# 1 Presentation

The Quanser Robotics Package for Research is based on the 6 DOF MICO Open-Architecture robotic platform from Kinova, a suitable platform for professors and graduate students to expand robotics theories to practical challenges and industrial applications. MICO arm is a light-weight robotic arm composed of six serially-linked segments and a two-finger gripper or hand. Using the QUARC® open-architecture the user can control the robot in three-dimensional space, grasp or release objects with the hand, visualize the robot in a virtual environment and examine the theories behind robot control such as DH parameters, forward and inverse kinematics, trajectory planning and motion control and more advanced concepts such as torque control, compliance control, teleoperation and more.


Figure 1.1: The Quanser Robotics Package for Research is based on 6 DOF MICO robotic arm

The interface to the target computer is Matlab® Simulink® with QUARC®. The 6 DOF MICO is accessible through MICO block sets available in QUARC®.

**Caution**

**This equipment is designed to be used for educational and research purposes and is not intended for use by the general public. The user is responsible to ensure that the equipment will be used by technically qualified personnel only.**

# 2 Kinematics

The coordinate system shown in Figure 2.1 is used for the Quanser Robotics Package for Research. Please see section 4.2 for more details about obtaining the forward kinematics.

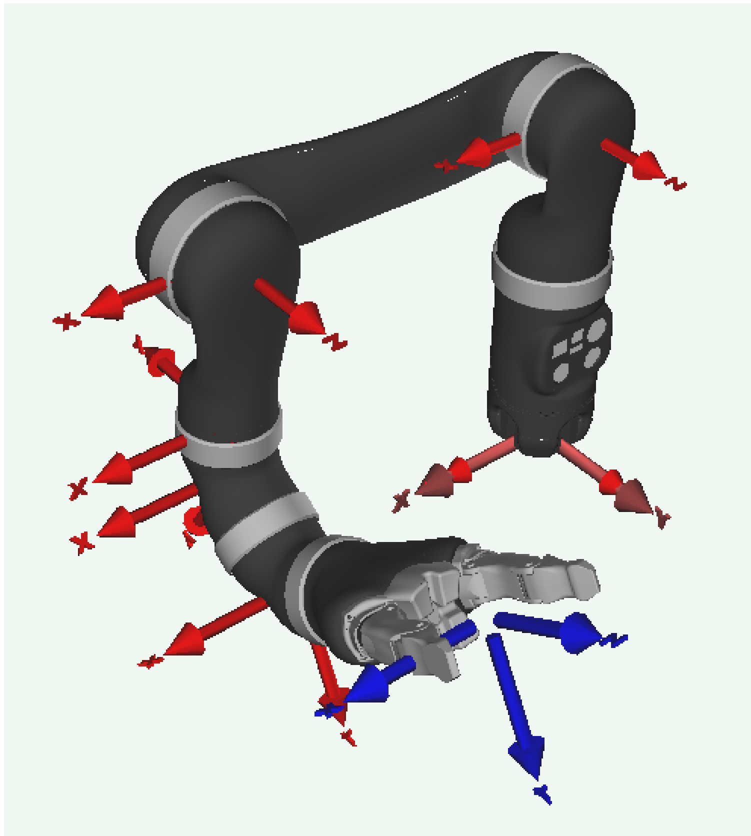


Figure 2.1: D-H Coordinate system used for kinematics of 6 DOF MICO

# 3 Running the Controllers

## 3.1 Overview of Supplied Files

### 3.1.1 General

Table 3.1 lists and describes the various computer files provided with the hardware.

| Item | Description |
|---|---|
| QUARC Help | Type `doc QUARC` in Matlab command prompt to access. Contains information about the 6 DOF MICO QUARC blocks used in all the supplied Simulink / QUARC models. |
| Fastcom DVD | Includes drivers necessary to install the Fastcom: FSCC RS485 Communications Adapter (PCI card), from Commtech Inc., that interfaces with the 6 DOF MICO. |
| 6 DOF MICO User Manual | Includes procedures to install and test the DAQ/FSCC serial card and the 6 DOF MICO itself as well as hardware specifications. |

Table 3.1: Overview of support files supplied with the Quanser Robotics Package for Research

### 3.1.2 Quick Start

Table 3.2 lists and describes the various quick start files provided with the hardware.

| Item | Description |
|---|---|
| 6 DOF MICO Quick Start Guide | Illustrative document that demonstrates how to setup the 6 DOF MICO. |
| MICO_Basic_IO_6DOF.slx | Quick start Simulink® model for position mode, that interfaces with the 6 DOF MICO using QUARC® software. Run this model first to test the 6 DOF MICO system after it has been installed as per the Quick Start Guide (or User Manual), and get accustomed with the 6 DOF MICO's position mode. |
| MICO_Limp_Mode.slx | Quick start Simulink® model for torque mode, that interfaces with the 6 DOF MICO using QUARC® software. Run this model first to test the 6 DOF MICO system after it has been installed as per the Quick Start Guide (or User Manual), and get accustomed with the 6 DOF MICO's torque mode. |

Table 3.2: Overview of Quick Start files supplied with the Quanser Robotics Package for Research

### 3.1.3 Position Mode

Table 3.3 lists and describes the various position mode files provided with the hardware.

| Item | Description |
|---|---|
| MICO_Basic_IO_6DOF.slx | Demo Simulink® model that interfaces with the 6 DOF MICO using QUARC® software. It implements joint-space position control. Users can apply sine wave position commands to each joint and the fingers separately. |
| MICO_Motion_Control_6DOF.slx | Implements both joint-space and task-space position controls. The joint-level position control can be done using the global coordinate system or the relative coordinate system (i.e. based on starting position). The task-space position control is done using the Jacobian through the rate commands. Joint-level rate commands can also be given. |
| MICO_Teleop_6DOF.slx | Contains the controllers to communicate with an input device using the QUARC Communication Stream blocks (i.e. client-server blocks). This can be used with Teleop_Joystick_Command.slx. |
| Teleop_Joystick_Command.slx | Client model that uses the joystick as an input device for the 6 DOF MICO tele-operation demo. Run this model before MICO_Teleop_6DOF.slx. |
| MICO_6DOF_Visualization_Sim.slx | Simulates the robot using QUARC Visualization framework (note: no hardware interface). |

Table 3.3: Overview of Position Mode files supplied with the Quanser Robotics Package for Research

### 3.1.4 Torque Mode

Table 3.4 lists and describes the various torque mode files provided with the hardware.

| Item | Description |
|---|---|
| MICO_Limp_Mode.slx | Using torque control, this model allows the user to put the 6 DOF MICO into *limp* mode (zero torque command) and apply manual non-zero torque commands to each joint. This model can be used to limp the robot and manually move the robot to a specific pose. |
| MICO_Position_Control_Torque _Mode.slx | Similarly to MICO_Basic_IO_6DOF.slx, this model controls the position of the 6 DOF MICO in joint-space, but does so using torque-mode control with position feedback (i.e. not position control mode). |
| MICO_Position_Control_Torque _Mode_Inverted_Pendulum.slx | Similar to MICO_Position_Control_Torque_Mode.slx, but in this model the measured angle of joint 2 is fed as a command to joint 3 to make the robot arm behave like a balancing vertical pendulum. |
| MICO_Spring_and_Assist.slx | Implements a simple spring (force resistance) and compliance (force assistance) control. In spring mode, the robot remains at a fixed position with certain stiffness. In assist mode, the robot tracks and follows the motions of the user as they manually move the robot manipulator arm. |

Table 3.4: Overview of files supplied with the Quanser Robotics Package for Research

## 3.2 Basic I/O Joint-Level Position Control

The MICO_Basic_IO_6DOF.slx model, shown in Figure 3.1 implements a basic joint-level control based on the relative coordinate frame. The joint commands are therefore expressed with respect to the initial joint position. For example, a zero input will result in the robot staying at the initial position (i.e., no motion). Finger commands are with respect to (w.r.t.) the middle (half open/close) position. For example, a zero input will result in the fingers staying at the initial (middle) position (i.e., no motion).

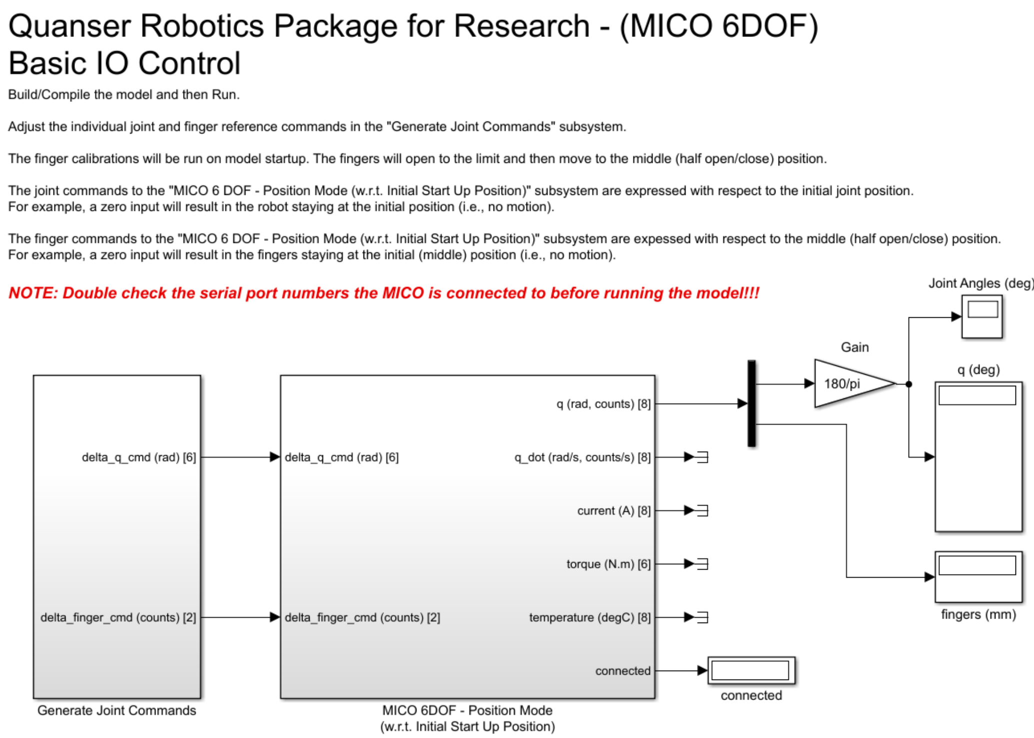



Figure 3.1: MICO_Basic_IO_6DOF.slx model

Follow the steps to run the model on the 6 DOF MICO:

1. Make sure everything is connected as per the Quick Start Guide / User Manual.
2. Ensure the robot is powered OFF and move the 6 DOF MICO to the approximate ”home” position (in the unpowered state, the system is effectively in *limp mode* and allows the user to move the robot to a certain pose).
3. Turn ON the power to engage the position control.
4. Set the serial port numbers of the FSCC RS-485 adapter in the QUARC Kinova Read block (e.g. [3 4]). Take a look at the User manual Section 3.1.3 #9.
5. Go to QUARC | Build to compile the model (generate the code).
6. Go to QUARC | Start to start running the model. The finger calibrations will be ran on model startup. The fingers will open to the limit and then move to the middle (half open/close) position. Once this is done, you can apply different joint and finger commands.
7. In the *Generate Joint Commands* subsystem shown in Figure 3.2, set the *Manual Switch Global Enable* to the upward *Enable Commands* position.
8. Adjust the individual joint and finger reference commands. Each joint has a manual switch for the sine wave amplitude. Set these to the upward position to apply the sine position command to that joint. The frequency of

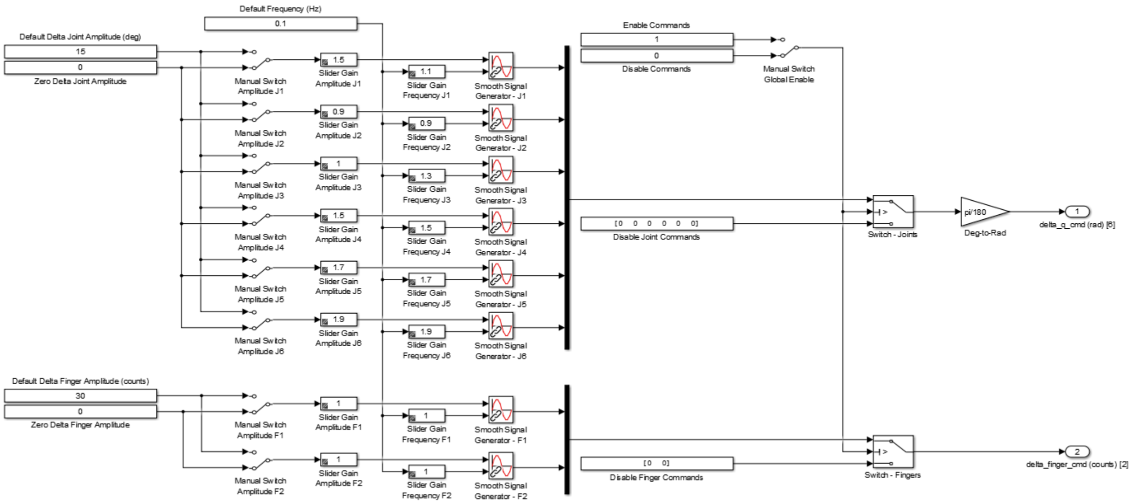



Figure 3.2: Relative joint commands in MICO_Basic_IO_6DOF.slx model

the sine wave can be changed in each *Smooth Signal Generator* block (double-click on the block and change the frequency field). **Caution:** Ensure that the arm does not collide with any surrounding objects, or with itself.

9. Stop the QUARC controller when the experiment is complete by clicking on the `Stop` button in the Simulink® diagram tool bar or go to `QUARC | Stop` from the menu bar.

10. If no additional experiments are going to be run, power off the 6 DOF MICO.

## 3.3 Task-Space Motion Control

The MICO_Motion_Control_6DOF.slx model is shown in Figure 3.3. Using QUARC, this model implements different modes of position control:

1. Relative joint position control: based on the initial position of the robot arm
2. Global joint position control: based absolute position of robot arm
3. Joint rate control (relative)
4. Task-space rate control

For more information about how these different control modes are implemented, see section 4.2.

Follow the steps described below to implement the MICO_Motion_Control_6DOF.slx model in QUARC:

1. Make sure the 6 DOF MICO has been installed and tested using the MICO_Basic_IO_6DOF.slx and MICO_Limp_Mode.slx models first. Refer to the Quick Start Guide and User Manual documentation.
2. As explained in section 3.2, place the robot in the home position and make sure the system is powered.
3. Set the control mode according to the type of control desired.
   (a) **Rate or Position Control**: Set the *Command Mode* manual switch to *Rate Command Mode* to apply rate commands or *Position Command Mode* for joint position commands.
   (b) **Task or joint space RATE control**:
      i. Set the *Command Space* manual switch to *Task Space* in order to apply rate end-effector commands in the task-space coordinate system using *Dummy xdot* constant block.

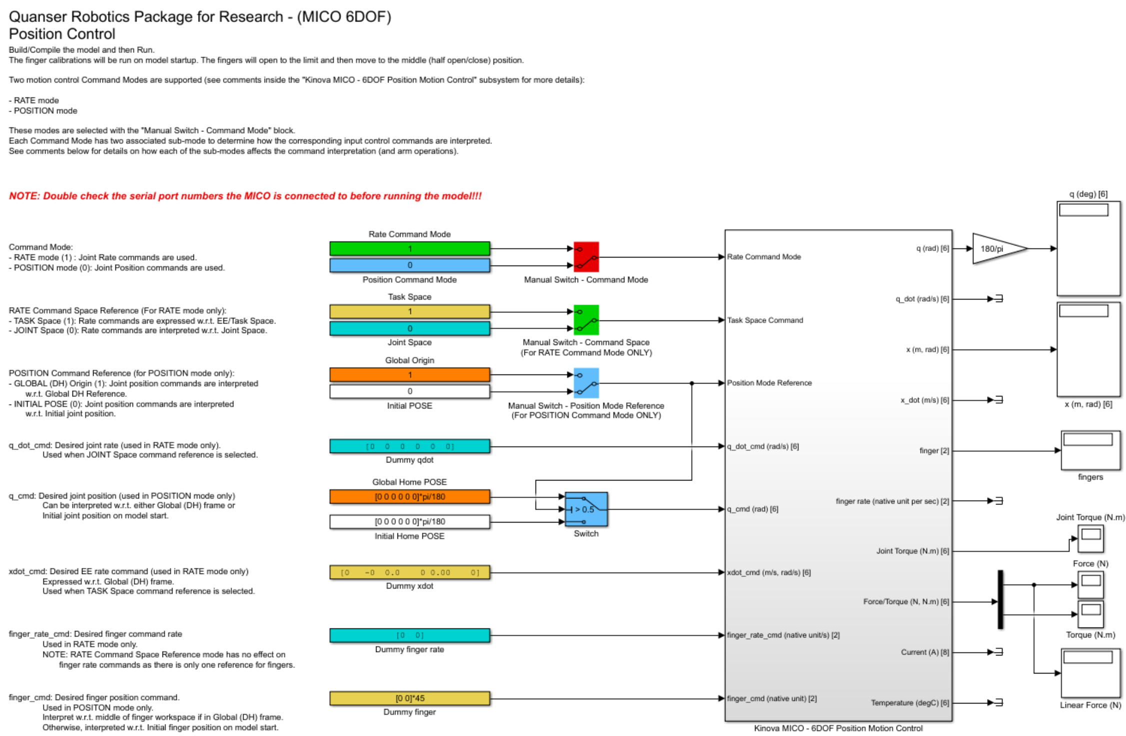



Figure 3.3: MICO_Motion_Control_6DOF.slx model

   ii. Set the switch to *Joint Space* to apply rate commands to the joints using the *Dummy qdot* constant block.

(c) **Relative or global POSITION control** :

   i. Set the *Position Mode Reference* manual switch to *Global Origin* to apply joint-level commands based on the absolute or global frame of reference using *Global Home POSE* constant block.

   ii. Set the switch to *Initial Pose* to apply relative or incremental joint position commands based on the initial position of the robot using *Initial Home POSE* constant block.

**Note:** Select the Command mode (Rate vs. Position) first. In Rate mode, both task space and joint space modes are available. In Position mode, positions can be specified through global DH frames, or as incremental positions w.r.t. an initial starting position.

4. Go to QUARC | Build to compile the model (generate the code).

5. Go to QUARC | Start to start running the model. The finger calibrations will be ran on model startup. The fingers will open to the limit and then move to the middle (half open/close) position. Once this is done, you can apply different end-effector, joint and finger commands.

6. Depending on the control mode selected, choose the appropriate constant block to apply either rate commands in the joint-space or world-space or joint position commands based on the starting pose or global origin.

Caution

**Be careful when setting values in the constant blocks and be mindful of the available workspace of the robot. In particular, start with very small rate commands and make sure you can set it back to zero promtply.**

7. Stop the model when finished.

8. Power off the 6 DOF MICO when finished to avoid unnecessary strain on the joints.

**Note:** The MICO_Motion_Control_6DOF.slx model is an example of how to implement various motion profiles. Setting commands via constant blocks is meant to simplify the diagram. The Simulink model can be modified to include different command signals (e.g. sine wave as in MICO_Basic_IO_6DOF.slx).

## 3.4 Teleoperation

This section presents example models that can be used to control the 6 DOF MICO robot arm using a joystick input device. The teleoperation MICO_Teleop_6DOF.slx *server* model is shown in Figure 3.4 and the input joystick Teleop_Joystick_Command.slx *client* model is shown in Figure 3.5. Both of these models have to be ran together in order to run the demo.

The MICO_Teleop_6DOF.slx model operates using rate control in task-space mode, similarly as done in the MICO_Motion_Control_6DOF.slx in Section 3.3. Using the QUARC Communications API, the MICO_Teleop_6DOF.slx includes a *Stream Server* block that accepts a connection from a local or remote client (running a *Stream Client* block) where the desired position commands are sent from.

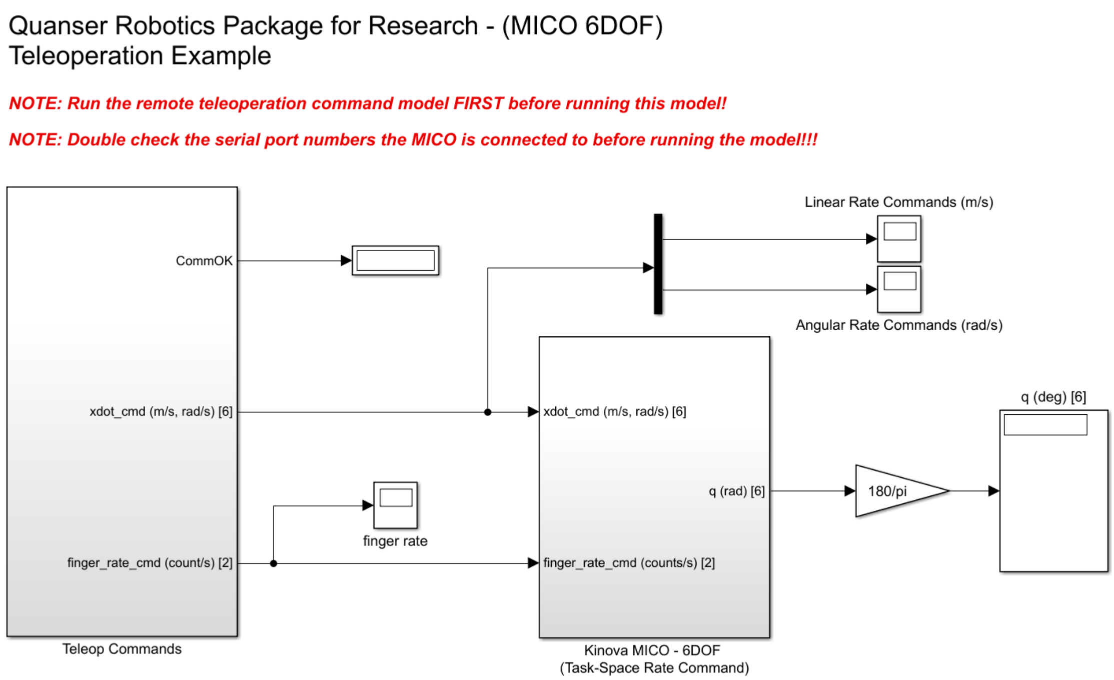



Figure 3.4: MICO_Teleop_6DOF.slx model

The Teleop_Joystick_Command.slx model is an example model that uses the QUARC Host Peripheral framework to interface to a joystick. The model is calibrated such that the joystick generates Cartesian task-space rate commands. These Cartesian rate commands are sent through QUARC Stream Client block to the MICO_Teleop_6DOF.slx model. This model was created with the Logitech® Extreme™ 3D Pro (X3D) (see Figure 3.6) in mind, but can be converted to use other input devices as well.

⚠ **Caution**

**Always make sure the QUARC Teleop_Joystick_Command.slx model is ran BEFORE the MICO_Teleop_6DOF.slx model.**

Follow these instructions to run the teleoperation demo models on the 6 DOF MICO:

1. Make sure the 6 DOF MICO has been installed and tested using the MICO_Basic_IO_6DOF.slx and

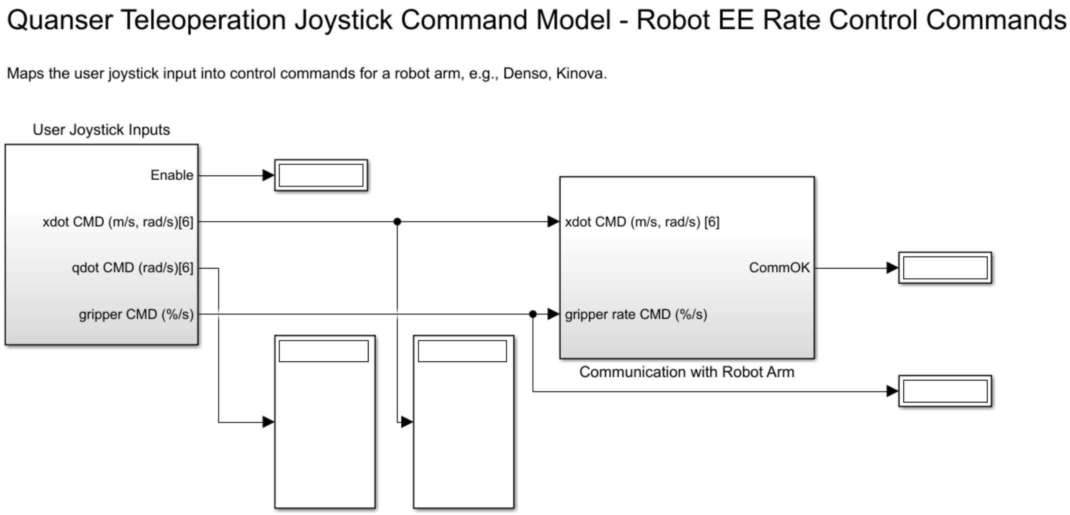



Figure 3.5: Teleop_Joystick_Command.slx model

MICO_Basic_IO_6DOF.slx models first. Refer to the Quick Start Guide and User Manual documentation.

2. Make sure the joystick is connected to the same PC.
3. Move the 6 DOF MICO robot arm to the home position.
4. Turn on the power on the manipulator.
5. Open the Teleop_Joystick_Command.slx model.
6. Go to QUARC | Build to compile the model (generate the code).
7. Go to QUARC | Start to start running the model.
8. Open the MICO_Teleop_6DOF.slx model.
9. Go to QUARC | Build to compile the model (generate the code).
10. Go to QUARC | Start to start running the model. The finger calibrations will be ran on model start up. The fingers will open to the limit and then move to the middle (half open/close) position.
11. Make sure the joystick knob is at least half-way between the minimum and maximum range to enable position commands.
12. Move the joystick and examine the motions of the robot arm. The joystick button mapping is provided in Figure 3.6.

Caution

**Always be mindful of the available workspace and any obstructions when running any of the models.**

13. To stop running the demo, first go to MICO_Teleop_6DOF.slx model and stop QUARC.
14. Then, go to Teleop_Joystick_Command.slx model and select QUARC | Stop.
15. Power OFF the robot if no more experiments will be conducted.

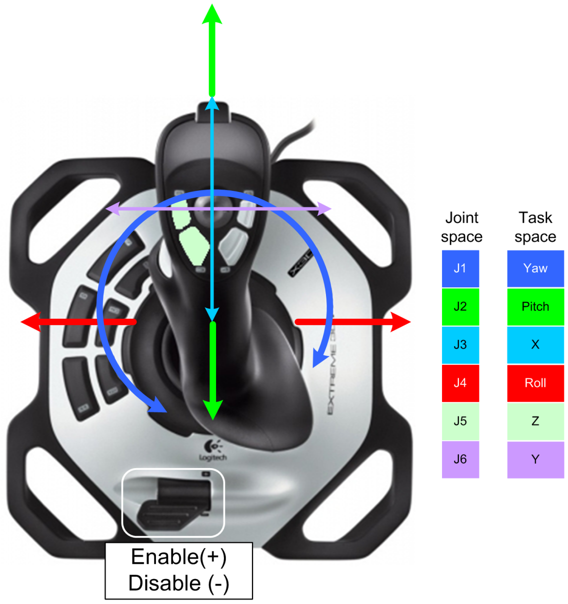



Figure 3.6: Joystick mapping for teleoperation

## 3.5 Visualization Simulation

The MICO_6DOF_Visualization_Sim.slx model shown in Figure 3.7 can be used to simulate the 6 DOF MICO system with a visualization. The visualization is done using the QUARC Visualization framework. The robot visualization has the same movements and pose as in the actual hardware.

The joint commands to the *Visualization - MICO 6DOF* subsystem are expressed with respect to the Global D-H reference position. For example, a zero input for all 6 joints will result in the robot staying at the D-H HOME position (displayed in Figure 2.1 or Figure 3.8). The finger commands in the *Visualization - MICO 6DOF* subsystem are with respect to the middle (half open/close) position. For example, a zero input will result in the fingers staying at the initial (middle) position (i.e. no motion).

**Note:** There is no dynamic model of the 6 DOF MICO robot arm in this Simulink file. The reference commands are applied directly to the visualization.

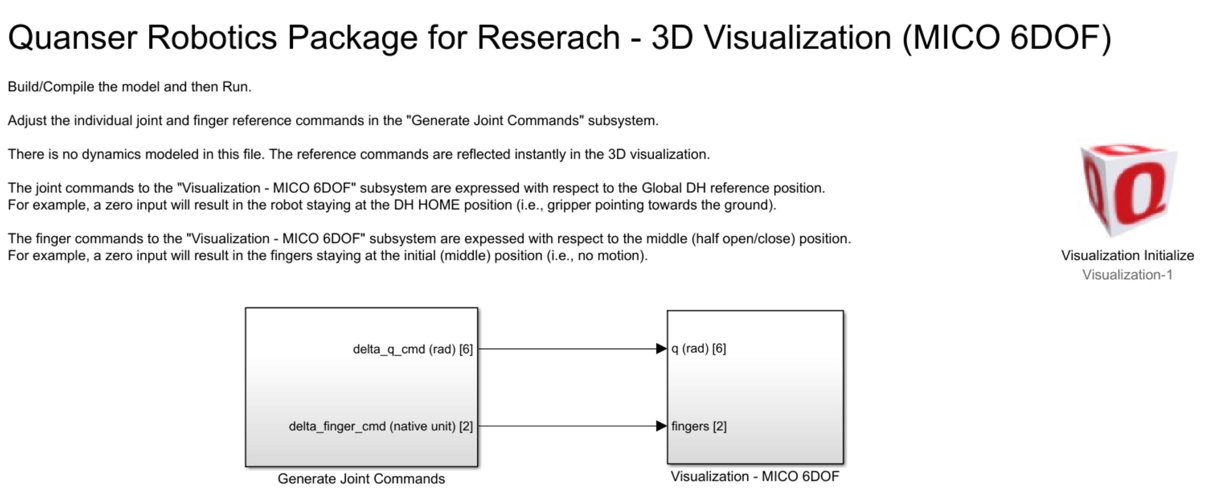



Figure 3.7: MICO_6DOF_Visualization_Sim.slx model

Follow these steps to run the model:

1. Open up the MICO_6DOF_Visualization_Sim.slx Simulink model in Matlab.
2. Go to QUARC | Build to compile the model (generate the code).
3. Go to QUARC | Start to start running the model. The visualization shown Figure 3.8 should load (i.e. no need to build code).

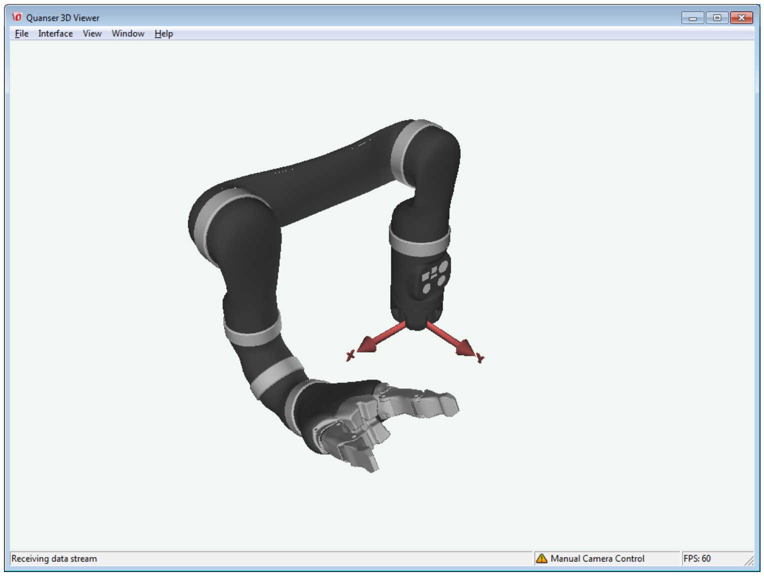



Figure 3.8: Visualization of the 6 DOF MICO

4. Vary the joint-level position commands in the *Generate Joint Commands* block and examine how the position of the robot changes in the visualization.
5. Click on the *Stop* button when finished.

## 3.6 Torque Control

The MICO_Limp_Mode.slx model shown in Figure 3.9 has an option to apply zero torque to all joints to place it in *limp mode* or apply torque commands to each joint individually.

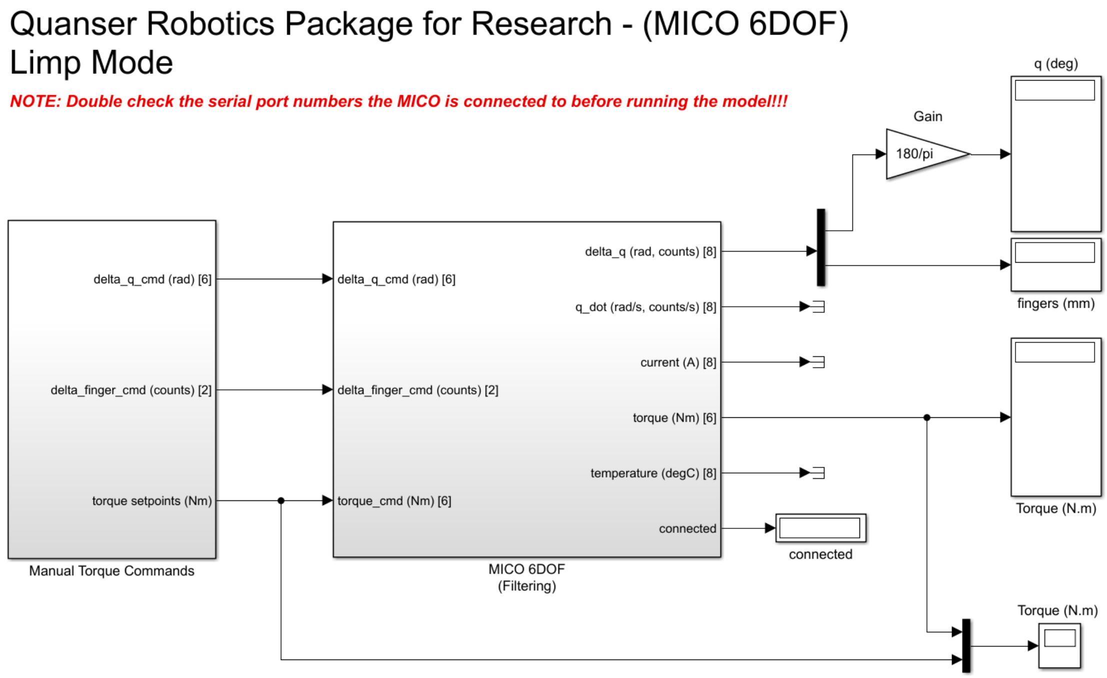



Figure 3.9: MICO_Limp_Mode.slx model

All the torque-based control models supplied have these features:

- **Torque mode entry/transition:** In order to enter/transition to the *Torque Control Mode* in the 6 DOF MICO system, the commanded torques must be relatively close to the torques measured by the sensors. This subsystem implements this functionality. See 4.1.2 for more details.
- **Safety stop for internal collision:** The models include safety watchdogs on all the joint angles. If the manipulator is about to enter a state where it may collide with itself, the QUARC model stops running and the 6 DOF MICO stops moving and maintains its current position. An error message pops up in the QUARC® model.

Follow these instructions to run the model on the 6 DOF MICO:

1. Make sure the 6 DOF MICO has been installed and tested using the MICO_Basic_IO_6DOF.slx and MICO_Limp_Mode.slx models first. Refer to the Quick Start Guide and User Manual documentation.
2. As explained in the 3.2, place the robot in the home position and make sure the system is powered.
3. Go to QUARC | Build to compile the model (generate the code).
4. Go to QUARC | Start to start running the model. The finger calibrations will run on model startup. The fingers will open to the limit and then move to the middle (half open/close) position.
5. In the *Manual Torque Commands* subsystem, shown in Figure 3.10, set the *Limp Mode* Manual Switch to the downward position to enable *limp mode*. This applies zero torque command to all the joints and allows the user to move the robot arm freely.
6. To apply direct torque commands to each joint, set the Manual Switch block to the upward position and change the Slider Gain block values for any of the joints.

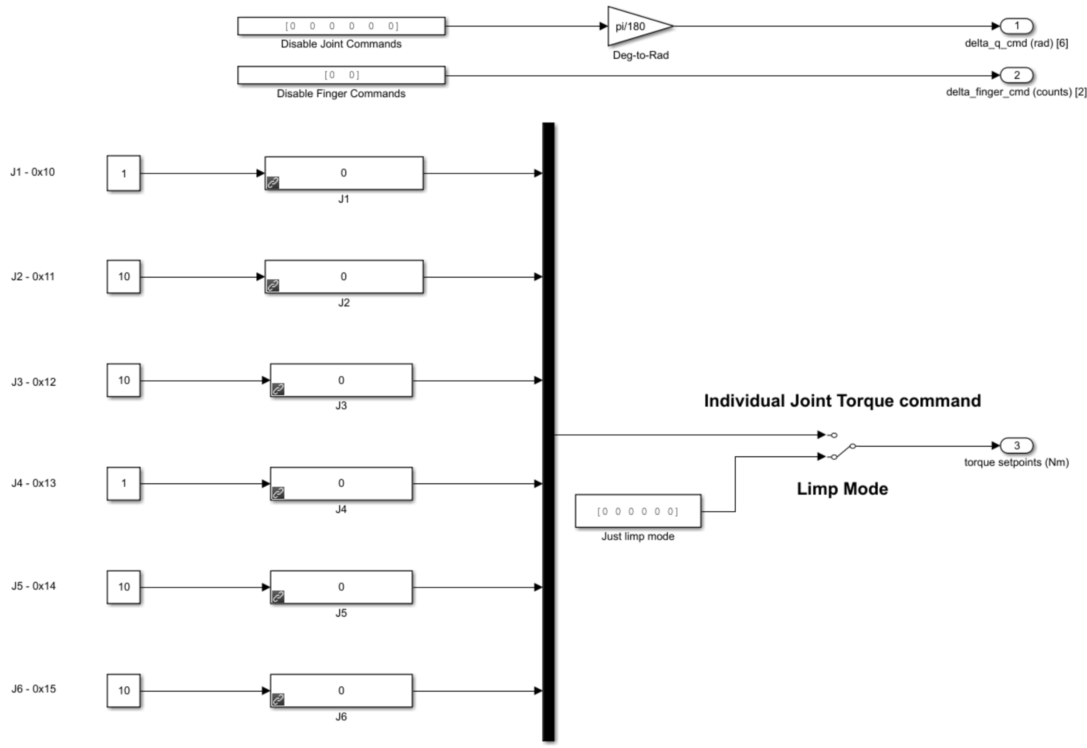



Figure 3.10: MICO_Limp_Mode.slx model commands

Caution

**Keep the initial torque commands low and make sure you can reset the torque commands back to zero quickly. Make sure there is no users or obstructions in the workspace area as the tests are conducted. User the Pause/Break key on your keyboard to stop the model if necessary.**

7. Stop the QUARC controller and power OFF the robot if no more experiments will be conducted.

## 3.7 Position Control in Torque Mode

The MICO_Position_Control_Torque_Mode.slx model shown in Figure 3.11 implements position feedback control in torque mode. Similarly to MICO_Basic_IO_6DOF.slx, joint position commands can be applied directly to each joint using the sine wave blocks. However, joint position control is implemented through torque-mode with position feedback. The position commands are converted into torque commands using a PID control scheme. Using torque commands and feedback, the desired joint angles are achieved. All joints are controlled individually, assuming no coupling between the joints.

The controller uses 6-parallel PID controllers in a vectorized format. This allows users to set a diagonal matrix for the joint gains. Non-diagonal elements can be used to introduce coupling between the joints. This is implemented in the *PID Controller* block, shown in Figure 3.12, in MICO_Position_Control_Torque_Mode.slx as well as in MICO_Position_Control_Torque_Mode_Inverted_Pendulum.slx and MICO_Spring_and_Assist.slx.

Caution

**Make sure the 6 DOF MICO is moved to the upward vertical configuration (i.e. pointing towards the ceiling, see Figure 3.13) BEFORE starting the QUARC controller model.**

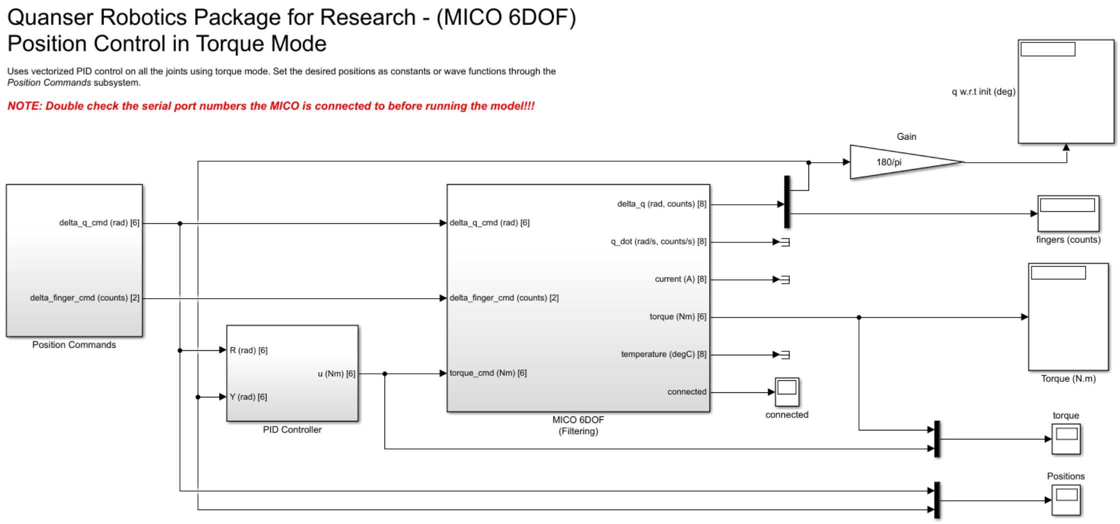



Figure 3.11: MICO_Position_Control_Torque_Mode.slx model

Follow these instructions to run the model on the 6 DOF MICO:

1. Make sure the 6 DOF MICO has been installed and tested using the MICO_Basic_IO_6DOF.slx and MICO_Limp_Mode.slx models first. Refer to the Quick Start Guide and User Manual documentation.
2. When un-powered, position the 6 DOF MICO robot arm so it is pointing vertically upwards towards the ceiling.
3. Turn on the power.
4. Go to QUARC | Build to compile the model (generate the code).
5. Go to QUARC | Start to start running the model. The finger calibrations will be ran on model startup. The fingers will open to the limit and then move to the middle (half open/close) position.
6. In the *Position Commands* subsystem, shown in Figure 3.14, set *Manual Switch Global Enable* to the enable position.
7. Adjust the individual joint and finger reference commands. Each joint has a manual switch for the sine wave amplitude. Set these to the upward position to apply the sine position command to that joint. The frequency of the sine wave can be changed in each *Smooth Signal Generator* block (double-click on the block and change the frequency field).

⚠ Caution

**Make sure to not command a position where the manipulator might collide with itself. If this is about to happen, the model will stop and an error message will pop up.**

8. To reset the robot back to the originally position, set the *Manual Switch Global Enable* back to the *Disable Commands* configuration to set all joint angle commands to zero.
9. Stop the QUARC controller and power OFF the robot if no more experiments will be conducted.

### 3.7.1 Position Control in Torque Mode with Inverted Pendulum behaviour

Similarly as in the MICO_Position_Control_Torque_Mode.slx model, the MICO_Position_Control_Torque_Mode_Inverted_Pendulum.slx model, shown in Figure 3.15, implements position feedback control in torque mode, that

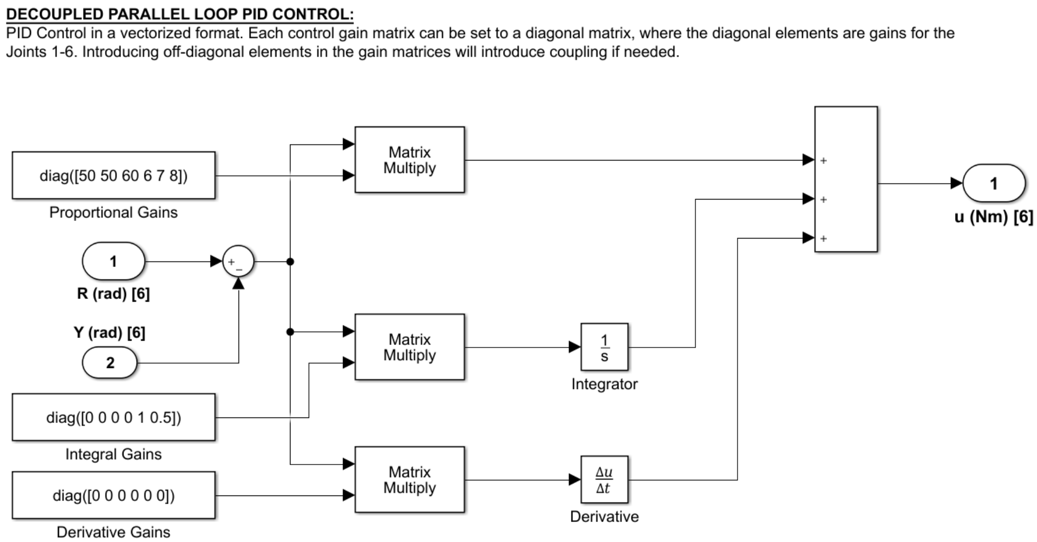



Figure 3.12: PID Parallel Loops used in Torque Control models

allows the user to change joint position commands. However, it uses the current joint 2 position measurement and feeds it back as the desired joint position for joint 3, through a negative gain and filter. Thus joint 1, 2, 4, 5, and 6 can accept manual commands but joint 3 will mirror the actions of joint 2. This will keep the link after joint 3 in the same configuration with respect to the link before joint 2 during startup.

Run this model initially with the MICO arm pointing vertically upwards towards the ceiling, as seen in Figure 3.16a. With a 45 deg increment applied to joint 2, the manipulator configuration should resemble Figure 3.16b. Also, try with the MICO arm straight from joint 2 onwards, but joint 2 bent, so that the links after joint 2 are parallel to the table-top/floor/ground.

Caution

**Make sure the 6 DOF MICO is moved to the upward vertical configuration (i.e. pointing towards the ceiling, Figure 3.16a) BEFORE starting the QUARC controller model.**

Follow these instructions to run the model on the 6 DOF MICO:

1. Make sure the 6 DOF MICO has been installed and tested using the MICO_Basic_IO_6DOF.slx and MICO_Limp_Mode.slx models first. Refer to the Quick Start Guide and User Manual documentation.
2. When un-powered, position the 6 DOF MICO robot arm so it is pointing vertically upwards towards the ceiling (Figure 3.16a).
3. Turn on the power.
4. Go to QUARC | Build to compile the model (generate the code).
5. Go to QUARC | Start to start running the model. The finger calibrations will be ran on model startup. The fingers will open to the limit and then move to the middle (half open/close) position.
6. In the *Position Commands* subsystem, shown in Figure 3.14, set *Manual Switch Global Enable* to the enable position.
7. Adjust the individual joint and finger reference commands. Each joint has a manual switch for the sine wave amplitude. Set these to the upward position to apply the sine position command to that joint. The frequency of the sine wave can be changed in each *Smooth Signal Generator* block (double-click on the block and change the frequency field).

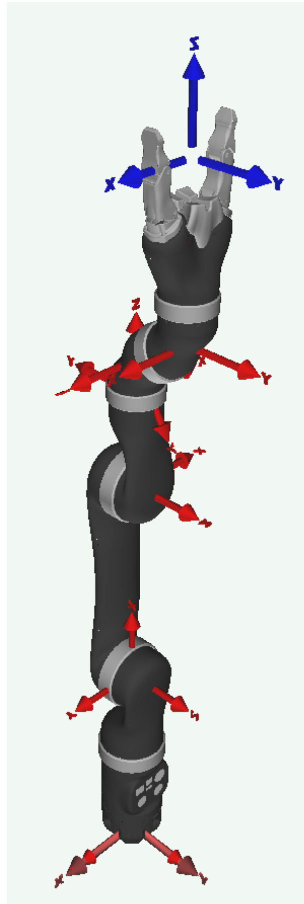


Figure 3.13: 6DOF MICO in the upward vertical configuration

8. For example, if all the joints are commanded a $\pm 15$ degrees sine wave then all the joints will move in that fashion, but joint 2 and 3 will mirror each other (because the measured joint 2 position is fed back as joint 3 command).
9. To reset the robot back to the originally position, set the *Manual Switch Global Enable* back to the *Disable Commands* configuration to set all joint angle commands to zero.
10. Stop the QUARC controller and power OFF the robot if no more experiments will be conducted.

## 3.8 Spring and Assist Control

The MICO_Spring_and_Assist.slx model shown in Figure 3.17 allows the user to switch between two modes: spring (stiffness) and assist (compliance). In *stiffness mode*, pure proportional control is used, and the manipulator acts as a spring about the desired position. In *compliance mode*, pure derivative control is used, and the manipulator provides torques that assist the arm in the direction of motion. This is achieved through negative derivative gains.

The manipulator is in spring mode by default. When the A, S and D keys are ALL pressed down, the manipulator enters compliance mode, making it easy to move the manipulator around. When the manipulator enters spring mode again, the current manipulator position becomes the new desired position, about which, the manipulator acts like a spring again.

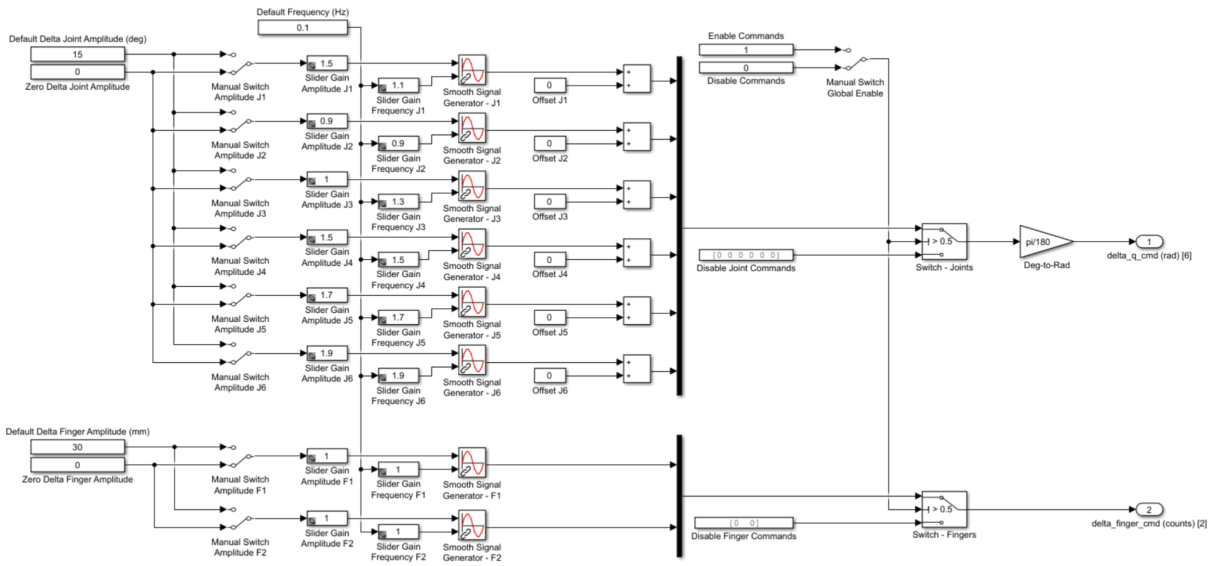



Figure 3.14: MICO_Position_Control_Torque_Mode.slx model commands

**Caution** **Always make sure someone is holding the 6 DOF MICO robot arm when in compliance mode. Never release the robot when the A, S and D keys are all pressed down as gravity will pull the arm down forcefully and the assist mode will aid gravity further.**

Follow these instructions to run the model on the 6 DOF MICO:

1. Make sure the 6 DOF MICO has been installed and tested using the MICO_Basic_IO_6DOF.slx and MICO_Limp_Mode.slx models first. Refer to the Quick Start Guide and User Manual documentation.
2. When un-powered, position the 6 DOF MICO robot arm so it is pointing vertically upwards towards the ceiling (as in Figure 3.13).
3. Turn on the power.
4. Go to QUARC | Build to compile the model (generate the code).
5. Go to QUARC | Start to start running the model. The finger calibrations will be ran on model startup. The fingers will open to the limit and then move to the middle (half open/close) position.
6. **Spring mode:** By default, the model ran in *stiffness* mode. Manually grasp the robot arm and move it gently. Feel the stiffness as it behaves like a spring.
7. **Assist mode**: Make sure someone is holding on to the robot and press DOWN on all three of the A, S and D keys. Gently move the robot arm manually. Notice how the 6 DOF MICO assists the motions of the user. Negative damping is used to help move the robot arm in the same direction as the user tries to move it.
8. Release one, two or all three of the A, S or D keys to enter the stiffness mode again, and maintain the current 6 DOF MICO position.

**Caution** **When you switch modes from Assist to Spring, the manipulator may still move a little. This is normal as gravity will move the manipulator in spring mode.**

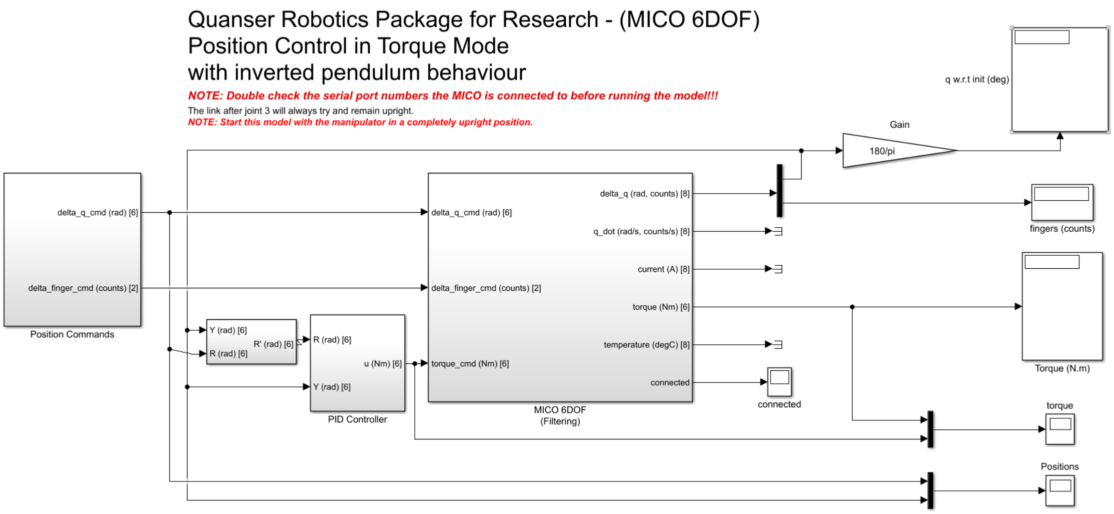
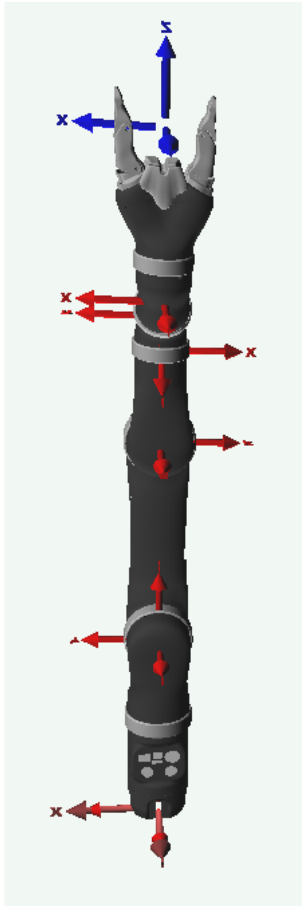
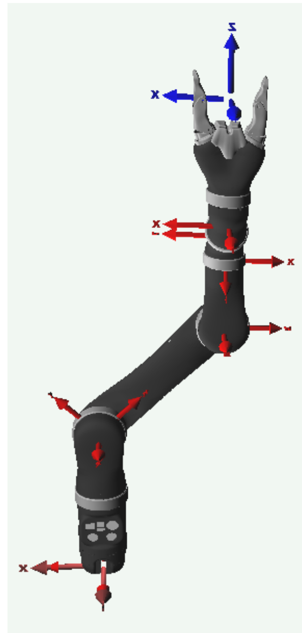



Figure 3.15: MICO_Position_Control_Torque_Mode_Inverted_Pendulum.slx model

9. Stop the QUARC controller and power OFF the robot if no more experiments will be conducted.



(a) 6DOF MICO vertically upward



(b) 6DOF MICO with 45 deg joint 2 increment

Figure 3.16: 6DOF MICO straight and bent configurations

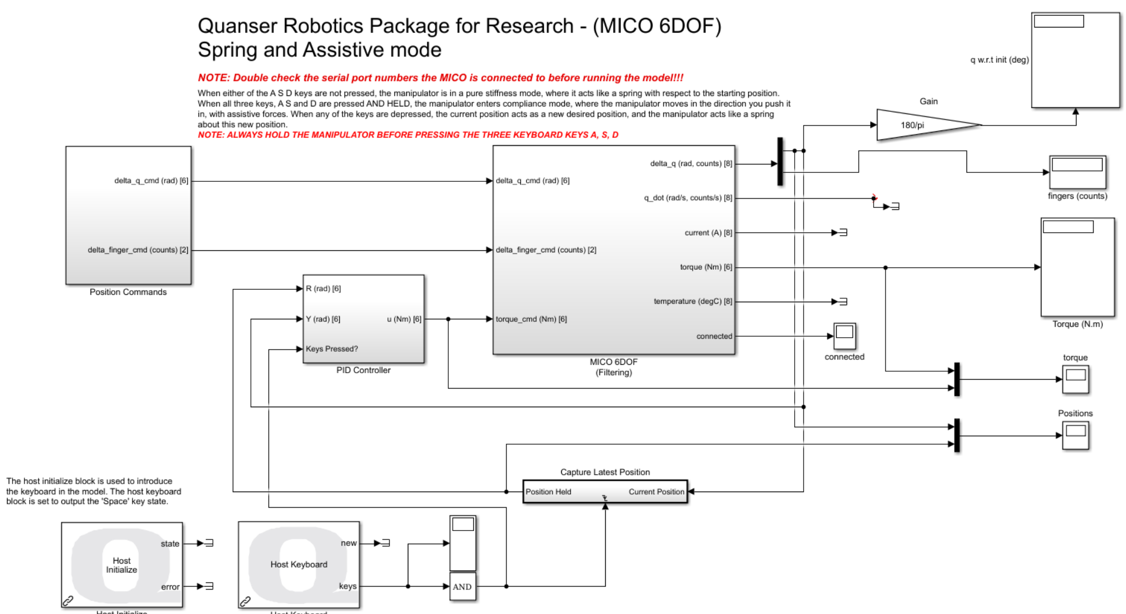



Figure 3.17: MICO_Spring_and_Assist.slx model

# 4 Model Description

This sections provides an overview of the different subsystems used in the Simulink models that supplied with the 6 DOF MICO.

## 4.1 6 DOF MICO I/O Subsystem

The innermost subsystem *6 DOF MICO I/O* is shown in Figure 4.1. This is shared between all the position and torque mode controllers. This block receives a control mode (position or torque) as well as commands. In position mode, this block provides an absolute DH joint position command input. In torque mode, the block provides an absolute DH torque command input. It converts the absolute DH values to absolute physical values, and writes them to the manipulator through the KINOVA Write block. It also reads absolute physical joint angles as well as absolute physical torques from the sensors, converts them to absolute DH, and outputs these values. It also has joint safety limits implemented, as well as a torque matching functionality during torque mode transition as explained below.

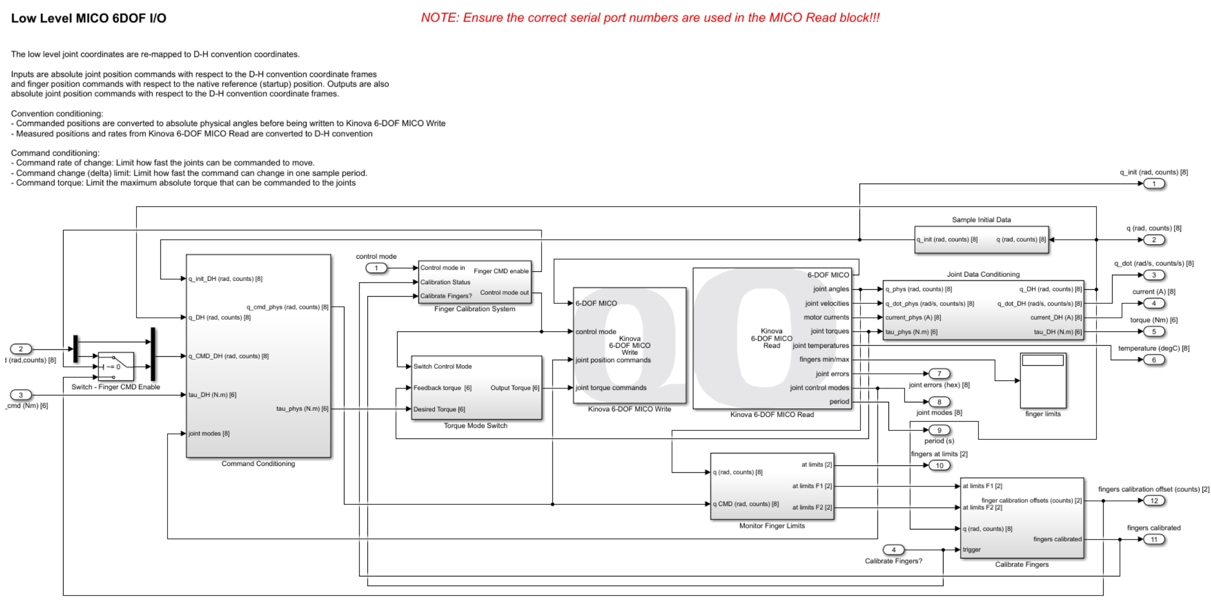



Figure 4.1: 6 DOF Position Mode Control subsystem

### 4.1.1 Command Conditioning before and after Read/Write

The QUARC *Kinova 6-DOF MICO Write* and *Kinova 6-DOF MICO Read* blocks are described in the QUARC Help documentation. The joint position commands applied to the write block are relative to the physical frame of reference (shown in the QUARC Help page for this block. Similarly, the measured joint positions from the read block is also based on the 6 DOF MICO physical coordinate system. Therefore the D-H commands from *q_cmd* input node are converted into *q_cmd_phys* before being applied to the write block. The measured low level joint coordinates are then re-mapped to D-H convention coordinates. The DH home position can be seen in Figure 2.1.

In addition to converting the joint angles from D-H into low-level physical joint angles, the *Command Conditioning* block imposes safety limits on the inputs:

- **Command rate of change**: Limit how fast the joints can be commanded to move.
- **Command change (delta) limit**: Limit how fast the command can change in one sample period.
- **Command torque limit**: Limit the maximum and minimum torques that can be applied (in torque mode).
- **Joint Safety**: Stops the model and pops an error message if the manipulator is about to collide with itself.

## 4.1.2 Torque Mode Switch

When the manipulator is either powered ON (but no QUARC® model is running) or the manipulator has a QUARC® model running on it in Position Control mode, the low level joint controllers on the manipulator apply torques to maintain the corresponding joint position. In order to enter/transition to the *Torque Control Mode* in the 6 DOF MICO system, the torque mode torque commands must be relatively close to the torques measured by the sensors. This subsystem commands the negative of the measured torques (applied torque) when the model starts running for two samples, before switching to the torque commanded by the user.

## 4.1.3 Sample Initial Data

This subsystem samples the initial pose of the manipulator to enable conversion from absolute DH angles to incremental DH angles, for both desired and measured joint positions.

# 4.2 6 DOF MICO Position Control Subsystems

The MICO_Motion_Control_6DOF.slx model includes a subsystem block called *Kinova MICO - 6 DOF Position Motion Control* shown in Figure 4.2. It is a generic block that provides the ability to apply different types of commands in different modes: angle mode or rate mode.

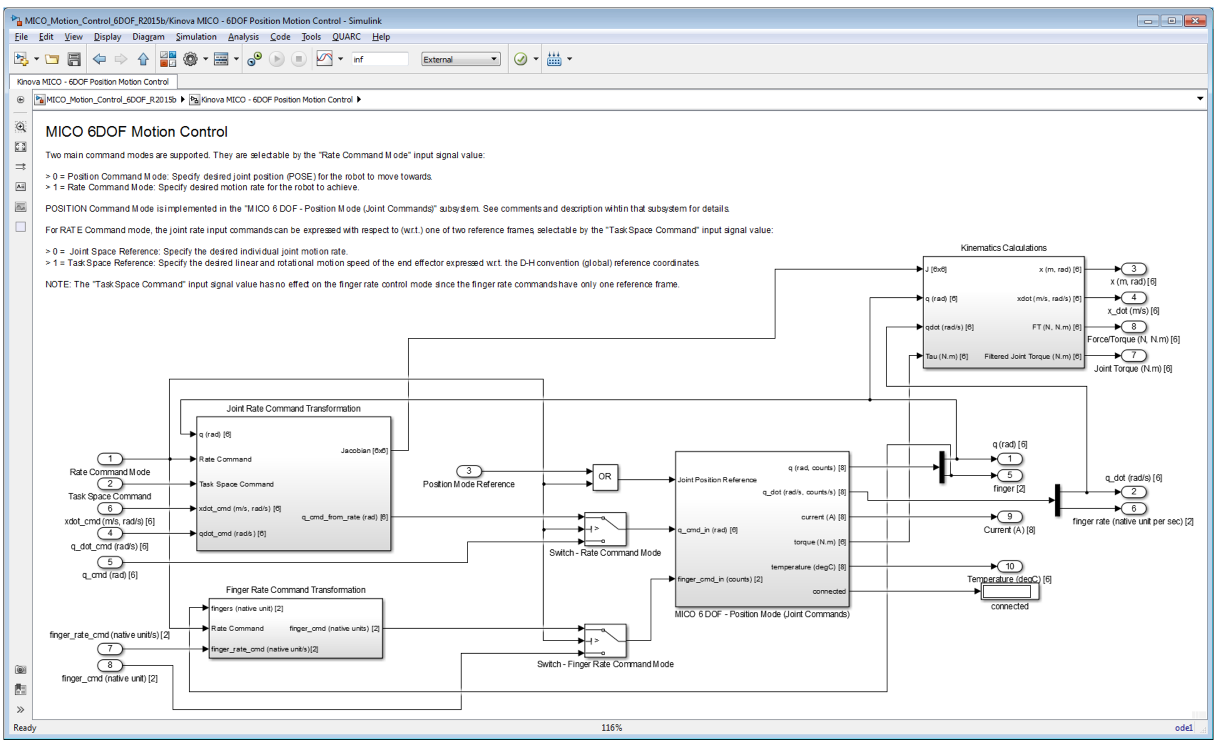



Figure 4.2: Kinova MICO - 6 DOF Position Motion Control block

The *Rate Command Mode* input signal enables you to choose between two different types of commands:

- **Position Command Mode (0):** Specify desired joint position (POSE) for the robot to move towards. Position Command Mode is implemented in the *MICO 6 DOF - Position Mode (Joint Commands)* subsystem described in 4.2.1. See also comments and description within that subsystem in Simulink for details.
- **Rate Command Mode (1):** Specify desired motion rate for the robot to achieve. For *Rate Command* mode, the rate input commands can be expressed with respect to one of two reference frames, selectable by the *Task Space Command* input signal value:
  - **Joint Space Reference (0):** Specify the desired individual joint motion rate.

– **Task Space Reference (1):** Specify the desired linear and rotational motion speed of the end effector expressed w.r.t. the D-H convention (global) reference coordinates.

**Note:** The *Task Space Command* input signal value has no effect on the finger rate control mode since the finger rate commands have only one reference frame.

Table 4.1 summarizes what each block in the *Kinova MICO - 6 DOF Position Motion Control* subsystem does.

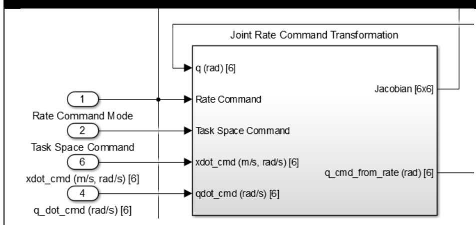
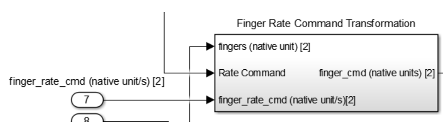
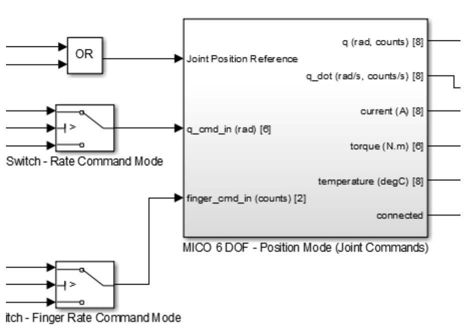
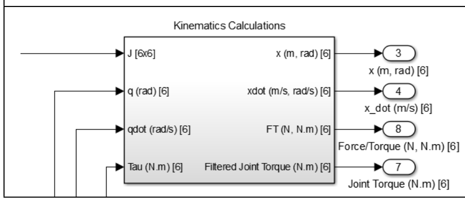
| Subsystem | Description |
|---|---|
|  | *Joint Rate Command Transformation:* Integrates the rate commands to corresponding position commands. If Task Space Command is used, the rate commands are first transformed into joint space before being integrated using the Jacobian. This is based on the desired rate: either the task-space rate commands *x_dot_cmd* or the joint-level rate commands *q_dot_cmd*. Task-space and joint-space limits on the rates are applied using Rate Limiter blocks. |
|  | *Finger Rate Command Transformation*: Integrates the desired finger rate command into corresponding desired finger position commands. The integrator is reset whenever the *Rate Command Mode* is activated. Speed limits are imposed on the finger rate commands using *Rate Limiter* block. |
|  | *MICO 6 DOF - Position Mode (Joint Commands)*: Applies joint-level commands based on either the absolute (DH) joint position or incremental joint position commands based on the initial pose of the robot. See 4.2.1 for more information. |
|  | *Kinematics Calculations*: Calculate the position of the end-effector in Cartesian task-space. Also calculates the equivalent end-effector torque based on measured joint torques. |

Table 4.1: Simulink blocks common to models supplied with 6 DOF MICO

### 4.2.1 MICO 6 DOF - Position Mode (Joint Commands)

The interior of the *MICO 6 DOF - Position Mode (Joint Command)* subsystem is shown in Figure 4.3. The input joint commands can be either absolute (DH) joint positions or incremental joint position commands (based on the initial joint positions). The position reference mode is selected by the value of the "Position Reference" input:

- **Incremental (0)**: position commands based on initial joint positions
- **Absolute/Global (1)**: position commands based on D-H convention reference (HOME) coordinates

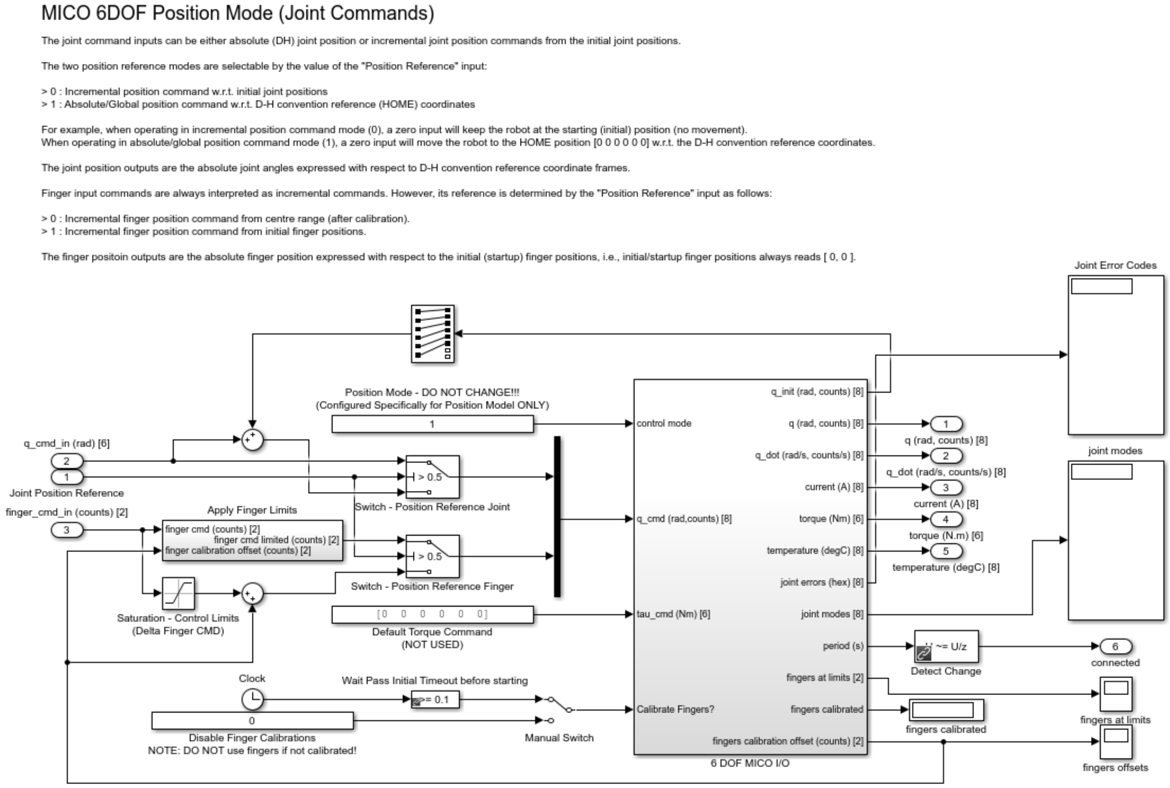



Figure 4.3: 6 DOF Position Mode (Joint Commands) subsystem

For example, when operating in incremental position command mode (0), a zero input will keep the robot at the starting (initial) position (no movement). When operating in absolute/global position command mode (1), a zero input will move the robot to the HOME position $[0, 0, 0, 0, 0, 0]$ with respect to (w.r.t.) the D-H convention reference coordinates (see Figure 2.1).

The joint position outputs are the absolute joint angles expressed with respect to D-H convention reference coordinate frames.

Finger input commands are always interpreted as incremental commands. However, its reference is determined by the ”Position Reference” input as follows:

- **From center (0):** Incremental finger position commands are relative to the center range (after calibration).
- **From initial (1):** Incremental finger position commands are based on the initial finger positions.

The finger position outputs are the absolute finger position expressed with respect to the initial (start-up) finger positions, i.e., initial/start-up finger positions always reads $[0, 0]$.

## 4.3 6 DOF MICO Torque Control Subsystems

This is analogous to the subsystems described in 4.2. Torque-based control models such as MICO_Position_Control_Torque_Mod include a subsystem block called *MICO 6DOF Torque Mode (Torque Commands)* shown in Figure 4.4. It has some of the functionality as the *Kinova MICO - 6 DOF Position Motion Control* shown in Figure 4.3, but includes the ability to apply torque commands to each joint directly (instead of commanding joint positions) and includes the *Joint Safety* block. For torque mode models, this block does not provide a Joint Position Reference option, that is, in position mode, joint commands and readings are always incremental.

Note that the subsystem 6DOF MICO I/O is the same as that presented in 4.1, and is used in both torque and position mode models as the innermost read/write block.

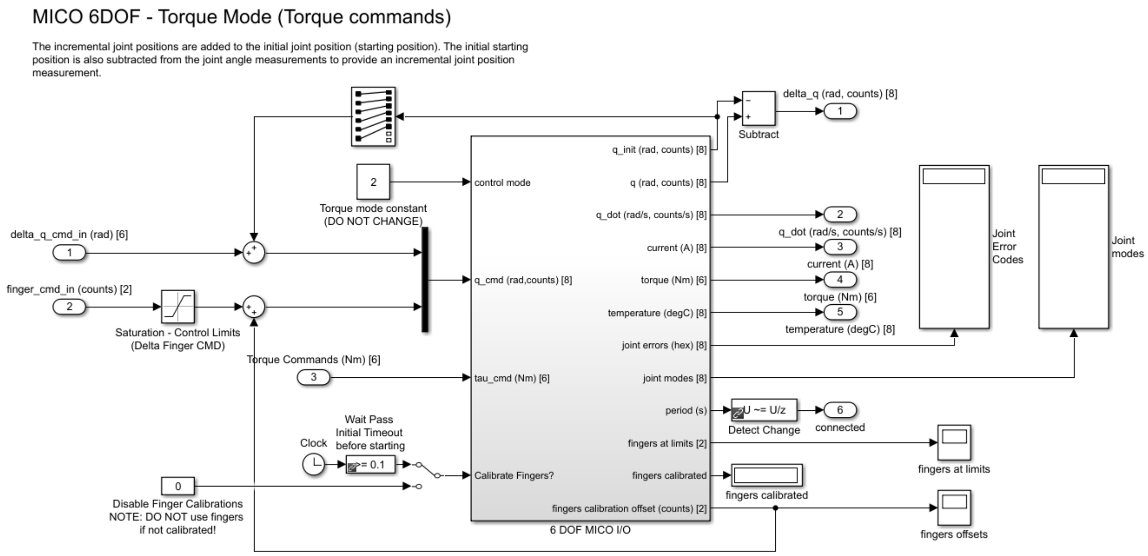



Figure 4.4: MICO 6 DOF - Torque Mode subsystem block

Quanser Inc.
119 Spy Court
Markham, Ontario
L3R 5H6
Canada
info@quanser.com
Phone: 1-905-940-3575
Fax: 1-905-940-3576

Printed in Markham, Ontario.

For more information on the solutions Quanser Inc. offers, please visit the web site at:
http://www.quanser.com