



**h\_da**

HOCHSCHULE DARMSTADT  
UNIVERSITY OF APPLIED SCIENCES

**fbi**  
FACHBEREICH INFORMATIK

MIKROPROZESSORPRAKTIKUM

SS2023

**Termin 2**

C-Programmierung für eingebettete Systeme Pointer, Peripherie, PIO,  
Interrupt

Name, Vorname	Matrikelnummer	Anmerkungen
Datum	Raster (z.B. Mi3x)	Testat/Datum

Legende: V:Vorbereitung, D: Durchführung, P: Protokoll/Dokumentation, T: Testat

## Lernziele:

Der Umgang mit digitalen Ein- und Ausgängen und Unterbrechungen (Interrupt).

## Arbeitsverzeichnis:

Kopieren Sie sich aus dem Ordner [user.v.shell.fbi.h-da.de/share/LabDisk/MI/](https://user.v.shell.fbi.h-da.de/share/LabDisk/MI/) den Ordner mpsDUESS2023. Dort finden Sie zu jedem Termin vorgegebene Dateien.

## Vorgehensweise:

Speichern Sie zu jeder gestellten Aufgabe Ihre Lösung unter Termin2AufgabeX.c (X steht für Nummer der Aufgabe). In den angebotenen offenen Laboren bekommen Sie die Möglichkeit Ihre Lösungen praktisch vorzuführen und zu testen und/oder mit den Lösungen Ihrer Kommilitonen zu vergleichen/zu diskutieren.

Zum zugehörigen Praktikumstermin sollen Sie Ihre Lösungen präsentieren können.

## Aufgabe 1:

Wir werden uns nach Termin 1 nun nochmals mit den Registern der PIOB des hier eingesetzten Mikrocontrollerboard Arduino DUE beschäftigen. Durch die strukturierte Anordnung der vielen verschiedenen Register der Peripherie des Mikrocontroller ist es möglich durch angelegte Strukturen sich die Programmierung und den Umgang mit der Peripherie zu vereinfachen. Schauen Sie sich Termin2Aufgabe1.c an. Testen und beschreiben Sie die Funktion des Programms. Hierzu schauen Sie sich die Dokumentation zum Board Arduino DUE und die Doku des eingesetzten Mikrocontrollers Atmel SAM3X8E an.

Mit welchen Anweisungen wird die gelbe LED „L“ ein- bzw. ausgeschaltet und warum?

..

Auf welchen Speicherstellen und/oder in welchen Registern können Sie erkennen ob die gelbe LED „L“ ein- oder ausgeschaltet ist?

..

## Aufgabe 2:

Gut, wir können nun die vorhandene gelbe Leuchtdiode „L“ auf dem Board kontrollieren. Ändern und erweitern Sie das Programm so, dass die gelbe LED „L“ leuchtet, wenn die Taste 1 (angeschlossen an PIOD0) betätigt wird. Speichern Sie Ihre Lösung unter Termin2Aufgabe2.c. Sollte die Taste nicht funktionieren, so überprüfen Sie ob im Power Management Controller (PMC) der Clock für PIOD eingeschaltet ist.

In welchen Registern müssen welche Bit gesetzt sein/werden, damit der Zustand der Taste über das Pin Data Status Register (PDSR) erfasst werden kann?

..

Auf welchen Speicherstellen und/oder in welchen Registern können Sie erkennen ob die Taste gedrückt ist?

..

## Aufgabe 3:

Schreiben Sie ein Programm, welches die gelbe LED „L“ mit ca. 0,5Hz (einfache Zeitschleife) blinken lässt.

Mit welcher Taktfrequenz läuft der Prozessor?

Untersuchen Sie die Funktion ihres Programms auch mit verschiedenen Optimierungsstufen.

## Aufgabe 4:

Kombinieren Sie die Programme aus Aufgabe 2 und 3. Schalten Sie bei jedem Tastendruck die LED „L“ aus.

Was erwarten Sie, wie der Tastendruck reagiert?

..

Untersuchen Sie die Funktion ihres Programms auch mit verschiedenen Optimierungsstufen.

..

### **Aufgabe 5:**

Schreiben Sie für die Taste 1 eine passende Interruptserviceroutine. Erklären Sie die Wechsel der ARM-Betriebsmodi. Die Taste welche Low-Aktiv ist soll beim loslassen die Aktion auslösen.

Wie reagiert nun der Tastendruck?

..

Hat das Bedienen der Taste einen Einfluss auf die Blinkfrequenz.

..

Sollten Sie in der Vorlesung noch kein Interrupt behandelt haben, kann diese Aufgabe auch nach Termin3 verschoben werden.

### **Aufgabe 6:**

Erstellen Sie zu diesem Termin ein Protokoll mit den Lösungen zu den Aufgaben und Ihren Erkenntnissen. Das Protokoll sollen Sie zum nächsten Termin vorlegen können. Denken Sie daran, dass einige Programmteile nochmals benötigt werden könnten.

Beschäftigen Sie sich auch schon mit den möglichen Aufgabenstellungen von Termin6 (Ausschankstation, Münzzähler, Pendel, oder ..).

Programmgerüstbeispiele:

```
// Loesung zu Termin 2
// Aufgabe 1
// Namen: _____; _____
// Matr.: _____; _____
// vom : _____
//
```