

CS 273 Lab 7: Miniboard Programming

Dept. of Computer Science, NMSU

Assigned October 11, 2006
Fall 2006

Due Date: a week after your lab session.

1 Programming the Miniboard

This assignment is a simple program running on your Miniboard. Your board will need one working switch connected to it and one light sensor. You need to write two versions of the same program, one that uses the switch and one that uses the light sensor.

The program should turn the red motor lights on, one at a time, for a 1/4 second each. Thus, the red row of motor LED lights should blink in sequence, one second per iteration over the four lights. Timing is important and will be part of the grade. The LEDs should blink in the order of increasing motor numbers – i.e., motor 1, motor 2, motor 3, motor 4, motor 1, In essence it is like a string of blinking Christmas lights.

For the switch version, if the switch is pressed your program should blink the green motor lights instead, as long as the switch is held down. When the switch is released, the program should go back to blinking the red lights. For the light sensor version, if a dark surface passes over the sensor (such as your thumb!), your program should blink the green motor lights, until the dark surface is removed.

When your program switches between red and green lights, either red-to-green or green-to-red, the sequencing must be maintained correctly. That is, if the third red light is on and your program notices the switch depressed or the light sensor darkened, then it should turn on the fourth green light next. So no matter which color of light is on, the 1-2-3-4-1-2-3-4... sequence should never be broken.

You do not need to detect a switch/sensor and switch to the other color in the middle of the 1/4-second one period. You can wait until it is time to switch the next light in sequence, and then check your switch/sensor.

The program is a team assignment. Each team will work together on it, and each person will submit the team's solutions.

Your program should include a subroutine that simply delays for a 1/4 of a second. It does not need to have any parameters or return values. You will have to use a counting loop (probably on a 16-bit register) in the subroutine to delay for a time period.

2 Building a car

Also, each team should work on building a car. This is not part of the lab grade, but is something you should begin anyway.

The design is, essentially, up to you. But keep in mind the tasks that it will have to do:

1. It will have to recognize running into a wall or similar structure, thus you probably want a couple of switches facing forward on the front corners of the car.
2. The car will detect a black line on a white floor and make a decision.

Most cars in the past have used a set of three light sensors facing downward, and placed horizontally across the width of the car.

Some designs that placed the sensors under the middle of the car have had problems because it is too dark under the center of the car. Placing the sensors where they have good ambient light seems to work best.

3. You should use just two motors, one each to drive a main wheel on each side. A key part of the design is making the car turn. Having four wheels on stiff axles will not let the car turn. The two drive wheels can be on stiff axles (connected directly to a motor, or to gears), but your other wheel or wheels need to allow the car to turn. A tri-wheel design might be easiest, but be creative! We do have some little “lazy-susan” turnstyles that can make a swivel wheel, but a team is only allowed one of these (due to short supply), and in general they don’t seem to perform well.

3 Demonstration and lab turn-in

You should finish the following items in this lab:

1. Your team should demonstrate a working program/board to the TA.
2. You should submit your assembly programs (each person) through the web,
3. Submit a lab report for this lab. Each person must submit a lab report, although you may collaborate with your team members your report.