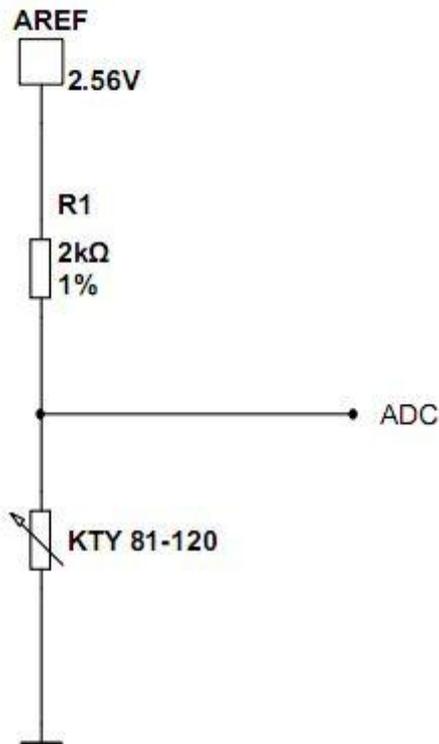


# Temperaturmessung mit dem ATmega8

Messaufbau:



Der ADC des ATmega8 stellt uns 6 Kanäle zur Analog/Digital Wandlung zur Verfügung und wird mit der internen Spannungsquelle bzw. externen Spannungsquelle betrieben. Wir arbeiten mit der internen Spannungsquelle des ADC, daher ergibt sich für uns eine Betriebsspannung von

$$U_{\text{AREF}} = 2,56 \text{ V.}$$

Die Berechnung zur Linearisierung des KTY-81 120 hat einen Vorwiderstand von 2kΩ ergeben um eine möglichst geringe Abweichung bei der Messung zu haben. Bei dem gewählten Vorwiderstand haben wir einen maximalen  $F_{\text{ABS}} = \pm 0,35 \text{ } ^\circ \text{C}$  bei  $-10^\circ \text{C}$ . Durch die Temperaturänderung entsteht eine sich ständig ändernde ADC Spannung die dem Mikrocontroller zur Umsetzung zugeführt wird. Um das Rauschen am Eingang des ADC's so gering wie möglich zu halten kann dem ADC noch ein Kondensator als Tiefpaß vorgeschaltet werden. Für unsere Messzwecke können wir auf diese Maßnahme verzichten.

Messbereich:	-30°C bis 50°C
Genauigkeit:	±0,5 ° C
Messart:	periodische Messung
Referenzspannung:	2,56 V bzw. 5 V
ADC Kanal:	0 ... 5
Auflösung:	10 Bit -> 1024 Stufen
Anzeige:	Temperatur in 0.5 ° C Schritten

## Berechnungen zur Linearisierung eines temperaturabhängigen Widerstandes

Bevor dem ADC die Spannung über dem KTY-81 120 zugeführt werden kann, muss die Spannung des Bauelementes linearisiert werden, da es ein nicht lineares Bauelement ist.

Wir haben dafür eine Excel Tabelle entwickelt, in der die Werte aus dem Datenblatt eingetragen werden müssen. So können wir direkt aus der Tabelle (siehe Abbildung 1) unsere Werte ablesen und für weitere Berechnungen verwenden.

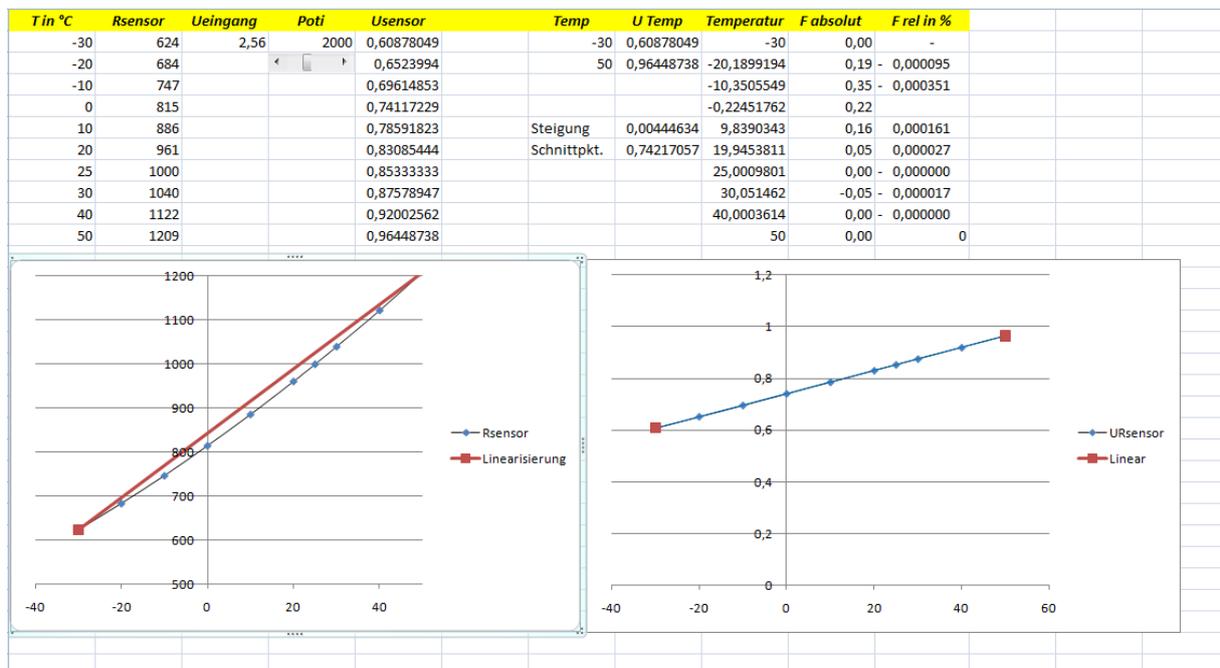


Abbildung 1

Die Grafik in Abbildung 2 zeigt den Verlauf des KTY (schwarze Kennlinie) in Abhängigkeit der Temperatur. Hier wird schnell ersichtlich, dass eine solche Kennlinie zu falschen Messergebnissen führen würde. Die Grüne Kennlinie zeigt den idealen Verlauf an. Die Linearisierung macht nun nichts anderes, als diesen Verlauf zu erreichen. Man beachte aber, dass lediglich die Spannung des Bauteils linearisiert werden kann. Da das Bauteil an sich unveränderbar ist.

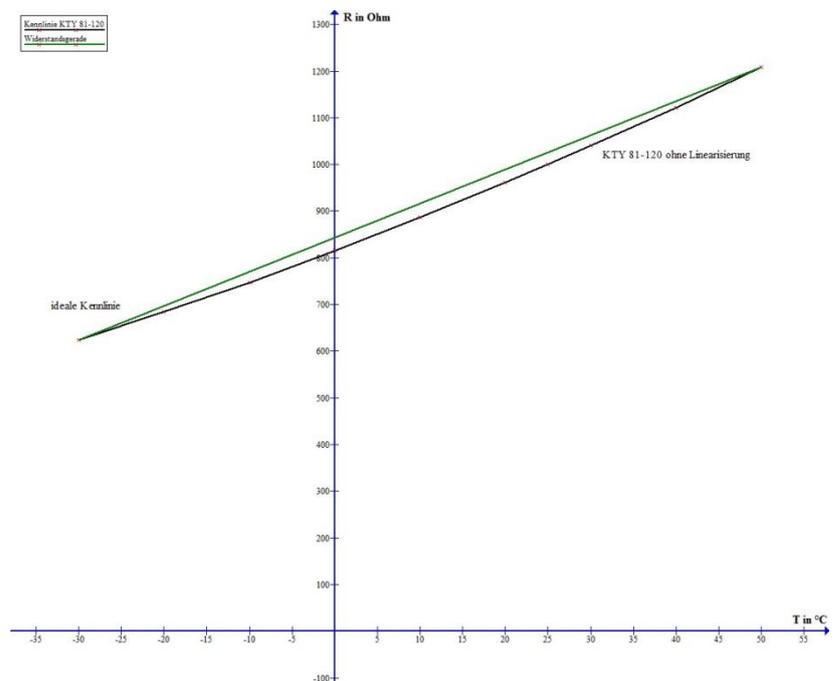


Abbildung 2

Aus dem Datenblatt des ATmega8 kann man entnehmen das der ADC ein 10 Bit Register zur Verfügung stellt, mit dem die Messungen umgewandelt werden können. Somit stehen uns 1024 Stufen zur Verfügung. Durch einen Vorteiler kann man diese Stufenzahl reduzieren. Da wir aber ein möglichst genaues Messergebnis haben wollen, verwenden wir die volle Bitbreite. Um das Messergebnis möglichst genau zu haben, könnte noch ein Operationsverstärker zum Einsatz kommen, der das Messsignal weiter spreizt und somit eine höhere Empfindlichkeit hat. Aus Einfachheit unsere Messung haben wir darauf verzichtet.

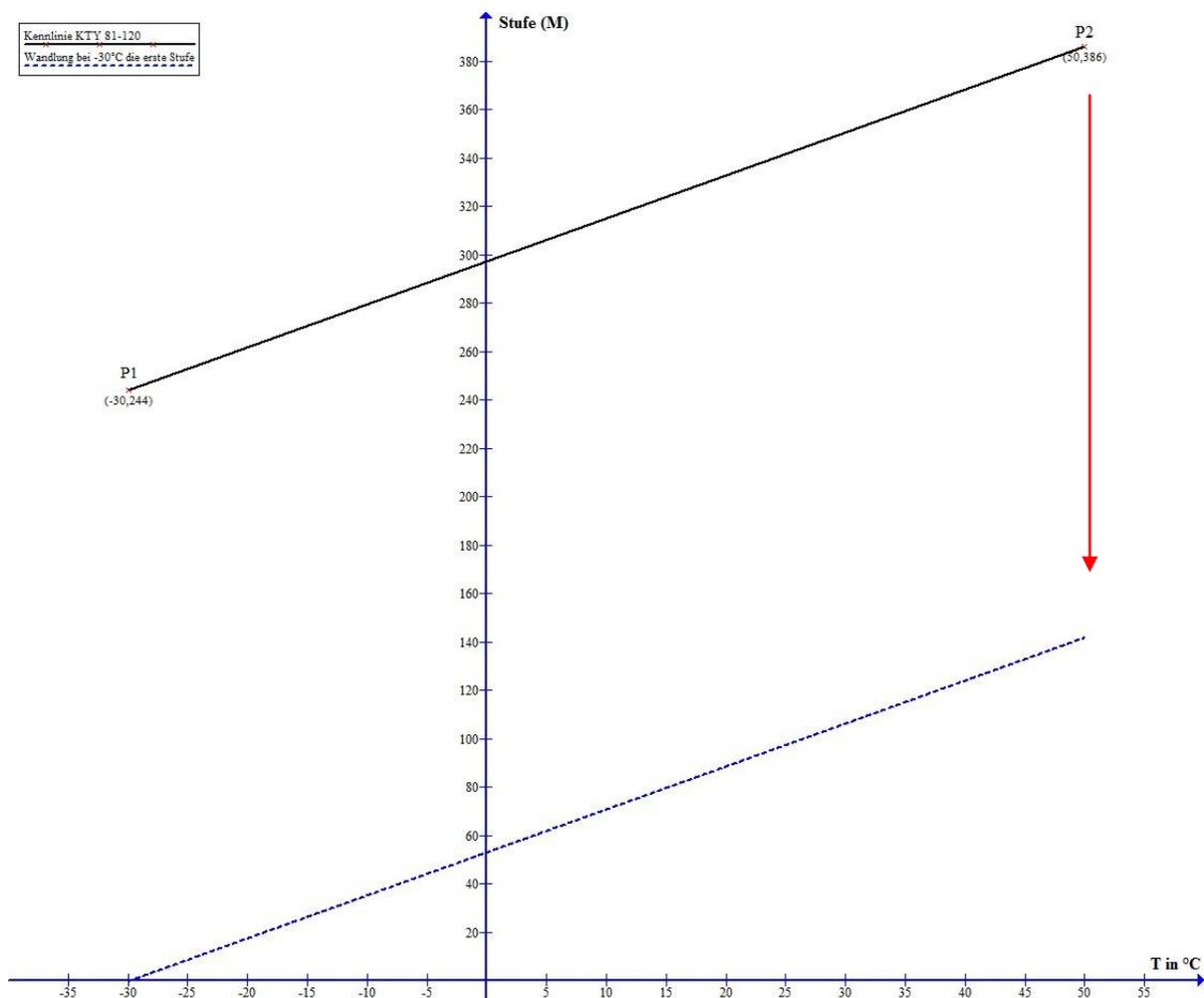
Formel zur Berechnung der Stufe:

$$ADC\ Wert = \frac{U_{ADC}}{U_{AREF}} * 1024$$

Mit Hilfe unserer Excel Tabelle können wir nun die Stufen bei den Punkten -30°C und 50°C berechnen. Da sich unsere Messung nur in diesem Bereich zwischen -30°C und 50°C erstrecken soll, werden wir für unsere Berechnung die erste Stufe bei -30°C beginnen lassen.

$U_{0°C} = 0,742\ V \rightarrow ADC = 53\ Stufe$

$$\rightarrow f(x) = 53 + \frac{(0,964\ V - 0,608\ V)}{50°C - (-30°C)} * ADC\ Wert$$



## Umsetzung in Bascom

---

```
' Der ADC(5) ist mit dem ADC Potenzial verbunden
' Der Messwert wird in der Variable W (Integer, da nur 0-1023) gespeichert
' Die errechnete Temperatur in der Variable T (Single, da Flieskommarechnung)
' Steigung wird als Konstante const m = 1.7785 gerechnet (aus Berechnung)
'T1 und T2 sind Hilfsvariablen, da wir den Messwert auf 0.5 ° C genau haben wollen
'Code erstellt durch Markus Schultheis © 2010
```

---

```
Dim W as Integer
Dim T as Single, T1 as Single, T2 as Integer
const M = 1.77853446
Config ADC = Single, Prescaler = Auto, Reference = Internal
Declare Sub Temperaturmessung()
```

```
Sub Temperaturmessung():
```

```
    Wait 2                                'Alle 2 Sekunden eine Messung
    Start ADC
    W = Getadc(5)
    Stop ADC
    W = W - 244                            'Bei -30°C = Stufe 1
    T = W / M
    T = T - 30
    T1 = T - T2
    T2 = T
    Cls
    Locate 2 , 1
    LCD "Temp:"
```

```
If W < 52 then                            'Wenn Temperatur < 0°C dann - T ° C
    If T2 = -1 Then                        'in 0,5°C Schritten
        T2 = 30 - T
        T2 = T2 + 2
        LCD "-"; T2 ; ".5" ; Chr(&H00df) ; " C"
    Else
        T2 = 30 - T
        T2 = T2 + 2
        LCD "-"; T2 ; Chr(&H00df) ; " C"
    End If
```

```
End if
```

```
IF W = 52 Then                            'Wenn Temperatur = 0°C dann anzeigen
    LCD "0" ; Chr(&H00df) ; " C"          'SPEZIALFALL!
End if
```

```

If W > 52 then
    T2 = T1 + 0.5
    If T2 = 1 then
        T2 = T
        LCD T2 ; ".5" ; Chr(&H00df) ; " C"
    Else
        T2 = T
        LCD T2 ; Chr(&H00df) ; " C"
    End If
End If
End if

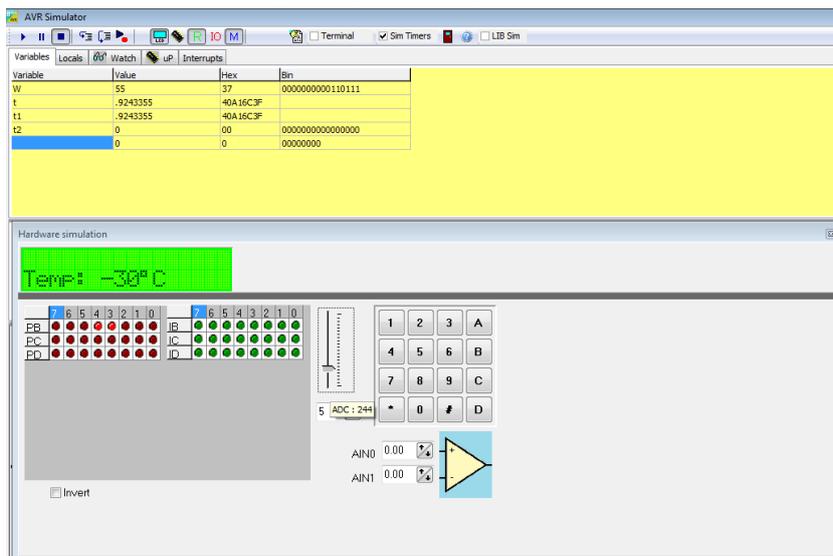
End Sub

```

'Wenn Temperatur > 0°C dann +T° C  
'in 0,5°C Schritten

## Die Simulationsumgebung

Die Simulation der Daten zeigt, dass die errechneten Werte korrekt angezeigt werden. Dies ist ein sehr gutes Hilfsmittel um auch negative, bzw. beliebige Werte einzugeben und anzeigen zu lassen ohne sich in dieser Umgebung zu bewegen.



Die Grafiken zeigen jeweils die beiden Extremwerte. Die Forderung nach der Maximalabweichung werden somit eingehalten.

