Rodrigo Becerril Ferreyra
CECS 346 Section 03
Lab 4
02 December 2020

1 Introduction

The purpose of this lab was to set up and test an external part called a proximity sensor. The proximity sensor is a 5 V system that changes its output signal to reflect whether or not it detects an object in front of it. The program used to test the sensor uses the on-board RGB LED to display the state of the entire system: if the LED is green, then the sensor does not sense anything; if the LED is blue, then the sensor senses an object in front of it; a flashing red LED represents a transition between the previous two states. In addition, all inputs are treated for mechanical bounce.

2 Hardware

As previously stated, I am using an infrared-radiation proximity sensor to drive the project. Its `signal` output is connected to pin `B[0]`. To avoid random movement and unpredictable triggering of the sensor, it is taped to a cardboard box for stability.

To change the desired minimum range that the sensor detects objects at, it is required to turn the potentiometer to adjust its resistance. However, when I tried to turn it with a screwdriver, the plastic wore out and I was unable to turn it. I had to unsolder the potentiometer and re-solder in one of my own. Even with this modification, I could not get the sensor to detect anything farther away than approximately 8 cm.

3 Software

Internally, the system uses three different interrupts: one for Port B (where the sensor was connected), one for Port F (where the two on-board buttons `SW1` and `SW2` are connected), and one for the SysTick timer. The SysTick timer is used to time the delays for the flashes for the transition between the two main states. If the current state of the LED is green, when the sensor detects an object, the LED transitions from green to blue (green $\rightarrow$ flashing red $\rightarrow$ blue). Pressing `SW1` also achieves this, while pressing `SW2` has no effect. Conversely, if the current state of the LED is blue, when the sensor stops detecting an object, the LED transitions from blue to green (blue $\rightarrow$ flashing red $\rightarrow$ green). Pressing `SW2` also achieves this, while pressing `SW1` has no effect.

Debounce is implemented simply by waiting a certain amount of time after an interrupt is triggered before handling the interrupt. If the interrupt trigger condition is no longer true, then the interrupt service routine is not executed.

4 Media

- A YouTube video of a demonstration of the system can be found here: https://youtu.be/Y_fqIKj0XAg
- The Figure is a schematic diagram of the system.

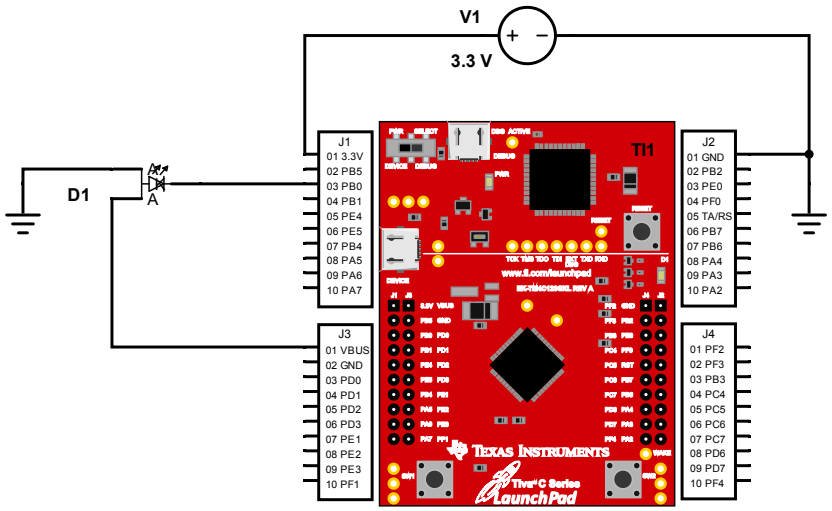

Figure 1: Schematic diagram of the system.

- The Figure 2 is an image of the system.

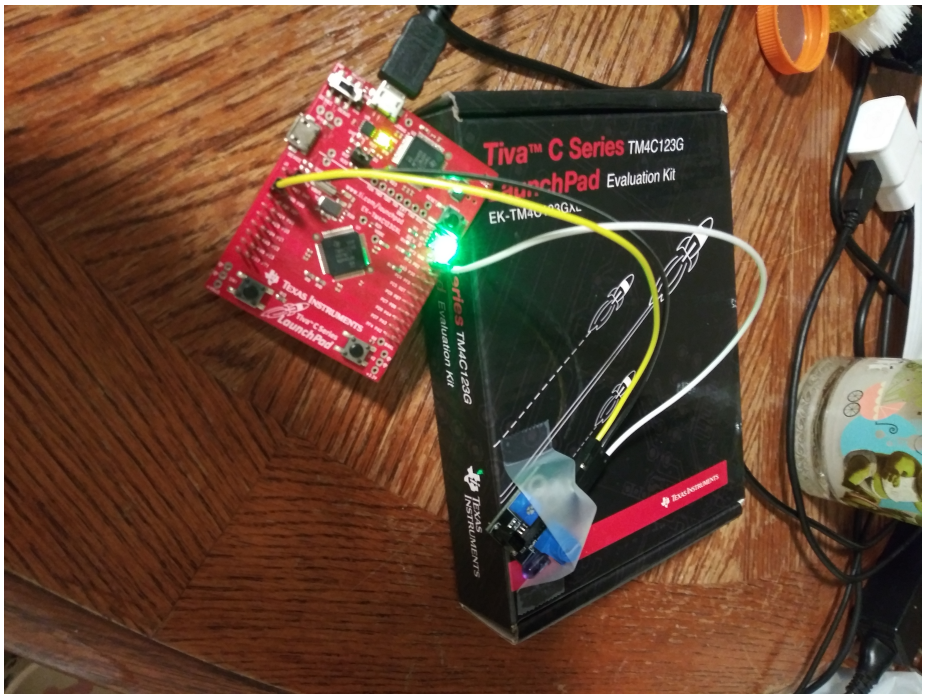

Figure 2: Image of system running.