

## Experiment C4 Single Water Tank Procedure

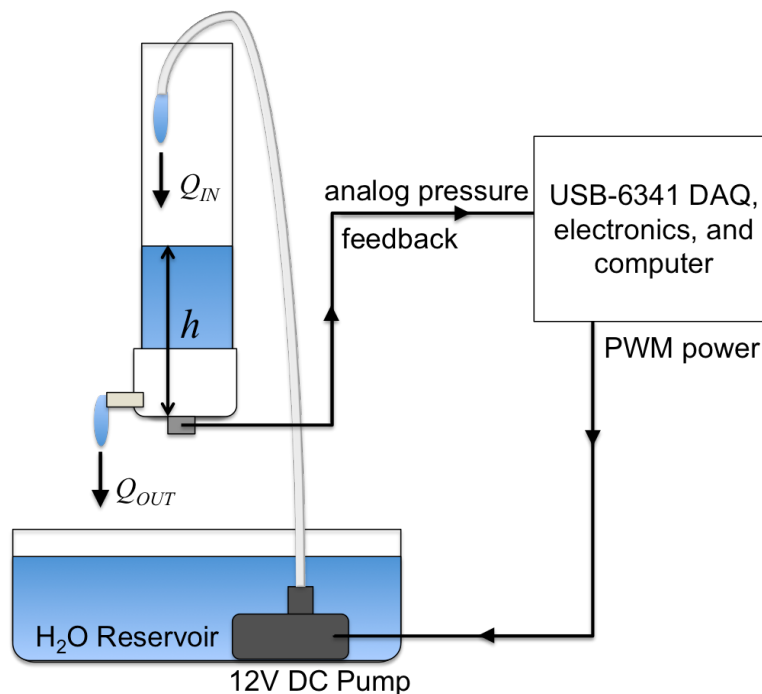
Deliverables: Checked lab notebook, Technical Memo

### Overview

In this lab, you will implement several different methods to control the water level in a flow-through tank. Shown in Figure 1, the tank is continuously drained into a reservoir and replenished by a 12V DC submersible pump. The volumetric flow rate out of the tank  $Q_{OUT}$  is related to the hydrostatic pressure  $p = \rho gh$  by Poiseuille's Law

$$p = RQ_{OUT} \quad (1)$$

where  $R$  is the resistance of the outlet nozzle. (Poiseuille's Law is analogous to Ohm's law, where voltage is replaced by pressure, and current replaced by flow rate.) A 12V DC submersible pump drives the volumetric flow rate in to the tank  $Q_{IN}$ , which can be controlled using a PWM MOSFET circuit! The tank will reach some steady-state equilibrium height  $h_{\infty}$  when  $Q_{IN} = Q_{OUT}$ , and a feedback controller can be used to very precisely control that height.



**Figure 1** – Feedback from an analog pressure transducer is used to control the height in a flow-through water tank.

## Part I: Sensor Calibration and Data Acquisition

### Background

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 as a function of time.

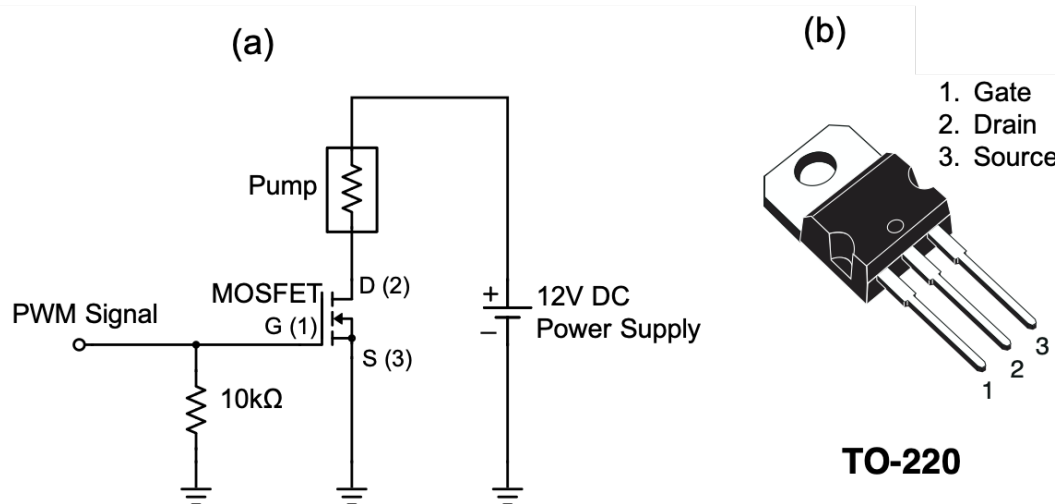
### Experimental Procedure

1. Hang up your coat and put your backpack off to the side.
2. Close the valve at the bottom of the tank. Check that the tank is firmly mounted by two beaker clamps above the dishpan (reservoir).
3. Make sure the breadboard is turned OFF. (You will not need the bread board power supplies. It should be off for the entire lab.)
4. 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.
5. Wire up the analog pressure transducer using the following connections:
  - a. Red  $\rightarrow$  +5V
  - b. Black  $\rightarrow$  GND
  - c. White  $\rightarrow V_{OUT}$  (output voltage, see next steps)
6. Use the handheld DMM to measure the output voltage  $V_{OUT}$  as a function of the water height in the tank  **$h$  in units of inches**. Apply a linear fit to the data to determine the linear calibration equation  $h = kV_{OUT} + z$ .
7. Create a copy of the thermistor V.I. from the first C1 lab. Rename the copy with an intelligent file name (i.e. "C4\_water\_height\_yourName.vi").
8. Connect the pressure sensor to the USB-6341. Open the express VI in your LabView code to determine where to connect the white  $V_{OUT}$  wire.
9. Modify the LabView code. Replace the thermistor calibration formulae with the linear calibration curve you just measured for the pressure sensor. Re-label the graphs and indicators with appropriate units.
10. Test the program to make sure it works correctly.

## Part II: System Identification and Characterization

### Background

You will now measure the flow characteristics of the pump and tank, and determine parameter values for the model you developed in the pre-lab assignment.



**Figure 2** – (a) An N-Channel MOSFET allows a low power digital signal to control a large amount of current passing through the 12V DC pump. (b) The pin-out for the TO-220 power MOSFET.

### Experimental Procedure

1. Measure the inner diameter of the tank. Use it to calculate the cross-sectional area  $A_T = \pi r^2$ . Record these values in your lab notebook.

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

2. Make sure the breadboard is turned OFF. (You will not need the bread board power supplies. It should be off for the entire lab.) You will use an external 12V DC power, because the breadboard power supply cannot produce enough current to drive the pump.
  - a. Make sure the kill switch is in the OFF position. Connect the 5.5mm barrel connector on the 12V DC power supply to the inline kill switch.
  - b. Connect the other end of the kill switch to the screw terminal adapter.
  - c. Establish a 12V bus line separate from the 5V bus line used to power the sensor. Use red and black wires to connect the + and – screw terminals to another set of vertical bus lines on the breadboard.
3. Construct the PWM MOSFET circuit shown in Fig. 2 with the 12V DC pump. Note that the red wire on the pump should go to the positive side of the circuit, and the pump will not work if it is connected in reverse polarity.
4. Use the function generator to produce a PWM input to drive the gate. Use a “Pulse” function with a frequency of 500 Hz, a high value of 5V and low value of 0V.

5. 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.
6. Make sure the tank is positioned so it will drain into the reservoir and the valve is open.
7. 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:** Be prepared to quickly turn OFF the pump to avoid a catastrophic water spill. Keep your finger on the kill switch, so you can quickly disable the pump.

8. 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.
9. Turn off the pump by turning off the output from the function generator.
10. Close the valve on the tank. Use the stopwatch to determine the flow rate in ( $\text{in}^3/\text{s}$ ) for a small value of % duty cycle.
11. Drain the tank and measure the flow rate for several more duty cycles up to about 50%
12. Plot the flow rate in ( $\text{in}^3/\text{s}$ ) as a function of % duty cycle.
13. Assume the flow rate is linearly related to the duty cycle, such that  $S = a(\% \text{PWM}) + b$ . Apply a linear curve fit, and write down the parameters  $a$  and  $b$  in your lab notebook. Note that  $-b/a$  is the threshold for flow to begin.
14. Determine the time constant  $\tau$  using a method similar to the temperature controller. Beginning with an empty tank, open the valve and start the LabView program. Turn on the pump with a PWM duty cycle about 5 - 10% above the threshold. Allow the tank to fill up to steady state, then turn off the PWM and allow the tank to fully drain.
15. Save the data of height vs. time. Use it to determine the resistance of the time constant  $\tau$ .
16. Disconnect the function generator and turn it off.

## Part III: Open-loop Controller

### Background

You will now use the model you derived in the pre-lab assignment and the information from Part II to control the height of water in the tank. You may use either the Arduino or LabView to implement the controller, but you must use LabView to record the data.

### Procedure

1. Decide which programming environment you will use to implement the open-loop controller: the Arduino or LabView.
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. Consider the formula for the steady-state equilibrium height  $h_s$  you derived in pre-lab assignment. Use this formula, along with the parameters measured in Part II, to determine the relationship between flow rate  $S_s$  and equilibrium height  $h_s$ .

4. Write a program takes a height set-point  $h_s$  as an input and drives the pump at the correct flow rate to achieve  $h_s$  as the equilibrium. The program should calculate a constant pump speed at the beginning using the formula you derived in the pre-lab. You may have to tweak your time constant a bit to make it more accurate.
5. Test the program to make sure it has been implemented correctly. Note that the open loop controller is *not* expected to perform well at all, which is why we will implement feedback in the next part of the lab.
6. Use LabView to save the height vs. time data for a few different set points.

## Part IV: PI Feedback Control

You will now implement a PI feedback controller to control the height of water in the tank.

### Procedure

1. Begin by using your C3 PID temperature controller Arduino code. Modify it as necessary. Delete any derivative feedback; you will not need it.
2. Note that there is a minimal threshold PWM signal for the flow  $Q_{IN}$  to begin, which effectively corresponds to zero flow. In your control loop, you will need to add this threshold value to your calculation. Specifically, you want to keep the tube filled to the top with water, even if the flow rate is zero.
3. Test the control loop at a set point of about  $h_s = 4''$ . Begin with  $k_I = 0$  and a very small value of  $k_p$ . Make sure you are confident the system is working correctly before you move on to the next step.

**CAUTION:** Keep your finger on the kill switch!

4. Manually tune the controller. Test it with several different set points.
5. Use LabView to save the height vs. time data for a few different set points.

## Part V: Introduction to LQR

### Background

We will now use a method known as the *Linear Quadratic Regulator* (LQR) that combines the open-loop controller and proportional feedback control. First, we will calculate the necessary flow rate  $S_s$  for a desired equilibrium height  $h_s$ , as in Part III. Then, we will add proportional feedback as in Part IV, such that the flow rate is determined by

$$S = S_s + k_p (h_s - h). \quad (3)$$

The linear quadratic regulator method can be used to determine the optimal gain  $k_p$  for a system governed by a first order linear differential equation

$$\dot{x} = Ax + Bu, \quad (4)$$

where  $x$  is the parameter you want to control,  $u$  is the input you provide, and  $A$  and  $B$  are constants determined from system identification and characterization. For the case of the single water tank, we have  $x = h - h_s$ ,  $u = S - S_s$ ,  $A = -1/\tau$ , and  $B = 1/\pi r^2$ . This is a simple *single* parameter application of the LQR method. Over the next few weeks, we will unleash its full potential and use it to control systems with *multiple* parameters.

As an important side note, we do not directly set the flow rate, rather we set the duty cycle of a PWM signal. Recall from Part II that the flow rate  $S$  is linear related to % duty cycle

$$S = a \cdot (\%PWM) + b, \quad (5)$$

where  $a$  and  $b$  are constants you determined from your linear fit. Combining Eqs. (3) and (5) yields

$$\%PWM = \frac{S_s - b + k_p (h_s - h)}{a}, \quad (6)$$

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. Write a feedback controller that uses the proportional feedback given by Eq. (6).
  - a. You will need to calculate the flow rate  $S_s$  for the set-point  $h_s$  as you did in Part III using the formula from the pre-lab assignment.
  - b. You will then implement the proportional feedback as you did in Part IV. Do *not* include any integral feedback.
  - c. Use the values of  $a$  and  $b$  you determined in Part II.
2. Use the `lqr()` function in Matlab to determine the optimal proportional gain  $k_p$ .
3. Test the controller using the optimal gain with several different set points.
4. Use LabView to save the height vs. time data for a few different set points.

## 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. A plot of the measured flow rate in  $\text{in}^3/\text{s}$  as a function of the % duty cycle for the PWM circuit.
2. A table containing the measured system parameters area  $A_T$  [ $\text{in}^2$ ], time constant  $\tau$  [s], PI feedback coefficients  $k_p$  [ $\text{in}^2/\text{s}$ ] and  $k_I$  [ $\text{in}^2/\text{s}^2$ ], and calculated LQR coefficient  $k$  [ $\text{in}^2/\text{s}$ ].
3. A plot of the water height vs. time for a few different set points for the PI feedback controller (Part IV).
4. A plot of the water height vs. time for a few different set points for the LQR feedback controller (Part V).

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

- Include any relevant equations you derived in the pre-lab assignment.
- Which behaved better? The PI controller or the LQR controller?
- Discuss the pros and cons of the open-loop controller, the PI feedback controller, and the LQR feedback controller.

## Appendix A

### Equipment

- USB-6341 DAQ
- PC computer with LabView
- Standard A&C lab kit in large tote (above lab bench)
- Dish pan with 12V DC pump and 2" tube
- Lab stand with 2 beaker clamps
- Single tank with sensor