
Klausur

Prüfungsfach: Embedded Systems 2
Datum/Uhrzeit: 29. Januar 2015 / 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)

Warum ist die **Embedded Programmierung schwieriger** als die Programmierung eines Desktop PC?

Aufgabe 2 (4 Punkte)

Wie sieht ein **Arbeitsplatz** für die Entwicklung von Programmen für Embedded Systems aus? Geben Sie in Stichpunkten die **nötigen Geräte** und auch die zu **installierenden Programme** auf dem Entwicklungsrechner an.

Aufgabe 3 (2 Punkte)

In der Datei `startupcode.h` befindet sich in der Zeile 14 das Makro:

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

Welche Aufgabe hat das Makro?

Aufgabe 4 (4 Punkte)

Welche Aufgaben haben im **Linkerscript** die Abschnitte **MEMORY** {...} und **SECTIONS** {...}?

Aufgabe 5 (4 Punkte)

Zeichnen Sie ein **Blockdiagramm** des **OpenOCD Debuggers**, das alle wesentlichen **Hard- und Software-Komponenten** umfasst.

Aufgabe 6 (5 Punkte)

Ordnen Sie die **Peripheriegeräte** den **I/O Schnittstellen** zu:

	GPIO	UART	SPI	I2C	PWM	AD	DA
Taster							
Drehgeber							
WII Nunchuck							
Relais							
Photowiderstand							
Gleichstrommotor							
Servomotor							
LED							
Temperatursensor (analog)							
7-Segment Anzeige							
Potentiometer							
Analoger Signalgenerator							

Aufgabe 7 (4 Punkte)

Was versteht man unter dem **SysTick** und wie nutzt man ihn in einem Programm?

Aufgabe 8 (4 Punkte)

Beschreiben Sie knapp die **vier Kategorien** (Leistungsklassen) von eingebetteten Systemen, die wir in der Vorlesung unterschieden haben. Nennen Sie jeweils einen konkreten Mikrocontroller.

Aufgabe 9 (4 Punkte)

Klassifizieren Sie allgemein die heute verfügbaren **Halbleiterspeicher** in einem **baumförmigen Diagramm**.

Aufgabe 10 (8 Punkte)

Fragen aus dem Buch von Elecia White, Kap. 5.1 und 5.2: **Managing the Flow of Activity**

- a) Wie kann man mehrere Aufgaben **gleichzeitig** auf einem Mikrocontroller erledigen?
- b) Wie kann man zwischen **Tasks kommunizieren** und welche **Probleme** können dabei auftreten?
- c) Was ist **Priority Inversion**?
- d) Welches **Entwurfsmuster** wendet man bei Embedded Systems häufig an? Geben Sie ein praktisches Beispiel an!

Aufgabe 11 (6 Punkte)

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

- a) Welche **Betriebsarten zum Stromsparen** hat ein moderner Mikrocontroller? (gerne in Form einer Tabelle)
- b) 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.
- c) Ist es besser, zum Stromsparen **einen langsamen oder einen schnellen** Mikrocontroller zu nehmen? (Achtung: Die Stromaufnahme erhöht sich proportional mit der Taktfrequenz!)

Aufgabe 12 (6 Punkte)

Bei der Programmierung von eingebetteten Systemen haben Sie die Wahl zwischen (a) **bare metal**, (b) **Echtzeit Betriebssystem** (RTOS), und (c) **grossem Betriebssystem** (Linux/Windows). Welches sind die **Vor- und Nachteile** dieser drei Möglichkeiten?

Ende der Klausur
