

## PREFACE

Preparing laboratory experiments can be time-consuming. Quanser understands time constraints of teaching and research professors. That's why the QUBE-Servo 2 experiment comes with a new generation of mix-and-match courseware that allows easy adaptation of material to a specific course. The courseware is also aligned with requirements of ABET accreditation. All this allows professors to get their labs running faster, saving months of time typically required to develop lab materials and exercises.

Quanser QUBE-Servo 2 courseware provides step-by-step pedagogy for a wide range of control challenges. You can select a pre-defined lab sequence where students start with the basic principles and progress to more advanced applications of control theories. Or you can select a specific topic and use the exercises to supplement the theory students learnt in class with hands-on experience in lab.

To make the courseware easily adaptable to your specific course, Quanser also offers a comprehensive mapping of courseware topics to the most popular control engineering textbooks:

- Control Systems Engineering by Norman S. Nise
- Feedback Systems by K.J. Åström, R.M. Murray
- Feedback Control of Dynamic Systems by G.F. Franklin, J.D. Powell, A. Emai-Naeini
- Modern Control Systems by R.C. Dorf, R.H. Bishop
- Modern Control Engineering by K. Ogata
- Automatic Control Systems by F. Golnaraghi, B.C. Kuo
- Control Systems Engineering by I.J. Nagrath, M. Gopal
- Mechatronics by W. Bolton

***This document provides an abbreviated example of background and in-lab exercise courseware sections for the QUBE-Servo 2 experiment. Please note that the examples are not complete as they are intended to give you a brief overview of the structure and content of the courseware you will receive with the QUBE-Servo 2.***

This courseware sample based on the material prepared for users of MATLAB®/Simulink® software.



The QUBE-Servo courseware is aligned with requirements of ABET accreditation.



# 1. QUBE™-SERVO 2 COURSEWARE TABLE OF CONTENTS

The full Table of Contents of the Quanser QUBE-Servo 2 Instructor Workbook is shown here:

- 1. QUBE-SERVO 2 INTEGRATION**
  - 1.1. BACKGROUND
    - 1.1.1. QUARC SOFTWARE
    - 1.1.2. DC MOTOR
    - 1.1.3. ENCODERS
  - 1.2. IN-LAB EXERCISES
    - 1.2.1. CONFIGURING A SIMULINK MODEL FOR THE QUBE-SERVO 2
    - 1.2.2. READING THE ENCODER
    - 1.2.3. DRIVING THE DC MOTOR
- 2. FILTERING**
  - 2.1. BACKGROUND
  - 2.2. IN-LAB EXERCISES
- 3. STABILITY ANALYSIS**
  - 3.1. BACKGROUND
    - 3.1.1. SERVO MODEL
    - 3.1.2. STABILITY
  - 3.2. IN-LAB EXERCISES
- 4. BUMP TEST MODELING**
  - 4.1. BACKGROUND
    - 4.1.1. APPLYING THIS TO THE QUBE-SERVO 2
  - 4.2. IN-LAB EXERCISES
- 5. FIRST PRINCIPLES MODELING**
  - 5.1. BACKGROUND
  - 5.2. IN-LAB EXERCISES
- 6. SECOND-ORDER SYSTEMS**
  - 6.1. BACKGROUND
    - 6.1.1. SECOND-ORDER STEP RESPONSE
    - 6.1.2. PEAK TIME AND OVERSHOOT
    - 6.1.3. UNITY FEEDBACK
  - 6.2. IN-LAB EXERCISES
- 7. PD CONTROL**
  - 7.1. BACKGROUND
    - 7.1.1. SERVO MODEL
    - 7.1.2. PID CONTROL
    - 7.1.3. PV POSITION CONTROL
  - 7.2. IN-LAB EXERCISES
- 8. LEAD COMPENSATOR**
  - 8.1. BACKGROUND
    - 8.1.1. LEAD COMPENSATOR DESIGN PROCEDURE
  - 8.2. IN-LAB EXERCISE

- 9. PENDULUM MOMENT OF INERTIA**
  - 9.1. BACKGROUND
  - 9.2. IN-LAB EXERCISES
- 10. ROTARY PENDULUM MODELING**
  - 10.1. BACKGROUND
  - 10.2. IN-LAB EXERCISES
- 11. BALANCE CONTROL**
  - 11.1. BACKGROUND
  - 11.2. IN-LAB EXERCISES
- 12. STATE SPACE MODELING**
  - 12.1. BACKGROUND
    - 12.1.1. PENDULUM MODEL
    - 12.1.2. LINEAR STATE-SPACE MODEL
  - 12.2. IN-LAB EXERCISE
    - 12.2.1. PENDULUM STATE-SPACE MODEL
- 13. SWING-UP CONTROL**
  - 13.1. BACKGROUND
    - 13.1.1. ENERGY CONTROL
    - 13.1.2. HYBRID SWING-UP CONTROL
  - 13.2. IN-LAB EXERCISES
    - 13.2.1. ENERGY CONTROL
    - 13.2.2. HYBRID SWING-UP CONTROL
- 14. OPTIMAL LQR CONTROL**
  - 14.1. BACKGROUND
  - 14.2. IN-LAB EXERCISES
    - 14.2.1. LQR CONTROL DESIGN
    - 14.2.2. LQR-BASED BALANCE CONTROL

### 3. BACKGROUND SECTION - SAMPLE

#### Bump Test Modeling

The bump test is a simple test based on the step response of a stable system. A step input is given to the system and its response is recorded. As an example, consider a system given by the following transfer function:

$$\frac{Y(s)}{U(s)} = \frac{K}{\tau s + 1} \quad (1.1)$$

The step response shown in Figure 1.1 is generated using this transfer function with  $K = 5 \text{ rad/V.s}$  and  $\tau = 0.05 \text{ s}$ .

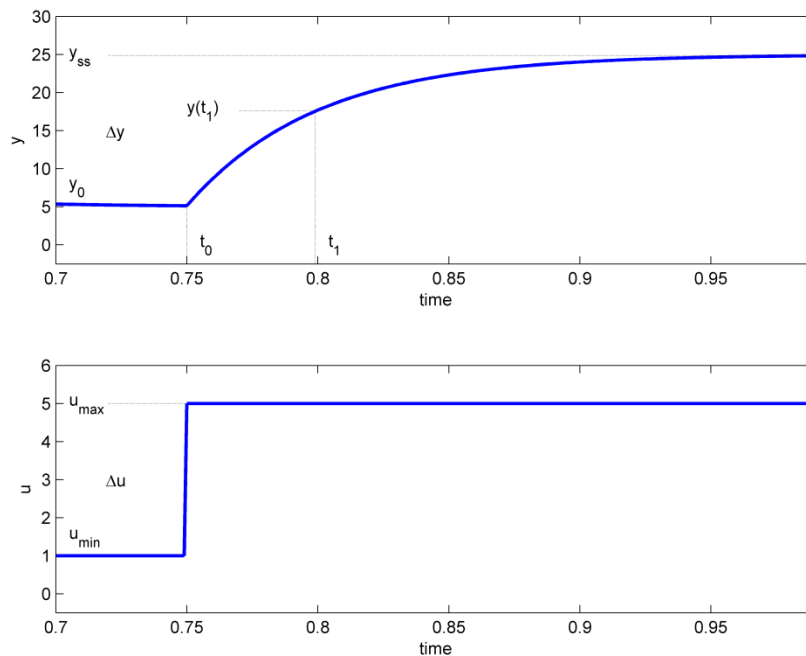


Figure 1.1: Input and output signal used in the bump test method

The step input begins at time  $t_0$ . The input signal has a minimum value of  $u_{min}$  and a maximum value of  $u_{max}$ . The resulting output signal is initially at  $y_0$ . Once the step is applied, the output tries to follow it and eventually settles at its steady-state value  $y_{ss}$ . From the output and input signals, the steady-state gain is

$$K = \frac{\Delta y}{\Delta u} \quad (1.2)$$

where  $\Delta y = y_{ss} - y_0$  and  $\Delta u = u_{max} - u_{min}$ . The time constant of the system  $\tau$  is defined as the time it takes the system to respond to the application of a step input to reach  $1 - 1/e \approx 63.2\%$  of its steady-state value, i.e. for Figure 1.1

$$t_1 = t_0 + \tau$$

where

$$y(t_1) = 0.632\Delta y + y_0 \quad (1.3)$$

Then, we can read the time  $t_1$  that corresponds to  $y(t_1)$  from the response data in Figure 1.1. From this, the model time constant can be found as:

$$\tau = t_1 - t_0 \quad (1.4)$$

## 4. IN-LAB EXERCISES - SAMPLE

### First Principle Modeling

Based on the models already designed in the Integration and Filtering laboratory experiment, design a model that applies a 1-3 V 0.4 Hz square wave to the motor and reads the servo velocity using the encoder as shown in Figure 2.1

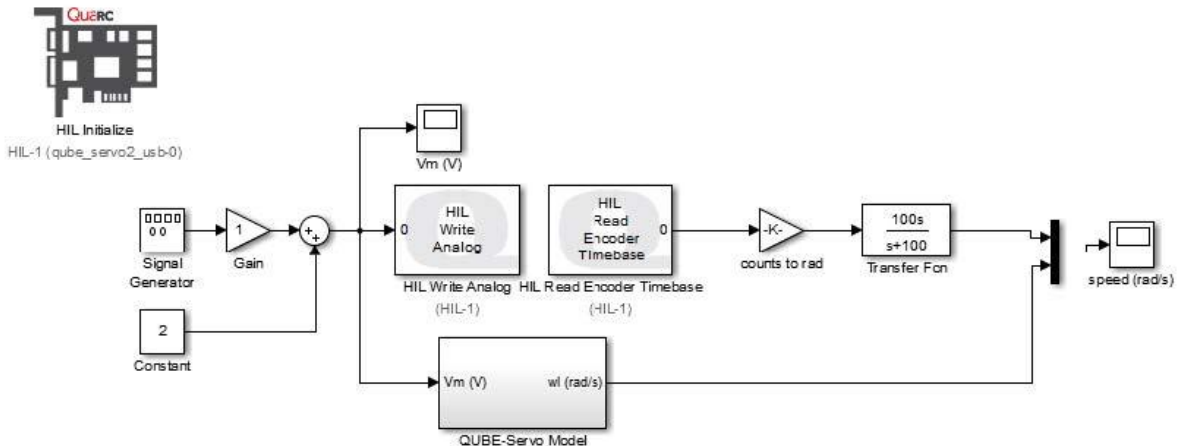


Figure 2.1: Applies a step voltage and displays measured and simulated QUBE-Servo 2 speed.

Create subsystem called *QUBE-Servo 2 Model*, as shown in Figure 2.1, that contains blocks to model the QUBE-Servo 2 system. Thus using the equations given above, assemble a simple block diagram in Simulink to model the system. You'll need a few Gain blocks, a Subtract block, and an Integrator block (to go from acceleration to speed). Part of the solution is shown in Figure 2.2.

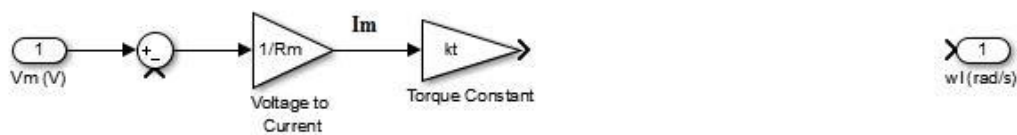


Figure 2.2: Incomplete *QUBE-Servo 2 Model* subsystem.

It may also help to write a short Matlab script that sets the various system parameters in Matlab, so you can use the symbol instead of entering the value numerically in the Gain blocks. In the example shown in Figure 2.2, we are using  $R_m$  for motor resistance and  $kt$  for the current-torque constant. To define these, write a script like:

```
% Resistance
Rm = 8.4;
% Current-torque (N-m/A)
kt = 0.042;
```

1. **A-1, A-2** The motor shaft of the QUBE-Servo 2 is attached to a *load hub* and a disc load. Based on the parameters given in Table 1.1, calculate the equivalent moment of inertia that is acting on the motor shaft.

**Answer 2.1**

**Outcome Solution**

A-1 From Figure 1.1, the total moment of inertia acting on the motor shaft is the sum of the motor armature / rotor inertia,  $J_m$ , the hub inertia,  $J_h$ , and the disc inertia,  $J_d$ . The equivalent moment of inertia is therefore

$$J_{eq} = J_m + J_h + J_d \quad (\text{Ans. 2.1})$$

Given the disc moment of inertia in Equation 1.7 and the parameters defined in Figure 1.1, the moment of inertia of the hub and disc load are:

$$J_h = \frac{1}{2} m_d r_h^2 \quad (\text{Ans. 2.2})$$

and

$$J_d = \frac{1}{2} m_d r_d^2 \quad (\text{Ans. 2.3})$$

A-2 Using the parameters from Table 1.1, evaluate Equation Ans.2.1 to obtain

$$J_{eq} = 4.0 \times 10^{-6} + \frac{1}{2} 0.0106(0.0111)^2 + \frac{1}{2} 0.053(0.0248)^2 = 2.09 \times 10^{-5} \quad (\text{Ans. 2.4})$$

□ □ □

2. **K-3** Design the *QUBE-Servo 2 Model* subsystem as described above. Attach a screen capture of your model and the Matlab script (if you used one).

**Answer 2.2**

**Outcome Solution**

K-3 The completed model is shown in Figure Ans. 2.1. The current depends on the angular rate of the shaft and the applied voltage, as expressed in Equation 1.4. The acceleration of the shaft equals the torque divided by the equivalent moment of inertia, as described in Equation 1.5. The Matlab script used for this is:

```
Rm = 8.4;
kt = 0.042;
km = 0.042;
Jr = 4e-6;
mh = 0.0106;
rh = 22.2/1000/2;
Jh = 0.5*mh*rh^2;
md = 0.053;
rd = 49.5/1000/2;
Jd = 0.5*md*rd^2;
Jeq = Jr + Jh + Jd;
```

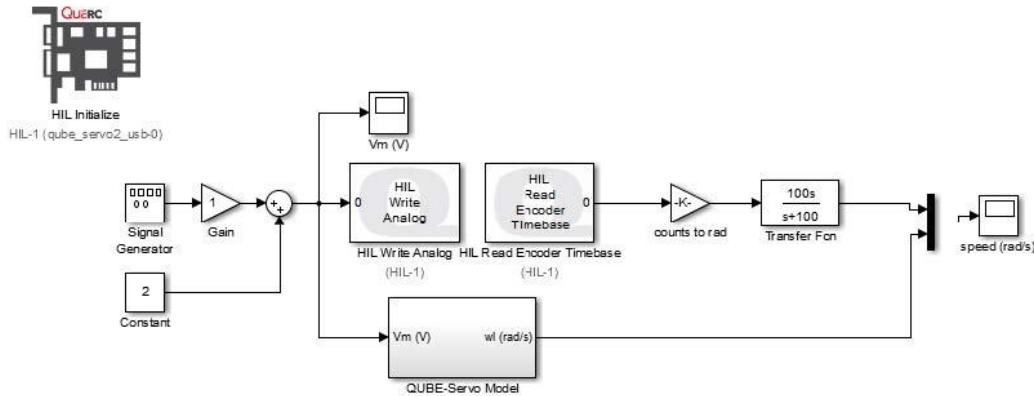
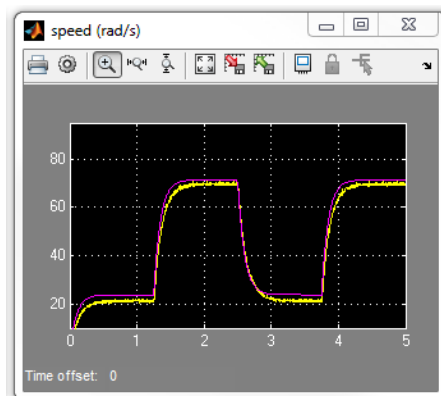


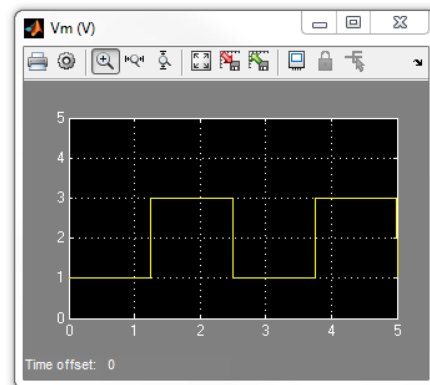
Figure Ans. 2.1: Completed *QUBE-Servo 2 Model* subsystem.

□ □ □

3. **B-5, B-9** Build and run the QUARC controller with your QUBE-Servo 2 model. The scope response should be similar to Figure 2.3. Attach a screen capture of your scopes. Does your model represent the QUBE-Servo 2 well? Explain.



(a) Motor Speed



(b) Motor Voltage

Figure 2.3: QUBE Response

**Answer 2.3**

**Outcome Solution**

B-5 If the experimental procedure was followed correctly, the user should be able to run the QUARC controller and obtain a response similar to Figure 2.3.

B-9 The model represents the actual QUBE-Servo 2 system accurately because the simulated response (purple) matches the measured response (yellow) quite well in Figure 2.3.

□ □ □

4. Stop the QUARC controller.
5. Power OFF the QUBE-Servo 2.