The University of Rhode Island
Department of Mechanical, Industrial and Systems Engineering

MCE433 – Mechatronics
Fall 2022

Final Project Assignment
Assigned: Mon 11/21 Due: Wed 12/21 (2.00 – 7.00 pm demo time)

The purpose of this assignment is to apply the concepts learned in the course to the operation and control of a mechatronics system. You are given the choice to select the particular system you can work on. You can work on a system provided for you in the lab or another system related to your capstone project, research project, ... etc. The basic requirements in this project include:

1. The use of a microcontroller, a PC or a combination of these platforms to implement the code processing for the project.
2. The use of state transition diagrams to design the control software for the project. The code operation should not be trivial (should have several distinct states of operation).
3. The project operation should include at least 3 of the following operations: A/D conversion, PWM actuation or D/A conversion, timing, and digital input/digital output.
4. Your project should include at least four different electronic components such as transistors, diodes, logic gates, LEDs, or other circuit elements such as switches or relays.
5. The project should have a scrolling plot of the controlled variable if you use the PC as the control platform.
6. Use of serial interfacing to send control commands from the PC as well as displaying status information on the operation of the system (if you used a microcontroller as the control platform).

Below is a list of projects that you can do that are available in the Mechatronics Lab.

- **Control of an oven** (similar to that described in SP4 in the Lab book). The oven will be simulated using a copper plate heated by a 10-W flexible heater. Use an RC servo to give visual indication of the temperature and a computer fan to provide cooling of the plate. The oven temperature should be within the range of 30-50 °C. Also use a buzzer to give an audible signal that the oven is ready. The buzzer should be turned on once for 3 seconds the first time the oven temperature reaches the desired temperature when the desired temperature has changed from the previous setting by at least 3 degrees.

- **Feedback Control of a DC linear actuator system** (similar to that described in SP3). You can use also use a PC to control the actuator. The actuator should be controlled to move to one of three defined positions, called In, Middle, and Out. Use a buzzer to give an audible signal for 3 seconds to indicate that the actuator has reached the midpoint of one of the three defined positions. Use running LEDs to visually show direction of motion, and different color LEDs to indicate location in one of the three defined intervals. The controller should also display the desired and actual position of the actuator.

- **Simulate the operation of a tank filling system** using Python and Arduino. Assume a tank with a constant cross-sectional area and a height of 100 cm. The tank should have three modes of operation: Off, On and Auto. In the *Off* mode, the input flow valve is turned off, while in the *On* mode, the input flow valve is turned on. In the *Auto* mode, the input flow valve is turned on/off to maintain a user specified desired tank height level (should be set to be below the maximum height level of 100 cm). In the *Auto* mode, the input valve is turned on when the tank height is 0.5 cm below the desired tank

height. Assume that the tank dispenses fluid through a horizontal outlet pipe and valve located at the bottom of the tank. Fluid is dispensed when the command *DispenseOn* is pressed and is shut off when the command *DispenseOff* is pressed. When the input flow valve is turned on and the output valve is turned off, the tank height should increase at the rate of 2 cm/sec. When the input flow valve is turned off, and the output valve is turned on, the tank height should decrease at the rate of 0.5 cm per second. Use a combination of Tkinter warning labels, a plot, and actual LEDs to show the status of the tank. For example:

- Use yellow LEDs to indicate if tank height is either above the high-level mark (such as 90 cm) or below the low-level mark (such as 10 cm). These LEDs show be off if the tank height is within the normal range. Use also warning labels to convey the same information.
- Use another 5 LEDS to indicate the tank height (each LED will represent 20% of the tank height).
- Use a green LED that keeps flashing every 1 second when the tank is dispensing fluid.
- Use a buzzer (insert a 100Ω resistor between the digital line and the positive side of the active buzzer) that turns on for 3 seconds the first time the tank reaches the desired height if run in auto mode.
- Use a potentiometer to select a filling rate that vary from 1 to 4 cm/sec. Display in your Python code the currently selected filling rate.
- Use a plot to dynamically show tank height during the last 60 seconds.

When the *DispenseOn* button is pressed, the button is made invisible, and the *DispenseOff* button is made to be visible. Similarly, When the *DispenseOff* button is pressed, the button is made invisible, and the *DispenseON* button is made to be visible. At startup, no warning label should be displayed. Also at startup, the tank control mode should be the off mode, the tank starting height level is 50 cm, and only the *DispenseOn* button is visible.

Note: It is suggested that you work in groups of two in any of these projects unless your work or class schedule makes it extremely difficult to do so.

An important aspect of this assignment is the design and documentation of the software for your project. You should start with a block diagram that shows all the components of your system and their interaction. Document the overall structure of your code, and how the different parts of the code interact together. For each control task in your project, you also need to provide a state transition diagram.

You are responsible for any hardware interfacing that might need to be performed if you decided to use systems other than those available in the Mechatronics Lab**.**

Project Selection

You are required to investigate further the project ideas suggested above and submit by email (one from each group) before or by Monday, November 28 (2 pm) your top two choices for a project or your own project idea. If you plan to work on a project of your own, submit a short document that contains the following: 1) a brief description of the project, 2) and a block diagram of the interaction of the main components. I will get back to you by November 29th on your selection.

Project Demo

1. You need to demonstrate the operation of your system in the lab on Wed, December 21, 2021 starting at 2 pm. A limited number of demonstration slots will be available between 4.00 to 6.00 pm on Tuesday December 20 for those who are not available on the 21st.
2. A short oral exam will be conducted to determine the contribution of team members.

Project Submission Requirements:

You need to hand in on that day **just before your demo time slot** one package from each group **(**in paper form as well in a single PDF file format) that has the following items**:**

a. A short report (2-3 double-spaced pages) explaining the approach for solving the project and how your program works (describe so that one can understand how the different pieces of the code were structured and integrated).
b. Detailed circuit diagram for interfacing the components in the project.
c. State transition diagrams for code with all transitions between states shown and the states in the diagram correspond exactly to those used in the program.
d. Listing of all developed code and screenshots of the code in operation.

Important Note: You should not copy or use code from other students or teams. Any such usage is considered plagiarism and all involved students will receive zero for this assignment and possible university disciplinary action.