

Hochschule Reutlingen

Reutlingen University

– Studiengang Mechatronik Bachelor –

Bachelor–Thesis

Internet der Dinge in Schüler-Technikprojekten

Carolin Vanessa Hahn
Joseph-Haydn-Straße 9
71254 Ditzingen

Matrikelnummer : 720115

Betreuer: Prof. Dr. rer. nat Stefan Mack
Zweitbetreuer: Prof. Dr. rer. nat Eberhard Binder
Abgabedatum: 02.10.2017



Danksagung

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die mich während der Anfertigung dieser Bachelor-Thesis unterstützt und motiviert haben.

Zuerst gebührt mein Dank Herrn Prof. Dr. rer. nat Stefan Mack, der meine Bachelor-Thesis betreut und begutachtet hat. Für die hilfreichen Anregungen bei der Erstellung dieser Arbeit möchte ich mich herzlich bedanken.

Ebenfalls möchte ich mich bei meinen Betreuern Anian Bühler und Michael Hermann bedanken, die mir mit viel Geduld, Interesse und Hilfsbereitschaft zur Seite standen.

Meinem Mann, Micha, meiner Familie und meinen Freunden danke ich besonders für den starken emotionalen Rückhalt über die Dauer meines gesamten Studiums.

Abschließend möchte ich mich bei meinen Eltern bedanken, die mir mein Studium durch ihre Unterstützung ermöglicht haben und stets ein offenes Ohr für meine Sorgen hatten.

Abstract

The „Internet of Things“ (IoT) is a current issue that is gaining importance in industry as well as in the private sector. In this thesis possibilities to integrate the topic IoT into student projects will be investigated. It is being examined particularly for the university-school cooperation project letsgoING, to what extent this is suitable for bringing the topic of IoT closer to students.

The importance of the „Internet of Things“ in an engineer’s everyday working life for industry and private life is to be discussed.

It is to be investigated whether other technology projects for pupils are addressing the topic of IoT and which methods are used. In order to develop an effective concept, the results with regard to letsgoING are assessed comparatively.

Basics that are important for the development of the learning module are presented. This includes the basics of Internet communication as well as the function of the required software and hardware. A microcontroller with WLAN chip is the core hardware element of this work. It is chosen in order to link existing letsgoING hardware components to the Internet. Different possibilities how this can be done are presented. Learning objectives are defined and methods for their implementation are presented.

Ultimately, the project is to be implemented in the most effective and efficient way in terms of expenses, implementation time and complexity.

Finally, an evaluation of the achieved goals is to be conducted and an outlook on possibilities to extend and improve the learning module.

Inhaltsverzeichnis

1	Einleitung	1
2	Motivation	3
2.1	Das Kooperationsmodell letsgoING	3
2.2	Internet der Dinge	6
3	Vergleich bestehender Konzepte	9
3.1	Der Baukasten „SenseBox“	9
3.1.1	Test der SenseBox:edu mit dem Wlan-Shield „BlueFly“	10
3.2	Genius - ein Projekt von Daimler	12
3.3	Roberta - Eine Initiative des Fraunhofer Institut IAIS	14
3.4	Calliope - Projekt, Mikrocontrollerboard und gemeinnützige GmbH	15
3.5	Vergleich und Bewertung der Projekte	17
4	Grundlagen	19
4.1	Typischer Aufbau eines Mikrocontrollers	19
4.2	Kommunikation im Internet	22
4.2.1	Internet Protocol und TCP/UDP	23
4.2.2	Webserver	23
4.2.3	Hypertext Transfer Protocol - HTTP	24
4.2.4	HTML, CSS und Javascript - Aufbau einer Webpage	25
4.3	Physische Komponenten der Internetkommunikation	26
4.3.1	Wireless LAN	26
4.3.2	Router	27
4.3.3	Modem	27
4.3.4	Gateway	28
4.3.5	Wireless Access Point	28
5	Anforderungen und Konzeption	29
5.1	Lernziele für Schüler mit dem IoT-Lernmodul	29
5.2	Notwendige Infrastruktur an Schulen	30
5.3	Anforderungen an die Hardware des IoT-Moduls	31
5.4	Hardwarekonzepte für das IoT-Modul	32
5.4.1	IoT-Modul als Schnittstellenkonverter auf WLAN	32
5.4.2	IoT-Modul als eigenständiger, programmierbarer Mikrocontroller	33
5.4.3	IoT-Modul als Input-Output-Modul	34
5.4.4	Auswahl des am besten geeigneten Hardware-Konzepts	35
5.5	Input-Output-IoT-Modul: Software- und Kommunikationskonzepte	36
5.5.1	Kommunikation mit dem IoT-Modul über eine App auf dem Smartphone oder Tablet	36
5.5.2	Kommunikation mit dem IoT-Modul über einen Cloudservice im Internet	38
5.5.3	Kommunikation mit dem IoT-Modul über dessen lokalen Webserver	39

5.5.4	Auswahl des am besten geeigneten Software- und Kommunikations-Konzepts	41
6	Realisierung	43
6.1	Hardware	43
6.1.1	Vergleich der Mikrocontrollerchips ESP8266 und ESP32	43
6.1.2	Aufbau der Hardwarekomponenten	44
6.2	Software	46
6.2.1	Programmablauf	46
6.2.2	Verbindungsaufbau mit Netzwerken	47
6.2.3	Digitale und analoge Werte verarbeiten und austauschen	48
6.2.4	Konfiguration und Bibliotheken	49
6.2.5	Website-Design und Umsetzung der Lernziele	50
6.3	Inbetriebnahme des IoT-Moduls	52
7	Zusammenfassung und Ausblick	53
A	Eidesstattliche Erklärung	59

Abbildungsverzeichnis

2.1	letsgoing Fahrzeug mit Arduino und aufgesteckten letsgoING-Modulen	5
2.2	Die grafische Programmierumgebung Ardublock.	6
3.1	Der Inhalt der SenseBox:edu.[18]	10
3.2	OpenSenseMap mit Interpolation der Temperatur.	11
3.3	Grafische Programmieroberfläche Open Roberta Lab.	15
3.4	Schematische Darstellung des CalliopeMini [?].	16
3.5	Vergleich der Projekte und Bewertung im Sinne der genannten Kriterien: grün: positiv, rot: negativ, gelb: unbefriedigend oder neutral.	17
4.1	Kommunikation zwischen einem Mikrocontroller und seiner Umgebung.	19
4.2	Kommunikation zwischen den Komponenten eines μ Cs und seiner Umgebung. Vgl. [22].	21
4.3	Server-Client-Prinzip des HTTP [29].	25
4.4	Kommunikationsschnittstellen eines Routers [33].	27
5.1	Kommunikation innerhalb eines Netzwerks.	29
5.2	Das WLAN-Modul als WLAN-Shield, schematische Darstellung der Kommunikation.	33
5.3	Das WLAN-Modul als alleinstehender Mikrocontroller. Schematische Darstellung der Kommunikation.	34
5.4	Das WLAN-Modul als IO-Modul. Schematische Darstellung der Kommunikation. . .	35
5.5	Diagramm: Entscheidung für das geeignetste Hardware-Konzept.	35
5.6	Tabelle: Bewertung der vorgestellten Hardwarekonzepte.	36
5.7	Anwendung mit Mobile App. Schematische Darstellung der Kommunikation.	37
5.8	Anwendung mit Cloudservice. Schematische Darstellung der Kommunikation.	38
5.9	Website auf ESP32 als Webserver und Access Point. Schematische Darstellung der Kommunikation	39
5.10	Entwurf des Layouts für die Website im Modul Internet der Dinge.	40
5.11	Tabelle: Bewertung der vorgestellten Softwarekonzepte.	41
5.12	Diagramm: Entscheidung für das geeignetste Hardwarekonzept und Softwarekonzept.	42
6.1	Vergleich der wichtigsten Merkmale des ESP8266 und seinem Nachfolger ESP32. Angaben nach [37] und [38].	43
6.2	SparkFun ESP32 Thing Entwicklungsplattform [39].	44
6.3	Versuchsaufbau: oben: ESP32 Thing mit Spannungswandler, unten: (v.l. n.r.) Die LGI-Module Potentiometer, LDR-Lichtsensor, RGB-LED, Stromadaptermodul . . .	45
6.4	Schaltbild für den Aufbau.	45
6.5	Pinbelegung der Sensoren und Aktoren.	46
6.6	Über die Funktion „Event-Handler“ wird der URI-Code durchsucht	47
6.7	Quellcode: Passwort und Netzwerkname (SSID).	47
6.8	Optionen für den Start des IoT-Moduls.	48
6.9	Konfiguration der PWM-Pins.	49
6.10	Layout der Webseite.	50

6.11	Adresszeile nach Klicken des Buttons „Ein”	51
6.12	Adresszeile nach Klicken des Buttons „senden” neben den PWM-Werten.	51
6.13	Adresszeile nach senden eines Schwellwertes.	52

Kapitel 1

Einleitung

Das „Internet der Dinge“ ist ein aktuelles Thema, das sowohl in der Industrie als auch im privaten Bereich an Bedeutung gewinnt. In dieser Arbeit sollen Möglichkeiten untersucht werden, das Thema „Internet der Dinge“ (IoT) in Schülerprojekte zu integrieren.

Es wird speziell für das Hochschul-Schul-Kooperationsprojekt letsgoING untersucht, inwiefern dieses sich dafür eignet, Schülern das Thema IoT näher zu bringen. Das Projekt findet im regulären Unterricht statt und in enger Kooperation zwischen Schülern, Lehrern, Lehramtsstudenten und der Hochschule. Dadurch hebt es sich von vielen anderen Projekten ab. Das Projekt ist modular aufgebaut und besteht aus mehreren Lerneinheiten. Innerhalb dieser Arbeit soll eine weitere Lerneinheit entstehen. Das Ziel ist, dafür ein Konzept zu erstellen, bestehend aus einem Hardware-IoT-Modul und Begleitmaterial für die Lehre.

Die Bedeutung des „Internets der Dinge“ im Berufsalltag eines Ingenieurs, für die Industrie und den privaten Lebensbereich sollen erörtert werden.

Es soll untersucht werden, ob andere Technikprojekte für Schüler das Thema IoT aufgreifen. Dabei ist von Bedeutung wie das Lehrkonzept gestaltet ist, welche didaktischen Methoden Anwendung finden und mit welchem Schwerpunkt und Tiefgang IoT behandelt wird. Um ein effektives Konzept zu erarbeiten, werden die Ergebnisse in Bezug auf letsgoING vergleichsweise beurteilt.

Im nächsten Schritt werden Grundlagen vorgestellt, die für die Erarbeitung des Lernmoduls wichtig sind. Dazu gehören Grundlagen über die Kommunikation im Internet sowie die Funktion dafür benötigter Soft- und Hardware.

Ein Mikrocontroller mit WLAN-Chip bildet das Zentrale Hardware-Element dieser Arbeit. Er wird ausgesucht, um bestehende letsgoING-Hardwarekomponenten mit dem Internet zu verknüpfen. Unterschiedliche Möglichkeiten wie dies geschehen kann werden vorgestellt. Es werden Lernziele festgelegt und Methoden für deren Umsetzung vorgestellt. Letztendlich soll der Entwurf umgesetzt werden, der die Lernziele am effektivsten und am effizientesten in Bezug auf Ausgaben, Zeitaufwand für die Umsetzung und Komplexität im Einsatz verfolgt.

Abschließend soll eine Bewertung der erreichten Ziele erfolgen und ein Ausblick über Möglichkeiten um das Lernmodul zu erweitern und verbessern.

Kapitel 2

Motivation

2.1 Das Kooperationsmodell letsGoING

letsGoING ist ein Kooperationsprojekt des Fachbereichs Mechatronik der Hochschule in Reutlingen mit Gymnasien im Umkreis. Unter dem Motto „Mikrocontroller macht Schule“ wird im Rahmen des Naturwissenschaft-Technik-Unterrichts (NwT) in der Sekundarstufe Wissen aus dem Arbeitsbereich eines Ingenieurs vermittelt. Das finanzielle Fundament für die Initiative bietet seit Projektbeginn im Jahr 2012 die Vector Stiftung. Die wesentliche Idee von letsGoING ist es, Schüler und vor allem auch Schülerinnen, im NwT-Unterricht für technische Aufgaben zu motivieren und bis zur Studienwahl zu begleiten. Durch den dauerhaften Kontakt zum Fachbereich Mechatronik der Hochschule soll so für das Berufsbild des Ingenieurs geworben werden.

Bislang einzigartig für eine Schulkooperation ist, dass alle Schüler einbezogen werden, da das Projekt im Rahmen des regulären NwT-Unterrichts stattfindet und nicht in einer freiwilligen Arbeitsgruppe. So werden alle erreicht, auch Schüler die sich noch nicht mit technischen Fragestellungen auseinandergesetzt haben. Das ist eine Chance, mehr Schüler und vor allem auch Schülerinnen für technische Berufe zu motivieren. Denn Frauen sind in Ingenieurstudiengängen immer noch eine Minderheit. Nur 23% der Hochschulabgänger in Ingenieurstudiengängen in Deutschland sind Frauen. Dabei stehen die Chancen gut: die meisten der unbesetzten MINT¹-Arbeitsstellen machen technische Berufe aus, darunter vor allem solche für Informatiker und Ingenieure. [1] Den potentiellen Studentinnen gilt daher ein besonderes Augenmerk.

Die Begleitung des Unterrichts durch Studenten als Mentoren schafft einen direkten Kontakt zur Hochschule und erweitert die Möglichkeit für ein Studium zu motivieren. Die Aufgaben im Projekt sind in Anlehnung an die Arbeitsweise eines Ingenieurs gestaltet und führen auf eine zugängliche, spielerische Weise in technische Themen ein. Die Schüler bearbeiten technische Aufgaben und werden auch handwerklich gefordert.

Der technische Fortschritt ist nicht aufzuhalten und die zunehmende Digitalisierung erstreckt sich bis in die Kinderzimmer, wo schon die Kleinen mit Lerncomputern umgehen und kaum ein Teenager ist noch ohne eigenes Smartphone, wie eine Studie des IZI zeigt [2]. Innerhalb der Schulbildung bietet sich eine gute Gelegenheit, junge Menschen frühzeitig zu sensibilisieren und ihnen Technikmündigkeit zu vermitteln. So ist es auch im Bildungsplan für NwT vorgesehen: „Die Vielfalt der technologischen Entwicklungen fordert von den Schülerinnen und Schülern heute und in Zukunft ein hohes Maß an Bewertungs-, Urteils- und Entscheidungsfähigkeit. Sie erwerben durch die Bearbeitung naturwissenschaftlicher und technischer Fragestellungen in Verknüpfung mit gesellschaftlichen, ökonomischen und ökologischen Aspekten eine naturwissenschaftlich-technische Allgemeinbildung und entwickeln eine Technikmündigkeit“ [3]. Es ist wichtig, die junge Generation auf das vorzubereiten was sie in ihrer Zukunft erwartet, unabhängig davon, ob der Einzelne einen technischen Berufsweg einschlägt oder nicht.

¹MINT: Mathematik Informatik Naturwissenschaft Technik.

Die Wissensvermittlung unterscheidet sich von anderen Projekten dadurch, dass zunächst Grundlagen geschaffen werden. Dies geschieht in einer Form, die an die Herangehensweise von Ingenieuren erinnert, nämlich dem Denken in Prozessen und Systemen. Das Niveau und die Auswahl der Themen sind auf die Zielgruppe abgestimmt. Es geht beispielsweise nicht nur darum, ein Programm schreiben zu können, das die geforderten Funktionalitäten erfüllt, sondern, vergleichbar mit der Arbeitsweise eines Ingenieurs, auch um das Gestalten eines Konzepts, welches den Weg zum Ziel aufzeigt. Das Verständnis dafür, wie ein Programm aus Codeteilen seine Funktion überhaupt erfüllen kann und wofür Datentypen und Variablen eingesetzt werden können, schafft eine Basis für eigenständiges, kreatives Lösen von Aufgaben statt das Kopieren von vorgefertigten Lösungen zu forcieren.

Einen hohen Stellenwert nimmt das Vermitteln von Grundlagenwissen über die Verwendung der Hardware ein. Die Schüler sollen verstehen, wie die Komponenten miteinander interagieren, mit denen sie die gestellten Aufgaben lösen. Dabei wird jedoch nicht physikalisches, tiefes Grundlagenwissen vermittelt, denn für Ingenieure ist es vor allem wichtig, Schnittstellen zu verstehen. Wenn etwas nicht funktioniert, was im Alltag oft vorkommt, muss ein Ingenieur nach der Fehlerquelle suchen. Daher liegt der Fokus im Projekt auf der systematischen Fehlersuche, dem Troubleshooting. Unter anderem unterstützen Blockschaltbilder dabei, das Zusammenspiel der Komponenten besser zu verstehen. Elektrotechnische Grundlagen werden nur vertieft, sofern es für das Lösen der Aufgaben hilfreich ist. Frustrationstoleranz und Selbstständigkeit werden geübt und den Anforderungen des Bildungsplans, „Lösungen zur Darstellung der Wechselwirkungen und des Gesamtzusammenhangs“ zu entwickeln, werden genüge geleistet [3].

Die Schüler arbeiten in Teams an Aufgaben oder Projekten und übernehmen Verantwortung, vor allem beim individuell gestaltbaren Abschlussprojekt. Dabei werden die Schüler kreativ und setzen eigene Ideen in einem „Ingenieur“-Projekt um, das sie selbständig planen. Auch darin steht letsGoING im Einklang mit dem Bildungsplan für NwT in dem es heißt, durch die „Durchführung von Projekten [...] wird [das] Durchhaltevermögen und [die] Beharrlichkeit gestärkt sowie [die] Leistungsbereitschaft und [das] Leistungsvermögen [der Schüler] gefördert [3].

Das Projekt bietet ein umfassendes Konzept und stellt Hardware, Software und didaktisch aufwendig aufbereitetes Unterrichtsmaterial mit Lösungen als Lehrerhandreiche zur Verfügung. Die Lerneinheiten sind modular gestaltet und an den Lehrplan anpassbar. Der Lehrer kann entscheiden welche Themen er in den Unterricht einbinden möchte und wie tief jedes Thema bearbeitet werden soll. Für Schüler gibt es, neben dem halbjährigen NwT-Kurs im Unterricht, eintägige Schnupperkurse an der Hochschule und Wochenkurse im Angebot. Lehrer werden in Fortbildungskursen fachlich sowie organisatorisch vorbereitet, bekommen eine Einführung in das letsGoING-Projekt und Grundlagenwissen vermittelt. Die Kurse bieten einen Einblick in Ingenieurstätigkeiten und sind Voraussetzung für den NwT-Schülerkurs im Unterricht.

Die Ausbildung der NwT-Lehrer ist inzwischen einer der Schwerpunkte bei letsGoING, denn nur wer selbst über fundierte Kenntnisse verfügt, kann eine gute Ausbildung gewährleisten. Daher hat letsGoING das Konzept „Lernen und Weiterbilden“ entwickelt, welches Lehrer und NwT-Lehramtsstudierende auf nachhaltige Weise zusammenbringt. Lehrer, die sich auf das Projekt einlassen, bekommen einen Lehramtsstudenten als Tutor an die Hand, der mit ihnen gemeinsam das Thema im Unterricht einführt. Außerdem bekommt der Lehrer einen Klassensatz des gesamten Materials. Der Lehramtsstudent profitiert ebenfalls, denn er hat für die Zukunft ein Unterrichtskonzept, welches er bereits in der Praxis anwenden konnte.

Wie wichtig solche Initiativen wie „Lernen und Weiterbilden“ für die Digitale Bildung an Schulen ist, zeigte sich auf dem 5. MINT-Gipfel. Experten wurden sich dort darüber einig, dass Investitionen in digitale Bildungsangebote nur dann sinnvoll genutzt werden können, wenn auch Lehrer für die Vermittlung digitaler Inhalte geschult werden[4].

Das Herzstück der Hardware im Projekt ist ein Arduino Uno Mikrocontroller. An diesen können

die letsgoING-Module, also die passend entworfenen Platinen mit Sensoren und Aktoren angesteckt werden. Eine der im Projekt realisierten Anwendungen, ist ein kleines Fahrzeug, an das weitere Sensoren für Spur- und Abstandhaltung sowie ein Akku angebaut sind. Der Arduino mitsamt den LetsgoING-Modulen wird darauf aufgesetzt, um die Komponenten des Fahrzeugs zusammen mit dem Arduino zu nutzen.

Das Stecksystem für die Module ist bewusst gewählt. Es bietet einerseits einen besseren Überblick und unterstützt beim Verständnis für die Kommunikation und den Stromfluss zwischen der Hardwareumgebung und dem Mikrocontroller. Es stellt alle Komponenten in einen leichter erkennbaren Zusammenhang, als bei fertig bestückten Platinen. Andererseits erhöht selbstständiges Löten und Zusammenbauen einzelner Komponenten natürlich gewollt die Fehlerquelle, was oben genannten didaktischen Ansatz des Troubleshootings unterstützt. Die Schüler programmieren mit der gra-

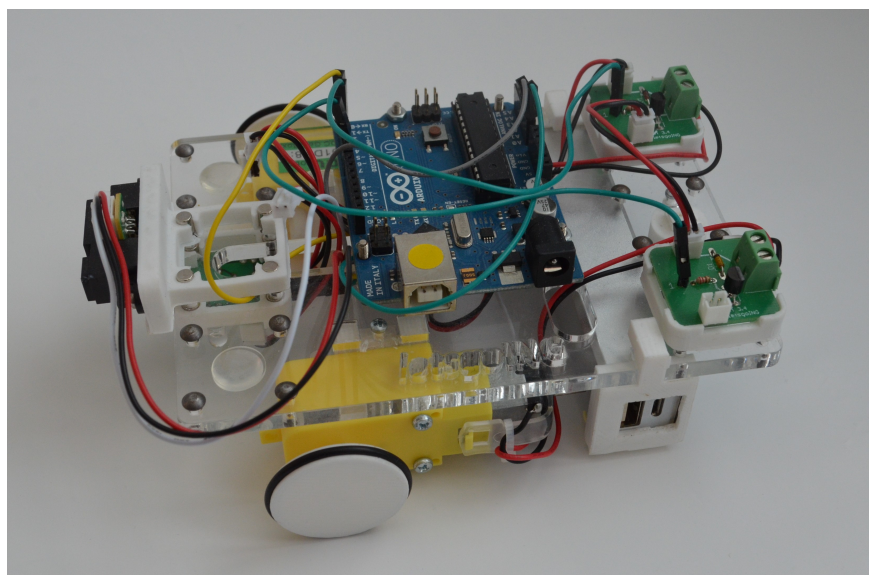


Abbildung 2.1: letsgoing Fahrzeug mit Arduino und aufgesteckten letsgoING-Modulen

phischen Programmierumgebung „Ardublock“. Das ist eine im Sinne von Open Source speziell für Arduino entwickelte Plattform, die in die Programmierumgebung des Arduino eingebettet ist und auf einfache Art in die Programmierung einführt. Ardublock ist auf der Programmiersprache des Arduino aufgebaut. Ein Programm wird durch Zusammenfügen von grafisch dargestellten Programmblöcken erstellt². Für das letsgoING Projekt werden auf die Hardware und Aufgaben abgestimmte Programmblöcke eigens entworfen. Beim Kompilieren wird im Hintergrund der textbasierte Arduino-Code generiert, den sich interessierte Schüler anschauen können. Sie haben zudem die Möglichkeit, den Code in einer professionellen Entwicklungsumgebung zu verändern. Es steht außerdem eine Internetplattform zur Verfügung, die als Nachschlagewerk für Schüler, Lehrer und betreuende Studenten dient. Die im Projekt verwendeten Komponenten und ihre jeweiligen Einsatzmöglichkeiten werden dort beschrieben und erklärt. Die Plattform unterstützt eine selbstständige Arbeitsweise beim Lösen von Problemen, die während der Projektdurchführung auftreten³. Kooperationen gibt es derzeit mit fünf Schulen aus Reutlingen, Tübingen und Nagold. Ein Ziel für die Zukunft der letsgoING-Initiative ist es, ein Konzept zu entwickeln, welches es erlaubt, die Kurse ohne Begleitung durch Studenten durchführen zu können. Daran wird derzeit gemeinsam mit dem Otto-Hahn-Gymnasium Nagold gearbeitet.

²Das Erscheinungsbild erinnert an Scratch, das aber auf ein ganz anderes Konzept zur Einführung in die Programmierung setzt. Denn Scratch ist eine von Wissenschaftlern entworfene, grafische Programmiersprache die bewusst nur einfache Funktionen unterstützt und vor allem für das Programmieren einfacher visueller Anwendungen gedacht ist. Diese können im eigenen Scratch-Player abgespielt werden. <https://scratch.mit.edu/>. Abrufdatum: 02.07.2017

³letsngING Wiki: <http://letsgoingwiki.reutlingen-university.de/mediawiki/index.php/Hauptseite>

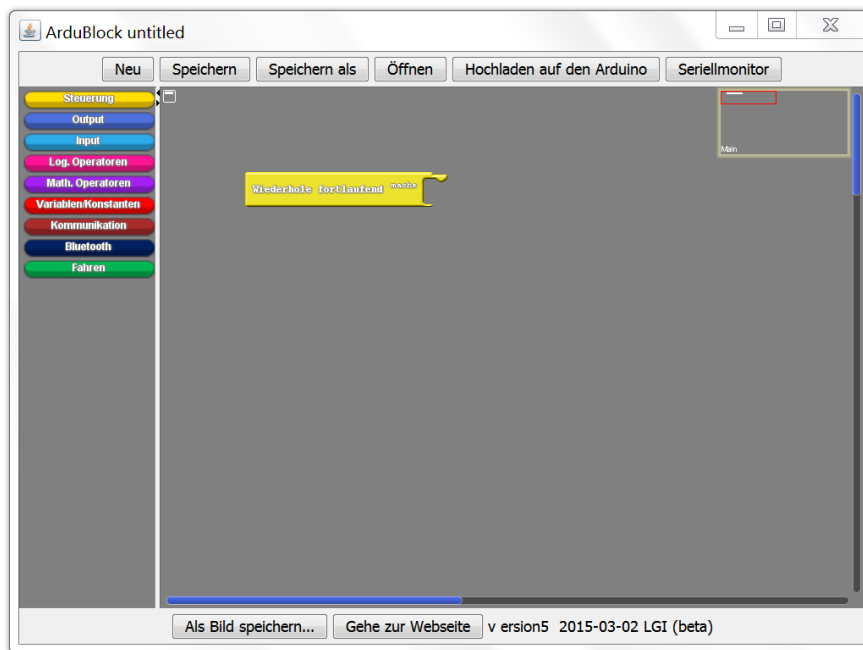


Abbildung 2.2: Die grafische Programmierumgebung ArduBlock.

2.2 Internet der Dinge

Das Internet wird heute von nahezu jedem Menschen täglich genutzt und ist fest in den Alltag integriert. Es ist für viele zu einer Selbstverständlichkeit geworden, jederzeit „online“ zu sein. Über Smartphones, Laptops und Tablets, von zu Hause oder unterwegs, greifen Menschen auf Inhalte zu. Diese sind auf Rechnern gespeichert, deren Standorte sich irgendwo weltweit befinden. Es ist möglich Inhalte über das Internet abzurufen oder zwischen Endgeräten zu kommunizieren.

Schon 1966 sagte Karl Steinbuch, ein Pionier der deutschen Informatik, voraus, es gebe bald „kaum mehr Industrieprodukte [...], in welche die Computer nicht hineingewoben sind, etwa so, wie das Nervensystem in Organismen...“ [5] [6]. Dies lässt sich inzwischen im Alltag beobachten. Viele der Gegenstände des täglichen Gebrauchs, haben Mikrocontroller und kleinste Rechenchips eingebaut. Die Komponenten für diese Chips und für Sensoren werden seit Jahrzehnten immer kleiner und energieeffizienter und können daher in kleinsten Geräten verbaut werden. Diese Umstände haben den Weg bereitet, für das „Internet der Dinge“ (IoT). Es geht dabei nicht mehr darum, dass der Mensch über das Internet kommuniziert. Wie der Begriff schon treffend beschreibt, sind die Dinge, also die Gegenstände über das Netzwerk des Internets verbunden. Sie können selbständig Informationen austauschen.

Der englische Begriff „Internet of Things“ beschreibt, genauso wie der deutsche „Internet der Dinge“, treffend den Umstand, dass „Dinge“ oder Gegenstände über eine Internetverbindung kommunizieren. Es ist außerdem möglich über Endgeräte wie Laptops, Smartphones und Tablets mit diesen Gegenständen drahtlos zu kommunizieren.

Auf eine einheitliche Definition konnten sich Wissenschaftler des National Institute of Standards and Technology (NIST) bislang nicht einigen [7]. Annäherungsversuche gab es schon. Es scheint naheliegend, sich an der Definition für das „Network of Things“ (NoT) zu orientieren, bei dem Gegenstände nicht über das Internet sondern über ein lokales Netz kommunizieren [8]. Daher sind sich alle, die an der Definition mitarbeiten, einig darüber, was Anwendungen im Internet der Dinge gemeinsam haben: das Tasten durch Sensoren, woraufhin Berechnungen anhand der ertasteten Werte durch Computer erfolgen und die anschließende Kommunikation über ein Netzwerk, um eine

Aktion auszulösen. Gegenstände, die dem Begriff „Internet der Dinge“ zugeordnet werden, beinhalten also Sensorik, Berechnung, Kommunikation und Aktion.

Ein anschauliches Beispiel für eine IoT-Anwendung ist das „Smart Home“. Der Begriff bezeichnet ein System, mit dem es möglich ist, Licht, Jalousien, die Heizung oder sogar Küchengeräte von der Ferne zu steuern oder ganz autonom arbeiten zu lassen. Scheint die Sonne auf die Fensterfront, nehmen Sensoren dies wahr. Der Computer stellt Berechnungen an und kommuniziert mit der im Rolladen und der Heizung verbauten Computereinheit. Der Rolladenmotor reagiert mit dem Herablassen der Jalousie und eventuell wird die Heizung reguliert. Licht kann über das Smartphone oder Tablet gesteuert werden. Der Anwender bekommt außerdem Benachrichtigungen auf sein mobiles Gerät.[9]

Andere Anwendungen sehen die Verbindung mit dem Internet von weiteren Geräten vor. Küchenmaschinen, Kühlschränke, Backöfen und andere Haushaltsgeräte sind vereinzelt schon mit einer Internetverbindung ausgestattet.

Google und Amazon⁴ bieten Geräte für eine Sprachsteuerung verschiedener Komponenten im Haus (Licht, Fernseher, Musikanlage etc.) und um Bestellungen im Internet zu tätigen. Diese Geräte können auch per Sprachsteuerung und -ausgabe Fragen beantworten indem sie Informationen aus dem Internet abrufen. Dafür „hören“ sie ständig den Geräuschen in ihrer Umgebung zu.

Eine Technisierung eines alltäglichen Gebrauchsgegenstands und ein weiteres Beispiel für eine IoT-Anwendung sind Fitnessarmbänder. Weiterentwickelt aus der digitalen Armbanduhr, senden sie Daten über das Bewegungsverhalten einer Person zur Aufzeichnung an Server. Anhand daraus erbrachter Berechnungen werden Empfehlungen und Warnungen ausgesprochen. Ziel ist die Einhaltung eines festgelegten Fitnessplans zu unterstützen, den Schlaf zu überwachen und darüber zu benachrichtigen wenn es „Auffälligkeiten“ gibt, wie zum Beispiel zu langes Sitzen am Schreibtisch. Solche Armbänder sind schon in vielen Ausführungen von unterschiedlichen Herstellern erhältlich [10]. Server halten die persönlichen Informationen bereit.

An den beiden Beispielen der Sprachsteuerung für zu Hause und am Fitnessarmband wird besonders deutlich, dass mit Voranschreiten der Entwicklung ein differenzierter Umgang mit dem Thema „Internet der Dinge“ einhergehen sollte. Fragen könnten dabei sein, wo Daten im Internet gespeichert sind, welche Daten gespeichert werden und wer Zugriff darauf hat. Diese Aspekte werden im Abschnitt 5.1 behandelt. Mit dem Thema IoT ist auch der Bereich der künstlichen Intelligenz (KI) verbunden. Hierauf wird in der vorliegenden Arbeit jedoch nicht eingegangen.

IoT-Anwendungen werden im Rahmen von Industrie 4.0 in Produktionsstätten und Logistikzentren von Firmen verwendet. Industrie 4.0 wird daher im Englischen oft als „Industrial Internet of Things“ (IIoT) bezeichnet. Das Bundesministerium für Bildung und Forschung (BMBF) beschreibt, welche „Vision [...] hinter dem Schlagwort Industrie 4.0 [steckt]: Maschinen, die miteinander kommunizieren, sich gegenseitig über Fehler im Fertigungsprozess informieren, knappe Materialbestände identifizieren und nachbestellen - das ist eine intelligente Fabrik“ [11]. Die Plattform Industrie 4.0 wurde von den Verbänden BITKOM, VDMA und ZVEI gegründet und ging aus der Hightech-Strategie der Bundesregierung hervor [12]. Die Einbindung in den Geschäftsablauf der Ideen die Industrie 4.0 zugrunde liegen, kann einem Unternehmen ein hohes Potenzial in der Produktionssteigerung, sowie Zeit- und Kosteneinsparung bieten [13]. Andere Länder verfolgen ähnliche Ziele unter anderem Namen: die USA mit Industrial Internet Consortium und Japan mit der Industrial Value Chain-Initiative [14].

Industrie 4.0 wird Ingenieure in der Zukunft daher in ihrem Berufsalltag begleiten. Die Entwicklung wird auch als „4. industrielle Revolution“ bezeichnet, worauf sich auch die Zahl „4.0“ in dem Begriff bezieht und die Verzahnung der industriellen Produktion mit digitalen Technologien gemeint ist[15].

⁴Google Home: https://madeby.google.com/intl/de_de/home/ und Amazon Echo: <https://www.amazon.de/Amazon-SK705DI-Echo-Schwarz/dp/B01GAGVCUY>

Die Anwendungen im privaten und industriellen Bereich machen deutlich, dass sich die Gesellschaft in einem digitalen Wandel befindet. Computer stecken schon seit längerer Zeit in Alltagsgegenständen. Diese Gegenstände werden um eine Internetverbindung erweitert und vernetzt, obwohl dies für das Erfüllen ihrer eigentlichen Kernfunktionen (Backen, Uhrzeit Anzeigen, Heizen etc.) nicht notwendig ist. In Wirtschaft und Politik wird im Zusammenhang mit IoT auch von der „digitalen Transformation gesprochen“ [16].

Die Sinnhaftigkeit von IoT-Anwendungen, vor allem im privaten Bereich, können aber auch in Frage gestellt werden. Kriterien können sein: spare ich durch die Anwendung tatsächlich Zeit oder Kosten ein? Gestaltet sie meinen Alltag angenehmer? Nimmt sie mir Aufgaben ab oder benötigt sie genauso viel oder gar mehr Aufmerksamkeit, nur in anderer Form? Ein verantwortungsbewusster Umgang mit neuen technologischen Errungenschaften und ein technisches Grundverständnis, ist für jeden Menschen wichtig um sich zurechtzufinden. Für einen Ingenieur ist neben dem technischen Verständnis und dem praktischen Ansatz, beim Bewältigen von Aufgaben auch ein differenzierter Umgang mit der verwendeten Technologie nützlich und hilfreich. Je früher kritisches Herangehen eingeprägt wird, desto nachhaltiger können neue Produkte von Ingenieuren aber auch von jedem damit in Berührung stehenden, sei es im Beruf oder privat, verwendet werden. Denn die digitale Transformation findet nicht nur in der Wirtschaft statt. Alle Bereiche unserer Gesellschaft sind durch immer mehr digitale Produkte, direkt oder indirekt betroffen und involviert.

Bereits für jetzt tätige und noch viel mehr für zukünftige Ingenieure ist es daher unumgänglich, sich mit den Technologien auseinanderzusetzen die mit dem Internet der Dinge verknüpft sind. Für den Nachwuchs ist es wichtig, den Umgang mit IoT-Anwendungen früh zu erlernen, um das Wachsen von Hemmschwellen, die mit neuer Technik in Verbindung stehen, zu verhindern. Parallel sollen junge Menschen zu einem kritischen Umgang und zur eigenen Meinungsbildung angehalten werden.

Der VDI-Präsident Prof. Udo Ungeheuer sieht in den aktuellen Umwälzungen in Gesellschaft und Wirtschaft eine Chance für „die Weiterentwicklung der Ingenieurskunst“. Damit diese gelingen kann, müssten Ingenieure entsprechend ausgebildet sein und sich nicht nur in der Technik, sondern auch in digitalen Geschäftsmodellen auskennen. Die Basis dafür sollte laut Ungeheuer früh im Leben gelegt werden. Daher fordert er eine „digitale Bildungsoffensive für alle Schulformen und Jahrgangsstufen“ ohne langwierige Reformen. Eu-Kommissar Günther Oettinger fügt dem hinzu, das Land brauche „mehr Menschen die aus ihrem Hobby einen Beruf machen“. [16]

Schülerprojekte bieten die Chance, junge Menschen mit unterschiedlich ausgeprägten Interessen zu erreichen und zu motivieren, sich mit dem Thema „Internet der Dinge“ auseinanderzusetzen. Bei letsgoING ist aufgrund der Projektdurchführung im regulären Unterricht die Chance gegeben, eine große Bandbreite an Schülern zu erreichen (siehe Abschnitt 2.1). Es wird also nicht nur den ohnehin Interessierten der Wissensdurst gestillt und deren Lust auf Mehr angeregt. Auch alle anderen werden motiviert sich auf technische und digitale Themen aus dem Ingenieursberuf einzulassen. Das Projekt ist somit sehr gut geeignet, junge Menschen auf Themen vorzubereiten, die in der Zukunft eine wichtige Rolle spielen. Es beinhaltet sogar das selbst gesteckte Ziel, aktuelle Themen aus dem Alltag eines Ingenieurs zu vermitteln und auf den Beruf vorzubereiten. Zu diesen Themen gehört, aufgrund seiner zunehmenden Aktualität, auch das Internet der Dinge.

Um ein Lernmodul „Internet der Dinge“ zu entwickeln, soll untersucht werden, welche Angebote es diesbezüglich derzeit gibt. Es ist von Interesse, wie andere Projekte vorgehen, um Schülern IoT nahezubringen und welcher Lernerfolg von diesen Projekten zu erwarten ist. Aufgrund dieser Untersuchungen soll ein Konzept für ein IoT-Lernmodul im Projekt letsgoING erarbeitet werden.

Kapitel 3

Vergleich bestehender Konzepte

Es gibt einige Projekte und Initiativen für den Einsatz in der Schulbildung im Bereich MINT die Arduinos oder ähnliche Mikrocontroller nutzen. Es soll untersucht werden, ob und inwiefern diese das Thema „Internet der Dinge“ aufgreifen.

Dabei werden vor allem Kriterien zur Bewertung des Lehrangebots herangezogen. Es wird untersucht,

- welche Inhalte vermittelt werden, ob sowohl auf Software und Programmierung als auch auf die im Projekt jeweils verwendete Hardware eingegangen wird und in welcher Weise das geschieht.
- Bei der Wissensvermittlung ist wichtig, wie gut der didaktische Ansatz bezogen auf Niveau und Inhalt, für die jeweils angesprochene Zielgruppe geeignet ist. Davon hängt ab, welche Lernerfolge erwartet werden können.
- Es kommt vor allem auf die Qualität des mitgegebenen theoretischen Materials an, ob Anleitungen vorhanden sind und ob diese verständlich sind und ob gestellte Aufgaben zielführend für die Lehre sind. Dazu soll jeweils eine Einschätzung erfolgen.
- Eine wichtige Komponente für den Erfolg eines Projekts ist die Schulung der Lehrer durch theoretisches Begleitmaterial und Fortbildungen.
- Die anfallenden Kosten spielen natürlich auch eine Rolle und dürfen das Budget der Bildungseinrichtungen nicht zu sehr belasten.

3.1 Der Baukasten „SenseBox“

Die SenseBox ist ein Baukasten mit Sensoren und einem Arduino-Mikrocontroller¹, wie in Abbildung 3.1 dargestellt[17]. Der Arduino kann mit dem beiliegenden Netzteil, mit dem Batteriepack oder über den USB-Anschluss am PC mit Strom versorgt werden. Für die Kommunikation liegt ein WLAN-Shield oder wahlweise ein Ethernetshield bei. Auf der SD Karte können Sensorwerte (zwischen)gespeichert werden. Aus den Komponenten lässt sich eine Wetterstation bauen, deren Messwerte dann auf der dafür bereitgestellten Internetplattform OpenSenseMap² veröffentlicht werden können und somit für jeden zugänglich sind. Der Plattform, wie in Abbildung 3.2 zu sehen ist, liegt eine Landkarte zugrunde, auf der alle Messstationen gekennzeichnet sind und aufgenommene Messwerte angezeigt werden.

Die Entwickler sind Mitarbeiter des Instituts für Geoinformatik der Universität Münster und verstehen die SenseBox als Citizen-Science-Werkzeug. Citizen Science bedeutet, Bürger forschen oder nehmen an der Forschung teil. In diesem Fall, durch das Sammeln von Geodaten, die der Allgemeinheit zur weiteren Verwendung, unter der Public Domain Dedication and Licesce 1.0, zur Verfügung stehen. Darüber hinaus soll die SenseBox als Werkzeug für Schulen dienen und im Unterricht fächerübergreifend eingesetzt werden können.

¹Es liegt eigentlich ein Genuino-Mikrocontroller bei, der aber baugleich mit dem Arduino ist und sich lediglich durch einen anderen Hersteller unterscheidet.

²<https://opensensemap.org/>



Abbildung 3.1: Der Inhalt der SenseBox:edu.[18]

Es gibt die Box in zwei Ausführungen, zum einen die SenseBox:home für den Heimgebrauch, bei der weniger Sensoren beiliegen und nicht gelötet werden muss. Fertiger Code steht zum Download bereit. Der Programmcode ist ebenfalls Open Source unter der LGPL 3.0 Lizenz verfügbar. Zum Anderen gibt es die SenseBox:edu welche mehr Sensoren beinhaltet und mehr Freiraum zum kreativen Basteln bietet, weshalb sie für den Einsatz in Bildungseinrichtungen gedacht ist. Programmcode kann entweder selbst entworfen werden oder von der SenseBox:home übernommen und um die zusätzlichen Sensoren erweitert werden. Für beide SenseBoxes gibt es eine Anleitung für den Aufbau und die Inbetriebnahme, sowie die Registrierung auf der Plattform OpenSenseMap. Die Anleitungen sind als Open Educational Resources unter der Creative Commons Lizenz CC BY SA 4.0 frei zugänglich.[17] Die Sensebox:edu kann derzeit für etwa 160 Euro bei Watterott bestellt werden, die SenseBox:home für etwa 100 Euro.³

3.1.1 Test der SenseBox:edu mit dem Wlan-Shield „BlueFly“

Außer den schon genannten Komponenten liegen der SenseBox:edu einige Jumper-Kabel, ein Breadboard und folgende Sensoren und Aktoren bei:

- Kombiniertes Temperatur- und Luftfeuchtigkeitssensor (HDC1008)
- Luftdrucksensor (BMP280)
- Infrarot-Distanzsensoren (GP2YA)

³ <http://www.watterott.com/de/Sensoren/senseBox> Stand: 20.04.2017

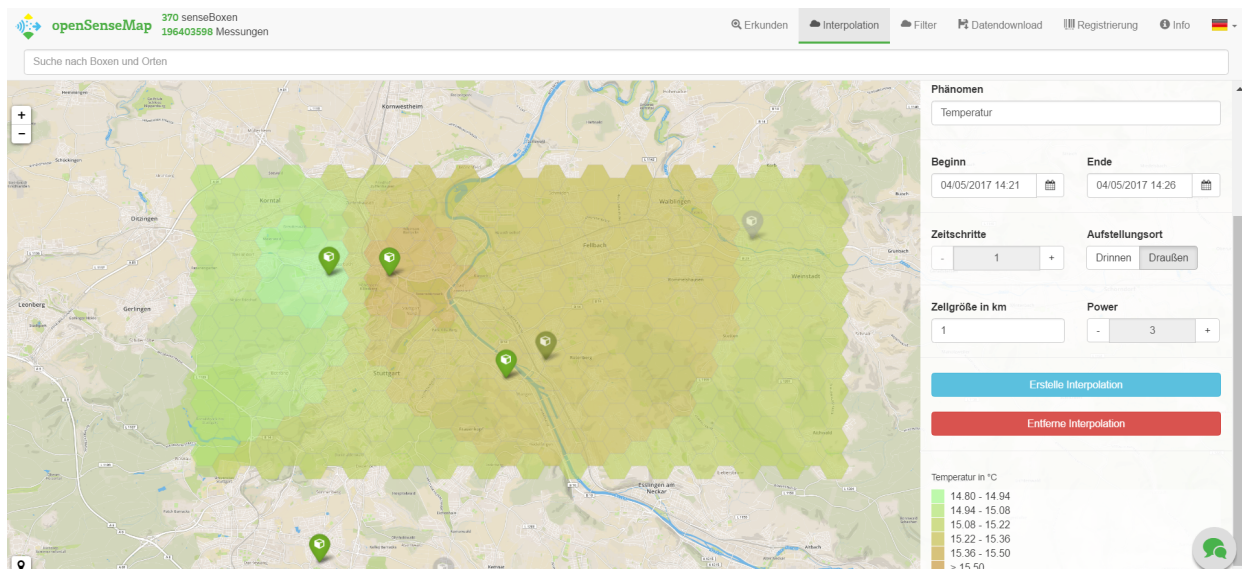


Abbildung 3.2: OpenSenseMap mit Interpolation der Temperatur.

- Ultraschall-Distanzsensoren (HC-SR04)
- Lichtsensoren (TSL45315)
- UV-Sensoren (VEML6070)
- Lichtabhängiger Widerstand
- Mikrofon (CEM-C9745JAD462P2.54R)
- Leuchtdioden (LED)
- RGB-LED (BL-L515)
- Widerstände
- Druckknöpfe
- Potentiometer
- Piezo-Summen.

Anhand der bereitgestellten Anleitung wurde der Arduino mit dem PC verbunden. Nach und nach wurden die vorhandenen Sensoren mit dem Arduino getestet.

Die beiliegenden Jumper-Kabel reichen für den Aufbau nicht aus. Es wurden noch noch kleine Verbindungen hinzugezogen, die man auf dem Breadboard benutzen kann um nicht so viele lange Kabel zu haben. Denn das verwirrt und erschwert die Fehlersuche beim Anschließen.

Es werden keine zusätzlichen Tipps zur geschickten Verwendung eines Breadboards und zur Verteilung der Pins und Anschlüsse mit mehreren Sensoren gegeben.

Das Einbinden mehrerer Sensoren, ausschließlich mit Zuhilfenahme der Anleitung, konnte leider nicht realisiert werden und auch die Chat-Hotline konnte dabei nicht weiterhelfen. Die Fehler im Code wurden dann selbstständig ausgebessert.

Es hat sich leider herausgestellt, dass das mitgelieferte WLAN-Shield defekt ist und sich der Arduino daher auch mit korrigiertem Code nicht mit dem WLAN-Netzwerk verbinden konnte. Es wurde umgehend von einem Mitarbeiter des Instituts für Geoinformatik mitgeteilt, dass das Shield ersetzt wird. Bis zur Fertigstellung dieser Arbeit lag das neue Shield noch nicht vor, sodass keine weiteren Tests durchgeführt werden konnten.

Die Registrierung der SenseBox auf der OpenSenseMap machte mit Zuhilfenahme der Anleitung keine Probleme. Es werden einige Aufgaben mitgeliefert, die wohl für den Unterricht in Schulen oder Projektarbeiten gedacht sind.

Für die meisten Aufgaben stehen keine Lösungen oder zumindest Lösungsansätze zur Verfügung und insgesamt wird kaum auf häufig auftretende Fehler eingegangen.

Hintergründe der Technik werden außer in einem kurzen Abschnitt über I2C und serieller Monitor, nicht behandelt.

Fazit

Wie groß der Lerneffekt mit dem gegebenen Material ausfallen kann, ist nicht direkt ersichtlich und könnte an einer Schülergruppe getestet werden. Um die Schüler Stück für Stück an die Themen Mikrocontroller, Elektronik und Programmierung heranzuführen, wird kein Material angeboten und so muss die Lehrkraft noch ein Konzept und Aufgaben ausarbeiten. Grundlagen über die Hardware und deren Funktionsweise werden nicht mitgegeben. In die Programmierung wird nicht von Grund auf eingeführt. Der zusammengestellte Baukasten enthält alle Hardwarekomponenten, um sinnvolle Versuche und Projekte mit Schülern durchzuführen. Für Lehrkräfte, die mit der Hardware und mit der Programmierung von Arduino oder ähnlichen Mikrocontrollern noch keine Erfahrung haben, wird keine Anleitung oder Einführung angeboten.

Schließlich wird das Thema IoT nicht behandelt oder aufbereitet. Das Projekt ist also eher der Wissenschaft im Sinne von Citizen-Science verschrieben als eine Bildungsinitiative für den Schulunterricht. Für Technikaffine bietet die SenseBox einen fertig zusammengestellten Baukasten mit allem was nötig ist, um einige Versuche und Projekte mit Sensoren durchzuführen. Die Hotline bietet zumindest ein Stück weit Unterstützung.

3.2 Genius - ein Projekt von Daimler

„Genius“, heißt das Projekt für Kinder und Jugendliche von der Daimler AG. Im Subtext des Logos auf der Website steht weiter: „die junge Wissenscommunity von Daimler“. Auf der Startseite der Webpräsenz ist zu lesen, es handele sich um eine MINT-Bildungsinitiative für Kinder und Jugendliche, wobei weiter ganz allgemein angegeben wird, es werden Themen aus Naturwissenschaft und Technik behandelt.⁴ Zielgruppen sind Schulen, sowohl die Grundstufe als auch die Sekundarstufen von Gymnasien und Realschulen. Partner von Genius sind Bildungsbehörden einiger Länder, verschiedene Institute und Landesbildungsinitiativen. In Baden-Württemberg sind diese das Kultusministerium, das Landesinstitut für Schulentwicklung sowie der Landesverband naturwissenschaftlich-technische Jugendbildung.

Das umfangreiche Lehr- und Lernmaterial wird vom KlettMINT-Verlag in Zusammenarbeit mit Lehrkräften verschiedener Schularten, Technik-Didaktikern und Ingenieuren erstellt, ist lizenziert und steht jeweils kostenlos als Download zur Verfügung. Begleitend zu den Themen der Lernmodule werden halbtägige Fahrzeugtechnik-Seminare für Lehrer angeboten, die in das didaktisch aufwendig aufbereitete Unterrichtsmaterial einführen sollen. Es gibt eine Reihe von Arbeitsheften, für die auch Lösungen und Anregungen für die Gestaltung des Unterrichts zur Verfügung stehen. Für einige Lernmodule sind auch praktische Aufgaben vorgesehen für die es eigens entworfene Bausätze käuflich zu erwerben gibt. Die meisten sind über Handelspartner von Genius zu beziehen. Ein Bausatz, passend zum Modul „Aerodynamik“, beinhaltet unter anderem Styroporteile zum ausschneiden, einen kleinen Windkanal und einen Styroporschneider für zusammen etwa vierzehn Euro. Zur Veranschaulichung einer Produktionsstraße können Modellautos mit Fischertechnikteilen in einer Mini-Montagelinie zusammengebaut werden, die als Bausatz im Set für eine Klasse für circa dreihundert Euro angeboten werden. Der Experimentiersatz „Digitaltechnik“ beinhaltet verschiedene Aufbauten und Experimente mit einem Arduino Nano um unterschiedliche Funktionen der

⁴Die in diesem Abschnitt zusammengetragenen Informationen, stammen aus der eigenen Webpräsenz[19] der Initiative.

in einem Auto eingebauten Elektronik zu erlernen. Für einen Teil des Moduls „Sicherheitstechnik“ ist die Verwendung eines Lego-Mindstorms[®]-Roboters vorgesehen, dessen Anschaffungspreis von ab rund vierhundert Euro, allein für die Verwendung im Rahmen dieses Projekts sehr hoch ist. Die Programmierung steht aber insgesamt im Projekt eher im Hintergrund und wird kaum behandelt. Themen, die das Lernmaterial in einzelnen Modulen umfasst, sind folgende:

- Antriebstechnik
- Sicherheitstechnik
- Design, Aerodynamik, Produktionstechnik
- Vernetzte Welt
- Aufbau und Funktion von Automobilen
- Mobilität und Verkehr
- Brennstoffzellen
- Sensoren in der Sicherheits- und Komforttechnik
- Verbrennungsmotor
- Aerodynamik / Strömungswiderstände
- Produktionstechnik
- Künstlerisches und technisches Zeichnen

Die Module sind unabhängig voneinander und lassen der Lehrkraft viel Entscheidungsfreiheit bei der Wahl der Themen. Es ist hier bei einigen Modulen anhand deren Namen schon deutlich zu erkennen, dass der thematische Schwerpunkt in der Automobilindustrie liegt. Dies wird bei näherer Betrachtung der einzelnen Themen noch einmal deutlicher. Die Funktionsweise von Verbrennungsmotor, Elektromotor und Brennstoffzelle wird jeweils in Bezug auf die Verwendung in Automobilen erläutert. Das Modul Sicherheitstechnik bezieht sich auf die technische Ausrüstung in Automobilen und deren technologische Entwicklung in den letzten Jahrzehnten. Funktion und Verwendung von Sensoren in Verbindung mit Fahrerassistenzsystemen werden besprochen.

Das Modul „Vernetzte Welt“ enthält ein Kapitel in dem es um Industrie 4.0 geht. Obwohl es den allgemeinen Titel „Internet der Dinge“ trägt, bleibt der Fokus wieder auf der Automobilindustrie, hier vor allem auf der Fertigung. Fertigungskonzepte, Mensch-Maschine-Interaktion, Vernetzte Produktionsanlagen und Industrie 4.0 sind die Hauptthemen. Hardware zur praktischen Erfahrung des „Internets der Dinge“ wird nicht angeboten.

Das Projekt bietet durchdachtes Lehrmaterial für einen variablen Preis.

Je nachdem welche praktischen Inhalte umgesetzt werden, fallen einige Euro (z.B. 14 Euro für den Aerodynamik Bausatz) bis einige hundert Euro (falls Lego-Mindstorms[®]-Roboter zum Einsatz kommen) an.

Die Begleithefte sind didaktisch intensiv aufbereitet und wie die Seminare kostenlos erhältlich.

Für die Durchführung einiger Projekte ist, trotz des umfangreichen Begleitmaterials, ein größerer Zeitaufwand für die Vorbereitung einzurechnen oder teilweise eben ein hoher Kostenaufwand.

Fazit

Der thematisch starke Bezug zur Automobilindustrie wird nachvollziehbar wenn man sich die Hintergründe des Projektes genauer anschaut. Ziel der Initiative ist es, Nachwuchs heranzuziehen. Daher soll mit einem Einblick in die Berufsbilder der Automobilindustrie Interesse geweckt werden, um dem Mangel an Fachkräften entgegenzuwirken.

Problematisch ist der direkte Bezug zu nur einem Vertreter der Industrie. Susanne Gaschke spricht in ihrem Buch „Die verkaufte Kindheit“ über die „Signalwirkung“ die das „Eindringen eines Unternehmens in den Schutzraum Schule [hat]“ [20] und stellt die Frage in den Raum ob es richtig ist, „Bildung auf berufliche Verwertbarkeit zu reduzieren“ und ob das Bild das Daimler von einem Ingenieur hat, auch das ist, welches die übrige Gesellschaft sucht und von anderen Branchen gebraucht wird. Die Nutzung der Unterlagen erfordert jedenfalls Akzeptanz für die Präsenz eines Konzerns in Schulen, insofern, dass das Logo der Daimler AG auf den Kopiervorlagen des Lernmaterials aufgedruckt ist.

Wie oben beschrieben, wird IoT leider nur am Rande, ohne Praxis und mit alleinigem Fokus auf die Automobilindustrie behandelt.

3.3 Roberta - Eine Initiative des Fraunhofer Institut IAIS

„Roberta - Lernen mit Robotern“ ist ein Projekt des Fraunhofer-Instituts für Intelligente Analyse- und Informationssysteme in Sank Augustin, das seit 2002 besteht. Ziel ist es vor allem Mädchen für Berufe im Bereich von MINT zu begeistern. Das Angebot richtet sich an Schüler und Schülerinnen ab zehn Jahren. Es gibt fundiertes, didaktisch aufbereitetes Lehr- und Lernmaterial für den Einsatz im Schulunterricht. Lehrer können an unterschiedlichen Seminaren teilnehmen und sich in „RobertaRegioZentren“ von Roberta-Teachers Wissen vermitteln lassen, das sie an ihre Schüler weitergeben. Regio Zentren können unter bestimmten Voraussetzungen von Bildungseinrichtungen eröffnet werden. Diese Zentren vermieten Roboterbaukästen, bieten Seminare für Lehrer und Kurse für Schüler an und funktionieren so als Multiplikator. Ein Grundlagenseminar für Lehrer, das über 2 Tage andauert, kostet 290 Euro. Lehrer können sich zum Roberta-Teacher ausbilden lassen und bekommen eine Schulung. In einer didaktischen Handreichung wird die Verwendung des Lehrmaterials und das unterschiedliche Verhalten von Mädchen und Jungen in Schülerkursen thematisiert. Außerdem haben zertifizierte Lehrer Zugang zu zusätzlichem Lehrmaterial.

Zentrales Medium sind die Lego-Mindstorms[®]-Roboter. Ein zusammengestellter Grundbausatz der auf die Verwendung im Projekt abgestimmt ist, kann für circa 400 Euro erworben werden. Eine Schulbox für zwei bis drei Schüler wird für circa 600 Euro angeboten. Zusätzlich ist ein Adapter notwendig, der mit zwanzig Euro zu Buche schlägt.

In Zusammenarbeit mit und Unterstützung von Google ist, als Weiterentwicklung aus Roberta, Open Roberta entstanden. Um technische Hürden in Roberta-Kursen zu minimieren wurde Open Roberta Lab entworfen. Open Roberta Lab ist eine Plattform zur grafischen Programmierung⁵ unterschiedlicher Roboter der Lego-Mindstorms[®]-Serie sowie NAO, Bot'n Roll, micro:bit und Callopie (siehe Kapitel 3.4). Hierfür hat das Fraunhofer IAIS mit „NEPO“ eine Programmiersprache für die grafische Programmierung entwickelt. Der Code wird beim Upload auf den Roboter in textbasierte Sprache umgewandelt. Die cloudbasierte Plattform kann unabhängig vom verwendeten Gerät und Betriebssystem im Browser verwendet werden. Ziel ist den Einstieg in das Programmieren einfacher zu gestalten. Es können keine Syntaxfehler gemacht werden und die Anwender kommen schnell zu einem Ziel. In Abbildung 3.3 ist die Oberfläche zu sehen und auch, dass man sich den textbasierten Code anzeigen lassen kann. Das Fraunhofer IAIS bietet didaktisch aufbereitetes Lehrmaterial in zahlreichen Büchern und CD-Roms, die für rund 25 bis 40 Euro über die Webseite des Projekts bestellt werden können.

Fazit

Insgesamt gehört Open Roberta und Open Roberta Lab somit, unter den in dieser Arbeit im Vergleich stehenden Projekten, zu den eher kostspieligen Möglichkeiten MINT-Themen in den Schulalltag zu integrieren. Der Fokus liegt ganz klar auf der Programmierung. Grundlagenwissen

⁵ „Open Roberta ist von Scratch inspiriert und basiert auf der open-source Bibliothek blockly.“ <https://scratch-dach.info/wiki/Open.Roberta>, Abrufdatum 02.07.2017

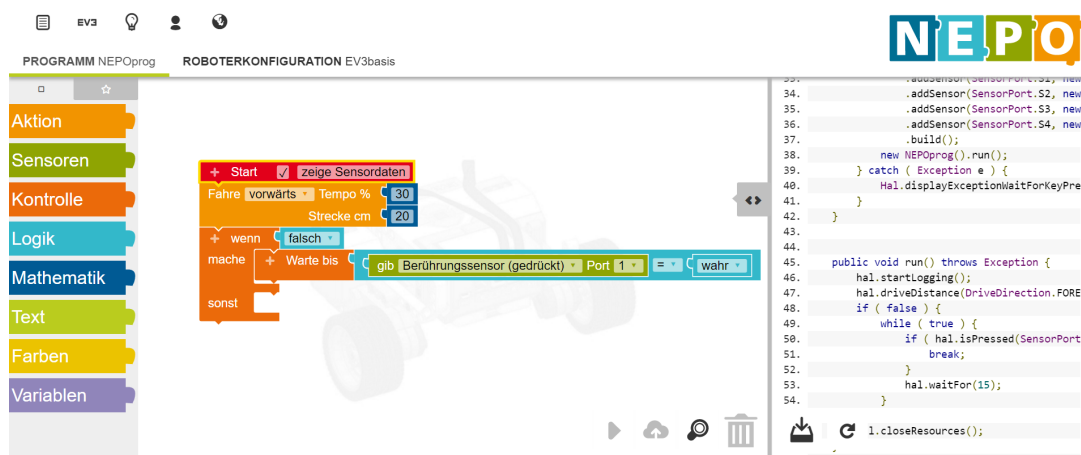


Abbildung 3.3: Grafische Programmieroberfläche Open Roberta Lab.

über die verwendete Hardware und Technik wird im Vergleich kaum vermittelt. IoT kommt in diesem Projekt nicht vor.

3.4 Calliope - Projekt, Mikrocontrollerboard und gemeinnützige GmbH

Calliope ist eine Initiative der gleichnamigen gemeinnützigen GmbH, die mit dem Ziel gegründet wurde, digitale Bildung für Kinder ab 8 Jahren zu bieten. Die Gründer ist neben Geesche Joost, (Professorin für Designforschung an der „Universität der Künste Berlin“, Internetbotschafterin der Bundesregierung und Aufsichtsratsmitglied der SAP SE,) ein bunt gemischtes Team aus einer Inhaberin und einem Interaktionsdesigner eines Unternehmens, das digitales Lernspielzeug entwickelt⁶, einem Diplom-Psychologen und Internet-Unternehmer, einem IT-Berater, einem Mitbegründer der Telekom Innovation Laboratories und einem Informatiker des Fraunhofer Instituts. Zentrales Element ist die Mikrocontrollerplatine CalliopeMini, welche mit den Zielen entworfen wurde, kindgerecht in der Handhabung zu sein und digitale Kompetenzen in Grundschulen zu fördern. Die Platine ist mit Sensoren und Aktoren bestückt. Sie beherrscht Bluetooth und ihre Sternform soll die Handhabung für Kinderhände einfacher machen. Die Sternform (siehe Abbildung 3.4) soll verhindern dass es beim Experimentieren zu Kurzschlüssen kommt. Die Anschlüsse liegen weit genug voneinander entfernt. Das Mikrocontrollerboard kann per USB an einen PC angeschlossen werden, um Programme hochzuladen und ist mit folgenden Komponenten bestückt:

- 25 rote LEDs als Matrix
- eine RGB-LED
- programmierbaren Buttons
- kombinierter Lagesensor mit Bewegungssensor und Kompass
- Bluetooth-Modul
- Helligkeits- und Temperatursensor
- Lautsprecher und Mikrofon
- Sensoren in der Sicherheits- und Komforttechnik

⁶Name des Unternehmens: urbnpockets

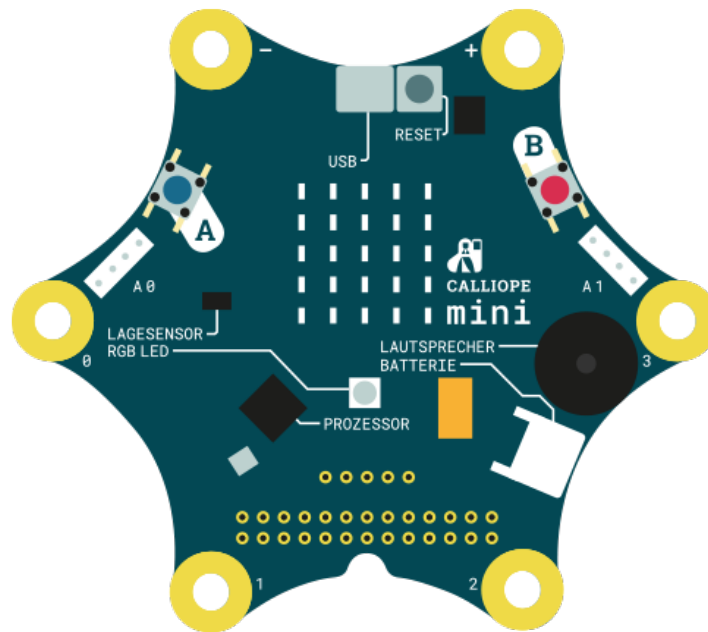


Abbildung 3.4: Schematische Darstellung des CalliopeMini [?].

- Motor-Anschluss

Die Komponenten sind schon auf dem Board verbaut und integriert, es kann nichts selbst gesteckt oder verbunden werden. Es ist aber möglich zusätzliche Sensoren oder Aktoren mit Krokodilklemmen anzuschließen oder die Grove-Stecker⁷, passend für Komponenten vom Hersteller Grove hierfür zu verwenden.

Die Programmierung steht bei dem Projekt im Mittelpunkt. Auf die Funktionsweise der verwendeten Hardware wird bislang, innerhalb des didaktischen Materials, das in Zusammenarbeit mit dem Cornelsen-Verlag entsteht, nicht eingegangen. Microsoft unterstützt das Projekt, indem das Unternehmen die Möglichkeit bietet, den CalliopeMini innerhalb ihrer pxt-Programmierplattform zu programmieren. Außerdem wird er in das Microsoft-Projekt „Code Your Live“ integriert. Weitere Unterstützer sind unter anderem Google, Bosch, NXP, SAP SA, Telekom Stiftung und Roberta, indem sie den Calliope in die Programmierplattform OpenRoberta integriert haben. In einem Pilotprojekt sollen Calliope-Mikrocontroller kostenlos an Grundschulen in Deutschland für Schüler ab Klasse 3 verteilt werden. Das Lehrmaterial führt in die Programmierung mit dem eigenen CalliopeMini-Editor ein, bei dem, wie auch bei pxt und OpenRoberta, in grafischen Blöcken programmiert wird. Eine Lehrerhandreiche kann für rund 13 Euro über die Webseite von Cornelsen⁸ bestellt werden. Der CalliopeMini kann dort für rund 35 Euro vorbestellt werden.

Fazit

Von den in dieser Arbeit untersuchten Projekten bietet Calliope mit dem „Mini“ eines der günstigsten Hardwaremodule an.

Vor allem für eine jüngere Zielgruppe ab acht Jahren ist der CalliopeMini geeignet, um in die Programmierung einzuführen und erste Erfahrungen mit einem Mikrocontroller zu machen.

Die Hardware wird nicht didaktisch thematisiert. Dies würde in Anbetracht der jungen Zielgruppe auch ein besonders abgestimmtes Konzept verlangen, da in der Grundschule nicht auf Vorwissen aus Naturwissenschaften aufgebaut werden kann. Das Lehrmaterial ist gut aufbereitet aber es fehlt

⁷Die ursprünglich von Seedstudio (Hardwareentwicklerplattform) entwickelten Hardwaremodule haben alle ein besonderes Stecksystem für Jumperkabel (Verbindungskabel), welches das korrekte anschließen der Komponenten erleichtert.

⁸<https://www.cornelsen.de/calliope/> stand 08.05.2017.

die Einführung in die Verwendung Hardware.
Das Thema IoT kommt bei Calliope nicht vor.

3.5 Vergleich und Bewertung der Projekte

In der Abbildung 3.5 sind die vorgestellten Projekte tabellarisch gegenübergestellt und die Bewertungskriterien, welche zu Beginn dieses Kapitels schon erläutert wurden, aufgeführt.

Projekt	SenseBox	Genius	(Open) Roberta	Calliope	LetsgoING
IoT	Clouddienst	Theoretisch	Nein	Nein	In Arbeit
Software	Ansätze	Ansätze, 1 Modul	Programmierung	Programmierung	Programmierung
Hardware	nicht thematisiert	Automobiltechnik	kaum	nicht thematisiert	Systemorientiert
Anleitung und didaktische Aufbereitung	Kurz, keine Lehrerhandreiche Hotlinedienst	Ausführlich, auch didaktisches Material, KlettMINT Verlag, Seminare	Seminare für Lehrer, Online Anleitung „Erste Schritte“, Lehrerhandreiche	Lehrerhandreiche, Kurze Anleitungen, Unterrichtsmaterial von Cornelsen	Seminare, Unterrichts- begleitung, Modular aufgebaute An- leitungen, Auf- gaben, Lösungen
Kosten / Schülerset (ca.)	160 €	14 – 30 € jeweils 400 € Roboter	600 € Roboter + 290€ Seminar	35€ Board + 12€ Anleitung	Ca. 60 €
Fazit	Teuer Lehre fehlt Daten sammeln	Leider sehr Automobil- und Konzernbezogen	Teuer Technik nicht thematisiert	Guter Ansatz aber Technik fehlt Jüngere Zielgruppe!	Preiswert, didaktisch gut aufgebaut, I und T

Abbildung 3.5: Vergleich der Projekte und Bewertung im Sinne der genannten Kriterien: grün: positiv, rot: negativ, gelb: unbefriedigend oder neutral.

Themen aus dem Bereich IoT beinhalten nur die beiden Projekte SenseBox und Genius. Mit der SenseBox könnte IoT, anhand des Clouddienstes den diese mit ihrer OpenSenseMap nutzt, behandelt werden. Das Thema bleibt aber außen vor. Genius thematisiert IoT zwar, jedoch passiert das nur theoretisch, nicht in Zusammenhang mit Hardware und nur mit Bezug auf die Automobilbranche und Industrie 4.0. Im Gegensatz dazu stehen die beiden Projekte Roberta und Calliope, bei denen das Internet der Dinge ganz außen vor bleibt.

Fundiert aufbereitetes Begleit- und Unterrichtsmaterial bieten die beiden Projekte Calliope und Roberta. Bei Genius gibt es viele ergänzende Informationen für Schüler und Lehrkräfte. letsgoING hält für Lehrer ein umfangreiches Angebot für die Einarbeitung und für den Unterricht bereit. Neben Seminaren und Anleitungen werden mit einer jüngst begonnenen Initiative schon Lehramtsstudenten im Studium einbezogen und so der Schneeballeffekt ausgenutzt. Die genannten Projekte, Calliope, Roberta, Genius und letsgoING sind also für den Einsatz im Unterricht gedacht und konzipiert. Wie in Abschnitt 3.1 beschrieben, bietet die SenseBox wenig didaktisch aufbereitetes Material und ist in der vorliegenden Form für den Einsatz im Schulunterricht nicht sehr gut geeignet. Nur wenn die Lehrkraft über ausreichende Vorkenntnisse im Umgang mit den Komponenten verfügt, ist die SenseBox geeignet. Dann muss aber der Unterricht noch vorbereitet werden. Aufgaben müssen zudem noch ausgearbeitet werden, um den Baukasten sinnvoll einsetzen zu können.

In den ausgesuchten Themen für Lernziele unterscheiden sich die Projekte. Die SenseBox bietet insgesamt, wie schon erwähnt, wenig Informationen dazu, was die Schüler mit dem Projekt lernen können. Roberta und Calliope sind in diesem Punkt wieder sehr ähnlich. Die Programmierung steht bei beiden stark im Vordergrund und ist das zentrale Lernziel. Sie sind so strukturiert, dass in den Kapiteln und Modulen schrittweise das Niveau der Programmieraufgaben steigt. Die Erklärung der Hardware bleibt bei beiden Projekten jedoch aus. Es zieht sich aber jeweils ein roter Faden durch die Didaktik bei der Projektausführung.

Dieser ist bei Genius so nicht zu erkennen. Die Genius-Module bauen inhaltlich nicht aufeinander auf und nur der Bezug zum Automobil verbindet alle miteinander. Das zentrale Thema ist also das Automobil und Ziel ist das kennen lernen der Automobilherstellung.

letsgoING dagegen verfolgt ein umfassendes Konzept, das sowohl in die Programmierung als auch in die Verwendung der Hardware einführt und sich in den Methodenkompetenzen nahe am Bildungsplan für NwT bewegt. letsgoING steht in engem Kontakt zu den Schulen und Lehrkräften, wie im Kapitel 2 dargelegt ist. Es werden Grundlagen in Programmierung und Benutzung der Hardware vermittelt. In jedem Projektabschnitt wird auf bisherige Erfahrung und Gelerntes zurückgegriffen und neues Wissen darauf aufgebaut.

Die Kosten fallen bei den Projekten sehr unterschiedlich aus. Roberta beinhaltet hohe Seminar-kosten. Die Anschaffungskosten für die benötigte Hardware sind ebenfalls sehr hoch. Bei Genius variieren die Preise zwischen den Modulen stark. Calliope ist mit 35 Euro für das Mikrocontroller-board am günstigsten und letsgoING mit rund 60 Euro ein wenig teurer. Das letsgoING-Material bietet aber auch eine umfangreichere Ausstattung.

Die Hardware von Calliope ist am ähnlichsten zu letsgoING. Ein entscheidender Unterschied ist, dass bei Calliope die Sensoren und Aktoren auf der Platine schon integriert sind. Bei letsgoING werden diese von den Schülern selbst angeschlossen und sind greifbar, was ein besseres Verständnis und eine nachhaltige Arbeitsweise, wie im Kapitel 2 beschrieben, unterstützt. Das Denken in Schnittstellen und Verwenden von Blockschaltbildern wird gefordert und gefördert.

Die didaktische Aufbereitung und Begleitung der Lehrkräfte bei letsgoING ist intensiv, nachhaltig und zielführend. Die Aufgaben sind in Niveau, Inhalt und Lernweise auf die Zielgruppe abgestimmt. Der modulare Aufbau des Inhalts ist übersichtlich und bietet der Lehrkraft Flexibilität bei der Unterrichtsgestaltung.

Zusammenfassend kann man feststellen, dass sich letsgoING von anderen Projekten vor allem durch sein didaktisches Konzept unterscheidet. Dieses deckt sich in einigen Punkten mit dem Bildungsplan für das Fach NwT. Das Projekt findet im regulären Unterricht statt. Ganz bewusst werden damit alle Schüler angesprochen, nicht nur die ohnehin interessierten, die sich bei anderen Projekten in Arbeitsgruppen zusammenfinden können. Der Preis ist, mit zirka sechzig Euro pro Materialset, im Vergleich noch günstig.

Somit hebt sich letsgoING, mit seinem inhaltlich und thematisch umfassenden Lehrkonzept und der engen Zusammenarbeit mit Schulen, von den übrigen untersuchten Projekten ab. Es eignet sich aufgrund des modularen Projektaufbaus gut, um IoT als weiteres Lernmodul aufzunehmen. Sowohl die Struktur des bestehenden Lernmaterials als auch der Hardware ermöglichen ein gut angepasstes Einbinden von IoT-Anwendungen.

Kapitel 4

Grundlagen

4.1 Typischer Aufbau eines Mikrocontrollers

Im Schülerprojekt kommen mehrere Sensoren und Aktoren zum Einsatz, deren Funktionen koordiniert werden müssen. Sensoren messen Phänomene in ihrer Umgebung und stellen ihre Ergebnisse als Information bereit. Aktoren können auf diese Information entsprechend reagieren. Jedoch nur, wenn zwischen beiden Komponenten ein Vermittler agiert, der die Information so aufbereitet, dass sie für alle Beteiligten brauchbar wird. Hierfür kommt eine besondere Art Computer zum Einsatz. Er kann auch als das „Gehirn“ eines aus vielen Komponenten bestehenden Systems bezeichnet werden, denn er bekommt Information („Sinneswahrnehmung“), stellt Berechnungen an („denkt nach“) und erteilt, entsprechend der Ergebnisse, Befehle an andere Komponenten. Der Computer, der diese Steuerungs- und Kommunikationsaufgaben erledigt, ist ein Mikrocontroller (kurz: μC). In der Abbildung 4.1 sind die Kommunikationswege schematisch dargestellt. Der μC stellt auch Verbindungen zum Benutzer zur bereit. Über diese Schnittstellen kann Software auf den μC geladen

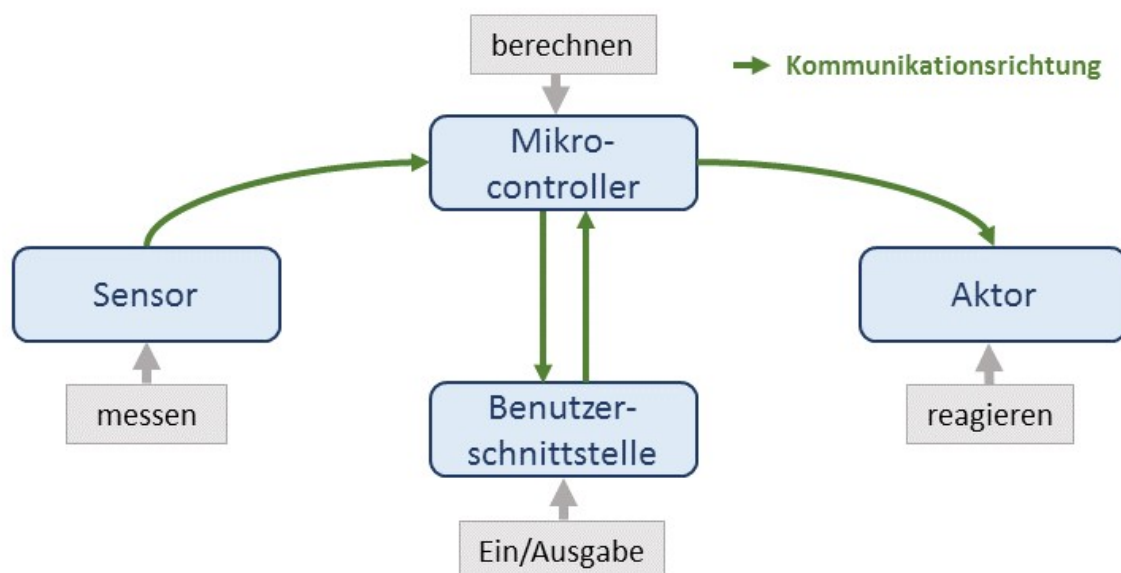


Abbildung 4.1: Kommunikation zwischen einem Mikrocontroller und seiner Umgebung.

werden und die Ein- und Ausgabe von Information erfolgen.

Der μC ist ein Computer, der dem PC zwar sehr ähnlich ist, sich aber in wesentlichen Aspekten von ihm unterscheidet. Ein PC besteht aus mehreren getrennten Komponenten: Prozessor, RAM-

und ROM-Speicher, Netzwerkkarte, Soundkarte, Grafikkarte, USB-Karte, WLAN-Karte und Ein-/Ausgabeschnittstellen für den Benutzer (Tastatur und Bildschirm). Der PC hat ein Betriebssystem mit benutzerfreundlicher (grafischer) Bedienoberfläche und ist sehr flexibel einsetzbar und personalisierbar. Es können Programme installiert und deinstalliert, sowie verschiedene Einstellungen vorgenommen werden. Der μC wird meistens ohne Betriebssystem ausgeliefert und nur einmal programmiert. Dafür ist sogar ein spezieller Adapter und Software notwendig. Einmal programmiert, erfüllt er dann nur eine für ihn bestimmte Aufgabe. Er hat in der Regel weniger Komponenten als ein PC. Wenn Kommunikationsschnittstellen für den Benutzer vorhanden sind, bestehen sie meistens nur aus ein paar Tastern und einem kleinen Display. μC haben gegenüber PCs einige wichtige Vorteile: sie sind viel günstiger, kleiner und energieeffizienter.

Das Kernstück eines μC ist, wie beim PC, sein Prozessor. μC sind in Haushaltsgeräten wie Kaffeemaschinen, Mikrowellen, Spülmaschinen, Waschmaschinen und Kühlschränken, in Unterhaltungselektronik (Fernseher, Audioanlage), in Druckern, Bildschirmen und vielen anderen Elektrogeräten aber auch in Kleinelektronik wie Uhren und sogar in Chipkarten zu finden. In Autos kommen gleich eine Vielzahl der kleinen Computer zum Einsatz. Oft sind diese noch um weitere externe Komponenten, zum Beispiel Eingabe- und Anzeigemodule, ergänzt. Man nennt sie dann „eingebettete Systeme“ und im Auto auch „Steuergeräte“. In der industriellen Automation heißen sie einfach „Steuerung“. Der Begriff leitet sich aus dem Umstand ab, dass die μC in einem solchen technischen Umfeld, die Datenverarbeitung für die Überwachung und Steuerung übernehmen [22].

Es gibt jedoch maßgebende Unterschiede zum PC, die Grund dafür sind, dass gerade ein μC in vielen dieser Geräte Verwendung findet. Vor allem seine geringe Größe, die Integration vieler Bausteine auf einem einzigen Chip und die Möglichkeit den Prozessor des μC so herzustellen, dass er für eine bestimmte Anwendung spezialisiert ist, unterscheidet ihn von PCs. Ein μC braucht keine Kühlung, verbraucht wenig Strom, ist ohne weitere externe Elemente betriebsbereit und sehr günstig.

In der Abbildung 4.2 sind die wichtigsten Bausteine dargestellt, aus denen ein μC besteht. Die drei Hauptkomponenten sind Prozessor, Speicher und Ein-/Ausgabeeinheiten, die oft als „Pins“ bezeichnet werden. Das Kernstück ist der Prozessor (CPU, Central Processing Unit). In μC verbaute Prozessoren werden aufgrund ihrer Größe oft auch als Mikroprozessoren bezeichnet. Diese sind auf einem Halbleiterchip untergebracht und in zwei Einheiten unterteilt. Das für Rechenoperationen zuständige Operationswerk (ALU, Arithmetical Logical Unit), führt arithmetische Berechnungen und logische Operationen (boolesche Algebra) durch. Die Steuereinheit (CU, Control Unit) koordiniert den Ablauf der Rechenoperationen. Der Prozessor kann unterschiedliche Verarbeitungsbreiten (Datenbreiten) haben, meistens 8, 16 oder 32 Bit, in einzelnen oft älteren Modellen oder wenn nur wenig Leistung bei geringen Kosten notwendig ist, auch noch 4 Bit. [22]. Der Arduino Uno welcher im Projekt letsGOING eingesetzt ist, hat einen Prozessor mit 8 Bit Datenbreite integriert¹. Die Datenbreite bestimmt die Größe der Zahlen mit denen der Prozessor rechnen kann. Über die Software können aber auch größere Zahlen bearbeitet werden, allerdings oft zulasten der Geschwindigkeit. Es gibt zwei grundlegende Arten von Prozessorarchitekturen, die sich durch ihren Befehlssatz unterscheiden. CISC-Prozessoren (Complex Instruction Set Computer) können viele verschiedene Befehle verarbeiten, die in einigen Fällen aber nicht nötig sind. Es entstanden aus den CISC- die RISC-Prozessoren (Reduced Instruction Set Computer), welche eine reduzierte Anzahl Befehle beherrschen und schneller sein können.

Neben dem Prozessor gibt es einen Speicher, in dem Programme und Daten stehen. Diese liegen in der Regel in einem getrennten Speicher (bei RISC). Es wird zwischen flüchtigem Schreib- und Lesespeicher für Daten und nicht flüchtigem Festwertspeicher für Programme unterschieden. Der Datenspeicher wird als flüchtig bezeichnet, da in ihm gespeicherte Information kurzfristig zur

¹Datenblatt des im Arduino verbauten Mikrokontrollers von Atmel: http://www.atmel.com/images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P_datasheet_Complete.pdf

Verfügung steht und nur solange Betriebsspannung anliegt. Es ist meist ein SRAM-Speicher (Static Random Access Memory) integriert, der lediglich wenige Bytes oder nur einige KBytes groß ist. Der Programmspeicher ist ein ROM-Speicher, den es in verschiedenen Varianten gibt und dessen Speicherinhalt auch nach Abschalten der Betriebsspannung erhalten bleibt. Der ursprüngliche ROM-Speicher (Read Only Memory) wird bei der Herstellung beschrieben und kann nicht mehr verändert werden. Mit diesem Speicher wurden die ersten μC ausgestattet und er ist heute noch durch die günstige Herstellung in Produkten mit hoher Stückzahl zu finden. PROM-Speicher (Programmable ROM) können dagegen auch nach der Herstellung programmiert werden, jedoch nur ein Mal. Wenn der Programmcode sein Herstellerhaus nicht verlassen soll, ist PROM oft die Wahl bei großen Stückzahlen. In μC sind inzwischen aber oft EEPROM-Speicher (Electrically Erasable PROM) zu finden, welche beliebig oft beschrieben und mit einer bestimmten elektrischen Spannung gelöscht werden können. Sehr schnelle EEPROM sind Flash-Speicher. Der ESP32 verfügt unter anderem über einen solchen kleinen Flash-Speicher, direkt auf dem Chip, für Endanwenderprogramme. [22]

Damit der μC seine zentrale Aufgabe erfüllen kann, nämlich mit der Peripherie zu kommunizieren und Sensoren und Aktoren zu koordinieren, verfügt er über Ein- /Ausgabeeinheiten. Dies sind einerseits digitale Allzweckkanäle (General Purpose I/O, GPIO) die oft verwendet werden um Zustände (HIGH/LOW) zu übertragen. Außerdem gibt es analoge Ein- /Ausgabeeinheiten über die Signale in Form von Spannungen übertragen werden. Analoge Eingangssignale müssen von einem Analog-Digital-Wandler in digitale Signale gewandelt werden, denn der Prozessor kann nur digitale Werte verarbeiten. Für die Gegenrichtung ist ein Digital-Analog-Wandler notwendig, der die digitalen Signale des μC für Aktoren in analoge Spannungswerte wandelt. Außerdem gibt es oft noch serielle Schnittstellen für die Kommunikation, zum Beispiel für das Übertragen von Programmen auf den Controller. [22]

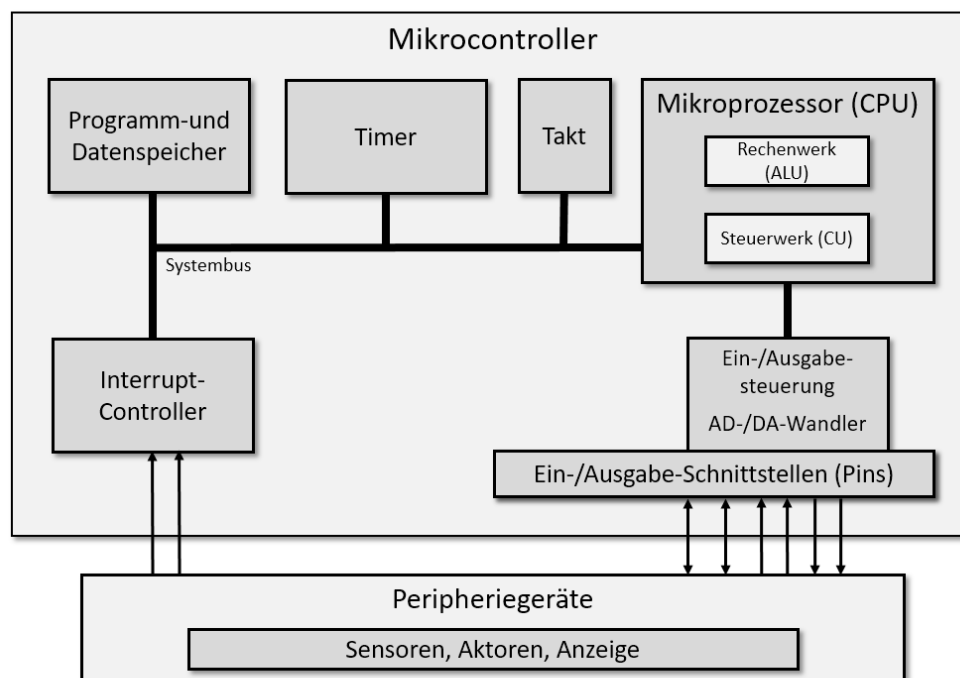


Abbildung 4.2: Kommunikation zwischen den Komponenten eines μC s und seiner Umgebung. Vgl. [22].

Die meisten μC verfügen noch über weitere Komponenten. Ein Zeitgeber (Timer) oder Zähler misst

die Zeit oder zählt einen Takt, den der Taktgeber des Controllers vorgibt und erfüllt eine Reihe wichtiger Prozesssteueraufgaben. Der Zeitgeber ist unter anderem notwendig um nach Ablauf einer bestimmten Zeit einen regelmäßigen Impuls auszulösen und für die Real Time Clock (RTC), welche die Zeit bis auf Zehntel Sekunden genau angibt. Der Interrupt-Controller verwaltet Unterbrechungsanfragen, die an den Prozessor gestellt werden. Diese können, von über Pins angeschlossene Hardwarekomponenten oder von der Softwareseite, ausgelöst werden. Das Programm wird dann unterbrochen bis die Interrupt-Routine ausgeführt wurde und anschließend fortgesetzt. Interrupts sind wichtig, um das Blockieren des Programmablaufs zu verhindern, wenn kurze Anweisungen immer wieder zyklisch ablaufen müssen.

Eine besonderer Typ von μC ist ein SoC (System on a Chip). Die Grenze zwischen μC und SoC verläuft fließend. SoCs haben mehr Komponenten als ein μC , wie z.B. eine WLAN-Schnittstelle oder Sensoren auf einem einzigen Chip vereint. SoCs lassen eine noch kleinere Gesamtgröße und geringere Stromverluste durch enge Bauweise zu. Sie sind meistens spezialisiert und angepasst um bestimmte Aufgaben zu erfüllen und leistungsfähiger als μCs . [23]

Der ESP32 ist ein solches SoC. Für diese Arbeit wird ein Modell verwendet (Sparkfun ESP32 Thing), das mit anderen Komponenten auf einem Entwicklerboard integriert ist. Auf diesem sind Anschlüsse vom Chip herausgeführt, um das Anschließen von Peripheriekomponenten zu erleichtern. Das System wird in Kapitel 6 genauer beschrieben.

4.2 Kommunikation im Internet

Die Wurzeln des uns heute bekannten Internets gehen bis auf das ARPANET zurück, ein Militärprojekt der USA, das zuverlässige Kommunikation im Kriegsfall gewährleisten sollte. Das Netzwerk wurde 1968 verwirklicht und verband zunächst einige über die USA verteilte Rechner. Seine Robustheit erlangte es durch die Topologie (Struktur) des Netzes, das bei Ausfall eines Rechners erlaubte, mit den übrigen weiter zu kommunizieren. Eine Abspaltung vom ARPANET wurde dann auch für wissenschaftliche Zwecke von Universitäten in den USA genutzt. Die Übertragung der Information fand schon in diesen Netzwerken über Datenpakete statt. Auch in Europa gab es, neben nationalen Netzen, eine Verknüpfung von Forschungsinstituten durch ein europaweites Netzwerk. In der Wissenschaft kam bald der Ruf nach einem größeren, weiter reichenden Netzwerk für den interkontinentalen Informationsaustausch auf. Es entstand ein Dienst, der uns heute als WWW (World Wide Web) bekannt ist. Dieses logische Netzwerk konnte auf das bestehende physikalische Netzwerk aus Rechnern aufbauen. Wachsende Datenmengen verlangten nach einem neuen System zum Aufrufen von Informationsquellen, denn bisher war dafür der Serveradministrator zuständig. Hyperlinks, mit denen Dokumente direkt von ihren Eigentümern untereinander verlinkt werden, lösten den Administrator ab. Außerdem wurde der noch heute verwendete URL (Uniform Resource Locator) als Dokumenten-Referenz eingeführt. HTML (Hypertext Markup Language) ist die Seitenbeschreibungssprache in der Dokumente verfasst sind und die ebenfalls bis heute, inzwischen in der Version HTML 5, Verwendung findet. Genauso HTTP (Hypertext Transfer Protocol, Abschnitt 4.2.3), ein Protokoll, das für die Verknüpfung der Dokumente im WWW zuständig ist. Wie alle Internetdienste (z. B. E-Mail, Cloudservice), nutzt das WWW als physikalische Grundlage Server, also Rechner die sich weltweit verteilt befinden und über das Internet vernetzt sind. Für die Darstellung der Inhalte war schon damals ein Browser auf dem eigenen Rechner zuständig, der das HTML-Protokoll versteht und interpretiert. Dieser, auch als Client bezeichnete Gegenspieler des Server, ruft Informationen auf diesem ab und zeigt sie an. Der Dienst des WWW machte die Koordination des Datenaustausches in einem weltweiten Netzwerk möglich. Im Internet werden Informationen in Datenpaketen übertragen. Die dafür verwendeten Protokolle heißen TCP (Transmission Control Protocol) und UDP (User Datagram Protocol) und basieren auf dem IP (Internet Protocol). Da diese Protokolle die Grundlage des Internets bilden, wird dieses auch als IP-Netzwerk bezeichnet, genauso wie jedes (auch lokale) Netzwerk das auf IP basiert. [24] Schließlich sind dann noch die physikalischen Komponenten notwendig, die eine Verbindung im Netzwerk herstellen. Dies

sind unter anderem Modems, Router, Gateways und Access Points.

4.2.1 Internet Protocol und TCP/UDP

Die Kommunikation im Internet, also die Übertragung von Information zwischen einem Sender und Empfänger findet über das Internet Protocol (IP) statt. Informationen werden in Pakete aufgeteilt bevor sie übertragen werden. Dies wird als Paketvermittlung bezeichnet. Dem IP liegt ein Adressierungsschema zu Grunde, das auf numerischen IP-Adressen beruht. Jedes Gerät in einem Netzwerk hat eine eigene IP-Adresse, über die es identifiziert werden kann. Ähnlich wie die postalische Anschrift einer Person, bestehend aus Ort und Straße. Diese gilt immer innerhalb des Netzwerks. Das Internet ist ein Netzwerk, an das viele Subnetze angebunden sind. Als Beispiel soll das Heimnetzwerk dienen. Es ist auch ein solches Subnetz. Der Router verbindet das Subnetz über ein Modem mit dem Internet. Der Router hat eine im Internet bekannte IP-Adresse. Die IP-Adressen der Geräte die zu Hause mit dem Router verbunden sind, bleiben dem Internet verschlossen. Ein Gerät kann, wenn es in ein anderes (Sub)netzwerk verlegt wird, eine andere IP-Adresse haben. Denn diese vergibt der Router und nur er kennt die IP-Adressen der Geräte, die mit ihm verbunden sind. Der Router sendet Daten so ins Internet, als würden sie von ihm selbst kommen und empfängt alle Daten für „seine“ Geräte, die aber an ihn adressiert sind. So kann das Ziel der Datenpakete mittels des Routers eindeutig festgelegt werden.

Das Internet Protocol kümmert sich lediglich um die Übertragung von Datenpaketen und nicht darum „ob ein Paket sein Ziel erreicht, dass die Reihenfolge der Pakete bei der Übertragung erhalten bleibt oder welche Laufzeit die Pakete haben“ [25]. Eine Nachricht wird also in Pakete unterteilt, ähnlich wie beim Versand einer großen Bestellung im Internet, etwa das Ausstellungsstück eines Kleiderschranks von einem Möbelhaus. Dieser wird zerlegt und seine Komponenten in mehrere Pakete aufgeteilt. Die einen enthalten Schrauben, andere Regalböden und wieder andere Türen. Die Pakete werden durchnummeriert. Dann bekommen alle einen Adressaufkleber mit dem Empfänger und eventuell mit dem Absender. Dies erledigt das IP indem es jedes Datenpaket mit einem Header versieht. Zuhause wird dann Stück für Stück, nach der Nummerierung und einer Anleitung alles wieder zusammgebaut. Zuvor müssen die Pakete aber sicher transportiert werden. Dies erledigen hauptsächlich zwei Protokolle: entweder das Transmission Control Protocol (TCP) oder das User Datagram Protocol (UDP). TCP ist zuverlässiger, da es eine Fehlerabfrage beinhaltet, sodass ein beschädigtes oder fehlerhaftes Paket nochmals durch den Empfänger, vom Sender angefordert werden kann. UDP ist schneller, aber Übertragungsfehler können nicht behoben werden. [26]

4.2.2 Webserver

Wird im Browser eine Website aufgerufen, werden Inhalte wie Text und Bilder oder Videos angezeigt. Meistens ist sogar noch weitere Interaktion möglich, indem man einen Link klickt, ein Feld ausfüllt, auf eine Schaltfläche klickt oder dergleichen.

Dabei stellt ein so genannter Client (der Browser) eine Anfrage an den Server, welcher diese dann beantwortet. Webserver sind Programme auf Rechnern, die ebenfalls als Server bezeichnet werden und immer online sind. Als zentraler Bestandteil des Internets bieten Server Dienste an und warten ständig auf Anfragen. Information, die über das Internet abrufbar ist, wird von Servern bereitgestellt und ist auf Serverrechnern gespeichert. Webseiten beispielsweise, setzen sich aus vielen Dateien zusammen, in denen deren Inhalt und Darstellung festgelegt ist. Diese Dateien sind auf Servern gespeichert. [27]

Das Serverprogramm, auch als Serverprozess bezeichnet, bekommt vom Client, ebenfalls ein Programm oder Prozess (z. B. der Browser), eine Anfrage für einen Dienst und sendet eine Antwort zurück. Ein Dienst ist ein Programm, das eine bestimmte definierte Funktion erfüllt. Webserver weisen ihren Diensten Portnummern zu. Anfragen werden dann an die entsprechenden Programme auf dem Server weitergereicht. Eine Anfrage an Port 80 betrifft das Webserverprogramm, das

sich um das Anzeigen von Internetseiten kümmert. Wird im Browser eine Adresse eingegeben, z.B. „<http://www.tec.reutlingen-university.de/projekte/letsgoing/>“ kontaktiert der Browser (Client) den Server über www.tec.reutlingen-university.de auf Port 80. Das Webserverprogramm akzeptiert die Verbindung wiederum. Mit dem Zusatz „/projekte/letsgoing/“ in der URL zeigt der Browser an, dass er eine bestimmte Seite anzeigen möchte. Das Webserverprogramm übergibt die Information über Inhalt, Darstellung und Funktionalitäten der Seite an den Browser und schließt die Verbindung. Dieser wertet die Information (meist eine HTML-Datei, siehe Abschnitt 4.2.4) aus und zeigt die Seite an. Beim Klicken einer Schaltfläche auf der Seite wird erneut eine Anfrage vom Browser an den Webserver gesendet und der Ablauf wiederholt sich. Für den Aufbau einer Webpage sind meistens mehrere Anfragen an den Server notwendig. Dies hängt von der Komplexität der Seite ab. [28]

Unter Webservice versteht man die Dienstleistung eines oft als „Cloud“ bezeichneten Servers. Bildlich gesprochen steht die „Wolke“ (engl. Cloud) für das Netzwerk, in dem sich Server und PCs befinden und Informationen austauschen. Dabei hat sich der Begriff „Cloud Computing“ verbreitet, den das Bundesamt für Sicherheit und Informationstechnik als „das dynamisch an den Bedarf angepasste Anbieten, Nutzen und Abrechnen von IT-Dienstleistungen über ein Netz“ definiert.

Dienstleistungen im Rahmen von Cloud Computing werden extern angeboten, also ohne Installation auf dem eigenen PC. Sie umfassen Software, Datenspeicher und Rechenleistung. Bekannte und von vielen privat aber auch beruflich genutzte Clouddienstleistungen sind externe Datenspeicher, die dazu dienen, Dokumente und Bilder oder andere Medien aufzubewahren. Diese Daten sind dann auf Servern gespeichert und können von überall aus, über das Internet erreicht werden. Beispiele hierfür sind OneDrive für Windows oder iCloud für Apple-Anwendungen und die unabhängig vom Betriebssystem zu verwendende Dropbox. Die OpenSenseMap aus Abschnitt 3.1 ist ebenfalls eine Cloudanwendung. Die Messdaten werden auf Servern gespeichert und sind über das Internet dort wieder abrufbar. Firmen und andere große Einrichtungen haben oft eine Virtual Private Cloud, die im Firmeneigenen Netzwerk lokal oder über einen Sicherheitszugang über das Internet genutzt werden kann.

Die Bezeichnung der Webservices als „Wolke“ macht auch deutlich, dass die Wege, welche die Daten nehmen und deren genauer Ort für den Nutzer nicht offensichtlich und unwichtig sind. Server können überall auf der Welt stehen und müssen sich lediglich in der „Internetwolke“, befinden, also Zugang zum Internet haben. [27]

4.2.3 Hypertext Transfer Protocol - HTTP

Das Hypertext Transfer Protocol (HTTP) ist essentiell für die Kommunikation und das am weitesten verbreitete Kommunikationsprotokoll im Internet. Es kümmert sich um die Übermittlung von Daten zwischen Webclient und Webserver [24]. Webserver die HTTP nutzen werden auch HTTP-Server genannt. Über HTTP werden Dateien vom Webserver angefordert, um sie in den Browser zu laden. Dieser kümmert sich dann um die Darstellung der enthaltenen Daten, wie Texte, Bilder und Videos.

Der Kommunikation mit HTTP liegt ein Client-Server-Prinzip zugrunde. Ein Benutzer fordert im Browser (Client) zum Beispiel durch Klicken eines Links eine Webseite an. Der Browser sendet dann eine Anforderung, einen sogenannten HTTP-Request an den Server. Die Adresse des Servers ist als URL angegeben. Der Server erhält die Anfrage und sendet eine Antwort in Form einer HTTP-Response. Diese Antwort enthält die Daten für den Client (Browser). Die HTTP-Verbindung wird nach dem Senden der Antwort beendet. Meistens finden mehrere dieser Verbindungen gleichzeitig statt. Abbildung 4.3 zeigt eine schematische Darstellung des Kommunikationsmodells.

Die Anfrage (Request) bekommt der Server über das Protokoll „TCP“ (Abschnitt 4.2.1) auf seinem Port 80, der wie in Abschnitt 4.2.2 beschrieben, Webserverdiensten zugewiesen ist.

Request- und Responsemeldungen bestehen aus einem Text, der in „Header“ und Daten unterteilt

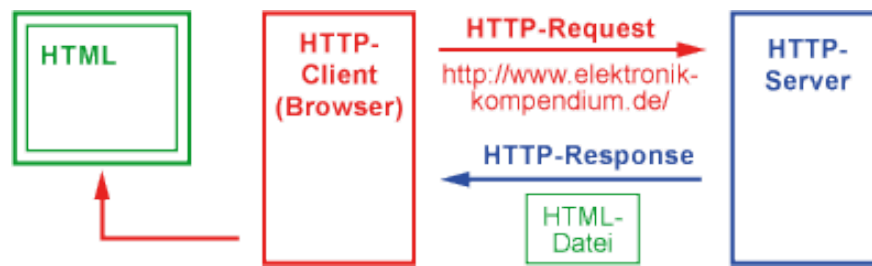


Abbildung 4.3: Server-Client-Prinzip des HTTP [29].

ist. Der Header enthält Informationen (für den Server oder den Client) darüber, was mit der Meldung und den Daten zu tun ist oder passiert ist. Die Meldung wird jeweils über den URL (in der Adresszeile des Browsers zu finden) übermittelt.

Der Client sendet einen Request an den Server in dem er angibt, welche Daten er erhalten möchte. Im URL wird zuerst das Transportprotokoll angegeben mit „http://“. Darauf folgt der Name der Domain und anschließend der Port. Wenn die Verbindung über den Port 80 abgewickelt wird, kann diese Angabe wegfallen. Dann folgt die Angabe des Dateipfades und schließlich die Datei. Der Header könnte dann so aussehen: „http://beispiel-domain.de/beispiele/beispielEins.html“.

Der HTTP-Request und die HTTP-Response unterscheiden sich natürlicherweise inhaltlich.

HTTP-Request

Der Request durch den Client beginnt mit der Angabe einer Methode. Diese gibt dem Server an, was er machen soll. Häufig werden die beiden Methoden „GET“ und „POST“ verwendet. Mit der GET-Methode wird eine Quelle, zum Beispiel eine HTML-Datei angefordert. Mit GET können Formulardaten aus einem Webformular übermittelt werden. Wenn in einem Bestellschritt in einem Onlineshop die persönliche Anschrift abgefragt wird, erfolgt dies über ein Formular. Mit einem Button-Klick (z.B. „weiter“ oder „OK“) werden die Daten mittels Request an den Server übermittelt. Die Daten stehen dann codiert und getrennt durch ein Fragezeichen in dem URL. Die POST-Methode wird ebenfalls für Formulardaten verwendet, und zwar wenn diese an ein Programm übermittelt werden. Sie funktioniert ähnlich wie die GET-Methode ist aber abweichend codiert. Oft wird die POST-Methode für Formulardaten bevorzugt, da die Daten nicht wie bei der GET-Methode im URL für den Anwender sichtbar sind. Es gibt noch einige weitere Methoden. Eine ebenfalls häufig verwendete Methode ist „PUT“. Sie dient zum Ändern von Daten auf dem Server.

HTTP-Response

Nachdem der Server einen Request erhalten hat, sendet er eine Response-Meldung an den Client. Diese enthält einen Response-Code, der vom Browser identifiziert wird. Es sind eine Reihe von Codes festgelegt, die unterschiedliche Funktionen haben. Ein Beispiel ist das Anzeigen einer Fehlermeldung, falls eine angeforderte Datei nicht gefunden werden konnte (Fehlercode 404). Auf den Response Code folgt eine Textmeldung entsprechend dem Request des Clients. Die Response enthält noch weitere Informationen über den Server und die Verbindung.

4.2.4 HTML, CSS und Javascript - Aufbau einer Webpage

Eine Webpage besteht aus einzelnen, auf bestimmte Weise angeordneten Elementen. Diese haben ein unterschiedliches Erscheinungsbild (Farben, Textgrößen, Position etc.). Meistens sind auch noch Funktionen implementiert, wie zum Beispiel bei Buttons. Klickt man darauf, soll etwas geschehen. Am Erscheinungsbild einer Webpage sind drei Sprachen beteiligt, die ein Zusammenspiel aus Struktur, Stil und Verhalten bilden. Dies sind HTML, CSS (Cascading Style Sheets) und JavaScript (JS).

Mit HTML werden die Inhalte einer Seite strukturiert, wie Überschriften, Abschnitte und Absätze. Stellt man sich eine Webpage als Analogie zu einem Zimmer vor, dann bestimmt HTML das „Was“, also welche Möbel in dem Zimmer stehen sollen (Stuhl, Tisch, Sofa)².

CSS bestimmt das „Wie“ und wird benutzt um über die Darstellung der Elemente zu entscheiden, also deren Stil („style“) vorzugeben. CSS legt deren Aussehen, die Position und Sichtbarkeit fest und somit welche Schriften, Farben, Abstände usw. die Elemente haben sollen. Es sorgt für ein gemeinsames Layout aller Elemente. In dem Zimmer würde CSS bestimmen, welche Farbe zum Beispiel das Sofa hat und wie viel Abstand es zum Tisch haben soll.

JavaScript bestimmt das Verhalten der Elemente und ihre Reaktion. Es wird verwendet um Meldungen auszugeben, zu bestimmen was passieren soll wenn ein Button geklickt wird oder wenn die Maus über ein Element bewegt wird. CSS und HTML sind beschreibende Sprachen. JS hingegen ist eine Programmiersprache für dynamisches Verhalten. Mit JS wird die Frage nach dem „Wohin“ beantwortet. Wohin sich ein Element bewegen oder verändern soll. In dem Zimmer könnte das bedeuten, dass sich die Farbe des Sofas ändert sobald es jemand (mit der Maus) berührt.[30]

HTML-Dateien werden auf einem Server gespeichert und von dort in den Browser geladen. CSS- und JS-Dateien werden dann vom Browser angefordert sobald im HTML-Dokument ein Hinweis darauf erscheint. Seltener stehen CSS und JS auch direkt im HTML-Text.

4.3 Physische Komponenten der Internetkommunikation

4.3.1 Wireless LAN

Mit „Wireless LAN“ (WLAN) werden eigentlich alle drahtlosen und lokalen Datennetze bezeichnet, also auch zum Beispiel solche, die Bluetooth nutzen. Im allgemeinen Sprachgebrauch hat sich der Begriff „WLAN“ jedoch für Funknetzwerke mit dem Standard IEEE 802.11 etabliert. WLAN steht für „Wireless Local Area Network“ was übersetzt ins Deutsche „drahtloses lokales Netzwerk“ bedeutet. WLAN wird in der Regel verwendet, um Geräte in einem lokalen Netzwerk mit dem Internet zu verbinden. Meistens stellt ein WLAN Router die kabelgebundene Verbindung zum Internet und das lokale drahtlose Netzwerk bereit.[32]

Die Arbeitsgruppe IEEE begann 1990 eine standardisierte Technik für die drahtlose Kommunikation zu erarbeiten. Daraus entstand ein Protokoll und ein Übertragungsverfahren. Der Standard IEEE 802.11 basiert auf Ethernet und besteht seit dem Jahr 1997. Er ist die meist genutzte Technik für Funknetzwerke. Mit der Einführung des Standards ist eine verbindliche Luftschnittstelle für lokale Funknetzwerke entstanden. Dieser Umstand ermöglichte die Verbreitung solcher lokaler Funknetze. Die ursprüngliche Übertragungsrate von IEEE 802.11 war mit 1 bis 2 MBit/s im Vergleich zu heute noch sehr gering. Der Standard wurde laufend erweitert und nicht nur die Übertragungsrate wurde erhöht, sondern auch an Reichweite, Datensicherheit und Kompatibilität zwischen verschiedenen Herstellern wurde gearbeitet. Die Bezeichnungen der unterschiedlichen Standards sind an den kleinen Buchstaben „a“, „b“, „n“ etc. zu erkennen, die „IEEE 802.11“ angefügt sind. Das Übertragungsmedium von WLAN ist Luft anstelle von Kabeln wie beim LAN (local area network). WLAN nutzte zu Beginn eine Frequenz von 2,4 GHz. Später kamen Frequenzbänder von 5 GHz dazu. Die Nutzung von Frequenzbändern ist weltweit nicht einheitlich geregelt. In Europa wird für WLAN meistens 2,4 GHz genutzt. Für eine schnellere Übertragung, oft auch 5 GHz. Kaum verbreitet ist eine Frequenz von 60 GHz, bei der nur eine sehr geringe Reichweite erreicht wird und die Wände nicht durchdringen kann.

Mobile Geräte die Internet nutzen haben einen WLAN-Chip integriert. Die Chips sind über die Jahre kleiner geworden und effizienter im Energieverbrauch. Sie kommen daher mit weniger Akkuleistung (oder Strom aus anderen Quellen) aus und können deshalb heute in kleinsten Geräten verbaut werden. Das macht WLAN für IoT Anwendungen verfügbar.[33]

² Analogie orientiert an [31]

Oft ist auf Geräten und in Beschreibungen der Begriff „WiFi“ zu lesen. Dieser Markenbegriff wurde von der gleichnamigen Wi-Fi-Alliance vorgestellt. Diese verkörpert unter anderem eine Art Prüfstelle für WLAN-fähige Geräte. Es hatte sich herausgestellt, dass der IEEE 802.11 Standard nicht konsequent umgesetzt wurde. Die Wi-Fi-Alliance schloss sich daher im Jahr 1999 zunächst unter dem Namen „WECA“ (Wireless Ethernet Compatibility Alliance) zusammen. Ihr Ziel ist es, seitdem dafür zu sorgen, dass der Standard besser implementiert und weiter verbreitet wird. Sie prüft Geräte nach eigenen Richtlinien und vergibt ein „Wi-Fi“-Zertifikat. Die Überprüfung erfolgt gegen eine Gebühr. Trägt ein Produkt kein Wi-Fi Zertifikat ist dies kein Hinweis auf Inkompatibilität. Es kann trotzdem alle Standards erfüllen. Die Organisation umfasst derzeit circa 600 Mitglieder (meist Unternehmen). [35]

4.3.2 Router

Mit Routern werden Netzwerkpakete transportiert und zwischen Netzwerken weitergeleitet. Ein Internetrouter ist das Bindeglied zwischen lokalen Geräten, die mit ihm verbunden sind und dem Internet. Er empfängt Datenpakete und leitet diese weiter oder blockiert unerwünschte Daten. Er hat eine lokale IP-Adresse, die für alle Geräte im lokalen Netzwerk sichtbar ist. Router, die mit dem Internet verbunden sind, haben außerdem eine IP-Adresse, mit der sie im Internet erkennbar sind. Diese ist im lokalen Netzwerk aber nicht zu erkennen. Der Router kann an ihn angeschlossene Geräte voreinander verbergen und private IP-Adressen zuordnen, die nur für lokal angeschlossene Geräte relevant sind. Datenpakete leitet er so weiter als kämen sie von ihm. Der Router bestimmt also den Datenfluss. Auf diese Weise werden Geräte im lokalen Netzwerk durch den Router nach außen abgeschirmt. Er hat zwei Kommunikationsschnittstellen. Eine nach außen, in das übergeordnete Netzwerk, zum Beispiel das Internet. Diese ist die WAN-Schnittstelle. Die andere Schnittstelle stellt die Verbindung zum lokalen Netzwerk her, in dem die privaten IP-Adressen verwendet werden. Sie wird LAN-Schnittstelle genannt. In Abbildung 4.4 ist die Kommunikation schematisch dargestellt.

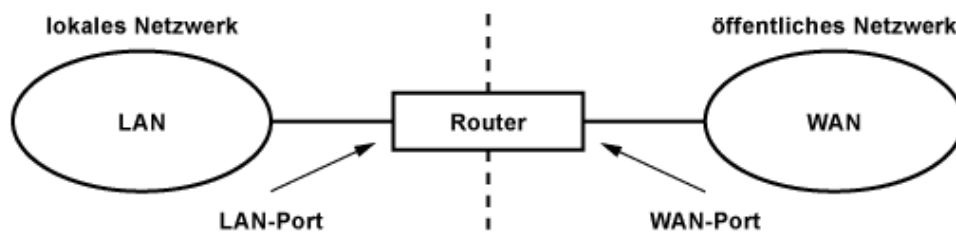


Abbildung 4.4: Kommunikationsschnittstellen eines Routers [33].

Anhand einer Routing-Tabelle entscheidet der Router über den Weg eines Datenpakets. Er ermittelt verfügbare Routen, wählt die am besten geeignete Route aus und stellt eine physikalische Verbindung zu anderen Netzwerken her. Schließlich passt er die Datenpakete an die Gegebenheiten an und fragmentiert sie entsprechend der verwendeten Übertragungstechnik.[26]

Sogenannte WLAN-Router haben einen Wireless Access Point integriert (siehe Abschnitt 4.3.5) um einen kabellosen Anschluss von Geräten mit WLAN-Adapter zu ermöglichen.

4.3.3 Modem

Das Modem wandelt digitale Signale in Spannungs-, Funk- oder Lichtsignale und umgekehrt. Daher hat es auch seinen Namen aus dem Englischen: MODulation - DEModulation. Es wandelt die analogen Signale, welche im Fall von DSL über die Telefonleitung übertragen werden, durch Modulation in digitale Signale um. So kann sie der Router verarbeiten. Andersrum wandelt das Modem die digitalen Signale des Routers in analoge Signale um, damit diese übertragen werden können

(Demodulation). In der Internetkommunikation übernehmen Modems die Aufgabe den Router mit dem physischen Netzwerk des Internets zu verbinden. Inzwischen haben viele WLAN-Router ein integriertes Modem.[33]

4.3.4 Gateway

Im Zusammenhang mit Netzwerken hat der Begriff Gateway unterschiedliche Bedeutungen und Definitionen. In dieser Arbeit wird mit Gateway ein Verbindungsmodul zum Internet bezeichnet. Ein Gateway stellt im Allgemeinen eine Verbindung zwischen zwei Netzwerken her. Diese Netzwerke können inkompatibel sein, also unterschiedliche Protokolle oder Übertragungsverfahren benutzen, die der Gateway dann aneinander anpasst, um die Kommunikation zu ermöglichen. Er stellt eine logische Verbindung her und übermittelt die Daten zwischen Quelle und Ziel. Dabei konvertiert ein Gateway wenn nötig eine oder mehrere der folgenden Komponenten: Physikalische Bedingungen, Übertragungsgeschwindigkeit, Protokolle, Adressierung, Datenformate.[33]

4.3.5 Wireless Access Point

Ein „Wireless Access Point“ (WAP) oft abgekürzt zu „Access Point“ (AP) stellt ein lokales drahtloses Netzwerk her. Im Deutschen bedeutet der Begriff „drahtloser Zugangspunkt“. Geräte können sich mit dem AP verbinden. Der AP ist wiederum mit einem Netzwerk verbunden, das meistens einen kabelgebundenen Zugang zum Internet hat. Ein AP kann aber auch ausschließlich ein lokales Netzwerk aufbauen, ohne eine Verbindung zu einem weiteren Netzwerk herzustellen. APs werden im WLAN genutzt und können für unterschiedliche Betriebsarten konfiguriert werden. Neben dem beschriebenen AP-Betrieb ist es außerdem möglich den AP als Client zu konfigurieren. Ein Gerät mit kabelgebundener Ethernetschnittstelle kann so drahtlos in einem WLAN kommunizieren. Konfiguriert man den AP als Repeater kann er die Reichweite eines WLANs erhöhen. Er funktioniert dann wie ein Verlängerungskabel in einem kabelgebundenen Netzwerk. Als Bridge konfiguriert, kann der AP zwei kabelgebundene Netzwerke über ein WLAN miteinander verbinden. [33]

Kapitel 5

Anforderungen und Konzeption

5.1 Lernziele für Schüler mit dem IoT-Lernmodul

Nach dem Durchführen der Lerneinheit sollen die Schüler verstehen, was der Begriff „Internet der Dinge“ bedeutet und IoT-Anwendungen erkennen können und von ähnlichen Anwendungen, die ebenfalls im Zusammenhang mit dem Internet stehen, abgrenzen können. Die Funktionsweise der Kommunikation und die Rollenverteilung soll schematisch verstanden werden. Es soll klar sein welche Instanzen den Client oder Server repräsentieren und wer von beiden eine Anfrage sendet oder antwortet, so wie in der Abbildung 5.1 schematisch dargestellt. Die Notwendigkeit dieser Vorgehensweise der Kommunikation soll verstanden werden. Es ist dabei nicht relevant, wie Protokolle funktionieren. Die Schüler lernen die wichtigsten Begriffe der Internetkommunikation und deren Bedeutung: Client, Server, IP und eventuell einige andere, die für das Lösen bestimmter Aufgaben notwendig sind. Im Zusammenhang mit dem Kommunikationsablauf soll auch deutlich werden,

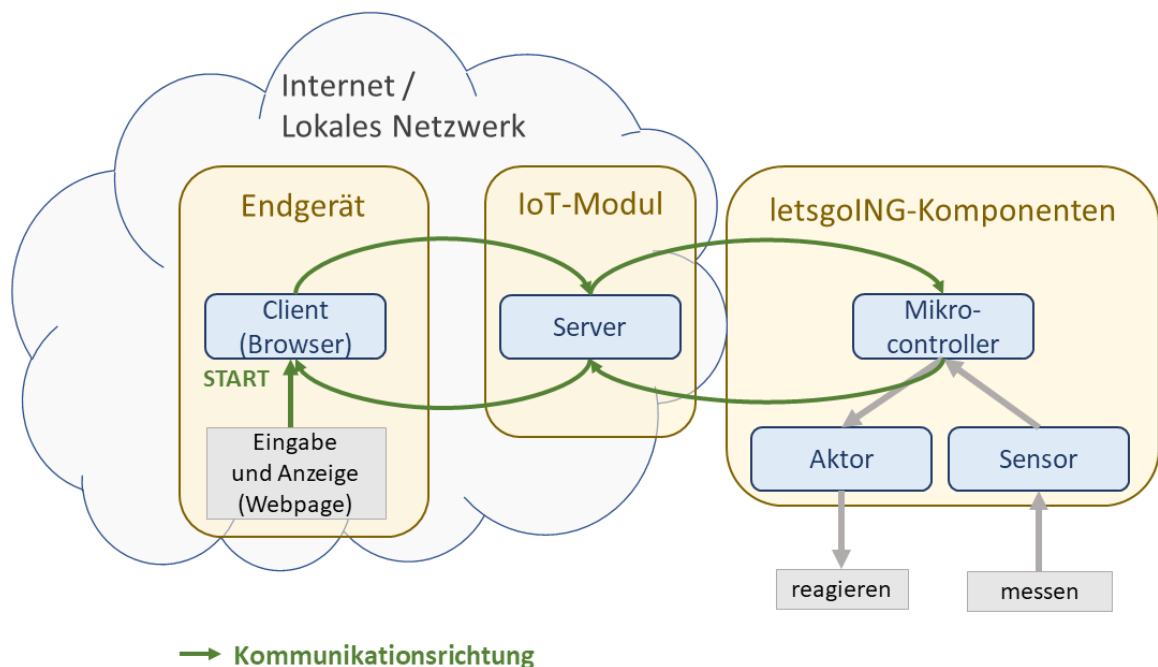


Abbildung 5.1: Kommunikation innerhalb eines Netzwerks.

dass es durch die weite Verbreitung des Internet Protocol (IP) und des WLAN, bei Wahl geeigneter Softwarekomponenten, einfach möglich ist unabhängig von der verwendeten Plattform (Betriebssys-

tem und Endgerät) zu kommunizieren. Dies soll, wie im Abschnitt 5.5 erörtert wird, anhand einer Anwendung im Browser geschehen. Denn Browser sind für unterschiedliche Betriebssysteme und Endgeräte verfügbar und außerdem fähig, denselben Inhalt anzuzeigen. So kann die Anwendung im Rahmen des Lehrmoduls im Schulunterricht auf Laptops, Smartphones und sofern vorhanden, Tablets ausgeführt werden.

Die Sensibilisierung für das Thema Datenschutz soll ebenfalls eine Rolle spielen, da es im Unterricht noch zu wenig Ansprache findet und sich im Internet auf andere Weise als in der analogen Welt manifestiert. Der Umgang im Internet mit personenbezogenen und von Sensoren aufgezeichneten Daten soll thematisiert werden. Dabei sollen die Fragen, „Wo landen meine Daten im Internet?“, „Wer hat Zugriff darauf?“, „Ist es möglich die Daten wieder zu löschen?“ gestellt und für bestimmte Fallbeispiele beantwortet werden. Aufgaben zu diesem Thema, können in einer zukünftigen Arbeit entworfen werden.

Wie im Abschnitt 2.2 schon angekündigt, ist ein wichtiges Ziel des Lernmoduls, der kritische Umgang mit Anwendungen in IoT und das Hinterfragen der Sinnhaftigkeit von neuen Entwicklungen in diesem Sektor. Die Schüler sollen nicht den Eindruck bekommen, dass das „Internet der Dinge“ per se nur nützliche und hilfreiche Anwendungen bietet, sondern, dass im Einzelfall ein Unterscheiden und Abwägen notwendig ist. Dies auch im Hinblick auf das zuvor genannte Thema Datenschutz. Es gibt momentan schon zahlreiche Anwendungen auf dem Markt und die Tendenz ist steigend. Schüler sollen diesem Umstand in der Gegenwart und Zukunft gewachsen sein und fähig kritisch zu entscheiden, welche Produkte sie tatsächlich nutzen wollen. Ein Aspekt ist hierbei zu hinterfragen, welchen tatsächlichen Mehrwert ein IoT-Produkt im Alltag bietet, ob sich eine Investition lohnt oder ob etwaige Gefahren von einer Anwendung ausgehen. Clemens Gleich beschreibt die Flut an Produkten sehr bildlich in seinem Kommentar „The Internet of shitty things“ und regt damit zum Nachdenken an, wenn etwa die Sicherheit betroffen ist, weil ein Heizungsthermostat über das Internet gebootet werden kann oder die tatsächliche Zeit- oder Arbeitersparnis, durch kommunizierende Haushaltsgeräte nicht erkennbar zu sein scheint [36]. Dieses Lernziel könnte zunächst theoretisch erörtert werden, innerhalb einer Diskussionsrunde, auf die sich jeder Schüler mittels Recherche vorbereitet. Der Sicherheitsaspekt könnte an Beispielen eines simulierten Hackerangriffs mit Hilfe von Netzen aus IoT-Produkten theoretisch thematisiert werden. Für die Zukunft wäre auch denkbar ein Netz aus WLAN-Modulen im Klassenzimmer miteinander kommunizieren zu lassen, um die Vernetzung der Dinge untereinander praktisch erfahrbar zu machen.

Die angestrebten Lernziele sind also:

- die Bedeutung des Begriffs „Internet der Dinge“ verstehen
- den Kommunikationsablauf im Internet und damit zusammenhängende Begriffe kennen lernen
- die Bedeutung der Plattformunabhängigkeit im Internet der Dinge verstehen
- die Sensibilisierung für Datenschutz
- die Sinnhaftigkeit von Anwendungen im Internet der Dinge diskutieren und abwägen.

5.2 Notwendige Infrastruktur an Schulen

Um das Lernmodul durchführen zu können muss an den Schulen eine passende Infrastruktur vorhanden sein. Endgeräte zur Bedienung und Ausgabe der Anwendungen des IoT-Moduls sind in erster Linie Laptops. Diese sollten Schülern in Zweiergruppen zur Verfügung stehen. Des weiteren sind zur Veranschaulichung der Plattformunabhängigkeit Smartphones oder Tablets notwendig. Sofern durch die Lehrkraft autorisiert, können die Schüler ihre eigenen Smartphones benutzen. Um Gleichberechtigung zu gewährleisten, sollten jedoch weitere Smartphones oder Tablets bereitgestellt

werden, um auch Schüler zu versorgen die ein solches Gerät nicht besitzen.

Die Schule sollte ein kabelloses Netzwerk bereitstellen. Für in Zukunft geplante IoT-Lerneinheiten ist außerdem ein Internetzugang notwendig. Im besten Fall liegt ein WLAN-Netzwerk vor, dessen Zugangsdaten für die Schüler offen gelegt werden können. Ist dies nicht möglich, sollte eine kabelgebundene LAN-Verbindung vorhanden sein, dann kann ein WLAN Router von letsgoING bereitgestellt werden, über den die Module eine Verbindung mit dem Internet herstellen. Sollte auch keine LAN-Verbindung mit Internetzugang zur Verfügung stehen, könnte die Möglichkeit erörtert werden, dass letsgoING Router mit mobilem Internetzugang und ausreichend Datenvolumen unterhält, die in den Schulen eingesetzt werden. Dies hängt natürlich zunächst von der Netzabdeckung und der Empfangsqualität in den Schulräumen ab. Die Überlegungen erübrigen sich jedoch zunächst für die in dieser Arbeit entwickelte Lerneinheit. Es ist noch keine Internetverbindung erforderlich, da das IoT-Modul ein eigenes lokales Netzwerk aufbaut, in welches sich Endgeräte einloggen können.

Somit wird für das nachfolgend in Abschnitt 5.4.3 beschriebene und in dieser Arbeit umgesetzte IoT-Modul, lediglich ein digitales Endgerät (Desktop PC, Laptop, Tablet oder Smartphone) mit WLAN-Schnittstelle und einer beliebigen Browsersoftware benötigt.

Die Marke und das Betriebssystem des Endgeräts sowie der Typ des Browsers spielen keine Rolle. Für weitere hier beschriebene Umsetzungskonzepte von Lerneinheiten und Kommunikationsmodellen mit dem IoT-Modul wird zusätzlich noch ein Internetzugang über einen WLAN Router benötigt.

5.3 Anforderungen an die Hardware des IoT-Moduls

Ein zentrales Hardwareelement von letsgoING ist der Arduino Uno Mikrocontroller. An diesen sind die letsgoING-Hardware-Module mit ihren Sensoren oder Aktoren angeschlossen.

Das IoT-Modul sollte zu diesen Komponenten kompatibel sein. Um „Internet der Dinge“ greifbar zu machen, ist eine Standardkommunikationstechnik notwendig, die weit verbreitet ist. Hierbei bietet sich IP an. Es ist Standard der meisten Netzwerke und der Router, die für Internetkommunikation genutzt werden. Die Kommunikation soll kabellos ablaufen, sodass es naheliegt ein WLAN-Modul als IoT-Modul für das Projekt einzusetzen. WLAN verfügt außerdem über eine größere Reichweite und Übertragungsgeschwindigkeit als Bluetooth[33]. Im Hinblick auf die Anwendung im Schulunterricht muss das IoT-Modul einfach zu bedienen sein, weshalb einige Aspekte in der Handhabung beachtet werden sollten. Das IoT-Modul sollte robust in der Anwendung sein und nicht anfällig für Systemabstürze. Es sollte bei Anschluss der Stromversorgung selbständig das Programm mit allen Funktionen für das IoT-Lernmodul starten, also „Plug and Play“ unterstützen. Die beiden Betriebsarten als Access Point, also der Aufbau eines eigenen lokalen Netzwerks sowie als Gateway, also als „Brücke“ ins Internet oder zu einem anderen Netzwerk, sollen unterstützt werden. Die Schnittstellen zwischen IoT-Modul und Peripherie sollen einfache Ein- und Ausgabe unterstützen. Flexibilität im Einsatz ist wichtig. Denn die in letsgoING bereits vorhandenen Hardware- und Softwarekomponenten müssen sowohl zusammen mit dem IoT-Modul, wie auch weiterhin unabhängig, ohne dass das IoT-Modul angeschlossen ist, verwendet werden können. Der Aufwand für den Entwurf und die Umsetzung des IoT-Moduls muss überschaubar bleiben, genauso müssen sich die anfallenden Kosten im Rahmen halten. Die Anwendung darf nicht zu komplex sein denn alle letsgoING-Module sollen an den Schulen auch durch Personen (Schüler und Lehrer), die wenig Wissen und Erfahrung in der Anwendung von Mikrocontrollern mitbringen, problemlos verwendet werden können. Fehler müssen leicht gefunden und behoben werden können. Möglichst sollte kein, im Vergleich zu bestehenden Komponenten, komplett neues und komplexes System eingeführt werden.

Ein Mikrocontroller mit umfangreichem Betriebssystem und grafischer Benutzeroberfläche, wie beispielsweise ein Raspberry Pi, der viele Komponenten für zahlreiche Funktionen integriert, ist daher nicht für ein IoT-Modul in letsgoING geeignet. Der Raspberry Pi wird mit einer Linux-Version oder einer Version von Microsoft 10 betrieben, die eine reduzierte oder keine grafische Benutzeroberfläche bietet. Die Einarbeitung der Lehrer und Schüler in das System würde zu viel Aufwand und Zeit

für die Einarbeitung in Anspruch nehmen. Denn der Raspberry Pi bietet nicht die Vorzüge von „Plug and Play“ und es muss, möglicherweise manuell, jedes mal beim Einschalten das entsprechende Programm manuell gestartet werden. Treten Fehler auf, können diese in einem umfangreichen Betriebssystem nicht so leicht identifiziert werden wie in einem einfachen System, das nur wenige Funktionen bereitstellt und das es zulässt einen Debugger zu verwenden um Fehler zu finden. Software müsste neu entworfen und angepasst werden und das Zusammenspiel mit dem Arduino wäre mit größerem Aufwand zu realisieren als mit einem einfachen Mikrocontroller.

Ein Mikrocontroller, der diese Ansprüche erfüllt, ist der ESP32 von espressif. Es ist das Nachfolgemodell des ESP8266. Die auf dem Betriebssystem FreeRtos basierende Systemsoftware befindet sich derzeit noch in der Entwicklung. Alle Funktionen die im Rahmen dieser Arbeit zunächst für das letsGoING-IoT-Modul notwendig sind, stehen zur Verfügung und laufen stabil. Zur vorläufigen Entwicklung des IoT-Moduls wird in dieser Arbeit ein ESP32 Thing Breakoutboard von SparkFun verwendet, welches im Kapitel 6 beschrieben ist. Dieses Wird als WLAN-Modul konfiguriert und eingesetzt.

5.4 Hardwarekonzepte für das IoT-Modul

Mehrere Konzepte für die technische Umsetzung eines IoT-Moduls sind möglich. Drei Verschiedene sollen im Folgenden dargestellt und untersucht werden. Ein Konzept soll ausgedacht und in dieser Arbeit umgesetzt werden. Auswahlkriterien sind vor allem die Funktionalität, welche das Konzept bietet und die Größe des Arbeitsaufwands bei der Entwicklung und Umsetzung. Die Flexibilität in der späteren Anwendung soll möglichst groß sein. Das bedeutet, es muss möglich sein, das bestehende System ohne Änderung, wie bisher, ohne das IoT-Modul zu verwenden. Das IoT-Modul soll aber auch in Verbindung mit bisher bestehender Hardware einfach verwendet und verbunden werden können. Die Anwendung soll für die Schüler und Lehrer nicht zu komplex sein, sodass mit einer einfachen Anleitung ohne ein umfangreiches technisches Hintergrundwissen gearbeitet werden kann. Die Kosten sollen den geringst möglichen Betrag im einstelligen Eurobereich, bei gleichzeitiger Erfüllung zuvor genannter Kriterien, nicht übersteigen.

5.4.1 IoT-Modul als Schnittstellenkonverter auf WLAN

Das erste Konzept welches hier vorgestellt werden soll sieht vor, das IoT-Modul als Schnittstellenkonverter auf WLAN (WLAN-Shield¹) einzusetzen. Dabei wird die Verbindung zwischen Arduino und ESP32 (als „WLAN-Shield“) über die Protokolle UART und SPI hergestellt. Das bedeutet, dass der Arduino direkt über das IoT-Modul im Netzwerk kommunizieren kann und so, alle an den Arduino angeschlossenen Komponenten, Daten direkt senden können. In der Abbildung 5.2 sind diese Komponenten grau hinterlegt. Dies hat zur Folge, dass Änderungen an Teilen des bestehenden Codes vorgenommen werden müssen. Für die Verwendung des WLAN-Moduls muss eine Bibliothek eingebunden werden und die Kommunikation über die serielle Schnittstelle muss angepasst werden, denn der Arduino verfügt über nur eine solche. Es ist zwar möglich eine zweite virtuelle serielle Schnittstelle einzubinden, zum Beispiel für die Ausgabe am seriellen Monitor. Die Kommunikation über diese „Softserial“-Schnittstelle ist aber langsamer und unzuverlässiger. Das Lehrmaterial müsste teilweise angepasst werden, es müssten korrigierte Beispielprogramme erstellt und grafisch in die Lehrunterlagen eingefügt werden. An manchen Stellen müsste die Beschreibung und Anleitung angepasst werden. Das Auffinden und korrigieren der betroffenen Abschnitte in den Lernmodulen würde Zeit in Anspruch nehmen und auch das Testen des korrigierten Programmcodes muss berücksichtigt werden. Die Umsetzung dieses Konzeptes wäre also softwareseitig und redaktionell mit relativ großem Aufwand verbunden. Zudem wäre es, nach Änderung des Codes, nicht mehr möglich das Board ohne WLAN-Modul, mit denselben Beispielprogrammen und Programmieraufgaben zu nutzen.

¹ähnliche Hardware wird oft als „WLAN-Shield“ bezeichnet

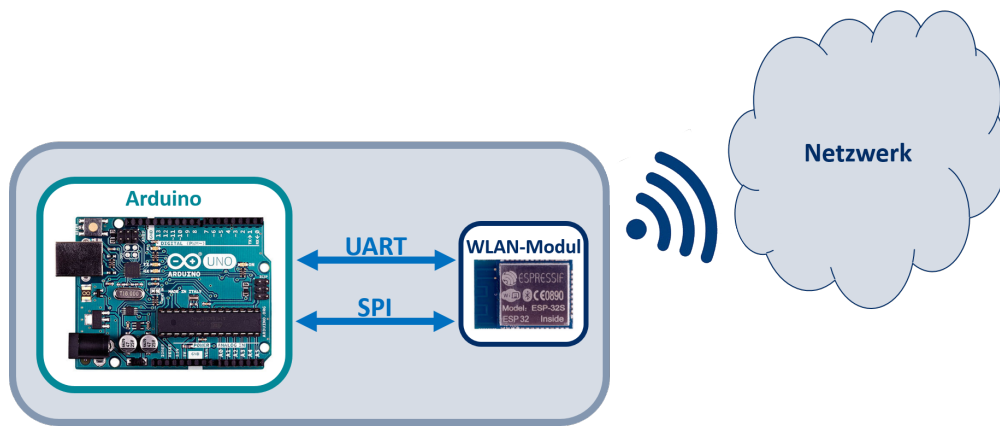


Abbildung 5.2: Das WLAN-Modul als WLAN-Shield, schematische Darstellung der Kommunikation.

5.4.2 IoT-Modul als eigenständiger, programmierbarer Mikrocontroller

Eine andere Möglichkeit für ein IoT-Modul ist, den ESP32 nicht mit dem Arduino zu verbinden, sondern als eigenständigen Mikrocontroller („Stand-Alone-Mikrocontroller“) zu verwenden. Ein großer Vorteil des ESP32 ist, dass er durch die zahlreichen digitalen und analogen I/O-Pins wie UART, SPI, I²C und einige PWM-Pins, und seine Leistung, den Arduino ersetzen kann. Die letsgoING-Module, also die Breakoutboards der Sensoren und Aktoren (LGI-Module) sollen dann über digitale, analoge oder I²C-Schnittstellen direkt mit dem WLAN-Modul verbunden werden. Somit können ebenfalls alle Komponenten über das IoT-Modul direkt im Netzwerk kommunizieren, wie es in der Abbildung 5.3 schematisch dargestellt ist. Bei diesem Konzept müssen einige Teile des Programmcodes angepasst werden um auf den ESP32 übertragbar zu sein. Auch einige Programmblöcke in Ardublock müssen angepasst werden. Alle schon bestehenden Softwarekomponenten könnten weiterhin uneingeschränkt wie bisher in Verbindung mit dem Arduino verwendet werden, sofern sie unverändert bleiben. Dafür ist also eine zweite komplette Version Software (Programme, Beispiele, Programmblöcke) notwendig, die aus der bisher vorhandenen entstehen kann. Somit gäbe es zwei Versionen, die jeweils entweder mit dem Arduino oder mit dem ESP32 funktionieren. Es wäre also ein neues, abgeschlossenes System, das so entstehen würde. Ein neuer Aufbau, auf dem der Mikrocontroller fest platziert ist und die LGI-Module aufgesteckt werden können, müsste entworfen und hergestellt werden. Das ist mit Kosten und Zeitaufwand verbunden, sowie auch das Entwerfen und Testen der Software und die Umsetzung zweier Systeme. Da das letsgoING-Fahrzeug derzeit mit einem Arduino Mikrocontroller betrieben ist, könnte es zunächst nicht im Rahmen „Internet der Dinge“ eingesetzt werden. Damit das möglich ist, müssen Überlegungen angestellt werden, wie die Verwendung des Fahrzeugs mit dem IoT-Hardwaremodul abgestimmt werden kann. Es muss ein neues Konzept ausgearbeitet und entwickelt werden.

Für einen Pilotversuch eignet sich dieses Konzept zunächst sehr gut. Ein erster Entwurf der Software für den ESP32 kann so unabhängig und mit wenig Aufwand getestet werden. Ob das IoT-Modul nach diesem Konzept in Schulen einzusetzen ist, ob es sich realisieren lässt zwei unterschiedliche Softwarepakete zu unterhalten oder ob das nicht zu Verwirrung und Frustration in der Anwendung an den Schulen führt, gilt es im Laufe der Arbeit zu erörtern. Möglicherweise ist das im Folgenden vorgestellte Konzept für den Schuleinsatz besser geeignet.

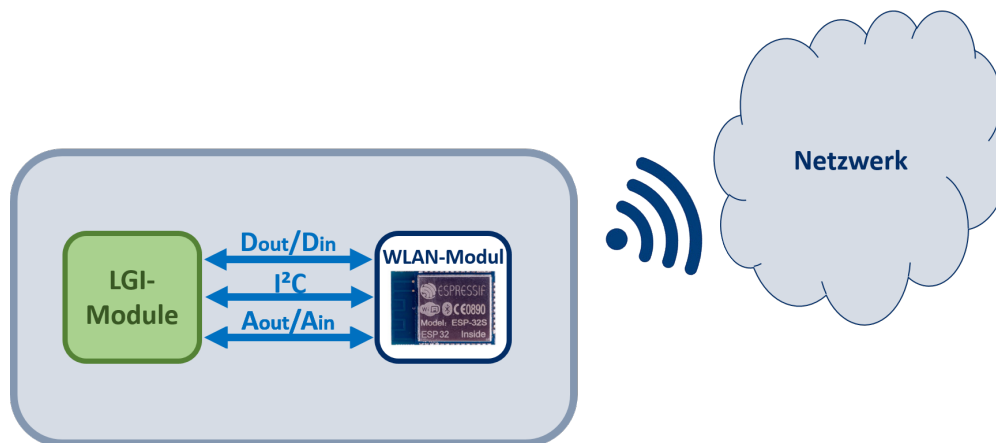


Abbildung 5.3: Das WLAN-Modul als alleinstehender Mikrocontroller. Schematische Darstellung der Kommunikation.

5.4.3 IoT-Modul als Input-Output-Modul

Bei diesem dritten Konzept kommt der Arduino zum Einsatz. Das IoT-Modul soll mit dem Arduino jedoch nicht, wie im Konzept aus dem Abschnitt 5.4.1 „IoT-Modul als Schnittstellenkonverter auf WLAN“ beschrieben, über UART und SPI verbunden sein. Zwischen IoT-Modul und Arduino sollen wie in Abbildung 5.4 schematisch dargestellt, nur Zustände und Spannungspegel an jeweilige Pins übertragen werden. Das IoT-Modul hat nur digitale (binäre) und analoge Ein- und Ausgänge. Der „analoge“ Ausgang ist wie beim Arduino Uno technisch als PWM-Ausgang realisiert. Es würden also die LGI-Module nicht direkt über das IoT-Modul im Netzwerk kommunizieren, wie bei den beiden zuvor vorgestellten Konzepten „als Schnittstellenkonverter auf WLAN“ (Abschnitt 5.2) und „als eigenständiger Mikrocontroller“ (Abschnitt 5.3). Für die Endanwendung ist der Umstand nicht relevant, denn diese kann so angepasst werden, dass der Kommunikationsweg zwischen LGI-Modulen für die Umsetzung der Lernziele keine Rolle spielt. Die Anpassung der Software findet in der Anbindung statt, also auf Seite des ESP32. Die bisher für den Arduino bestehende Software bleibt unberührt und kann genauso wie die Dokumentation weiter verwendet werden, ohne das didaktische Begleitmaterial umschreiben zu müssen. Flexibilität ist gegeben, denn alle bestehenden Komponenten sind wie bisher, auch ohne das IoT-Modul einsetzbar. Nur die Firmware für das IoT-Modul wird neu entwickelt und getestet. Der Aufwand in der Umsetzung ist so gering wie möglich. Es ist denkbar, das Design für das Breakoutboard des IoT-Moduls so zu gestalten, dass es sich neben den LGI-Modulen auf das Fahrzeug aufstecken lässt. Somit können in Zukunft auch andere Ideen für Lerneinheiten umgesetzt werden, wie zum Beispiel eine Motorsteuerung über WLAN oder Anwendungen die Bluetooth nutzen, welches der ESP32 wie in Abschnitt 5.3 beschrieben unterstützt. Das Konzept wäre somit günstig, da kein neuer Aufbau notwendig ist, nachhaltig, da es Möglichkeiten für andere Anwendungen bietet, der Aufwand für die Realisierung ist überschaubar, da bestehende Software, Hardware und Dokumentation weiter verwendet werden kann. Die Akzeptanzschwelle und der Aufwand für die Einarbeitung bleibt für die Lehrkräfte niedrig.

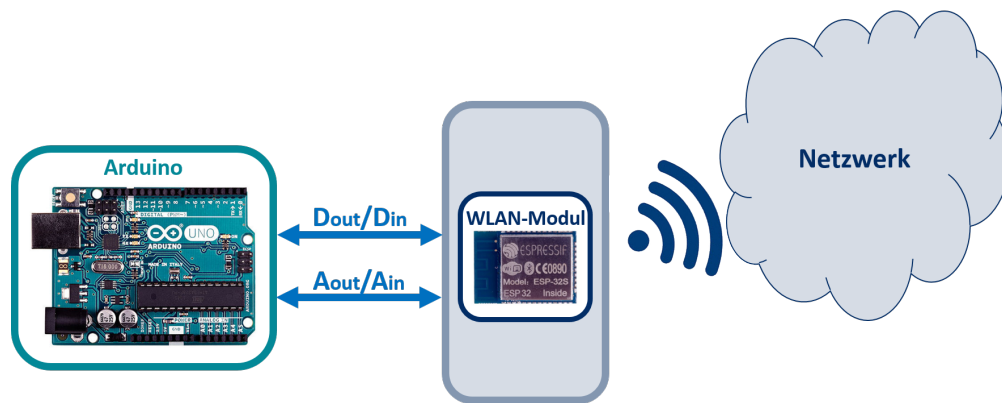


Abbildung 5.4: Das WLAN-Modul als IO-Modul. Schematische Darstellung der Kommunikation.

5.4.4 Auswahl des am besten geeigneten Hardware-Konzepts

Um die Software für das WLAN-Modul zu entwickeln, eignet sich am besten der ESP32 der auf einem Breakoutboard erhältlich ist. Auf diesem sind Pins bereits ausgeführt um die LIGI-Module anzuschließen. Außerdem sind wichtige Buttons und geeignete Anschlüsse für die Stromversorgung verbaut. Eine solche Entwicklungsplattform kann fertig gelötet, mit USB-Schnittstelle gekauft werden und ist sofort einsatzbereit[39]. Es kann über USB mit dem PC verbunden werden, um Programme hochzuladen und zu testen. Die Abbildung 5.5 stellt die beschriebenen Hardwarekonzepte zusammenfassend dar.

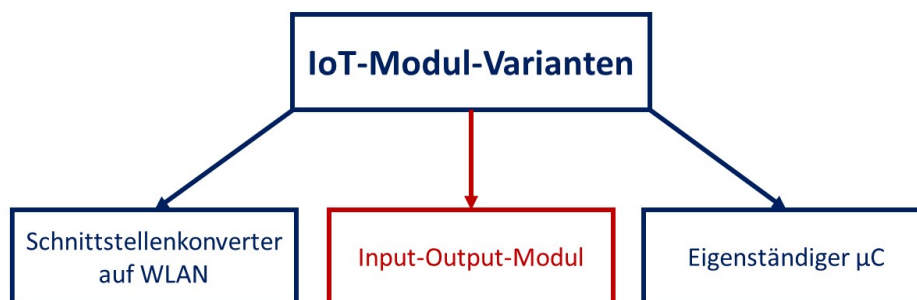


Abbildung 5.5: Diagramm: Entscheidung für das geeignetste Hardware-Konzept.

Die vorgestellten Konzepte und Kriterien zur Auswahl in der Tabelle in der Abbildung 5.6 zusammengefasst dargestellt und bewertet. Der Erstentwurf des IoT-Moduls wird mit einer ESP32 Thing Entwicklungsplattform, wie in Abbildung 6.2 zu sehen, ausgearbeitet. Damit wird zunächst das Konzept verfolgt, das WLAN-Modul als alleinstehenden Mikrocontroller zu verwenden, jedoch nur für erste Tests und die Entwicklung der Software. Denn das Ziel soll sein, das IoT-Modul nach dem letztgenannten Konzept, als Input-Output-Modul (Abschnitt 5.4) im Projekt einzusetzen und somit auch die Firmware entsprechend zu entwerfen. Diese Wahl wird getroffen, da dieses Konzept die festgelegten Kriterien für die Funktionalität, den Aufwand der Umsetzung, die Flexibilität im Einsatz (mit oder ohne IoT-Modul), geringe Komplexität in der Anwendung für Schüler und Lehrer sowie geringe Kosten, am besten erfüllt. Die Punkte sind im Abschnitt 5.4.3 jeweils erklärt. Das IoT-Modul als Schnittstellenkonverter auf WLAN (Abschnitt 5.4.1) würde vor allem nicht genügend Flexibilität bieten, das bestehende System aus Hard- und Softwarekomponenten wie bisher ohne WLAN zu verwenden. Das IoT-Modul als eigenständiger Mikrocontroller wäre zu komplex in der Anwendung und beide Konzepte verlangen einen hohen Aufwand für die Umsetzung und die Einbindung in das Projektumfeld.

Hardwarekonzept	Komplexität für Lehrende	Umsetzung des Konzepts	Flexibilität: LGI-Komponenten mit / ohne IoT	Kosten	Lokales Netzwerk oder Internet benötigt
Schnittstellenkonverter auf WLAN (5.4.1)	Sehr komplex	Großer Aufwand	Nur mit IoT, sonst sehr umständlich	gering	Mindestens lokales Netzwerk
Alleinstehender Mikrocontroller (5.4.2)	Sehr Komplex, da ein komplett neues System	Großer Aufwand	Nicht alle Komponenten mit IoT einfach verwendbar	Neuer Aufbau ist mit größeren Kosten verbunden	Nein
Input-Output-Modul (5.4.3)	Gering, bekanntes System	Überschaubarer Aufwand	Komplette Flexibilität	gering	Nein

Abbildung 5.6: Tabelle: Bewertung der vorgestellten Hardwarekonzepte.

5.5 Input-Output-IoT-Modul: Software- und Kommunikationskonzepte

Die im Abschnitt 5.1 erläuterten Lernziele sollen die Schüler durch praktisches Erfahren und Selbstaktiv-Sein bei der Umsetzung von Aufgaben, erreichen. Um Aufgaben stellen und bearbeiten zu können, ist eine Umgebung in Form einer Anwendung zu schaffen. Auf dieser Grundlage können dann Aufgaben gestellt und von den Schülern bearbeitet werden. Die Anwendung sollte nicht zu komplex sein, um die Schüler nicht zu überfordern und den Fokus auf das Bearbeiten von Aufgaben zu lenken und nicht auf das in Betrieb nehmen der Anwendung. Die Anwendung sollte am besten so gestaltet sein, dass Parallelen zu bekannten Anwendungen aus dem Alltag zu erkennen sind. Bestehendes Wissen sowie im Projekt bereits erlangte Erfahrung soll helfen die Aufgaben zu bearbeiten. Eine anwenderfreundliche Bedienoberfläche ist dabei wichtig, damit sich die Schüler schnell zurechtfinden. Bei den Überlegungen spielt auch die Autonomie des gesamten eingesetzten Systems eine wichtige Rolle. Eine Abhängigkeit von Drittanbietern, die Softwarepakete, ergänzende Anwendungen und Speicherplatz auf Onlineplattformen anbieten ist zu vermeiden, da die Anwendung dann nur mit einer Verbindung ins Internet genutzt werden kann. Autonomie durch das Aufbauen eines lokalen Netzwerks ist die bevorzugte Möglichkeit für die Anwendung, denn an den Schulen ist die Nutzung des WLAN für Schüler in der Regel nicht freigegeben und nicht immer ist ein LAN-Anschluss erreichbar. Damit das IoT-Lehrmodul in jedem Fall reibungslos eingesetzt werden kann, sollte es am besten als Access Point betrieben werden damit die Anwendung somit ohne Internetverbindung über ein selbst bereitgestelltes Netzwerk laufen kann. Möglich ist der Einsatz von Drittanbietersoftware (z.B. Clouddienste) dann, wenn diese offline in ausreichendem Umfang lauffähig ist. Drei unterschiedliche Ideen und Ansätze für Anwendungen werden in diesem Kapitel vorgestellt. Dabei kommuniziert der Arduino jeweils über das IoT-Modul „als Input-Output-Modul“ mit der Anwendung, wie im Abschnitt 5.4.3 vorgestellt.

5.5.1 Kommunikation mit dem IoT-Modul über eine App auf dem Smartphone oder Tablet

Der Arduino mit angeschlossenem IoT-Modul kommuniziert bei dieser Anwendung über eine Applikation (App) auf einem Smartphone oder Tablet. Das mobile Endgerät kann sich in dem vom IoT-Modul aufgebauten lokalen Netzwerk registrieren und so über das angebundene IoT-Modul eine kabellose Verbindung zum Arduino herstellen. Die Abbildung 5.7 zeigt die Kommunikationswege in schematischer Darstellung.

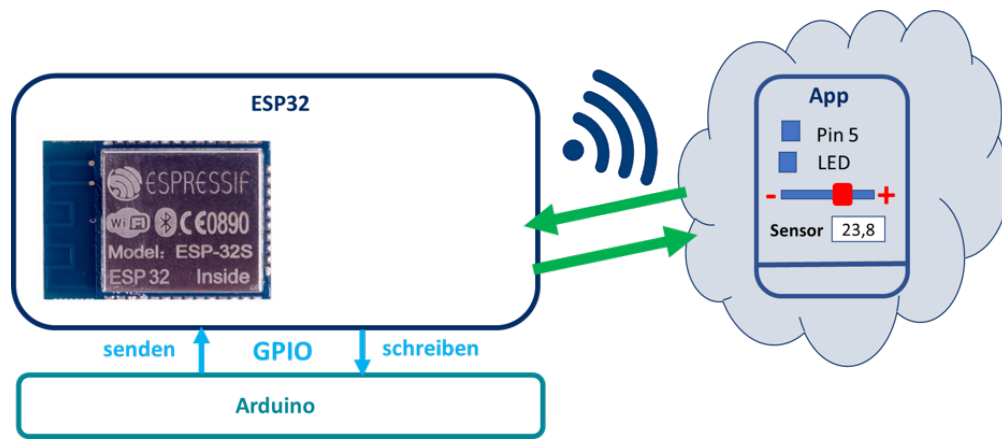


Abbildung 5.7: Anwendung mit Mobile App. Schematische Darstellung der Kommunikation.

Die App soll Sensorwerte anzeigen, auf Benutzereingaben reagieren, Befehle an Pins geben und den Status von Pins und angeschlossener Hardware anzeigen können. Dies soll innerhalb einer einfach und übersichtlich gestalteten grafischen Benutzeroberfläche geschehen, die auch mindestens einen Schieberegler für ein Potentiometer oder für PWM beinhaltet.

Eine App kann für Android-Betriebssysteme mit dem kostenlosen Programm App Inventor des MIT in wenigen Schritten erstellt werden. Mit dem Editor hat man die Möglichkeit in Blöcken statt textbasiert zu programmieren, was mit überschaubarem Aufwand zu bewerkstelligen ist. Mit Apple Geräten, die das Betriebssystem iOS nutzen, ist der Editor aber nicht kompatibel. Des Weiteren lassen sich Apps bei Android lokal installieren, für Apple Geräte ist das nicht vorgesehen und so gibt es nur umständliche Wege für eine Installation. Der für die Umsetzung beider Apps notwendige Aufwand, ist im Rahmen des Projekts zu hoch. Schüler die kein Smartphone mit Android-Betriebssystem zur Verfügung haben, sind somit von den Aufgaben ausgeschlossen und es kommt zu ungleicher Behandlung, dies gilt es zu vermeiden.

Plattformunabhängigkeit im Internet der Dinge kann mit dieser Anwendung nicht demonstriert werden. Eine App kann nur mit dem Betriebssystem funktionieren für welches sie entwickelt wurde.

Der Ablauf der Netzwerkkommunikation und die Begriffe der zum Einsatz kommenden Komponenten lässt sich mit einer App anschaulich erklären. Die Geräte sind den Schülern aus dem Alltag bekannt und so können sie leicht parallelen zu bekannten Anwendungen erkennen. Da die Kommunikation in zwei Richtungen abläuft, vom Anwender ausgehend und zu ihm hin oder automatisiert zwischen den an den Arduino angeschlossenen Hardware-Modulen ist das Thema „Kommunikation im Internet“ gut abgedeckt.

Die Sinnhaftigkeit von Anwendungen die zahlreiche Apps im Internet der Dinge bereitstellen, kann mit den Schülern diskutiert werden und auf bekannte Apps und Anwendungen Bezug genommen werden. Als Beispiel können Fitnessarmbänder dienen die derzeit immer mehr Popularität auf dem Markt gewinnen. In diesem Zusammenhang bietet es sich an, das Thema Datenschutz zu diskutieren, wie im Abschnitt 5.1 bereits ausgeführt. Im folgenden sind die Lernziele welche mit dieser Anwendung umgesetzt werden können aufgezählt:

- Bedeutung des Begriffs „Internet der Dinge“ verstehen
- Kommunikationsablauf im Internet und damit zusammenhängende Begriffe kennen
- Sensibilisierung für Datenschutz
- Sinnhaftigkeit von Anwendungen im Internet der Dinge diskutieren.

Lernziel das nicht erreicht wird, ist die:

- Bedeutung der Plattformunabhängigkeit im Internet der Dinge.

Das IoT-Modul kann für diese Anwendung nach Bedarf konfiguriert werden, so, dass es als Access Point funktioniert und ein lokales Netzwerk aufbaut oder als Gateway zu einem bestehenden WLAN-Netzwerk mit oder ohne Verbindung zum Internet.

5.5.2 Kommunikation mit dem IoT-Modul über einen Cloudservice im Internet

Das IoT-Modul wird als Gateway ins Internet verwendet. Der Arduino sendet über das IoT-Modul Daten an einen Cloudservice im Internet. Aktuelle Sensorwerte werden in regelmäßigen Abständen oder bei Eintreten eines Ereignisses, an einen Cloudservice übertragen und statistisch sowie grafisch aufbereitet und dargestellt. Die Ergebnisse sind dann über einen Browser auf dem Endgerät abrufbar. Sensoren könnten an unterschiedlichen Orten angebracht werden, beispielsweise ein Zähler an der Klassenzimmertüre welcher erfasst, wie viele Personen sich zu bestimmten Uhrzeiten im Raum befinden. Kostenlos einsetzbar ist die Plattform ThingSpeak der Mathworks Inc. mit welcher Sensordaten in einem privaten Bereich gesammelt werden können [34]. Es stehen Möglichkeiten zur grafischen Aufbereitung zur Verfügung. Nach der Definition von „Internet der Dinge“ aus dem Abschnitt 2.2, als Kommunikation zwischen Geräten oder mit einem Benutzer, ist ein Cloudservice für sich noch keine Anwendung im Bereich Internet der Dinge. Die Plattform bietet die Möglichkeit mit ThingTweet Nachrichten per Twitter zu versenden. Mit Einbindung dieser Funktion entsteht eine Internet-der-Dinge-Anwendung. Der Zähler an der Türe könnte also dafür sorgen, dass am Ende jeder Pause, eine Nachricht mit der Anzahl der vor und nach der Pause anwesenden Personen, getwittert wird. ThingSpeak unterstützt TalkBack, womit an Hardware automatisiert Aktionen ausgelöst werden können, wenn festgelegte Sensorwerte vorliegen.

Bei der Einbindung eines Clouddienstes besteht eine Abhängigkeit von einem Fremdanbieter, in diesem Fall von Mathworks Inc.. Es müsste ein eigener Clouddienst entworfen werden der auf einem eigenen Server von letsgoING läuft um unabhängig zu sein. Damit die Notwendigkeit eines Internetanschlusses hinfällig wird, kann ThingSpeak offline verwendet werden wenn die Applikation lokal gespeichert ist und im lokalen Netzwerk zur Verfügung steht. Abhängigkeit besteht dann aber trotzdem, wenn ein Update notwendig ist. Das IoT-Modul ist dann als Access Point konfiguriert.

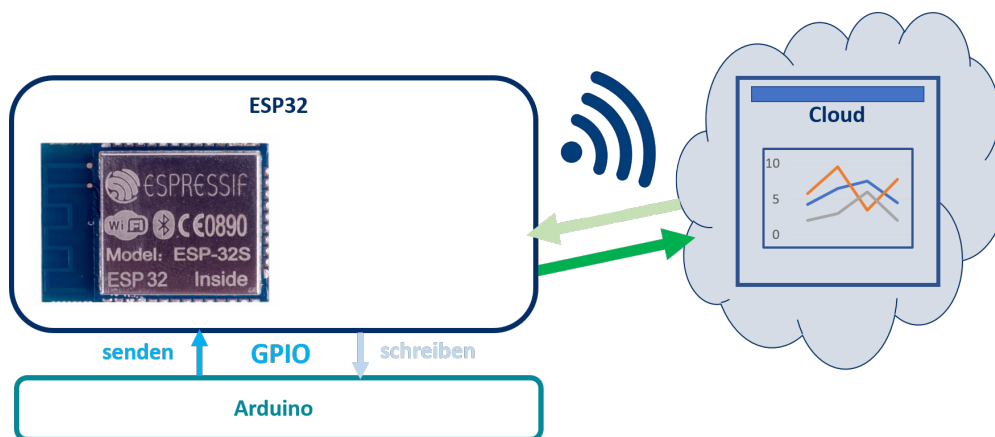


Abbildung 5.8: Anwendung mit Cloudservice. Schematische Darstellung der Kommunikation.

Eine Cloud-Anwendung bietet also nur unter bestimmten Voraussetzungen die Möglichkeit, den Schülern Wissen über die Kommunikationswege im Internet nahezubringen. Denn nur mit einer geeigneten Applikation findet Kommunikation in beide Richtungen, vom Arduino ausgehend und zu ihm zurück oder automatisiert über die Cloud, zwischen den an den Arduino angeschlossenen Hardware-Modulen, statt. Wird ThingSpeak mit den genannten zusätzlichen Komponenten Thingtweet oder TalkBack verwendet, kann die für IoT-Anwendungen typische Eigenschaft der

Selbständigkeit greifbar gemacht werden. Die Kommunikation ist in der Abbildung 5.8 dargestellt. Die hellblauen und hellgrünen Wege sind nur begehbar wenn eine der genannten zusätzlichen Applikationen verwendet wird. Eine Cloud-Anwendung eignet sich gut um das Thema Datenschutz zu diskutieren. Denn hier werden von Sensoren aufgezeichnete Daten auf Server-Rechnern gespeichert, deren Standort nicht offensichtlich ist. Auch was mit den Daten geschieht ist oft nicht eindeutig ersichtlich oder nachvollziehbar. Wie sinnvoll eine Anwendung ist, die einen Clouddienst nutzt und welche Beispiele es gibt, lässt sich thematisieren und mit den Schülern gemeinsam ergründen. Eine Fragestellung könnte lauten, welche Anwendungen bekannt sind, die eine Cloud nutzen und ob sich die Aufgabe der Anwendung auch ohne Cloudservice erfüllen lässt. Da die gesammelten und verarbeiteten Sensordaten mit einem Browser abzurufen sind, ist die Plattformunabhängigkeit gegeben und es kann gezeigt werden warum für die kommunizierenden Dinge, das Internet als Plattform gewählt wurde. Die umsetzbaren Lernziele sind

- Bedeutung des Begriffs „Internet der Dinge“ verstehen
- Sensibilisierung für Datenschutz
- Sinnhaftigkeit von Anwendungen im Internet der Dinge diskutieren.
- Bedeutung der Plattformunabhängigkeit im Internet der Dinge.

Lernziel das nur unter genannten Voraussetzungen erreicht wird, ist der:

- Kommunikationsablauf im Internet und damit zusammenhängende Begriffe

5.5.3 Kommunikation mit dem IoT-Modul über dessen lokalen Webserver

Der ESP32 verfügt über ausreichend Speicherkapazität für einen einfachen Webserver. Das WLAN-Modul wird als Access Point und als Gateway programmiert. Wenn kein WLAN gefunden wird, in das sich das Modul einwählen kann, wird ein lokales Netzwerk aufgebaut, in das sich andere Endgeräte (Laptop, Smartphones etc.) einwählen können. Diese Geräte kommunizieren mit dem IoT-Modul über eine Website, die auf dem Server (auf dem IoT-Modul) gespeichert ist. Das System funktioniert unabhängig von einer Internetverbindung. Die Kommunikationswege sind in der Abbildung 5.9 schematisch dargestellt und laufen in beide Richtungen ab, vom IoT-Modul zum Endgerät (Website im Browser) und umgekehrt. Auf Benutzereingaben kann also entsprechend reagiert werden, was für das Umsetzen des Lernziels vorteilhaft ist. Das IoT-Modul (genauer: der Webserver) soll, ähnlich der

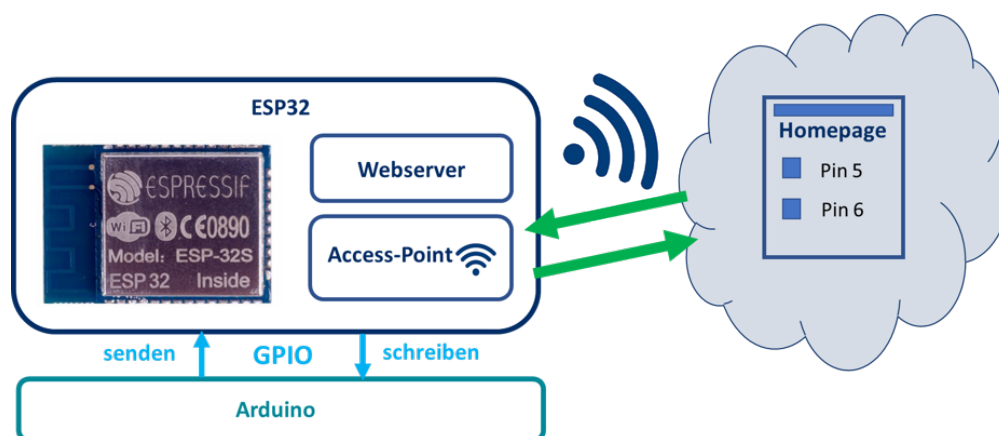


Abbildung 5.9: Website auf ESP32 als Webserver und Access Point. Schematische Darstellung der Kommunikation

App aus dem Abschnitt 5.5.1, Sensorwerte empfangen und im Browser anzeigen. Außerdem sollen Schaltflächen zum regulieren der PWM für eine RGB-LED und um eine LED an und auszuschalten, angezeigt werden. In der Abbildung 5.10 ist ein Entwurf für das geplante Layout der Startwebseite zu sehen.

The image shows a web interface for 'letsGOING' (Mechatronik Hochschule Reutlingen) titled 'Internet der Dinge'. It features several control elements:

- LED:** Two buttons labeled 'Ein' and 'Aus'.
- RGB-PWM:** Three input fields labeled 'Rot', 'Grün', and 'Blau', each containing '#WERT'. A 'senden' button is positioned to the right of the 'Blau' field.
- Analoger Schwellwert:** An input field containing '#WERT' and a 'senden' button to its right.
- Digitaler Eingang:** An input field containing 'Anzeige'.
- Sensor 1:** An input field containing 'Anzeige'.
- Sensor 2:** An input field containing 'Anzeige' and a 'anzeigen' button to its right.

Abbildung 5.10: Entwurf des Layouts für die Website im Modul Internet der Dinge.

Es sind verschiedene Anwendungen möglich, die mit der Website ausgeführt werden können. Über die Schaltflächen „Ein“ und „Aus“ kann eine LED (= digitaler Ausgang) gesteuert werden und gleichzeitig ganz einfach geprüft werden ob das IoT-Modul mit dem Netzwerk verbunden ist. Reagiert die LED, funktioniert das IoT-Modul und ist einsatzbereit. Beim Klicken des Button soll ein Text an die URL angehängt werden und in der Adresszeile erscheinen. Wird der Button „Ein“ gedrückt steht dort „LEDON“. Das ist für die Schüler sichtbare Kommunikation im Internet und kann didaktisch gut verwendet werden um den Kommunikationsablauf zu erklären. Beim Senden der PWM-Werte (= drei analoge Ausgänge) soll ebenfalls die Farbe und der zugehörige Wert in der Adresszeile sichtbar sein. Die Schüler haben dann die Möglichkeit auch direkt über die Adresszeile Werte einzugeben und so die LED zu steuern². Damit wird klar, dass die Kommunikation zwischen Endgerät und IoT-Modul, vom Endgerät (genauer: Browser bzw. Client) ausgelöst wird

Im Feld „analoger Schwellwert“ Kann ein Wert (= analoger Ausgang) eingegeben werden, der an das IoT-Modul gesendet wird. Dieser Wert dient dann z.B. als Schwellwert für eine sensorbasierte Steuerung innerhalb eines letsGOING Projekts. Sobald die Verbindung zwischen IoT-Modul und Arduino implementiert und funktionsfähig ist, kann der Wert in einer Aufgabe für die Schüler in Ardublock verwendet werden. Ein Helligkeitssensor beispielsweise, bekommt einen Schwellwert zugewiesen. Nach dessen Über- oder Unterschreitung soll automatisch eine LED angeschaltet werden.

Eine Ähnliche Anwendung könnte mit einem Lautstärkesensor realisiert werden. Die LED leuchtet grün wenn es ruhig ist, und rot wenn der Schwellwert überschritten wurde. Dabei können die Schüler ganz praktisch die Wege der Kommunikation im Internet, von einem Endgerät (hier der Browser auf dem Computer oder Smartphone) über die Logik auf der Recheneinheit zum Sensor und zum Aktor, erfahren.

Es können außerdem ein digitaler Zustand (= digitaler Eingang) und die Werte von zwei Sensoren (= zwei analoge Eingänge) zur Überwachung angezeigt werden. Hier findet Kommunikation

²Dieses Kommunikationsverfahren, das Anfügen von Parametern an eine http-URL entspricht dem Programmierparadigma „Representational State Transfer“ oder „REST“. Nach diesem Verfahren werden z.B. auch Leuchten in der Hausautomation (Smart Home, siehe Abschnitt 2.2) gesteuert.

nur scheinbar in umgekehrter Richtung, also zunächst vom IoT-Modul (bzw. Server) ausgelöst zum Endgerät (bzw. Browser als Client) statt. Denn eigentlich wird durch eine Anfrage des Endgeräts, einen Teil der Seite zu aktualisieren und neu zu laden, wiederum eine Anfrage an das IoT-Modul gesendet. Dieser Vorgang findet automatisiert statt und kann den Schülern anhand einer Aufgabe nahegebracht werden, wenn zum Beispiel erst nach klicken einer Schaltfläche der automatische Seitenneuaufbau startet. So wird deutlich, dass auch hier der Client, also der Browser, Auslöser für die Kommunikation ist und diese in Gang setzt.

Da hier ein Browser als Benutzeroberfläche dient, ist Plattformunabhängigkeit gewährleistet. Dieser Aspekt kann sehr gut demonstriert werden, wenn unterschiedliche Endgeräte wie Laptops, Smartphones oder Tablets benutzt werden um die Website anzuzeigen.

Unter Beachtung dieser Ausführungen bei der Erstellung von Aufgaben für die Schüler, können alle gesteckten Lernziele erreicht werden:

- „Dinge des Internets“ werden durch die Verwendung der Sensoren als LGI-Module greifbar. Die Module können von den Schülern selbst, als von den übrigen Hardwarekomponenten getrennte Module, an das IoT-Modul angeschlossen werden. Sie werden dadurch als eigenständige „Dinge“ erkannt.
- Der Kommunikationsablauf im Internet und damit zusammenhängende Begriffe werden durch das Websitekonzept sichtbar und verständlich. Im Abschnitt 6.2.5 beschriebene sukzessiv aufeinander aufbauende Aufgaben unterstützen denn Lernprozess.
- die Bedeutung der Plattformunabhängigkeit wird durch die Verwendung von Browsern und Endgeräten unterschiedlicher Hersteller deutlich.
- Die Sensibilisierung für Datenschutz kann wie im Abschnitt 5.1 beschrieben, in zukünftigen Erweiterungen der Arbeit und unter Einbeziehung der Sensoren (LGI-Module) beigebracht werden.
- Die Sinnhaftigkeit von Anwendungen im Internet der Dinge kann, wie im Abschnitt 5.1 erläutert, mit den Schülern diskutiert werden.

5.5.4 Auswahl des am besten geeigneten Software- und Kommunikations-Konzepts

In dieser Arbeit wird das in Abschnitt 5.5.3 vorgestellte Softwarekonzept „Kommunikation mit dem IoT-Modul über dessen lokalen Webserver“ umgesetzt. Die Auswahlkriterien und die Bewertung der drei vorgestellten Softwarekonzepte sind in der Tabelle in der Abbildung 5.11 zusammengefasst.

Softwarekonzept	Komplexität für Lehrende	Lernziel: Kommunikation im Internet	Lernziel: Plattform-unabhängigkeit	Lernziel: Datenschutz	Lernziel: Kritischer Umgang
Mobile App (5.5.1)	Gering: Bekannte Komponenten	Gut umsetzbar	App ist nur für eine Plattform ausgelegt	Gut umsetzbar	Gut umsetzbar
Cloudservice (5.5.2)	Neue Komponenten (Clouddienst)	Nur eingeschränkt	Browser ist Plattform-unabhängig	Gut umsetzbar	umsetzbar
Website auf lokalem Webserver (5.5.3)	Gering: Bekannte Komponenten	Gut umsetzbar	Browser ist Plattform-unabhängig	Gut umsetzbar	Gut umsetzbar

Abbildung 5.11: Tabelle: Bewertung der vorgestellten Softwarekonzepte.

Die geringe Komplexität für Lehrer und Schüler bei der Bearbeitung der Aufgaben ist ein Vorteil des Gewählten Konzepts. Außerdem spricht die gute Umsetzbarkeit der Lernziele die sich Schüler teilweise selbst erarbeiten können für dieses Konzept. Wie dies geschehen kann wird im Abschnitt ?? beschrieben.

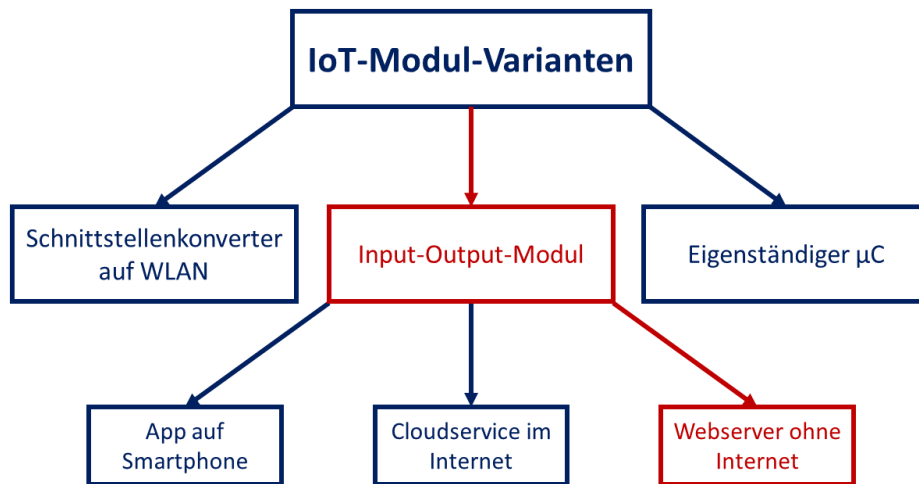


Abbildung 5.12: Diagramm: Entscheidung für das geeignetste Hardwarekonzept und Softwarekonzept.

Das in dieser Arbeit umgesetzte Hardware-Konzept ist eine Vorstufe zum gewählten Zielkonzept. Zunächst soll der Webserver und die Website mit dem ESP32 als alleinstehenden Mikrocontroller entworfen und getestet werden. Das langfristige Ziel ist mit diesem Entwurf ein Input-Output-Modul wie im Abschnitt 5.4.3 beschrieben zu entwickeln, das zusammen mit dem Arduino verwendet wird.

Kapitel 6

Realisierung

6.1 Hardware

6.1.1 Vergleich der Mikrocontrollerchips ESP8266 und ESP32

Der ESP32 hat im Vergleich zu seinem Vorgängermodell einige Vorzüge. Er läuft stabiler, unter anderem wohl bedingt durch die zwei Kerne des verbauten Prozessors Tensilica Xtensa LX6. Die Systemsoftware wird offen entwickelt, es kann also jeder testen und Verbesserungen vorschlagen. So werden Instabilitäten schneller und besser gefunden und beseitigt. Der Prozessor hat eine schnellere

	ESP32	ESP8266
Prozessor	Tensilica Xtensa LX6 32 bit Dual-Core bis 240 MHz	Tensilica L106 32-bit Single-Core 80 MHz
RAM	512 kB	<50 kB verfügbar
Flashspeicher	Extern bis zu 4x 16 Mbyte, zusätzlich: On chip embedded flash	Extern bis zu 16 Mbyte --
Pins / GPIO	32	17
PWM-Pins	Bis zu 16	4
Bluetooth	BLE und klassisches BT	--
WLAN	802.11 b/g/n/e/i	802.11 b/g/n/e/i

Abbildung 6.1: Vergleich der wichtigsten Merkmale des ESP8266 und seinem Nachfolger ESP32. Angaben nach [37] und [38].

Taktrate von bis zu 240 MHz statt zuvor maximal 180 MHz. Auf dem Chip ist ein eingebetteter Flashspeicher für das Speichern von Anwendungsdateien und Programmen vorhanden. Der extern anschließbare Speicherplatz ist größer dimensioniert. Der ESP32 hat mehr Pins und vor allem mehr PWM-Pins als der ESP8266. WLAN wird wieder mit den Spezifikationen 802.11 b/g/n/e/i unterstützt und als Neuerung zusätzlich Bluetooth, was für künftige Lehrmodule in lets goING interessant sein kann und somit bei Verwendung des ESP32 für die Zukunft schon verfügbar ist. Weitere Einzelheiten sind der Abbildung 6.1 zu entnehmen. [37] [38] Aufgrund seiner Vorteile wird für diese Arbeit das Modell ESP32 verwendet.

Der ESP32 wird von SparkFun¹ auf einem Entwicklerboard angeboten. Auf diesem Board sind Anschlüsse und Pins so ausgeführt, dass das Board sofort einsatzbereit ist und auf einer Steckplatine verwendet werden kann. Es können auf einfache Weise Komponenten (Sensoren, Aktoren) wie die LGI-Module angeschlossen werden. Für den Entwurf des IoT-Moduls in dieser Arbeit wird ein solches ESP32-Thing-Entwicklerboard verwendet.

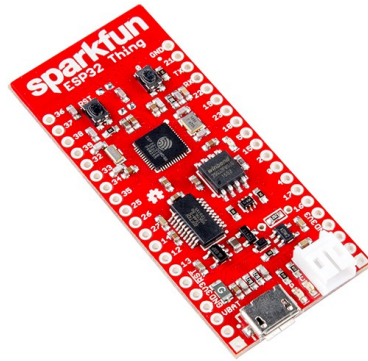


Abbildung 6.2: SparkFun ESP32 Thing Entwicklungsplattform [39].

6.1.2 Aufbau der Hardwarekomponenten

Das ESP32-Thing-Entwicklerboard (ESP32) ist auf eine Steckplatine aufgebracht. Die letsGoING-Module mit ihren Sensoren und Aktoren sind an den ESP32 angeschlossen. Es wird eine dreifarbige RGB-LED, ein Potentiometer, ein Helligkeitssensor (LDR-Sensor) und die auf dem ESP32Thing-Board verbaute LED für Tests verwendet. Die drei Farben der RGB-LED, rot, grün und blau, können beliebig zu Mischfarben kombiniert werden.

¹<https://learn.sparkfun.com/tutorials/esp32-thing-hookup-guide>

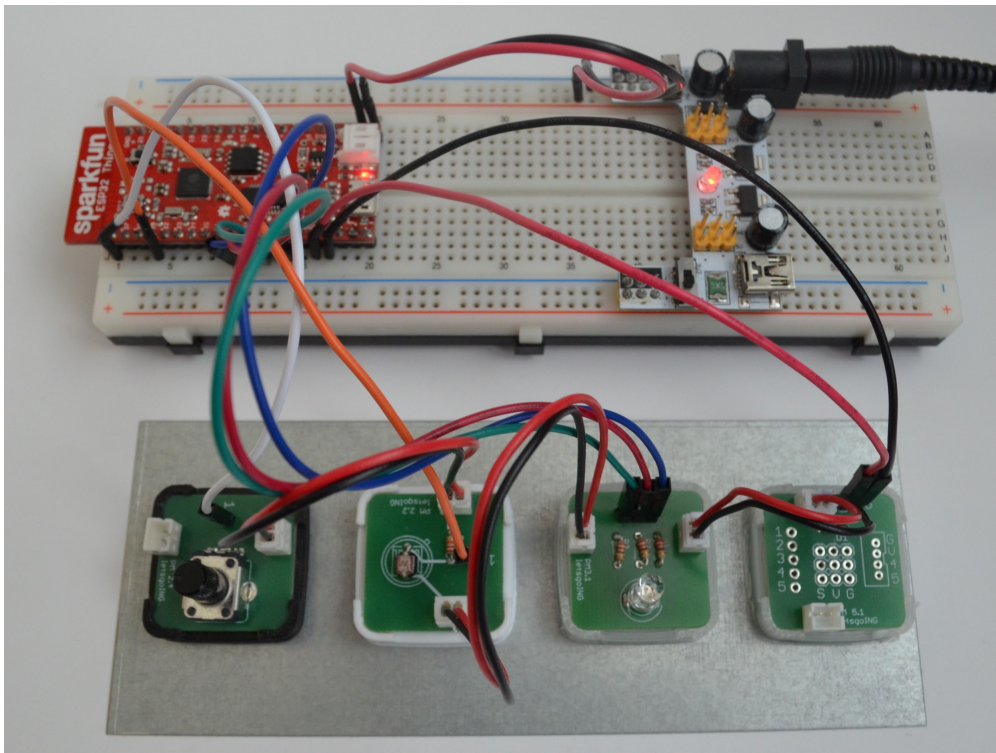


Abbildung 6.3: Versuchsaufbau: oben: ESP32 Thing mit Spannungswandler, unten: (v.l. n.r.) Die LIG-Module Potentiometer, LDR-Lichtsensord, RGB-LED, Stromadaptermodul

Der Aufbau ist zunächst provisorisch und kann für zukünftige Projekte weiterentwickelt werden. Das ESP32-Thing-Entwicklerboard könnte durch ein eigenes, für letsgoING entworfenes Board ausgetauscht werden, welches an die bestehenden letsgoING-Komponenten angepasst wird. In Zukunft soll das IoT-Modul mit den angeschlossenen LIG-Modulen alleinstehend verwendet werden oder gemeinsam mit dem Arduino auf das letsgoING-Fahrzeug aufgebracht werden.

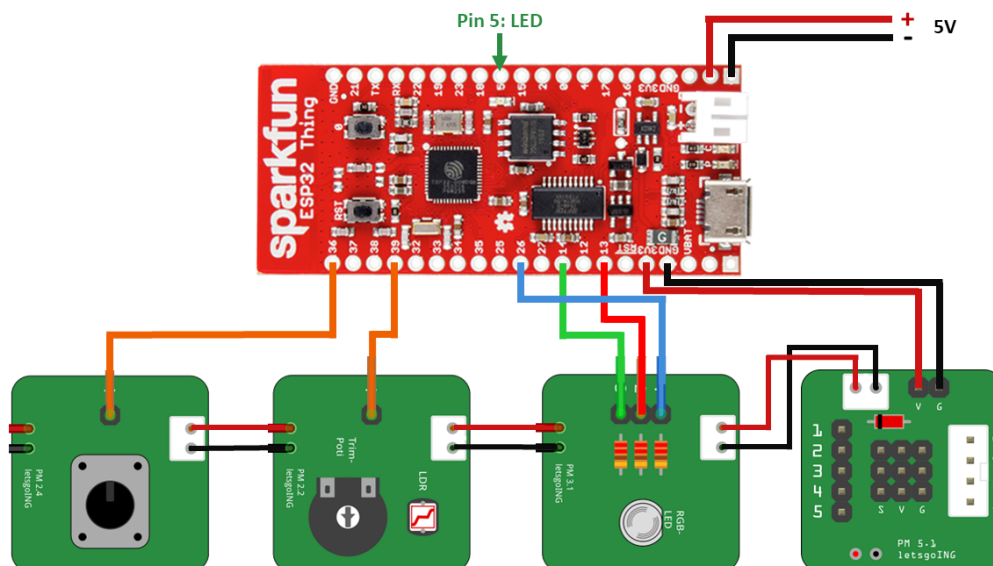


Abbildung 6.4: Schaltbild für den Aufbau.

Das Potentiometer wird an Pin 36 angeschlossen, der Helligkeitssensord an Pin 39. Die RGB-LED

wird an Pin 13 (rot), Pin 14 (grün) und Pin 26 (blau) angeschlossen. Über den 3,3-Volt-Anschluss des Boards und Ground werden die LIG-Module mit Strom versorgt. Der genaue Anschlussplan ist dem Blockschaltbild in Abbildung 6.4 zu entnehmen. Die letsGoING-RBG-LED ist für den Betrieb mit 5V Spannung ausgelegt. Im Rahmen dieser Arbeit reichen aber 3,3V aus. Für die Weiterentwicklung des IoT-Moduls sollte dies jedoch beachtet werden.

Pinbelegung

Die Anschlüsse für Input und Output der Sensoren und Aktoren befinden sich an folgenden Pins:

Sensor/Aktor	Pinnummer
On-Board-LED	5
digitaler Eingang	15
Potentiometer	36
LDR-Sensor	39
RGB-LED: rot	13
RGB-LED: grün	14
RGB-LED: blau	26

Abbildung 6.5: Pinbelegung der Sensoren und Aktoren.

Das Potentiometer entspricht dem „Sensor 1“ auf der Homepage, der Wert des LDR-Sensors wird im Feld „Sensor 2“ angezeigt.

Stromversorgung

Die Stromversorgung erfolgt entweder über ein USB-Kabel das mit dem USB-Port des Computers verbunden ist oder über ein externes Netzgerät, das das ESP32-Entwicklerboard mit 5 Volt Spannung versorgt. In Abbildung 6.3 erfolgt die Stromversorgung über ein Netzteil. Dazu muss der ESP32 auf ein Breadboard gesteckt werden und mit einem Spannungswandler verbunden sein. Im Zuge der Weiterentwicklung des IoT-Moduls sollte auch über eine geeignete Stromquelle nachgedacht werden. Es würde sich anbieten den vorhandenen, für das Fahrzeug genutzten Akku zu verwenden. Dieser liefert 5V Spannung.

6.2 Software

6.2.1 Programmablauf

Auf den ESP32 wird einmalig eine Firmware geladen, die beim Durchführen des Lehrmoduls nicht mehr verändert wird. Die vorangehenden letsGoING-Lernmodule behandeln die Programmierung mit Ardublock bereits intensiv. Die Schüler sollen sich in diesem Lernmodul auf das Thema „Internet der Dinge“ und „Kommunikation im Internet“ konzentrieren. Die Kommunikation wird anhand der gestellten Aufgaben mit der Website erarbeitet. Das Eingreifen in den Programmcode ist dafür nicht zwingend notwendig. Die Fehlerquellen wären sehr groß wenn Schüler den Programmcode verändern und der Zeitaufwand würde dadurch erheblich steigen, ohne dabei neue Themen zu erschließen.

Das Softwareprogramm besteht aus mehreren Teilen: a) Logik für den Mikrocontroller und die Netzwerkverbindung, b) die Website und c) Logik für den Server. Das Softwareprogramm ist in

einer C-Plus-Plus-Datei (.cpp-Datei) auf dem PC gespeichert und wird über die Entwicklungsumgebung auf den ESP32 geladen.

Der erste Schritt im Programmablauf ist die Herstellung einer Netzwerkverbindung wie im Abschnitt 6.2.2 erläutert. Dann kann sich ein Endgerät (Client) mit dem Server (IoT-Modul) auf dem ESP32 verbinden und die Kommunikation kann über einen Server- und eine Client-Instanz ablaufen. Die Website kann dann über den Browser auf dem Endgerät aufgerufen werden.

Der Code für die Funktionen des Mikrocontroller umfasst die Steuerung angeschlossener Komponenten. Es werden ständig zyklisch die Sensorwerte für das Potentiometer und den LDR-Sensor eingelesen. Digitale Werte für die auf dem ESPThing-Board verbaute LED und drei PWM-Werte für eine angeschlossene RGB-LED werden gesteuert. Dafür ist eine Abfrage implementiert, welche den vom Browser ankommenden URI-Code durchsucht und je nach Inhalt des URI weitere Funktionen aufruft. Dies geschieht bei jeder Client-Anfrage.

```
305     void eventHandler() {
```

Abbildung 6.6: Über die Funktion „Event-Handler“ wird der URI-Code durchsucht

Der Programmcode wurde mit der Entwicklungsumgebung „Atom“² erstellt. Er ist in einer C-plus-plus-Datei auf dem PC gespeichert und wird über eine USB-Schnittstelle auf den ESP32 übertragen.

6.2.2 Verbindungsaufbau mit Netzwerken

Der ESP32 kann entweder als Access Point programmiert werden oder er kann eine Verbindung zu einem bestehenden Netzwerk aufbauen. Im Quellcode müssen dafür die Daten eines vorhandenen WLAN-Netzwerks hinterlegt sein. Es ist möglich den Netzwerknamen (SSID) und das Passwort eines bestehenden WLAN-Netzwerks im Quellcode einzugeben, damit sich der ESP32 mit diesem verbindet. Außerdem kann an der gleichen Stelle die SSID und das Passwort für den Access Point des ESP32 abgelesen (oder geändert) werden. Die SSID lautet „LGI-IoT-Modul“. Das Passwort muss genau 8 Zeichen lang sein und lautet: „12345678“. Nach dem Starten des Programms (=Anschluss an Stromversorgung), wird versucht eine Verbindung mit dem angegebenen Netzwerk herzustellen. Gelingt dies nicht, baut der ESP32 einen Access Point auf. Es ist aber nicht gewünscht, dass an den

```
15     /*Wifi Access data*/
16     const char* ssid_ap = "LGI-IoT-Modul";
17     const char* password_ap = "12345678";
18
19     const char * ssid_sta     = "SSID";
20     const char * password_sta = "PW";
21
```

Abbildung 6.7: Quellcode: Passwort und Netzwerkname (SSID).

Schulen Änderungen am Quellcode vorgenommen werden. Für die Eingabe soll daher in Zukunft eine Abfrage eingefügt werden, die automatisch beim Einschalten des ESP32 startet. Der Benutzer gibt dann den Namen (SSID) und das Passwort seines WLAN-Netzwerks ein und nach Bestätigung verbindet sich der ESP32 mit dem gewünschten WLAN-Netzwerk. Da die WLAN-Netzwerke an vielen Schulen oft nur eingeschränkt zur Verfügung stehen, ist eine weitere Möglichkeit, ein Netzwerk aufzubauen, notwendig. Denkbar ist ein eigenes Netzwerk mit Internetzugang bereit zu stellen. Dies könnte ein eigenes Mobiles Netzwerk sein das von letsGoING bereitgestellt wird, wie in Abschnitt 5.2 beschrieben ist.

²Weitere Informationen sind auf der Website zu finden: <https://atom.io/>

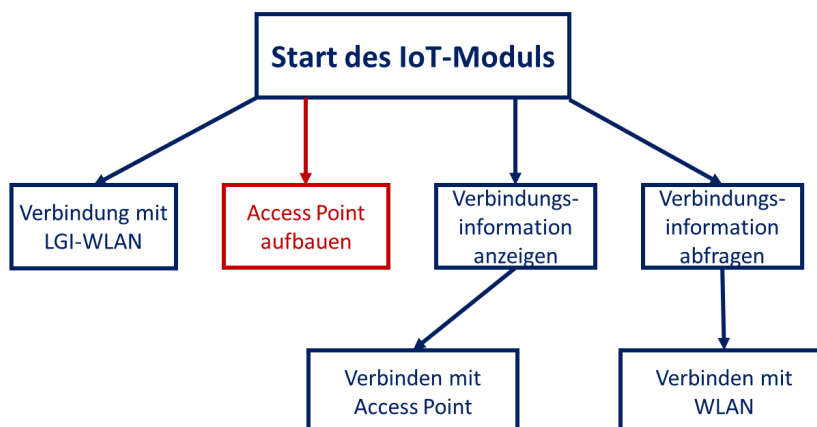


Abbildung 6.8: Optionen für den Start des IoT-Moduls.

Die in dieser Arbeit umgesetzten Komponenten sind jedoch ohne Internetverbindung voll funktionsfähig. Daher startet der ESP32 generell im Access-Point-Modus und baut ein lokales Netzwerk auf. Laptops, Smartphones und andere Geräte können sich mit diesem Netzwerk verbinden. Dafür könnte es ebenfalls eine Anzeige geben, die über die Verbindungsdaten mit dem Access Point informiert, sobald ein Endgerät versucht sich mit dem Access-Point-Netzwerk zu verbinden. Die verschiedenen Optionen für den Start des IoT-Moduls sind im Baumdiagramm in der Abbildung 6.8 dargestellt.

Wenn mehrere IoT-Module im Unterricht parallel zum Einsatz kommen, muss noch geklärt werden, wie jedes einzelne Modul für die Verbindung erkannt werden kann. Jede Schülergruppe soll ihre Geräte (Laptop, Smartphone, Tablet) mit ihrem eigenen IoT-Modul-Netzwerk verbinden können. Eventuell kann eine Unterscheidung über den Netzwerknamen erfolgen oder über die MAC-Adresse. Es ist nicht vorgesehen die Firmware für jedes IoT-Modul zu ändern. Etwaige Änderungen müssten beim Start über eine Abfrage geschehen, wie oben und in der Abbildung 6.8 als Option genannt. Eine weitere Möglichkeit wären Dip-Schalter die mit dem ESP32 auf einer Platine verbaut werden³.

6.2.3 Digitale und analoge Werte verarbeiten und austauschen

Durch Klicken der Schaltflächen „Ein“, „Aus“ oder „senden“ auf der Homepage, wird Information (Text und Zahlenwerte) vom Endgerät (Client) an das IoT-Modul (Server auf ESP32) gesendet oder umgekehrt. Dies geschieht über HTTP.

In der Adresszeile (im URI) ist der Textteil der Protokollnachricht zu sehen. Im Programm auf dem Server (ESP32) wird dieser Text verarbeitet. Der Text befindet sich dort in der String-Variable `sHTMLRequest` und stellt eine GET-Methode dar, wie im Abschnitt 4.2.3 erklärt. Der String wird von einer Methode im Programm ausgewertet (Abbildung 6.6). Die vom Client übergebenen Werte können so herausgefunden und auf die entsprechenden Pins gesetzt werden.

Wird die Schaltfläche „aktualisieren“, auf der Webseite neben den Sensorwerten, geklickt, bekommt der Server vom Client einen Befehl die Seite neu zu laden. Dabei werden dem Client aktuelle Messwerte übergeben, die dieser im Browserfenster anzeigt.

³Über das Setzen von 8 Dip-Schaltern könnte eine Zahl zwischen 0 und 255 generiert werden die an die SSID angehängt wird.

6.2.4 Konfiguration und Bibliotheken

Für den Webserver wird die Bibliothek „Wifi.h“ für den ESP32 verwendet⁴. Sie beinhaltet Befehle zum Übergeben von Information an der Browser. Mit „client.print(html_document())“ wird ein String übergeben in welchem sich der HTML- und JavaScript-Code für die Website befindet.

Da sich der ESP32 noch in der Entwicklung befindet, sind Bibliotheken noch nicht für alle Funktionen verfügbar. Das Dateisystem für den internen Speicher kann noch nicht verwendet werden. Der Quellcode für die Website wird daher in einen String geschrieben. Wenn in Zukunft eine stabil laufende Bibliothek verfügbar ist, kann ein anderer Weg für die Übertragung des Quellcodes und der Messwerte der Sensoren entwickelt werden.

Um die PWM-Werte für die RGB-LED setzten zu können, müssen analoge Pins entsprechend konfiguriert werden, damit sie als PWM-Pins verfügbar sind. Die Pins nutzen einen analog-Digital-Konverter. Um diesen zu verwenden, ist die Bibliothek „adc.h“⁵ erforderlich. Es müssen einige Parameter und die Kanäle eingestellt werden, die der Konverter nutzen soll.

```
9  #include "driver/adc.h"

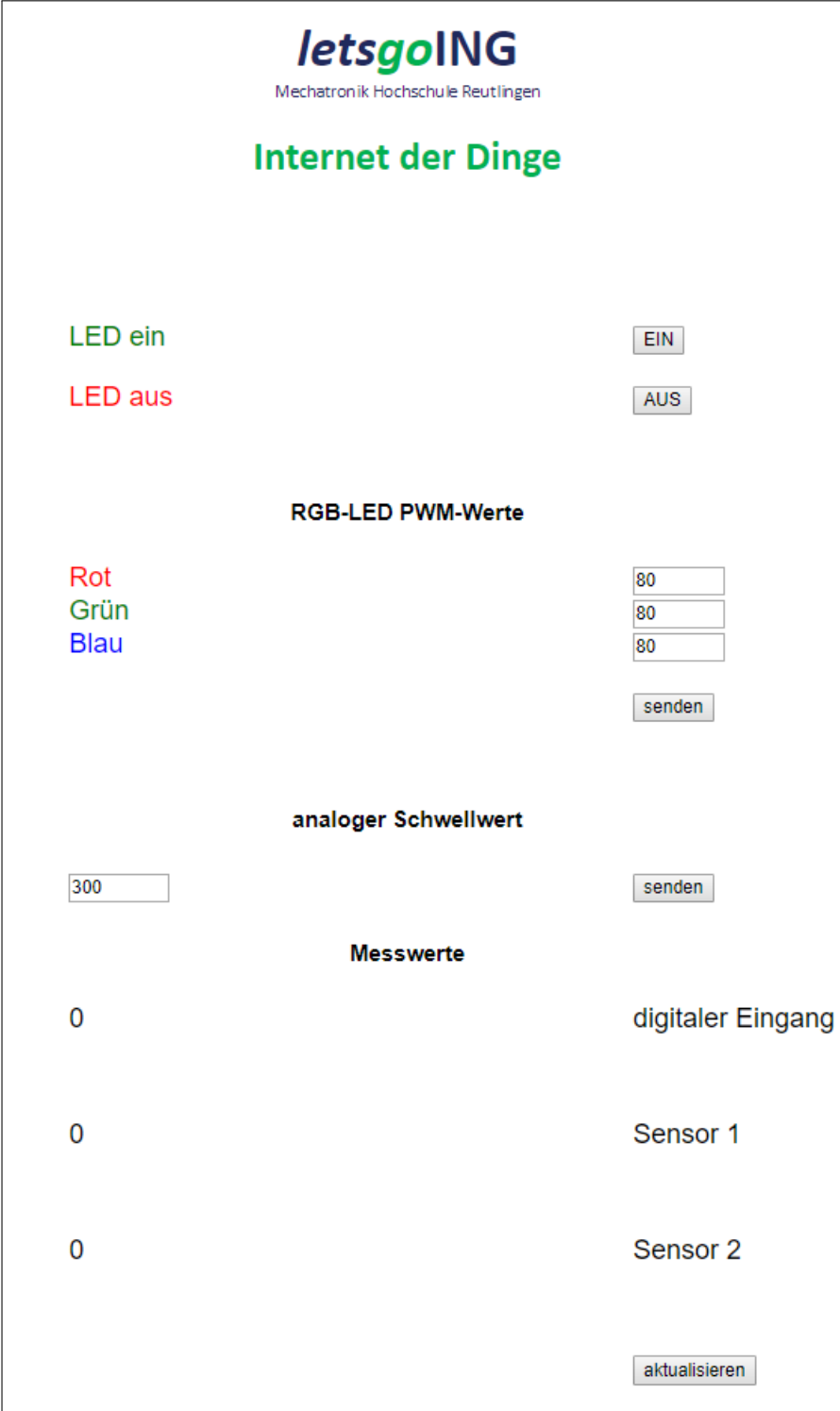
349  pinMode(13, OUTPUT); //red
350  pinMode(14, OUTPUT); //green
351  pinMode(26, OUTPUT); //blue
352  //pinMode(15, INPUT); //digital sensor (z.B. Taster)
353
354  ledcAttachPin(pinRot, 1); // assign RGB led pins to channels
355  ledcAttachPin(pinGruen, 2);
356  ledcAttachPin(pinBlau, 3);
357  /* Initialize channels: channels 0-15, resolution 1-16 bits, freq limits depend on resolution
358   * ledcSetup(uint8_t channel, uint32_t freq, uint8_t resolution_bits);*/
359  ledcSetup(1, 12000, 8); // 12 kHz PWM, 8-bit resolution
360  ledcSetup(2, 12000, 8);
361  ledcSetup(3, 12000, 8);
362
363  adc1_config_width(ADC_WIDTH_12Bit);
364  adc1_config_channel_atten(ADC1_CHANNEL_0, ADC_ATTEN_0db);
```

Abbildung 6.9: Konfiguration der PWM-Pins.

⁴Link auf github: <https://github.com/espressif/arduino-esp32/tree/master/libraries/WiFi>

⁵Für genaue Informationen zur Konfiguration und Funktionsweise siehe: <http://espressif.readthedocs.io/en/latest/api-reference/peripherals/adc.html>. Abrufdatum 06.09.2017.

6.2.5 Website-Design und Umsetzung der Lernziele



letsGOING
Mechatronik Hochschule Reutlingen

Internet der Dinge

LED ein

LED aus

RGB-LED PWM-Werte

Rot

Grün

Blau

analoger Schwellwert

Messwerte

0 digitaler Eingang

0 Sensor 1

0 Sensor 2

Abbildung 6.10: Layout der Webseite.

Die Webseite ist bewusst in einem einfachen Design gehalten. Die Schüler sollen nicht durch aufwendige Grafiken oder Animationen, abgelenkt werden. Der Entwurf ist in Abbildung 6.10 zu sehen.

Alle Komponenten sind untereinander und zentriert angeordnet, damit die Seite auf Geräten mit unterschiedlichen Bildschirm- und Displaygrößen angezeigt werden kann. Die Anzeige wurde auf dem Laptop in den Browsern Google Chrome, Mozilla Firefox und Microsoft Edge und auf dem

Smartphone (Sony Xperia Z5 compact) im Browser Google Chrome getestet.

Die gewählte Anordnung der Elemente soll eine gute Übersicht bieten. Von oben nach unten wird eine Zeile oder Elementgruppe nach der anderen in das didaktische Konzept einbezogen. Die Aufgaben sind so gestellt, dass Stück für Stück die Kommunikation im Internet verstanden wird. Jede Aufgabe baut auf den vorhergehenden auf und vertieft das zuvor erlangte Wissen. Anhand der Aufgabenstellungen sollen die Schüler mittels Transferleistung selbst Muster und Funktionsweisen der Internetkommunikation erkennen. Die Aufgaben und ihre Reihenfolge könnte sich an folgenden Grundgerüst orientieren:

Aufgabe 1

- a) Schalte die LED durch Klicken der beiden Buttons ein und aus. Beobachte was in der Adresszeile passiert und notiere deine Beobachtungen.
- b) Versuche nun die LED ein- und auszuschalten, ohne die Buttons zu klicken.

Anmerkung: Hier wird klar, dass URL-Code über die Adresszeile übertragen wird der anschließend vom Server verarbeitet und als Befehl interpretiert wird. Es erscheinen die Worte „LEDON“ bzw. „LEDOFF“ in der Adresszeile (siehe Abbildung 6.11. Diesen Text sollen die Schüler in der Aufgabe direkt in die Adresszeile schreiben und mit der Eingabetaste oder der entsprechenden Schaltfläche im Browser bestätigen. Die Befehle (LEDON / LEDOFF) werden dann vom Server „gelesen“ und verarbeitet. Der Server reagiert mit einem Befehl an den Mikrocontroller, die LED ein- und auszuschalten. Im Unterricht sollten diese Vorgänge dann erläutert werden. Denkbar wäre das anhand eines Schaubildes wie in Abbildung 5.1 zu sehen ist. In einer Vertiefungsaufgabe könnten die Felder im Schaubild dann von den Schülern ausgefüllt werden oder das Schaubild eigenständig skizziert werden.



Abbildung 6.11: Adresszeile nach Klicken des Buttons „Ein“.

Aufgabe 2

- a) Sende PWM-Werte an die RGB-LED und mische unterschiedliche Farben und Helligkeiten. Beobachte wieder, was in der Adresszeile steht und notiere deine Beobachtungen.
- b) Versuche nun die Farben einzustellen, ohne die Schaltflächen zu benutzen.

Anmerkung: An dieser Stelle kann das Benutzen von Variablen in der URL kurz besprochen werden. Aus anderen letsgoING-Modulen sollten Variablen schon bekannt sein.

Das Client-Server Konzept für die Kommunikation kann auch hier, wie unter Aufgabe 1 erläutert, erklärt werden. Der Client löst das Senden der Werte an den Server aus (siehe Abbildung 6.12. Dort wird dieser weiter verarbeitet.

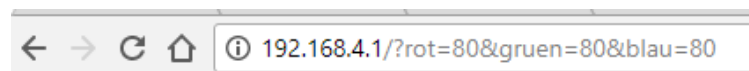


Abbildung 6.12: Adresszeile nach Klicken des Buttons „senden“ neben den PWM-Werten.

Aufgabe 3

Diese Aufgabe setzt voraus, dass das Modul mit dem Arduino verbunden ist und Ardublock benutzt werden kann. Das ist innerhalb dieser Arbeit noch nicht der Fall. In Zukunft könnte eine Aufgabe dann wie folgt lauten:

Sende einen analogen Schwellwert an den Arduino. Verwende diesen in Ardublock: Entwerfe ein kurzes Programm für einen Lautstärkesensor, mit dem übertragenen Wert als Grenzwert. Wenn der Wert überschritten wird, soll die LED leuchten.

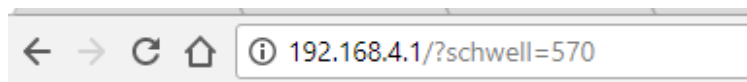


Abbildung 6.13: Adresszeile nach senden eines Schwellwertes.

In weiteren Aufgaben kann erklärt werden wie die Kommunikation in umgekehrter Richtung verläuft. Dazu werden die Felder für die Anzeige von Sensorwerten genutzt. Beim Klicken des Buttons „aktualisieren“ werden die vom Programm zuletzt gespeicherten Sensordaten auf der Homepage aktualisiert.

Aufgabe 4

- Drehe am Poti und beobachte das Feld für Sensor 1. Passiert etwas? Notiere deine Beobachtung.
- Klicke den Button „aktualisieren“ und beobachte erneut. Notiere deine Beobachtung.

Du hast in den Aufgaben 1-3 gelernt, wie die Kommunikation zwischen dem IoT-Modul (=Server) und der Webseite (=Client) funktioniert. Was ist die Voraussetzung? Was ist die logische Erklärung dafür, dass nicht ständig aktuelle Werte automatisch auf der Webseite angezeigt werden?

In Zukunft können weitere Aufgaben ausgearbeitet werden, um die Lernziele zu vertiefen.

6.3 Inbetriebnahme des IoT-Moduls

Sobald der ESP32 an die Stromversorgung angeschlossen ist (siehe Abschnitt 6.1.2), startet das gespeicherte Programm. Sollte dies nicht geschehen, kann das Drücken des Reset-Tasters („RST“) auf dem Board helfen.

Verbindung mit Endgeräten

Wie in Abschnitt 6.2.2 beschrieben, kann sich der ESP32 mit einem bestehenden Netzwerk verbinden sofern dessen Name und Passwort angegeben wurde. Andernfalls baut der ESP32 ein eigenes Netzwerk auf. Mit diesem Netzwerk können sich Endgeräte unter Verwendung des Namens „LGI-IoT-Modul“ und des Passworts „12345678“ verbinden.

Nach erfolgreicher Verbindung kann über den Internetbrowser des Endgeräts die Website auf dem Server aufgerufen werden. Dazu wird in die Adresszeile des Browsers die IP-Adresse des ESP32 eingegeben: „192.168.4.1“.

Kapitel 7

Zusammenfassung und Ausblick

In dieser Arbeit ist ein Erstentwurf für ein IoT-Modul für das Hochschul-Schul-Kooperationsprojekt letsgoING entstanden. Es ist in der aktuell vorliegenden Form noch nicht für den Einsatz in der Lehre geeignet. Es wurde ein Hardwarekonzept umgesetzt, das einen alleinstehenden Mikrocontroller verwendet. Ziel ist es, dieses weiterzuentwickeln und ein Input-Output-Modul zu gestalten, wie es im Abschnitt 5.4.3 beschrieben ist. Dieses soll dann zusammen mit dem Arduino und den letsgoING-Modulen eingesetzt werden.

Ein für die letsgoING-Komponenten passendes Modulbreakoutboard kann noch entworfen werden. So könnte das IoT-Modul für künftige Aufgaben auf das letsgoING-Fahrzeug aufgesteckt werden. In diesem Zusammenhang ist dann zu klären, wie der ESP32 mit Strom versorgt wird.

Das in dieser Arbeit entstandene Softwarekonzept für das IoT-Modul kann noch weiterentwickelt werden. Sobald für den Chip ESP32 bislang fehlende Bibliotheken zur Verfügung stehen, kann die Software verbessert werden. Die Homepage kann um weitere Seiten für neue Projekte und Aufgabestellungen ergänzt werden.

Im Projekt arbeiten die Schüler in Kleingruppen. Es ist noch zu erörtern wie mit mehreren IoT-Modulen umgegangen wird, die in einem Raum benutzt werden. Diese Problematik ist im Abschnitt 6.2.2 beschrieben.

Die Verbindung mit Endgeräten im Access-Point-Modus bricht bei Testläufen nach einiger Zeit ab. Es ist zu klären, wann und warum das geschieht und ob es möglich ist, den Verbindungsabbruch zu verhindern.

Die Lernziele zum Thema „Datenschutz“ und „Sinnhaftigkeit von IoT-Anwendungen“ können noch weiter ausgearbeitet werden. Es könnte ein Konzept entwickelt werden, das diese Themen aufgreift. Bestenfalls erarbeiten sich die Schüler diese Themen selbstständig und mittels einer praktischen Anwendung.

Literaturverzeichnis

- [1] BUNDESAGENTUR FÜR ARBEIT STATISTIK/ARBEITSMARKTBERICHTERSTATTUNG: *Der Arbeitsmarkt in Deutschland - MINT-Berufe*. Bundesagentur für Arbeit Statistik/Arbeitsmarktberichterstattung, Nürnberg, 2016. <https://statistik.arbeitsagentur.de/Statischer-Content/Arbeitsmarktberichte/Branchen-Berufe/generische-Publikationen/Broschuere-MINT-2016.pdf> , Abrufdatum 30.05.2017
- [2] DURNER, DR. ALEXANDRA, HEIKE VOM ORDE: *Aktuelle Ergebnisse zur Mediennutzung von Jugendlichen in Deutschland*. Internationales Zentralinstitut für das Jugend- und Bildungsfernsehen, München, 2017. http://www.br-online.de/jugend/izi/deutsch/Grunddaten_Jugend_Medien.pdf , Abrufdatum 30.05.2017.
- [3] MINISTERIUM FÜR KULTUS JUGEND UND SPORT: *Bildungsplan 2016 - Gymnasium Naturwissenschaft und Technik (NwT) - Profulfach*. Ministerium für Kultus Jugend und Sport, Stuttgart, 2016. http://www.bildungsplaene-bw.de/site/bildungsplan/get/documents/lsbw/export-pdf/depot-pdf/ALLG/BP2016BW_ALLG_GYM_NWT.pdf , Abrufdatum 30.05.2017.
- [4] KNOKE, MAREIKE: *Mint im Stand-by-Modus* in VDI nachrichten, Verein Deutscher Ingenieure Verlag GmbH, Düsseldorf, Ausgabe Nr. 24, 16. Juni 2017.
- [5] WIDROW, BERNARDREINER HARTENSTEINROBERT HECHT-NIELSEN: *1917 Karl Steinbuch 2005*. IEEE Computational Intelligence Society, Satnford, Kaiserslautern, San Diego, 2005. <http://helios.informatik.uni-kl.de/euology.pdf>, Abrufdatum 06.09.2017
- [6] STEINBUCH, KARL: *Die informierte Gesellschaft: Geschichte und Zukunft der Nachrichtentechnik*. Deutsche Verlags-Anstalt, Stuttgart, 1966.
- [7] NATIONAL INSTITUTE OS STANDARDS AND TECHNOLOGY: *NIST's Network-of-Things Model Builds Foundation to Help Define the Internet of Things*. 16.07.2016. <https://www.nist.gov/news-events/news/2016/07/nists-network-things-model-builds-foundation-help-define-internet-things>, Abrufdatum 05.05.2017.
- [8] VOAS, JEFFREY: *NIST Special Publication 800-183, Networks of 'Things'*. National Institute of standards an Technology, Gaithersburg, 2016. <http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-183.pdf>, Abrufdatum 05.05.2017.
- [9] SCHILLER,KAI: *Was ist ein Smart Home - Geräte und Systeme* homeandsmart, Karlsruhe, 2016. <https://www.homeandsmart.de/was-ist-ein-smart-home>, Abrufdatum 18.08.2017.
- [10] CHIP.DE: *Fitnesstracker im Preis-Leistungs-Vergleich*. Chip Media GmbH, München, 26.01.2017. <http://www.chip.de/artikel/Fitness-Armband-guenstig-Alle-Fitness-Tracker-im-Preis-Leistungs-Vergleich.74630696.html>, Abrufdatum 05.05.2017.
- [11] BUNDESMINISTERIUM FÜR BILDUNG UND FORSCHUNG: *Zukunftsprojekt Industrie 4.0*. Bundesministerium für Bildung und Forschung, 2016. <https://www.bmbf.de/de/zukunftsprojekt-industrie-4-0-848.html>, Abrufdatum 05.05.2017.

- [12] BUNDESMINISTERIUM FÜR WIRTSCHAFT UND ENERGIE: *Zukunftsprojekt Industrie 4.0, Häufig gestellte Fragen*. Bundesministerium für Wirtschaft und Energie, 2017. <http://www.plattform-i40.de/I40/Navigation/DE/Service/FAQ/faq.html>, Abrufdatum 05.05.2017.
- [13] SCHMIDT, RAINER, MICHAEL MÖHRING, RALF-CHRISTIAN HÄRTING, CHRISTOPHER REICHSTEIN, PASCAL NEUMAIER, PHILIP JOZINOVIC: *Industry 4.0 - Potentials for Creating Smart Products: Empirical Research Results*. In *Lecture Notes in Business Information Processing* Nr. 208. Springer International Publishing, 2015.
- [14] HEILMANN, DIRK, LUDWIG EICKEMEYER, JAN KLEIBRINK: *Industrie 4.0 im Internationalen Vergleich*. Handelsblatt Research Institute, Düsseldorf, 2016.
- [15] SCHWAB, KLAUS: *Die Vierte Industrielle Revolution*. Pantheon, München, 2016.
- [16] STEINMÜLLER, PETER: *Die Digitale Transformation als Jobmotor für Deutschland* in VDI nachrichten, Verein Deutscher Ingenieure Verlag GmbH, Düsseldorf, Ausgabe Nr. 20, 19. Mai 2017.
- [17] INSTITUT FÜR GEOINFORMATIK: *SenseBox*. Institut für Geoinformatik, Universität Münster, Münster, 2017. <https://sensebox.de/>, Abrufdatum 20.04.2017.
- [18] WATTEROTT ELECTRONIC GMBH: *SenseBox mit WLAN*. Watterott electronic GmbH, Leinfelde, 2017. <http://www.watterott.com/de/senseBox-EDU-Kit-WLAN>, Abrufdatum 20.04.2017.
- [19] DAIMLER AG (HRSG.): *Genius die junge Wissenscommunity von Daimler*. Daimler AG, Stuttgart, 2017. <https://www.genius-community.com/>, Abgerufen am 06.05.2017.
- [20] GASCHKE, SUSANNE: *Die verkaufte Kindheit: Wie Kinderwünsche vermarktet werden und was Eltern dagegen tun können*. Pantheon, München, 2011
- [21] CALLIPOPE GMBH: *Über den mini*. Calliope gemeinnützige GmbH, Berlin, 2017. <https://calliope.cc/ueber-mini>, Abgerufen am 06.05.2017.
- [22] BÄHRING, HELMUT: *Anwendungsorientierte Mikroprozessoren - Mikrocontroller und Digitale Signalprozessoren*. Springer-Verlag, Berlin Heidelberg 4. Auflage, 2010.
- [23] NURMI, JARI: *Processor Design - System-on-Chip Computing for ASICs and FPGAs*. Springer-Verlag, Dordrecht, 2007.
- [24] BADACH, ANATOL, ERWIN HOFFMANN: *Technik der IP-Netze*. Carl Hanser Verlag, München, 3. Auflage, 2015.
- [25] OBERMANN, KRISTOF, MARTIN HORNEFFER: *Datennetztechnologien für Next Generation Networks*. Springer Vieweg, Wiesbaden, 2. Auflage 2013.
- [26] IGOE, TOM: *Making Things Talk - Die Welt hören sehen fühlen*. O'Reilly, Köln, 2012.
- [27] BUNDESAMT FÜR SICHERHEIT IN DER INFORMATIONSTECHNIK: *IT-Grundschatzkataloge 15 EL 2016*. Bundesamt für Sicherheit in der Informationstechnik, Berlin, 2016. Abschnitt B 5.4. https://www.bsi.bund.de/DE/Themen/ITGrundschatz/ITGrundschatzKataloge/Bezugsquellen/bezugsquellen_node.html, Abrufdatum 24.03.2017.
- [28] LAURIE, BEN, PETER LAURIE: *Apache - Das umfassende Handbuch*. O'Reilly, Köln, 2. Auflage, 2003.
- [29] SCHNABEL, PATRICK: *Kommunikationstechnik-Fibel - Grundlagen der Kommunikationstechnik* Elektronik-Kompendium.de, Ludwigsburg, 2016. <https://www.elektronik-kompendium.de/sites/kom/index.htm>, Abrufdatum 24.08.2017

- [30] FREEMAN, ERIC, ELISABETH ROBSON: *JavaScript-Programmierung von Kopf bis Fuß*. O'Reilly, Köln, 2014.
- [31] KERESZTURI, ALEX: *Aller guten Dinge sind 3: HTML, CSS und JavaScript im Zusammenspiel*. Digicomp academy, Zürich, 2017. <http://news.digicomp.ch/de/2014/08/21/aller-guten-dinge-sind-3-html-css-und-javascript-im-zusammenspiel/>, Abrufdatum 25.08.2017.
- [32] GRIMM, MARKUS: *DSL, WLAN, FRITZ!Box und Co.* HERDT-Verlag für Bildungsmedien, Bodenheim, 2012.
- [33] SCHNABEL, PATRICK: *Netzwerktechnik-Fibel - Grundlagen der Netzwerktechnik* Elektronik-Kompodium.de, Ludwigsburg, 2016. <https://www.elektronik-kompodium.de/sites/net/index.htm>, Abrufdatum 21.08.2017
- [34] MATHWORKS INC.: *ThingSpeak*. Mathworks Inc., Natick, Massachusetts, USA, 2017. <https://thingspeak.com/>, Abrufdatum 21.05.2017.
- [35] WI-FI ALLIANCE: *The worldwide network of companies that brings you Wi-Fi*. Wi-Fi Alliance, Austin, 2017. <https://www.wi-fi.org/>, Abrufdatum 23.08.2017.
- [36] GLEICH, CLEMENS: *Kommentar: The Internet of Shitty Things*. Heise Online, 2016. <https://www.heise.de/newsticker/meldung/Kommentar-The-Internet-of-Shitty-Things-3492221.html>, Abrufdatum 20.05.2017.
- [37] ESPRESSIF SYSTEMS: *ESP32 Datasheet*. Espressif Systems, Shanghai, 2017. https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf, Abrufdatum 16.04.2017.
- [38] ESPRESSIF SYSTEMS: *ESP32 Datasheet*. Espressif Systems, Shanghai, 2017. https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf, Abrufdatum 16.04.2017.
- [39] SPARKFUN ELECTRONICS: *SparkFun ESP32 Thing*. SparkFun Electronics, Colorado, USA, 2017. <https://www.sparkfun.com/products/13907>, Abrufdatum 20.05.2017.

Anhang A

Eidesstattliche Erklärung

Ich versichere, dass ich diese Arbeit ohne fremde Hilfe selbständig verfasst, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie alle wörtlichen oder sinngemäß übernommenen Stellen in der Arbeit gekennzeichnet habe. Die Arbeit wurde noch keiner Kommission zur Prüfung vorgelegt und verletzt in keiner Weise Rechte Dritter.