Laborskript

Grundlagen der elektronischen Messtechnik

Aufgaben zum Laborpraktikum

Prof. Dr. Clemens Gühmann
Daniel Thomanek, M.Sc.

25. Oktober 2024

Technische Universität Berlin
Fakultät IV
Institut für Energie und Automatisierungstechnik
Fachgebiet Elektronische Mess- und Diagnosetechnik

Inhaltsverzeichnis

0 Allgemeine Hinweise

Für das Bestehen des Praktikums gibt es einige Regeln und Hinweise zum Ablauf und zu den Anforderungen eines Protokolls.

0.1 Hinweise zur Vorbereitung

Damit die Versuche durchgeführt werden können und während des Versuchs verstanden werden, ist es unbedingt notwendig, sich auf den Versuch vorzubereiten. Dazu gehören:

- Selbstständige Erarbeitung der Theorie (mittels VL-Folien, Laborskript und ggf. zusätzlicher Literatur)
- Die Theorieaufgaben müssen vor Beginn des Labortermins bearbeitet worden sein.
- Alle Gruppenmitglieder müssen sich vor dem Labortermin mit den Laboraufgaben und Schaltungen vertraut machen

0.2 Regeln

1. Labor
   (a) Es besteht Anwesenheitspflicht für alle Gruppenmitglieder. Bevor das Labor betreten werden darf, müssen die Sicherheitshinweise gelesen, verstanden und akzeptiert werden. Die hierzu geltenden Dokumente *Sicherheitsvorschriften* befindet sich im ISIS-Kurs und müssen von allen Studierenden eingehalten und aktiv (gelesen und verstanden) zur Kenntnis genommen werden.
   (b) Das Labor gilt als bestanden, wenn sowohl die **Pythonabgabe (Aufgabe 1 in Einzelarbeit)**, als auch **alle 5 Labortermine (Aufgaben 2 - 6 in Gruppenarbeit)** abgeschlossen wurden.
      i. Die Pythonabgabe (Aufgabe 1) gilt als abgeschlossen, wenn Sie innerhalb der Abgabefrist ausführbaren Pythoncode (Quelltext: **.py* + dazugehörige Daten: **.npy*) abgegeben haben, welcher bei Ausführung die in Aufgabe 1.3 und 1.4 geforderten Unteraufgaben löst. Zu diesem ersten Versuch muss nur der ausführbare Code und kein Protokoll abgegeben werden (Aufgaben 1.2 kann, muss aber nicht abgegeben werden)
      ii. Ein Labortermin (Aufgabe 2 - 6) gilt als abgeschlossen, wenn Sie innerhalb der Abgabefrist die gelösten Vorbereitungsaufgaben abgeben, alle Gruppenmitglieder pünktlich zum Labortermin erscheinen, die Laboraufgaben erfolgreich abarbeiten und Sie im Anschluss innerhalb der Abgabefrist das Laborprotokoll abgeben. Das Laborprotokoll muss den Anforderungen aus Punkt 0.3 genügen und im Anschluss durch Ihre Tutorin/Ihren Tutor positiv bewertet worden sein.
   (c) Diese Leistung wird in MOSES verbucht und behält auch bei einer nicht bestandenen bzw. nicht angetretenen Klausur ihre Gültigkeit.

2. Vorbereitungsaufgaben

   (a) Vor jedem Labortermin müssen Vorbereitungsaufgaben gelöst werden.

   (b) Die Vorbereitungsaufgaben werden vor dem Beginn des Labortermins über ISIS im PDF Format abgegeben.

   (c) Ohne abgegebene Vorbereitungsaufgaben kann an dem jeweiligen Labortermin nicht teilgenommen werden.

3. Protokolle

   (a) Für jeden Laborversuch wird ein Protokoll erstellt, in dem die Ergebnisse aus dem Laborversuch aufgezeichnet und ausgewertet werden.

   (b) Mindestanforderungen: siehe Abschnitt 0.3 *Was gehört in ein Protokoll*?

   (c) Es steht eine LaTeX-Protokollvorlage im ISIS-Kurs zur Verfügung. Diese kann durch die/den jeweilige/n Tutor/in verpflichtend vorausgesetzt werden. Der jeweilige Tutor wird dies rechtzeitig ankündigen

   (d) Abgabefrist ist in der Regel 7 Tage nach dem Labortermin. Die Abgabe erfolgt in Gruppen über ISIS im PDF Format. Näheres zu den Abgaben und Fristen erfahren Sie in Ihrem jeweiligen ISIS-Kurs

   (e) Die Tutorinnen und Tutoren bewerten die Protokolle „digital“ mit „ausreichend“ oder „nicht ausreichend“ und teilen dies den Teilnehmern wiederum eine Woche später über ISIS mit.

   (f) Es besteht jeweils eine einmalige Korrekturmöglichkeit der Protokolle (Abgabe spätestens 7 Tage nach dem darauffolgenden Labortermin).

4. Abwesenheit/Krankheit

   (a) Jedes Labor muss von allen Gruppenmitgliedern besucht werden. Falls es Gruppen gibt, in denen das aufgrund nachweisbarer einschlägiger Gründe (z.B. Zugehörigkeit zu einer Risikogruppe, wie Schwangere, chronisch Kranke, etc.) nicht möglich ist, sprechen Sie Ihre Tutorin oder Ihren Tutor an.

   (b) Bei Krankheit muss ein entsprechendes Attest im Sekretariat des Fachgebiets eingereicht werden. Ist der Rest der Gruppe weiterhin *arbeitsfähig* muss der Labortermin nicht nachgeholt werden.

5. Fehlleistungen:

   (a) Es dürfen maximal 2 Fehlleistungen während der gesamten Labordurchführung anfallen. Diese sind folgend aufgelistet:

      i. ungenügende Vorbereitung auf das Labor (z.B. fehlende Vorbereitungsaufgaben)
      ii. verspätete Abgabe der Vorbereitungsaufgaben
      iii. verspätete Abgabe eines Protokolls
      iv. „nicht ausreichendes“ Protokoll nach einer Korrektur

0.3 Was gehört in ein Protokoll?

- Deckblatt
  - Thema, Gruppe, Teilnehmerinnen und Teilnehmer, Betreuerin oder Betreuer

- Inhalt allgemein
  - evtl. Inhaltsverzeichnis
  - Übersichtlichkeit (Seiten, Bilder, Tabellen und Formeln werden nummeriert etc.)
  - Literaturverzeichnis
- Vorbereitende Theorie
  - Die gelösten Aufgaben sind Teil des Protokolls
- Durchführung der Versuche
  - Was ist das Ziel des Versuchs?
  - Beschreibung des Messaufbaus (inklusive der Messgeräte und deren Messgenauigkeiten)
  - Wie wurden die Versuche durchgeführt?
  - Besonderheiten?
  - Messergebnisse, Darstellung (Achsenbeschriftung!)
- Diskussion und Auswertung
  - Was besagen die Ergebnisse?
  - Vergleich mit den theoretischen Ergebnissen
  - Erklärung der Abweichungen
  - Zusammenfassung

1 Einführung in Python

Qualifikationsziele

- Grundkenntnisse im Umgang mit Python
- Grundkenntnisse im Umgang mit den Bibliotheken Numpy und Matplotlib

1.1 Einleitung

Ziel dieser Aufgaben ist das Erlernen der Grundfunktionen von Python und den Bibliotheken *Numpy* und *Matplotlib*, die im Laufe der Labortermine immer wieder zum Einsatz kommen werden.
Die Aufgaben werden – anders als die darauffolgenden Versuche – von jeder/jedem einzeln in Eigenarbeit zu Hause gelöst und im Haupt-ISIS Kurs abgegeben.
Als Hilfestellung werden von den Tutoren/Tutorinnen während der ausgewiesenen Bearbeitungszeit (s. Zeitplan im ISIS-Kurs) Sprechstunden angeboten. Die entsprechenden Zeiten sowie die Zoom-Links oder Räume entnehmen Sie bitte dem ISIS-Kurs. Bei dieser Aufgabe kann zu jeder Sprechstunde gegangen werden und nicht nur zu jener des/der eigenen Tutors/Tutorin
Für diesen ersten Teil des Praktikums muss kein Bericht abgegeben werden, sondern der ausführbare Pythoncode zusammen mit den gespeicherten Messdaten.

1.2 Wichtige Befehle

Lösen Sie die folgenden Aufgaben mithilfe von Funktionen der Numpy- und Matplotlib-Bibliothek. Evtl. benötigte Befehle können in der Python, Numpy und Matplotlib Dokumentation gefunden werden.
Diese Aufgabe 1.2 dient der Überprüfung der mindestens benötigten Pythonkenntnisse. Sie muss nicht mit abgegeben werden

1. Erzeugen Sie auf zwei verschiedene Weisen einen Vektor x, dessen Elemente von 1 bis 1000 laufen.
2. *Schneiden* Sie die Werte $10 - 50$ aus diesem Vektor heraus und ordnen Sie sie einem neuen Vektor zu.
3. Weisen Sie dem ersten Element des neuen Vektors den Wert 0 zu. Welche Auswirkungen hat dies auf den ursprünglichen Vektor? Überprüfen Sie das entsprechende Element des ursprünglichen Vektors und schreiben Sie einen Kommentar in den Code.
4. Erzeugen Sie einen Vektor $y = [0.2; 0.4; 0.6; ...; 200.0]$ und bilden Sie das Skalarprodukt von x und y.
5. Multiplizieren Sie x und y nun *elementweise*
6. Bilden Sie die Summe aller Elemente des Vektors x ohne eine Schleife zu benutzen.
7. Erzeugen Sie eine 3×4 Matrix mit gleich-verteilten Zufallszahlen zwischen 0 und 1.
8. Fügen Sie zu dieser Matrix eine weitere Zeile hinzu, die nur Einsen enthält.
9. Berechnen Sie nun die Transponierte, Inverse und Determinante dieser Matrix.

10. Erzeugen sie eine Periode eines Sinus-Signals mit einer Frequenz von 50 Hz und einer Amplitude von 5 V. Erzeugen Sie zuerst einen Vektor t der die Zeitpunkte enthält und anschließend den Vektor y, der die Spannungswerte enthält. Das Signal sollte für eine gute Auflösung mindestens 30 Stützstellen (Elemente im Vektor) aufweisen.

11. Plotten Sie mithilfe der Bibliothek *matplotlib.pyplot* den Spannungsverlauf und fügen Sie Achsenbeschriftungen und einen Titel hinzu.

12. Bestimmen Sie mit *Numpy* die Zeit, bei der das erzeugte Sinus-Signal seinen Maximalwert annimmt. Bestimmen Sie außerdem den Index, an dem dieser Maximalwert aufgetreten ist und mit dessen Hilfe den Zeitpunkt. Plotten Sie diesen Punkt in die Abbildung des Zeitverlaufs.

13. Numpy kann Datensätze mit den Befehlen *np.save()* und *np.load()* speichern und laden. Testen Sie die Befehle, in dem Sie t und y in je eine Datei speichern. Zur Probe löschen Sie ihren Workspace indem Sie *%reset* in die Konsole eingeben und laden den Datensatz erneut in den Workspace.

14. Plotten Sie den Sinus und ein normalverteiltes Rauschsignal in zwei *Unterfenstern (Subplots)* übereinander. Erzeugen Sie dazu das Rauschsignal mit der Funktion *np.random.randn()* und verwenden Sie für beide Signale denselben Zeitvektor.

15. Speichern Sie nun den Plot in eine PDF Datei.

1.3 Datenerfassung mit der Messkarte NI USB-6009

Im Labor können Messungen mit der Messkarte NI USB-6009 durchgeführt werden. Zur Vorbereitung hierauf wurde eine virtuelle Messkarte mit der gleichen API in dem Python-Modul *mdt.py* implementiert. Der virtuelle Frequenzgenerator wurde so eingestellt, dass er eine sinusförmige Spannung mit einer Amplitude von 2.5 V und einer Frequenz von 50 Hz erzeugt. Dieser wurde an den ersten Kanal ihrer virtuellen Messkarte angeschlossen.
Auf die Messkarte kann mittels der Funktion *mdt.dataRead()* zugegriffen werden. Parametrieren Sie diese Funktion so, dass sie mindestens eine vollständige Periode aus dem gemessenen Signal erhalten. Die Karte kann Signale mit einer maximalen Abtastrate von 48 kHz und einer Auflösung von maximal 14 Bit einlesen. Die Nummerierung der Kanäle beginnt mit 0.
Speichern Sie anschließend das Signal in einen Datensatz zur Weiterverarbeitung ab.
Gehen Sie zur Lösung dieser Aufgabe wie folgt vor:

1. Laden Sie sich das Python-Modul *mdt* aus dem ISIS-Kurs herunter. Mit *Python-Modul* ist die Datei *mdt.py* aus dem ISIS-Kurs gemeint. Diese muss in den selben Ordner wie Ihr Pythonskript kopiert werden. Den Speicherort Ihres Pythonskriptes wird in Spyder unter der Haupt-Iconleiste angezeigt (siehe Abbildung 1.1)

2. Importieren Sie die mdt Funktionen mittels *import mdt*

3. Die Messung erfolgt mittels der Funktion *mdt.dataRead()*. Wird diese ohne Parameter ausgeführt erscheint ein Hilfetext. Aus diesem ist die Parametrierung der Funktion ersichtlich. Verwenden sie als Abtastrate 48 kHz und eine Auflösung von 14 Bit

4. Die Daten werden bei jeder virtuellen Messung leicht unterschiedlich aussehen. Speichern Sie daher die Daten in eine Datei und Laden Sie diese anschließend wieder.
   Vergessen Sie nicht die hier generierten .npy-Dateien mit abzugeben

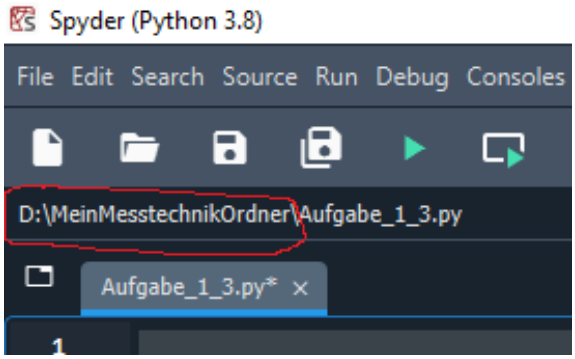

Abbildung 1.1: Ordner in welchen die Datei *mdt.py* aus dem ISIS-Kurs kopiert werden muss (Beispiel)

5. Schneiden Sie eine vollständige Periode des Sinus aus. Plotten Sie dazu das gesamte Signal und suchen sie manuell den Index des Beginns und des Endes der Periode.

6. Stellen Sie das gemessene und das ausgeschnittene Signal in jeweils einem Unterfenster (Subplot) dar. Erzeugen Sie sich dazu auch einen Zeitvektor, damit die Beschriftungen an der x-Achse korrekt sind.

1.4 Matrizen und lineare Gleichungssysteme

Es wird angenommen, dass die Stellung eines Kfz-Fahrpedals durch einen Spannungswert zwischen 3.1 V und 0.45 V wiedergegeben wird. Die kleinere Spannung entspricht dabei einem Winkel von $35\,^\circ$ und die größere Spannung einem Winkel von $0\,^\circ$. Bestimmen Sie die Spannung als Funktion der Winkelstellung.
Gehen Sie zur Lösung dieser Aufgabe wie folgt vor:

1. Nehmen Sie eine lineare Beziehung an

2. Stellen Sie das lineare Gleichungssystem (Matrizenform) auf

3. Lösen Sie das Gleichungssystem z. B. durch das Invertieren der Matrix (*np.linalg.inv()*)

1.4.1 Selbstdefinierte Funktionen

Schreiben Sie eine Funktion *angle2voltage*, mit der Sie eine Winkelstellung in eine Spannung umrechnen. Überprüfen Sie Ihre Funktion, indem Sie die Winkel $0\,^\circ$ und $35\,^\circ$ eingeben.

1.4.2 Grafiken

Das Fahrpedal wird nun betätigt. Die Veränderung der Winkelstellung *angle* über der Zeit t soll mit Hilfe der Funktion $angle(t) = 1/2 \cdot (\tanh(t-5)+1) \cdot 35$ simuliert werden. Plotten Sie den Verlauf der Funktion im Bereich $t \in [0\,\mathrm{s}, 10\,\mathrm{s}]$. Stellen Sie weiterhin den zugehörigen Verlauf der Spannung im gleichen Graphikfenster dar. Beschriften Sie die Achsen und fügen Sie eine Legende ein.