

Name	Matrikel	Anmerkungen
Datum	Raster (z.B. Mo-2x)	Testat/Datum

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

Praktikum 4

Lernziele: Timer, Prescaler und Tonerzeugung. Berechnung Prescaler, Timerperiode für Frequenzerzeugung, Erzeugung einer Tonfolge.

Für die Bearbeitung der Aufgaben ist ein Termin angesetzt. **Eine gute Vorbereitung ist zwingend erforderlich!**

- Bei üblichen Mikrocontrollern werden Timer und ähnliche SoC-Komponenten über schaltbare Prescaler angesteuert. Dies wird in der Schaltung im *TopDesign* simuliert [1].
(Bemerkung: Auf einem PSoC ginge es einfacher, aber wir wollen ja die Realität üben ...)
Alle notwendigen Informationen zu den verwendeten Komponenten erhalten Sie über die Datenblätter zu den Komponenten.
Die Schaltung zur Tonerzeugung ist bis auf die Konfiguration des Pin soweit funktionsfähig, muss also nicht verändert werden.

Bearbeiten Sie zur **Vorbereitung**, wird verlangt, die folgenden Fragen:

- Was bewirken die `FreqDiv_1` bis `FreqDiv_4`?
 - Mit welchem Takt werden Sie angesteuert?
 - Wie wird die `clock` des Timers ausgewählt?
 - Welche Signalform erzeugt der Timer am Ausgang `tc`?
 - Was für eine Signalform ist für eine einfache Tonerzeugung sinnvoll?
 - Was bewirkt die FF-Schaltung? Welche Frequenz liegt am Ausgang `q` an?
 - Wie wird das Signal am `Pin_Buzz` freigegeben? Erklären Sie!
- Planen Sie die effiziente Implementierung! Spart viel Code und mögliche Fehler! **(wird als Vorbereitung verlangt)**
 - Betrachten Sie die Tonfrequenzen in Header-Datei `scale.h`
 - Berechnen Sie einen geeigneten Vorteiler (prescaler) für die gewünschten Frequenzen.
 - Welchen Drive Mode wählen Sie für `Pin_Buzz`, damit er den Buzzer gut treiben kann?
 - Nutzen Sie Makros, um bequem und einfach Einstellungen ändern zu können und die Timer-Periode(n) aus der gewünschten Frequenz und den Voreinstellungen berechnen zu können.
 - Starten Sie **PSoC-Creator** und laden Sie das Projekt Termin 4: *MPS22_Prakt_4*.
 - Suchen Sie die Pins für den Anschluss des Piezzo-Buzzers und lassen Sie sich die Schaltung abnehmen!
 - Testen Sie zuerst den Prescaler über die Tastatureingaben '0' bis '3'.
(Diesen Teil dann später eventuell auskommentieren). Beachten Sie auch die Tastatureingaben 'T' und 't' (gerne benutzen zur Schonung Ihrer und unserer Nerven!)
 - Realisieren Sie über die Tastatur ein einfaches Klavier für die Töne C, D, E, F, G, A, H und C'.

4. Spielen Sie mit der Firmware ein einfaches Lied ab. *Delay* darf für die Tondauer benutzt werden.
 - a. Als Vorschlag für die Definition der Tonhöhe und Dauer siehe die Vorschläge `sNoteSimple_t` und die Tonfolge `sNoteSimple_t harm[]`.
 - b. Realisieren Sie eine Funktion `play(melody, ...)`
 - c. Sie können a. und b. auch gerne anders nach Ihren Vorstellungen realisieren!
 5. Kommentieren Sie – gegebenenfalls nach dem Praktikum zu Hause – Ihren Code. Archivieren Sie Ihr Projekt zu Ihrem späteren Gebrauch.
 6. Schreiben Sie ein kurzes Protokoll und fassen Sie Ihre Erkenntnisse zusammen und fügen Sie die jeweiligen Codeabschnitte hinzu. Laden Sie Ihren Code⁺⁾ als *.zip und Ihr Protokoll als *.pdf in Moodle hoch bis **maximal** 1 Woche nach dem Termin.
-

Bereiten Sie sich auf den Praktikumstermin 4 so vor, dass die Zeit zur Durchführung während des Termins sicher ausreicht. (*Lesen Sie bitte die Aufgabenstellung und Begleitmaterial vor dem Praktikumstermin.*)

Die Themen und Erkenntnisse aus diesem Praktikum werden im Lauf des Semesters weiter benötigt! Arbeiten und dokumentieren Sie Ihre Ergebnisse sorgfältig!

Die Teilaufgaben sind schriftlich zu dokumentieren. Laden Sie Ihr Protokoll wie in 6. beschrieben zu Termins 4 hoch.

Viel Spaß und Erfolg

⁺⁾ im *.zip bitte **nur** den Ordner mit *.c, *.h und gegebenenfalls Projektdatei.

[1] Projekt-Template MPS22_Prakt_4.cydsn.zip auf Moodle