EE 337: Interfacing to DC Motors
Lab 6

Rajbabu Velmurugan and Shankar Balachandran

31^{st} August, 2015

1 Interfacing a Motor

In the set of experiments for the week, we shall learn how to drive a DC motor using micro-controllers.

The output ports of a micro-controller cannot provide sufficient current to drive a motor. Therefore, we use an interface chip like the H bridge L293D which receives standard digital signals from a micro-controller and provides large currents to an external device. Data sheet for L293D has been posted on your Moodle site.

For this experiment, you will be provided with a breadboard, the L293D IC, a DC motor and an external power supply. You have to set up the driver IC connections to the micro-controller and motor on the breadboard to drive the motor.

DC motors can be controlled by controlling the average DC voltage applied to the motor. Analog control of this voltage is not very easy for a micro-controller (though this can be done, as we shall learn later). However, a micro-controller can change the duty cycle of the voltage applied. This can be done by applying either a voltage V or 0 through an H bridge, as controlled by a port pin of the micro-controller. If the voltage is switched between V and 0 very rapidly, the motor cannot respond to the instantaneous voltages. It responds to the average value of voltage applied to it. We will control the speed of the DC motor by using pulse width modulation (PWM). If a voltage V is applied for time T1 and 0 V is applied for time T2, the average applied DC voltage is $V \cdot T1/(T1 + T2)$. This average value can be controlled by varying T1 and T2, typically keeping (T1+T2) constant.

Problem Description

1. The DC motor drives a slotted wheel, which interrupts an infrared light beam from an LED to a photo detector. There are 30 slots in the wheel. A wheel that rotates at 1 rps (revolutions per second) would give 30 interruptions in 1 second. The circuit for this has been given in an accompanying document. By counting the number of light interruptions per second, the speed of the DC motor can be estimated. The motor's rating is maximum of 300 RPM (revolutions per minute).

   The flow for the problem is given below:

   (a) The input switches are read from the user.

   (b) The value of the input switch dictates the ON time of a 30 ms pulse that is used to drive the motor driver.

   (c) The motor driver produces an appropriate voltage which is given to the motor.

   (d) The motor runs at a corresponding RPM.

   (e) This RPM is measured back by the uController. The input switch's value and the RPM measured is displayed on the LCD. The uController should count the interruptions from the IR sensor for every 1s and use that to calculate the RPM.

This problem requires 4 timed events. Event E1 is timed interrupt every 2 ms (30ms / maximum input value 15). Event E2 is to ensure that a single PWM cycle is of time period 30 ms. Event E3 is the interruption created by the IR sensor. This has to be counted using a hardware counter.

Two consecutive E4 events will mark the time period for which the interruptions are counted so that the RPM can be measured. We require this period to be 1s.

We suggest that E1 is done using a hardware timer and E2 is managed inside the ISR of the same timer. Event E3 will require another timer to be configured as a counter. Event E4 can be managed using either the timer used for event E1 or using a new timer. Note that there are three timers in the device.

The display format must be:

```
IN RPM
XX YYY
```

Ensure that the LCD is initialized only once and use the cursor to manipulate the positions at which the values are displayed. Do not clear the LCD every time. The output format requires only the second line to be updated. Line 2 of LCD is updated only on a E4 event.

Homework

1. Write a program that uses a hardware timer to control the duty cycle of ON and OFF time of any of the LEDs on the board. You should read a 4-bit value from the switches. If the input from the switches is k, the ON time $T1$ must be $2 * k$ms. The sum of ON and OFF times ($T1 + T2$) should always be 30 ms. For example, if the input is 4, the ON time must be 8ms and OFF time must be 22ms.

   The switches must be read every 1s. You can use the same ISR routine or use a different timer for controlling the rate of sampling of the switches.

   The whole process has to run indefinitely and you should observe the LED's brightness change as per input.

   Also, connect the port to an oscilloscope during the lab to observe the PWM waveform that is being fed to the motor controller.

2. Do all the calculations required for displaying the RPM.

3. Read the documentation of the L293D and of the IR circuitry.

Lab Work

Rest of the work is part of your lab work. This includes writing ISR for the counter and displaying on the LCD. You also have to make appropriate changes to the homework if necessary.

2 Reaction Timer

In the last class, you wrote a program to measure reaction time and put the values in a register. Modify the exercise so that your program measures the actual reaction time (to be displayed in ms). Also, modify the program so that a user is prompted 5 times, each time showing the reaction time. At the end of 5 turns of a user, display the average reaction time also.

To do this, you have to convert the counter values to ms.

This homework is not part of this week and can be shown to your RA any time before 4^{th} September 2015.