

Hvilken værdi skal der skrives til tælleregisteret TCNTn, hvis der ønskes x sekunder til timeren laver overflow?

$$TCNTn = Max_{Timer} - \left(\frac{CPU_{hz}}{prescaler} * x \right)$$

Asynkron seriel kommunikation implementeres via en UART, som er initieret til whatever. Hvor mange 8 bit tegn kan der maksimalt sendes på 1 sekund?

Svar: Tæl antallet af bits

BAUD rate = 19200
1 startbit
8 databit
Ulige paritet
2 stopbit

"Ulige paritet" er også en bit. I ovenstående tilfælde $\frac{19200}{1+8+1+2} = 1600$

Hvor mange databits og hvilken paritet anvendes?

UART DRIVERS MED INTERRUPTS SLIDES side 12 (for bit-størrelse) og side 10 (for paritet). Afhænger af hvordan UCSRA/UCSR0B/UCSR0C er sat op (altså hvilke bits der er tændt).

BAUD rate findes på side 13.

En n-bit A/D-konverter anvendes til at måle en spænding i området 0 volt til 5 volt.

Ved indgangsspændingen 0 volt vil A/D-konverteren give et digitalt resultat på 0.

Ved indgangsspændingen 5 volt vil A/D-konverteren netop give maksimalt digitalt resultat.

Hvad er A/D-konverterens opløsning?

$$V_{\text{resolution}} = \frac{V_{\text{fullscale}}}{2^n - 1}$$

$$\frac{V_{\text{in}}}{V_{\text{fullscale}}} = \frac{X}{2^n - 1}$$

Hvilken værdi får x når AD/konverterens indgang tilsluttes en spænding på 1.5 volt (V_{in})?

```
int main()
{
    unsigned int x;
    ADMUX = 0b11000000;
    ADCSRA = 0b10000111;
    ADCSRB = 0b00000000;
    while(1)
    {
        ADCSRA |= 0b01000000;
        while ( (ADCSRA & 0b01000000) != 0 )
        {}
        x = ADCW;
    }
}
```

ADMUX 2 FØRSTE BIT BESKRIVER $V_{fullscale}$

REFS1	REFS0	Voltage Reference Selection ⁽¹⁾
0	0	AREF, Internal V _{REF} turned off
0	1	AVCC with external capacitor at AREF pin
1	0	Internal 1.1V Voltage Reference with external capacitor
1	1	Internal 2.56V Voltage Reference with external capacitor

AREF = Ekstern pin
AVCC = 5 volt
Intern 1,1 volt
Intern 2,56 volt

Hvis bit-størrelse ikke er specificeret, så er den 10. F.eks. $2^{10} - 1$

Efter afvikling af en interrupt Service Rutine (ISR) ønsker man at forsætte i programmet fra hvor man blev afbrudt af interruptet. Hvordan sker dette?

Programtælleren gemmes automatisk på stacken. Hentes automatisk tilbage til programtælleren når interruptrutinen afsluttes.

```
#define F_CPU 16000000
```

```
#define F_CPU 8000000
```

Porten vil nu toggles 2 gange hvert sekund (omvendt intuition).

Hvor lang tid tager det at afvikle koden?

$$\frac{\text{Cykler}}{CPU_{freq}}$$

Mhz = output i mikrosekunder

Hz = output i sekunder

Hvilke flag bliver sat?

Se hvilke ting der påvirker flag ved at slå instruktionerne op. F.eks. påvirker ADD flag, men LDI gør ikke.

En Mega2560's timer X anvendes til at generere et PWM signal og er i Y-bit PWM mode. PWM signalets frekvens er Z Hz. Vi ændrer nu X's mode til Y' bit PWM. Hvad bliver den nye frekvens?

Først skal man regne CPU-frekvensen ud (vi får givet ben-frekvensen Z), og så dernæst regne den nye benfrekvens ud. Man kan slå op hvad TOP er for given timer/mode. **HUSK AT LÆGGE +1 TIL TOP.** Hvis prescaler ikke er givet er den 1. Alternativt kan man finde top ved at sige 2^{bit} (hvor man så ikke behøver at lægge én til).

PWM Mode:

$$\text{Duty cycle} = 1 - (\text{OCRn} / \text{TOP})$$

$$\text{Frekvens} = f_{\text{cpu}} / (N * 2 * \text{TOP})$$

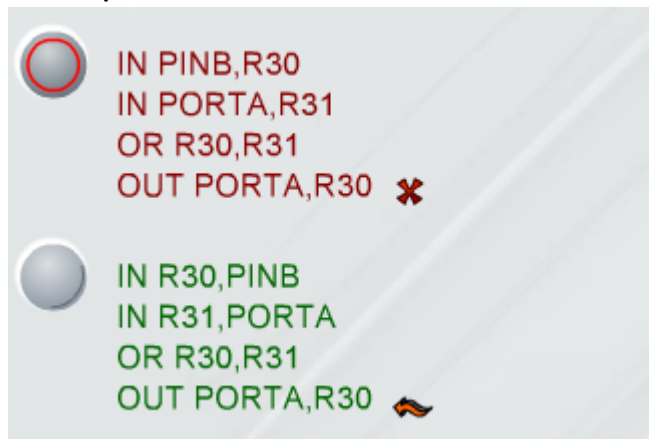
CTC Mode:

$$\text{Ben frekvens} = f_{\text{cpu}} / (2 * N * (1 + \text{OCRn}))$$

N er timerens prescaler-værdi

Denne C-kode statement compileres til Atmel AVR assembly kode:

PORTA |= PINB



En Mega2560 har en CPU clockfrekvens på X Mhz. Vi sætter Timer 2 i Normal Mode. Hvad er den længste tid, som vi kunne få mellem to efterfølgende Timer 2 overflows?

Timer 2 Max = 255+1

Timer 2 max prescaler = 1024

$$\frac{\max}{CPU_{freq}} * \maxPrescaler$$

Husk at justere efter outputdatatype.

OR/AND/XOR

OR eller |= bruges til at sætte en bit til 1. F.eks. $| (1 \ll 4)$

AND eller &= bruges til at sætte en bit til 0. F.eks. $\& \sim(1 \ll 4)$

AND eller &= kan bruges til at se om en bit er 0 eller 1. F.eks. $\& (1 \ll 4)$

XOR eller ^= bruges til at tage komplementet.

Hvilken timer interrupter først (prioritet)?

Svaret kan ses på **Interrupt vektorer** i INTERRUPT-SLIDES på side 13/14. Jo højere på listen = højere prioritet.

Hvilken metode er korrekt for at afsende/modtage i USART?

Svaret ses på **side 29 UART driver slides**. F.eks. for at se om USART er klar til at sende, så kræver det at UDREn = 1 på 5. bitplads (som tjekkes i et whileloop med en && condition).

Overflow per sekund:

$$Overflow_{per\ sekund} = \frac{CPU_{hz}}{\frac{Prescaler}{Timer_{max} + 1}}$$

Hvad gælder lige efter en reset:

Programtælleren er 0, og I/O registre indeholder kendte værdier.

Hvordan enables timer interrupts?

INTERRUPT SLIDES side 34+

Aflæs hvornår A/D konverter kan starte/slutte eller er startet/færdig

Kan findes på A/D CONVERSION SLIDES SIDE 32

Hvad kaldes seriel kommunikation der kan foregå i begge retninger, men kun i en retning ad gangen?

UART SLIDES side 15

Eksterne interrupt INT_X. Desuden skal den trigge rising edge

Løsningen kan findes under interrupts på side 28-29.

EIMSK skal være enabled (1) i den værdi der er X.

EICRA og EICRB skal være enabled (1) i forhold til beskrivelsen. I dette tilfælde skal ISC7_X og ISC7_X begge være 1 da rising edge skal være til.