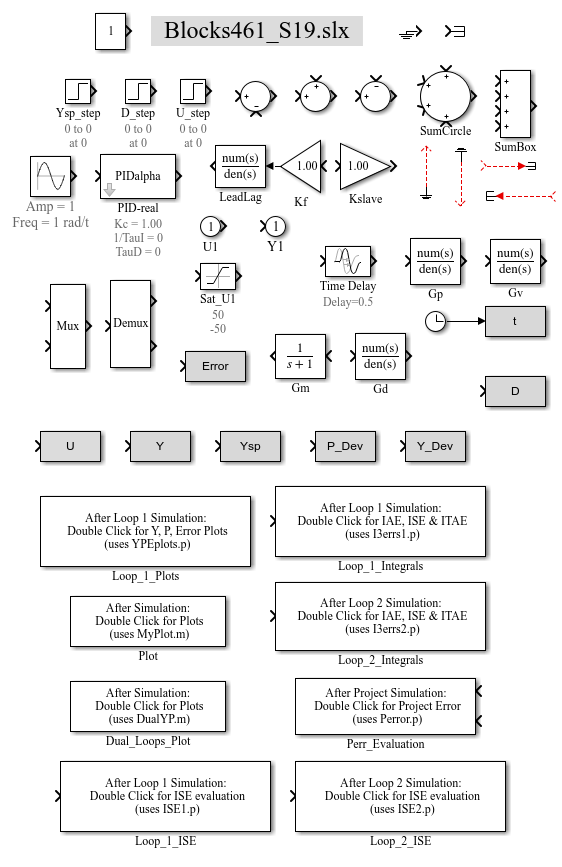
**Studio 2: Oven Block P-arty! CHE 461**

In this studio you are going to model a conventional kitchen oven as a ***simple*** first order process and carry out “open-loop” experiments (i.e. no control), and “closed-loop” experiments (i.e. controller connected) with a simple proportional-only controller (P-only). Before running any experiments you will build a simple block diagram in Simulink to represent the oven control system.

**CHE 461 Blocks = Toolbox for Simulink Work**

A library of common function blocks is contained in the file: “**Blocks461\_S20.slx.**” All of the block diagrams that we’ll use in CHE 461 can be built by copying blocks from this library (ctl+c, clt+v). There is no need to search for these blocks in Simulink (time-consuming). A number of these blocks have been custom-made particularly for the analysis that we’ll do in CHE 461 and require you to also download a script file (MATLAB file) to your working directory (current folder). These files will be made available on Canvas.

For studio 2 you will need to download the following files and place them in your working directory:

**YPEplots.p**

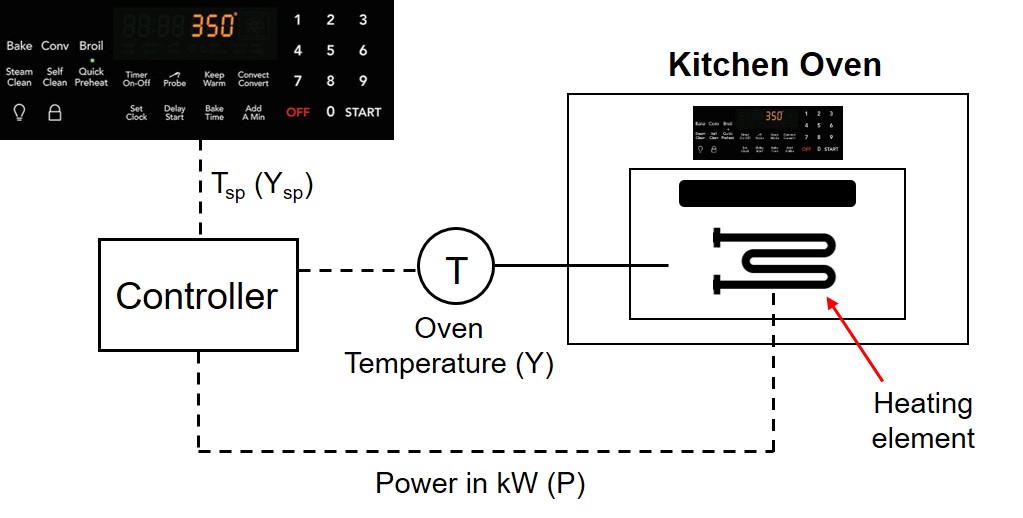
**fnicfig.m**

**Studio2\_StepTest.m**

**Blank\_Model\_461.slx**

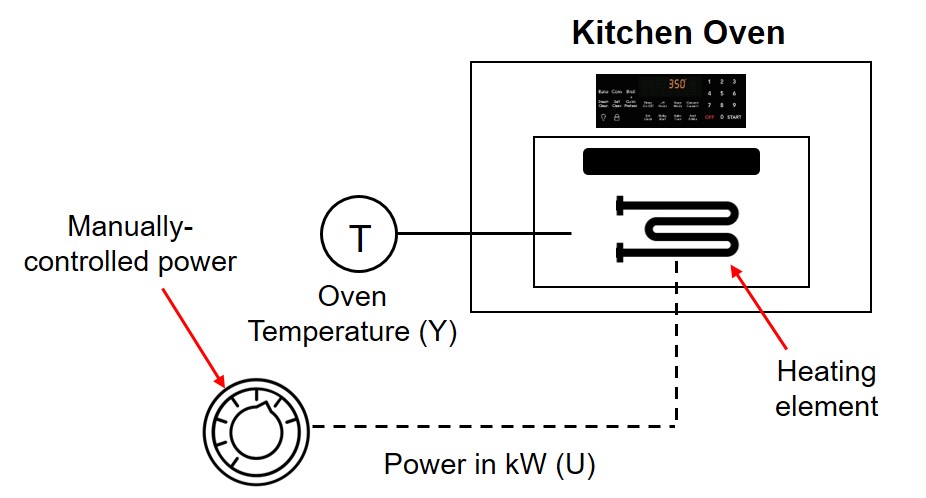
To begin studio work today open the Blank\_Model\_461.slx and Blocks461\_S20.slx files and watch the brief video on how to set up the first simulation. Alternatively, you can jump to the instructions below if you are comfortable with the MATLAB/Simulink interface

**A *Simplified* Conventional Kitchen Oven Temperature Control System:**



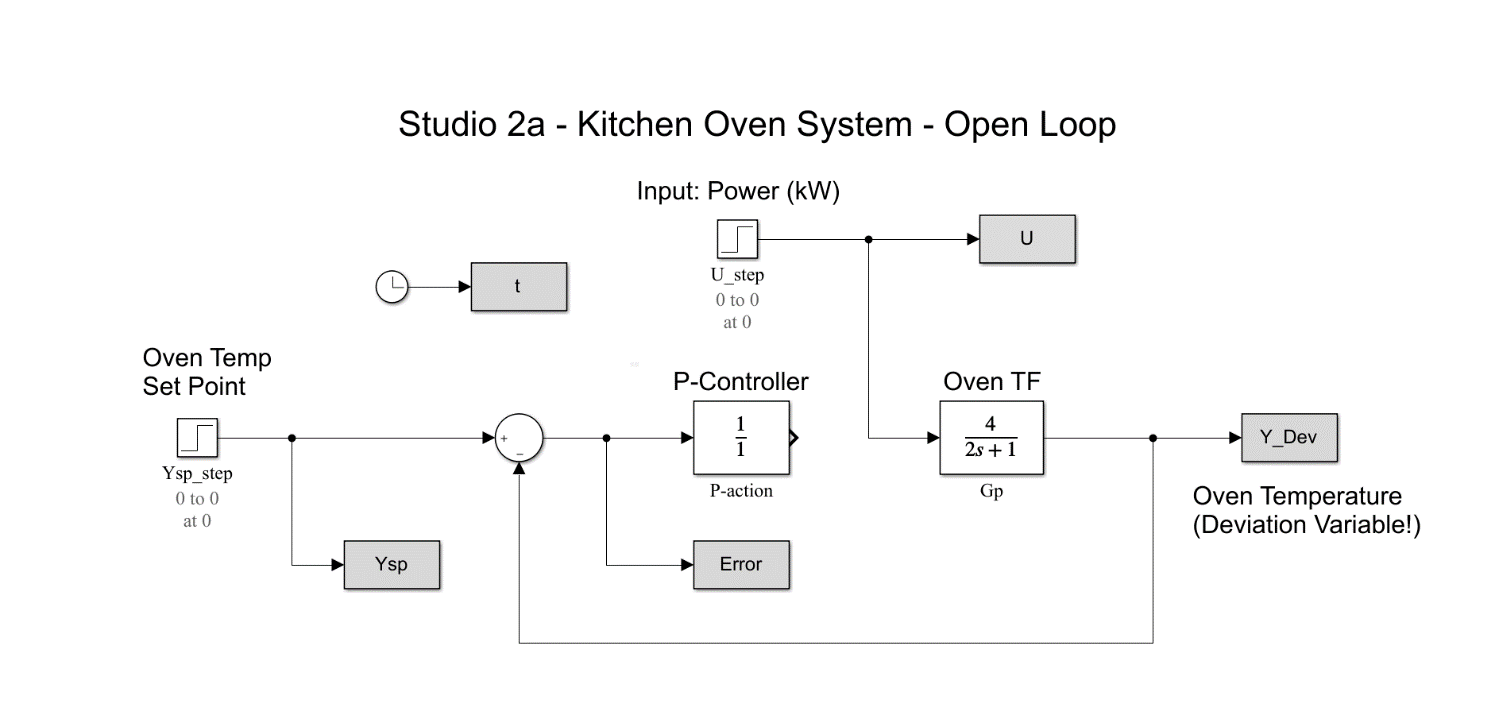
**a. Build a block diagram to model our oven system as an “open loop” (i.e. no control)**

Before running experiments on our closed-loop oven control system, we’ll first do a set of “open-loop” tests to learn how to perform step tests in Simulink and look at the dynamics of the oven. To run an “open-loop” test we simply “disconnect” the controller and manually adjust the power to the oven. The picture below presents a simplified example of how we would run our oven in “manual” open-loop mode (**don’t try this at home!**):

In the “open-loop” mode shown to the left, the controller is disconnected and the operator (you) has the ability to manually adjust power to the oven. You can do a step test by quickly turning up the power dial and observing the response of oven temperature

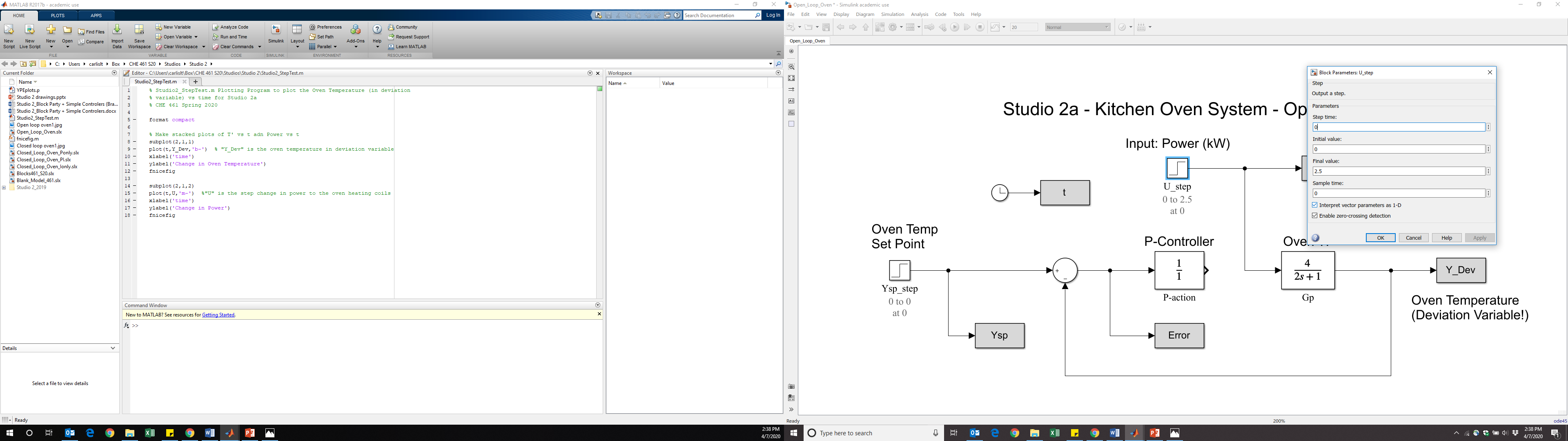
**Open the file Blank\_Model\_461.slx and build the open-loop block diagram shown below by copy/pasting blocks from Blocks461\_S20.slx and connecting wires as shown**

**\*\*\*You should first save this file as “Studio2a.slx”\*\*\***

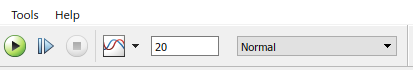


**1.** Before running a simulation, what is the input/output pair for the oven transfer function (Gp = *Y’/U’*)?

Input (*U’*):\_\_\_\_\_\_\_\_\_\_\_\_ Output (*Y’*):\_\_\_\_\_\_\_\_\_\_\_\_\_

**2.** Double click on the “U\_step” block and configure a step of +2.5 (in power, U) to occur at time = 0 (step time).

Click OK, then run the simulation by clicking the green play button. Make sure the run time is set to t=20



After running the simulation, go to the MATLAB editor window and run the plotting file Studio2\_StepTest.m. Your plots should look just like those shown below:

****

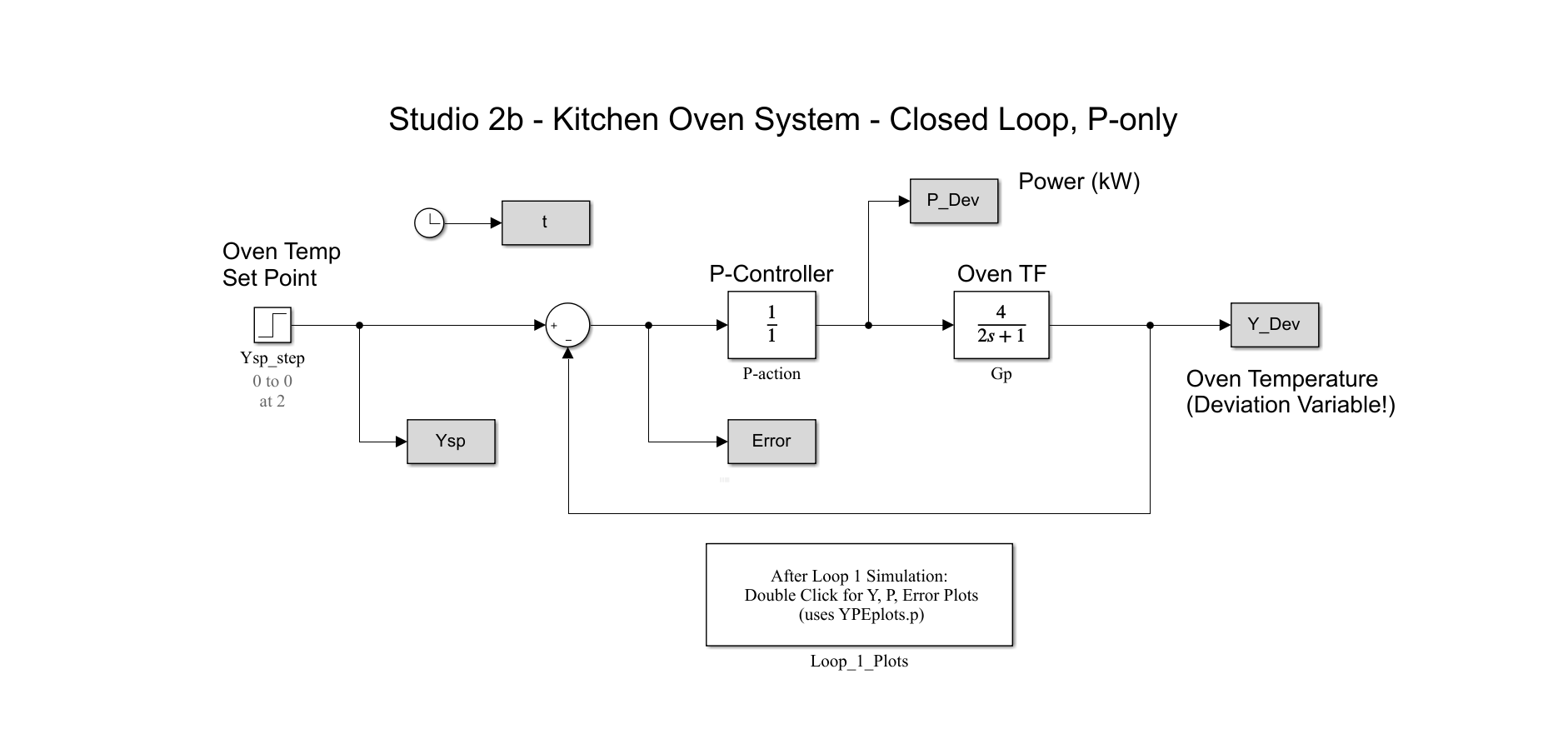
**3.** Run another step experiment, but this time configure the step to occur time = 2. Again, after the simulation is complete run the program Studio2\_StepTest.m and observe how the two plots changed from those shown to the left.

**4.** What is the change in steady state oven temperature for step change of +2.5 kW? (you can simply look at the plots)

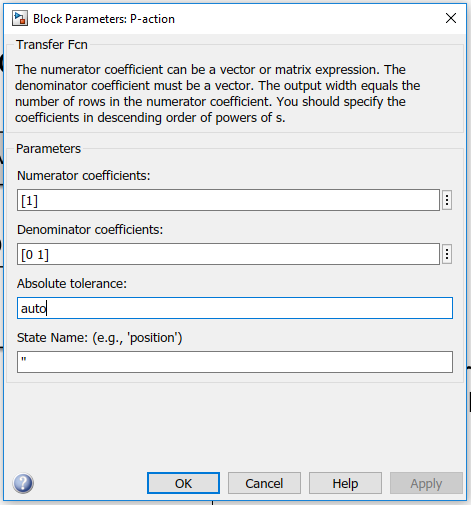
***T’* = \_\_\_\_\_\_\_\_\_\_\_\_ for a step of +2.5 kW**

**b. Modify your oven system to run experiments with a “closed-loop” P-only control scheme (i.e. “reconnect” the controller), as shown below:**

**\*\*\*Before making changes to Studio2a.slx, save this file as “Studio2b.slx”\*\*\***



Don’t forget this block!

**1.** Set the P controller gain to an initial value of *Kc* = 1 by double clicking on the function block and configuring the settings as shown here:

Now run a closed-loop servo (setpoint) experiment by double-clicking on the “Ysp\_step” block and configuring a step setpoint change of +10 degrees at time = 2. This would be equivalent to adjusting the temperature on your oven from 350 to 360 °F (using dial or digital display). After running the simulation double click the “YPEplots.p” block to generate three plots: Controlled Variable (Y, temperature), Controller Output (P, power), and Error (E).

**Inspect the three plots and answer the following questions:**

**a.** Does the oven temperature reach the desired setpoint? If not, what is the steady state error (or offset)? (You can use the “hand” tool to move plots around if data is off the scale. Could also go to MATLAB command window and type the command Y\_dev(end) to retrieve the final value of oven temperature)

Steady state error = \_\_\_\_\_\_\_\_\_\_ degrees

**b.** What is the gain of the closed-loop servo transfer function (*GCL*)? You could derive this using block algebra, **OR** simply determine from your servo experiment data/plots (*GCL* *=* *Y\_dev/Ysp = T’/T’sp*)

Closed-loop servo gain = \_\_\_\_\_\_\_\_\_

**c.** Thinking like a process engineer, what steady state offset and closed-loop servo gain would you desire in your control system?

Desired steady state error = \_\_\_\_\_\_\_\_ degrees

Desired closed-loop servo gain = \_\_\_\_\_\_\_\_

**d.** When is “P-action” largest? Smallest? Why? (Since this is a P-only controller, the controller output “P\_dev” is due only to P-action)

P-action is largest…\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

P-action is smallest…\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**e.** From your open-loop experiments in part (a), what value of P\_dev (controller output, kW) is required to obtain a 10 degree steady state temperature change? What is the actual value of P\_dev at steady state?

Required value of P\_dev = \_\_\_\_\_\_\_\_ kW

Actual value of P\_dev (controller output) = \_\_\_\_\_\_\_\_ kW

**f.** Can you ***eliminate*** steady state offset (steady state error) by making the gain *Kc* larger? Try it! (Use the plotting zoom tool to check!)

Can s.s. offset be **eliminated** with a P-only controller? **YES** or **NO**

**g.** Are there physical, or practical, device limitations that set an upper limit to how large Kc can be? **Hint:** Try running a simulation with Kc = 100 and look at the curve for controller output (P, kW)

List some physical/practical limitations for Kc:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_