

Aufgabe 2: Geschicklichkeit (12 Punkte, Abgabe Di. 24.11.09 12:00 Uhr)

Programmieren Sie ein Geschicklichkeitsspiel in der Datei *gesch.c*. Ziel das Spiels ist es alle 8 LEDs zum Leuchten zu bringen. Hierzu wandert die LED immer hin und her (die LED wird invertiert). Wird auf den Taster 0 gedrückt, wird die aktuelle LED dauerhaft invertiert. Dies bedeutet auch, dass das eine bereits eingeschaltete LED wieder ausgeschaltet wird. Sind alle LEDs erfolgreich ausgewählt kommt die Sieges-Sequenz und das Spiel beginnt von neuem. Diesmal in einem höheren Level. Der aktuelle Level wird in der Sieben-Segmentanzeige angezeigt.

Die Siegessequenz:

- Die eingeschaltete LED wandert von links nach rechts und wieder zurück.
- Die Leds füllen sich von links und gehen wieder zurück.
- Die Leds füllen sich noch einmal. Diesmal leeren sie sich in die andere Richtung.

Das Programm soll in zwei Hauptteile unterteilt werden: Die Funktionen *play()* und *showwin()*.

Beachten Sie bei der Programmierung der Spiel-Funktion:

- Schreiben Sie eine Funktion *wait_key()* welche abhängig vom level eine bestimmte Zeit wartet. Beachten Sie hierbei, dass ein Tastendurck erkannt werden muss und die Wartezeit nicht von der Dauer des Tastendurcks abhängen darf. Achten Sie auch darauf, dass der Taster nicht schon vor dem Aufleuchten der LED gedrückt ist (gegebenenfalls darauf warten, dass der Taster losgelassen und wieder gedrückt wird).
- Schreiben Sie eine Funktion *show_led()*, welche als Parameter die bereits gesetzten LEDs, wie auch die aktuell ausgewählte LED übergeben bekommt.

Beachten Sie bei der Programmierung der Siegessequenz:

- Verwenden Sie Schleifen und Bitoperationen um die Muster zu ersellen.
- Finden Sie eine Möglichkeit die Geschwindigkeit flexibel anzupassen (z.B. durch ändern des Programmcodes). Verwenden Sie hierzu keine Variable.

Sonstiges:

- Verzichten sie auf globale oder statische Variablen.
- Verwenden Sie nur die Funktion *sb_led_set_all_leds()* um die Leds anzusteueren.
- Im Verzeichnis *pub/aufgabe2* befindet sich die Datei *gesch.hex* welche eine Beispielimplementierung enthält.