



Hochschule Konstanz
Technik, Wirtschaft und Gestaltung

Signale, Systeme und Sensoren

Aufbau, Kalibrierung und Einsatz eines einfachen Entfernungsmessers

T. Schoch, L. Stratmann

Konstanz, 10. April 2019

Zusammenfassung (Abstract)

Thema:	Aufbau, Kalibrierung und Einsatz eines einfachen Entfernungsmessers	
Autoren:	T. Schoch	tobias.schoch@htwg-konstanz.de
	L. Stratmann	luca.stratmann@htwg-konstanz.de
Betreuer:	Prof. Dr. Matthias O. Franz	mfranz@htwg-konstanz.de
	Jürgen Keppler	juergen.keppler@htwg-konstanz.de
	Mert Zeybek	me431zey@htwg-konstanz.de

In dem Versuch haben wir einen Entfernungsmesser dazu verwendet, um die bereits in der Vorlesung behandelten Vorgehensweisen zum Thema Kalibrierung, Fehlerbehandlung und Fehlerrechnung anzuwenden. Der Distanzsensor der Marke "Sharp" benutzt für das Triangulationsprinzip Infrarot-LEDs mit einer Linse. Diese geben Lichtstrahlen von sich, um dann wiederum reflektiert zu werden und durch die zweite Linse zu gelangen. Je nachdem wo der Lichtstrahl auftrifft, wandelt der Signalprozessor die Leitfähigkeit in eine Spannung um. Da das Ausgangssignal anti-proportional ist, wird mit der zunehmenden Entfernung das Ausgangssignal kleiner. Die Entfernungen liegen zwischen 10cm und 70cm. Durch ein Oszilloskop können wir den Spannungsverlauf des Abstandssensors überprüfen.

Inhaltsverzeichnis

Abbildungsverzeichnis	IV
Tabellenverzeichnis	V
Listingverzeichnis	VI
1 Einleitung	1
2 Versuch 1: Ermittlung der Kennlinie des Abstandssensors	2
2.1 Fragestellung, Messprinzip, Aufbau, Messmittel	2
2.2 Messwerte	3
2.3 Auswertung	4
2.4 Interpretation	4
3 Versuch 2: Modellierung der Kennlinie durch lineare Regression	5
3.1 Fragestellung, Messprinzip, Aufbau, Messmittel	5
3.2 Messwerte	5
3.3 Auswertung	5
3.4 Interpretation	5
4 Versuch 3: Flächenmessung mit Fehlerrechnung	6
4.1 Fragestellung, Messprinzip, Aufbau, Messmittel	6
4.2 Messwerte	6
4.3 Auswertung	6
4.4 Interpretation	6
Anhang	7
A.1 Quellcode	7
A.1.1 Quellcode Versuch 1: Messung Kalibrierung	7

A.1.2	Quellcode Versuch 2	8
A.1.3	Quellcode Versuch 3	8
A.2	Messergebnisse	8

Abbildungsverzeichnis

Tabellenverzeichnis

2.1	Messwerte Kalibrierung	3
-----	----------------------------------	---

Listingverzeichnis

5.1	Messung	7
-----	-------------------	---

1

Einleitung

[?] [?]

2

Versuch 1: Ermittlung der Kennlinie des Abstandssensors

2.1 Fragestellung, Messprinzip, Aufbau, Messmittel

Im ersten Versuch werden wir die Kennlinie des Abstandssensors ermitteln. Für den Aufbau des Projektes verbinden den Abstandssensor an 'Output 3' des Labornetzgerätes 'EA-PS 2342-06 B' durch Ground (-) und dem 5V Anschluss (+) angeschlossen. Der Abstandssensor lautet 'GP2Y0A21YK0F' und wurde von der Firma 'Sharp' entworfen. Das Netzgerät wird auf 5V Gleichspannung eingestellt. Das Oszilloskop von 'Tektronix' mit dem Namen 'TDS 2022B' wird an den Abstandssensor mit Ground(-), sowie an den Signalausgang angeschlossen. Dies wird im Oszilloskop mit einem Adapter an Channel1 verbunden. Nachdem das Oszilloskop richtig eingestellt wurde, haben wir es mit dem PC verbunden. Über ein Programm konnten wir so das Oszilloskop mit dem Computer verbinden. Ein Programm hat uns geholfen den aktuellen Bildschirm des Oszilloskopes auf dem Bildschirm zu empfangen. Zudem kann das Programm die empfangenen Daten über die Ausgangsspannung in eine '.csv' Datei ausgeben. So konnten wir für jede Messung einen Screenshot und eine '.csv' Datei erstellen. Ein hochkant stehendes Holzbrett definiert den Abstand. Die 21 zu messenden Werte liegen zwischen 10cm und 70cm in jeweils 3cm Abständen. Mit einem Meterstab haben wir einen Richtwert für den Abstand zwischen Abstandssensor und Holzbrett. Nachdem wir durch die erschwerten Lichtverhältnisse die richtige Lage des Abstandssensors gefunden haben, haben wir über das Programm sowohl Screenshots vom Bildschirm des Oszilloskopes gemacht, als auch die Daten in einer '.csv' Datei gespeichert. Zudem haben wir die gemessenen Längen mit deren dazugehörigen Ausgangsspannung handschriftlich in einer

Tabelle aufgeschrieben, welche am Ende des Versuches vom Tutor unterschrieben wurde. Anschließend haben wir in Python programmiert, um die Dateien aus den '.csv' einzulesen mit `genfromtxt()`. Um den Einschwingvorgang nicht mit zu berechnen haben wir die ersten 1000 Zeilen übersprungen. Nachdem werden wir den Durchschnitt sowie die Standardabweichung berechnet haben, visualisieren wir in einer Kennlinie den Durchschnitt sowie die Standardabweichung mittels `matplotlib`.

2.2 Messwerte

Tabelle [2.2] zeigt die von Hand notierten sowie die per Skript erfassten Messwerte

Distanz	Spannung	Durchschnitt	Standartabweichung
10 cm	1.34 V	0.6659393794704831	0.6660828058406754
13 cm	1.15 V	0.5748604885613923	0.5750357147324959
16 cm	1.05 V	0.5234618726058222	0.5236664822795073
19 cm	0.935 V	0.4653799639943883	0.4656458563403725
22 cm	0.838 V	0.4172580848541211	0.41750096830677325
25 cm	0.775 V	0.4172580848541211	0.41750096830677325
28 cm	0.696 V	0.3457795703901795	0.3462834179341396
31 cm	0.657 V	0.3270183282633423	0.32727442926760836
34 cm	0.617 V	0.30706827965420097	0.3073858905206787
37 cm	0.580 V	0.2883470006555391	0.2886328902698045
40 cm	0.560 V	0.28011523262263754	0.28047480730534874
43 cm	0.519 V	0.25865669034127736	0.2589851949585083
46 cm	0.499 V	0.24820714314341125	0.2486386557159006
49 cm	0.479 V	0.23764770166808308	0.2380554924324234
52 cm	0.457 V	0.2263390103963483	0.22679061638041825
55 cm	0.434 V	0.21142392568638377	0.22598395876834088
58 cm	0.412 V	0.20893641211394368	0.20936113207355245
61 cm	0.395 V	0.19631902915868696	0.19675309457785908
64 cm	0.374 V	0.18642892104853862	0.1869330192510634
67 cm	0.395 V	0.19552981861590926	0.19618095153123555
70 cm	0.374 V	0.18458076957849764	0.18511617366642327

Tabelle 2.1: Messwerte Kalibrierung

2.3 Auswertung

In der folgenden Abbildung sind die Messergebnisse nochmals dargestellt. Die Messergebnisse wurden mit matplotlib in Python visualisiert.

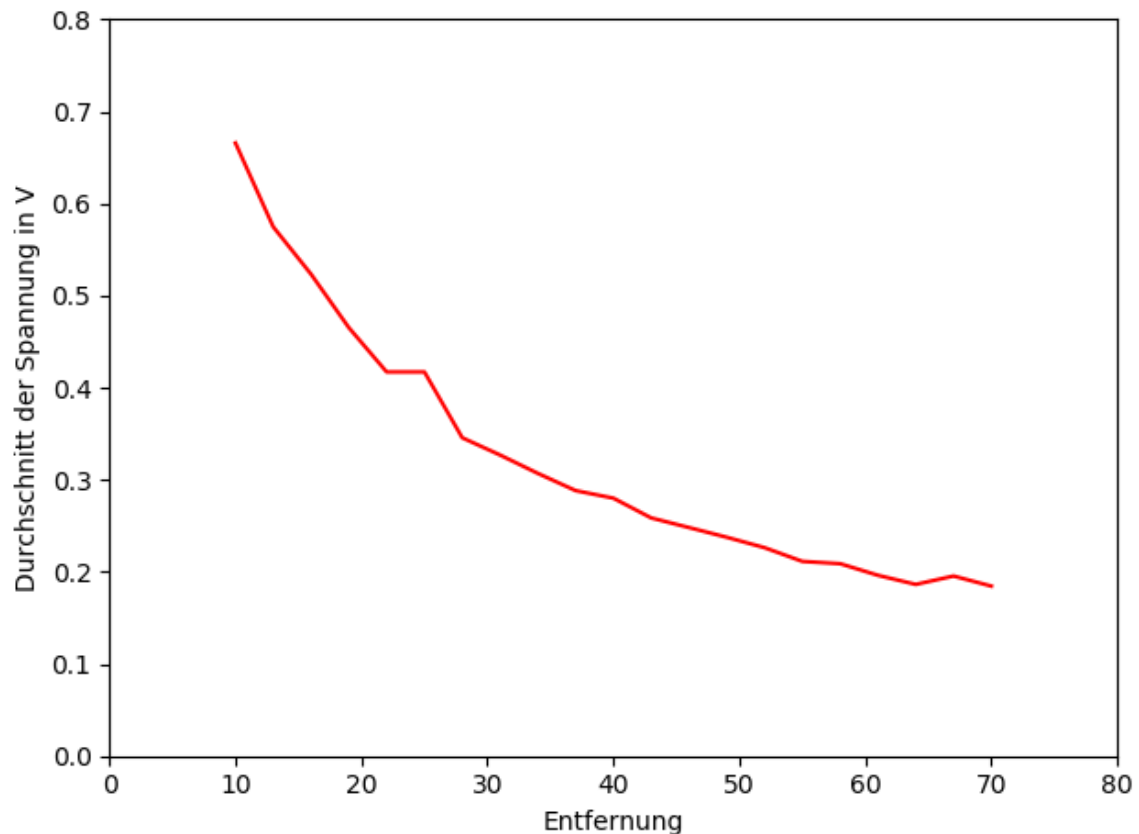


Abbildung 2.1: Plot Messungen

2.4 Interpretation

Wie man gut sehen kann wird die Spannung stets niedriger. Dies liegt an der Anti-proportionalität, dass mit der zunehmenden Entfernung zwischen Holzbrett und Abstandssensor die vom Signalprozessor übertragene Spannung geringer wird.

Leider haben wir beim Generieren der Dateien einen Fehler gemacht und bei 22cm und 25cm zu spät die Single Sequenz aktualisiert, weshalb eine Gerade in dem Plot zwischen 22cm und 25cm genau gleich ist. Ansonsten sieht man im Vergleich zu den manuellen Messdaten keine sichtbaren Unterschiede.

3

Versuch 2: Modellierung der Kennlinie durch lineare Regression

3.1 Fragestellung, Messprinzip, Aufbau, Messmittel

3.2 Messwerte

3.3 Auswertung

3.4 Interpretation

4

Versuch 3: Flächenmessung mit Fehlerrechnung

4.1 Fragestellung, Messprinzip, Aufbau, Messmittel

4.2 Messwerte

4.3 Auswertung

4.4 Interpretation

Anhang

A.1 Quellcode

A.1.1 Quellcode Versuch 1: Messung Kalibrierung

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 ten = -1
5 data = 0
6 data2 = 0
7 data3 = 0
8
9 vec = np.zeros((21, 3))
10 vol = np.zeros((21, 3))
11
12 rang = ["10", "13", "16", "19", "22", "25", "28", "31", "34", "37", "40", "43", "46", "49", "52", "55", "58", "61",
13         "64", "67", "70"]
14
15 for x in rang:
16     ten += 1
17     data = np.genfromtxt('data/' + str(rang[ten]) + '.csv', delimiter=",", skip_header=1000, skip_footer=499)
18
19     print("Durchschnitt " + str(rang[ten]) + "cm:" + str(np.nanmean(data)))
20     print("Standartabweichung " + str(rang[ten]) + "cm:" + str(np.nanstd(data)) + "\n")
21
22     vec[ten, 0] = rang[ten]
23     vec[ten, 1] = np.nanmean(data)
24     vec[ten, 2] = np.nanstd(data)
25
26 plt.axis([0,80,0,0.8])
27 plt.xlabel('Entfernung')
28 plt.ylabel('Durchschnitt der Spannung in V')
```

```

29 plt.plot([vec[0, 0], vec[1, 0], vec[2, 0],vec[3, 0], vec[4, 0],
30          vec[5, 0],vec[6, 0],vec[7, 0],vec[8, 0],vec[9, 0],
31          vec[10, 0],vec[11, 0],vec[12, 0],vec[13, 0],vec[14, 0],
32          vec[15, 0],vec[16, 0],vec[17, 0],vec[18, 0],vec[19, 0],
33          vec[20, 0]],
34          [vec[0, 1],vec[1, 1],vec[2, 1],vec[3, 1],vec[4, 1],
35          vec[5, 1],vec[6, 1],vec[7, 1],vec[8, 1],vec[9, 1],
36          vec[10, 1],vec[11, 1],vec[12, 1],vec[13, 1],
37          vec[14, 1],vec[15, 1],vec[16, 1],vec[17, 1],
38          vec[18, 1],vec[19, 1],vec[20, 1]], 'r')
39 plt.show()
40
41
42 plt.axis([0,80,0,0.8])
43 plt.xlabel('Entfernung')
44 plt.ylabel('Standartabweichung')
45 plt.plot([vec[0, 0], vec[1, 0], vec[2, 0],vec[3, 0], vec[4, 0],
46          vec[5, 0],vec[6, 0],vec[7, 0],vec[8, 0],vec[9, 0],
47          vec[10, 0],vec[11, 0],vec[12, 0],vec[13, 0],vec[14, 0],
48          vec[15, 0],vec[16, 0],vec[17, 0],vec[18, 0],vec[19, 0],
49          vec[20, 0]],
50          [vec[0, 2],vec[1, 2],vec[2, 2],vec[3, 2],vec[4, 2],
51          vec[5, 2],vec[6, 2],vec[7, 2],vec[8, 2],vec[9, 2],
52          vec[10, 2],vec[11, 2],vec[12, 2],vec[13, 2],
53          vec[14, 2],vec[15, 2],vec[16, 2],vec[17, 2],
54          vec[18, 2],vec[19, 2],vec[20, 2]], 'b')
55 plt.show()

```

Listing 5.1: Messung

A.1.2 Quellcode Versuch 2

A.1.3 Quellcode Versuch 3

A.2 Messergebnisse