

Übungen zu Systemnahe Programmierung in C

Abschnitt WS.3: Aufgabe (izaehler)

16.11.2020

Tim Rheinfels
Benedict Herzog
Bernhard Heinloth

Lehrstuhl für Informatik 4
Friedrich-Alexander-Universität Erlangen-Nürnberg



Lehrstuhl für Verteilte Systeme
und Betriebssysteme



FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

TECHNISCHE FAKULTÄT



- Zählen der Tastendrucke an BUTTON0
- Vorübergehendes Aktivieren der Anzeige durch BUTTON1
 - Deaktivieren der Anzeige nach 1 - 10 Sekunden (einstellbar über das Potentiometer)
 - Darstellung über 7-Segmentanzeige und LEDs
 - Bei Verlassen des anzeigbaren Wertebereichs Zähler zurücksetzen
 - Aktive Anzeige bei Änderung des Zählerstandes aktualisieren
- Erkennung der Tastendrucke ohne Polling
 - Interrupts verwenden (fallende Flanke)
 - CPU in den Schlafmodus versetzen, wenn nichts zu tun ist
- Hinweise:
 - Erkennung der Tastendrucke **ohne** libspicboard
 - Ansteuerung des Zeitgebers **mittels** libspicboard
 - Interrupts nur kurzzeitig sperren und ISR kurz halten
 - Auf richtige Synchronisation achten



```
01 static void alarm_handler(void) {
02     alarm_event = 1;
03     alarm = NULL;
04 }
05
06 void main(void) {
07     sei();
08     alarm = sb_timer_setAlarm(alarm_handler, 1000, 0);
09
10     // [...]
11
12     cli();
13     if(alarm) {
14         sb_timer_cancelAlarm(alarm);
15     }
16     sei();
17 }
```

- Handler im Interrupt-Kontext (\leadsto gesperrte Interrupts)
- Single-Shot Alarme (`cycle = 0`) dürfen nur abgebrochen werden, **bevor** sie ausgelöst haben (Nebenläufigkeit!)