



SEMINARARBEIT

Rahmenthema des Wissenschaftspropädeutischen Seminars:

Soft - und Hardwarehacks für den schulischen Alltag

Leitfach: Informatik

Thema der Arbeit:

Bau einer Zeitmessenanlage für Sprintsportarten

Verfasser/in:

Tobias Jungmann

Abgabetermin:

Kursleiter/in:

Sebastian Paintner

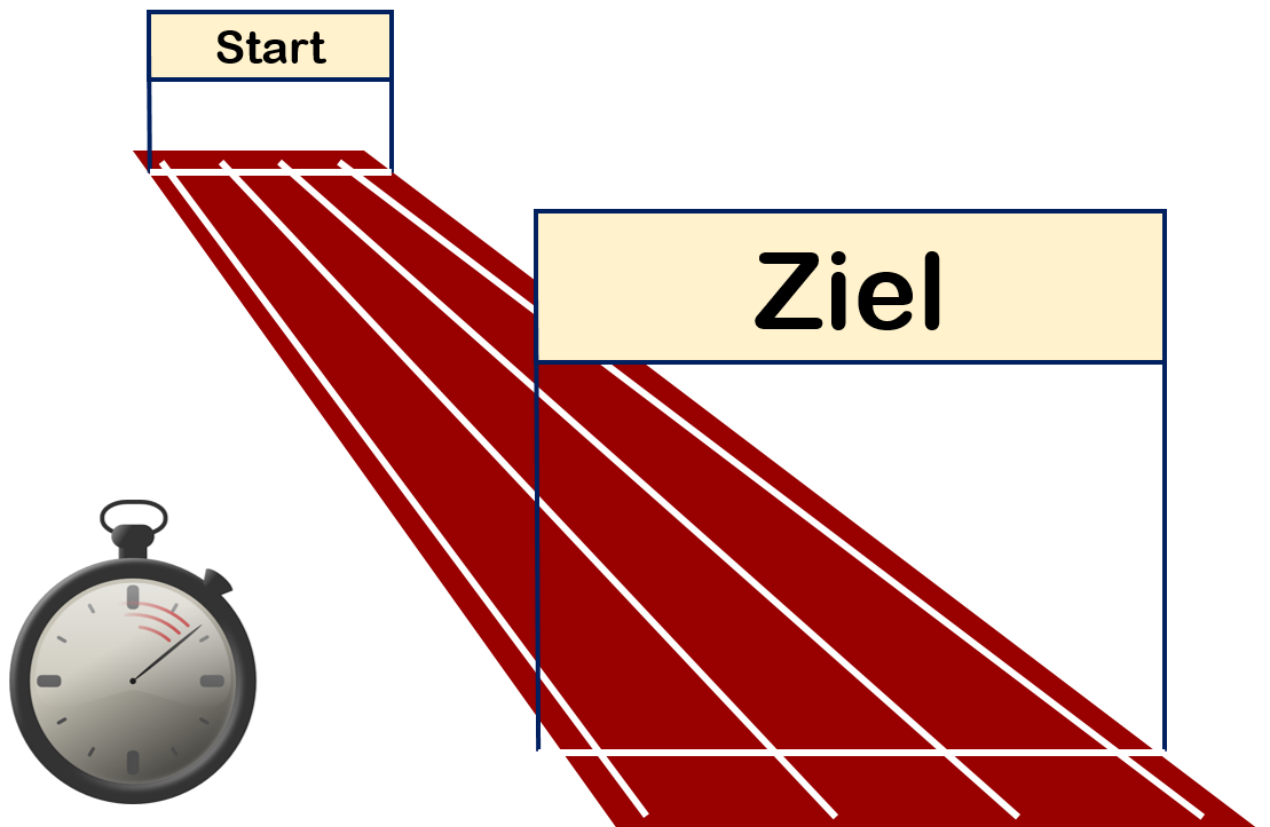
07. November 2017

Bewertung	Note	Notenstufe in Worten	Punkte		Punkte
schriftliche Arbeit				x 3	
Abschlusspräsentation				x 1	
Summe:					
Gesamtleistung nach § 61 (7) GSO = Summe : 2 (gerundet)					

Datum und Unterschrift der Kursleiterin / des Kursleiters

Bau einer Zeitmessaanlage für Sprintsportarten

Tobias Jungmann 12Q3



Inhaltsverzeichnis

1	Notwendigkeit einer Zeitmessanlage für Sprintsportarten	2
2	Anforderungen	2
3	Implementierung.....	3
3.1	System	3
3.1.1	Systemstruktur	3
3.1.2	Bedienung.....	4
3.2	Hardware.....	6
3.2.1	Gemeinsame Eigenschaften beider Geräte	6
3.2.2	Startgerät.....	9
3.2.3	Zielgerät.....	11
3.3	Software	13
3.3.1	Methoden.....	13
3.3.2	Softwaremodule	16
3.3.3	Softwarebibliotheken	18
4	Besondere Herausforderungen	18
4.1	Funkübertragung	18
4.2	Ultraschallmessung	21
4.2.1	Kontinuierliche Messung mit einem Sensor	21
4.2.2	Gleichzeitige Messung auf zwei Bahnen	24
5	Ausblick	26
6	Anhang	27
7	Literaturverzeichnis.....	28
7.1	Bücher	28
7.2	Internet Quellen	28
7.3	Abbildungsverzeichnis.....	30
7.4	Bildquellen.....	30
8	Eidesstattliche Erklärung.....	31

1 Notwendigkeit einer Zeitmessenanlage für Sprintsportarten

Um als Schüler der zwölften Klasse am Gymnasium im 100m Sprint 15 Notenpunkte zu erzielen, muss die vorgegebene Strecke in weniger als 12,3 Sekunden¹ zurückgelegt werden.

Dabei würde die Durchschnittsgeschwindigkeit mindestens 8 Meter pro Sekunde betragen. Das eigentliche Überqueren der Ziellinie geschieht somit innerhalb des Bruchteils einer Sekunde. Durch viele verschiedene Umstände kann die manuell gemessene Zeit verfälscht werden. Beispielsweise wird das Startsignal häufig von Schülern ausgeführt. Dabei kann durch das Zuschlagen der „Startklappe“ vor oder nach dem eigentlichen Startkommando, der genaue Zeitpunkt des Startes nicht ermittelt werden. Beim Zieleinlauf, insbesondere wenn zwei Schüler das Ziel kurz hintereinander überqueren, ergibt sich aus der nahezu gleichzeitigen Bedienung zweier Stoppuhren leicht eine Verfälschung der Messung um eine zehntel Sekunde. Selbst eine so geringe Verfälschung entscheidet, vor allem in den höheren Klassenstufen, schnell über die Note.

Wird für einen Schüler der zwölften Klasse für den 100 Meter Sprint 12,4 Sekunden gemessen, kann dieser durch diese Verfälschungen von +/- einer Zehntelsekunde 15, 14 oder 13 Notenpunkte erzielen.

Die hier vorliegende Arbeit bietet eine Lösung dieses Problems an.

2 Anforderungen

Die Zeitmessenanlage soll die Zeit messen, die zwei Sprinter für eine zuvor festgelegte Strecke benötigen und daraus die passende Note bestimmen können, in Abhängigkeit

¹ Staatsinstitut für Schulqualität und Bildungsforschung: „Durchführung des Sportunterrichts in den Jahrgangsstufen 11 und 12“, in: <https://goo.gl/F6jAt6>, Zugriff am 1.11.2017, S. 10

von der benötigten Zeit, von den aktuellen Leistungsvorgaben im Rahmen des Sportunterrichts sowie von Klassenstufe und Geschlecht.

Die Anlage muss außerdem das Startsignal „Auf die Plätze fertig - Los!“ am Start erklingen lassen und das Überqueren der Ziellinie überwachen. Sie soll nach einer kurzen Einweisung von zwei Personen aufgebaut und bedient werden können.

Nach der Inbetriebnahme können die zur Notenbestimmung nötigen Parameter eingegeben werden.

Bei Ausdauersportarten soll das Gerät als erweiterte Stoppuhr benutzt werden können. Dabei soll es neben der bereits gelaufenen Zeit die dieser Zeit entsprechende Note ausgeben.

Die von mir entwickelte und in dieser Arbeit beschriebene Zeitmessanlage erfüllt alle diese Anforderungen.

3 Implementierung

3.1 System

3.1.1 Systemstruktur

Die Zeitmessanlage besteht aus zwei verschiedenen Geräten, dem Start- und dem Zielgerät. Beide besitzen den gleichen Einplatinen-Computer, die gleiche mobile Stromversorgung und die gleiche Einrichtung zur Datenübertragung. Nur beim Benutzerinterface weisen beide Geräte Unterschiede auf. Das Startgerät besitzt zusätzlich eine Möglichkeit zur Tonausgabe, das Zielgerät Sensoren, um die Ziellinie zu überwachen sowie ein Display für Zeit-/Notenanzeige und Eingaben.

3.1.2 Bedienung

3.1.2.1 Inbetriebnahme

Aktiviert werden die beiden Geräte durch das Anschließen an je eine Akku-Bank. Danach sind sie nach etwa 35 Sekunden arbeitsbereit.

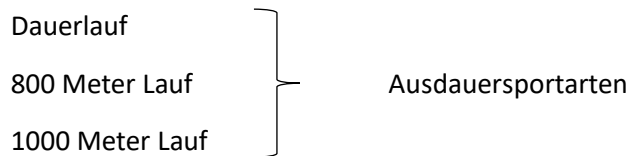
Das Startmodul wird an der Startlinie zwischen den beiden startenden Läufern platziert und kann während der Sprints von dessen Bediener in der Hand gehalten werden. Die Antennen müssen in die dafür vorgesehenen Steckplätze eingesteckt werden, wobei auf die farbigen Markierungen zu achten ist.

Das Zielgerät wird einen halben Meter hinter der Ziellinie und parallel zu dieser platziert und durch Einstecken der Antennen sowie Ausklappen und Ausrichten beider Sensorhalterungen aufgebaut. Es muss auf einen Sockel gestellt werden, beispielsweise auf einen Tisch (Höhe ca. 1 Meter), um sicherzustellen, dass die Sensoren bei allen Schülern mindestens die Höhe des Unterleibes erreichen. Innerhalb der drei Meter langen, geradlinig verlaufenden Messbereiche der Sensoren darf sich kein Hindernis, wie beispielsweise vorbeilaufende Schüler oder ein Gebäude, befinden.

3.1.2.2 Durchführung einer Messung

Am Zielgerät müssen zu Beginn Klassenstufe, Geschlecht und Sportart eingegeben werden. Die Eingaben werden durch die drei Taster unterhalb des LCD-Displays getätigt. Dabei geht der rechte Taster zur nächsten Auswahlmöglichkeit über, während sich mit dem mittleren sowie dem linken Taster die jeweiligen Werte verstellen lassen (links -; rechts +). Auswählbar sind, je nach Klassenstufe und Geschlecht, die folgenden Sportarten:

50 Meter Sprint	}	Sprintsportarten
75 Meter Sprint		
100 Meter Sprint		
110 Meter Hürden		
100 Meter Hürden		



Nach dem Abschluss der Einstellungen beginnt eine der beiden im Folgenden beschriebenen, beliebig oft wiederholbaren Sequenzen. Dabei kann nach Ablauf jeder vollständigen Messung durch Drücken des linken Tasters in das Einstellungs Menü zurückgekehrt werden.

Bei Sprintsportarten (Sequenz 1) wird dem Startgerät durch Drücken des rechten Tasters signalisiert, dass ein neuer Lauf beginnen kann. Dem Helfer am Start wird dieser Zustand durch eine grün leuchtende LED signalisiert. Dieser löst durch Drücken des grünen Tasters das Signal "Auf die Plätze, fertig - Los!" aus und startet damit den Lauf. Beide Läufer sprinten daraufhin in ihren Bahnen die zu absolvierende Strecke. An der Ziellinie muss je einer der Läufer auf der Bahn rechts und links neben dem Zielgerät die Ziellinie überqueren. Nach dem Zieleinlauf werden die benötigte Zeit und die entsprechende Note für beide Bahnen ausgegeben. Nachdem beide Läufer das Ziel durchquert haben und die Lehrkraft die Daten übernommen hat, kann ein neuer Lauf gestartet werden. Sollte ein Läufer das Ziel nicht erreichen, kann durch Bedecken des Sensors die Zeitmessung angehalten werden.

Bei Ausdauersportarten (Sequenz 2) wird nur das Zielgerät benötigt. Außerdem spielt die Ausrichtung des Zielgerätes keine Rolle, die Sensorhalterungen können eingeklappt bleiben und die Antennen müssen nicht angesteckt werden (siehe 3.1.2.1 Inbetriebnahme). Der Lauf wird durch ein entsprechendes Signal der Lehrkraft gestartet und gleichzeitig die Zeitmessung durch Drücken des rechten Tasters am Zielgerät ausgelöst. Am Display erscheint neben der bereits gelaufenen Zeit die entsprechende Note. Ein Druck auf mittleren Taster stoppt die Zeitanzeige für die Dauer des Tastendruckes, nicht aber die Zeitmessung.

3.1.2.3 Abschalten

Beide Geräte sind mit einem Taster zwischen den Antennen ausgestattet. Durch längeres Drücken dieses Tasters wird das jeweilige Gerät heruntergefahren. Das Zielgerät erinnert auf seinem LCD-Display auch nach dem Herunterfahren daran, die Stromversorgungen abzustecken.



Abbildung 1 Taster "Herunterfahren"

3.2 Hardware

3.2.1 Gemeinsame Eigenschaften beider Geräte

3.2.1.1 Rechner

Beide Geräte enthalten jeweils einen Raspberry Pi. Ursprünglich waren zwei Model B+ ausgewählt, wegen eines Defekts wurde der im Startgerät verbaute später durch einen Raspberry Pi 3 Model B ersetzt.

Für einen Raspberry Pi habe ich mich entschieden, weil er:

1. Die für die Zielüberwachung notwendige Rechenleistung aufweist.
2. Die für die Hardware essenziellen GPIOs aufweist.
3. Einen 3,5mm Klinkenstecker sowie Audiointerface zum Verwenden eines Lautsprechers aufweist.
4. Mittels einer mobilen Spannungsversorgung betrieben werden kann.
5. Mindestens eine USB Anschlussmöglichkeit bietet.
6. Außerdem habe ich mit einem solchen bereits vorher erste Erfahrungen gesammelt.

Die GPIOs des Startgerätes werden mithilfe eines Schneidklemmsteckers mit der Platine verbunden. Beim Zielgerät werden mehr Kabel benötigt. Deshalb findet dafür ein 80-poliges IDE Kabel Verwendung, um Kurzschlüsse zu vermeiden, die sonst bei der Nutzung eines Schneidklemmsteckers mit dieser hohen Kabelmenge entstehen würden.

Das IDE-Kabel besitzt den gleichen Stecker. Problematisch ist jedoch, dass es in IDE Kabeln interne Verbindungen gibt. Dadurch auftretende Kurzschlüsse lassen sich durch Entfernen der entsprechenden GPIOs am Raspberry Pi vermeiden.²

3.2.1.2 Funkübertragung

Die Signale zwischen den beiden Geräten werden per Funk übertragen. Als Empfängermodul wird das MX-05V, als Sendemodul das MX-FS-03V verwendet. Die Übertragung erfolgt digital auf dem in Deutschland frei verwendbaren 433MHz Band.³

Die für die Funkstrecke verwendeten Antennen wurden aus Kostengründen selbst gebaut. Der Bau der Antennen wird in Kapitel 4.1 Funkübertragung näher erläutert. Die Antennen sind durch eine Steckverbindung abnehmbar.

Dazu besitzt jede Antenne einen Koaxstecker und eine farbliche Markierung. Diese Markierung findet sich auch in den dafür vorgesehenen Löchern im Gehäuse des jeweiligen Gerätes wieder. Eine Vertauschung der Antennen ist durch die Kodierung der Stecker ausgeschlossen.



Abbildung 2 Holzsockel mit Koaxsteckern

² Wikimedia Foundation Inc.: „ATA/ATAPI - Steckerbelegung“, in: <https://de.wikipedia.org/wiki/ATA/ATAPI>, Zugriff am 1.11.2017

³ Bundesnetzagentur: „Frequenzplan – Amateurfunk“, in: <https://goo.gl/Koaetv>, Zugriff am 28.09.2017, S.272

Beide Geräte besitzen innerhalb des Gehäuses einen Holzsockel auf den zwei weitere Koaxstecker geschraubt sind, sodass diese in gerader Linie unter den Löchern im Gehäuse liegen. Zwei kleine und ein etwas größerer T-förmiger Einschnitt ermöglichen es, die Masse und die eigentliche Antenne an den dafür vorgesehenen Stellen anzubringen.

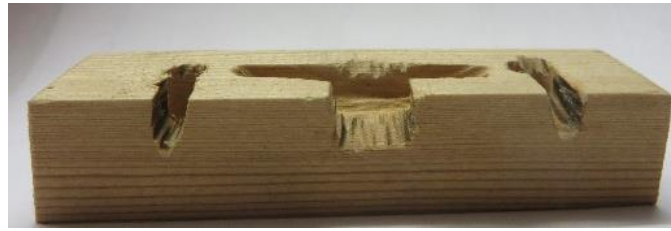


Abbildung 3 Holzsockel

3.2.1.3 Stromversorgung

Beide Geräte werden durch je eine „EasyAcc Smart 10000mAh Powerbank“ mit zwei USB A-Ausgängen betrieben. Beide USB-Ausgänge liefern jeweils 5V.⁴ Die Powerbank, die durch Einstecken eines Verbrauchers aktiviert wird, zeigt den aktuellen Ladestand mithilfe von vier kleinen LEDs an. Beide Akkubanks werden per Klettverschluss am Gehäuse befestigt und liefern ausreichend Strom, um beide Geräte einen Schultag lang nutzen zu können. Das Aufladen der Akkus erfolgt durch ein gewöhnliches Handy Ladegerät durch den Micro USB-Eingang auf einer der Seitenflächen.

⁴ easyacc.com, Inc.: „Kaufangebot mit Information über die Akkubank“, in: <http://bit.ly/2h68bNV>, Zugriff am 1.11.2017

3.2.2 Startgerät

3.2.2.1 Benutzerinterface

Das Benutzerinterface besteht aus zwei Tastern, wovon einer, vor versehentlichem Drücken geschützt, zwischen den Antennen liegt.

Zwei tageslichttaugliche LEDs, eine rote (21000 mcd)⁵ und eine grüne (37500 mcd)⁶, die auch bei starker Sonneneinstrahlung durch den farblosen Kopf sehr gut ablesbar sind, signalisieren dem Helfer am Start, ob ein Lauf gestartet werden darf (grün) oder nicht (rot).



Abbildung 4 Benutzerinterface Startgerät



Abbildung 5 Benutzerinterface Rote LED

3.2.2.2 Tonausgabe

Das Startsignal ertönt über einen „Speedlink Twoxo Stereo“ Lautsprecher. Dieser erhält ein Aux-Signal des Raspberry Pi über den 3,5mm Klinkenstecker. Da die benötigte Versorgungsspannung genau 5V⁷ beträgt, erfolgt die Stromzufuhr durch den zweiten USB A Steckplatz an der Powerbank.

⁵ Conrad Electronic SE(Hrsg.): „Katalogeintrag rote LED“, in: <https://goo.gl/wWP7Mq>, Zugriff am 24.09.2017

⁶ Conrad Electronic SE(Hrsg.): „Katalogeintrag grüne LED“, in: <https://goo.gl/bhb68X>, Zugriff am 24.09.2017

⁷ Amazon.com Inc.: „Kaufangebot mit Informationen über Lautsprecher“, in: <https://goo.gl/M9dFEZ>, Zugriff am 28.07.2017

3.2.2.3 Gehäuse

Alle vier verwendeten Gehäuse bestehen aus Kunststoff, da diese deutlich leichter als vergleichbare Gehäuse aus Holz sind.

Das Gehäuse des Startmoduls ist relativ kompakt. Deshalb sind die Lautsprecher aus Platzgründen außerhalb auf einer der Seitenwände des 16 x 10 x 6 cm großen Plastikgehäuses mit Klettverschlüssen befestigt. Die Powerbank befindet sich auf der Unterseite des Gehäuses. Damit kann diese geladen sowie an- und abgesteckt werden, ohne die Box aufschrauben zu müssen. Die Steckplätze für die Antennen befinden sich samt allen Bedienelementen auf der Oberseite. Sowohl der 9V-Block als auch das Relais sind mit Isolierband umwickelt um Kurzschlüsse zu Verhindern.



Abbildung 6 Startgerät offen



Abbildung 7 Startgerät

Im Inneren des Gehäuses befindet sich auf dem mit vier Schrauben befestigten Boden, auf einer Seite der Raspberry Pi, in der Mitte eine kleine Platine, beide auf zwei kleinen Leisten aufgeschraubt und auf der anderen Seite der in Kapitel 3.2.1.2 Funkübertragung beschriebene Sockel samt Steckern.

3.2.3 Zielgerät

3.2.3.1 Benutzerinterface

Alle Ausgaben erfolgen über ein LCD-Display mit zwei Zeilen, die jeweils 16 Zeichen anzeigen können. Für Eingaben durch den Benutzer stehen drei Taster zur Verfügung, die je nach Situation unterschiedliche Funktionen besitzen. Ein weiterer, gegen



Abbildung 8 Benutzerinterface Zielgerät

gesicherter Taster ermöglicht, das Zielgerät herunter zu fahren.

3.2.3.2 Sensorik

Zu beiden Seiten des Gehäuses befindet sich je ein HC-SR04 Ultraschallsensor. Diese ermöglichen das Messen der Entfernung ausgehend von den Sensoren bis zum nächsten Hindernis innerhalb einer Entfernung von 5cm bis 300cm⁸. Beide Sensoren besitzen eine eigene kleine Platine. Eine ihrer Funktionen ist es, mithilfe eines NPN Transistors die 3,3V des „Trigger“-Signales des GPIOs auf die am „trig“-Eingang des Sensors notwendige Spannung von 5V zu erhöhen. Platine und Sensor befinden sich in je einem 11,2 x 6,3 x 3,2cm großen Kunststoffgehäuse, wobei nur der Sender und der Empfänger der Ultraschallsensoren heraus schauen.



Abbildung 9 HC-SR04

Banggood Ltd: „Verkaufsangebot HC-SR04“, in: <https://goo.gl/Z2eKYC>, Zugriff am 1.11.2017

⁸ KT-Elektronik: „Datenblatt HC SR 04“, in: <https://goo.gl/6PvZQf>, Zugriff am 26.09.2017, S.1

Eine mögliche Alternative würde eine Lichtschranke darstellen. Diese benötigt aber einen zusätzlichen Reflektor auf der anderen Seite der Bahn, wodurch zum einen Aufbau und Transport der Anlage erschwert würde, zum anderen die Gefahr des Umkippens und des Umstoßens durch Schüler bestünde.

3.2.3.3 Gehäuse

Ein $40 \times 30 \times 17\text{cm}$ großer Sicherungskasten bietet aufgrund seiner großen Grundfläche einen stabilen und damit sicheren Stand. Die beiden Ultraschallsensoren sind im vorderen Bereich des Deckels an je einem circa 24cm langen, schwenkbaren Aluminiumrohr angeschraubt. Dadurch wird eine deutlich platzsparendere Lagerung der Zeitmessanlage möglich. Zusätzlich kann das deutlich kompaktere Startgerät in dem Zielgerät gelagert werden. Die Elemente des Benutzerinterfaces, die Platine und der Raspberry Pi sind am Deckel befestigt. Damit werden beim Öffnen keine Kabel abgerissen. Die Platine sowie der Raspberry Pi sind auf an die Deckelunterseite geklebte



Abbildung 10 Zielgerät Transport/Lagerung

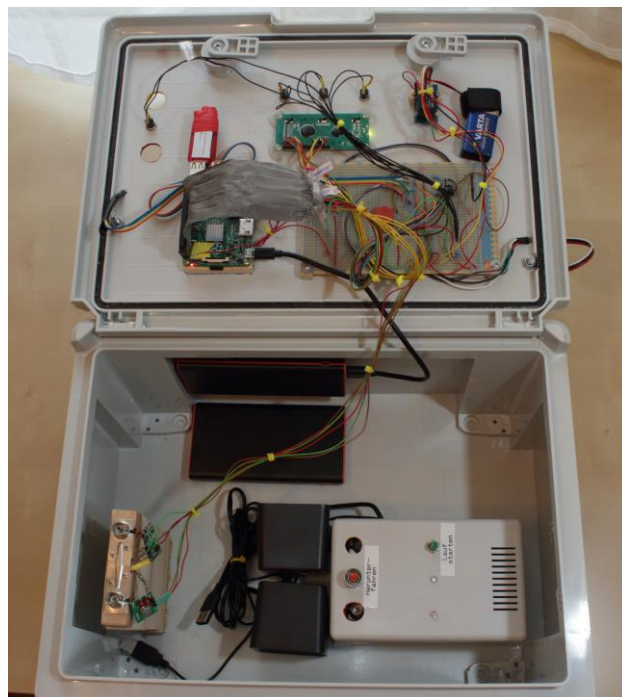


Abbildung 11 Lagerung Startgerät

Holzleisten geschraubt. An der Vorderseite des Zielgerätes befinden sich acht ISO-Rohr-Klemmen, die die Lagerung aller vier Antennen der beiden Geräte ermöglichen.

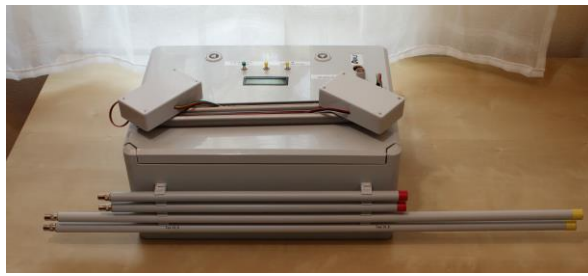


Abbildung 12 Lagerung der Antennen

3.3 Software

Die im Start- und Zielgerät verwendeten Programme sind in C-Code geschrieben. Diese Sprache wurde aus den folgenden Gründen gewählt:

1. Ausgereifte Compiler sind verfügbar und erlauben schnelle und effiziente Programmdurchläufe.
2. Eine Bibliothek zur Ansteuerung der GPIOs ist verfügbar.
3. Ich habe mit dieser Sprache bereits etwas Erfahrung.

3.3.1 Methoden

Nach dem Starten des Programmes wird immer genau eine Methode an einem Gerät abgearbeitet. Begonnen wird bei beiden Geräten mit der Methode „init()“. Diese führt alle, für das jeweilige Gerät notwendigen Initialisierungen durch.

Für das Zielgerät zusätzlich verwendete Methoden:

Für die Eingabe der Einstellungen dient die Methode „einstellungen()“, die zunächst die vier folgenden Methoden aufruft

- „klassenstufeEingeben()“
- „geschlechtEingeben()“
- „sportartEingeben()“
- „auslesen()“

Nach Rückkehr in die Methode „einstellungen()“ ruft diese, je nach ausgewählter Sportart, eine der beiden Methoden auf.

- „sprintsequenz()“
 - „senden()“
 - „empfangen()“
 - „zielkontrolle()“
 - „stateMachine()“
- „laufsequenz()“

Für Startgerät zusätzlich verwendete Methoden

- „statusROT()“
- „statusGRUEN()“
- „statusAus()“

Gemeinsame Methoden beider Geräte:

- „senden()“
- „empfangen()“
- „herunterfahren()“

Im Folgenden wird das Zusammenspiel und die Funktionsweise der aufgelisteten Methoden erläutert.

Die Methode „herunterfahren()“ wird in beiden Geräten in verschiedenen Zuständen aufgerufen. Sie sorgt dafür, dass sich das jeweilige Gerät nach einem einsekündigen Druck auf den Taster zwischen den Antennen abschaltet.

Die erste Methode die nach der Initialisierung am Zielgerät aufgerufen wird ist: „einstellungen()“. Diese wartet bis der rechte Taster gedrückt wurde und ruft dann nacheinander die vier weiteren Methoden auf.

„KlassenstufeEingeben()“, „geschlechtEingeben()“ und „SportartEingeben()“ warten zunächst, bis der rechte Taster nicht mehr gedrückt wird, woraufhin alle drei eine Schleife solange wiederholen, bis der rechte Taster erneut gedrückt wird. Innerhalb der Schleife wird geprüft, ob einer der Taster gedrückt wird und dementsprechend die ausgewählte Möglichkeit in der unteren Zeile des LCD Displays verändert.

In der letzten der drei Methoden, „SportartEingeben()“, werden unter Zuhilfenahme der Klassenstufe und des Geschlechts die möglichen Sportarten bestimmt und zur Auswahl vorgeschlagen.

Die vierte Methode „auslesen()“ liest die Datei „daten.txt“ mit den Notenzuordnungen Zeile für Zeile ein, bis die ersten drei Ziffern einer Zeile mit denen der eben getätigten Einstellungen übereinstimmen. Die Zeilen

```
ascwert=werte[i];
cwert[0]=(char)ascwert;
zahlwerte[i]=atoi(cwert);
```

speichern den aus „daten.txt“ eingelesenen Ascii-Wert zunächst als Integer. Durch die beiden folgenden „Type-Casts“ wird die Zahl zunächst in den, dem Ascii-Code entsprechenden, Character-Wert, dann in den passenden Integer-Wert umgewandelt. Würde „werte[i]“ direkt mit „atoi“ in „zahlenwerte[i]“ konvertiert werden, wird beispielsweise eine in „daten.txt“ gespeicherte 1 als 49 in „zahlenwerte[i]“ gespeichert.

Danach werden die restlichen Ziffern, immer vier am Stück, zu einer vierstelligen Zahl konvertiert und in dem Feld „Notenwerte“ gespeichert. Der Inhalt dieses Feldes wird mit einem Faktor multipliziert, um ihn in Mikrosekunden anzugeben.

Nachdem alle vier Methoden ausgeführt sind, prüft die Methode „einstellungen()“, ob es sich bei der ausgewählten Sportart um eine Sprintsportart handelt, in diesem Fall wird Methode „sprintsequenz()“ aufgerufen, anderenfalls die Methode „laufsequenz()“.

Die Methode „sprintsequenz()“ ist nur in Verbindung mit dem Startgerät zu verwenden und ruft nacheinander die benötigten Methoden auf. Im Folgenden wird der Ablauf den „sprintsequenz()“ ermöglicht beschrieben. Die Funktionsweise der einzelnen Methoden wird in Kapitel 4 Besondere Herausforderungen erläutert. Das Zielgerät wartet zu Beginn der Methode auf einen Druck des rechten Tasters. Daraufhin teilt es dem Startgerät durch ein Signal die Startbereitschaft mit. Selber wechselt das Zielgerät nun in die Empfangsbereitschaft und wartet, bis das Startgerät das „Startsignal“ sendet. Erst danach wechselt das Startgerät auf den Zustand „Startbereit“, wobei die grüne LED aufleuchtet, die rote erlischt. Jetzt kann durch Drücken des Tasters am Startgerät der Start ausgelöst werden.

Nach dem Drücken des Tasters am Startgerät wechseln die LEDs wieder in ihre Ausgangszustände, das „Startsignal“ wird gesendet und das Startkommando abgespielt. Danach wechselt das Startgerät automatisch in den anfänglichen Zustand. Das Zielgerät hingegen wechselt nach dem Empfangen des „Startsignales“ in den Zustand der Zielüberwachung. Dabei überwachen die Ultraschallsensoren die Ziellinie. Sobald beide Sensoren einen Zieleinlauf registriert haben, werden die gemessenen Zeiten ausgewertet und anschließend auf dem LCD-Display ausgegeben. Daraufhin kann durch Drücken des rechten Tasters in den anfänglichen Zustand gewechselt und ein neues „Startbereit“-Signal gesendet werden.

„laufsequenz()“ zeigt die Zeit kontinuierlich an, bis die Zeit für die bestmögliche Note erreicht ist. Mithilfe des Rechenoperators Modulo wird die Zeitspanne von Mikrosekunden in Minuten, Sekunden und Zehntelsekunden umgerechnet. Der Laufzeit entsprechend wird die Note, neben dieser, auf dem LCD-Display ausgegeben. Ein Druck auf den mittleren Taster lässt die Zeitanzeige für die Zeitspanne des Druckes pausieren. Die eigentliche Zeitmessung wird dabei nicht unterbrochen.

Bei der Auswertung wird die gemessene Zeit mit den Werten des Feldes „Notenwerte“ verglichen. Je nachdem, an welcher Position des Feldes die benötigte Zeit erstmals größer ist als der Wert des Feldinhalte, wird die entsprechende Note ausgegeben.

3.3.2 Softwaremodule

Um auf die GPIOs des Raspberry Pis zuzugreifen, wird die Open Source Bibliothek „wiringPi“ von Gordon Henderson⁹ verwendet. Diese Bibliothek bietet weiterhin die Methoden zum Pausieren eines Programms für eine im Mikrosekundenbereich wählbare, beliebige Zeit sowie um die seit dem Start eines Programmes vergangene Zeit zu bestimmen.¹⁰

⁹ Henderson, G.: „wiringPi“, in: <https://projects.drogon.net/contact/>, Zugriff am 24.9.2017

¹⁰ Henderson, G.: „wiringPi Funktionen“, in: <http://bit.ly/2A0YofT>, Zugriff am 24.09.2017

Für die Ausgaben auf dem LCD-Display wird neben „wiringPi“ auch die Bibliothek „lcd“ benötigt. Um diese verwenden zu können, muss auch das gleichnamige h-File ergänzt werden.¹¹

Ein USB-Stick im Raspberry Pi des Zielgerätes enthält eine Datei namens „daten.txt“ die alle für die Notenvergabe relevanten Zeiten, abhängig von Klassenstufe, Geschlecht und Sportart enthält. Auf dieses File wird mit den Standard C-Bibliotheken „stdio“ und „stdlib“ zugegriffen. Diese ermöglichen unter anderem, eine Datei zu öffnen, zu schließen, Stück für Stück einzulesen und Variablen von einem in einen anderen Datentyp umzuwandeln.¹²

Um das selbst aufgenommene Startsignal, welches sich als „mp3“-Datei gespeichert ist, abspielen zu können wird ein im Internet frei verfügbarer Codeausschnitt verwendet.¹³

Für beide Geräte existiert jeweils ein Makefile¹⁴, welches mithilfe des GNU Compilers den Code des jeweiligen Gerätes sowie der passenden selbstgeschriebenen Bibliotheken kompiliert und linkt.¹⁵

Damit das jeweilige Programm automatisch beim Hochfahren des Raspberry Pi startet, wird das mit „sudo nano /etc/rc.local“ im Terminal des Raspberry Pis aufrufbare „rc.local“-File verwendet, welches bei jedem Start erneut durchlaufen wird.¹⁶

¹¹ Campbell, S.: „How to Setup an LCD on the Raspberry Pi and Program it With C“, in: <https://goo.gl/WpbtRP>, Zugriff am 31.10.2017

¹² Erlenkötter, H., Grundkurs Computerpraxis C Programmieren von Anfang an, 23. Auflage Februar 2016, S.165ff

¹³ Huang, J.: „Play MP3 with libmpg123 and libao“, in: <http://bit.ly/2gWMDj0>, Zugriff am 31.10.2017

¹⁴ Becker, M.: „Eine Einführung in Makefiles“, in: <http://bit.ly/2xNkOQU>, Zugriff am 1.11.2017

¹⁵ Free Software Foundation, Inc.: „GCC, the GNU Compiler Collection“, in: <https://gcc.gnu.org/>, Zugriff am 1.11.2017

¹⁶ Schweizer, P.: „Autostart mit rc.local“, in: <https://goo.gl/HhfnMf>, Zugriff am 1.11.2017

3.3.3 Softwarebibliotheken

Um das Programm übersichtlicher zu gestalten, sind die meisten verwendeten Methoden in thematisch sortierten Bibliotheken zusammengefasst.

- Die Bibliothek „Einstellungen“ enthält die fünf dafür notwendigen Methoden.
- Die Bibliothek „433MhzGrundlagen“ enthält alle für die Funkverbindung relevanten Methoden.
- Die Bibliothek „Startsignal“ enthält die gleichnamige Methode.
- Die Bibliothek „Daten“ enthält keine Methoden, stellt aber globale Variablen und Konstanten bereit, auf die aus mehreren Bibliotheken zugegriffen werden kann.
- Die Bibliothek „StatusLED“ enthält Methoden, um die beiden LEDs an und aus zu schalten.
- Die Bibliothek „herunterfahren“ stellt die entsprechende Methode bereit.

Zu jeder Bibliothek existiert jeweils ein gleichnamiges „.c“ und „.h“ File. Dabei enthält das „.c“-File den Code der Methoden. Das „.h“-File ermöglicht den Zugriff anderer Programme auf den Code dieser Bibliothek.

4 Besondere Herausforderungen

4.1 Funkübertragung

Die Reichweite der Funkmodule ist mit 200 Metern angegeben.¹⁷ Dieser Wert ist nur unter optimalen Bedingungen, wie beispielsweise einer hohen Betriebsspannung und professionellen Antennen zu realisieren. Mit einfachen Antennen und mit einer Versorgungsspannung von 3,3V erreichen die beiden Module lediglich eine Reichweite von etwa einem Meter. Ein zur Optimierung der Reichweite gebauter Prototyp sendet

¹⁷ Amazon.com, Inc.: „Kaufangebot mit Informationen der Sende- und Empfangseinheit“, in: <https://goo.gl/Q12P3z>, Zugriff am 28.9.2017

ein 10 Millisekunden langes Signal, welches nach einer ebenso langen Pause immer wieder wiederholt wird.

Die Methode „empfangen()“ wartet, bis es von dem „Data“-Pin des Empfängers ein „HIGH“-Signal erhält. Daraufhin misst sie die Zeit, die vergeht, bis ein „LOW“-Signal anliegt. Eine Antenne aus einfachem Draht mit einer Länge von 17cm¹⁸, diese entspricht bei einer Frequenz von 433Mhz $\lambda/4$, erhöht die Reichweite bereits auf wenige Meter, das Signal bleibt aber immer noch sehr unzuverlässig. Nach weiteren Tests fiel die endgültige Wahl auf selbst gebaute Dipol Antennen. Diese werden aus einem Koaxkabel gebaut. Für eine Antenne eines Empfängers werden 17cm des Kabels abisoliert und die unter der Isolationsschicht liegende Abschirmung wird nach hinten über das restliche Kabel gestülpt. Der vordere, abisolierte und nicht mehr abgeschirmte Teil stellt die eigentliche Antenne dar, der hintere Teil den Gegenpol.¹⁹ Die Antennen der

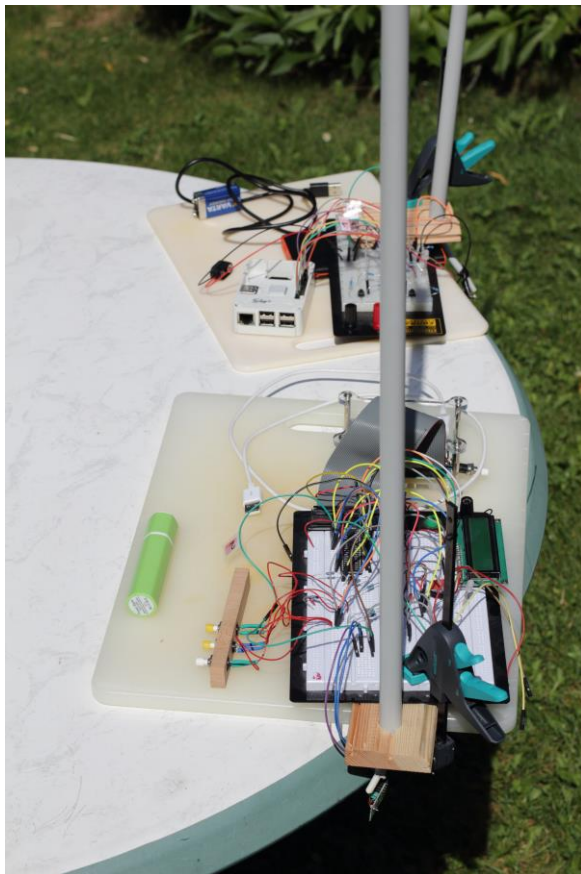


Abbildung 14 Prototypen für Reichweitentests



Abbildung 13 Dipol Antennen

¹⁸ Schwarz, A.: „433Mhz Funkübertragung – Bau einer Dipol Antenne“, in:

https://www.mikrocontroller.net/articles/433_MHz_Funk%C3%BCbertragung, Zugriff am 28.09.2017

¹⁹ ebd.

Sender werden nach dem gleichen Prinzip gebaut, der abisolierte Teil hat jedoch eine Länge von $\lambda/2$ (34cm). Damit wird die Sendeleistung größer.

Der Empfänger behält die kleinere Variante, da dieser sonst zu viele Störsignale empfängt. Die daraus resultierende Länge der Antennen macht die Geräte unhandlich, deshalb sind die Antennen durch Koaxsteckverbinder abnehmbar.

Dadurch kann die Reichweite der Funkverbindung auf circa 50 Meter gesteigert werden. Durch Erhöhen der Versorgungsspannung des Senders auf

9V wird eine Reichweite von 100 Metern möglich. Um den 9V-Block zu schonen, wird die Versorgungsspannung nur für die Zeit des Sendens durch ein Relais angelegt.

Da die kurzen Signale anfälliger für Störungen sind, werden in der finalen Version Signale mit einer Länge von einer Sekunde gesendet. Damit in den endgültigen Geräten Fehlmessungen ausgeschlossen werden können, führt das Programm jeweils zehn Messungen direkt hintereinander aus. Sollte dabei mehr als eine Messung ein „HIGH“-Signal des Empfängers bekommen, gilt die Messung als empfangen. Diese Messungen laufen kontinuierlich ab. Die Zeitspanne, in der der Empfänger ein „HIGH“-Signal erkennt, wird gemessen. Wenn die gemessene Länge des erkannten Signals zwischen 1.050.000 und 805.000 Mikrosekunden liegt, wurde ein entsprechendes Signal des jeweils anderen Gerätes erhalten und diese Funktion beendet. Wenn der Sender ein einzelnes Signal sendet, verfälscht sich dessen Länge teils massiv. Durch ein kurzes Signal vor dem eigentlichen Signal kann dieses Problem behoben werden.

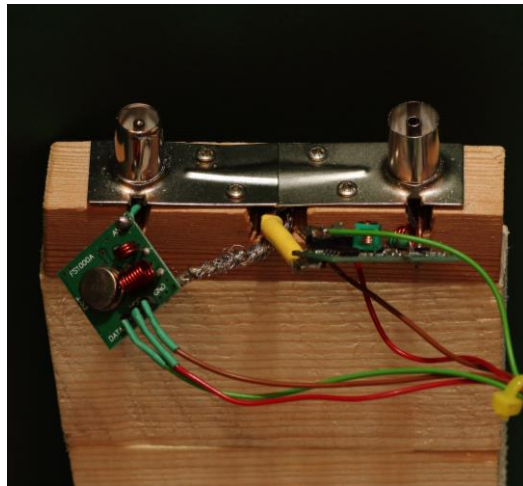


Abbildung 15 Sender und Empfänger mit Koaxsteckern

4.2 Ultraschallmessung

4.2.1 Kontinuierliche Messung mit einem Sensor

Die Funktionsweise des Ultraschallsensors HC-SR04, zum Messen von Entfernungen, wird in dessen Datenblatt wie folgt beschrieben:

„Das Auslösen eines Messzyklus geschieht durch eine fallende Flanke am Triggereingang (...) für mindestens 10µs. Das Modul sendet darauf nach ca. 250µs ein 40 kHz Burst-Signal für die Dauer von 200µs. Danach geht der Ausgang (...) sofort auf H[IGH] -Pegel und das Modul wartet auf den Empfang des Echos. Wird dieses detektiert fällt der Ausgang auf L[OW] -Pegel. 20ms nach Triggierung kann eine weitere Messung stattfinden. Wird kein Echo detektiert verweilt der Ausgang für insgesamt 200ms auf H[IGH]-Pegel und zeigt so die erfolglose Messung an. Danach wartet das Modul auf die nächste fallende Flanke am Triggereingang und die Messung beginnt neu.“²⁰

Durch dieses Prinzip ist es leicht möglich, eine Entfernung zentimetergenau zu bestimmen. Vergleichbare Sensoren werden beispielsweise bei Einparkhilfen verwendet. Implementiert werden kann dieses Prinzip durch zwei „while“-Schleifen. Die erste wartet, bis am „echo“-Pin ein „HIGH“-Signal anliegt, die Zweite wartet, bis dieses wieder auf „LOW“ wechselt. Im Anschluss kann aus der Dauer des „HIGH“-Signals die Entfernung bis zum nächsten Hindernisses bestimmt werden. Dieses Prinzip wird so lange wiederholt, bis zwei aufeinander folgende Messungen ergeben, dass das nächste Hindernis weniger als zwei Meter weit entfernt ist. Sollte dies der Fall sein, brechen die Wiederholung der Messung sowie das eigentliche Stoppen der Zeit ab. Im Anschluss wird die Note, wie im Kapitel 3.3.2 Softwaremodule beschrieben, bestimmt.

Auf die im weiteren Verlauf des Datenblattes beschriebene Kompensation der Temperatur kann bei dieser Anwendung verzichtet werden, da die dadurch mögliche Zunahme der Genauigkeit keinen Einfluss auf die Messung hätte.

Sollte der Sensor kein Echo empfangen, wartet er, wie beschrieben, 200ms um diesen Zustand zu signalisieren.²¹ Dadurch sind ohne ein Hindernis etwa vier bis fünf

²⁰ KT-Elektronic: „Datenblatt HC SR 04“, in: <https://goo.gl/6PvZQf>, Zugriff am 26.09.2017, S.1

²¹ ebd.

Messungen pro Sekunde möglich. Damit ein Läufer sicher erkannt wird, sind jedoch mindestens 48 Messungen pro Sekunde nötig. Diese Zahl ergibt sich, da ein sprintender Schüler mit einer Maximalgeschwindigkeit von 12m/s den an der dünnsten Stelle 50cm breiten Messkorridor durchläuft und von den Sensoren erkannt werden muss.

Demzufolge muss der Sensor innerhalb von 1/24 Sekunden einen Läufer erkennen. Um sicherzustellen, dass es sich um einen das Ziel durchquerenden Läufer handelt und nicht um eine Fehlmessung, sind zwei Messungen hintereinander notwendig.

Damit das Programm fast zehn Mal so schnell arbeiten kann, enthält es innerhalb der zweiten „while“-Schleife eine Abbruchbedingung, die ab einer Entfernung von mehr als zwei Metern die Messung automatisch beendet und eine neue Messung startet.

Während das Programm auf das „HIGH“-Signal wartet wird dafür eine Variable hochgezählt, anhand deren Größe im Anschluss entschieden wird, ob die Messung abgebrochen werden muss.

Die Abbruchbedingung ermöglicht die notwendige Anzahl an Messungen. Sollte die Anforderung, dass sich kein Hindernis hinter dem Messbereich befinden darf, nicht erfüllt sein, können Reflexionen einer abgebrochenen Messung nach dem „triggern“ einer darauf folgenden neuen Messung auf den Sensor treffen. Dadurch würde diese Messung verfälscht werden.

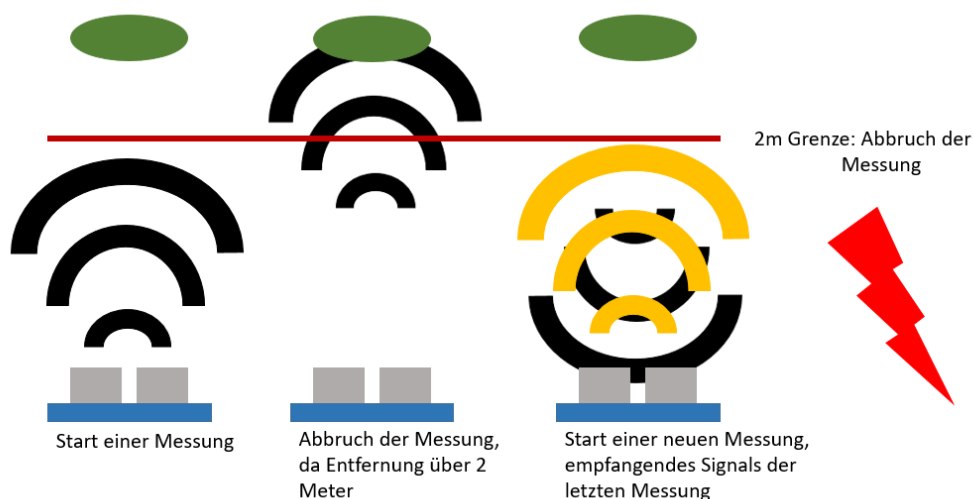


Abbildung 16 Grafik Ultraschallsensor Messfehler

Dieser Effekt entsteht, da die Schallwellen der vorherigen Messung (schwarz), nach Beginn der nächsten Messung (gelb), den Empfänger erreichen. Die aktuelle, fehlerhafte Messung erkennt diese als ihr eigenes Signal. Falls die Situation am Ziel nicht den nötigen Platz bietet, können alternativ schallschluckende Schaumstoffmatten eingesetzt werden.



Abbildung 18 Schallschluckende Schaumstoffmatten

Final sind beide Sensoren in einem 55° Winkel zur Basis angebracht, sodass sie nicht zur Seite, sondern schräg nach vorne messen, wodurch der 15° große Öffnungswinkel des Messbereiches optimal für die 1,22 Meter breite Bahn²² ausgenutzt wird. Dadurch entsteht ein Optimum aus nötiger Messentfernung und entsprechendem Querschnitt, den ein Läufer minimal durchlaufen kann.

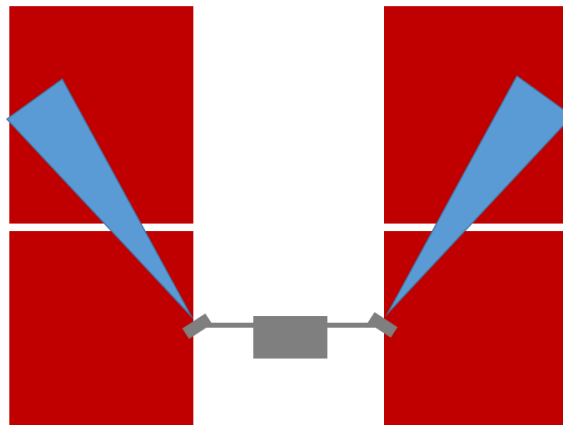
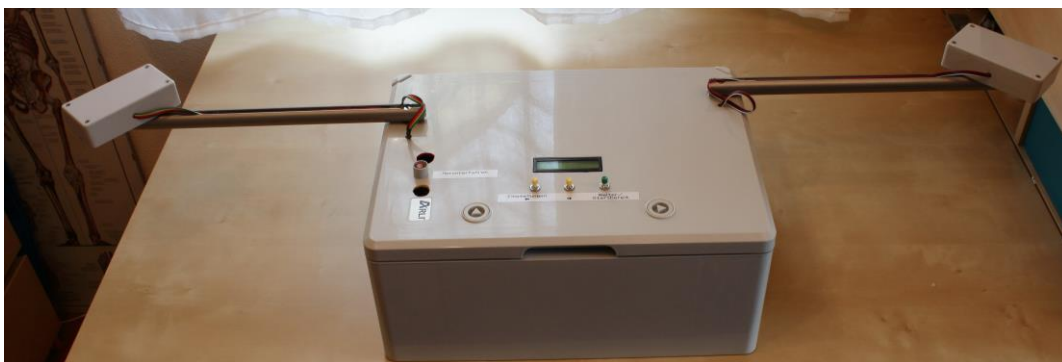


Abbildung 17 Optimaler Messbereich - Theorie



²² K+S Sportstätten Service West: "Technische Information Anforderungen, Maße und Linierungen für Kleinspielfelder und Leichtathletikanlagen", in: <http://bit.ly/2hxPGic>, Zugriff am 27.09.2017

Die Messung könnte weiter optimiert werden, indem die Sensoren direkt auf dem Boden stehen, anstatt auf einem Sockel und schräg nach vorne/oben messen. Dadurch kann die Anlage auch an Laufbahnen, an denen direkt ein Hindernis wie beispielsweise einem Gebäude angrenzt, verwendet werden. Die dafür notwendige Anlage benötigt entweder eine komplexere Hardware oder die Bedienung wird erschwert.

4.2.2 Gleichzeitige Messung auf zwei Bahnen

Die im vorherigen Kapitel beschriebene Ansteuerung eines Ultraschallsensors findet in der finalen Version als Grundprinzip Verwendung.

Eine Möglichkeit, beide Bahnen zu überwachen, wäre es, beide Messungen abwechselnd abzuarbeiten. Zeitlich ist es nicht möglich, die 48 benötigten Messungen pro Sekunde nacheinander auf beiden Bahnen durchzuführen, deshalb müssen beide Messungen parallel ausgeführt werden. Ebenfalls nicht möglich ist es, die Sensoren mithilfe der selben „while“-Schleifen auszulesen, denn sie müssen zu unterschiedlichen Zeitpunkten den Zustand wechseln und „Triggern“ können.

Damit beide Sensoren ihre Messung zeitlich parallel durchführen, auch wenn sie sich in unterschiedlichen Stadien der Messungen befinden, kommt eine Softwarestruktur namens „State Machine“ zur Verwendung.²³ Dadurch wird ein extrem schneller Programmdurchlauf ermöglicht, der fähig ist, beide Messungen parallel durchzuführen. Diese Struktur besteht aus zwei Methoden, „zielkontrolle()“ und „stateMachine()“. Die Methode „zielkontrolle()“ ruft „stateMachine()“ in einem bestimmten Takt, mittels einer „while“-Schleife, immer wieder auf. Die Methode „stateMachine()“ beinhaltet zwei nahezu identische „switch - cases“, einen pro Ultraschallsensor. In jedem „switch - case“ wird, abhängig vom Zustand der jeweiligen Messung, genau eine Aktion ausgeführt, damit das Programm nicht zu lange in einem Teilschritt verharrt und die jeweils andere Messung verfälscht. Sollte das Programm an einer Stelle eine bestimmte Zeit lang warten müssen, wird in jedem Durchlauf lediglich eine Mikrosekunde gewartet und

²³ Schwarz, A.: „Erklärung der Softwarestruktur Statemachine“, in: <http://bit.ly/2z9kyxo>, Zugriff am 10.08.2017

danach der Durchlauf fortgesetzt. Außerdem enthält die Methode „stateMachine()“ eine „if“-Option zur Bestimmung der Noten, die nur durchlaufen wird, nachdem beide Sensoren einen Zieldurchlauf erkannt haben. Im Folgenden wird die Funktionsweise des ersten Ultraschallsensors erklärt, der zweite arbeitet identisch, Unterschiede finden sich ausschließlich in der Benennung der verwendeten Variablen.

Die Variable „zustandEINS“ speichert den Zustand, in dem sich das Programm momentan befindet. Sie wird, sobald ein Case vollständig abgearbeitet wurde, auf den nachfolgenden Zustand gesetzt. Der komplette Durchlauf kann unter Umständen mehrere Aufrufe der Methode „stateMachine()“ in Anspruch nehmen.

Case 0:	Initialisierung aller messungsspezifischen Variablen und setzen des „Triggers“, um diesen später auslösen zu können.
Case 1:	Wartet, bis der Ultraschallsensor das Signal am „trig“-Eingang sicher wahrgenommen hat. Dazu wird dieser Zustand für 26 Durchläufe beibehalten, bevor der nächste Case aufgerufen wird. Bei jedem der 26 Durchläufe wird jeweils eine Mikrosekunde gewartet.
Case 2:	„Triggern“ des Sensors.
Case 3:	Wartet einmalig eine Mikrosekunde, sodass der GPIO sicher seinen Zustand wechselt und somit sichergestellt ist, dass der Sensor wirklich getriggert wurde. Um den Zustand eines GPIOs zu ändern, muss das Programm kurz unterbrochen werden, um dem System zu ermöglichen, auf den jeweiligen GPIO zuzugreifen.
Case 4:	Wartet, bis erstes „HIGH“-Signal am „Echo“-Pin gelesen wird. Sollte dies nach 1000 Durchläufen nicht geschehen sein, bricht diese Messung ab und es wird wieder mit Case 0 begonnen.
Case 5:	Wartet auf ein „LOW“-Signal am „Echo“-Pin, sollte dies vor dem Timeout anliegen, wird die Länge des „HIGH“-Signals gemessen, darauf folgt Case 6. Die Timeout Funktion bricht die Messung ab, falls ein Signal zu lange dauert, was ein weiter als 2 Meter entferntes Ziel voraussetzt. Auf den Abbruch folgt Case 0.
Case 6:	Prüft anhand der Dauer des HIGH Signals, wie weit ein Hindernis entfernt ist. Sollte diese Messung, sowie die Vorherige eine Dauer von weniger als 9860 Mikrosekunden haben, wurde ein Läufer erfasst. Daraufhin folgt „Case 7“. Sollte nur diese Messung weniger als 9860 Mikrosekunden

	benötigt (Entfernung bis zum ersten Hindernis <1,80 Meter) haben, wird die Variable „vorherig_ZielerreichtEINS“ auf Eins gesetzt, anderenfalls auf Null. In beiden Fällen geht es danach mit Case 0 weiter.
Case 7:	Beim ersten Durchlauf dieses Cases wird die Variable „abgeschlossenEINS“ auf Eins gesetzt. Bei allen weiteren Durchläufen wird nur noch der jeweils andere Sensor ausgewertet. Sobald dieser Case in beiden „Switch-cases“ durchlaufen wurde, beginnt die Auswertung, die beide Messungen beendet.

5 Ausblick

Bei der beschriebenen Zeitmessenanlage handelt es sich um einen optimierbaren Prototypen. Es sind noch weitere Ausbaustufen denkbar, für die der zeitliche Rahmen dieser Arbeit jedoch nicht ausreicht. Denkbar wären die folgenden Erweiterungen beziehungsweise Optimierungen:

- Die Einstellungen könnten durch einen weiteren Punkt „Wertungsart“ und eine passende Auswertung so erweitert werden, dass zusätzlich die Punktvergabe nach den Kriterien der Bundesjugendspiele möglich wäre.
- Um das Aufstellen der Zeitmessenanlage dauerhaft einfacher zu gestalten, ließe sich das Zielgerät um ein höhenverstellbares Stativ erweitern.
- Die Funkverbindungen könnten durch professionelle und damit leistungsfähigere Antennen verstärkt und dadurch weniger anfällig für Störsignale werden. Die im Handel verfügbaren, allerdings teureren Antennen wären dann gleichzeitig weniger platzaufwändig.
- Das Startgerät könnte um einen Taster zum Signalisieren eines Fehlstartes erweitert werden, auf dessen Druck hin das Startgerät ein entsprechendes Signal an das Zielgerät sendet.

Im Rahmen dieser Arbeit konnte ich vielfältige Erfahrungen sammeln. Durch die Vermischung der Bereiche Software und Hardware konnte ich erste, sehr intensive Eindrücke von der Komplexität beim Zusammenspiel der einzelnen Komponenten

gewinnen. Ich musste die Erfahrung machen, dass auch die Umsetzung scheinbar einfacher Details sehr aufwändig sein kann und dass hierbei eine gründliche Planung sehr wichtig ist.

6 Anhang

- Zeitmessanlage für Sprintsportarten
- Datenträger mit vollständigem Programm-Code
- Preisliste

7 Literaturverzeichnis

7.1 Bücher

Erlenkötter, H.: „Grundkurs Computerpraxis C Programmieren von Anfang an“, 23. Auflage Februar 2016, S.165ff

7.2 Internet Quellen

Staatsinstitut für Schulqualität und Bildungsforschung: „Durchführung des Sportunterrichts in den Jahrgangsstufen 11 und 12“, in: <https://goo.gl/F6jAt6>, Zugriff am 1.11.2017, S. 10

Wikimedia Foundation Inc.: „ATA/ATAPI - Steckerbelegung“, in: <https://de.wikipedia.org/wiki/ATA/ATAPI>, Zugriff am 1.11.2017

Bundesnetzagentur: „Frequenzplan – Amateurfunk“, in: <https://goo.gl/Koaetv>, Zugriff am 28.09.2017, S.272

easyacc.com, Inc.: „Kaufangebot mit Information über die Akkubank“, in: <http://bit.ly/2h68bNV>, Zugriff am 1.11.2017

Conrad Electronic SE(Hrsg.): „Katalogeintrag rote LED“, in: <https://goo.gl/wWP7Mq>, Zugriff am 24.09.2017

Conrad Electronic SE(Hrsg.): „Katalogeintrag grüne LED“, in: <https://goo.gl/bhb68X>, Zugriff am 24.09.2017

Amazon.com Inc.: „Kaufangebot mit Informationen über Lautsprecher“, in: <https://goo.gl/M9dFE7>, Zugriff am 28.07.2017

KT-Elektronik: „Datenblatt HC-SR04“, in: <https://goo.gl/6PvZQf>, Zugriff am 26.09.2017

Henderson, G.: „wiringPi“, in: <https://projects.drogon.net/contact/>, Zugriff am 24.9.2017

Henderson, G.: „wiringPi Funktionen“, in: <http://bit.ly/2A0YofT>, Zugriff am 24.09.2017

Campbell, S.: „How to Setup an LCD on the Raspberry Pi and Program it With C“, in: <https://goo.gl/WpbtrP>, Zugriff am 31.10.2017

Huang, J.: „Play MP3 with libmpg123 and libao“, in: <http://bit.ly/2gWMDi0>, Zugriff am 31.10.2017

Becker, M.: „Eine Einführung in Makefiles“, in: <http://bit.ly/2xNkOQU>, Zugriff am 1.11.2017

Free Software Foundation, Inc.: „GCC, the GNU Compiler Collection“, in: <https://gcc.gnu.org/>, Zugriff am 1.11.2017

Schweizer, P.: „Autostart mit rc.local“, in: <https://goo.gl/HhfnMf>, Zugriff am 1.11.2017

Amazon.com, Inc.: „Kaufangebot mit Informationen der Sende- und Empfangseinheit“, in: <https://goo.gl/Q12P3z>, Zugriff am 28.9.2017

Schwarz, A.: „433Mhz Funkübertragung – Bau einer Dipol Antenne“, in: https://www.mikrocontroller.net/articles/433_MHz_Funk%C3%BCbertragung, Zugriff am 28.09.2017

K+S Sportstätten Service West: „Technische Information Anforderungen, Maße und Linierungen für Kleinspielfelder und Leichtathletikanlagen“, in: <http://bit.ly/2hxPGic>, Zugriff am 27.09.2017

Schwarz, A.: „Erklärung der Softwarestruktur Statemachine“, in: <http://bit.ly/2z9kyxo>, Zugriff am 10.08.2017

Hipp,M. und Orgis,T.: „API documentation for libmpg123 and libout123“, in: <https://www.mpg123.de/api/>, Zugriff am 31.10.2017

Xiph.Org.: „Download libao“, in: <https://www.xiph.org/downloads/>, Zugriff am 1.11.2017

Raspberry Pi Foundation: „Shutdown in an C program“, in: <https://goo.gl/nbJ7TY>, Zugriff am 1.11.2017

Free Software Foundation, Inc.: „GCC, the GNU Compiler Collection“, in: <https://gcc.gnu.org/>, Zugriff am 1.11.2017

Staatsinstitut für Schulqualität und Bildungsforschung: „Leistungsbewertung im Fach Sport für alle weiterführenden Schulen“, in: <https://goo.gl/D4mqEY>, Zugriff am 1.11.2017

Seeed Development Limited: „Grove – Relay Introduction“, in: <http://wiki.seeed.cc/Grove-Relay/>, Zugriff am 1.11.2017

123apps LLC : „Audio Converter online“, in: <http://online-audio-converter.com/de/>, Zugriff am 1.11.2017

7.3 Abbildungsverzeichnis

Abbildung 1 Taster "Herunterfahren"	6
Abbildung 2 Holzsockel mit Koaxsteckern	7
Abbildung 3 Holzsockel	8
Abbildung 4 Benutzerinterface Startgerät.....	9
Abbildung 5 Benutzerinterface Rote LED	9
Abbildung 6 Startgerät offen	10
Abbildung 7 Startgerät	10
Abbildung 8 Benutzerinterface Zielgerät.....	11
Abbildung 9 HC-SR04	11
Abbildung 10 Zielgerät Transport/Lagerung.....	12
Abbildung 11 Lagerung Startgerät.....	12
Abbildung 12 Lagerung der Antennen.....	13
Abbildung 13 Dipol Antennen	19
Abbildung 14 Prototypen für Reichweitentests	19
Abbildung 15 Sender und Empfänger mit Koaxsteckern	20
Abbildung 16 Grafik Ultraschallsensor Messfehler	22
Abbildung 17 Optimaler Messbereich - Theorie	23
Abbildung 18 Schallschluckende Schaumstoffmatten.....	23
Abbildung 19 Optimaler Messbereich - Umsetzung.....	23

Abbildung 1;2;3;4;5;6;7;8;10;11;12;13;14;15;17 sowie 19 sind selbst fotografiert.

Abbildungen 16; 18 selbst erstellt.

7.4 Bildquellen

Stoppuhr Titelblatt

Publicdomainvectors.org: „Chronometer-Vektor-Bild“, in: <https://goo.gl/vhTFCD>, Zugriff am 1.11.2017

Abbildung 9: Foto HC-SR04

Banggood Ltd: „Verkaufsangebot HC-SR04“, in: <https://goo.gl/Z2eKYC>, Zugriff am 1.11.2017

8 Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich diese Arbeit selbstständig und ohne fremde Hilfe, nur unter der Zuhilfenahme der angegebenen Quellen verfasst habe.

Ort, Datum

Unterschrift des Verfassers