

WEEK 4

ONDERWERPEN

- interrupts in C
- TM1638 led&key
- UART, PWM en ADC
- using a TTC-scheduler
- state diagram

GROEN GEMARKEERD = AFTEKEN OPDRACHT

OPDRACHT 1 : AANSTURING DISPLAY LED&KEY

Op Blackboard is het programma "demo TM1638.c" te vinden. Compileer dit programma en flash het op het Arduino bordje. Pas het programma zodanig aan dat jullie namen op het display worden getoond.

OPDRACHT 2 : REACTIETESTER

- Schrijf een programma dat een teller op de display weergeeft die elke 100 milliseconde met één wordt verhoogd.
- Je gaat nu dit programma uitbreiden naar een reactietester. Pas de teller uit (a) aan zodat nu elke 10 milliseconde de teller wordt verhoogd. Verbind 2 knoppen op het bordje, deze noemen we swA en swB. Drukt gebruiker A op swA, dan begint de teller te lopen, net zolang tot gebruiker B op swB heeft gedrukt. De reactietijd wordt op het display getoond.

Op Blackbord is de code voor de AVR_TTC_scheduler (.c en .h) te vinden. Deze scheduler moet je gebruiken. Maak voordat je met programmeren begint eerst een ontwerp. Bespreek in je ontwerp welke periodieke taken (functies) je nodig hebt. Bijvoorbeeld : hoe vaak moet je naar de knopjes kijken, hoe vaak moet het display worden bijgewerkt ? En wat zijn bij de functies de in- en/of uitgaande variabelen ?

- c) Vul je ontwerp aan met een timing diagram als hieronder. Hierin geeft je aan om de hoeveel timer ticks de periodieke taken worden uitgevoerd.

[illegible]

OPDRACHT 3 : AFSTANDSMETER

- a) Om de duur van een interval te meten maakt iemand het onderstaand programma. Je mag er van uitgaan dat het programma compileert. Toch is er iets (subtiel) mis met dit programma, wat is het ?

Tip : kan gv_counter ook negatief zijn ?

```
volatile int gv_counter;
init_timer();
// start counting
TCNT1 = 0;

... wait some time ...

// read value timer1
gv_counter = TCNT1;
// display timer value
display_value(gv_counter);
```

- b) De SCMI-kit bevat een ultrasone sensor HC-SR04 voor het meten van afstand. De user manual hiervan kun je vinden op Blackboard. Zoals je kunt lezen in deze datasheet moet je een korte puls (tenminste 10 μ s) naar de trigger input sturen, en de breedte van de echo puls gaan meten. Het resultaat van de meting, de afstand in cm, moet worden weergegeven op de display.

Een groot deel van het programma is al gegeven. De weking van het programma is toelicht dmv een sequence diagram. Zie de files op Blackboard.

Met de statement `TCCR1B = _BV(CS10);` start je de teller, en met `TCCR1B = 0;` stop je de teller.

Sluit de sensor als volgt aan: Trigger op PD0 en Echo op PD3 (INT1). En sluit de LED&KEY als volgt aan : data op PB0 en clock op PB1.

Op Youtube zijn diverse filmpjes over de te vinden over de HC-SR04 zoals bijvoorbeeld

<https://www.youtube.com/watch?v=jDFU4noShQA>.

Maak voordat je met programmeren begint eerst een ontwerp. Welke functies heb je nodig ?

OPDRACHT 4 : VERBINDING VIA UART

In deze opdracht ga je een RS232-verbinding opzetten tussen PC en STK500.

Installeer op je pc een RS232 terminal emulator, bijvoorbeeld Putty of Realterm (v2). Bedenk dat PD0=RX en PD1=TX.

- a) Maak een programma "echo", dat alle karakters die op de Atmega UART worden ontvangen direct terug stuurt. Je moet zowel transmitter als receiver aan zetten (bits RXEN0 en TXEN0). En in je functie receive() wachten tot bit RXC0 '1' is (is set).
- b) Hoe lang duurt het bij de door jou gekozen baudrate voordat een karakter volledig is ingelezen in de UART buffer ? (M.a.w. volledig verzonden is uit de UART buffer ?)
- c) Breid dit programma uit zodat je vanuit je RS232-terminal op de PC de 1 LED op het Arduino bordje kunt aansturen. Bijvoorbeeld na het versturen van '1' gaat de LED aan, en na het versturen van '0' gaat de LED uit.

OPDRACHT 5 : MODELLEREN CRUISE CONTROL

Een cruise control systeem is een embedded applicatie in een auto met als doel het constant houden van de snelheid. De applicatie moet goedkoop maar ook zeer betrouwbaar zijn.

Het systeem heeft de volgende interactie met de gebruiker :

- set button: start cruise control, doelsnelheid = huidige snelheid
- inc button: verhoog de doelsnelheid (+ 5 km/u)
- dec button: verlaag de doelsnelheid (- 5 km/u)
- brake/clutch pedal : stop de cruise control

Maak een state diagram voor dit systeem.

Tip : Naast de toestand 'init' zijn er nog drie toestanden.