

Multiplexing mit dem Multifunctionshield

1. Dokumentation

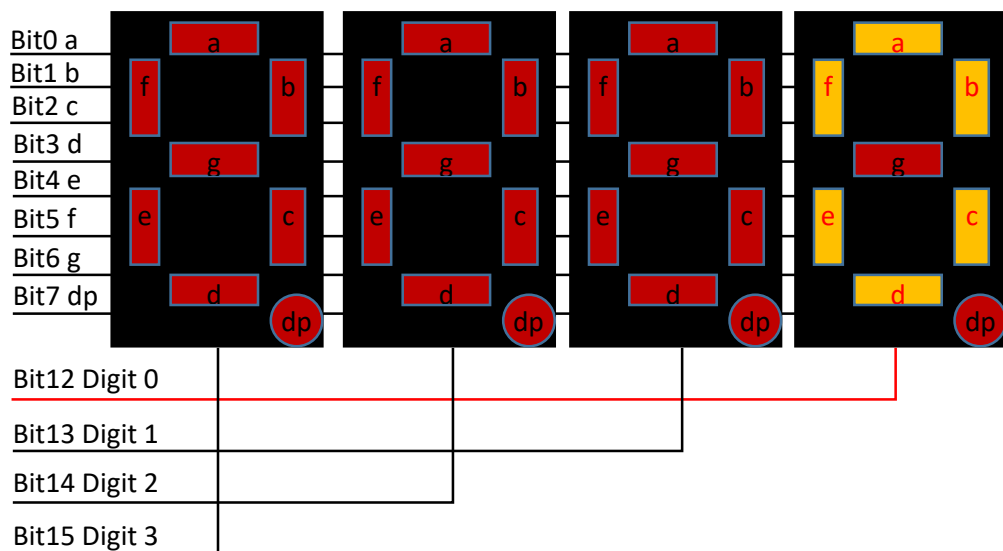
- Neues Programm: Vorlage: mbed OS Blinky LED Hello World
- Import der Bibliothek
 - Offline MBED-Studio: Import Library. URL der Library:
https://os.mbed.com/users/jack1930/code/MFS_Display_HAL/
 - Online MBED: Import Library, Suchen mit GSOE
 MFS_Display_HAL doppelklicken
- Einbinden der Bibliothek in main.cpp

```
/* mbed Microcontroller Library
 * Copyright (c) 2019 ARM Limited
 * SPDX-License-Identifier: Apache-2.0
 */

#include "mbed.h"
#include "MFS.h" //Einbinden der Bibliothek

// Blinking rate in milliseconds
#define BLINKING_RATE 500ms
```

- Globale Deklaration des Multifunctionshields
MFS display; //Anstatt PortOut display verwenden wir MFS display
- Verwendung



char seg7[10]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};

Beispiele: display=0b0001000000111111; //Gibt eine 0 auf Digit 0 aus

display=0x2000+seg7[3]; //gibt 3 auf Digit 1 aus

2. Stufe 1: Zahl auf Digit 0 ausgeben

Programmgerüst:

```
#include "mbed.h"
#include "MFS.h"

MFS display;
char seg7[10]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};
int main()
{
    // Initialise the digital pin LED1 as an output
    DigitalOut led(LED1);

    while (true) {
        display=0x1000+seg7[5]; //gibt eine 5 auf Digit 0 aus
        led = !led;
        HAL_Delay(500);
    }
}
```

- Geben Sie ebenso eine 3 auf Digit 1 aus
- Ergänzen sie eine Zählvariable, die im Sekundentakt immer von 0 bis 9 zählt und dann wieder von vorne beginnt

3. Stufe 2: Die Zahlen 0 .. 3 nacheinander auf die Digits 0 .. 3 mit externem Interrupt ausgeben.

- Die Ausgabe erfolgt mit externem Interrupt an Taste PA1

- Global

- InterruptIn Taste(PA_1);
- int dig=0; //Stellenzähler
- int digTab[4]={0x1000,0x2000,0x4000,0x8000};

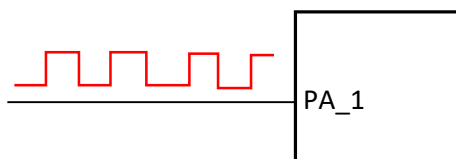
- ISR void TasteISR()

- Ausgabe display=digTab[dig]+seg7[dig]
- dig++
- falls dig=4 dann dig wieder bei 0 beginnen

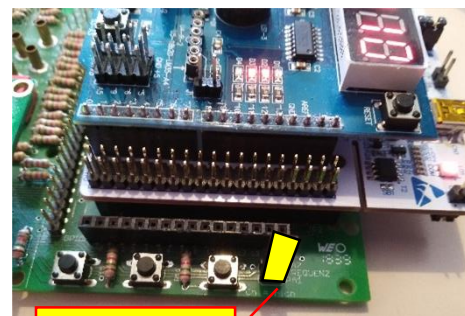
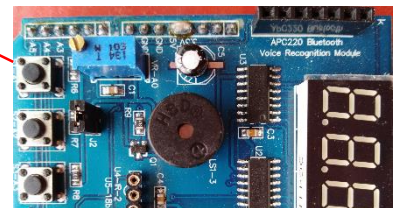
- in main

- Externen Interrupt für fallende Flanke initialisieren
- Die Testbefehle aus Stufe 1 entfernen

- Multiplexbetrieb: Schließen Sie die Brücke, damit werden mit hoher Frequenz Interrupts ausgelöst.



Taste PA1



Brücke

4. Stufe 3: Ausgabe des analogen Poti-Einstellwertes 4-stellig auf das Display

- a. Ergänzungen global
 - `AnalogIn poti(PA_0);` //Analogeingang an Portin PA_0
 - `int ausgabewert=0;` //Diese Zahl soll ausgegeben werden
- b. In der ISR
 - Berechnung der lokalen Variablen int stelle (Einer, Zehner, Hunderter und Tausender)
 - Wenn `dig=0` (Einerstelle) dann `stelle=ausgabewert%10;`
 - Wenn `dig=1` (Zehnerstelle) dann `stelle=ausgabewert/10%10;`
 - Usw.
 - Ausgabe: `display=digTab[dig]+seg7[stelle]`
 - dig erhöhen bis 3, bei 4 von vorne
- c. main
 - In der Endlosschleife: `ausgabewert=poti*10000;`
//poti geht von 0..1, die Anzeige soll aber von 0-9999 gehen.

5. Beantworten Sie folgende Fragen

- a. Warum bemerkt das Auge den Multiplexbetrieb nicht?
- b. Was bewirkt die Anweisung: **`stelle=ausgabewert/10%10;`**
- c. Wie funktioniert die Anweisung: `display=digTab[dig]+seg7[stelle]`