

Cleveland State University
Department of Electrical and Computer Engineering
Control Systems Laboratory

Experiment #1
Torsion Mechanism: Familiarization and Basic Closed Loop Control

INTRODUCTION

The objectives of this experiment are to:

- (i) familiarize you with the torsion mechanism which in this lab is the system whose behavior is to be controlled and which forms the basis of all but one of the experiments;
- (ii) familiarize you with the control computer and the Real-Time Simulink software that are used to control the mechanism;
- (iii) familiarize you with the A/D card and the encoder card and their associated software that bring feedback signals from the mechanism into the control computer and Simulink;
- (iv) familiarize you with D/A board that brings control signals from the control computer and Simulink to the mechanism; and
- (v) have you perform simple closed loop control of the mechanism.

What you will do in this experiment to achieve these objectives is first make tests on the individual components or subsystems of the basic feedback control loop, and then you'll hook up the entire control loop and test out its performance. The paragraphs below describe in more detail the subsystems that make up the complete closed loop control system.

Torsion Mechanism: The torsion mechanism whose behavior you are to control is shown in Figure 1. The general control objective—illustrated in Figure 2—is to control the angular position of the mechanism's bottom plate (the mechanism in the lab has the middle and top plates removed). The plate is driven by a brushless DC Servomotor connected to the plate's shaft through a rigid-belt/pulley system with a 3:1 speed reduction, and the motor itself—which has a torque constant (gain) of 0.086 Nm/A and negligible dynamics compared to the those of the torsion mechanism—is driven by a power amplifier which also has negligible dynamics compared to the torsion mechanism, and a gain of 1.5 A/V. The plate position is measured with a high resolution incremental encoder that generates 16,000 pulses per revolution of the plate: the change in plate angular position is measured by counting pulses—up for one direction of rotation, down for the other. Although this type of encoder only gives a measurement of change in position rather than absolute position, it is useful to us because position change is what we will be controlling, i.e. we'll want to change the plate position by, say, 14.8 degrees from wherever it is, rather than move it from, say, exactly 22.9 degrees to 44.2 degrees. Finally, the angular velocity of the plate—useful in addition to the plate position measurement for controlling plate position—is measured with a tachometer mounted on the crossbar midway up the mechanism. The tachometer (tach) is a DC generator that

outputs a voltage proportional to its speed, and its shaft is coupled to the plate's shaft with a small belt/pulley system.

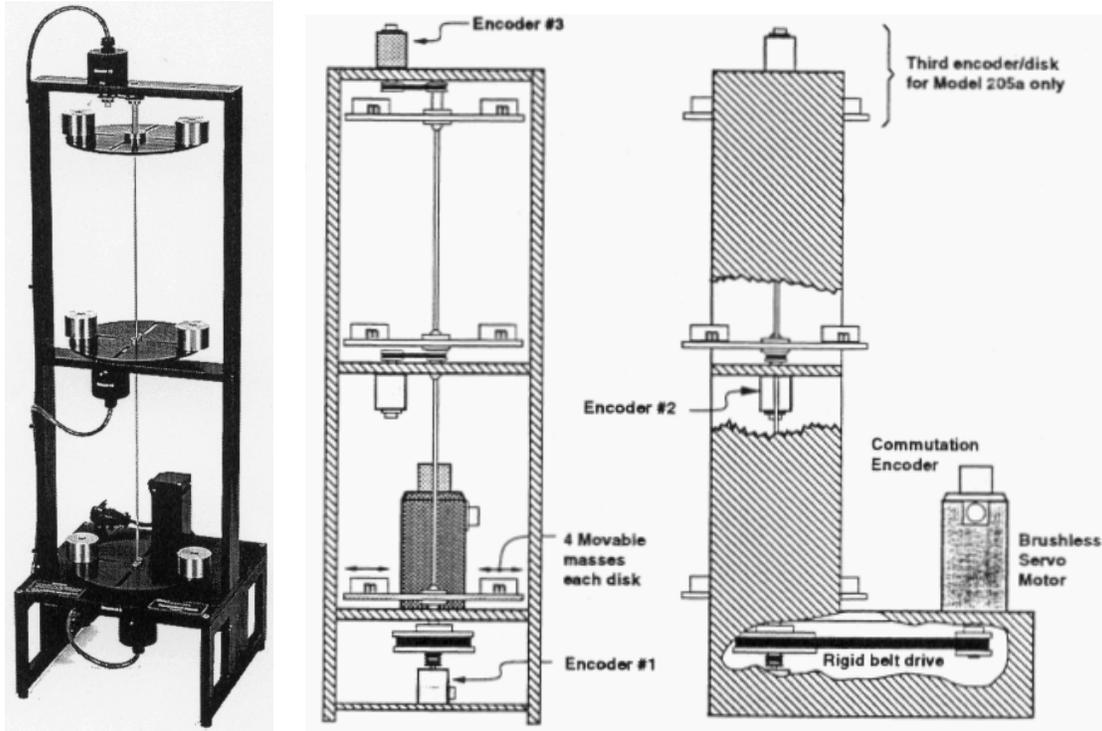
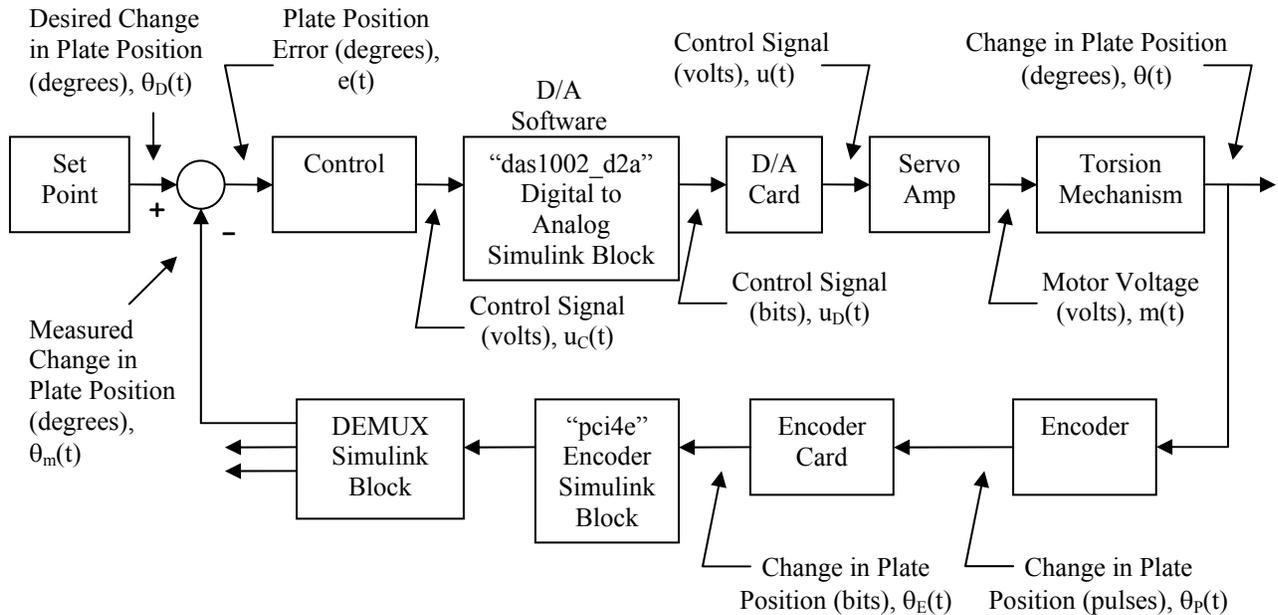


Figure 1: Torsion Mechanism

Figure 2: Complete Feedback Control System



Encoder Card: The pulses from the encoder (see Figure 2) are fed to an encoder card (mounted in the PC) that counts encoder pulses and maintains the count in a register. The contents of this register are read by the **Optical Encoder Board Interface (pci4e)** block (developed in-house) in Simulink, and the output of this block is the plate position change in degrees. This measurement is then available to the control program that is running in real-time in the computer and controlling the voltage to the plate servomotor. The torsion mechanism has three encoders—one for each plate—but we are only using encoder #1 for plate #1. The **Optical Encoder Board Interface** can be found in the “**CSU Library**” in the Simulink command window. The dialog box for the encoder block should be set up as shown in Figure 3.

A/D Converter Card: The tach voltage representing plate speed is fed to the A/D converter card (mounted in the computer) which converts the analog tach voltage to digital form (16 bits). The card’s output is read by the **Analog to Digital Converter (das1002_a2d)** block in Simulink, and this block’s output—which is a digital version of plate speed—is available for use by any other Simulink block. The **Analog to Digital Converter** block can be found in the “**CSU Library**” in the Simulink command window. The dialog box for this block should be set up as shown in Figure 3.

D/A Converter Card: The control signal generated by the control program (which is running in real-time in the computer) is a digital signal that must be converted to an analog voltage and then output to the servomotor’s power amplifier (the power amplifier—or servo amp—drives the servomotor, and the servomotor, in turn, rotates the plate). The conversion from digital to analog is done by the combination of the **Digital to Analog Converter (das1002_d2a)** block in Simulink and the D/A converter card mounted in the computer: the Simulink block takes the digital control signal and applies it to the D/A card where it is converted to an analog voltage. The analog voltage is then applied to the Servo Amp’s input. This block can be found in Simulink the same way the A/D block was found. The dialog box for the **Digital to Analog Converter** should be set up as shown in Figure 4.

Real-Time Clock: Since the control program is controlling the mechanism in real time and not internal—or computer—time, it must itself run in real time. This is accomplished by placing the **Pseudo Real-Time Clock (realtimeclock)** block in Simulink (found in the “**CSU Library**”) on any Simulink control program that you want to run in real time. The dialog box for this block should be set up as shown in Figure 4.

PROCEDURE

MAKE SURE ALL POWER IS TURNED OFF. Now hook up that portion of the control loop shown in Figure 2 that is external to the control computer, i.e. connect the D/A card output to the servo amp input, the servo amp output to the mechanism’s servomotor input, and the encoder output to the encoder’ card input.

Part I—D/A Output: The objective of this first part of the experiment is to check and calibrate the D/A output capability of the system. Remember: the D/A is used to send the control signal from the controller to the servo amp.

- a) **DISCONNECT THE D/A OUTPUT FROM THE SERVOAMP INPUT.** Then set up a Simulink program that will (i) run in real time, and (ii) use a “constant” block and the **Digital to Analog Converter** block to output a constant value from the D/A (in a control loop, this would be the control signal). Make sure the parameters for the D/A block are set properly.

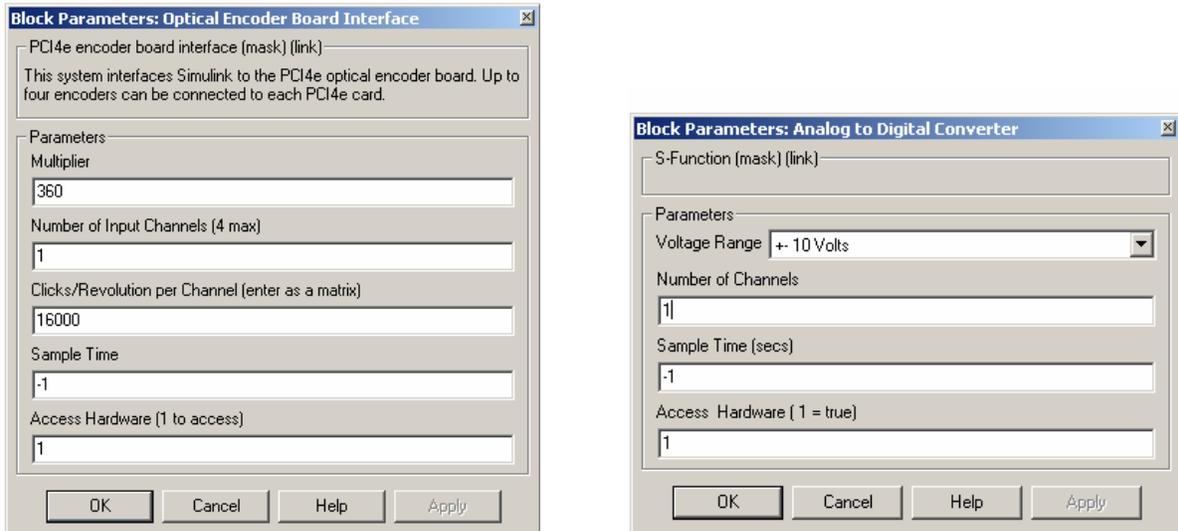


Figure 3: Dialog Boxes for Optical Encoder Board Interface (pci4e), and Analog to Digital Converter (das1002_a2d)

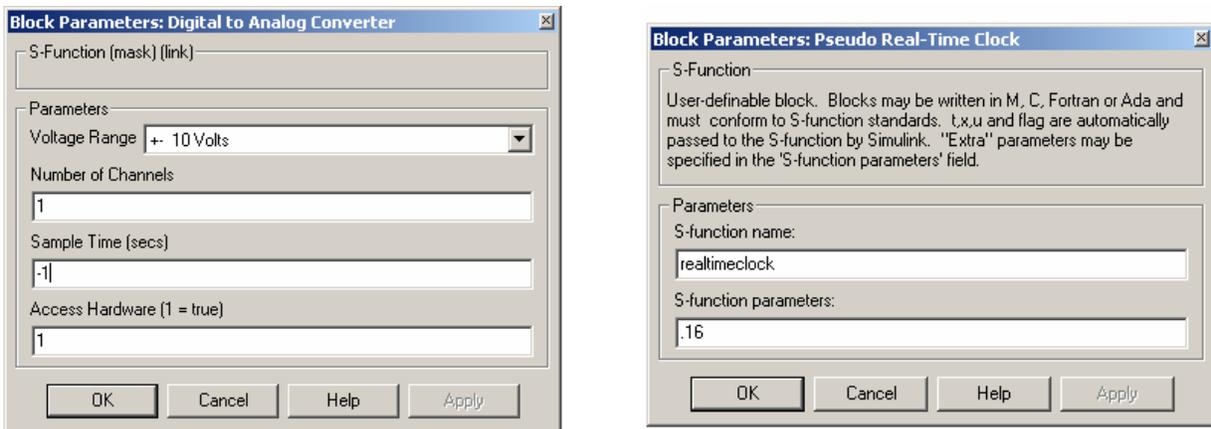


Figure 4: Dialog Boxes for Digital to Analog Converter (das1002_d2a), and Pseudo Real-Time Clock (realtimeclock)

- b) Run the program and use a multi-meter to measure the D/A output as you vary the constant being output from 0 to 5.0 in steps of 0.5. Then repeat from 0 to -5.0 in steps of -0.5. Sketch the D/A's input-output relationship and discuss it. What is the D/A's gain? **Have the instructor check your data before going further.**
- c) **MAKE SURE THE SERVOAMP IS TURNED OFF AND THE CONTROL PROGRAM IS NOT RUNNING.** Connect the D/A output to the servo amp input, and set the constant output value in the program to 0.4. **Before running the program**, turn on the servo amp—if the plate turns, turn off the servo amp and seek help from the instructor. Now run the program. The plate should rotate at a constant speed since you are applying a constant control voltage to the motor via the D/A and the servo amp. Stop the program and change the sign of the control voltage. What happens to the rotation of the plate? Doing this and observing the rotation of the mechanism sets up the next part of the experiment.

DISCUSSION QUESTIONS

1. Plot the input/output relationship for the combination of the **Digital to Analog Converter** block and the D/A card using the data from part (b).
2. Assuming there are no significant dynamics in the **Digital to Analog Converter** block and the D/A card, what is the transfer function between the controller output and the D/A output?
3. Assuming the servo amp has no significant dynamics, what is its transfer function? Hint: reread the Introduction where the torsion mechanism is described.

Part II—Position Measurement/Encoder Input: The objective of this part of the experiment is to check the incremental encoder that produces a measurement of plate position.

- a) Add a **Optical Encoder Board Interface** block to the program from Part I. The **Optical Encoder Board Interface** has a vector output [position, sample time, velocity (per second)] for each channel configured (3,6,9 or 12 outputs). Connect the block's encoder #1 output (which measures the position of plate #1) to a graph block so you can observe plate position. **DO NOT YET RUN THE CONTROL PROGRAM.**
- b) **DISCONNECT THE LEADS FROM THE D/A OUTPUT TO THE SERVOAMP** (this will prevent the control program from turning the plate). Now run the control program, rotate the plate by hand through approximately ± 45 degrees and ± 90 degrees, and observe the encoder's output on the graph (you may have to adjust the graph's parameters so that you can see everything clearly). Do you see what you think you should see? Make some notes about what you see and check with the instructor.

- c) While the control program is running, set the constant going out through the D/A to zero (this is the control signal, and setting it to zero now will ensure that the plate won't turn when you first reconnect the D/A to the servo amp). Now stop the control program, set the graph's parameters to 10 seconds and about 10,000 degrees, set the control signal to 0.25, and reconnect the D/A output leads to the servo amp. Now run the control program and observe the plate position in the real world and on the graph. Does what you see make sense?

DISCUSSION QUESTIONS

1. Assuming there are no significant dynamics associated with the encoder, the encoder card, and the **Optical Encoder Board Interface** block (see Figure 2), and assuming none of these make any errors in measuring a change in plate position, what is the transfer function of the combination of the three blocks—from encoder input to **Optical Encoder Board Interface** block output?
2. Describe why the response in part (c) looked the way it did.

Part III—Plate Speed Measurement and A/D Card: The purpose of this part of the experiment is to check and calibrate the tachometer and A/D card that together provide a measurement of plate speed for use in future experiments. The tach provides a voltage proportional to plate speed; you must determine that proportionality constant so that you can convert speed in volts to speed in rpm.

- a) **DISCONNECT THE D/A OUTPUT LEADS FROM THE SERVOAMP.** Set the control signal to 0 and run the control program (this will ensure that the plate won't turn when you connect the D/A to the servo amp). Now stop the control program and reconnect the D/A leads to the servo amp.
- b) Connect the tach output to the A/D card input, and connect a multi-meter to read the tach output voltage. Now add a **Analog to Digital Converter** block to the control program so that the tach voltage can be brought into the control program, and run the output of this block to a graph block so the value of the tach voltage in the control program can be read. Finally, set the constant in the control program that determines the servo amp voltage to a value of 0.1.
- c) You are now going to use a strobe light to determine the proportionality constant (discussed above) that converts tach voltage to plate speed in rpm. Run the control program and vary the control voltage from about 0.1 up to 1.0 (each value changes plate speed). For each value, record three things: tach voltage straight off the tach, tach voltage in Simulink, and actual speed of the plate in rpm (this latter is measured with the strobe light). Make about ten different speed measurements and use control voltage values so that the speed points are somewhat evenly distributed over the entire speed range (don't have a bunch of points at one end and few in the middle). What is the transfer function

between tach voltage straight off the tach and tach voltage in Simulink? What is the conversion constant that converts tach voltage straight off the tach to plate speed in rpm? Discuss this with the instructor.

DISCUSSION QUESTIONS

1. Plot tach voltage vs. plate speed in rpm. What is the proportionality constant that converts tach voltage to plate speed in rpm? **Describe—in words**—why you selected the value you did, and show calculations if necessary.
2. The plate angular speed measurement system studied in this part consists of four subsystems: the tachometer, the A/D Card, the **Analog to Digital Converter** block in Simulink, and a Simulink constant block that contains your proportionality constant. Draw a block diagram of this system using four blocks, each block with an input signal and an output signal. Give a word name to each block. Each **signal** needs **three names: a word name, a symbol name—like $m(t)$ —and a units name. Label each of the signals (there are five of them) with its three names. If you think you have finished this part but can see a signal without three names—then you're not finished.**
3. Assuming none of the blocks in the above block diagram have significant dynamics, what is the overall transfer function of the speed measurement system? Why?—explain.

Part IV—Closed Loop System Performance: The objective of this part of the experiment is to put all of the earlier pieces of the experiment together, close the loop around plate position, and observe the transient performance steady state accuracy of the closed loop system.

- a) **MAKE SURE THE SERVOAMP POWER IS TURNED OFF.** Although the tach and A/D card just studied are useful in later experiments, they are not used in the remainder of this experiment and therefore should be deleted from the control program. As shown in Figure 2, the feedback signal for controlling plate position change is plate position change as measured by the encoder. Hook up the plate position change control loop shown in Figure 2—including the control program shown—and use a gain block for the controller (this makes the controller a proportional controller, i.e. the control signal is proportional to the error). Use a step function block to get step changes in the set point, and set the block up so that the steps occur at time 2.0 rather than at zero (so you can more easily see them). Use graph blocks to observe both control signal and plate position change.
- b) Start with a controller gain of 0.001 and a desired change in plate position of 180 degrees. Observe the mechanism's response and also look at the response using the graph blocks. Does the plate reach the set point? Change the controller gain from 0.001 to 0.01 in steps of 0.001 and observe the response. Print a few of the more interesting responses as you vary the gain settings.

DISCUSSION QUESTIONS

1. Describe and explain the results you observed in this part of the experiment. Remember: it's not enough to just describe what happened—you must also explain why what you are describing happened.
2. Explain why the closed loop system is or is not a unity feedback system. In doing this you should consider your answers to the questions for Part II.
3. Redraw Figure 2 in as simplified a form as possible. To help you, use your answers to the questions for Parts I and II, and your answer to the above question. **REMEMBER: ALL SIGNALS HAVE THREE NAMES, AND ALL BLOCKS HAVE AT LEAST ONE NAME.**