

# Embedded Systems I

## Lab 5: Hausautomation Teil 2 - Interrupts



Beschreibung:

In dieser Übung werden Sie die Lichtschaltung aus der vorherigen Übung um Taster erweitern, welche zum einen einen Stromstoßschalter simulieren und zum anderen eine "Alles aus"-Funktion ermöglichen.

### Aufgaben:

1. Öffnen Sie das Projekt aus der vorherigen Übung (Lab 3: Hausautomation Teil 1 – Intelligente Lampensteuerung) und schließen Sie die LEDs entsprechend an.

Erweitern Sie das Projekt um eine weitere, fünfte LED, an Pin 8 des Arduinos an welche ein Treppenhauslicht mit Ein-/ Ausschalter simuliert. Schließen Sie hierzu einen Taster an Pin 2 des Arduino an.

- Der Taster soll einen Interrupt auslösen, welcher das Treppenlicht abwechselnd ein und wieder ausschaltet.
- Nutzen Sie für den Status des Treppenlichtes eine globale Variable, welche durch die Interrupt-Routine gesetzt wird.

### Hinweise:

Der Arduino Uno kann nur über Pin 2 und Pin 3 einen Interrupt auslösen. Den Taster schalten Sie diesmal direkt gegen Masse (0V), da intern bereits ein Pullup-Widerstand geschaltet wird (s.u.). Zur Verwendung des Interrupts an Pin 2 müssen Sie eine weitere Routine, die sogenannte **Interrupt Service Routine** programmieren. Verwenden Sie hierzu einfach eine weitere Funktion `void isr()`, welche Sie im Programmcode nach `void loop()` deklarieren.

Um den Interrupt an Pin 2 zu setzen, verwenden Sie folgende Einträge in der **setup()** Routine:

```
pinMode(2, INPUT_PULLUP); // hierdurch wird automatisch ein
                             Pullup-Widerstand geschaltet
attachInterrupt(0, isr, FALLING); // 0 ist der erste Interrupt an Pin 2,
                                   1 der zweite Interrupt an Pin 3
```

2. Erweitern Sie das Projekt um einen weiteren Taster, welcher bei Betätigung alle LEDs ausschaltet.
3. Zusatzaufgabe: Experimentieren Sie mit Interrupts, z.B.
  - Lösen Sie z.B. den Interrupt auf einer steigenden Flanke mit dem Schlüsselwort RISING aus (Achtung: Nun benötigen Sie einen Pull-Down Widerstand in Hardware).
  - Verwenden Sie den anderen PIN 3 (Interrupt 1) für den Interrupt.
  - Verwenden Sie delay() innerhalb der Interrupt Service Routine. Warum geht das nicht?