Submission deadlines:

- **Each** EE 200 student must demonstrate their LabVIEW realization of the finite state machine for loading passengers into minecarts in the solution for Problem Set 4 Problem 15 using LabVIEW.
- Demonstrate completion of project 2 during evening laboratory sessions on either Tuesday April 22, or Wednesday April 23, from 6:30 pm to 8:30 pm in 302 EE West. You must sign up for a ten minute slot on either Tuesday or Wednesday using SignUpGenius. You will receive an email message via your Penn State email address when the sign up sheet is available online at SignUpGenius. Slots are filled on a first-come, first-served basis.

Realize the finite state machine finite state machine for loading passengers into minecarts in the solution using LabVIEW and the myDAQ and demonstrate the system to the laboratory instructor. To receive full credit, your system must comply with the following specifications:

1. Use the signal assignments in Table 1.

Signal Designation	**Signal Description**	**myDAQ DIO line**
X	Output: X	DIO 0
Y	Output: Y	DIO 1
Z	Output: Z	DIO 2
E	Input: Enter	DIO 5
L	Input: Leave	DIO 6
R	reset: command	DIO 7

Table 1: The myDAQ DIO line assignments.

2. Connect the input switches and output LEDs as indicated in Figure 1. Note that pressing a given button sets the signal to a logic high state, while the corresponding logic level at the myDAQ connector is set to a logic low state. For example, when the user presses the Leave button, the value of L is logic high while the corresponding logic level at DIO 6 is low.
3. Implement the FSM in LabVIEW using the architecture shown on slide 19 of Laboratory #20 as a guide. Use the provided code in the directory project_2_LabVIEW_Solution_Template as a template.
4. Use a LabVIEW project file and virtual folders to organize your subVIs and control files.
5. A subVI READ INPUT should a return 16-bit unsigned integer output, called input, to represent E and L as shown in Table 2.

Input	**input**
$\overline{L}\ \overline{E}$	0
$\overline{L}$ E	1
L $\overline{E}$	2
L E	3

Table 2: Representation of the input signals E and L using a single integer variable.

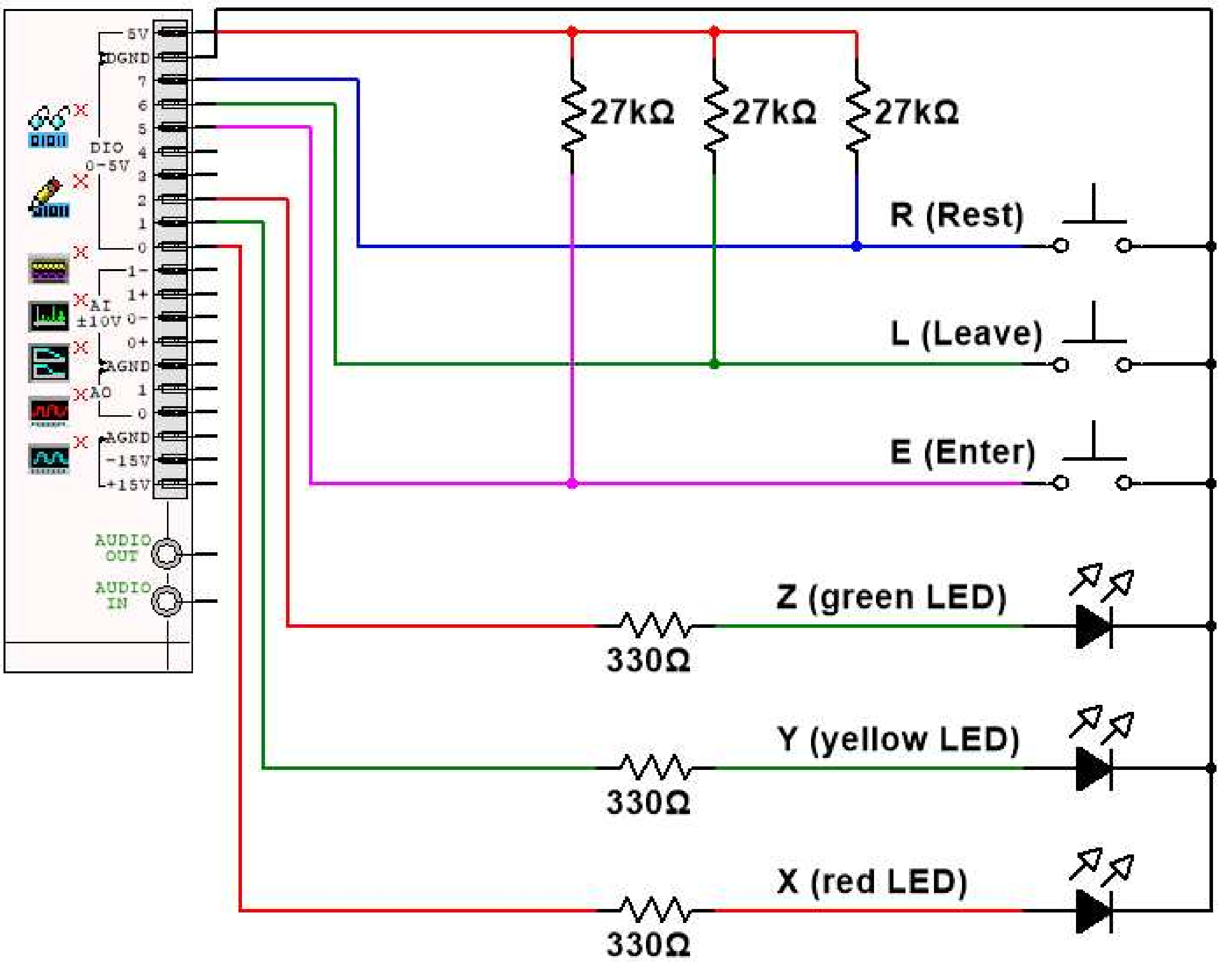

Figure 1: myDAQ connections to input switches and output LEDs.

6. Use a functional global variable to store the state. The state must be typed defined using a control file.
7. Use a subVI TRANS STATE to transition between states as shown in the slow loop of the VI on slide 19 of Laboratory #20. This subVI must realize the finite state machine using two nested switch statements as in Laboratory #20. The inner switch statement acts on the output integer generated by the subVI READ INPUT.
8. Read the inputs E and L once per second, and appropriately update the state and outputs.
9. Display the current state, SO, S1, S2, or S3, on the front panel using a type defined control.
10. Poll the reset signal R and the front panel stop control every 10 ms using a fast sloop as shown on slide 19 of Laboratory #20 . Pressing the button R must sets outputs X, Y, and and Z to the logic low state and the state to S0.

During the project demonstration, the instructor will award points using the following criteria.

1. (5 points) LabVIEW code must display the current state, SO, S1, S2, or S3, on the front panel using a type defined control.
2. (20 points) LabVIEW code must follow the architecture shown on slide 19 of Laboratory #20 and use a LabVIEW project file and virtual folders to organize your subVIs and control files.
3. (20 points) The LabVIEW code must use two nested switch statements to realize the finite state machine, and that the inner switch statement acts on an integer representing the values of the inputs E and L as defined in Table 2.
4. (15 points) Demonstrate that pressing the rest button triggers returns the system to state S0 and appropriately sets the outputs within 10 ms.
5. (20 points) Demonstrate that system responds correctly to the inputs E and L.
6. (20 points) The grader will ask you a question regarding the LabVIEW realization of the finite state machine.