

## 2-stellige Siebensegmentanzeige

Auf der Mikrocontrollerplatine befinden sich 2 Siebensegmentanzeigen DIS\_PC15 und DIS\_PC14

Auf der Mikrocontrollerplatine befinden sich 2 Siebensegmentanzeigen: Digit 0 und Digit 1.

Boardbeschreibung GSOE Base-Shield V1.0

### 2. LED und Siebensegmentanzeige

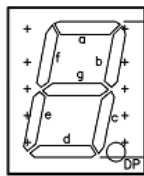
Die LEDs und die Siebensegmentanzeige sind an den Ports PC0 .. PC7 angeschlossen. Die Ports sind als Outputs vorprogrammiert.

Hinweis: An den selben Ports sind auch die Hochstromausgänge angeschlossen.

Die Ausgänge sind highaktiv.

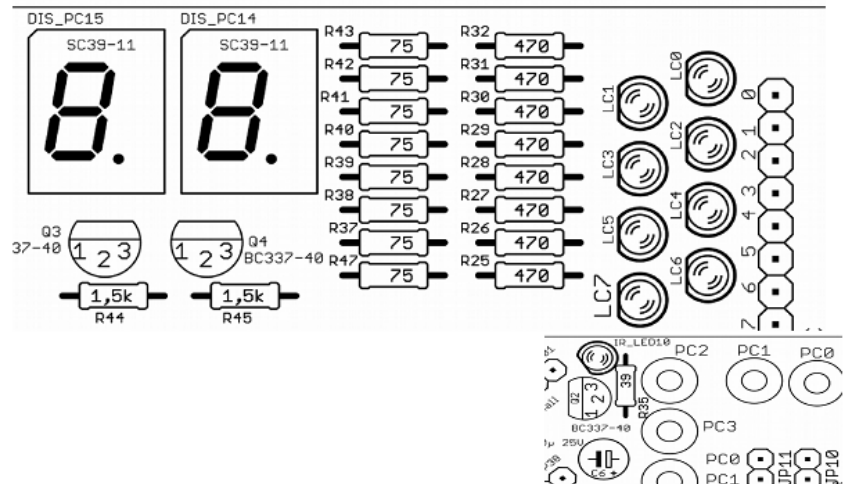
Die beiden Digits der Siebensegmentanzeige werden gemultiplext (d. h. In schnellem Wechsel abwechselnd eingeschaltet).

PC15=1 schaltet die Zehnerstelle, PC14=1 die Einerstelle, ein.



Zuordnung:

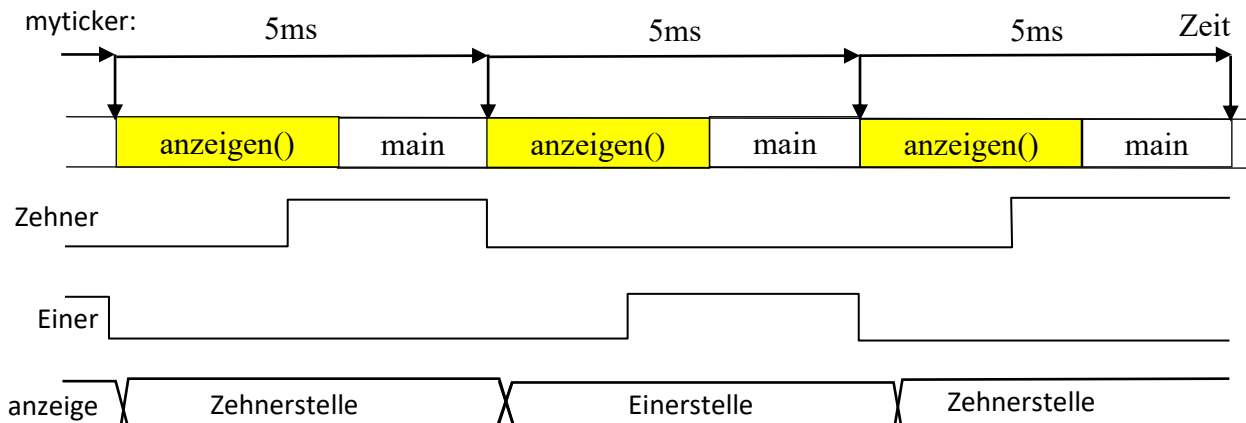
a = PC0 g=PC6  
b = PC1 dp = PC7  
c = PC2  
d = PC3  
e = PC4  
f = PC5



Siebensegmentanzeigen heißen Siebensegmentanzeige weil sie 7 Segmente hat. Beide Digits sind parallel an Port PortC (GPIOC) angeschlossen. Die Auswahl, welches Digit leuchtet erfolgt mit PC\_14 und PC\_15.

Die Ausgabe auf die Siebensegmentanzeige soll jetzt 2-stellig erfolgen! Es sollen die Zahlen von 0 bis 99 ausgegeben werden. Als Zähler für Taste PA10 soll eine Variable int z; verwendet werden.

Der Trick dabei ist: Die zweistellige Zahl wird im Zeitmultiplexverfahren ausgegeben:



Funktionsprinzip:

Der Ticker ruft zyklisch alle 5ms die Operation void anzeigen() auf.

Deklarationen:

- PC\_15 als DigitalOut Zehner
- PC\_14 als DigitalOut Einer
- PortC Bit6..Bit0 als PortOut anzeige
- Zählvariable z als int, Startwert 0
- Ticker myticker
- Port PA\_10 als DigitalIn Taste

Die Operation anzeigen():

- Wenn Einer=1
  - Einer:=0
  - Zehnerstelle ausgeben (anzeige=seg7[z/10];)
  - Zehner:=1
- Sonst
  - Zehner:=0
  - Einerstelle ausgeben (anzeige=seg7[z%10];)
  - Einer:=1

Im Hauptprogramm:

- Initialisieren
  - Ticker mit callback &anzeigen und 5ms
  - Taste mit PullDown
- Endlosschleife
  - Wenn Taste=1
    - z:=z+1
    - warten solange Taste=1

Siebensegmenttabelle:

```
int seg7[10]={0b00111111, 0b00000110, 0b01011011, 0b01001111, 0b01100110,  
             0b01101101, 0b01111101, 0b00000111, 0b01111111, 0b01101111};
```

Aufgaben:

1. Programmieren Sie den Tastenzähler
2. Schreiben ein Programm, das den Einstellwert des Potis: AnalogIn poti(PA\_0); als Spannungswert 0.0 bis 3.3 2-stellig auf die Anzeige bringt.  
Hinweis: poti liefert Kommazahlen (float) im Bereich 0.0 .. 1.0
3. Erläutern Sie, wie der Eindruck einer 2-stelligen Zahl entsteht.