Cyber Exploration Laboratory Experiments

Cyber Exploration Laboratory

Experiment 1.1

Objective To verify the behavior of closed-loop systems as described in the Chapter 1 Case Study.

Minimum Required Software Packages LabVIEW and the LabVIEW Control Design and Simulation Module. *Note*: While no knowledge of LabVIEW is required for this experiment, see Appendix D to learn more about LabVIEW, which will be pursued in more detail in later chapters.

Prelab

1. From the discussion in the Case Study, describe the effect of the gain of a closed-loop system upon transient response.
2. From the discussion in the Case Study about steady-state error, sketch a graph of a step input superimposed with a step response output and show the steady-state error. Assume any transient response. Repeat for a ramp input and ramp output response. Describe the effect of gain upon the steady-state error.

Lab

1. Launch LabVIEW and open **Find Examples . . .**
2. In the **NI Example Finder** window, open **CDEx Effect of Controller Type.vi,** found by navigating to it though **Toolkits and Modules/Control and Simulation/ Control Design/Time Analysis/CDEx Effect of Controller Type vi.**
3. On the tool bar click the circulating arrows located next to the solid arrow on the left. The program is now running.
4. Move the slider **Controller Gain** and note the effect of high and low gains.
5. Change the controller by clicking the arrows for **Controller Type** and repeat Step 4.

Postlab

1. Correlate the responses found in the experiment with those described in your Prelab. Explore other examples provided in the LabVIEW example folders.

Cyber Exploration Laboratory

Experiment 2.1

Objectives To learn to use MATLAB to (1) generate polynomials, (2) manipulate polynomials, (3) generate transfer functions, (4) manipulate transfer functions, and (5) perform partial-fraction expansions.

Minimum Required Software Packages MATLAB and the Control System Toolbox

Prelab

1. Calculate the following by hand or with a calculator:

    a. The roots of $P_1 = s^6 + 7s^5 + 2s^4 + 9s^3 + 10s^2 + 12s + 15$

    b. The roots of $P_2 = s^6 + 9s^5 + 8s^4 + 9s^3 + 12s^2 + 15s + 20$

    c. $P_3 = P_1 + P_2$; $P_4 = P_1 - P_2$; $P_5 = P_1 P_2$

2. Calculate by hand or with a calculator the polynomial

$$P_6 = (s + 7)(s + 8)(s + 3)(s + 5)(s + 9)(s + 10)$$

3. Calculate by hand or with a calculator the following transfer functions:

    a. $G_1(s) = \dfrac{20(s + 2)(s + 3)(s + 6)(s + 8)}{s(s + 7)(s + 9)(s + 10)(s + 15)}$,

    represented as a numerator polynomial divided by a denominator polynomial.

    b. $G_2(s) = \dfrac{s^4 + 17s^3 + 99s^2 + 223s + 140}{s^5 + 32s^4 + 363s^3 + 2092s^2 + 5052s + 4320}$,

    expressed as factors in the numerator divided by factors in the denominator, similar to the form of $G_1(s)$ in Prelab 3**a.**

    c. $G_3(s) = G_1(s) + G_2(s)$; $G_4(s) = G_1(s) - G_2(s)$; $G_5(s) = G_1(s)G_2(s)$
    expressed as factors divided by factors and expressed as polynomials divided by polynomials.

4. Calculate by hand or with a calculator the partial-fraction expansion of the following transfer functions:

    a. $G_6 = \dfrac{5(s + 2)}{s(s^2 + 8s + 15)}$

    b. $G_7 = \dfrac{5(s + 2)}{s(s^2 + 6s + 9)}$

    c. $G_8 = \dfrac{5(s + 2)}{s(s^2 + 6s + 34)}$

Lab

1. Use MATLAB to find P_3, P_4, and P_5 in Prelab 1.
2. Use only one MATLAB command to find P_6 in Prelab 2.
3. Use only two MATLAB commands to find $G_1(s)$ in Prelab 3**a** represented as a polynomial divided by a polynomial.

4. Use only two MATLAB commands to find $G_2(s)$ expressed as factors in the numerator divided by factors in the denominator.

5. Using various combinations of $G_1(s)$ and $G_2(s)$, find $G_3(s)$, $G_4(s)$, and $G_5(s)$. Various combinations implies mixing and matching $G_1(s)$ and $G_2(s)$ expressed as factors and polynomials. For example, in finding $G_3(s)$, $G_1(s)$ can be expressed in factored form and $G_2(s)$ can be expressed in polynomial form. Another combination is $G_1(s)$ and $G_2(s)$ both expressed as polynomials. Still another combination is $G_1(s)$ and $G_2(s)$ both expressed in factored form.

6. Use MATLAB to evaluate the partial fraction expansions shown in Prelab 4.

Postlab

1. Discuss your findings for Lab 5. What can you conclude?

2. Discuss the use of MATLAB to manipulate transfer functions and polynomials. Discuss any shortcomings in using MATLAB to evaluate partial fraction expansions.

Experiment 2.2

Objectives To learn to use MATLAB and the Symbolic Math Toolbox to (1) find Laplace transforms for time functions, (2) find time functions from Laplace transforms, (3) create LTI transfer functions from symbolic transfer functions, and (4) perform solutions of symbolic simultaneous equations.

Minimum Required Software Packages MATLAB, the Symbolic Math Toolbox, and the Control System Toolbox

Prelab

1. Using a hand calculation, find the Laplace transform of:
$$f(t) = 0.0075 - 0.00034e^{-2.5t}\cos(22t) + 0.087e^{-2.5t}\sin(22t) - 0.0072e^{-8t}$$

2. Using a hand calculation, find the inverse Laplace transform of
$$F(s) = \frac{2(s+3)(s+5)(s+7)}{s(s+8)(s^2+10s+100)}$$

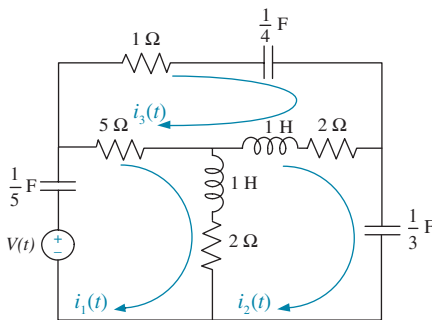**3.** Use a hand calculation to solve the circuit for the loop currents shown in Figure P2.42.

FIGURE P2.42

Lab

1. Use MATLAB and the Symbolic Math Toolbox to

a. Generate symbolically the time function $f(t)$ shown in Prelab 1.

b. Generate symbolically $F(s)$ shown in Prelab 2. Obtain your result symbolically in both factored and polynomial forms.

c. Find the Laplace transform of $f(t)$ shown in Prelab 1.

d. Find the inverse Laplace transform of $F(s)$ shown in Prelab 2.

e. Generate an LTI transfer function for your symbolic representation of $F(s)$ in Prelab 2 in both polynomial form and factored form. Start with the $F(s)$ you generated symbolically.

f. Solve for the loop currents in Prelab 3.

Postlab

1. Discuss the advantages and disadvantages between the Symbolic Math Toolbox and MATLAB alone to convert a transfer function from factored form to polynomial form and vice versa.

2. Discuss the advantages and disadvantages of using the Symbolic Math Toolbox to generate LTI transfer functions.

3. Discuss the advantages of using the Symbolic Math Toolbox to solve simultaneous equations of the type generated by the electrical network in Prelab 3. Is it possible to solve the equations via MATLAB alone? Explain.

4. Discuss any other observations you had using the Symbolic Math Toolbox.

Experiment 2.3

Objective   To learn to use LabVIEW to generate and manipulate polynomials and transfer functions.

Minimum Required Software Packages   LabVIEW and the LabVIEW Control Design and Simulation Module.

Prelab

1. Study Appendix D, Sections D.1 through Section D.4, Example D.1.

2. Perform by hand the calculations stated in Prelab 1 of Experiment 2.1.

3. Find by a hand calculation the polynomial whose roots are: $-7, -8, -3, -5, -9$, and -10.

4. Perform by hand a partial-fraction expansion of $G(s) = \dfrac{5s + 10}{s^3 + 8s^2 + 15s}$.

5. Find by a hand calculation $G_1(s) + G_2(s)$, $G_1(s) - G_2(s)$, and $G_1(s)G_2(s)$, where
$G_1(s) = \dfrac{1}{s^2 + s + 2}$ and $G_2(s) = \dfrac{s + 1}{s^2 + 4s + 3}$.

Lab

1. Open the LabVIEW functions palette and select the **Mathematics/Polynomial** palette.

2. Generate the polynomials enumerated in Prelab 1a and 1b of Experiment 2.1.

3. Generate the polynomial operations stated in Prelab 1c of Experiment 2.1.

4. Generate a polynomial whose roots are those stated in Prelab 3 of this experiment.

5. Generate the partial fraction expansion of the transfer function given in Prelab 4 of this experiment.

6. Using the **Control Design and Simulation/Control Design/Model Construction** palette, construct the two transfer functions enumerated in Prelab 5.

7. Using the **Control Design and Simulation/Control Design/Model Inter-connection** palette, display the results of the mathematical operations enumerated in Prelab 5 of this experiment.

Postlab

1. Compare the polynomial operations obtained in Lab 3 to those obtained in Prelab 2.

2. Compare the polynomial displayed in Lab 4 with that calculated in Prelab 3.

3. Compare the partial-fraction expansion obtained in Lab 5 with that calculated in Prelab 4.

4. Compare the results of the mathematical operations found in Lab 7 to those calculated in Prelab 5.

Experiment 3.1

Objectives To learn to use MATLAB to (1) generate an LTI state-space representation of a system and (2) convert an LTI state-space representation of a system to an LTI transfer function.

Minimum Required Software Packages MATLAB and the Control System Toolbox

Prelab

1. Derive the state-space representation of the translational mechanical system shown in Skill-Assessment Exercise 3.2 if you have not already done so. Consider the output to be $x_3(t)$.
2. Derive the transfer function, $\frac{X_3(s)}{F(s)}$, from the equations of motion for the translational mechanical system shown in Skill-Assessment Exercise 3.2.

Lab

1. Use MATLAB to generate the LTI state-space representation derived in Prelab 1.
2. Use MATLAB to convert the LTI state-space representation found in Lab 1 to the LTI transfer function found in Prelab 2.

Postlab

1. Compare your transfer functions as found from Prelab 2 and Lab 2.
2. Discuss the use of MATLAB to create LTI state-space representations and the use of MATLAB to convert these representations to transfer functions.

Experiment 3.2

Objectives To learn to use MATLAB and the Symbolic Math Toolbox to (1) find a symbolic transfer function from the state-space representation and (2) find a state-space representation from the equations of motion.

Minimum Required Software Packages MATLAB, the Symbolic Math Toolbox, and the Control System Toolbox

Prelab

1. Perform Prelab 1 and Prelab 2 of Experiment 3.1 if you have not already done so.
2. Using the equation $T(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}$ to find a transfer function from a state-space representation, write a MATLAB program using the Symbolic Math Toolbox to find the symbolic transfer function from the state-space representation of the translational mechanical system shown in Skill-Assessment Exercise 3.2 and found as a step in Prelab 1.
3. Using the equations of motion of the translational mechanical system shown in Skill-Assessment Exercise 3.2 and found in Prelab 1, write a symbolic MATLAB program to find the transfer function, $\frac{X_3(s)}{F(s)}$, for this system.

Lab

1. Run the programs composed in Prelabs 2 and Prelab 3 and obtain the symbolic transfer functions by the two methods.

Postlab

1. Compare the symbolic transfer function obtained from $T(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}$ with the symbolic transfer function obtained from the equations of motion.
2. Discuss the advantages and disadvantages between the two methods.
3. Describe how you would obtain an LTI state-space representation and an LTI transfer function from your symbolic transfer function.

Experiment 3.3

Objectives   To learn to use LabVIEW to (1) generate state-space representations of transfer functions, (2) generate transfer functions from state-space representations, and (3) verify that there are multiple state-space representations for a transfer function.

Minimum Required Software Packages   LabVIEW, the LabVIEW Control Design and Simulation Module, and the MathScript RT Module.

Prelab

1. Study Appendix D, Sections D.1 through Section D.4, Example D.1.
2. Solve Skill-Assessment Exercise 3.3 in Chapter 3.
3. Use your solution to Prelab 2 and convert back to the transfer function.

Lab

1. Use LabVIEW to convert the transfer function, $G(s) = \dfrac{2s + 1}{s^2 + 7s + 9}$, into a state-space representation using both the graphical and MathScript approaches. The front panel will contain controls for the entry of the transfer function and indicators of the transfer function and the two state-space results. Functions for this experiment can be found in the following palettes: (1) **Control Design and Simulation/Control Design/Model Construction,** (2) **Control Design and Simulation/Control Design/Model Conversion,** and (3) **Programming/Structures** Hint: Coefficients are entered in reverse order when using MathScript with MATLAB.

2. Use LabVIEW to convert all state-space representations found in Lab 1 to a transfer function. All state-space conversions should yield the transfer function given in Lab 1. The front panel will contain controls for entering state-space representations and indicators of the transfer function results as well as the state equations used.

Postlab

1. Describe any correlation found between the results of Lab 1 and calculations made in the Prelab.
2. Describe and account for any differences between the results of Lab 1 and calculations made in the Prelab.
3. Explain the results of Lab 2 and draw conclusions from the results.

Experiment 4.1

Objective To evaluate the effect of pole and zero location upon the time response of first- and second-order systems.

Minimum Required Software Packages MATLAB, Simulink, and the Control System Toolbox

Prelab

1. Given the transfer function $G(s) = \dfrac{a}{s+a}$, evaluate settling time and rise time for the following values of a: 1, 2, 3, 4. Also, plot the poles.

2. Given the transfer function $G(s) = \dfrac{b}{s^2 + as + b}$:

   a. Evaluate percent overshoot, settling time, peak time, and rise time for the following values: $a = 4$, $b = 25$. Also, plot the poles.

   b. Calculate the values of a and b so that the imaginary part of the poles remains the same but the real part is increased two times over that of Prelab 2**a**, and repeat Prelab 2**a**.

   c. Calculate the values of a and b so that the imaginary part of the poles remains the same but the real part is decreased $\frac{1}{2}$ time over that of Prelab 2**a**, and repeat Prelab 2**a**.

3. **a.** For the system of Prelab 2**a**, calculate the values of a and b so that the real part of the poles remains the same but the imaginary part is increased two times over that of Prelab 2**a**, and repeat Prelab 2**a**.

   b. For the system of Prelab 2**a**, calculate the values of a and b so that the real part of the poles remains the same but the imaginary part is increased four times over that of Prelab 2**a**, and repeat Prelab 2**a**.

4. **a.** For the system of Prelab 2**a**, calculate the values of a and b so that the damping ratio remains the same but the natural frequency is increased two times over that of Prelab 2**a**, and repeat Prelab 2**a**.

   b. For the system of Prelab 2**a**, calculate the values of a and b so that the damping ratio remains the same but the natural frequency is increased four times over that of Prelab 2**a**, and repeat Prelab 2**a**.

5. Briefly describe the effects on the time response as the poles are changed in each of Prelab 2, 3, and 4.

Lab

1. Using Simulink, set up the systems of Prelab 1 and plot the step response of each of the four transfer functions on a single graph by using the Simulink LTI Viewer. Also, record the values of settling time and rise time for each step response.

2. Using Simulink, set up the systems of Prelab 2. Using the Simulink LTI Viewer, plot the step response of each of the three transfer functions on a single graph. Also, record the values of percent overshoot, settling time, peak time, and rise time for each step response.

3. Using Simulink, set up the systems of Prelab 2**a** and Prelab 3. Using the Simulink LTI Viewer, plot the step response of each of the three transfer functions on a single graph. Also, record the values of percent overshoot, settling time, peak time, and rise time for each step response.

4. Using Simulink, set up the systems of Prelab 2**a** and Prelab 4. Using the Simulink LTI Viewer, plot the step response of each of the three transfer functions on a single graph. Also, record the values of percent overshoot, settling time, peak time, and rise time for each step response.

Postlab

1. For the first-order systems, make a table of calculated and experimental values of settling time, rise time, and pole location.

2. For the second-order systems of Prelab 2, make a table of calculated and experimental values of percent overshoot, settling time, peak time, rise time, and pole location.

3. For the second-order systems of Prelab 2**a** and Prelab 3, make a table of calculated and experimental values of percent overshoot, settling time, peak time, rise time, and pole location.

4. For the second-order systems of Prelab 2**a** and Prelab 4, make a table of calculated and experimental values of percent overshoot, settling time, peak time, rise time, and pole location.

5. Discuss the effects of pole location upon the time response for both first- and second-order systems. Discuss any discrepancies between your calculated and experimental values.

Experiment 4.2

Objective To evaluate the effect of additional poles and zeros upon the time response of second-order systems.

Minimum Required Software Packages MATLAB, Simulink, and the Control System Toolbox

Prelab

1. **a.** Given the transfer function $G(s) = \dfrac{25}{s^2 + 4s + 25}$, evaluate the percent overshoot, settling time, peak time, and rise time. Also, plot the poles.

   b. Add a pole at -200 to the system of Prelab 1**a**. Estimate whether the transient response in Prelab 1**a** will be appreciably affected.

   c. Repeat Prelab 1**b** with the pole successively placed at -20, -10, and -2.

2. A zero is added to the system of Prelab 1**a** at -200 and then moved to -50, -20, -10, -5, and -2. List the values of zero location in the order of the greatest to the least effect upon the pure second-order transient response.

3. Given the transfer function $G(s) = \dfrac{(25b/a)(s + a)}{(s + b)(s^2 + 4s + 25)}$, let $a = 3$ and $b = 3.01$, 3.1, 3.3, 3.5, and 4.0. Which values of b will have minimal effect upon the pure second-order transient response?

4. Given the transfer function $G(s) = \dfrac{(2500b/a)(s + a)}{(s + b)(s^2 + 40s + 2500)}$, let $a = 30$ and $b = 30.01$, 30.1, 30.5, 31, 35, and 40. Which values of b will have minimal effect upon the pure second-order transient response?

Lab

1. Using Simulink, add a pole to the second-order system of Prelab 1**a** and plot the step responses of the system when the higher-order pole is nonexistent, at $-200, \ -20, \ -10$, and -2. Make your plots on a single graph, using the Simulink LTI Viewer. Normalize all plots to a steady-state value of unity. Record percent overshoot, settling time, peak time, and rise time for each response.

2. Using Simulink, add a zero to the second-order system of Prelab 1**a** and plot the step responses of the system when the zero is nonexistent, at $-200, \ -50$, $-20, \ -10, \ -5$, and -2. Make your plots on a single graph, using the Simulink LTI Viewer. Normalize all plots to a steady-state value of unity. Record percent overshoot, settling time, peak time, and rise time for each response.

3. Using Simulink and the transfer function of Prelab 3 with $a = 3$, plot the step responses of the system when the value of b is 3, 3.01, 3.1, 3.3, 3.5, and 4.0. Make your plots on a single graph using the Simulink LTI Viewer. Record percent overshoot, settling time, peak time, and rise time for each response.

4. Using Simulink and the transfer function of Prelab 4 with $a = 30$, plot the step responses of the system when the value of b is 30, 30.01, 30.1, 30.5, 31, 35, and 40. Make your plots on a single graph, using the Simulink LTI Viewer. Record percent overshoot, settling time, peak time, and rise time for each response.

Postlab

1. Discuss the effect upon the transient response of the proximity of a higher-order pole to the dominant second-order pole pair.

2. Discuss the effect upon the transient response of the proximity of a zero to the dominant second-order pole pair. Explore the relationship between the length of the vector from the zero to the dominant pole and the zero's effect upon the pure second-order step response.

3. Discuss the effect of pole-zero cancellation upon the transient response of a dominant second-order pole pair. Allude to how close the canceling pole and zero should be and the relationships of (1) the distance between them and (2) the distance between the zero and the dominant second-order poles.

Experiment 4.3

Objective   To use LabVIEW Control Design and Simulation Module for time performance analysis of systems.

Minimum Required Software Packages   LabVIEW with the Control Design and Simulation Module

Prelab   One of the experimental direct drive robotic arms built at the MTT Artificial Intelligence Laboratory and the CMU Robotics Institute can be represented as a feedback control system with a desired angular position input for the robot's joint position and an angular position output representing the actual robot's joint position.

The forward path consists of three transfer functions in cascade; (1) a compensator, $G_c(s)$, to improve performance; (2) a power amplifier of gain, $K_a = 1$; and (3) the

transfer function of the motor and load, $G(s) = 2292/s(s + 75.6)$. Assume a unity-feedback system. Initially the system will be controlled with $G_c(s) = 0.6234$, which is called a proportional controller (*McKerrow, 1991*).

1. Obtain the closed-loop system transfer function and use MATLAB to make a plot of the resulting unit step response.
2. Repeat with $G_c(s) = 3.05 + 0.04s$, which is called a PD controller.
3. Compare both responses and draw conclusions regarding their time domain specifications.

Lab   Create a LabVIEW VI that uses a simulation loop to implement both controllers given in the Prelab. Plot the responses on the same graph for comparison purposes.

Postlab   Compare the responses obtained using your LabVIEW VI with those obtained in the Prelab.

Experiment 4.4

Objective   To use the LabVIEW Control Design and Simulation Module to evaluate the effect of pole location upon the time response of second-order systems.

Minimum Required Software Packages   LabVIEW with the Control Design and Simulation Module.

Prelab   Solve the Cyber Exploration Laboratory Experiment 4.1 Prelab, Part 2.

Lab   Build a LabVIEW VI to implement the functions studied in the Prelab of Cyber Exploration Laboratory 4.I, Part 2.
　　　Specifically for Prelab Part **a**, your front panel will have the coefficients of the second-order transfer function as inputs. The front panel will also have the following indicators: (1) the transfer function; (2) the state-space representation; (3) the pole locations; (4) the step response graph; (5) the time response of the two states on the same graph; (6) the time response parametric data including rise time, peak time, settling time, percent overshoot, peak value; and final value.
　　　For Prelab, Part **b**, your front panel will also have the following indicators: (1) the step response graph, and (2) the parametric data listed above for Prelab, Part **a**, but specific to Part **b**.
　　　For Prelab, Part **c**, your front panel will also have the following indicators: (1) the step response graph, and (2) the parametric data listed above for Prelab, Part **a**, but specific to Part **c**.
　　　Run the VI to obtain the data from the indicators.

Postlab   Use your results to discuss the effect of pole location upon the step response.

Experiment 5.1

Objectives To verify the equivalency of the basic forms, including cascade, parallel, and feedback forms. To verify the equivalency of the basic moves, including moving blocks past summing junctions, and moving blocks past pickoff points.

Minimum Required Software Packages MATLAB, Simulink, and the Control System Toolbox

Prelab

1. Find the equivalent transfer function of three cascaded blocks, $G_1(s) = \dfrac{1}{s+1}$. $G_2(s) = \dfrac{1}{s+4}$, and $G_3(s) = \dfrac{s+3}{s+5}$.

2. Find the equivalent transfer function of three parallel blocks, $G_1(s) = \dfrac{1}{s+4}$. $G_2(s) = \dfrac{1}{s+4}$, and $G_3(s) = \dfrac{s+3}{s+5}$.

3. Find the equivalent transfer function of the negative feedback system of Figure P5.55 if $G(s) = \dfrac{s+1}{s(s+2)}$, and $H(s) = \dfrac{s+3}{s+4}$.

FIGURE P5.55

4. For the system of Prelab 3, push $H(s)$ to the left past the summing junction and draw the equivalent system.

5. For the system of Prelab 3, push $H(s)$ to the right past the pickoff point and draw the equivalent system.

Lab
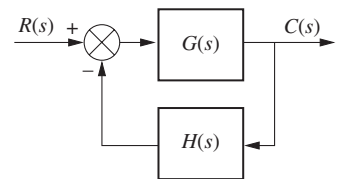
1. Using Simulink, set up the cascade system of Prelab 1 and the equivalent single block. Make separate plots of the step response of the cascaded system and its equivalent single block. Record the values of settling time and rise time for each step response.

2. Using Simulink, set up the parallel system of Prelab 2 and the equivalent single block. Make separate plots of the step response of the parallel system and its equivalent single block. Record the values of settling time and rise time for each step response.

3. Using Simulink, set up the negative feedback system of Prelab 3 and the equivalent single block. Make separate plots of the step response of the negative feedback system and its equivalent single block. Record the values of settling time and rise time for each step response.

4. Using Simulink, set up the negative feedback systems of Prelabs 3, 4, and 5. Make separate plots of the step response of each of the systems. Record the values of settling time and rise time for each step response.

Postlab

1. Using your lab data, verify the equivalent transfer function of blocks in cascade.

2. Using your lab data, verify the equivalent transfer function of blocks in parallel.

3. Using your lab data, verify the equivalent transfer function of negative feedback systems.

4. Using your lab data, verify the moving of blocks past summing junctions and pickoff points.
5. Discuss your results. Were the equivalencies verified?

Experiment 5.2

Objective To use the various functions within LabVIEW's Control Design and Simulation Module to implement block diagram reduction.

Minimum Required Software Package LabVIEW with the Control Design Simulation Module

Prelab Given the block diagram from Example 5.2, replace $G_1, G_2, G_3, H_1, H_2, H_3$ with the following transfer functions and obtain an equivalent transfer function.

$$G_1 = \frac{1}{s+10}; G_2 = \frac{1}{s+1}; G_3 = \frac{s+1}{s^2+4s+4}; H_1 = \frac{s+1}{s+2}; H_2 = 2; H_3 = 1$$

Lab Use LabVIEW to implement the block diagram from Example 5.2 using the transfer functions given in the Prelab.

Postlab Verify your calculations from the Prelab with that of the equivalent transfer function obtained with LabVIEW.

Experiment 5.3

Objective To use the various functions within LabVIEW's Control Design and Simulation Module and the Mathematics/Polynomial palette to implement Mason's rule for block diagram reduction.

Minimum Required Software Package LabVIEW with Control Design and Simulation Module, Math Script RT Module, and the Mathematics/Polynomial palette.

Prelab Given the block diagram created in the Prelab of Cyber Exploration Laboratory 5.2, use Mason's rule to obtain an equivalent transfer function.

Lab Use LabVIEW's Control Design and Simulation Module as well as the Mathematics/Polynomial functions to implement block diagram reduction using Mason's rule.

Postlab Verify your calculations from the Prelab with that of the equivalent transfer function obtained with LabVIEW.

Cyber Exploration Laboratory

Experiment 6.1

Objectives   To verify the effect of pole location upon stability. To verify the effect upon stability of loop gain in a negative feedback system.

Minimum Required Software Packages   MATLAB, Simulink, and the Control System Toolbox

FIGURE P6.22

Prelab

1. Find the equivalent transfer function of the negative feedback system of Figure P6.22 if

$$G(s) = \frac{K}{s(s+2)^2} \text{ and } H(s) = 1$$

2. For the system of Prelab 1, find two values of gain that will yield closed-loop, overdamped, second-order poles. Repeat for underdamped poles.

3. For the system of Prelab 1, find the value of gain, K, that will make the system critically damped.

4. For the system of Prelab 1, find the value of gain, K, that will make the system marginally stable. Also, find the frequency of oscillation at that value of K that makes the system marginally stable.

5. For each of Prelab 2 through 4, plot on one graph the pole locations for each case and write the corresponding value of gain, K, at each pole.

Lab

1. Using Simulink, set up the negative feedback system of Prelab 1. Plot the step response of the system at each value of gain calculated to yield overdamped, underdamped, critically damped, and marginally stable responses.

2. Plot the step responses for two values of gain, K, above that calculated to yield marginal stability.

3. At the output of the negative feedback system, cascade the transfer function

$$G_1(s) = \frac{1}{s^2 + 4}$$

Set the gain, K, at a value below that calculated for marginal stability and plot the step response. Repeat for K calculated to yield marginal stability.

Postlab

1. From your plots, discuss the conditions that lead to unstable responses.

2. Discuss the effect of gain upon the nature of the step response of a closed-loop system.

Experiment 6.2

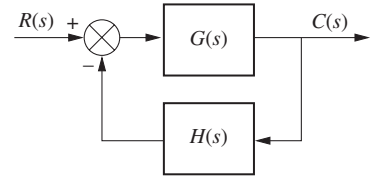**Objective**   To use the LabVIEW Control Design and Simulation Module for stability analysis.

Minimum Required Software Package   LabVIEW with the Control Design and Simulation Module

Prelab   Select six transfer functions of various orders and use Routh-Hurwitz to determine their stability.

Lab   Create a LabVIEW VI that receives the order and the coefficients of the characteristic equation and outputs the location of the poles and information regarding stability.

Postlab   Verify the stability of the systems from your Prelab.

Experiment 7.1

Objective   To verify the effect of input waveform, loop gain, and system type upon steady-state errors.

Minimum Required Software Packages   MATLAB, Simulink, and the Control System Toolbox

Prelab

1. What system types will yield zero steady-state error for step inputs?
2. What system types will yield zero steady-state error for ramp inputs?
3. What system types will yield infinite steady-state error for ramp inputs?
4. What system types will yield zero steady-state error for parabolic inputs?
5. What system types will yield infinite steady-state error for parabolic inputs?

FIGURE P7.36

6. For the negative feedback system of Figure P7.36, where $G(s) = \dfrac{K(s+6)}{(s+4)(s+7)(s+9)(s+12)}$ and $H(s) = 1$, calculate the steady-state error in terms of K for the following inputs: $5u(t)$, $5tu(t)$, and $5t^2u(t)$.
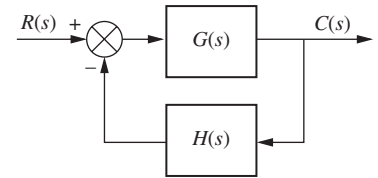
7. Repeat Prelab 6 for $G(s) = \dfrac{K(s+6)(s+8)}{s(s+4)(s+7)(s+9)(s+12)}$ and $H(s) = 1$.

8. Repeat Prelab 6 for $G(s) = \dfrac{K(s+1)(s+6)(s+8)}{s^2(s+4)(s+7)(s+9)(s+12)}$ and $H(s) = 1$.

Lab

1. Using Simulink, set up the negative feedback system of Prelab 6. Plot on one graph the error signal of the system for an input of $5u(t)$ and $K = 50, 500, 1000$, and 5000. Repeat for inputs of $5tu(t)$ and $5t^2u(t)$.

2. Using Simulink, set up the negative feedback system of Prelab 7. Plot on one graph the error signal of the system for an input of $5u(t)$ and $K = 50, 500, 1000$, and 5000. Repeat for inputs of $5tu(t)$ and $5t^2u(t)$.

3. Using Simulink, set up the negative feedback system of Prelab 8. Plot on one graph the error signal of the system for an input of $5u(t)$ and $K = 200, 400, 800$, and 1000. Repeat for inputs of $5tu(t)$ and $5t^2u(t)$.

Postlab

1. Use your plots from Lab 1 and compare the expected steady-state errors to those calculated in the Prelab. Explain the reasons for any discrepancies.

2. Use your plots from Lab 2 and compare the expected steady-state errors to those calculated in the Prelab. Explain the reasons for any discrepancies.

3. Use your plots from Lab 3 and compare the expected steady-state errors to those calculated in the Prelab. Explain the reasons for any discrepancies.

Experiment 7.2

Objective   To use the LabVIEW Control Design and Simulation Module for analysis of steady-state performance for step and ramp inputs.

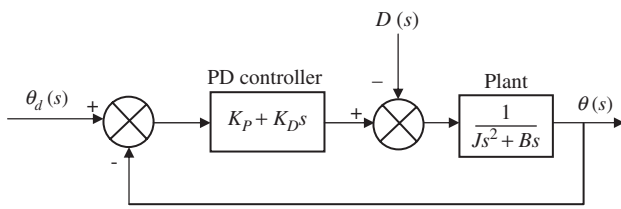**Minimum Required Software Package** Lab VIEW with the Control Design and Simulation Module

FIGURE P7.37

Prelab You are given the model of a single joint of a robotic manipulator shown in Figure P7.37 (*Spong, 2005*), where B is the coefficient of viscous friction, $\theta_d(s)$ is the desired angle, $\theta(s)$ is the output angle, and $D(s)$ is the disturbance. We want to track the joint angle using a PD controller, which we will study in Chapter 9. Assume $J = B = 1$. Find the step and ramp responses of this system for the following combinations of PD gains (K_P, K_D): (16, 7), (64, 15), and (144, 23).

Lab

1. Create a LabVIEW VI to simulate the response of this system to a step and a ramp inputs, under no-disturbance conditions. Use the functions available in the **Control Design and Simulation/Control Design** palette.

2. Create a LabVIEW VI using the functions available in the **Control Design and Simulation/Simulation** palette, to track an input set-point of 10 under a disturbance of $D = 40$.

Postlab Compare your results with those of the Prelab. What conclusions can you draw from the various responses of this system to different inputs and different PD parameters? What is the system type? Does the steady-state behavior corroborate the theory you learned regarding system type and the steady-state error for various inputs? Explain your answer.

Experiment 8.1

Objective To verify the effect of open-loop poles and zeros upon the shape of the root locus. To verify the root locus as a tool for estimating the effect of open-loop gain upon the transient response of closed-loop systems.

Minimum Required Software Packages MATLAB and the Control System Toolbox

Prelab

1. Sketch two possibilities for the root locus of a unity negative-feedback system with the open-loop pole-zero configuration shown in Figure P8.27.
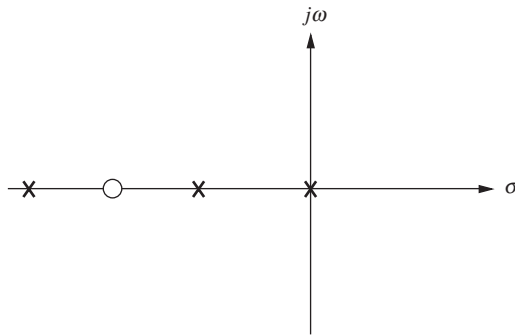

FIGURE P8.27

2. If the open-loop system of Prelab 1 is $G(s) = \dfrac{K(s+1.5)}{s(s+0.5)(s+10)}$, estimate the percent overshoot at the following values of gain, K: 20, 50, 85, 200, and 700.

Lab

1. Using Matlab's SISO Design Tool, set up a negative unity feedback system with $G(s) = \dfrac{K(s+6)}{s(s+0.5)(s+10)}$ to produce a root locus. For convenience, set up the zero at -6 using SISO Design Tool's compensator function by simply dragging a zero to -6 on the resulting root locus. Print the root locus for the zero at -6. Move the zero to the following locations and print out a root locus at each location: -2, -1.5, -1.37, and -1.2.

2. Using Matlab's SISO Design Tool, set up a negative unity feedback system with $G(s) = \dfrac{K(s+1.5)}{s(s+0.5)(s+10)}$ to produce a root locus. Open the LTI Viewer for SISO Design Tool to show step responses. Using the values of K specified in Prelab 2, record the percent overshoot and settling time and print the root loci and step response for each value of K.

Postlab

1. Discuss your findings from Prelab 1 and Lab 1. What conclusions can you draw?

2. Make a table comparing percent overshoot and settling time from your calculations in Prelab 2 and your experimental values found in Lab 2. Discuss the reasons for any discrepancies. What conclusions can you draw?
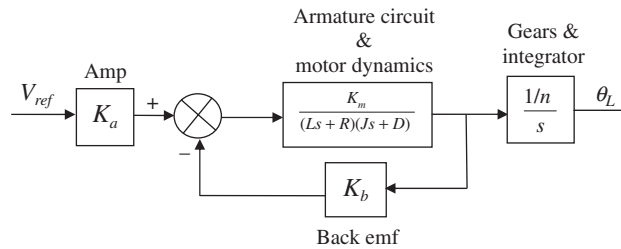
Experiment 8.2

Objective To use MATLAB to design the gain of a controller via root locus.

Minimum Required Software Package MATLAB with the Control Systems Toolbox.

Prelab The open-loop system dynamics model for the NASA eight-axis Advanced Research Manipulator II (ARM II) electromechanical shoulder joint/link, actuated by an armature-controlled dc servomotor is shown in Figure P8.28.

The ARM II shoulder joint constant parameters are $K_a = 12$, $L = 0.006$ H, $R = 1.4\,\Omega$, $K_b = 0.00867$, $n = 200$, $K_m = 4.375$, $J = J_m + J_L/n^2$, $D = D_m + D_L/n^2$, $J_L = 1$, $D_L = 0.5$, $J_m = 0.00844$, and $D_m = 0.00013$(*Craig, 2005*), (*Nyzen, 1999*), (*Williams, 1994*).

a. Obtain the equivalent open-loop transfer function, $G(s) = \dfrac{\theta_L(s)}{V_{ref}(s)}$.

b. The loop is to be closed by cascading a controller, $G_c(s) = K_D s + K_P$, with $G(s)$ in the forward path forming an equivalent forward-transfer function, $G_e(s) = G_c(s)G(s)$. Parameters of $G_c(s)$ will be used to design a desired transient performance. The input to the closed-loop system is a voltage, $V_I(s)$, representing the desired angular displacement of the robotic joint with a ratio of 1 volt equals 1 radian. The output of the closed-loop system is the actual angular displacement of the joint, $\theta_L(s)$. An encoder in the feedback path, K_e, converts the actual joint displacement to a voltage with a ratio of 1 radian equals 1 volt. Draw the closed-loop system showing all transfer functions.

c. Find the closed-loop transfer function.

FIGURE P8.28 Open-loop model for ARM ll

Lab Let $\dfrac{K_P}{K_D} = 4$ and use MATLAB to design the value of K_D to yield a step response with a maximum percent overshoot of 0.2%.

Postlab

1. Discuss the success of your design.

2. Is the steady-state error what you would expect? Give reasons for your answer.

Experiment 8.3

Objective To use LabVIEW to design the gain of a controller via root locus.

Minimum Required Software Package   LabVIEW with the Control Design and Simulation Module, and the MathScript RT Module.

Prelab   Complete the Prelab to Experiment 8.2 if you have not already done so.

Lab   Let $\dfrac{K_P}{K_D} = 4$. Use LabVIEW to open and customize the Interactive Root Locus VI from the Examples in order to implement a design of K_D to yield a step response with a maximum percent overshoot of 0.2%. Use a hybrid graphical/ MathScript approach.

Postlab

1. Discuss the success of your design.
2. Is the steady-state error what you would expect? Give reasons for your answer.

Experiment 9.1

Objectives To perform a trade-off study for lead compensation. To design a PI controller and see its effect upon steady-state error.

Minimum Required Software Packages MATLAB, and the Control System Toolbox

Prelab

1. How many lead compensator designs will meet the transient response specifications of a system?

2. What differences do the lead compensators of Prelab 1 make?

3. Design a lead compensator for a unity negative feedback system with a forward transfer function of $G(s) = \dfrac{K}{s(s+3)(s+6)}$ to meet the following specifications: percent overshoot $= 20\%$; settling time $= 2$ seconds. Specify the required gain, K. Estimate the validity of the second-order approximation.

4. What is the total angular contribution of the lead compensator of Prelab 3?

5. Determine the pole and zero of two more lead compensators that will meet the requirements of Prelab 3.

6. What is the expected steady-state error for a step input for each of the lead-compensated systems?

7. What is the expected steady-state error for a ramp input for each of the lead-compensated systems?

8. Select one of the lead compensator designs and specify a PI controller that can be cascaded with the lead compensator that will produce a system with zero steady-state error for both step and ramp inputs.

Lab

1. Using the SISO Design Tool, create the design in Prelab 3 and plot the root locus, step response, and ramp response. Take data to determine the percent overshoot, settling time, and step and ramp steady-state errors. Record the gain, K.

2. Repeat Lab 1 for each of the designs in Prelab 5.

3. For the design selected in Prelab 8, use the SISO Design Tool and insert the PI controller. Plot the step response and measure the percent overshoot, settling time, and steady-state error. Also, plot the ramp response for the design and measure the steady-state error.

4. Plot the step and ramp responses for two more values of the PI controller zero.

Postlab

1. Make a table showing calculated and actual values for percent overshoot, settling time, gain, K, steady-state error for step inputs, and steady-state error for ramp inputs. Use the three systems without the PI controller and the single system with the PI controller from Lab 3.

2. Itemize the benefits of each system without the PI controller.

3. Choose a final design and discuss the reasons for your choice.

Experiment 9.2

Objective To design a PID controller via LabVIEW

Minimum Required Software Packages LabVIEW with the Control Design and Simulation Module

Prelab

1. Perform Cyber Exploration Laboratory Experiment 8.3.
2. Use the system described in Cyber Exploration Laboratory Experiment 8.3 and replace the controller described there, $G_c(s) = K_D s + K_P$, with a PID controller.
3. Design the controller to meet the following requirements: (1) shorten the settling time found in the design of Cyber Exploration Laboratory Experiment 8.3 to less than 1 sec., and (2) limit the percent overshoot to no more than 5%.
4. Design a LabVIEW VI to test your design. The front panel inputs will be the PID gains and the numerator and denominator of the plant. The indicators will be the transfer functions of the plant, PID controller, and closed-loop system. Finally, provide an indicator for the step-response graph.

Lab Run your LabVIEW VI and obtain the step response of the closed-loop system.

Postlab Compare the transient and steady-state error performance between the closed-loop step responses of Cyber Exploration Laboratory Experiment 8.3 and this experiment.

Experiment 10.1

Objective To examine the relationships between open-loop frequency response and stability, open-loop frequency response and closed-loop transient response, and the effect of additional closed-loop poles and zeros upon the ability to predict closed-loop transient response

Minimum Required Software Packages MATLAB, and the Control System Toolbox

Prelab

1. Sketch the Nyquist diagram for a unity negative feedback system with a forward transfer function of $G(s) = \dfrac{K}{s(s+2)(s+10)}$. From your Nyquist plot, determine the range of gain, K, for stability.

2. Find the phase margins required for second-order closed-loop step responses with the following percent overshoots: 5%, 10%, 20%, 30%.

Lab

1. Using the SISO Design Tool, produce the following plots simultaneously for the system of Prelab 1: root locus, Nyquist diagram, and step response. Make plots for the following values of K: 50, 100, the value for marginal stability found in Prelab 1, and a value above that found for marginal stability. Use the zoom tools when required to produce an illustrative plot. Finally, change the gain by grabbing and moving the closed-loop poles along the root locus and note the changes in the Nyquist diagram and step response.

2. Using the SISO Design Tool, produce Bode plots and closed-loop step responses for a unity negative feedback system with a forward transfer function of $G(s) = \dfrac{K}{s(s+10)^2}$. Produce these plots for each value of phase margin found in Prelab 2. Adjust the gain to arrive at the desired phase margin by grabbing the Bode magnitude curve and moving it up or down. Observe the effects, if any, upon the Bode phase plot. For each case, record the value of gain and the location of the closed-loop poles.

3. Repeat Lab 2 for $G(s) = \dfrac{K}{s(s+10)}$.

Postlab

1. Make a table showing calculated and actual values for the range of gain for stability as found in Prelab 1 and Lab 1.

2. Make a table from the data obtained in Lab 2 itemizing phase margin, percent overshoot, and the location of the closed-loop poles.

3. Make a table from the data obtained in Lab 3 itemizing phase margin, percent overshoot, and the location of the closed-loop poles.

4. For each Postlab task 1 to 3, explain any discrepancies between the actual values obtained and those expected.

Experiment 10.2

Objective   To use LabVIEW and Nichols charts to determine the closed-loop time response performance.

Minimum Required Software Packages   LabVIEW, Control Design and Simulation Module, MathScript RT Module, and MATLAB

Prelab

1. Assume a unity-feedback system with a forward-path transfer function, $G(s) = \dfrac{100}{s(s+5)}$. Use MATLAB or any method to determine gain and phase margins. In addition, find the percent overshoot, settling time, and peak time of the closed-loop step response.

2. Design a LabVIEW VI that will create a Nichols chart. Adjust the Nichols chart's scale to estimate gain and phase margins. Then, prompt the user to enter the values of gain and phase margin found from the Nichols chart. In response, your VI will produce the percent overshoot, settling time, and peak time of the closed-loop step response.

Lab   Run your VI for the system given in the Prelab. Test your VI with other systems of your choice.

Postlab   Compare the closed-loop performance calculated in the Prelab with those produced by your VI.

Experiment 11.1

Objectives   To design a PID controller using MATLAB's SISO Design Tool. To see the effect of a PI and a PD controller on the magnitude and phase responses at each step of the design of a PID controller.

Minimum Required Software Packages   MATLAB, and the Control System Toolbox

Prelab

1. What is the phase margin required for 12% overshoot?
2. What is the bandwidth required for 12% overshoot and a peak time of 2 seconds?
3. Given a unity feedback system with $G(s) = \dfrac{K}{s(s+1)(s+4)}$, what is the gain, K, required to yield the phase margin found in Prelab 1? What is the phase-margin frequency?
4. Design a PI controller to yield a phase margin $5°$ more than that found in Prelab 1.
5. Complete the design of a PID controller for the system of Prelab 3.

Lab

1. Using MATLAB's SISO Design Tool, set up the system of Prelab 3 and display the open-loop Bode plots and the closed-loop step response.
2. Drag the Bode magnitude plot in a vertical direction until the phase margin found in Prelab 1 is obtained. Record the gain K, the phase margin, the phase-margin frequency, the percent overshoot, and the peak time. Move the magnitude curve up and down and note the effect upon the phase curve, the phase margin, and the phase-margin frequency.
3. Design the PI controller by adding a pole at the origin and a zero one decade below the phase-margin frequency found in Lab 2. Readjust the gain to yield a phase margin $5°$ higher than that found in Prelab 1. Record the gain K, the phase margin, the phase-margin frequency, the percent overshoot, and the peak time. Move the zero back and forth in the vicinity of its current location and note the effect on the magnitude and phase curve. Move the magnitude curve up and down and note its effect on the phase curve, the phase margin, and the phase-margin frequency.
4. Design the PD portion of the PID controller by first adjusting the magnitude curve to yield a phase-margin frequency slightly below the bandwidth calculated in Prelab 2. Add a zero to the system and move it until you obtain the phase margin calculated in Prelab 1. Move the zero and note its effect. Move the magnitude curve and note its effect.

Postlab

1. Compare the Prelab PID design with that obtained via the SISO Design Tool. In particular, compare the gain K, the phase margin, the phase-margin frequency, the percent overshoot, and the peak time.
2. For the uncompensated system, describe the effect of changing gain on the phase curve, the phase margin, and the phase-margin frequency.

3. For the PI-compensated system, describe the effect of changing gain on the phase curve, the phase margin, and the phase-margin frequency. Repeat for changes in the zero location.

4. For the PID-compensated system, describe the effect of changing gain on the phase curve, the phase margin, and the phase-margin frequency. Repeat for changes in the PD zero location.

Experiment 12.1

Objective   To simulate a system that has been designed for transient response via a state-space controller and observer.

Minimum Required Software Packages   MATLAB, Simulink, and the Control System Toolbox

Prelab

1. This experiment is based upon your design of a controller and observer as specified in the Case Study Challenge problem in Chapter 12. Once you have completed the controller and observer design in that problem, go on to Prelab 2.

2. What is the controller gain vector for your design of the system specified in the Case Study Challenge problem in Chapter 12?

3. What is the observer gain vector for your design of the system specified in the Case Study Challenge problem in Chapter 12?

4. Draw a Simulink diagram to simulate the system. Show the system, the controller, and the observer using the physical variables specified in the Case Study Challenge problem in Chapter 12.

Lab

1. Using Simulink and your diagram from Prelab 4, produce the Simulink diagram from which you can simulate the response.

2. Produce response plots of the system and the observer for a step input.

3. Measure the percent overshoot and the settling time for both plots.

Postlab

1. Make a table showing the design specifications and the simulation results for percent overshoot and settling time.

2. Compare the design specifications with the simulation results for both the system response and the observer response. Explain any discrepancies.

3. Describe any problems you had implementing your design.

Experiment 12.2

Objective   To use LabVIEW to design a controller and observer

Minimum Required Software Packages   LabVIEW, the Control Design and Simulation Module, and the MathScript RT Module.

Prelab   Design a LabVIEW VI that will design the controller and observer for the Antenna Control Case Study in this chapter. Your VI will have the following inputs: phase-variable form of the plant, the controller poles, and the observer poles to meet

the requirements. Your indicators will display the following: the phase-variable equation of the plant, whether or not the system is controllable, the observer canonical equation of the observer, whether or not the system is observable, the gains for the controller, and the gains for the observer. Also provide the impulse response and initial response curves shown in Figure 12.27. In addition, provide similar response curves for the state variables.

Lab   Run your VI and collect the data from which to compare the results of the case study with those found from your VI.

Postlab   Compare and summarize the results found from your VI with those of the Chapter 12 Antenna Control Case Study.

Experiment 13.1

Objective To design the gain of a digital control system to meet a transient response requirement; to simulate a digital control system to test a design; to see the effect of sampling rate upon the time response of a digital system.

Minimum Required Software Packages MATLAB, Simulink, and the Control System Toolbox

Prelab

1. Given the antenna azimuth control system shown on the front endpapers, use Configuration 2 to find the discrete transfer function of the plant. Neglect the dynamics of the power amplifier and include the preamplifier, motor, gears, and load. Assume a zero-order hold and a sampling interval of 0.01 second.

2. Using the digital plant found in Prelab 1, find the preamplifier gain required for a closed-loop digital system response with 10% overshoot and a sampling interval of 0.01 second. What is the peak time?

3. Given the antenna azimuth control system shown on the front endpapers, use Configuration 2 to find the preamplifier gain required for the continuous system to yield a closed-loop step response with 10% overshoot. Consider the open-loop system to be the preamplifier, motor, gears, and load. Neglect the dynamics of the power amplifier.

Lab

1. Verify your value of preamplifier gain found in Prelab 2 using the SISO Design Tool to generate the root locus for the digital open-loop transfer function found in Prelab 1. Use the Design Constraints capability to generate the 10% overshoot curve and place your closed-loop poles at this boundary. Obtain a plot of the root locus and the design boundary. Record the value of gain for 10% overshoot. Also, obtain a plot of the closed-loop step response using the LTI Viewer and record the values of percent overshoot and peak time. Use the same tool to find the range of gain for stability.

2. Using Simulink set up the closed-loop digital system whose plant was found in Prelab 1. Make two diagrams: one with the digital transfer function for the plant and another using the continuous transfer function for the plant preceded by a zero-order sample-and-hold. Use the same step input for both diagrams and obtain the step response of each. Measure the percent overshoot and peak time.

3. Using Simulink, set up both the digital and continuous systems calculated in Prelabs 2 and Prelab 3, respectively, to yield 10% overshoot. Build the digital system with a sample-and-hold rather than the *z*-transform function. Plot the step response of each system and record the percent overshoot and the peak time.

4. For one of the digital systems built in Lab 2, vary the sampling interval and record the responses for a few values of sampling interval above 0.01 second. Record sampling interval, percent overshoot, and peak time. Also, find the value of sampling interval that makes the system unstable.

Postlab

1. Make a table containing the percent overshoot, peak time, and gain for each of the following closed-loop responses: the digital system using the SISO Design Tool; the digital system using Simulink and the digital transfer functions; the digital system using Simulink and the continuous transfer functions with the zero-order sample-and-hold; and the continuous system using Simulink.

2. Using the data from Lab 4, make a table containing sampling interval, percent overshoot, and peak time. Also, state the sampling interval that makes the system unstable.

3. Compare the responses of all of the digital systems with a sampling interval of 0.01 second and the continuous system. Explain any discrepancies.

4. Compare the responses of the digital system at different sampling intervals with the continuous system. Explain the differences.

5. Draw some conclusions about the effect of sampling.

Experiment 13.2

Objective To use the various functions from the LabVIEW Control Design and Simulation Module for the analysis of digital control systems.

Minimum Required Software Packages LabVIEW with the Control Design and Simulation Module and the MathScript RT Module; MATLAB with the Control Systems Toolbox.

Prelab You are given Figure P8.28 and the parameters listed in the Prelab of Cyber Exploration Laboratory Experiment 8.2 for the open-loop NASA eight-axis ARMII (Advanced Research Manipulator II) electromechanical shoulder joint/link, actuated by an armature-controlled dc servomotor.

1. Obtain the open-loop transfer function of the shoulder joint/link, $G(s) = \dfrac{\theta_L(s)}{V_{ref}(s)}$, or use your calculation from Cyber Exploration Laboratory Experiment 8.2.

2. Use MATLAB and design a digital compensator to yield a closed-loop response with zero steady-state error and a damping ratio of 0.7. If you already have performed Cyber Exploration Laboratory Experiment 8.2, modify your M-file from that experiment. Test your design using MATLAB.

Lab Simulate your Prelab design using a Simulation Loop from the LabVIEW Control Design and Simulation Module. Plot the step response of two loops as follows: (1) a unity feedback with the forward path consisting of the continuous system transfer function preceded by a zero-order hold, and (2) a unity feedback with the forward path consisting of the equivalent discrete transfer function of your compensator in cascade with the open-loop plant.

Postlab Compare the results obtained with those from your prelab MATLAB program. Comment on time-performance specifications.