
Allgemeine Hinweise zu den GSPiC-Übungen

- Die Aufgaben sind alleine zu bearbeiten.
- Jeder Benutzer erhält für SPiC ein spezielles Projektverzeichnis mit dem Namen `/proj/i4gspic/LOGIN/`, wobei `LOGIN` für den eigenen Login-Namen steht. Die Projektverzeichnisse werden für alle Teilnehmer angelegt, die sich im Waffel angemeldet haben. Eine Anmeldung im Waffel ist daher zwingend zur Übungsteilnahme erforderlich!
- Der Verzeichnisbaum für die Aufgaben ist folgendermaßen aufzubauen: `/proj/i4gspic/LOGIN/aufgabe1`, `/proj/i4gspic/LOGIN/aufgabe2`, usw.
- Die Aufgaben sind bis spätestens zum Abgabetermin durch Aufruf des Programms `/proj/i4gspic/pub/abgabe aufgabeX`, wobei $X = 1 \dots n$, abzugeben. Dieses Programm überprüft die Verzeichnisstruktur und die Namen der Dateien, die nach der Aufgabenstellung vorhanden sein müssen, und erzeugt dann ein Archiv der abgegebenen Dateien. Bis zum Abgabetermin kann ein Programm beliebig oft abgegeben werden – es gilt der letzte, vor dem Abgabetermin vorgenommene Aufruf des Abgabeprogramms.
- Sie können Aufgaben auch nach dem Abgabetermin abgeben, indem Sie das Abgabeskript mit dem Parameter `-force` aufrufen:
`/proj/i4gspic/pub/abgabe aufgabeX -force X =0 ... n`
In diesem Fall ist jedoch eine Rücksprache mit Ihrem Übungsleiter erforderlich, der dann entscheiden kann, ob die verspätete Abgabe noch gewertet wird. Eine frühere, fristgerechte Abgabe wird durch eine verspätete Abgabe **nicht** überschrieben und im Zweifelsfall gewertet.
- Verwenden Sie für den Namen der C-Quelldatei, soweit in der Aufgabenstellung nicht anders angegeben, den Namen des Programms entsprechend dem Titel der jeweiligen Aufgabenstellung. Ist der Titel der Aufgabenstellung also z. B. *blink*, so legen Sie den Quellcode in einer Datei `blink.c` ab.

GSPiCAufgabe #1: wuerfel

(6 Punkte, Dienstag, 09.11.2010, um 22:00, keine Gruppen)

Programmieren Sie zum Kennenlernen der `libspicboard` einen Würfel in der Datei `wuerfel.c`. Das Programm soll auf der 7-Segmentanzeige beim Drücken des Taster 0 eine Zahl zwischen 1 und 6 ausgeben. Um eine zufällige Zahl zu generieren soll eine Variable in einer Schleife hochgezählt werden. Sobald der Taster gedrückt wird, wird ein neues Würfelergbnis ausgegeben. Während der Taster gedrückt ist, muss der Zähler bereits zur Generierung des nächsten Ergebnisses weiterzählen. Die Anzeige soll sich, während der Taster gedrückt ist nicht mehr ändern. Zum Abfragen des Tasters soll ausschließlich die Funktion `sb.button.getState()` verwendet werden. Achten Sie darauf, dass zur Verwendung der 7-Segment Anzeige die Interrupts mit durch den einmaligen Aufruf von `sei()` aktiviert werden müssen.