

# Skript

## Student Group

First Name	Surname	Matrikel Nr.

## Table of Contents

<b>Skript</b> .....	2
<i>SW2 Hello Display World - fast Counter</i> .....	2

# Skript

Bildschirmleupe an!

## SW2 Hello Display World - fast Counter

1. Wdh. Hello Blinking World:
  1. DDRx, PORTx, \_Delay\_ms() --> R steht fast immer für Register
  2. --> einmal kompilieren und in Simulide aufbauen
  3. Welche "Vorgaben für die SW-Entwicklung" wurden verletzt? --> Keine magic numbers, sondern #defines !  
siehe Weiterführende Fragen und Infos
2. Heute "Hello World" in echt! Timer + Displayausgabe
3. "Kapitel 2 Sound und Timer bitte nachträglich anschauen"
4. Frage an Studis "Wer weiß nicht was PWM ist?"

In MC Studio

1. neues Projekt "02\_timer"
2. jetzt neu: mit Display!
  1. --> Bibliothek aus wiki herunterladen!
  2. Project --> Add --> existing Item (NICHT drag & drop)
  3. bei mir --> F2 Namen ändern auf "lcd\_lib\_de.h"
  4. Split Screen
  5. Was tun, um Lib in main einzufügen?
  6. #include!
    1. #inc + <Tab>
    2. --> Unterschied <lib.h> vs "lib.h"
3. Durchsicht der lcd\_lib\_de.h
  1. F\_CPU
    1. --> CPU Frequenz, wichtig für genaues Timing der delays
    2. hier 18,432 MHz --> Minimexle Frequenz
    3. Warum 18,432?
      1. ILIAS --> Elektronik Labor --> MiniMEXLE Schaltbild
      2. "Schreck!" sowie Krams auf dem Schaltplan!
      3. Wo ist der Quarz? Quarz schwingt mechanisch im E-Feld --> Schaut im Bild aus wie ein Kondensator
  2. defines --> keine Magic numbers
  3. Funktionsprototypen --> bitte immer am anfang angeben --> gut für eine Übersicht
4. als erstes immer Initialisierung (anlegen der Variablen, verschiedene Konfigurationen etc.)
  1. lcd\_i + <tab>
  2. schon mal kompilieren (immer mal kompilieren zum test, ob noch alles klappt)
  3. noch nicht lauffähig, da nichts angezeigt !
5. einen String ausgeben!
  1. welche Unterfunktion wohl geeignet?
  2. Hinweis auf Inkonsistenz bei Namensgebung
  3. Eingabe lcd\_displayMessage("Hallo!", 0,0 ) --> Hinweis auf Zählanfang 0 nicht 1!
6. Flashen auf Minimexle
  1. Add Target --> STK500 --> ersten COM Port auswählen (und - falls es nicht passt - den nächsten)

2. Tools --> Device Programming
  1. Apply --> Device Signature sichtbar?
  2. --> Memories --> Program
7. Ausgabe von "Hallo! Zähler: "
  1. kann "ä" nicht schreiben , sondern schreibt "μ", warum?
    1. --> Datasheet lesen!
    2. Am besten in der Schaltung den Namen suchen
    3. Googeln nach DEM16216 Datasheet --> Datenblatt etwas kurz? Blockdiagramm (immer schön Bilder in eigene Dokus machen!) --> ST7066U!
    4. Googeln nach ST7066U Datasheet
    5. Kurzes darüberscrollen über das Datasheet
    6. --> Character code Table! --> ist da "ä" drin? In einer schon... In der anderen is "μ" beim gleichen Bitmuster
    7. Also: was tun? entweder "á" nutzen, oder `ldc_putc(111100001);` --> was wurde vergessen? --> % !
  2. Vergleich in Simulide:
    1. Aufbau der Schaltung: mega88 + Hd44780 (ist kompatibel zu ST7066U)
    2. Wie verbinden? Siehe lib (wenn gut beschrieben) oder MEXLE Schaltung
    3. In lib: Port-Bits. PIN\_EN, PIN\_RS --> wo in Simulide?
    4. Für was steht EN? --> Enable. RS --> Register Select
    5. PORT\_DATA: von PORTC nur die ersten 4 bits (0...3)
    6. 18,432 MHz eingeben!
    7. hex file Flashen
    8. --> animation einschalten (High/Low wird angezeigt)
    9. es wird noch nichts ausgegeben?? --> im Code schauen oder im Schaltplan!
    10. PC0 auf D4, PC1 auf D5, PC2 auf D6, PC3 auf D7
    11. jetzt klappts , aber ä an falscher Position
  3. `lcd_gotoxy` einfügen
    1. In Simulide autoloader einschalten!

Jetzt: aufsteigende Zahlen ausgeben Was tun?

1. Laufvariable anlegen und nutzen: `uint8_t i=0;` und `i++` in der Schleife
2. wie gibt man Zahlen aus? `sprintf`? (kann in String einen Zahlenwert ausgeben)
3. `sprintf(output_str, "i:%03u", i);` 3 --> drei Stellen, u --> unsigned
4. `output_str` deklarieren
5. kompilieren --> `#include <stdio.h>` vergessen
6. Simulation herunterdrehen

aktuell zählt er nicht so schnell wie die CPU kann, sondern so schnell wie er es ausgeben kann. Die CPU kann aber schneller!

1. Blick ins Datenblatt des atmega88
2. Blockbild
3. T/C im Inhaltsverzeichnis --> 16 bit Timer/Counter
4. wichtig sind immer die Register
  1. TCNTn --> timer Counter ; für was steht n? in Mathe?
  2. OCRnA --> Output Compare
  3. TCCRnA
5. `sprintf(output_str, "TC:%03u", TCNT1);` --> an zweiter Pos ausgeben. --> wichtig: 3 in 5 ändern!
  1. kommt noch nix heraus!
  2. Control logic --> steht im folgenden in der Register Description

3. Blick in die Tabellen, was bei initialisierung mit 0 passiert
4. --> CS auf 000 hat keine Clock!
5. also CS10 setzen für schnellsten Zähler
6. `TCCR1B |= 1<<CS10;`

From:

<https://first.mexle.te.hs-heilbronn.de/> - **MEXLE Wiki**

Permanent link:

<https://first.mexle.te.hs-heilbronn.de/microcontrollertechnik/skript?rev=1728252299>

Last update: **2024/10/07 00:04**

