Lab-4: Stepper and DC Motors

In this set of experiments, we shall learn how to drive DC and stepper motors using micro-controllers. Please review the mini-tutorial on stepper motors uploaded to your moodle page earlier. A lot of other material is available on the web. (However, information about other types of stepper motors might be confusing!) Notice that the mini-tutorial describes a simple construction with 4 pole pieces. In actual practice, many more pole pieces are used, but the operating principle remains the same.

As the mini-tutorial on stepper motors explains, the motor is rotated by applying a vertical magnetic field, then a horizontal magnetic field, then applying the vertical magnetic field in the reverse direction and finally a horizontal magnetic field in the reverse direction. A practical stepper motor has many more magnetic pole pairs than were shown in the mini-tutorial, but the sequence of operations remains the same. We can use two port pins to feed the vertical coil and another two to feed the horizontal coil. A coil with 0 applied to both ends will not be enegized at all. One which has 1 at one end and 0 at the other will generate an electromagnetic field whose direction will depend on which end is at 1 and which one is at 0. Thus the sequence of digital outputs to produce rotation will be:

Vertical Coil		Horizontal Coil	
P2.3	P2.2	P2.1	P2.0
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

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. We had used the H bridge for the previous experiment on note generation. Data sheet for L293D was posted on your Moodle site last week. A diagram showing connections of L293D H-bridge to the motor kit has also been uploaded earlier.

After applying each output, we pause for a programmable amount of time and then apply the next output. The amount of time between successive outputs will determine the stepping rate and hence, the speed of the motor. By applying the sequence in the reverse order, the motor can be made to rotate in the reverse direction.

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. 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.

The speed of a stepper can be easily determined by counting the pulses applied to it per unit of time and the number of poles in the motor. How do we determine the speed of a DC motor?

We can mount a light interrupter on the rotor and count the number of times the light beam is interrupted. This will give us a measure of the speed or rotation of the DC motor.

Homework

1. This program will eventually be used for running a stepper motor. However, as home work, we shall test it by lighting the 4 on-board LEDs in proper sequence.

   Write a program which will light the 4 LEDs on the board in the sequence suggested for driving the stepper motor. The time between each successive output should be a multiple of 100 ms, as read from the slide switches. Use timer 0 for controlling this time. If the slide switches are set to 0000, the program should terminate.

2. Write a program which will control the duty cycle of ON and OFF time of any of the LEDs on the board. The ON time should be the 4 bit number read from slide switches in milliseconds. The sum of ON and OFF times should always be 20 milliseconds.

Lab Assignments

For this experiment, you will use a breadboard, an H bridge IC (L293D), a stepper motor, a DC motor and an external power supply. The DC motor drives a toothed wheel, which interrupts the light path between an LED and a photo-detector. You have to set up the driver IC connections to the uC and motors on the breadboard to drive the motors, according to the scheme uploaded on your moodle page.

1. We define 8 speeds for the stepper motor. Speed 0 means 0 speed (static). Speed 7 is the highest speed, which should be approximately 100 RPM. Other speed levels should be proportionately spaced.

   Define one of the switches on the uC board as the 'up' button and another as the 'down' button.

You can think of speeds as ranging between +7 and -7 with the sign representing the direction of rotation. The 'up' switch should act as increment and 'down' switch should act as decrement.

Each change of the state ('on' to 'off' or 'off' to 'on') on the 'up' button should increase the speed in the clockwise direction and decrease the speed if currently rotating in counterclockwise direction. Similarly a change of state on the 'down' button should decrease the speed in the clockwise direction and increase the speed if the motor is currently rotating counterclockwise. A toggle on the 'up' button should be ignored if one is already on maximum speed in the clockwise direction.

Similarly, a toggle on the 'down' button while at speed 0 should take the motor to counter clockwise rotation with speed 1. Toggles on the 'down' button should be ignored if the speed is maximum in the counter clockwise direction.

2. The DC motor drives a slotted wheel, which interrupts an infra red light beam from an LED to a photo detector. The circuit for this has been given in a document uploaded to moodle. By counting the number of light interruptions per second, the speed of the DC motor can be estimated. Configure T0 as a counter to count the number of pulses produced by the toothed wheel on the DC motor. Use T1 to count these pulses for exactly 1 second. Drive the DC motor using an H bridge with duty cycle controlled as in the home work problem-2. Display the 4 most significant digits of this count on the LEDs on the board.