CS466 Lab 6 – Quadrature Decoding.
Due Thursday 4/17/2021.

This lab is the beginning for the final lab (still called #7) It's important that you get it working before moving on to #7. Start with a minimal single task template with serial working and interrupt handlers.

The motors that we will be using have incremental rotary encoders attached to them. (https://en.wikipedia.org/wiki/Rotary_encoder) When the encoder LED is powered properly 'phase-a' and 'phase-b' provide quadrature information. From the pa and pb signals you will need to keep track of motor position and velocity.

You are free to power the motor with an external power supply between 5 and 12 volts DC. Don't leave it running at 12V indefinitely as it will get hot over time. Switching polarity.

In order to calculate RPM you will need to determine how many ¼ cycle tics there are in a single revolution.

Requirements for Lab6

1. The power for the optical encoder may come off the Tiva 3.3V buss
2. The heartbeat shall report twice a second over the serial port and report;
    a. The absolute physical position of the motor in ¼ phase ticks.
        i. Absolute position 0 is set as the current position when Tiva is restarted
    b. The average RPM of the motor taken over the prior 500ms
3. Pressing Reset will reset the code and set absolute count to 0.
4. The Tiva board shall blink the green LED as a heartbeat at 1Hz

Recommended Steps for Lab 6

1. ☐ Take a look at the DC motors
    a) The motors connect to the tiny breakout board with an FFC (Flexible Flat Cable) cable. It just presses into the motor and is held via spring pressure of the contacts. The breakout board has a mechanical lock which should be opened slightly before inserting the cable then pressed back in. In all cases be sure that you are connecting the FFC so that the contacts on the cable match the contacts in the connector.
        1. Phase-A
        2. V3.3
        3. Phase-B
        4. GND
        5. MotorA
        6. MotorB

    The motor-a and motor-b lines are used to apply a DC voltage across the motor. They will not be required to be used in this lab but you need to be able to tell what is what. Using a voltmeter you can easily identify the pins for A and B as they are directly hooked to the large solder lugs on the back of the motor. Unless directed, don't apply more than 13VDC to the motor.

2. ☐ On your Tiva board Setup PortE pins 2 and 3 as input to receive the quadrature Phase-A and Phase-B signals respectively.

3. ☐ Setup an interrupt handler to receive edge transitions of A and B

4. ☐ In your ISR track the position of the motor based on the encoder interrupts. In your idle task, dump the position of the motor in encoder ticks. There is a QuadraturePhase.pdf handout in the repo directory.

5. ☐ Have your heartbeat display the position of the motor in encoder ticks to the serial port, say once every 500ms.

6. ☐ Use a free running timer (TIMER_0) to calculate the speed of the motor in RPM and display average speed over your report cycle with the end motor position. The Tiva Driver Library timer example (…/TiveDriver/examples/timers/timers.c) has example setups but don't generate interrupts, You're going to access a free running counter in your existing ISR to determine the time between encoder tics. The Driver Library API has another funciton, TimerValueGet() to query a timer value.

7. ☐ Determine an accurate way to discover the number of ¼ cycle tics your motor encoder delivers. Quantify and verify the error you expect in the ticks/rotation. This will directly map to an RPM error.

8. ☐ Add the RPM to your heartbeat output. display.. Find a way to calculate a reasonable average over the 500ms. What are the potential errors that your RPM method may encounter.