Lab Procedure

Routh-Hurwitz Stability

Introduction

Ensure the following:

1. You have reviewed the Application Guide – Routh-Hurwitz
2. Make sure you have Quanser Interactive Labs open in the Qube 3 - DC Motor → Servo Workspace.
3. Launch MATLAB and browse to the working directory that includes the Simulink models for this lab.

The **Hardware Interfacing** and **Filtering** labs explained the basic blocks to read and write from the Qube-Servo 3. For simplicity, all labs forward will use a Qube-Servo 3 block that sets up the system beforehand and outputs the available information from the Qube.

Using the gains found to convert encoder counts into rads from the instrumentation labs update the qs3_routh_hurwitz.slx file to match Figure 1. Leave the disconnected gain block.

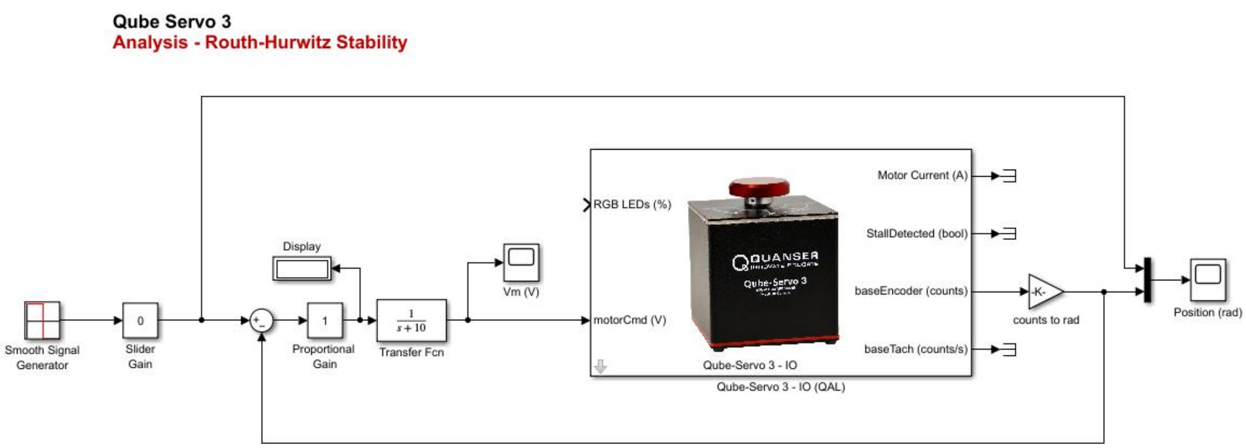


Figure 1: Simulink model used to test stability of Qube-Servo 3

Build the Simulink model shown in Figure 1, if it has not already been provided, use a previous model as a start.

Routh Stability Analysis

1. Given the feedback loop diagram in Figure 2, determine the closed-loop transfer function $\frac{Y(s)}{R(s)}$ in terms of k_p. Then substitute values for K and τ in from any of the modeling labs, or if no modeling lab has been done, $K = 24$ and $\tau = 0.1$ are good defaults.

   Use the following definitions for Transfer Functions:

$$P(s) = \frac{K}{s(\tau s + 1)}$$

$$C(s) = \frac{1}{s + 10}$$

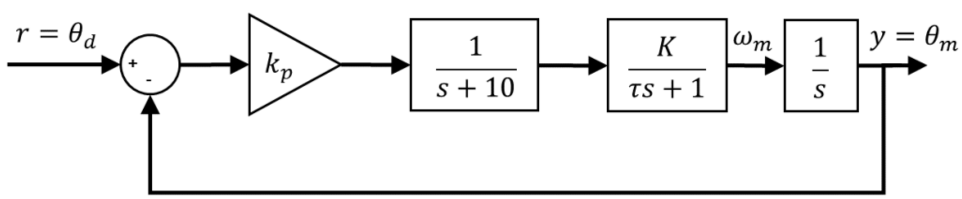


Figure 2: Closed-loop feedback block diagram.

2. Create a Routh Table for the closed-loop transfer function.

3. Find the range of k_p for which the system is stable.

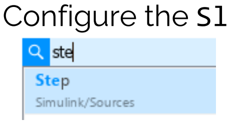
4. Is there a k_p for which the system is marginally stable? Comment on the pole locations of the closed-loop transfer function.

Testing Stability

Now that the theoretical stability of the system has been calculated, these calculations can be validated on hardware. Create or open the Simulink model from Figure 1.

1. Open the *qs3_routh_hurwitz* model shown in Figure 1.

2. Add a `Smooth Signal Generator` and a `Slider Gain` block into your model, by double clicking on an empty part of your model and typing the block name. Configure the signal generator to output a square wave with an amplitude of ± 5 at $0.4\,Hz$. Configure the `Slider Gain` with a low value of 0 and high of 10. t



3. Set the `Slider Gain` block to 0 to apply a motor position reference signal of zero rads (i.e. zero setpoint).

4. Set the `Proportional Gain` to 1.

5. Run the QUARC controller using the `Run` button on the `Simulation` tab.

6. While the controller is running, manually perturb the disc on the Qube-Servo 3 and attach a screenshot of the scopes that represents the behaviour observed. Is the response stable? Is there steady state error once you have perturbed the system? An example response is shown below in Figure 3:

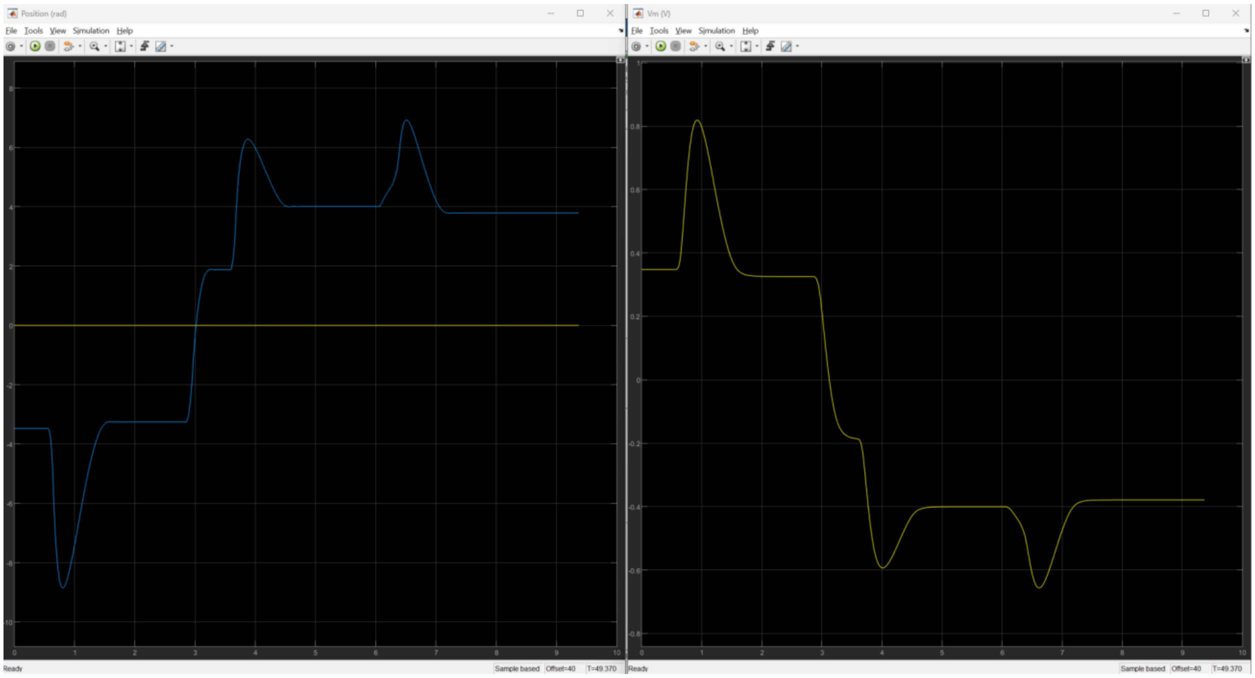


Figure 3: Example response when perturbing the disk with k_p =1.

7. While the controller is still running, slowly increase the proportional gain to about 5V/rad. You can double click the `Slider Gain` block and increase the value slowly.

8. Take a screenshot that is representative of any new behaviors. Take note of the effect this change has on the system response compared to a $k_p = 1$?

9. Increase the proportional gain until you reach the value for k_p for which the system is theoretically marginally stable. Take a screenshot that represents the behaviour at this k_p value.
   CAUTION: Be ready to turn OFF the Qube-Servo 3 from the switch in the back if the system goes unstable.

10. If the result was not marginally stable in the previous step, try gently tuning k_p until the system is marginally stable. Write down the k_p for which the system is marginally stable.
    NOTE: Use increments of 0.1 when tuning k_p.
    CAUTION: Be ready to turn OFF the Qube-Servo 3 from the switch in the back if the system goes unstable.

11. Stop and close your model. Ensure you save a copy of the files for review later.