

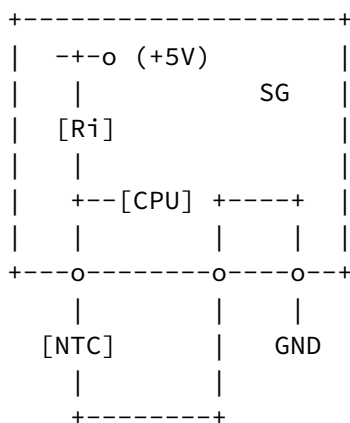
**Thema:** Temperatursensor (NTC) - temperaturabhängigen Widerstand**Keywords**

- ✓ PYTHON ANACONDA <sup>1</sup>
- ✓ Jupyter Notebook / matplotlib / Erstellung von Diagramme
- ✓ ChatGPT <sup>2</sup>
- ✓ Temperatursensor (NTC)

Kennlinie des temperaturabhängigen Widerstands bestimmen

- NTC: Je höher die Temperatur wird, desto niedriger ist ihr Widerstandswert.
- PTC: Je höher die Temperatur wird, desto größer ist ihr Widerstandswert.

# Schaltplan: Temperatursensor NTC + SG



# Quellcode in Python

```

import math
import matplotlib
matplotlib.rcParams['text.usetex'] = True # Latex code
import matplotlib.pyplot as plt

# geg
Io = 0.0025 # Strom (A) (gemessen) HINWEIS: Stecker vom SG trennen
Uo = 5      # Referenzspannung (V) (gemessen) HINWEIS: Stecker vom SG trennen
Ri = Uo/Io  # Innenwiderstand (Ohm) von SG berechnet
# NTC: Je höher die Temperatur wird, desto niedriger ist ihr Widerstandswert.
# NTC Warm: R_Sensor = 222 Ohm => U_Sensor = 0,5 V
# NTC Kalt: R_Sensor = 18000 Ohm => U_Sensor = 4,5 V
# U_Sensor = 0.5 # Spannung (V) am Sensor_NTC gemessen HINWEIS: Zündung ein,
#             ⇨ Motor an
# Kennlinie des temperaturabhängigen Widerstands (NTC) bestimmen
# R_Sensor = U_Sensor*Ri/(Uo-U_Sensor)

#zahlen = [x/10 for x in range(0, 50)] # Zahlen von 0,0 bis 4,9 erstellen

```

<sup>1</sup><https://docs.anaconda.com/free/anaconda/>

<sup>2</sup><https://chat.openai.com/>

```
#liste = Liste[5:46] # Auswahl Zahlen von 0,5 bis 4,5
X = [x/10 for x in range(2, 48)] # Zahlen von 0,5 bis 4,5
Y1 = [U_Sensor*Ri/(Uo-U_Sensor) for U_Sensor in X]

# Latex
# Farbe: Orange #F28C64 grau2 #B2B2B2 rot5 #A71916 blau5 #0D468E
plt.plot(X,Y1, label='NTC', color="#0D468E")#blau5
plt.title(r'Kennlinie NTC',fontsize=12)
plt.xlabel(r'\textbf{U_Sensor (V)}')
plt.ylabel(r'\textbf{R_Sensor ($\Omega$)}')
# x-y Wertebereich bestimmen
plt.xlim(0.0,+5.0)
plt.ylim(0,+30000)
plt.legend()
plt.savefig("Diag_Tempersensor.svg")# SVG-Vektorgrafik
plt.show()
```

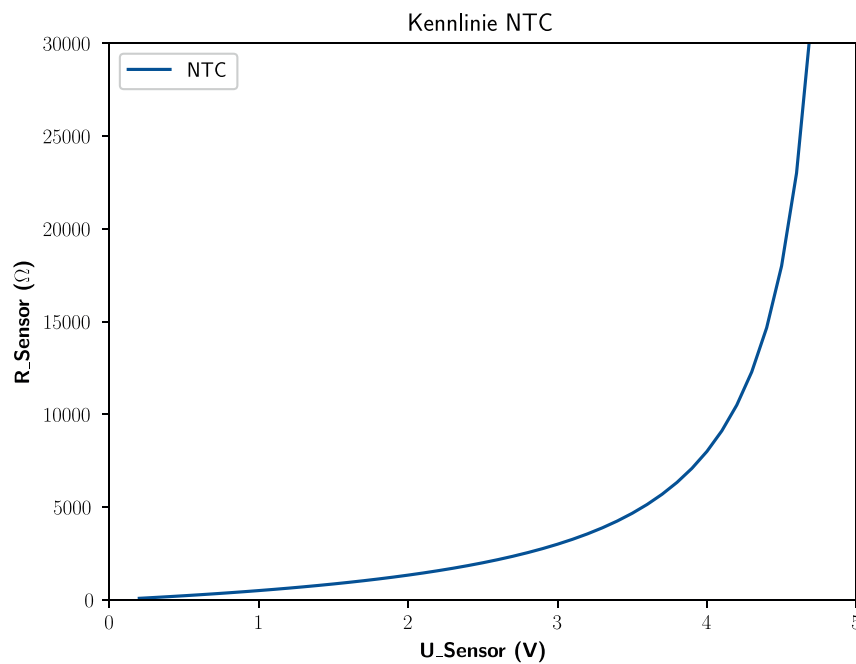


Abb. 1: Kennlinie des temperaturabhängigen Widerstands (NTC)