

Matthias von Wachter

Professur Elektronische Bauelemente und Integrierte Schaltungen

# Extraktion von Modellparametern für ein Kompaktmodell am Beispiel einer PN Diode

Hauptseminar Mikro- und Nanoelektronik // Dresden, 10. Juni 2020

# Outline

Einführung

Messungen

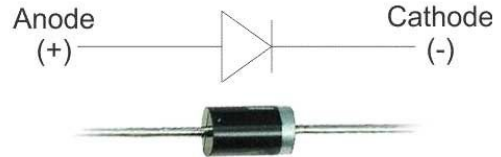
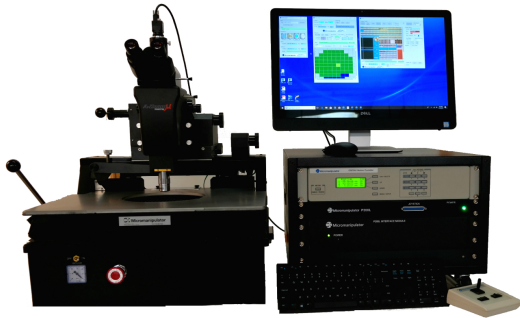
Modell

Implementierung in Python

Zusammenfassung und Ausblick

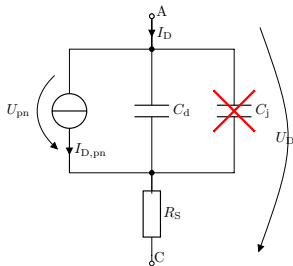
# Einführung

# Automated Wafer Characterization



The Micromanipulator Company [1]

# Grundgleichungen



Shockley Diodengleichung :

$$I_{D,pn} = I_S \left[ \exp\left(\frac{U_{pn}}{mU_T}\right) - 1 \right]$$

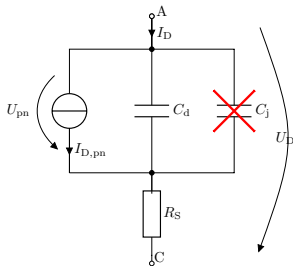
$I_S$ : Sättigungsstrom

$m$ : Idealitätsfaktor

$U_T = \frac{kT}{q}$ : Temperaturspannung

Spannungsgesteuerte Stromquelle  $I_{D,pn}$

# Grundgleichungen



Shockley Diodengleichung :

$$I_{D,pn} = I_S \left[ \exp\left(\frac{U_{pn}}{mU_T}\right) - 1 \right]$$

$I_S$ : Sättigungsstrom

$m$ : Idealitätsfaktor

$U_T = \frac{kT}{q}$ : Temperaturspannung

Spannungsgesteuerte Stromquelle  $I_{D,pn}$

Diffusionskapazität  $C_d$ :

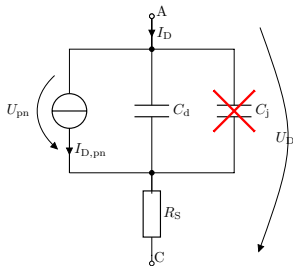
Diodenkapazität  $C_D$  = Diffusionskapazität

$C_d$  + Sperrschichtkapazität  ~~$C_j$~~

$$C_d = \tau_T * \frac{I_D}{U_T}$$

$\tau_T$ : Transferzeit

# Grundgleichungen



Shockley Diodengleichung :

$$I_{D,pn} = I_S \left[ \exp\left(\frac{U_{pn}}{mU_T}\right) - 1 \right]$$

$I_S$ : Sättigungsstrom

$m$ : Idealitätsfaktor

$U_T = \frac{kT}{q}$ : Temperaturspannung

Spannungsgesteuerte Stromquelle  $I_{D,pn}$

Diffusionskapazität  $C_d$ :

Diodenkapazität  $C_D$  = Diffusionskapazität

$C_d$  + Sperrschichtkapazität  ~~$C_j$~~

$$C_d = \tau_T * \frac{I_D}{U_T}$$

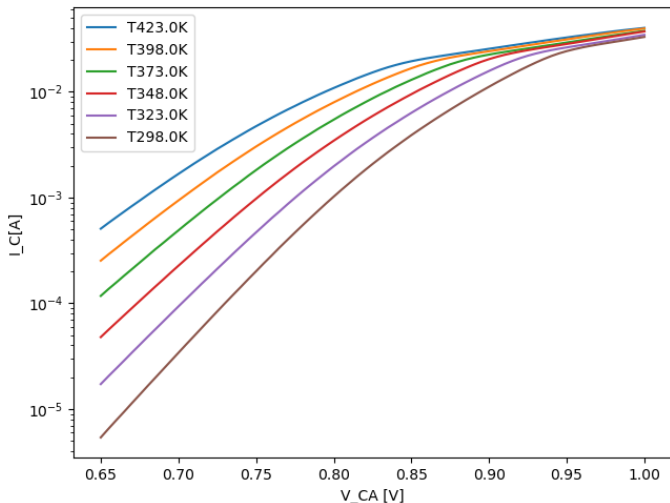
Bahnwiderstand  $R_S$ .

$\tau_T$ : Transferzeit

# Messungen

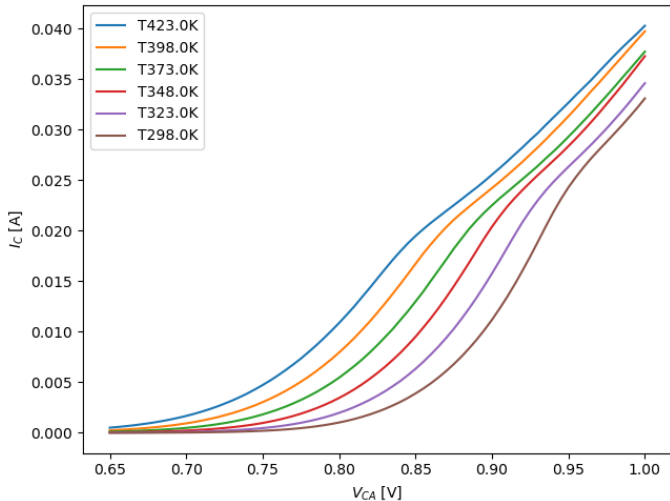


# Messungen: Diodenstrom $I_C$ logarithmisch



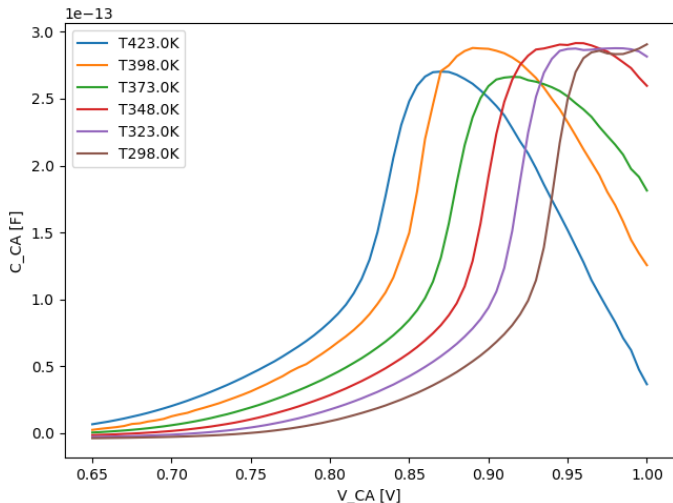
- Messbereich  
 $0.65 \text{ V} \leq V_{CA} \leq 1.00 \text{ V}$
- Rein exponentieller Verlauf  
für  $V_{CA} \leq 0.75 \text{ V}$
- Positiver  
Temperaturkoeffizient

# Messungen: Diodenstrom $I_C$ linear



- Linear steigend für  $V_{CA} > 0.9 \text{ V}$

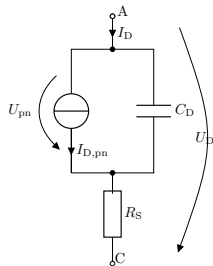
# Messungen: Diodenkapazität $C_{CA}$



- Ähnlicher Verlauf wie  $I_C$  für  $V_{CA} < 0.85$  V
- Nicht mehr steigend für höhere Spannungen
- Ungeklärter Temperatureffekt!

# Modell

# Grundgleichungen (Wiederholung)



Shockley Diodengleichung :

$$I_{D,pn} = I_S \left[ \exp\left(\frac{U_{pn}}{mU_T}\right) - 1 \right]$$

$I_S$ : Sättigungsstrom

$m$ : Idealitätsfaktor

Spannungsgesteuerte Stromquelle  $I_{D,pn}$

Diffusionskapazität  $C_d$ :

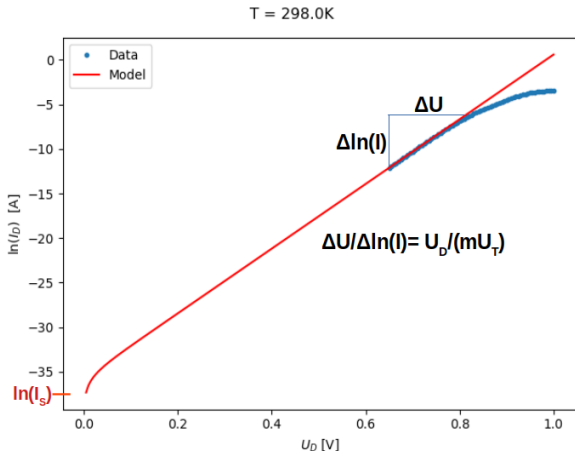
Diodenkapazität  $C_D \approx$  Diffusionskapazität  $C_d$

$$C_d \approx \tau_T * \frac{I_D}{U_T}$$

Bahnwiderstand  $R_S$ .

$\tau_T$ : Transferzeit

# Modell: Exponentieller Diodenstrom $I_D$



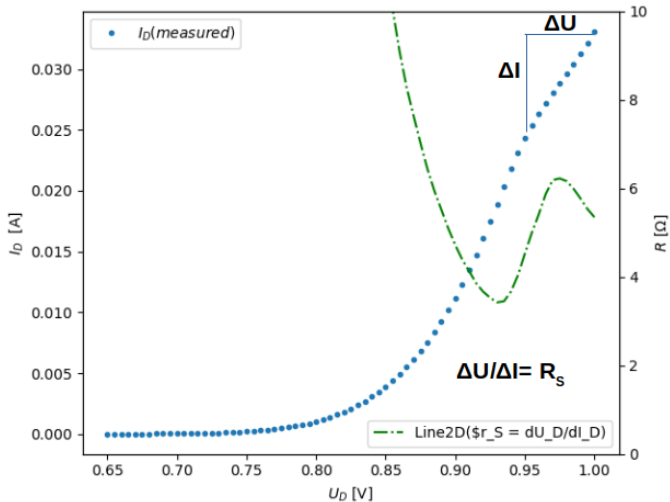
Linearisiere Diodengleichung

$$I_D = I_S \left[ \exp\left(\frac{U_D}{m U_T}\right) - 1 \right] \text{ für}$$

$$\frac{U_D}{m U_T} > 10 \quad (\Rightarrow U_D > 0.25 \text{ V}) :$$

$$\ln(I_D) = \ln(I_S) + \frac{U_D}{m U_T} I_D$$

# Modell: Bahnwiderstand $R_S$

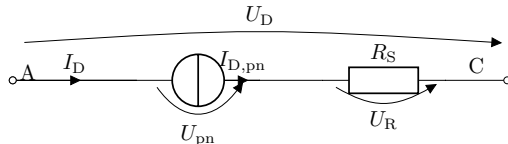


$$R_S = \frac{\Delta U_D}{\Delta I_D} =$$

$$\frac{1.0\text{V} - 0.9\text{V}}{I_D(U_D = 1.0\text{V}) - I_D(U_D = 0.9\text{V})}$$

Alternative:  $R_S$  als  
Durchschnitt des  
differentiellen Widerstands  $r_D$

# DC Modell: Diode mit ohmschem Widerstand



$$U_D(I_D) = U_{pn} + U_R = \left[ \ln \left[ \frac{I_D + I_S}{I_S} \right] * U_T + I_D * I_S \right] + I_D * R_S$$

Analytische Lösung mit Lambertscher W-Funktion:

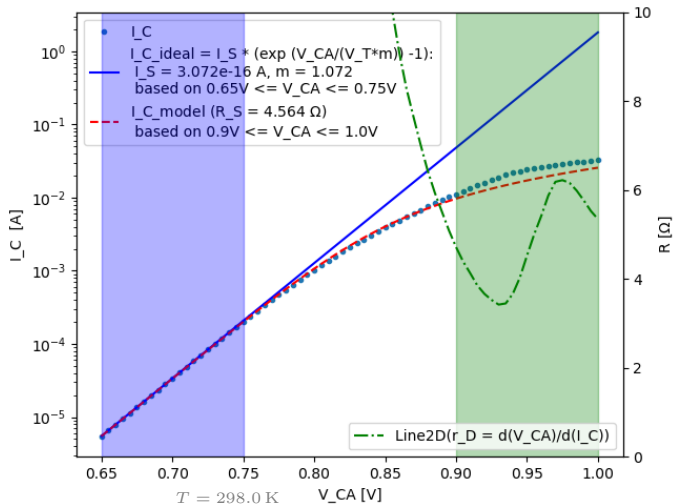
$$f(x) = x \exp(x) \Leftrightarrow W(x) = f^{-1}(x)$$

$$I_D(U_D) = \Re \left[ W \left[ I_S R_S * \exp \left[ \frac{I_S R_S + U_D}{U_T} \right] * \frac{1}{U_T} \right) * \frac{U_T}{R_S} - I_S R_S \right]$$

$$I_D = I_S \left[ \exp\left(\frac{U_D}{m U_T}\right) - 1 \right]$$



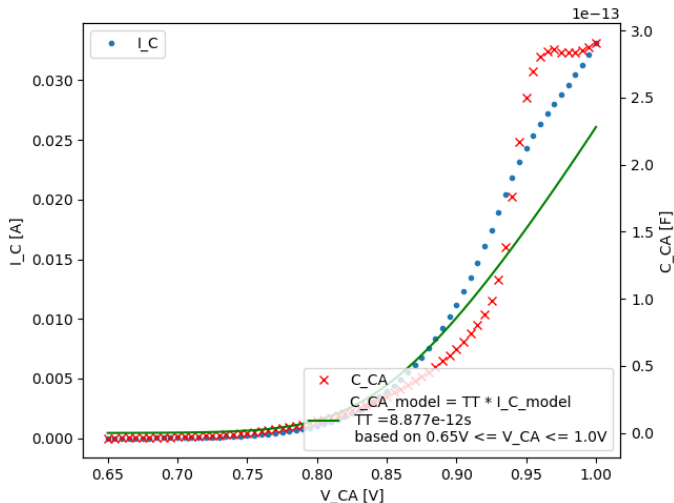
# DC Modell: Diode mit ohmschem Widerstand



Ideale Diode

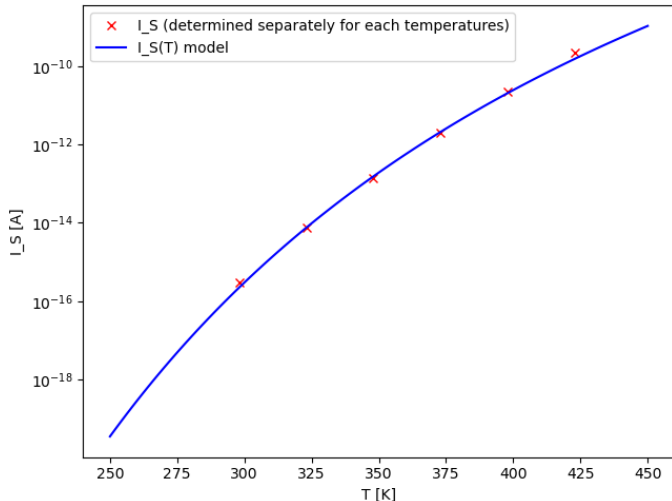
Ohmscher Widerstand

# Modell: Kapazität $C_D$



$$C_D \cong C_d = \tau_T * \frac{I_D}{U_T}$$

# Modell: Sättigungsstrom $I_S(T)$



Vereinfachtes HICUM-Modell  
für eine interne  
Basis-Emitter-Diode:

$$I_{S, \text{BEi}}(T) \approx I_{S, \text{simplified}}(T) =$$

$$I_{S0} * \exp \left[ \frac{E_g}{U_T} * \frac{T}{T_0} \right]$$

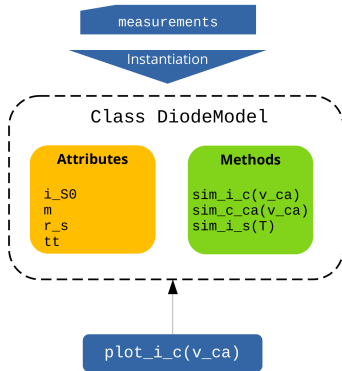
$$E_g = E_{\text{bandgap, silicon}} = 1.17 \text{ V}$$

$$T_0 = 300.15 \text{ K}$$

$$U_T = \frac{kT}{q}$$

# Implementierung in Python

# Objektorientierter Ansatz



## Klasse DiodeModel:

- Initialisierung mit Messdaten
- $\Rightarrow$  Berechnen und Speichern der Modellparameter
- Funktionen zur Simulation von Messgrößen

# Verwendete Python Module

- Numpy für numerische Berechnungen
- Scipy für Curve Fitting:
  - `scipy.optimize.curve_fit()` basiert auf Methode der kleinsten Quadrate
- Matplotlib für Diagramme



# Zusammenfassung und Ausblick

# Zusammenfassung

- DC: gute Genauigkeit für Reihenschaltung von idealer Diode und ohmscher Widerstand



# Zusammenfassung

- DC: gute Genauigkeit für Reihenschaltung von idealer Diode und ohmscher Widerstand
- AC: Bei höheren Spannungen/Temperaturen gilt nicht mehr
$$C_D \cong C_d = \tau_T * \frac{I_D}{U_T}$$

# Zusammenfassung

- DC: gute Genauigkeit für Reihenschaltung von idealer Diode und ohmscher Widerstand
- AC: Bei höheren Spannungen/Temperaturen gilt nicht mehr
$$C_D \cong C_d = \tau_T * \frac{I_D}{U_T}$$
- Open-Source Python Pakete NumPy, SciPy und matplotlib ermöglichen effiziente Datenanalyse und Modellierung

# Ausblick

- Ungeklärte Hochtemperatur-Anomalie der Kapazität der Basis-Emitter Diode

# Ausblick

- Ungeklärte Hochtemperatur-Anomalie der Kapazität der Basis-Emitter Diode
- Weiterentwicklung der Software:
  - Test Driven Development
  - Continuous Integration
  - Effizientere Nutzung von NumPy Funktionen






# Ausblick

- Ungeklärte Hochtemperatur-Anomalie der Kapazität der Basis-Emitter Diode
- Weiterentwicklung der Software:
  - Test Driven Development
  - Continuous Integration
  - Effizientere Nutzung von NumPy Funktionen
- Gitlab Repository:  
[https://gitlab.hrz.tu-chemnitz.de/s1760196--tu-dresden.de/diode\\_model\\_parameters.git](https://gitlab.hrz.tu-chemnitz.de/s1760196--tu-dresden.de/diode_model_parameters.git)

# Fragen?

# Anhang

# Literatur I

-  The Micromanipulator Company. *P300L Semi-Automatic Probe Station*. P200LSemiAutomaticProber.jpg. URL: <https://www.micromanipulator.com/products/probe-stations/p300l-300mm/>.
-  Elprocus. *Rectifier Diode Circuit Working And Its Applications*. URL: <https://www.elprocus.com/rectifier-diode-working-applications/>.
-  Holger Göbel. *Einführung in die Halbleiter-Schaltungstechnik*. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2011.
-  Reinhold Paul. *Halbleiterdioden Grundlagen und Anwendung*. Berlin: Verl. Technik, 1976.
-  Michael Schroter. *HICUM - A scalable physics-based compact bipolar transistor model*. 2000.

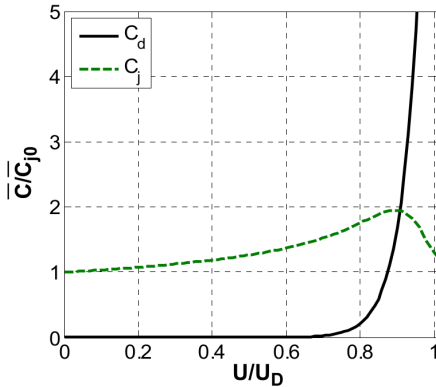


# Literatur II

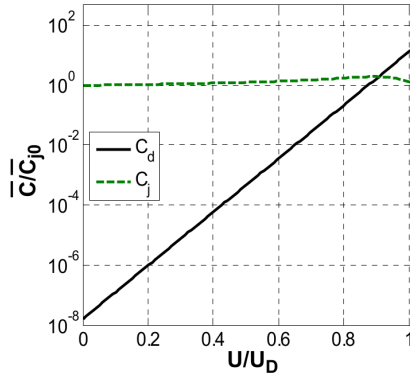


Michael Schröter. *Elektronische Bauelemente. Vorlesungsmanuskript.*

# p-n Kapazitäten im Bauelemente Skript



(a)

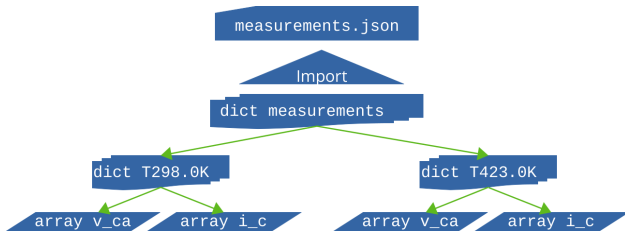


(b)

Fig. 3.4.0/3: Diffusionskapazität als Funktion der angelegten Spannung: (a) lineare Darstellung; (b) halblogarithmische Darstellung.

Elektronische  
Bauelemente.  
Schröter.  
[6, S. 63]

# Daten Import und Export



```
with open('data.json', 'r') as f:
    measurements = json.load(f)
for measurement in measurements.items():
    v_ca_a = measurement[1]['V_CA'][:]
    i_c_a = # ...

model = DiodeModel(v_ca_a, i_c_a, c_ca_a, T)

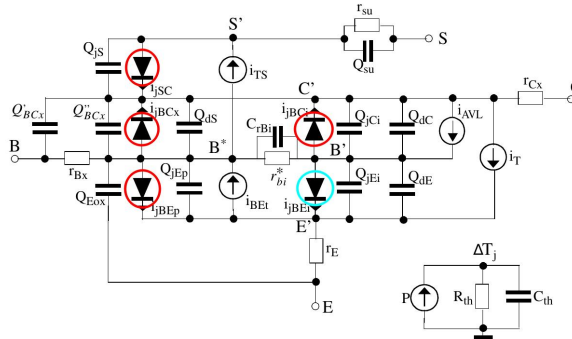
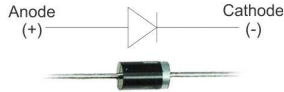
with open('model_parameters.json', 'w') as f:
    json.dump(base_model.params, f)
```

Import (Messdaten) und Export (Modellparameter) im JSON Format

# Berechnung der Modellparameter

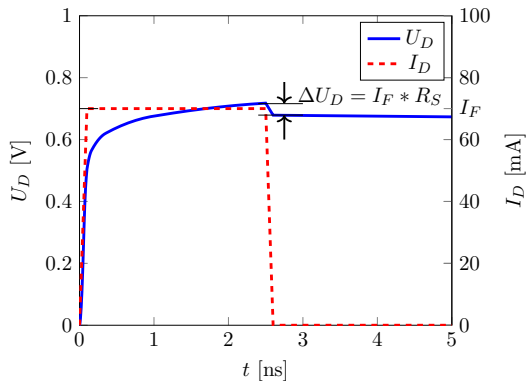
```
def diode_model_params_isotherm(v_ca_a, i_c_a , c_ca_a , T):  
    # Curve fit C_CA to I_C curve  
    tt = diode_capacitance_model(v_ca_a, i_c_a , c_ca_a)  
    model_params = {'TT': tt}  
    # ...  
    return model_params  
  
def diode_capacitance_model(v_ca_a , i_c_a , c_ca_a):  
    p_opt, pcov = scipy.optimize.curve_fit(diode_cap_TT_eq, i_c_a, c_ca_a)  
    tt = p_opt[0]    # Transit time  
    return tt  
  
def diode_cap_TT_eq(i_c, tt):  
    c_ca = tt * i_c  
    return c_ca
```

# Dioden im HICUM Modell



Elprocus [2]  
HICUM Model [5]

# Bestimmung des Widerstands durch Stromsprungantwort



$$R_S = \frac{\Delta U_D}{I_F}$$

# Probleme und Lösungen

- Numerische Instabilität bei Curve Fitting auf eine Exponentialfunktion  
⇒ Linearisiere physikalische Gleichung und Eingangsparameter mit Logarithmus
- Kein ideales Verhalten der Eingangsdaten  
⇒ Nutze Datenbereiche mit (fast) rein exponentiellem/linearem Anstieg für Modell
- Bestimmung des ohmschen Widerstands durch Stromsprung nicht möglich  
⇒ Verwende Steigung des Diodenstroms bei höherer Spannung