Practicum

Track 4 + 5



**Cas Koopmans**

**Berend Vet**

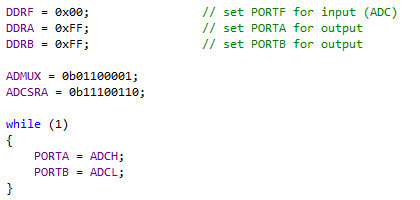
**09/03/2017**

# Repository

Deze repository bevat de volledige code die voor deze opdrachten gemaakt zijn.

# <https://github.com/CasKoopmans/MicrocontrollersAssignments>

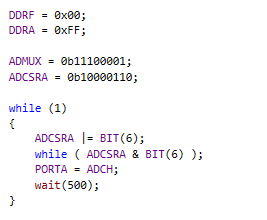
# Opdracht B1



In de ADMUX wordt op referentiespanning 5V gezet, ADLAR op 1 gezet en de analog bits op PF1.

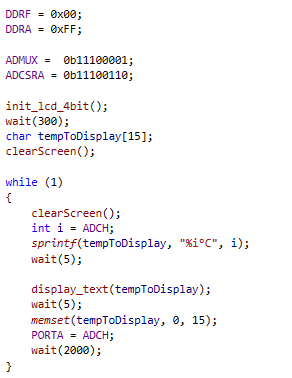
ADSCRA wordt de ADC enabled, gestart en op free running gezet. De devision factor is hier 64.

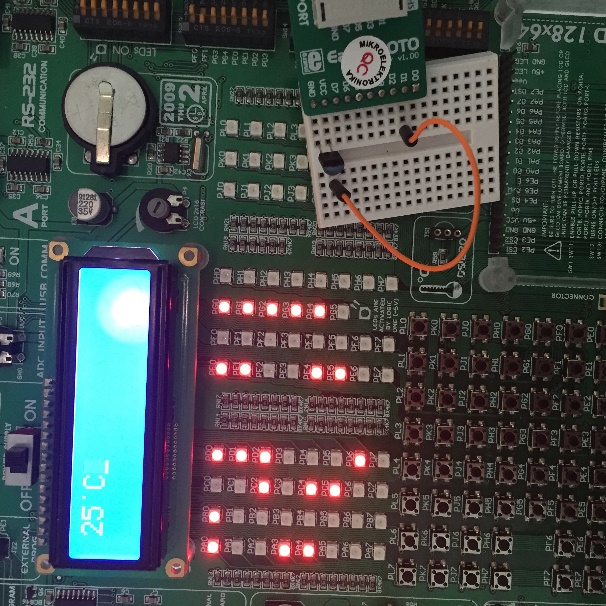
# Opdracht B2



De ADCSRA start iedere halve seconde. Hierdoor worden iedere halve seconde de 8 bits van ADCH afgepeeld met de ledjes van PORTA.

# Opdracht B3





# Opdracht C



a)

Alleen de eerste twee digits worden aangestuurd.

B1)

Om alle digits aan te sturen en 1-2-3-4 te schrijven moet het volgende aangepast worden:

spi\_slaveSelect(0); // Select display chip

spi\_write(0x0B); // Register 0B: Scan-limit

spi\_write(0x01); // -> 1 = Display digits 0..1

spi\_slaveDeSelect(0); // Deselect display chip

naar

spi\_slaveSelect(0); // Select display chip

spi\_write(0x0B); // Register 0B: Scan-limit

spi\_write(0x03); // -> 1 = Display digits 0..1

spi\_slaveDeSelect(0); // Deselect display chip

met deze commando word het scan-limit vergroot naar alle 4 de digits.

en

for (char i =1; i<=2; i++)

{

spi\_slaveSelect(0); // Select display chip

spi\_write(i); // digit adress: (digit place)

spi\_write(i); // digit value: i (= digit place)

spi\_slaveDeSelect(0); // Deselect display chip

wait(1000);

}

Naar

for (char i =1; i<=4; i++)

{

spi\_slaveSelect(0); // Select display chip

spi\_write(i); // digit adress: (digit place)

spi\_write(5-i); // digit value: i (= digit place)

spi\_slaveDeSelect(0); // Deselect display chip

wait(1000);

}

Hier worden commandos om 1tm4 te schrijven naar alle 4 de digits gestuurd.

B2)

Om de intensiteit aan te passen word het volgende aangepast.

spi\_write(0x0A); // Register 0A: Intensity

spi\_write(0x0F); // -> Level 4 (in range [1..F])

De range van helderheid gaat van laag naar hoog(1…F)

c)

void spi\_writeWord(unsigned char adress, unsigned char data){

spi\_slaveSelect(0); // Select display chip

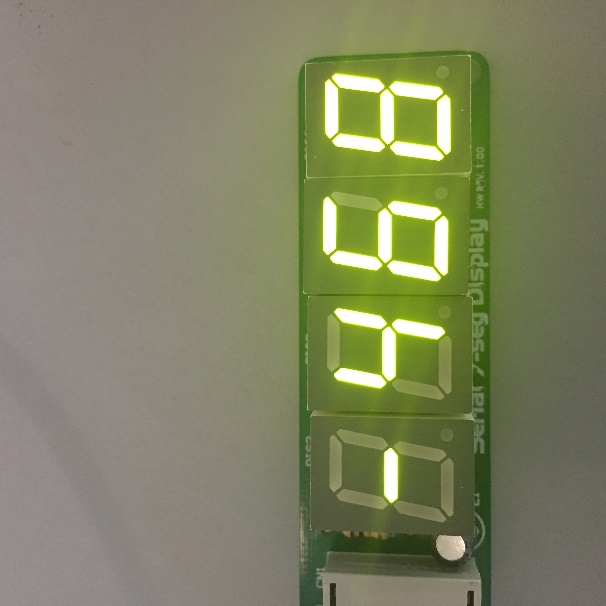
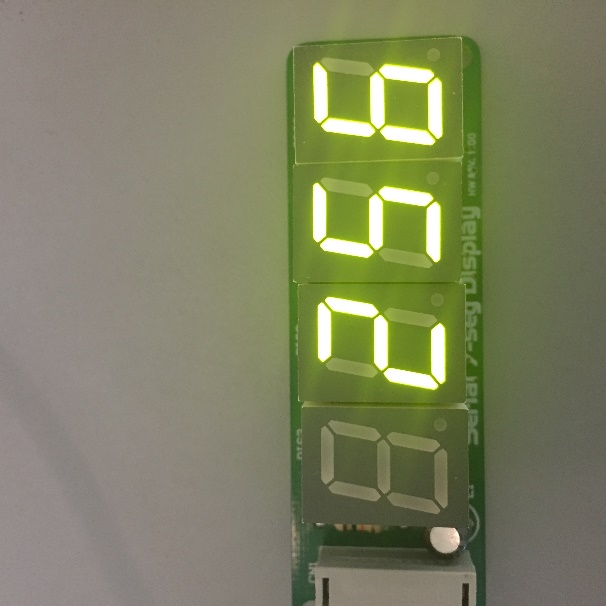
spi\_write(adress); // send adress

spi\_write(data); // send value

spi\_slaveDeSelect(0); // Deselect display chip

}

# Opdracht C1



void writeLedDisplay(int value){

bool negative = value < 0; // check if value is negative

int length = *floor*(*log10*(*abs*(value))) + 1; // check how long the int is

if(length > 4 || (negative && length > 3)) // check whether it is too big

return;

if(negative) // if it is negative turn it positive

value = *abs*(value);

for(int i = 1; i<5; i++){ //for each number in the int print

int digit = value % 10; // cycle through the numbers

value /= 10;

if(i>length)

spi\_writeWord(i,0b1111); // if number is over print blank

else

spi\_writeWord(i, (digit)); // else print the number

}

if(negative)

spi\_writeWord(length+1,0b1010); // if it is a negative print a 'minus' sign

}