

MIC aflevering 5 – PWM signal og Timer

Det første i koden er som altid vores setup afdeling. Her sættes PortA og PortB op til henholdsvis input/output. Derudover sættes indstillingerne for Potentiometeret op. Der sendes en byte som fortæller at den skal benytte Prescale 8 da med vores 1MHz derved får under 200KHz.

$$1\text{MHz} / 8 = 125 \text{ KHz}$$

Ligeledes sættes Referencen til ARef ved at sende den korrekte bit til ADMUX. Der sendes så der benyttes en ekstern 5V ref (potentiometeret) med venstre-justering. Der benyttes venstre-justering da det er de bit som har størst betydning.

Herefter sættes op så det kun er et display segment der lyser.

Vi kommer så til MAIN loopet.

Det første der gøres er at sætte PWM op til "Fast, Inverted". Inverted da kontrollen mhs. Hvor den hæver/sænker lys-niveauet er omvendt i forhold til opgavens krav.

Der sendes også til Timer0 hvilken Prescale den skal benytter, nemlig 1.

$$\text{PWM Frequency } 4\text{Kz} = 1\text{MHz}/(4\text{KHz}*256) = 1$$

Så begynder vores WAIT loop.

Den tjekker allerførst om den er klar til at læse input fra potentiometeret ved at tjekke om ADIF bitten er sat i ADCSR.

Hvis bitten ikke er sat hopper den tilbage til starten af WAIT. Når bittet er sat så gå den videre.

Først sættes ADIF til 0 igen. Dette gøres ved at sende et 1-tal til den, da det er sådan den er lavet.

Herefter indlæses de Lave bits af potentiometeret og derefter de Høje bits.

Det er lavet så man skal indlæse lave bits først selv om vi kun skal bruge de høje. Det er også derfor at de høje bits gemmes i samme register som de lave og derved overskrider dem.

Herfra sendes et out signal til OCR0 med indholdet af de høje bits. OCR0 er "Duty-cycle" af PWM signalet og kan derfor styrer lysintensiteten.

Til sidst startes WAIT loopet igen.

Frederik Mazur Andersen