Mosaic Industries

Quadrature Encoder Detection

APPLICATION NOTE
MI-AN-051

Summary

The following explains the quadrature encoder detection.

Description

The QED Board can decipher the quadrature A and B signals from a quadrature shaft encoder. Typical applications are for monitoring and control of motor speed. A typical encoder produces two square waves of 50% duty cycle called A and B. When the shaft turns in the “clockwise” (CW) direction, the A waveform leads the B waveform by 90 degrees. When the shaft turns in the “counter clockwise” (CCW) direction, their order is reversed; the B waveform leads the A waveform by 90 degrees. The number of pulses of A and B that are produced per revolution, N, is determined by a codewheel internal to the encoder that is attached to the motor shaft. (Of course, the definition of CW and CCW depends on the physical configuration of the system; we'll adopt the stated convention of A leading B for CW rotation for simplicity in the following description).

From these two signals, A and B, the QED Board needs to extract both direction and speed information. At speeds of 1000 pulses per second or slower, software can be used to reliably extract this information using an Input Capture (IC) pin on the QED Board.

At speeds greater than 1000 pulses per second, the pulse accumulator input (PA7) on the QED Board should be used to calculate the speed. The pulse accumulator counts up to 256 pulses and then can automatically interrupt the processor. An interrupt service routine treats the pulse accumulator as the lower 8 bits of a count and increments a memory location for the most significant 8, 16, or 32 bits of a larger count. By keeping track of the time between interrupts (by reading the 68HC11’s free-running counter, TCNT, or using the timeslicers count, or even using the battery-backed real time clock), the average speed is calculated. In addition, a single D-type flip flop can be used to latch the direction: By applying the B input as a positive-edge clock input, and the A inputs the D (data) input to the flip flop, the Q output is high for theCW direction and low for the CCW direction.

The Software

At the relatively slower speeds (under 1000 pulses/sec), an input capture can be configured to interrupt the processor each time a rising edge is detected on the B encoder signal. The input captures are available on PortA pins PA0 (IC3), PA1 (IC2), PA2 (IC3), and PA3 (IC4/OC5) on the QED Board. Note that PA3 is not available if you are taking advantage of the secondary serial port. The encoder's A signal should be connected to another input (for example, on PortA or PortE) so it can be read as described below.

The QED Hardware Manual presents a detailed description of Input Captures along with Forth and Assembly language routines in the “Programmable Timer and Pulse Accumulator” Chapter in the sections titled “Configuring and Using Input Capture Functions”, “Input Capture Interrupts”, and “Input Signal Pulse Width Measurement”.

Configure the input capture to trigger an input on the rising edge of B. In the interrupt service routine, read the state of the A input. If it is high, the shaft is turning CW; if A is low, the shaft is turning CCW. This direction determination is a software implementation of the flip-flop circuit described above. The direction can be saved in a variable for use by a foreground program.

The interrupt service routine could also increment a 32-bit accumulator that tracks the total number of pulses (revolutions) counted to date. CW pulses could increment the accumulator, and CCW pulses could decrement it. This accumulator can be used by the foreground to calculate average speed (as elapsed counts divided by elapsed time between speed calculations).

The interrupt service routine should save the value in the TCNT register (this is the exact time at which the B pulse went high) in a variable in RAM. To calculate instantaneous speed, the service routine subtracts the prior saved TCNT count from the new TCNT count. This yields the elapsed time per pulse in units of 2 microseconds. Knowing the number of pulses per revolution, N, determined by the codewheel, the speed in revolutions per second can be calculated. Often it is advantageous for the interrupt routine to perform only the rudimentary subtraction, and leave it to the foreground program to perform unit conversions which require more time. This keeps the interrupt routines short and maximizes system performance.

Mosaic Industries

5437 Central Ave Suite 1, Newark, CA 94560 Telephone: (510) 790-8222 Fax: (510) 790-0925