
Klausur

Prüfungsfach: Embedded Systems 2
Datum/Uhrzeit: 26. Januar 2017 / 14:30 Uhr
Raum: J3.19
Prüfer: Dr. Hubert Högl
Dauer: 60 Minuten
Hilfsmittel: keine

Hinweise:

1. Diese Angabenblatt hat auch eine **Rückseite**. Bitte sofort überprüfen.
2. **Sie dürfen dieses Angabenblatt behalten**. Ihre Lösungen schreiben Sie bitte auf die separat ausgeteilten karierten Bögen.
3. Schreiben Sie bitte nicht mit Bleistift und nicht mit roter Farbe.

Viel Glück!

Aufgabe 1 (4 Punkte)

Nennen Sie **zwei konkrete eingebettete Systeme** die Sie aus dem Alltag kennen und beschreiben Sie in Stichpunkten deren **Zweck** und wie die **Technik** darin funktionieren könnte.

Aufgabe 2 (10 Punkte)

Vergleichen Sie die **Programmierung auf einem PC und auf einem Embedded System (ES)**. Stellen Sie für die folgenden zehn Stichpunkte jeweils PC und ES gegenüber:
(a) CPU (b) Hauptspeicher (c) Festplatte (d) C Compiler (e) Debuggen (f) printf (g) malloc/free (h) C-Bibliothek (i) Betriebssystem (j) GPIO

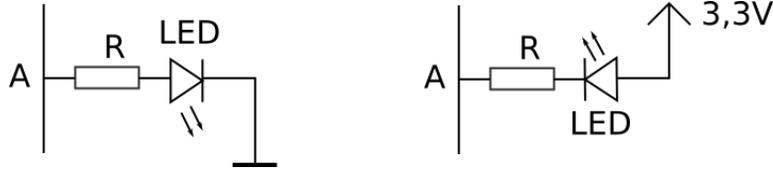
Aufgabe 3 (4 Punkte)

Warum ist die **Embedded Programmierung schwieriger** als die Programmierung eines Desktop PC? Nennen Sie **vier Gründe**.

Aufgabe 4 (8 Punkte)

Einige einfache Fragen zur **Elektronik**:

- (a) Was ist beim Anschluss eines Tasters an einen Eingang zu beachten (bitte mit Schaltungsskizze)?
- (b) Die folgende Abbildung zeigt zwei Möglichkeiten, um eine Leuchtdiode an einen Ausgang A anzuschliessen:



Vergleichen Sie die beiden Möglichkeiten und geben Sie in etwa den Wert des Vorwiderstandes R bei 3,3V Versorgungsspannung an.

(c) Wie steuert man eine Siebensegmentanzeige an, so dass man möglichst wenige Ausgänge benötigt? (Stichwort „multiplexen“).

Punkte: (a) 2, (b) 4, (c) 2

Aufgabe 5 (8 Punkte)

Fragen zur Toolchain

- (a) Zeichnen Sie ein Diagramm, das die einzelnen Schritte beim Kompilieren eines Programmes mit dem GNU Compiler wiedergibt. Denken Sie auch an mögliche Konfigurationsdateien.
- (b) Was steht in der (optionalen) Datei `syscalls.c` ?
- (c) Was bedeutet der Begriff **semihosting**?

Punkte: (a) 4, (b) 2, (c) 2

Aufgabe 6 (13 Punkte)

Zwischen dem **Reset** und der Ausführung der `main()` Funktion wird der Mikrocontroller initialisiert. Der Quelltext ist in den Dateien `startupcode.c` und `startupcode.h`.

- (a) Welche Aufgaben werden im Startupcode nacheinander erledigt?
- (b) Was ist die **data section** eines Programmes und wie muss diese initialisiert werden?
- (c) Was ist die **bss section** eines Programmes und wie muss diese initialisiert werden?
- (d) Wozu wird folgendes **Makro** verwendet?

```
#define WEAK_ALIAS(a) __attribute__((weak, alias(#a)))
```

- (e) Wie steuert man, dass bestimmte Teile eines Programmes, z.B. die Interruptvektoren, an eine **vordefinierte Adresse** im Speicher kommen?
- (f) Was passiert nach der **Rückkehr** von `main()`?

Punkte: (a) 4, (b) 2, (c) 2, (d) 2, (e) 2, (f) 1

Aufgabe 7 (6 Punkte)

Der folgende **sysTick Interrupt Handler** hat nur eine Zeile:

```
void SYSTICK_Handler() {  
    GPIOC_ODR ^= (LED_GR | LED_BL);  
}
```

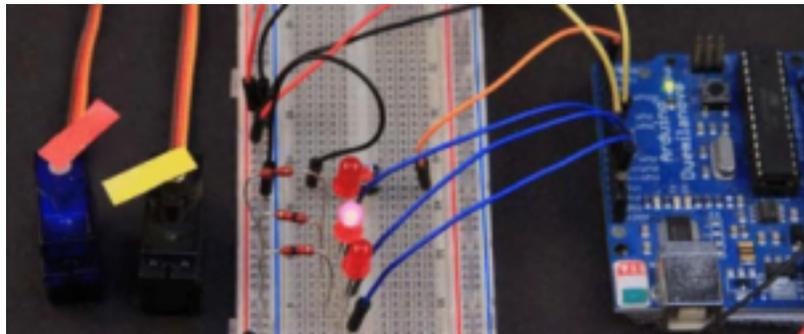
(a) Was macht diese Zeile?

(b) Sie möchten das Programm debuggen, merken aber, dass nach einem Breakpoint der sysTick aufhört. Was müssen Sie tun, damit der sysTick Interrupt unter der Kontrolle des Debuggers weiterläuft?

Punkte: (a) 2, (b) 4

Aufgabe 8 (8 Punkte)

Sie erinnern sich an das **Pseudo-Multitasking** auf dem Arduino, die folgende Abbildung zeigt zwei Servomotoren und drei LEDs, die unabhängig voneinander gesteuert werden.



(a) Beschreiben Sie die **Grundidee** hinter dieser Art der Programmierung ohne RTOS.

(b) Wie würde man die Programmierung mit Hilfe eines **Echtzeitbetriebssystems**, z.B. FreeRTOS lösen?

Punkte: (a) 4, (b) 4

Aufgabe 9 (6 Punkte)

Beschreiben Sie kurz, was die folgenden Begriffe bedeuten:

(a) CMSIS

(b) STM32F1 Cube

(c) Board Support Package (BSP)

Aufgabe 10 (16 Punkte)

Füllen Sie folgende Tabelle zu verschiedenen **Speicher-Technologien** auf Ihrem Arbeitsblatt aus.

	Funktionsprinzip	Flüchtig ja/nein	Typ. Grösse	Wo verwendet?
Statisches RAM (SRAM)				
Dynamisches RAM (DRAM)				
NOR Flash				
NAND Flash				

Aufgabe 11 (6 Punkte)

Fragen zum Artikel von R. Keil, F. Grobe, **Designing energy efficient systems with Cortex-M Microcontrollers, 2009**.

- Welche **Betriebsarten zum Stromsparen** hat ein moderner Mikrocontroller? (gerne in Form einer Tabelle)
- Mit welchem **Trick bei der Programmierung** kann man ein Gerät mehrere Jahre aus einer kleinen Batterie betreiben? Zeichnen Sie die Leistungsaufnahme über der Zeit hin.
- Ist es besser, zum Stromsparen **einen langsamen oder einen schnellen** Mikrocontroller zu nehmen? (Achtung: Die Stromaufnahme erhöht sich proportional mit der Taktfrequenz!)