



MIKROCONTROLLERTECHNIK

## Labor 5

Vorbereitungsaufgaben

*Autor:* Richard GRÜNERT

15.4.2020

# 1 Tasterprellen

Das Prellen der Taster ist mechanischen Ursprungs. Ursache ist das elastische Zurückprallen gegen die Federung (Wikipedia), wodurch der Kontakt oft mehrmals geöffnet und geschlossen wird bevor der eigentlich stabile Tasterzustand entsteht, was dazu führen kann, dass fehlerhaft mehrfache Anschläge registriert werden.

Hardwaremäßig kann man ein Tiefpassfilter einsetzen um die kurzen, ungewollten Impulse zu filtern. Nachfolgend wird dann ein Schmitt-Trigger zur Erzeugung der korrekten Signalwerte (High/Low) geschaltet.

In der Software kann man eine bestimmte Entprellzeit abwarten, bevor der Tasterzustand gelesen wird. Dort wird z.B. der Interrupt durch das erste Prellen ausgelöst, der Zustand registriert und mit dem Zustand nach der Entprellzeit verglichen.

Tasterprellzeiten liegen im Bereich 100  $\mu$ s bis 10 ms.

# 2 Codierung des Tastendrucks

Mithilfe einer integrierten Encoderschaltung, wie dem 74HC148, kann ein Tastendruck aus acht Tastern in einen 3-bit Code gewandelt werden. Jeder Taster erhält also einen eigenen Code, z.B. 101, über den er von den anderen unterscheidbar ist. Werden mehrere Taster gleichzeitig gedrückt, liegt u.U. eine Priorisierungslogik vor, die den Taster/Code mit der höheren Priorität einstellt und den/die mit der niedrigeren damit unterdrückt.

Der Group Select (GS) Pin am 74HC148 wird bei Eingangsänderung gesetzt und kann zur Interrupterzeugung verwendet werden.

### 3 Interruptkonfiguration

**Global Interrupt Enable (GIE) im SR muss aktiviert sein**

```
__bis_SR_register(GIE);
```

**Port Interrupt Enable (PxIE) muss aktiviert sein**

```
P1IE |= BIT4;
```

**Interrupt Edge Select (IES) wählen**

```
P1IES |= BIT4; // Interrupt bei High-Low-Flanke  
P1IES &= ~BIT4; // Interrupt bei Low-High-Flanke
```

**Interrupt Service Routine, Vektor zuweisen**

```
#pragma vector=PORT1_VECTOR  
__interrupt void port1_isr_name() {  
    // ISR  
}
```

### 4 Tastendruck einlesen

- Portinitialisierung, Interruptkonfiguration
- ISR: Ausgelöst durch GS
- 3-bit Wert aufnehmen

- Entprellzeit abwarten
- Entsprechend Zustand setzen, ob Taster gedrückt
- Weiterverarbeitung außerhalb der ISR abhängig vom Tasterzustand