2 januari 2022

Jeroen Visser, Teun Leenders

Versie 1

Groeidocument

Microcontrollers

Inhoudsopgave

[GitRepo 2](#_Toc95341867)

[Week 1 3](#_Toc95341868)

[Opdracht A.1 3](#_Toc95341869)

[Opdracht A.2 3](#_Toc95341870)

[Opdracht A.3 3](#_Toc95341871)

[Opdracht B.1 4](#_Toc95341872)

[Opdracht B.2 4](#_Toc95341873)

[Opdracht B.3 4](#_Toc95341874)

[Opdracht B.4 4](#_Toc95341875)

[Opdracht B.5 4](#_Toc95341876)

[Week 2 5](#_Toc95341877)

[Opdracht A1 5](#_Toc95341878)

[Opdracht A2 5](#_Toc95341879)

[Opdracht B1 5](#_Toc95341880)

[Opdracht B2 5](#_Toc95341881)

[Opdracht B3 6](#_Toc95341882)

[Opdracht B4 6](#_Toc95341883)

[Opdracht B5 7](#_Toc95341884)

# GitRepo

**De git repository is te vinden op:** <https://github.com/TONI330/microcontrollers_2_3>

# Week 1

## Opdracht A.1

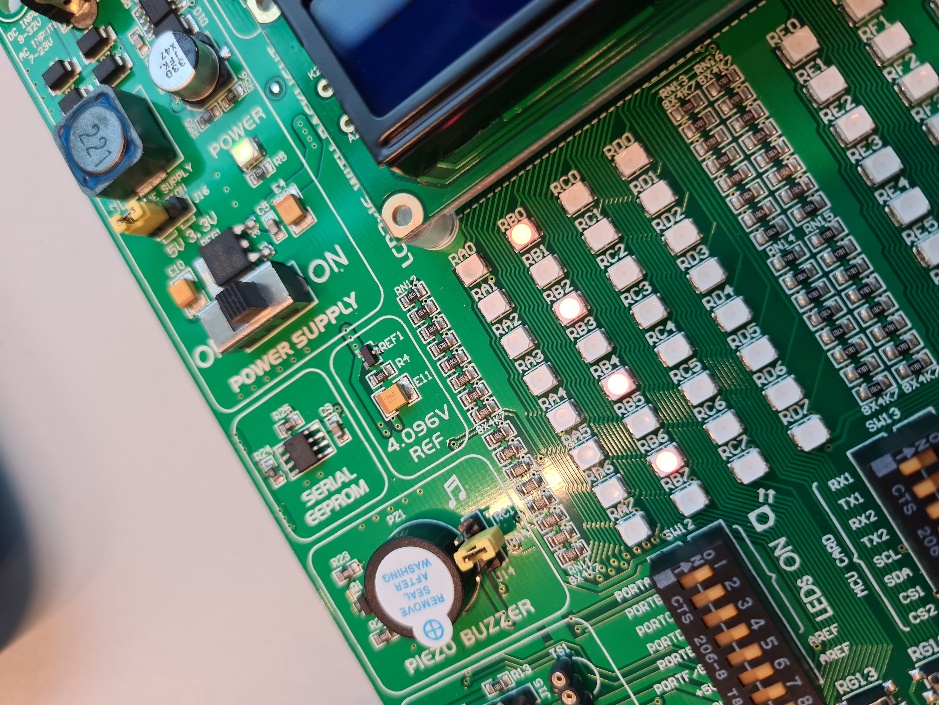
**Opdracht: Verzamel alle materialen.**

Alle materialen zijn verzameld.

## Opdracht A.2

Figuur

**Opdracht: Run code in de simulator.**

De code runt in de simulator. Er is te zien dat de code de juiste I/O aanstuurt, omdat dit onderaan te zien is in de autos. Zie figuur 1.

Figuur

Verder hebben we ook de code op het echte bord getest. Hier werkte het ook zoals te zien is op figuur 2. Of [Klik hier voor de video.](https://mymedia.avans.nl/media/Microcontrollers+opdracht+A2/1_12c1pxg2)

## Opdracht A.3

**Opdracht: Loop doormiddel van de JTAG module door de code heen.**

Het is gelukt om de JTAG aan te sluiten en doormiddel hiervan de code te debuggen kijk daarvoor [deze](https://mymedia.avans.nl/media/Microcontrollers+opdracht+A3/1_6wwlt8qf) video.

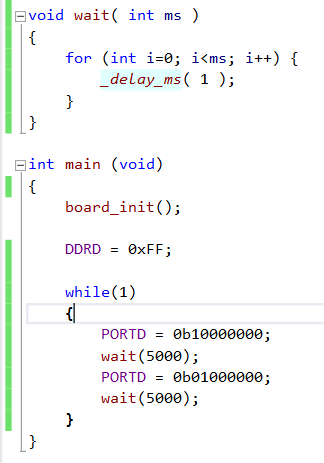
## Opdracht B.1

**Opdracht: Beantwoord de volgende vragen vanuit de literatuur.**

Hoe groot is het program memory van de ATmega128? De grootte is 128KB.

Wat is het adres van Data direction register van PORTE (DDRE)? Het adres is

## Opdracht B.2

**Opdracht: Ontwikkel een applicatie die bepaalde LED’s om en om laat knipperen.**

Figuur

Voor de code voor de applicatie is te zien in figuur 3.

Het resultaat van de code is te zien in de video [hier](https://mymedia.avans.nl/media/MC+Opdracht+B2/1_7lqfmtt8).

## Opdracht B.3

**Opdracht:** Maak een applicatie die een led laat knipperen als de drukknop ingedrukt en stopt bij het loslaten van de drukknop.

Zie de repository voor de code van de opdracht en klik [hier](https://mymedia.avans.nl/media/Microcontrollers+opdracht+B3/1_4z7f2yct) voor de video.

## Opdracht B.4

**Opdracht: Implementeer een looplicht applicatie op de LED’s van PORTD. Tussen elke verandering van output zit 50ms (milliseconden). Hoe zou je dit kunnen meten? Om een eenvoudig looplicht te maken kun je gebruik maken van de shift operatoren in C (de >> en de <<). Dit heb je ook al gedaan in periode TI-1.1 op het GUI board.**

**//TODO uitwerken**

## Opdracht B.5

**Opdracht:**

We hebben debug code erin staan die op poort DDRC laat zien hoe groot de lijst met instructies is in binair.

Hieronder staat de belangrijkste code:

while (1) {

// Set index to begin of pattern array

for(int i=0; i < sizeof(pattern) / sizeof(LIGHTS\_PATTERN); i++)

{

PORTD = pattern[i].adress; //set light value

wait(pattern[i].delayms); //wait the given delay

}

}

De video is [hier](https://mymedia.avans.nl/media/Microcontrollers+opdracht+B5/1_13akj9jp) te vinden.

# Week 2

## Opdracht A1

**//TODO uitwerken**

## Opdracht A2

**//TODO uitwerken**

## Opdracht B1

**Opdracht:** Download ioisr.c uit de repository en executeer deze op het UNI-DS board. Verklaar de werking.

De video is [hier](https://mymedia.avans.nl/media/Microcontrollers+week+2+opdracht++B1/1_k4el2np5) te vinden.

In de main() functie worden er twee interrupts geïnitialiseerd. Verder bestaat deze functie uit een while loop met het knipperen van de led op D7 met een pauze van 500ms ertussen.

Er zijn twee andere methodes, deze beschrijven wat er gebeurd op de twee interrupts. De eerste interrupt staat op knop D0. Deze zet de led op D5 aan. De tweede interrupt staat op D1. Deze zet de led op D5 uit.

Dit zorgt ervoor dat ondanks dat de main() functie met while loop altijd bezig is, de twee knoppen een eigen functie hebben.

## Opdracht B2

**Opdracht:** Implementeer een applicatie die met de ISR\_INT1 en ISR\_INT2 het looplichtje naar voren en naar achteren verplaatst. In de main() methode mag alleen een initialisatie en een lege while loop staan.

De video is [hier](https://mymedia.avans.nl/media/Microcontrollers+week+2+opdracht++B2/1_plfmeajf) te vinden.

ISR (INT1\_vect)

{

//shift the led on forwards

if(PORTC == 0x01) {

PORTC = 0x80;

} else PORTC >>= 1;

}

We gebruiken bitshift operaties om de ledjes te verplaatsen.

## Opdracht B3

**Opdracht:** implementeer een display() functie die een hexadecimaal getal van 0 tot 15 kan weergeven. Valt dit getal hierbuiten, laat dan de “E” van Error zien. Loop door de getallenreeks heen met twee knoppen.

Het resultaat van de opdracht staat [hier](https://mymedia.avans.nl/media/Microcontrollers+week+2+opdracht++B3/1_maevo5vt).

void display(int digit) {

if(0 <= digit && digit <= 15) {

//display het getal

PORTA = numbers[digit];

} else {

//display "e" van error

PORTA = numbers[14];

}

}

Hierboven staat de display methode gebruikt om de nummers weer te geven.

## Opdracht B4

**Opdracht:** Gebruik inspiratie uit lookup.c van week 1 om een lichteffect te maken met het 7 segment display.

De video is [hier](https://mymedia.avans.nl/media/Microcontrollers+week+2+opdracht++B4/1_kew1zqce) te bekijken.

DDRA = 0xFF; //Set all A ports to output

int index = 0;

while(1==1) {

//Display

PORTA = pattern[index].data;

//Wait

wait(pattern[index].delay);

//Set index

if(index <= sizeof(pattern) / sizeof(PATTERN\_STRUCT)) {

index++;

} else index = 0;

}

Hierboven staat hoe er door de struct[] wordt gelopen om het lichteffect weer te geven.

## Opdracht B5

**Opdracht:** Ontwerp een LCD module (als lcd.c file) waarin de init, display\_text en set\_cursor functies zijn geïmplementeerd. Zodat deze als generieke module gebruikt kan worden om de LCD aan te sturen. Schrijf hier voor ook een testprogramma in een aparte file.

Figuur

Het resultaat van het testprogramma is hiernaast te zien in figuur 4. De volledige code staat uiteraard op github.