actuator; measured in A
	Voltage of each actuator; measured in V
	Motor temperature; measured in °C
	Core temperature; measured in °C

Part	Information displayed
End Effector	X, Y, and Z; measured in mm
	Theta X, theta Y, and theta Z; measured in °
	Tool twist linear X, Y, and Z; measured in m/s
	Tool twist angular X, Y, and Z; measured in °/s
	Commanded tool pose X, Y, and Z; measured in mm
	Commanded tool pose Theta X, theta Y, and theta Z; measured in °
	External tool wrench torque x, y, z measured in N·m
	External tool wrench force x, y, z measured in N
Flange Pose	X, Y, and Z; measured in mm
	Theta X, theta Y, and theta Z; measured in °

Event notifications

Notifications about the robot and actions taken in *Kortex Web App* are sent through Kortex Web App.

Access notifications of events by tapping the notification bell.

Figure 223: Notification bell when not selected

Figure 224: Notification bell when selected

View events that have happened from the **Events** page.

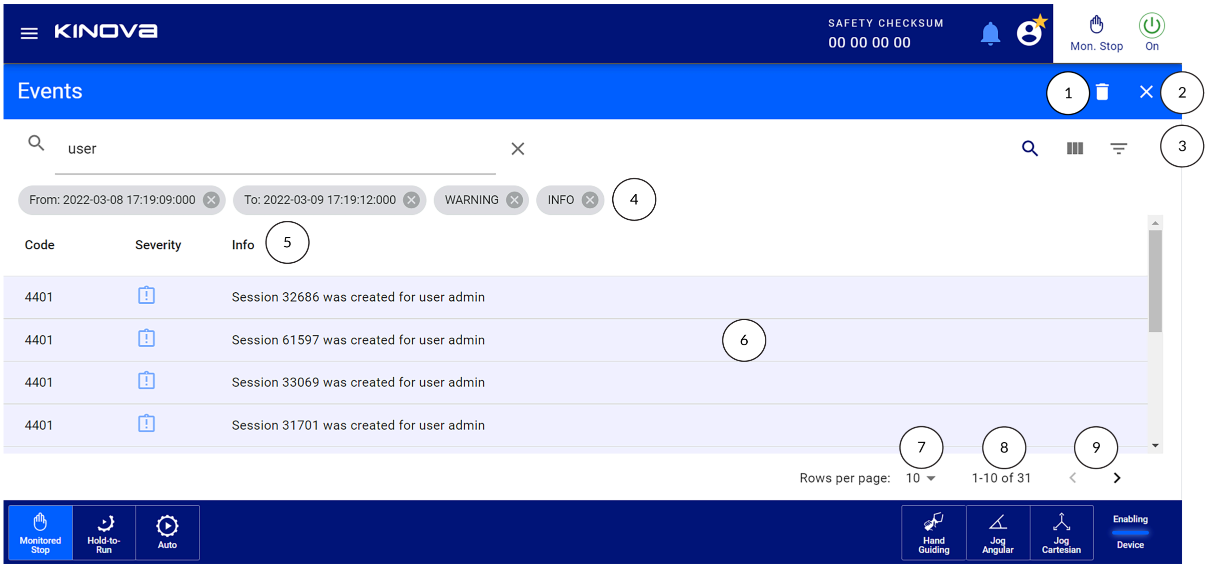


Figure 225: The Events page and its parts

1 **Delete** icon ( )

Tap to delete the events on the page.

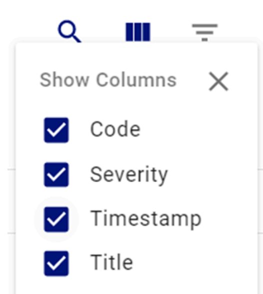
2 **Close** icon (X)

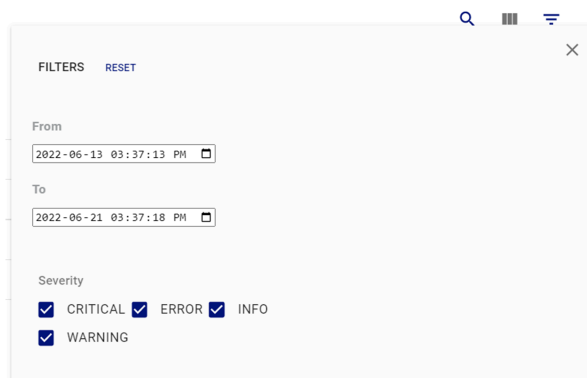
Tap to close the page.

3 Filtering tools

All filtering tools can be used independently or with each other.

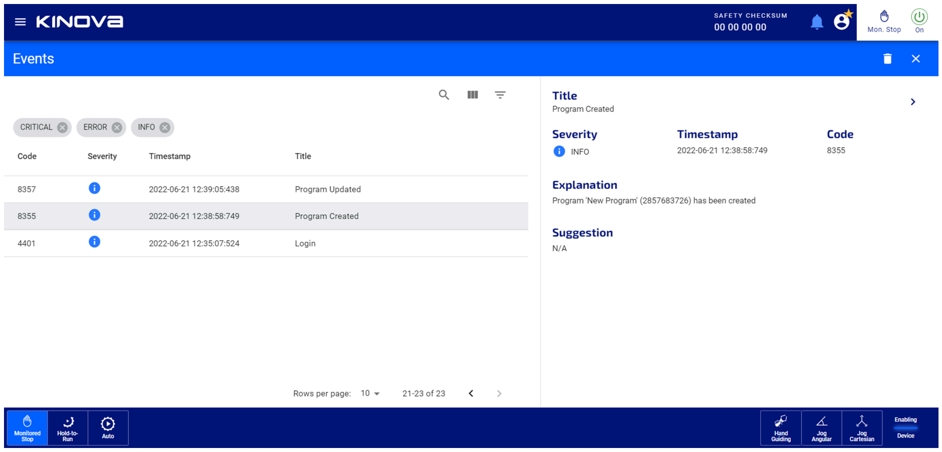
Search bar	Enter one or more words that may be found in the title of the notification to view all similar notifications. Tap **X** to remove the search.
Search icon ( )	Tap the icon to launch the **Search** bar.
Show Columns ( )	Tap to launch the **Show Columns** pane.



		Select one or more of the types of information you want to see. • Code • Severity • Timestamp • Title Tap **X** to close the **Show Columns** pane.
	Filter icon (☰)	Tap to launch the **Filters** pane.  Enter start and end dates if you want to view notifications within a time range. Select and deselect filters based on **Severity** of the notification. • **CRITICAL** • **ERROR** • **INFO** • **WARNING** Tap **RESET** to clear the current filters. Tap **X** to close the **Filters** pane.
4	Active filters	
	The filters that have been selected using the filtering tools. Tap **X** to close the filter that is not applicable.	
5	Table column headers	
	The column headers indicate the type of information that is displayed.	

6 Information

A summary of each event that meets the filtering criterion is displayed in the Details pane.



Title Summary of the event

Severity Level of severity of the event

Timestamp Date and time the event took place

Code Number associated with the event to make reporting easy

Explanation Reason the event is triggered

Suggestion Details what can be done to circumvent the event if applicable

Tap **>** to close the **Details** pane.

7 Number of rows, or item entries, of information that is to be displayed on each page

Current options are **10**, **25**, **50**, **100**.

8 The range of item entries currently displayed, with respect to the number of item entries to be displayed on each page, and the total number of entries that meet the filtering criterion defined with the filtering tools.

9 Previous and Next arrows

Tap to navigate the pages of events.

Plugins

The modular design of Link 6 makes it a robot with expanding capabilities. The expanding capabilities are added on as *plugins*.

Link 6 comes with two pre-installed plugins and cannot be uninstalled.

Arm plugin	The plugin handles all actions needed to control the *arm*. The configuration and settings of the arm plugin are not visible because they are not modifiable.
Industrial I/O plugin	The plugin handles all general *I/O* between the *controller* and on the *wrist*. That is, the plugin exchanges information using pure electrical signals through cables connected to some other piece of equipment.

Administrators can install new plugins to expand the capabilities of their robot. The plugins are on the Kinova website under Resources.

All pre-installed plugins start to run when the controller is switched on. However, they can be stopped so the settings can be modified. All plugins that are running are, by default, active.

Many plugins contain actions that the user can use when programming the robot with the visual programming interface in Kortex Web App. Each plugin may also contain parameters that can be modified in Kortex Web App from the corresponding plugin page.

All plugins have two pages: **Configurations** and **About**.

Related topics

Visual programming on page 212

Installing plugins on page 175

Overview of working with plugins on page 179

Plugins on page 173

Arm plugin

Link 6 comes with the *arm plugin* pre-installed.

There are no configurable parameters for the arm plugin. The **Configurations** portion of the **Arm plugin** page merely states that it does not have any modifiable configurations.

The **About** portion lists information specific to the arm, such as name, package name, version, type of installation, and a description of the plugin.

Industrial I/O Plugin

Link 6 comes with the Industrial I/O *plugin* pre-installed.

The Industrial I/O plugin has tiles in the visual programming interface in *Kortex Web App* to exchange information using pure electrical signals through cables connected to some other piece of equipment. The equipment is connected to the general purpose and high speed connectors in the *controller*.

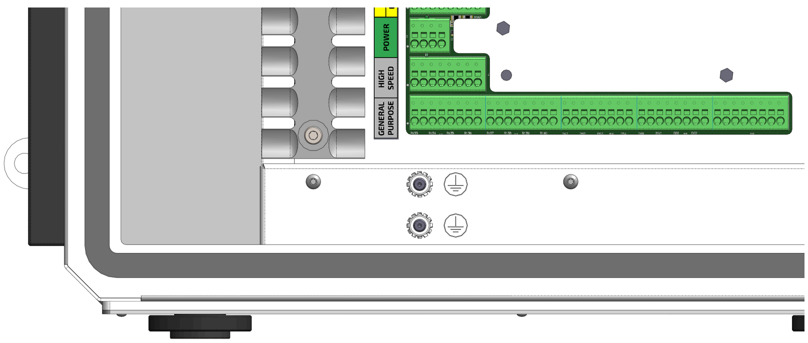


Figure 226: Industrial equipment connection on the controller

There are many configurable parameters for the Industrial I/O plugin. Access all parameters from **Systems** > **Plugins** > **Industrial I/O Plugin** > **Configurations**. The plugin has several pages of configurations, each page dedicated to one aspect of industrial I/Os.

- **Controller Digital Outputs**
- **Controller Digital Inputs**
- **Controller Analog**
- **Wrist Analog**
- **Wrist Digital**

Important: To configure any part of the Industrial I/O plugin, you must first tap **Stop** to stop running the plugin.

Related topics

Variables on page 215

Variable Management page on page 224

Controller analog on the Industrial I/O plugin page

All analog inputs for the *controller* can be configured on the **Controller Analog** pane of the **Industrial I/O** plugin page.

Analog has a voltage resolution of 0.1 V and a current resolution of 1 mA.

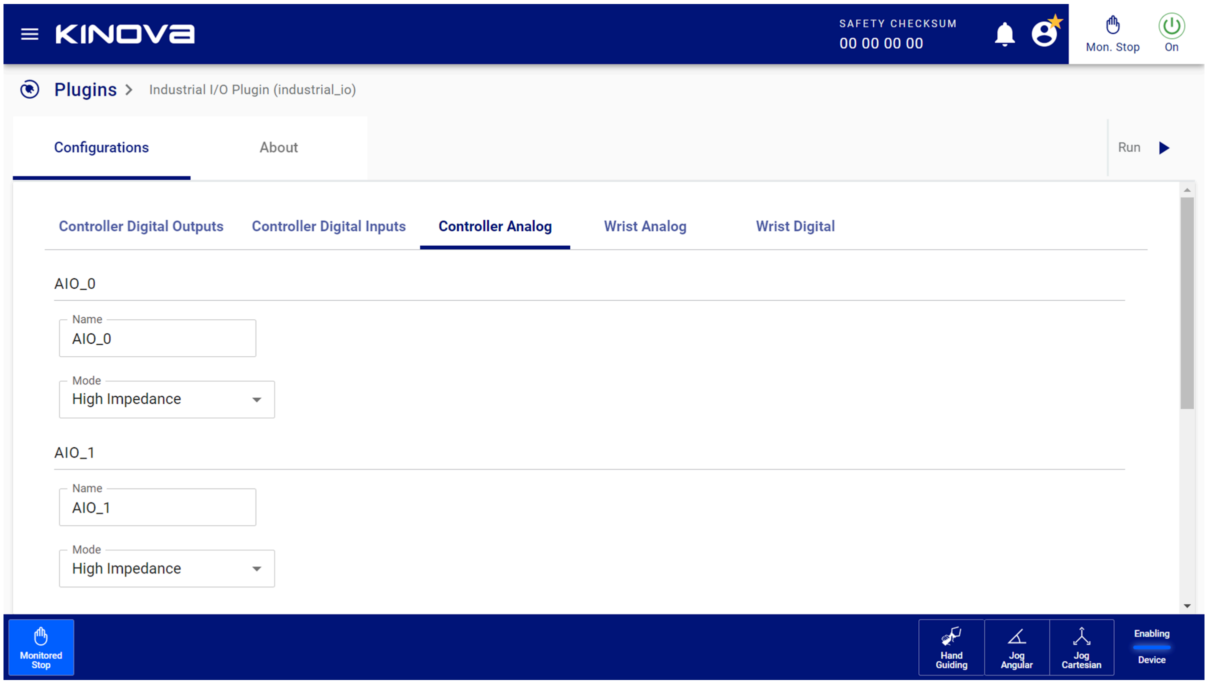


Figure 227: Controller Analog pane

Each analog I/O of the controller has its own smaller pane on the **Controller Analog** pane. Each one of **AIO_0** to **AIO_3** corresponds to each of the terminal connections on the controller.

Table 102: Controller analog configurations

Parameter	Description
Mode	Select a driver mode. • **High Impedance** (disable) • **Voltage Output** • **Current Output** • **Voltage Input** • **Current Input**
`Name`	Enter a meaningful name for that specific analog input or output for the controller.
	The name is visible in the visual programming interface.

Controller digital inputs on the Industrial I/O plugin page

All digital inputs for the *controller* must be configured on the **Controller Digital Inputs** pane of the **Industrial I/O** plugin page.

Each digital input for the controller is defined independently from the other digital input for the controller.

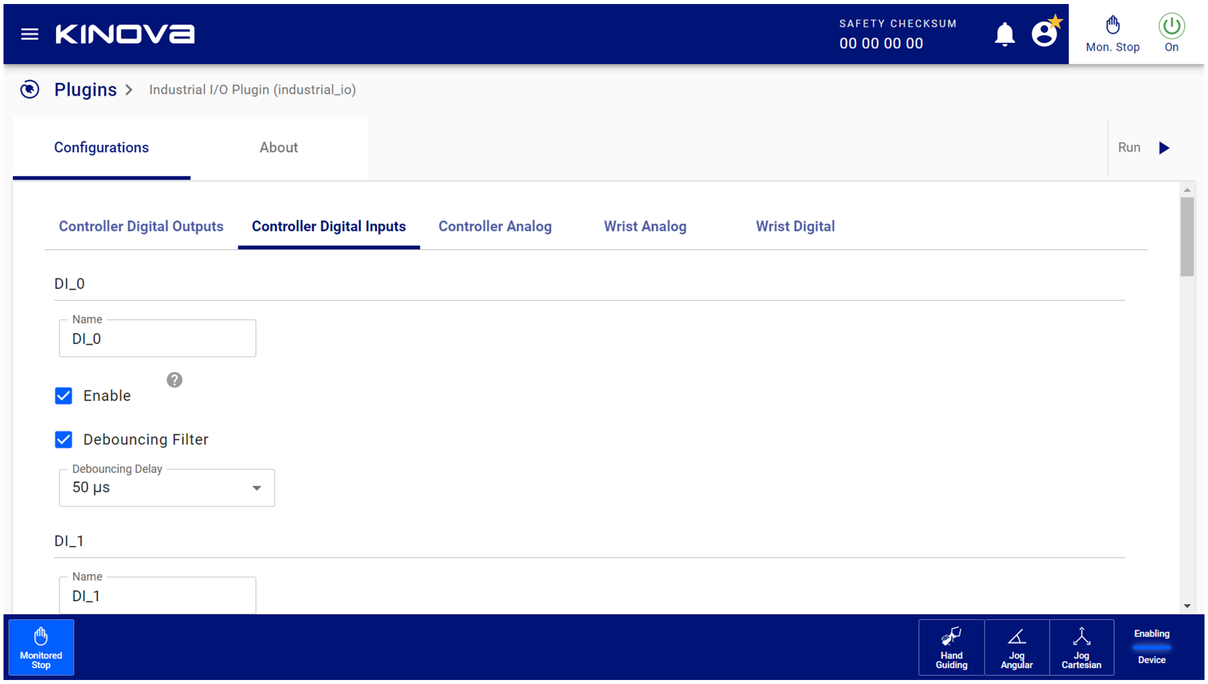


Figure 228: Controller Digital Inputs pane

Each digital I/O of the controller has its own smaller pane on the **Controller Digital Inputs** pane. Each one of **DI_0** to **DI_7** corresponds to each of the terminal connections on the controller.

Parameter	Description
`Name`	Enter a meaningful name for that specific digital input for the controller.
	The name is visible in the visual programming interface.
Enable	Select when you want to enable the input.
Debouncing Filter	Select to configure the *debouncing* delay.
Debouncing Delay	The debouncing delay is set in microseconds. Valid values are 50 µs, 100 µs, 400 µs, 800 µs, 1600 µs, 3200 µs, 12800 µs, and 20000 µs.

Related topics

General-purpose digital inputs on page 35

Controller electrical and communications interface on page 27

Controller digital outputs on the Industrial I/O plugin page

All digital outputs for the controller must be configured on the **Controller Digital Outputs** pane of the **Industrial I/O** plugin page.

Each digital output for the *controller* is defined independently from the other digital output for the controller.

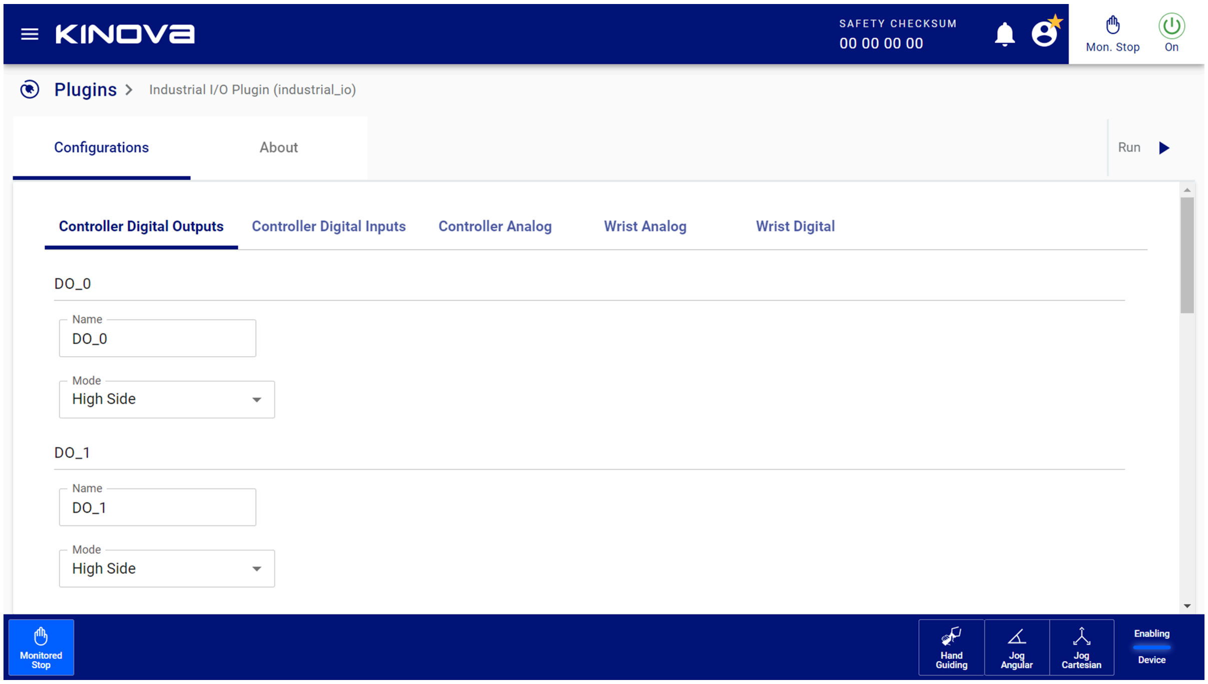


Figure 229: Controller Digital Outputs pane

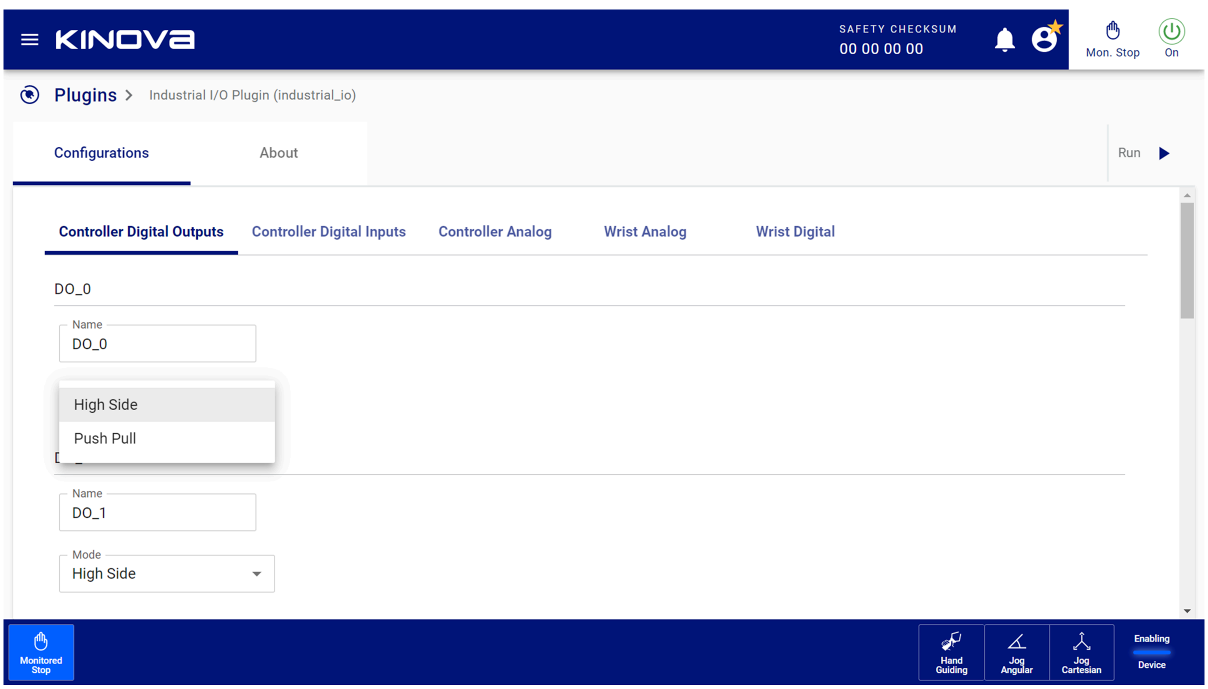


Figure 230: Controller Digital Outputs pane with Mode drop-down

Each digital I/O of the controller has its own smaller pane on the **Controller Digital Outputs** pane. Each one of **DO_0** to **DO_7** corresponds to each of the terminal connections on the controller.

Parameter	Description
`Name`	Enter a meaningful name for that specific digital output for the controller.
	The name is visible in the visual programming interface.
Mode	Select a driver mode. • **High Side** • **Push Pull**

Related topics

General-purpose digital outputs on page 36

Controller electrical and communications interface on page 27

Wrist analog on the Industrial I/O plugin page

All analog inputs for the *wrist* can be configured on the **Wrist Analog** pane of the **Industrial I/O** plugin page.

Analog has a voltage resolution of 0.1 V and a current resolution of 1 mA.

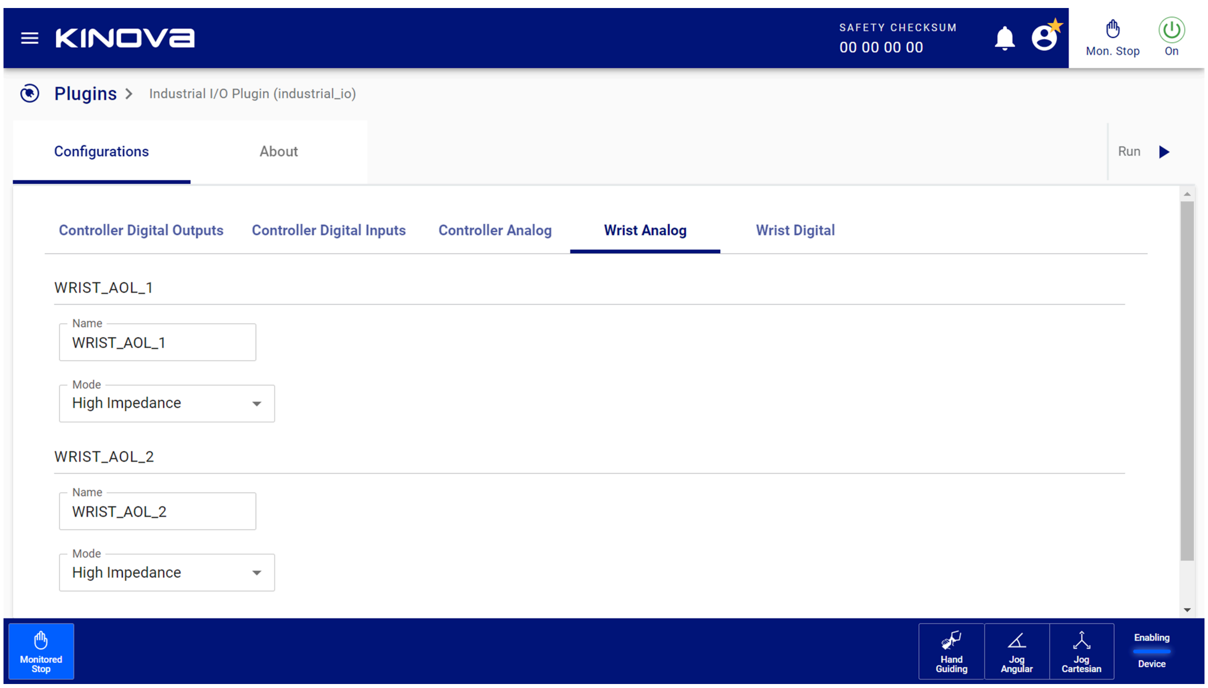


Figure 231: Wrist Analog pane

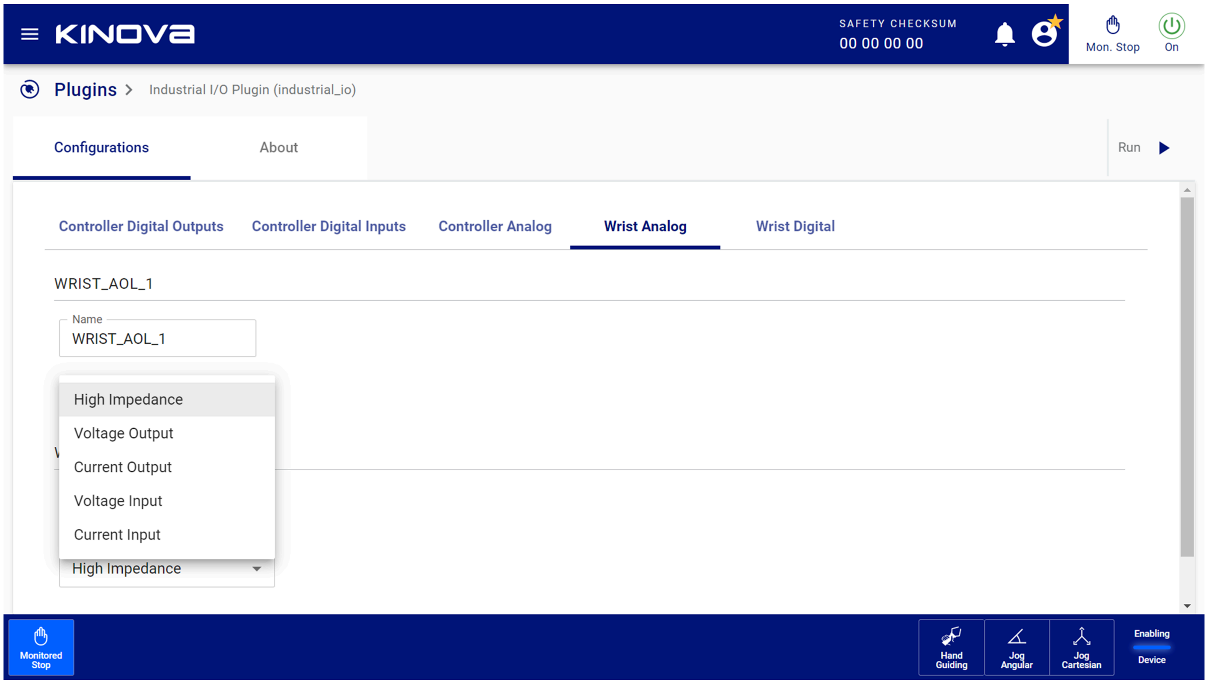


Figure 232: Wrist Analog pane with Mode drop-down

Each analog I/O of the wrist has its own smaller pane on the **Wrist Digital** pane. Each one of **WRIST_AOI_1** and **WRIST_AOI_2** corresponds to the two RS 485 connectors on the wrist.

Table 103: Wrist analog configurations

Parameter	Description
`Name`	Enter a meaningful name for that specific analog input or output for the controller.
	The name is visible in the visual programming interface.
Mode	Select a mode. • **High Impedance** (disabled) • **Voltage Output** • **Current Output** • **Voltage Input** • **Current Input**

Related topics

Electrical interface on page 18

Wrist digital on the Industrial I/O plugin page

All digital inputs and outputs for the *wrist* must be configured on the **Wrist Digital** pane of the **Industrial I/O** plugin page.

Each digital input and output of the wrist is defined independently from the other digital input and output of the wrist.

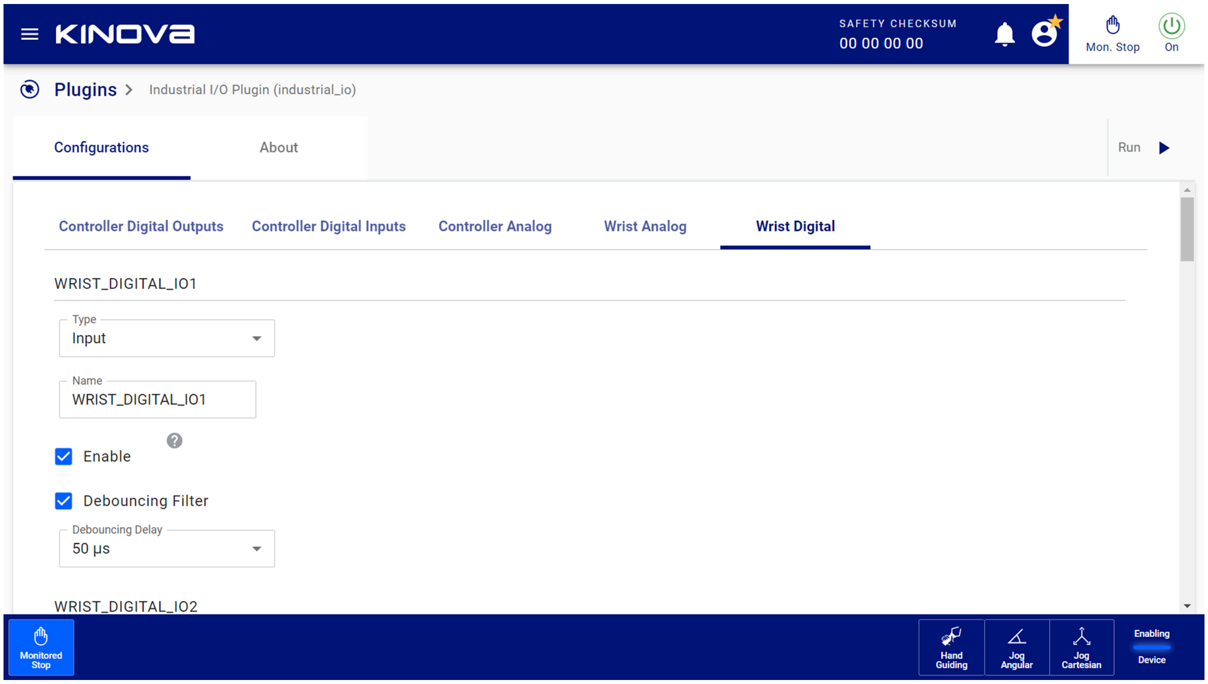


Figure 233: Wrist Digital pane when the pin is set an input

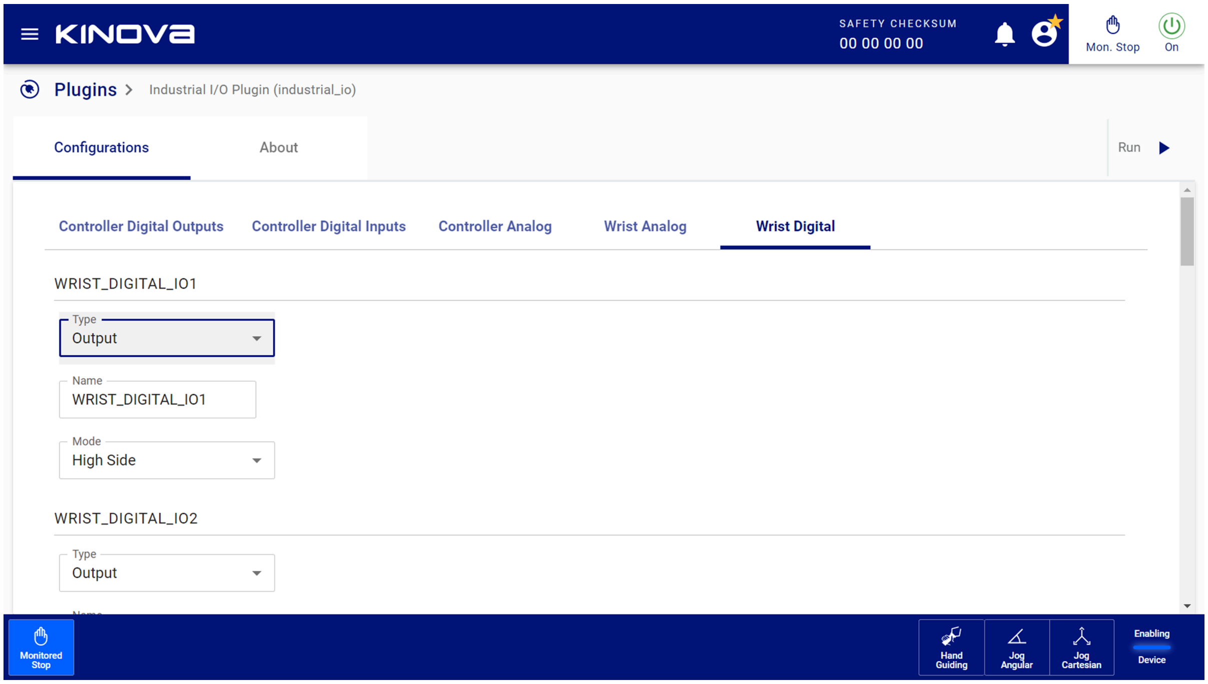


Figure 234: Wrist Digital pane when the pin is set an output

Each digital I/O of the wrist has its own smaller pane on the **Wrist Digital** pane. Each one of **WRIST_DIGITAL_IO1** to **WRIST_DIGITAL_IO8** corresponds to each of the eight pins on the M8 connector on the wrist.

Table 104: Wrist digital configurations

Parameter	Description
Type	Select either **Input** or **Output**.
`Name`	Enter a meaningful name for that specific analog input or output for the controller.
	The name is visible in the visual programming interface.
Parameters specific to **Input** pins	
Enable	Select when you want to enable the input.
Debouncing Filter	Select to configure the debouncing delay.
Debouncing Delay	The *debouncing* delay is set in microseconds. Valid values are 50 μs, 100 μs, 400 μs, 800 μs, 1600 μs, 3200 μs, 12800 μs, and 20000 μs.
Parameters specific to **Output** pins	

Parameter	Description
Mode	Select one of two modes. • **High Side** • **Push Pull**

Related topics

Wrist I/O on page 43

Accessories

Accessories can enhance the Link 6 experience, either by keeping the work environment aesthetically pleasing or by increasing the functional capabilities of the robot.

Accessories range from mounts to adapters to vision systems that simplify complex tasks.

In addition to the accessories by Kinova, there is a growing number of accessories that Link 6 can support.

Table 105: Partner accessories that are compatible with Link 6

Accessory	Plugin	Plugin version
OnRobot RG2 Gripper	OnRobot Plugin	1.1.0
Kinova Robotiq adapter	n/a	n/a
Robotiq 2F-85 Gripper	Robotiq Plugin	1.1.0
Robotiq 2F-140 Gripper	Robotiq Plugin	1.1.0
Robotiq Hand-E Gripper	Robotiq Plugin	1.1.0
Vision module	Vision Plugin	1.1.0

Related topics
Link 6 overview on page 1

External emergency stop station

The *E-stop* station is an optional component that provides *E-stop* functions in case the *teach pendant* is not used or additional *E-stop* locations are required. It is connected directly to the terminals in the *controller*.

Photo: Rockwell Automation

Figure 235: Rockwell Automation 800F-1YMQ3

Related topics
Safety I/O on page 87

Installing an external emergency stop station

Whenever you need to install an external *E-stop* station, be sure to read and follow all mounting instructions from the manufacturer.

Before you begin

Personnel:

- **Number of Workers:** 1

Supplies

- 1 x Emergency stop station
- 1 x Cable

Tools

- Ferrule (optional)

About this task

View or download publications at rok.auto/literature .

Procedure

1. Make sure the *controller* is switched off.
2. Connect the *E-stop* to the controller. Use the wire order on the E-stop connector.

Terminal (Controller side)	Wire color	Pin (Device side)
A1	Black	2
A2	Red	1
B1	White	4
B2	Yellow	3

3. Connect the 4-pin connector to the Emergency stop station.

Related topics
Safety I/O on page 87

External 3-position enabling device

The external 3-position *enabling device* is an optional component that provides enabling device functions in case the *teach pendant* is not used. It is connected directly to the terminals in the *controller*.

It is connected directly to the terminals in the controller.

Photo: IDEC

Figure 236: IDEC HE1G-21SM

Related topics
Safety I/O on page 87

Installing the external 3-position enabling device

Whenever you need to install an external 3-position *enabling device*, be sure to read and follow all mounting instructions from the manufacturer.

Before you begin

Personnel:

- **Number of Workers:** 1

Supplies

- 1 x 3-position enabling device
- 1 x Cable

Tools

- Ferrule (optional)

About this task

View or download publications at www.ca.idec.com.

Procedure

1. Make sure the *controller* is switched off.
2. Follow the wiring instructions in Industrial I/O Safety I/O inputs.
   Wiring for use of Channel 0

Terminal (Controller side)	Wire color	Pin (Device side)
AI_0	Black	2
24_o (Bank A)	Red	1
BI_0	White	4
24_o (Bank B)	Yellow	3

3. Set the Safety I/O to **Enabling Device**.

Related topics

Safety I/O on page 87

Maintenance

Preventive maintenance of Link 6 averts major issues that can arise, and keeps the robot and users safe.

There are three main things to do to give the robot a long and productive life.

- Clean the robot.
- Perform a visual inspection of the robot.
- Service the robot according to schedule.

To perform basics in maintenance, always have a good supply of consumables.

Maintenance is made of preventive measures and maintaining fasteners.

Consumables

Some parts of Link 6 can be removed and misplaced. Other parts may wear out. Always make sure you have a good supply of spare parts to lessen downtime.

The number of each part that should be kept on hand is only a recommendation of the minimum number of each part. It may be wise to have more than the recommended minimum quantity listed.

Note: Blank cells in the table Recommended spare parts list indicate there is no information available at the time of writing.

Table 106: Recommended spare parts list

Part	Manufacturer	Part number	Order number	Quantity on hand	Note
Plastic dust protection cap - M23	Phoenix Contact	RC-Z2469	1611797	1	
Plastic dust protection cap - M16	Sealcon			1	
Sealed cap for wrist Pogo pins	Kinova	KR21537		1	
Controller cable entry plate	Icotek	KEL-DPZ 24|25	50710	0	Replace if needed. The plate will not seal the controller if the cables are removed.

Part	Manufacturer	Part number	Order number	Quantity on hand	Note
Controller cable entry plate plug kit	Icotek	ST-B	42801	0	Seal the controller after removing the cables.
Fuse 3AG, Fact-Acting, 15A	Bussman	ACD-15-R		2	
Fan filter	Rübsamen & Herr	AM 115P		5	Quantity may vary, based on working environment.

Preventive maintenance

Always perform preventive maintenance within the time scheduled maintenance time frame to avoid downtime of the robot.

Different parts of Link 6 require maintenance checks on a daily, weekly, quarterly, or yearly schedule. Some preventive maintenance does not adhere to a specific schedule.

Daily maintenance

Always perform daily tasks daily to avoid downtime of the robot.

Table 107: Daily checks for preventive maintenance

Validation task	Corrective action
Check the emergency stop functions as expected.	Contact support at support@kinova.ca
Check all states of the 3-position enabling device function as expected if it is expected to be used.	Contact support at support@kinova.ca
Check all terminal connections are secured.	Redo connections.
Check the cables are in good condition with no rips, tears, or frays.	Contact support at support@kinova.ca
Check the robot joint caps are installed properly on the robot with no visible gaps and secure.	Contact support at support@kinova.ca
Check the tools and accessories are attached correctly and are not loose.	Secure the tools and accessories.
Check the joints are not loose.	Contact support at support@kinova.ca

Validation task	Corrective action
Check the bolts on the base are tightened according to the specifications and the base does not move when the robot moves.	Secure the base of the arm.
Check the casing of the controller is closed, with no openings along the edges or elsewhere.	Contact support at support@kinova.ca
Check the I/O panel on the controller is closed.	Close the panels on the controller.
Check the end caps are on all electrical connectors that are not in use.	Place end caps on all exposed electrical connectors.
Check the environment is the same as usual, with no additional objects in the area.	Remove objects that normally are not in the environment.

Weekly maintenance

Always perform weekly tasks weekly to avoid downtime of the robot.

Table 108: Weekly checks for preventive maintenance

Validation task	Corrective action
Check the air filter in the controller is clean.	Replace the air filter.

Quarterly maintenance

Always perform quarterly tasks every three months to avoid downtime of the robot.

Table 109: Quarterly checks for preventive maintenance

Validation task	Corrective action
Check the main power is secured to the controller.	Reattach the latch.
Check all connected safety inputs and outputs are secured and functioning.	Contact support at support@kinova.ca
Check the cable and connector of the Teach pendant are in good shape.	Contact support at support@kinova.ca
Torque check the base fastener.	Retorque.
Torque check the wrist fastener and end effector tooling.	Retorque.
Make sure the actuator covers are not loose.	Retorque.

Validation task	Corrective action
Check that the fan in the controller is rotating freely.	Contact support at support@kinova.ca

Yearly maintenance

Always perform yearly tasks every year to avoid downtime of the robot.

Table 110: Yearly checks for preventive maintenance

Validation task	Corrective action
Check Protective earth grounding resistance is less than 0.1 Ω to the controller.	Replace the cable.
Check Protective earth grounding resistance is less than 0.1 Ω to the arm.	Replace the cable.

Unscheduled maintenance

Even by following a maintenance schedule, unexpected maintenance is required.

Some of the unscheduled maintenance tasks are also scheduled maintenance tasks under certain circumstances. For instance, when there is a sudden power outage or when an accident happens with the robot, it is recommended to perform a visual inspection of the robot.

Table 111: Unscheduled maintenance tasks

Validation task	Corrective action
Cables are in good condition with no rips, tears, or frays.	Contact support at support@kinova.ca
Make sure the joint caps are on the robot. Also, make sure there are no visible gaps and that they are secured in place.	Contact support at support@kinova.ca
Tools and accessories are attached correctly and are not loose.	Secure the tools and accessories.
Joints are not loose.	Contact support at support@kinova.ca
The base is fixed correctly. In other words, the bolts on the base are tightened according to the specifications and the base dose not move when the robot moves.	Secure the base of the arm.

Validation task	Corrective action
The casing of the controller is closed, with no openings along the edges or elsewhere.	Contact support at support@kinova.ca
Panels on the controller are closed.	Close the panels on the controller.
End caps are on all electrical connectors that are not in use.	Place end caps on all exposed electrical connectors.
The environment is the same as usual, with no additional objects in the area.	Remove objects that normally are not in the environment.
Make sure the fuse is functioning.	Replace the fuse.

Fastener maintenance

To keep Link 6 safe, the fasteners require regular maintenance.

Table 112: Fastener locations and maintenance

Robot part	Location	Quantity	Fastener type	Recommended torque	Maximum thread engagement
Base	Base to table	4	M8×1.25	20 N·m	
	Protective earth screw	1	M5×0.8		
Arm	Actuator covers - Size 80	3	M4×0.7 - 8	0.5 N·m	
	Actuator covers - Size 110	4	M4x0.7 - 8	0.5 N·m	
Wrist	End effector tooling	4	M6×1.0	7-9 N·m	6.0 mm
	Bracket for GigE adapter (2x side)	2	M4×0.7		5.0 mm
Controller	Cable pass-through plate	4	M5×0.8		
	Protective earth stud	1	M5×0.8		
	Mounting bracket fasteners to controller	4	M6×1.0 - 10		

Robot part	Location	Quantity	Fastener type	Recommended torque	Maximum thread engagement
	Mounting bracket fasteners to wall	4	M8×1.25 - 14		
Teach pendant	Mounting bracket fasteners to wall	4			

Servicing

Some routine maintenance and repair work must be performed by a Kinova expert.

Not servicing Link 6 in a timely manner voids the Kinova warranty.

Contact Kinova support to learn how to have the robot serviced.

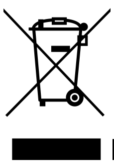
Disposal

The product contains parts that are deemed to be hazardous waste at the end of the life of the product.

After Link 6 has reached its life expectancy, you have two options.

- Continue using the device.
- Replace the robot and dispose of the old device appropriately.

If you wish to continue using Link 6 after its expected lifespan, Kinova highly recommends that you return the device temporarily to Kinova for routine servicing that includes inspection of the robot, as well as replacement and repair of any internal components that may be worn out.



If you wish to dispose of the robot, note that the robot contains materials that can be recycled. However, some of the materials are deemed hazardous. Specialized companies can dismantle the unit and sort out these materials. When you dispose of the unit, inform yourself about local regulations concerning waste management.

Troubleshooting

Knowing how to correct and recover from robotic issues is important to keeping the Link 6 working.

When Link 6 encounters a problem, it usually triggers a fault. When a fault is triggered, the information on the **Status** page should be enough to come to a diagnostic. There may be cases in which some users view certain behaviors as undesirable; however, from the point of view of the robot, the supposed undesirable behavior is expected. There is no diagnostic information available for expected behavior.

There are many examples of expected robot behavior that some users may deem undesirable listed. If you believe you have an issue with Link 6, check to see if you can find a similar issue. If you cannot find an issue similar to yours, please write to us at support@kinova.ca.

Issues related to hardware

Examples of hardware-related issues that you may have deal with the controller, the arm, the buttons on the wrist, the tool, and external equipment.

Why does the controller have no power?

Condition

The power button on the controller is switched on, but there is no power in the controller.

Cause

The power cable may not be connected to the controller or to a power source.

Remedy

Procedure

- Connect the power cable to the controller.
- Connect the other end of the power cable to the power source.

Why is the robot not powering on?

Condition

The controller is on, but the robot does not power on.

Cause

The cable attached to the arm of the robot is not connected to the controller.

Remedy

Connect the robot cable to the controller.

Why is the robot moving in joint Hand Guiding mode by itself?

Condition

The robot is in Joint Hand Guiding mode and it is moving without the application of external forces.

Cause

The masses or centers of mass of the tool or payload is not correct.

Remedy

Procedure

- Make sure the mass of the tool entered in Kortex Web App matches that of the tool.
- Make sure the center of mass of the tool entered in Kortex Web App matches that of the center of mass of the tool.
- Make sure the mass of the payload entered in Kortex Web App matches that of the payload.
- Make sure the center of mass of the payload entered in Kortex Web App matches that of the center of mass of the payload.

Why is the snapshot button not working?

Condition #1

When the waypoint capture, or snapshot, button is pressed, the waypoint does not appear in the program that is currently on the **Program** page.

Cause

The current context on the **Program** page is selecting a block that is not a waypoint block.

Remedy

Procedure

1. Look at the **Sequence editor** to see which block is selected.

2. Select the waypoint capture block that represents where you currently are in the program.
3. Press the waypoint capture button.

Cause #2

The snapshot button is held for too long.

Remedy for cause #2

Press the snapshot button quickly.

Why is my tool not working?

Condition

The tool connected to the robot is not working.

Cause

The plugin associated with the tool is not installed or is not activated.

Remedy

Procedure

1. Tap **Systems** > **Plugins** to access the **Plugins** page.
2. Install the plugin associated with the tool if it is not already installed.
3. Activate the plugin associated with the tool.
4. Reset or initialize the plugin associated with the tool if the plugin requires this action.

Related topics
Installing plugins on page 175

Emergency stop

Condition

An emergency stop signal is sent to the robot.

Cause

There are two possible causes.

- The E-stop button on the teach pendant, or elsewhere in the system, is pressed.
- A signal is detected on the input reserved for emergency stops.

Remedy

Procedure

1. Make sure nothing is interfering with the robot and that it is safe to proceed.
2. Remove the signal.
   Release the E-stop button, such as the one on the teach pendant, by turning the button.
3. Clear the fault.
4. Make sure the jumper wire configuration matches the installation.

Why is the wrist enabling device not enabling Hand Guiding mode?

Condition

The robot cannot be moved by hand. The wrist ring flashes green once each second.

Cause

The enabling device on the wrist is not pressed and held in its center position.

Remedy

Press the Wrist enabling device in the center position until the wrist ring illuminates a steady blue. Continue to hold the button in the center position while you move the arm.

Why is the controller hot?

Condition

The *controller* is hot or the diagnostics in *Kortex Web App* indicate the *SCU* or the *MCU* are overheating.

Cause

The temperature of the SCU core or MCU board is too high.

Remedy

Procedure

- Check the controller fans are spinning freely.
- Check that neither the controller fans nor the fan filter are clogged.

Why did my robot collide with the tool installed on its end effector?

Condition

While the robot was moving, it collided with the tool installed on its *end effector*.

Cause

The tool sphere protection zone is not configured to contain the entire tool.

Remedy

Procedure

1. Switch off the arm.
2. Navigate to **Safety** > **Protection Zones**.
3. Configure the Tool Sphere protection zone so that it contains the entire volume of your tool.

Why is my tool, connected to the M8 connector, not responding?

Condition

A tool is connected to an M8 connector on the wrist, but the tool is not responding to commands received from a program.

Cause

The connection to the M8 connector is incorrect.

Remedy

Validate that the correct M8 connector is used and that the connector is not damaged.

Why does my robot refuse to move in a particular direction?

Condition

While moving the robot in Manual Jog or Hand Guiding mode, the robot refuses to move in a certain direction; however, it does move in other directions.

Cause

One or multiple joints are reaching their position limits.

Remedy

Use Jog Angular mode to move the actuator back to its zero position.

Why is my teach pendant only displaying a console?

Condition

The teach pendant is connected and the controller is switched on, but only a console is displayed on the teach pendant.

Cause

The connector of the teach pendant is inserted incorrectly in the controller, which causes some of the pins to be misaligned.

Remedy

Disconnect and reconnect the teach pendant properly.

Why are my digital outputs not operating at 24 V?

Condition

Multiple elements are connected to consecutive digital outputs. Applying an output signal does not yield 24V.

Cause

The load on a terminal block, composed of 4 I/Os, is too high.

Remedy

Distribute the electrical load over more terminal blocks.

Issues related to safety elements

Examples of safety-related issues that you may have deal with the safety configurations and safety I/O connections.

Why can I not edit any safety configurations?

Condition

None of the parameters on any of the **Safety** pages are modifiable.

Note: This condition only happens when the user is an *operator* and is not an *admin user*.

Cause

The user is an operator. Only admin users can modify safety configurations.

Remedy

Procedure

- Tap > **ELEVATE PERMISSIONS** and enter the admin password.
- Contact an admin user to discuss which safeties may need to be modified.

Why is the external safety device not working?

Condition

The external device attached to the robot is not working. The robot is in a *stop category (SC)* 0, where there is no electrical power in the robot.

Cause

The device is not configured correctly on the safety I/O channel.

Remedy

Make sure the safety I/O channel associated with the external device is configured correctly.

Select the correct safety type from the safety type drop-down on the **Safety I/Os** page in *Kortex Web App* for the channel where the device is connected.

Why is the robot stuck in Reduced mode?

Condition

The robot is moving at Reduced speed. All **Safety** pages indicate the robot should be moving at Normal speeds.

Cause

The robot is receiving a signal to operate at Reduced speed.

Remedy

Make sure no safety I/O channel is sending a signal to the robot to operate at Reduced speed.

Why are my digital outputs not operating at 24 V?

Condition

When operating in joint Hand Guiding mode, the robot triggers a fault for exceeding speed limits.

Cause

When the arm is driven in joint Hand Guiding mode near the outer edge of its workspace, small angular velocities can yield large *Cartesian* velocities on the *TCP* and trigger speed limit faults.

Remedy

Move the *TCP* closer to the arm when you move the first two joints of the robot in Hand Guiding mode.

Issues related to modes of operation

Examples of issues with modes of operation that you may have include modes being disabled and modes that seem stuck.

Why is Manual mode disabled?

Condition

The arrows of the virtual joystick of Jog Angular and Jog Cartesian in *Kortex Web App* are grayed out.

The *enabling device* in Kortex Web App illuminates white.

Cause

The enabling device is not held in its center position.

Remedy

Procedure

- Make sure you are not pressing the enabling device all the way to the bottom.
- Make sure you are pressing the enabling device in its middle position.

Why can the robot not exit Recovery mode?

Condition

The robot cannot exit Recovery mode. The wrist ring flashes yellow two times each second.

Remember: As long as the robot is in Recovery mode, all safety position-related limits are ignored and all joint speed safety limits are limited to a maximum speed of 30 °/s.

Cause

There are two situations where the robot remains in recovery mode.

- The initial cause of the fault is still in effect.
- After clearing the initial fault, another fault is triggered in the current configuration of the robot.

Remedy

Procedure

- Make sure the robot is far from its position limits.
- Make sure the robot is outside of the enabled protection zones.
  a) Make sure the speed limit of each of the joints is set to 30 °/s.
  b) Tap **Safety** > **Protection Zones**.
  c) Visualize the current pose of the robot relative to the protection zones.
  d) Move the robot to a safe position using either Hand Guiding or Manual Jog mode.

Related topics

Protection zones on page 77

Recovery mode on page 122

State: Fault on page 113

Why is my robot not moving even if it is in an operating mode where it should be possible?

Condition

When operating in Automatic, Hold-to-Run, or Manual mode, the robot does not perform any *Cartesian* command. A notification is displayed on the teach pendant, but the robot does not enter a fault state.

Cause

The robot is near a singularity.

Remedy

Use any angular control mode to move the robot away from the singularity.

Issues related to plugins

Examples of issues you may have with plugins deal with plugin installation, configuration, and activation.

Why can I not configure Industrial I/Os?

Condition

The fields on the **Configurations** page of the **Industrial I/O** plugin page are not selectable.

Cause

The Industrial I/Os need to be switched on before they can be configured.

Remedy

Make sure the Industrial I/Os are receiving power.

Related topics

Industrial I/O Plugin on page 324

Industrial I/O panel overview on page 32

Issues related to programs and programming

Examples of issues related to programs and programming deal with variables, action blocks, and programs.

Why can I not select a variable in the Select Variable page?

Condition

A variable in the **Select Variable** page is grayed out and cannot be selected.

Cause

The format of the desired variable does not match the format expected by the field.

Remedy

Make sure the variable you are trying to select has the same type, schema, and unit as the field you are trying to fill.

Why are tiles missing in the visual programming interface?

Condition

The tiles cannot be found anywhere under the **Actions** menu on the **Program** page.

Cause of the trigger #1

The plugin associated with the missing tiles is not installed or is not activated.

Remedy for trigger #1

Procedure

1. Tap **Systems** > **Plugins** to see if the plugin associated with the missing actions is installed.

Option	Description
Plugin is not installed	Install the plugin.

2. Tap the plugin to access its **Configurations** page.

Option	Description
Run ▸ **is displayed**	Plugin is not active. Tap **Run** to activate the plugin.
Stop ■ **is displayed**	Plugin is active.

3. Return to the **Program** page of the program where the issue occurred.
   The plugin is displayed on the page with all its available actions.

Cause of the trigger #2

A fault occurs in the arm.

Remedy for trigger #2

Procedure

1. Restart the program.
2. Refresh the page.

Related topics
Installing plugins on page 175

Why can I not find a program in the list of programs?

Condition

A specific program is not found on the **Programs** page in Kortex Web App.

Note: This condition only happens when the user is an *operator* and is not an *admin user*.

Cause

The user is an operator and the admin user has not validated the program.

Remedy

The admin user must validate the program. After it is validated, the program appears in the list of programs for all users, including operators.

Why do some of the tiles of an imported program show a question mark?

Condition

A program is imported. One or more tiles on the **Program** page displays a question mark.

Cause

The imported program contains tiles from plugins that currently are not installed or activated on the controller.

Procedure

1. Remove the program from Kortex Web App.
2. Install the missing plugins.
3. Activate the plugins.
4. Re-import the program.

Why are the Industrial I/O tiles not loading?

Condition

When trying to run a program that uses tiles from the Industrial I/O plugin, the Industrial I/O tile configuration does not load properly.

Cause

There is a bug that happens occasionally when the Industrial I/Os on the controller are connected to power when controller is already switched on.

Remedy

Force the Industrial I/O plugin to reinitialize by dragging any Industrial I/O tile anywhere in the program. Optionally, delete the tile and after the plugin re-initializes, add it back to the program.

Why is my program running, but not making progress?

Condition

A running program does not make any progress, but it is not interrupted and there are no faults being triggered.

Cause

A `Wait` or `Wait for Input` tile is not configured correctly.

Remedy

Validate the configuration of the `Wait` or `Wait for Input` tile. Their duration parameters are in seconds.

Why is my robot extremely slow?

Condition

A program is running a waypoint trajectory at an abnormally low speed.

Cause #1

The safety limits of the robot are configured to enforce low velocities.

Remedy #1

Switch off the arm and increase the velocity limits provided the risk assessment is not impacted by the change.

Cause #2

Constraints applied to the waypoint are forcing the robot to operate at lower speeds to make the trajectory feasible.

Remedy #2

Procedure

- Validate that the local speed and acceleration limits of each waypoint are correct.
- Validate that the global waypoint trajectory speed and acceleration limits are correct.

Cause #3

When blending during the trajectory requires accelerations that are too fast, speed is automatically reduced so that it is feasible to travel the trajectory.

Remedy #3

Perform one or more of the actions to fix the situation.

- Increase the acceleration limits.
- Increase the size of the blending radius.
- Activate the **Optimal Blending** solution for the tile.

Issues related to Kortex Web App

Examples of issues related to *Kortex Web App* include such things as software and configuring problems.

Why is the robot not going to the programmed position?

Condition

The robot is programmed to go to specific locations. The robot is not moving to the expected locations.

Cause of the trigger #1

The tool is not configured correctly.

Remedy for trigger #1

Procedure

1. Tap **Systems** > **Tools** to access the **Tools** page.
2. Make sure the tool is added in the correct stacking position.
3. Make sure all configurable parameters of the tool have the correct values.

Cause of the trigger #2

The robot is not calibrated.

Remedy for trigger #2

Procedure

1. Tap **Systems** > **Robot** > **Arm**.
2. Expand the **Calibration** pane.
3. Verify that the calibration status is `Not Calibrated`.
4. Tap **Import** in the **Import calibration file** panel to upload the calibration file to the robot.

    Important: Contact support@kinova.ca if the administrator does not have a backup of the calibration file.

5. Verify that the calibration status is `Calibrated`

Related topics

Adding tools on page 182

Configuring tools on page 180

Configuring the arm for robot controls on page 187

Why is the computer not connecting to Kortex Web App?

Condition

The controller is on, but *Kortex Web App* is not displayed on the computer.

Cause #1

The Ethernet cable is not connected to the computer or the controller.

Remedy #1

Make sure one end of the Ethernet cable is connected to the controller and that the other end is connected to the computer.

Cause #2

The netmask of the Ethernet interface on the computer does not match the IP address of the controller.

Remedy #2

Make sure the netmask of your Ethernet interface is appropriate for the range of the two or more devices that are communicating with the robot. The default IP address of the controller is `192.168.1.10`.

Cause #3

A user is logged on with the teach pendant.

Remedy #3

Make the user who is using the teach pendant log out.

Why is Kortex Web App not responding on my computer?

Condition

Kortex Web App is running on a computer, but there is no response from interacting with the application. The application behaves as if it were frozen.

Cause

The computer has memory performance issues.

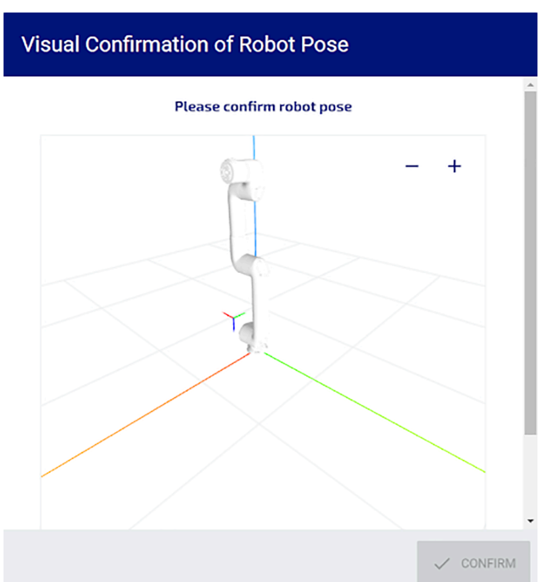
Remedy

Refresh the browser.

Why is there a pop-up requesting visual confirmation of the robot pose after the robot reboots?

Condition

A pop up appears after boot. It requests confirmation of pose of the robot.



Cause

After the robot is transported or when the robot is backdriven while it is switched off, a mismatch occurs between the last known position in memory and the current position.

Remedy

If the position of robot matches the visual representation, tap **CONFIRM**.

If the position of robot does not match the visual representation, do not confirm a mismatched pose. Contact support at support@kinova.ca for investigation.

Why do the +/- buttons on the wrist not work with my tool?

Condition

A plugin that is supposed to support controlling a tool with the end effector action (+/-) button on the wrist is running, but pressing the end effector action (+/-) button does not operate the tool.

Cause

The tool is not configured as active in the **Tools** page.

Remedy

Procedure

1. Navigate to **Systems** > **Tools**.
2. Drag the desired tool from the **Inactive (drag & drop)** list to the top of the **Active** tool list.

   Note: The tool must be listed in the **Inactive (drag & drop)**" list before the *flange*.

Glossary

A

AC

type of electrical current

alternating current (AC)

actuator

a device that is responsible for robot motion

administrator

a person who configures and uses the robot

synonym: admin user, admin

angular

refers to an object with joints and its rotational capabilities

arm

the part of the robot that is connected to the base and that moves

API

a software interface between the Kortex software and your own software

Application Programming Interface(API) />)

AWG

a standardized wire gauge system; unit of measurement

American Wire Gauge (AWG)

B

backdrive

to move the robot manually without using electrical power

base

the bottom-most part of the arm of the robot that is placed on a surface for installation; support of the robot

BCD

a way to represent each decimal digit in its binary equivalent using no more than 4 binary digits

Binary Coded Decimal (BCD)

bending radius

the radius of the inner-most curvature when bending an object, such as a cable; an object with a smaller radius have higher flexibility to bend before potentially becoming damaged

boot

to switch on, or start

C

camlock

a type of fastener

Cartesian

linear movement along the x, y, and z coordinate lines

collaborative

the ability for an integrated robot to work in a shared space, or in close proximity of people and other equipment, based on its integration

controller

the part of the robot from where the software operates

CPU

the portion of the computer that retrieves and performs instructions

central processing unit (CPU)

CRC

code that detects errors in changes to data

Cyclic redundancy check (CRC)

CRC

cUL

tested to Canadian standards by Underwriters' Laboratories

D

DC

type of electrical current

direct current (DC)

debounce

the act of ensuring one and only one signal is used when opening or closing a contact in an electronic device

DIN

a German standardization organization that is part of the ISO standards body

Deutsches Institut für Normung (DIN)

synonym: German Institute for Standardization

DoF

the number of variables needed to define the configuration of a robot, particularly for all the directions in which the motion can occur

degrees of freedom (DoF)

E

EFT

a series of quick, high frequency pulses often caused by a lack of electricity flowing through the circuitry

Electrical Fast Transient(EFT)

synonym: fast transient, burst

EFT/B

refer to *EFT*

elbow

the location of the 3rd actuator in the robot arm

EMI

the operation of an electronic device is disrupted by another electronic device that is nearby

electromagnetic interference (EMI)

enabling device

a device used to send a signal to the robot that the user is going to move the robot either in Manual Jog mode or Hand Guiding mode.

synonym: pendant enabling device, wrist enabling device

end effector

the end of the arm of a robot; also, a tool or *flange* connected to the end of the arm of the robot

ESD

the sudden flow of electricity between two objects that have different charges

electrostatic discharge (ESD)

E-stop

a sudden removal of all electrical power to the entire robot

(E-stop)

synonym: SF01

F

firmware

in computers, it is low-level software that is programmed into read-only memory and is used directly by hardware

flange

a rim used for attaching another object

G

GPU

originally created to enhance graphic rendering, it also has proven itself useful in many other applications, such as artificial intelligence

graphics processing unit (GPU)

GUI

a way of interacting with a program of a computer that involves windows, dialogs, icons, and menus

Graphical User Interface (GUI)

synonym: human machine interface (HMI)

H

hamburger menu

three vertical lines that, when tapped, expose a menu of actions or tasks from which the user can choose; usually used for main menus

synonym: menu, hamburger, main menu

hazard identification

the act of identifying all hazards that may cause harm

HMI

a way of interacting with a machine with such items as push buttons connected to a relay or a menu on a touch screen

human machine interface (HMI)

I

I/O

input output

input output (I/O)

IEC

an organization that creates international standards for electrotechnology

International Electrotechnical Commission (IEC)

industrial I/O panel

a panel on the *controller* where the *integrator* connects external devices

integrator

a person who incorporates the robot in a larger system or environment

J

joint position

the rotational position of a joint, measured in °

synonym: SF03

joint speed

the velocity at which the joint rotates, measured in °/s

synonym: SF04

K

kebab menu

three vertical dots that, when tapped, expose a menu of actions or tasks from which the user can choose; usually used for contextual menus

⋮

synonym: contextual menu, menu options, options, menu

Kortex Web App

the *GUI* that runs on a web server on the *controller* and is accessed on the *teach pendant* or on an external computer

synonym: Web App

L

Lockout Tagout

safety procedure where power is removed from the machinery, one or more locks is placed on the power supplies, and a tag is added to the lock with the name or initials of the worker who locked out the machinery

Lockout Tagout (LOTO)

LSB

the right-most bit in a binary number

Least Significant Bit (LSB)

M

MAC

unique identifier, similar to a room number inside a hotel

MCU

a device used in the actuators to read sensors and report sensor information

Main Control Unit (MCU)

microcontroller

in computers, a miniature computer integrated on a computer chip; often has firmware installed on it

mode

describes how a function is performed

Monitored stop

the robot is not moving, has power, and is actively monitoring that it is not doing anything

MPU

a device in the controller that manages external communication and inverse kinematics, sends commands to each actuator, and collects information from all other devices

Main Processing Unit (MPU)

N

namespace

a way in computer science of grouping together all objects that work together such that each object has a unique name

NC

electricity flows through a contact in its normal state

Normally Closed (NC)

non-blocking

the ability for a robotic program to proceed to its next step even when the current, non-blocking, action is not finished

non-collaborative

the inability for the robot to work in a shared space, or in close proximity of people and other equipment; the need for the robot to be in an enclosed, self-contained area to operate safely

NPN

the meanings of binary signal levels are interchanged such that high voltage is 0 and low voltage is 1

Negative logic (NPN)

O

operator

a person who uses, but who does not configure, the robot

OSSD

a device that outputs a signal to shut down the machine; used as part of a safety system of a machine

Output Signal Switching Device (OSSD)

P

payload

the load the robot carries excluding the tool that the robot carries

PCB

a structure that is made up of conductive and insulating layers

printed circuit board (PCB)

PE

a system that connects a power supply with the ground

protective earth (PE)

synonym: ground

pendant enabling device

enabling device located on a teach pendant, usually as a button

enabling device on page 365

synonym: enabling device, teach pendant enabling device

pendant

a device used to configure and control the movements of a robot

synonym: teach pendant

pinch point

an area in which a body part can get in between a stationary part and a moving part of the machinery

PLC

an industrial computer used in controlling manufacturing processes

programmable logic controller (PLC)

PLd

Performance Level D

Performance Level d (PLd)

plugin

software that is added to an pre-existing base software so as to give the application additional capabilities

program sequence

the sequence of actions that a program must use to complete its task

program sequence

synonym: program

protection zone

an area where the robot cannot enter

synonym: SF06

protective stop

a stop in which the robot stops, but there still is power in the robot

synonym: SF02

R

RAM

the memory of the computer that stores data for the *CPU* to use

random access memory (RAM)

reboot

to restart the computer

risk analysis

a study of the potential dangers in using something, often equipment

risk control

the plan to avoid or lessen the risk involved

RJ45

a jack with 8 pins

robot

refers to the computer or teach pendant, controller, arm, and end end effector; alternatively, refers to just the arm and the tools attached to it

robot cell

a set of one or more machines that work in conjunction with one or more programmable robots

robot system

an entire automated system consisting entirely of robots or automated equipment

role

the position held that is capable of performing a specific set of tasks

RS-485

a standard that defines electrical traits that are used in serial communication systems

RTU

an electronic device controlled by microprocessors that interfaces with other systems and is used to send and receive data

remote terminal unit (RTU)

S

safety mode

one of two speeds that are used when determining the maximum speed at which the robot can operate

SC2

stop category 2

synonym: *protective stop*

scope

in computer science, the area of code that is valid within the confines of another area of code

SCU

a processor in the controller that processes all safety functions

Safety Control Unit (SCU)

state

describes the overall condition of the system, in our case Link 6

stop category

a type of safety control function

stop category (SC)

SWU

a file format, used by Kinova, to package firmware updates

firmware update package (SWU)

T

TCP

the position and orientation of the tool

Tool Center Point (TCP)

TCP/IP

set of communication protocols used in computer networks and in the Internet

Transmission Control Protocol/Internet Protocol (TCP/IP)

teach pendant

a device used to configure and control the movements of a robot

synonym: pendant

teach pendant enabling device

enabling device located on a teach pendant, usually as a button

enabling device on page 365

synonym: enabling device, pendant enabling device

U

UL

Underwriters Laboratories Inc., an organization dedicated to testing and creating standards for electrical products

Underwriters Laboratories (UL)

USB

industry computer bus standard that makes it easier to transfer data between computers

Universal Serial Bus (USB)

W

wrist

the part found at the end of the robot arm to which end effectors may be attached

wrist enabling device

enabling device located on the wrist of the robot, usually as a button

enabling device on page 365

synonym: enabling device

wrist I/O

the area of connector on the *wrist* where the *integrator* connects external devices

wrist ring

the LED ring found around the wrist

TOGETHER IN ROBOTICS

PROUDLY MADE IN

QUEBEC, CANADA

EN-UG-020-Link-6-r2.0-202301