

Entwicklung und testen eines Ultraschall-Entfernungsmessers als Vorbereitung eines Produktentwurfes

Projektarbeit

erstellt an der
Fachschule für Technik des Carl-Severing-Berufskolleg
für Metall- und Elektrotechnik der Stadt Bielefeld



Erstellt durch:

Eduard Meiser
Omar Hachimi
Stephan Dannat
FET6A

in Zusammenarbeit mit der Fa. Tinkerforge
betreut durch
Herr Simon

Bielefeld, 22. März 2018

Persönliche Erklärung

Hiermit bestätigen wir, dass die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Hilfsmittel benutzt wurden. Die Stellen der Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken (dazu zählen auch Internet-quellen) entnommen sind, wurden unter Angabe der Quellen kenntlich gemacht.

Bielefeld, _____

Eduard Meiser

Omar Hachimi

Stephan Dannat

Inhaltsverzeichnis

1	Einleitung	1
1.1	Zusammenfassung	1
1.2	Lastenheft	1
1.2.1	Über Tinkerforge	1
1.2.2	Motivation	1
1.2.3	Aufgabenbeschreibung	1
1.3	Management des Projektes	2
1.3.1	Trello	2
1.3.2	Github	2
1.4	Richtlinien zu Erstellung von Schaltplänen und Platinenlayout anhand einem ECAD- Programmpaket	3
1.4.1	Schaltplan	3
1.4.2	Platinenlayout	3
2	Vorbereitung	4
2.1	Recherche der Funktionsweise	4
2.1.1	Mikrocontroller	4
2.1.2	Sender	4
2.1.3	Hochsetzsteller	5
2.1.4	Ultraschallkapsel	5
2.1.5	Filter	5
2.1.6	Empfänger	5
2.2	Verwendete sowie Dimensionierung der Bauteile	5
2.2.1	Mikrocontroller	5
2.2.2	Sender	6
2.2.3	Hochsetzsteller	6
2.2.4	Ultraschallkapsel	7
2.2.5	Filter	7
2.2.6	Empfänger	7
2.3	Entwicklung der Software zum Betrieb des Prototypen	8
2.3.1	Benötigte Kenntnisse	8
2.3.2	Quellcodeentwurf	9
3	Messungen und Auswertung der Ergebnisse	12
3.1	Prototyp 1	12
3.1.1	Senderkreis	12
3.1.2	Empfängerkreis	13
3.2	Prototyp 2	15
3.3	Fazit aus den Ergebnissen für den Auftraggeber	17
4	Reflektion über den Projektablauf	19
5	Anhänge	20

1 Einleitung

1.1 Zusammenfassung

Bei dieser Projektarbeit wurden Messungen an einem zuvor selbst entworfenen Prototyp eines Ultraschall-Entfernungsmessers durchgeführt. Das geschah im Auftrag der Firma Tinkerforge, um herauszufinden, wie sich ein solcher Sensor realisieren lässt. Dabei gab es zwei Möglichkeiten und entsprechend dieser auch zwei unterschiedliche Prototyp-Versionen. Begonnen wurde mit einer Variante mit getrenntem Sende- und Empfangsbetrieb, auf zwei Platinen aufgebaut. Danach wurde eine zweite Variante mit kombiniertem Betrieb, auf einer Platine aufgebaut. Diese zweite Prototyp-Version besaß nur noch eine Ultraschallkapsel für den Sende- und Empfangsbetrieb. Trotz mehrerer notwendiger Veränderungen an den Platinen ließen sich Aussagekräftige Messungen durchführen.

1.2 Lastenheft

1.2.1 Über Tinkerforge

Die Tinkerforge GmbH wurde Ende 2011 mit dem Ziel gegründet, die Handhabung eingebetteter Systeme zu vereinfachen. Das Tinkerforge Baukastensystem besteht aus aktuell fast 80 verschiedenen Modulen, die vom Anwender flexibel für die jeweilige Aufgabe kombiniert werden können. Zu den Modulen zählen diverse Sensor- Aktor- und Schnittstellenmodule, die alle über Hochsprachen wie C#, Python und Java gesteuert werden können. Tinkerforge unterstützt aktuell 17 verschiedene Programmiersprachen. Sowohl Hardware als auch die Software aller Module sind OpenSource. Die Stärke des Baukastensystems ist aus Anwendersicht die enorme Flexibilität, die Einfachheit und die Schnelligkeit mit der Projekte realisiert werden können. Es eignet sich daher besonders im Bereich Rapid Prototyping. Daher findet das Tinkerforge Baukastensystem Anwendung in vielen Forschungsinstituten, in diversen Entwicklungsabteilungen bekannter Automobilhersteller und Ingenieurbüros.

1.2.2 Motivation

Diese Technikerarbeit soll die Grundlage zur Entwicklung eines Entfernungssensors für das Baukastensystem bilden, der auf einer Ultraschall-Entfernungsmessung basiert. Das Baukastensystem verfügt aktuell über so einen Sensor. Bei diesem handelt es sich um ein zugekauftes Modul, welches nicht die gewünschten Leistungen liefert. Daher soll an einem zu entwerfenden Prototypen Forschung betrieben werden, um eine eigene Lösung entwerfen zu können.

1.2.3 Aufgabenbeschreibung

Innerhalb dieser Arbeit soll der Entwurf eines Prototypen des Entfernungssensors und die damit verbundene Forschung durchgeführt werden. Dabei ist durch Recherche zu erarbeiten, welche Möglichkeiten zur Realisierung zur Verfügung stehen. Durch Messungen am Prototypen soll festgestellt werden, welche dieser Möglichkeiten funktional realisierbar sind, um ein eigenes Produkt zu erstellen.

Diverse Teilaufgaben sind zu erledigen:

- **Recherche**

Zu Anfang muss recherchiert werden, welche Möglichkeiten es gibt mittels Ultraschall eine

Entfernung zu ermitteln und wie diese technisch umgesetzt werden können. Zusätzlich müssen die Techniker sich mit dem Tinkerforge Baukastensystem und seiner internen Funktionsweise vertraut machen.

- **Bauteilauswahl**

Abhängig von der gewählten technischen Umsetzung müssen geeignete Komponenten ausgewählt werden. Die Auswahl sollte auch unter dem Gesichtspunkten Preis, der Bauteilverfügbarkeit und der technischen Anforderungen erfolgen.

- **Schaltplanentwurf und Layouterstellung**

Von Tinkerforge wird das Open Source CAD Programm KiCad verwendet. Mit diesem Programm ist ein Schaltplan für den Prototypen und anschließend ein Leiterplattenlayout zu erstellen.

- **Leiterplattenbestückung**

Die erstellte Leiterplatte wird von Tinkerforge in Auftrag gegeben. Diese muss mit den gewählten Komponenten bestückt werden. Die Tinkerforge GmbH stellt dazu die notwendigen Werkzeuge bereit.

- **Einrichten und Einarbeitung in die Tinkerforge Toolchain**

Viele Softwarekomponenten werden von der Tinkerforge Toolchain automatisch generiert. Um diese Nutzen zu können muss ein Linux System eingerichtet werden. Anschließend muss sich mit der Funktionsweise des Generators und der Softwareversionsverwaltung „Git“ vertraut gemacht werden.

- **Testsoftware und Forschung**

Um Messungen an der Hardware durchführen zu können gilt es Programmblöcke zu entwerfen, mit denen die einzelnen Funktionen der Baugruppen getestet werden können. So soll ermittelt werden, wie zum einen das Ultraschallsignal effektiv ausgegeben werden kann und wie sich die Signalamplitude auf die Reichweite auswirkt. Zum anderen gilt es zu recherchieren, wie das zurückkommende Signal verarbeitet werden kann. Auch soll erarbeitet werden, wie gut das Signal unter verschiedenen Bedingungen verarbeitet werden kann und ob eine zuverlässige Verarbeitungsqualität ohne großen Aufwand realisierbar ist.

1.3 Management des Projektes

1.3.1 Trello

Zur zeitlichen Planung und Übersicht des Ablaufes wurde auf das Onlinetool Trello zurückgegriffen. Dieses ist ein kostenfreies, webbasiertes Projektmanagementtool. Es ermöglicht den Gruppenmitgliedern gleichzeitig von verschiedenen Orten auf die Oberfläche zuzugreifen und Änderungen vorzunehmen. So kann ein Teilnehmer auch neue Termine mit Kennzeichnung der Fälligkeit für andere Gruppenmitglieder einfügen, oder bereits erledigte Aufgaben für alle abhaken. Auch können hier relevante Dokumente, die alle Gruppenmitglieder lesen sollen hochgeladen, und bei Bedarf noch kommentiert werden. Für die Dokumentation lässt sich an diesem System auch einfach abgleichen, zu welchen Zeitpunkten die einzelnen Aufgaben abgeschlossen wurden.

1.3.2 Github

Bei Github handelt es sich um einen webbasierten Online-Dienst, der Server für Entwicklungsprojekte mit einer Versionsverwaltung bereitstellt. So können alle Daten nach einer Änderung im Programm wieder hochgeladen und mit einem Kommentar versehen werden. Sollte nach mehreren Änderungen ein Problem auftreten, kann einfach auf eine ältere Version zurück gegriffen und so der Fehler eingegrenzt werden. Auch kann ein Projekt in mehrere Abschnitte aufgeteilt werden, damit

mehrere Personen unabhängig voneinander daran arbeiten können. Nach der Bearbeitung können diese wieder zusammengefügt werden. Dabei ist erkennbar, welche Änderungen von wem vorgenommen wurden. So können alle Vorgänge jederzeit verfolgt werden, um eine größtmögliche Übersicht zu gewährleisten. Durch das Kommentieren der Änderungen kann die Nachvollziehbarkeit dieser ebenfalls deutlich gesteigert werden. Des weiteren ist diese Plattform gerade für Unternehmen wie Tinkerforge, die ihren Quellcode als Open-Source anbieten besonders praktisch, da den Nutzern hier alle veröffentlichten Daten direkt zur Verfügung stehen.

1.4 Richtlinien zu Erstellung von Schaltplänen und Platinenlayout anhand einem ECAD-Programmpaket

Das Open Source ECAD-Programm KiCAD ist eine Anwendung zum Erstellen von Schaltplänen und elektronischen Leiterplatten. Mit einem solchen Programm lassen sich die erstellten Schaltpläne vor einer Leiterplattenerstellung auf Verdrahtungsfehler prüfen, um spätere Probleme zu vermeiden.

1.4.1 Schaltplan

Beim Schaltplanentwurf gilt es auf gewisse Regeln zu achten, zu dem ist auf die Übersichtlichkeit des Schaltplans zu achten. So sollten beispielsweise Filterkondensatoren an der Spannungsversorgung des Mikrocontrollers eingeplant werden, um die Versorgungsspannung zu stabilisieren. Auch ist schon bei dem Entwurf des Schaltplans an das spätere Platinenlayout zu denken. So muss bei der Platzierung der Bauteile darauf geachtet werden, dass die Signalintegrität gewährleistet ist, und es zu keiner Potential Verschiebung kommt.

Im Schaltplan wurden die Baugruppen Empfänger, Sender, Hochsetzsteller und die Anschlüsse vom Controller getrennt und so positioniert, dass die beste Übersicht dargestellt wird.

Um die einzelnen Bestandteile separat testen zu können, wurden dem Design NULL Ohm Widerstände, an den Verbindungspunkten, hinzugefügt. Außerdem ist es dadurch möglich, eine Zerstörung einzelner Baugruppen durch einen Verdrahtungsfehler bei der Erstinitialisierung zu vermeiden.

1.4.2 Platinenlayout

Beim Entwerfen eines Platinenlayouts gibt es viele Möglichkeiten ein Ergebnis zu erzielen. So können alle Bauteile so angeordnet werden, dass alle parallelen Bauteile nebeneinander aufgereiht werden und die in Reihe dazu liegenden Bauteile darunter angeordnet sind. So sähe die Platine zwar ähnlich eines Kontaktplanes aus, allerdings ist diese Variante aus Sicht der EMV nicht sonderlich empfehlenswert.

Eine andere Möglichkeit wäre, die Bauteile schon im Schaltplan in Gruppen zusammenzulegen und den Schaltplan auf der Platine exakt nachzubilden. Auch bei dieser Variante ergeben sich gelegentlich Probleme, was die Führung der Leitungen und vor allem den Verlauf der Ströme angeht.

So sollte ein Augenmerk auf den stromführenden Leitungen liegen. Je höher der Strom ist, desto breiter und kürzer ist die Leitung auszulegen, um weniger EMV-Störungen zu erzeugen. Auch sollte die Rückführung (GND) günstigerweise als eigene Leiterschicht ausgeführt werden, um einen großen Leiterquerschnitt zu gewährleisten. So kann bei der Rückführung der Ströme auch das Risiko vermieden werden, durch Bildung von größeren Schleifen, Antennen zu erzeugen. Die GND-Schicht sollte so wenig wie möglich unterbrochen werden, vor allem sind Unterbrechungen quer zur Stromflussrichtung zu vermeiden. Zusätzlich ist zu beachten, dass Kondensatoren, die der Verringerung von Störenden Spannungsschwankungen dienen, nahe schaltenden Bauteilen angebracht werden. Die optimale Platzierung ist direkt am VDD oder VCC des ICs, so dass die Leiterbahn vor dem IC mit einem höheren Querschnitt am Kondensator liegt und dann mit leicht verringertem Querschnitt an das IC angeschlossen ist.

2 Vorbereitung

Bevor ein erster Entwurf der Schaltpläne durchgeführt werden konnte, mussten Informationen zu den zu verwendenden Bauteilen eingeholt werden, um sicherzustellen, dass diese auch die passenden Betriebsparameter haben und untereinander kompatibel sind.

2.1 Recherche der Funktionsweise

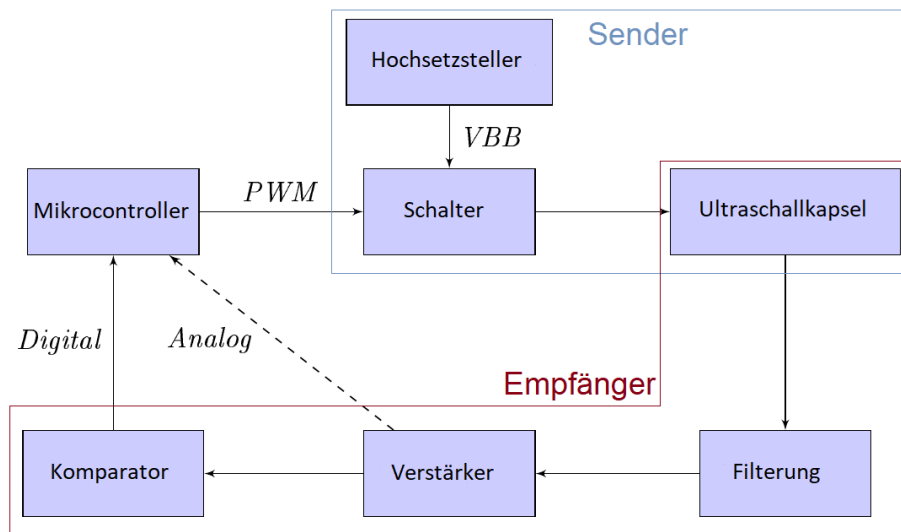


Abbildung 2.1: Blockschaltbild des Ultraschall-Entfernungsmessers

Die Abbildung 2.1 zeigt eine vereinfachte Funktionsübersicht der Platine. Die Schaltpläne zu den verschiedenen Ausführungen der Platine sind den Anlagen beigelegt.

2.1.1 Mikrocontroller

Der Mikrocontroller soll eine stabile und variable PWM ausgeben können. Es werden für die Zeitfassung und die PWM Ausgabe mehrere Timer benötigt, die sowohl intern als auch über die Hardware angesteuert werden können. Ein Analog/Digital-Wandler sollte ebenfalls vorhanden sein.

2.1.2 Sender

Der Sender soll Schallwellen im Ultraschallbereich (40 kHz) aussenden. Diese sollen einen ausreichenden Schalldruck haben um von Hindernissen, die mehrere Meter entfernt sind, ein Echo erhalten zu können. Dafür muss die Ultraschallkapsel mit einem sinusähnlichen Signal angesteuert werden. Dessen Amplitude muss angemessen hoch sein, um den gewünschten Schalldruck zu erzeugen. Der Controller ist auf eine Spannungsebene von 3,3 V begrenzt. Um höhere Spannungen an der Ultraschallkapsel realisieren zu können, muss ein Schalter als weiteres Bauteil dazwischen geschaltet werden. Dieses zusätzliche Bauteil dient als Trennung zwischen den 3,3 V des Mikrocontrollers und der höheren Spannungsebene der Lautsprecherkapsel. Dabei muss darauf geachtet werden, dass der Schalter schnell genug arbeitet, um die 40 kHz auch zuverlässig schalten zu können.

2.1.3 Hochsetzsteller

Der Hochsetzsteller generiert aus den 5 V Versorgungsspannung eine höhere Spannung für den Sendebetrieb. So kann der Schalldruck, der ausgegeben wird erhöht werden (größere Reichweite/-größeres Rücksignal) siehe Abbildung 3.14.

Die Funktionsweise des Hochsetzstellers (Spannungspumpe/Aufwärtswandler/Aufwärtsregler) ist überschaubar und findet in vielen Bereichen Anwendung. Grundsätzlich wird eine Induktivität in Reihe mit einer Freilaufdiode vor einen Ladekondensator geschaltet. Dieser liegt parallel zum Ausgang. Zwischen der Spule und der Diode ist ein Schalter angeschlossen, der die Spule gegen Masse schaltet. Die Spule lädt sich bei Betätigung des Schalters auf. Durch den Stromfluss entsteht ein Magnetfeld und beim Öffnen steigt die Spannung am sekundären Ende der Spule. Durch das zusammenbrechende Magnetfeld an und lädt den Kondensator auf. Dieser Vorgang wird wiederholt bis der Kondensator so weit aufgeladen ist, dass die Ausgangsspannung den gewünschten Wert hält. Die mögliche Ausgangsspannung ist nicht unbegrenzt über das Schaltspiel regelbar. Sie ist von den Eigenschaften der Komponenten abhängig.

2.1.4 Ultraschallkapsel

Für das Senden und Empfangen des Ultraschalls wird eine auf Piezomodulen basierende Ultraschallkapsel verwendet, weil Sie die Möglichkeit hat ein Signal auszugeben sowie zu empfangen und in Kleinbauform erhältlich ist. Wie stark das Nachschwingen der Kapsel eine Auswirkung auf die Empfängerschaltung hat, sowie die Größe der Störfrequenz bei verschiedenen Amplituden auf unterschiedlichen Entfernungen sollte überprüft werden. Es sollte auch getestet werden welche Kapseln sich für einen stand-alone Betrieb am besten eignen.

2.1.5 Filter

Die Filterschaltung soll am Eingang des Empfängers mögliche Störfrequenzen herauszufiltern um eine Verfälschung der gemessenen Strecke zu vermeiden, da die Ultraschallkapsel nicht ausschließlich Ultraschallsignale aufnimmt. Für Hohefrequenzen eignet sich eine passive Hochpassfilter-Schaltung, es wird zu einem Kondensator oder einer Spule ein Widerstand parallel zur Masse bestückt.

2.1.6 Empfänger

Im Empfangsbetrieb werden die zurückkommenden Schallsignale, die auf die Ultraschallkapsel treffen, in sinusförmige Spannungssignale umgewandelt. Die Amplitude hängt von dem Schalldruck der empfangenen Signale ab und ist deutlich niedriger als die Amplitude der gesendeten Signale. Diese Signale müssen anschließend verstärkt werden, um sie mit dem Mikrocontroller auswerten zu können. Zur Vereinfachung der Auswertung wird das eingehende analoge Signal in ein digitales Signal umgewandelt, um Verarbeitungsaufwand einsparen zu können. Die Auswertung eines Analogsignals erfordert mehr Programmieraufwand als die Auswertung eines Digitalsignals. Um die Qualität der gemessenen Entfernung besser zu beurteilen sollte eventuell auch das Analogsignal ausgewertet und mit dem digitalen Signal verglichen werden.

2.2 Verwendete sowie Dimensionierung der Bauteile

Bevor ein erster Prototyp erstellt werden konnte, mussten die richtigen Bauteile ausgewählt werden. Dies geschah unter anderem anhand der gewonnenen Informationen aus dem Kapitel 2 Vorbereitung.

2.2.1 Mikrocontroller

Der Infineon XMC 1404_Q048 gehört zur der Familie der ARM Cortex -M0 Prozessoren und ist ein 32-bit Industrial Microcontroller, wird mit 48 MHz externer Clock betrieben. Die 48 im Namen des Prozessors steht für die Anzahl der Pins. Der interne Timer läuft mit 96Mhz. Die CCU4

des Prozessor bietet ein 2x4 16 bit Timer. Außerdem hat der XMC einen 12 bit A/D Wandler, welcher für die Analogmessung eine genauere Auflösung bieten kann als ein 8 bit A/D Wandler. Die Betriebsspannung des Prozessors beträgt 3,3 V. Die Auswahl des Controller wurde getroffen, weil dieser standardmäßig auch schon bei Tinkerforge eingesetzt wird und uns für den Prototypen vorgegeben wurde.

2.2.2 Sender

Um eine Trennung zwischen der CPU (3,3 V) und dem Hochsetzsteller (5 V-20 V) zu ermöglichen, wurde ein weiteres Bauteil benötigt. Dafür wurde das IC A5950 (Voll Brücke) ausgewählt. Diese H-Brücke kann eine angeschlossene Last mit der vom Hochsetzstellers erzeugten Spannung versorgen. Deren Frequenz wird über das Signal an dem Anschluss Phase von der CPU vorgegeben. An den Anschlüssen Out1 und Out2 wird das Ausgangssignal abgegriffen. Für die genaue Beschaltung des ICs siehe Anhang Datenblatt "Schimatic A5950".

2.2.3 Hochsetzsteller

Tinkerforge nutzt schon eine Variante eines Hochsetzstellers auf ihren Platinen. Der Aufbau und die Bauteilauswahl musste nur auf die variable Ausgangsspannung erweitert werden. Die Standardschaltung, von Tinkerforge, wurde an dem Eingang der Feedbackspannung mit einem Potentiometer (RV1) erweitert, um die gewünschte Variable Spannung zu generieren.

Die Ausgangsspannung wird mit den externen Widerständen RV1, R13 und R2 eingestellt (siehe Grundschtung Datenblatt vom IC LMR62014x, Seite 11). Ein Wert von ca. 13,3 kΩ wurde für R2 empfohlen. Mit den Potentiometer (RV1) lässt sich die Ausgangsspannung bei 5 V Eingangsspannung bis 21,5 V einstellen. In der Formel werden die Widerstände RV1 und R13 zum Ersatzwiderstand(Re) zusammengefasst.

$$Re = R2 * \left(\frac{V_{out}}{1,23} - 1 \right) \Rightarrow V_{out} = \left(\frac{Re}{R2} + 1 \right) * 1,23$$

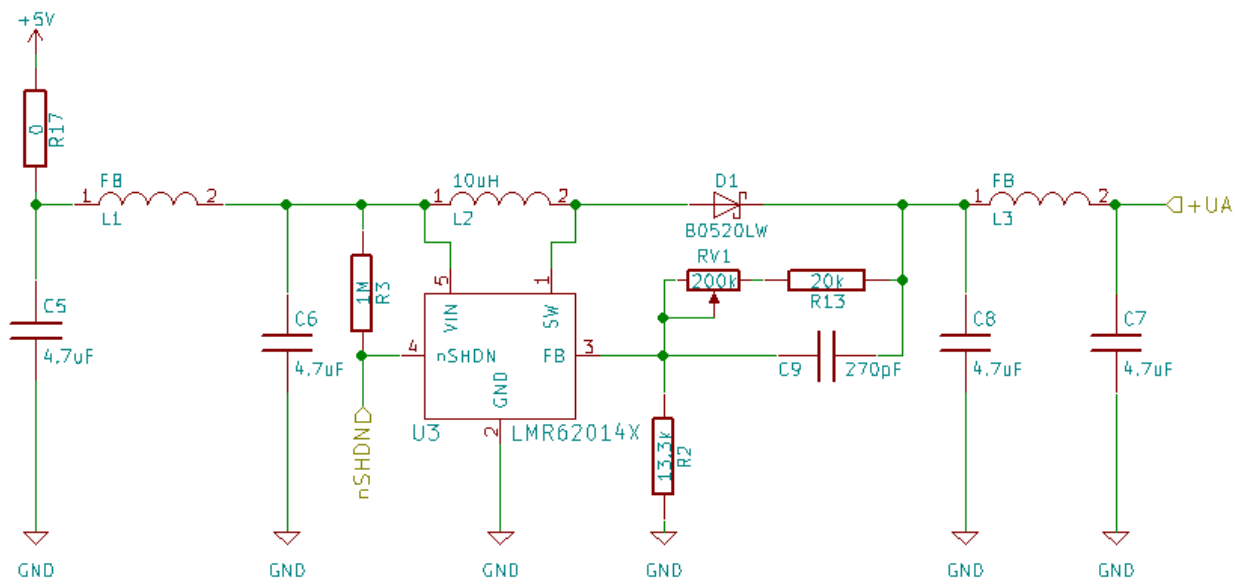


Abbildung 2.2: Schaltplan des beschalteten Hochsetzstellers

2.2.4 Ultraschallkapsel

Für den Prototypen wurden mehrere Kapseln verschiedener Hersteller bestellt. Dieses geschah um Unterschiede der verschiedenpreisigen Bauteile zu ermitteln und festzustellen welches Preissegment die nötige Qualität für die vorliegende Anwendung erfüllt.

2.2.5 Filter

Um unerwünschte Signalanteile mit Frequenzen, die unter 40 kHz liegen, zu unterdrücken musste die Filterschaltung mit einem Hochpassfilter (CR Glied) bestehend aus einem Kondensator und einem Widerstand bestückt. Der Widerstand wurde nach der e24 Reihe ausgewählt. Die Kapazität des Kondensators wurde an die Grenzfrequenz von 40 kHz und den Widerstand angepasst.

$$C = \frac{1}{2 * \pi * f_g * R} \Rightarrow \frac{1}{2 * \pi * 40 \text{ kHz} * 100 \text{ K}\Omega} \approx 40 \text{ pF}$$

Anhand der Berechnung wurde für den Hochpassfilter ein Kondensator mit 39 pF genommen.

2.2.6 Empfänger

Die Abbildung 2.3 zeigt die Empfängerschaltung. Durch diese Verschaltung von Operationsverstärkern wird das ankommende sinusförmige Signal verstärkt und in ein digitales Signal umgewandelt. Für die Verstärkung der Amplitude so wie der Umwandlung des analogen Signals in ein Rechtecksignal mit 40 KHz wurde der TLC272 ausgewählt, weil er günstigem Stückpreis ist und die Technischenspezifikationen ausreichend sind. Für die Verstärkung der Amplitude ist der Operationsverstärker TLC272 U2B als nicht invertierender Verstärker geschaltet.

Für die Umwandlung des analogen Signales in ein digitales wurde der Operationsverstärker TLC272 U2C als Komparator geschaltet. Beim Auftreten von Differenzen zwischen den Eingangssignalen, wechselt der Ausgang des Komparators zwischen Low (0 Volt) auf High (3,3 Volt).

Die Referenzspannung (U_{ref}) wird durch den Spannungsteiler R9 und R8 bestimmt.

$$U_{ref} = \frac{U_{ges} * R9}{R8 + R9} \Rightarrow \frac{3,3 \text{ V} * 120 \text{ K}\Omega}{100 \text{ K}\Omega + 120 \text{ K}\Omega} = 1,8 \text{ V}$$

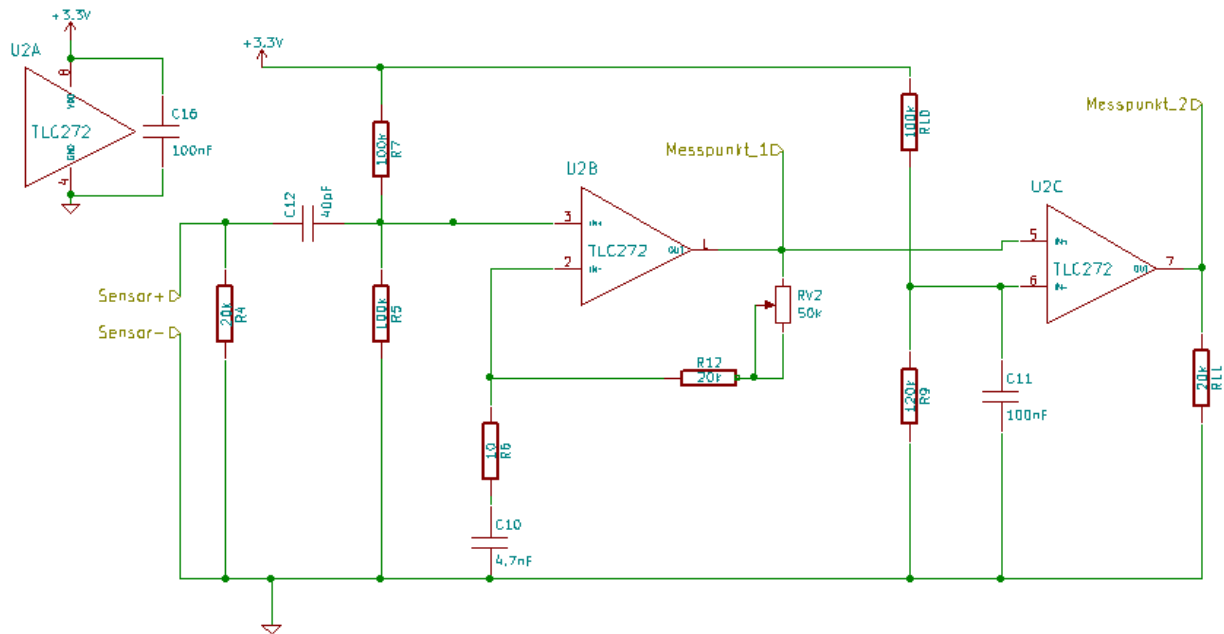


Abbildung 2.3: Schaltplan zweier Operationsverstärker mit einer vorgelagerten Filterung als Empfänger-schaltung

2.3 Entwicklung der Software zum Betrieb des Prototypen

2.3.1 Benötigte Kenntnisse

Um das Programm zu erstellen sollten Kenntnisse z.B. für die Capture Compare Unit des Mikrocontrollers vorhanden sein. Die CCU4 hat insgesamt 4 Einheiten und jede einzelne besitzt mehrer Slices von CC40 bis CC43. Slices können untereinander verschachtelt werden um so z.B. Interrupt Service Routinen aufzurufen, siehe dazu das Reference Manuel. Die Werte der Register mit denen die Zeit gemessen wird müssen sofort zum beginn der Interrupt Service Routine zwischengespeichert werden damit sichergestellt wird das nicht die Zeit zur Brechung mit gemessen wird.

Durchgeführte Berechnungen

Auch für die Programmierung waren diverse Berechnungen notwendig. Zur Erzeugung der Ultraschallimpulse wurde ein pulswertenmoduliertes Rechtecksignal mit einer Frequenz von 40 kHz generiert. Dafür war es notwendig mit Hilfe der CCU4 einen Timer zu erstellen. So musste bei einem Timertakt von 96 MHz eine Periodendauer von 2400 Takten und einem Compare_Wert von 1200 Takten konfiguriert werden. Im Zählvorgang des Timers wird der Ausgang nach Erreichen des Compare-Wertes auf 1 gesetzt, und nach Erreichen der Periodendauer wieder auf 0 zurückgesetzt. Dadurch ergibt sich eine Periodendauer von 25 μ s, was einer Frequenz von 40 kHz entspricht. Die Zeit die vergeht bis das Echo des Ultraschall-Impulses zurück kommt wird über einen Timer erfasst.

$$Periodendauer = \frac{96 \text{ MHz}}{40 \text{ kHz}} = 2400, \quad Compare_Wert = \frac{2400}{2} = 1200$$

2.3.2 Quellcodeentwurf

Programmstruktur:

In dem Hauptprogramm `main.c` befinden sich nur Funktionsaufrufe die eine Grundkonfiguration für den Mikrocontroller beinhaltet und eine Schleife die immer wieder abgerufen wird um z.B. eine neue Entfernungsmessung zu starten, siehe Abbildung 2.4. Anstatt den gesamten Programmcode im Hauptprogramm zu verfassen was bei komplexen Programmen schnell zu Unübersichtlichkeit führt hat das Auslagern den Vorteil, dass der Quellcode Logisch getrennt werden kann was eine Verschlinkung des Quellcodes mit sich bringt. Somit stehen in dem Hauptprogramm nur noch die Aufrufe der separat verfassten Funktionen. Auch vereinfacht diese Struktur gerade bei Prototypen das Testen der Funktion. So kann im Falle einer fehlerhaften Funktion einfach ein Aufruf auskommentiert werden um zu testen, ob der Fehler wirklich von der Funktion herrührt. Dadurch müssen nicht etliche Zeilen Programmcode einer Funktion auskommentiert werden. Dadurch ist die Fehlerrate durch übriggebliebene Zeichen, oder gar beim entfernen der Auskommentierung gelöschte Zeichen deutlich gesenkt.

```
1 #include <stdio.h>
2 #include <stdbool.h>
3 #include "bricklib2/logging/logging.h"
4 #include "bricklib2/bootloader/bootloader.h"
5 #include "communication.h"
6 /****Eigene Include Dateien*****/
7 #include "configs/config.h"
8 #include "system_timer/system_timer.h"
9 #include "a16pt.h"
10 int main(void)
11 {
12     logging_init();
13     logd("Start Distance US V2 Bricklet/n/r"); //For the Debugmodus
14     communication_init(); //Function call
15     a16pt_init(); //Function call
16     while(true)
17     {
18         a16pt_tick(); //Function call
19         bootloader_tick(); //Function call
20         communication_tick(); //Function call
21     }
22 }
23 }
```

Abbildung 2.4: Quellcode der `main.c` des Distance US

Die verwendete Vorgehensweise bei der Programmierung arbeitet mit einer gewissen Verschachtelung. In den *.c Dateien befindet sich zwar der Großteil des des aktiv geschriebenen Programms, doch ste-

hen die Funktionsdefinitionen in den Headerdateien, wie der `a16pt.h` 2.5

```
1 #ifndef A16PT_H
2 #define A16PT_H
3 #include <stdint.h>
4 void a16pt_init(void); //Fu
5 void a16pt_tick(void); //Func
6 uint16_t a16pt_get_distance(void);
7 #endif
```

Abbildung 2

Init Aufruf: In der Abbildung 2.6 ist ein Teilausschnitt aus der `a16pt.c` zu sehen. Dieser enthält zwei verschiedene PWM konfigurationen, die auf den Ports `P4_4` und `P4_6` ausgegeben werden.

```

1 void a16pt_init(void)
2 {
3  /*****Externe_Interrupt*****/
4
5  eru_init(eru_port);
6
7  /*****PWM_Init*****/
8
9  XMC_CCU4_Init(CCU41, XMC_CCU4_SLICE_MCMS_ACTION_TRANSFER_PR_CR_PCMP);
10 XMC_CCU4_StartPrescaler(CCU41);
11
12 ccu4_pwm_init(pwm_port_0, cc40, period_1); //P4_4
13 ccu4_pwm_set_duty_cycle(cc40, compare_1);
14
15 ccu4_pwm_init(pwm_port_1, cc42, period_0); //P4_6
16 ccu4_pwm_set_duty_cycle(cc42, compare_0);
17 ...

```

Abbildung 2.6: *Ein Teilausschnitt aus der a16pt.c mit Konfigurationen von Funktionen*

Interrupt Aufruf: In der Abbildung 2.7 :a16pt.c Interrupt Request, werden die für die Entfernungsmessung notwendigen Funktionen und die Interrupt Anweisungen, in dem Fall die IRQ21, abgearbeitet. Außerdem werden die Timer synchron abgeschaltet und aus experimentellen Gründen wurde ein weiterer Impuls generiert um zu beobachten wie sich das Nachschwingen bei einer längeren Kurzschlusszeit an der Ultraschallkapsel verhält. Die IRQ wird auch als Interrupt Request bezeichnet und wird von der Hardware oder von der Software ausgelöst.

```

1
2 /*****Interrupt_Funktionen*****/
3
4 void IRQ_Hdlr_21(void) // Compare Interrupt counter 10
5 {
6
7     // Disable IRQs so we can't be interrupted
8     __disable_irq();
9
10    // Set CCU trigger to low, otherwise ccu counter is restarted
11    XMC_SCU_SetCcuTriggerLow(XMC_SCU_CCU_TRIGGER_CCU41);
12
13    // Stop slice 2
14    XMC_CCU4_SLICE_StopClearTimer(CCU41_CC40);
15
16    // For slice 1 we wait until PWM is run through (to get exactly 10 pwm peaks on
17    P4_4 and P4_6)
18    while(XMC_CCU4_SLICE_GetTimerValue(CCU41_CC42) > compare_1) {
19
20        __NOP();
21    }
22
23    //New pin configuration
24    const XMC_GPIO_CONFIG_t pin_out_config = {
25        .mode           = XMC_GPIO_MODE_OUTPUT_PUSH_PULL,
26        .output_level    = XMC_GPIO_OUTPUT_LEVEL_HIGH,
27    };
28
29    XMC_GPIO_Init(P4_6, &pin_out_config);
30    //Creatw a high impulse
31    for(s=0; s<50; s++)
32    {
33        __NOP();
34    }
35    // Stop slice 0
36    XMC_CCU4_SLICE_StopClearTimer(CCU41_CC42);
37
38    //Pin configuration back to the PWM-Mode
39    const XMC_GPIO_CONFIG_t gpio_out_config1 = {
40        .mode           = XMC_GPIO_MODE_OUTPUT_PUSH_PULL_ALT9,
41        .input_hysteresis = XMC_GPIO_INPUT_HYSTERESIS_STANDARD,
42        .output_level    = XMC_GPIO_OUTPUT_LEVEL_LOW,
43    };
44
45    XMC_GPIO_Init(P4_6, &gpio_out_config1);
46    // Enable IRQs again
47    __enable_irq();
48
49 }

```

Abbildung 2.7: Ein Teilausschnitt von der *a16pt.c* mit einem Interrupt Request

3 Messungen und Auswertung der Ergebnisse

Für die Messungen wurden im Laufe des Projekts zwei Versionen an Prototyp-Platinen entworfen. Im Verlauf der Versuche wurden Veränderungen an den Platinen vorgenommen, um den Betrieb dieser zu verbessern.

3.1 Prototyp 1

Bei dem ersten Prototyp wurden die Sendereinheit und die Empfängereinheit auf getrennten Platinen aufgebaut. So bestand die Möglichkeit, den Senderkreis und den Empfängerkreis getrennt zu untersuchen, ohne dass sich elektrische Signale der beiden Schaltkreise überlagern konnten. Anfangs wurde für den Senderkreis eine komplette H-Brücke als Schalter verwendet. Allerdings war der Aufbau nicht funktional, da ein Beschaltungsfehler vorhanden war. Dieser wurde mit Fädeldraht provisorisch behoben. Dadurch, dass das A5950 eine H-Brücke mit integrierter Kontroll-Logik und eigener Spannungspumpe ist, wurde anhand der Schaltweise der Baugruppe festgestellt, dass diese H-Brücke nicht für die Schaltung einsetzbar ist. Um nicht sofort eine neue Platine in Auftrag geben zu müssen, was einiges an Zeit gekostet hätte, wurde das IC einfach ausgelötet. Danach wurde zunächst ein MOSFET als HIGH-Side an der Stelle eingesetzt. Dadurch ließen sich erste Versuche durchführen, allerdings entsprach das Ergebnis noch nicht den Anforderungen. Deswegen wurde die Beschaltung an dieser Stelle auf eine Halbbrücke erweitert. Damit war die Funktion des Senderkreises für die ersten Versuche und Messungen gegeben.

Der Empfängerkreis war von Beginn an funktional, konnte aber erst mit Inbetriebnahme des Senderkreises richtig getestet werden. Am Empfängerkreis wurden lediglich zu Testzwecken Änderungen an der Filterung, vor dem Verstärker, vorgenommen. Die Filterung wurde in ihrer Dimensionierung verändert, um herauszufinden, wie sich die Qualität des Echo-Signals verbessern lässt.

3.1.1 Senderkreis

Zu erst wurden Signale direkt an der CPU gemessen, um sicher zu stellen, dass die Einstellungen im Programm auch die gewünschten Ausgaben zur Folge haben, und keine Gefährdung der Bauteile entsteht. Um das Signal für die Entfernungsmessung zu generieren wurde der Mikrocontroller so programmiert, dass zehn Impulse mit einer Frequenz von 40 kHz ausgegeben werden. Danach erfolgt eine Pause, um das zurückkehrende Signal abzuwarten und auszuwerten.

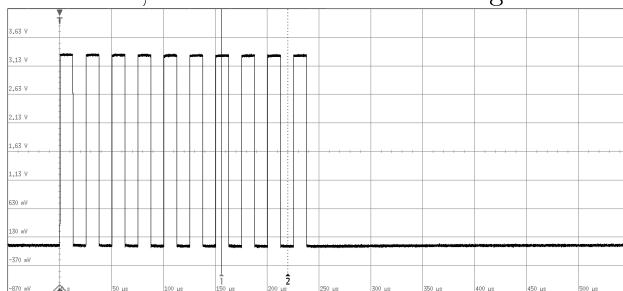


Abbildung 3.1: *PWM-Burst auf 40 kHz Basis an der CPU*

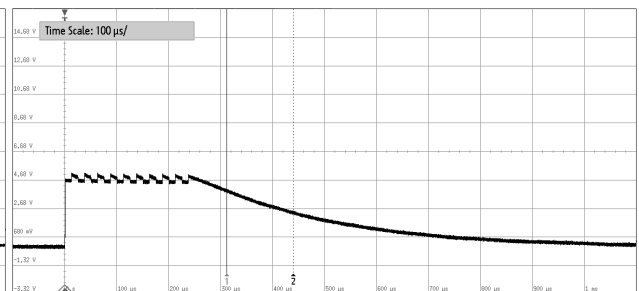


Abbildung 3.2: *PWM Ausgabe über einen HIGH-Side*

In der Abbildung 3.1 ist zu sehen, dass der gewünschte Burst aus zehn Impulsen mit einer Periodendauer von jeweils 25 μ s vom Mikrocontroller generiert wurde. Diese Messung wurde auch vorgenommen, um zu überprüfen, wie sich das Signal durch die eingesetzten Bauteile verändert.

Die Abbildung 3.2 zeigt, wie das Ausgangssignal nach einer HIGH-Side aussieht. So wird zwar im Takt des PWM-Signals geschaltet, allerdings fehlt es an einem Gegenpol, um das Potential in den Schaltphasen wieder auf Null zu ziehen. Dadurch bleibt die Spannung während des Schaltens immer auf einem erhöhten Pegel und sinkt erst nach Ende des PWM-Signals langsam ab. Dadurch kann natürlich keine vernünftige Ausgabe am Lautsprecher erzeugt werden, denn ohne deutliche Potentialunterschiede kann dieser auch nicht in Schwingungen versetzt werden. Der ausgegebene Schalldruck würde nur für kurze Entfernungsmessungen reichen. Das zurückkommende Signal wird von der abklingenden Spannung des HIGH-Side überlagert. Somit ist dieser Aufbau nicht operabel. Um die Spannung nicht nur auf einen HIGH-Pegel, sondern auch auf einen LOW-Pegel schalten zu können wurde danach auf eine Halbbrücke gewechselt. Mit dieser lässt sich der Ausgang, über zwei durch das PWM-Signal gesteuerte MOSFETs, sauber auf HIGH- oder LOW-Pegel schalten. Mit der verwendeten Halbbrücke ergab sich die Abbildung 3.3

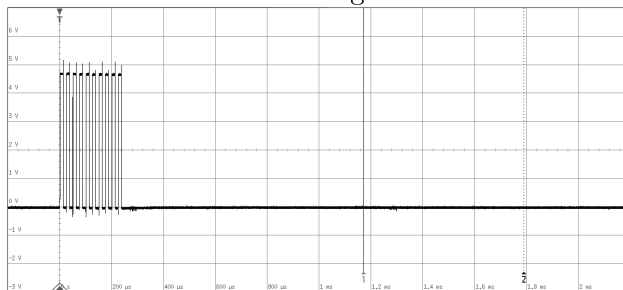


Abbildung 3.3: PWM Ausgabe über eine Halbbrücke

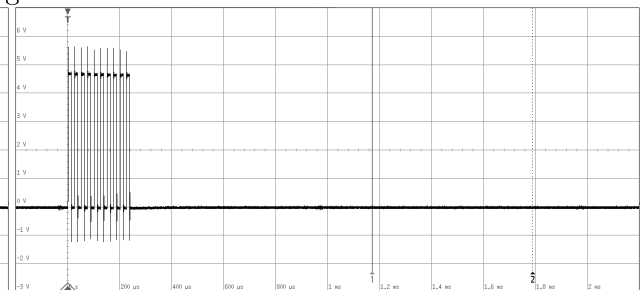


Abbildung 3.4: Ausgabe der PWM an der Ultraschallkapsel

Es zeigt sich, dass das Signal nach der Erweiterung auf eine Halbbrücke wieder wie das von der CPU ausgegebene PWM-Signal 3.1 aussieht. Die Amplitude fällt wie geplant höher aus. Somit kann die Höhe der Amplitude über die Spannungspumpe variiert werden um die Stärke des ausgegebenen Signals zu verändern, ohne die CPU durch die höhere Spannung zu beschädigen. Wie in der Abbildung 3.4 zu entnehmen ist, entstehen durch die angeschlossene Ultraschallkapsel höhere Spannungsimpulse in den Schaltmomenten. Diese Spannungsspitzen, die durch die Ultraschallkapsel entstehen, wurden in den Versuchen vernachlässigt, da keine Gefährdung anderer Bauteile entstand. Das somit generierte Ausgangssignal entsprach den Anforderungen und musste für die Versuche nicht weiter bearbeitet werden.

3.1.2 Empfängerkreis

Die Platinen des Sender- und Empfängerkreises wurden gemeinsam auf einer Halterung montiert, so dass die Ultraschallkapseln zum senden und empfangen der Signale nebeneinander befestigt werden konnten. Ziel war es, durch verschieben eines Hindernisses die Signaländerungen an den Platinen beobachten zu können, ohne die gesamten Messaufbauten bewegen zu müssen. Bei der Aufnahme der Messungen wurde der Empfängerkreis Schritt für Schritt überprüft um zu erfahren, wie sich das empfangene Signal durch die einzelnen Bauteile verändert.

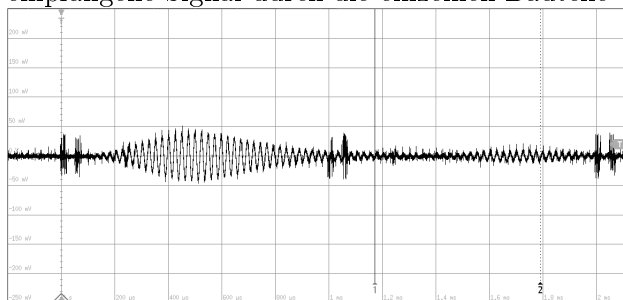


Abbildung 3.5: Signal Empfang

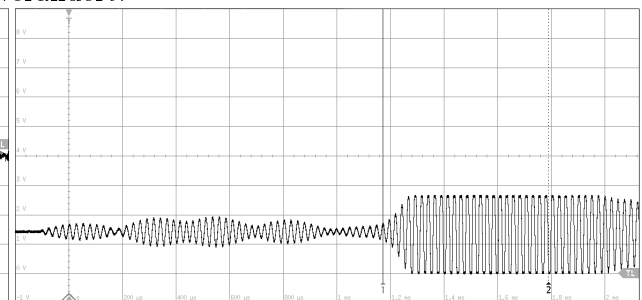


Abbildung 3.6: Signal nach Verstärkung

Die Abbildung 3.5 zeigt das Signal, das direkt am Empfänger zu messen war. Hier sind verschiedene vorerst nicht zuordenbare Signale zu sehen. Allein aus diesem Bild lässt sich daher keine Aussage

zu den einzelnen Signalen machen. Fest steht nur, dass ebenfalls Signale die nicht der gewünschten Frequenz entsprechen, vom Empfänger aufgenommen werden. Dies gilt es natürlich schnellst möglich auszumerzen, um unerwünschte Störungen zu vermeiden. Die Abbildungen 3.6 und 3.7 zeigen den Verlauf des Signals nach der Filterung und Verstärkung in zwei verschiedenen Zeitauflösungen. Dabei entspricht 3.6 den ersten drei Kästchen von 3.7 und dient um darzustellen, dass die Verstärkung eine maximale Aussteuerung von 3,3 V nicht überschreitet.

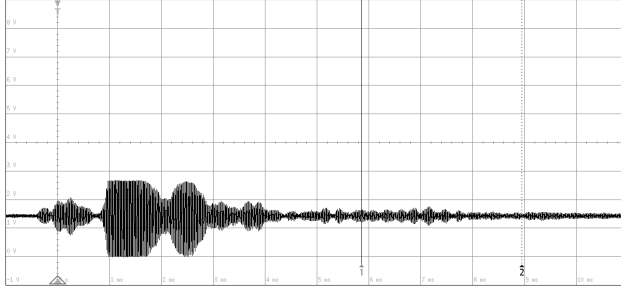


Abbildung 3.7: *Signal nach Verstärkung2*

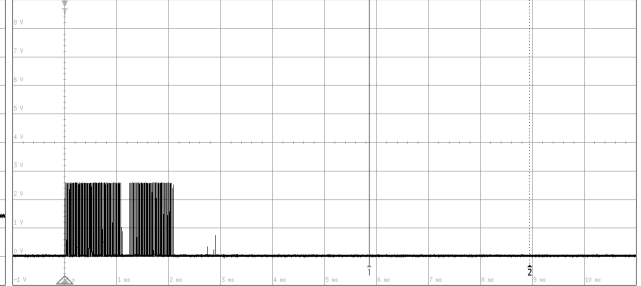


Abbildung 3.8: *Signal nach Komparator*

Nach dem das Signal den Komparator passiert hat, ergibt sich das Bild wie in Abbildung 3.8 zu sehen ist. Bei einem Vergleich mit dem Signal nach der Verstärkung 3.7 wird sichtbar, dass der Komparator nur Signale, die über seinem Schwellwert von 1,8 V liegen, durchschaltet. Die Aufteilung in zwei Signalblöcke in den Abbildungen kommt daher, dass der erste Block das Signal der Sender-Kapsel ist, das direkt beim Senden seitlich auf die Empfänger-Kapsel abgestrahlt wurde. Der zweite Block ist bereits das Echo, das vom 20 cm entfernten Hindernis zurückgeworfen wurde. Anhand dieser Ergebnisse kann festgehalten werden, dass die einfachere Version des Ultraschall-Entfernungsmessers durchaus simpel umzusetzen ist. Es fehlt nur noch ein Programm, das die Zeit zum Eintreffen des Echo-Signals in einen Abstand zum Hindernis konvertiert. Besagtes Programm wurde nicht für diese Prototypversion erstellt, da der Anspruch bestand, den Betrieb nicht nur über eine Platine, sondern auch noch über eine Ultraschallkapsel ablaufen zu lassen.

3.2 Prototyp 2

Nachdem mit der ersten Prototyp-Version bereits Versuche an der Elektronik durchgeführt wurden und auch erste Messungen mit einem beweglichen Hindernis auswertbare Ergebnisse brachten, wurde eine zweite Prototyp-Version entworfen. Bei der zweiten Prototyp-Version wurden der Sender- und der Empfängerkreis auf einer Platine aufgebaut und es wurde nur noch eine Ultraschallkapsel für beide Anwendungen vorgesehen. Dadurch wurden für den fehlerfreien Betrieb drei und nicht zwei Schaltzustände der Halbbrücke benötigt. Deswegen wurde die Halbbrücke der ersten Prototyp-Version durch eine voll gesteuerte Halbbrücke ersetzt. Nicht nur sollte die Ultraschallkapsel mit HIGH-, oder LOW-Signal ansteuerbar sein, auch ein dritter potentialfreier Zustand war nötig, damit die Echo-Signale auch empfangen werden konnten. Um bei diesem Aufbau einen fehlerfreien Betrieb der verwendeten voll gesteuerten Halbbrücke sicherzustellen wurden durch den Mikrocontroller zwei getrennte PWM-Signale generiert die wie in Abbildung 3.9 zu sehen ist durch Lücken getrennt sind. Verzögerungen im Schaltbetrieb der Halbleiter konnten daher, keine Kurzschlüsse mehr verursachen. Außerdem konnten durch den neuen Platinenentwurf die ab der ersten Prototyp-Version vorgenommenen Änderungen direkt in den Schaltplan übernommen werden. Dadurch ließ sich die Störanfälligkeit durch empfindliche Drahtbrücken deutlich reduzieren. Bedingt durch anfängliche Befürchtungen wurden für die MOSFETs der Halbbrücke anfangs jeweils ein weiteres MOSFET vorgeschaltet. Dieser Zusatz stellte sich als unnötig heraus, da durch eines der zusätzlichen Bauteile eines der PWM-Signale invertiert wurde. Somit wurde das überflüssige MOSFET entfernt. Durch zwei Drahtbrücken konnte die komplette Schaltung nach Entlarvung eines weiteren Verdrahtungsfehlers, in Betrieb genommen werden.

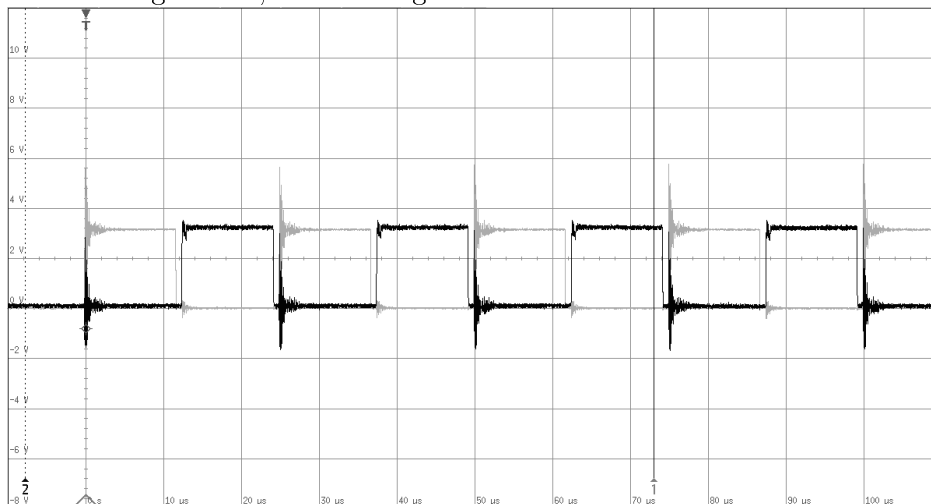


Abbildung 3.9: *Verlauf der zwei generierten PWMs für den Betrieb der voll gesteuerten Halbbrücke*

Nachdem dieser Betrieb sichergestellt war, wurden Messungen am Verstärker (obere Linie), und am Komparator (untere Linie) vorgenommen. Dabei wurde die Verstärkung so eingestellt, dass unerwünschte Störungen nicht vom Komparator weitergegeben wurden. Die Spannung für den Sendebetrieb wurde für die Versuche zwischen 5V und 20V variiert, um betrachten zu können, wie sich das auf die Reichweite und Genauigkeit der Messungen auswirkt. Als Hindernis wurde bei allen Versuchen eine glatte Holzplatte der Maße 50x64cm verwendet und in einem Abstand von ein bis fünf Metern von der Ultraschallkapsel entfernt aufgestellt. In den Abbildungen 3.10 bis 3.13 sind die Ergebnisse einer Messreihe mit einer Spannung von 5 V für den Sendebetrieb dargestellt. Die Ansicht wurde so eingestellt, dass zwei Sendeimpulse zu sehen sind. Dadurch wird deutlicher, welches die Sende Impulse sind, und welches die von der Entfernung abhängigen Echos sind.

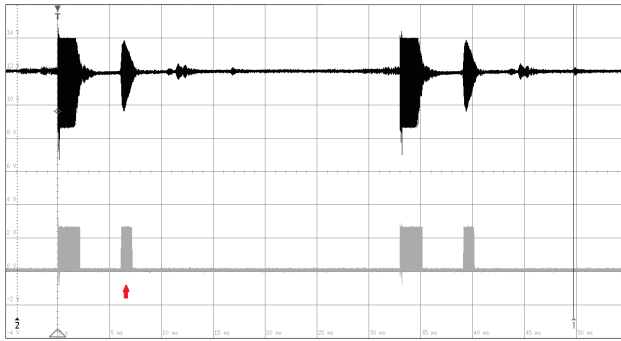


Abbildung 3.10: *Signalverlauf bei 5 V auf 1 m Abstand*

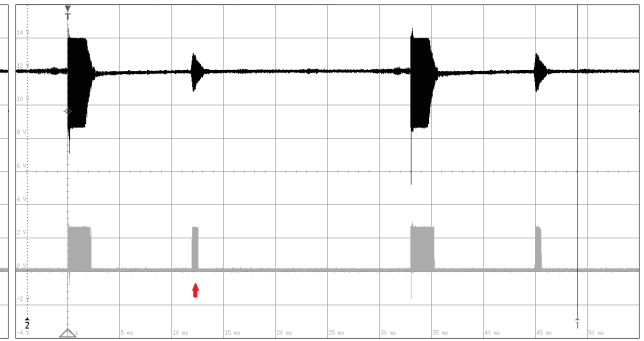


Abbildung 3.11: *Signalverlauf bei 5 V auf 2 m Abstand*

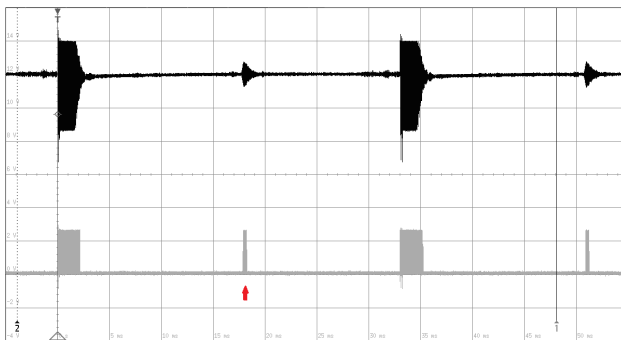


Abbildung 3.12: *Signalverlauf bei 5 V auf 3 m Abstand*

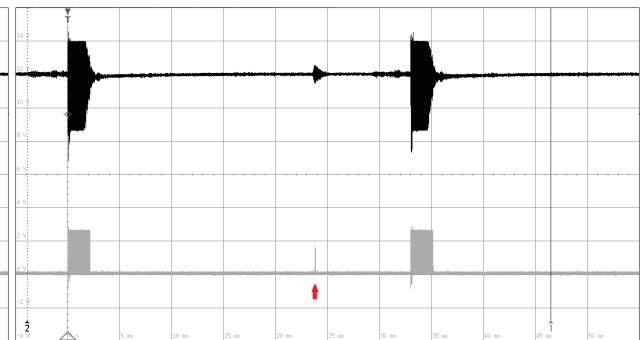


Abbildung 3.13: *Signalverlauf bei 5 V auf 4 m Abstand*

Bei den Abbildungen ist zu sehen, dass das Echo-Signal mit zunehmender Entfernung immer schwächer wird. Bei einer Entfernung von vier Metern (Abbildung 3.13) wird das Echo-Signal so schwach, dass die Signalstärke nach dem Komparator nicht mehr für eine eindeutige Auswertung über den Mikrocontroller ausreicht. Nachfolgend sind die Abbildungen einer Messreihe mit verschiedenen Spannungseinstellungen für den Sendebetrieb zu sehen. Anhand dieser Messreihe soll dargestellt werden, welchen Einfluss die eingestellte Spannung im Sendebetrieb auf die Reichweite des Ultraschallsignals hat. Für die Darstellung wurden die Messungen bei 5 Meter Abstand ausgewählt.

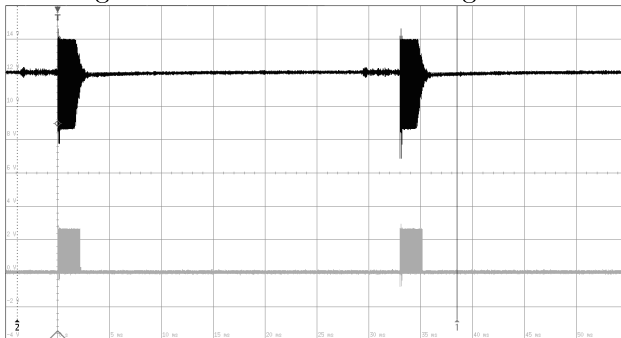


Abbildung 3.14: *Signalverlauf bei 5 V auf 5 m Abstand*

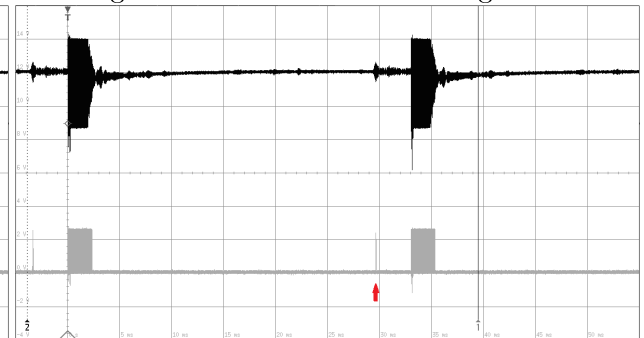


Abbildung 3.15: *Signalverlauf bei 10 V auf 5 m Abstand*

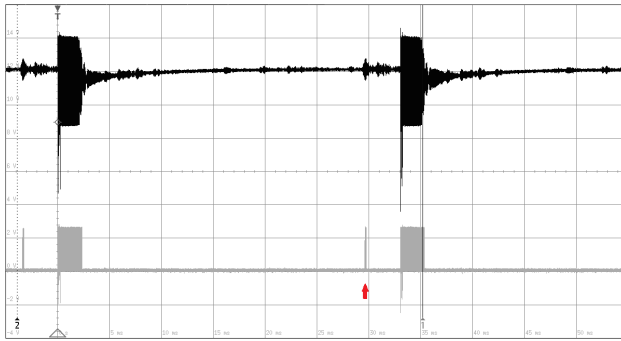


Abbildung 3.16: *Signalverlauf bei 15 V auf 5 m Abstand*

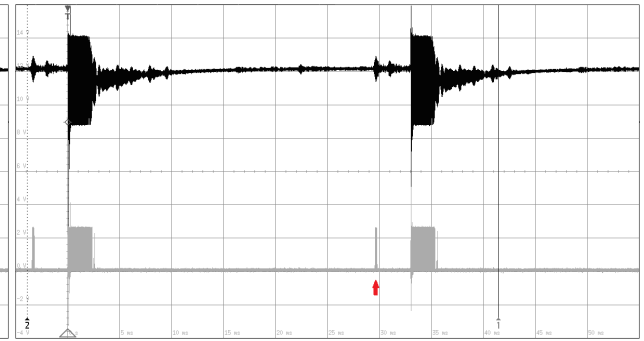


Abbildung 3.17: *Signalverlauf bei 20 V auf 5 m Abstand*

Bei Vergleich der Abbildungen 3.14 und 3.15 mit den Abbildungen 3.16 und 3.17 ist zu sehen, dass bei einem Abstand von 5 Metern erst bei einer Sendespannung von über 10 V, auch am Komparator ein über den Mikrocontroller auswertbares Signal vorhanden ist.

In der nachfolgenden Tabelle 3.1 wurden die Zeitabstände vom ersten Impuls des gesendeten Signals, bis zum ersten Impuls des Echo-Signals aufgetragen. Anhand der Schallgeschwindigkeit die Entfernung, die der Schall zurückgelegt hat berechnet. Dazu wurde noch die Abweichung der berechneten Entfernung von der eingestellten Entfernung angegeben.

Tabelle 3.1: *Entfernungsmessung mit Abweichung bei 20 V Sendespannung*

Entfernung [m]	Zeit bis Anfang Echo [ms]	Errechnete Entfernung [m]	Abweichung [cm]
1	6,07	1,0416	4,16
1,5	8,97	1,5392	3,92
2	11,92	2,0454	4,54
2,5	14,8	2,5396	3,96
3	17,75	3,0459	4,59
3,5	20,65	3,5435	4,35
4	23,56	4,0428	4,28
4,5	26,49	4,5456	4,56
5	29,46	5,0553	5,53

Wie befürchtet sind bei der errechneten Entfernung Abweichungen im Bereich weniger Zentimeter aufgetreten. Bei Betrachtung der Abweichungen wird allerdings deutlich, dass die Werte bis auf einen, alle im Bereich von 4 cm bis 4,5 cm liegen. Ähnliche Abweichungen waren auch bei den anderen Messreihen zu beobachten. Somit ließe sich die Abweichung durch einen Korrekturwert auf ein Minimum reduzieren und würde einen Zentimeter nur noch selten überschreiten. Der erste Impuls des Echo-Signals als Referenz für die Entfernungsberechnung zu nutzen ist. Die anfänglichen Überlegungen, den letzten Impuls des Echo-Signals, oder einen Mittelwert aus allen empfangenen Impulsen zu verwenden beinhaltet ein höheres Fehlerpotential. Die Dauer des Echo-Signals kann durch niederfrequente Störgeräusche deutlich verlängert werden. Dies ließ sich bei einem der Tests beobachten, als im Hintergrund ein Lasercutter betrieben wurde. Dabei entstanden entgegen der Befürchtung keine Störsignale, stattdessen war die Signalintensität des Echo-Signals deutlich höher als bei Versuchen in einer stillen Umgebung.

3.3 Fazit aus den Ergebnissen für den Auftraggeber

Aus den Versuchen und Messungen lassen sich mehrere Aussagen treffen.

Als erstes, eine Ultraschall-Entfernungsmessung ist mit wenigen Bauteilen, sowohl als Zwei-Kapsel-Variante, als auch als Ein-Kapsel-Variante durchführbar. Bei der Ein-Kapsel-Variante ist darauf zu achten, dass der Verstärker eine ausreichende Spannungsfestigkeit besitzt, um nicht durch das

Sendersignal gestört zu werden. Auch ist wichtig, dass bei der Erzeugung des PW-Modulierten Ausgangssignals die MOSFETs so angesteuert werden, dass den MOSFETs zwischen den Schaltsignalen genug Zeit gegeben wird, um die Schaltzustände zu erreichen. Vergleiche Abbildung 3.9. Dadurch lassen sich Kurzschlüsse an dieser Stelle vermeiden.

Eine Auswertung des Echo-Signals ist prinzipiell sowohl digital, als auch analog möglich. bei der digitalen Auswertung muss die Zeit erfasst werden, die zwischen dem Senden des PWM-Signals und dem Empfangen des Echo-Signals vergeht. Die errechnete Strecke ist zu halbieren, da die vergangene Zeit sowohl den Hin-, als auch den Rückweg beinhaltet. Bei der analogen Auswertung besteht zwar die Möglichkeit über einen Frequenzvergleich auch Signale mit kleinerer Echo-Amplitude zu erkennen und auszuwerten, allerdings beinhaltet dieses Vorgehen einen deutlich höheren Programieraufwand.

Bei der Berechnung der Zeiten muss berücksichtigt werden, dass das eingehende Echo-Signal die Ultraschallkapsel langsam in Schwingungen versetzt. Die ersten eintreffenden Schwingungen eines PWM-Signals erzeugen kleinere Spannungssignale als die darauf folgenden. Außerdem schwingt die Ultraschallkapsel auch nach Ende des eingehenden Signals noch etwas nach, was zur Folge hat, dass das analoge Abbild des gesendeten PWM-Signals leicht versetzt und etwas verlängert wirkt. All diese Faktoren müssen für eine genauere Berechnung der Strecke, die das Signal zurück gelegt hat, berücksichtigt werden. Die Tatsache, dass das empfangene Signal an der Ultraschallkapsel verändert wird, ist eine absolut exakte Messung nicht möglich. Eine Beschränkung des Fehlers auf einzelne Zentimeter ist realisierbar.

Mit den aktuellen, korrigierten Schaltplänen ist der Aufbau eines Ultraschall-Entfernungsmessers durchführbar. Was noch aussteht, ist eine Festlegung auf definierte Werte für die Verstärkung und die Sendespannung, um die beiden Potentiometer durch Widerstände zu ersetzen. Denn schließlich soll ein Produkt nicht erst eingestellt werden, sondern nach einschalten funktionieren.

4 Reflektion über den Projektablauf

Nach Erhalt der Aufgabenbeschreibung war das Bild, das man sich von den bevorstehenden Aufgaben machte, doch sehr anders, als die Arbeiten später aussahen. So war mit der Programmierung, auf Grund des doch deutlich komplexeren Mikrocontrollers, ein enormer Lernaufwandt verbunden. Die Recherche der Hardwarebauteile und das Erstellen der Schaltpläne und Platinen hingegen verlief ähnlich den Erwartungen.

Eddy: Die Einarbeitung in die Tinkerforge verwendeten Programme, KiCad und GitHub, erforderten genausoviel Aufmerksamkeit wie die Programmierung trotz der Einarbeitungsphasen war stest konstant ein Lernerfolg zu verbuchen und konnten bis zum ende des Projektes uns neues Wissen sowie Fähigkeiten erwerben.

5 Anhänge

Abbildungen der Messreihen mit verschiedenen Spannungen bei Abständen von ein bis fünf Metern.

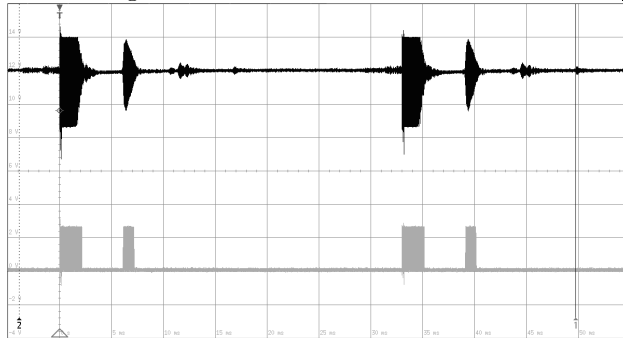


Abbildung 5.1: *Signalverlauf bei 5V auf 1m Abstand*

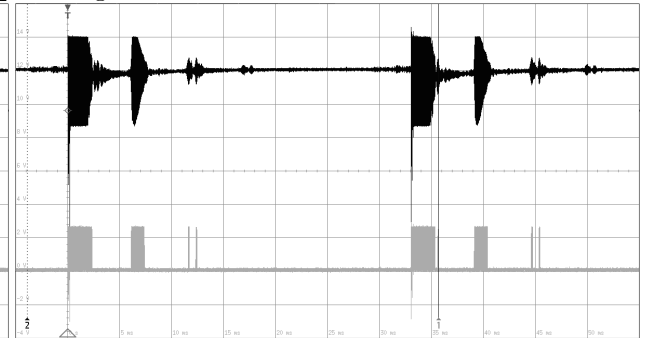


Abbildung 5.2: *Signalverlauf bei 10V auf 1m Abstand*

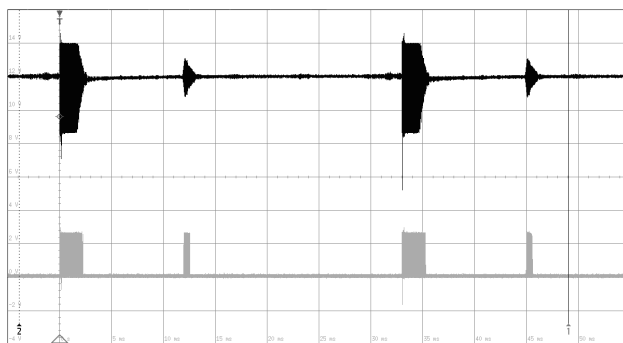


Abbildung 5.3: *Signalverlauf bei 5V auf 2m Abstand*

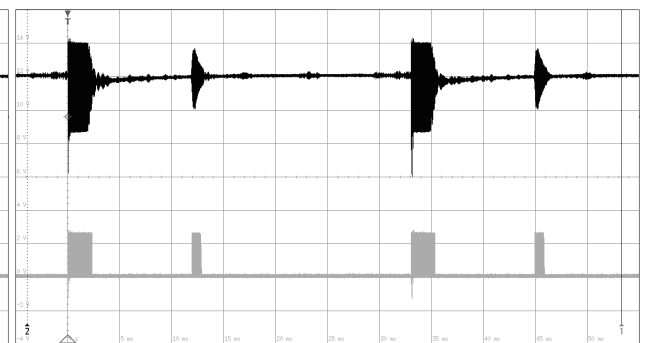


Abbildung 5.4: *Signalverlauf bei 10V auf 2m Abstand*

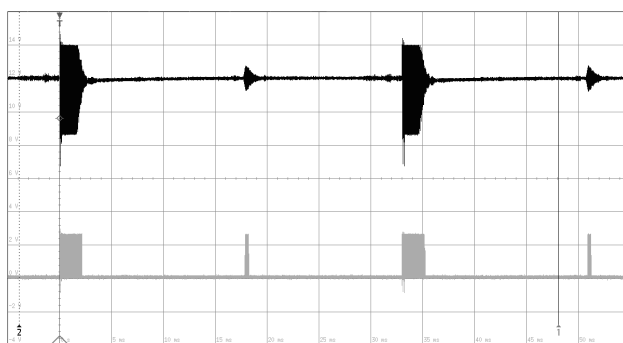


Abbildung 5.5: *Signalverlauf bei 5V auf 3m Abstand*

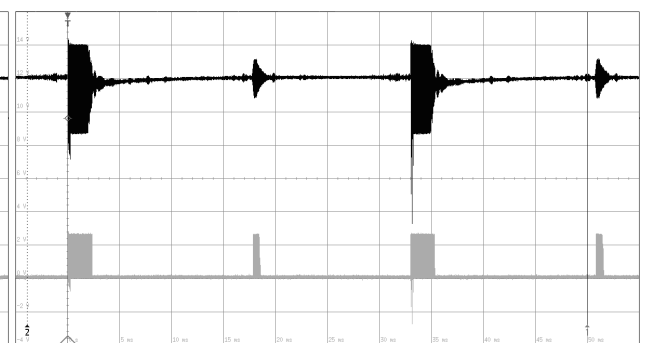


Abbildung 5.6: *Signalverlauf bei 10V auf 3m Abstand*

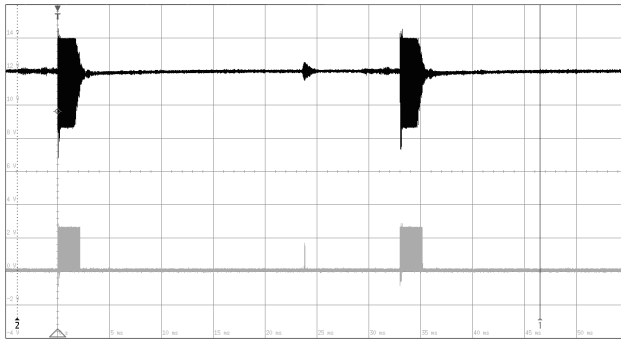


Abbildung 5.7: *Signalverlauf bei 5V auf 4m Abstand*

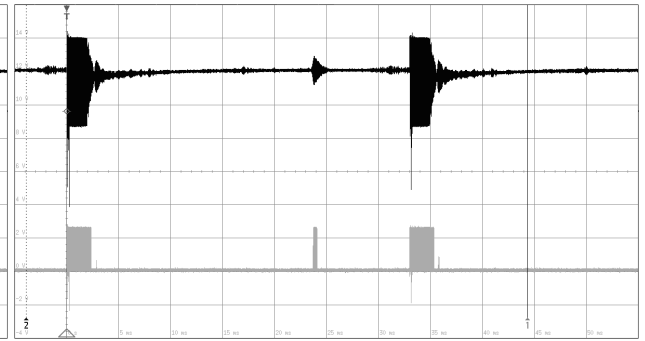


Abbildung 5.8: *Signalverlauf bei 10V auf 4m Abstand*

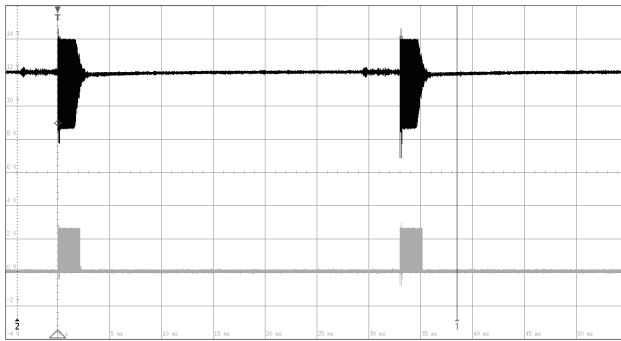


Abbildung 5.9: *Signalverlauf bei 5V auf 5m Abstand*

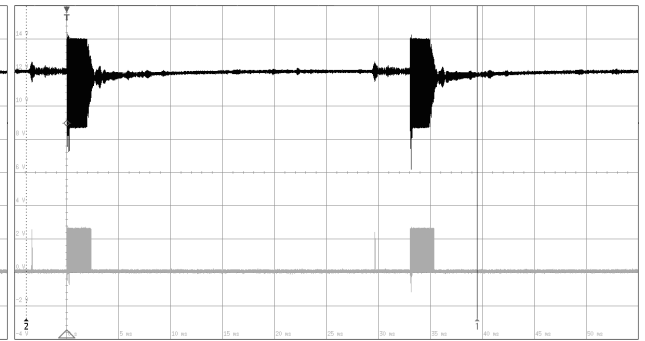


Abbildung 5.10: *Signalverlauf bei 10V auf 5m Abstand*

Entfernung [m]	Zeit bis Anfang Echo [ms]	Zeit bis Ende Echo [ms]	Errechnete Entfernung Anfang [m]	Errechnete Entfernung Ende [m]
1	6,11	7,23	1,0485	1,2407
1,5	9,02	9,89	1,5478	1,6971
2	11,97	12,76	2,0541	2,1896
2,5	14,88	15,5	2,5534	2,659
3	17,84	18,2	3,06134	3,1231
3,5	20,8	21,11	3,569	3,6225
4	23,71	23,81	4,0686	4,0858
4,5	26,61	26,73	4,5663	4,5869
5	29,59	29,68	5,0776	5,0961

Tabelle 5.1: *Entfernungsmessung bei 5V Sendespannung*

Entfernung [m]	Zeit bis Anfang Echo [ms]	Zeit bis Ende Echo [ms]	Errechnete Entfernung Anfang [m]	Errechnete Entfernung Ende [m]
1	6,07	7,37	1,0416	1,2647
1,5	8,99	10,02	1,5427	1,7194
2	11,94	12,9	2,0489	2,2136
2,5	14,83	15,7	2,5448	2,6941
3	17,8	18,48	3,0545	3,1712
3,5	20,7	21,36	3,5521	3,6654
4	23,62	24	4,0532	4,1184
4,5	26,55	26,9	4,5560	4,6160
5	29,5	29,9	5,0622	5,1308

Tabelle 5.2: *Entfernungsmessung bei 10V Sendespannung*

Abbildungsverzeichnis

2.1	Blockschaltbild des Ultraschall-Entfernungsmessers	4
2.2	Schaltplan des beschalteten Hochsetzstellers	6
2.3	Schaltplan zweier Operationsverstärker mit einer vorgelagerten Filterung als Empfänger-schaltung	8
2.4	Quellcode der main.c des Distance US	9
2.5	Inhalt der Headerdatei a16pt.h	9
2.6	Ein Teilausschnitt aus der a16pt.c mit Konfigurationen von Funktionen	10
2.7	Ein Teilausschnitt von der a16pt.c mit einem Interrupt Request	11
3.1	PWM-Burst auf 40 kHz Basis an der CPU	12
3.2	PWM Ausgabe über einen HIGH-Side	12
3.3	PWM Ausgabe über eine Halbbrücke	13
3.4	Ausgabe der PWM an der Ultraschallkapsel	13
3.5	Signal Empfang	13
3.6	Signal nach Verstärkung	13
3.7	Signal nach Verstärkung2	14
3.8	Signal nach Komparator	14
3.9	Verlauf der zwei generierten PWMs für den Betrieb der voll gesteuerte Halbbrücke	15
3.10	Signalverlauf bei 5 V auf 1 m Abstand	16
3.11	Signalverlauf bei 5 V auf 2 m Abstand	16
3.12	Signalverlauf bei 5 V auf 3 m Abstand	16
3.13	Signalverlauf bei 5 V auf 4 m Abstand	16
3.14	Signalverlauf bei 5 V auf 5 m Abstand	16
3.15	Signalverlauf bei 10 V auf 5 m Abstand	16
3.16	Signalverlauf bei 15 V auf 5 m Abstand	17
3.17	Signalverlauf bei 20 V auf 5 m Abstand	17
5.1	Signalverlauf bei 5V auf 1m Abstand	20
5.2	Signalverlauf bei 10V auf 1m Abstand	20
5.3	Signalverlauf bei 5V auf 2m Abstand	20
5.4	Signalverlauf bei 10V auf 2m Abstand	20
5.5	Signalverlauf bei 5V auf 3m Abstand	20
5.6	Signalverlauf bei 10V auf 3m Abstand	20
5.7	Signalverlauf bei 5V auf 4m Abstand	21
5.8	Signalverlauf bei 10V auf 4m Abstand	21
5.9	Signalverlauf bei 5V auf 5m Abstand	21
5.10	Signalverlauf bei 10V auf 5m Abstand	21

Tabellenverzeichnis

3.1	Entfernungsmessung mit Abweichung bei 20 V Sendespannung	17
5.1	Entfernungsmessung bei 5V Sendespannung	22
5.2	Entfernungsmessung bei 10V Sendespannung	22