

Fachgruppe Technische  
Informatik

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



## PRAKTIKUM EINGEBETTETE SYSTEME

SS2024

Termin 3

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

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

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

## Lernziele:

Im dritten Termin beschäftigen wir uns mit der Distanzmessung mittels eines Ultraschallsensors. Hierbei nutzen wir einen Timer-Counter-Baustein TIOA8 (Echo Pin) des Mikrocontrollers im Capture-Modus.

## Aufgabe 1

Initialisieren Sie die verwendeten Timerbausteine TIOB7 (Trigger) und TIOA8 (Echo) durch den Aufruf der Funktion `sonic_init`.

## Aufgabe 2

Implementieren Sie das Auslesen der eigentlichen Messung innerhalb der Funktion `sonic_loop`. Verwenden Sie dafür die Register RA und RB, sowie das Statusregister (SR). Eine Messung kann über das CCR gestartet werden. (Verwenden Sie Busy-Waiting oder Polling.)

## Aufgabe 3

Implementieren Sie nun die gleiche Messung mithilfe von Interrupts. Implementieren Sie dazu die Prozedur `TC8_Handler` zur Behandlung des Interrupts. Den Interrupt können Sie mit `NVIC_Enable` aktivieren.

## Aufgabe 4

Zeigen sie die Distanz zum Hindernis auf der LED Matrix an, wenn eine Abstandsmessung vorliegt und dies durch den Status `state.sonic` (`SONIC_ON`) innerhalb der `matrix_loop` Funktion gewählt ist. Verwenden sie dafür `sample_series_get`, `gap_target` und `matrix_progress`.

## Optional

Berechnen Sie die Geschwindigkeit eines vor dem Ultraschallsensor bewegten Objektes.

## Leitfragen

- Welche Distanzen können gemessen werden?
- Welche Zeiten benötigen Abstandsmessungen?