

# Entwicklung und testen eines Ultraschall-Entfernungsmessers als Vorbereitung eines Produktentwurfes

---

## Projektarbeit

erstellt an der  
Fachschule für Technik des Carl-Severing-Berufskollegs  
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, 14. 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

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Lastenheft . . . . .	1
1.1.1	Über Tinkerforge . . . . .	1
1.1.2	Motivation . . . . .	1
1.1.3	Aufgabenbeschreibung . . . . .	1
1.2	Management des Projektes . . . . .	2
1.2.1	Trello . . . . .	2
1.2.2	Github . . . . .	2
1.3	Open Source CAD-Programm Paket zur Erstellung von elektronischen Leiterplatten	3
1.3.1	Schaltplan . . . . .	3
1.3.2	Platinenlayout . . . . .	3
<b>2</b>	<b>Vorbereitung</b>	<b>4</b>
2.1	Recherche der Funktionsweise . . . . .	4
2.1.1	Sender . . . . .	4
2.1.2	Empfänger . . . . .	4
2.2	Verwendete sowie Dimensionierung der Bauteile . . . . .	5
2.2.1	Sender . . . . .	5
2.2.2	Umschaltung . . . . .	5
2.2.3	Empfänger . . . . .	5
2.2.4	Hochsetzsteller . . . . .	6
2.2.5	Mikroprozessor . . . . .	6
2.3	Entwicklung der Software zum Betrieb des Prototypen . . . . .	6
2.3.1	Benötigte Register . . . . .	6
2.3.2	Quellcodeentwurf . . . . .	7
<b>3</b>	<b>Messungen und Auswertung der Ergebnisse</b>	<b>9</b>
3.1	Prototyp 1 . . . . .	9
3.1.1	Senderkreis . . . . .	9
3.1.2	Empfängerkreis . . . . .	10
3.2	Prototyp 2 . . . . .	12
3.3	Ergebnisse der Forschung . . . . .	14
3.4	Fazit aus den Ergebnissen für denAuftraggeber . . . . .	14
<b>4</b>	<b>Reflektion über den Projektablauf</b>	<b>15</b>
<b>5</b>	<b>Anhänge</b>	<b>16</b>
5.1	Schaltpläne . . . . .	16
5.2	Platinenlayout . . . . .	16
5.3	Quellcode . . . . .	16

# 1 Einleitung

## 1.1 Lastenheft

### 1.1.1 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.

### 1.1.2 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 aber im wesentlichen 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.

### 1.1.3 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 und finanziell realisierbar sind, um ein eigenes Produkt zu erstellen. Sollte im Anschluss noch die Möglichkeit bestehen, sind die Ergebnisse in ein serienreifes Modul umzusetzen.

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 und zum anderen 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.

## 1.2 Management des Projektes

### 1.2.1 Trello

Zur zeitlichen Planung und Übersicht des Ablaufes wurde auf das Onlinetool Trello zurückgegriffen. Dieses ist ein kostenfreies, web basiertes 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 wunderbar abgleichen, zu welchen Zeitpunkten die einzelnen Aufgaben abgeschlossen wurden.

### 1.2.2 Github

Bei Github handelt es sich um einen webbasierten Online-Dienst, der Server für Software- 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 auf mehrere Abschnitte aufgeteilt werden, damit mehrere Personen unabhängig voneinander daran arbeiten können. Nach der Bearbeitung können die Programmteile 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 reizvoll, da die Nutzer hier schnell an alle Dateien ran kommen.

## 1.3 Open Source CAD-Programm Paket zur Erstellung von elektronischen Leiterplatten

Das Open Source CAD-Programm KiCAD ist eine Anwendung zum Erstellen von Schaltplänen und elektronischen Leiterplatten. Hier lassen sich auf einfache Weise Schaltpläne erstellen und nach einer Prüfung auf fehlerfreie Verdrahtung zur vereinfachten Platinenerstellung nutzen.

### 1.3.1 Schaltplan

Zur Signalerzeugung wurde ein Hochsetzsteller eingesetzt, um eine höhere Spannung erzeugen zu können, als auf dem System zur Verfügung steht. So höhere Amplituden auf den Sender gegeben werden, um eine höhere Signalstärke zu erreichen. Danach wird das Signal auf eine H-Brücke geleitet, diese dient dazu aus dem Gleichsignal ein Wechselsignal mit einer Frequenz von 40kHz zu generieren. Dieses Wechselsignal wird auf den Sender gegeben um das Spannungssignal in Schallwellen umzuwandeln.

Wird ein zurückkommendes Signal empfangen, so wandelt der Empfänger die Schallwellen in ein Sinussignal um. Dieses Signal wird auf einen nicht invertierenden Verstärker geleitet, um die Signal-Amplitude zu verstärken und anschließend durch einen Komperator in ein Rechteckiges(digitales) Signal umgewandelt. Dieses Signal wird im Prozessor verarbeitet, um aus der Zeit zwischen senden und empfangen des Signals in die Entfernung zum betreffenden Objekt umzuwandeln.

Beim Schaltplanentwurf gilt es auch auf gewisse Physikalische Eigenschaften von Leitungen und Bauteilen zu achten. So sollten bei Anschluss längerer Leitungen, Condensatoren zum ausfiltern eingefangener Funksignale und anderer EMV-Belastungen, angebracht werden. Auch benötigen Bauteile wie Platinen Kondensatoren an der Spannungsversorgung, um auch kleinste Schwankungen dieser zu vermeiden.

### 1.3.2 Platinenlayout

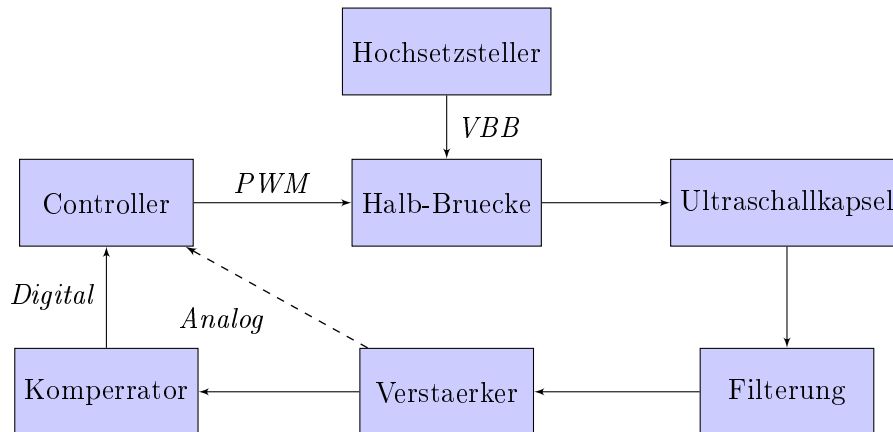
Bei dem Entwurf eines Platinenlayouts gibt es viele Möglichkeiten ein Ergebniss zu erzielen. So können alle Bauteile so angeordnet werden, dass alle parallelen Bauteile sauber nebeneinander aufgereiht werden, und die in Reihe dazu liegenden Bauteile auch darunter angeordnet sind. So sähe die Platine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

Auch können die Bauteile wie im Schaltplan in Gruppen zusammen gelegt werden und der Schaltplan auf der Platine 1:1 nachgebildet werden. 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 ist die Leitung auszulegen und sie sollte auch möglichst kurz gehalten werden 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 die Bildung von größeren Schleifen Antennen zu erzeugen. Die GND Schicht sollte so wenig wie möglich unterbrochen werden, vorallem sind Unterbrechungen quer zur Stromflussrichtung zu vermeiden. Zusätzlich ist zu beachten, dass Kondensatoren meistens zur Verringerung von Störungen nahe digitalen Bauteilen angebracht werden sollten. Die optimale Platzierung ist direkt an den Pinns, so dass die Leiterbahn mit einem höheren Querschnitt auf den Kondensator geht, und dann mit leicht verringertem Querschnitt direkt auf die Pinne des IC-Bauteils verläuft.

## 2 Vorbereitung

### 2.1 Recherche der Funktionsweise



#### 2.1.1 Sender

Der Sender soll Schallwellen im Ultraschallbereich aussenden, die einen ausreichenden Schalldruck haben, um von Hindernissen, die bis zu mehrere Meter entfernt sind ein Echo erhalten zu können. Dafür muss die Lautsprecherkapsel mit einem Sinusähnlichen Signal angesteuert werden, dessen Amplitude angemessen hoch ist, um den gewünschten Schalldruck zu erzeugen. Mit dem Mikrocontroller kann ein Rechtecksignal erzeugt werden, bei dem die Pulsweite variierbar ist (PWM). Der Nachteil ist, dass der Mikrocontroller nur auf seine Spannungsebene von 3,3V begrenzt ist und durch höhere Spannungen zerstört würde. Um höhere Spannungen an der Lautsprecherkapsel realisieren zu können, muss ein weiteres Bauteil dazwischen geschaltet werden. Dieses zusätzliche Bauteil dient als Trennung zwischen den 3,3V des Mikrocontrollers und der höheren Spannungsebene der Lautsprecherkapsel. Dabei wird darauf zu achten sein, dass dieser Schalter schnell genug arbeitet, um die 40kHz (Frequenz von Ultraschall) auch sauber schalten zu können.

#### 2.1.2 Empfänger

Im Empfangsbetrieb werden die zurückkommenden Schallsignale, die auf die Lautsprecherkapsel treffen in Sinusförmige Spannungssignale umgewandelt. Die Amplitude dieser Signale 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 macht es auch Sinn, das eingehende analoge Signal in ein Digitales Signal umzuwandeln, dies kann einiges an Verarbeitungsaufwand einsparen. Zusätzlich ist das eingehende Signal zu filtern, da die Lautsprecherkapsel nicht ausschließlich Ultraschallsignale aufnimmt. Mit der Zeit, die zwischen dem gesendeten Signal und dem empfangenen Signal vergeht, muss letztendlich der Abstand zwischen dem Sensor und dem Hinderniss berechnet werden.

## 2.2 Verwendete sowie Dimensionierung der Bauteile

### 2.2.1 Sender

Für den Sender (und empfänger) wurden auf Piezomodulen basierende Kapsellautsprecher verwendet. Dabei wurden für den Prototypen 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. Wichtig sind bei der Überprüfung das nachschwingen der Kapsel, nach Ende des zu sendenden Signals, die Empfindlichkeit auf eingehende Störfrequenzen und die zu erreichende Reichweite mit verschiedenen Signalstärken.

### 2.2.2 Umschaltung

Da der Sende-, und der Empfangsbetrieb über eine Kapsel laufen sollen und die beiden Betriebsarten mit verschiedenen Spannungspegeln arbeiten, ist eine Umschaltung zwischen den Anschlüssen notwendig.

Zu Beginn wurde eine H-Brücke als optionale Lösung des Problems ins Auge genommen. Dafür wurde das IC A5950 verwendet. Wie in Abbildung ?? zu sehen ist, kann die H-Brücke eine angeschlossene Last mit einer Wechselspannung versorgen, deren Frequenz über das Signal am Anschluss PHASE vorgegeben werden kann.

Später wurde auf eine Halbbrücke umgeschwenkt, bei der die MOSFETs einzeln ansteuerbar waren. Diese Halbbrücke bietet den Vorteil, dass bei der Ansteuerung der einzelnen MOSFETs gezielt Zwischenzeiten zwischen den Schaltvorgängen eingepflegt werden können, damit ein Abstand zwischen den Schaltflanken entsteht.

### 2.2.3 Empfänger

Die Abbildung ?? zeigt die Empfängerschaltung. Durch diese Verschaltung von Operationsverstärkern(OPVs) wird das ankommende Sinusförmige Signal verstärkt und in ein digitales Signal umgewandelt. Die Schaltung wurde mit einem Hochpassfilter (CR Glied) bestückt bestehend aus C12 und R5 um unerwünschte Signalanteile mit Frequenzen, die unter 40kHz liegen, zu unterdrücken. Der Widerstand wurde nach der e24 Reihe, anhand der folgende Berechnung, ausgewählt:

$$C12 = \frac{1}{2 * \pi * f_g * R} \Rightarrow \frac{1}{2 * \pi * 40kHz * 100\Omega} \approx 40pF$$

Die Kapazität des Kondensators C12 wurde an die Grenzfrequenz von 40 kHz und den Widerstand angepasst.

Für die Verstärkung der Amplitude so wie der Umwandlung des analogen Signals in ein Rechtecksignal mit 40 KHz dienen die Operationsverstärker des Bausteines TLC272. Die Versorgungsspannung der OPV's von 3,3V wird durch den Kondensator C16 (EMV Störfilter) stabilisiert.

Für die Verstärkung der Amplitude ist der Operationsverstärker TLC272 U2B als nicht invertierender Verstärker geschaltet. Wenn eine Gleichspannung anliegt, wirkt der Kondensator (C10) in der Operationsverstärkerschaltung als Impedanzwandler, also mit einer Verstärkung von eins, geht nun die Eingangsfrequenz hoch, nimmt der Widerstand des Kondensators (C10) ab, somit beginnt der Operationsverstärker auch zu verstärken, und zwar mit zunehmender Verstärkung, bis irgendwann die Impedanz des Kondensators vernachlässigt werden kann und die Verstärkung nur noch durch das Verhältnis der Widerstände beeinflusst wird, R6 ist zudem notwendig um das schwingen der Amplitude zu verhindern, somit kann die Verstärkung mit folgender Formel berechnet werden:

$$V_u = R6 + R8 + \frac{R12}{R6}.$$

Die Z-Diode D2 ist für die Spannungsstabilisierung und als Sicherung da.

Für die Umwandlung des Analogen Signales in ein Digitales wurde der Operationsverstärker TLC272 U2C als Komparator geschaltet. Beim auftreten von Differenzen zwischen den eingangs Signalen, wechselt der Ausgang des Komparators zwischen Low (0 Volt) auf High (3,3 Volt).



## 2.2.4 Hochsetzsteller

Der Hochsetzsteller dient dazu, aus den 5V Versorgungsspannung eine (für die Versuche variable) höhere Spannung für den Sendebetrieb zu schaffen. So kann der Schalldruck der ausgegeben wird erhöht werden (größere Reichweite/größeres Rücksignal)

Die Funktionsweise des Hochsetzstellers (Spannungspumpe/Aufwärtswandler/Aufwärtsregler) ist relativ simpel und findet in vielen Bereichen Anwendung. Grundsätzlich wird eine Induktivität in Reihe mit einer Freilaufdiode vor einen Ladecondensator geschaltet. Dieser liegt parallel zum Ausgang. Zwischen der Spule und der Diode ist ein Schalter angeschlossen, der die Spule gegen Masse schaltet. So lädt sich die Spule 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. Dann wird die Schaltfrequenz auf das mindestmaß verringert, um den Wert zu halten. Natürlich ist die mögliche Ausgangsspannung nicht unbegrenzt über das Schaltspiel regelbar, sondern ist auch von den Baugrößen der Bauteile abhängig. Mit einer Induktivität von 10mH und einer Kapazität von 40uF lässt sich die Ausgangsspannung bei 5V Eingangsspannung zwischen 6V und 20V einstellen.

$$R1 = R2 * \left( \frac{V_{out}}{1,23} - 1 \right) \Rightarrow V_{out} = \left( \frac{R1}{R2} + 1 \right) * 1,23$$

## 2.2.5 Mikroprozessor

Der Infineon XMC 1xxx48 gehört zu der Familie der ARM Cortex -M0 Prozessoren und ist ein 32-bit Industrial Microcontroller und wird mit 48MHz externer Clock betrieben. Die 48 im Namen des Prozessors steht für die Anzahl der Pins. Der interne Timer läuft mit 96MHz. Außerdem bietet der XMC einen 12 bit A/D Wandler, welcher für die Analogmessung eine viel genauere Auflösung bieten kann als ein 8 bit A/D Wandler. Die Betriebsspannung des Prozessors beträgt 3,3V.

<https://www.infineon.com/cms/en/product/microcontroller/32-bit-industrial-microcontroller-based-on-arm-cortex-m/32-bit-xmc1000-industrial-microcontroller-arm-cortex-m0/#>

# 2.3 Entwicklung der Software zum Betrieb des Prototypen

## 2.3.1 Benötigte Register

Um das Programm zu entwerfen gab es die Möglichkeit, alle Zähler per Hand zu programmieren und so ein recht aufwendiges und langsames System zu entwerfen, oder die im Datenblatt hinterlegten Informationen zu bereits im Prozessor hinterlegten Zählregistern und interruptfähigen Eingängen zu erarbeiten. Durch diese aufwendige Recherche konnte ein grundlegendes Verständnis der Zählereinheit ccu40 erarbeitet werden, mit deren Hilfe sowohl ein Pulsweitenmoduliertes Rechtecksignal, als auch mehrere Zähler realisiert werden konnten.

## Durchgeführte Berechnungen

Auch für die Programmierung waren diverse Berechnungen notwendig. So musste zur Erzeugung der Ultraschallimpulse ein Pulsweitenmoduliertes Rechteck signal geschaffen werden. Dafür wurde ein Timer der ccu40 auf einen Takt von 40kHz eingestellt. So musste bei einem Timertakt von 96MHz eine Periodendauer von 2400 Takten und ein Compare-Wert von 1200 Takten eingestellt werden. Im Zählvorgang des Timers wird der Ausgang nach Erreichen des Compare-Wertes auf 1, und nach Erreichen der Periodendauer wieder auf 0 gesetzt. Dadurch ergibt sich eine Periodendauer von 25us, was einer Frequenz von 40kHz entspricht.

Auch zur Erfassung der Zeit, die vergeht bis das Echo des Ultraschall-Impulses zurück kommt wird über einen Timer der ccu40 erfasst.

## Einstellung im Programm

### 2.3.2 Quellcodeentwurf

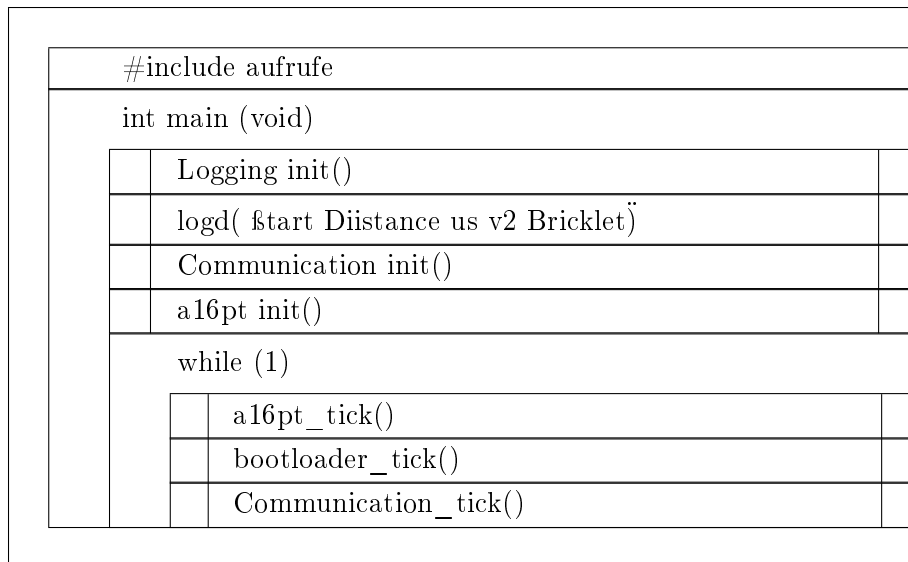
Um Fehler in der Zeitnahme zu verringern sollten Zeitwerte, die ausgelesen werden sollen direkt am Anfang des Interrupts in Variablen gespeichert werden und nicht erst innerhalb anderer Anweisungen oder Schleifen, da das schon deutliche Abweichungen mit sich bringt.

Programmstruktur: Anstatt alles in der main.c an Programmcode zu verfassen was bei sehr komplexen Programmen schnell zu unübersichtlichkeit führt sowie die bearbeitung an Bugs was dadurch noch extrem erschwert wird. Somit werden die Funktionen die in der main.c auftauchen ausgelagert und in einzelnen Funktion verschachtel.

```
1 #include <stdio.h>
2 #include <stdbool.h>
3 #include "configs/config.h"
4 /****Eigene Include Dateien****/
5
6 #include "bricklib2/bootloader/bootloader.h"
7 #include "system_timer/system_timer.h"
8 #include "bricklib2/logging/logging.h"
9 #include "communication.h"
10 #include "a16pt.h"
11 int main(void)
12 {
13     logging_init();
14     logd("Start Distance US V2 Bricklet/n/r"); Fuer den DBUGModus TXPin P0_12
15     communication_init(); //Funktionsaufruf
16     a16pt_init(); //Funktionsaufruf
17     while(true)
18     {
19         a16pt_tick(); //Funktionsaufruf
20         bootloader_tick(); //Funktionsaufruf
21         communication_tick(); //
22         Funktionsaufruf
23     }
24 }
```

Den Inhalt von den „Eigene-Include\_Dateien“ bis zur a16pt.h können wir vernachlässigen weil sie Standard mässig vor eingestellt werden. Die meiste Änderung findet in der a16pt.h statt wo der Fokus hauptsächlich liegt. In der a16pt.h werden die Funktionen definiert die dann in der main.c aufgerufen werden. In der a16pt.c stehen die Funktionsanweisungen dazu. So können auch einzelne Header-Dateien erstellt werden wo nur die Ports oder andere Werte definiert werden. Dieses geschieht mit dem Ziel nur einzelne Dateien bearbeiten zu müssen, um Änderung an Ports schnell und effizient durchführen zu können, ohne mehrere Dateien suchen zu müssen um eine Änderung zu bewirken.

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



**Abbildung 2.1:** *Struktogramm der main*

Im Main stehen vor allem die Aufrufe der verschiedenen benötigten Funktionen. Diese Aufteilung auf verschiedene Blöcke (und in diesem Falle auch Dateien) dient der Vereinfachung der Abläufe und der Übersicht. So sieht man in der Main jetzt deutlich, welche Funktionen beim Starten initialisiert werden, und welche Unterprogramme regelmäßig aufgerufen werden. Auch vereinfacht diese Struktur gerade bei Prototypen das Testen der Funktion, so kann im Falle einer fehlerhaften Funktion einfach der Aufruf auskommentiert werden um zu testen, ob der Fehler wirklich von der Funktion herrührt. Dadurch müssen nicht etliche Zeilen Programmcode der Funktion auskommentiert werden, wodurch schnell Fehler entstehen könnten, durch übriggebliebene Zeichen, oder gar beim entfernen der Auskommentierung gelöschte Zeichen.

In der a16pt werden die für die Entfernungsmessung notwendigen Funktionen und die Interrupt anweisungen abgearbeitet, außerdem werden die Ein- und Ausgangsports hier geschaltet.

# 3 Messungen und Auswertung der Ergebnisse

## 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.

### 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 40kHz ausgegeben werden. Danach erfolgt eine Pause, um das zurückkehrende Signal abzuwarten und auszuwerten.

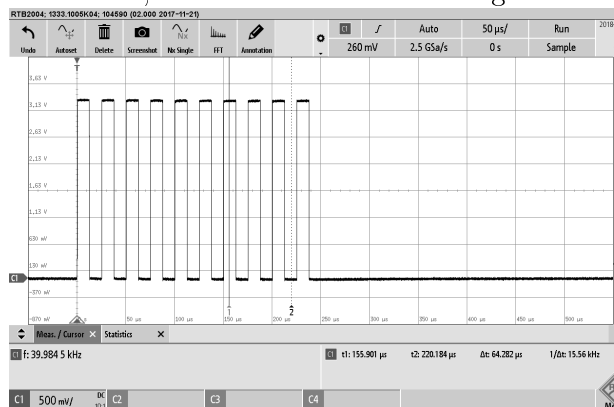


Abbildung 3.1: PWM-Burst auf 40kHz Basis an der CPU

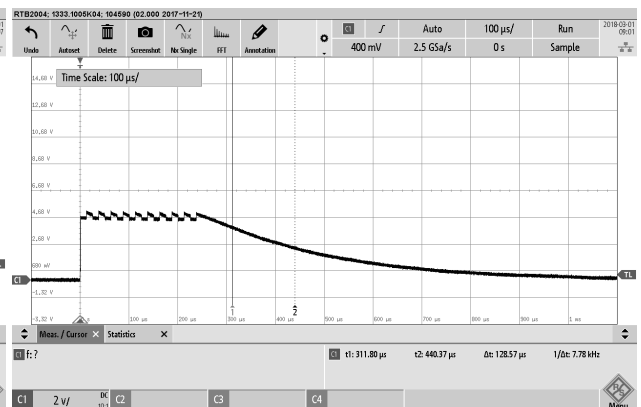


Abbildung 3.2: PWM Ausgabe über einen Hi-Side

In der Abbildung 3.1 ist zu sehen, dass ein Burst aus zehn Impulsen mit einer Periodendauer von jeweils 25µs 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 Hi-Side aussieht. So wird zwar im Takt des PWM-Signals geschaltet, allerdings fehlt es an einem Gegenpool, 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 maximal für kürzeste Entfernungsmessungen reichen, wenn überhaupt und dann würde das zurückkommende Signal noch von der abklingenden Spannung des Hi-Side überlagert. Dadurch ist dieser Aufbau nicht operabel.

Um die Spannung sauber auf Hi- und LOW-Pegel schalten zu können wurde somit auf eine Halbbrücke gewechselt. Bei der ersten, einfach gesteuerten Version, entstand das Problem, dass beim Schalten der Halbbrücke, in den Schaltmomenten, Kurzschlüsse entstanden, weil beide MOSFETs gleichzeitig geschaltet haben. Somit wurde auf eine voll gesteuerte Halbbrücke gewechselt und es wurden zwei PWM-Signale moduliert, bei denen eines invertiert und die Flanken derart verschoben waren, dass die angesteuerten MOSFETs nie Kurzschlüsse schalten können. Mit der verwendeten, vollgesteuerten Halbbrücke ergab sich die Abbildung 3.3

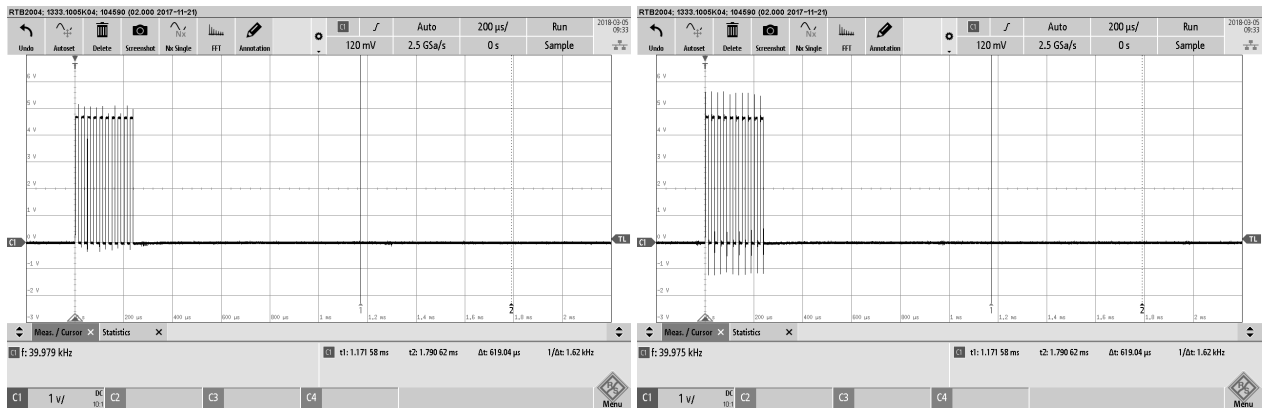


Abbildung 3.3: PWM Ausgabe über eine Halbbrücke

Abbildung 3.4: Ausgabe der PWM mit erhöhter Amplitude am Sender

Es zeigt sich, dass das Signal nach der Erweiterung auf eine Halbbrücke wieder wie das von der CPU ausgegebene PWM-Signal 3.1 aussieht, nur dass die Amplitude wie geplant höher ausfällt. Somit kann die Höhe der Amplitude über die Spannungspumpe variiert werden um die Stärke des ausgegebenen Signals zu variieren, ohne die CPU durch die höhere Spannung zu beschädigen. Wie in der Abbildung 3.4 zu entnehmen ist, entstehen durch den angeschlossenen Sender höhere Spannungsimpulse im Einschaltmoment.

### 3.1.2 Empfängerkreis

Der auf einer zweiten Platine aufgebaute Empfängerkreis wurde getrennt vom Senderkreis getestet, um eine mögliche Überlagerung der elektrischen Signale zu verhindern. Dafür wurden beide Platinen auf einem Versuchsaufbau befestigt und die Sender- und die Empfängerkapsel nebeneinander, parallel ausgerichtet, befestigt.

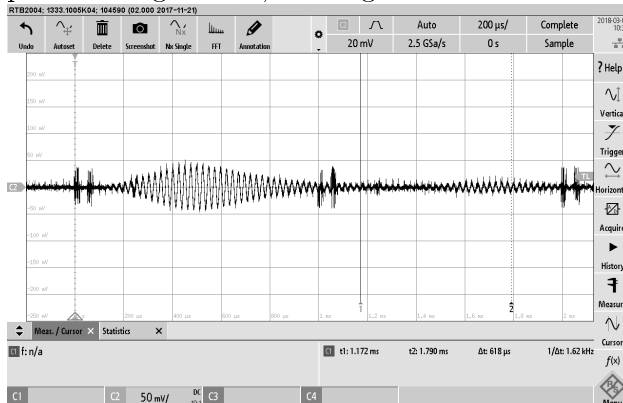


Abbildung 3.5: Signal Empfang

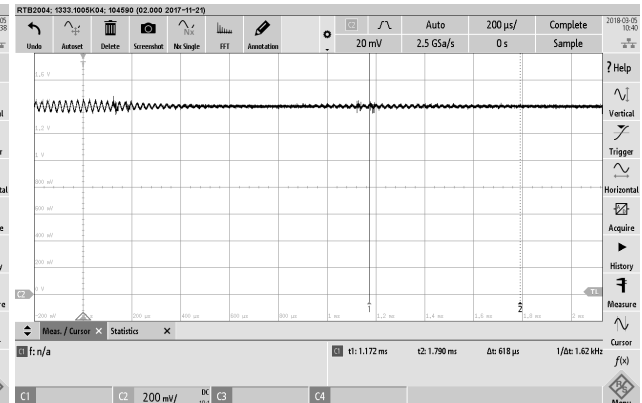


Abbildung 3.6: Signal nach der Filterung

Die Abbildung 3.5 zeigt das Signal, das direkt am Empfänger zu messen war. Hier sind verschiedene vorerst nicht zuordnenbare Signale zu sehen. Allein aus diesem Bild lässt sich aber keine Aussage zu den Signalen machen, fest steht nur dass ebenfalls Signale, die nicht der gewünschten Frequenz entsprechen ebenfalls vom Empfänger aufgenommen werden. Diese gilt es natürlich schnellst möglich auszumerzen, um unerwünschte Störungen zu vermeiden. In der Abbildung 3.6 ist das Signal nach dem eingebauten Hochpassfilter zu sehen.

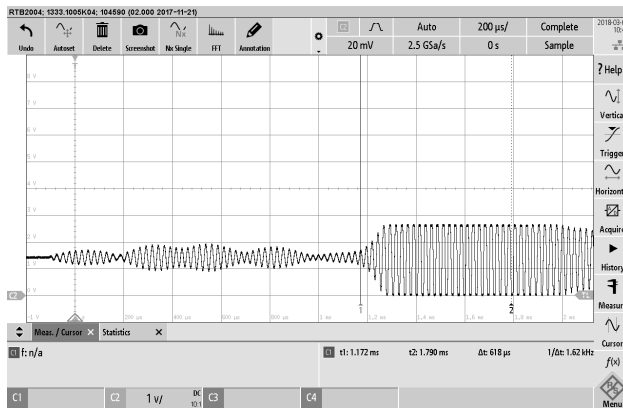


Abbildung 3.7: Signal nach Verstärkung

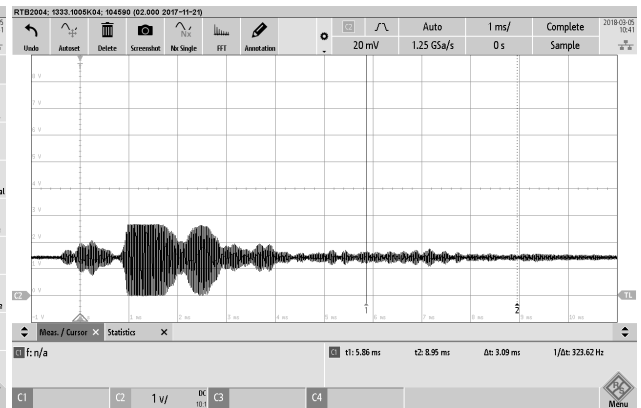


Abbildung 3.8: Signal nach Verstärkung2

Der erste Operationsverstärker dient der Verstärkung des gefilterten Eingangssignals. Dieses ist notwendig, da das ausgesendete Schallsignal mit zunehmendem Weg an Schalldruck verliert, und beim Eintreffen am Empfänger deutlich niedrigere Spannungen erzeugt, als beim absenden eingesetzt wurden.

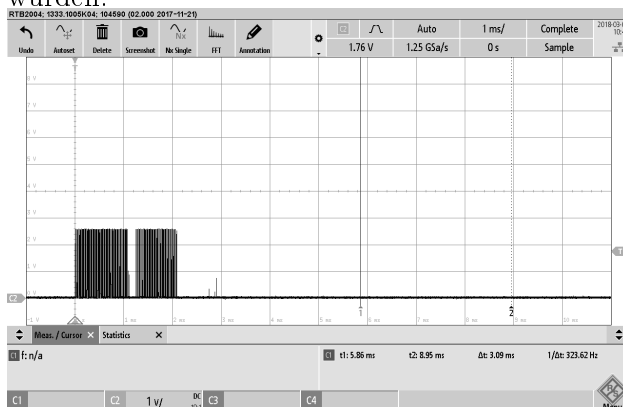


Abbildung 3.9: Signal nach Komparator

Nach dem das Signal den Komparator passiert hat, ergibt sich das Bild wie in Abbildung 3.9 zu sehen ist. Aus dem anfänglichen Eingangssignal, aus dem nichts verwertbares abzulesen war, ist ein wesentlich übersichtlicheres Signal entstanden. Nun sind lediglich zwei Signalblöcke übriggeblieben, die beide eine Frequenz von 40kHz haben. Bei dem ersten Signalblock handelt es sich um ein Stör-signal, denn es ist das vom Sender ausgesandte Signal, das seitlich auf den Empfänger abstrahlt. Der zweite Signalblock ist dagegen das Echo, also das Signal, das vom Hindernis zurückgeworfen wurde. Mit diesem Signal ließe sich die Entfernung zum Hindernis berechnen.

Nachfolgend wurden die Signalverläufe mit verschiedenen Ultraschallkapseln aufgenommen, um vergleichen zu können, wie die Signalqualität bei verschiedenen Produkten schwankt. Dabei wurde die Amplitude von 4,6V für das Sendersignal und die Verstärkung des Empfängers auf ein Minimum eingestellt.

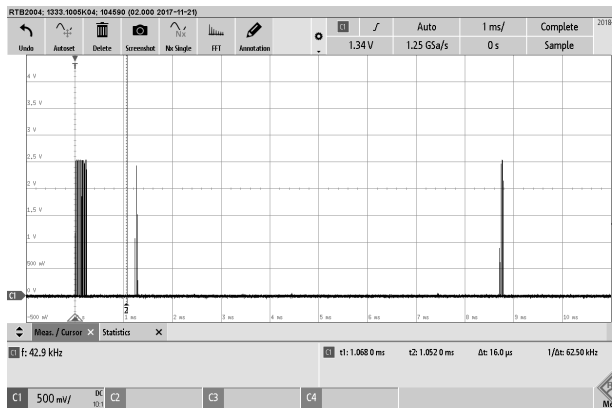


Abbildung 3.10: *EKULIT 1,5m*

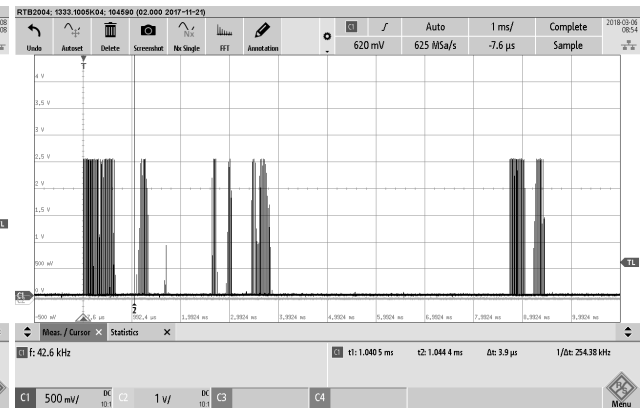


Abbildung 3.11: *MURATA reciver 1,5m*

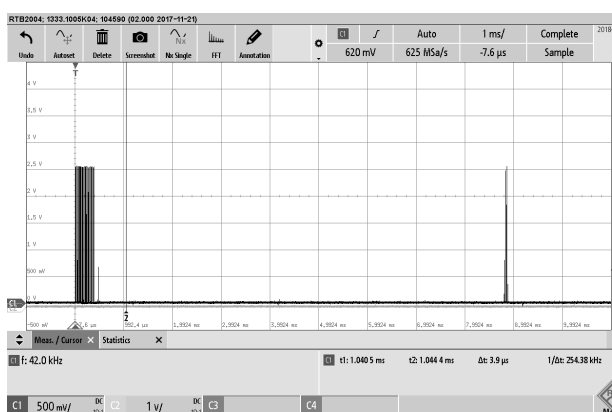


Abbildung 3.12: *MURATA sender 1,5m*

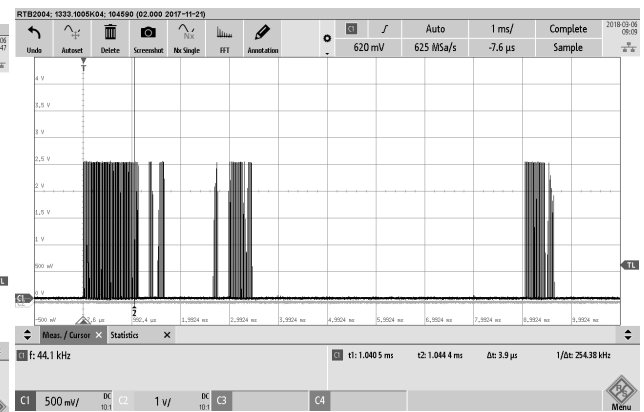


Abbildung 3.13: *MURATA sender + reciver 1,5m*

Bei den Ultraschallkapseln von EKULIT zeigte sowohl eine Verpoolung der Anschlüsse, als auch ein Tausch von Sender- und Empfängerkapsel keinen Unterschied. Bei den Ultraschallkapseln von MURATA sind bei der Wahl der Kapseln dagegen deutliche Unterschiede zu sehen. Werden nur die Sender-Kapseln als Sender und Empfänger verwendet, entsteht das in Abbildung ?? zu sehende Signal. Die Breite des eingehenden Echo Signals ist deutlich geringer, als bei den anderen verwendeten Kombinationen. Am besten sind die Signale bei der Kombination aus Sender und Empfänger von MURATA, Abbildung 3.13 oder bei den EKULIT-Kapseln, Abbildung 3.10.

## 3.2 Prototyp 2

Bei einer Betrachtung der Signalverläufe einer Entfernungsmessung, ließen sich die folgenden Signale aufnehmen. Die Spannung von der Spannungspumpe wurde auf 10v eingestellt, zu sehen sind die Signalverläufe nach dem Verstärker und nach dem Komparator.

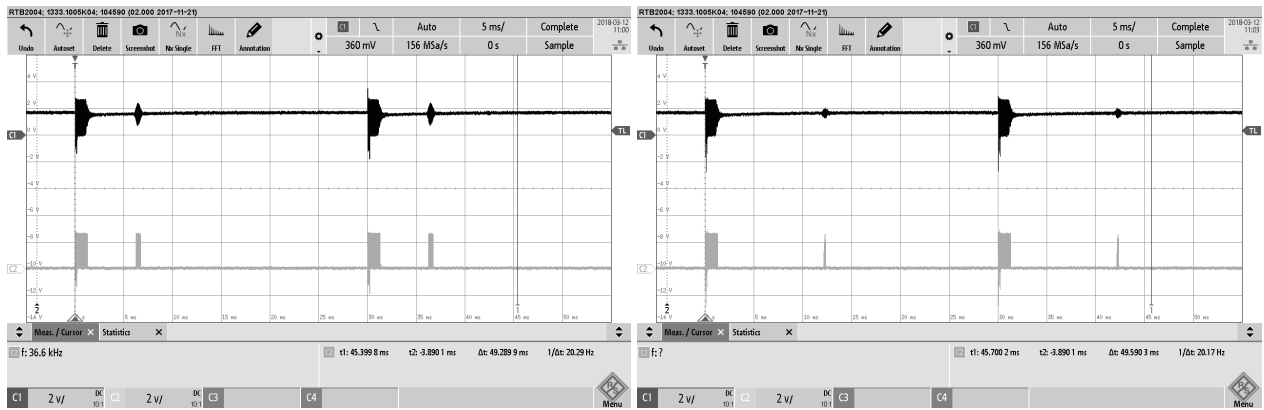


Abbildung 3.14: EKULIT 1m

Abbildung 3.15: EKULIT 2m

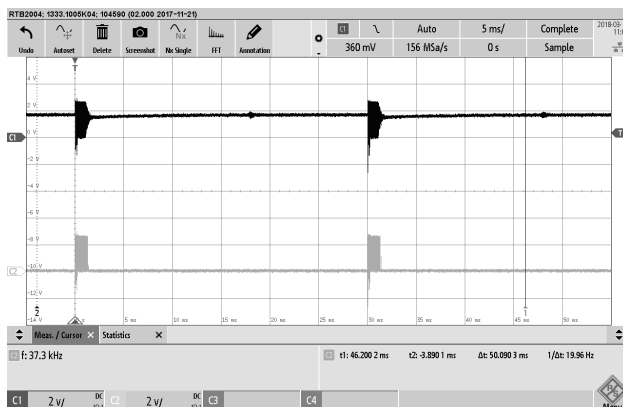


Abbildung 3.16: EKULIT 3m

In den Abbildungen sind jeweils zwei PWM-Ausgaben und die darauf folgenden Echos zu sehen. So trifft das Echo in Abbildung 3.14 etwa 6,ms nach dem absenden des ersten Impulses wieder an der Ultraschallkapsel ein. In der Abbildung ?? ist das Echo so schwach, dass die Verstärkung bereits nicht mehr ausreicht, um noch ein Signal nach dem Komparator zu ermöglichen.

Bei einer Erhöhung der Spannung von der Spannungspumpe auf einen Wert von 15,75V (Spitzenwerte an der Ultraschallkapsel von 20V) sind die nachfolgenden Signalverläufe entstanden.

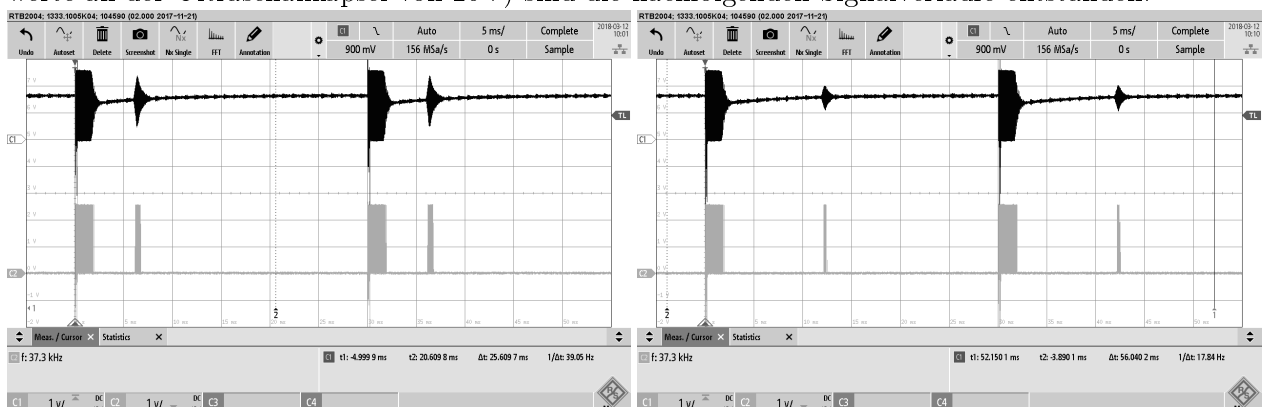


Abbildung 3.17: EKULIT 1m

Abbildung 3.18: EKULIT 2m





Abbildung 3.19: EKULIT 3m

In den obigen Abbildungen ist zu sehen, dass die Signalstärke der Echos durch die Erhöhung der Spannung leicht zugenommen hat. Gerade bei einem Vergleich der Abbildungen 3.16 und 3.19 ist zu sehen, dass der Komparator durch die Signalerhöhung auch bei dem Echo das Schalten beginnt. Das heißt, dass bereits durch die Signalstärke am Sender ein Betrieb bis 3m Abstand möglich ist.

### 3.3 Ergebnisse der Forschung

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 ist durchführbar. Bei der ein Kapsel Variante ist darauf zu achten, dass der Verstärker eine ausreichende Spannungsfestigkeit besitzt, um nicht durch das Sendersignal zerstört zu werden. Auch ist wichtig, dass bei der Erzeugung des PW-Modulierten Ausgangssignals die MOSFETs so angesteuert werden, dass das MOSFET, welches die High-Impulse schaltet, genug Zeit zum abschalten hat, bevor das MOSFET, das das LOW-Signal schaltet, einschaltet, und umgekehrt um Kurzschlüsse an dieser Stelle zu vermeiden.

Eine Auswertung des Echo-Signals ist prinzipiell sowohl Digital, als auch Analog möglich. Die Digitale Auswertung läuft recht simpel ab, prinzipiell muss nur 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 noch kleinerer Echo-Amplitude zu erkennen und auszuwerten, allerdings beinhaltet dieses Vorgehen einen deutlich höheren Programmieraufwand.

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 also 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. Allein durch die Tatsache, dass das empfangene Signal an der Ultraschallkapsel derart verändert wird, macht eine absolut exakte Messung nicht möglich. Eine Beschränkung des Fehlers auf einzelne Zentimeter dürfte aber durchaus realisierbar sein.

### 3.4 Fazit aus den Ergebnissen für den Auftraggeber

## 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.

# **5 Anhänge**

## **5.1 Schaltpläne**

## **5.2 Platinenlayout**

## **5.3 Quellcode**

Hier könnte Ihr Quellcode (inklusive Werbung) stehen  
weitere Abbildungen

# Abbildungsverzeichnis

2.1	Struktogramm der main . . . . .	8
3.1	PWM-Burst auf 40kHz Basis an der CPU . . . . .	9
3.2	PWM Ausgabe über einen Hi-Side . . . . .	9
3.3	PWM Ausgabe über eine Halbbrücke . . . . .	10
3.4	Ausgabe der PWM mit erhöhter Amplitude am Sender . . . . .	10
3.5	Signal Empfang . . . . .	10
3.6	Signal nach der Filterung . . . . .	10
3.7	Signal nach Verstärkung . . . . .	11
3.8	Signal nach Verstärkung2 . . . . .	11
3.9	Signal nach Komparator . . . . .	11
3.10	EKULIT 1,5m . . . . .	12
3.11	MURATA reciver 1,5m . . . . .	12
3.12	MURATA sender 1,5m . . . . .	12
3.13	MURATA sender + reciver 1,5m . . . . .	12
3.14	EKULIT 1m . . . . .	13
3.15	EKULIT 2m . . . . .	13
3.16	EKULIT 3m . . . . .	13
3.17	EKULIT 1m . . . . .	13
3.18	EKULIT 2m . . . . .	13
3.19	EKULIT 3m . . . . .	14

# Tabellenverzeichnis