Lab Procedure for Simulink
Inverse Kinematics

Setup

1. It is recommended that you review Lab 2 – Application Guide before starting this lab.

2. Turn on the QBot Platform by pressing the power button once. To ensure the robot is ready for the lab, check the following conditions.

    a. The LEDs on the robot base should be solid red.

    b. The LCD should display the battery level. It is recommended that the battery level is over 12.5V.

    c. The Logitech F710 joystick's wireless receiver is connected to the QBot Platform. Before use, **always make sure the switch on top is in the X position and that the LED next to the Mode button is off.**

    d. Make sure your computer is connected to the same network that the QBot Platform is on. If using the provided router, the network should be `Quanser_UVS-5G`.

    e. Test connectivity to the QBot, using the IP displayed in the robot's LCD display, enter the following command in your local computer terminal and hit enter: `ping 192.168.2.x` .

3. Deploy and run qbot_platform_driver_physical on QBot Platform:

    a. Right click on qbot_platform_driver_physical.rt-linux_qbot_platform, select "Show more options", then select "Run on target".

    b. Change Target URI to: `tcpip://192.168.2.x:17000`

    c. Change Model Arguments to `-d /tmp -uri tcpip://192.168.2.x:17099`

    d. Click Run.

    e. The QBot Platform LEDs should pulse white if the driver is deployed and running successfully.

4. Open the Simulink Model inv_kin.slx, as shown in Figure 1. Configure the model so that it can be deployed to the QBot Platform:

    a. Open Hardware Settings under the Hardware ribbon in your model.

    b. Expand and browse to `Code Generation > Interface`.

    c. Change the MEX-file arguments to the following string including single quotes,

    `'-w -d /tmp -uri %u','tcpip://192.168.2.x:17001'`

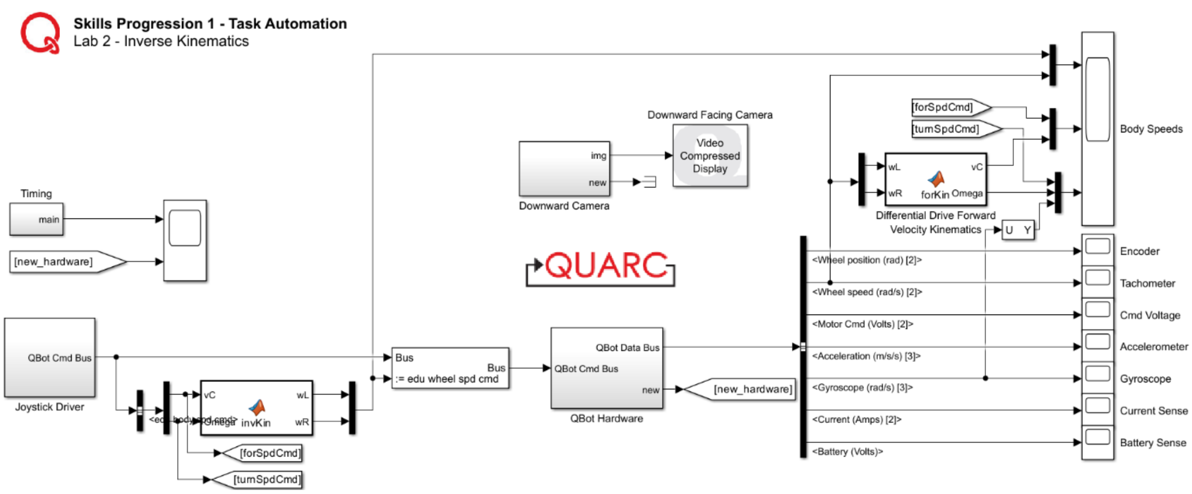

Figure 1. Inverse Kinematics Simulink Model

Inverse Kinematics Formulation

1. Double click **MATLAB Function** block labeled *forKin* and verify that it is completed for you. If not, copy over the completed forward kinematics function from the previous lab.

2. When considering a pure forward velocity for a differential drive robot, what do you expect each wheel to do? How about when turning?

3. When commanding a positive turn velocity for the QBot (counterclockwise), which wheel should spin faster?

4. Open the **MATLAB Function** block labeled *invKin*. This function is incomplete and must be compete before use. This function should take body speeds as input and output QBot wheel speeds in rad/s.

5. Based on the completed forward kinematics equations, derive the inverse kinematics equations, and complete *invKin*.

6. Close the function, run your Simulink model, and drive the QBot around. Using the *Body speeds* scope, verify that the output from your inverse kinematics function (commanded wheel velocities) closely match those from the tachometer. Revise your function if needed.

7. Open the Body speeds scope. How closely do the commanded body velocities match the measured velocities through forward kinematics?

8. When you're satisfied with the output of your function, take a screenshot of the scope windows.

9. What is the key difference between this application and the drive mode in the Play lab?

10. What is the maximum forward or turn speed of the robot you can command? What wheel velocity command does this correspond to?

11. Stop the Simulink model when complete. Ensure that you save a copy of your completed files for review later.

12. Turn OFF the robot by single pressing the power button (do not keep it pressed until it turns off). Post shutdown, all the LEDs should be completely OFF.