
GSPiC-Aufgabe #4: LED

(14 Punkte, Abgabe bis Freitag, 29.06.2012, um 14:00, in Zweier-Gruppen)

Reimplementieren Sie das LED-Modul `led.c` der `libspicboard`. Schreiben Sie außerdem ein Testprogramm `test.c`, welches alle Funktionen des Moduls testet. Die Beschreibung der zu implementierenden Schnittstelle finden Sie in der API-Dokumentation auf der GSPiC-Webseite unter:

http://www4.cs.fau.de/Lehre/SS12/V_GSPIC/Uebungen/doc/group_LED.shtml

LED-Modul

- Halten Sie sich genau an die vorgegebene Schnittstelle. Binden Sie hierzu die `led.h` (verfügbar in `/proj/i4gspic/pub/i4/led.h`, bzw. unter Laufwerk Q:) aus der `libspicboard` in Ihre `led.c` ein.
- Weitere Funktionen oder globale Variablen, die in diesem Header nicht deklariert werden, sind in ihrer Sichtbarkeit auf das Modul zu beschränken.
- Verzichten Sie zum Ansteuern der Pins auf `if`-Kaskaden und `switch`-Anweisungen. Verwenden Sie stattdessen Arrays (und ggf. Zeiger), um Port- und Pinnummer zu bestimmen.
- Beachten Sie, dass die I/O-Ports des ATmega-Mikrocontrollers vor der ersten Verwendung entsprechend initialisiert werden müssen. Dieses Detail soll jedoch (wie im Original) innerhalb des LED-Moduls verborgen bleiben.

Testprogramm

- Ihr Testprogramm soll alle Funktionen testen. Vergessen Sie hierbei nicht, auch die Fehlerfälle zu überprüfen.
- Überprüfen Sie Ihr Testprogramm auch mit der Referenzimplementierung aus der `libspicboard` und vergleichen Sie das Verhalten.
- Für Ihr Testprogramm können Sie auch die anderen Module der `libspicboard` verwenden. Sie können z. B. das ADC-Modul verwenden, um den Wert des Photosensors auszulesen und mit der LED-Pegelanzeige den Helligkeitspegel im Raum anzuzeigen.

Hinweise

Der Linker verwendet stets die „lokalen“ Funktionen, die in den lokalen Quelldateien definiert sind. Nur wenn eine Funktion dort nicht definiert ist, werden die Bibliotheken durchsucht. Um testweise statt der eigenen LED-Funktionen in der Datei `led.c` die LED-Funktionen der `libspicboard` zu verwenden, können Sie einfach ihre eigene Implementierung auskommentieren, z. B. durch eine Zeile

```
#if 0
```

am Anfang der Datei und eine Zeile

```
#endif
```

am Ende der Datei. Aktivieren können Sie Ihre Implementierung dann wieder, indem Sie die `0` in eine `1` abändern.

Ihr Programm muss mit der Build-Compiler-Konfiguration kompilieren und funktionieren; diese Konfiguration wird zur Bewertung herangezogen.