

## Experiment C5

### Coupled Water Tanks

#### Procedure

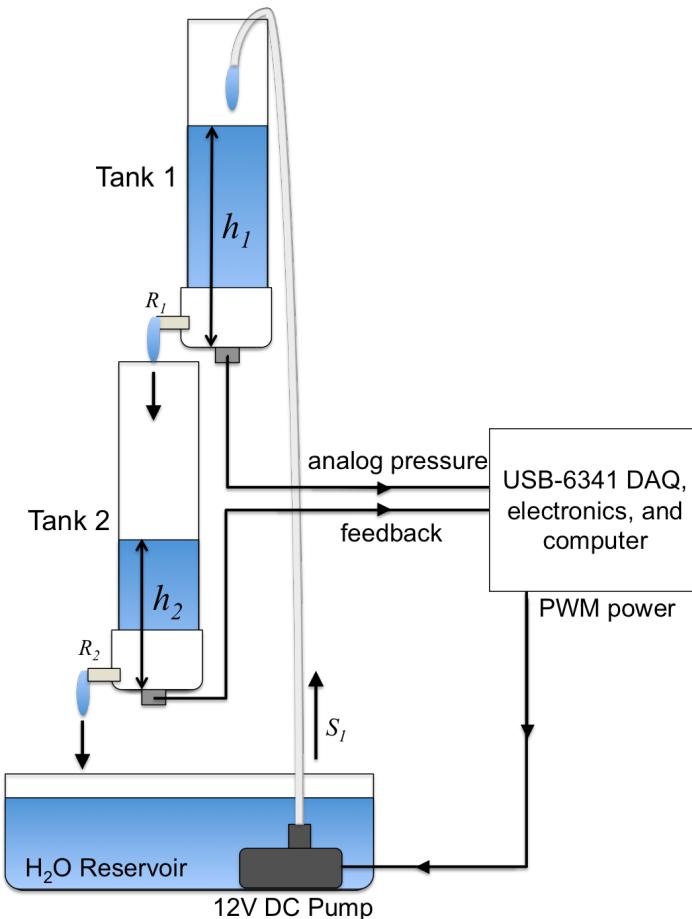
Deliverables: Checked lab notebook, Technical Memo

### Overview

Thus far, we have only had one system parameter to control (i.e. temperature or water height) and one input parameter to control it (i.e. heater power or pump power). These systems were fairly easy and intuitive to control. In this lab, we will have multiple system parameters that interact with each other (the height in both tanks) and multiple input parameters to control them (two pumps in Part II). **Controlling multiple system parameters with multiple inputs is not at all intuitive.** Thus, we will rely on the formal framework of the LQR method to guide us.

In Part I, we will have a single input (a pump) and two outputs (two water heights), as shown in Figure 1. In Part II, we will have two inputs (two pumps) and two outputs (two water heights), as shown in Figure 2.

### Part I: Single Input, Multiple Output (SIMO)



**Figure 1** – Two identical flow-through tanks are connected in series, such that water from Tank 1 drains into Tank 2. Feedback from pressure sensors in both tanks is used to control the water height.

## Background

Similar to last week, the water height is measured using an analog pressure sensor at the bottom of the tank. The sensor outputs a voltage proportional to the water pressure  $p = \rho gh$ , and you will need to calibrate it. Then, you will write a LabView program to record the water height in both tanks as a function of time.

### Hardware Setup

### Experimental Procedure

1. Hang up your coat and put your backpack off to the side.
2. Visually inspect the drain nozzle on both tanks. Confirm that they are the same.
3. Close the valve at the bottom of both tanks. Check that both tanks are firmly mounted by two beaker clamps above the dishpan (reservoir). The lab stand holding the bottom tank and reservoir should be on the floor. The top tank should be on the lab bench with outlet nozzle draining into the bottom tank, as shown in Fig. 1.
4. Make sure the breadboard is turned OFF. (You will not need the bread board power supplies. It should be off for the entire lab.)
5. Connect the +5V output and a GND on the USB-6341 to a pair of *vertical* bus lines on the breadboard. Cut some new lengths of 22 gauge wire to do this.
6. Wire up the analog pressure transducers using the following connections:
  - a. Red  $\rightarrow$  +5V
  - b. Black  $\rightarrow$  GND
  - c. White  $\rightarrow V_{OUT}$  (output voltage, see next steps)
7. Pour a few swigs of water into the tanks to make sure the sensor are working properly—the output voltage should increase as you add water.
8. Create a copy of the LabView program from last week's lab. Rename the copy with an intelligent file name (i.e. "C5\_2\_TANKS\_yourName.vi"), and save it to your C5 folder.
9. Connect the pressure sensor to the USB-6341. Double click the express VI in your LabView code. In the DAQ Assistant window, add a new analog voltage measurement. Refer to the connection diagram to determine where to connect the white  $V_{OUT}$  wire.
10. Press OK to return to the block diagram. Wait for LabView to build the Express VI.
11. Right click out "data" pin on the Express VI you just modified and select "Signal Manipulation Palette" > "Split Signals".
12. Convert the voltages to water height in inches using the linear calibration curve from last week. You may need to tweak the intercepts of the calibration equations slightly.
13. Create a new array with shift registers to save the data from Tank 2.
14. Make a copy of the Express XY Graph. You should have two XY graphs: one with the height in Tank 1, and another with the height in Tank 2, both as a function of time.
15. Modify the code to save the data from both tanks.

16. Test the program to make sure it works correctly.

**Important:** Be careful to NOT connect 12V to the pressure sensor. It will burn out the sensor.

17. Construct the PWM MOSFET circuit with the 12V DC pump, as you did last week. Use the external 12V DC power to drive the circuit, NOT the breadboard power supply.

18. Fill the bottom reservoir with water from the sink. Submerge the pump, and run the outlet tube from the top of the pump up and into the tank, as shown in Fig. 1. Make sure the tank is positioned so it will drain into the bottom tank and **both valves are open**.

19. Test the PWM MOSFET pump driver circuit using either the function generator. Confirm that it behaves similar to last week. Make a note if the threshold duty cycle has changed.

**CAUTION:** Keep your finger on the “Output ON” button on the function generator, so you can quickly disable the PWM and turn off the pump.

20. Last week, we assumed the flow rate is linearly related to duty cycle,  $S = a(\% \text{PWM}) + b$ . Note that  $-b/a$  is the threshold for flow to begin, and you may have to slightly adjust  $b$  depending on whether your threshold is different this week.

21. Assume the time constant  $\tau$  is the same for both tanks and is equal to the value you measured last week. Look up the value in your notebook and re-write in this week’s notebook entry.

22. Disconnect the function generator.

### *LQR Control*

We will now use LQR to control the height in *both tanks*. First, we will calculate the necessary flow rate  $S_s$  for a desired equilibrium height  $h_s$ , which was derived in the pre-lab assignment. Then, we will add proportional feedback from *both* sensors, such that the flow rate is determined by

$$S = S_s + k_{p1}(h_{1s} - h_1) + k_{p2}(h_{2s} - h_2), \quad (1)$$

where  $h_1$  and  $h_2$  are the water heights,  $h_{1s}$  and  $h_{2s}$  are the setpoints, and  $k_{p1}$  and  $k_{p2}$  are the proportional gains for tanks 1 and 2, respectively. (In the pre-lab assignment, you will see that there are constraints on what values of the set-points are physically stable.)

Again, the LQR method can be used to determine the optimal gains  $k_{p1}$  and  $k_{p2}$  for a system governed by a first order linear differential equation

$$\dot{x} = Ax + Bu, \quad (2)$$

where  $x$  is a column vector containing the parameter you want to control,  $u$  is the input you provide, and  $A$  is a  $2 \times 2$  matrix, and  $B$  is a  $2 \times 1$  column vector. Both  $A$  and  $B$  contain constants determined from system identification and characterization. (You will derive formulas for  $A$  and  $B$  in the pre-lab assignment.)

Note that we do not directly set the flow rate, rather we set the duty cycle of a PWM signal. Recall that the flow rate  $S$  is linearly related to % duty cycle

$$S = a \cdot (\% \text{PWM}) + b, \quad (3)$$

where  $a$  and  $b$  are the constants you determined when you calibrated the pump last week. Combining Eqs. (1) and (3) yields

$$\%PWM = \frac{S_s - b + k_{p1}(h_{1S} - h_1) + k_{p2}(h_{2S} - h_2)}{a}, \quad (4)$$

which you will use to calculate the % duty cycle from the measured height  $h$ . (Remember that  $S_s$  depends on  $h_s$ , as it did in Part III.)

## Procedure

1. Decide which programming environment you will use to implement the open-loop controller. You may use either the Arduino or LabView to implement the controller, but you must use LabView to record the data.
2. Make the necessary electrical connections. Include a **kill switch** so you can quickly cut power to the pump and avoid a catastrophic spill.
3. Implement the LQR controller using the proportional feedback given by Eq. (4).
  - a. You will need to calculate the flow rate  $S_s$  needed to attain the set-points that you derived in the pre-lab assignment.
  - b. You will then implement the proportional feedback using data from *both sensors*.
  - c. Use the values of  $a$  and  $b$  you determined from the pump flow rate characterization.
4. Use the `lqr()` function in Matlab to determine the optimal proportional gains and  $k_{p1}$  and  $k_{p2}$ .
5. Test the controller using the optimal gain with several different set points, but make sure that  $h_{1S} = h_{2S}$ . Use LabView to save the height vs. time data.

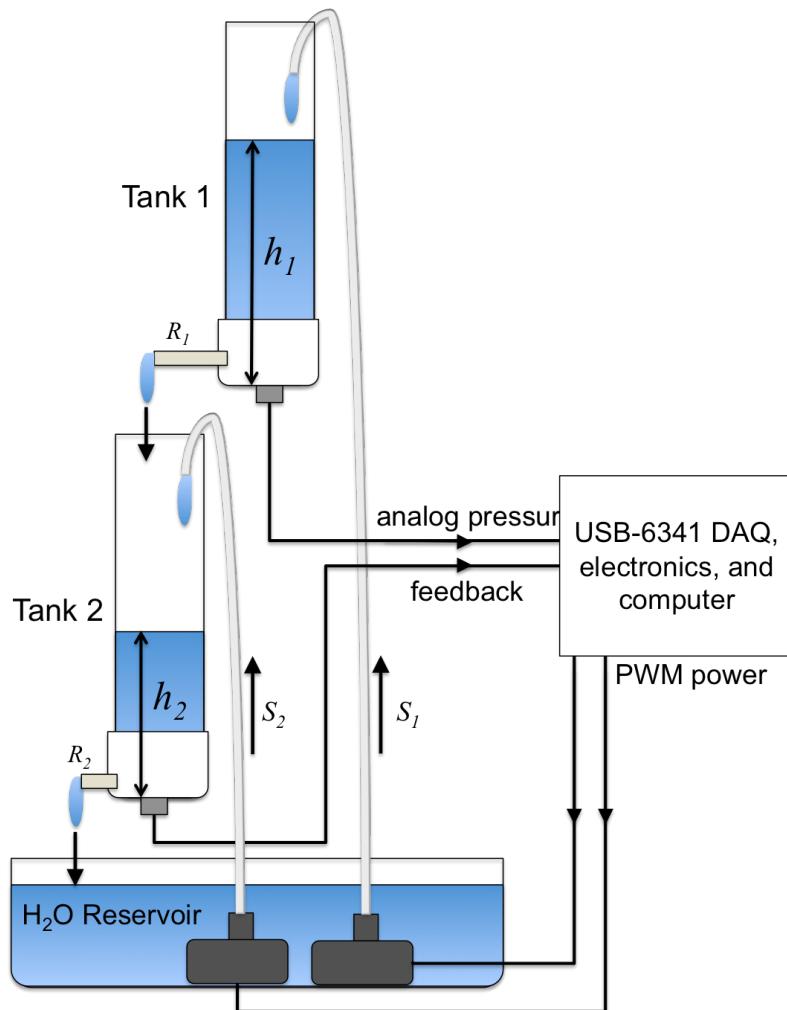
**CAUTION:** Keep your finger on the kill switch to avoid spills!

6. You will now attempt to defy the laws of physics. Test the controller using significantly unequal set-points  $h_{1S} \neq h_{2S}$ . Use LabView to save the data. Keep your finger on the kill switch in case things get messy!

## Part II: Multiple Input, Multiple Output (MIMO)

### Background

We will now add a slight modification to the previous system: an extra pump will give us an extra degree of freedom to control the level in Tank 2. By judiciously selecting different outlet nozzles, Tank 2 will only be weakly coupled to Tank 1, which will allow for a wider range of set-points where  $h_{1S} \neq h_{2S}$ .



**Figure 2** – Two identical flow-through tanks are connected in series, such that water from Tank 1 drains into Tank 2. Feedback from pressure sensors in both tanks is used to control the water height.

### System Identification and Characterization

You will now modify the system and characterize the new pump output  $S_2$ , the new nozzle resistance  $R_1$ , and new time constant  $\tau_1$ .

1. Make sure both tanks are drained. Unplug the external 12V DC power supply and disconnect it from the breadboard bus lines.
2. Replace the nozzle on Tank 1 with a different nozzle. Ask the instructor for wrenches to do this.

3. Use calipers to measure the inner diameter of both nozzles. Record the values in your lab notebook.
4. Obtain the second pump. They have a gray plastic fitting on their outlet and are in a box on the cart.
5. Construct a second PWM MOSFET circuit with the second pump.
6. Run a tube from the new pump into Tank 2. Note that the second pump requires a smaller tube diameter.
7. Use the function generator to produce a PWM input to drive the gate for the new pump. Use a “Pulse” function with a frequency of 500 Hz, a high value of 5V and low value of 0V.
8. Make sure the bottom valve is open.
9. Reconnect the 12V DC power supply to the bus lines. Use the **kill switch** to connect the positive line, so you can quickly shut down the entire system if necessary.
10. Start at a duty cycle of 1%. Turn the output of the function generator ON. Gradually increase the duty cycle until water begins to flow.

**CAUTION:** Keep your finger on the “Output ON” button on the function generator, so you can quickly disable the PWM and turn off the pump.

11. Note that there is a threshold % duty cycle for the water to reach the top of the tube and begin flowing into the tank. Record this threshold duty cycle in your lab notebook.
12. Turn off the pump by turning off the output from the function generator.
13. Close the valve on the **bottom** tank. Use the stopwatch to determine the flow rate in ( $\text{in}^3/\text{s}$ ) for a small value of % duty cycle.
14. Drain the tank and measure the flow rate for several a couple more duty cycles up to about 40%
15. Assume the flow rate is linearly related to the duty cycle, such that  $S_2 = a_2(\% \text{PWM}) + b_2$ . Apply a linear curve fit, and write down the parameters  $a_2$  and  $b_2$  in your lab notebook. Note that  $-b_2/a_2$  is the threshold for flow to begin. (The pump going to Tank 2 has not been modified, so we will assume its parameter  $a_1$  and  $b_1$  are the same as Part I.)
16. Determine time constant  $\tau_1$  for Tank 1 using a method similar to last week. (Tank 2 has not been changed, so we will assume it still has the roughly same resistance and time constant as last week.)
17. Disconnect the function generator and turn it off.

### LQR Control

As before, LQR will be used to control the height in *both tanks*. First, we will calculate the necessary flow rates  $S_{1s}$  and  $S_{2s}$  for a desired heights  $h_{1s}$  and  $h_{2s}$  using the formulas derived in the pre-lab assignment. Since we have two pumps, the input  $u$  will become a 2x1 column vector

$$u = \begin{bmatrix} S_1 - S_{1s} \\ S_2 - S_{2s} \end{bmatrix}, \quad (5)$$

and  $\mathbf{B}$  will become as 2x2 matrix. The LQR method says that the optimal feedback  $u = -\mathbf{k}x$ , where  $\mathbf{k}$  is a 2x2 matrix containing the proportional gains. Thus, the feedback becomes

$$\begin{bmatrix} S_1 \\ S_2 \end{bmatrix} = \begin{bmatrix} S_{1s} \\ S_{2s} \end{bmatrix} + \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} \begin{bmatrix} h_{1s} - h_1 \\ h_{2s} - h_2 \end{bmatrix}, \quad (6)$$

where  $S_{1s}$  and  $S_{2s}$  are the flow rate necessary for a desired equilibrium heights  $h_{1s}$  and  $h_{2s}$ . Again, assume the flow rate is linearly related to the duty cycle, such that  $S = a(\% \text{PWM}) + b$  for each pump, so you will need rewrite Eq. (6) to determine the correct % duty cycle via the pump calibration constants from Part II(a).

### Procedure

1. Create a copy of your previous Arduino script, and give it an intelligent file name. Modify the Arduino code to have an additional PWM output for the new pump.
2. Make the necessary electrical connection from the new PWM output to the new PWM MOSFET circuit.
3. Implement the LQR controller using the proportional feedback given by Eq. (6).
  - a. You will need to calculate the equilibrium flow rate  $S_s$  for the set-points that you derived in the pre-lab assignment.
  - b. You will then implement the proportional feedback using data from *both sensors*.
  - c. Use the values of  $a_i$  and  $b_i$  you determined from both pump flow rate characterizations to convert flow rate to %PWM. (See problem 2 of the pre-lab assignment.)
4. Use the `lqr()` function in Matlab to determine the optimal proportional gain matrix.
5. Test the controller using the optimal gain with several different set points where  $h_{1s} > h_{2s}$ . Use LabView to save the height vs. time data.
6. Test the controller using the optimal gain with several different set points where  $h_{1s} < h_{2s}$ . Use LabView to save the height vs. time data.

**CAUTION:** Keep your finger on the kill switch to avoid spills!

## Data Analysis and Deliverables

Using LaTeX or MS Word, make the following items and give them concise, intelligent captions. Make sure the axes are clearly labeled with units. Plots with multiple data sets on them should have a legend. **Additionally, write several paragraphs describing the plots/tables. Any relevant equations should go in these paragraphs.**

**IMPORTANT NOTE:** Add a horizontal line denoting the set-point whenever it is applicable.

1. From Part I, a table containing the system parameters: tank area  $A_T$  [in<sup>2</sup>], time constant  $\tau$  [s], and calculated LQR coefficients  $k_{p1}$  and  $k_{p2}$  [in<sup>2</sup>/s].
2. From Part I, a plot of the water heights vs. time (both on same graph w/ legend) for the case where  $h_{1S} = h_{2S}$ .
3. From Part I, a plot of the water heights vs. time (both on same graph w/ legend) for the case where  $h_{1S} \neq h_{2S}$ .
4. From Part II, a table containing the system parameters: tank area  $A_T$  [in<sup>2</sup>], time constants  $\tau_1$  and  $\tau_2$  [s], and calculated LQR coefficients  $k_{11}$ ,  $k_{12}$ ,  $k_{22}$ , and  $k_{21}$  [in<sup>2</sup>/s].
5. From Part II, a plot of the water height vs. time (both on same graph w/ legend) for the case where  $h_{1S} > h_{2S}$ .

**Talking Points** – Discuss these in your paragraphs.

- Include any relevant equations you derived in the pre-lab assignment.
- What were the most challenging aspects of this laboratory exercise?

## **Appendix A**

### **Equipment**

- USB-6341 DAQ
- Arduino UNO
- PC computer with LabView 2017
- Bread board
- Two pump
- Two tanks with pressure transducers
- Dishpan
- Spare nozzle