
SPiC-Aufgabe #4: Ampel

(15 Punkte, Montag, 24.11.2014, 10:00, keine Gruppen)

Entwerfen Sie eine Steuerung für eine Bedarfsampel an einem Fußgängerüberweg in einer Datei `ampel.c`. Hierbei wird die den Autos zugewandte Ampel durch die LEDs `REDO`, `YELLOWO` und `GREENO` dargestellt, die Fußgängerampel durch die LEDs `RED1` und `GREEN1` (kein gelbes Licht). Durch das Drücken von `BUTTON0` können Fußgänger eine Umschaltung anfordern. Die LED `BLUE1` signalisiert den Fußgängern, dass eine Umschaltanforderung entgegengenommen wurde.

Die Steuerung soll im Detail wie folgt arbeiten:

- Die Grünphase für die Autos dauert mindestens 10 s.
- Eine Umschaltanforderung wird durch Druck auf `BUTTON0` ausgelöst. Der Druck der Taste wird durch Aktivierung der LED `BLUE1` (Signal kommt) bestätigt. Diese LED wird beim Start der Umschaltung wieder deaktiviert. Tastendrucke werden bis zum Ende der Grünphase für Fußgänger ignoriert.
- Die Ampel schaltet genau 5 s nach dem Tasterdruck, frühestens jedoch nach Ablauf der minimalen Grünphase für die Autos, um.
- Bei der Umschaltung wird die Autoampel zunächst auf Gelb (500 ms), dann Rot geschaltet. Nach einer Verzögerung von 1.5 s schaltet die Fußgängerampel dann auf Grün.
- Die Grünphase der Fußgängerampel soll 5 s andauern. Die Fußgänger haben anschließend 2 s Zeit um die Fahrbahn zu räumen (beide Rot). Anschließend wechselt die Autofahrerampel auf Gelb-Rot (1 s) und wieder auf Grün.
- Auf der Siebensegmentanzeige soll die verbleibende Rot-Dauer für Fußgänger in Sekunden angezeigt werden (0 entspricht der Umschaltung auf Grün). Wird die Siebensegmentanzeige nicht mehr benötigt (0), soll diese abgeschaltet werden.
- In Ruhephasen soll der Mikrocontroller in einen Sleep-Modus wechseln (kein aktives Warten). Der Taster `BUTTON0` soll von Ihnen so konfiguriert werden, dass ein externer Interrupt den Mikrocontroller aus dem Sleep-Modus aufwecken kann.
- Verwenden Sie die Module `LED` und `7seg` der `libspicboard`, sowie die Funktionen `sb_timer_setAlarm` und `sb_timer_cancelAlarm` des `Timer`-Moduls (nicht `sb_timer_delay`). Die Verwendung des `Button`-Moduls ist nicht zulässig.

Hinweise:

- Das `7seg`-Modul verwendet das `Alarm`-Modul, welches Interrupts verwendet.
- Die Callbackfunktion des `Timer` modules wird aus dem Interrupt-Kontext aufgerufen.
- Unnötigen Berechnungen nach dem Aufwachen aus dem Sleep-Modus sollen vermieden werden.
- Achten Sie auf die korrekte Verwendung des `volatile`-Schlüsselworts. Beschreiben Sie in einem Kommentar zu jeder verwendeten `volatile`-Variable, weshalb Sie dieses Schlüsselwort dort benötigen.
- Ihr Programm muss mit der `Build`-Compiler-Konfiguration kompilieren und funktionieren; diese Konfiguration wird zur Bewertung herangezogen.
- Im Verzeichnis `/proj/i4spic/pub/aufgabe4/` unter Linux bzw. in `Q:\aufgabe4\` unter Windows befindet sich die Datei `ampel.hex`, welche eine Beispielimplementierung enthält.