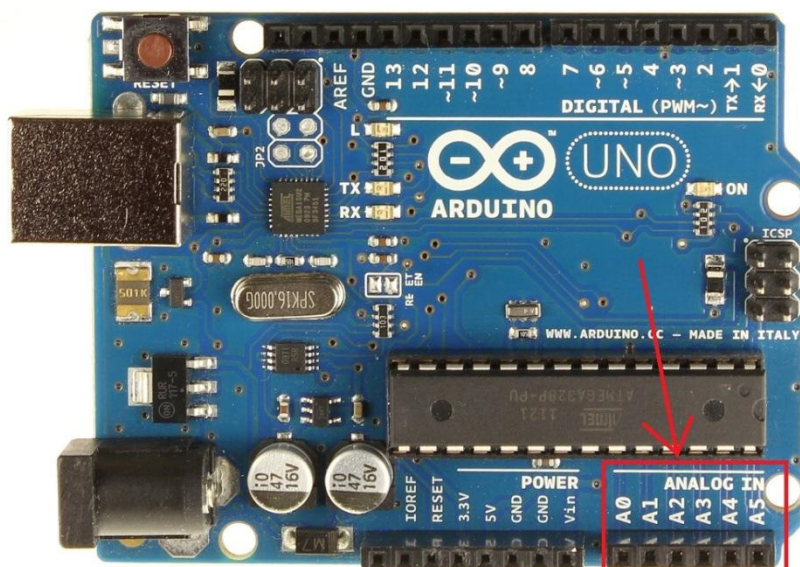


## A/D Converter

Hvis man har behov for at måle en analog spænding med en mikroprocessor, skal den analoge spænding konverteres til et digitalt signal. Til dette formål bruges en Analog til Digital Converter. Denne kaldes også for en A/D Converter eller bare en ADC.

### A/D Converter i Arduino

Der er indbygget seks A/D Convertere i Arduino-mikroprocessoren. De har betegnelsen: A0 til A5. Connectorerne til disse ses på Figur 1 markeret med en rød kasse.



Figur 1 Input til de seks A/D Convertere er A0 til A5

Hver A/D Converter (herefter kaldet for ADC) arbejder med 10 bit. Det betyder, at de har 10 bit til rådighed til at omsætte hvilken spænding der er blevet målt.

### Måle en analog spænding

Når man i programmet ønsker at måle en spænding, skrives:

```
// read the value from the sensor:  
sensorValue = analogRead(sensorPin);
```

Øverst oppe i koden har man defineret både `sensorValue` og `sensorPin` variableerne:

```
int sensorPin = A0;    // select the input pin for the ADC  
int sensorValue = 0;  // variable to store the value coming from the sensor
```

`sensorPin` har fået værdien A0, og det betyder, at når man laver en `analogRead(sensorPin)` læses spændingen på A0 benet. Resultatet gemmes i variabelen `sensorValue`. Den værdi der gemmes er IKKE spændingen i volt, men i stedet for et *step-nummer*. I følgende afsnit forklares det hvad dette betyder.

## 10 bit - hvad betyder det?

Som tidligere beskrevet, så arbejder ADC'erne i Arduino'en med 10 bit. Det betyder, at der er 10 bit til rådighed til at repræsentere den spænding der er målt.

Med 10 bit kan der repræsenteres  $2^{10} = 1024$  værdier eller step.

Hvis spændingen som ADC'en måler er 0 V, giver det en værdi på 0 (step 0). Hvis spændingen som ADC'en måler er 5 V (dvs. den maksimale spænding som ADC'en kan måle), giver det en værdi på 1023 (step 1023). Der er altså 1024 steps til at repræsentere spændingerne mellem 0 V og 5 V.

Man kan beregne hvor stor en spænding hvert step dækker over:

$$U_{step} = \frac{5V}{1024 \text{ step}} = 0,00488 \frac{V}{step} = 4,88 \frac{mV}{step}$$

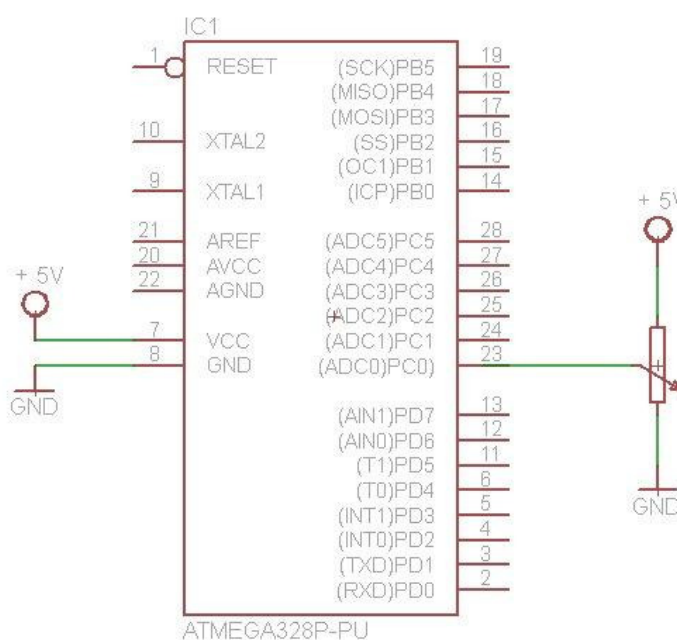
Hvis der måles en værdi på 688 (dvs. step 688), kan det beregnes, hvor stor en spænding der hermed er på ADC input:

$$U_{ADC} = n_{step} \cdot U_{step} = 688 \text{ step} \cdot 4,88 \frac{mV}{step} = 3,36V$$

Hermed kan der omregnes mellem spænding og step-nummer.

## Input til ADC

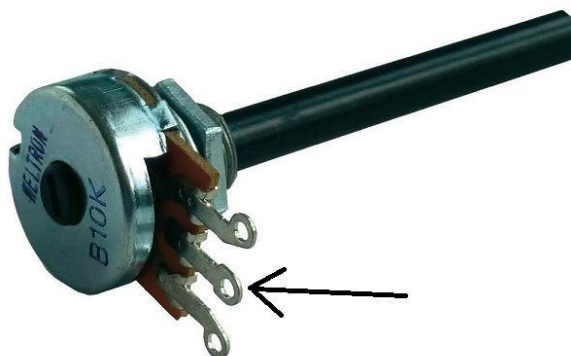
En simpel måde at teste hvordan ADC'en oversætter spændinger til step-værdier, er at forbinde et potentiometer til Arduino'en via A0. Dette ses på Figur 2 hvor diagrammet for dette ses.



Figur 2 Potentiometer tilsluttet A0 på Arduino processoren.

Potentiometret kan f.eks. være et 10 k $\Omega$  eller 100 k $\Omega$ . Figur 3 viser et potentiometer. Det ses, at potentiometret har tre ben. Måler man modstandsværdien mellem de to yderben er den 10 k $\Omega$  (hvis det er et 10 k $\Omega$  potentiometer). Modstandsværdien mellem midterbenet og yderbenene afhænger af hvor akslen står. Altid gælder det dog, at

Modstand mellem ben 1 og ben 2 (yderben og midterben) + modstand mellem ben 2 og ben 3 (midterben og andet yderben) = potentiometerets modstandsværdi.



Figur 3 Et potentiometer har tre ben; Når akslen drejes, ændres modstandsværdien mellem midterbenet og de to yderben.

Midterbenet forbindes til A0 på Arduino boardet. Det ene yderben forbindes til +5 V, og det andet yderben forbindes til GND.

Følgende program afvikles på Arduino'en:

```
int sensorPin = A0; // select the input pin for the potentiometer
int sensorValue = 0; // variable to store the value coming from the sensor

void setup() {
  //Initialize serial and wait for port to open:
  Serial.begin(9600);

  // prints title with ending line
  break Serial.println("ADC values
from A0");
}

void loop() {
  // read the value from the
  sensor: sensorValue =
  analogRead(sensorPin);

  // Print ADC value via Serial
  Monitor Serial.print("ADC value:
"); Serial.println(sensorValue);
}
```

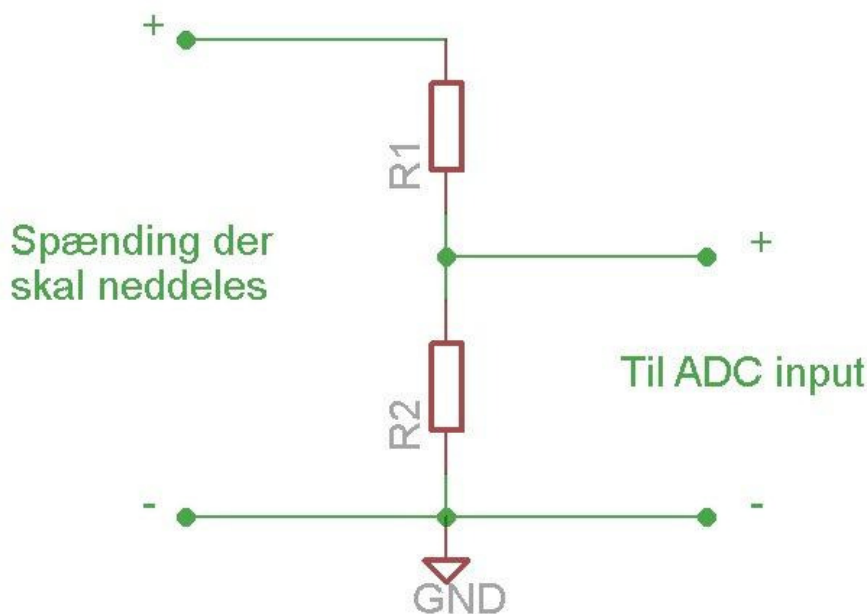
Programmet vil måle spændingen ind på A0 på Arduino'en, oversætte den til den tilhørende 8 bits værdi, og skrive resultatet ud på Serial Monitor.

## Spændingsdeler

Når man skal bruge ADC'en på Arduino-board'et, så skal man have styr på, hvordan spændingen på input signalet kan variere. Hvad er den laveste spænding og hvad er den højeste spænding?

Da Arduino-board'et arbejder med 5V, og mikroprocessoren på board'et dermed også er forsynet med en spænding på 5V, må man ikke have en inputspænding over 5V på ADC inputtet. Har man det, ødelægger man mikroprocessoren.

Hvis spændingen man ønsker at sende ind på ADC'en faktisk er større end de maksimale 5V, kan man lave en spændingsdeler, der neddelser inputspændingen. Figur 4 viser diagrammet over en spændingsdeler. Forholdet, som spændingen bliver neddelt i, bestemmes af modstandene R1 og R2.



Figur 4 Spændingsdeler, der kan neddele en spænding med en given faktor.

### Udregning af spændingsdelerforholdet

Man kan beregne den neddelte spænding, ud fra følgende formel:

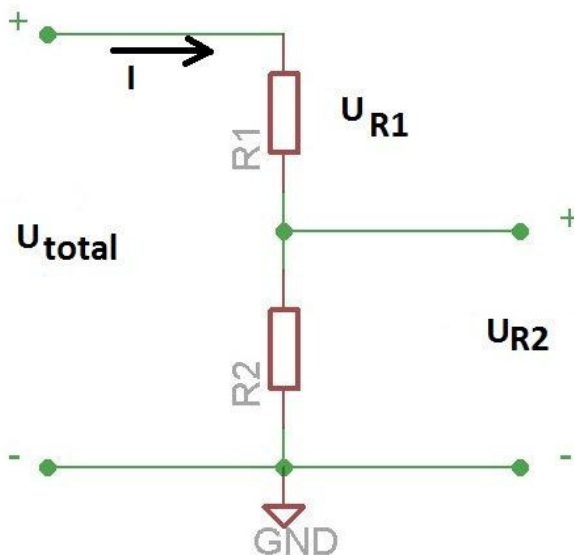
$$U_{Til\ ADC\ input} = \frac{U_{Spænding\ der\ skal\ neddeles} \cdot R_2}{R_1 + R_2}$$

Det viser sig, at hvis man vælger R1 og R2 ens (altså den samme modstandsværdi) så får man halveret sin spænding: Hvis man har en spænding på 10V, der skal neddeles til 5V, kan man indsætte følgende værdier i formlen:

$$U_{Til\ ADC\ input} = \frac{10V \cdot 10k\Omega}{10k\Omega + 10k\Omega} = 5V$$

## Udledning af spændingsdelerformlen

Denne formel fremkommer ud fra Ohms lov. Figur 5 viser spændingsdeleren, hvor der er sat "nye" labels på i forhold til diagrammet vist på Figur 4 (der står  $U_{total}$ ,  $U_{R1}$ ,  $U_{R2}$  og  $I$ )



Figur 5 Udledning af spændingsdelerformel

Spændingen  $U_{total}$  er summen af spændingerne over  $R1$  og  $R2$ :

$$U_{total} = U_{R1} + U_{R2}$$

Derefter udregnes strømmen  $I$ , der løber gennem  $R1$  og  $R2$ . Det gøres ud fra Ohms lov, hvor strømmen,  $I$ , isoleres:

$$U = R \cdot I \rightarrow I = \frac{U}{R}$$

Strømmen beregnes som spændingen over  $R1$  og  $R2$  ( $U_{total}$ ) delt med den samlede modstand ( $R1 + R2$ ):

$$I = \frac{U_{total}}{R_1 + R_2}$$

Spændingen over  $R2$  kan nu beregnes (igen ud fra Ohms lov):

$$U = R \cdot I$$

$$U_{R2} = R_2 \cdot I = R_2 \cdot \frac{U_{total}}{R_1 + R_2}$$

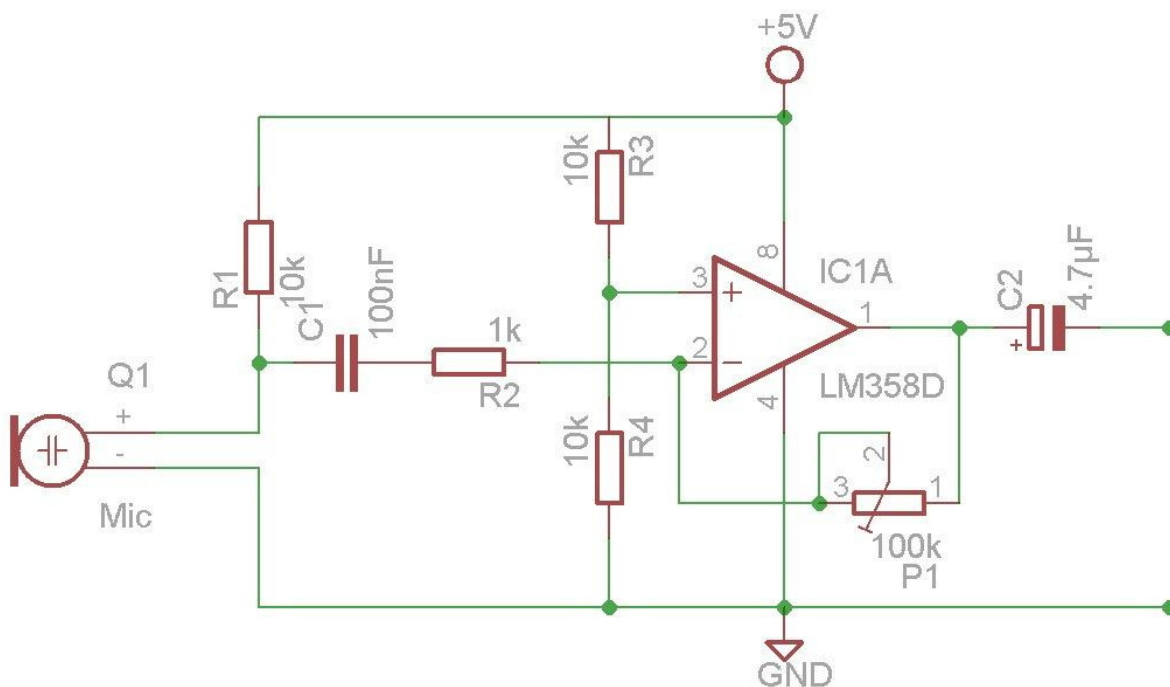
Dermed er spændingsdelerformlen udledt:

$$U_{R2} = \frac{U_{total} \cdot R_2}{R_1 + R_2}$$

## Signalforstærker (mikrofon forforstærker)

Hvis man har et analogt inputsignal som spændingsmæssigt ikke er ret stort, kan man forstærke signalet op, inden man sender det ind i ADC'en. Det kunne for eksempel være tilfældet hvis man vil koble en mikrofon til Arduino'en, og måle lyd. En lille kondensator mikrofon (også kaldet en electret mikrofon), giver kun et spændingsoutput på nogle få mV. Fra afsnittet "10 bit, hvad betyder det?" på side 2 konstateredes det, at hvert af de 1024 steps som ADC'en måler inputspændingen i, er 4,88 mV. Hvis inputspændingen kun er nogle få mV, så bliver opløsningen på det AD konverterede signal ikke stort nok til at være brugbar. Dermed er der brug for at forstærke mikrofonsignalet op, så det udnytter flest mulige af de 1024 steps.

En simpel forforstærker til en mikrofon ses på Figur 6. Forstærkningsgraden stilles på potentiometer P1.



Figur 6 Mikrofon forforstærker.