MAE156a Arduino-Motor Driver Lab Sections

(Delson/Roberts, rev 09/2017)

This lab section will acquaint you with some of the I/O capabilities of the Arduino platform, using a driver board to control speed and direction of a DC motor. You will also learn about using a pulse-width-modulation (PWM) approach to get an apparently continuous analog linear output from a digital device (the MCU), with application to motor speed control. And finally you will do some software experiments to determine execution times of several Arduino program instructions. You will have four tasks to accomplish for this lab:

1. **PWM output:** Students will work in pairs. Student “A” will write a very short Arduino program [**analogWrite**(pin,duty)] to produce a PWM output of an arbitrary duty cycle. Student “B” will use the oscilloscope cursors to measure the duty cycle thus produced. Record the results (A’s programmed duty cycle and B’s measured duty cycle) on the worksheet form which follows. A and B then swap roles and repeat the exercise at some other duty cycle.
2. **Motor Speed:** Again working in pairs, using the motor driver board on the Arduino and the 12-volt DC power supply, write a short Arduino program to drive the motor at speeds of 50%, 75%, and 100% duty cycle. Include in your program instructions to (a) repeatedly read the potentiometer voltage output [**analogRead**(pin)] as the motor starts running, and (b) also plot **(Tools/Serial Plotter)** the potentiometer output results (0-1023) from the analogRead instruction.
3. **Execution times:** Different Arduino program instructions take different amounts of time to execute. This section will be done using the oscilloscope

You will be given a short Arduino test program which you type in to the Arduino editor. The loop() section of the program will show timing information by producing a timing pulse with a low-to-high transition on a pin just prior to executing an instruction, and then a high-to-low transition immediately after the instruction. You use the scope to monitor the timing pin output and determine the execution time by measuring the length of the pulse using the scope cursors.

In the program you will notice that initially all the math and I/O instructions are commented out with the **//** operator. The idea here is to determine the time to execute an “empty” loop. You then subtract the empty loop time from the measured time when executing one of the instructions. When doing this test be sure to have only one uncommented instruction in the loop.

1. **Data storage:** Repeat one of the duty cycles from step 2, but rather than plotting the measurements immediately online, put the data [analogRead()] and the time [millis()] into two arrays. Then plot the data (x=time, y=pot output). The tradeoff should be that you can improve the time resolution of the data but there will be a memory limitation on the array sizes.

*Suggestion:* Write the data into x and y arrays. Then at the end of your program read back the arrays and do a Serial.print to the Serial Plotter.

MAE156a Arduino-Motor Driver Lab - Worksheet

1. PWM output.

Programmed PWM Measured PWM

Stu A % duty cycle value (0-255) high pulse width (ms) total period % duty cycle

Stu B

1. Execution speeds. Timing using the oscilloscope.

First, run the program with all functions commented out, to determine the empty loop time. Enter your measured empty loop time here: uS.

Instruction measured time empty loop time execution time

(uS) (uS) (uS)

Serial.print (“A”) 9600:

“ 115200:

analogRead(1)

a=b+i (integers)

a=b/i (integers)

a=b+i (floating pt)

a=b/i (floating pt)

a=sin(i) (floating pt)

a=b/1.234 (floating pt)

**🡺 Report for this exercise is due at your following (week 3) lab section.**