

# Browserbasierte Programmierung und Interaktion mit Arduino-Aufbauten im Bereich Elektrotechnik

Michael Ebert<sup>1</sup> und Wolfram Haupt<sup>2</sup>

**Abstract:** Aufgaben zur Programmierung eingebetteter Systeme (Mikrocontroller) lassen sich für Programmieranfänger im Bereich der Elektrotechnik nur schwer, teilweise bedingt durch den hohen organisatorischen Aufwand, integrieren. Im Rahmen einer Lehrveranstaltung *Programmieren 1* mit Python, welche auf fachbezogene Beispiele während der Vorlesung und Übung setzt, wurde eine webbasierte Simulation von Ein- und Ausgängen eines Arduinos mit zusätzlichen Sensoren entwickelt. Arduino liefert ein einsteigerfreundliches Komplettpaket, bestehend aus Mikrocontroller, I/O-Board und entsprechender Software zur Ansteuerung. Die Besonderheit unserer Umsetzung ist die Interaktion mit der virtuellen Hardware, d. h., Veränderung von Schalterstellungen und regelbaren Sensorwerten. In diesem Paper möchten wir unsere Simulation und deren mögliche Einsatzszenarien vorstellen, welche Studierenden den ersten Kontakt mit der Hardware erleichtern und somit auch das Ausprobieren neuer Inhalte und Konzepte ermöglichen sollen.

**Keywords:** Programmierung, Python, Interaktives eBook, Arduino

## 1 Motivation und anwendungsbezogene Programmieraufgaben

Yacob et al. [AM12] haben unterschiedliche Motivationsfaktoren in Bezug auf das Lernen von Programmieren in Ingenieursstudiengängen untersucht. Neben extrinsischen Faktoren wie das soziale Umfeld oder Prüfungen spielen auch intrinsische Faktoren wie die Einstellung zum Thema und Erwartung an die Vorlesung eine Rolle [MMF]. An der Hochschule Coburg führen wir regelmäßig zu Semesterbeginn in der Lehrveranstaltung *Programmieren 1* für Elektrotechniker Evaluationen durch, die auf ähnliche Ergebnisse hindeuten. Die Fragen zielen unter anderem auf das Vorwissen, die empfundene Relevanz der Informatik aus Sicht der Studierenden, und auf die Erwartungen an die Veranstaltung ab. Die Ergebnisse lassen erkennen, dass die Studierenden zwar wenig Vorwissen mitbringen, jedoch ein Großteil Informatik und Programmieren als relevanten Studieninhalt erachten. Besonders Praxisnähe und Anwendungsbezug wurden als Erwartungshaltung an die Lehrveranstaltung genannt. Jedoch erfordert die Verwendung von praxisnahen und anwendungsspezifischen Systemen, wie z. B. Mikrocontrollern, meist einen hohen organisatorischen und zeitlichen Aufwand. Oft muss die Hardware ausgeteilt bzw. vorbereitet werden. In kleinen Fakultäten mit wenig Laborpersonal und begrenzter Raumkapazität ist dies nur schwer zu realisieren, jedoch können solche Übungen zum Lernerfolg und zur Motivation beitragen [MMF][FG05]. In dieser Arbeit

---

<sup>1</sup> Hochschule Coburg, Projekt EVELIN, Friedrich-Streib-Str. 2, 96450 Coburg, michael.ebert@hs-coburg.de

<sup>2</sup> Hochschule Coburg, Projekt EVELIN, Friedrich-Streib-Str. 2, 96450 Coburg, wolfram.haupt@hs-coburg.de

möchten wir unseren Ansatz der virtuellen Hardware - Arduinos (Mikrocontroller) mit angeschlossenen LEDs und Schaltern - vorstellen.

## 2 Ansatz

An der Hochschule Coburg wurde im Wintersemester 2014 eine spezielle Übung zur Ansteuerung und Bedienung von RaspberryPis abgehalten. RaspberryPis sind Mini-Computer mit Ein- und Ausgabe-Pins an die eine programmierbare LED-Matrix angeschlossen wurde. In diesen Übungsstunden wurden eine aktivere Teilnahme und ein größeres Interesse am Fach beobachtet, welches durch die Lehrveranstaltungsevaluation bestätigt werden konnte.

Mit nicht veränderbaren, virtuellen Hardware-Aufbauten, wie z. B. ein Mikrocontroller mit LCD-Display und Temperatursensor oder mehrere ansteuerbare LEDs sowie Schalter, möchten wir die Motivation und damit langfristig den Lernerfolg in Bezug auf die Programmierfähigkeit steigern. Die interaktiven Beispiele sollen ausgewählte Programmierkonzepte und deren Umsetzung in einer konkreten Programmiersprache veranschaulichen und eine greifbare Vorstellung der Auswirkungen von Anweisungen geben. Alle Beispiele sollen direkt im Browser, z. B. in einem interaktiven eBook oder in Präsentationen verwendet werden können. Somit können diese universell innerhalb von Vorlesungen (Live-Coding), in Übungen oder von zu Hause aus bearbeitet oder verwendet werden. Die Aufgaben und Beispiele sollen Konzepte wie Bedingungen, Schleifen und Bitoperationen und deren Anwendungen abdecken. Der Mikrocontroller und weitere Elemente, wie Schalter oder LEDs, werden dabei simuliert. Der Fokus liegt allerdings nicht auf der physisch korrekten Simulation der einzelnen Stromflüsse, sondern beschränkt sich auf die Ein- und Ausgabeschnittstellen. Zum Beispiel sollen Schalter von Studierenden betätigt werden, um die Auswirkungen zu beobachten. Python soll dabei als einfache Programmiersprache dienen, die über eine serielle Schnittstelle mit der Hardware kommuniziert und eine Bibliothek zur Steuerung verwendet. Dieselbe Bibliothek soll auch beim Einsatz richtiger Hardware in Übungen verwendet werden und so einen leichten Übergang ermöglichen.

## 3 Umsetzung

Die Implementierung der Simulation im Browser basiert auf einem JavaScript-Python-Interpreter [Sc14]. Dieser wurde speziell angepasst und Komponenten zur Simulation und Interaktion mit HTML-Elementen hinzugefügt. Abbildung 1 zeigt einen Aufbau eines Lauflichtes. Der Reset-Schalter sowie der Schalter auf dem Steckboard können durch Mausklicks ausgelöst werden. Die LEDs leuchten, wenn der jeweilige digitale Pin durch das Programm (vgl. Abb. 1) angesteuert wird. Zur Ansteuerung der LEDs wurde eine virtuelle serielle Schnittstelle geschaffen, die der über den Port zugeordneten Hardware Befehle schickt. Alle Elemente, wie der Arduino, Verbindungen, LEDs und Schal-

ter, werden in Fritzing<sup>3</sup> erstellt und sind nicht veränderbar.

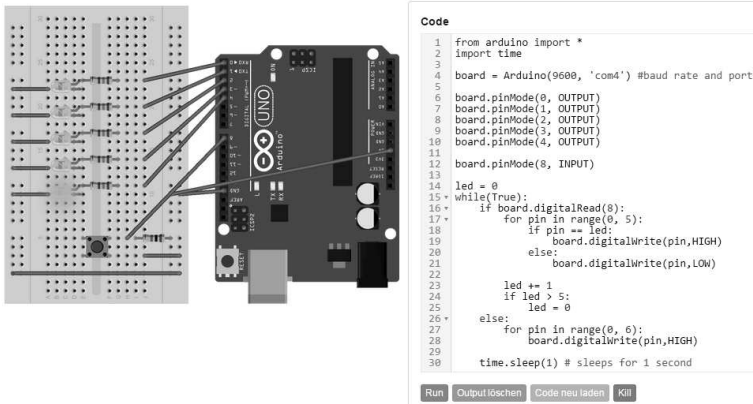


Abb. 1: Lauflicht mit zusätzlichem Schalter auf dem Breadboard zum Moduswechsel

## 4 Ausblick

Der hier vorgestellte Ansatz, welcher anwendungsbezogene und praxisrelevante Beispiele tiefer in Vorlesung und Übung verankert, soll im Wintersemester 2015 evaluiert werden. Die Motivation bzw. Interesse der Studierenden bildet hierbei das Hauptaugenmerk in der Evaluation. Kann diese durch anschauliche Beispiele, die jederzeit ausgeführt werden können, gesteigert werden, und tragen diese zur fachlichen Einordnung des Programmierens in das Studium bei?

## Literaturverzeichnis

- [AM12] Azliza Yacob; Mohd Yazid Md Saman: Assessing Level of Motivation in Learning Programming among Engineering Students. In (The Society of Digital Information and Wireless Communications, Hrsg.): The International Conference on Informatics and Applications (ICIA2012), S. 425–432. Shenyang, 2012.
- [FG05] Forte, A.; Guzdial, M.: Motivation and Nonmajors in Computer Science: Identifying Discrete Audiences for Introductory Courses. IEEE Transactions on Education, 48(2):248–253, 2005.
- [MMF] Martins, Scheila Wesley; Mendes, Antonio Jose; Figueiredo, Antonio Dias: A strategy to improve student’s motivation levels in programming courses. In: 2010 IEEE Frontiers in Education Conference (FIE). S. F4F–1–F4F–7.
- [Sc14] Skulpt, Python. Client side., www.skulpt.org, Stand: 27.10.2014.

<sup>3</sup> Eine Software zum Erstellen von Elektronikaufbauten (vgl. <http://fritzing.org>)