

---

## Klausur

Prüfungsfach: Embedded Systems 2  
Datum/Uhrzeit: 1. Februar 2018 / 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 (5 Punkte)

Beschreiben Sie, wie Sie im Praktikum die einzelnen Schritte **Editieren**, **Kompilieren**, **Flashen**, **Laden**, **Debuggen** erledigt haben.

### Aufgabe 2 (6 Punkte)

Welche Register hat der Cortex-M4 Prozessor? Ordnen Sie diese in die Gruppen **Vielzweck Register**, **Programmablauf Register** und **Spezielle Register**.

### Aufgabe 3 (9 Punkte)

Hier ist der Quelltext der `init()` Funktion:

```
1 void init()
2 {
3     uint8_t *src;    uint8_t *dst;
4     src = (uint8_t *)&_sidata;
5     dst = (uint8_t *)&_sdata;
6
7     while (dst < (uint8_t *)&_edata) {
8         *dst++ = *src++;
9     }
10    dst = (uint8_t *)&_sbss;
11    while (dst < (uint8_t *)&_ebss) {
```

```

12         *dst++ = 0;
13     }
14     main();
15 }

```

- (a) (4 Punkte) Was bedeuten die Symbole `_sidata`, `_sdata`, `_edata`, `_ebss`?
- (b) (1 Punkt) Wo sind die Symbole aus (a) definiert?
- (c) (4 Punkte) Was wird in den zwei `while`-Schleifen in `init()` gemacht? Bitte keine trivialen Angaben, sondern das übergeordnete Ziel erklären.

#### Aufgabe 4 (12 Punkte)

Übertragen Sie die folgende Tabelle auf Ihr Arbeitsblatt. Machen Sie für jeden der 12 Funktionsblöcke ein Kreuz in der Spalte, in der Sie den Block vermuten:

	CORE	CORE-PERIPH.	MCU
CPU			
NVIC			
SysTick			
SCB			
FPU			
SWD			
RCC			
EXTI			
SYSCFG			
PWR			
GPIO			
UART			

#### Aufgabe 5 (6 Punkte)

Welche Funktionsblöcke des STM32L476 spielen eine Rolle, wenn man externe GPIO Interrupts verwenden möchte? Versuchen Sie, das Tafelbild zu dem Thema wiederzugeben.

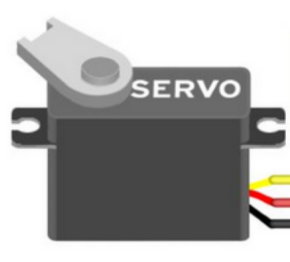
#### Aufgabe 6 (6 Punkte)

Sie können bei der Programmierung des STM32 Controllers zusätzliche Bibliotheken verwenden, damit Sie auf ein höheres Abstraktionsniveau und somit schneller zum Ziel kommen. Welche dieser Bibliotheken haben wir im Praktikum verwendet?

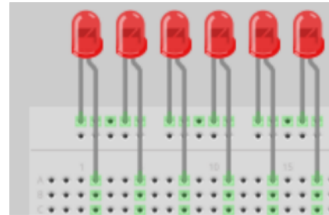
Nennen Sie **drei Namen** und geben Sie eine kurze **Beschreibung**, worum es sich jeweils handelt.

### Aufgabe 7 (6 Punkte)

Sie möchten ein Lauflicht und einen Servo-Motor quasi-gleichzeitig ansteuern, haben aber kein RTOS auf Ihrem Mikrocontroller. Das Lauflicht soll langsam im Sekundentakt von LED zu LED weiterschalten, das Servo soll sich „hektisch“ in Sekundenbruchteilen irgendwie hin- und herbewegen.



Servomotor



Lauflicht

Ihr erster Versuch besteht erst mal nur aus der Mainloop:

```
int main() {
    while (1) {
        servo();
        lauflicht();
    }
}
```

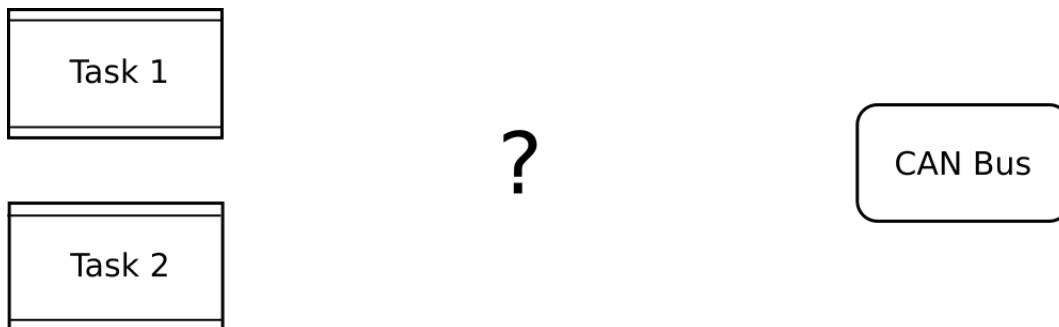
Sie erkennen, wenn Sie in `lauflicht()` eine Sekunde warten, dann „verhungert“ der Servomotor.

(a) (4 Punkte) Wie können Sie das Ganze zum Laufen bringen? Sie dürfen auch Interrupt Service Routinen zu Hilfe nehmen.

(b) (2 Punkte) Nennen Sie mindestens einen Nachteil, den diese Lösung trotzdem hat, im Vergleich mit einer guten Lösung mit RTOS.

### Aufgabe 8 (5 Punkte)

Zwei Tasks möchten gleichzeitig auf eine CAN Bus Schnittstelle zugreifen. Wie löst man das Problem mit FreeRTOS, so dass keine Nachrichten durcheinander kommen?



**Aufgabe 9** (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!)

---

Ende der Klausur

---