

Drehgeber

Ziel

Einen Zähler mittels eines Drehgebers verstellen.

Verbinden Sie einen Drehgeber mit dem Arduino. Entwickeln Sie ein Programm, das einen Zähler mit jedem Schritt des Drehgebers herauf-, bzw. herunterzählt. Der aktuelle Zählerwert soll über die serielle Schnittstelle ausgegeben werden.

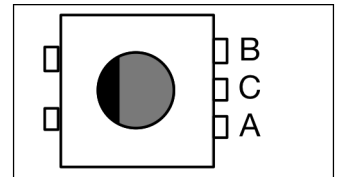


Abbildung 1) Signalleitungen

Vorgehensweise

Schließen Sie den Drehgeber so an den Arduino an, dass sowohl Signalleitung A, als auch Signalleitung B einen Interrupt auslösen können (siehe Abbildung 1). Ein Problem mit dem Drehgeber liegt darin, dass er prellt, d.h. zusätzlich zu dem Signalverlauf im Timing-Diagramm (siehe Abbildung 2) gibt es Störimpulse – diese lösen ebenfalls Interrupts aus. Um den Drehgeber korrekt auszulesen müssen Sie auf vier Ereignisse reagieren: Auf das Steigen und Fallen des Signals A, sowie auf das Steigen und Fallen des Signals B.

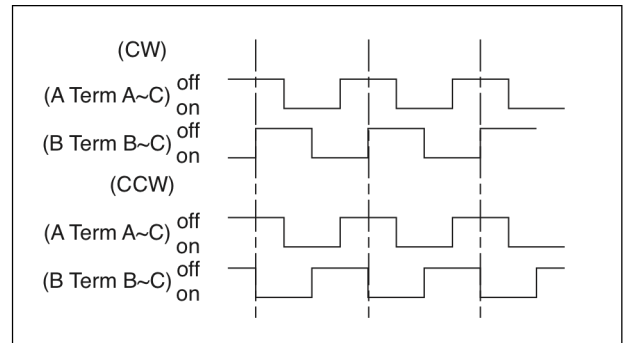


Abbildung 2) Timing Diagramm

Implementieren Sie für jedes Signal eine Interrupt-Routine, die auf einen Wechsel des Pegels das entsprechende Ereignis auslöst. Programmieren Sie dann eine Zustandsmaschine, die auf diese Ereignisse reagiert (siehe Abbildung 3). Den Zähler erhöhen Sie nach jedem vollen Durchlauf der Zustandsmaschine im Uhrzeigersinn, bzw. erniedrigen ihn nach einem vollen Durchlauf gegen den Uhrzeigersinn.

Achtung

- Kurze Interrupt-Routinen: kein `Serial`, kein `delay()`, keine Bibliotheksaufrufe.
- Benutzen Sie die internen Pull-Up Widerstände um einen definierten Pegel beim Auslesen des Drehgebers zu erreichen (invertiert das Signal)
- Bei der Übergabe von enums als Parameter muss die Signatur auf `int` getypt sein, der Wert muss per casting ermittelt werden.
- Es gibt nur zwei Interrupt Pins.

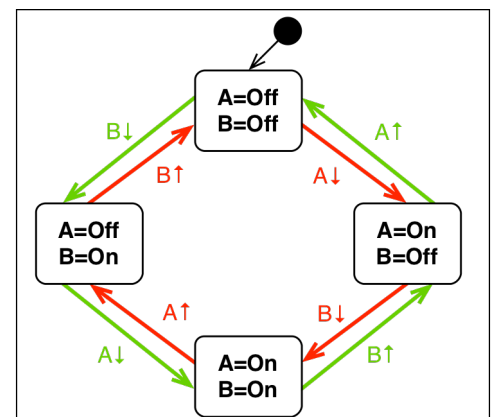


Abbildung 3) Zustandsmaschine

Vorbereitung

Machen Sie sich mit der Behandlung von Interrupts durch das Lesen der Arduino Dokumentation vertraut. Studieren Sie das Datenblatt für den Drehgeber. Beschäftigen Sie sich mit der Umsetzung von Zustandsmaschinen in C. Implementieren Sie eine Zustandsmaschine wie in Abbildung 3.

Wichtige Funktionen

- `attachInterrupt()`
- `digitalPinToInterrupt()`

Sie brauchen

- Arduino, USB Kabel, einen Drehgeber
- Das Datenblatt vom Drehgeber TW-700198

Notengebung

4,0 (Anwesend); 3,0 (Zählt, aber nicht vollkommen sauber); 2,0 (Korrekte, saubere Funktionsweise); 1,7 (+Code strukturiert und dokumentiert); 1,3 (+Kompakter, verständlicher Code für das Auslösen der Events); 1,0 (+Eleganter Code für die Zustandsmaschine)