

```
.include "m32def.inc"
.def temp = R16
.def vismin = R17
.def sek = R18
.def min = R19
.org 0x0 jmp start
.org 0x2 jmp bryter2
.org 0x4 jmp bryter3
.org 0xe jmp sekund
```

Start:

```
; Clear
clr temp
clr vismin
clr sek
clr min
```

```
; Morklegge dioder
ldi temp, 0xff
out DDRB, temp
```

```
; Initialisere stack-peker
ldi temp, 0x02
out sph, temp
ldi temp, 0x5f
out spl, temp
```

```
; Initialiser for avbrudd fra INT0 og INT1
```

```
ldi temp, 0xC0
Out GICR, temp
ldi temp, 0x0A
Out MCUCR, temp
```

```
; Initialiser for timer1
ldi temp, 0x10
Out TIMSK, temp
ldi temp, 0b00001011
out TCCR1B, temp
ldi temp, 0xf4
out OCR1AH, temp
ldi temp, 0x24
Out OCR1AL, temp
sei
```

```
;Interrupt rutine
```

```
Audun1:
Cpi vismin, 0
```

```
breq HP2
cli
com min
out portB, min
com min
Sei
jmp Audun1
```

```
HP2:
Cli
Com sek
Out portb, sek
com sek
Sei
jmp Audun1
```

; Bryter avbrudd

```
bryter2:
ldi vismin, 1
reti
```

```
bryter3:
ldi vismin, 0
reti
```

;Sekund avbrudd

```
Sekund:
push temp
ldi temp, sreg
push temp
```

```
ldi temp, 0x1
add sek, temp
```

```
cpi sek, 60
breq minpluss1
```

```
sekpluss1:
```

```
pop temp
out sreg, temp
pop temp
reti
```

; Minutt avbrudd

```
minpluss1:
ldi temp, 1
```

```
add min, temp  
clr sek
```

```
cpi min, 60  
brne sekpluss1  
clr min  
jmp sekpluss1
```