

Annahmen

- Ein SSD1306 OLED-Display mit 128x64 Pixel, Ansteuerung per I2C (7-Bit-Adresse meist 0x3C).
 - Es existieren I2C-Routinen (Start, Write Byte, Stop).
 - Der AVR-Assembler ist für einen gängigen AVR wie Atmega328P.
 - 8-Bit-, Registerorientierter Code.
-

Beispiel: Initialisierung der SSD1306 per I2C in AVR-Assembler

1. I2C Basisroutinen (sehr rudimentär/skizzenhaft)

/*

- i2c_start:
 - - startet I2C-Kommunikation
 - i2c_write_byte:
 - - schreibt ein Byte
 - i2c_stop:
 - - stoppt I2C-Kommunikation
 -
 - Diese Funktionen müssen je nach Target selbst implementiert sein.
 - /
-

2. Befehle für OLED Init

Jedes Kommando wird wie folgt gesendet:

- I2C-Start
 - Slave-Adresse (7Bit + Write)
 - Control-Byte: 0x00 (für Kommandos)
 - Kommando-Byte(n)
 - I2C-Stop
-

3. Beispielcode

```
; Register-Definitions
.equ OLED_I2C_ADDR = 0x3C      ; OLED I2C Adresse (7-Bit)
.equ OLED_COMMAND = 0x00      ; Control-Byte für Kommando über I2C
.equ OLED_DATA = 0x40         ; Control-Byte für Daten
```

```

; --- Funktionsprototypen (Labels) ---
; i2c_start
; i2c_write_byte
; i2c_stop

; Beispiel Subroutine zum Senden eines Kommandos
; Eingabe: Kommando in r16
oled_send_command:
    push r16
    push r17

    rcall i2c_start

    ldi r16, (OLED_I2C_ADDR << 1)      ; Adresse + Write-Bit (0)
    rcall i2c_write_byte

    ldi r16, OLED_COMMAND               ; Control Byte für Kommando
    rcall i2c_write_byte

    pop r17                             ; Kommando in r17
    pop r16

    rcall i2c_write_byte                ; Kommando-Byte senden (in r16)

    rcall i2c_stop

    ret

; Alternative: Übergabe kommando in r16, kein push/pop wenn nicht nötig

; Hauptinitroutine
oled_init:
    ; Folge von Kommandos senden (Beispielwerte für SSD1306)
    ldi r16, 0xAE                      ; DISPLAY_OFF
    rcall oled_send_command

    ldi r16, 0xD5                      ; SET_DISPLAY_CLOCK_DIV
    rcall oled_send_command

    ldi r16, 0x80                      ; CLOCK_DIV_VALUE
    rcall oled_send_command

    ldi r16, 0xA8                      ; SET_MULTIPLEX
    rcall oled_send_command

    ldi r16, 0x3F                      ; MUX_RATIO 64
    rcall oled_send_command

    ldi r16, 0xD3                      ; SET_DISPLAY_OFFSET
    rcall oled_send_command

    ldi r16, 0x00                      ; OFFSET 0
    rcall oled_send_command

    ldi r16, 0x40                      ; SET_START_LINE

    rcall oled_send_command

    ldi r16, 0x8D                      ; CHARGE_PUMP
    rcall oled_send_command

    ldi r16, 0x14                      ; Enable Charge Pump

```

```

rcall oled_send_command

ldi r16, 0x20          ; MEMORY_MODE
rcall oled_send_command

ldi r16, 0x00          ; Horizontal Addressing
rcall oled_send_command

ldi r16, 0xA1          ; SEG_REMAP
rcall oled_send_command

ldi r16, 0xC8          ; COM_SCAN_DEC
rcall oled_send_command

ldi r16, 0xDA          ; SET_COM_PINS
rcall oled_send_command

ldi r16, 0x12          ; COM_PINS_HW_CFG
rcall oled_send_command

ldi r16, 0x81          ; SET_CONTRAST
rcall oled_send_command

ldi r16, 0xCF          ; CONTRAST_VALUE
rcall oled_send_command

ldi r16, 0xD9          ; SET_PRECHARGE
rcall oled_send_command

ldi r16, 0xF1          ; PRECHARGE_VALUE
rcall oled_send_command

ldi r16, 0xDB          ; SET_VCOM_DETECT
rcall oled_send_command

ldi r16, 0x40          ; VCOM_DETECT_VALUE
rcall oled_send_command

ldi r16, 0xA4          ; DISPLAY_ALL_ON_RESUME
rcall oled_send_command

ldi r16, 0xA6          ; NORMAL_DISPLAY
rcall oled_send_command

ldi r16, 0xAF          ; DISPLAY_ON
rcall oled_send_command

ret

```

Hinweise:

- Die tatsächlichen I2C-Routinen müssen kontrollieren, ob das Slave-Device die Bytes mit ACK bestätigt.
- Einige Displays fordern nach einem Reset eine Pause – diese kann per Timer/Delay realisiert werden.
- Das Beispiel ist stark vereinfacht. Im echten Code muss auf Fehlerbehandlung geachtet werden.

- Für eine sequenzielle Übertragung mehrerer Kommandos könnte man auch mehrere Kommandos in einem I2C Paket senden, z.B. ControlByte + mehrere Kommando-Bytes. Das reduziert Overhead.

Falls gewünscht, kann ich Beispiele für die I2C-Routinen oder erweiterte Codes schreiben.