

KLAS(SEN) : EQ1	BLAD : 1 van 3 BLADEN
TOETS : Microcontroller Programmeren in C	DOCENT : Harry Broeders
CODE : MICPRG-sc1	DATUM :
KWARTAAL: 2	TYPE : voorbeeld
	TIJD :

Tijdens dit tentamen mogen **alle** boeken, dictaten, aantekeningen enz. worden gebruikt.

Bij elke opgave staat tussen haakjes het maximale aantal te behalen punten vermeld.
Eindcijfer = (aantal behaalde punten + 10) / 10.

1. Op de **C poort** van een ATmega32 zijn 8 LEDs aangesloten. Elke LED kan aangezet worden door een 1 naar de betreffende outputpin van de C poort van de ATmega32 te schrijven. Elke LED kan uitgezet worden door een 0 naar de betreffende outputpin te schrijven. Schrijf een programma waarin de LEDs één voor één, van links naar rechts, aangezet en weer uitgezet worden. Elke LED moet gedurende 250 ms branden. Hierdoor ontstaat een zogenoemd “looplicht”. De opzet voor het programma is hieronder gegeven.

```
#include <avr/io.h>
#include <stdint.h>
#include <util/delay.h>

void wait(void) {
    uint8_t i;
    for (i = 0; i < 10; ++i)
        _delay_ms(25);
}

int main(void) {
    /* DEZE CODE MOET JIJ SCHRIJVEN! */
    return 0;
}
```

(20 punten) Geef de code die in de main functie moet worden ingevuld.

2. In een C programma voor een ATmega32 moet regelmatig (zo nauwkeurig mogelijk) 1 seconde worden gewacht. Een programmeur heeft bedacht om hier een functie voor te schrijven. De opzet van deze functie is hieronder gegeven. Verder is gegeven:
 - de klokfrequentie van de ATmega32 is 4 MHz.
 - timer/counter 1 van de ATmega32 **moet** gebruikt worden.
 - er moet een prescaler waarde worden gekozen waarmee de wachttijd van 1 seconde **zo nauwkeurig mogelijk** benaderd wordt.
 - het is **niet** nodig om interrupts te gebruiken.

```
#include <avr/io.h>

void wait1sec(void) {
    /* DEZE CODE MOET JIJ SCHRIJVEN! */
}
```

(20 punten) Geef de code die in de wait1sec functie moet worden ingevuld.

Zie volgende blad ⇨

KLAS(SEN) : EQ1	BLAD : 2 van 3 BLADEN
TOETS : Microcontroller Programmeren in C	DOCENT : Harry Broeders
CODE : MICPRG-sc1	DATUM :
KWARTAAL: 2	TYPE : voorbeeld
	TIJD :

Tijdens dit tentamen mogen **alle** boeken, dictaten, aantekeningen enz. worden gebruikt.

3. In een C programma voor een ATmega32 moet pin B3 hoog gemaakt worden als de spanning op pin A3 kleiner dan 1.00 V is of groter dan 2.00 V is. Pin B3 moet laag gemaakt worden als de spanning op pin A3 groter dan, of gelijk aan, 1.00 V is én lager dan, of gelijk aan, 2.00 V is. De spanning op pin A3 moet met een bepaalde regelmaat worden bekeken. De programmeur van dit programma heeft ervoor gezorgd dat Timer/Counter 0 een Overflow geeft als pin A3 moet worden bekeken. De ADC van de ATmega32 moet zodanig worden ingesteld dat er automatisch een conversie gestart wordt als Timer/Counter 0 een Overflow heeft. Als de conversie voltooid is moet in een **Interrupt Service Routine** de spanning op pin A3 worden ingelezen en pin B3 hoog of laag maakt worden. De opzet van dit programma is hieronder gegeven. Verder is gegeven:

- er is **géén** referentiespanning aangesloten op de AREF pin van de ATmega32.
- er is een condensator aangesloten op de AREF pin van de ATmega32 zodat je gebruik kunt maken van de interne referentiespanning.
- de klokfrequentie van de ATmega32 is 4 MHz.

```
#include <avr/io.h>
#include <avr/interrupt.h>

void initTimer0(void) {
    /* Deze code zorgt voor een Overflow als de spanning op
       pin A3 bekeken moet worden. Deze code is al geschreven */
}

void initADC(void) {
    /* DEZE CODE MOET JIJ SCHRIJVEN! */
}

/* hier komt de benodigde Interrupt Service Routine */
/* DEZE CODE MOET JIJ SCHRIJVEN! */

#define SINT_JUTTEMIS 0

int main(void) {
    DDRB = 0xFF;
    initTimer0();
    initADC();
    sei();
    while (!SINT_JUTTEMIS);
    return 0;
}
```

(15 punten) Geef de code die in de `initADC` functie moet worden ingevuld.

(15 punten) Geef de code van de benodigde Interrupt Service Routine.

Zie volgende blad ⇨

KLAS(SEN) : EQ1	BLAD : 3 van 3 BLADEN
TOETS : Microcontroller Programmeren in C	DOCENT : Harry Broeders
CODE : MICPRG-sc1	DATUM :
KWARTAAL: 2	TYPE : voorbeeld
	TIJD :

Tijdens dit tentamen mogen **alle** boeken, dictaten, aantekeningen enz. worden gebruikt.

4. In het onderstaande C programma is de in de les besproken datastructuur Stand gegeven. De functie totaalPunten moet het totaal aantal punten van alle spelers samen berekenen.

```
#include <stdio.h>

typedef struct {
    char naam[80];
    int punten;
} Deelnemer;

typedef struct {
    Deelnemer speler[100];
    int aantalSpelers;
} Stand;

int main(void) {
    Stand leesData(void);
    int totaalPunten(Stand);
    Stand s;

    s = leesData();
    printf("Totaal aantal punten = %d", totaalPunten(s));

    getchar();
    return 0;
}

int totaalPunten(Stand st) {
    /* DEZE CODE MOET JIJ SCHRIJVEN! */
}

Stand leesData(void) {
    /* Deze code is al geschreven */
}
```

(20 punten) Geef de code die in de totaalPunten functie moet worden ingevuld.