EE 448: Sensors and Actuators

Laboratory Module #2

Modeling a First-Order System

Assigned: January 15, 2005

Due: 12:30 PM, Monday January 22, 2007 (In Class)

Objectives

The objectives of this lab are to learn how to use the TCL board with the I/O board, and to determine the parameters of a linear model of the system.

You Will Need...

To complete this module, you will need

1. A USB-1208FS I/O card, available in Sieg 428
2. A USB cable
3. A Windows PC
4. Software for communicating with the I/O board: *InstaCal*, *LabView*.
5. An oscilloscope and probe
6. A *Temperature Control Lab* (TCL) board and power supply
7. Wire and a small screwdriver (only needed if the hardware is not already cnnected)

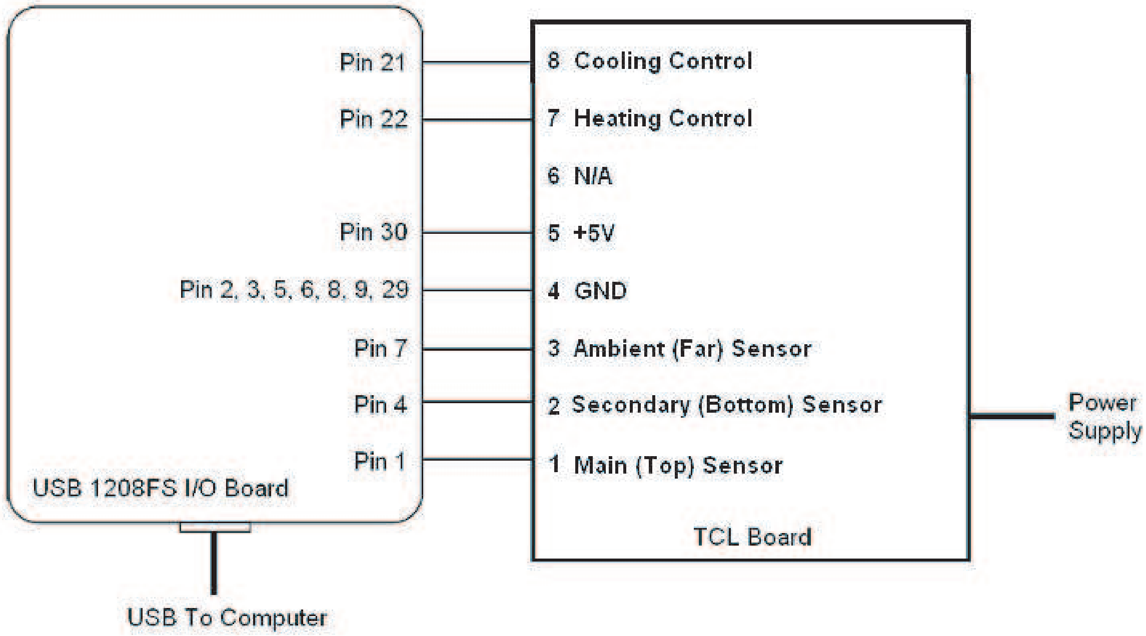

Figure 1: The connections between the I/O card and the TCL board for this module.

1 Set Up The Hardware

The hardware should already be connected in Sieg 428. If not, connect the I/O card to the computer and make sure that InstaCal recognizes it. Then start MATLAB. Connect the TCL Board as shown in Figure 1. Connect the power supply to the TCL Board last. Make sure that the green LED on the TCL board is on.

2 Check That Things Are Working

Configure LabView to output a 50% duty factor PWM signal to the digital output connected to the cooling channel. This is pin 21 on the I/O card. Use the template `.vi` given on the wiki to see how to configure and use the digital I/O lines. Note that pin 22 should be low (zero). Check that the top block is cold. If it is, everything is fine. If not, you have some debugging to do.

Repeat the above with pin 21 set to 0 and pin 22 with a 50% PWM signal. Check the the block gets warm.

> **For Your Writeup:** Describe any surprises.

Next modify your code to read the values from the sensors each step through the loop and display these on a stripchart. Don't forget to multiply by 100 to get the actual temperature in degrees Celsius. Or use Fahrenheit – just be sure to tell NASA what units you are using before they install your code on their next space probe.

Continue to output a 50% duty factor PWM signal and save the data you take to an array. Output it graphically so you can see it.

Important: Make sure you figure out how to turn off the peltiers at the end of an experiment (send zero duty factor). Otherwise they will stay on for days.

3 Obtain the K, τ_b and τ_d

Recall that the transfer function of the TCL board is

$$G(s) = \frac{K}{\tau_b s + 1} e^{-s\tau_d}.$$

The input u to the transfer function is the duty factor and the output y is the difference between the top sensor and the bottom sensor. The objective of this lab is to measure K, τ_b and τ_d.

Note that we will configure that $-100\% \leq u \leq 100\%$. If u positive, then use the duty factor u for pin 22 (heating) and keep pin 21 off. If u is negative, then use the duty factory $-u$ for pin 21 (cooling) and keep pin 22 off.

To measure K, you will need to determine the steady state value of y for several different values of u. You might consider $u \in \{-80\%, -40\%, 0\%, 40\%, 80\%\}$. To do this, send out a constant value for u until the system comes to steady state. Then write down or store the value of y for that u and go on the the next u. Fit your data with a line (don't forget error bars). The slope of the line is K.

To measure τ_b and τ_d, refer to the lecture notes. You will need to set u to a step input (of magnitude about $\pm 80\%$) and save the response to an array. You might also save the value of u and the time to additional arrays so you can plot both signals on one figure. This is important for measuring the delay. Do this process for heating and cooling.

Note that you need to make sure that your board comes to equilibrium $y = 0$ for $u = 0$ before you change to, for example, $u = 80\%$. Otherwise you are not getting a real step response. You might simply unplug the board and wait for it to cool down.

For Your Writeup: Describe the system and the equations you use to model it. Set up a mathematical framework for the rest of this module by describing what the variables are, what their units are and what equations approximately govern your system.

For Your Writeup: Explain why the procedure for measuring τ_b works.

For Your Writeup: Plot the data you obtain for measuring K and for measuring τ_b and τ_d. Explain how you obtained the data, include code snippets where relevant, and discuss the difference between the parameters measured for the two different sensors.

4 MATLAB

Note that MATLAB (or Mathematica, or Excel) can do a lot for you in terms of managing data. You should export your data from LabView into a file using the appropriate blocks. You can plot the data you are getting in real time, for example, by plotting a new point and redrawing the figure window in each step of your algorithm. Or you can save all the data ($u(t)$, $y(t)$ and anything else) into some data structure and plot it after your code finishes running. In any case, you will need to develop a suite of tools for such things and will used them all quarter.