

ARDUINO-BASED ELECTROTACTILE STIMULATION SYSTEM FOR VIRTUAL SENSORY FEEDBACK

Author: Javan Willock

Registration number: 1503772

Supervisor: Professor Francisco Sepulveda

Second Assessor: Dr John O'Hara

Degree: Electronic Engineering

Table of Contents

Chapter 1: Introduction	1
1.1. Motivation.....	1
1.2. Objectives.....	2
1.3. Report Summary	3
Chapter 2: Background Reading and Literature Review	5
2.1. Electrotactile Stimulation	5
2.2. Somatosensory System	7
2.3. Epidermal Impedance.....	9
Chapter 3: Software Design	13
3.1. Abstract	13
3.2. Requirements Form	13
3.3. Sequence Diagram.....	14
3.4. Flowchart	14
3.5. GUI	18
Chapter 4: Hardware Design and Integration.....	21
4.1. Abstract.....	21
4.2. Technical Background.....	21
4.3. Microcontroller	21
4.4. 5DT Data Glove.....	23
4.5. Electrode	26
4.6. Schematic Diagram	27
Chapter 5: Results, Evaluation and Discussion	30
5.1. Software Results.....	30
5.2. Hardware Results	33
Chapter 6: Project Management and Reflection	36
6.1 JIRA and Gitlab screenshots.....	36
6.2 Project Tracking.....	41
6.3 Self-Critical Reflections.....	42
Conclusion	44
References	45

Table of Figures

Figure 1: Stimulation Waveforms.....	5
Figure 2: Dorsal root ganglion location in spinal nerve	7
Figure 3: Cortical sensory homunculus.....	9
Figure 4: 3D Representation of anatomy of the skin	10
Figure 5: Intermittent stimulation waveform [13]	12
Figure 6: Sequence Diagram defining program behaviour	14
Figure 7: Glove setup flowchart	14
Figure 8: Setting up 5DT Glove in Vizard 5	15
Figure 9: Flowchart of top-level hierarchy.....	15
Figure 10: Virtual object detection function.....	17
Figure 11: Hand gesture array.....	17
Figure 12: Graphical User Interface that the wearer interacts with	18
Figure 13: PWM code snippet.....	20
Figure 14: Arduino Due Microcontroller	22
Figure 15: Output reading of finger flexure in Glove Manager.....	24
Figure 16: MotionBuilder capture of 5DT Data Glove binding	25
Figure 17: 5DT Data Glove Diagram	25
Figure 18: Axelgaard PALs Electrodes.....	26
Figure 19: Voltage waveform converter.....	27
Figure 20: Top level with sub circuit.....	27
Figure 21: MAX3232 circuit.....	28
Figure 22: System Architecture	29
Figure 23: Vizard Environments	30
Figure 24: Object Masking	31
Figure 25: Howland Current Pump with MAX3232	33
Figure 26: Input output waveforms	33
Figure 27: Output signal pulse width timings	34
Figure 28: Output voltage and current readings	34
Figure 29: Screenshot of JIRA Kanban Board	36
Figure 30: Screenshot of Gitlab Repository	37
Figure 31: Example User Story Description	38
Figure 32: JIRA Cumulative Flow Diagram	39
Figure 33: Issue Type Pie Chart Report and Table.....	39
Figure 34: Project Topography	40

Acknowledgments

I would like to thank Professor Francisco Sepulveda for his supervision and guidance throughout my final year project. His support was much needed and helped to provide a direction to my learning experience.

I would also like to extend this thanks to the University of Essex for being able to accommodate my needs and providing a platform for me to grow and my friends and family for being there every step of the way.

Chapter 1: Introduction

Electrotactile¹ feedback stimulation development is the subject of many research projects, with ideas for use in medicine, virtual reality and the industry. Due to the limitations presented by sensory impairment, the main idea is to communicate non-tactile information via electrical stimulation of the sense of touch to the user. Successful implementation of electrotactile feedback stimulation (EFS) has the potential result in a better standard of living and/or increased experiences and safety in a variety of environments. Individuals without the ability to process information in a conventional way, i.e. through sight, may find that they can still experience the world via EFS. This research project will explore a wearable electrotactile feedback system prototype, with focus on sensitivity and object handling in a virtual environment.

1.1. Motivation

Sensory impairment affects people from every demographic. There are many people around the world that are resilient to the limitations caused by this and persevere regardless to achieve great success through their own efforts or aided with the support and guidance from others. Though it can be said that the implementation of certain practices, see [1] where an electrotactile glove was developed for the for the blind to help identify graphics, with initial results demonstrating a 63.5% successful detection rate with random geometric shapes. Or [2] where an electrotactile system for the tongue was created to help individuals with low or lost vision gain more perception of their surrounding environment, demonstrating a strong outcome in achieving this objective. Can and do have the potential to positively impact on lives and have the potential in improve on quality of living.

The ability to provide electrotactile feedback stimulation to individuals could provide a means to open a relatively new and yet fully unexplored area of research to help individuals live around their lack of somatosensory² response.

This project has been tailored to my background as an electronic engineering student. In addition, my version of this system will use a wearable data gathering glove to track the position of the fingers. As the user performs hand grabbing gestures, the motion will be mimicked on-screen with his/her fingers, the device will detect which finger is being moved and where it is in the virtual space, and a multichannel electro-tactile stimulator will translate that movement and gesture result into a signal to be transmitted to the fingertips of the user, thus passing the action from the screen to the user's skin.

¹ Electrotactile means to provide stimulation to the skin using an array of small electrodes.

² Somatosensory refers to a sensation felt on the body, be that pressure, pain or warmth.

The electrotactile stimulator will be built around an Arduino microcontroller that will generate the necessary electrical pulses that will drive the skin electrode array. Therefore, in the completion of this project I will learn about signal characteristics for reliable, comfortable, and crucially safe electro-tactile stimulation. More information on this can be found in 'Chapter 3: Implementation'. The use of a virtual world³ can allow for interaction with countless of objects in different scenarios and so creates a great platform for testing and gathering data in a safe and reliable environment.

1.2. Objectives

The primary aim of this project is to design and implement a wearable electrotactile feedback system that will allow the wearer to develop a limited awareness in their fingertips to increase sensitivity and obtain object contact recognition. To achieve this a set of objectives were put in place, the principle ones being:

1. Ideation – Forming a plan/process, and figuring out what I need, what I need to do and why. Use of JIRA and Gitlab is fundamental in managing this objective;
2. Design – Drawing up flowcharts and diagrams to help with laying out the development groundwork of hardware and software creation;
3. Implementation – Actual building of hardware and software to allow for signal conversion for feedback stimulation and interaction with virtual world respectively. As well as compatibility testing for software and hardware;
4. Final Evaluation – This includes testing and proving that the prototype performs as intended with an impact on the user such that it satisfies the reason(s) for initial development;

The secondary objectives being:

5. Identification of stimulation parameters;
6. Testing framework for user optimisation;
7. Determination of electrode efficiency.

To recreate the sense of contact made with touching an object in the 3D environment, the user will experience localised stimulation on the tips of their fingers if their hand is 'grabbing' the object. The intensity of stimulation felt by each finger does not vary or indicate the level of force applied to the virtual object and will remain consistent. Distance to the object in relation to the hand will also not be a factor in the ability to interact with the object. It has been shown that each person has a unique tolerance to electrotactile stimulation and so, the

³ A virtual reality development platform capable of supporting peripherals real-time.

threshold will need to be determined each time by measuring the intensities of stimulation via psychophysical techniques⁴ [3].

Successful implementation of this project will allow the ability to perceive the environment and the object in real-time effectively and enable the user to experience their environment without use of the eyes or other aids. Given time this could allow for object localisation, obstacle avoidance and object identification without using the eyes.

Virtual environments are currently highly incompatible to people with visual impairment as most of the experiences given are visual. Primary use of such an application could also be used instead in locations where remote sensitivity to objects is needed. Or gaming, where electrotactile stimulation provides an increased user immersion experience.

By using the virtual environment as a test case scenario, we can derive results that can potentially benefit many people and be used in innovative ways. There is potential to provide a useful and safe environment for learning.

1.3. Report Summary

This report consists of six chapters, followed by a conclusion and references section. Detailing in order of occurrence each chapter will be listed below not including the introduction.

Chapter 2: This section covers the fundamental reading for the understanding and design of an electrotactile feedback system. You will find information regarding natural skin impedances as well as the basic principles of electrotactile stimulation here.

Chapter 3: Talks in detail on the implementation of the project covering the software and hardware aspects, in this section you can explore the design flow represented with the visual modelling language UML⁵ as well as the key features of the software and its programming, the language used and thresholds set in place. For the hardware you can find a thorough covering of the microcontroller and its features. As well as the circuitry developed detailing the reasons for its use, elaboration on key components and their main functions in the prototype and lastly how they all fit together to make a complete system covering steps needed to achieve this.

Chapter 4: This is the results and evaluation chapter; here you can read on my findings and thoughts on the results gained. There will be a small discussion on those results, though I will

⁴ Psychophysical technique refers to the method of measuring an individual's perception and performance, so that characteristics of the sensation can be specified and altered if necessary.

⁵ UML is Unified Modelling Language and is usually used as part of a software development process.

note here that due to sudden change in circumstances⁶ this section is not as fulfilling as it should be, but you can find figures and plots of simulations for both hardware and software here.

Chapter 5: This chapter will cover how I managed my project. Here I will self-criticize my work, include references to Gitlab and Jira and elaborate on some reflective questioning, talking on learning experiences and providing an example on how the agile methodology has helped me throughout this project development.

⁶ Due to the sudden outbreak of COVID-19, the time of which this paper was written. The results and evaluation of this product are very limited and are restricted to basic simulations and overview of the system currently in place.

Chapter 2: Background Reading and Literature Review

2.1. Electrotactile Stimulation

As defined by Schroeder in [4], the fundamental element of electrical stimulation is the application of an electrical potential difference to the surface of the skin via electrodes. The electrical balance across the nerve of the membrane between the location of stimulation is disturbed, which can then trigger an action potential on the nerves that are in the area between the electrodes.

Looking into different papers, there is a debate on which waveform is the most beneficial for electrotactile feedback. In a test performed in [2] three waveforms were evaluated on a concentric electrode. As shown in Figure 1: Stimulation Waveforms each waveform has common parameters, these being frequency, pulse width and amplitude and the electrical potentials that are applied to the skin are defined by these constraints.

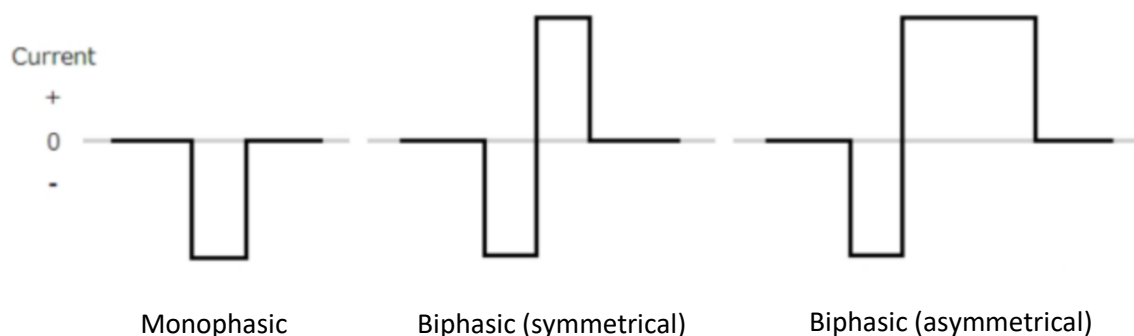


Figure 1: Stimulation Waveforms⁷

Nerve fibres respond uniquely dependent on the constraints of the applied pulses. Pulses with a shorter pulse width and a higher current are necessary to create a response and pulse; whereas a long pulse width requires a lower amplitude [4].

- Monophasic wave, with a pulse of $150\mu s$ at $0.5 - 0.25$ mA.
- Wave with alternating phase pulse of $150\mu s$ at $0.5 - 0.15$ mA.
- Biphasic, pulses at 10kHz at $0.5 - 0.15$ mA.

⁷A monophasic waveform current amplitude is in one polarity only either in the positive or negative direction. A biphasic waveform current amplitude has dual polarity and occurs in both the positive and negative direction. This should allow for a greater sensory feedback potential for the user.

After achieving a comfortable tactile threshold for each participant, by altering the current amplitude an average current level was chosen, with the charges per pulse being within $\pm 4\text{dB}$ despite the waveforms being very different. Intermittent inspections of the stimulation zone showed small reactions in the epidermis. In [5] Saunders findings showed biphasic stimulation as the most comfortable, because of its shorter pulse width required, a greater current amplitude to produce an equivalently strong sensation, and therefore produced a greater amount of transient skin reddening and other reversible changes. Monophasic stimulation with longer pulses required the least amount of current for satisfactory sensation and produced the least amount of transient skin reddening. The alternating phase stimulation came in second.

This is further backed by [6] where it is also mentioned that monophasic pulses can be the cause of tissue damage, skin irritation or rashes when used for prolonged periods, due to the ionic build up and polarising effect which is capable of breaking down tissue if not addressed. With biphasic pulses being able to reduce this side effect by allowing the stimulated tissue time to depolarise.

So overall no major safety concerns can be considered in choosing the type of waveform, just the application of said waveform. This experiment can be reinforced by the stimulation test carried out in [7] where it was confirmed that a certain percentage of participants experienced a change in 'comfort' levels as the pulse repetition was changed.

When a short pulse is sent through, the sensory nerve response is defined by the sensory threshold. When supplying stimulus waveforms within the sensory nerve threshold with an adequate pulse width, you can see muscle activation. As a result of this, stronger pulses can result in pain if the frequency of the stimulation is too high. If this happens nerves will fail to respond. This is because the refractory period can only be started after some time. It is mentioned in [8], that the maximum frequency for nerves to react is in the range of 4-5kHz.

In [9] it is mentioned that from the application of an electrical stimulation and electric current pulse from the electrode surface generates an electric field inside the skin. It is this field that induces nerve activity. It is also theorised in [9] that the changing of the current polarity and electrode size can lead to localised stimulation of the skin receptors.

2.2. Somatosensory System

The somatosensory system is multi-layered and is a part of the sensory system that deals with the conscious perception of touch, pressure, pain, temperature, position, movement, and vibration, which arise from the muscles, joints, skin, and fascia. [10] The somatosensory system is a 3-neuron system that relays sensations detected in the periphery and conveys them via pathways through the spinal cord, brainstem, and thalamic relay nuclei to the sensory cortex in the parietal lobe [11] ⁸.

The three sensory neurons are as follows:⁹

- Primary
- Secondary
- Tertiary

As mentioned in [12] the primary neuron's main role is to detect sensory stimuli as the sensory receptor. This neuron responds well to touch and temperature. The cell body of the neuron is found in the dorsal root ganglion of the spinal nerve.

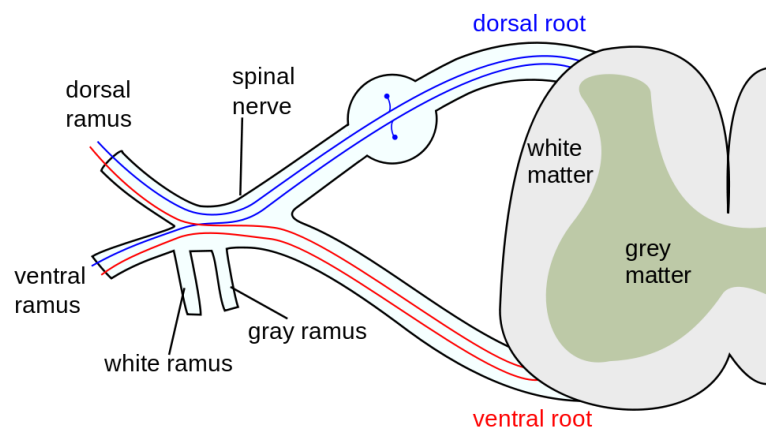


Figure 2: Dorsal root ganglion location in spinal nerve¹⁰

⁸ Referenced from: Gleveckas-Martens, N. and Crisan, E., 2013. Somatosensory System Anatomy. [online] Medscape. Available at: <<https://emedicine.medscape.com/article/1948621-overview>> [Accessed 6 April 2020].

⁹ Courses.lumenlearning.com. n.d. General Organization of The Somatosensory System. [online] Available at: <<https://courses.lumenlearning.com/boundless-ap/chapter/the-somatosensory-system/>> [Accessed 6 April 2020].

¹⁰ Image found on Wikipedia article: En.wikipedia.org. n.d. Dorsal Root Ganglion. [online] Available at: <https://en.wikipedia.org/wiki/Dorsal_root_ganglion#/media/File:Spinal_nerve.svg> [Accessed 6 April 2020].

The secondary neurons function is more of a relay located in the brainstem or the spinal cord and the tertiary neurons have cell bodies in the thalamus, forming a sensory homunculus when tactile stimulus is induced. This neuron is in the cerebellum [12].

Mechanoreceptors are sensory receptors in the skin that respond to mechanical pressure or distortion in [13] details records of several types of sensory endings:

- Meissner corpuscles¹¹ – Detect texture change
- Pacinian corpuscles – Detect extreme vibrational change
- Merkel cell-neurite complexes – Detect sustained touch and pressure
- Ruffini endings – Detect skin tension
- Mechanoreceiving – Free nerves that detect touch, pressure and stretching
- Hair follicles – Detect hair strand changes

It is key to know that there are several mechanoreceptors in the human skin, that determine perception of contact with an object. Also, that each one of them have different sensitivities to tactile pressure and receptive field sizes. Regardless of the amount of force needed to stimulate the area electrical stimulation of the mechanoreceptors creates the same feeling as a physical stimulation [4] [13].

These receptors all possess different nerve characteristics and stimulation requirements, for example if we look at the Meissner Corpuscles, it is said that for adequate tactile sensation, only an isolated stimulus in a localised area is needed as they are sensitive to light strokes on the skin. This sensation is transmitted by the rapidly adapting fibre. Whereas the Pacinian corpuscle needs an abrupt automatic sensation to activate the receptors, these sensations being transmitted by the Pacinian corpuscle-associated fibre [14]. To reduce or avoid activation of nerve trunks and muscles, stimulation can be delivered using concentric electrodes. This arrangement will generate surface currents, thereby avoiding unnecessary activation of the deep sensory motor structures [15].

Mechanoreceptors are mainly responsible for transforming tactile stimulus into electrical impulses, which travel through the neural network to the brain. See Figure 3: Cortical sensory homunculus for a graphical representation of the primary somatosensory cortex and how the anatomical portions of the body are mapped out.

Several studies have measured the skin's ability to transfer electrical stimulation to mechanoreceptors and their fibres by delivering either pulse voltage across all, more commonly, regulated current through the skin surface [2]. Pulse width is an important characteristic in determining the amount of charge that is delivered to the skin. It also affects

¹¹ A minute body or cell in an organism.

sensation of the electrical stimulation. The frequency of stimulation may also play a role in discriminating the location of a signal [16].

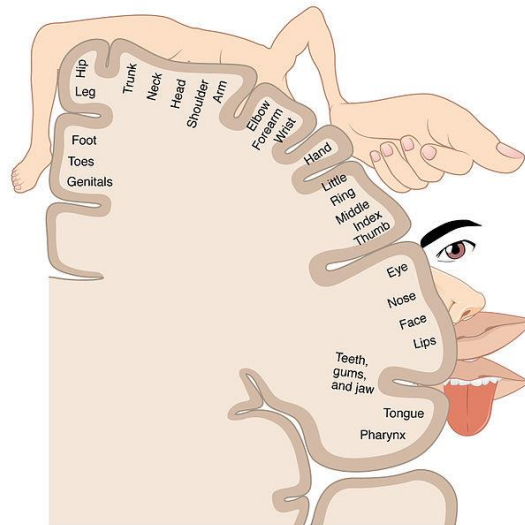


Figure 3: Cortical sensory homunculus¹²

The primary somatosensory area of the human cortex is in the postcentral gyrus of the parietal lobe. The postcentral gyrus is the location of the primary somatosensory area, the area of the cortex dedicated to the processing of touch information. At this location there is a map of sensory space referred to as a sensory homunculus [12].

2.3. Epidermal¹³ Impedance

Skin is a multi-layered system that provides a natural resistive property that could make it difficult to interpret test results. To get around this we need to look more into electrical conductance in biologic tissue. Electrical conductance here depends on the ion-permeability¹⁴ of the tissue in question [17]. Stratum corneum¹⁵ cells are a densely packed lattice of layered corneocytes¹⁶ composed of relatively high resistivity keratin compared to other tissues in the body, such as muscle, bone and underlying dermal tissue. The stratum corneum makes up the protective barrier of skin exposed to the environment [18] [19].

¹² Image found on Wikipedia article: Textimgs.s3.amazonaws.com. n.d. [online] Available at: <<https://textimgs.s3.amazonaws.com/boundless-anatomy-and-physiology/xa97n5hro6vd8gsmcvvi.jpg#fixme>> [Accessed 6 April 2020].

¹³ The outer nonvascular, non-sensitive layer of skin covering your skin.

¹⁴ How easily can the ion cross the membrane to provide electrical stimulation.

¹⁵ The outermost later of the skin.

¹⁶ Corneocytes form most of the stratum corneum, they are an essential part of the skin barrier property.

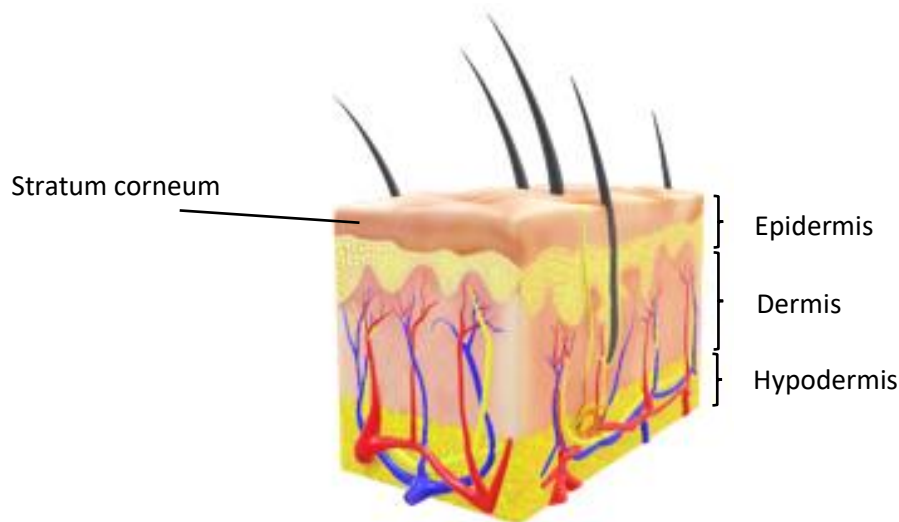


Figure 4: 3D Representation of anatomy of the skin¹⁷

Dry stratum corneum is an almost impermeable tissue, but wet corneum, by virtue of aqueous channels between the keratin plates and possibly between the keratin fibrils, is rather freely conductive [17]. This is also covered in [20] where the flow of aqueous electrolyte¹⁸ depends on the applied voltage magnitudes. The electrical resistance and capacitance of the skin are extrinsic properties; hence, skin impedance is dependent on effective electrode area against the stimulated skin surface which in turn depends on applied force between skin to electrode interfacing. In contrast, the resistivity of sub dermal skin tissue and other soft tissues is shown to be mostly unchanged when altering the frequency and amplitude (as seen in [16]) and is much lower and constant spatially in value compared to the stratum corneum. Whereas for the stratum corneum square wave electronic pulses evidence indicates that the electronic resistivity of the stratum corneum layer decreases almost inversely with increasing current amplitude [18].

The germinative layer by virtue of the large interstitial canaliculi¹⁹ in which even protein molecules have been observed to migrate is rather freely conductive, but the granular layer possibly constitutes a barrier as does the dermo epidermal junction, where the cells are tightly joined. The membranes of these cells may be relatively ion impermeable as are those of many cells because of the lipid or protein content [17].

Due to the uniqueness of the user's biology, the interpretation of skin conductance depends on the conditions on which it will be measured. This can be seen in [21] to achieve additional electrical conductivity between the electrodes on a skin. A small amount of conductive gel

¹⁷ 3D model taken from Microsoft Word's 3D Models database.

¹⁸ A substance that produces an electrically conducting solution when dissolved in water.

¹⁹ Tiny canals that connect all the unfilled space together.

was applied to the fingers of the users prior to the fitting of the gloves. This would enable the testers to get around some of the individual biological characteristics of the users and bypass any natural skin impedances.

As briefly mentioned above in the 'Electrotactile Stimulation' sub-chapter, it can be said that the shape and surface area have an impact on the surface effect of the sensation received on the skin. Circular electrodes have a better user experience than rectangle electrodes, this is because the current can peak at the corners of the rectangular electrodes. An electrode with a surface area of 1mm^2 or above 100mm^2 can cause a prickly or painful sensation [4].

Due to the high concentrations of mechanoreceptors found on human fingertips, exploration of the fingertips ability to discriminate electrotactile stimulation at four points has been explored. It is noted that during testing, stimulation was delivered to the volar aspect, with varying electrode sizes, interelectrode spacings and stimulation frequency as the main factors. Discrimination of stimulated locations under this parameter was significantly above chance levels in all cases. It is incredibly prevalent to note that electrode size or stimulation frequency in this test it was not a significant influencer when considered separately [16].

These findings are reinforced in [7], when the percentage of participants hit at least 70%, who of which could feel the stimulation when the pulse width or the pulse repetition frequency was varied. When verifying the percentage of participants who could differentiate tactile sensation, the adjustment of the pulse width or pulse repetition frequency increased the number of participants who could notice the difference to at least 60% no matter which factor was adjusted. It was therefore concluded that the higher the pulse repetition frequency the greater the number of participants who could notice the difference as the pulse width varied. The wider the pulse width was, the greater the number of participants who could do so as the pulse repetition frequency varied.

However, results showed in [16] that increased electrode spacing significantly increased subjects' ability to discriminate location of stimulated electrodes. Knowing this, it may be useful to investigate multi-channel electrotactile stimulation for sensory feedback of fingers. The work explored in [22] (see Figure 5: Intermittent stimulation waveform [13]), made efforts to find possibilities for two channel electrotactile stimulation through intermittent stimulation for better performance. This promoted the possibility for independent feedback of the forces of the fingers in question. By applying offset pulses, the stimuli of each channel would be recognised somewhat independently, and not get confused with each other. This would reduce the masking effect of simultaneous stimulation and improve recognition of the stimuli by users.

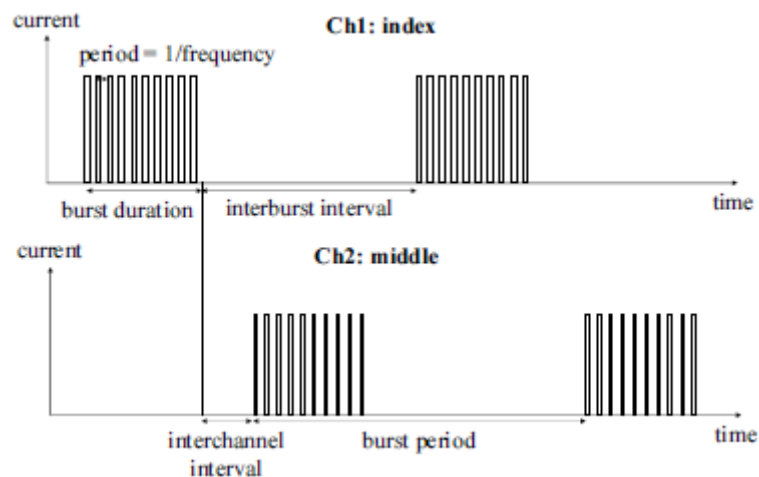


Figure 5: Intermittent stimulation waveform [13]

The spatial distribution of electrodes should be considered, as any force applied to a virtual object can be replicated if there is a selection of appropriate points of stimulation. For example, stimulation from a centre pin electrode can simulate the contact between the centre of the fingertip and an object edge. Whereas stimulation from one of the many pin outer electrodes can simulate the contact between an area of the fingertip's surfaces of the base of a cylindrical object [21].

Chapter 3: Software Design

3.1. Abstract

The electrotactile stimulator will be built around an Arduino microcontroller that will generate the required electrical pulses that will drive the electrodes in contact with the skin. As a result of this, a stable and safe stimulation is needed where the characteristics of the signal are clearly defined. The development of this software will be done on Vizard 5 and the Arduino IDE 1.8.10 written in python32 and C/C++ respectively. The Arduino microcontroller will generate the signal based on real-time input from the user in the virtual world. These signals will be sent serially to the circuit where a signal conversion takes place and current regulation takes place.

3.2. Requirements Form²⁰

Name	Electrotactile glove stimulator prototype
Purpose	To provide electrotactile stimulation to the fingertips once virtual contact of an object has been made, representative of the feeling of touch.
Inputs	USB power supply
Outputs	Waveform signal, electrodes
Functions	Finger position determination, virtual tactile sense provider
Performance	Provide real-time feedback of contact with minimal lag
Physical size/weight	No larger than a hand, no heavier than a standard glove
Power	Minimum supply of 290mW

Table 1: Requirements Form table of system architecture

²⁰ General representation of the system and its function and non-functional properties.

3.3. Sequence Diagram

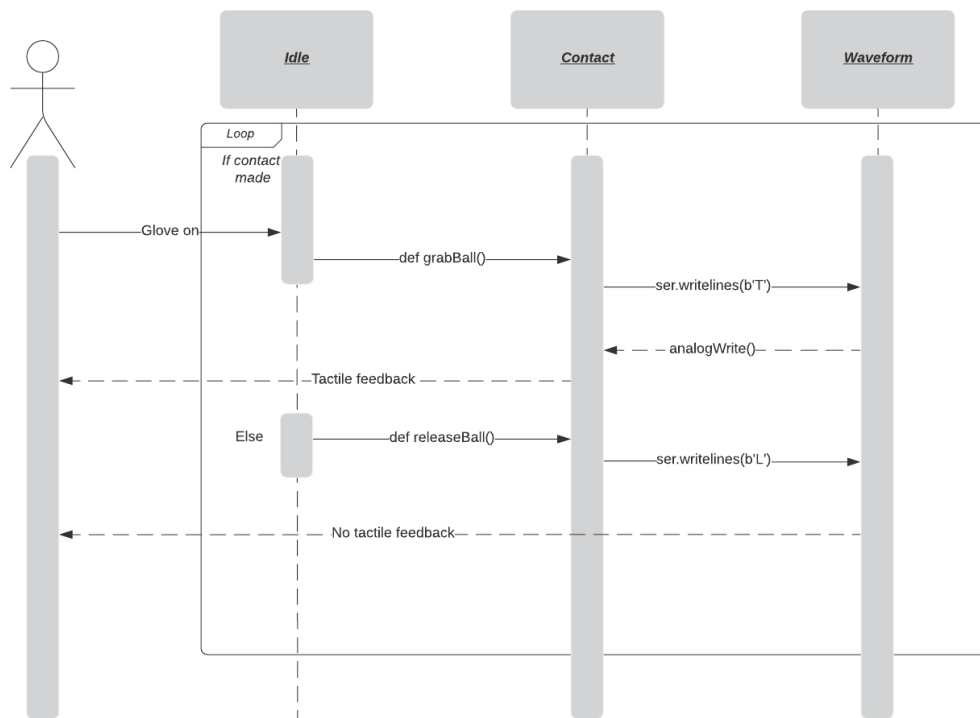


Figure 6: Sequence Diagram defining program behaviour

The sequence diagram in Figure 6: Sequence Diagram defining program behaviour shows object interactions in a time sequence. All the objects in the diagram are enclosed in a loop case to show a group scenario dependent on one another. With priority starting from the top left cascading downwards following the directional arrows.

3.4. Flowchart

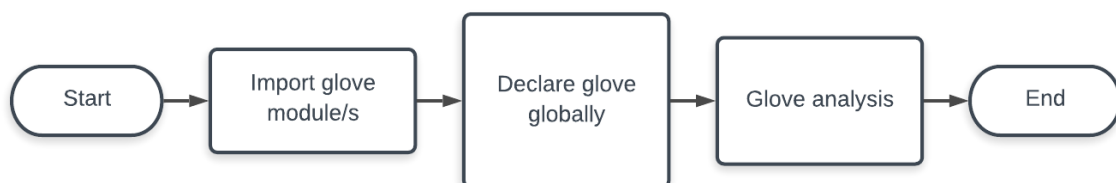


Figure 7: Glove setup flowchart

At the start of the program initiation, the code will import the 5DT Data Glove module '**import hand**'. In Figure 8: Setting up 5DT Glove in Vizard 5, we can see that the code first identifies what port the glove is on and then uses one of the built in Vizard modules '**addSensor**'²¹ to include the 5DT Data Glove plugin and request a sensor object from it. This sensor object will contain the output positions of the fingers as well as orientation data that is attached it to the sensor variable. The next line takes the keyword argument '**hand.GLOVE_5DT**' and the positional argument '**sensor**' to be passed to the function and assigns it to the '**glove**' variable. This allows for the necessary function to be called automatically based on the specific file type given and for a hand object to be created.

```
PORT_5DT_USB = 0
sensor = viz.addSensor('5dt.dls')
glove = hand.add(sensor, hand.GLOVE_5DT)
```

Figure 8: Setting up 5DT Glove in Vizard 5

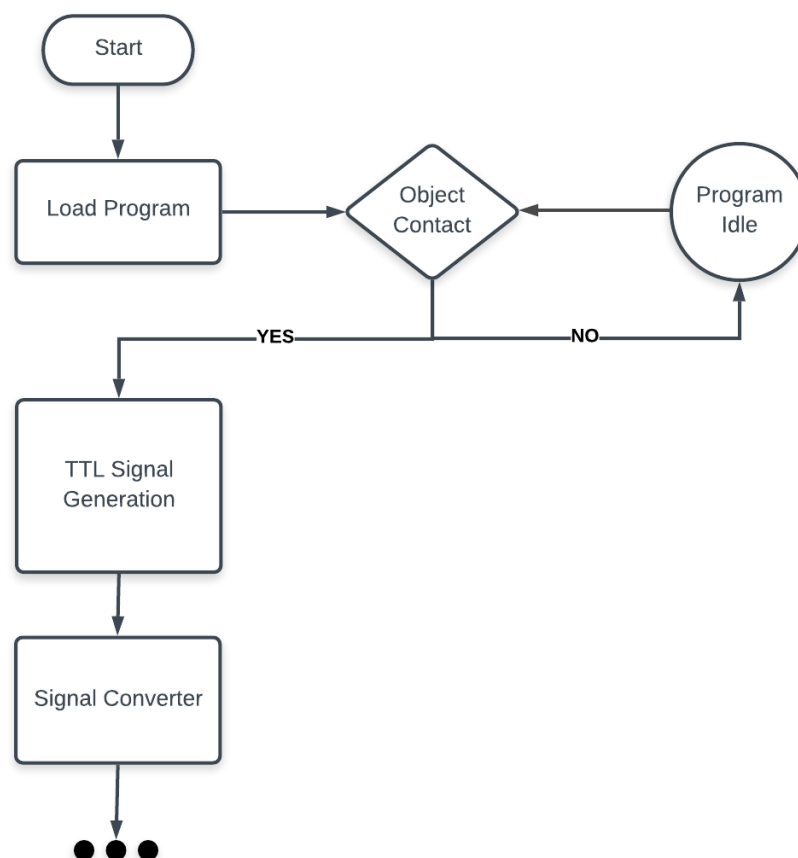


Figure 9: Flowchart of top-level hierarchy

²¹ Command to add sensor object to the world.

The top-level hierarchy encompasses the main features of the software relating to the contact of an object in the virtual world. This is a condition-based action that will activate whenever the boundaries of the function are met. There is one main function 'def grabBall'²² (see Figure 10: Virtual object detection function) that will determine if the hand has contacted the virtual object. It is seen that the 'gesture' variable references the sensor value mentioned above in Figure 8: Setting up 5DT Glove in Vizard 5. This variable contains the retrieved integer value of the positional values of the fingers. Each integer value from zero to sixteen represents a specific hand gesture. For example, the integer value zero would represent a closed fist and a two would represent an index finger point, to see the whole array refer to Figure 11: Hand gesture array.

The 'object' variable is assigned the command that returns the current geometry object that the cursor is pointing at. In this case we want it to be the object that the wearer's attention is focussed on. If the user clicks on an object, a return True value will produce a VizIntersect object²³, the VizIntersect command stores the start and end vector position points of the intersection line segment created in the virtual world and will check if that specified line segment intersects with any objects placed in the environment. If there is an intersection the 'object' variable will hold the name of the object intersected.

The if statement initiates when the user has made a fist gesture and the 'object' variable has matched the pre-defined object 'basketball'. So long as both conditions are logically true, the command 'viz.grab'²⁴ will create a link that keeps the orientation and position of the target object 'basketball' fixed to the relative source object which in this case is the cursor. Movement of the cursor will move the basketball, relative to the preserved orientation and position. 'ser.writelines(b'T')' sends a message via the serial port which gets picked up by the microcontroller, this is what allows for the initial square waveform to be generated.

If the wearer has made anything but a closed fist gesture, then the program will release the basketball, and send a different message serially to the microcontroller, stopping any waveform generation.

²² Function containing object contact detecting commands.

²³ A command that returns information about intersection results if the cursor hits any objects in the virtual environment.

²⁴ Command used to create a link that acts as if the source object has grabbed the target object.

```
def grabBall():
    gesture = int(sensor.get()[-1])
    gestureText.message(gestureName[gesture])
    object = viz.pick()

    if (gesture == 0) and (object == basketball):
        print
        "closed fist - gripped basketball"
        ser.writelines(b'T')
        global link
        link = viz.grab(hand, basketball)
        glove.getSensorRawAll()

    else:
        print
        "This is not a basketball"
        ser.writelines(b'L')
        vizact.ontimer(1, releaseBall)
```

Figure 10: Virtual object detection function

```
gestureName = ['Fist', 'Index finger point', 'Middle finger point',
               'Two finger point', 'Ring finger point', 'Ring-Index finger point',
               'Ring-middle finger point', 'Three finger point', 'Little finger point',
               'Index and little finger point', 'Little-middle finger point',
               'Not ring finger point', 'Little-ring finger point',
               'Not middle finger point', 'Not index finger point',
               'Flat hand', 'Undefined']
```

Figure 11: Hand gesture array

3.5. GUI

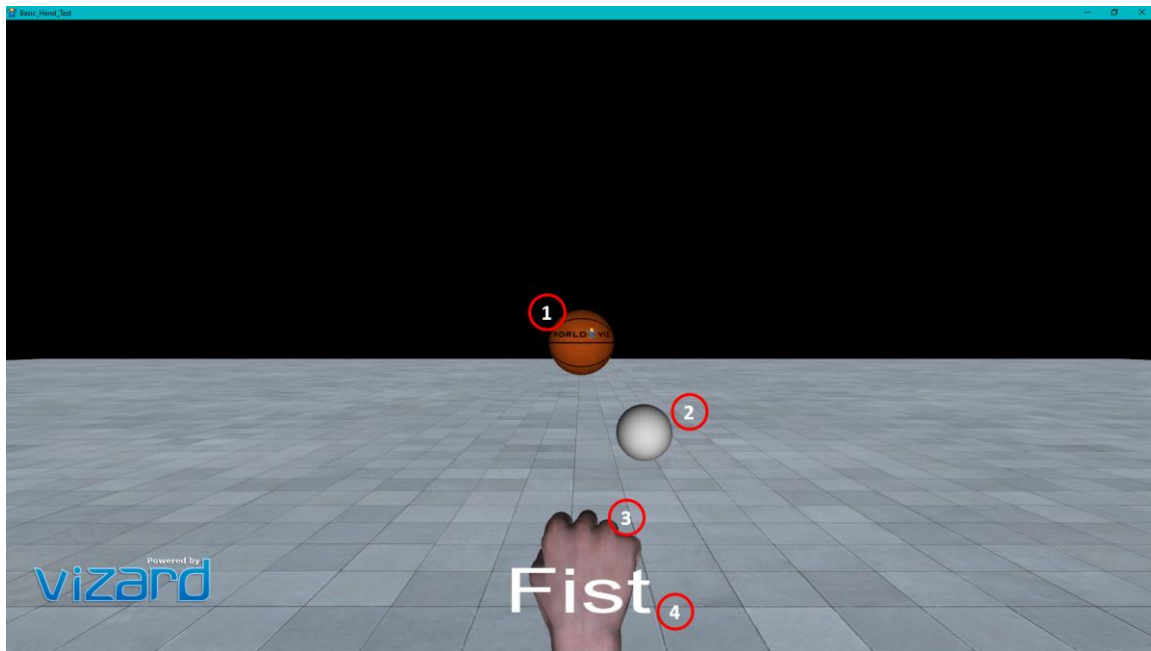


Figure 12: Graphical User Interface that the wearer interacts with

Figure 12: Graphical User Interface that the wearer interacts with reference points listed below:

1. The basketball object user is required to interact with: I wanted to be able to make it as easy as possible to 'grab' the basketball. So, I placed the object directly centre on the screen with a `<node3d>.setPosition([x, y, z])` command. The x, y, z being the vector co-ordinates used to translate the object in three-dimensional space. And `<node3d>` referring the node object to be targeted.
2. Moveable cursor that will act as a pointer to the object: Because of the lack of ability to move the virtual hand through the virtual space and towards the basketball I needed to be able to include a focus aspect to the grab. So, I made the decision to keep the cursor visual, and to make it stand out more, I attached it to a white ball object using the `viz.link(source object, destination object)`. This command links the destination object (white ball) to the source object (cursor). Once the two objects are linked, any change of the position and/or rotation of the source object will automatically be applied to the destination object.
3. Virtual hand that will mimic the wearer's own. Positioned so it looks as though the hand is coming out of the screen with the same command used for the basketball position. Thanks to the plugin provided the finger tracking data can be captured and mimicked with the hand model.

4. Text allowing wearer feedback on hand gestures being made. This is not a necessary feature but was added for the benefit of the wearer testing the device. Looking at Figure 10: Virtual object detection function `grabBall()` function, we can see the first two lines of code snippet where the code identifies the gesture being performed and then refers to an array (see Figure 11: Hand gesture array) and then prints a string to screen depending on what array element was retrieved.

Due to the nature of the project being focussed on the virtual environment I wanted to make a simple three-dimensional area that would allow the user perception of only the necessities. Thanks to the versatility of Vizard I was very quickly able to make this and throw in the above list of objects to increase user immersivity.

3.6. Arduino Code

```
void loop() {  
  
    analogWrite(thumb, 127);  
    analogWrite(index, 127);  
    analogWrite(middle, 127);  
    analogWrite(ring, 127);  
    analogWrite(pinky, 127);  
  
    delay(30);  
}  
  
    analogWrite(thumb, 0);  
    analogWrite(index, 0);  
    analogWrite(middle, 0);  
    analogWrite(ring, 0);  
    analogWrite(pinky, 0);  
  
    delay(30);  
}
```

Figure 13: PWM code snippet

In Figure 13: PWM code snippet there are two loops that are executed one after the other. The first to increase the output value and the second to decrease it, both done on the output pins 13, 12, 11, 7 and 6 to represent all five fingers on the left hand. The `analogWrite()` function writes an analog value to the specified pin. This pin will generate a stable rectangular wave of 50% duty cycle.

Chapter 4: Hardware Design and Integration

4.1. Abstract

Electrotactile feedback needs to be able to provide information back to the receiver, whether that be the amount of pressure being applied to an interaction, the sensitivity response of contact or determining the shape of an object through this stimulation. Whatever the function is the need for it to be channelled in some way to the user and more specifically to the target bodily area is important.

There have been many studies over the years that have utilised different methods to provide electrotactile feedback to the user. All of them wearable in some way and capable of delivering a safe and localised stimulus. This chapter will highlight the proposed design, elaborating on the key components used from the data glove, the microcontroller, electrodes to the circuit design.

4.2. Technical Background

In all electrotactile feedback systems there are main similarities. To create a successful electrotactile feedback system there needs to be a basal number of components implemented to produce a successful product. These components being the means to sense when contact has been made so the output can be forwarded to the pulse generator, in this project that would be determined by the code mentioned in Figure 10: Virtual object detection function. A means to create the pulse waveform, this would be the Arduino Due, explored more in the next section 4.3. Microcontroller and lastly the stimulation component in contact with the wearer, this being the electrodes covered in 4.5. Electrode.

The ability to produce a waveform signal with the Arduino Due makes it an optimal choice for a microprocessor. In this application the microcontroller should be able to receive an analog command digitally.

4.3. Microcontroller

The microcontroller of choice was an Arduino Due, this board contains a micro-USB cable port with an AC-to-DC adapter. It is important to note that unlike other microcontrollers this board runs off a maximum 3.3V. Any voltages higher than this will damage the input/output pins and the board itself. The board is based on the Atmel SAM3X8E ARM Cortex-M3 CPU. It is the first Arduino board based on a 32-bit ARM core microcontroller. That has 54 digital

input/output pins (of which 12 can be used as PWM outputs), 12 analog inputs, 4 UARTs (hardware serial ports), a 84 MHz clock, an USB OTG capable connection, 2 DAC (digital to analog), 2 TWI, a power jack, an SPI header, a JTAG header, a reset button and an erase button [23].



Figure 14: Arduino Due Microcontroller²⁵

The controller has several features allowing it to provide an interface and communicate with external devices and peripherals, a bit more detail on the few listed above, below:

- PWM²⁶ – Accessible via the `analogWrite()` command function, this command creates voltages between full on at 3.3v and full off at 0v. With this I'm able to generate a square waveform to get a varying analog output.
- Digital I/O pins – The board has 54 digital pins that can be utilised for either input or output accessible via the command functions, `pinMode()`, `digitalWrite()` and `digitalRead`. Each pin can provide a current source of 3mA or 15mA.
- Serial – There are four serial ports readily available that can receive RX and transmit TX, TTL serial data.

The pulse created by the microcontroller combined with an amplifier is necessary to create a sensation on wearers skin. The voltage in this project for an RS3232 type signal will be up to -100V to +100V. As the microcontroller only supplies 0V - 3.3V I will have to convert this voltage and make the signal biphasic and then amplify it.

²⁵ Image found on Arduino website: Store.arduino.cc. n.d. Arduino Due | Arduino Official Store. [online] Available at: <<https://store.arduino.cc/arduino-due>> [Accessed 6 April 2020].

²⁶ Pulse Width Modulation is a technique for getting analog results with digital means - Arduino.cc. n.d. Arduino - PWM. [online] Available at: <<https://www.arduino.cc/en/tutorial/PWM>> [Accessed 6 April 2020].

4.4. 5DT Data Glove

In this project I used a 5DT Data Glove Ultra 14, this is a hand motion capturing device that can be used in a variety of different ways, but its main purpose is for use in animation and virtual reality. Considering the nature of this project and its objectives, this glove is an ideal medium for use in delivering electrotactile stimulation to the wearer. The 5DT Data Glove measures finger flexure as well as the abduction between the fingers. This measurement is taken between the first and second joint knuckle on every finger [24].

The glove is designed for use of people of varying hand shapes and sizes, being made of a pliable black Lycra material that stretches and withdraws when necessary. Alongside this feature there are a few more mentioned below:

- Accurate and sensitive data response - In Figure 15: Output reading of finger flexure in Glove Manager we can see a real-time scaled sinusoidal output response normalised between zero and one in Glove Manager²⁷ where the finger flexures are being recorded and shown. The graph displays a raw sensor value coming from the glove. An extreme bending of the finger will produce a higher peak amplitude whereas a slight bend will generate a small peak. The speed at which you bend the finger is also considered allowing for a narrow pulse width and a sharper response or a wider pulse width with the same degree of accuracy.

Thanks to the fourteen inbuilt fibre optics cable sensors, two per finger and one for the knuckle the glove can generate clean and reliable signals with no need for additional filtering.

²⁷ A tool to record finger flexure readings and to perform diagnostic testing to see if glove is working.

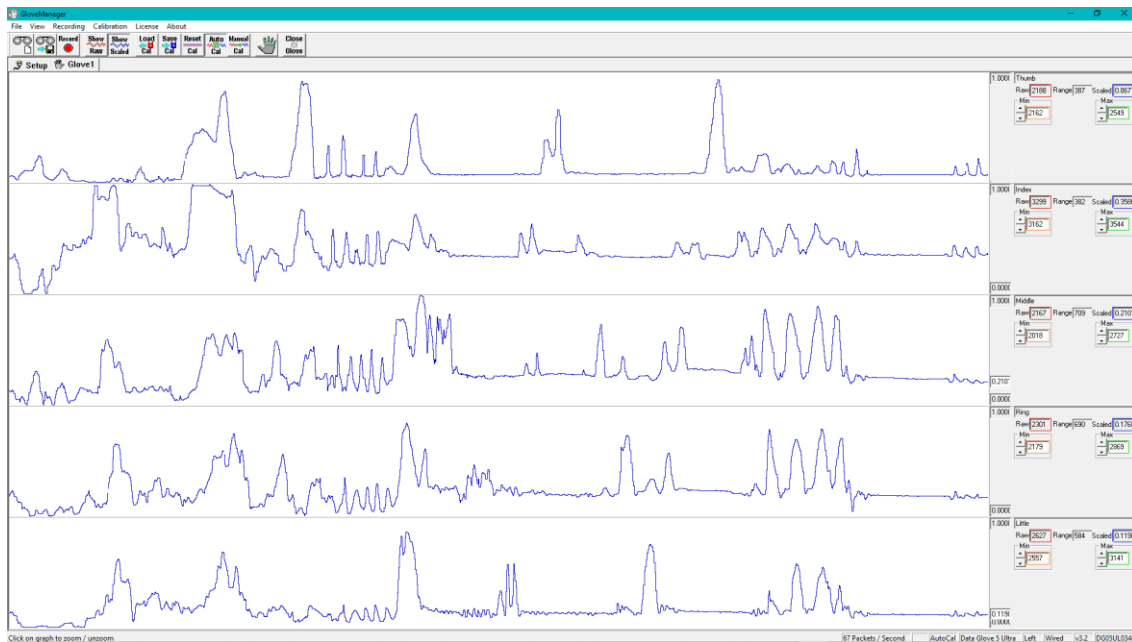


Figure 15: Output reading of finger flexure in Glove Manager

- **Diagnostic Software** - As briefly mentioned in footnote 27, and to elaborate more here, the Glove Manager software has an extensive array of features that make it beneficial for the developer to use throughout their project. With this software you're able to perform routine glove diagnostics, record live raw data values and general glove data to a file. There is also a multichannel feature that allows for two gloves to be connected at the same time with automatic calibration techniques built in. And TCP/IP remote data operation capabilities.
- **Multi-platform use** – The 5DT Data Glove can be used on a variety of platforms, initially coming with a plugin for use for Autodesk MotionBuilder. Nonetheless it has functionality on applications such as Discreet 3D Studio Max, and Kaydara MOCAP and of course Vizard. See Figure 16: MotionBuilder capture of 5DT Data Glove binding and Figure 12: Graphical User Interface that the wearer interacts with for a graphical representation.

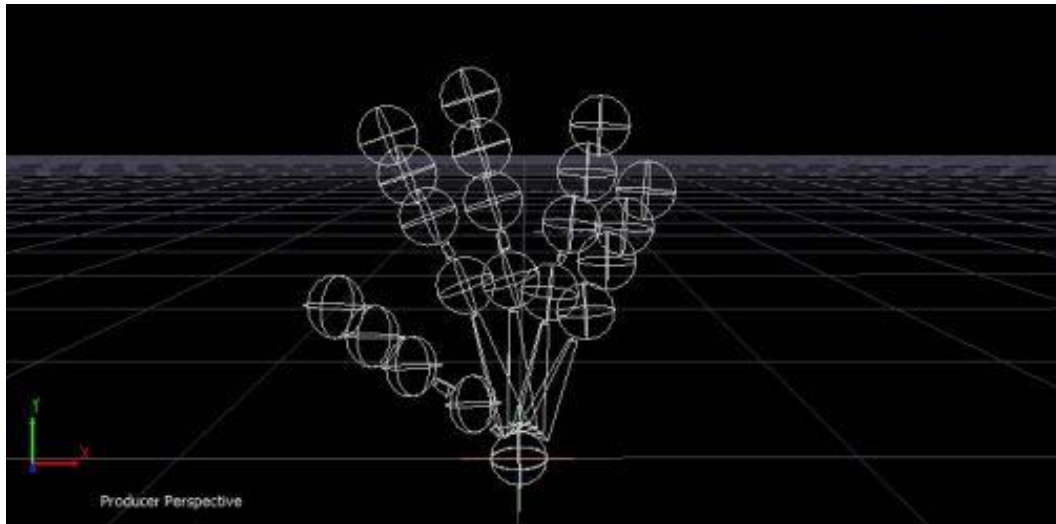


Figure 16: MotionBuilder capture of 5DT Data Glove binding

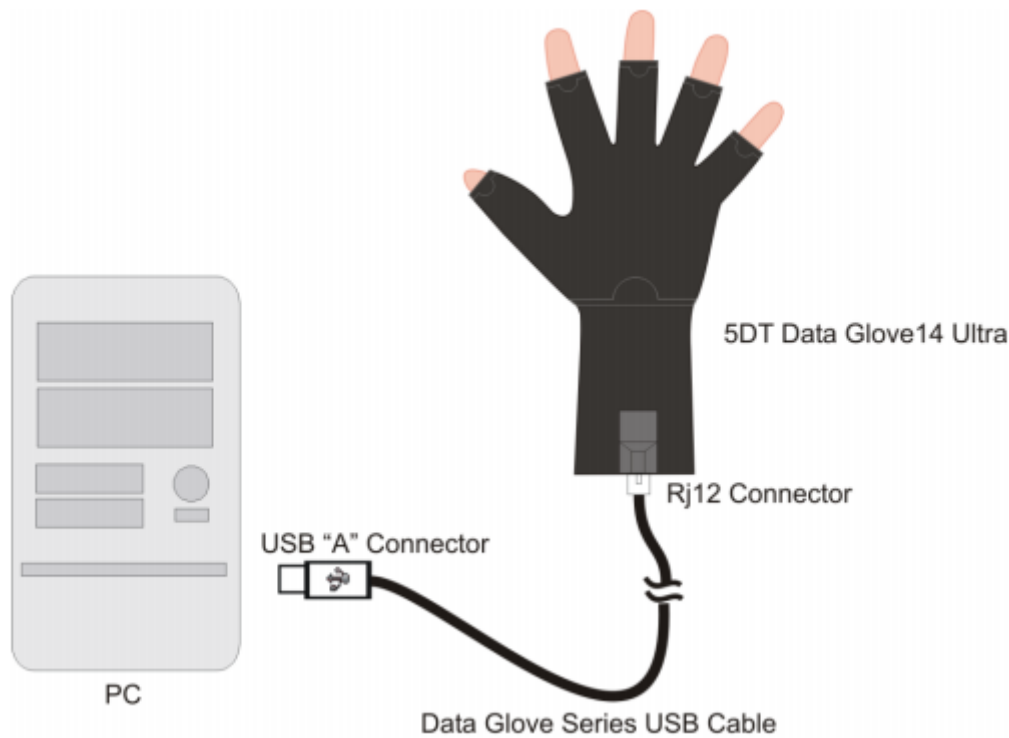


Figure 17: 5DT Data Glove Diagram²⁸

²⁸ Image found on 5DT website: 5dt.com. n.d. 5DT Data Glove Ultra Series. [online] Available at: <<https://www.5dt.com/downloads/dataglove/ultra/5DT%20Data%20Glove%20Ultra%20Manual%20v1.3.pdf>> [Accessed 6 April 2020].

4.5. Electrode

The electrodes used in this project are round Axelgaard PALS Electrodes, they're made from a conductive cloth. These electrodes provide neurostimulation²⁹, and flexibility and conformity to ensure optimal current distribution [25]. These electrodes can increase patient comfort during stimulation by dispersing current evenly across the electrode whilst eliminating stinging, edge biting, and hot spots [25]. As seen in Chapter 2: Background Reading and Literature Review that shape of the material of the electrodes has a positive effect on the stimulation sensation produced. As a result of this, as well as the shape and size of the fingertips I decided to cut the electrodes down to size, significantly reducing the surface area, but created a more localised stimulation point of contact making sure to keep the circular aspect to keep current peaking to a minimal.



Figure 18: Axelgaard PALS Electrodes

²⁹ Neurostimulation is the intentional modulation of the nervous system via invasive or non-invasive means.

4.6. Schematic Diagram

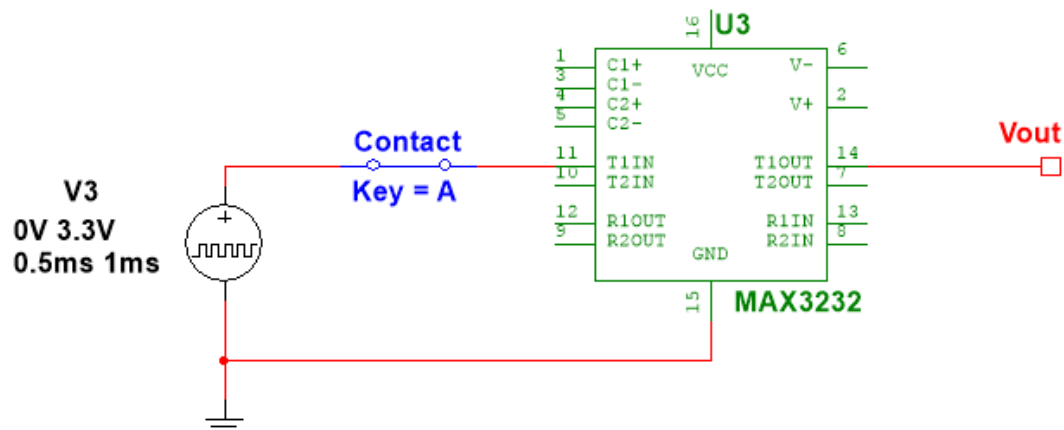


Figure 19: Voltage waveform converter

The above circuit in Figure 19: Voltage waveform converter is a top-level representation of the circuit to be implemented. The circuit consists of a voltage source, taken from the Arduino, the MAX3232 which is the chip that converts a TTL waveform to an RS232 one. As well as a switch which will act as the 'contact' for testing purposes to open and close the circuit in a representation of when the wearer has contacted an object. The MAX3232 chip in Multisim is merely a placeholder and cannot be used for simulations, so the next step is to create a sub-circuit in place of the chip that will perform the function required.

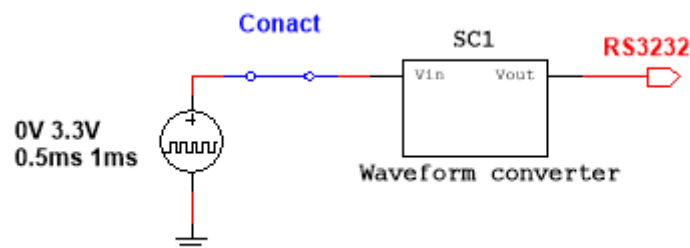


Figure 20: Top level with sub circuit

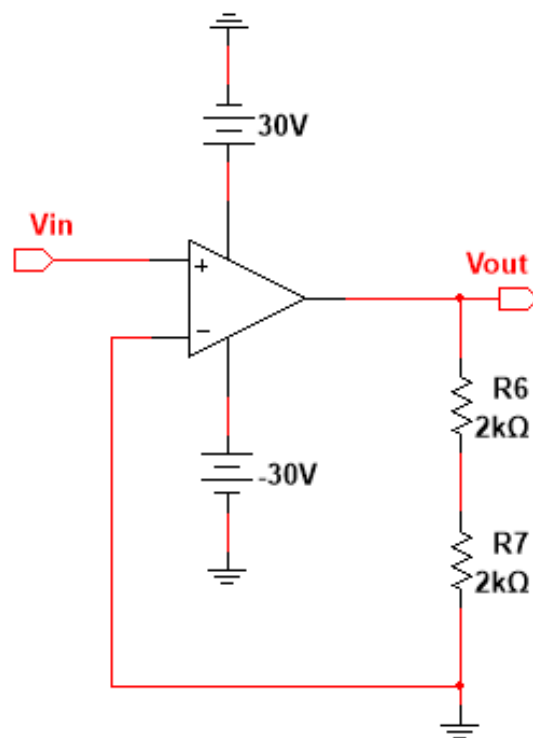
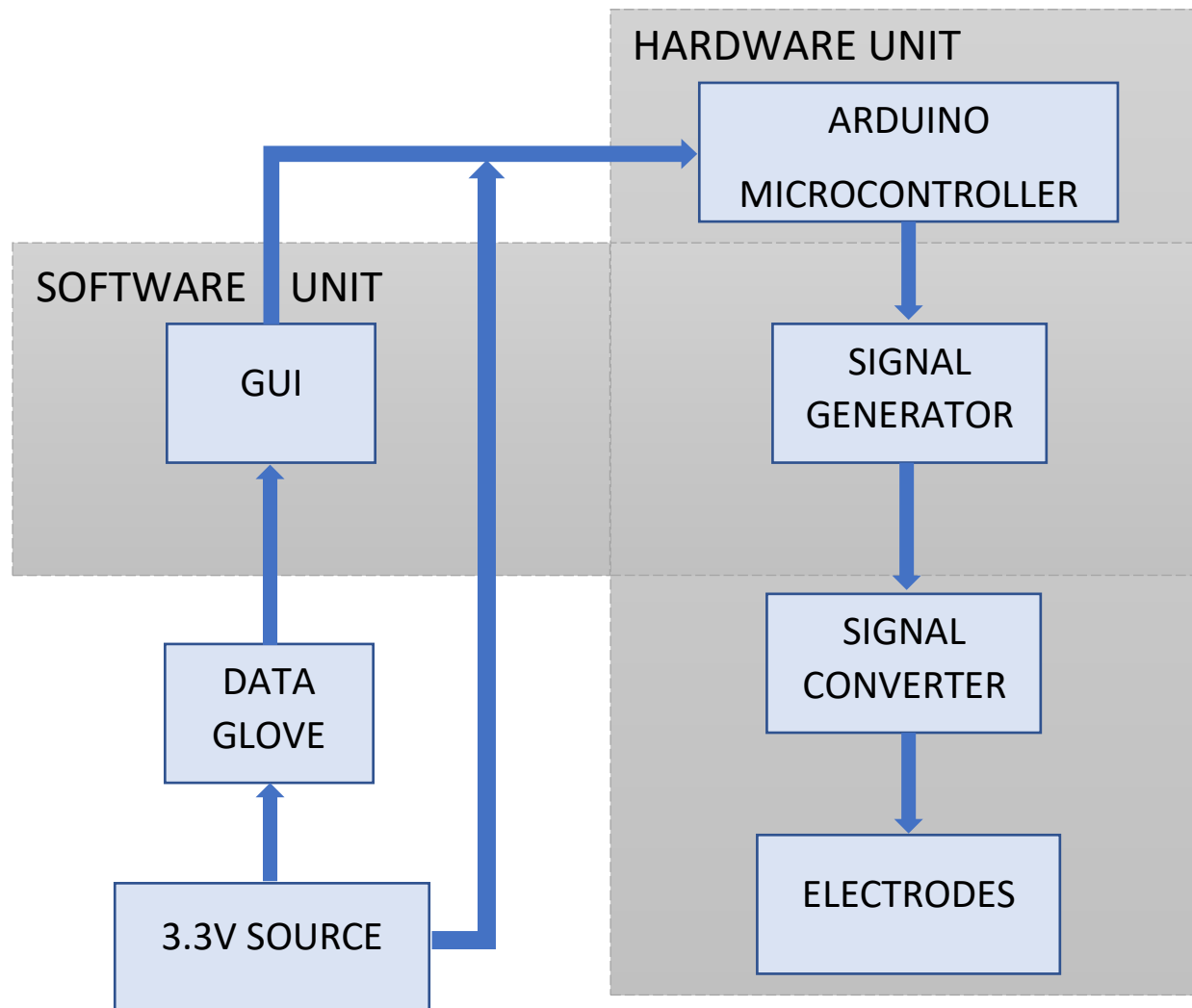


Figure 21: MAX3232 circuit

Figure 20: Top level with sub circuit is the new top-level design, with a sub-circuit block in place of the original MAX3232 chip placeholder. Once expanded you will see in Figure 21: MAX3232 circuit a simple non-inverting op-amp circuit. This op-amp will take the monophasic $0V - 3.3V$ square wave signal in with the purpose of converting the signal into a biphasic square wave signal with a slight voltage amplification.

4.7. Integration



A general overview of how the main system components are linked together. Figure 22: System Architecture shows a 3.3V power supply driving the 5DT Data Glove and Arduino microcontroller which in turn feeds into the waveform modulation section consisting of the signal generator, signal converter and electrodes.

Figure 22: System Architecture

Chapter 5: Results, Evaluation and Discussion

5.1. Software Results

The results from the software testing were generally consistent with what I wanted to achieve. For the software portion python implementation, I was able to isolate the raw data values of the 5DT Data Glove using the ctypes³⁰ function. This allowed me to see the fingers flexure as an integer and provide me with the means begin my first testing with a breadboard, resistors and LEDs by setting a boundary value for each finger and having the python script continuously check whether the finger had passed that value. Only once it had passed the boundary did the code send a message serially to the Arduino board.



Figure 23: Vizard Environments

Whilst working with Vizard I initially started experimenting on the surroundings and then the population of objects. The idea was to get something simple but interactive on screen that

³⁰ A function library that provides C compatible data types. With this function it is possible to call functions in DLLs.

would not distract from the projects main purpose but would allow for a successful demonstration. I worked with different environments (see Figure 23: Vizard Environments, from the top left going clockwise: gallery, dojo, piazza and ground) adding in different environment modules to see what they looked like in the virtual world, in the end settling for the ground environment, this being a very plain single gridded grey square platform. After which I looked to import the hand model into the new scene created, this was a simple process and I played around with the size, shape and orientation of the hand to see what felt right.

There are still some setbacks with the responsiveness of the grabbing feature. Where the wearer may initiate the closed fist hand gesture and the cursor is over the given target. Occasionally the program is unable to recognise that an object has been selected and will not perform the required movement operation. Though the grabbing feature has a high success rate with the current code. I have considered implementing an alternative solution that has the chance to produce better results. It involves spatial recognition, where the cursor is detected hovering over a barrier mask placed equidistantly around the object (see Figure 24: Object Masking for a 3D representation of this). The mask would be relatively tight to the object. When the wearer gets within the object mask the code can recognise that as a selection. As you can see below the purple hex represents the barrier mask and the plain white ball, the object.

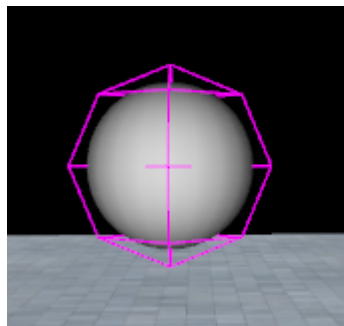


Figure 24: Object Masking

Considering the main objective of the project is to provide electrotactile stimulation feedback to the visually impaired once contact with an object has been made, the current implementation of the code is insufficient. This is because there is no way to move the hand through the virtual space and the use of a cursor would not be beneficial to someone who cannot see it. However, that is primarily due to the lack of a hardware component that can bring the spatial movement functionality to life. One way to resolve this issue would be to use the VIVE Tracker, this device would allow me to create a wireless connection with the 5DT Data Glove providing me the opportunity to develop a better version of the system where it would allow for gyroscopic tracking in the virtual world. So, if the wearer were to move their

hand forward, backwards, up or down the hand model in the virtual world would move with them mirroring the action performed.

With this new functionality and the successful spatial recognition code on the target object it would allow the wearer to move their hand around the virtual space until detection of the object is made.

5.2. Hardware Results

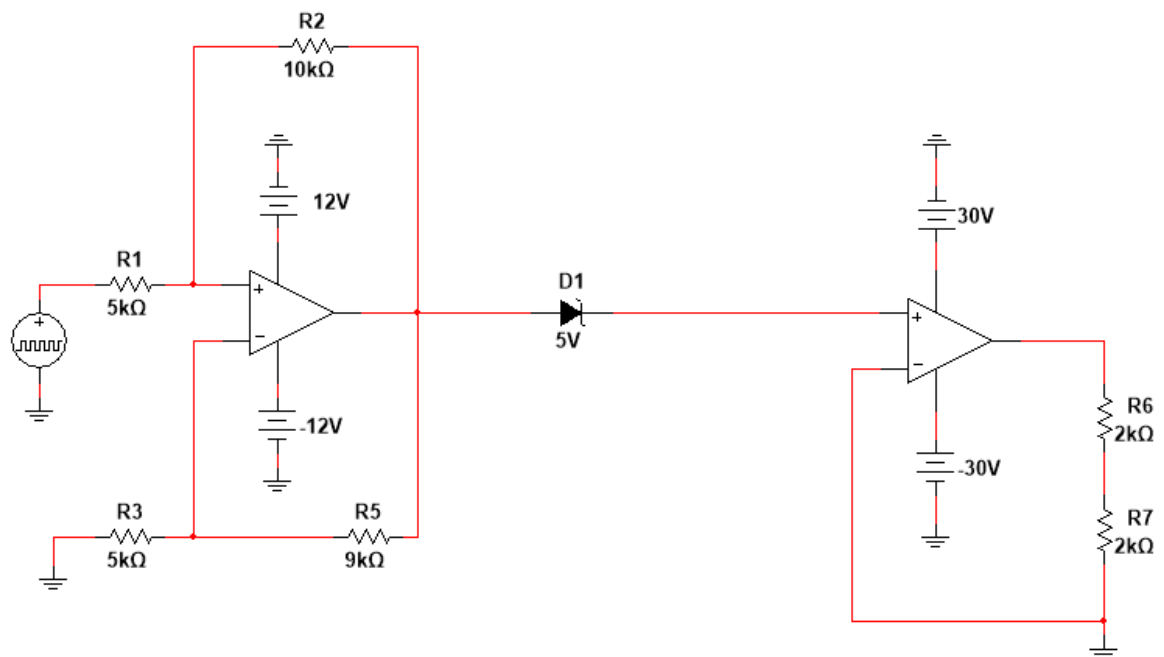


Figure 25: Howland Current Pump with MAX3232

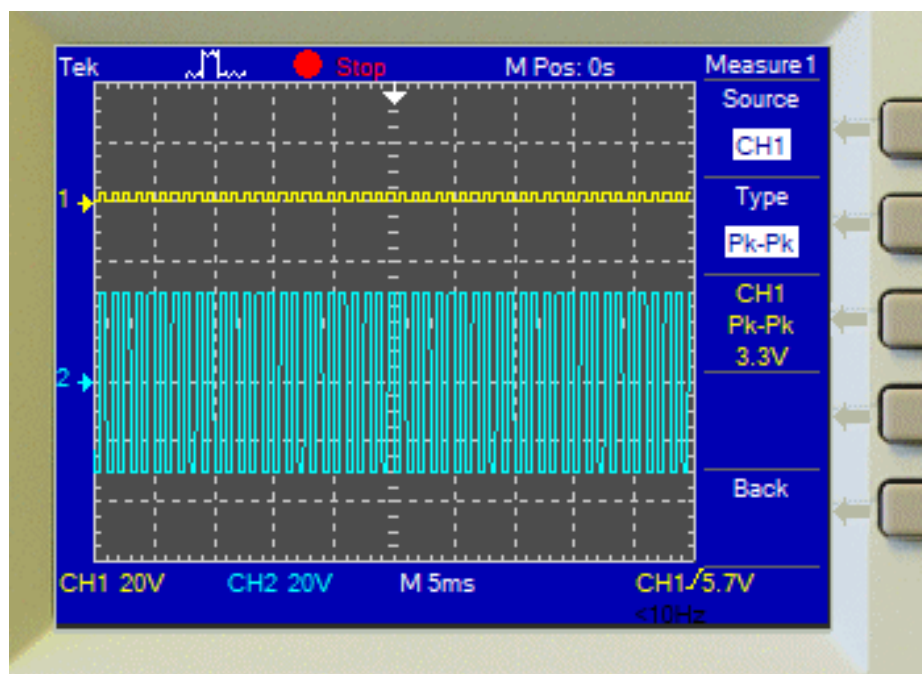


Figure 26: Input output waveforms

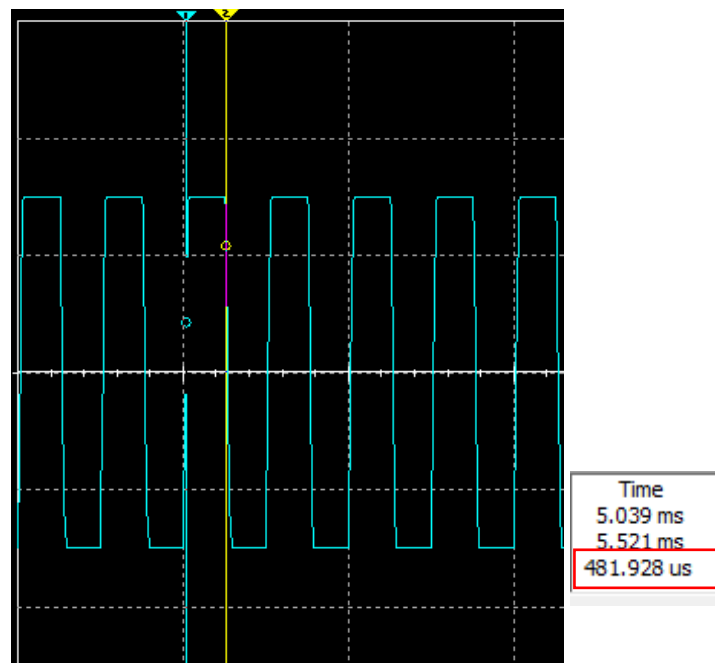


Figure 27: Output signal pulse width timings

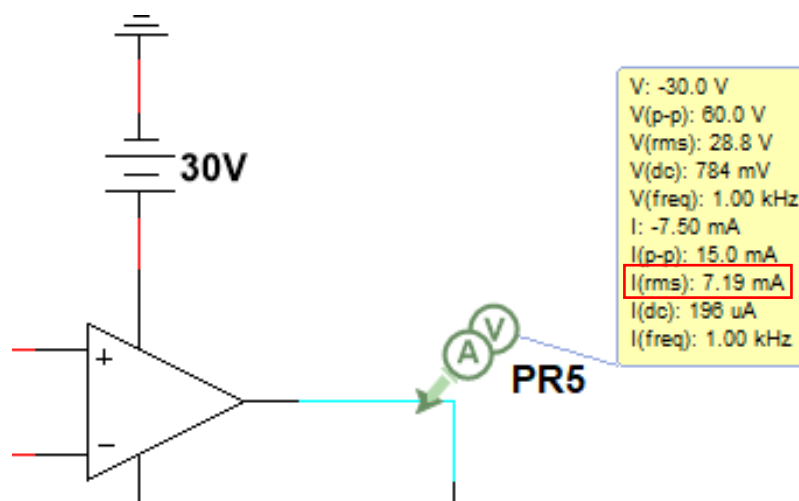


Figure 28: Output voltage and current readings

When taking the virtual oscilloscope reading from the Arduino Due pulse voltage simulator, ensuring the PWM outputs waveform matches what I would expect to see from the code written. Based on the results of the voltage waveform converter/amplifier simulation circuit, I can confirm that the recordings taken from the virtual four channel oscilloscope and voltage

and current probes fall into an acceptable range where participants can detect electrotactile stimulation with the proposed design.

In order to employ the design in a practical setting, it would be necessary to stimulate the mechanoreceptors and to record the response of its activity to better match the needs of the individual and determine it is performing as intended. In previous experiments historically, practical experiments show that participants tend to perceive differences in intensity of electrical stimulation depending on where the electrode is positioned and how big the stimulation surface area is. As such the instability of the concentration of stimulation may cause uncomfortable electrical sensations upon the skin. The reason for this seems to be that the impedance of the skin varies depending on the location. Due to this, the use of the Axelgaard PALS Electrodes becomes increasingly prevalent as it provides an even current distribution at a specified current threshold across all its surface area. It is also possible to alter sensation by altering the current threshold of electrical stimulation according to the position of the electrode on the user's finger and individual in general by increasing the feedback resistance or reducing the level of amplification on the converted waveform generated.

The device prototype requires a stimulator that can deliver a short high and low burst with a pulse width of roughly $10\mu s - 500\mu s$, the pulse must be delivered with a current in the range of $10mA - 20mA$ with a voltage up to $-100V$ to $+100V$ with close to zero DC offset and a 50% duty cycle. It is important that a 50% duty cycle is used, because if a continuous application of oscillating current is applied with prejudice to a greater magnitude in the negative region. The fingers will get a build-up of ions that will make the sensation uncomfortable if not painful. The circuit in Figure 25: Howland Current Pump with MAX3232 takes in the Arduino voltage square wave and converts it to an output current, with the MAX3232 providing the signal conversion and voltage amplification required to produce the particular output needed.

In Figure 26: Input output waveforms you can see a snippet of the input voltage waveform (yellow signal) and the output signal (blue signal). On initial assessment it can be observed that a distinct transformation of the waveform has occurred. Where the input signal is monophasic and has a significantly reduced amplitude the circuit diagram has converted that signal into a biphasic signal with a much greater amplitude. Figure 27: Output signal pulse width timings measurements show the timing of the pulse widths at approximately $480\mu s$

Chapter 6: Project Management and Reflection

Project management tools are an integral part of maintaining a projects lifecycle, this is because the ability to track objectives and issues becomes such a useful ability when developing a system/device. In this project I use Jira as it provides the ability to simplify the organisation of work and allows for easier collaboration amongst developers to ensure goals are clear and easy to follow. As an agile tool you can scale the project to your needs by breaking the objectives down into smaller tasks to achieve a solution. Alongside this I also use a version control system, GitLab to manage the workload in a centralised repository to aid in project planning and source code management.

6.1 JIRA and Gitlab screenshots

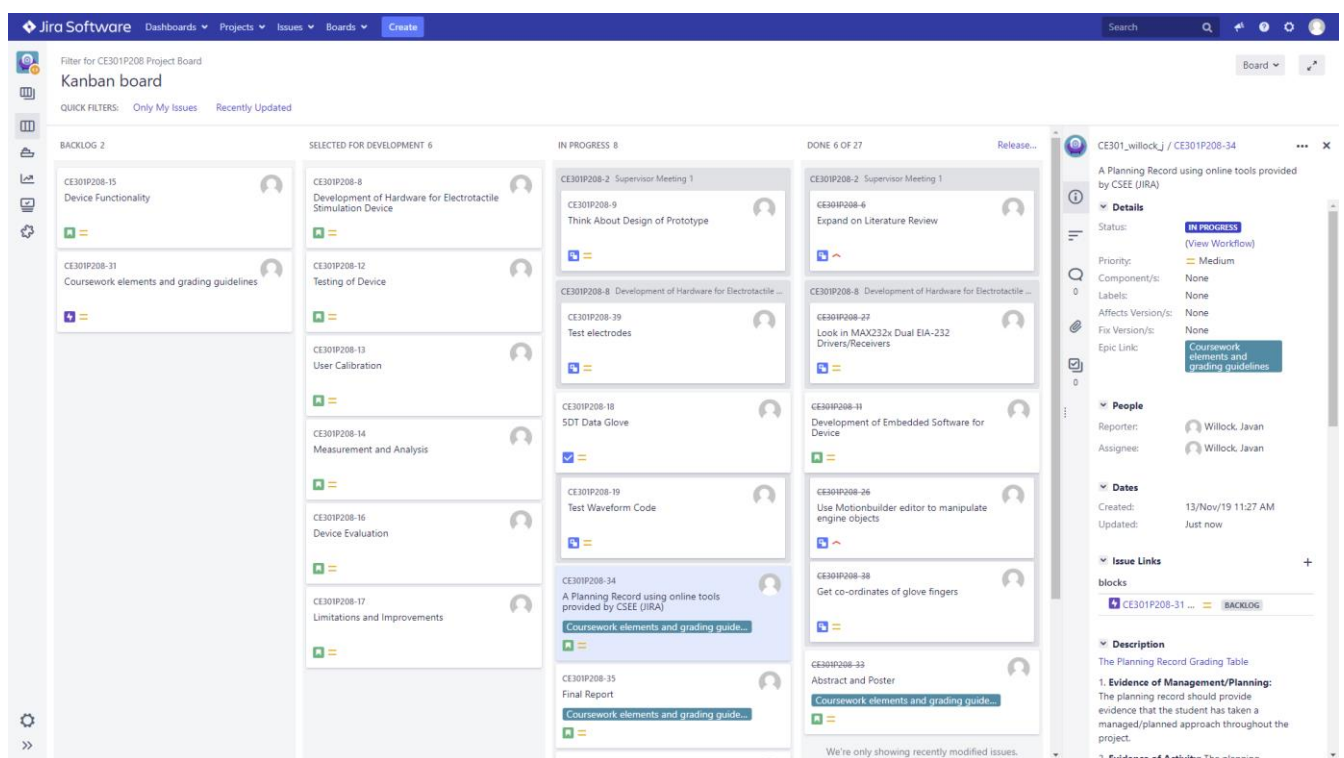


Figure 29: Screenshot of JIRA Kanban Board

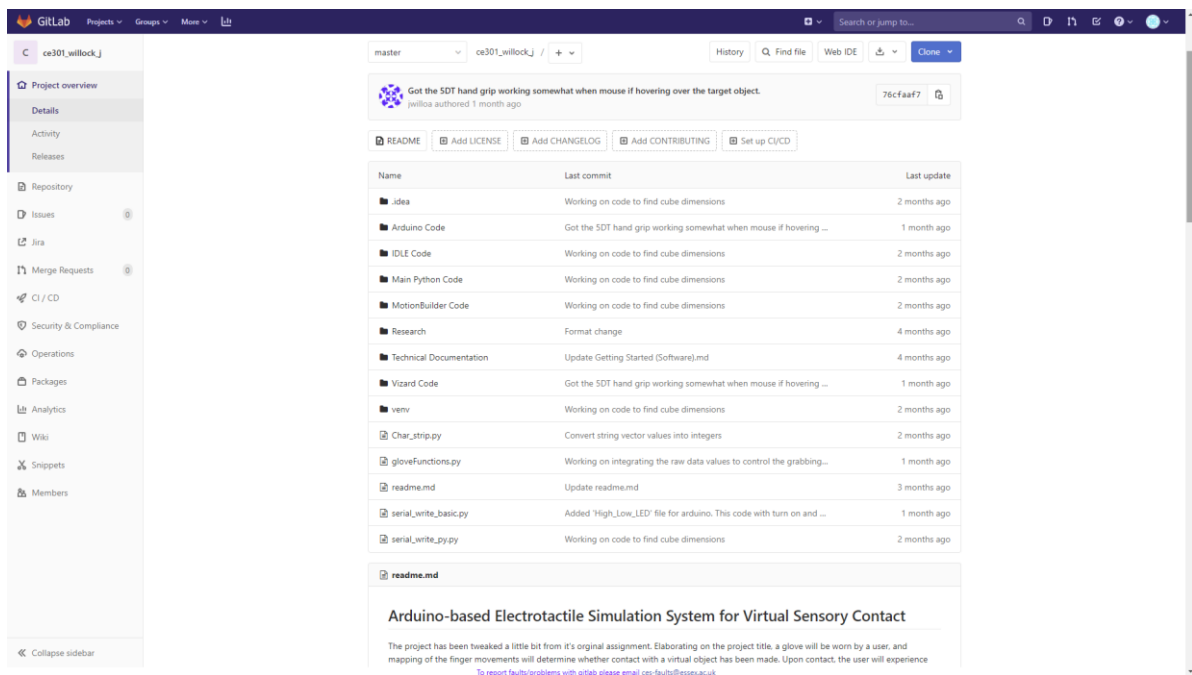


Figure 30: Screenshot of Gitlab Repository

CE301_willock_j / ... / CE301P208-6



Expand on Literature Review

▼ Details

Status: **DONE** (View Workflow)
Priority: High
Component/s: None
Labels: None
Affects Version/s: None
Fix Version/s: None
Epic Link: None

▼ People

Reporter: Willock, Javan
Assignee: Willock, Javan

▼ Dates

Created: 10/Oct/19 12:26 PM
Updated: 1 week ago 18:36 PM

▼ Description

Develop approximately five pages worth of background reading

▼ Comments



▼ Willock, Javan added a comment - 08/Apr/20 6:36 PM

Built up an extensive background literature review that highlights some key aspects relevant to the project

Comment

▼ Attachments

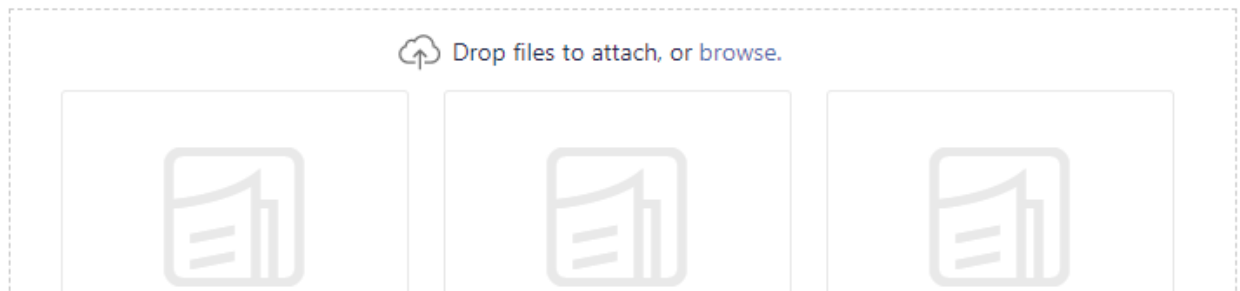


Figure 31: Example User Story Description

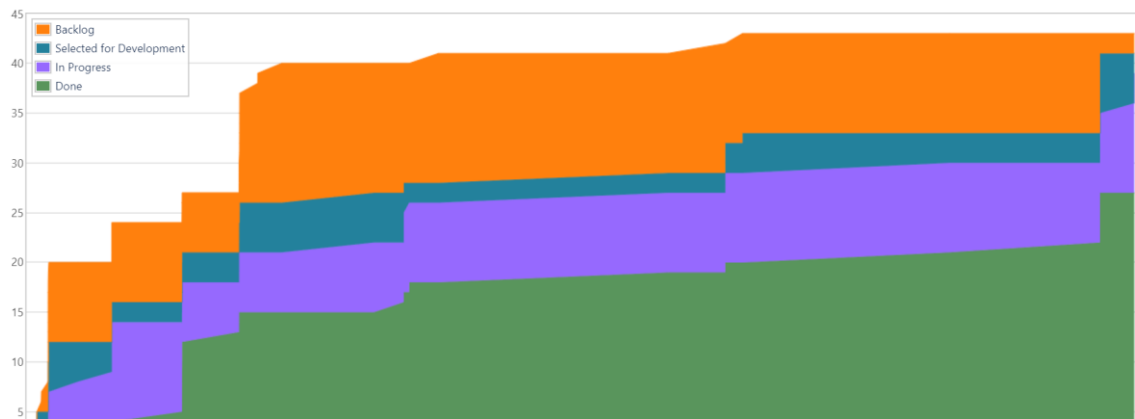
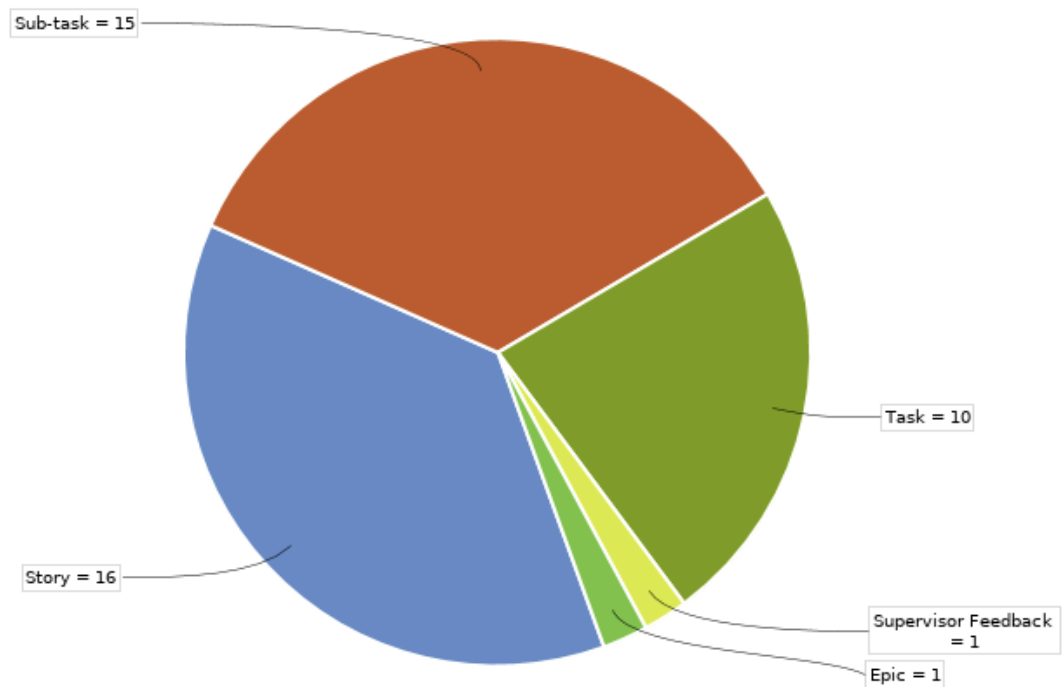


Figure 32: JIRA Cumulative Flow Diagram



Data Table

	Issues	%
Story	16	37%
Sub-task	15	34%
Task	10	23%
Supervisor Feedback	1	2%
Epic	1	2%

Figure 33: Issue Type Pie Chart Report and Table

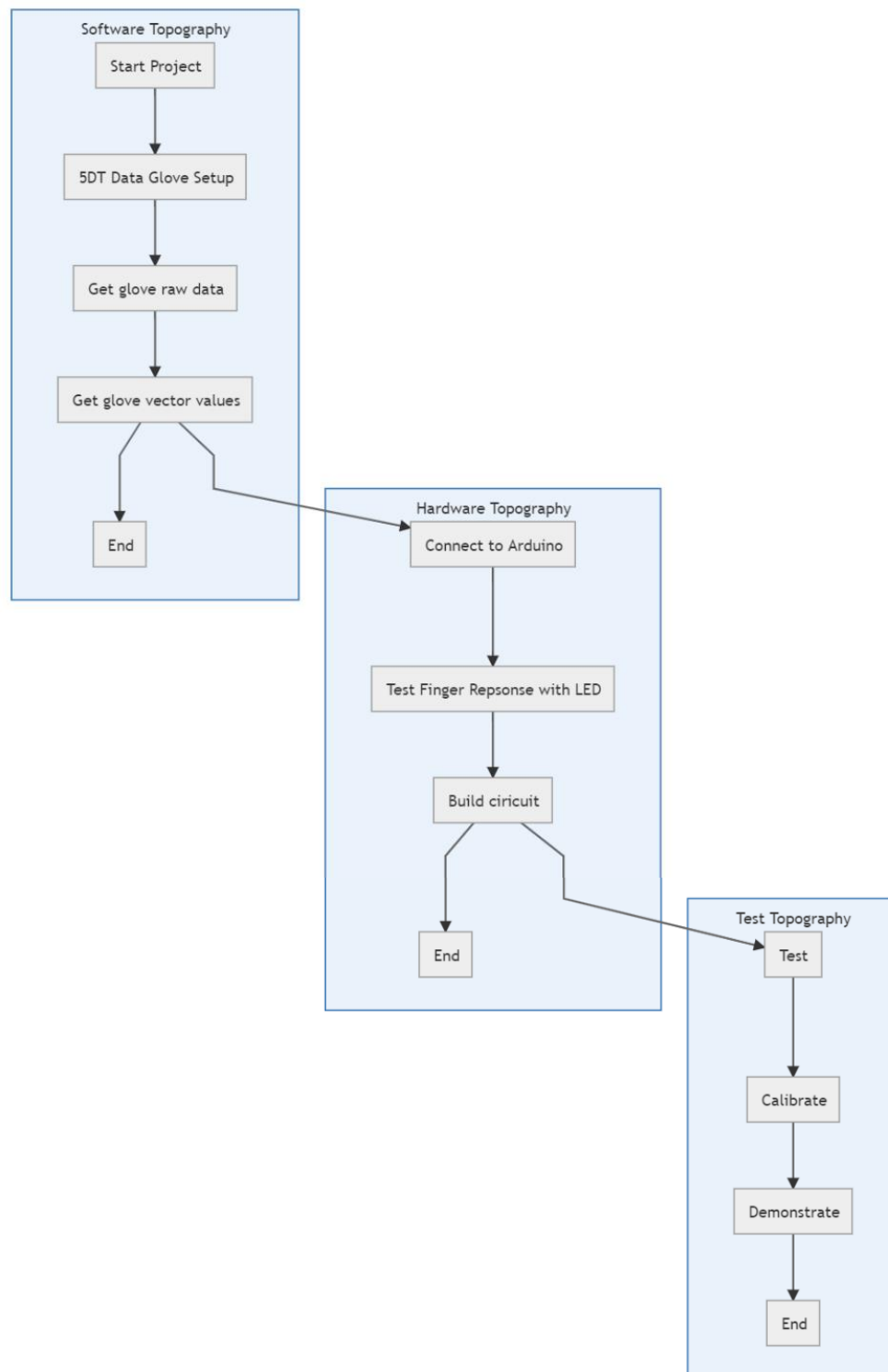


Figure 34: Project Topography

6.2 Project Tracking

The use of JIRA firstly allows me multiple ways of viewing information, one example being the Kanban Board as seen in Figure 29: Screenshot of JIRA Kanban Board. This is a 'to do' list with columns separated into four segments, backlog, selected for development, in progress and done. With this list I'm able to drag and drop any tasks, epics³¹ or stories³² in those respective columns. It is this feature that allows me to easily manage my workload and priorities. I can see what objectives take precedence at a glance by seeing where they are located on the board.

During the project building stage, I created as many issues as necessary to build up an overview of main objectives needed to get the project to completion and sub objectives that would help improve the functionality of the project. On the Kanban Board when I created an issue I was able to specify a summary title of the task, and give a description on what that task entails as well as providing an acceptance criteria to be met, this is principally what I expect to see by the completion of the task and I can use this to ensure that the task is following the requirements needed. Once I had ensured the task had been fulfilled, I marked it as 'Done' and put in the time it took to complete the task and any useful additional comments of attachments.

Gitlab on the other hand provides a very structured way to store, access, view and edit your files. See Figure 30: Screenshot of Gitlab Repository for a graphical representation. One of the main benefits of Gitlab is the ability to connect to the server remotely from any capable device and set up a local repository on your device to manage your files. Once this is done, I can use applications like PyCharm to directly push or pull my files to and from the server. In the instance where I needed to return to a previous state, I could look into the history of my 'commits' and fall back on to a version I know worked and fix it from there.

With the project in full swing the security of Gitlab allows me to work without the worry of losing any files. After connecting my Gitlab to PyCharm I was able to start making .py files and directory folders. The PyCharm interface allowed me to easily see where my files were being kept, providing a similar layout to Gitlab. It also served the added benefit of allowing me to access code outside of the standard python language, code such as C/C++ or the MotionBuilder python variation code and the Vizard variation code. At the start of the project I was able to make a readme.md file, this is a file that I edited to make the 'homepage' of my repository. This homepage provides context to the project, detailing a small abstract and providing a brief explanation of what electrotactile stimulation is to the readers. As well as

³¹ Epics are large quantities of work that need to be broken down into smaller tasks called stories.

³² Stories are a list of requirements written from the perspective one who is or will be using the product.

displaying a topography of the project (see Figure 34: Project Topography). In the readme.md application I attempted to build something of a website by linking .md files, these links would allow myself and the viewer to navigate to certain aspects of the project, e.g. Hardware or Software, and following a step-by-step walkthrough on how to get started, based off of my own experience as well as application used and what version, and where to get them.

6.3 Self-Critical Reflections

There were some key parts during the development stages of my project that required me to make some key decisions that would determine which way I decided to tackle my project. During the beginning stages of this project I spent roughly a week looking into different game engines (MotionBuilder, Unity and Unreal Engine) and testing their capabilities with the hopes that I could get a basic hand model shown in its virtual environment that was connected the 5DT Data Glove plugin. As such I spent a lot of time doing some research online and getting used to the different layouts each application provided. Though attempting to do this with the Unity resulted with no successful outcome and it seemed that I would need to develop some visual studio code to get it working. Unreal engine was of a similar result, and I decided not to take this application further as I would need to build an additional module to allow for the glove plugin to connect. MotionBuilder on the other hand came with a very useful guide on how to start up use of the data glove, detailing ways to implement the plugin whilst demonstrating its use. Due to the immediate success of MotionBuilder and its scripting language being Python I decided to go with this option.

Whilst messing around with MotionBuilder I spent a lot of time looking into their documentation, specifically their software development and application programming interface in the hopes I could learn their unique class hierarchies and modules. I found that there were little to no guides online that I could refer to, so this was the only option available to me. This provided a great learning opportunity and I spent time figuring out what functions I would need to get the software part working. Though this would eventually prove insufficient. I found that as I progressed and slowly built up the base application. The results produced during testing showed that the glove mask in the virtual environment by itself was capable of real-time movement, but any attempt at an interaction with an object would cause the scene to freeze as the script performed its operation. Only after the script had completed its iterations did the environment update. I think the reason for this is because MotionBuilder is primarily an animation software, but also because the MotionBuilder python scripting language functionality is a subset of the open reality C++. These functions being available through the application's graphical user interface. So, because the plugin is written in C++ any python code executed is not able to access the C++ plugins. I believe that this problem could've been partially resolved if the code was written in C++ and compiled into a dynamic link library. I have reasonable confidence that it would've allowed the code to detect when

contact had been made, as any object in the environment that would have been generated via in-built python scripting can be executed by C++ code. And so, data would already be flowing through the subset from the top level.

However instead of following this through, I found an alternative solution to the problem. Rather than using MotionBuilder I looked around and found Vizard, a python based virtual reality software. This being a much more versatile and data glove friendly, with the ability to respond to the wearer's movements real-time.

Conclusion

This paper covers the theoretical and practical applications of an electrotactile feedback system for use in a virtual environment using a voltage waveform converter and amplifier with a closed loop design. I have proposed a possible design capable of delivering the required amount of stimulation to the fingertips. With Multisim I simulated the waveform response from the projected voltage output of the Arduino Due, and the simulation results have shown the circuit is practicable for use in the real world. Even though live testing on human participants could not be performed with the device prototype I hope that this paper provides insight on things to consider and that the potential for use is possible with the correct tools. By implementing the code and hardware mentioned above in and Chapter 4: Hardware Design and Integration it may be possible to generate real world results on sensitivity and accuracy to electrotactile stimulation in the virtual environment by performing psychophysical³³ experimentation.

³³ Psychophysical experiments test a person's perception in stimulus detection and difference detection methods.

References

- [1] S. H. K. K. a. P. B.-y.-R. M. Tyler, "Development of an electrotactile glove for display of graphics for the blind: preliminary result," IEEE, Houston, TX, USA, USA, 2002.
- [2] T. H. Nguyen, T. H. Nguyen, T. L. Le, T. T. H. Tran, N. Vuillerme and T. P. Vuong, "A wireless assistive device for visually-impaired persons using tongue electrotactile system," IEEE, Ho Chi Minh City, Vietnam, 2013.
- [3] W. Liu and H. Tang, "An initial study on lip perception of electrotactile array stimulation," IEEE, 2005.
- [4] J. W. Schroeder, "Creating Tactile Feedback with Intelligent Electrical Stimulation to Compensate for Sensory Impairment," 2014.
- [5] A. J. Szeto and F. A. Saunders, "Electrocutaneous Stimulation for Sensory Communication in Rehabilitation Engineering," IEEE, 1982.
- [6] D. Agnello, "New-Generation Fully Programmable Controller For Functional Electrical Stimulation Applications," 2011.
- [7] D.-R. Tsai and W. Hsu, "A new type of a large-area multi-touch tactile device operated by electrotactile stimulation," IEEE, 2019.
- [8] A. R. Ward, S. L. Toumbourou and B. McCarthy, "A comparison of the analgesic efficacy of medium-frequency alternating current and TENS," Musculoskeletal Research Centre, 2009.
- [9] H. Kajimoto, N. Kawakami and S. Tachi, Psychophysical evaluation of receptor selectivity in electro-tactile display, ResearchGate, 2003.
- [10] A. d. Lahunta and E. Glass, "Somatic Afferent," Veterinary Neuroanatomy and Clinical Neurology, 2009.
- [11] N. Gleveckas-Martens and E. Crisan, "Somatosensory System Anatomy," 2013.
- [12] Lumen, "Lumen Learning," [Online]. Available: <https://courses.lumenlearning.com/boundless-ap/chapter/the-somatosensory-system/>.
- [13] V. G. Macefield, "Physiological characteristics of low-threshold mechanoreceptors in joints, muscle and skin in human subjects," CEPP, 2005.
- [14] S. Ha-Ju, O. Han-Young, K. Seong-Hyun and K. Dong-Wook, "A Study on Estimated Sub-threshold Intensities of Somatosensory Stimuli Based on Somatosensory Evoked Potentials of EEG," IEEE, 2013.

- [15] L. P. Parades, S. Dosen, F. Rattay, B. Graimann and D. Farina, "The impact of the stimulation frequency on closed-loop control with electrotactile feedback," *Journal of Neuroengineering and Rehabilitation*, 2015.
- [16] J. P. Warren, L. R. Bobich, M. Santello, J. D. Sweeney and S. I. Helms Tillery, "Receptive Field Characteristics Under Electrotactile Stimulation of the Fingertip," *IEEE*, 2007.
- [17] R. Edelberg, "Relation of Electrical Properties of Skin to Structure And Physiologic State," New Jersey, 1977.
- [18] J. W. Gregory, "Development Of Haptic Electrotactile Rendering Device Design Implementation And Testing," 2011.
- [19] T. Yamamoto and Y. Yamamoto, "Electrical Properties of The Epidermal Stratum Corneum," *Medical and biological engineering*, 1979.
- [20] U. Pliquet, R. Langer and J. C. Weaver, "Changes in the passive electrical properties of human stratum corneum due to electroporation," 1995.
- [21] S. Meers and K. Ward, "A Substitute Vision System for Providing 3D Perception and GPS Navigation via Electro-Tactile Stimulation," *IEEE*, 2005.
- [22] K. Choi, P. Kim, K.-S. Kim and S. Kim, "Two-Channel Electrotactile Stimulation for Sensory Feedback of Fingers of Prosthesis," *IEEE*, 2016.
- [23] Arduino, "Arduino Due," [Online]. Available: <https://store.arduino.cc/arduino-due>. [Accessed 9 April 2020].
- [24] 5DT, "5DT Data Glove Ultra Series," [Online]. Available: <https://www.5dt.com/downloads/dataglove/ultra/5DT%20Data%20Glove%20Ultra%20Manual%20v1.3.pdf>. [Accessed 9 April 2020].
- [25] Axelgaard, "PALS Electrodes," [Online]. Available: <https://www.axelgaard.com/Products/Electrodes/PALS/ProductLine>. [Accessed 11 April 2020].
- [26] K. A. Kaczmarek, K. M. Kramer, J. G. Webster and R. G. Radwin, "A 16-Channel 8-Parameter Waveform Electrotactile Stimulation System," *IEEE*, 1991.
- [27] K. Sato and S. Tachi, "Design of Electrotactile Stimulation to Represent Distribution of Force Vectors," *IEEE*, 2010.
- [28] R. J. Onesti, W. J. Tompkins, J. G. Webster and J. J. Wertsch, "Design of a Portable Electrotactile Stimulator for Sensory Substitution Applications," 1989.
- [29] T. Hirotsugu, X. Mingdi, H. Takehito and I. Keiji, "Effect of Electrical Stimulus Intensity to Hemodynamic Responses of Somatosensory Cortex," *IEEE*, 2007.
- [30] P. M. V. K. M. N. B. a. O. R. I. Khoo, "Design of a biofeedback device for gait rehabilitation in post-stroke patients," *IEEE*, Fort Collins, CO, USA, 2015.