SoftGlove

Vibrotactile palm feedback

T.H.M. Roos
K.M. Peelen

Abstract: VR gear is in the early stages of public use. A recent development is the use of haptic feedback and gloves to improve the immersion. This reports describes some of the challenges in this field, methods to stream haptic feedback in the form of audio, potential haptic protocols and contains the design process of a haptic feedback glove for SenseGlove.

TU Delft

SoftGlove
Vibrotactile palm feedback

by

T.H.M. Roos
K.M. Peelen

to obtain the degree of Bachelor of Science
at the Delft University of Technology,
to be defended on June 28, 2019

Authors:	T.H.M. Roos	K.M. Peelen
Student numbers:	4492242	4457692
Project duration:	April 23, 2019 – July 5, 2019	
Thesis committee:	Dr. M. Spirito	TU Delft, chair
	Dr. ir. C.J.M. Verhoeven	TU Delft, supervisor
	Dr. J. Dong	TU Delft
	Ing C. Lam	SenseGlove

This thesis is confidential and cannot be made public until May 1, 2025."'

Summary

The goal of the Bachelor graduation project in this thesis is to develop the electronics and firmware for a glove that can give a sense of touch to the virtual world. The glove will be an improvement compared to the current product: The SenseGlove. The designed glove will be an improvement in modularity, size and experience. The glove must has the following requirements (adapted from 2.2):

- A form factor that does not interfere with the movement of the hand.
- integration of the following feedback methods:
  - Per finger force.
  - Linear Resonant Actuators on the fingers
  - Lofelt actuator on the palm of the hand.
- No immersion-breaking latency.
- (Wish) Firmware for the glove which integrates with SenseGlove's systems.
- (Optional) A wireless datalink.
- (Optional) Powered by a battery.

The six students were split up in three groups of two, namely the Finger Force Feedback department, the Finger Vibrotactile Feedback department and the Palm Vibrotactile Feedback department. These departments all were responsible for one method of feedback, and together for the product as a whole.

In this report the Palm Vibrotactile Feedback department's design process is explained from research to prototype. It describes the discovery of methods which both decreased the size and data needed to drive actuator, as well as protocols that promise reliable QOS with minimal data transfer, high flexibility, and minimal delays. Finally a system has been made to drive the actuator that provides the feedback in the palm. Though not all functionalities that the team wanted to implement worked, the system still reached all the main objectives and scores well on most cost factors.

Preface and Acknowledgements

The Bachelor graduation project tests students on both the knowledge that was taught to them during the course of the bachelor and testing the other abilities students are suggested to develop during these years: To quickly grasp knowledge, know where and when to apply their knowledge, and to reach out to others who can teach them when they do not have the available knowledge, as well as key lessons in planning and both collaboration and independence. During these months everyone in this project has grown significantly in all of these fields.

But that did not happen without struggling. During this time we have asked and been offered help by many parties. We would like to thank these here.

First off the ones who were there with advice and insight. The ones who took the time to explain concepts big and small: Dan Shor, Michelle Corten and Chun Lam from SenseGlove and Chris Verhoeven, our supervisor. Without them this project would never have been as successful. Secondly we would like to thank Thiago Batista Soeiro for his expertise in power conversion and sitting in at our greenlight assessment. We would like to thank Max Lammers for his time and help with the demo. Our thanks goes out to Martin, who always came to see us, hooked us up with equipment if we needed it and gave us mental support when we were having trouble.

We would personally like to thank our colleagues from the other departments: Thanks for being able to laugh, think, run, drink coffee, practically live, struggle and finish this project with us. It was a pleasure. We would also like to thank the ones that gave an essential resource to our team: The ETV, the TU Delft and Coffeestar for the absurd amount of fixer elixir they provided over these months.

Contents

1 Introduction

Virtual reality (VR) and virtual environments (VE) has been the subject of many papers and articles. It is widely accepted that this technology has a place in the future and in some forms already in the present. Disney has been doing research in the VE field since atleast 2012 [1], but the first VR headsets were designed in the early 90's [2]. These products, however, were ahead of their time and limited by the technology of that era. This year Apple showed the world their 3D street view, supporting the rumours they are working on a bigger augmented reality (AR) product. Estate agents already give tours of houses that are still being build using VR[3], and therapists are experimenting with VR to help patiënts face their fears[4][5]. Similarly, the first 3D VR gaming stations and games have been released the last few years. The technology, however, is still in the early stages of public use and has to be futher developed and matured in order to gain better footing in this quickly evolving world.

1.1. SenseGlove, State of the art

In this market, there is a company named SenseGlove. A Delft startup based in Yes!Delft. Their main product is the similarly named SenseGlove, an exoskeleton around the hand that tracks the hands posture and position in realtime, has vibrating actuators in all fingers, and can deliver force feedback on each finger to make objects that only exist in the virtual world touchable. Though the company is still a startup, they are named among the best and biggest in the market like Oculus, Dexmo and HaptX [6]. Though all these companies have their own ideas and methods, they try to reach the same goal: Increasing immersion. Haptic feedback is a big factor in this[7]. This also mean they face the same challenges. One of these challenges is tight latency constraints. As the goal of the gloves is to increase immersion, the user must not experience too many delays when he or she interacts with this virtual world; Research has showed that people notice delays as small as 2ms [8], though people function fine with delays up to 50ms [9]. This is the point most people start reporting "simulation sickness" [10] which makes wearing VR gear an unpleasant experience.

1.1.1. The objective

SenseGlove, the client, is interested in making their glove not an exoskeleton, but an actual glove. This means smaller and if possible flexible electronics, increasing modularity and becoming wire independent. Our assignment is to create a prototype for the electronics for this version of the glove. This means transferring all current functionality except for the finger tracking to this model. The company also wants to add a widebandwidth vibrating actuator in the palm of the hand which can play music like data as feedback, without breaking immersion.

1.1.2. Thesis outline

This report is made by the Palm Vibrotactile Feedback department. The first part of this report, chapters 2 and 3 are made in collaboration with the colleagues from the Finger Force

Feedback department[11] and the Finger Vibrotactile Feedback department[12]. These colleagues will have the same 2 chapters in their report, with the exception of section 2.4, where the specific requirements for this department will be discussed. The report will then explain the process of researching , testing and prototyping that concluded a first and second revision of a prototype. In the final chapter recommendations are given for a next revision and the technology in general.

2
Requirements

This chapter discusses the general requirements that are the result of the assignment from the company SenseGlove. After detailed research the original assignment is changed to the final assignment. The assignments and requirements are a result of collaboration with SenseGlove about the time and practical limitations of the project. The final assignment and requirements will split the complete system in three subsystems. Finally the requirements that are specific for (TODO, jouw subsysteem) will be discussed.

2.1. Assignment

The current version of the product uses an exoskeleton. This design limits the capability and the scale of implementation for augmented reality applications. Therefore a soft/fabric version of the old design is an important development. This soft version should have at least similar capabilities as the current exoskeleton glove, with the exception of finger tracking and added vibrotactile feedback in the palm of the hand. The first assignment made by SenseGlove is discussed in Section 2.1.1. After discussions with the company about the project and research on the subject, the constraints did not completely fit the assignment. Therefore the assignment was modified in collaboration with SenseGlove, this assignment is discussed in Section 2.1.2.

2.1.1. Original Assignment

The original assignment was to design and realize a semi-flex PCB for the SoftGlove, which integrates per finger force feedback, linear resonant actuators in the fingertips and a Lofelt haptic actuator on the palm of the hand, including firmware, where communication to the PC through USB according to the SenseGlove protocol is possible. As an optional assignment, the glove can be outfitted with a wireless communication link. This assignment can be found in Appendix A.3.1.

2.1.2. Final Assignment

After detailed research it was apparent that some changes needed to be made to the assignment. The semi-flex PCB material is rated to bend a maximum amount of five times to make inserting the PCB in a housing easier [13]. It is not made to bend continuously back and forth and is therefore not suited bending with the movement of the wrist. Another option would be to use a fully flexible PCB However the design of a fully flexible PCB adds significant complexity to the design process, as described in [14]. Because of this, the use of a rigid PCB is chosen, which can be mounted on the wrist in the form of several modules.
Secondly there were some concerns about the assignments challenge level as the finger force feedback is already optimized for the current SenseGlove. Therefore it was decided to make the system work with a battery so the product could become entirely wireless. When making the SoftGlove wireless, power supply by a battery is needed which makes the power conversions for the finger force feedback more complicated. However, the SoftGlove must have the

ability to be powered via USB at 5 V with a maximum of 4 A. This results in a maximum available power of 20 W.

2.2. General Requirements

Based on the final assignment that is discussed in Section 2.1.2, requirements are set that are applicable for the whole system that should be made for the SoftGlove. The requirements can be divided in mandatory requirements, cost factors and stretch goals. All of these are listed below.

Mandatory

1. The glove must have per finger force feedback.
2. The glove must have per finger vibrotactile feedback.
3. The glove must have a larger vibrotactile feedback core in the palm of the hand.
4. The glove must support USB-based firmware updates.
5. The glove may not have a power consumption over 20 W.
6. The average latency of the PCB may be no more than 40 ms. How the latency is defined is discussed in Section 3.2.
7. The PCB must have over current protection.
8. The PCB must have over voltage protection.
9. The PCB must have reverse current protection.
10. The glove must stay under 40°C

Cost Factors

1. The latency of the glove should be as low as possible.
2. Extensions of the glove should take up minimal space on the wrist or other parts of the body.
3. The glove must have a minimal power consumption.
4. The feedback placement on the glove should be optimized where the sensitivity of the human skin is highest.
5. The glove should be as durable as possible.
6. The glove should fit a wide audience as comfortably as possible. This means the product should fit both men and women with a range of different sizes of wrists and hands.

Stretch Goals

1. The glove would benefit from being compatible with SenseGlove Communication Protocol [15].
2. The glove would benefit from having a wireless communication link.
3. The glove would benefit from using a mobile power source

2.3. Subsystems

It is clear the glove has three major feedback methods, finger force feedback, finger vibrotactile feedback and palm vibrotactile feedback. The finger force feedback can hold the fingers back when they are grasping an object in VR, creating the illusion of a solid object. The other two feedback methods are comprised of vibrations of actuators on the hand, creating the feeling of a buzz when touching something in the virtual environment. The finger vibrotactile feedback is comprised of a smaller actuator on each finger, whereas the palm vibrotactile feedback is a larger actuator in the hand palm. Because there are three types of feedback, the complete system is split up in this three subsystems. Based on the complexity of each subsystem, some secondary tasks are divided to the subsystems. An overview of the placement of all feedback subsystems is shown in Fig. 2.1.

2.3.1. Finger Force Feedback

The iconic form of feedback from the client is the Finger Force Feedback, allowing people to ”grab” or ”squeeze” items in a virtual environment, by applying force to the fingers that stops them from moving through a virtual object. This will be done using the actuators provided by SenseGlove. The actuators provide feedback on the top of all fingers, marked in blue, as shown in Fig. 2.1

This subsystem will use the most power and the highest voltage, and will therefore be accountable for designing the power converters.

2.3.2. Finger Vibrotactile Feedback

The more subtle but just as important way the current version of the glove provides feedback is through small actuators that vibrate the fingers. This system allows the user to experience for example button clicks and the smoothness of certain surfaces. This design is meant to be an improvement over the vibration motors currently in the SenseGlove. The finger vibrotactile feedback motors will be placed on the intermediate phalanges of the fingers and the proximal phalanx of the thumb, marked in green, as shown in Fig. 2.1.

2.3.3. Palm Vibrotactile Feedback

SenseGlove wants to add another way of feedback in their products, and they want it to be the Lofelt actuator based in the palm. This is a sensitive area that can provide general purpose feedback. The Lofelt actuator will be placed in the palm of the hand, marked in red, as shown in Fig. 2.1.

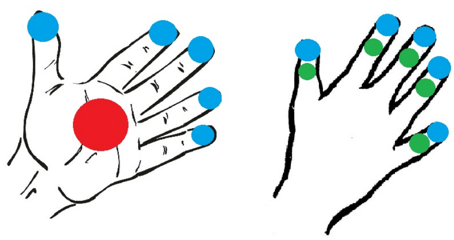


Figure 2.1: Overview of the placement of all subsystems on the hand of the user

2.4. Subsystem requirements

As is explained in chapter 1, the subsystem that will be discussed in this thesis is the Vibrotactile feedback based in the palm. This subsystem has some specific requirements and costfactors beside the general requirements discussed in section 2.2. The reasons behind these is discussed in chapter 4.

Requirements:

1. The feedback in the palm of the hand must be delivered by a Lofelt actuator.

2. The actuator should cause vibrations up to 1KHz.

Costfactors:

1. The maximal vibrational force should be as strong as possible .
2. The latency should be as low as possible.
3. The actuator profits from a flat frequency response.
4. The actuator profits from a continuous phase transfer.
5. The system must be as silent as possible.
6. The system should deliver the signal undistorted.

3
General Design

Next to the designs of the separate subsystems described in Section 2.3 some general design choices had to be made. These choices are applicable for all subsystems and are discussed in this chapter. The power supply consists of several parts that are split up between the subgroups. First the battery charger circuit which is done by the Palm Vibrotactile Feedback group, second the battery protection circuit which is made by the Finger Vibrotactile feedback group and third the type of battery which is chosen by the Finger Force Feedback group. Besides the power supply, the microcontroller and programming language were chosen. The way in which the systems cooperate can be found in Fig.3.2.

3.1. Power Supply

As described in the new assignment, which is shown in Section 2.1.2, the goal is to design a wireless glove. For the power supply this means a battery or multiple batteries have to be attached to the SoftGlove or to the human body. As can be seen in the program of requirements, which is shown in Section 2.2, the physical size is a major cost factor. Besides, a smaller system allows the gloves to be compatible for a wider audience, which is also a cost factor. Taking this into account, all considerations and final decisions for the battery type, charger and protection are outlined in this section.

3.1.1. Battery Type

Since the SoftGlove is designed for wireless application, a battery has been found that will not constrain the usage of the glove. From the general program of requirements in Section 2.2, some requirements for the battery follow. The battery should be able to deliver a peak power of 20 W and the battery, as an extension of the glove to the wrist, should take up minimal space.

Types of Batteries

The requirements immediately shorten the list of usable batteries for the application. The used voltages in the system are 3.3 V, 5 V and 24 V, where the 24 V subsystem uses the most power. The highest efficiency will be achieved with a battery input voltage of between 5 V and 24 V. This efficiency is mainly based on the boost from the input voltage to the output voltage of 24 V. When boosting an input voltage lower than 5 V to an output of 24 V, the efficiency of one the boost converter often becomes lower than 75% which is too low to meet the power specifications as described in Section 2.4. This efficiency will be further discussed in the Finger Force Feedback thesis [11]. The second option is to use two boost converters in cascade. However, this uses almost double the space, which is not available. Therefore the input voltage must be at least 5 V. Furthermore, for practicality and durability the battery needs to be rechargeable. Finally, the battery shape and weight influences comfort of the user of the SoftGlove. Taking all of this in account, five battery types were considered and discussed. Paper [16] was consulted, to further explain the differences between the different batteries. These battery types shown and discussed below. The best battery type is used in the design of the SoftGlove.

- Lead-Acid
- Nickel Cadmium(Ni-Cd)
- Nickel-Metal Hydride(Ni-MH) item Lithium-ion(Li-ion)
- Lithium-ion Polymer (Li-Po)

Lead Acid Batteries Lead Acid Batteries are created as very reliable and low-cost power sources. As disadvantage they have a low energy-to-weight ratio. Because of their big size and high weight in comparison to other battery types, this is not an option for wearable application.

Nickel Cadmium Batteries have a couple of useful advantages. For example, they can handle many charge/discharge cycles in comparison to the other types of batteries. On the other hand, there are disadvantages which are so crucial that this type of battery is not chosen for the SoftGlove. Firstly, the presence of the so called 'memory effect': The batteries lose their maximum capacity when they are being recharged after not being fully discharged. Secondly, This type of battery also contains toxic metals and the energy density is not as high as some other battery types. Another disadvantage is Nickel Cadmium batteries have a cylindrical shape, which is not ideal for efficient usage of the available space on the wrist.

Nickel-Metal Hydride Batteries have a higher energy density than Nickel Cadmium batteries but also have the cylindrical shape. The energy density also is not as high as with Lithium batteries. For the same capacity, a bigger and heavier battery is needed. Nickel-Metal Hydride batteries are not effected by the memory effect, which is an advantage. Despite this advantage, the self discharge rate is high and the maintenance to ensure a sufficient lifetime is very difficult. All the disadvantages makes the Nickel-Metal Hydride battery not suitable for usage by a wide and long term audience as for the SoftGlove.

Lithium-Ion Batteries are widely used for wearable applications. A disadvantage is that these batteries also have a cylindrical shape. This type of battery is comparable to Lithium-ion Polymer batteries [17], which have the advantage of a low profile and non-cylindrical shape. Their form factor makes it also easier to attach the batteries to the wrist. Li-Po batteries have a disadvantage of higher price comparing to Lithium-ion, however these costs small compared to the advantages. Lithium-ion has a sufficient discharge current for the case of maximal dissipation of 5 A, where maximally 2.5 A can be drawn. Lithium-Polymer generally has even higher discharge rates. Looking at safety differences, Lithium-Polymer is more sensitive compared to Lithium-Ion regarding over voltage and over current while charging and discharging. However, when using reliable and good protection circuits this can be prevented. In Table 3.1 the batteries together with their advantages and disadvantages are summarized. Taking all things into consideration, Lithium-Polymer is chosen as the optimal battery type.

Table 3.1: Decision Matrix Battery Type

Battery type	Advantages	Disadvantages
Lead-Acid	- Non-cylindrical shape - Reliable - Low Cost	- Low energy density - Big size, high weight
Nickel Cadmium	- Many charge/discharge cycles	- Memory Effect - Toxic metals - Moderate energy density - Cycindrical shape - Self-discharge rate high
Nickel-Metal Hydride	Similar to Nickel Cadmium but: - Higher specific energy - No toxic Metals - No memory effect	Similar to Nickel Cadmium but: - Less charge/discharge cycles
Lithium-ion	- High energy density	- Cylindrical shape - Requires specific protection system
Lithium-ion Polymer	- High energy density - Non-cyclindral shape - Low profile - High discharge rate	- Higher price - Requires specific protection system

Integration in Design

Lithium-Polymer batteries have a nominal voltage of 3.7 V. As stated above, it is inefficient to directly convert from this voltage to the 24 V, which is needed for the finger force feedback subsystem. To achieve higher efficiency, two battery cells can be connected in series. This gives a nominal voltage of 7.4 V. The disadvantage of connecting multiple cells in series is the mandatory use of a balancing system between the multiple cells to ensure safety and durability of the cells. From 7.4 V highly efficient boost converters are available that can convert this input voltage to 24 V. Connecting more than two cells in series makes balancing even more difficult and increases size as well. This makes connecting two cells in series the optimal design choice.

Next to choosing the amount of cells, the cell capacity also has to be chosen. This is the amount of energy stored in the batteries. As already said in Chapter 2 the glove must have equal or better specifications than the current model. The wireless kit, that is in development for the current SenseGlove, can last around 30 minutes on maximal power dissipation. To achieve this in the SoftGlove, the maximum power dissipation has to be estimated. Given the nominal battery voltage of 7.4 V, around 2.5 A can be drawn maximally. At this power dissipation the battery must last 30 minutes or more, so a capacity of at least 1250 mAh is needed. A battery is chosen with 1500 mAh capacity, where a maximum continuous current of 4.5 A can be drawn. The size is 66x32x6.5 mm, such that the battery can fit comfortably within the width of most wrists. The weight of two cells is 60 g, not more than the weight of an average watch. These two cells are connected in series to achieve the required input voltage of 7.4 V.

3.1.2. Battery Charger

Since the system will be charged over USB the charger needs to accept an input voltage of 5V. Unfortunately there is currently no IC available with support for boost mode charging, balancing and protection of a 2 cell (2S) lithium-polymer battery. Therefore a separate battery protection and charging IC is used. A single lithium-polymer cell is rated at a maximum of 4.2 V, two cells in series are rated at 8.4 V. Therefore the charger must be able charge the lithium-polymer battery to 8.4 V. The IC used for charging the battery is the BQ25883 from Texas Instruments. This is a 2S boost mode Li-Ion and Li-Po battery charger. It can charge the battery with a maximum current of 2 A. When using the battery as stated in Section 3.1.1 the charging time will be 45 minutes. The final circuit and layout of the charger can be found in Appendix A.1.2 and A.3 respectively.

3.1.3. Battery Protection

As stated above lithium polymer batteries need some types of protections. The cells of a Li-Po battery get damaged when they are charged or discharged too far. In case of over discharge the battery will lose some of its capacity and its self-discharge rate will increase. In the case of over charge, the battery might catch fire or even explode. This poses a safety hazard that is not ethically permissible in a consumer product. Because of this a solid protection circuit is needed. As stated in the section above there is no IC available that can charge, protect and balance a 2S battery. Therefore a separate protection IC is necessary. The battery protection IC that meets all these requirements is the BQ28Z610. While this IC is marketed as a gas gauge, a circuit meant to determine the state of charge of the battery, it also has many protections built in. The IC features over- and undervoltage protection, overcurrent protection, short circuit protection and overtemperature protection. Apart from these protections it also has the ability to balance a 2S battery. It therefore includes all the desired features that the battery charging circuit lacks. The final circuit and layout can be found in Appendix A.1.3.

Unfortunately the battery protection circuit is untested at time of writing. This is due tot the fact that the footprint of the IC was drawn incorrectly, both in terms of size and orientation. However, this has been rectified for the final prototype and the circuit has been checked multiple times to ensure there are no errors.

3.2. Microcontroller

The subsystems of the glove need to be controlled by an microcontroller. Since the desire was to make the system wireless a microcontroller with integrated wireless functionality is ideal. The ESP32 microcontroller was therefore chosen for the prototype as it provides a sufficient amount processing power, storage, IO pins and has integrated Bluetooth and WiFi connectivity. For the final version the ESP32 Pico was selected. The Pico has all the same functionality as the bigger modules, but is a lot smaller with its 7*7mm QFN package and requires no external components like crystals since they are builtin to the package. Even though the Pico has Bluetooth and WiFi functionality, it does not have a built-in antenna. Therefore an external antenna has to be used. The Proant 440 was selected, because of it's simplicity, small size and good performance.

3.3. Programming Language

The chosen ESP32 supports the use of multitude of programming languages, each with their respective advantages and disadvantages. The programming languages that were considered were Micropython, Arduino and ESP-IDF. The latter is the official development framework based on C provided by the manufacturer of the ESP32. Micropython has the advantage that it is easy to write and especially easy to debug since it is an interpreted programming language. This makes it possible to send commands and read out contents of variables over USB without needing to recompile and upload the code. There are however fairly major disadvantages to this approach. Micropython is slow when compared to Arduino and especially to using the ESP-IDF and it provides little flexibility in regard to for example assigning which pins the I^2C bus uses. Another disadvantage is that only a few people in the group have experience with Python and would therefore require some studying of the syntax and behaviour to write proper code. The Arduino programming language benefits from many built-in functions for controlling for example the I^2C or SPI bus and it supports the C and C++ languages. However since it is designed to run on a multitude of microcontrollers it features the same flexibility disadvantage as Micropython and is still not as fast as C or C++ code written specifically for the used microcontroller. This is provided by the ESP-IDF, which stands for the Espressif IoT Development Framework. This is the most low level language that has a similar structure as C and C++ and thus provides only limited pre-made functionality, it does,however provide a lot of flexibility and speed. Since a main limiting factor in this project is latency, execution speed of the commands is critical. Furthermore since the whole group has experience in writing C and C++ code from Bachelor courses this would be relatively familiar. Therefore the ESP-IDF was chosen for developing the software that would run on the final prototype. For software development reasons the ESP-IDF code for all subsystems has to integrate with the current SenseGlove communication protocol that is described in [15].

3.4. Latency Budget

One of the most immersion breaking parts of virtual reality experiences is latency. It is therefore part of one of the major requirements, namely that the average latency may not be more than 40ms. In order to understand which parts of the design have the highest latency a latency budget was constructed. First of all an estimation was made regarding the various components of the design. After the design and assembly, the actual latencies of the components was measured to check if the estimations were correct. The wireless communication, processing on the microcontroller, the driving of the finger force feedback actuators, the per finger vibrotactile feedback and the Lofelt circuitry were considered in the estimation of the latency budget. The estimated latency budget can be seen in Tab. 3.2. The latencies of the different subsystems have been measured and can be found in Table 3.3. The latency of the finger force feedback stays the same because it is based on the known switching delay and rise time of the MOSFETs.
An important matter to consider about latencies is the exact definition of the latency. The latency can be taken as the purely electrical or processing latency but it can also include the mechanical latency of the (vibration) motors. In deliberation with SenseGlove, it was determined that latency would be defined as the time between the computer sending the data to the moment the system sends the signal to the actuators. So mechanical latency and latency within the PC software is not taken into account. Additionally, the latency of the microcontroller was not measured in the final design as it is already included in the latencies of the subcomponents. The latency of the driver in the Palm vibrotactile Feebback department was hard to detemine. This is due to the nature of the output, which is explained in their report [18]. Their latency was estimated based on the datasheets.

Table 3.2: Estimated latency budget.

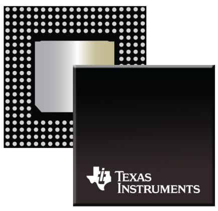
Component	Estimated latency
Wireless communication	10 ms
Microcontroller	1 ms
Per finger force feedback	0.1 ms
Per finger vibrotactile feedback	2.5 ms
Palm vibrotactile feedback	4 ms

Table 3.3: Measured latencies per subsystem.

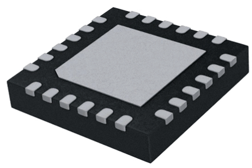
Component	Measured latency
Wireless communication	7 ms
Per finger force feedback	0.1 ms
Per finger vibrotactile feedback	1.9 ms
Palm vibrotactile feedback	0.1 ms

3.5. Broad Design Choices

Some general design decisions were during the design process. Firstly what component packages were going to be used. Since everything had to be soldered by hand BGA packages would be very difficult to solder properly. As can be seen in Fig.3.1a the package has pins on the bottom which are very hard to reach during soldering. BGA is therefore avoided. The same goes for QFN packages, while they are easier to solder than BGA they still pose a challenge, however, the QFN package ended up being almost impossible to avoid in some cases. In Fig.3.1b the QFN package is shown, it can be seen that the soldering pads are on the bottom but also reachable from the side. Another component choice was regarding the size for the passive components like resistors, capacitors, etc. Of course having smaller components would lead to an overall more comfortable design for the glove. This is due to a better fit on the wrist, because of the smaller PCB size. However, this would again make it hard to solder by hand. Therefore the imperial 0805 component size was chosen as a good compromise between size and ability to solder by hand. However, for the final prototype the space constraints were so tight that for the Finger Vibrotactile feedback subsystem, components with the size of 0603 were chosen. Another decision with a major impact on form factor was the amount of layers of the PCB. With more layers less space is required to route all the wires as well as the fact that it improves power distribution and shielding due to the ability to add more power and ground planes. The downside of going from a 2 to a 4 layer PCB is monetary cost, with a 4 layer PCB being almost twice as expensive [13]. For the first PCB a 2 layer design was made and manufactured. Because of this experience and space constrains it is decided to use a 4 layer PCB for the final prototype.



(a) BGA package [19].



(b) QFN package [20].

Figure 3.1: BGA and QFN packages

3.6. General System Overview

In Section 2.3 all subsystems that are integrated in the SoftGlove are discussed. In Fig.3.2 an overview of all connections between this subsystems is shown. The subsystems are abbreviated by FFF for per finger force feedback, FVF for per finger vibrotactile feedback and PVF for palm vibrotactile feedback. The blue lines represent the data lines between the modules, where the numbers show the amount of data lines. The red lines represent the power lines between the modules with the voltages shown on the lines. The USB block represents an USB micro input to charge the battery and connect to program the microcontroller which is shown as the ESP32 block. Furthermore, the power conversions block consists of a buck converter to create the required 5 V as well as a boost converter to generate the 24 V for the finger force feedback.

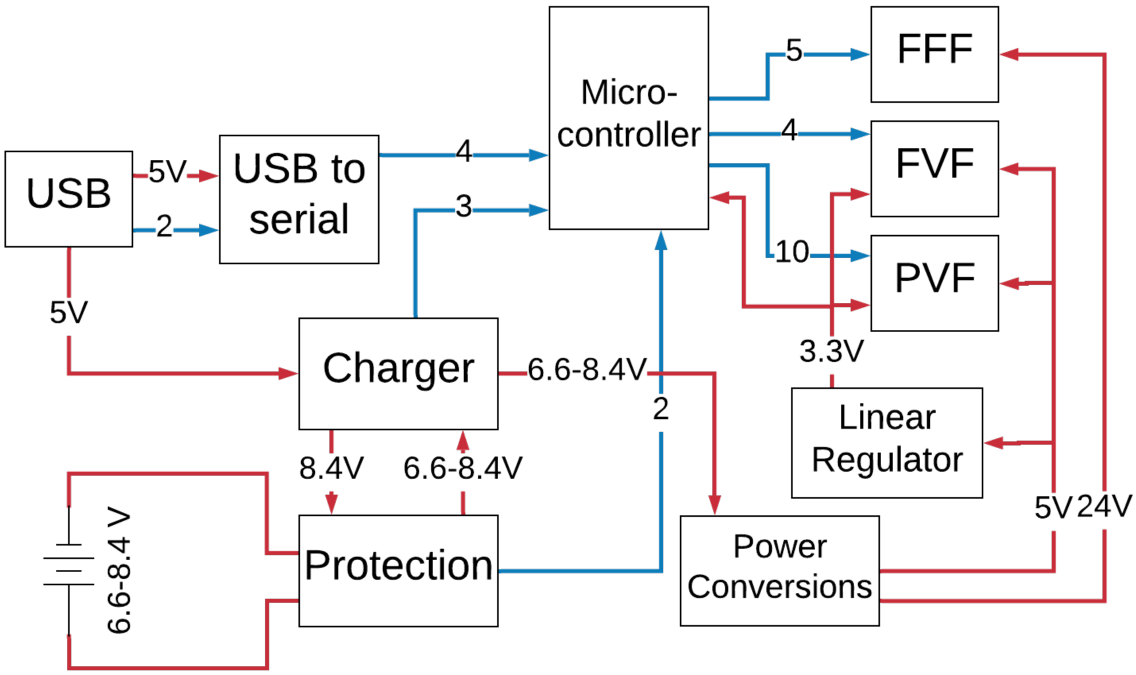


Figure 3.2: SoftGlove system overview. The subsystems on the top right are abbreviated as follows: Finger Force Feedback (FFF), Finger Vibrotactile Feedback (FVF) and Palm Vibrotactile Feedback (PVF).

All data lines are connected to the microcontroller. When determining all the data lines to the microcontroller, specifications had to be taken in account. First of all some pins output a PWM signal while the microcontroller is booting. Second, some pins are not allowed to be pulled up or down when the microcontroller is switching on. This is since these pins are responsible for selecting the boot mode. Third, some pins are specified to be just an input or just an output pin. The pin layout is therefore carefully designed and can be found in detail in Appendix A.7.

3.7. PCB Layout

The PCB stage consisted of two stages. A first PCB which is mainly focused on the functionality of the subsystems. The second PCB, which will be a revision of the first PCB, is mainly focused on the form factor and the placement of the subsystems. The second revisions will be the final prototype. The first PCB is 10.5 cm by 14.5 cm which is not the size that meets the requirement to fit on the wrist. The functionality of all subsystems is discussed and tested together with the revisions for the individual subsystem in the theses as described in Section 2.3. The layout of the second PCB, the final prototype, will be discussed in this section. As stated in Section 3.5, the first PCB is made with just 2 layers and the second PCB with 4 layers.

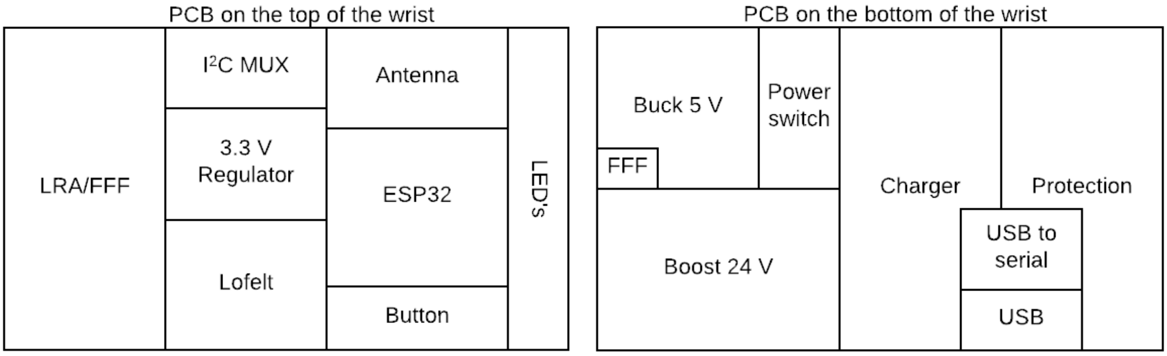
3.7.1. General Improvements for the Second PCB

After soldering and testing the first PCB, some general improvements had to be made when designing the second PCB. These improvements are listed below.

- A reset button for the microcontroller is needed.
- A power switch to turn the whole system on and of is needed.
- More test points need to be placed where possible.
- Pull-up resistors are required for both I^2C buses.
- Capacitors with a small capacitance need to be placed as close to the ICs as possible.

3.7.2. Final PCB Layout

All the improvements that are discussed in Section 3.7.1 together with the improved subsystems led to the final PCB layout that is shown in Fig.3.3. The circuits schematics of the final PCB can be found in Appendix A.1. The final layout consists of two PCBs that both have a size of 40 mm by 70 mm, which is considerably smaller that the first PCB. The choice for two small PCBs gives the possibility to mount one PCB on the top of the wrist and the other one on the bottom of the wrist. Each PCB is mounted with one of the lithium-polymer cells, so a cell on the top and bottom of the wrist which can together deliver the 7.4 V. In Fig.3.4 it is shown how this construction is set up. The PCB has all the components placed on one side to make sure nothing collides with the battery cells. The structure and design of all separate layers of the final complete PCB can be found in Appendix A.2.



(a) Top PCB layout. (b) Bottom PCB layout.

Figure 3.3: The layout of all subsystems on the final PCB

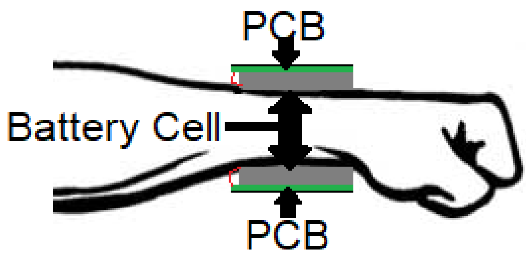


Figure 3.4: Mounting of the PCB and battery to the arm of the user.

4

Research and Conceptional tests

4.1. Lofelt

Lofelt is a company that makes high bandwidth vibration actuators and high speed digital signal processing (DSP) systems. These systems can drive the actuators based upon an full bandwidth audio signal of 22KHz[21]. Because of this they recently got used in the new Razer-hypersense headset, multiple studies [22] [23] and a small bracelet they produced called the Basslet, which was well received by the gadget world. SenseGlove believes this product is well suited for their needs and want to use their actuators for the SoftGlove version.

Figure 4.1: The L5 actuator as seen on the site of lofelt [18]

The goal SenseGlove hopes to achieve with a vibrotactile feedback core in the palm consists of two parts. To deliver information about what the user is experience in his or her palm, and to transfer a vibration experience to the entire hand. Because it is hard for humans to tell where on their palm a feeling originates, The palm actuator could work together with the finger vibrotactile feedback to fool the brain to think that the entire hand is being vibrated. The palm is also quite sensitive and used in a fair amount of actions [24], which makes it a logical spot to put a feedback source.

To validate the claims of Lofelt and understand its behaviour better, their latest L5 actuator was tested with the Lofelt evaluation kit, later revered to as the EVK, which has the DSP functionality.

4.1.1. Architecture and Design

The EVK has two options for drivers: the TAS2552 and the TPA2025D1. It used a STM32L431 Cortex-M4F microcontroller for the DSP. The STM32L431 was running firmware that is closed source and discouraged futher tweaking. It was packed with safety features that safeguarded the confidential code inside, including bricking itself when it sensed being touched a few times. All mentioned ICs were in a BGA packages, which is outside our design restrictions as explained in section3.5. Though the system was compact, the driving and DSP subsystems were on the large side compared to a wrist, where the PCB would eventually be placed.

The actuator was connected with 2 thin wires. If not handled and connected properly, these connections would break at higher intensity vibrations.

4.1.2. DSP

First off the DSP was tested as this has a direct effect the Lofelt's behavior and therefor needs to be understood before the frequency response of the L5 can be determined.

The DSP has multiple modes. Though the exact effects of the DSP is a trade secret defended by lofelt, the modes roughly seemed to have the following effects:

Mode	Effect
Pass through	Tunes down resonance peak, but no other effects
Cinema	Removes higher frequencies
Gaming	Boosts low frequencies, higher frequencies are shifted down.
Music	Bass frequency cause vibrations, inactive during the rest

It became clear this product has not been marketed towards pure haptic products, but towards products that want the haptic feedback as an augmentation of audio stimuli. The DSP provided is designed to create input for a haptic device out of audio, which is not the use case is in this project. This means the provided DSP is of little use to this project, which meant the system could be made smaller.

4.1.3. Frequency response of the L5

To verify the claims of the wide range the L5 could be be used in, frequency sweeps were played and blind AB testing was performed to judge if the actuator could still be felt. Frequency responses of the actuator were also performed in a later stage, but since they do not take into account the human aspect and its tactile thresholds these measurements are deemed less important.

Detectable frequencies

The datasheets of the actuator claims that frequency response of the actuator stay relatively flat up to 300Hz. However during our simple tests the vibrational force did not change significantly up to 1KHZ, though it did take more power. Frequencies higher than 1Khz were not felt anymore and could only be detected by the increasing amount of sound the actuator produces. This is to be expected: at higher frequencies the impedance of the Lofelt becomes higher and the response for a given power becomes less. This is why it was decided the goal was to produce vibrations up to 1Khz, and prevent higher frequencies reaching the actuator, as these only generate heat and sound. This means our driver has to deliver at least 2K samples per second to meet the Nyquist criterion.

Actuator characterization

Since the perception of humans can give quite a distorted view on the actual vibrations the actuator produces, full spectral analysis of the L5 was also performed. White noise with an sample rate of 8Khz and thus bandwidth of 4Khz was played through the actuator and its response was measured by an accelerometer[1]. The actuator was mounted to an 100gram weight and placed on an isolated surface. The accelerometer could only sample up to 2.4Khz, so the highest frequency that could be measured with this sensor is only 1.2KHz. Since this bandwidth is smaller than the bandwidth of the white noise played through the actuator, no distortions should be visible. The results can be found in Fig.4.2 and show that apart from the already known resonance peak, the frequency response of the L5 is indeed pretty flat. This resonance peak could be compensated for by reducing the power of frequencies in this range, or the resonance peak could be utilized as the most power efficient operating point for the L5. There also a small peak visible in around 800Hz, but this peak was not felt in our tests. There is an small rolloff visible starting at 1KHz, which coincides with the point at which the actuator cannot be felt anymore.

[1] The setup used for this was lend from the finger vibrotactile Feedback department.[12]

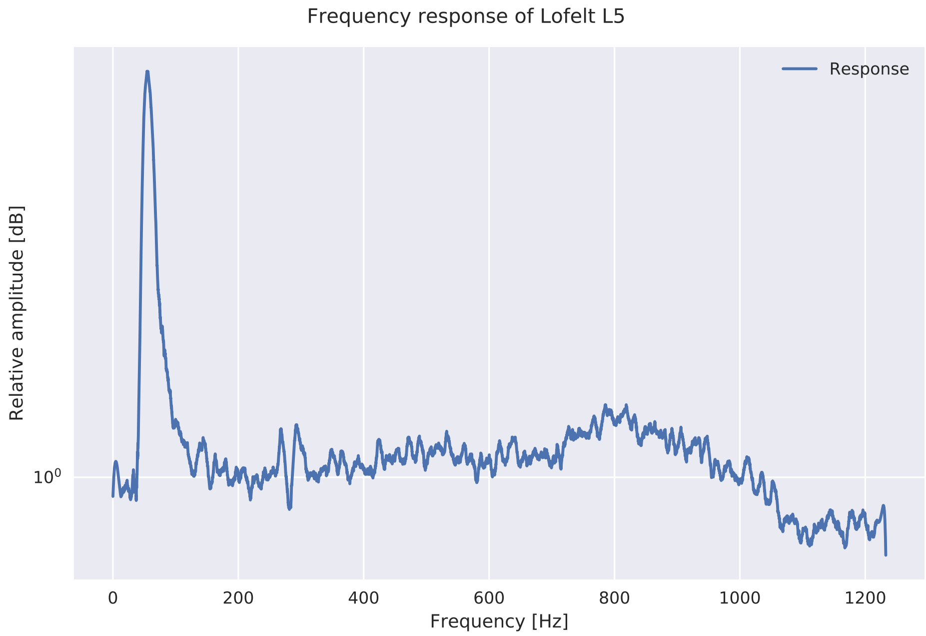


Figure 4.2: Measured frequency response of the Lofelt L5

4.1.4. Proof of concept

A rudimentary serial wired communication protocol was implemented, to conform the feasibility of driving the Lofelt using a ESP32. The system used in this setup had no DSP and analogue music was send to an class-D amplifier with a bandwidth of 4KHz. This was frequency was chosen so that the music was recognizable and major distortion would be noticeable. The L5 got warmer than in a normal use case, which was expected as higher frequencies are dissipated to heat.

At frequencies around 60 to 80 Hz the resonance caused clipping and distortions. This means two things. The EVK was indeed tracking and compensating for resonance, and it will be good for the cost factors is the SoftGlove would have this feature too.

The ESP is equipped with both Bluetooth and Wi-Fi functionality which was also tested. Replacing the USB link with a Bluetooth connection and a small buffer. At this point a sampling frequency of 8KHz was being transferred. During the test the buffer was 30 milliseconds and both overflowed and underflowed often due to non consistent Bluetooth connection.

From these test it was clear that streaming audio over a wireless connection could have potential. Since the goal was to use the actuator with frequencies up to 1KHz, the digital transmission only has to be be up 2KHz. To space out harmonic images and effectively filter any frequency above 1KHz away, it would be profitable to send an oversampled signal or interpolate the signal as it arrives.

4.1.5. Power dissipation

To measure the desired driver power requirements for maximizing the L5, an rudimentary test was performed to determine what the L5 can handle. In this test music with an bandwidth of 1KHz was send at an sample rate of 16KHz (to prevent any high frequency distortions) and amplified by an PAM8302A. This driver was chosen because it was simple and is capable of delivering 2.5 Watts. The signal power was raised slowly, until a point was reached that the actuator would get above 40°C. At this point the power usage of the entire system was measured, then the L5 would be plugged out. Then the power usage was measured again to estimate the power usage of only the L5 from the difference from these two measurements. This turned out to be about 0.9 Watt. From this test we concluded that our driver would need to be capable to deliver at least 1 Watt without creating a lot of distortion.

4.2. Transmission

As seen in the requirements in section 2.2, a connection to the main computer is obligatory. The current system uses an USB 2.0 connection over which an serial link is emulated. Therefore the current communication protocol [15] is based around serial communication ports. Virtual serial communication ports over USB can easily reach speeds up to 2M baud / second, making it plenty fast enough for the current system. The new here proposed systems should however transmit more data, preferable enough to drive the Lofelt in realtime like the other components currently are. It was also desired to make the total system wireless and that would also need the communication protocol to be wireless.

4.2.1. Wireless streaming

Due to the limited timescope, no new radio frequency protocol is proposed, but rather existing options have been validated. The chosen microcontroller, the ESP32, has support for 150Mbit/s Wi-Fi and both Bluetooth Classic and Bluetooth BLE and therefore these will be considerd

Instead of benchmarking these protocols and concluding from the benchmarks which technique is suitable, actual streaming implementations for each were implemented and tested. The streaming protocol sent over random data, but in the same way that an host computer would. Different sample rates and thus differrent L5 bandwidths were tested and also different collision refresh rates and thus packet sizes were also tested. The testing system measures the time it takes for the random data to arrive at the ESP32 at either an rate which is significantly faster then the sample rate or an entire packet completely. The time taken was computed by taking the the time difference from the sending of the random data and the receiving of the acknowledgement back from the ESP32. By including the time taken for acknowledgement to be received, all measured delays will be slightly higher than single direction delay. This increased delay was deemed acceptable as it both provides an upper bound and is not much higher than the single direction delay as for both Wi-Fi and Bluetooth most time is spent in the setting up of the connection and transferring the random data.

Data quality checks were also performed in which all random data received by the ESP32 was echoed back to the host computer and then compared to the random data that was sent out. In all cases the echoed data was exactly the same as the original data, as was to be expected since all the tested protocols include error checks and error correction techniques.

Bluetooth

Although current streaming protocols for Bluetooth like A2DP for audio exhibit significantly higher latency than desired, Bluetooth was still considered as a wireless protocol. Since the current Senseglove protocol [15] works over an serial link, it makes sense use the SPP (Serial Port Protocol) of the Bluetooth 2.0 EDR specification as an direct replacement for USB. Bluetooth BLE does not have an similar protocol and does not have a fixed API to implement such an protocol ourselves in Microsoft Windows versions before 10 and will therefore not be considered. All test results with of the Bluetooth 2.0 EDR performance can be found in appendix D and one of the most interesting tests is also shown in Fig.4.3. As can been seen in the figure, the delay of the Bluetooth connection is quite noticeable, but also pretty consistent. The performance of Bluetooth did not seem to differ from clean to noisy RF environments, which is to be expected due to the direct connection and adaptive frequency-hopping connection that is used. Whilst all the data packets that were transferred arrived correctly, not every data packet was always transferred as the Bluetooth connection sometime refused to send over an packet. This would result in missing samples on the glove and the glove would need to for example repeat old samples to fill the time for the next packet to arrive. The effective bandwidth of Bluetooth was found to be quite low and sample rates higher than 2Khz produced an significantly higher rate of packets that did send. We can conclude from these measurements that Bluetooth could just about serve as wireless protocol, but it functions optimal when there is not much data to transfer.

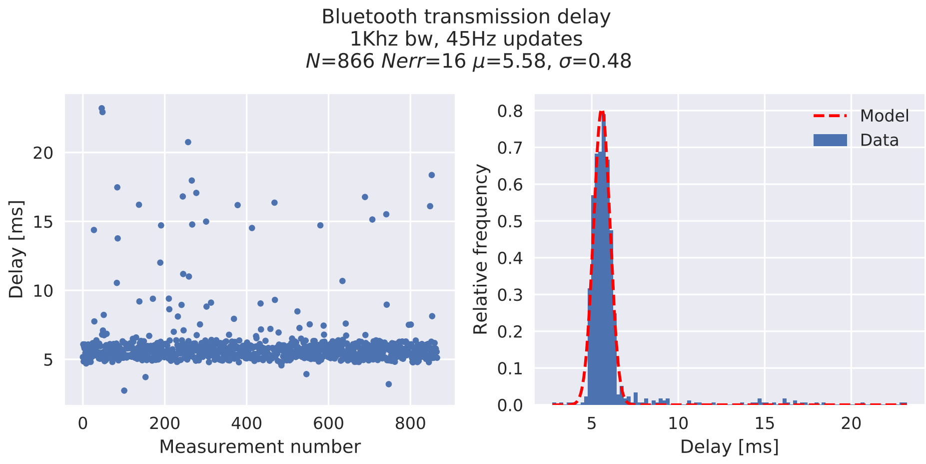


Figure 4.3: Bluetooth test with Fs=2000 , 45Hz

Wi-Fi

Unlike Bluetooth, Wi-Fi is usually used in an access point (AP) - client configuration and therefore direct connections between devices are tunneled over an central access point. Since it is important that our packets arrive correctly and in the correct order, TCP was used in all Wi-Fi tests. All performed tests can be found in appendix D and one of the more interesting AP tests is shown in Fig.4.4. As can be seen in this figure, the average latency is acceptable, but there are some spurious long delays that are not desirable.

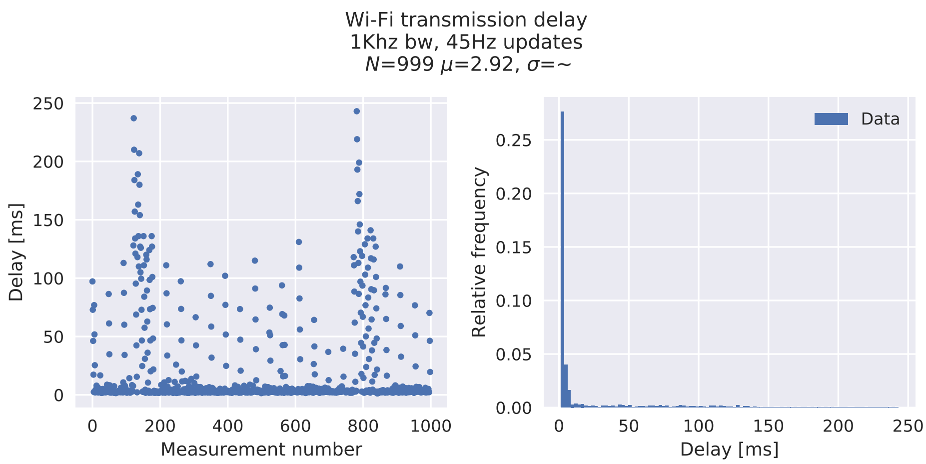


Figure 4.4: Wi-Fi with AP test with Fs=2000, 45Hz

Direct Wi-Fi tests were also performed in which the ESP32 functioned as AP and the host PC connected directly to the network of the ESP32. An representative result of this test is shown in Fig.4.5. The average latency of this setup is incredibly low in all tested configurations and therefore seems very suitable. Integrating it into the current Senseglove protocol [15] would need more work, since there is no native serial emulation protocol readily available for Wi-Fi.

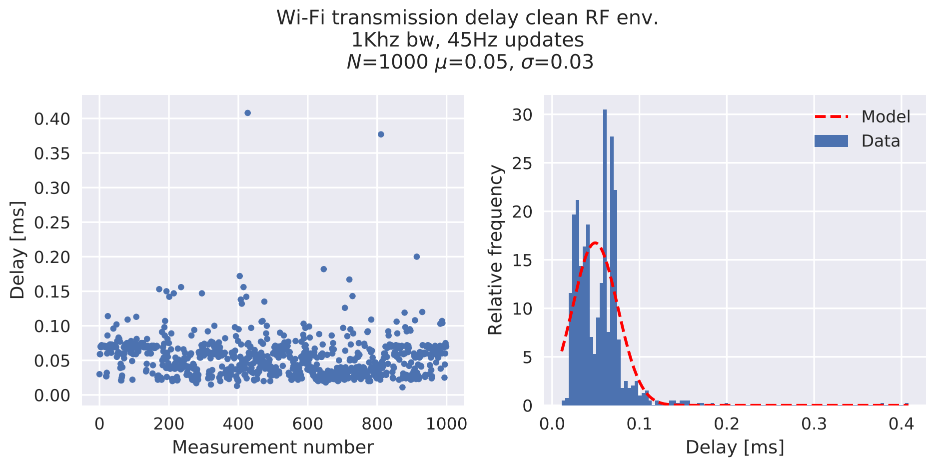


Figure 4.5: Wi-Fi direct test in clean RF env. with Fs=2000, 45Hz

Further tests were also performed in a less perfect RF environment to validate the robustness and repeatability of these nice results. An representative test is shown in Fig.4.6. From these test it can be concluded that direct Wi-Fi connections only work well in clean RF environments and are not very robust in noisy RF environments. Due to the unpredictable nature of quality that Wi-Fi links offer, this technology doesn't seem like the best to stream data in real-time.

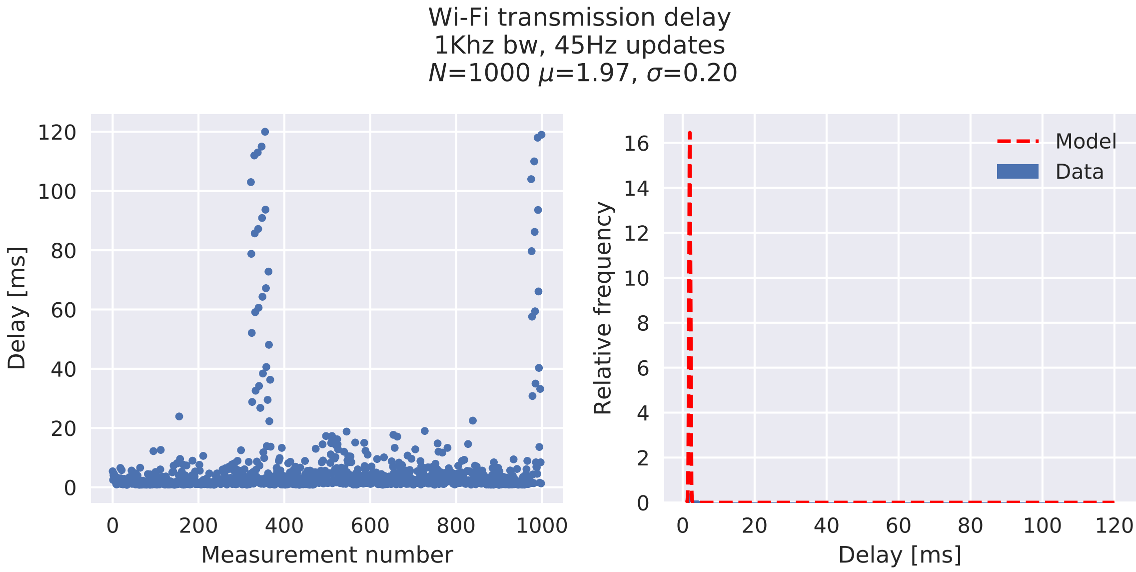


Figure 4.6: Wi-Fi direct test in noise RF env. with Fs=2000, 45Hz

4.2.2. Wired streaming

Wired communication should also be considered as an option, since it is very robust. Whilst it does not offer the flexibility that Wi-Fi and Bluetooth offer, the guaranteed bandwidth makes it very suitable for streaming in real-time. The ESP32 does not have an built-in USB peripheral, but has multiple TTL serial peripherals that support up to 5M baud with FIFO buffers in both directions. To connect this serial line to an host computer an Silicon Labs CP2102N was used. TTL serial to USB 2.0 converter IC has an maximum baud rate of 2M baud, which corresponds to 2Mbit/s of guaranteed asynchronous bandwidth.

Latency tests were also performed over this serial link at the maximum speed of 2M baud and show that the transmission latency is so little that it cannot be measured correctly. Since a packet is considered 'received' as soon as it starts arriving at a speed that is higher than the sample speed, this is practically instantaneous. Due to the asynchronous full-duplex nature of the serial link, the acknowledgement is sent while the data is still being received.

4.2.3. Preloading

Wireless connections are fundamentally unreliable in terms of latency. This means it would be beneficiary to depend on it as little as possible. That is why instead of steaming, 2 other methods were thought of that worked around the problems of steaming all the samples to the system. Preloading data is likely to be the most reliable and simple method. The idea is to load a library of short fragments that can be played repeatedly. Then when a the user is supposed to feel feedback, the computer sends which sample should be played and if it should be repeated until it receives a stop signal, or just one cycle. This system almost guarantees the feedback will be delivered on time and is rather flexible. The downside is however, that this system is limited by the amount the memory on board. Secondly there will need to be a loading period. This period can be at start up, but might need to be changed between "scenes", which challenges immersion. Finally there can be a disruptive effects if there is a phase jump when a fragment ends and the next one starts.

4.2.4. Parametric

A parametric system would send only a few values representing a frequency, amplitude and possibly an extra modulation effect, like a heartbeat rhythm or square wave. This could then be generated on board of the SoftGlove. This method would give the developer more options. More data would have to be send than preloading during use, but there is no need for data loading before use.

4.2.5. Overview

In short, all three methods have their advantages and disadvantages.

Table 4.1: The advantages and disadvantages of a variaty of methods

(a) The advantages and disadvantages of streaming

	Streaming
+	Absolute freedom in provided feedback
-	Requires stable connection
o	QOS dependent workload for SoftGlove

(b) The advantages and disadvantages of preloading

	Preloading
+	Very low data transfer during use
+	Lots of freedom in fragments choice
+	Lowest workload for SoftGlove
+	Easiest to implement
-	Loading time
-	Limited by memory
-	Phase jumps

(c) The advantages and disadvantages of parametric

	Parametric
+	Low data transfer
+	Flexible waveforms
+	Guaranteed continuous phase
-	No way to get different waveforms
-	High workload for Softglove

It is possible to combine multiple systems. One can imagine where a system where once it is determined the connection has a bigger delay than usual, the system switches to the parametric system. Another option is to have a few fragments preloaded, and still have the option to generate signals onboard.

4.3. Upsampling

As discussed in section4.2.1, A wireless transmission can not guarantee high samplerates without sacrificing latency. Since many driver ICs require higher input sample rates, the system might need to resample the data it receives. This causes undesirable effects: Imagines that appear on multitudes of the sampling frequency, distortion and sampling windows. These effects should be minimized. At the same time the sampling should not be too computational intensive as that would cause delays and increase the latency of the SoftGlove. These 2 cost factors will have to be weighed and an optimum will be chosen. To test this a few audio samples were taken and sampled down to 8KHz. As the effects caused by resampling does not depend of actual the frequency, but only the ratio of the frequency to the sampling frequency, these frequencies was to see if music would still be recognizable after interpolation. The fragment was reconstructed with a factor of 4 to 36KHz using sample and hold , linear interpolation, adding zeros and cubic spline interpolation.

4.4. Methods

Sample and hold

Sample and hold is the simplest method. Instead of trying to estimate where a point would be at that point, the latest sample is taken and repeated. This does cause an discontinuity at every original sample which will cause distortion. An example is given in Fig.4.7a

Linear interpolation

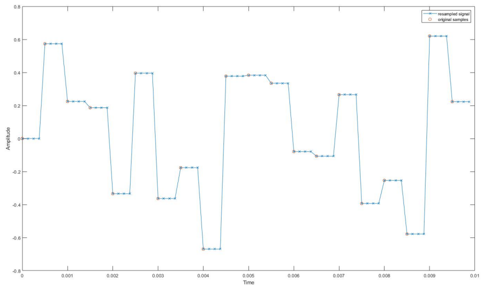
Linear interpolation takes 2 samples and adds points on a straight line between these samples. This is not computational intensive to do. An example can be seen in Fig.4.7b.

Add zero

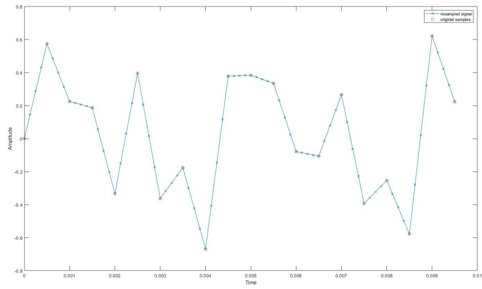
An experimental type of creating more samples was to add zeros instead of trying to estimate the actual signal. The idea is that in the frequency domain endless images appear without at the same intensity as the original every $F_{sampling}$. These copies could be filtered out if there is a DSP integrated in the audio driver, which has more advantages as discussed at 5.1(!). An example can be seen in Fig.4.7c

Cubic spline interpolation

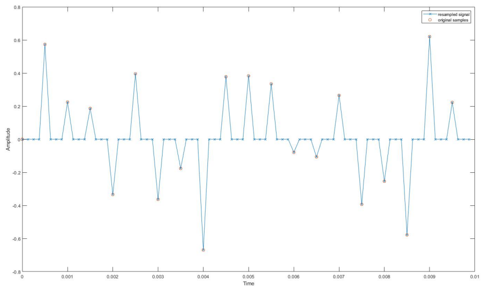
Cubic spline interpolation uses third polynomials estimate the values in between samples. It fits this polynomial to 3 samples and pins new points on this line. This line is continuous in contrast to the other methods which improves its performance when looking at higher frequencies.



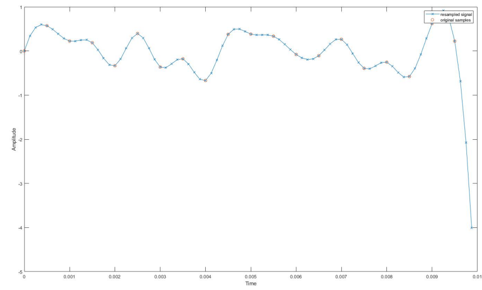
(a) Sample and hold example



(b) linear interpolation example



(c) Adding zeros example



(d) Cubic spline interpolation example

4.4.1. Effects

The effects of resampling can be seen in the frequency domain. Each method distorts the signal in their own way and creates images of the signal differently as well. These effects can be seen in Fig.4.8. A line has been added at each multiple of the sampling frequency. Spline and linear interpolation preform the best. Sample and hold causes noise all through the new bandwidth.

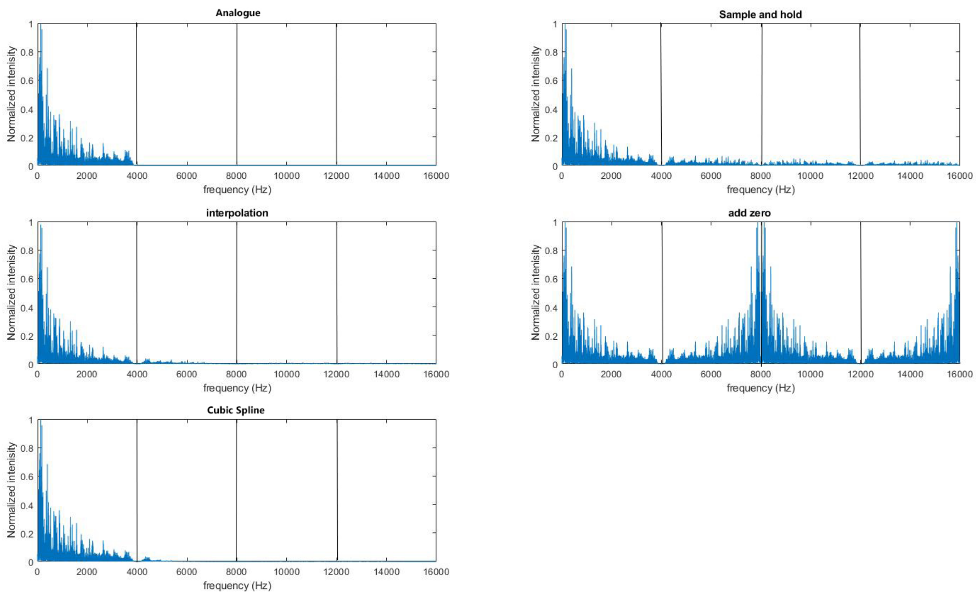


Figure 4.8: Fourier transform of the original signal and the resampled versions.

Because linear and spline interpolation preform similarly well, the original spectrum was subtracted from the resampled versions. This way the difference and thus distortion is clearly visible. This can be seen in Fig.4.9. Cubic spline outpreforms linear interpolation in both peak and average distortion, and the distortion in localized in a fixed frequency range. This makes it easier to work around. Interesting is the performance of adding zero's. It has no distortion in the original bandwidth, confirming the theory that this could be made functional when the original signal is sampled at a higher frequency than the actual bandwidth, if filtered with a sufficiently steep low pass filter.

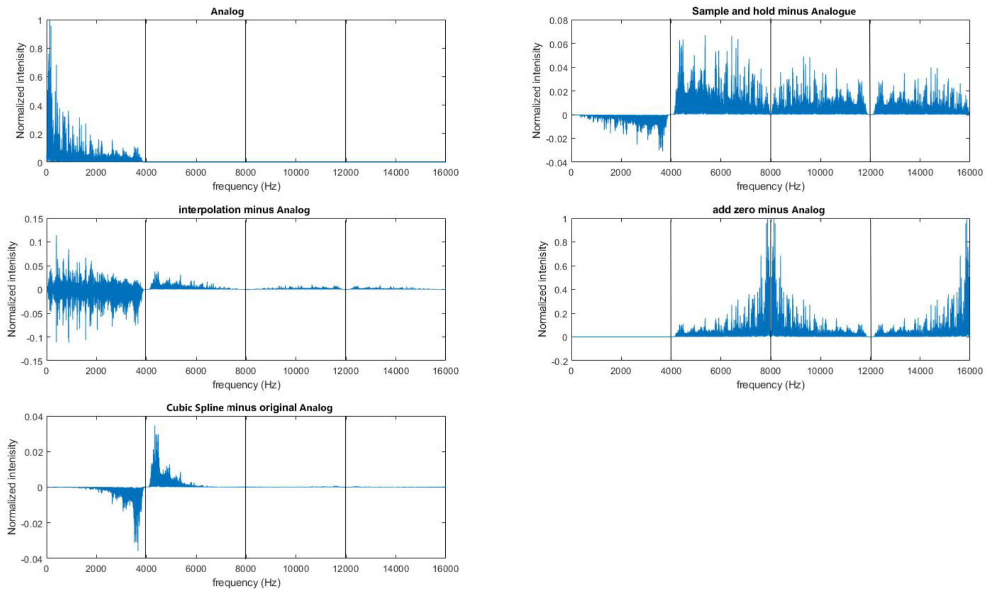


Figure 4.9: Distortion in the frequency domain caused by resampling

5 Prototype

The prototype will consist of 2 major components. The microcontroller that receives and processes the data send by the computer, and the driver of the L5. An overview of this system can be seen in Fig.5.1

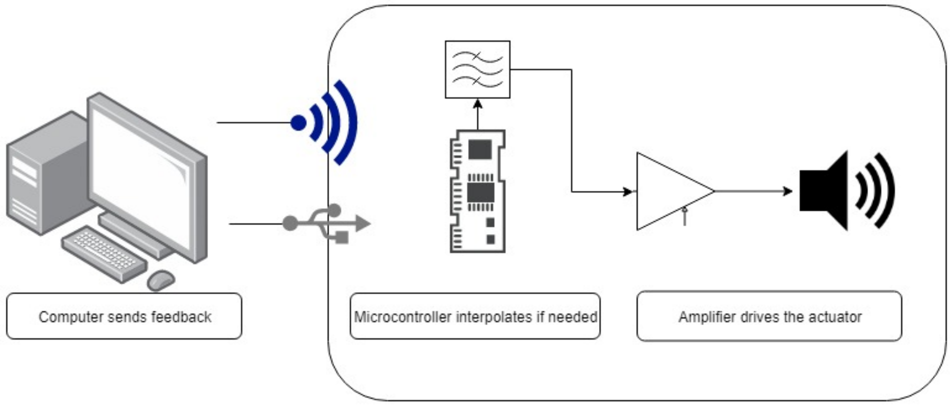


Figure 5.1: A simplified overview of the system

5.1. Driving the L5

The requirements demand that the L5 is functional as a vibrotactile source. The actuator requires a driving system but what exactly is the best driver is not obvious. The system is in a way both similar to a LRA and to a speaker, which both have different systems to drive it. Both these options will be considered, as well as adopting the system on the evalution kit that is provided by the manufacturer of the actuator.

5.1.1. EVK

Using the DSP and amplifier provided by Lofelt on the EVK should be considered. It accepts a wide variety of inputs for both setting registers and sending over the audio, is well designed and has build in powersense. It is arguably the best implementation to drive the L5 though it has options and parts we will not use. The DPS settings that would be used would be bypass, which does not do anything except filtering for the resonance frequency. The board's circuit is rather big and getting the necessary documentation for this option had already proven time intensive.

There are multiple ways to implement this system. A cut out of the EVK board could be used as well as a copy the circuit. The first method is a work around for the firmware and saves time, but the board is expensive and big. Copying the system requires firmware which is currently unavailable and hardware that is hard to install by hand, as it uses BGA packages.

It does allow for some minor modifications that can bring down the size by removing some overhead and rearranging components.

5.1.2. Audio versus LRA drivers

The L5 actuator is in definition still an LRA, and could be driven as one. LRA drivers seemed like an obvious choice at first as the needed power and resonance tracking of more readily available in this market. However LRAs generally are not made to function on frequencies above 300Hz [25]. This has as a result that LRA drivers rarely function up to 1KHz[26]. Secondly normal LRAs are meant to be driven at their resonating frequency instead of being tuned down. After searching for a suitable drivers none were found that met the requirements.

5.1.3. Audio drivers: Analogue versus Digital

The final option is to design a with a audio amplifier without copying the EVK. Than the next challenge would be choosing between a analogue and a digital input for the driver. Audio drivers have a digital (like I2S) input, an analogue input or both. Analogue is the simple option: a microcontroller would send out the necessary amplitude using the build in DAC at practically any frequency up to the Megahertz range. The main limitation to that frequency is how many samples can be send over and generated. The waveform coming out of the microcontroller will look like a sample and hold signal, so this would need to be sampled sufficiently high as observed during section 4.3. Digital signals like I2S have advantage that they can be filtered by drivers with DSP build in, which could undo some of the negative side effects of resampling. DSP could also offer one solution for the resonance frequency; it could implement a notch filter. This does mean a minimum sample rate of 8K sample/second must be achieved as that is minimum supported sample rate. I2S additionally is more complicated and needs not only a data line, but also a bit and word clock. Build in DSP also increases complexity as it has to be configured and puts constraints on the clock speed.

5.1.4. Resonance frequency

As discussed before, it would be good to tune the resonance frequency down to make the frequency response flat. This can be done using a notch filter. This is however only true if this frequency does not change between L5s and is not altered much by being attached to other things like connectors, hands and gloves. This is why IV sensing was considered. If the system can tell the actuator starts to resonate, it could tune down the input and correct itself. This does severely limit the choice in drivers. Luckily there was a moment where we could talk to an engineer at Lofelt, who claimed that the actuators are relatively unaffected by their surroundings and have a consistent resonance frequency; a filter with a bandwidth around 20 Hz should be enough. Taking the word of this engineer, A notch filter in DSP would suffice.

5.1.5. Power

As mentioned in section 4.1 the L5 will be placed in the palm of the hand. This means it will be connected directly to the main mass of a hand, which damps the vibrations. This is why it is important to use the full power of the L5. This means using a continuous power output up to 0.9 Watt of power. More than that and the L5 gets too warm as was discovered (see section 4.1.5)

5.1.6. Constraints

After these consideration there are still constraints that need to be kept in mind:

- No BGA or similar packages.
- Deliverable within a week
- Input voltage at most 5.5V

5.2. Design

After comparing all the options found during the research phase a design was chosen. in Tab.5.1 an overview can be seen of all the options described in 5.1. LRA drivers were deemed unsuitable and using the EVK was both too expensive, complex and not necessary. Instead, an audio amplifier would be the main component of our system

Table 5.1: Advantages and disadvantages of multiple driver topology options

	EVK cutout		**EVK Copy**		**LRA driver**
+	Wide input choice	+	Wide input choice	+	Simple
+	Easy to ”design”	+	Easy to ”design”	+	Power is plenty
+	Boost and Auto-resonance	+	Boost and Auto-resonance	-	Small bandwidth
-	Lofelt DPS	+	Can be made smaller.	-	Often boosts resonance frequency
-	DSP can not be altered	-	Lofelt DPS	-	No DSP
-	Documentation unavailable	-	DSP can not be altered		
-	Board is expensive	-	Documentation is hard to come by		
-	Board is big	-	Firmware is hard to come by		
		-	BGA packages		

	Analogue audio amplifier		**Digital audio amplifier**
+	Simple	+	DSP
+	Large Bandwidth	+	Large bandwidth
-	No resonance tracking	+	Resonance tracking
-	No DSP	-	Complicated
		-	Harder packages (BGA)
		-	Needs $F_{sampling}$ of at least 8KHz

5.2.1. Analogue versus Digital

In the end the choice between an analogue or digital input audio amplifier was delayed. There were simply too many different options and solutions to each problem to chose either one. With both options on the prototype both methods could be tested. On the analogue line a second order filter lowpass filter was made for 1KHz to filter away frequencies above that, as those mainly cause sounds and heat.

5.2.2. TLV320AIC3120

In the end the choice was rather limited. The combination of no BGA, DSP and both analogue and digital input only left a few options open. The ones that were still possible were compared in the other cost factors, especially how much power could be delivered and the power/harmonic distortion behaviour. This gave the TLV320AIC3120 [27] as conclusion. The chip is complex; It requires both a digital and a analogue ground and has a datasheet of over 150 pages, of which 100 is the manual describing registers that need to be set and how to choose the all the variables.

5.2.3. Details

The driver can function at 3.3V, but the performance increases a lot in terms of harmonic distortion when this voltage is increased to 5.2 V as seen in Fig.5.2. However, the driver will still need another 3.3 V input as well next to the 5.2 V. Creating a boost converter create the 5.2 V from 3.3 V is desirable as these big in size. A better solution is to use an LDO to create 3.3 V from the 5.2 V. Only some logic of the TLV320AIC3120, the Switch of the finger vibrotactile feedback department and the ESP require 3.3 V, Which

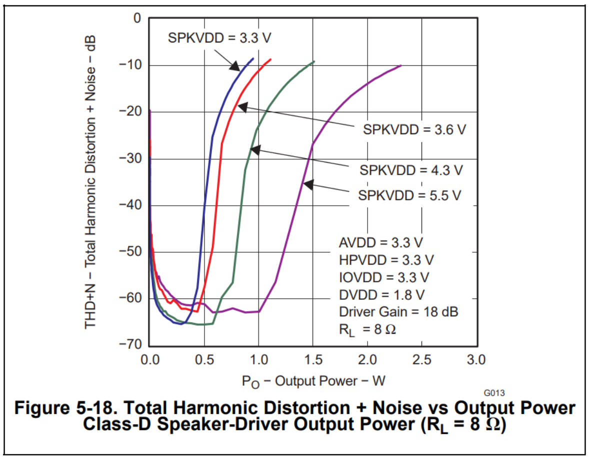


Figure 5.2: Distortion versus output power for a 8 ohm load

totals around a Watt of power during peak performance. [27] [28] [29]. The Driver also needs an LDO for 1.8 V for the internal DSP logica. This LDO became the TLV755P, because it can deliver enough power and is readily available.

5.2.4. Implementation

During the drawing of the schematics and later the routing on the PCB, The typical application noted in the manual was followed as close as possible. Only pins and features that would not be used were removed and disconnected or connected to the ground. The split between analogue and digital grounds was not possible; there is one digital pin that has to be connected digital ground as quick as possible, but placed on the analogue ground part of the chip. The splitted ground is meant to prevent EMI on the analogue ground. However, when the using analogue input the digital should not generate too much EMI, and during digital communication the analogue path is not being used. This means not following this design rule should not cause too much trouble. The prototype schematic and blueprint can be seen in Fig.5.4 and 5.3 respectively.

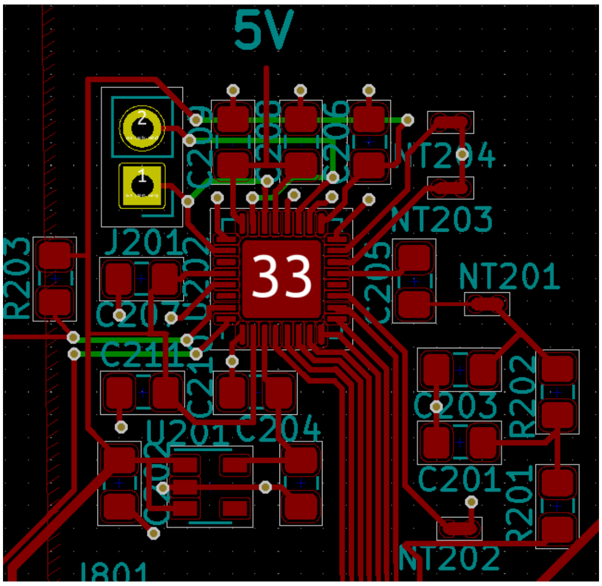


Figure 5.3: PCB layout of the prototype Lofelt driver

Currently the ESP uses a 8bit DAC, which means a SNR of[30]:

$$SNR(dB) = 6.02 \cdot Q = 6.02 \cdot 8 = 48.16dB \tag{5.1}$$

5.3. Results

Overall the subsystem worked and met its mandatory requirements. A more detailed result is discussed below.

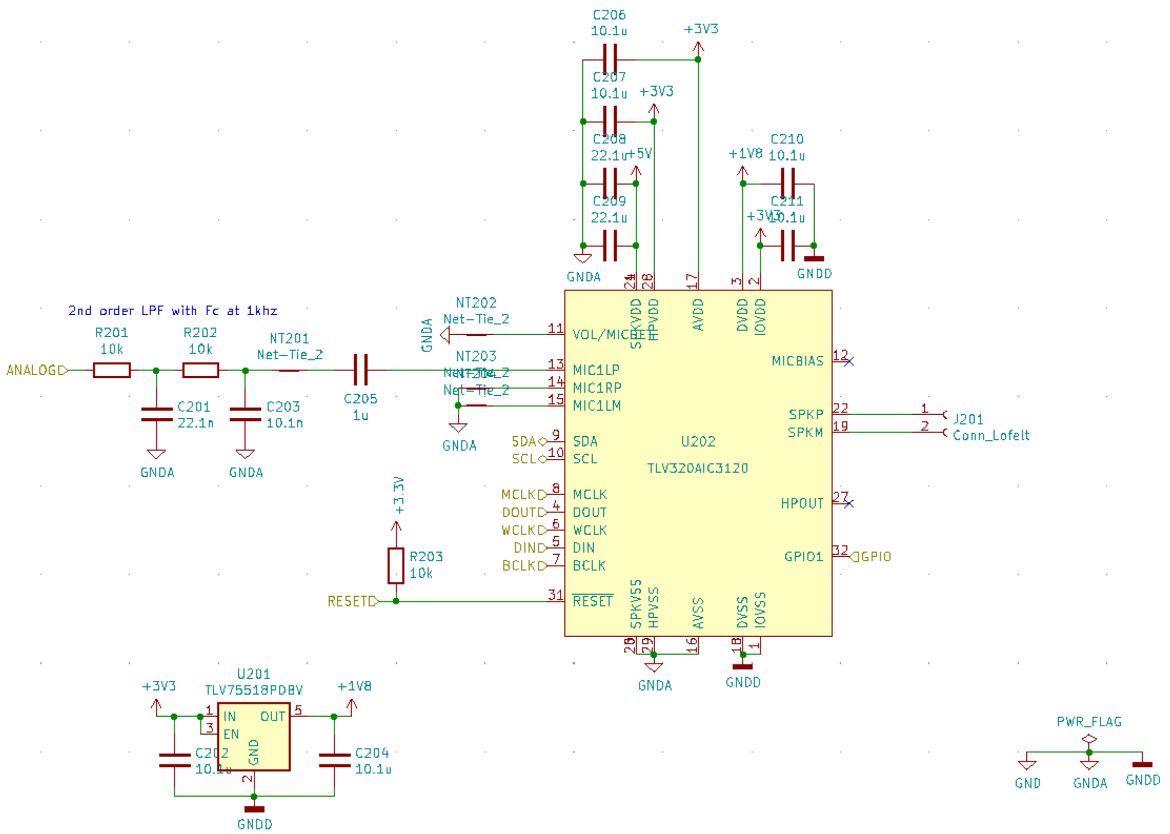


Figure 5.4: schematic design of the prototype Lofelt driver

5.3.1. Analogue

Analogue connection was a success and works as expected. However, the output of a class D amplifier is a PWM signal, which means it is rather complex to get a frequency response from it. The L5's frequency response can be measured with an accelerometer, but this is not accurate enough. There are some estimations made about the quality of the signal going into the driver.

Assuming the worst case scenario of a signal sampled at 2KHz, the signal will be interpolated to 8Khz. This will be the output of the DAC of the ESP. This signal looks a lot like a sample and hold signal. This could cause some more noise, that is why the simulation interpolates to 40KHz using sample and hold. 40KHz was chosen so that actual bandwidth contains the entire audible spectrum. The original signal was upsampled with both linear and cubic spline and then compared. In figure 5.5 and 5.6 the error normalized around the input is shown.

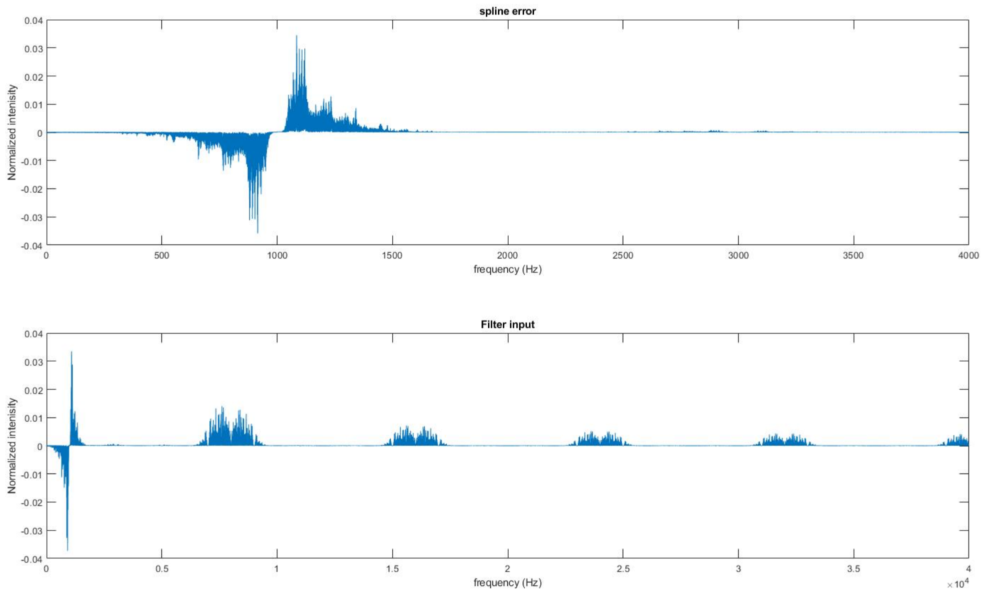


Figure 5.5: Effects of analogue sending when using Spline interpolation.

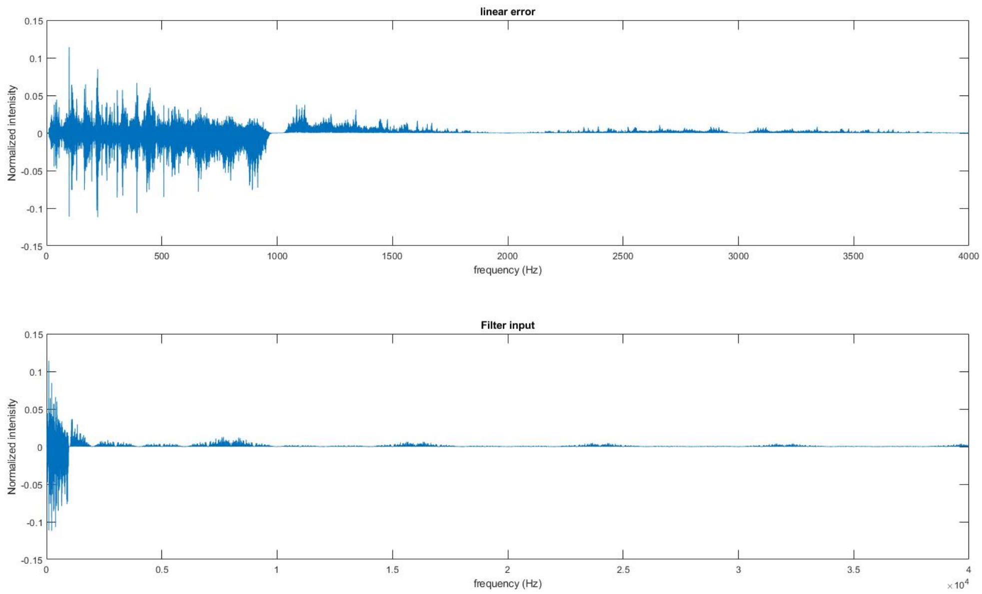


Figure 5.6: Effects of analogue sending when using linear interpolation.

Linear has more distortion over the entire bandwidth, whilst cubic spline mainly distorts the latest 100Hz of the spectrum. Cubic spline outperforms linear but is more complicated to implement. Looking back at the tests that are described in 4.1.3, linear interpolation is enough for this use case. One advantage of using an analogue input is that the signal is not processed by the DSP, which means less delay.

The delay of the driver is hard to measure. As soon as the driver IC is activated and configured for an analog input, it will start putting out an high frequency PWM signal like can be seen in Fig.5.7, of which the duty cycle determines the swing of the of the actuator. Due to the pick up and amplification of background noise, there will always be an PWM signal even if the ESP32 does not output an signal. This results in the fact that the available low bandwidth oscilloscope was not able no detect difference in the PWM signal and therefore an trigger based test in which the delay of the driver is compared to the delay of the toggling of an GPIO pin. The datasheet [27] promises a response time of at least 45 ns, which is so small that it would have also been hard to measure correctly and be insignificant in scope of the complete system.

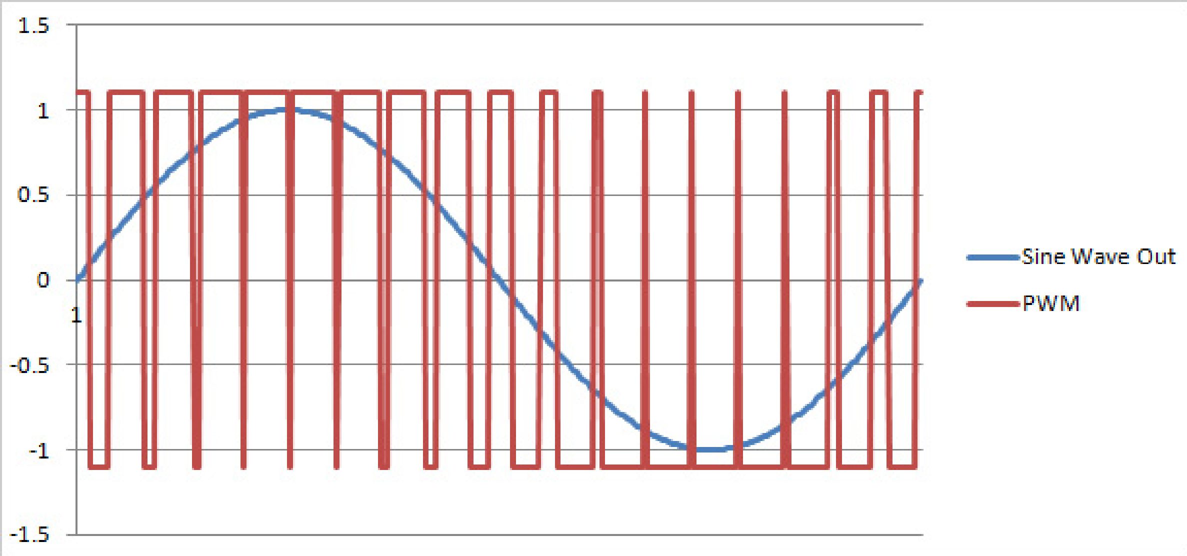


Figure 5.7: PWM amplification technique of Class D amplifier

5.3.2. Digital

the I2S connection was no success. No signal would come out of the driver when it was set up for a digital signal. The following has been tried to get it working, without success:

- Checking register values 3 times by different people.
- Different samplerates.
- Different clock speeds.
- Checked all the clock waveforms.
- Check if all the clocks are in sync/ need to be in sync
- Checked the I2S waveform.
- Switching Left/ right /averaged channels.
- Setting all amplifications to the maximum.

The input has been thoroughly examined and verified. As the driver does work when using an analogue signal, the most likely cause is a defect in the chip, a fault in either the example code and register manual, or human error on our side. After multiple days of attempts, this method was abandoned. For things that could still be tested, see appendix B.1.

The plan was to use DSP to filter out resonance frequency and use a lowpass filter. The first attempt at such a filter can be seen in Fig.5.8.

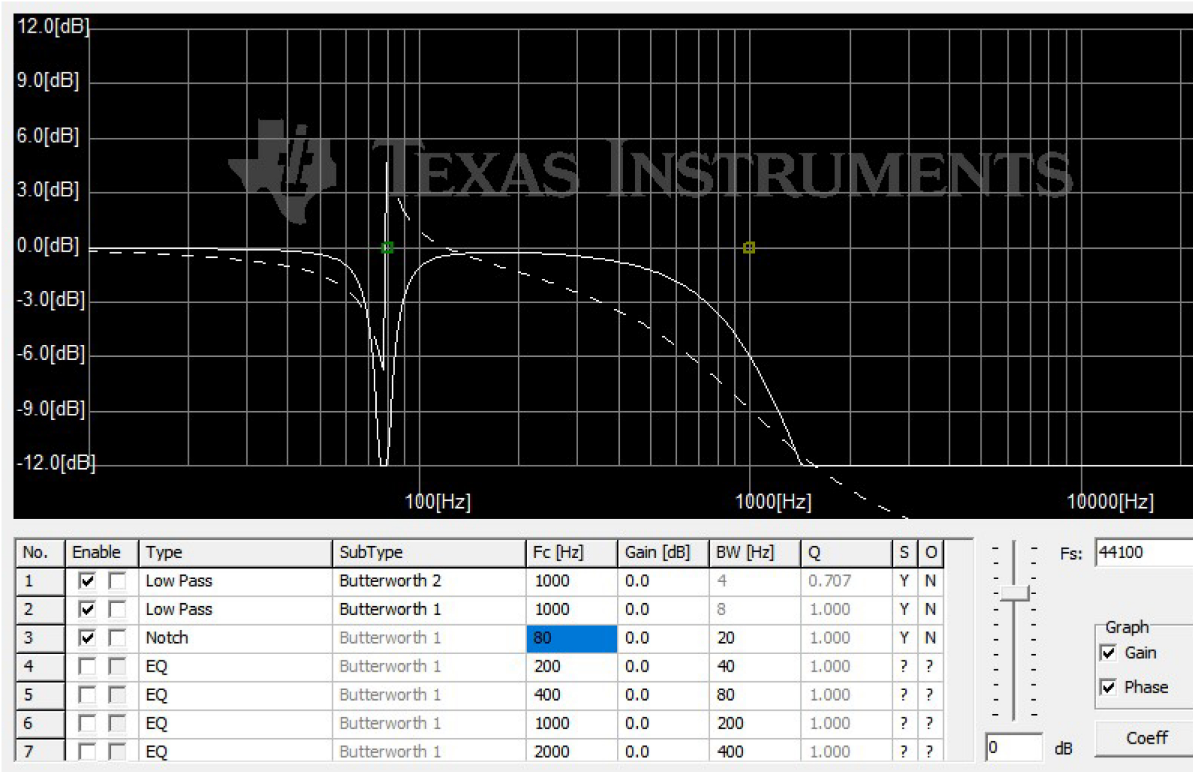


Figure 5.8: Plan for the DSP, made with a tool from TI specifically for this IC

5.3.3. Connectivity

Though wired connections could easily support live audio streaming, Wireless systems are less stable as they are subject to interference and other external factors. That is why the system can be set to work with on different frame rates and different sampling rates.

5.4. Revisions

5.4.1. Improvements from last revisions

During the test phase the digital I2S stream did not work. Finding out what was the cause was only made more difficult due to the lack of test points on the I2S channels and the general clock. This mistake was rectified in in the second edition. It had also come to attention that the clock required for the driver was higher than expected and had to be generated almost directly from the crystal in the ESP. This can only be done through certain pins. This was temporarily fixed by an extra wire on the prototype and noted down as a point of improvement for the second board.

5.4.2. 4 layers

One of the first decisions made for the revision was to use 4 layers. This allows for easier to route and compacter systems. This also simplifies separating different grounds. Though it did not cause any noticeable interference on the prototype for this system, There was no reason not to do it as it should improve performance.

5.5. Conclusion

The driver system is working and meets the mandatory requirements and scores well on the cost factors. Especially the hardware has proven flexible and very powerful; The driver can deliver plenty of power and is meant to drive audio equipment. As humans are much more sensitive to distorted audio than distorted vibrations[31]. The errors that are produced by the amplifier is smaller than a human can perceive. The main limit on the quality is in storing or receiving the needed data. When data is send live, the quality is mainly limited by the throughput that can be achieved within the latency constraints. When sending 2k samples/second and using linear interpolation to 8Khz, the SNR drops towards 10dB. This can be prevented by preloading, which replaces the constraints of latency with one of available memory and the amount of audio stored on the device. Any major improvement would be in this area and in the resonance tuning. The digital connection is not working due to what likely is a software error. The revision has more testing capabilities for this subsystem to get a digital system working. If this is successful the DSP could be used and the performance increased significantly. An overview of the final implementation can be seen in Fig.5.9

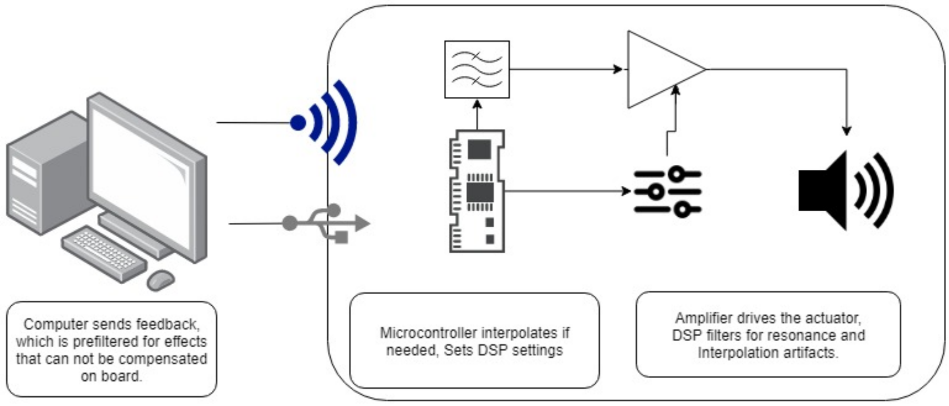


Figure 5.9: The overview updated to be in line with current topology

6 Recommendation

After going through two version of the product there are few conclusions and notes for the next version of the product.

6.1. Audio versus haptic

A lot of time has been spend on details as the feedback was treated as audio. It might be usefull to test if the effects that play big roles in audio are as big as a deal in vibrotactile vibrations. It would also be good to have a better understanding of the sensitivity and resolution that people perceive for this kind of feedback. This could significantly affect the constraints for further versions.

6.2. Pre-filtering

One of the major limits of the current system is the amount of onboard computing that has to be done in order to work with small delays and little data, whilst maintaining an output that is reasonable. During the project the team found that it was much easier to move the computing to the PC and send prefiltered signals to the setup. This idea is already portrayed in Fig.5.9

6.3. Wireless datalink

If SenseGlove or any other company wants to use a live audio link with their haptic feedback device, it is advised to use either a 2K or 4K samplerate. This is based on the experience that distortion is hard to notice for the users unless it is heard. Secondly, providing a 8K Sample rate would severely complicate high reliability low latency systems.

Sending audio during use is a big technical challenge and will always mean a degree of uncertainty and distortion. It would be much more practical to use preloaded data what has to be triggered using the same (wireless) link. This would avoid both the latency that an audiobuffer creates and the delay caused by resampling. The data can be stored in any resolution and is only limited by the amount of memory which can be stored, and the size of the message. A protocol for this could look like 6.1. This would support up to 255 audio files at 16 volumes and 16 effects like ramp up or down in just 3 bytes excluding headers.

Table 6.1: A possible data package for preloaded files

Order type	File number	(Optional) Volume & effect
1 Char	1 byte	1 byte total, 4bit each
8 bits	8 bits	8 bytes

Parametric would also be a viable option. with similar system the computer could send multiple frequencies and volumes for the ESP to generate. This would be a bigger workload for the microcontroller, but this that workload would be comparable to a more advanced interpolation method.

During working with wireless connections a few ideas were born which required more time than was available, but open the door to interesting solutions. The first idea is to make a Bluetooth based UDP system to reduce latency, though it would have to be combined with a system that would have a way with handling with missed packages. Secondly was to take a look at wireless mouses. These are highly reliable, extremely low latency and have a comparable throughput as preloading and parametric protocols.

6.4. time

The project was planned for 11 weeks of works, with 2 versions, research, and proof of concept included. This time could not be dedicated fully to the cause as some time had to be spend on an ethics course and a business plan. Even is the time was fully dedicated, the time constraints for the personal goals of the team were extremely tight. With an extra week in both revisions more thorough testing could have been done and better plans and designs could have been made. The planning can be found in appendix A.4,

A

General appendix

A.1. Schematic

A.1.1. Module overview

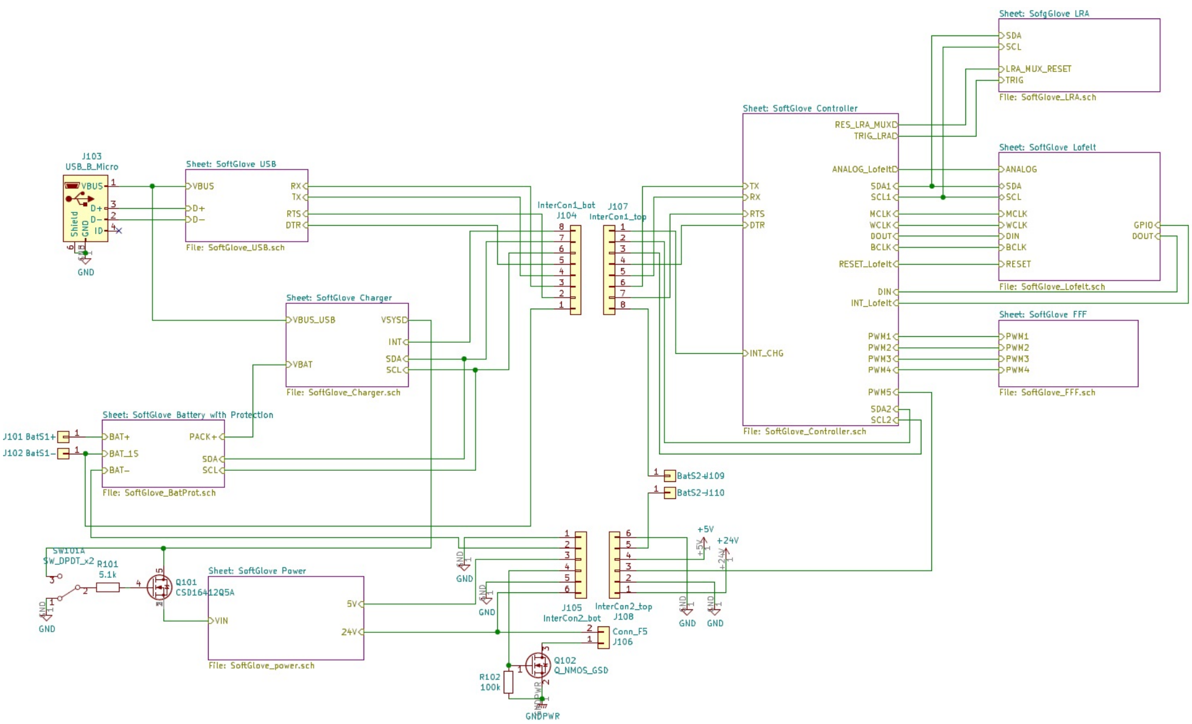


Figure A.1: Schematics of the complete system.

A.1.2. Battery charger

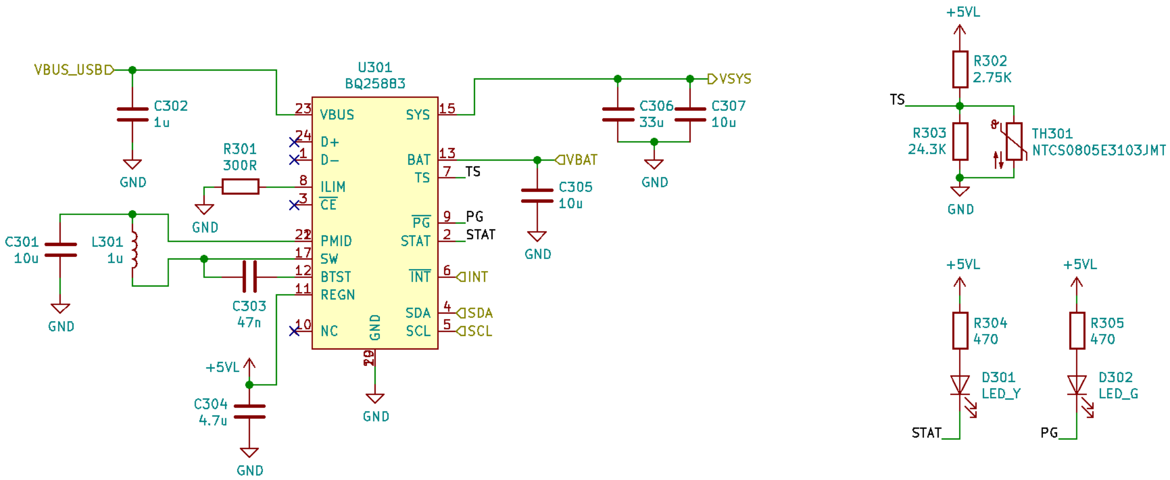


Figure A.2: Schematics of the battery charger.

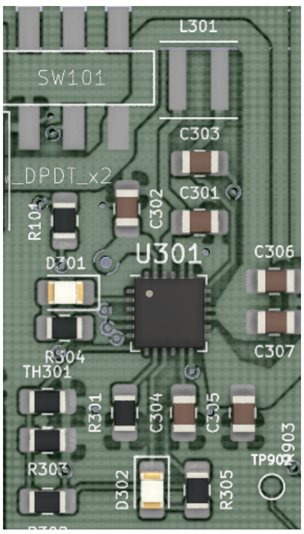


Figure A.3: PCB design of the battery charger.

A.1.3. Battery protection and USB

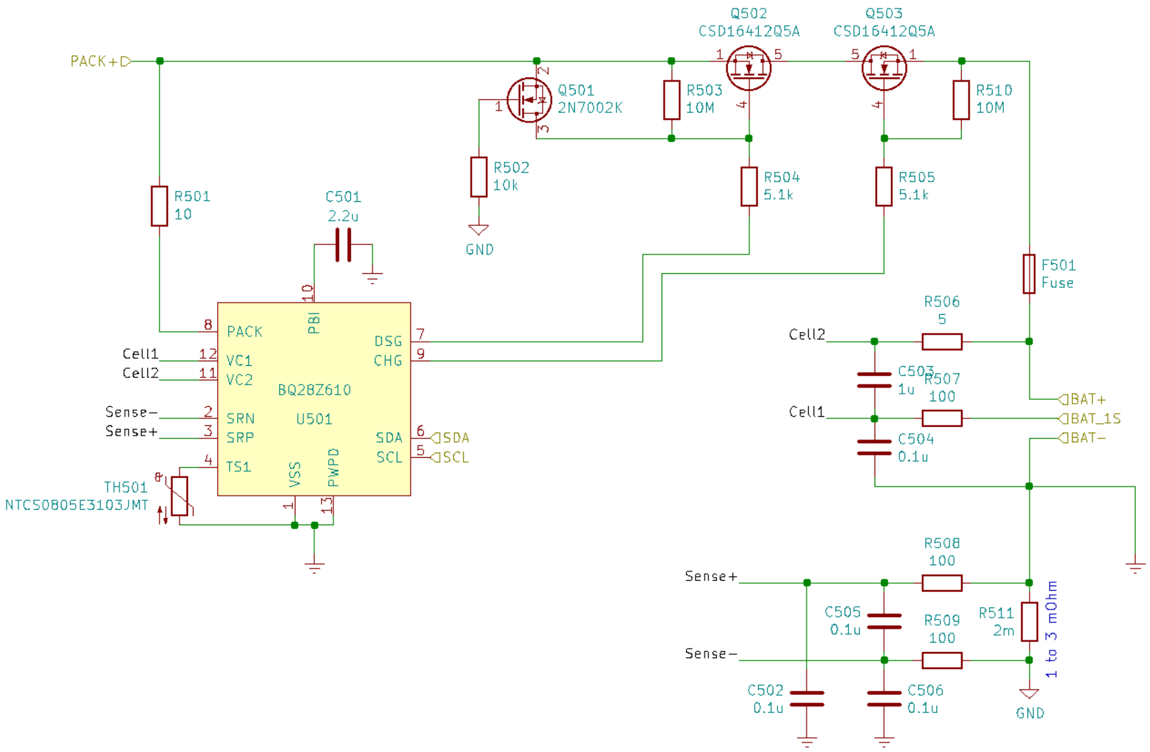


Figure A.4: Schematics of the battery protection.

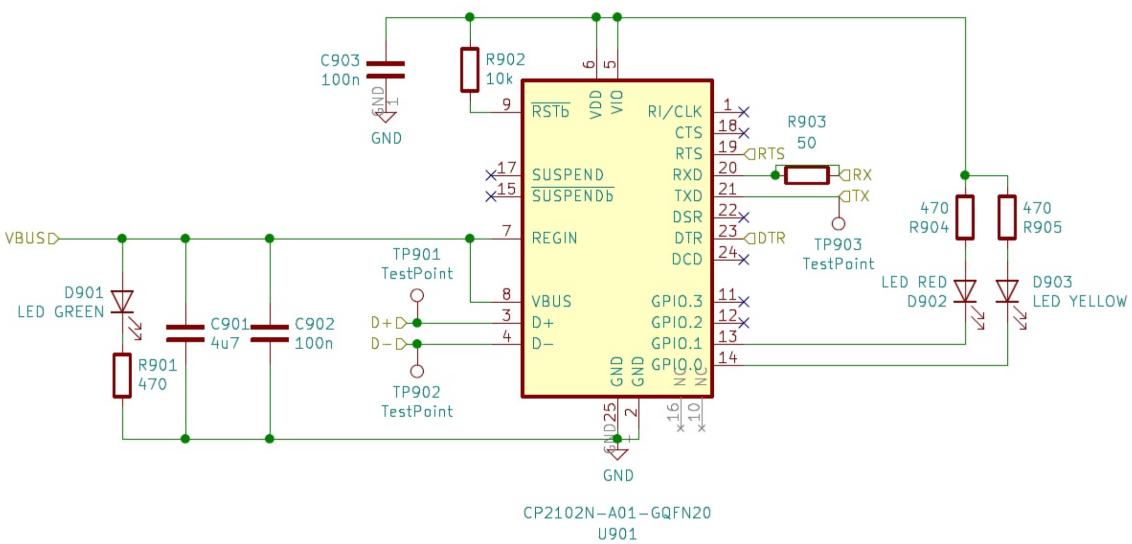


Figure A.5: Schematics of the USB to serial.

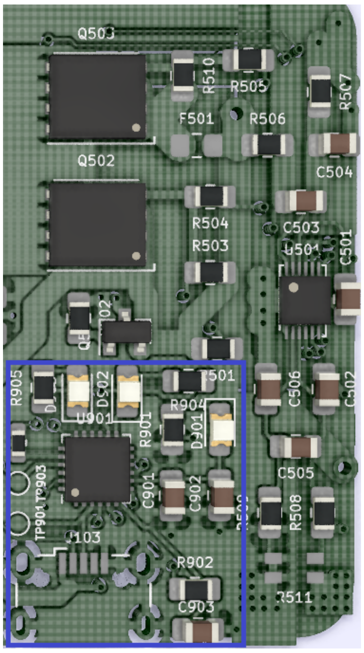


Figure A.6: PCB design of the battery protection and USB to serial design in the blue box.

A.1.4. ESP Layout

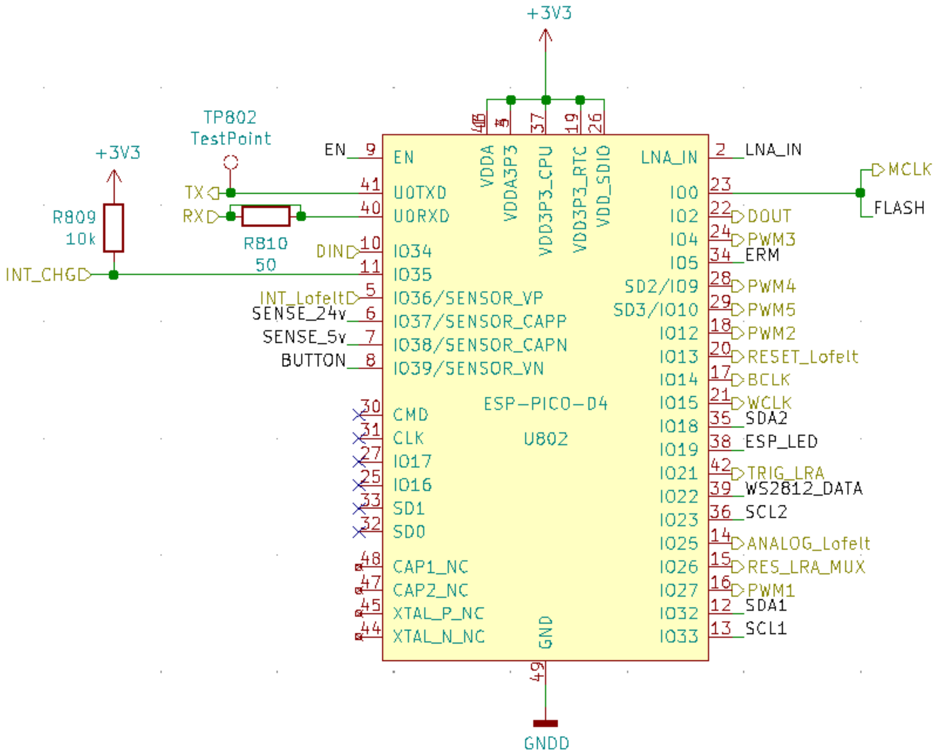


Figure A.7: The layout of the ESP with all pin connections.

A.1.5. ESP Schematics

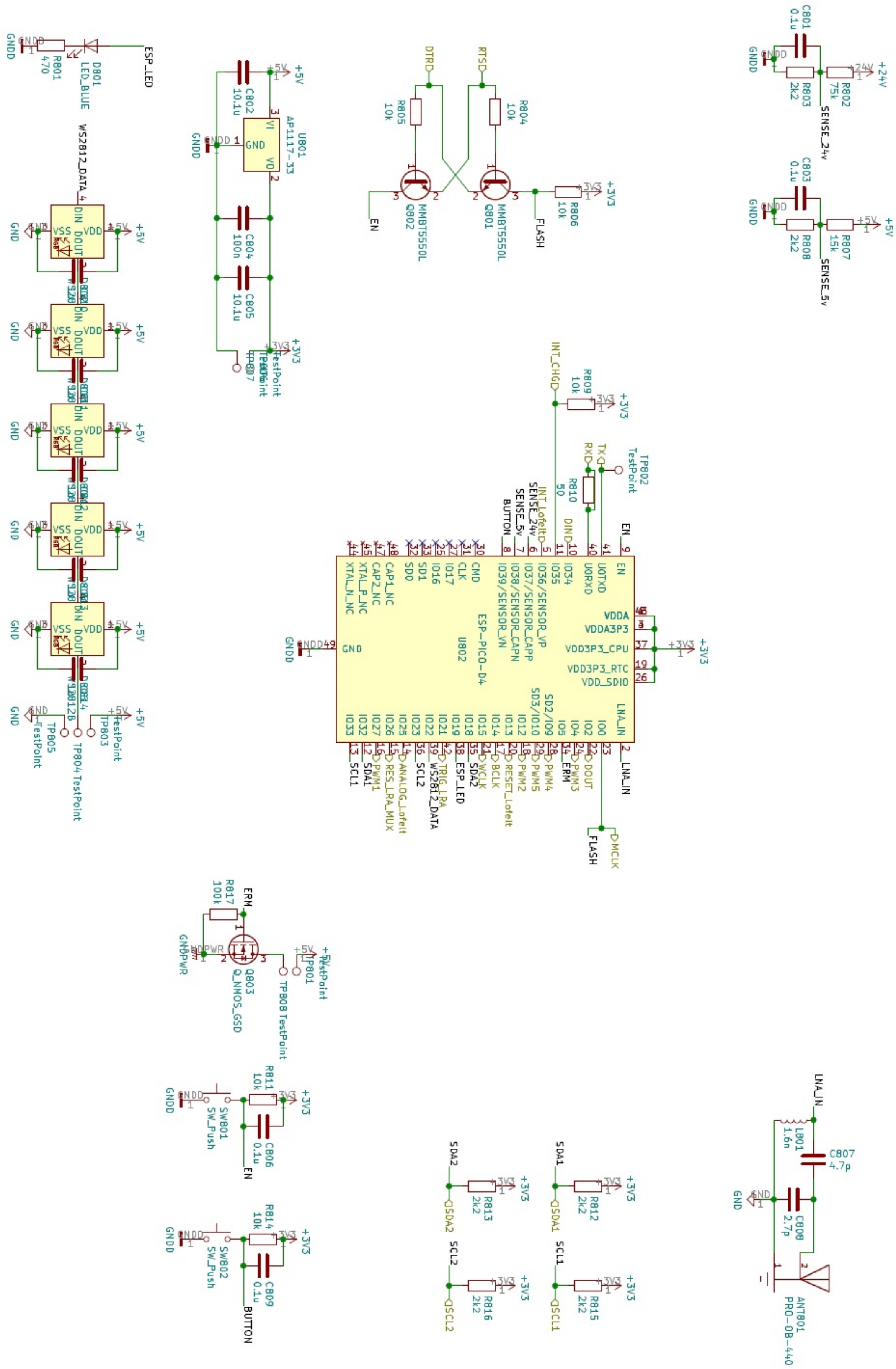
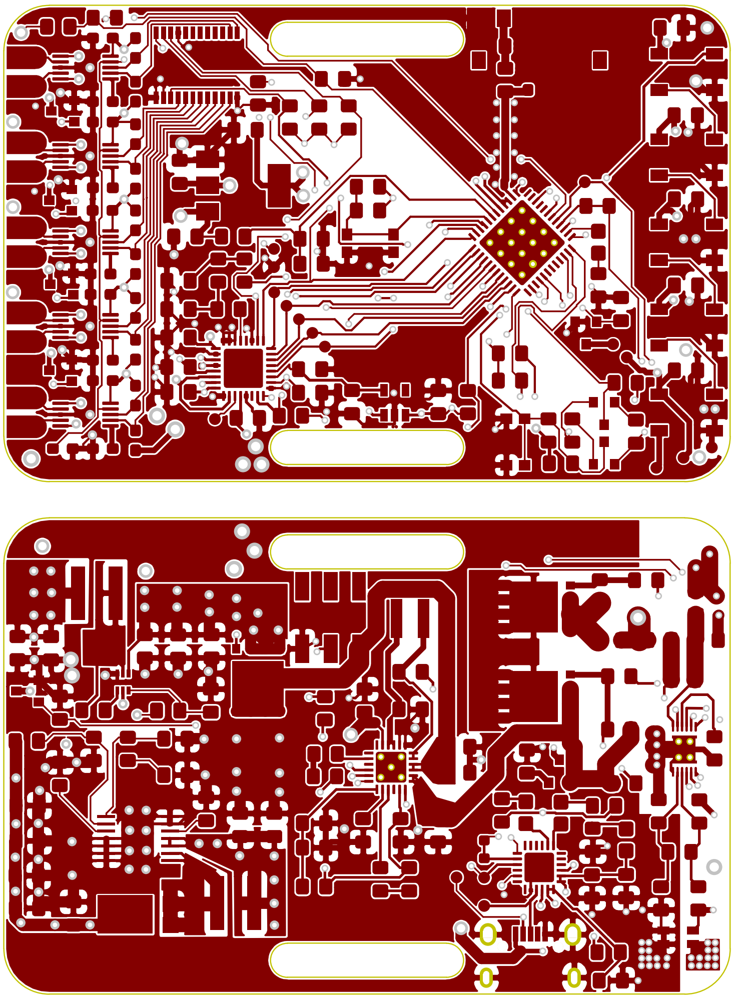


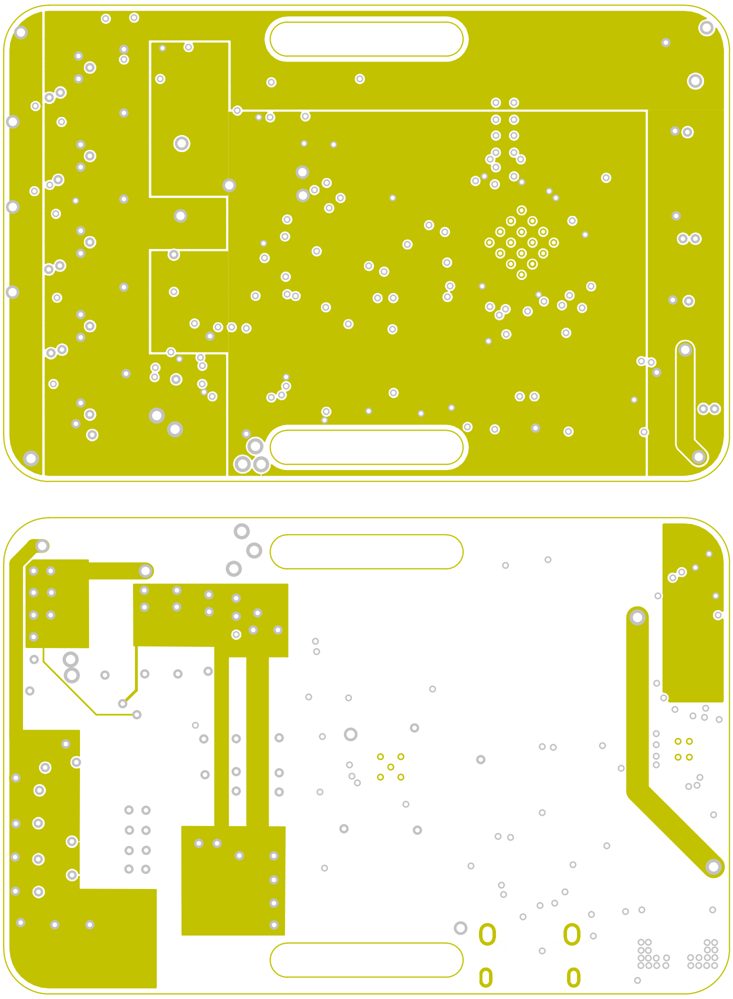
Figure A.8: Schematics of ESP.

A.2. PCB Structure of all layers

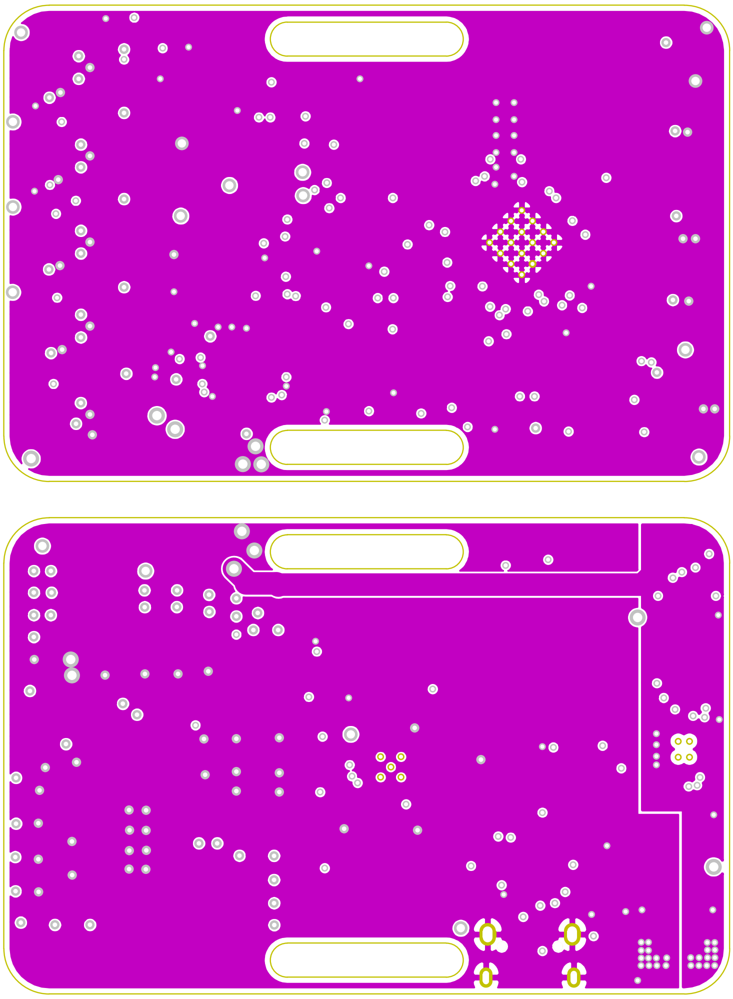
A.2.1. Copper layer 1



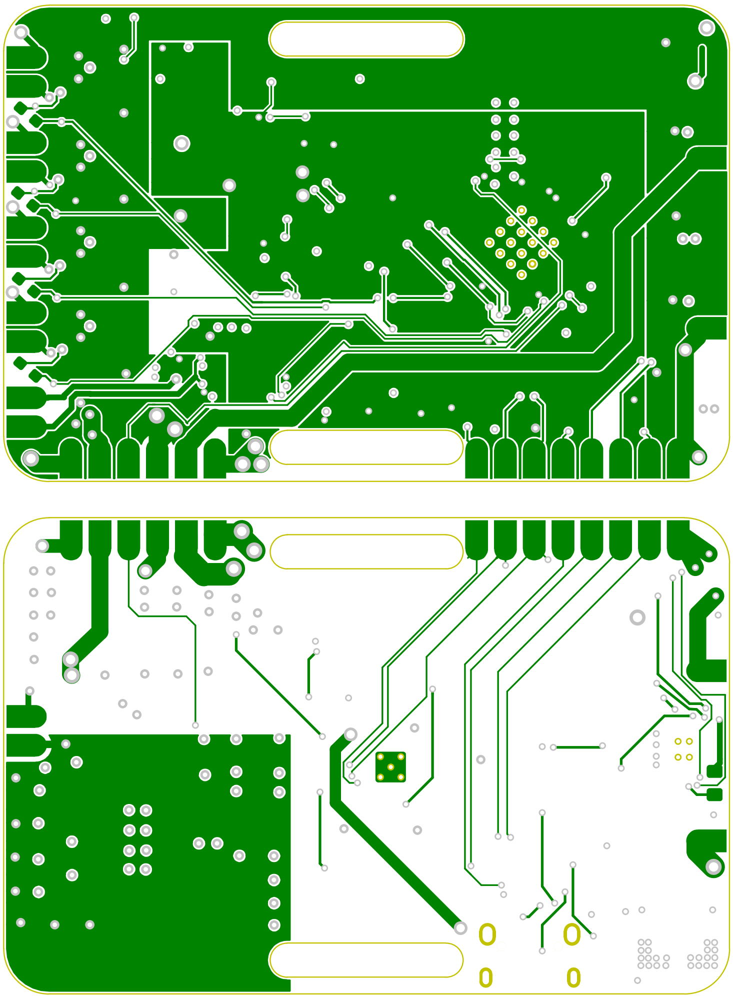
A.2.2. Copper layer 2



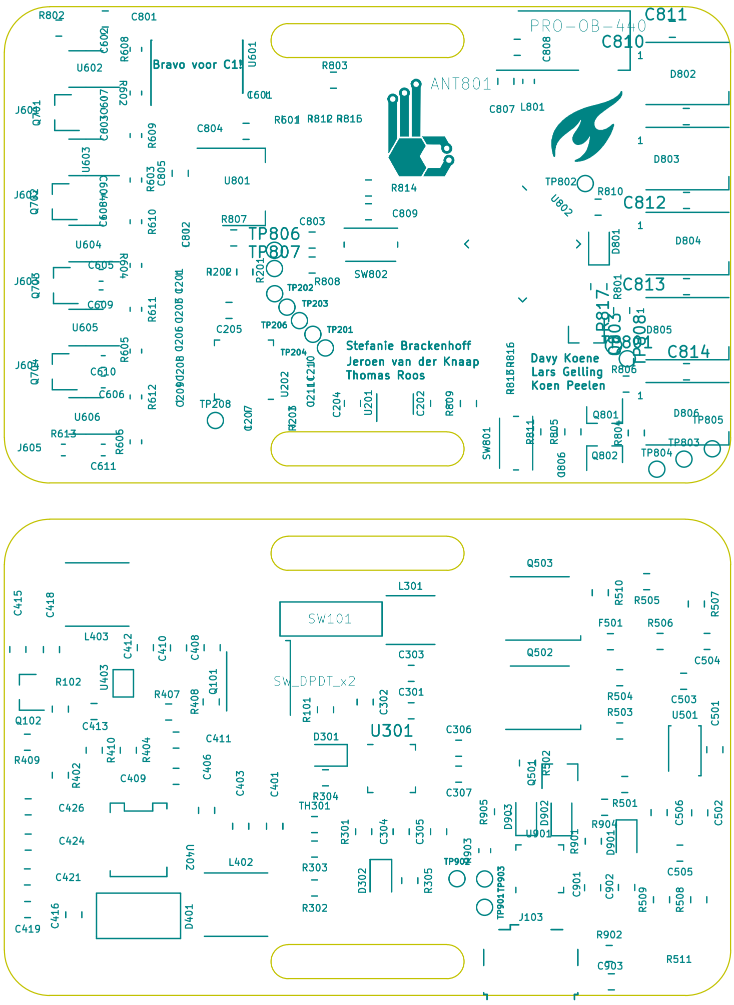
A.2.3. Copper layer 3



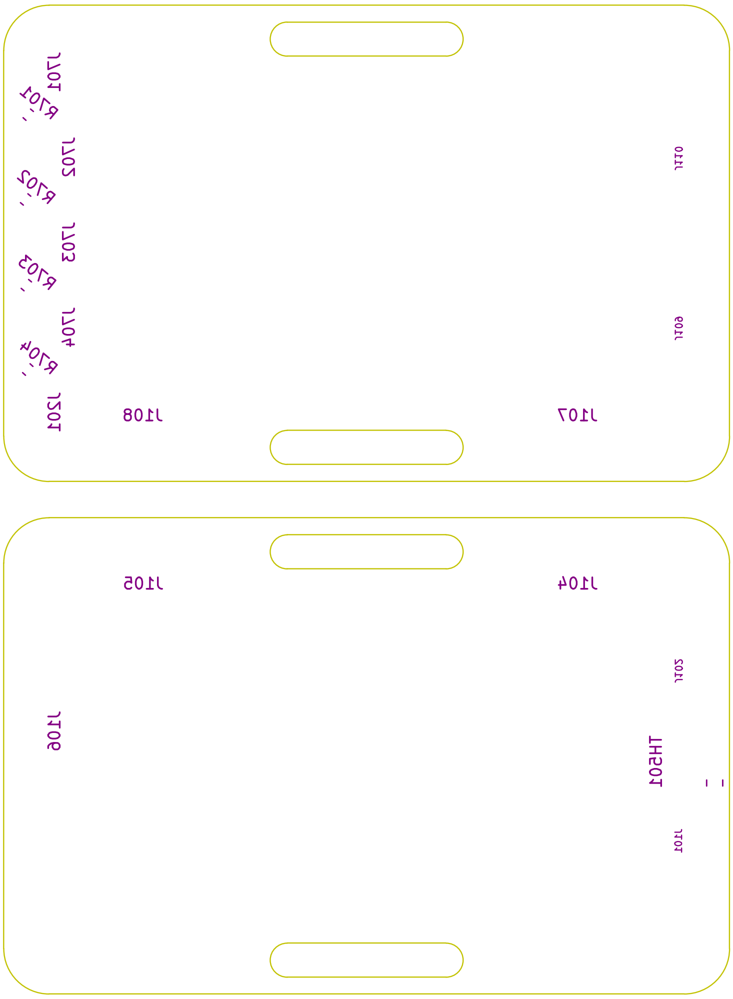
A.2.4. Copper layer 4



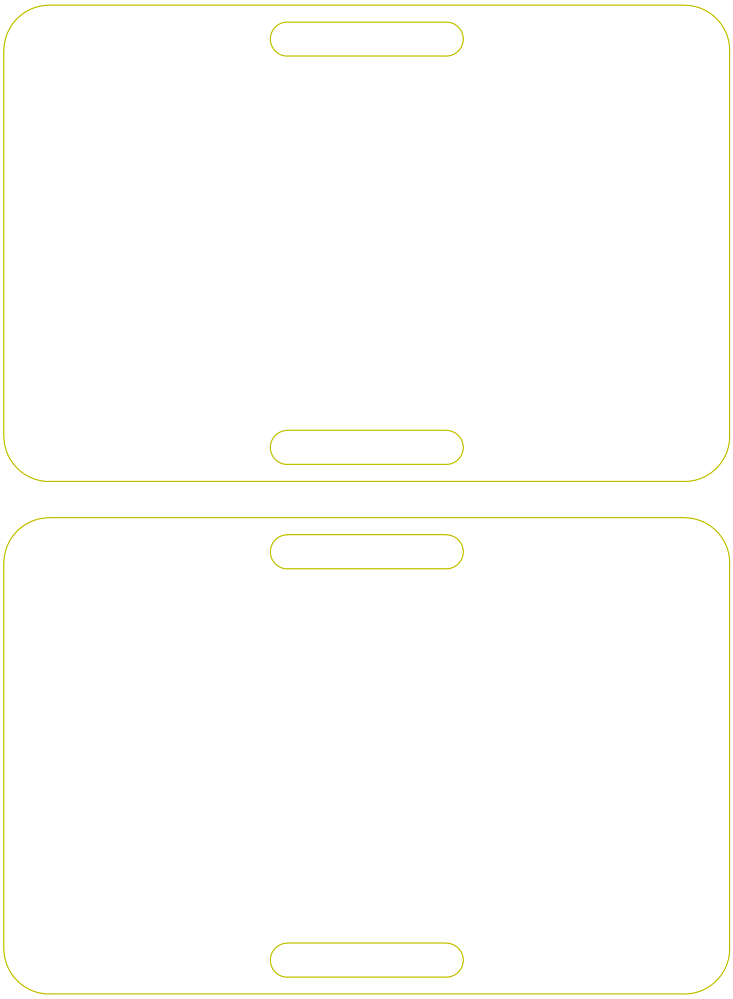
A.2.5. Silkscreen top



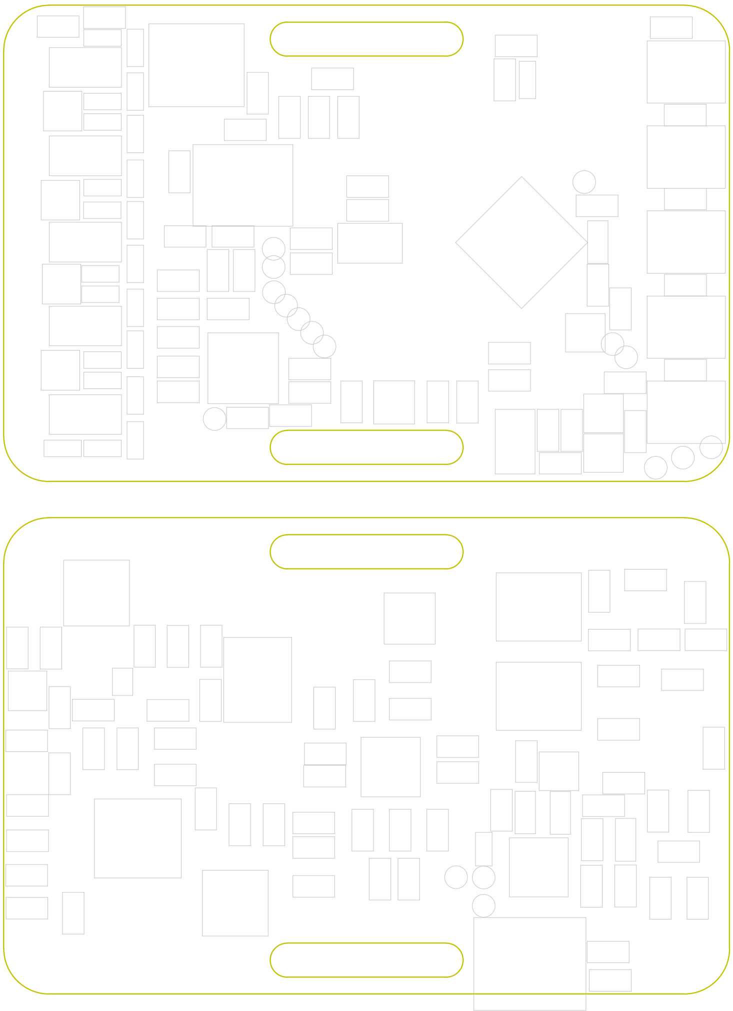
A.2.6. Silkscreen bottom



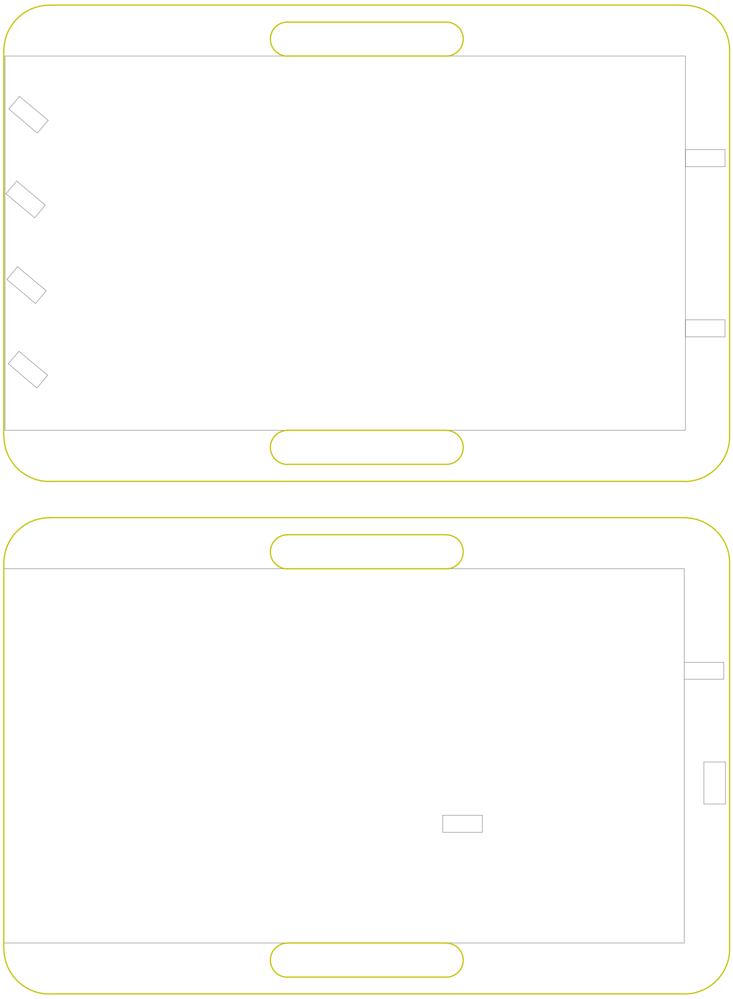
A.2.7. Edges and routing



A.2.8. Component placement top



A.2.9. Component placement bottom



A.3. Assignments

A.3.1. Old assignment

Sense Glove: Soft Glove Prototyping

Bachelor Final Project

Company:
At Sense Glove we develop a VR glove that translates the hands of a user to the virtual world: the Senseglove. The capabilities of the Senseglove allow a user to handle virtual objects the same as real objects. Capabilities such as per finger force- and vibrotactile feedback in addition to accurate self-contained hand tracking. The Senseglove is used in training simulators for car mechanics in a digital factory, VR CAD, proxy robotics and many more. Currently, Sense Glove has produced and sold their initial development kit. In addition to selling the Senseglove, Sense Glove helps companies to integrate interactable physics into existing VR environments. With the current development kits targeting the business-to-business market; a consumer version will be designed.

Problem:
The current Sense Glove uses an exoskeleton to track the position of the fingers and provide the force- and vibrotactile feedback. For Augmented Reality applications, an exoskeleton design is limiting the usability and its scale of implementation. Therefore, a "softglove" is required. The softglove needs to have similar capabilities as the SenseGlove exoskeleton, however the finger tracking will be excluded. With the launch of the Hololens 2, the finger tracking will be done with optical sensors from the head mounted displays.

Assignment
1. Design and realize a semi-flex PCB for the softglove, which integrates
   a. Per finger force feedback
   b. Linear Resonant Actuators in the fingertips
   c. Integration of LoFelt haptic drivers on the palm of the hand
2. Write firmware for the PCB, which can communicate to a PC through USB.
3. (Optional) Make it wireless through Bluetooth.

Sense Glove | MAKING VIRTUAL REALITY REAL

A.3.2. New assignment

Sense Glove: Soft Glove Prototyping
Bachelor Final Project

Company

At Sense Glove we develop a VR glove that translates the hands of a user to the virtual world: the Senseglove. The capabilities of the Senseglove allow a user to handle virtual objects the same as real objects. Capabilities such as per finger force- and vibrotactile feedback in addition to accurate self-contained hand tracking. The Senseglove is used in training simulators for car mechanics in a digital factory, VR CAD, proxy robotics and many more. Currently, Sense Glove has produced and sold their initial development kit. In addition to selling the Senseglove, Sense Glove helps companies to integrate interactable physics into existing VR environments. With the current development kits targeting the business-to-business market; a consumer version will be designed.

Problem

The current Sense Glove uses an exoskeleton to track the position of the fingers and provide the force- and vibrotactile feedback. For Augmented Reality applications, an exoskeleton design is limiting the usability and its scale of implementation. Therefore, a “softglove” is required. The softglove needs to have similar capabilities as the SenseGlove exoskeleton, however the finger tracking will be excluded. With the launch of the Hololens 2, the finger tracking will be done with optical sensors from the head mounted displays.

Assignment

Design and realize a PCB:

- With a formfactor that does not interfere with the movement of the hand.
- Which integrates the following feedback methods:
  - Per finger force.
  - Linear Resonant. Actuators on the fingers
  - Integration of LoFelt actuator on the palm of the hand.
- (Wish) Write firmware for the glove which integrates with SenseGlove’s systems.
- No immersion-breaking latency.
- (Optional) Make a wireless datalink.
- (Optional) Powered by a battery.

A.4. Planning

A.4.1. New assignment

Softglove - Work packages

ID	Subject	Start date	Finish date
48	Start project	23-04-2019	23-04-2019
43	Literature study	24-04-2019	01-05-2019
70	Reading up on HW	01-05-2019	04-05-2019
65	Proof of concept	01-05-2019	12-05-2019
46	Literatuur studie	02-05-2019	02-05-2019
69	Tests w/o micro or only Arduino	05-05-2019	12-05-2019
47	GreenLight Planning	10-05-2019	10-05-2019
55	Proto version	13-05-2019	07-06-2019
59	Draw schematic and PCB of proto version	13-05-2019	22-05-2019
53	Topic proposal Ethics	16-05-2019	16-05-2019
67	Code proto software	22-05-2019	01-06-2019
54	Proto PCB being manufactured and parts shipped	22-05-2019	29-05-2019
68	Order proto PCB	22-05-2019	22-05-2019
71	Greenlight deadline	27-05-2019	27-05-2019
64	Assemble proto version	29-05-2019	01-06-2019
63	Test and check prototype	01-06-2019	07-06-2019
49	First Full Draft Ethics	06-06-2019	06-06-2019
52	Final version	07-06-2019	01-07-2019
58	Redraw schematic and PCB	07-06-2019	14-06-2019
45	Final PCB being manufactured and parts shipped	14-06-2019	21-06-2019
72	Writing report	14-06-2019	20-06-2019
42	Report: final deadline	21-06-2019	21-06-2019
62	Assembling final version	21-06-2019	25-06-2019
61	Coding final demo code	24-06-2019	01-07-2019
50	Ethics: final deadline	27-06-2019	27-06-2019
60	Creating presentation	02-07-2019	04-07-2019

19-06-2019 *1/2*

B

Appendix extra information

B.1. What could be tested for digital connections

- Different I2C clock.
- Trying different ouput methods.
- Sending I2S clock whilst setting the necessary values in the registers.
- Higher NDAC values.
- Using a crystal/signal generator to generate a clock, avoiding PLL.

B.2. Lofelt EVK choives

B.2.1. should we use the EVK topology?

Advantages	Disadvantage
Wide input choice	DPS does little for this project
Easy to ”Design”	The DSP can not be altered by us
Boost and Auto-resonance is build in	Firmware and Documentation is hard to come by
	The board is bigger than need be.
	The board is expensive

B.2.2. Cutout vs copying

Cut out		Copy	
+	It is easier than copying	+	It can be made smaller than the EVK
+	It is quicker than copying	-	Hard packages to solder
-	The board is big	-	Requires firmware and documentation
-	EVK board is expensive		

C

Appendix Codes

```
% filename = 'toto.wav';
% [Samples,Fsampling] = audioread(filename);
% Samples = Samples.';


%% Set global variables
% Fsampling   = 2000;                     % set samplinbg frequency
Fplayback   = 20*8000;                  % set play frequency to 4*
Tsampling   = 1/Fsampling;              % set sample period
Tplayback   = 1/Fplayback;              % set playback period
Factor      = (Fplayback/Fsampling);% how many samples need to be made
Playtime    = 1;                        % sample time

%% Sample generation & few things
% Ftune = 458; %Hz

% old code for testing 1 frequency %
% Tsamples     = 0:Tsampling:Playtime-Tsampling;  %set sample time stamps
% Tplay        = 0:Tplayback:Playtime-Tplayback;  %set play time stamps

% Samples      = 0.3*sin(4*pi*Tsamples*Ftune)+0.3*sin(2*pi*Tsamples*Ftune)+0.3*sin(pi*Tsam
%sample

N_sample    = length(Samples);
N_playback  = length(Samples)*Factor;

for i = 1:N_sample
    Tsamples(i)    = (i-1)*Tsampling;
end %reconstructed original timestamps
for i = 1:N_playback
    Tplay(i)    = (i-1)*Tplayback;
end %reconstructed interpolated timestamps

L           = N_playback/2;                     % resolution FFT interpolation
F_axis      = 0:(Fplayback/(N_playback)):(Fplayback/2)-(Fplayback/(N_playback));
l           = N_sample/2;                       % resolution FFT original
F_axis_og   = 0:(Fsampling/(N_sample)):(Fsampling/2)-(Fsampling/(N_sample));

F_Original = fft(Samples); %2sided spec
```

```
F_Original = abs(F_Original); %absolute
N = max(F_Original);
F_Original = F_Original./N;
F_Original = F_Original(1:l); %take single side spec
F_Original = [F_Original,zeros(1,l*(Factor-1))]; %zero padding to make axis

subplot(3,2,1);
plot(F_axis,F_Original)
title('Original')
xlabel('frequency␣(Hz)')
ylabel('Normalized␣intenisity')



%% Sample and hold
S_sampleandhold = repelem(Samples,Factor);


% plot(Tplay,S_sampleandhold,'-x',Tsamples, Samples,'o')
% xlabel('Time')
% ylabel('Amplitude')
% legend('resampled signal','original samples')

F_sampleandhold = fft(S_sampleandhold); %2sided spec
F_sampleandhold = abs(F_sampleandhold)./(N*Factor); %absolute
F_sampleandhold = F_sampleandhold(1:L); %take single side spec

subplot(3,2,2);
plot(F_axis,F_sampleandhold)
title('Sample␣and␣hold')
xlabel('frequency␣(Hz)')
ylabel('Normalized␣intenisity')


%% Sample interpolation
% Tplayi      = 0:Tplayback:Playtime-Factor*Tplayback;  %set play time stamps
for i = 1:N_playback-Factor*2
    Tplayi(i)    = i*Tplayback;
end %reconstructed interpolated timestamps
% S_interpolate = interpft(Samples,N_playback);
S_interpolate = interp1(Tsamples,Samples,Tplayi);
F_interpolate = fft(S_interpolate); %2sided spec
F_interpolate = abs(F_interpolate)./(N*Factor); %absolute
F_interpolate = F_interpolate(1:L); %take single side spec

% plot(Tplay,S_interpolate,'-x',Tsamples, Samples,'o')
% legend('resampled signal','original samples')
% xlabel('Time')
% ylabel('Amplitude')

subplot(3,2,3);
plot(F_axis,F_interpolate)
title('interpolation')
xlabel('frequency␣(Hz)')
ylabel('Normalized␣intenisity')
```

```
%% Sample add zero
S_addzero=[zeros(1,N_playback)];
for i = 1:N_sample
    S_addzero(i+(i-1)*(Factor-1)) = Samples(i);
end
F_addzero = fft2(S_addzero); %2sided spec
F_addzero = abs(F_addzero)./N; %absolute
F_addzero = F_addzero(1:L); %take single side spec

% plot(Tplay,S_addzero,'-x',Tsamples, Samples,'o')
% legend('resampled signal','original samples')
% xlabel('Time')
% ylabel('Amplitude')

subplot(3,2,4);
plot(F_axis,F_addzero)
title('add zero FFT')
xlabel('frequency (Hz)')
ylabel('Normalized intenisity')


%% Cubic
S_cubic = spline(Tsamples,Samples,Tplay);

F_cubic = fft(S_cubic); %2sided spec
F_cubic = abs(F_cubic)./(N*Factor); %absolute
F_cubic = F_cubic(1:L); %take single side spec

% plot(Tplay,S_cubic,'-x',Tsamples, Samples,'o')
% xlabel('Time')
% ylabel('Amplitude')
% legend('resampled signal','original samples')

subplot(3,2,5);
plot(F_axis,F_cubic)
title('cubic FFT')
xlabel('frequency (Hz)')
ylabel('Normalized intenisity')


clearvars
Factor1         = 4;% how many samples need to be made
F_sampling = 2000;
F_playback1  = F_sampling*Factor1;
F_playback2  = 80000;    % set play frequency to 4*
Factor2      = F_playback2/F_playback1;

%% getting samples
filename = 'toto.wav';
[Samples,] = audioread(filename);
Samples = Samples.';

if (mod(length(Samples),2)== 1)
   Samples(length(Samples))= [];
end
N_samples= length(Samples);
```

```
F_sampling = 2000;
Tsampling  = 1/F_sampling;          % set sample period

for i = 1:N_samples
    Tsamples(i)    = (i-1)*Tsampling;
end %reconstructed original timestamps

l            = N_samples/2;                           % resolution FFT original
F_axis_og    = 0:(F_sampling/(N_samples)):(F_sampling/2)-(F_sampling/(N_samples));

F_Original = fft(Samples); %2sided spec
F_Original = abs(F_Original); %absolute
N = max(F_Original);
F_Original = F_Original./N;
F_Original = F_Original(1:l); %take single side spec
F_Original1 = [F_Original,zeros(1,l*(Factor1-1))]; %zero padding to make axis
F_Original2 = [F_Original,zeros(1,l*(Factor1*Factor2-1))]; %zero padding to make axis


%% Sample stats 1

N_playback1 = Factor1*N_samples;            % set play frequency to 4*
T_playback1  = 1/F_playback1;          % set playback period

for i = 1:N_playback1
    Tplay1(i)    = (i-1)*T_playback1;
end

L1            = N_playback1/2;                         % resolution FFT interpolation
F_axis1       = 0:(F_playback1/(N_playback1)):(F_playback1/2)-(F_playback1/(N_playback


%% Sample stats 2
                 % how many samples need to be made
N_playback2  = Factor2*N_playback1;
T_playback2  = 1/F_playback2;          % set playback period

for i = 1:N_playback2
    Tplay2(i)    = (i-1)*T_playback2;
end

L2            = N_playback2/2;                         % resolution FFT interpolation
F_axis2       = 0:(F_playback2/(N_playback2)):(F_playback2/2)-(F_playback2/(N_playback


%% Cubic
S_cubic = spline(Tsamples,Samples,Tplay1);

F_cubic = fft(S_cubic); %2sided spec
F_cubic = abs(F_cubic)./(N*Factor1); %absolute
F_cubic = F_cubic(1:L1); %take single side spec

% plot(Tplay,S_cubic,'-x',Tsamples, Samples,'o')
% xlabel('Time')
% ylabel('Amplitude')
% legend('resampled signal','original samples')
```

```
for i = 1:N_playback1-Factor1*2
    Tplayi(i)   = i*T_playback1;
end %reconstructed interpolated timestamps
% S_interpolate = interpft(Samples,N_playback);
S_interpolate = interp1(Tsamples,Samples,Tplayi);
F_interpolate = fft(S_interpolate); %2sided spec
F_interpolate = abs(F_interpolate)./(N*Factor1); %absolute
F_interpolate = F_interpolate(1:L1); %take single side spec

%% Sample and hold to 40.000 (what you hear)
 S_sampleandhold = repelem(S_interpolate,Factor2);


% plot(Tplay,S_sampleandhold,'-x',Tsamples, Samples,'o')
% xlabel('Time')
% ylabel('Amplitude')
% legend('resampled signal','original samples')

F_sampleandhold = fft(S_sampleandhold); %2sided spec
F_sampleandhold = abs(F_sampleandhold)./(N*Factor1*Factor2); %absolute
F_sampleandhold = F_sampleandhold(1:L2); %take single side spec

subplot(2,2,1);
plot(F_axis1,F_Original1)
title('Original')
xlabel('frequency␣(Hz)')
ylabel('Normalized␣intenisity')

subplot(2,2,2);
plot(F_axis1,F_interpolate)
title('linear')
xlabel('frequency␣(Hz)')
ylabel('Normalized␣intenisity')

subplot(2,2,3);
plot(F_axis2,F_Original2)
title('Original')
xlabel('frequency␣(Hz)')
ylabel('Normalized␣intenisity')
subplot(2,2,4);
plot(F_axis2,F_sampleandhold)
title('Filter␣input')
xlabel('frequency␣(Hz)')
ylabel('Normalized␣intenisity')
```

D

Appendix D, plots and schematics

D.1. schematics

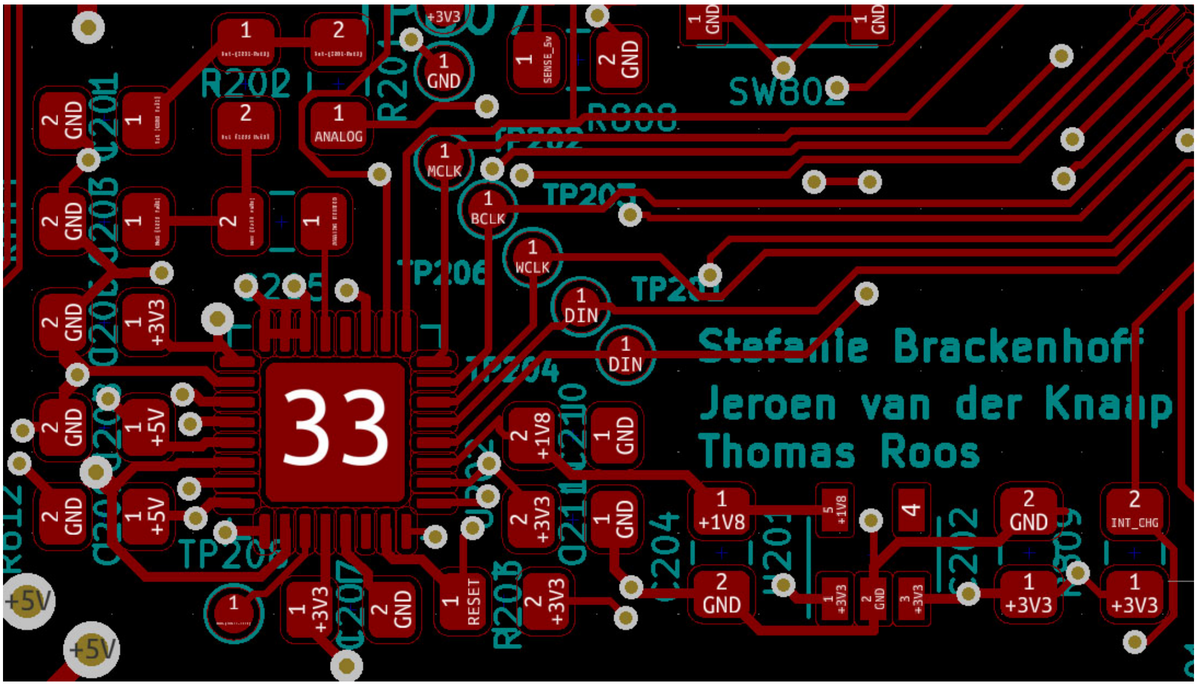


Figure D.1: Routed PCB, only the upper layer is rendered to improve clarity

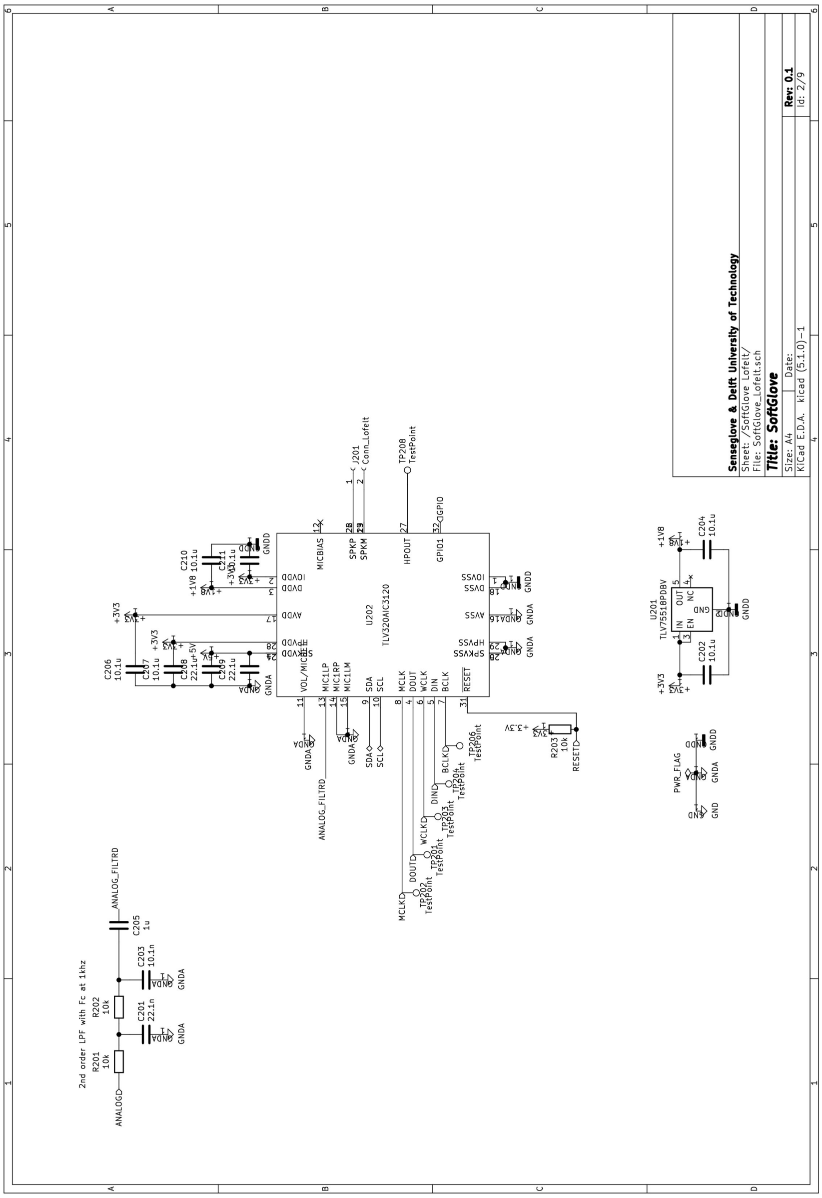


Figure D.2: revised schematic

D.2. interpolation

note: cubic interpolation and spline interpolation are the same. spline is a form of cubic interpolation.

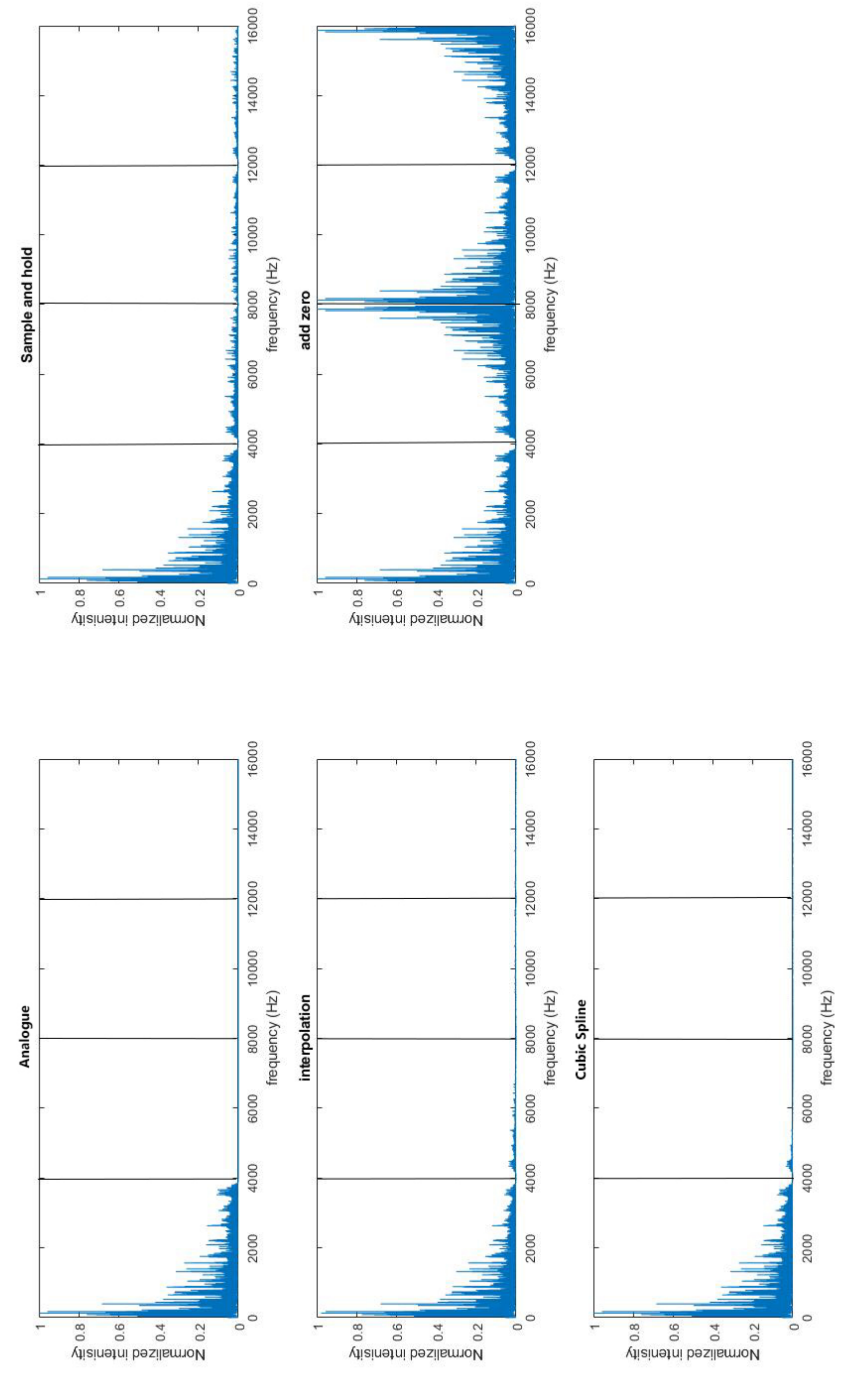


Figure D.3: overview of the effects of resampling methods

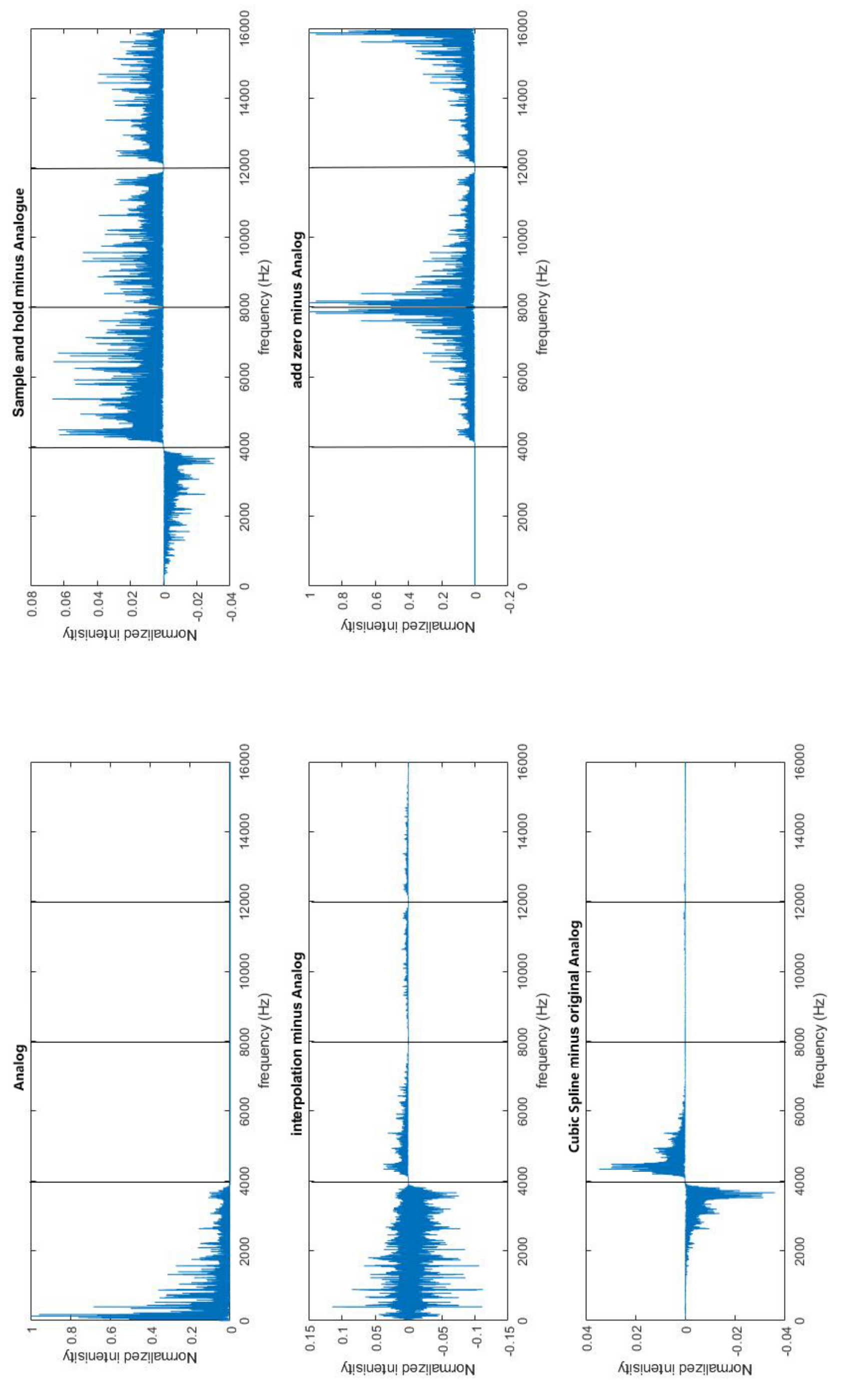
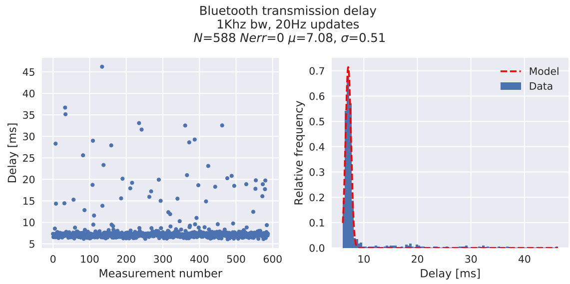
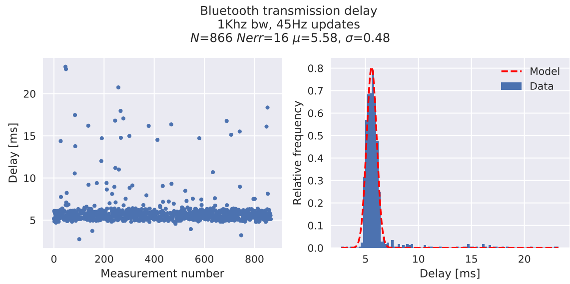


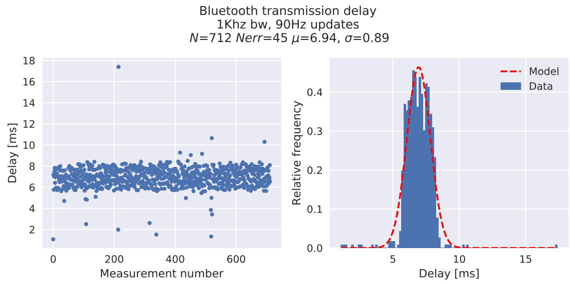
Figure D.4: overview of the errors of resampling methods, notice the sampling window and sidelobes of cubic and Sample and Hold



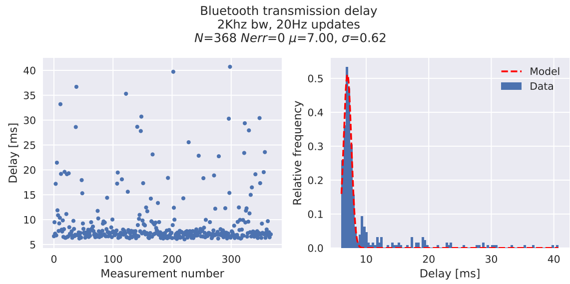
(a) Bluetooth test with Fs=2000, 20Hz



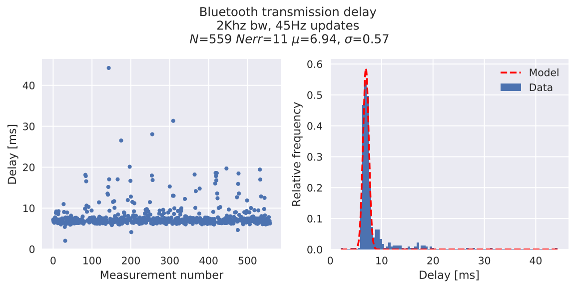
(b) Bluetooth test with Fs=2000, 45Hz



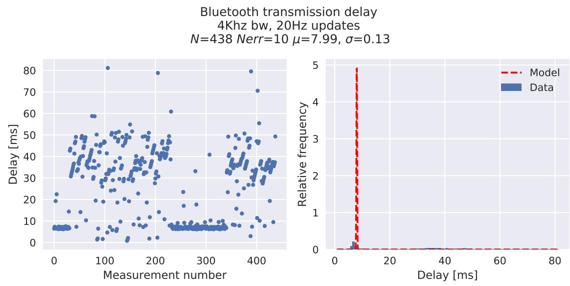
(c) Bluetooth test with Fs=2000, 90Hz



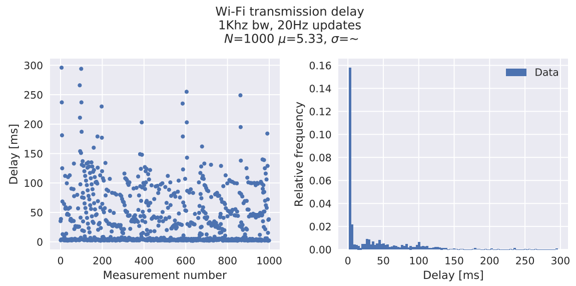
(d) Bluetooth test with Fs=4000, 20Hz



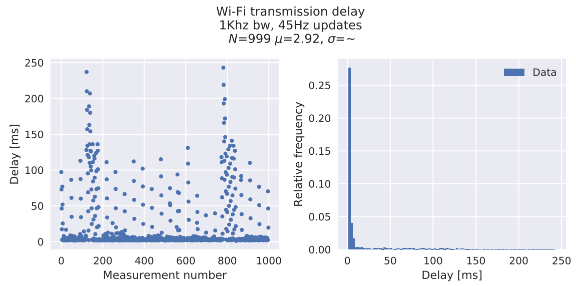
(e) Bluetooth test with Fs=4000, 45Hz



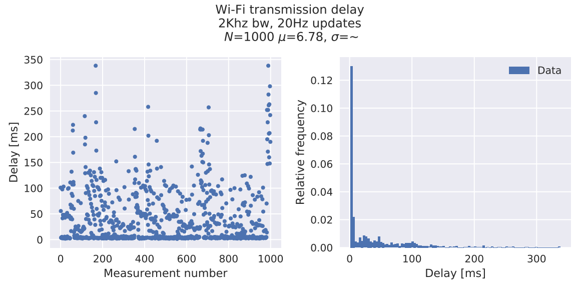
(f) Bluetooth test with Fs=8000, 90Hz



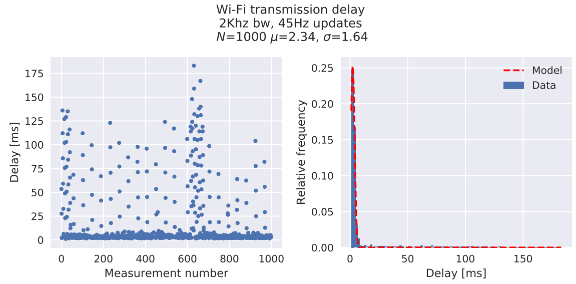
(a) Wi-Fi with AP test with Fs=2000, 20Hz



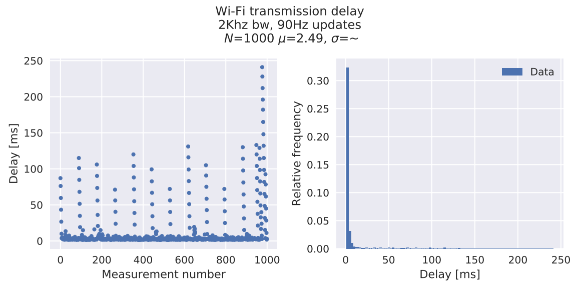
(b) Wi-Fi with AP test with Fs=2000, 45Hz



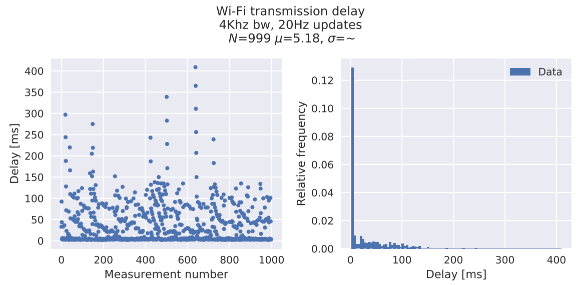
(c) Wi-Fi with AP test with Fs=4000, 20Hz



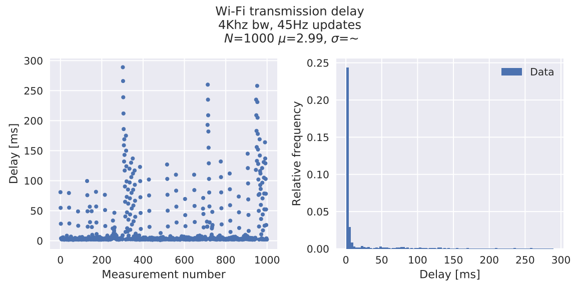
(d) Wi-Fi with AP test with Fs=4000, 45Hz



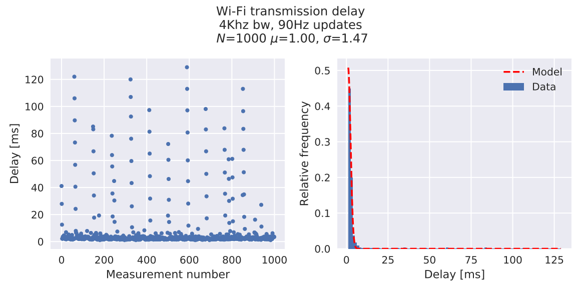
(e) Wi-Fi with AP test with Fs=4000, 90Hz



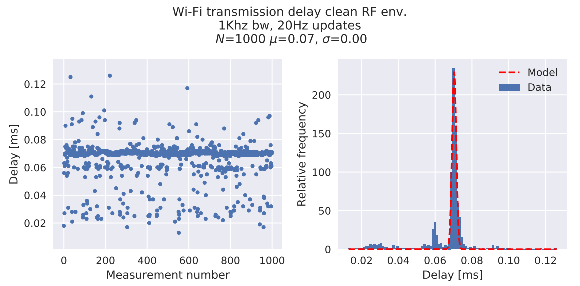
(f) Wi-Fi with AP test with Fs=8000, 20Hz



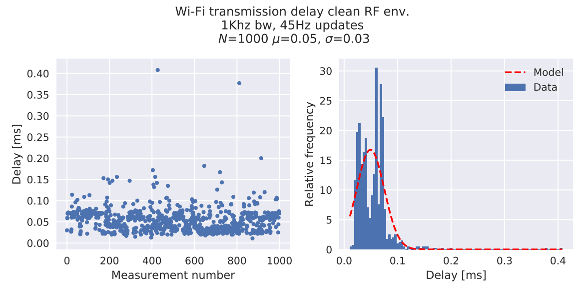
(g) Wi-Fi with AP test with Fs=8000, 45Hz



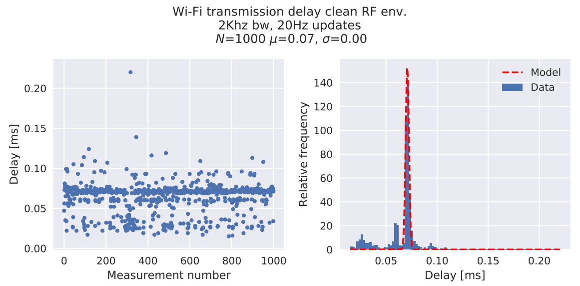
(h) Wi-Fi with AP test with Fs=8000, 90Hz



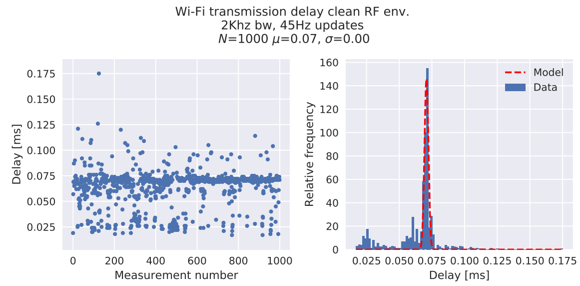
(a) Wi-Fi direct test in clean RF env. with Fs=2000, 20Hz



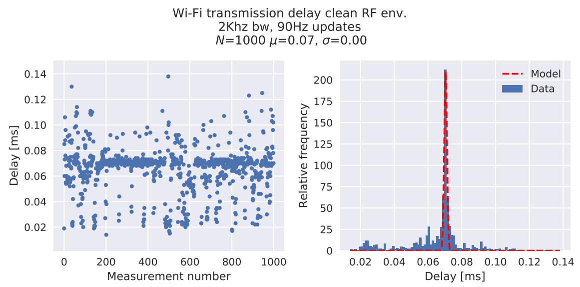
(b) Wi-Fi direct test in clean RF env. with Fs=2000, 45Hz



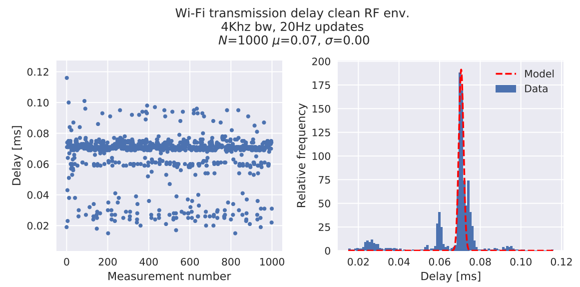
(c) Wi-Fi direct test in clean RF env. with Fs=4000, 20Hz



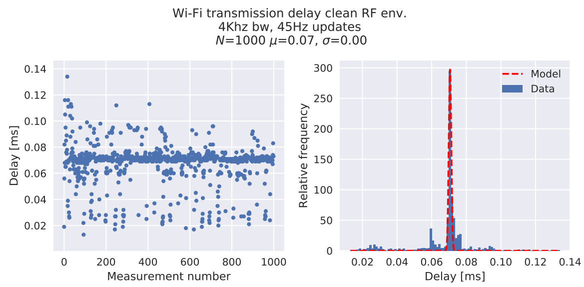
(d) Wi-Fi direct test in clean RF env. with Fs=4000, 45Hz



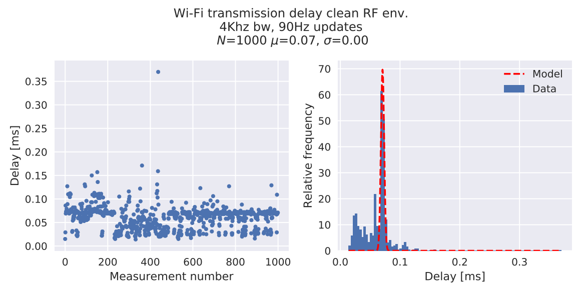
(e) Wi-Fi direct test in clean RF env. with Fs=4000, 90Hz



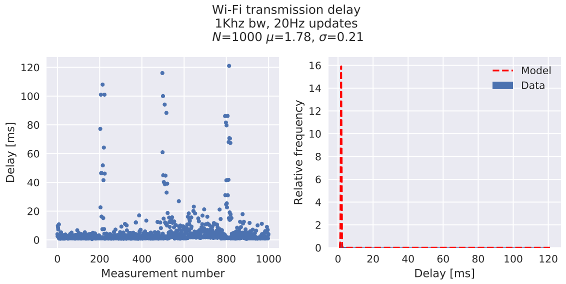
(f) Wi-Fi direct test in clean RF env. with Fs=8000, 20Hz



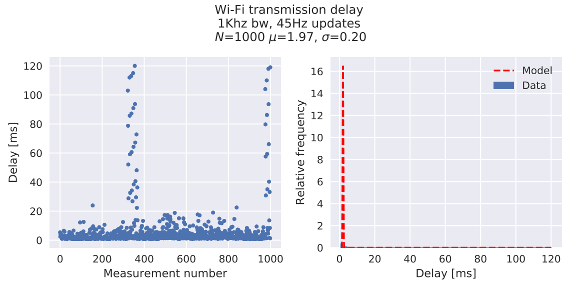
(g) Wi-Fi direct test in clean RF env. with Fs=8000, 45Hz



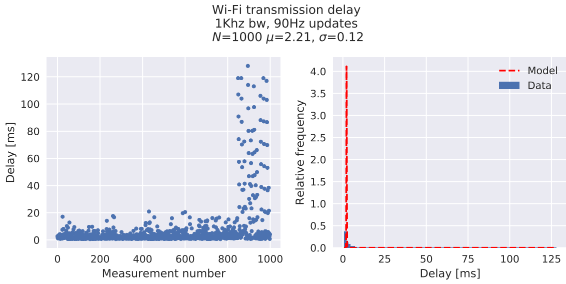
(h) Wi-Fi direct test in clean RF env. with Fs=8000, 90Hz



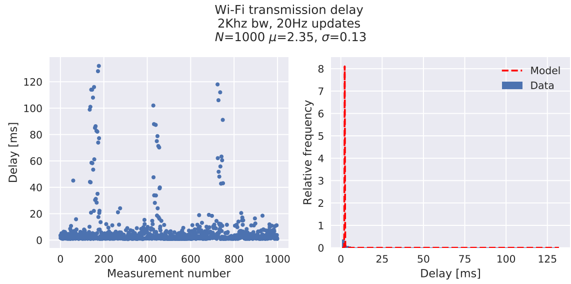
(a) Wi-Fi direct test in noise RF env. with Fs=2000, 20Hz



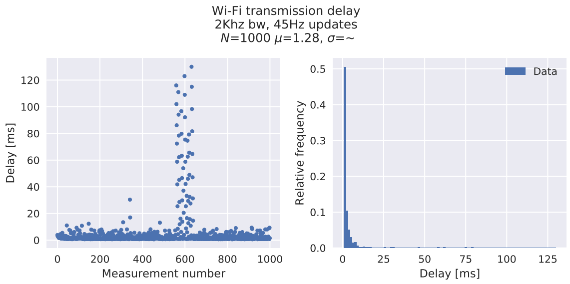
(b) Wi-Fi direct test in noise RF env. with Fs=2000, 45Hz



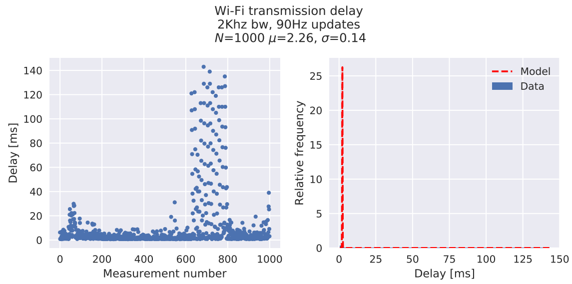
(c) Wi-Fi direct test in noise RF env. with Fs=2000, 90Hz



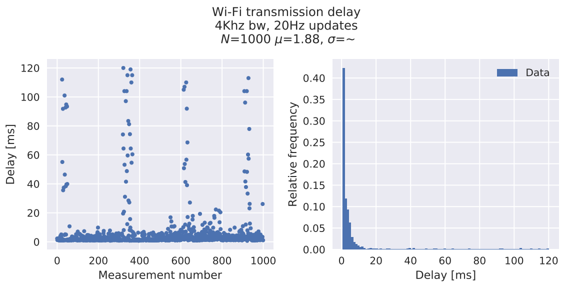
(d) Wi-Fi direct test in noise RF env. with Fs=4000, 20Hz



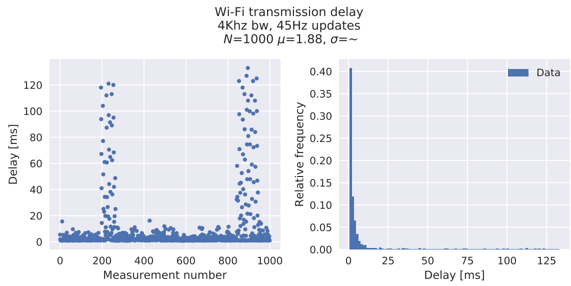
(e) Wi-Fi direct test in noise RF env. with Fs=4000, 45Hz



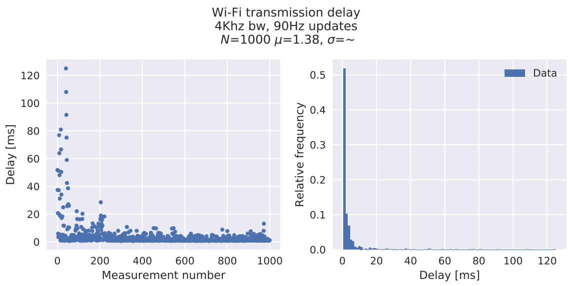
(f) Wi-Fi direct test in noise RF env. with Fs=4000, 90Hz



(g) Wi-Fi direct test in noise RF env. with Fs=8000, 20Hz



(h) Wi-Fi direct test in noise RF env. with Fs=8000, 45Hz



(i) Wi-Fi direct test in noise RF env. with Fs=8000, 90Hz

Bibliography

[1] O. Bau and I. Poupyrev, “Revel: Tactile feedback technology for augmented reality,” *ACM Trans. Graph.*, vol. 31, no. 4, pp. 89:1–89:11, Jul. 2012. [Online]. Available: http://doi.acm.org/10.1145/2185520.2185585

[2] Wikipedia, “Virtual reality headset,” 2019. [Online]. Available: https://en.wikipedia.org/wiki/Virtual_reality_headset

[3] transported, “Virtual reality for real estate,” 2019. [Online]. Available: https://www.transported.co/

[4] K. R. Lohse, C. G. E. Hilderman, K. L. Cheung, S. Tatla, and H. F. M. Van der Loos, “Virtual reality therapy for adults post-stroke: A systematic review and meta-analysis exploring virtual environments and commercial games in therapy,” *PLOS ONE*, vol. 9, no. 3, pp. 1–13, 03 2014. [Online]. Available: https://doi.org/10.1371/journal.pone.0093318

[5] P. P. M. E. E. B. M. J. G. R. H. B. S. A. Rizzo, J, “Development of vr therapy application for iraq war military personnel with ptsd,” in *Medicine Meets Virtual Reality 13: The Magical Next Becomes the Medical Now.* IOS press, 2005, pp. 407–414.

[6] E. V. d. P. Jerome Perret, “Review paper: Commercial haptic gloves,” 2018.

[7] E.-L. Sallnäs, “Haptic feedback increases perceived social presence,” in *Haptics: Generating and Perceiving Tangible Sensations*, A. M. L. Kappers, J. B. F. van Erp, W. M. Bergmann Tiest, and F. C. T. van der Helm, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 178–185.

[8] D. DiFilippo and D. K. Pai, “The ahi: An audio and haptic interface for contact interactions,” in *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology*, ser. UIST ’00. New York, NY, USA: ACM, 2000, pp. 149–158. [Online]. Available: http://doi.acm.org/10.1145/354401.354437

[9] C. Jay, M. Glencross, and R. Hubbold, “Modeling the effects of delayed haptic and visual feedback in a collaborative virtual environment,” *ACM Trans. Comput.-Hum. Interact.*, vol. 14, no. 2, Aug. 2007. [Online]. Available: http://doi.acm.org/10.1145/1275511.1275514

[10] T. J. Buker, D. A. Vincenzi, and J. E. Deaton, “The effect of apparent latency on simulator sickness while using a see-through helmet-mounted display: Reducing apparent latency with predictive compensation,” *Human Factors*, vol. 54, no. 2, pp. 235–249, 2012, pMID: 22624290. [Online]. Available: https://doi.org/10.1177/0018720811428734

[11] L. G. J. van der Knaap, “Softglove: Per finger force feedback,” Bachelor thesis, Delft University of Technology, 2019.

[12] S. B. D.A.M. Koene, “Softglove: Per finger vibrotactile feedback,” Bachelor thesis, Delft University of Technology, 2019.

[13] EuroCircuits, 2019. [Online]. Available: https://www.eurocircuits.com/

[14] J. Fjelstad, *Flexible Circuit Technology, Third Edition.* Br Publishing, Incorporated, 2007.

[15] C. Lam, “Senseglove communication protocol,” 2018.

[16] P. V. den Bossche, F. Vergels, J. V. Mierlo, J. Matheys, and W. V. Autenboer, "Subat: An assessment of sustainable battery technology," *Journal of Power Sources*, vol. 162, no. 2, pp. 913 – 919, 2006, special issue including selected papers from the International Power Sources Symposium 2005 together with regular papers. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378775305008761

[17] G. E. Blomgren, "Current status of lithium ion and lithium polymer secondary batteries," in *Fifteenth Annual Battery Conference on Applications and Advances (Cat. No. 00TH8490)*. IEEE, 2000, pp. 97–100.

[18] T. R. K. Peelen, "Softglove: Palm vibrotactile feedback," Bachelor thesis, Delft University of Technology, 2019.

[19] T. Instruments, 2019. [Online]. Available: https://www.ti.com/packaging/docs/searchtipackages.tsp?packageName=BGA

[20] Distrelec, 2019. [Online]. Available: https://www.distrelec.nl/nl/converter-ic-qfn-24-ic-haus-ic-tw4-qfn24/p/17350861

[21] Lofelt, "Elevating haptic technology with lofelt wave," January 2019.

[22] C. Cuijpers, "Enhancing the music festival experience for hearing-impaired people," TU Delft, November 2018, https://repository.tudelft.nl/islandora/object/uuid:fc203340-4f55-48cc-825a-b874cfe201a7.

[23] K. J. Burdick, S. K. Jorgensen, M. O. Holmberg, S. P. Kultgen, T. N. Combs, and J. J. Schlesinger, "Benefits of sonification and haptic displays with physiologic variables to improve patient safety," *Proceedings of Meetings on Acoustics*, vol. 35, no. 1, p. 020001, 2018. [Online]. Available: https://asa.scitation.org/doi/abs/10.1121/2.0000941

[24] F. Gonzalez, F. Gosselin, and W. Bachta, "A framework for the classification of dexterous haptic interfaces based on the identification of the most frequently used hand contact areas," in *2013 World Haptics Conference (WHC)*, April 2013, pp. 461–466.

[25] J. Long, "Linear vibration motors (lra's)," 2019. [Online]. Available: http://www.vibration-motor.com/products/lra-linear-resonant-actuator-vibration-motors.html

[26] L. F. R. S, "Drv2605: Which driver to use for linear wideband actuator," 2019. [Online]. Available: https://e2e.ti.com/support/motor-drivers/f/38/t/760733?DRV2605-Which-driver-to-use-for-Linear-Wideband-Actuator4

[27] TI, "Tlv320aic3120 low-power audio codec with minidsp and 2.5-w mono class-d speaker amplifier," 2019. [Online]. Available: http://www.ti.com/product/TLV320AIC3120#

[28] E. Systems, "Esp32 series," 2019.

[29] ——, "Tca9548a low-voltage 8-channel i2c switch with reset," 2016.

[30] D. M. J.G. Proakis, *Digital Signal Processing*, 4th ed. Pearson, 2014.

[31] M. Corten, "Senseglove haptics research 2018 (confidential)," 2018.