



**h\_da**

HOCHSCHULE DARMSTADT  
UNIVERSITY OF APPLIED SCIENCES

**fbi**  
FACHBEREICH INFORMATIK

**PRAKTIKUM EINGEBETTETE SYSTEME**

**SS2024**

**Termin 4**

C-Programmierung für eingebettete Systeme, Timer-Counter-Baustein,  
Wave-Modus

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

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

## Lernziele:

Im Termin 4 befassen wir uns mit der Motoransteuerung des Fahrzeugs. An jedem Arbeitsplatz der Versuchsaufbauten befindet sich ein Motortreiber mit einem entsprechenden Motor. Im Fahrzeug ist derselbe Motortreiber mit einem ähnlichen Motor verbaut. Es ist wichtig, bereits jetzt das Ansprechverhalten des Motors im Fahrzeug vom festen Versuchsaufbau unterscheiden zu können. Die Geschwindigkeit des Motors wird durch den Duty Cycle eines PWM-Signals gesteuert. Die Fahrtrichtung kann über die PIO-Ausgänge (PIOC23 und PIOC24) gesteuert werden, die durch entsprechende Pegel vorwärts oder rückwärts fahren ermöglichen.

## Aufgabe 1

Fügen Sie `drive_init` und `drive_loop` in `init` und `loop` ein.

## Aufgabe 2

Initialisieren Sie in der `drive_init` Funktion die an den Motortreiber angeschlossenen Pins (IN1, IN2 und ENA) mit den entsprechenden Peripheriekomponenten. Verwenden Sie hierfür den Timerbaustein im Wave-Modus. Das PWM-Signal ist an PIOC25 auszugeben.

## Aufgabe 3

Implementieren Sie die Steuerung des Motors in `drive_loop` abhängig von `duty_cycle` und `direction` aus `state.h`.

## Leitfragen

- Wie könnte sich das Ansprechverhalten des Motors zwischen dem Fahrzeug und dem festen Versuchsaufbau unterscheiden?
- Welche Möglichkeiten gibt es, den Treiberbaustein des Motors anzusprechen?