

---

## SPiCAufgabe #2: Interrupt Zähler

(12 Punkte, Mittwoch, 13.11.2013, 10:15 , keine Gruppen)

Schreiben sie das Programm `izaehter.c`, das die Tastendrucke an Taster 1 zählt (fallende Flanke). Wird der Taster 0 gedrückt, sollen die gezählten Tastendrücke angezeigt werden. Die Einerstelle wird auf der Siebensegmentanzeige und die Hunderterstelle mit den Leds angezeigt ( $\text{LED0} = 100$ ,  $\text{LED1} = 200$ , etc.).

- Die Anzeige soll bei einem Tastendruck auf Taster 0 aktualisiert werden.
- Um Energie zu sparen, soll der Mikrocontroller so oft wie möglich in den Ruhezustand versetzt werden.
- Des Weiteren soll die Anzeige nach 5 Sekunden ohne Tastendruck auf Taster 0 abgeschaltet werden.
- Sollte der Zähler den anzeigenbaren Wertebereich verlassen, wird er zurück auf 0 gesetzt.
- Dokumentieren Sie im Quellcode aus welchem Grund die Interrupts aktiviert bzw. deaktiviert werden können/müssen.

### Hinweise:

- Verwenden Sie zur Ansteuerung der Siebensegmentanzeige und der LEDs die `libspicboard`. Die Button-funktionen der libspicboard sind **nicht zulässig**.
- Um die Wahrscheinlichkeit, dass Tastendrücke verloren gehen, zu minimieren, dürfen die Interrupts während des Aufrufs der `sb_led-` und `sb_7seg`-Funktionen nicht gesperrt sein.
- Verwenden sie das Datenblatt um herauszufinden an welchen Interrupts die Taster (PD2, PD3) angeschlossen sind. Dort finden sie auch die nötigen Informationen um das MCUCR und GICR-Register richtig zu konfigurieren.
- Interrupts sollen, wie immer, möglichst kurz gehalten werden.