Lab Week 9: Liquid Level Physical System Modeling

0. Objectives

- Learn about block diagrams and modeling of the physical liquid level system

1. Overview

This multi-week lab consists of using MATLAB Simulink and Arduino microprocessors to control a liquid level system. The system operates by manipulating the outlet valve to maintain a desired water height regardless of inlet flowrate.

2. System Description

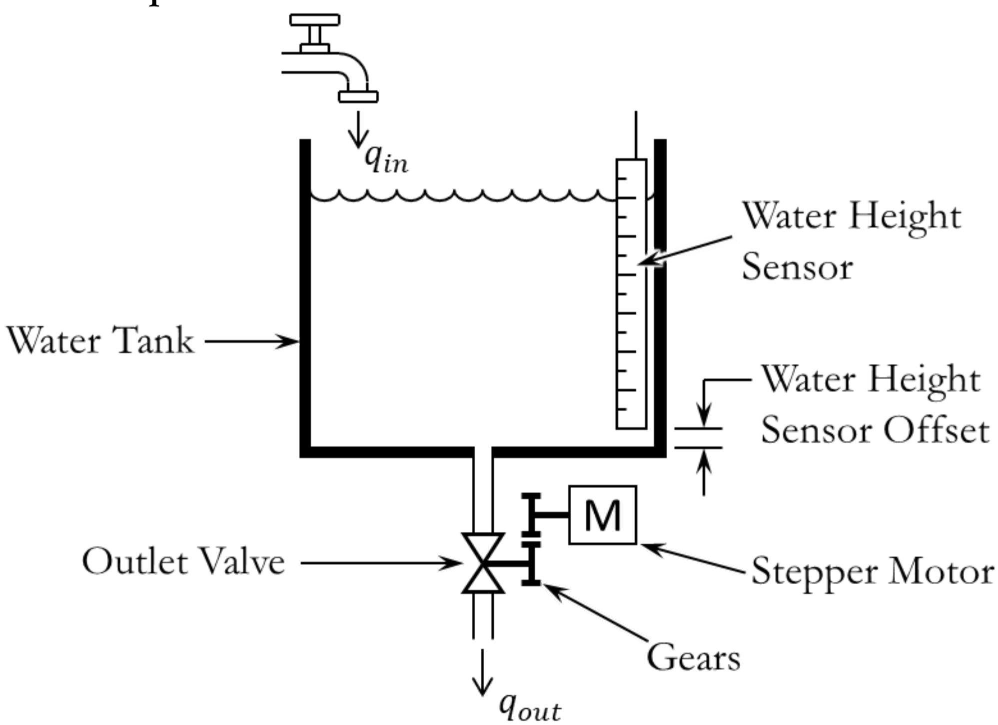


Figure 1: Mixing tank schematic

2.1 Main System Components:

- Outlet valve: controls outlet flowrate to maintain desired water height
- Stepper Motor
  - Opens and closes outlet valve via gears
  - When outlet valve is fully closed, the stepper motor position is ‘steps = 0’
  - When outlet valve is fully open, the stepper motor position is ‘steps = 1020’
- Water height sensor
  - Detects water height
  - Also referred to as an eTape sensor
- Water tank

2.2 Main system variables:

- Inlet flowrate (q_{in})
  - Inlet flowrate of water
  - Value of flowrate is unknown, uncontrollable, and can vary
  - Units: cm^3/s
- Outlet flowrate (q_{out})
  - Outlet flowrate of water
  - Flowrate will vary depending on outlet valve position **and** actual water height
  - Initial outlet flow rate is 0 cm^3/s
  - Units: cm^3/s
- Outlet valve position (X_{valve})
  - Fully closed position = 0 steps
  - Fully open position = 1020 steps
  - The valve position can be anywhere between 0 steps and 1020 steps
  - Initial valve position is 0 steps
  - Units: steps
- Water tank cross-sectional area (A_{tank})
  - Assume constant
  - 643 cm^2
  - Units: cm^2
- Setpoint water height ($h_{setpoint}$)
  - The desired water height
  - Units: cm
- Actual water height (h_{actual})
  - This is the output variable; the variable we want to control
  - The setpoint water height will be specified. By adjusting the outlet valve, the actual water height will change until it matches the setpoint water height
  - The initial actual water height is 0 cm
  - Units: cm
- Sensor water height (h_{sensor})
  - This is the output/reading of the water height sensor
  - Units: cm
- Water height sensor offset (h_{offset})
  - 3.5 cm
  - Units: cm

3. Objectives

Using MATLAB Simulink and Arduino microprocessors, a liquid level system will be controlled by manipulating the outlet valve to maintain a desired water level regardless of inlet flowrate. The basic steps for completing the multi-week lab are as follows:

1. Create a block diagram of the full system simulation model (Lab Week 4)
2. Characterize and model system components (Lab Week 5)
3. Build simulation model of full system using Simulink (Lab Week 5)
4. Run simulation model to optimize controller and predict behavior (Lab Week 6)
5. Build physical system control model
    - Create new block diagram that will be used for the physical control model
6. Compare simulated results and experimental results

4. Background: create block diagram and model for the physical system

The Simulink model that you have built allows for simulation of the mixing tank system and optimization of the controllers. In order to actually interact with the physical system, a new Simulink model will be used. Using the concepts and techniques from your Simulink model created in previous lab weeks, create a new Simulink model that will be used to interact with the physical system. The physical system has two components that you will interact with:

1. Outlet valve (via the stepper motor)
2. Water height sensor

The Simulink physical system control model will be used to send information to two Arduino microprocessors. These Arduinos will control the outlet valve (via the stepper motor). The information from the water height sensor is collected by the Arduinos and sent back to the Simulink control model. The Simulink control model will have two inputs. These inputs are sent <u>from</u> the Arduinos <u>to</u> the Simulink control model.

1. Actual Valve Position [steps]
    a. The current position of the outlet valve
    b. Valve fully closed: 0 steps
    c. Valve fully open: 1020 steps
2. eTape Water Height [ADC value]
    a. Note: the output of the water height sensor is different than the real-world water height because of the height sensor offset (see Figure 1)
    b. The water height sensor is attached to an Arduino's ADC; the value of the Arduino's ADC is sent to Simulink
    c. Experimental data will be provided that relates eTape ADC value to eTape height measurements (see Blackboard)
    d. Possible range of ADC value is 0 to 1023

The Simulink physical system control model will have one output. This output is sent from the Simulink control model to the Arduinos.

1. Valve Command Position [steps]
    a. The desired valve position (may or may not be different than actual valve position)
    b. Valve fully closed: 0 steps
    c. Valve fully open: 1020 steps

You will be provided with Simulink blocks to add to your Simulink physical control model (see Figure 2). You will **not** be adjusting or modifying any portion of these blocks. They are required in order to make your physical control model properly interact with the physical system.

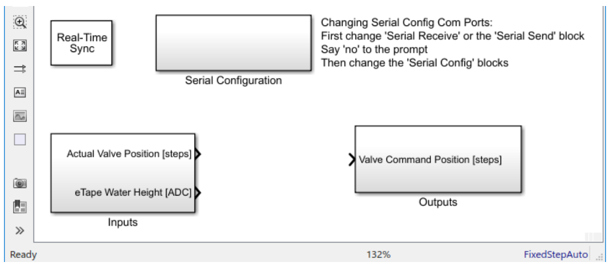


Figure 2: Simulink blocks provided for control model

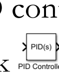
5. Assignment

5.1 Create Block Diagram

Using the two inputs and one output, create a block diagram of your physical system control model. Once again, the setpoint will be the desired water height with units [cm]. The block diagram for your physical system control model will not be identical to the block diagram created in Lab Week 4.

5.2 Create Simulink Model for Physical System Control Model

After the control system block diagram is finished, create a Simulink model that will be used to control the physical system. Below are specific requirements that your model must follow. You will not be able to run the model unless your computer is connected to a test stand. This week you are only constructing it based off of the block diagram in Section 5.1

- Solver type = Fixed-step
- Solver time step = 0.1 seconds
- Solver end time = 400 seconds
- You must build your own controller. For example, if you are implementing a PID controller for the water height portion of the model, you cannot use the Simulink PID block , you must build your own PID controller using basic Simulink blocks (see textbook Figure 5.3)
- If you use any Derivative, Integrator, Transfer Fcn, etc. blocks, you must use the discrete versions of those blocks, not the continuous versions.
- Using experimental data, convert water height sensor ADC values to cm (see Blackboard)
    - There are two test stands, so you will have two different calibration curves
- Valve position command [steps] has a valid range of 0 to 1020 steps.

6. Deliverables

- Problem 5.1
    - Digital print out of your block diagram (it cannot be hand-drawn)
    - Provide a label and units to all flow arrows
    - Provide a label for all blocks
- Problem 5.2
    - Screen shot of Simulink file
    - Provide a label and units to all flow arrows
    - Provide a label for all blocks
    - Email Simulink file to Gabe Wilfong (gwilfong@purdue.edu)
- You do not need an executive summary
- Don't forget a cover sheet