

Auswertung eines Quadratur- Encoders mittels CLC

Allgemein:

Für diese Aufgabe braucht man 2 CLC- Module, eines liefert einen Interrupt bei Drehung im Uhrzeigersinn, das zweite in der Gegenrichtung.

Zur Erprobung habe ich wieder den PIC16F1713 genommen. Wichtig ist die Möglichkeit, die beiden Encodersignale A und B jeweils an beide CLC-Module anzukoppeln. Peripheral Pin Select (PPS) macht's möglich. (Der 16F1503 wäre hier z.B. ungeeignet, weil man als Eingang für CLC1 nur RA2 und RA5 definieren kann, für CLC2 aber nur RC3 und RC4. Das würde 4 Input-Pins belegen.)

Funktionsweise:

Bei einer Pegeländerung an Signal A wird dieser neue Pegel mit dem Pegel an Signal B verglichen. Je nach Drehrichtung liegt gleiche oder entgegengesetzte Polarität an. Das wird ausgewertet und führt zu einem Setzen des CLC1IF oder CLC2IF.

Im Programmbeispiel wird eine Hilfsvariable i je nach Drehrichtung hoch- oder heruntergezählt und an PORTB ausgegeben.

Hinweis: Fast alle Encoder haben am Rastpunkt einen Pegelwechsel an Signal B. Damit ist der Pegel an B undefiniert. Der Pegelwechsel an A erfolgt stets **zwischen** den Rastpunkten, wo der Pegel an B stabil ist. Deshalb können A und B nicht vertauscht werden!

Encoder:

Die Beschaltung des Encoders erfolgte gemäß Schaltung.jpg. Beim Schließen eines Kontaktes wird im Schaltmoment der auf 5V geladene 10nF-Kondensator über 10kOhm entladen. Dazu kommt der Strom von 0,5 mA Von +5V über 10kOhm. Der Encoder sollte also min. 1mA verkraften, was fast alle tun. Ausnahme ist z.B. der EC20A von ALPS, der nur bis 0,5mA spezifiziert ist.

Beachte: Es gibt Encoder mit identischer Anzahl von Impulsen und Rastungen und solche mit der doppelten Anzahl von Rastungen.

Anschluss:

Der Encoder wird mit Signal A an PORTC.0 angeschlossen, Signal B kommt an PORTC.1 Beide Signale gehen intern sowohl an CLC1 als auch an CLC2.

Meine Testumgebung: Der Test erfolgte auf einem Testboard mit der Möglichkeit, den Pegel aller uC-Pins anzuzeigen.

Daran orientiert sich auch das mikroBASIC- Beispielprogramm Encoder.mbas.

An PORTA wird das Ausgangssignal der beiden CLC-Module angezeigt: CLC1 -> PORTA.0, CLC2 -> PORTA.1. Die Zählvariable i wird mit 15 initialisiert und an PORTB angezeigt.

Programm:

Im Hauptprogramm werden ANSEL- und TRIS- Register gesetzt. Weiterhin werden notwendige Interrupts aktiviert.

Dann werden noch CLC1 und CLC2 konfiguriert und der uC in die Endlosschleife geschickt.

Die Encoder- Auswertung erfolgt in der Interrupt- Routine, also hier nur das Hoch- und runterzählen und Anzeigen von i.

Zum Interrupt: Im vorliegenden Beispiel wird nur bei der steigenden Flanke am Ausgang der CLCs das CLCxIF-Bit gesetzt. Wenn es so ist, dass nur bei jeder 2. Rastung weitergezählt wird (das hängt vom Encoder-Typ ab), kann man auch **zusätzlich** den Interrupt auf die fallende Flanke aktivieren (grün eingekreist auf den CLC-Bildern). Die Register CLC1CON und CLC2CON werden dabei von 0x95 auf 0x9D geändert.

Viel Spaß beim Ausprobieren!