



KINOVA™ *Ultra lightweight robotic arm*

User Guide

Contents

| | |
|---|-----------|
| About this document..... | 6 |
| Symbols, definitions, and acronyms..... | 7 |
| Warranty..... | 8 |
| Warnings..... | 9 |
| Disclaimer..... | 10 |
| General Information..... | 11 |
| Robotic arm Configurations..... | 12 |
| 4 DOF Components..... | 15 |
| 6 DOF Curved Wrist Components..... | 16 |
| 6 DOF Spherical Wrist Components..... | 17 |
| 7 DOF Spherical Wrist Components..... | 18 |
| External Connectors..... | 19 |
| Comparing robotic arm configurations..... | 20 |
| Markings and Labels..... | 22 |
| Installation of the robotic arm..... | 23 |
| Mechanical mounting of the robotic arm..... | 24 |
| Electrical integration..... | 26 |
| Integrating a new end effector (optional)..... | 27 |
| Control integration..... | 29 |
| Control modes overview..... | 30 |

| | |
|--|-----------|
| Control peripheral options overview..... | 31 |
| Kinova joystick controller..... | 32 |
| Joystick movements and modes..... | 33 |
| Operating principles and Cartesian mode..... | 34 |
| Home / Retracted positions..... | 35 |
| Operating the arm via joystick..... | 36 |
| Joystick control quick start..... | 37 |
| Default joystick motion settings - Cartesian three-axis mode..... | 39 |
| Default joystick motion settings - Cartesian two-axis mode..... | 40 |
| Controlling the arm in Angular mode..... | 41 |
| Kinova Joystick LED feedback..... | 43 |
| Kinova joystick Blue LEDs feedback..... | 44 |
| Kinova joystick Green LEDs feedback..... | 45 |
| Kinova joystick Red LEDs feedback..... | 46 |
| Controlling the arm using Kinova software..... | 47 |
| Controlling the arm using the Kinova API..... | 49 |
| Singularity Avoidance..... | 50 |
| Self-collisions auto-avoidance..... | 53 |
| 7 DOF Spherical Null space motion..... | 54 |
| Protection zones..... | 55 |
| Rotating frame / Fixed frame..... | 56 |

| | |
|---|-----------|
| Usable workspace..... | 57 |
| Admittance control..... | 58 |
| Torque Control..... | 60 |
| Improving robot behavior in Torque mode..... | 61 |
| Optimal mode code example..... | 62 |
| Important considerations for setting Optimal mode..... | 64 |
| Specific Utilization Limitations..... | 65 |
| Recommended maximum actuators utilization..... | 66 |
| Software position limitations of actuators..... | 67 |
| Software position limitations of fingers..... | 69 |
| Advanced Configuration..... | 70 |
| Kinematics Parameters..... | 71 |
| Basic geometric parameters - 4 DOF..... | 72 |
| Basic geometric parameters - 6 DOF curved wrist..... | 74 |
| Alternate geometric parameters - 6 DOF curved wrist..... | 77 |
| Basic geometric parameters - 6 DOF spherical wrist..... | 79 |
| Basic geometric parameters - 7 DOF spherical wrist..... | 81 |
| Classic DH parameters - 4 DOF..... | 83 |
| Classic DH parameters - 6 DOF curved wrist..... | 84 |
| Classic DH parameters - 6 DOF spherical..... | 86 |
| Classic DH parameters - 7 DOF spherical..... | 88 |

| | |
|--|------------|
| Directions of joints in angular space..... | 90 |
| Advanced sensors information..... | 91 |
| Normal use definition..... | 92 |
| Electromagnetic interference from radio wave sources..... | 93 |
| Maintenance and Disposal..... | 94 |
| Packing Materials..... | 95 |
| Troubleshooting / FAQs..... | 96 |
| Contacting support..... | 98 |
| 3-axis mode joystick controls reminder..... | 99 |
| 2-axis mode joystick controls reminder..... | 100 |

About this document

 Read all instructions before using this product.

 Keep these instructions for future reference.

 Read all warnings on the product and in this guide.

 Follow all instructions.

This document contains information regarding product setup and the operation. It is intended for:

- Field service, customer support and sales employees of authorized Kinova distributors
- Kinova product end users

Symbols, definitions, and acronyms



Important information regarding the safety related to the product and the user.



Tip on the maintenance, operation and manipulation of Kinova's products.



Refer to accompanying documents.



Direct current.



Alternating current.



Operating temperature range.



Compliance with WEEE2 directive.



Compliance with ROHS3 directive.



Type BF Applied Part device.

Warranty

This section describes the Kinova warranty terms.

Subject to the terms of this clause, Kinova warrants to End User that the Products are free of defects in materials and workmanship that materially affect their performance for a period of two (2) years from the date Kinova ships the Products to the End User ("Delivery Date").

Kinova agrees to repair or replace (at Kinova's option) all Products which fail to conform to the relevant warranty provided that:

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

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

Allegedly defective Products returned to Kinova in accordance with this contract will, if found by Kinova on examination not to be defective, be returned to End User and Kinova may a 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.

Warnings

This section summaries important warnings and cautions related to use of the robotic arm.

 It is not recommended to use the arm under heavy rain or snow.

 Never use the HOME/RETRACTED function when carrying liquid. The HOME position is preset and the wrist may rotate and drop the liquid.

 Do not manipulate cutting, very sharp or any dangerous tools or objects with the arm.

 When the power is turned off, the arm will fall down and may cause damage to itself, depending on its position at the time of disconnection. Be sure to support its wrist before turning the power off.

 Do not force the fingers beyond their maximal opening. This could damage some internal components.

 Do not immerse any part of the arm under water or snow.

 When lifting weight near the maximum load and reach, if the red lights of the controller blink, put down the object in the gripper, bring back the arm to HOME or RETRACTED position and wait until the warning goes away before using it again.

Disclaimer

KINOVA™ and Kinova's logo are trademarks of Kinova Inc., herein referred to as Kinova. All other brand and product names are trademarks or registered trademarks of their respective corporations.

The mention of any product does not constitute an endorsement by Kinova. This manual is furnished under a lease agreement and may only be copied or used within accordance with the terms of such lease agreement. Except as permitted by such lease agreement, no part of this publication may be reproduced, stored in any retrieval system, or transmitted, in any form or by any means, electronic, mechanical, recording, or otherwise, without prior written consent of Kinova.

The content of this manual is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Kinova. Kinova assumes no responsibility or liability for any errors or inaccuracies that may appear in this document.

Changes are periodically made to the information herein; these changes will be incorporated into new editions of this publication. Kinova may make improvements and/or changes in the products and/or software programs described in this publication at any time.

Address any questions or comments concerning this document, the information it contains or the product it describes to:

support@kinovarobotics.com

Kinova may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligations to you.

General Information

The KINOVA™ Ultra lightweight robotic arm is a light-weight robot composed of four, six, or seven inter-linked segments. Through the controller or through a computer, the user can move the robot in three-dimensional space and grasp or release objects with the gripper (if a gripper is installed).

 Do not modify this equipment without authorization of the manufacturer.

 The Normal Use Definition contains some fundamental information to the proper operation of the robotic arm. arm.

 It is not recommended to use the arm under heavy rain or snow.

Robotic arm Configurations

This section describes the different configurations available for the robotic arms.

For Innovation applications, the robotic arm is available in four configurations:

- 4 degrees of freedom (DOF)
- 6 degrees of freedom (DOF) curved wrist
- 6 degrees of freedom (DOF) spherical wrist
- 7 degrees of freedom (DOF) spherical wrist



Figure 1: 4DOF



Figure 2: 6DOF curved wrist



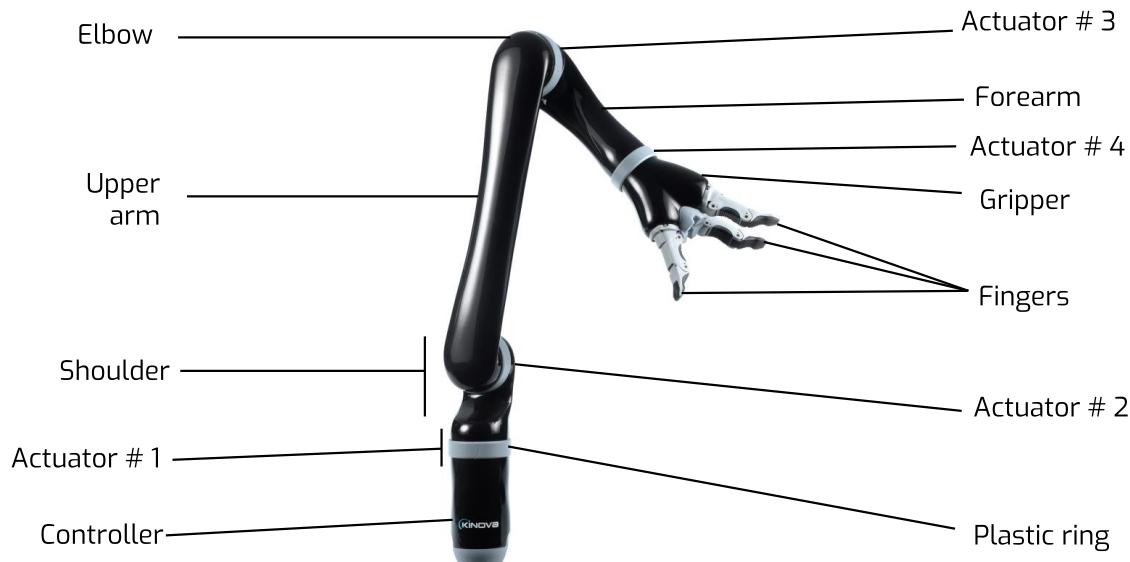
Figure 3: JACO 6DOF spherical wrist



Figure 4: 7DOF spherical wrist

4 DOF Components

This section shows the components of the 4 DOF robotic arm.



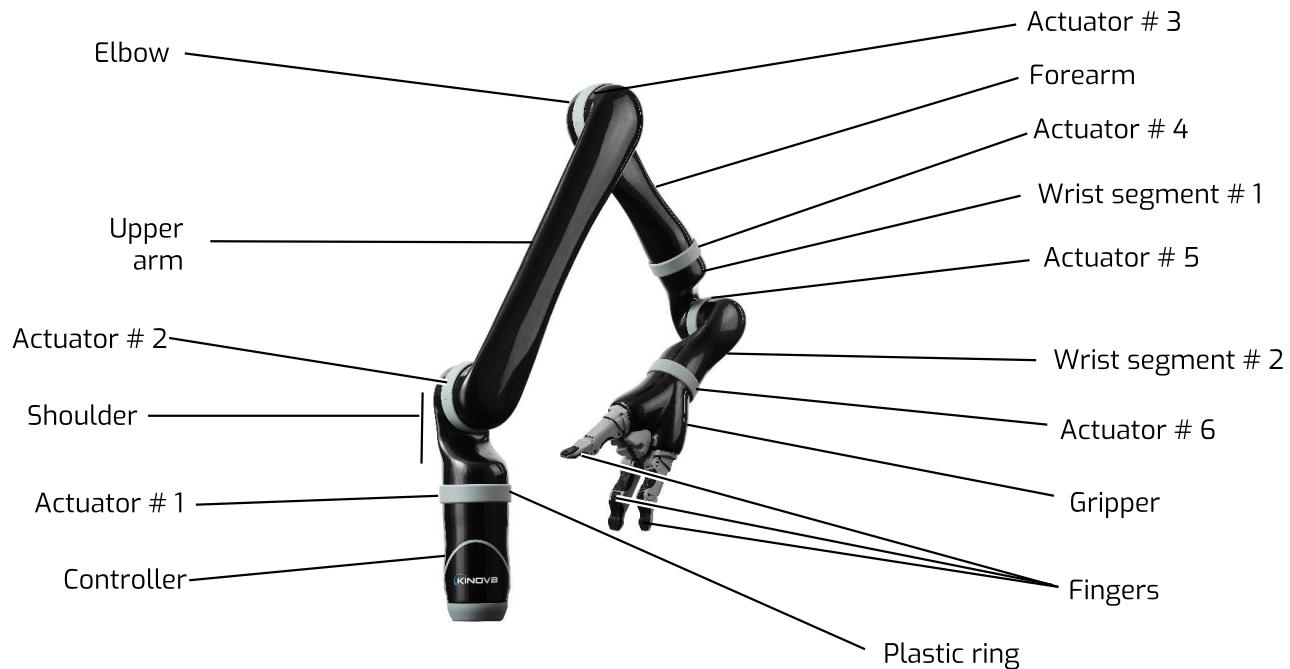
6 DOF Curved Wrist Components

This section shows the components of the 6 DOF curved wrist robotic arm.



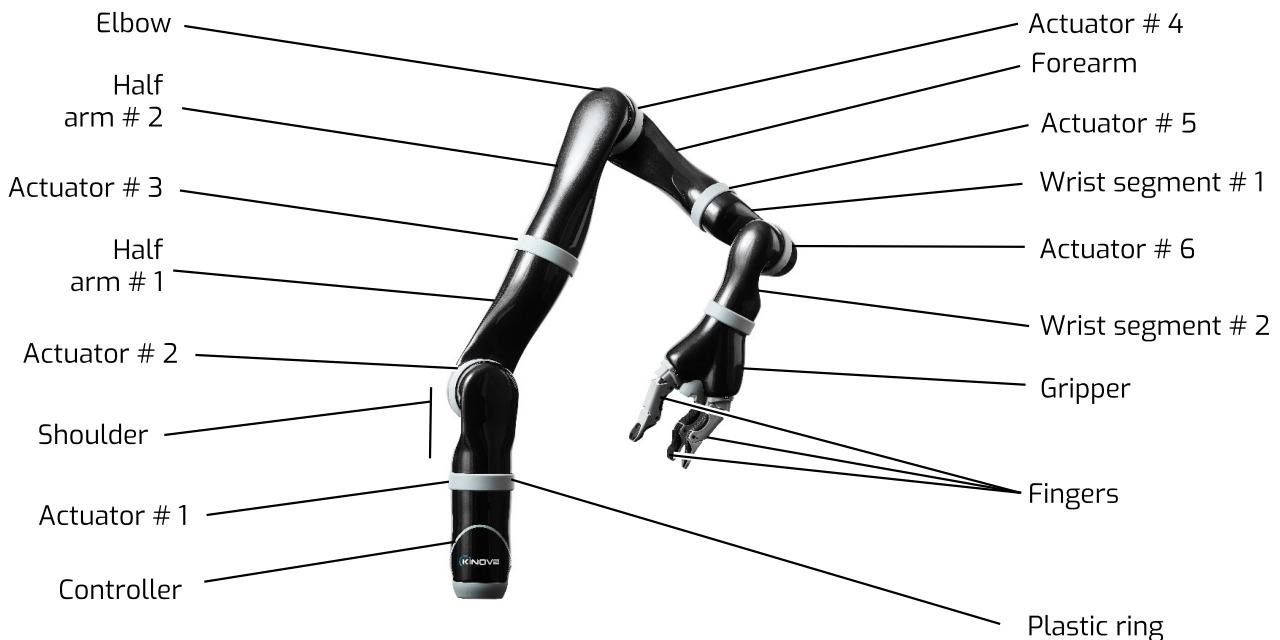
6 DOF Spherical Wrist Components

This section shows the components of the 6 DOF spherical wrist robotic arm.



7 DOF Spherical Wrist Components

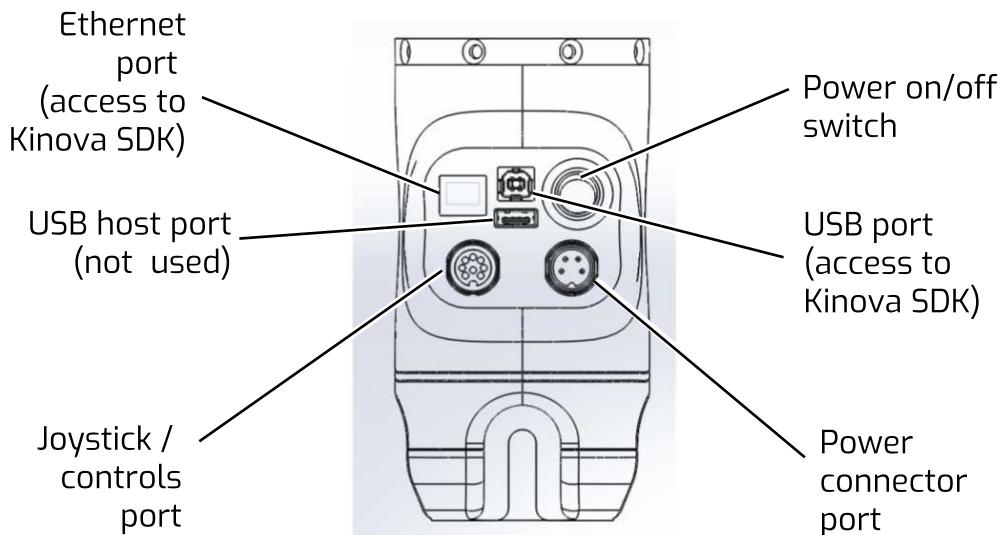
This section shows the components of the 7 DOF spherical wrist robotic arm.



External Connectors

This section describes the external connectors of the robotic arm controller base.

The following figure shows the external connectors located on the arm controller base.



The panel at the back of the controller has four connectors and a power on / off switch.

The **power on /off switch** is used to power up or power down the robotic arm.

The **power connector** port is used to connect the robotic arm to electrical power. The power connector port has four pins

The **joystick / controls port** is used to plug in controls for the arm. The joystick / controls port has six pins.

The **USB port** is used to connect to a computer.

⚠ The control Port and Power Connector are intended to be connected only with a Kinova approved device. Connecting other devices may result in bad performance or even make your arm inoperable and void your warranty.

⚠ Do not override the safety purpose of the polarized or grounding type plug. If the provided cable does not fit in your outlet, consult an electrician for replacement of obsolete outlet.

⚠ To prevent risk of fire or electric shock, avoid overloading wall outlets and extension.

⚠ Protect the cords from being walked on or pinched.

Comparing robotic arm configurations

This section compares the different arm configurations.

Each of the four available robotic arm configurations have their particular strengths. The best option depends on the needs of the specific users / group. The following table compares the four options.

Table 1: Configurations

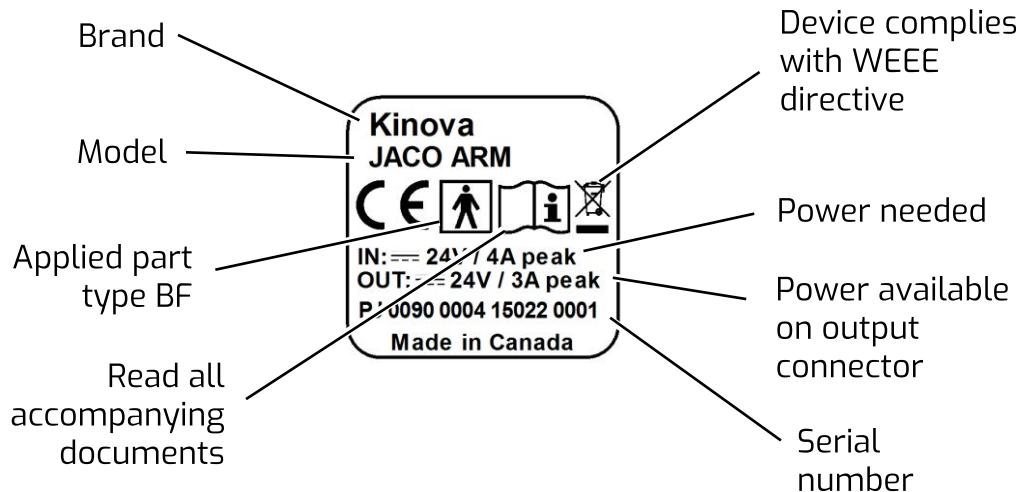
| | 4 DOF | 6 DOF curved wrist | 6 DOF spherical wrist | 7 DOF spherical wrist |
|-----------------------------------|---|---|---|---|
| |  |  |  |  |
| Total weight | 3.6 kg | 4.4 kg | 4.4 kg | 5.5 kg |
| Reach | 75 cm | 90 cm | 98.4 cm | 98.4 cm |
| Maximum payload | <ul style="list-style-type: none"> 4.4 kg (mid-range continuous) 3.5 kg (full-reach peak / temporary) | <ul style="list-style-type: none"> 2.6 kg (mid-range continuous) 2.2 kg (full-reach peak / temporary) | <ul style="list-style-type: none"> 2.6 kg (mid-range continuous) 2.2 kg (full-reach peak / temporary) | <ul style="list-style-type: none"> 2.4 kg (mid range continuous) 2.1 kg (full-reach peak / temporary) |
| Materials | Carbon fiber (links), Aluminum (actuators) | | | |
| Joint range (software limitation) | ± 27.7 turns | | | |
| Maximum linear arm speed | 20 cm / s | | | |
| Power supply voltage | 18 to 29 VDC | | | |
| Average power | 25 W (5 W in standby) | | | 25 W (15W standby) |
| Peak power | 100W | | | |
| Communication protocol | RS485 | | | |
| Communication cables | 20 pins flat flex cable | | | |
| Water resistance | IPX2 | | | |

| | 4 DOF | 6 DOF curved wrist | 6 DOF spherical wrist | 7 DOF spherical wrist |
|-----------------------|--|--|--|--|
| Operating temperature | -10 °C to 40 °C | | | |
| Pros: | <ul style="list-style-type: none"> • Position control • Simplicity • More payload compared to 6DOF and 7DOF | <ul style="list-style-type: none"> • No pinch points • Proven technology • “Unique feel” wrist motion | <ul style="list-style-type: none"> • Simpler kinematics • Better reach • More payload than 7DOF | <ul style="list-style-type: none"> • Ability to move arm without moving hand • More flexibility with movements • Less difficulties with singularities |

Markings and Labels

This section describes markings and labels on the robotic arm.

Please note that these labels may slightly differ from the ones accompanying your device depending of your country. The following figure depicts the information about the label affixed on the robotic arm controller.



Installation of the robotic arm

This section describes the main high level steps of the installation process.

The robotic arm's installation consists of four high-level steps:

1. Mechanical integration
2. Electrical integration
3. End-effector electrical integration (optional)
4. Control integration

Mechanical mounting of the robotic arm

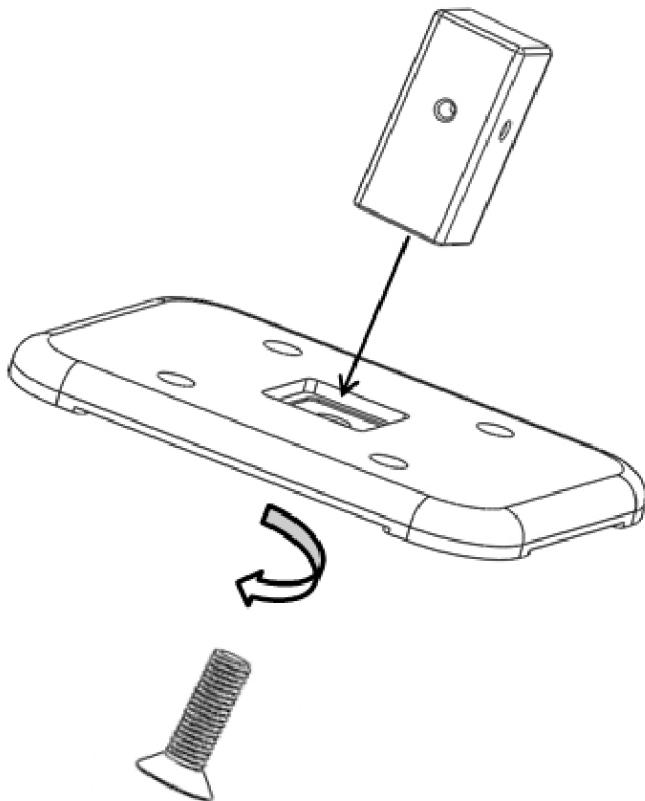
This section describes the steps for mechanical mounting of the robotic arm.

About this task

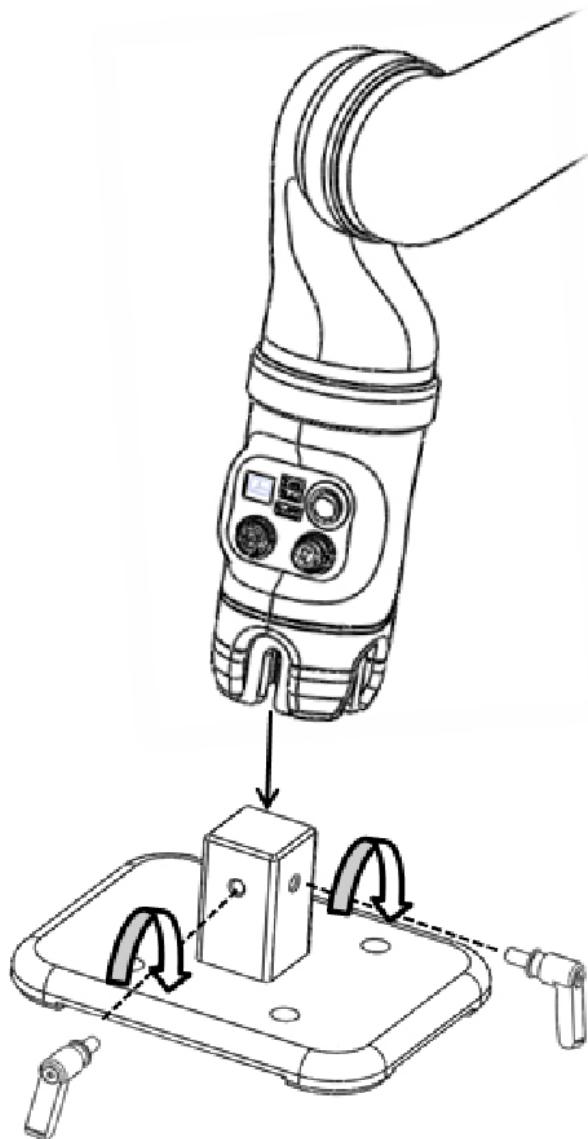
The arm is designed to be installed on a fixed surface or mobile platform. Please make sure the arm is fixed in such a way that its base cannot fall or break during operations involving maximum reach of the arm. Here is a guide on how to install the arm on the mounting kit (XK 0000 0014) supplied with your robotic arm.

Procedure

1. Assemble the mounting kit. Insert the mounting post into the square cavity on the top of the mounting plate and use an 8 mm hex key to attach from the bottom of the mounting plate.



2. Affix the mounting kit to a flat surface. You can either place the larger side of the mounting kit on the edge of a solid flat surface and clamp it as firmly as possible by placing the two clamps supplied with the package on each side of the mounting post or secure four M12 screws through the holes in the mounting plate.
3. Insert the robot arm on the top of the mounting post. Screw the two M8 lever screws into the mounting post, one in the back of the controller and the other on one of the sides of the robot.



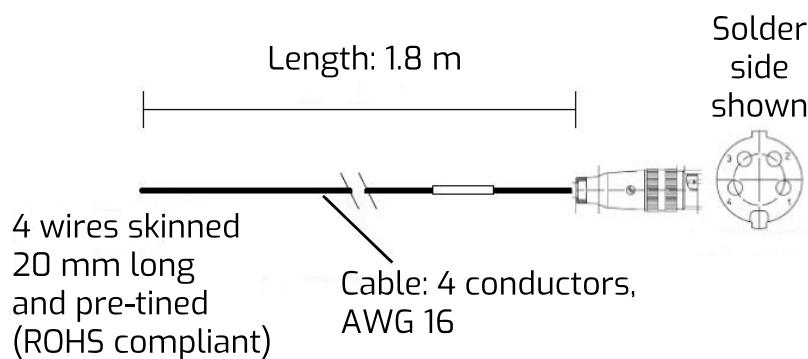
Electrical integration

This section describes how to connect the robotic arm to an electrical power source.

There are two ways of powering the robotic arm:

- Wall electrical outlet
- Battery power

Electrical outlet - You can power your robotic arm using a standard 110/220 V power outlet by plugging the power cord (EH 0300 0001 (USA), EH 2500 0001 (EUR), EH 2500 0002 (AUS), EH 2500 0003 (UK)) into the Power Supply Unit (PSU - AE 0000 0029) on one end and into a power outlet on the other. Then plug the PSU into the base controller power connector.



Battery power - You can use the battery power cord (EH 01M8 0003) by plugging one end into the base controller power connector and connecting the four wires at the other end to a 24V battery. The following table shows the relationships between power connector pinout, the signal, and the wire color.

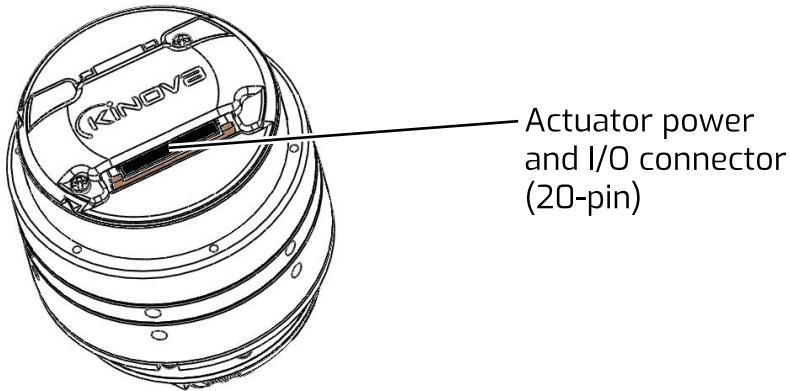
| Pinout table | | |
|--------------|--------|------------|
| Pin # | Signal | Wire color |
| 1 | 24V | Red |
| 2 | 24V | Brown |
| 3 | GND | Black |
| 4 | GND | Orange |

⚠ Make sure that your battery respects the electrical specifications of the robotic arm.

Integrating a new end effector (optional)

This section describes how to integrate a new end effector with Kinova actuators, whether in a Kinova robotic arm or custom application.

Kinova actuators, whether as part of a Kinova robotic arm, or as part of a researcher's custom-built robotics application, have the ability to be connected to end effectors. The output end of each actuator has a 20-pin power and I/O connector. Two of these pins are set aside as dedicated expansion communication and power lines. This allows you the option to connect an additional device to the end of a robotic arm or the end of a chain of actuators in a custom-built set-up.

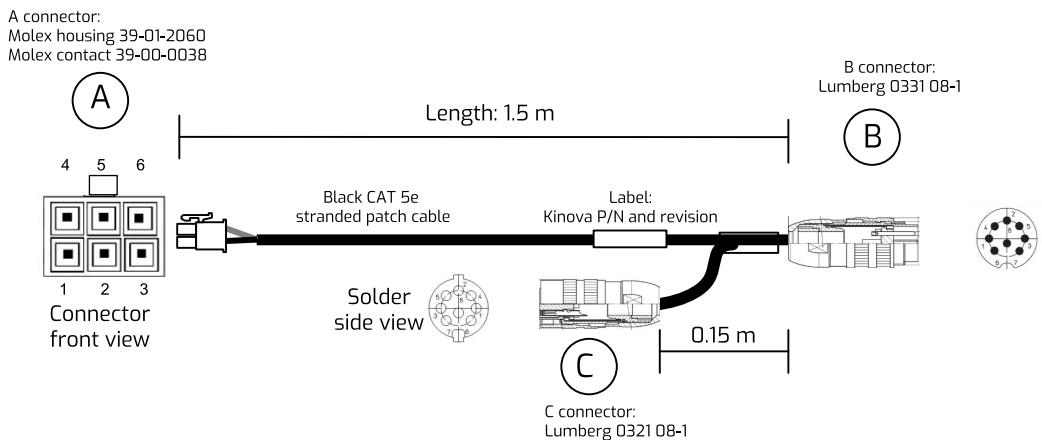


The pins and their function are described in the table below:

Table 2: K-58 actuator pinout

| Pin # | Signal |
|---------|-------------|
| 1 to 8 | 24V input |
| 9 to 16 | GND |
| 17 | RS485 low |
| 18 | RS485 high |
| 19 | Expansion 0 |
| 20 | Expansion 1 |

⚠ Make sure to connect your end effector using only pins 19 and/or 20. Using other pins could severely damage your robot. These two expansion lines are accessible for power and controls purposes via pins at the joystick port. The A "Y" cable is supplied with the robot to allow you to access both the joystick and the expansion lines.



The output pinout of the two expansion lines is indicated in the table below.

Table 3: EH 01M5 0001 pinouts

| Connector A | | |
|-------------|-------------------------|-------------|
| Pin # | Signal | Function |
| 1 | COM1 | RS485_low |
| 2 | GND | GND |
| 3 | COM3 | Expansion 0 |
| 4 | COM2 | RS485_high |
| 5 | 24V (max current: 1.5A) | 24V |
| 6 | COM4 | Expansion 1 |

Control integration

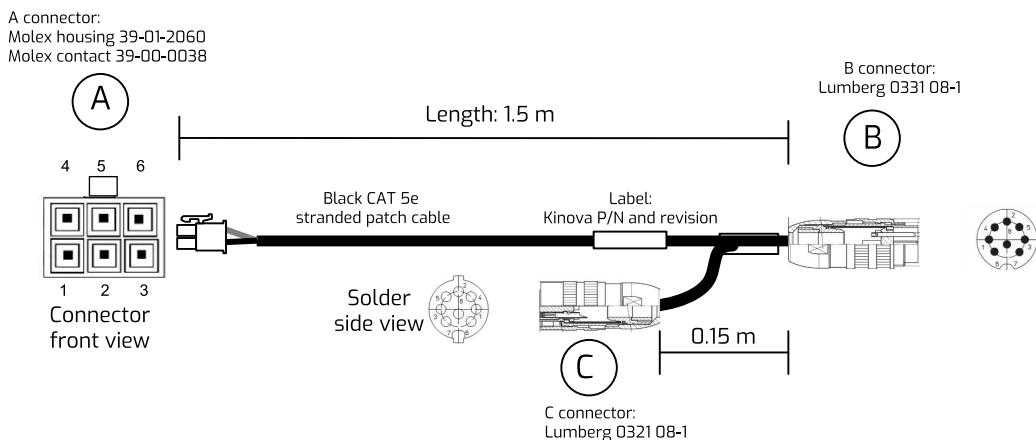
This section describes the controls integration for Kinova robotic applications, whether for Kinova robotic arms or custom-built applications.

Once the mechanical and electrical integration are completed (as well as the end effector integration if applicable), you can power on the robot by flipping the power switch on the back of the controller to ON. To control the robot, you can use either the Kinova API or Kinova's joystick.

API - Connect the USB cable supplied with your package. Connect one end to the controller USB port, and the other to a USB port on the development computer. Install and open the Kinova SDK Development Center on the development computer and follow the procedure and documentation included in the SDK.

Note: Refer to the Development Center user guide and the Kinova API documentation for more guidance on controlling the robotic arm via the API.

Kinova Joystick - Connect the joystick to the joystick port or to the C connector if you are using a "Y" cable. Refer to Kinova's Joystick section in the user guide for all the details regarding the use of the joystick.



Control modes overview

This section explains how to operate the robotic arm with factory configuration.

Kinova robot actuators can be controlled by end effector position, actuators' angular position or actuators' torque. Kinova robots offer the following control mode options:

Table 4: Control modes

| Control mode | Description |
|--|---|
| Cartesian position | Specifies end-effector's position and end-effector's orientation (Euler angles, X-Y'-Z" convention) in the base frame. |
| Cartesian velocity | Specifies end-effector's translation velocities in the base frame and end-effector's rotation velocities in the effector's frame. |
| Angular position | Specifies each actuator's angle. |
| Angular velocity | Specifies each actuator's angular (rotational) velocity. |
| Cartesian admittance (Reactive Force control in Cartesian space) | Applies forces and torques on the end-effector and get a Cartesian motion (translation/rotation) in the appropriate direction. |
| Angular admittance (Reactive Force control in joint space) | Applies torques on actuators and get an angular motion (joint rotation) in the appropriate direction. |
| Direct torque control | Specifies each actuator's torque. By default, each actuator receives its corresponding gravity torque so the robot compensates its own weight. |
| Force control | Specifies forces and torques at the end-effector. The robot automatically computes the torque at each actuator required to generate the appropriate forces/torques at the end-effector. |

Control peripheral options overview

Three options are available to control Kinova robotic arms. Two of these options use the API. The

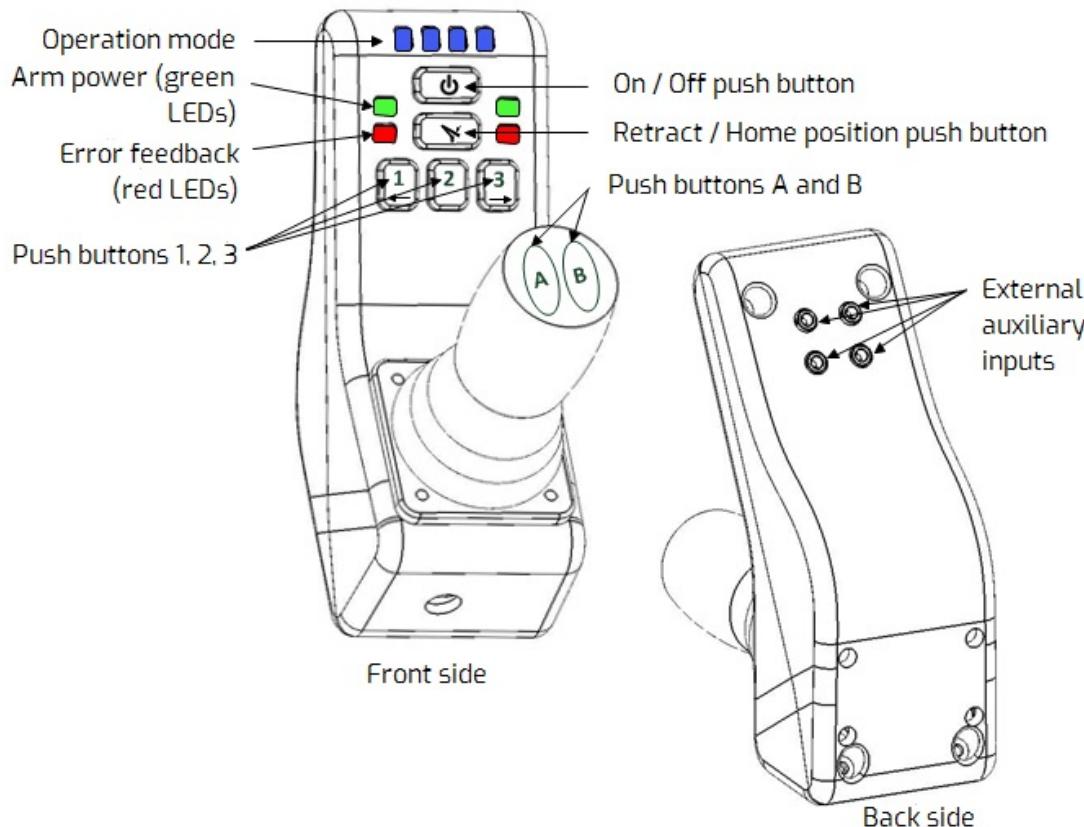
third option is the joystick control. The three options are:

- **(Kinova) joystick control:** This controls the arm in Cartesian velocity (by default as soon as the robot reaches its READY position) or in Angular velocity (if angular control is activated). Joystick control in Cartesian mode is the only control peripheral option Assistive users should be familiar with.
- **Kinova software control:** Kinova provides two different software control panels that allow you to control the arm via a graphical user interface.: the Development Center and the Torque Console. These two software panels allow users to command the arm in position, velocity, and trajectories. It also allows users to activate admittance control (inside the Development Center) and direct torque control/force control (inside the Torque Console). For more details on Kinova software, please see the Development Center User guide (downloadable from Kinova website).
- **API control:** Kinova has a library of C++ functions to control its robotic arms. This library of functions is referred to as the Kinova API. The API (.dll files and .h files) is downloadable from Kinova's website as part of the Kinova software development kit (SDK). The SDK is supplemented by HTML-based documentation detailing all the available functions. The Kinova API is supported on both Windows and Ubuntu. Kinova also offers developers the possibility for developers to control the robotic arm through a [ROS](#) interface (ROS Indigo on Ubuntu 14.04 currently supported). For more information, see the Kinova ROS Github page at <https://github.com/Kinovarobotics/kinova-ros>.

A fourth option is to control the arm by directly interacting with it (pushing/pulling on some link or at the end-effector). The robot becomes reactive to direct interaction in admittance control and direct torque/force control.

Kinova joystick controller

The Kinova standard controller is a three-axis joystick mounted on a support. The controller includes five independent push buttons and four external auxiliary inputs (on the back side).



Joystick movements and modes

The Kinova joystick allows to control the arm in a “2-axis” or “3-axis” mode. The “2-axis” mode will disable the joystick twist rotation.

The following table shows the button default factory settings for using the joystick in a 2-axis and 3-axis modes.

Table 5: Default joystick button settings

| Buttons | One click | Hold 2 sec (Hold until position is reached) |
|---|---|---|
|  | Deactivate / Activate Joystick | Change joystick operating mode (2-axis Vs 3-axis) |
|  | --- | Home / Retracted function |
| 3-Axis | | |
| 1 | Deactivate / Activate Drinking mode | --- |
| 2 | --- | Set Position |
| 3 | --- | Go to pre-set position |
| A | Reach Finger mode | Decrease speed |
| B | Reach Translation & Wrist mode | Increase speed |
| Ext1 | Reach Finger mode | Decrease speed |
| Ext2 | Reach Translation and Wrist mode | Increase speed |
| Ext3 | -- | Home / Retracted function |
| Ext4 | Deactivate / Activate Drinking mode | -- |
| 2-Axis | | |
| 1 | Deactivate / Activate Drinking mode | --- |
| 2 | Reach Wrist orientation & Finger mode | Decrease speed |
| 3 | Reach Translation-X/Y & Translation-Z / Wrist rotation mode | Increase speed |
| A | --- | -- |
| B | --- | -- |
| Ext1 | Reach Wrist orientation & Finger mode | Decrease speed |
| Ext2 | Reach Translation-X/Y & Translation-Z / Wrist rotation mode | Increase speed |
| Ext3 | -- | Home / Retracted function |
| Ext4 | Deactivate / Activate Drinking mode | -- |

Operating principles and Cartesian mode

This section describes at a high level the control of the arm using the joystick in Cartesian mode.

Operating principles

The operating principles are very simple and intuitive. The robotic arm may be operated through several controllers. The following sections present the general control principles through Kinova's joystick.

Basic movements

The normal control of the robotic arm with the joystick is said to be Cartesian. The user commands the end-effector's translations (position variations) with respect to the base and the rotations (orientation variations) around the end-effector's reference point. The different joints are piloted automatically following the given command.

In "Translation mode", the user controls the position of the gripper in space. The gripper will always keep its parallelism with the arm's base. Translation X refers to left/right movements of the gripper. Translation Y refers to front/back movements of the gripper. Translation Z refers to up/down movements of the gripper.

In the "Wrist mode", the user controls the position of the gripper around its center point (reference point) which will not move (or move slightly) when operating in this mode. Lateral orientation refers to a thumb/index circular movement of the wrist around the reference point. Vertical orientation refers to a top/bottom circular movement of the wrist around the reference point. Wrist rotation refers to a circular movement of the gripper around itself.

The "Drinking mode" is to be used with the wrist rotation only. While operating the Jaco² arm in "Drinking mode", the reference point (normally set in the middle of the gripper), is offset in height and length to produce a rotation around another point in the space of the arm.

In the "Finger mode", the user controls the opening and closing of the fingers.

Note: The arm will sometimes respond differently to a given command than described in this section. This may be due to the singularity and collision avoidance algorithms embedded in the kinematics. It is a normal protective behaviour of the arm and is position dependent. Both these avoidance algorithms can be deactivated by the user.

Home / Retracted positions

This section describes the Home and Retracted positions of the robotic arm.

The arm comes with two factory default pre-set positions that may be configured in the Kinova Development Center:

- Home position and
- Retracted position.

The Home position refers to the position of the arm when it is ready to be used. In the Home position, the arm is awaiting commands from the joystick.

The Retracted position refers to the position of the arm when it is not used. The user should always place the arm in the Retracted position when it is unused as it decreases the physical volume occupied by the arm. In the Retracted position, the arm is in standby mode; the joystick features are disabled and power consumption is much lower.

 Never use the Home / Retracted function when carrying liquid. The Home position is pre-set and the wrist may have to rotate and will drop the liquid.

Operating the arm via joystick

This section describes operation of the arm using the joystick.

This section explains how to operate the arm with factory configuration. Contact your reseller for operation instructions in the case of an adapted configuration.



Before operating the arm, please make sure it is properly installed.



Do not manipulate cutting, very sharp or any dangerous tools or objects with the arm.



This equipment is not designed to act as a lift.



This equipment is not designed to be used in presence of flammable mixture. (Not AP or APG rated).



Do not install the arm near any heat sources, such as radiators. Do not use it to directly manipulate hot objects.

Joystick control quick start

This section describes how to get started using the Kinova joystick to control the arm in the default configuration.

About this task

Procedure

1. Turn ON the device by pushing the ON/OFF switch located on the arm base.
2. Wait until the green lights on the controller stop flashing.
3. Put the arm in its Home position by holding down the HOME/RETRACTED button until the arm stops moving. The arm will slowly reach the Home position.

Note: When starting the arm, you are in 3-Axis operation mode, "Translation control mode", meaning that any movement of the joystick will move the center of the gripper parallel to the floor.

4. You may move the 3 axes of the joystick to experience the Translation control mode.

Note: To change the operating mode of the Joystick, hold the ON/OFF button for 2 seconds. At this point, you are in 2-axis mode and the stick rotation is deactivated.

5. One press of Button B will bring you in Wrist control mode meaning that any movement of the joystick will result in a rotation of the gripper around its center.

Note: Another press of Button B will bring you back in Translation control mode.

6. One press of Button 1 will activate the Drinking mode which may be used only in Wrist mode. When rotating the joystick lever, you will see that the arm's wrist rotation now compensates for the height and radius of a virtual glass. This movement is ideal when trying to drink directly from a glass.

Note: Another press of Button 1 will disable Drinking mode.

7. One press of Button A will bring you in Finger control mode. The fingers will move per a left/right inclination of the joystick.

Note: At any time, you may use the Home / Retracted button until the arm stops moving to bring it back to its Home position.

Note: If you hold the Home / Retracted button again, the arm will start to move toward the Retracted position.

8. Hold the On/Off Button for 2 seconds to change the operating mode. This will disable the stick rotation. You are now in a 2-Axis Translation control mode. Stick rotation won't have any effect and you will only be able to control the horizontal translation of the arm (X- and Y- axis).

9. One press of Button 3 will bring you to control the vertical translation of the gripper (Translation-Z) and Wrist rotation.

Note: Another hit on Button 3 will bring you back in Translation-X and Translation-Y control mode.

10. One press of Button 1 will activate the Drinking mode which may be used only in Wrist mode. When rotating the joystick lever, you will see that the arm's wrist rotation now compensates for the height and radius of a virtual glass. This movement is ideal when drinking directly from a glass.

11. One press of Button 2 will bring you to control the wrist orientation (Lateral orientation and Vertical orientation).

12. Another press of Button 2 will bring you to Finger control mode. The fingers will move according to a left/right inclination of the joystick.

Note: Another press of Button 2 will bring you back in Lateral orientation and Vertical orientation control mode.

Default joystick motion settings - Cartesian three-axis mode

This section describes default motion settings in Cartesian three-axis mode.

Table 6:

| Joystick movement | Arm movement | Availability |
|-------------------------------|------------------------------------|---------------------------|
| Translation Mode | | |
| Incline FRONT | Gripper moves forward | 4 / 6 / 6 -S / 7 DOF-S |
| Incline BACK | Gripper moves backward | 4 / 6 / 6 -S / 7 DOF-S |
| Incline LEFT | Gripper moves left | 4 / 6 / 6 -S / 7 DOF-S |
| Incline RIGHT | Gripper moves right | 4 / 6 / 6 -S / 7 DOF-S |
| Rotate stick CLOCKWISE | Gripper moves up | 4 / 6 / 6 -S / 7 DOF-S |
| Rotate stick COUNTERCLOCKWISE | Gripper moves down | 4 / 6 / 6 -S / 7 DOF-S |
| Wrist Mode | | |
| Incline FRONT | Vertical orientation – Top side | 6 / 6 -S / 7 DOF-S |
| Incline BACK | Vertical orientation – Bottom side | 6 / 6 -S / 7 DOF-S |
| Incline LEFT | Lateral orientation – Thumb side | 6 / 6 -S / 7 DOF-S |
| Incline RIGHT | Lateral orientation – Index side | 6 / 6 -S / 7 DOF-S |
| Rotate stick CLOCKWISE | Wrist rotation clockwise | 4 / 6 / 6 -S / 7 DOF-S |
| Rotate stick COUNTERCLOCKWISE | Wrist rotation counterclockwise | 4 / 6 / 6 DOF-S / 7 DOF-S |
| Finger Mode | | |
| Incline LEFT | Close Fingers (3 finger mode) | 4 / 6 / 6 -S / 7 DOF-S |
| Incline RIGHT | Open Fingers (3 finger mode) | 4 / 6 / 6 -S / 7 DOF-S |
| Incline FRONT | Open Fingers (2 finger mode) | 6 -S / 7 DOF-S |
| Incline BACK | Close Fingers (2 finger mode) | 6 -S / 7 DOF-S |

Default joystick motion settings - Cartesian two-axis mode

This section describes default motion settings in Cartesian two-axis mode.

Table 7:

| Joystick movement | JACO arm movement | Availability |
|---|------------------------------------|---------------------|
| Translation-X and Translation-Y | | |
| Incline FRONT | Gripper moves forward | 4 / 6 / 6S / 7S DOF |
| Incline BACK | Gripper moves backward | 4 / 6 / 6S / 7S DOF |
| Incline LEFT | Gripper moves left | 4 / 6 / 6S / 7S DOF |
| Incline RIGHT | Gripper moves right | 4 / 6 / 6S / 7S DOF |
| Translation-Z and Wrist Rotation | | |
| Incline FRONT | Gripper moves up | 4 / 6 / 6S / 7S DOF |
| Incline BACK | Gripper moves down | 4 / 6 / 6S / 7S DOF |
| Incline LEFT | Wrist rotation clockwise | 4 / 6 / 6S / 7S DOF |
| Incline RIGHT | Wrist rotation counter-clockwise | 4 / 6 / 6S / 7S DOF |
| Wrist Orientation | | |
| Incline FRONT | Vertical orientation – Top side | 6 / 6S / 7S DOF |
| Incline BACK | Vertical orientation – Bottom side | 6 / 6S / 7S DOF |
| Incline LEFT | Lateral orientation – Thumb side | 6 / 6S / 7S DOF |
| Incline RIGHT | Lateral orientation – Index side | 6 / 6S / 7S DOF |
| Finger Mode | | |
| Incline LEFT | Close Fingers (3 finger mode) | 4 / 6 / 6S / 7S DOF |
| Incline RIGHT | Open Fingers (3 finger mode) | 4 / 6 / 6S / 7S DOF |
| Incline FRONT | Open Fingers (2 finger mode) | 6S / 7S DOF |
| Incline BACK | Close Fingers (2 finger mode) | 6S / 7S DOF |

Controlling the arm in Angular mode

In Angular mode, the user commands each individual actuator's rotation. Angular mode must be activated using the Kinova Development Center software or Kinova API. Multiple joints can be commanded simultaneously using the joystick (when diagonal control is enabled). The joystick default settings in Angular mode are summarized in the following table.

Table 8:

| Joystick movement | Arm movement | Availability |
|--------------------------------|---|--------------|
| Translation mode | | |
| Incline LEFT | Joint 1 rotates positively (angle increases) | 6S /7S DOF |
| Incline RIGHT | Joint 1 rotates negatively (angle decreases) | 6S /7S DOF |
| Incline FRONT | Joint 2 rotates negatively (angle decreases) | 6S /7S DOF |
| Incline BACK | Joint 2 rotates positively (angle increases) | 6S /7S DOF |
| Rotate CLOCKWISE | Joint 3 rotates positively (angle increases) | 6S /7S DOF |
| Rotate stick COUNTER-CLOCKWISE | Joint 3 rotates negatively (angle decreases) | 6S /7S DOF |
| Wrist mode | | |
| Incline LEFT | Joint 4 rotates negatively (angle decreases) | 6S /7S DOF |
| Incline RIGHT | Joint 4 rotates positively (angle increases) | 6S /7S DOF |
| Incline FRONT | Joint 5 rotates positively (angle increases) | 6S /7S DOF |
| Incline BACK | Joint 5 rotates negatively (angle decreases) | 6S /7S DOF |
| Rotate CLOCKWISE | Joint 6 rotates positively (angle increases) | 6S /7S DOF |
| Rotate stick COUNTER-CLOCKWISE | Joint 6 rotates negatively (angle decreases) | 6S /7S DOF |
| Finger mode | | |
| Incline LEFT | Close Fingers (3 finger mode) | 6S /7S DOF |
| Incline RIGHT | Open Fingers (3 finger mode) | 6S /7S DOF |
| Incline FRONT | Open Fingers (2 finger mode) | 6S /7S DOF |
| Incline BACK | Close Fingers (2 finger mode) | 6S /7S DOF |
| Rotate CLOCKWISE | Joint 7 rotates positively (angle increases) | 7S DOF |

| Joystick movement | Arm movement | Availability |
|--------------------------------|--|--------------|
| Rotate stick COUNTER-CLOCKWISE | Joint 7 rotates negatively (angle decreases) | 7S DOF |

Kinova Joystick LED feedback

The Kinova joystick offers visual feedback:

- Blue lights : Feedback on control mode
- Green lights : Feedback on arm power
- Red lights : Feedback on error

Kinova joystick Blue LEDs feedback

This section describes the blue LED feedback on the Kinova controller.

The blue LEDs on the controller give feedback on the current control mode. The interpretation of the blue mode LED indicators is described in the following table.

Table 9: Control mode feedback

| Blue LED indication | Control Mode |
|---------------------|---|
| 3-Axis | Translation (X-Y-Z) |
| | Wrist |
| | Fingers |
| | Drinking mode (to be used with wrist rotation mode) |
| | Disabled controller |
| 2-Axis | Translation (X-Y) |
| | Translation (Z) / Wrist Rotation |
| | Wrist Orientation |
| | Fingers |
| | Drinking mode (to be used with wrist rotation mode) |
| Disabled controller | |

When no blue lights are visible, the controller is disabled. To enable the controller, you must either proceed with the following options:

- The On/Off button must be pushed
- The arm must be set in its HOME position by holding the HOME/RETRACTED function until the arm stops moving.

Kinova joystick Green LEDs feedback

This section describes the green LED feedback on the Kinova controller.

The green lights offer visual feedback on the power status of the arm.

Table 10: Power status feedback

| Green LED indication | Power Status |
|----------------------|---|
| Flashing | The arm has just been turned on and the internal communication is synchronizing. The arm is not yet ready to use. |
| Solid | The arm is powered and ready to use. |

Kinova joystick Red LEDs feedback

This section describes the red LED feedback on the Kinova controller.

The red lights offer visual feedback on possible errors that may occur while operating the arm:

Table 11: Error status feedback

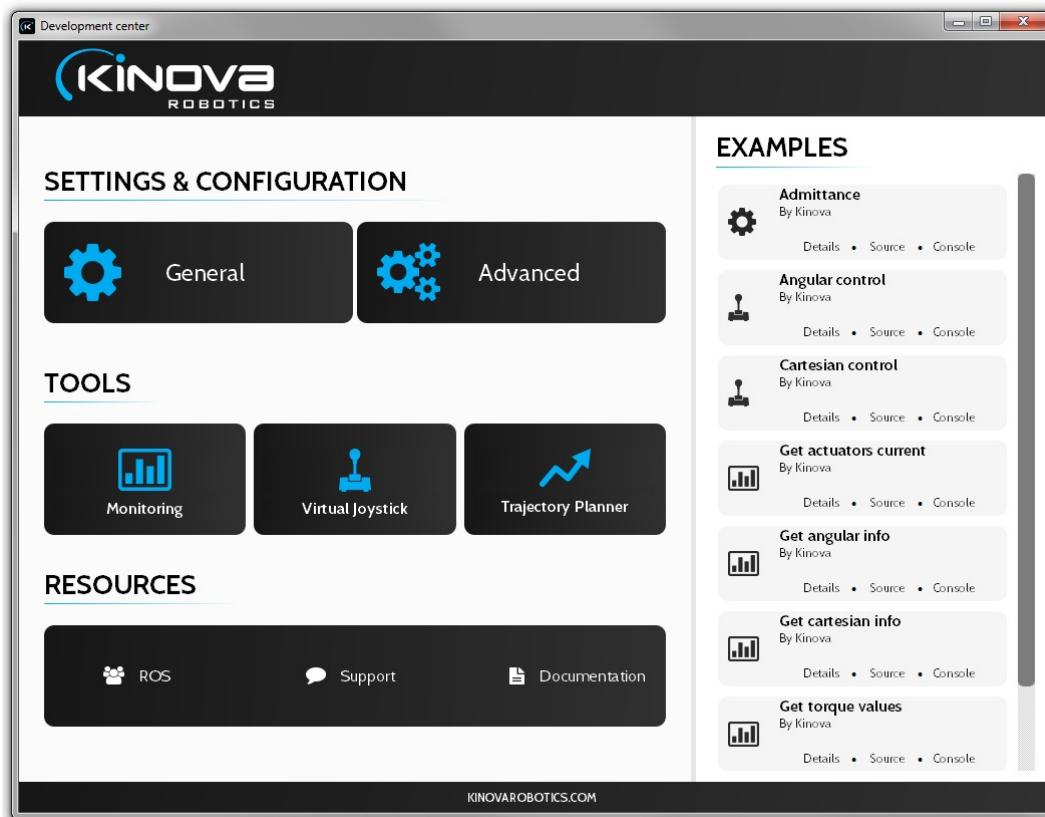
| Red LED indication | Cause of the Error Status | Action to resolve the situation |
|--------------------|---|--|
| Flashing | The weight that is being lifted is too heavy or too much force is applied on the arm. | Safely put down the object, or release force applied on the arm, and wait until red lights turn off. |
| | The temperature of a section of the arm is too high. | The usage of the arm is excessive and doesn't respect the normal use definition. Safely put down any object that is in the gripper, bring back the arm to its RETRACTED position, and wait until red lights turn off. |
| Solid | The input voltage to the arm (or batteries) is too low or too high. | Safely put down any object that is in the gripper, bring back the arm to its RETRACTED position. Ensure the power supply is appropriate, connections are secure and batteries are charged properly before using the arm again. |
| | The arm is in fault | Turn off the arm and turn it back on. If the problem remains, contact your distributor or Kinova. |

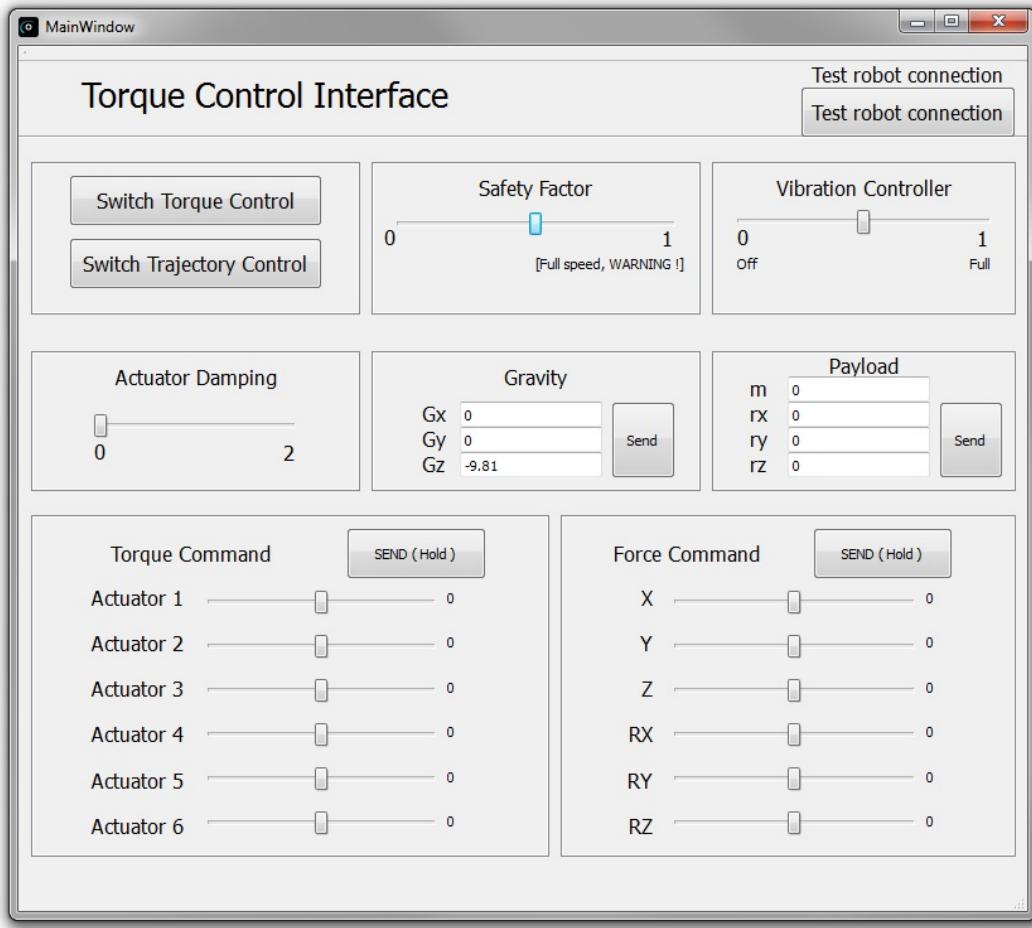
Controlling the arm using Kinova software

This section describes at a high level control of the arm using Kinova software.

Multiple functionalities are offered by Kinova software. Using the Development Center and the Torque Console, users can (among other things) send trajectories, monitor their robot's state, activate admittance and switch between Cartesian and angular control. Kinova software is also useful to update the robot's firmware and to diagnose different problems.

The Development Center and Torque Console are available for download on the Kinova website. For a complete list of Kinova software functionalities and use instructions, please see the separate Kinova Development Center User Guide.





Controlling the arm using the Kinova API

This section describes at a high level control of the arm using Kinova API.

As with the Development Center and the Torque Console, information about the API is downloadable from Kinova's website. The Development Center comes with HTML documentation describing the C++ programming functions that can be used to access the API. A good way to start with the API is to look at the examples provided with the Kinova Development Center. For more details, please see the Kinova Development Center User Guide and the HTML-based API documentation.

Singularity Avoidance

This section discusses singularity-avoidance in the robotic arm.

In Cartesian mode, there are some configurations in which the robot loses one or more degrees of freedom (meaning the robot is not able to move in one direction or the other.). These configurations are called singularities and Kinova robots avoid them automatically. This means that Cartesian commands sent by the user may be modified somewhat to avoid a singularity.

Singularity auto-avoidance behavior can be deactivated using the `ActivateAutomaticSingularityAvoidance` API function and setting its parameter to false.

Table 12: 6DOF Spherical Singularities

| Singularity | Explanation | Robot Behavior |
|---|--|--|
| Boundary singularity  | The arm is at full reach. It cannot move anymore in the direction it is currently reaching out. | Not possible to bring elbow at 180° in Cartesian mode. |
| Wrist-over-base singularity  | The wrist point is aligned with the first joint axis. Joint 1 cannot move the hand in translation anymore. | Not possible to bring the wrist inside a virtual cylinder located around the base. The virtual cylinder has a 15 cm radius approximately. |
| Wrist alignment singularity  | Joint 4 and 6 are aligned and have the same effect. The hand cannot rotate in one direction anymore. | Executed motion when the wrist is near alignment configuration might be a little different from the commanded motion. But the robot is essentially able to go 'through' this singular configuration. |

Table 13: 7DOF Spherical Singularities

| Singularity | Explanation | Robot Behavior |
|--|--|---|
| Boundary singularity  | The arm is at full reach. Joint 4 is at 180°. The arm cannot move in the direction it is currently reaching out. | When singularity avoidance is activated, it's not possible to bring the elbow at 180° in Cartesian mode. |
| Joints 2 and 3 singularity  | Joint 2 is at 180° so joints 1 and 3 are perfectly aligned and have the same effect. Joint 3 is at 90° or at 270° so joint 2 and joint 4's axis is perpendicular. The robot cannot move purely along an axis in translation anymore. | When singularity avoidance is activated, it's not possible to bring joint 3 near 90° or 270° when joint 2 is near 180° (or vice versa, to bring joint 2 near 180° when joint 3 is near 90° or 270°) in Cartesian mode. Besides, the fitness function will try to avoid the singularity by moving joint 2 away from 180° and joint 3 away from 90° or 270° while moving in the robot's null space. For more information on the fitness function, please see section "Null space motion." |
| Joints 2 and 6 singularity  | Joint 2 is at 180° so joints 1 and 3 are perfectly aligned and have the same effect. Joint 6 is at 180° so joints 5 and 7 are perfectly aligned and have the same effect. The hand cannot rotate in one direction anymore. | When singularity avoidance is activated, it's not possible to bring joint 2 near 180° when joint 6 is near 180° in Cartesian mode. Besides, the fitness function will try to avoid the singularity by moving joints 2 and 6 away from 180° while moving in the robot's null space. For more information on the fitness function, please see section "Null space motion." |

| Singularity | Explanation | Robot Behavior |
|--|---|--|
| Joints 5 and 6 singularity  | Joint 6 is at 180° so joints 5 and 7 are perfectly aligned and have the same effect. Joint 5 is at 90° or at 270° so joint 4 and joint 6's axis is perpendicular. The robot can't complete pure rotations around an axis. | When singularity avoidance is activated, it's not possible to bring joint 5 near 90° or 270° when joint 6 is near 180° in Cartesian mode. Besides, the fitness function will try to avoid the singularity by moving joint 5 away from 90° or 270° and joint 6 away from 180° while moving in the robot's null space. For more information on the fitness function, please see section "Null space motion." |

Self-collisions auto-avoidance

In Cartesian mode, Kinova robots avoid self-collisions (essentially collisions between the hand and the base but also between the hand and the arm segment). This means the Cartesian command sent by the user may be modified by the control software to avoid a self-collision. Singularity auto-avoidance behavior can be deactivated using the `ActivateAutomaticCollisionAvoidance` API function and setting its parameter to false.

Please note that collisions with other objects present in the environment are not automatically handled by the robot.



Self-collisions are not automatically avoided during angular control. Only joint limits are handled.

7 DOF Spherical Null space motion

The 7 DOF-S robot is redundant because it only requires six degrees of freedom (=six actuators) to move and orient its effector in 3D space although it has seven actuators. A robot with six actuators can only reach a given end-effector position and orientation with a few configurations, which are very different from one another. The 7th degree of freedom (or 7th actuator) gives more flexibility to the robot and lets it reach a given end-effector position and orientation with a multiple/infinite number of configurations.

In fact, the robot can even move its elbow (joint 4) without modifying its end-effector position and orientation. This type of motion is called “null space motion” because it does not affect where the end-effector is (i.e. it has no effect in the task space). During Cartesian control, Kinova automatically optimizes the robot motion in its null space with a special optimisation function called the fitness function. The fitness function will try to find the best compromise to avoid singularities, position the elbow and avoid angular limits of the 7 DOF-S arm without modifying the user’s Cartesian command. Right now, the robot’s null space motion can be deactivated, but the fitness function parameters cannot be modified.

⚠ Because of null space motion, the robot’s elbow can move quite a lot during Cartesian control. This is because the robot is trying to avoid singularities and angle limits while trying to keep a preferred angle position and without modifying the user’s Cartesian command.

⚠ If the currently implemented null space motion algorithm does not suit your needs, you can control the robot in angular space or deactivate the null space motion using the API function.

Protection zones

Protection zones can be defined using the API (For research users) or with the help of a Kinova qualified service personnel member (for Assistive users). Protection zones must be box-shaped. Once a protection zone is defined, the robot will avoid getting inside this zone. If possible, it will slide on the zone. If not, it will stop. By default, a protection zone is defined near the base connectors.

 If you give a command and the robot stops moving because it is too close to a protection zone, try moving the robot in another direction. If you notice that your robot is not moving under any command, your robot's behaviour is abnormal. Try sending the arm to its Ready position (to continue using it) and contact Kinova support (to report the problem).

 Protection zones are not avoided during angular control. Only joint limits are handled.

Rotating frame / Fixed frame

This section describes the difference between Rotating frame and Fixed frame for the hand/gripper.

By default, Kinova robots are configured in Rotating frame. In Rotating frame, the hand/gripper rotates automatically to follow the arm's motion in the horizontal plane. This gives a more human-like behavior. In Fixed frame, the hand's orientation will not change unless explicitly commanded.

Users can switch between Rotating frame and Fixed frame using the Kinova Development Center software or API. Frame selection (Rotating or Fixed) is recorded inside the robot's memory. This means that a robot that was configured in Fixed frame will not reset to Rotating frame at reboot.

Usable workspace

This section describes the usable workspace of the robotic arm.

The robotic arm's usable workspace is represented in the graphic below. The workspace provided is for angular control.

When the robot is in Cartesian mode, its effective usable workspace will be slightly smaller because of singularities, self-collisions and protection zone automatic avoidance algorithms.

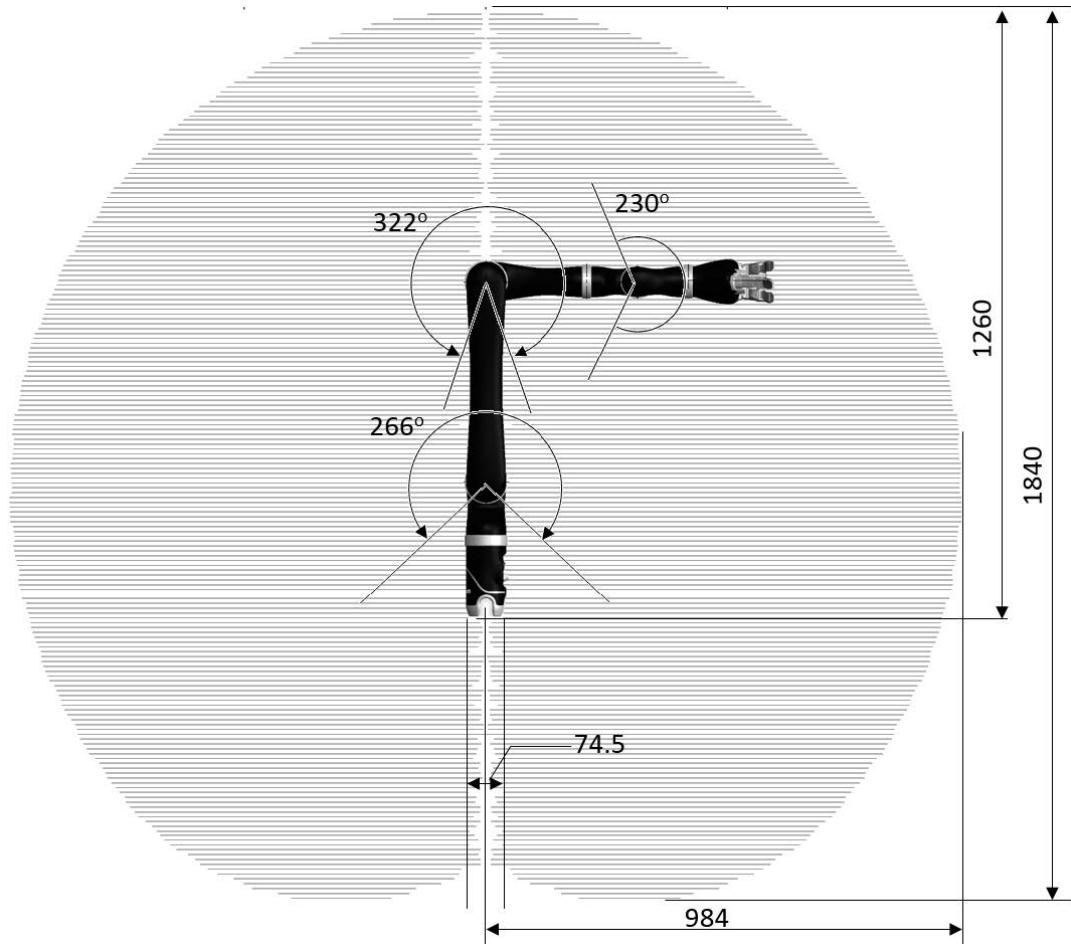


Figure 5: Usable workspace - dimensions in mm. 6 DOF shown for illustrative purposes

Admittance control

This section describes admittance control both in Cartesian and in angular mode.

Admittance control (also called Reactive Force control) can be activated and deactivated using the Kinova Development Center software or API. When admittance is active, it becomes possible to move the robot by hand. For Cartesian admittance, the robot must be in Cartesian mode and admittance must be activated. For angular/joint admittance, the robot must be in angular mode and admittance must be activated.

During Cartesian admittance control, the robot continues to avoid singularities, self collisions and protection zones automatically (unless these functions are deactivated using the API). The null space motion will also stay active (unless deactivated using the API). Finally, maximum velocity and acceleration limits stay active during admittance control in angular or Cartesian space. User force input is automatically bounded between a minimum and a maximum value inside the code. These minimum and maximum values are configurable inside the API. Damping and inertia parameters present in the robot's admittance model are also configurable.

The table below shows default admittance parameters for the 7 DOF spherical configuration. Details will differ for other configurations.

Table 14: Admittance default parameters for 7 DOF Spherical configuration.

| Cartesian admittance parameters | Angular admittance parameters |
|--|---|
| Damping_X=18.18 N/(m/s) | Damping_Joint1=3.95 Nm/(rad/s) |
| Damping_Y=31.82 N/(m/s) | Damping_Joint2=9.2 Nm/(rad/s) |
| Damping_Z=54.55 N/(m/s) | Damping_Joint3=3.95 Nm/(rad/s) |
| Damping_ThetaX=0.85 Nm/(rad/s) | Damping_Joint4=3.95 Nm/(rad/s) |
| Damping_ThetaY=0.85 Nm/(rad/s) | Damping_Joint5=0.5 Nm/(rad/s) |
| Damping_ThetaZ=0.85 Nm/(rad/s) | Damping_Joint6=0.5 Nm/(rad/s) |
| Inertia_X=6.36 N/(m/s ²) | Damping_Joint7=0.5 Nm/(rad/s) |
| Inertia_Y=11.14 N/(m/s ²) | Inertia_Joint1=0.711 Nm/(rad/s ²) |
| Inertia_Z=19.10 N/(m/s ²) | Inertia_Joint2=1.656 Nm/(rad/s ²) |
| Inertia_ThetaX=0.21 Nm/(rad/s ²) | Inertia_Joint3=0.711 Nm/(rad/s ²) |
| Inertia_ThetaY=0.21 Nm/(rad/s ²) | Inertia_Joint4=0.711 Nm/(rad/s ²) |
| Inertia_ThetaZ=0.21 Nm/(rad/s ²) | Inertia_Joint5=0.09 Nm/(rad/s ²) |
| CartesianForceMin_X = 8.0N; | Inertia_Joint6=0.21 Nm/(rad/s ²) |
| CartesianForceMin_Y = 8.0N; | Inertia_Joint7=0.21 Nm/(rad/s ²) |
| CartesianForceMin_Z = 8.0N; | AngularForceMin_Joint1 = 4.0Nm |
| CartesianForceMin_ThetaX = 1.5Nm | AngularForceMin_Joint2 = 7.0Nm |
| CartesianForceMin_ThetaY = 1.5Nm | AngularForceMin_Joint3 = 2.0Nm |
| CartesianForceMin_ThetaZ= 1.5Nm | AngularForceMin_Joint4 = 2.0Nm |
| CartesianForceMax_X= 12.0N | AngularForceMin_Joint5 = 1.5Nm |
| CartesianForceMax_Y = 15.0N | AngularForceMin_Joint6 = 1.5Nm |
| CartesianForceMax_Z = 20.0N | AngularForceMin_Joint7= 1.5Nm |
| CartesianForceMax_ThetaX = 2.5Nm | AngularForceMax_Joint1= 7.0Nm |
| CartesianForceMax_ThetaY= 2.5Nm | AngularForceMax_Joint2 = 14.0Nm |
| CartesianForceMax_ThetaZ = 2.5Nm | AngularForceMax_Joint3 = 5.0Nm |
| | AngularForceMax_Joint4 = 5.0Nm |
| | AngularForceMax_Joint5 = 2.0Nm |
| | AngularForceMax_Joint6 = 2.0Nm |
| | AngularForceMax_Joint7 = 2.0Nm |

⚠ In Cartesian admittance mode, you should grab the robot by its hand when you interact with it. If you grab it by another link (e.g. grab the robot at elbow level), the admittance will still work, but the hand might rotate in an unusual way.

⚠ If you have the impression that the robot is moving unusually in admittance mode, please check your torque sensor calibration.

Torque Control

This section describes torque control.

As its name suggests, torque control lets the user control the motors in torque rather than in position. By default, the arm compensates its own weight in torque mode. Users can also send torque commands and force commands using Kinova Torque Console or API.

By default, the robot also includes some safety features. The first safety is that the arm will not switch from position mode to torque mode unless the torque read by the actuators and the gravity torques computed by the gravity model inside the robot are too different. On reboot, the arm will always go back to position mode.

Once in torque mode, some safety features can bring back the arm in position mode. This is the case when one of the motors' rotation velocity gets too high. Users can disable this safety feature at their own risk using the Kinova Torque Console or API (Safety Factor to 0). By default, the robot will also switch back to position mode if the hand gets too close from the base or if a specific joint gets too close from its angle limits. Users can disable the base collision avoidance safety feature at their own risk using the API.

⚠ Please note that when the robot is in torque mode, it is NOT possible to move the robot's fingers. This is a limit to our actual system and it should be fixed in the next firmware release.

⚠ Do not disable the velocity safety feature unless you are sure that your robot's torque sensors are well calibrated.

⚠ Be very careful when you disable a torque control safety feature. Ideally, keep the robot in an open environment free of near potential obstacles. Please keep in mind that commanding your robot in torque mode with safeties disabled could damage your robot if the motors start turning too fast or if the robot collides with itself or the environment. If you disable the base collision avoidance safety, keep in mind that the robot could collide with itself. A lot of safety parameters are customizable in torque mode (see API HTML documentation for more details), but be aware that these customizations require knowledge to use appropriately.

⚠ Moving the robot very quickly in torque mode can lead, in rare occasions, to an unexpected reboot - the robot stops moving or switches back to position mode and waits for a Home/Ready command. If you observe this behaviour, please contact the Kinova Support team (see Contacting Support).

Improving robot behavior in Torque mode

This section describes how to improve behavior of the robot in Torque mode.

Kinova robots allow to compensate the gravity torques associated with their own weight. Two gravity compensation modes are available:

- Manual (default mode)
- Optimal (must be activated by user)

In **Manual mode**, each robot segment mass and center of mass is specified. For a robot with 6 motors, this makes 24 parameters. Users can change segment masses and centers of mass through the function `SetManualInputParam()`.

In **Optimal mode**, the robot computes gravity torques from a series of parameters linking the angle and torque readings. To find these parameters, the robot must perform an automatic trajectory. The accuracy of the gravity torques estimation with the Optimal mode is usually about three times better than with the Manual mode.

To use the Optimal gravity compensation mode, you need to create a custom C++ program calling three main functions from Kinova API:

1. a. `RunGravityZEstimationSequence()` - for 6 DOF Spherical
b. `RunGravityZEstimationSequence7DOF()` - for 7 DOF Spherical
2. `SetOptimalZParam()`
3. `SetGravityType()`

The first function to be called is either `RunGravityZEstimationSequence()` for the 6 DOF Spherical, or `RunGravityZEstimationSequence7DOF()` for the 7 DOF Spherical. `RunGravityZEstimationSequence()` takes for input the robot type and an empty array of 16 floats, while `RunGravityZEstimationSequence7DOF()` takes the robot type and an empty array of 19 floats. This will compute the 16 (or 19) optimal gravity parameters and place them in the empty array of 16 (or 19) floats. `RunGravityZEstimationSequence()` will also write the parameters in a text file called `ParametersOptimal_Z.txt`. This file is found in the same folder as your C++ project. This function is not called every time you want to use the robot. Once the optimal parameters have been obtained, this function does not need to be called. Only call this function if you want to recalibrate the gravity parameters (which is the case if you tighten/untighten some screws, or if the robot seems to have an unusual behaviour after resetting the 'zero' of the torque sensors).

The second function to be called is `SetOptimalZParam()`, which takes for input the array of optimal parameters from the previous step. `SetOptimalZParam()` informs the robot on the new optimal parameters. This function must be called at every reboot. When you call this function without prior calling to `RunGravityZEstimationSequence()`, you must initialize the array of 16 floats with the values found in the text file `ParametersOptimal_Z.txt`.

The third function to be called is `SetGravityType()`, which takes for input either `OPTIMAL` (activates the Optimal gravity compensation mode) or `MANUAL_INPUT` (activates the `MANUAL` gravity compensation mode). This function must also be called at every reboot.

Optimal mode code example

This section gives an example of C++ code to use Optimal mode with the arm.

The following sample C++ program demonstrates how to use Optimal mode

```
1      #include <Windows.h>
2      #include <conio.h>
3      #include <iostream>
4      #include "Lib_Examples\CommandLayer.h"
5      #include "Lib_Examples\CommunicationLayerWindows.h"
6      #include "Lib_Examples\KinovaTypes.h"
7      #include <fstream>
8
9
10     using namespace std;
11
12     //A handle to the API.
13     HINSTANCE commandLayer_handle;
14
15     //Function pointers to the functions we need
16     int(*MyInitAPI)();
17     int(*MyCloseAPI)();
18     int(*MyGetAngularCommand)(AngularPosition &Response);
19
20     int(*MyRunGravityZEstimationSequence7DOF)(ROBOT_TYPE type, float
21     OptimalzParam[OPTIMAL_Z_PARAM_SIZE_7DOF]);
22     int(*MySetGravityOptimalParameter)(float
23     Command[GRAVITY_PARAM_SIZE]);
24     int(*MySetGravityType)(GRAVITY_TYPE Type);
25
26     int main(int argc, char* argv[]) {
27
28         //We load the API.
29         commandLayer_handle = LoadLibrary(L"CommandLayerWindows.dll");
30         AngularPosition current;
31         int result;
32         int programResult = 0;
33
34         //We load the functions from the library
35         MyInitAPI = (int(*)()) GetProcAddress(commandLayer_handle, "InitAPI");
36         MyCloseAPI = (int(*)()) GetProcAddress(commandLayer_handle, "CloseAPI");
37         MyGetAngularCommand = (int(*)(AngularPosition &Response))
38             GetProcAddress(commandLayer_handle, "GetAngularCommand");
39         MyRunGravityZEstimationSequence7DOF = (int(*)(ROBOT_TYPE type, float
40         OptimalzParam[OPTIMAL_Z_PARAM_SIZE]))
41             GetProcAddress(commandLayer_handle,
42                 "RunGravityZEstimationSequence7DOF");
43         MySetGravityOptimalParameter = (int*)(float
44             Command[GRAVITY_PARAM_SIZE])) GetProcAddress(commandLayer_handle,
45             "SetGravityOptimalZParam");
46         MySetGravityType = (int*)(GRAVITY_TYPE Type))
```

```
46     GetProcAddress(commandLayer_handle, "SetGravityType");  
47  
48     //Verify that all functions have been loaded correctly  
49     if ((MyInitAPI == NULL) || (MyCloseAPI == NULL) ||  
50         (MyGetAngularCommand == NULL) ||  
51         (MySetGravityOptimalParameter == NULL) ||  
52         (MyRunGravityZEstimationSequence == NULL) ||  
53         (MySetGravityType==NULL)) {  
54         cout << "* * * E R R O R D U R I N G I N I T I A L I Z A T I O N * * *" <<  
55         endl;  
56         programResult = 0;  
57     }  
58     else {  
59         cout << "I N I T I A L I Z A T I O N C O M P L E T E D" << endl << endl;  
60         int result = (*MyInitAPI)();  
61         int resultComm;  
62         AngularPosition DataCommand;  
63  
64         // Get the angular command to test the communication with the robot  
65         resultComm = MyGetAngularCommand(DataCommand);  
66         KinovaDevice list[MAX_KINOVA_DEVICE];  
67  
68         // If the API is initialized and the communication with the robot is working  
69         if (result == 1 && resultComm == 1) {  
70             // Choose robot type  
71             ROBOT_TYPE type = SPHERICAL_7DOF_SERVICE;  
72             float OptimalzParam[OPTIMAL_Z_PARAM_SIZE_7DOF];  
73  
74             // Run identification sequence  
75             cout << "Running gravity parameters estimation trajectory..." << endl;  
76             MyRunGravityZEstimationSequence7DOF (type, OptimalzParam);  
77             MySetGravityOptimalParameter(OptimalzParam);  
78  
79             // informs the robot on the new optimal gravity parameters  
80             MySetGravityType(OPTIMAL); //sets the gravity compensation mode to  
81             Optimal  
82         }  
83         cout << endl << "C L O S I N G A P I" << endl;  
84         result = (*MyCloseAPI)();  
85         programResult = 1;  
86     }  
87     FreeLibrary(commandLayer_handle);  
88  
89     return programResult;  
90 }
```

Important considerations for setting Optimal mode



IMPORTANT: Before launching the calibration trajectory with the function `RunGravityZEstimationSequence()`, please ensure the robot has enough space to move. The top 1.5 meter-radius spherical space around the base should be free of obstacles. The hand should technically never go below the base, so the base can be fixed on a table. For ideal results, you can attach the base on a small stand so the space below the robot (except the space below the base) is free.

Before launching the calibration trajectory to find the optimal gravity parameters (`RunGravityZEstimationSequence()` function), [reset the torque sensors zero value](#). Whenever your torque sensors readings seem off, start by resetting the torque sensors zero value, especially if you want to use admittance or torque mode.

The Optimal gravity compensation mode works only when the arm is standing on a flat surface (the gravity vector is in the direction [0, 0, -9.81]).

The robot does not save the optimal gravity parameters. If you want to use the Optimal gravity compensation mode, you will have to load the parameters from the 'ParametersOptimal_Z.txt' text file, send them to the robot using the `SetOptimalZParam()` function and activate the Optimal mode using the `SetGravityType()` function after each reboot.

Specific Utilization Limitations

The robotic arm has certain utilization limitations that research users should be cautious of. This includes:

- Recommended maximum actuators utilization
- Software position limitations of actuators
- Software position limitations of fingers

Recommended maximum actuators utilization

| | Big actuators (75mm) | Small actuators (58mm) | Fingers actuators |
|----------------------------|---------------------------------|-----------------------------------|------------------------------|
| Maximum RPM | 6 RPM | 8 RPM | 600 RPM |
| Maximum command / sec | 36° / sec | 48° / sec | 300 mm / sec 10800° / sec |
| Maximum repetitive current | 1.5A | 1.6A | 1.4 |
| Maximum temperature | 80°C | 80°C | 80°C |

Utilization over these maximum recommended parameters may affect lifetime of the arm and its modules. Please refer to the specification sheet information for your particular arm configuration for additional information

Software position limitations of actuators

This section provides a reference of software position limitations of various robotic arm configurations.

The following limitations indicate the software limitations that are present in the robotic arm controller base to ensure safety of the robot. These limitations are there to protect the arm and its environment.

When moving the actuators, the following minimum and maximum positions should be followed. If the command sent to any of these actuators goes further than these values, the actuators will stop moving.

Table 15: 4 DOF software limitations

| Joint | Min(°) | Max(°) |
|-------|---------|--------|
| 1 | -10 000 | 10 000 |
| 2 | 47 | 313 |
| 3 | 19 | 341 |
| 4 | -10 000 | 10 000 |

Table 16: 6DOF curved wrist software limitations

| Joint | Min(°) | Max(°) |
|-------|---------|--------|
| 1 | -10 000 | 10 000 |
| 2 | 50 | 310 |
| 3 | 19 | 341 |
| 4 | -10 000 | 10 000 |
| 5 | -10 000 | 10 000 |
| 6 | -10 000 | 10 000 |

Table 17: 6 DOF Spherical software limitations

| Joint | Min(°) | Max(°) |
|-------|---------|--------|
| 1 | -10 000 | 10 000 |
| 2 | 47 | 313 |
| 3 | 19 | 341 |
| 4 | -10 000 | 10 000 |
| 5 | 65 | 295 |
| 6 | -10 000 | 10 000 |

Table 18: 7 DOF Spherical software limitations

| Joint | Min(°) | Max(°) |
|-------|---------|--------|
| 1 | -10 000 | 10 000 |

| Joint | Min(°) | Max(°) |
|-------|---------|--------|
| 2 | 47 | 313 |
| 3 | -10 000 | 10 000 |
| 4 | 30 | 330 |
| 5 | -10 000 | 10 000 |
| 6 | 65 | 295 |
| 7 | -10 000 | 10 000 |

These limitations are in angular mode. In Cartesian mode, the limits for joints 2 and 3 are different.

For right-hand mode:

- Joint 2 - **min**: 140°, **max**: same as angular
- Joint 3 - **min**: same as angular, **max**: 165°

For left-hand mode:

- Joint 2 - **min**: same as angular, **max**: 220°
- Joint 3 - **min**: 195°, **max**: same as angular.

Software position limitations of fingers

When moving the robotic arm fingers, the following minimum and maximum positions should be respected. If the command sent to any of these fingers goes further than these values, the fingers will stop moving.

| Finger # | Minimum | Maximum |
|----------|-----------|-----------------|
| 1 | 0 mm (0°) | 18.9 mm (6800°) |
| 2 | 0 mm (0°) | 18.9 mm (6800°) |
| 3 | 0 mm (0°) | 18.9 mm (6800°) |

Advanced Configuration

This section describes considerations for the arm position indexation process.

Position indexation



Figure 6: Set zero positions joint angles

Resetting torque sensors zero value

In order to reset the torque sensors zero, you must first place the arm in a position where gravity does not influence the joint torques. The set zero position is good but it is suggested to use the positions as indicated in the table below since this position also limits perpendicular torques on the actuators.

Table 19: Recommended reset positions

| Config | Joint #1 angle | Joint #2 angle | Joint #3 angle | Joint #4 angle | Joint #5 angle | Joint #6 angle | Joint #7 angle |
|--------------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 4 DOF | * | 180 | 180 | 0 | N/A | N/A | N/A |
| 6 DOF curved wrist | * | 180 | 180 | 0 | 0 | 180 | N/A |
| 6 DOF spherical | * | 180 | 180 | 0 | 0 | 180 | N/A |
| 7 DOF spherical | * | 180 | * | 180 | * | 180 | * |

* Any angle

Kinematics Parameters

There are several sets of useful kinematic parameters:

- Basic geometric parameters of the arm
- Classic Denavit-Hartenberg (DH) parameters
- Directions of joints in angular space
- Inertial parameters

Basic geometric parameters - 4 DOF

This section describes the basic geometric parameters of the 4 DOF configuration.

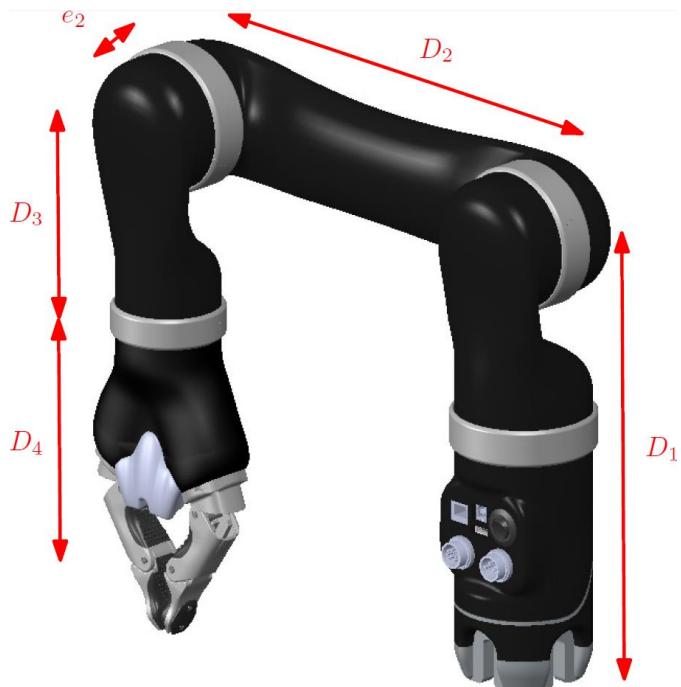
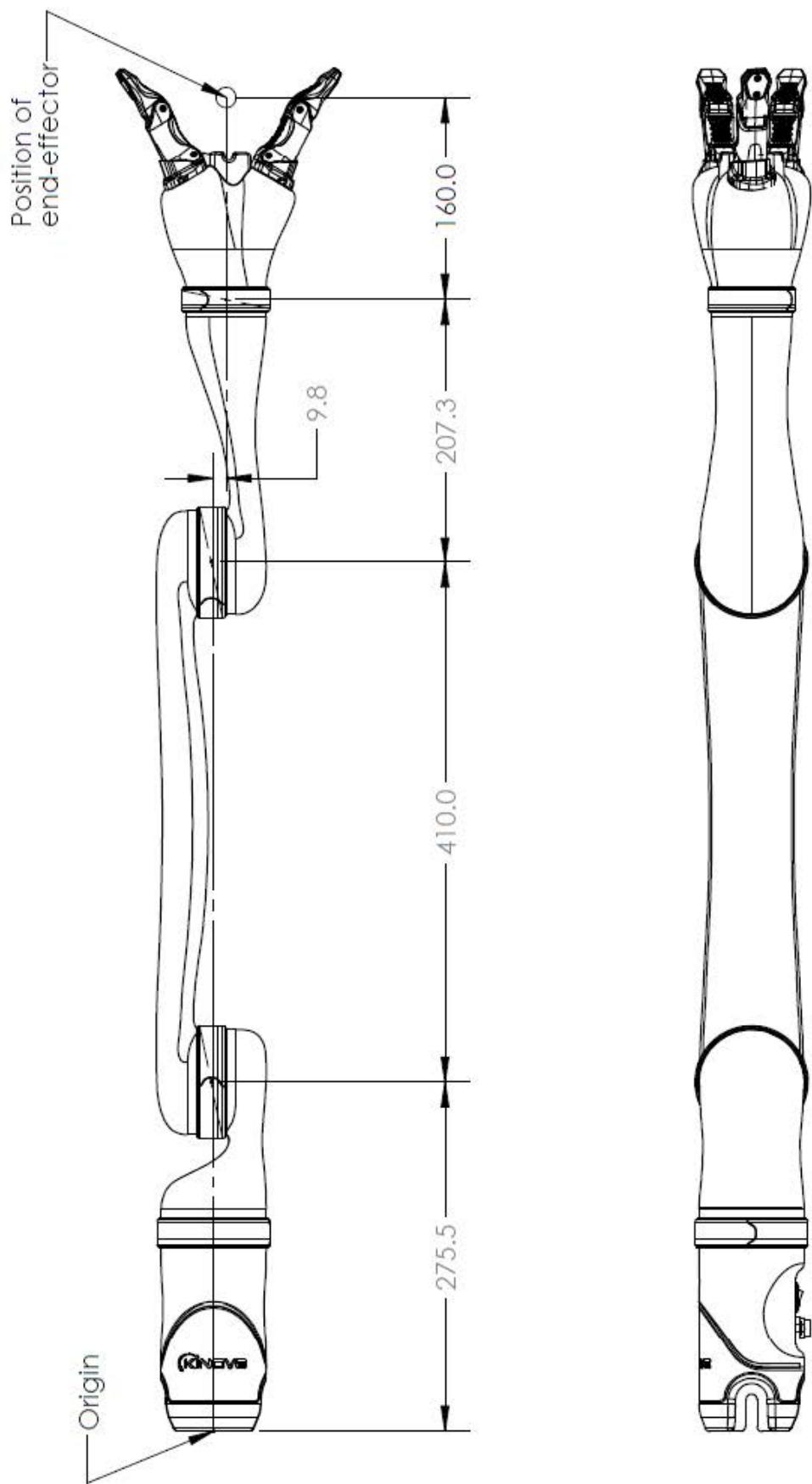


Table 20: 4 DOF basic geometric parameters

| Parameter | Description | Length (m) |
|-----------|---|------------|
| D1 | Base to shoulder | 0.2755 |
| D2 | Upper arm length (shoulder to elbow) | 0.4100 |
| D3 | Forearm length (elbow to wrist) | 0.2073 |
| D4 | First wrist length (center of actuator 4 to center of actuator 5) | 0.1600 |
| e2 | Joint 3-4 lateral offset | 0.0098 |



Basic geometric parameters - 6 DOF curved wrist

This section describes the basic geometric parameters of the 6 DOF curved wrist configuration.

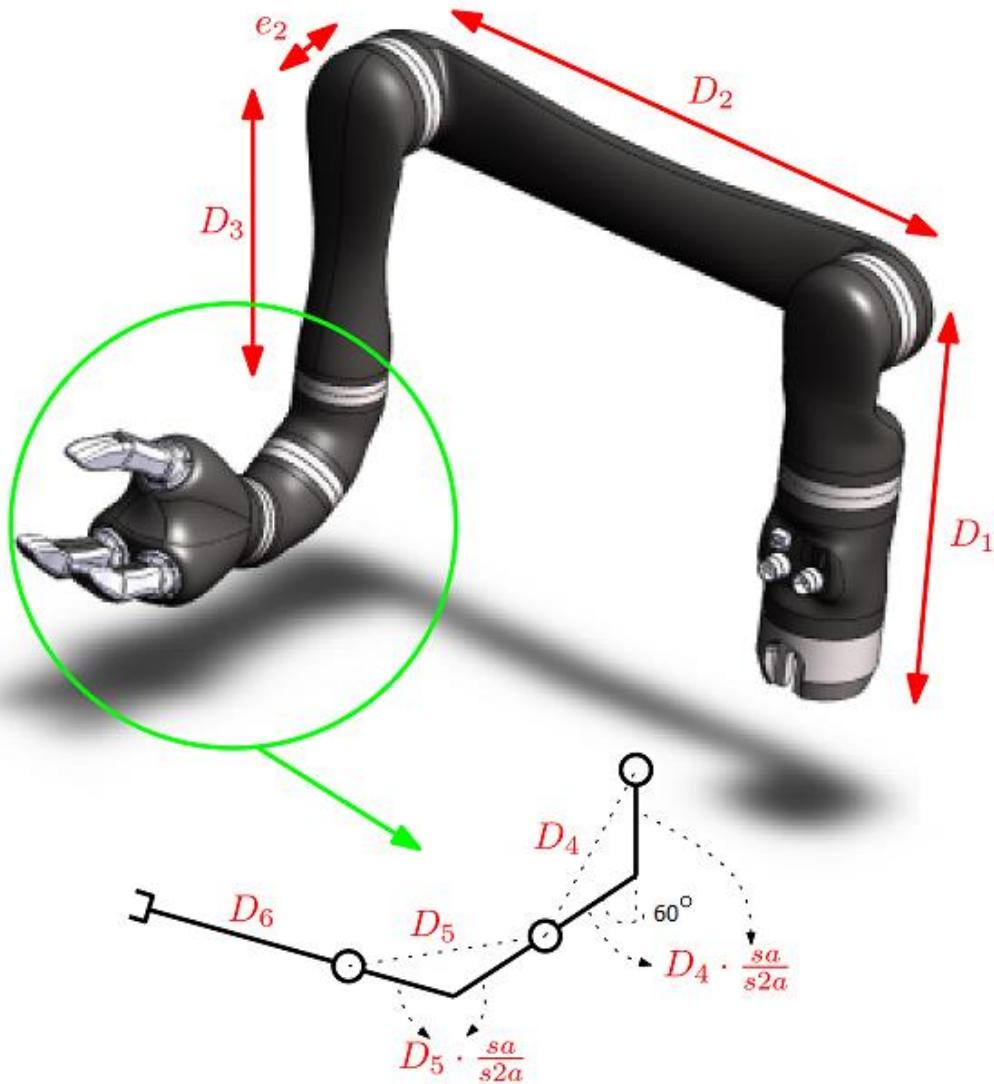
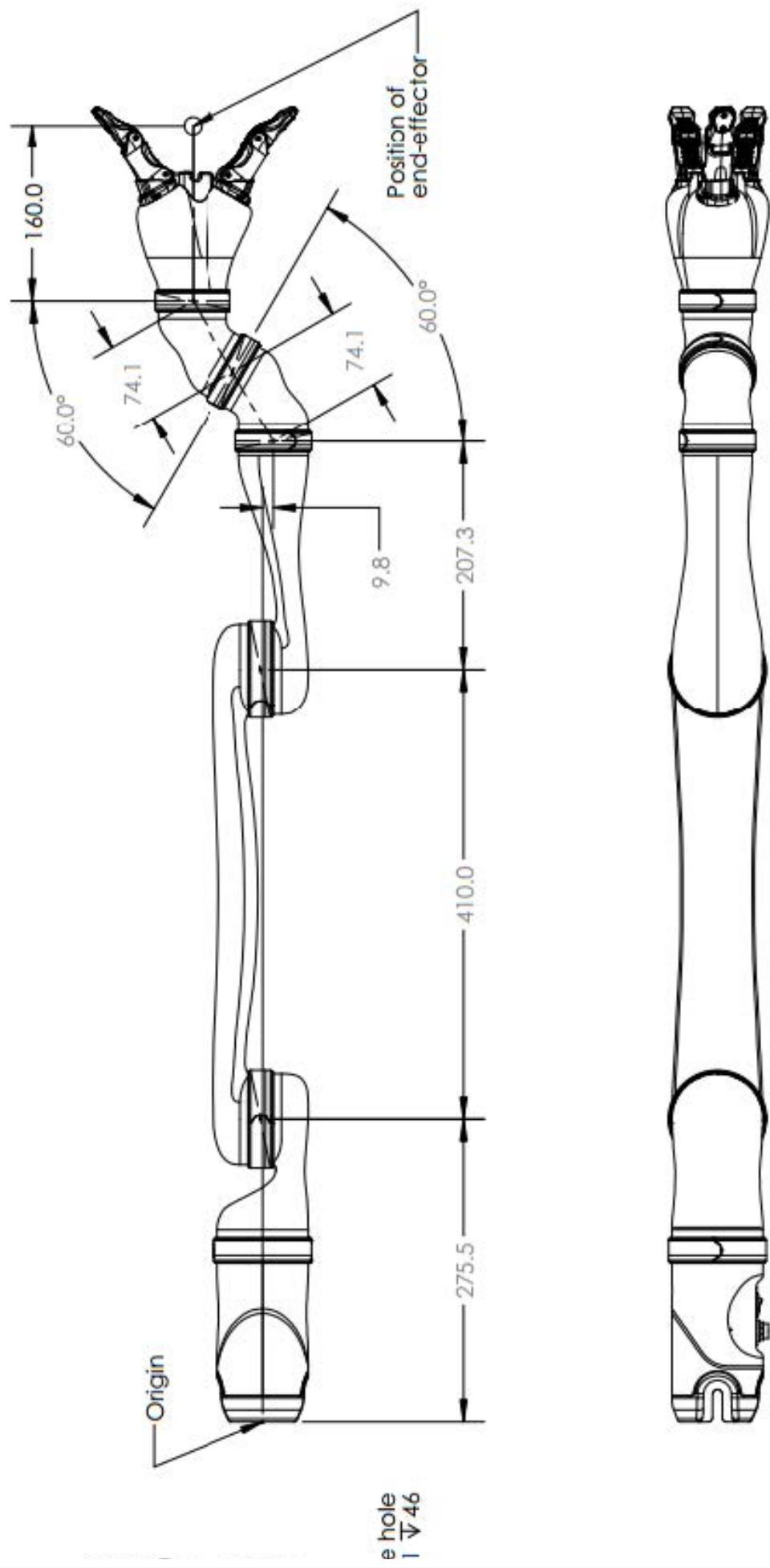


Table 21: 6 DOF curved wrist basic geometric parameters

| Parameter | Description | Length (m) |
|-----------|--|------------|
| D1 | Base to shoulder | 0.2755 |
| D2 | Upper arm length (shoulder to elbow) | 0.4100 |
| D3 | Forearm length (elbow to wrist) | 0.2073 |
| D4 | First wrist length (center of actuator 4 to center of actuator 5) | 0.0741 |
| D5 | Second wrist length (center of actuator 5 to center of actuator 6) | 0.0741 |

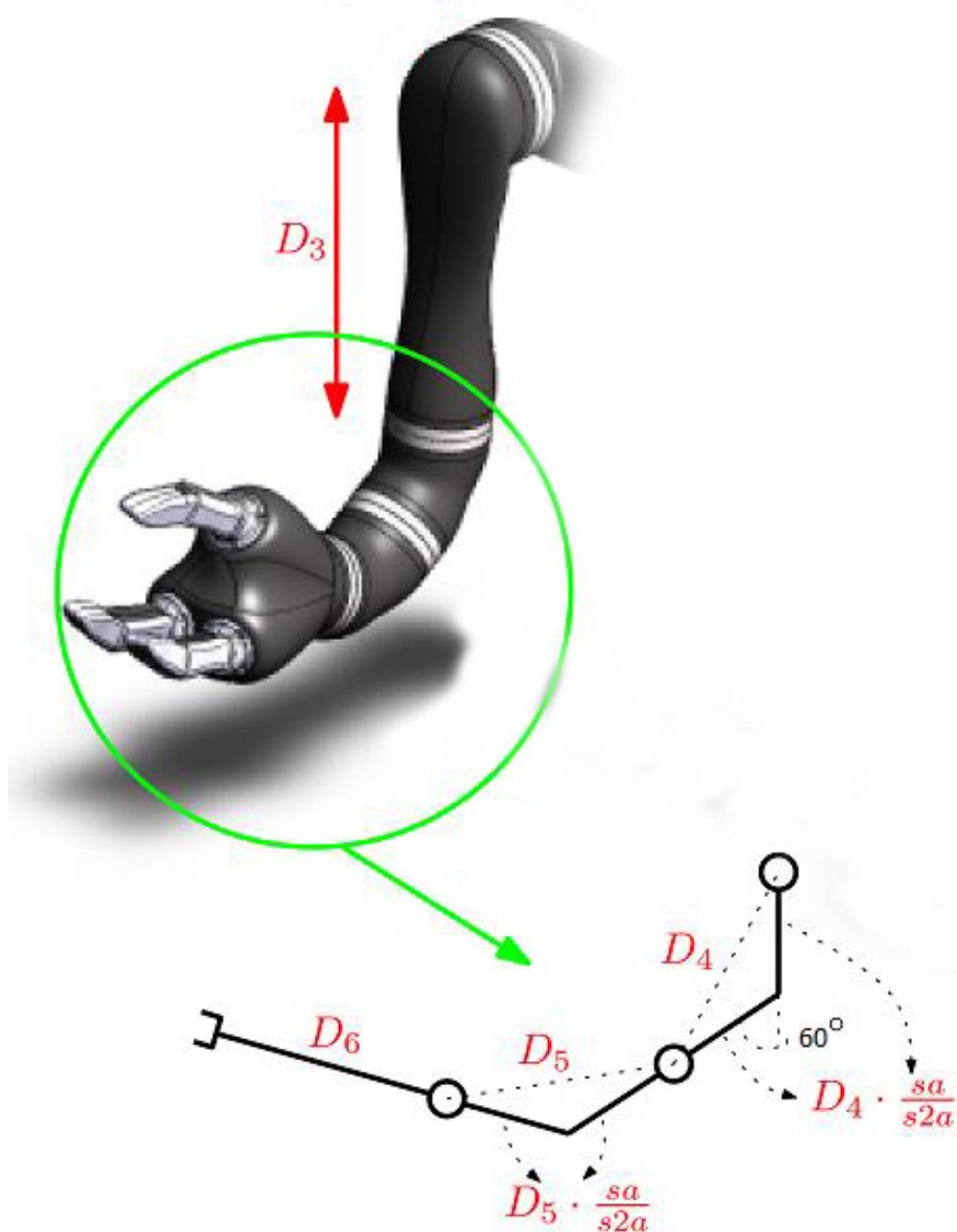
| Parameter | Description | Length (m) |
|-----------|-----------------------------|------------|
| D6 | Wrist to center of the hand | 0.1600 |
| e2 | Joint 3-4 lateral offset | 0.0098 |



Alternate geometric parameters - 6 DOF curved wrist

This section describes alternate parameters that are useful for describing the geometry for kinematics of the 6 DOF curved wrist configuration.

The kinematics of the 6 DOF curved wrist configuration are more complicated than for a spherical wrist due to the more complicated geometry. To simplify the analysis, it is useful to break down each of the two curved wrist segments into two component straight-line sub-segments of equal length, with the second sub-segment angled 60° from the first.



In this way, the arm from the elbow to the center of the hand can be analyzed as three straight-line segments:

- d4b - distance from elbow to end of first sub-segment of first wrist segment
- d5b - distance from end of first sub-segment of first wrist segment to end of first sub-segment of second wrist segment
- d6b - distance from end of second sub-segment of second wrist segment to center of hand

Table 22: Alternate parameters

| Parameter | Description | Value |
|-----------|--|-------------------------------------|
| aa | Half of the angle of curvature of each wrist segment (60°), measured in radians | $(30.0 * \pi) / 180.0$ |
| sa | Sine of half the angle of curvature of wrist segment | $\sin(aa)$ |
| s2a | Sine of angle of curvature of wrist segment | $\sin(2*aa)$ |
| d4b | Length of straight-line segment from elbow to end of first sub-segment of first wrist segment. | $D3 + (sa / s2a) * D4$ |
| d5b | Length of straight-line segment consisting of second sub-segment of first wrist segment and first sub-segment of second wrist segment | $(sa / s2a) * D4 + (sa / s2a) * D5$ |
| d6b | Length of straight-line segment consisting of second sub-segment of second wrist segment and distance from wrist to the center of the hand | $(sa / s2a) * D5 + D6$ |

The [DH parameters](#) for the lower part of the robotic arm are most naturally expressed in terms of these alternate parameters.

Basic geometric parameters - 6 DOF spherical wrist

This section describes the basic geometric parameters of the 6 DOF spherical wrist configuration.

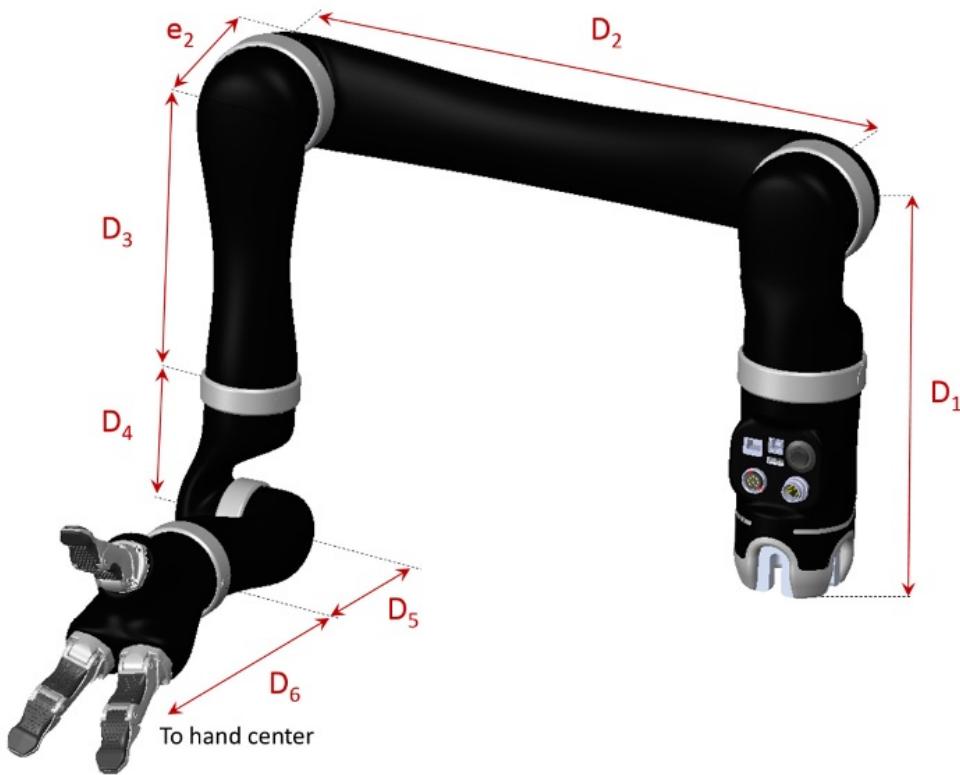


Table 23: 6 DOF spherical basic geometric parameters

| Parameter | Description | Length (m) |
|-----------|-------------------------------------|------------|
| D1 | Base to shoulder | 0.2755 |
| D2 | Upper length (shoulder to elbow) | 0.4100 |
| D3 | Forearm length (elbow to wrist) | 0.2073 |
| D4 | First wrist length | 0.0741 |
| D5 | Second wrist length | 0.0741 |
| D6 | Wrist to center of the hand | 0.1600 |
| e2 | Joint 3-4 lateral offset | 0.0098 |

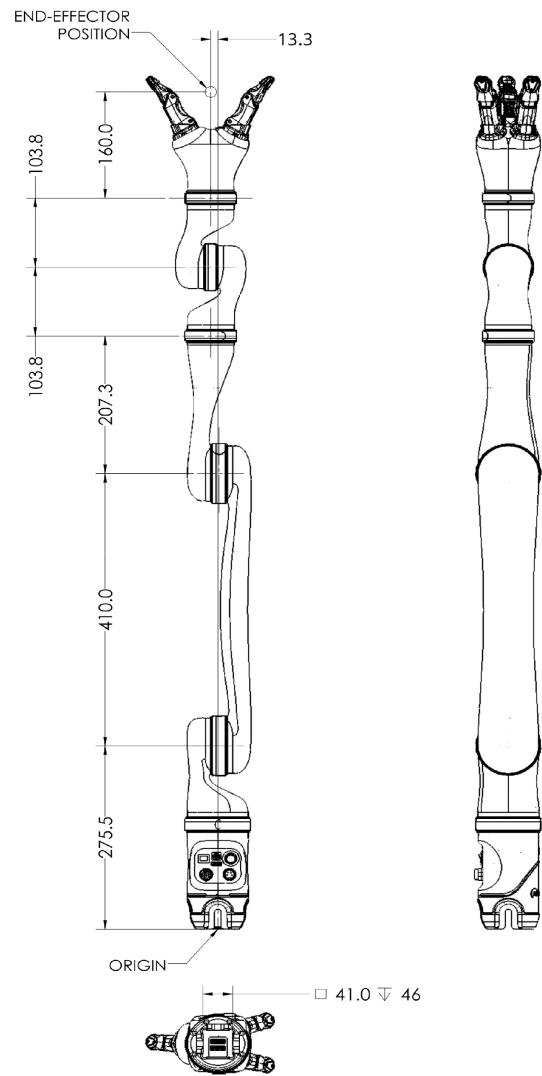


Figure 9: Detailed 6 DOF spherical robot length values (units in mm)

Basic geometric parameters - 7 DOF spherical wrist

This section describes the basic geometric parameters of the 7 DOF spherical wrist.

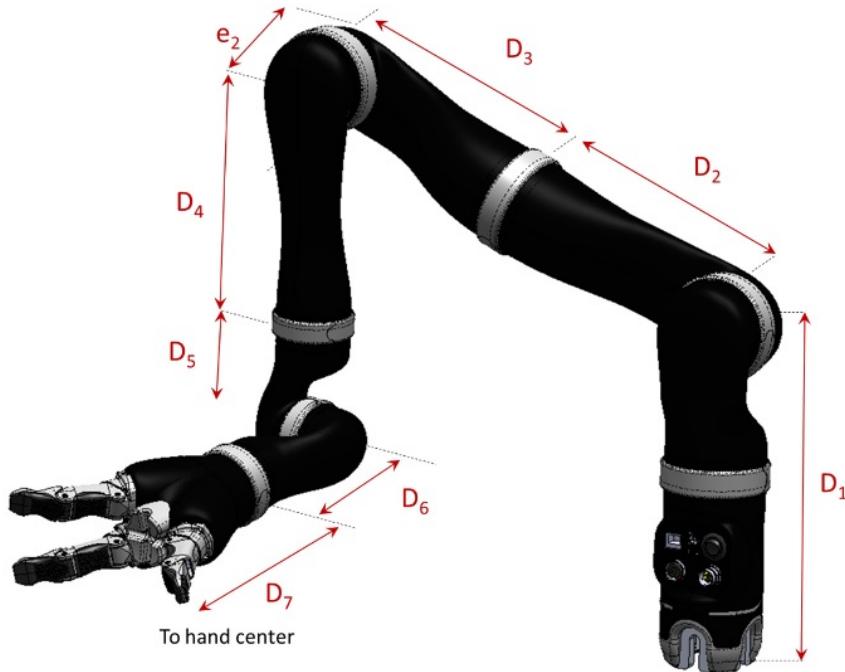


Table 24: 7 DOF spherical basic geometric parameters

| Parameter | Description | Length (m) |
|-----------|------------------------------------|------------|
| D1 | Base to shoulder | 0.2755 |
| D2 | First half upper arm length | 0.2050 |
| D3 | Second half upper arm length | 0.2050 |
| D4 | Forearm length (elbow to wrist) | 0.2073 |
| D5 | First wrist length | 0.1038 |
| D6 | Second wrist length | 0.1038 |
| D7 | Wrist to center of the hand | 0.1600 |
| e2 | Joint 3-4 lateral offset | 0.0098 |

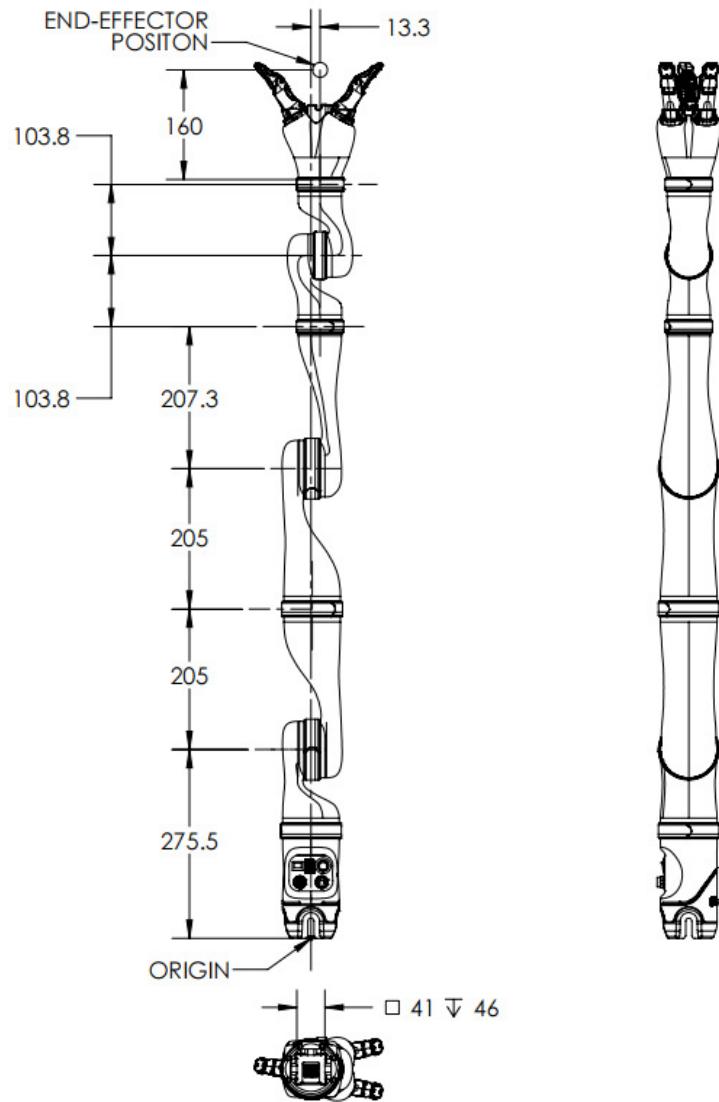


Figure 10: Detailed 7 DOF spherical robot length values (units in mm)

Classic DH parameters - 4 DOF

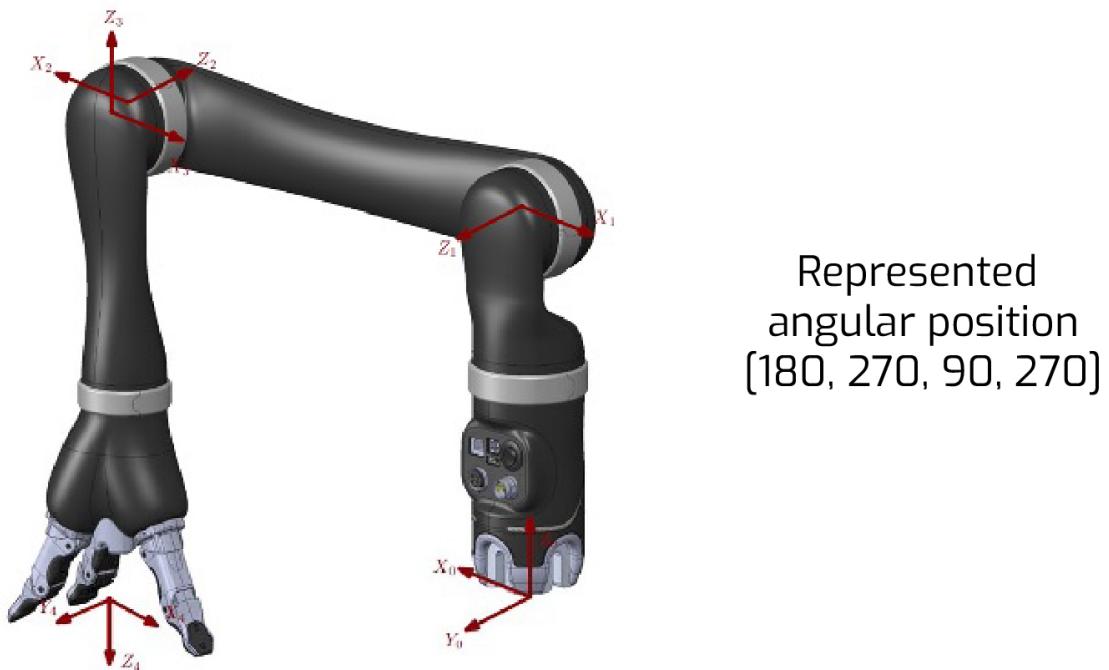
This section describes the Classic DH parameters for the 4 DOF.

Table 25: 4 DOF DH parameters

| i | a_{i-1} | d_i | θ_i |
|---|-----------|-------|------------|
| 1 | $\pi / 2$ | 0 | D1 |
| 2 | π | D2 | 0 |
| 3 | $\pi / 2$ | 0 | -e2 |
| 4 | π | 0 | D3 + D4 |

Table 26: Transformation from DH algorithm to robotic arm physical angles

| |
|----------------------------------|
| Q1(physical) = -Q1(DH algo) |
| Q2(physical) = Q2(DH algo) + 90 |
| Q3(physical) = Q3(DH algo) - 90 |
| Q4(physical) = Q4(DH algo) - 270 |



Classic DH parameters - 6 DOF curved wrist

This section describes the Classic DH parameters for the 6 DOF with curved wrist.

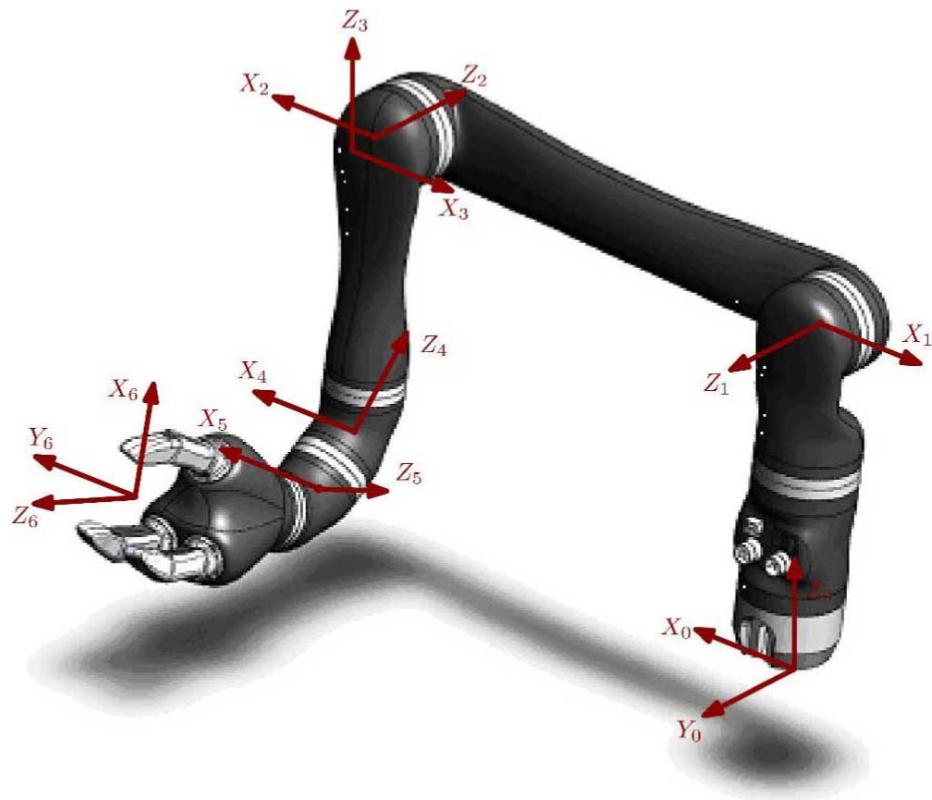
The DH parameters for the 6 DOF curved arm robotic arm are naturally expressed in terms of [alternate parameters](#).

Table 27: 6 DOF curved wrist DH parameters

| i | α_{i-1} | a_{i-1} | d_i | θ_i |
|---|----------------|-----------|-------|------------|
| 1 | $\pi / 2$ | 0 | D1 | q1 |
| 2 | π | D2 | 0 | q2 |
| 3 | $\pi / 2$ | 0 | -e2 | q3 |
| 4 | $2^{*}aa$ | 0 | -d4b | q4 |
| 5 | $2^{*}aa$ | 0 | -d5b | q5 |
| 6 | π | 0 | -d6b | q6 |

Table 28: Transformation from DH algorithm to robotic arm physical angles

| |
|----------------------------------|
| Q1(physical) = -Q1(DH algo) |
| Q2(physical) = Q2(DH algo) + 90 |
| Q3(physical) = Q3(DH algo) - 90 |
| Q4(physical) = Q4(DH algo) |
| Q5(physical) = Q5(DH algo) + 180 |
| Q6(physical) = Q6(DH algo) - 90 |



Represented
angular position:
[180, 270, 90, 180, 180, 0]

Classic DH parameters - 6 DOF spherical

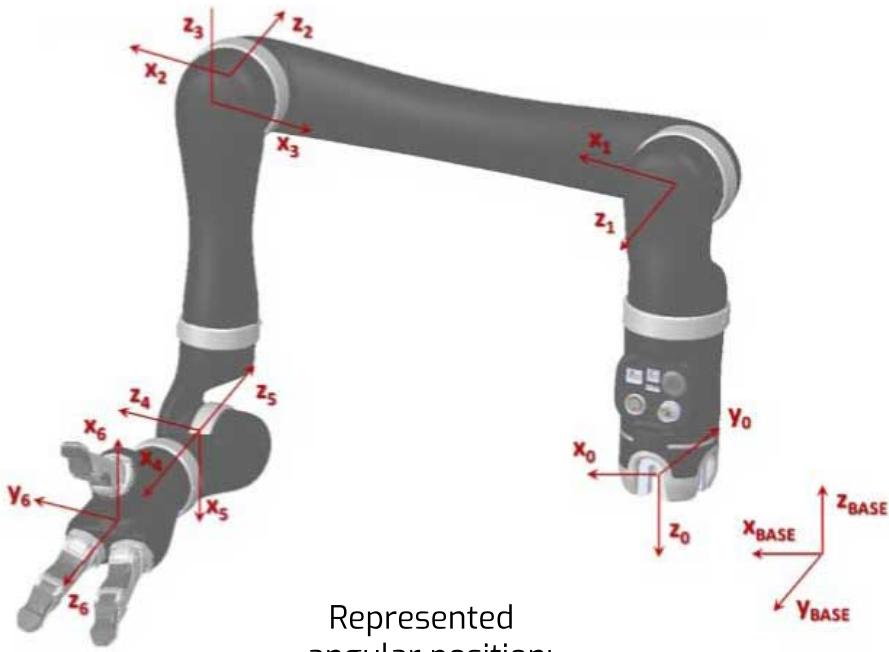
This section describes the Classic DH parameters for the 6 DOF spherical.

Table 29: 6DOF spherical DH parameters

| i | α_{i-1} | a_{i-1} | d_i | θ_i |
|---|----------------|-----------|------------|------------|
| 1 | $\pi / 2$ | 0 | D1 | q1 |
| 2 | π | D2 | 0 | q2 |
| 3 | $\pi / 2$ | 0 | -e2 | q3 |
| 4 | $\pi / 2$ | 0 | -(D3 + D4) | q4 |
| 5 | $\pi / 2$ | 0 | 0 | q5 |
| 6 | π | 0 | -(D5 + D6) | q6 |

Table 30: Transformation from DH algorithm to robotic arm physical angles

| |
|-----------------------------------|
| Q1(physical) = -Q1(DH algo) + 180 |
| Q2(physical) = Q2(DH algo) - 90 |
| Q3(physical) = Q3(DH algo) - 90 |
| Q4(physical) = Q4(DH algo) |
| Q5(physical) = Q5(DH algo) |
| Q6(physical) = Q6(DH algo) + 90 |



Represented
angular position:
[180, 270, 90, 270, 270, 270]

Transformation matrix
from BASE
to 0 axis system:

$$R = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

Classic DH parameters - 7 DOF spherical

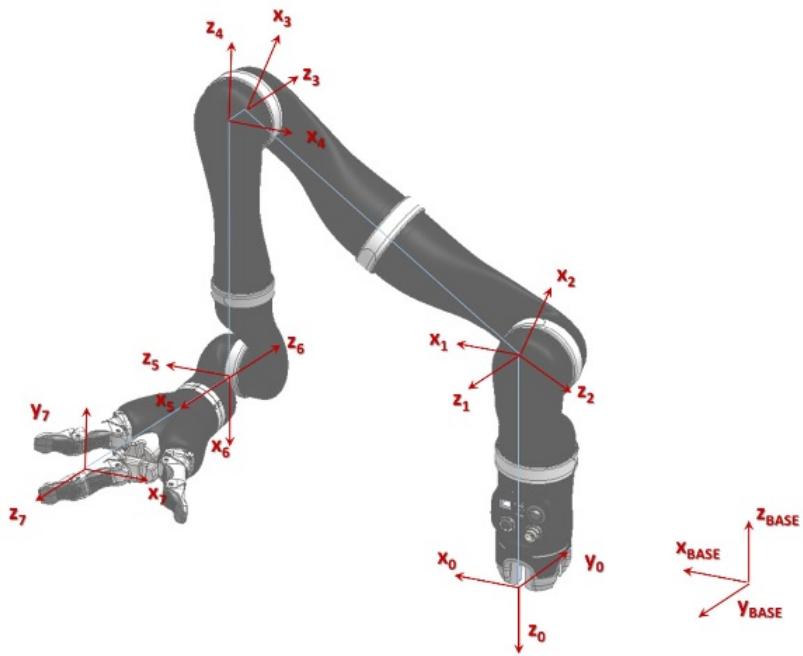
This section describes the Classic DH parameters for the 7 DOF spherical.

Table 31: 7 DOF spherical DH parameters

| i | α_{i-1} | a_{i-1} | d_i | θ_i |
|---|----------------|-----------|------------|------------|
| 1 | $\pi / 2$ | 0 | -D1 | q1 |
| 2 | $\pi / 2$ | 0 | 0 | q2 |
| 3 | $\pi / 2$ | 0 | -(D2 + D3) | q3 |
| 4 | $\pi / 2$ | 0 | -e2 | q4 |
| 5 | $\pi / 2$ | 0 | -(D4 + D5) | q5 |
| 6 | $\pi / 2$ | 0 | 0 | q6 |
| 7 | π | 0 | -(D6 + D7) | q7 |

Table 32: Transformation from DH algorithm to robotic arm physical angles

| |
|----------------------------------|
| Q1(physical) = Q1(DH algo) + 180 |
| Q2(physical) = Q2(DH algo) |
| Q3(physical) = Q3(DH algo) |
| Q4(physical) = Q4(DH algo) |
| Q5(physical) = Q5(DH algo) |
| Q6(physical) = Q6(DH algo) + 90 |



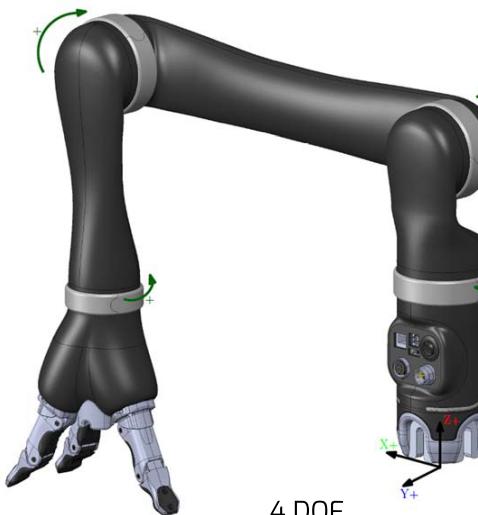
Transformation matrix
from BASE
to 0 axis system:

$$R = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

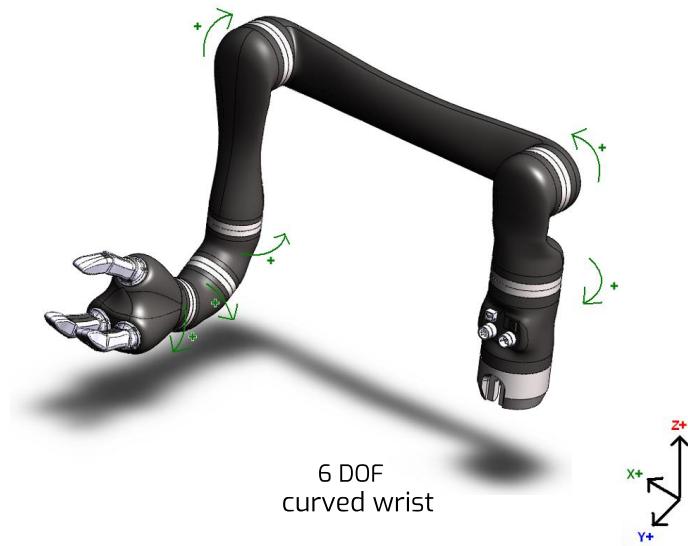
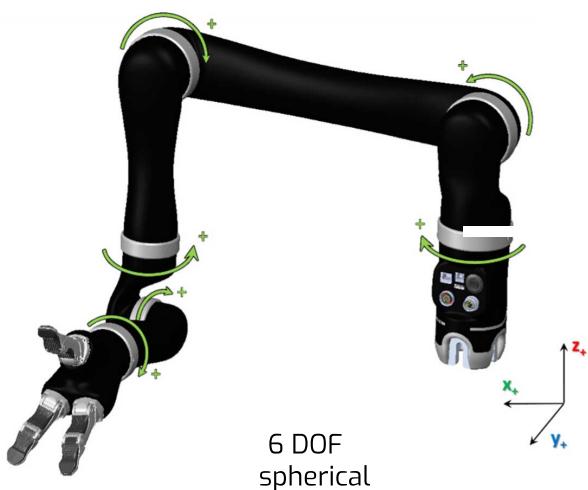
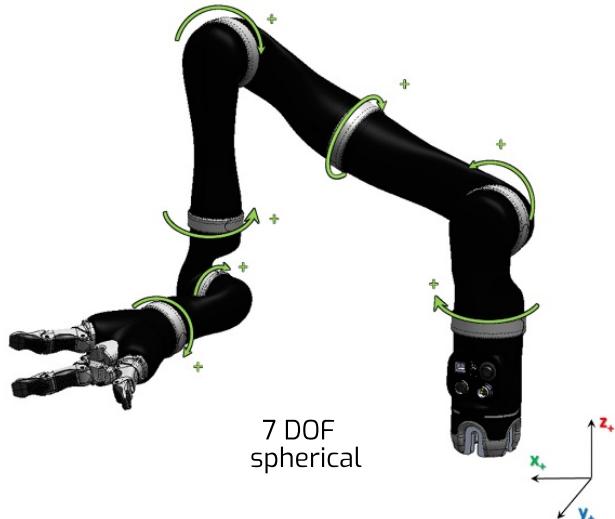
Represented
angular position:
[180, 240, 0, 60, 90, 90, 180]

Directions of joints in angular space

The following image represents the positive direction of rotation of each actuator for the different configurations of the robotic arm:



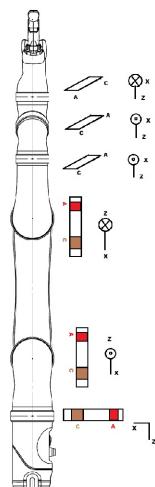
4 DOF

6 DOF
curved wrist6 DOF
spherical7 DOF
spherical

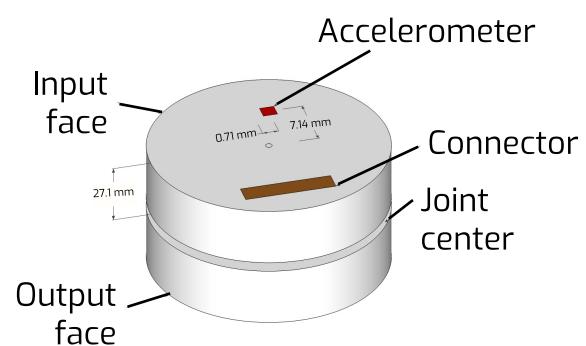
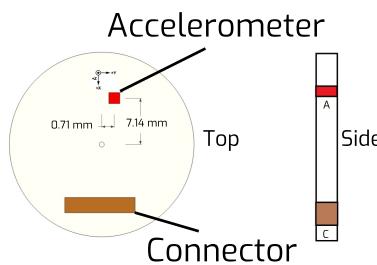
Advanced sensors information

This section has information about the accelerometers in each actuator.

The image below shows information about the accelerometers in each joint actuator.



6 DOF arm shown for illustration purposes



Normal use definition

This section describes the normal use of the robotic arm.

The definition of a normal use of the robotic arm includes that you can lift, push, pull or manipulate a maximum load of:

- **Continuously** 2.6 kg from minimum to middle reach for 6 DOF, 6 DOF-S, and 7 DOF-S (45-49 cm distance between the actuator #2 and the load, depending on the configuration) and 4.4 kg from minimum to middle reach for the 4 DOF (35 cm distance between actuator #2 and the load).
- **Temporary** 2.2 kg from middle to full reach for 6 DOF, 6 DOF-S, 7 DOF-S (90-98 cm distance between actuator #2 and the load, depending on the configuration) and 3.5 kg from middle to full reach for 4DOF (75 cm distance between actuator #2 and the load).

The arm is designed to be able to hold objects in the environment of the user, but it is a manipulator that in some positions and loads near the maximum reach and maximum loads holds for a long period, it can heat. When this occurs, before overheating and being dangerous for either the user or the arm, red lights on the joystick will blink. This is a warning. Simply put down any object in the gripper, and bring back the arm to the HOME or RETRACTED positions and wait until the warning goes away before using the arm.

If you don't use a Joystick in your application, make sure to read all the error statuses and temperature of all actuators modules via the API to ensure that they do not go higher than recommended parameters. If this occurs, the arm should be held in an idle position near the base for a certain time without any object in the gripper to cool down the arm.

 When lifting weight near the maximum load and reach, if the red lights of the controller blinks, put down the object in the gripper, and bring back the arm to HOME or RETRACTED position and wait until the warning goes away before using it.

Note: During normal operation, the joints are subject to heating. The joints are normally covered with plastic rings which will protect the user from any danger that may be occurred by the heating of the metal parts.

The fingers of the arm are made flexible in order to protect the internal mechanism. When using the fingers to push on objects, the user must take special care not flex the fingers beyond their maximal opening as this could damage the internal mechanism.

 Do not force the fingers beyond their maximal opening as this could damage some internal components.

Electromagnetic interference from radio wave sources

This section describes electromagnetic interference considerations for the JACO robotic arm.

Even if the product complies with all relevant standards, your arm may still be susceptible to electromagnetic interference (EMI), which is interfering electromagnetic energy (EM) emitted from sources such as radio stations, TV stations, amateur radio (HAM) transmitters, two way radios, and cellular phones. The interference (from radio wave sources) can cause the product to stop moving for a period of 10 seconds. In this case, the device will simply re-initialize and you will be able to continue to use it. In extremely rare case, it can also permanently damage the control system.

The intensity of the interfering EM energy can be measured in volts per meter (V/m). The product can resist EMI up to certain intensity. This is called "immunity level". The higher the immunity level is, the greater is the protection. At this time, current technology is capable of achieving at least a 20 V/m immunity level, which would provide useful protection from the more common sources of radiated EMI.

There are a number of sources of relatively intense electromagnetic fields in the everyday environment. Some of these sources are obvious and easy to avoid. Others are not apparent and exposure is unavoidable. However, we believe that by following the warnings listed below, your risk to EMI will be minimized.

The sources of radiated EMI can be broadly classified into three types:

1. Gripper-held portable transceivers (e.g. transmitters-receivers with the antenna mounted directly on the transmitting unit, including citizens band (CB) radios, walkie-talkie, security, fire and police transceivers, cellular phones, and other personal communication devices). Some cellular phones and similar devices transmit signals while they are ON, even if not being actively used.
2. Medium-range mobile transceivers, such as those used in police cars, fire trucks, ambulances and taxis. These usually have the antenna mounted on the outside of the vehicle.
3. Long-range transmitters and transceivers, such as commercial broadcast transmitters (radio and TV broadcast antenna towers) and amateur (HAM) radios. Other types of gripper-held devices, such as cordless phones, laptop computers, AM/FM radios, TV sets, CD players, cassette players, and small appliances, such as electric shavers and hair dryers, so far as we know, are not likely to cause EMI problems to your device.

Because EM energy rapidly becomes more intense as one move closer to the transmitting antenna (source), the EM fields from gripper-held radio wave sources (transceivers) are of special concern. It is possible to unintentionally bring high levels of EM energy very close to the control system while using these sorts of devices. Therefore, the warnings listed below are recommended to reduce the effects of possible interference with the control system.

⚠ Do not operate gripper-held transceivers (transmitter's receivers), such as citizens band (CB) radios, or turn ON personal communication devices, such as cellular phones, while the device is turned ON.

⚠ Be aware of nearby transmitters, such as radio or TV stations, and try to avoid coming close to them.

⚠ Be aware that adding accessories or components, close to the device may make it more susceptible to EMI.

⚠ Report all incidents of unintended shut down to your local distributor, and note whether there is a source of EMI nearby.

Maintenance and Disposal

This section describes maintenance and disposal considerations.

Cleaning instructions

Only the external surfaces of the product may be cleaned. Cleaning may be done using a damp cloth and light detergent. The following described the steps for the cleaning the product:

- Prepare a water/soap preparation using a proportion of about 2ml of dish soap for 100ml of water
- Immerse a clean cotton cloth in the preparation
- Take out the cloth and wring out thoroughly
- Gently rub the external surface to be cleaned

 Do not wash more than three times per day.

 Do not immerse any part of the product under water or snow.

 The product is not intended to be sterile. No sterilization process should be applied to the product.

 Do not rub the external surfaces with abrasive materials.

Preventive Maintenance

The product requires no maintenance. Fingers should be cleaned and lubricated every 6 months.

 Refer all services to qualified service personnel. A service is required when the apparatus has been damaged in any way, for example if the power-supply cord or plug is damaged, if the product does not operate normally or has been dropped.

 There is no “home serviceable” part inside the product. Do not open..

Disposal



 The product contains parts that are deemed to be hazardous waste at the end of their life. For further information on recycling, contact your local recycling authority or local Kinova distributor. In any way, always dispose of product through a recognized agent.

Packing Materials

The product packing material can be disposed as recyclable material.

Metal parts

Metal parts can be disposed as recyclable scrap metal.

Electrical parts, circuit boards, and carbon fiber

Please contact your local distributor to have information regarding disposal of such parts. You can also address questions directly to Kinova through our website (see Contacting Support).

Troubleshooting / FAQs

This section gives guidance in troubleshooting common issues

Table 33:

| Issues | |
|--|---|
| My arm will not switch from position mode to torque mode | Make sure nothing is touching your robot when you try to switch to torque mode. Monitor your robot's torque readings, computed gravity-free torques and force/torque at the hand using the Kinova Development Center or API. Normally, gravity-free torques and force/torque at the hand should be low. If your torque readings seem unusual, redo your torque sensors zero calibration. Normally, a good position to switch from position mode to torque mode is the robot's Ready position (although any position is technically acceptable). |
| My arm keeps falling down in torque mode | Torque mode is very sensitive to gravity compensation. First, check that your torque sensors' zero was adequately calibrated. You can also follow the tips given in Appendix 5. If your robot keeps falling down, you can try adding a negative ($m < 0$) virtual mass using the Kinova Torque Console or API. * Note: If instead of falling down, your arm is always going up, you should follow the same tips but with adding a positive virtual mass at the end instead of a negative one. |

| Issues | |
|---|---|
| <p>I sent a trajectory in Cartesian mode and the robot stops at some point in the trajectory.</p> | <p>This problem can take several forms: the robot is in a random position when the trajectory is launched, the robot ‘reaches’ the first point but does not continue passed that. In this case, the robot never really reaches the first point, but goes towards it without reaching it; the robot starts the trajectory but gets stuck trying to reach a point; the robot takes a very unusual twist trying to reach a point. All of these problems are in fact one problem: the robot cannot reach the next trajectory point if it gets into singularity, self-collision or runs into a protection zone. “But I moved the robot with the joystick to create the trajectory”. When you are using the joystick, the robot receives a succession of small movements that eventually lead it to the points you have recorded in the trajectory. Besides, when the robot receives a joystick point (or Cartesian velocity point), it tries to reach it, then erases it and waits for the next point. When the robot receives a Cartesian trajectory point (or Cartesian position), it computes a straight line from its current position to the next trajectory point. Then it computes the next reachable point on this straight line considering maximum velocity and acceleration limits and gets deviated by its automatic self-collision and singularity avoidance algorithms. Finally, the robot verifies if the desired trajectory point was reached. If it did not reach the trajectory point, it repeats the process and it NEVER erases the trajectory point until it reaches it (within a certain error margin). Because the robot is not able to reach the next trajectory point while avoiding singularities and self-collision, and because this trajectory point does not get erased from the list, you get the impression that the robot either does not receive all the trajectory point or freezes in the middle of the trajectory.</p> <p>Now, what to do? Here are a few tips:</p> <ul style="list-style-type: none"> • Record more points in the trajectory, especially for difficult parts of the trajectory (will also reduce the risk of having your robot doing unusual moves between your trajectory points) • Familiarize yourself with your robot’s singularity configurations; • If you do not want to be bothered with singularity and self-collision avoidance algorithms, use angular trajectories. |
| <p>My robot has an unusual behaviour in admittance/force control mode.</p> | <p>It is probably because the gravity compensation is not right. Please try to follow the tips given in Appendix 5 (check torque sensors’ zero, use optimal gravity model, etc.).</p> |
| <p>My arm is in angular mode and it will not switch back to Cartesian mode? However, when I send the robot back to its HOME position, it will switch back to Cartesian mode</p> | <p>The robot will not switch back to Cartesian mode if it is not in a valid Cartesian pose, i.e. too close from a singularity, too close from the base for self-collision avoidance, too close from a joint limit and inside a protection zone. The reason why the robot is always switching back to Cartesian control when you bring it to the Home position is because the Home position is chosen as a valid Cartesian position.</p> |

Contacting support

If you need help or have any questions about this product, this guide or the information detailed in it, please contact a Kinova representative at support@kinovarobotics.com.

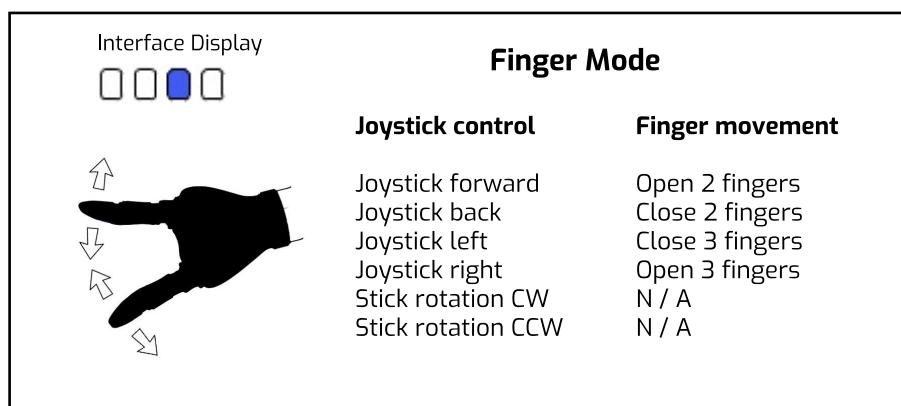
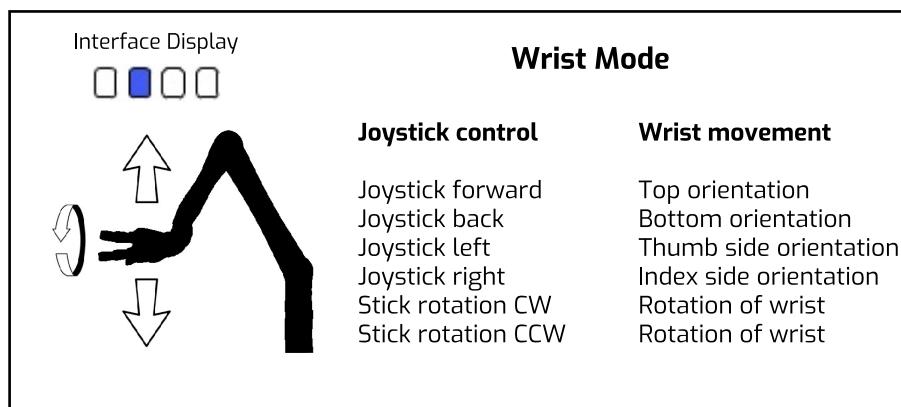
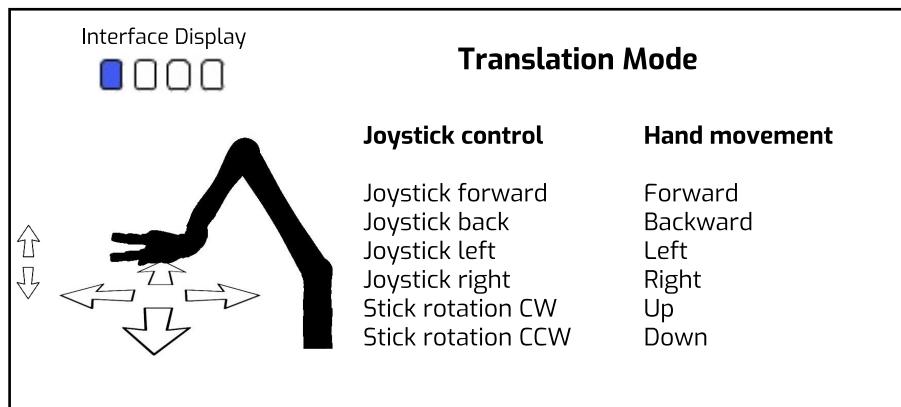
We value your comments!

To help us assist you more effectively with problem reports, the following information will be required when contacting Kinova or your distributor support:

- Product serial number (This will allow the support agent to have all the information regarding your product as the software version running in the device, the part revisions and characteristics, etc.)
- Date/Time of the problem
- Environment where the problem occurred
- Actions performed immediately before the problem occurred

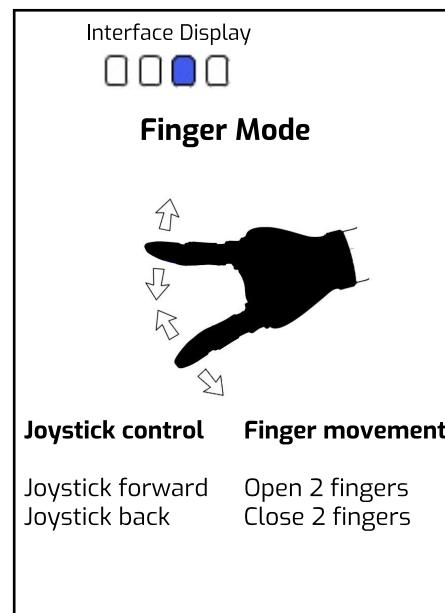
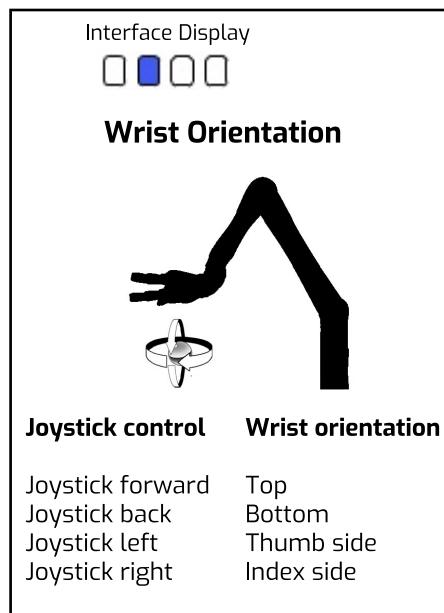
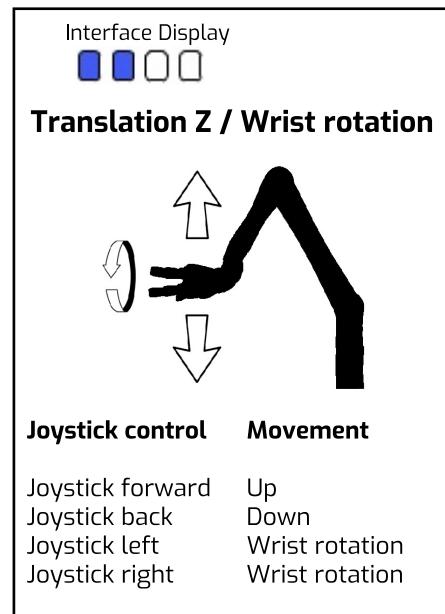
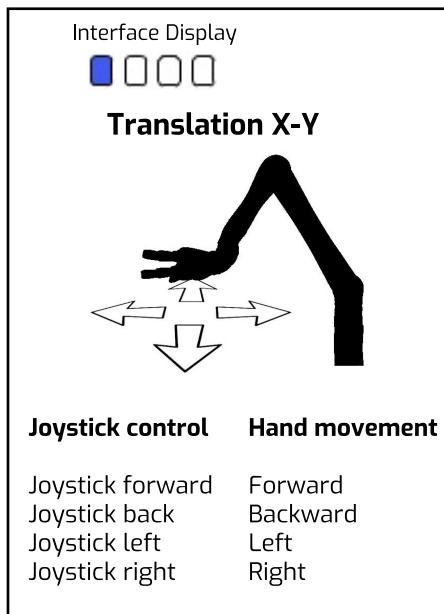
3-axis mode joystick controls reminder

This section is a visual reminder of the joystick controls in 3-axis mode.



2-axis mode joystick controls reminder

This section is a visual reminder of the joystick controls in 2-axis mode.



**Kinova Inc.**

(Headquarters)

4333 Boul. de la Grande Allée
Boisbriand, QC, Canada
J7H 1M7

+1 (514) 277 – 3777

www.kinovarobotics.com
info@kinovarobotics.com

Kinova Europe GmbH

Friedrich-Ebert-Allee 13
53113 Bonn, Deutschland

+49 (0) 228-9293-9148

www.kinovarobotics.de
support@kinovarobotics.de