

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

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

Praktikum 5

Lernziele: Auslesen einer 4x4-Matrixtastatur, Pin-Konfiguration, Bit-Manipulation. I2C am Beispiel I/O-Expander mit LED-Ansteuerung, Recherche in technischen Dokumentationen.

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

1. Dieses Praktikum wiederholt elektrische Pin-Konfiguration inkl. Drive-Mode, pullup/pulldown wo erforderlich. Alles, was Sie in den vorangegangenen Praktika erarbeitet und dokumentiert haben, kann von Nutzen sein.
Die Port-Belegung (port.pin(s)) ist funktionsfähig und sollte nicht verändert werden. Jedoch ist die Konfiguration der Ports für die Tastatur nicht funktionsfähig!

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

- a. Elektrische Pin- und Port-**Konfiguration** für die Tastatur vorschlagen
- b. **Funktion** einer Matrixtastatur, siehe [2], erklären können.
- c. **Skizze**, wie Sie Zeilen und Spalten der Tastatur ansteuern und auslesen wollen, siehe [2].
- d. **C-Code** für die Ansteuerung und Auslese der Tastatur **vorbereiten**.
- e. **Grundfunktion** des I2C **PCF8574** erläutern können, [3] und/oder [4]
- f. Bestimmen Sie die **Basisadresse** des **PCF8574**, [3], [4]
- g. Welche Subadressen sind möglich (werden im Praktikum durch Jumper eingestellt)
- h. Elektrische Pin- und Port-**Konfiguration** für den I2C-Slave vorschlagen und begründen.

2. *Lesen Sie die Tasten der 4x4-Matrix mit Ihrem vorbereiteten Programm aus.* Nutzen Sie dazu das Projekt-Template Termin 5: *MPS22_Prakt_5*, [1].
 - a. Schließen Sie das Keypad vorsichtig und korrekt am FreeSoC-Board an und lassen Sie es kontrollieren.
*Lassen Sie vorher das Traffic-Light Shield vom Betreuer entfernen, **nicht selbst!***
 - b. Zeigen Sie die Tastendrücke (Spalte, Zeile) auf dem UART-Terminal an
 - c. Zeigen Sie das jeweils gedrückte Zeichen auf dem Terminal an.
Hinweis: Erzeugen Sie ein Array aus `char`, welches mit Zeile u. Spalte adressiert werden kann.
 - d. *Freiwillig:* Aufzeichnung einer Tastenfolge, Start mit '*', Ende mit '#' und Anzeige auf dem Terminal.

Lassen Sie sich diesen ersten Teil abnehmen.

3. Machen Sie sich mit dem I2C 8-bit I/O-Expander vertraut. Suchen Sie in [3] und/oder [4] die für Sie und die Aufgabe relevanten Informationen.
 - a. An den 8 I/O-Pins des Expanders ist jeweils eine LED angeschlossen, welche active-low leuchtet.
 - b. Wie müssen die Pins `SDA` und `SCL` elektrisch konfiguriert werden, damit die Kommunikation funktionieren kann?

- c. Schließen Sie SDA, SCL, PWR und GND korrekt am FreeSoC2 an *und lassen Sie b) und c) vom Betreuer kontrollieren.*
 - d. Definieren Sie die Device-Adresse `I2C_PCF8574_BASE_ADDR` siehe [3], [4]
 - e. Definieren Sie die Sub-Adresse `I2C_PCF8574_DEV_ADDR` Ihres I/O-Expanders. *Betrachten Sie dazu die Erläuterungen am Ende des Dokuments.*
 - f. Wie wird daraus die korrekte vollständige Adresse `I2C_PCF8574_ADDR` bestimmt?
 - g. Entfernen Sie den Kommentar vor `#define I2C_PROJ_ON`, wenn die Adresse stimmt, dann blinkt's!
4. Erweitern/ändern Sie den I2C-Programmteil im Folgenden ab:
- a. Senden Sie an den I/O-Expander nacheinander ein Byte so, dass ein Binärzähler entsteht, **bei 0** (alles aus) **beginnend**. (Wählen sie ein Delay, z.B. 200ms damit man etwas sieht!)
 - b. *Freiwillig*: Nutzen Sie das Keypad, um die LED's zu steuern:
0: alles aus, 1: nur erste LED, 2: zweite LED, ..., 8: achte LED, *: alle an.
5. Fragen für's Protokoll:
- a. Erklären Sie den 2. Parameter in `I2C_MasterSendStart(addr, ?);`
 - b. Könnte man mit dem I/O-Expander auch das Keypad ansteuern?
Man würde statt den 8 Ports am PSoC in unserer Lösung nur 2 für I2C benötigen, (und könnte obendrein noch andere I2C-Geräte anschließen)
 - c. *Freiwillig*: Warum würde der I/O-Expander mit active-high LED's nicht funktionieren? Siehe [3] oder [4]
6. Kommentieren Sie – gegebenenfalls nach dem Praktikum zu Hause – Ihren Code.
Archivieren Sie Ihr Projekt zu Ihrem späteren Gebrauch.
7. 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 5 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_5.cydsn.zip auf Moodle

[2] AZ-Delivery 4x4-Matrixtastatur

https://cdn.shopify.com/s/files/1/1509/1638/files/4x4_Matrix_Array_Keypad_Datenblatt_AZ-Delivery_Vertriebs_GmbH.pdf?v=1608632986

[3] PCF8574; PCF8574A Remote 8-bit I/O expander for I 2C-bus with interrupt

https://www.nxp.com/docs/en/data-sheet/PCF8574_PCF8574A.pdf

[4] PCF8574 Remote 8-Bit I/O Expander for I2C Bus

<https://www.ti.com/lit/ds/symlink/pcf8574.pdf>

[5] Anhang: letzte Seite

Anhang:

Pinbelegung Matrixtastatur

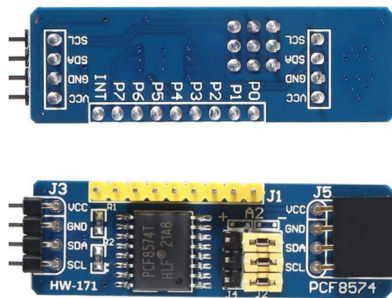


Matrix	R1	R2	R3	R4	C1	C2	C3	C4
PSoC-Pins (port.pin)	0.0	0.1	0.2	0.3	3.0	3.1	3.2	3.4
<i>TopDesign</i> , und *.cydwr	P_Row				P_Col			

Beachten Sie *TopDesign*, dort sind jeweils 4 Pins zusammengefasst und können jeweils zusammen im C-Code mit `P_Row_Write(b); P_Row_Read(b);` und `P_Col_Write(b); P_Col_Read(b);` angesprochen werden. Siehe auch Pin-Belegung im PSoC-Creator.

Die elektrische Konfiguration für diese Anwendung muss von Ihnen noch entsprechend eingestellt werden (sonst funktioniert nichts!).

I/O-Expander



Die Pins des I/O-Expander-Boards sind wie im Bild belegt. Das LED-Shield (LED's active low) ist bereits korrekt aufgesteckt, die Pins für die Subadressen dort nach oben herausgeführt: unten A0, in der Mitte A1 und oben A2. links + (high), rechts – (low) Für die Bestimmung der Subadresse siehe [2] oder [3]