

Das LiFi-Projekt {.unnumbered}

Digitalisierung und Programmierung

Prof. Dr. Nicolas Meseth

Inhaltsverzeichnis

Worum geht es im LiFi-Projekt?	1
Was sind die Ziele?	2
Wie gehen wir vor?	2
Was benötigen wir für das LiFi-Projekt?	3
Hardware	3
Software	5
Übungsaufgaben	5

Worum geht es im LiFi-Projekt?

LiFi ist eine Technologie, die die Übertragung von Informationen mithilfe von Licht ermöglicht. Sie nutzt das sichtbare Lichtspektrum, um ein Signal zu erzeugen, das von einem Fotodetektor empfangen werden kann, welcher das von einer LED ausgestrahlte Licht erfasst. Durch die Veränderung der Lichteigenschaften über die Zeit, wie etwa der Wellenlänge oder der Helligkeit, können wir Daten kodieren und übertragen. LiFi bietet großes Potenzial für den Einsatz in Umgebungen, in denen Hochfrequenzsignale Mikroorganismen oder andere empfindliche elektronische Geräte stören könnten. Darüber hinaus könnte LiFi im Gegensatz zu Bluetooth oder WiFi in Robotern eingesetzt werden, die unter Wasser arbeiten.

Deine Aufgabe als Teil eines interdisziplinären F&E-Teams in einem Hightech-Unternehmen ist die Entwicklung eines LiFi-Kommunikationsgeräts. Das Unternehmen entwickelt Roboter für Lebensmittel- und Landwirtschaftsanwendungen, und das Gerät soll in die nächste Robotergeneration integriert werden. Es besteht aus zwei Hauptkomponenten: eine kleine LED, die mehr als 16 Millionen Farben des RGB-Farbsystems darstellen kann, und einen Farbsensor, der die Intensität der RGB-Farbkanäle und die Lichthelligkeit misst. Ein sogenannter Master Brick – ein Minicomputer – steuert diese Komponenten und gewährleistet die reibungslose Kommunikation zwischen dem Roboter und seinen Peripheriegeräten.

Was sind die Ziele?

Das LiFi-Projekt stellt ein typisches Ingenieursproblem dar: die Kombination von Hardware und Software zur Lösung eines Praxisproblems. Die zentralen inhaltlichen Fragen dieses Projekts lauten:

- Wie können physikalische Größen wie die Temperatur oder Licht gemessen und von einer analogen in eine digitale Form überführt werden?
- Wie können Algorithmen Entscheidungen auf Basis der digitalen Eingabedaten treffen?
- Wie können wir mit Lichtsignalen Informationen darstellen?
- Wie können wir mithilfe einer LED und eines Farbsensors Informationen übertragen?
- Welches Protokoll eignet sich am besten für die LED-basierte Datenübertragung?
- Wie verlässlich ist die Datenübertragung?
- Welche maximale Übertragungsdistanz ist mit Licht möglich?
- Welche Umgebungsbedingungen sind für eine erfolgreiche Übertragung erforderlich?
- Wie lässt sich die Datenübertragung sicher gestalten?
- Welche Datenübertragungsrate können wir erreichen?
- Wie können wir möglichst effizient kommunizieren?

Eine wichtige Einschränkung besteht darin, dass wir all diese Fragen der uns bereitgestellten Hardware beantworten müssen: einer LED und einem Farbsensor.

Am Ende dieses Projekts wirst du nicht nur das Ingenieursproblem gelöst, sondern auch Antworten auf die genannten Fragen gefunden haben. Als zusätzlichen Bonus erwirbst du dabei tiefere Einblicke in die digitale Welt sowie grundlegende Programmierkenntnisse.

Wie gehen wir vor?

Ein solch großes und komplexes Ingenieursproblem wie das LiFi-Projekt erfordert ein durchdachtes Vorgehen. Da es sich vornehmlich um ein Projekt zur Einführung in die digitale Welt handelt, gehen wir schrittweise vor und lernen bei jedem Schritt wichtige Grundlagen, die uns bei der Umsetzung helfen.

Zuerst widmen wir uns dem Basteln: Nach Anleitung setzen wir aus den bereitgestellten Hardware-Bauteilen den LiFi-Prototypen zusammen. Dieser bildet die Grundlage für unsere praktische Arbeit im Projekt. Während die Hardware nur einmalig zu Beginn aufgebaut werden muss, entwickeln wir die Software kontinuierlich über das gesamte Projekt hinweg weiter.

Um das Problem besser zu verstehen und in kleinere Teilprobleme zu zerlegen, betrachten wir in **sec-problem-solving** zunächst geeignete Techniken zur Problemlösung. Parallel dazu beginnen wir mit der Programmierung in Python – der Sprache, die wir für die Entwicklung der LiFi-Software nutzen werden. Diese Kenntnisse erweitern wir in jedem Kapitel, führen

wichtige Konzepte der Programmierung ein und lernen, wie wir mit Python die Hardware des LiFi-Prototypen steuern können.

Das Buch ist in vier Teile gegliedert. Nach jedem Teil stellen wir eine neue Version unseres LiFi-Prototypen fertig. Die finale Version enthält dann alle notwendigen Funktionen, die zur Beantwortung der Fragestellungen erforderlich sind.

Was benötigen wir für das LiFi-Projekt?

Hardware

Komponenten

Für den LiFi-Hardware-Prototyp benötigen wir folgende Komponenten:

- 1 x [Master Brick 3.1](#)
- 1 x [RGB LED Bricklet 2.0](#)
- 1 x [Color Bricklet 2.0](#)
- 1 x [OLED 128x64 Bricklet 2.0](#)
- 4 x [Bricklet Cable 15 cm \(7p-7p\)](#)
- 1 x [USB-A to USB-C Cable 100 cm](#)
- 2 x [Mounting Plate 22x10](#)
- 4 x [Mounting Kit 12 mm](#)

Bitte überprüfe vor dem Fortfahren mit den folgenden Anweisungen, ob dein LiFi-Kit alle Komponenten in den angegebenen Mengen enthält.



Aufbauanleitung

1. Schutzfolie auf den Befestigungsplatten entfernen
2. Abstandshalter an beide Befestigungsplatten anbringen

Befestigungsplatte 1

Befestigungsplatte 2

3. Master Brick befestigen
4. Verbindungskabel einstecken
5. Befestigungsplatten miteinander verbinden
6. OLED-Anzeige montieren
7. LED und Farbsensor montieren
8. Peripheriegeräte mit Master Brick verbinden

Software

Neben der Hardware benötigen wir für die Entwicklung des LiFi-Prototyps verschiedene Software-Komponenten. Die Software ist komplett Open-Source und dadurch kostenlos nutzbar. Alle Programme sind für Windows, Mac OS und Linux verfügbar. Hier zunächst die Übersicht, bevor wir jede Software im Detail vorstellen:

- [Brick Daemon](#) und [Brick Viewer](#)
- [Visual Studio Code](#)
- [Python](#)
- [Git](#)

Brick Daemon

Brick Viewer

Visual Studio Code

Python

- Virtuelle Python-Umgebungen
- Externe Module installieren

Git

- GitHub
- Ein Repository klonen
- Den lokalen Code aktualisieren

Übungsaufgaben

1. Verwendet den Brick Viewer und verbindet euch mit eurem Master Brick. Stellt sicher, dass die Verbindung funktioniert:
 - a. Öffnet den Tab für die RGB LED und erkundet die verfügbaren Funktionen.
 - b. Testet den Farbsensor, indem ihr seine Reaktionen unter verschiedenen Lichtverhältnissen beobachtet.
 - c. Ermittelt die eindeutige Identifikationsnummer (UID) für jeden eurer Sensoren im Brick Viewer.
2. Startet das Visual Studio Code und öffnet dort ein neues Terminalfenster:

- a. Wechselt über die Kommandozeile in ein Verzeichnis eurer Wahl und klonst das GitHub-Repository mit dem Beispielcode für dieses Buch mittels des `git`-Befehls.
- b. Erstellt eine neue virtuelle Python-Umgebung im Ordner `.env` und aktiviert diese. Installiert die für den Beispielcode benötigten Python-Module `tinkerforge` und `pyyaml`.