Lab 6: Reading the Encoder Inputs Using Interrupts.

Objective: Using interrupts to monitor external hardware.

Description of Lab 6:

It should be noted that inputs, especially those driven by devices, can change very quickly and therefore need to be serviced quickly. Microprocessor systems have developed hardware known as interrupts for handling such inputs. From our standpoint an interrupt is actually a function that runs anytime an input or signal changes or is in a given state. The application that we will be employing interrupts for is the input from the knob of our encoder.

A shaft encoder uses two signals, also known as channels, to indicate the motion of a shaft as well as its direction. The signals from an encoder are shown in Figure 6-1. In this graph it can be seen that only one of the signals changes at a time. Also when moving in a ClockWise (CW) or CounterClockWise (CCW) direction, the order in which the signals change is different. Note we will arbitrarily think of CW as the positive direction and CCW as the negative direction.

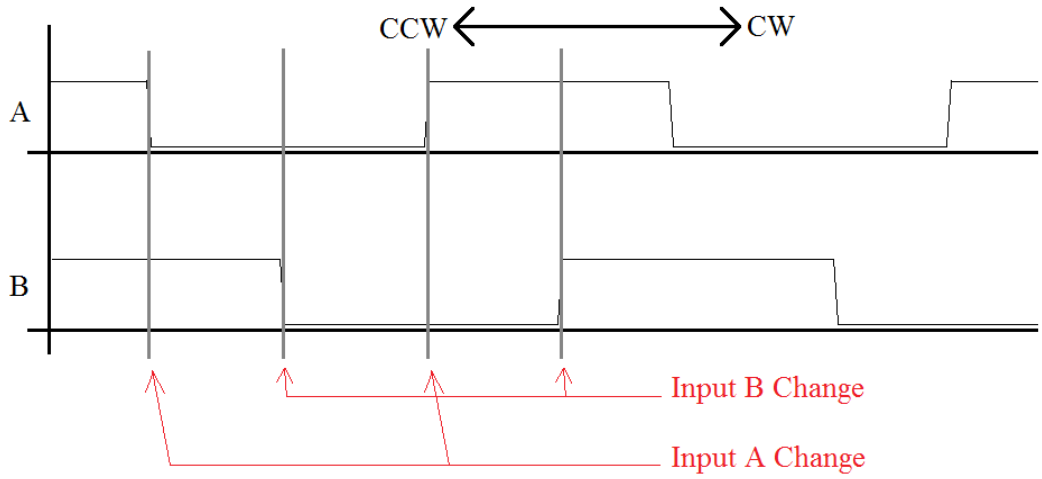

Figure 6-1. Encoder Signal with Transitions Marked.

The encoder signals A and B are connected to pins 2 and 3 of the Arduino Nano, respectively. Pins 2 and 3 were chosen since they are attached to interrupts 0 and 1 respectively. In this way a function known as an Interrupt Service Routine (ISR) can be established that will be run each time one of the signals changes. The question is, What should the routine do in order to keep track of the motion of the shaft?

Note that if we move CW and examine the transition events only on channel A, channel A always transitions into the opposite value of channel B. However, if we move CCW channel A transitions into the same value as channel B. Examining the transitions of B, we see that the opposite is true. Thus once a transition occurs in one of the signals, an interrupt routine will run, and depending upon the state of the two inputs (immediately after the transition), it will increments or decrements a variable indicating the shafts position.

ECE 241 **Lab 6** **Fall 2017**
October 3-9, 2017 **Points: 20**

The structure of the interrupt routines for handling the encoders are included below. The test of the encoders and our code will be to simply display the encoder value every 0.1 seconds.

Functions Needed: attachInterrupt(Interrupt, ISR, mode);

Program Structure:
Global Variable encoderPosition

```
// Interrupt service routines.
void MonitorA()
    if input A and input B are equal // input A and input B are pins 2 and 3 respectively.
        Increment encoderPosition.
    else
        Decrement encoderPosition.

void MonitorB()
    if input A and input B are equal
        Decrement encoderPosition.   // Note this is the opposite of what happens above!
    else
        Increment encoderPosition.

SetUp:
    Attach MonitorA to interrupt 0, (Pin 2) and set mode for any change.
    Attach MonitorB to interrupt 1, (Pin 3) and set mode for any change.

Loop:
    Every 100 milliseconds display encoderPosition on LCD screen.
```

Lab Assignment:

Prelab: Write the program described in the program section.

Lab 6:

1) Download and demonstrate the program in the prelab.
   Documenting the following as accurately as you can.
   - Number of encoder counts per revolution.
   - Number of encoder counts per "detent".

2) Change your program to decrement encoderPosition every time the push-button is pressed.
   Note for this, the code from lab 5 should be used to insure that only one decrement occurs per button press.