ENGINEERING TECHNOLOGY
& INDUSTRIAL DISTRIBUTION
TEXAS A&M UNIVERSITY

Department of Engineering Technology and Industrial Distribution
ESET 349 Microcontroller Architecture

Lab 5 – Detection of Obstacles[1]

Objectives

1. Design an algorithm to handle multiple input-output circuits simultaneously.
2. Write a complete assembly language program to implement the algorithm.

Overview

The infrared obstacle avoidance sensor module as shown in Fig. 1 detects the presence of an object. Hence, the sensor can be used in obstacle avoidance for robots, line detection for a line follower and so on. The detection distance is limited to 12 inches only. When an obstacle is detected, a low digital signal is generated at its output pin which can be used as an input signal for further processing. The potentiometer on the sensor can be tuned to vary the sensitivity and hence the effective range of obstacle detection. You may need to tune the potentiometer (longer range sensor) based on the functioning of your sensors.

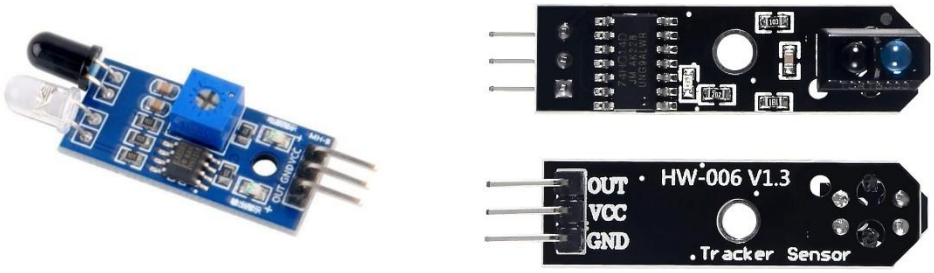

Figure 1. Infrared obstacle sensors, longer range 0-30cm (blue) short range 0-25mm(black)

Your Tasks

For the experiment, consider three of these sensors in your "vehicle": one on the left side, one on the right side and one on the rear side. As the vehicle moves, sensors will detect obstacles close to each of them. When an obstacle is detected, the corresponding LED should glow. The LED should turn off once the obstacle is not present anymore. Additionally, add a photoresistor to remind drivers to use the headlights at dusk.

Place the three obstacle avoidance sensors and a photoresistor on your breadboard instead of the "vehicle". Use 4 LEDs – one LED for each direction (rear, left and right) and one as the headlight. You may refer to the handout of Lab 4 for information on using a photoresistor.

The following partial flowchart (Fig. 2) shows one possible way of completing this task. You are encouraged to develop a detailed flowchart as per need, or to come up with another solution.

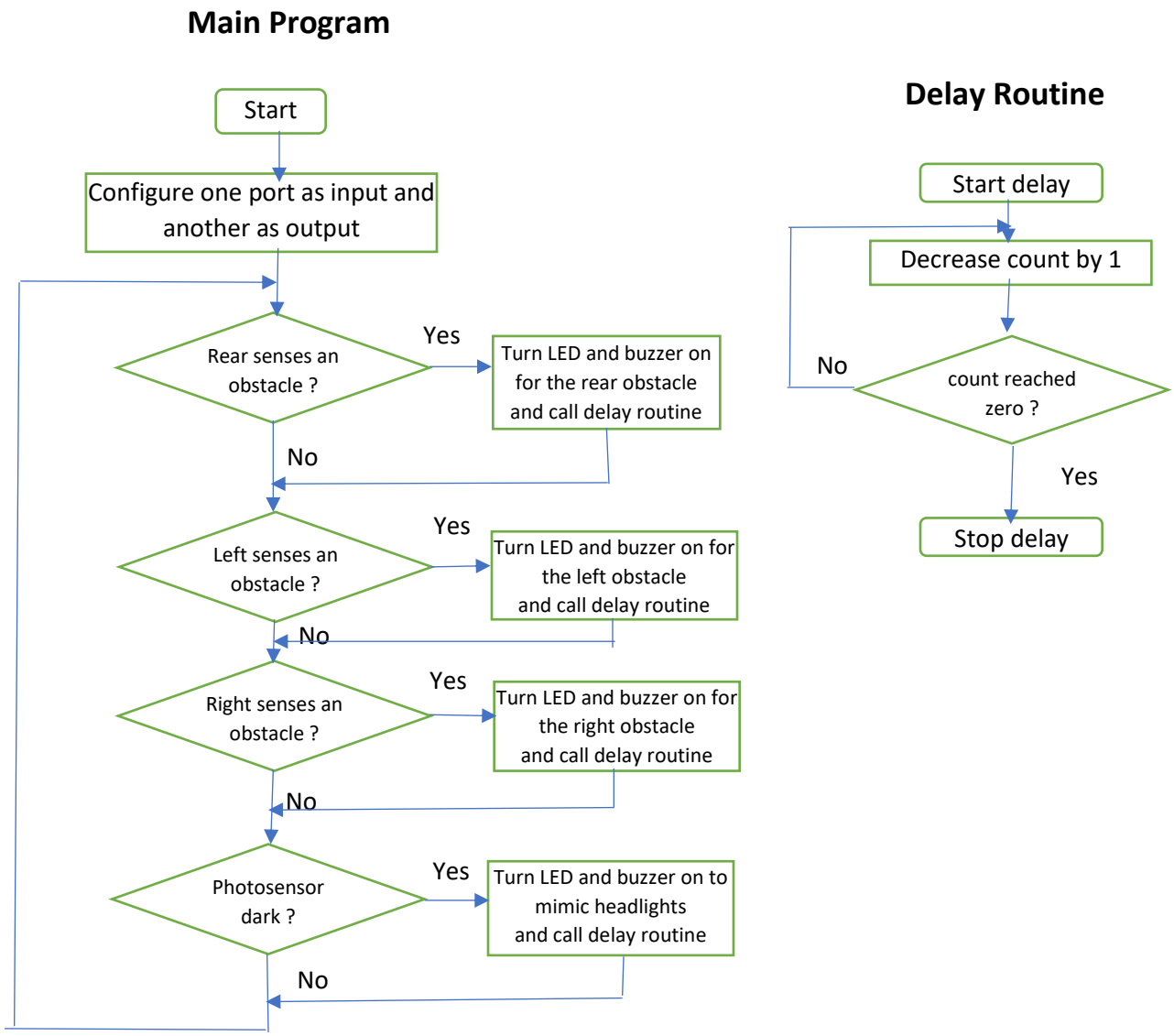

Figure 2. An incomplete flowchart for this lab

The above flowchart is a guide to understand the mechanism of writing a program to interface the obstacle detectors and a photoresistor. **If you do not have a buzzer, ignore the pertinent instructions.**

Is the delay routine required as shown in the flowchart? Could using a delay routine prevent inputs from being captured? Remove the delay routine if you believe it is not essential.