Creating an ergonomic glove instrument for portable beat-making and MIDI interaction

Development of a wearable device for musicians with hybrid compatibility as a standalone instrument and MIDI controller

Francisco Rosado Correia

Sound and Music Computing, XXXX, 2021-05

Master's Project

STUDENT REPORT
AALBORG UNIVERSITY

AALBORG UNIVERSITY
STUDENT REPORT

Sound and Music Computing
Aalborg University
http://www.aau.dk

Title:
Creating an ergonomic glove instrument for portable beat-making and MIDI interaction

Theme:
New Interfaces for Musical Expression

Project Period:
Spring Semester 2021

Project Group:
XXX

Participant(s):
Francisco Rosado Correia

Supervisor(s):
Daniel Overholt

Copies: 1

Page Numbers: 58

Date of Completion:
May 19, 2021

Abstract:

This project focuses on the creation of an affordable, ergonomic, fingerless glove device that functions in both MIDI mode allowing for interaction with commercially available DAWs as well as a Standalone mode that lets the device function as its own instrument. Sounds and MIDI notes are triggered by piezoelectric sensors placed at the fingers. A complementary hybrid tool dubbed the control panel was developed to further improve the musical interaction, allowing for MIDI modulation and on-site control. This thesis places emphasis on the analysis of several different design and implementation methods to develop hand-worn musical instruments, as well as highlighting differences between two developed design iterations. A brief usability test scheme used to validate the device's concept was performed, focusing on video demonstrations and participant feedback. The test was performed on an initial iteration of the device where only MIDI capabilities were shown and the feedback acquired allowed development to shape the presented version. The proposed project is deemed a successful implementation of a responsive hybrid musical interaction tool, where the research performed upon explored the topic could provide a baseline for further development of both hand-wearable and hybrid compatibility devices.

Contents

Preface

Aalborg University Copenhagen, May 19, 2021

Francisco R Correia

Francisco Rosado Correia
<frosad19@student.aau.dk>

Chapter 1

Introduction

The art of traditional beat-making processes helps bridge recorded music and machine interaction in a very interesting way where artists like J Dilla, Nujabes and Flying Lotus are famed examples of how this bridging occurs through several techniques, where the producer also functions as the composer, by methods of sampling and warping said samples into new compositions [27]. These artists and many more have made use of drum samplers and drum machines, such as the Roland SP-404 and the legendary Akai MPC series of samplers, used notably in late J Dilla's critically acclaimed record "Donuts" [5], more specifically the Akai MPC3000.

This type of technology that was once considered quite hard to come across has through time become more readily and easily available, both in a hardware format and a digital format. Prime examples of these easily available drum samplers can be observed in Teenage Engineering's *Pocket Operator* series, where small, circuit board-like pieces of hardware can be used to record samples and play them back, such as the PO-33 model [33].

Nowadays, the bridging between analog and digital music composition is seen all around, mainly through the use of the MIDI (Musical Instrument Digital Interface) protocol [15], offering methods for musicians to quickly send analog inputs through their controllers that can be converted into digital messages, which through the use of Digital Audio Workstations, commonly known as DAWs, has been a complete revolution in the music production methodology. Beat-making no longer requires drum samplers, but yet, they are still very much alive and well within the community.

The technological progression of drum machines has been in a state of continuous change and improvement, where more compact designs are being pushed alongside new ways to interact with music. Instruments such as the MI.MU gloves developed by artist and musician Imogen Heap keep pushing the boundaries. By creating a set of gloves with various sensors on them such as accelerometers, one is able to control music with movement [17]. Combining these concepts of portability, sampling and MIDI interaction sparked an idea for a new way to interact with the art of beat-making and music production, by creating a wearable glove that can be used to create beats by finger tapping. The finger tapping beat synchronisation habit [23] that is commonly observed in humans was also a powerful influence in the creation of this project.

In this thesis, an innovative and intuitive approach to musical interaction is presented in the form of an ergonomic wearable musical tool that focuses on the finger tapping motion rather than interaction with an external device. Such an approach is achieved by placing piezoelectric sensors on a glove's fingers that capture the finger's pressure when tapping on a surface. With hybrid compatibility, functioning both as a MIDI controller and as its own instrument, not needing interfacing through DAWs, a device of this nature proves to be an important contribution to the field of portable, wearable, beat-making and MIDI controlling devices by not being restrained to a singular practice, aiming at belonging in all four fields, bridging the gaps between them. This project is aimed at musicians, performers and beat-makers and presents to them a new complementary tool for MIDI production techniques as well as a portable musical tool to jot down ideas and to play with on the go.

The MIDI mode is optimised for the Fruity Loops Studio DAW (more specifically FL Studio 12), as the user can register this glove as a MIDI input where through the native FL Studio plugin FPC, which can be considered as a digital emulator of the MPC for Fruity Loops, different user-selected sounds can be assigned to whichever finger they prefer by mapping them to the corresponding MIDI note value. Another feature that was added as inspired by the MI.MU glove was a sensor that allowed for modulation of a user-selected effect by means of an accelerometer. This easy to setup feature can be assigned to whatever controllable parameter from any VST (native or not) as long as MIDI CC (MIDI Control Change) is tolerated, and by moving or tilting the hand up and down, the parameter is modulated. This modulator can also be considered as a real-time automation drawing tool, that when recorded, can register the rate of change of the parameters over time. In the Standalone mode, the device no longer requires any type of interfacing with a DAW as the drum sounds are implemented directly in the device. The user can connect their favourite set of earphones directly into the headphone jack and start creating immediately.

The prototype being presented is not limited to just a glove, having a complementary control panel that can be mounted on the forearm or placed near the user's desired performing space, dedicated to extending the usability of this device even further, boasting more interactive options for the user to experiment with. These include two knobs and four buttons with their own unique interactability depending on the mode being used.

Along this report, a literature review of similar equipment and devices that in some way can contribute to this research is presented, providing relevant insight into them, and showcasing ways that certain approaches taken can be adapted to fulfil the topic at hand as well as identifying important contributions to the field and where certain improvements could be made, as seen on chapter **??**. Furthermore, a scientific evaluation of the use of piezoelectric sensors in this scope is performed, accompanied by extensive research on these sensors and other commercially available options. Although this report mainly focuses on the latest iteration of the prototype, the first official iteration is also discussed extensively, evaluating its downfalls and how the latest iteration corrects them. It is important to place focus on this first iteration as an evaluation and usability test was performed for it, as elaborated in chapter 3, where participant feedback was taken into account into developing the latest prototype.

In this chapter, research materials gathered throughout the development stage of this thesis are to be presented, ranging from products and fellow projects in the same scope that helped inspire in some form the functionality and design of the glove, to the components and materials used in the manufacturing process, as well as other potential ones. Details are provided on why and how these items were considered for the development of this thesis during its progress.

The main scope of this thesis aims to create a wearable instrument that can function with both connection to a DAW and in a Standalone mode, meaning that no connection to a computer is necessary. Finger drumming utilised by musicians with a primary focus on the art of beat-making served as the main inspiration for the thesis, noticing the interaction between the artist and the drum sampler, tapping drum pads that trigger a sample or drum sound. These drum samplers and controllers are widely available, ranging from MIDI controllers such as Novation's Launchpad, to self-sufficient ones like the Roland SP-404. These are usually compact and portable enough and boast many functionalities outside of simply finger drumming such as controlling MIDI CC messages.

It is possible to argue that these MIDI controllers are not fully intuitive to make the most use of, usually having DAW specific interactions (as an example, Arturia and some Novation products have native compatibility to Ableton Live, or different to some extent in comparison to Fruity Loops Studio) which could make interchangeability between mediums a complex task. It should be noted that MIDI-only products always require a computer to interface between the user and music, which severely reduces portability and usability outside the recording space. Another trait that could be argued to make these products less intuitive, shared with Standalone controllers, is the specific input combinations necessary to achieve certain features, although these arguably allow for extended capabilities, and can be powerful tools when mastered.

By no means are the aforementioned traits diminishing to a user's musicianship, however, this project is aiming at exploring this middle ground left out in these products, aiming at functioning both in MIDI and Standalone modes, keeping interactions simple and intuitive, as portable as possible while being quite user customisable. These aspects are combined into a simple glove controller, both MIDI compatible and Standalone, with five inputs corresponding to each finger, an accelerometer for effect control and accompanied by a control panel that can be used alongside the glove should the user desire, which can be mounted on the forearm, or placed on a surface. These features, which will be further elaborated throughout this report, culminate in a wearable device that aptly functions as both a MIDI controller as well as an instrument on its own.

Wearable musical instruments seem to have been garnering attention in recent years, where one of the most notorious pieces is Imogen Heap's MI.MU Gloves [17] as seen in figure 1.1 which allows the user to manoeuvre their hand as a controller or modulator, mainly utilised for live performances and production when used with the accompanying Glover VST [11], to help map hand gestures detected by the incorporated flex sensors to sounds within a DAW or to control MIDI and OSC messages. The fingerless design allows for the user to perform other tasks while wearing the gloves, which can both be beneficial due to less intrusion while performing said tasks or even to map gestures for them. The device communicates over Wi-Fi allowing for complete freedom during performance.

Figure 1.1: A MI.MU glove, taken from [17].

Contributors to the Adafruit DIY community have submitted their projects inside this scope. Notably, contributors Becky Stern and Limor Fried submitted their own MIDI glove [31] [32] that makes use of piezoelectric microphones at the glove's fingertips, connected to a FLORA board. This design makes use of solely four piezos, one for each finger excluding the ring finger, and they are programmed to replicate keyboard keys, which in most DAWs are linked to a certain MIDI note. Another project by John Park, the CLUE BLE MIDI Glove [21], makes use of an accelerometer sewn into a glove to send MIDI CC signals to a BLE synthesiser.

These designs were taken into consideration when creating the proposed thesis project and adapted to a more unique design, such as using the fingerless approach while using the piezos as inputs, placing them on the back of the finger joints rather than the fingertips.

Utilising piezoelectric sensors allows inputs to be registered from the vibrational response from the components facing a finger tap, however, other components have also been used in other projects to achieve similar effects. The Tap Strap 2 keyboard [26] in figure 1.2 for instance is a wearable product used for typing, assigning certain finger tapping combinations to a respective letter. The way this works is by strapping a band containing individual rings for each finger, where each ring contains a 3-axis accelerometer. The downward motion of finger tapping is detected by the accelerometer which sends a Bluetooth signal to the corresponding input. Not only that but the thumb also boasts an optical mouse, which allows the Tap device to be converted to a mouse if the hand rests on a surface. Such an approach could be carried over for musical processes, using the same method as the aforementioned MIDI glove using the computer keyboard to trigger notes, however, it is costly to produce and can be seen as impractical as it requires the user to learn a new way to type.

Figure 1.2: The TAP Strap 2 keyboard, taken from [26].

Another item of note on wearable instruments would be the DrumPants project [8], which is based on flexible drum pads that the user can attach to their clothing and shoes that when hit will send a MIDI or OSC message via Bluetooth. This project is compatible with DAWs such as Ableton Live and Reason and is even compatible with video games or other tools like Google Slideshows, and boasts a controller for the user to swap between sound banks. The creators do not disclose the materials used in their trigger pads, but the concept of portability used in this product is maximised as the positioning of the trigger pads is not limited to the legs and feet as they can be attached wherever the user desires and is even showcased being incorporated onto a drumkit kick drum to also trigger MIDI notes.

The Sphero Specdrums [6] project focuses on using a singular ring to trigger sounds on its dedicated apps. The compact ring is to be used against colours which will trigger an assigned sound and it is aimed towards a similar target group as this thesis project, however, over time the developers dropped MIDI support.

An important contribution towards the field of wearable devices is the TapID [16], a wearable device that is placed on the wrist that interacts with VR technology for various uses including keyboard typing and piano playing. The compact design focuses on a thin wristband with sensors that track finger movements and replicate them in a VR environment, aiming to break standard VR practices that mostly make use of camera-based movement tracking. The sensors are comprised of four accelerometers, and the data processing is based on a machine-learning algorithm that detects the finger tap probabilities to then process which hand movements are being performed. This type of advancement would certainly be beneficial for musical interaction, allowing users to use various instruments in the VR realm. Furthermore, this type of design could show promise of a significant improvement in the current glove-based instruments or interfaces, abandoning the glove altogether, allowing for much freer movement.

Although not a wearable instrument, an interesting item to note would be Koka's Pocket Beatbox [20], a 4-pad portable drumkit that allows users to load their own samples via SD card. The pads are velocity-sensitive, and an interesting feature about the velocity sensitivity is how the velocity of the hit can also be used to trigger different sounds, as a certain velocity value will trigger a certain sound, which allows for a very interesting and dynamic approach to beat-making. This device also allows for both MIDI input and output alongside standard headphone output, and there is even an expression pedal input. The pressure-sensitive pads work at a 12-bit resolution that allows for accurate input response.

Chapter 2

Development

In this chapter, the project design and development will be discussed. Amongst this discussion, comparison and contrast between different building strategies are performed, as well as highlighting the current features implemented and their functionality.

Throughout this chapter there will also be a focus on the developing stages of the proposed glove project, noting its progress and evolution and providing adequate reasoning for the alterations made, from the beginning moments of its creation until the current, proposed iteration. Any further possible development and concepts that had to be postponed or outright scrapped in the presented iteration are elaborated in chapter 5.

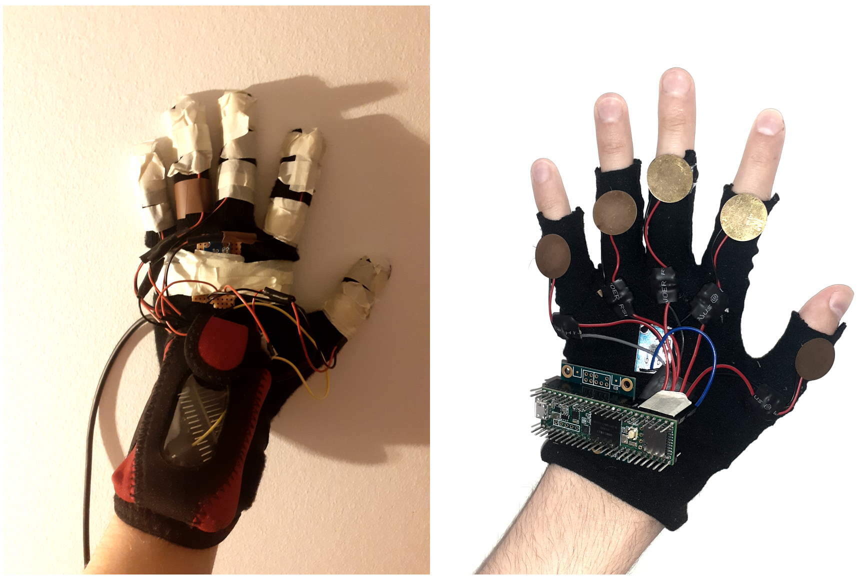

Figure 2.1: The glove instrument. On the left is the first iteration of the glove, with the sensors at the fingertips. On the right is the latest iteration, showcasing the fingerless design.

2.1 Glove Design

The proposed thesis project focuses on bringing a platform for beat-makers, musicians, and performers to interact with their craft, with a design inspired by the interactions of musicians with drum samplers and sequencers as well as the common practice of following a beat by tapping the fingers along. As such, the design aims to be ergonomic and comfortable, being a wearable glove with sensors placed on its fingers. Furthermore, the platform being a glove allows for tapping on any surface to trigger MIDI notes or sounds. Through development, two main design iterations were created: an initial one that followed a similar approach to the aforementioned FLORA-based MIDI Glove [32], having the sensors placed at the fingertips, whereas the latest one focuses on a fingerless design approach where the sensors are placed at the back of the finger joints. The first iteration functioned only as a MIDI controller, whereas the latest supports both a MIDI controller mode and a Standalone mode. Both iterations can be seen in figure 2.2 and are discussed in this section and adequate reasoning for why certain approaches differ is provided alongside it.

Before proceeding into elaborating the hardware and software implementation, the materials, components, and equipment used as well as the glove's assembly should be discussed, explaining their relevance within this project's context.

The glove material required that it fit two criteria: comfortableness and flexibility to ensure a pleasant performance experience. Two gloves made with different materials were trialled and tested, one mainly composed of acrylic and polyester and another being mostly made out of cotton. The former glove was used for the first iteration of the glove, where it proved to fill the two criteria, however, it became apparent that due to their main intent to be used as cold weather outerwear that prolonged indoor use would become fatiguing, as the hands would become too warm. In contrast, the cotton gloves were lighter, thinner, and much less constraining. Both gloves were trialled by wearing them and performing the finger tapping motion for approximately ten minutes in two formats: unadulterated glove and fingerless, where the fingers of the gloves were cut approximately halfway to each finger's joint. These tests were performed by a single person as performing the test during the ongoing COVID-19 pandemic could pose health risks that were unwilling to be taken. This point is further detailed in chapter 3. Upon trialling, it was concluded that the fingerless cotton glove was the better option for the finished product. It should be stated that for the first iteration, the unadulterated polyester glove was used, with no other comparative data. These two gloves can be observed in figure 2.2.

To detect the finger tapping motion, piezoelectric sensors were used due to the capacity to register vibrations from the tapping. These sensors function by registering the vibrational or pressure information as electrical charge, from which

Figure 2.2: The two gloves used during development. Left: polyester glove. Right: cotton glove.

the data can be recorded analogically and converted into a digital format. These types of sensors have seen usage in various other fields such as biomedical practices [12] to create cardiac sensors or blood pressure monitors by harvesting the vibrational energy outputted by the heart. Applying the fundamentals of piezoelectric sensing, it can be deduced that these materials will perform well to detect finger tapping, as exerted pressure from this action and its residual vibration are factored in. These sensors are available in various formats, from the ceramic (PZT) piezoelectric contact microphones such as the ones used in these project to the polyvinylidene fluoride (PVDF) films, each with their appropriate uses and advantages. PVDF films have been also utilised for musical expression similarly to the usage of the ceramic sensors as evidenced in [25].

Ideally, PVDF sensors such as the ones in figure 2.3 would have been used within this context mainly due to their flexibility and dimensions as they are usually just thin films, as the brittleness of the PZT sensors caused severe issues during development, such as constant breaking or deformation, but the decision to use them ultimately resulted from their affordability [1]. Other components to perform the same task were also considered, namely accelerometers and force-sensitive resistors (FSR).

Accelerometers could be implemented by placing them on a finger and tracking the position relative to the Y-axis. The rate at which each finger moves downwards for a tap can be registered in a way that by establishing a minimum threshold of velocity, the tap will be registered. As such, for MIDI functions, the corresponding note could be held if there is no significant change in the acceleration after a tap has been registered and can be disengaged once a change is performed. The velocity or gain could also easily be mapped to the tracked finger's velocity. The Tap Strap keyboard and TapID devices make use of such technology [26] [16]. The use of accelerometers as the primary tap tracking components ended up being scrapped as these devices could be impractical due to noise from a user just moving their hand and due to their high price per component. Although accelerometers were not used to track finger tapping, one (ADXL335) was used to function as a MIDI CC controller, placed at the back of the hand, tracking movements along the Y-axis. This feature is desired mainly for performance and will be expanded on further in section 2.3.

FSRs would be a possibility for finger tapping, as these components are sensitive to applied force, such as squeezing and pressure. By placing them at the fingertips it would be possible to register the resistance at the moment of tapping, which in turn could be mapped to the MIDI velocity or gain value, and similarly to the accelerometer approach, the corresponding note could be held down if resistance is maintained below a certain threshold (as resistance decreases the more force is applied). These components were not used as the design shifted for a fingerless approach, however, if the fingerless design were not to be carried out, these would have replaced the PZT piezo components, as these are also flexible materials and somewhat affordable.

In the first iteration of the glove, the piezo sensors were placed underneath the fingertips, being held on to with masking tape. Measured with a diameter of approximately 1cm, these components fit comfortably at the fingertips non-intrusively. This method accurately relayed tapping information back to the microcontroller, but not without its drawbacks. The main concern that arose was component damage due to the direct interaction between the contact microphone and the surface, which lead to damage, bending and chipping as there was minimal protection over the components. Moreover, the direct interaction would strain the connections which repeatedly lead to disconnections between the piezos and the wiring, leading to several reworks.

At this stage, a new method had to be placed that shifted focus away from the direct impact between component and surface, hence a fingerless approach. By placing the piezos further down the finger, on the backside between the tendon and the joint, it could still capture vibrations and still function properly, with the correct calibration. Since the fingertip section of the gloves were no longer necessary, they were cut out, and the leftover fabric reused to keep the piezos in place by sewing a section over the component, leaving a fingerless glove, hence the name. Some core advantages of this design approach include the significant reduction in proneness for component wearing since there was no more direct contact; free fingertips allow the user to perform other tasks with no obstructions (e.g. playing another instrument, using the phone) increasing product comfort and usability;

keeping the sensors closer to the Teensy meant that less wiring would be necessary which allows for not only a more organised design but less potential noise due to shorter wiring. Larger dimension piezos were used later, at approximately 2.5cm diameter. The larger diameter components appeared to be more sensitive to the tapping strength than the smaller ones, which is accurate to [18] [19] as it denotes that larger diameter PZT sensors within an approximate 1 to 5cm range output higher signal amplitude, meaning in this context that the tapping signal is more pronounced. The masking tape was abandoned and instead the sensors were held on using hot glue and the wires were sewn onto the glove. The sensors used during development can be seen in figure 2.4. The dimensions of the piezo sensors had to be adequate to fit the glove, as despite larger piezos providing higher peak data, they still had to be limited to a certain diameter to not interfere with each other and to appropriately fit within the design.

Figure 2.3: A PVDF film sensor.

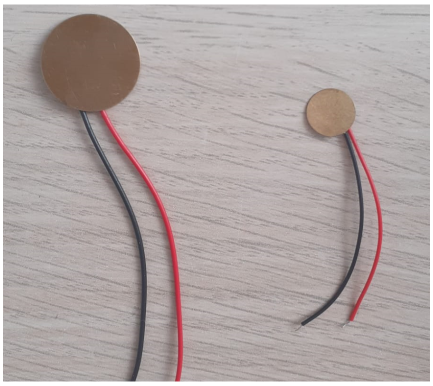

Figure 2.4: The PZT piezo sensors used in this project. Left: Used in the latest iteration, diameter = 2.5cm. Right: Used in the first iteration, diameter = 1cm

Using a Teensy microcontroller was seen as the best option available due to being a microcontroller dedicated to audio-visual projects such as this one. The Teensy is known for its easy compatibility with MIDI, with its own Arduino library facilitating these connections via USB. Two models of the Teensy were used during development: the 4.0 version and the 3.6 version. The Teensy 4.0 was used in the first iteration as it only focused on MIDI interfacing, where the 3.6 version was used in the latest one. This apparent downgrade in versions was a conscious decision due to the implementation of the Standalone mode of operation. Other boards were considered, such as the TT-Go Audio board due to its design and Bluetooth functionality which was quickly discarded due to fear of latency issues, opting instead for a direct USB connection. The FLORA board as used in the aforementioned MIDI Glove Adafruit project [31] was also considered since it was proven to work, however the Teensy being an audio-focused board seemed to easily outshine it, as it has direct compatibilities with audio features with its own library. The Teensy 4.0 boasts a much higher processing speed at 600MHz compared to the 3.6 at 180MHz, however, the main difference in this context is the number of available pins [35], the former having 24 pins, 10 of which are analog, without the need for soldering whereas the 3.6 has 40, where 21 are analog as evident in figure 2.5. For MIDI only operations, the Teensy 4.0 has more than the necessary amount even when paired with the accessory control panel (further elaborated on), however, for the Standalone mode, it only leaves four analog pins usable out of the minimum necessary five for each finger due to the connection with the Audio Adaptor Board [34], which requires most pins to be used with it. This board is required for the Standalone operation, with the in-built headphone jack and audio processing integration. The Teensy 3.6 even with the board has the necessary number of pins required for all functionalities, for both MIDI and Standalone modes.

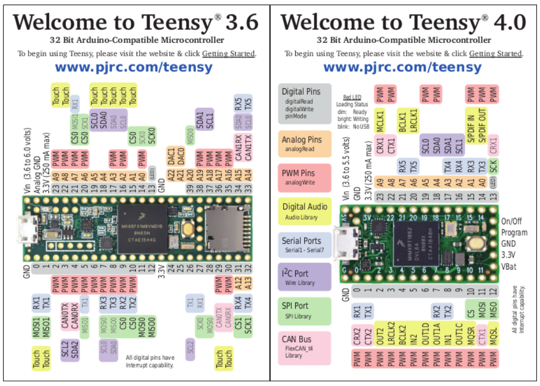

Figure 2.5: Comparison between the Teensy 3.6 and 4.0, adapted from [35].

To circumvent this issue, a potential solution could be to make use of multiplexers, devices that allow several inputs or outputs to be selected through a singular pin. This method would be effective as it would allow for easier segmenting of the entire design, where one multiplexer would be placed on the glove for the analog inputs, and two on the control panel, for the digital and analog pins, leaving only very few wires coming out of the Teensy, resulting in a slicker design. However, implementing these devices would be less cost and time effective than simply replacing the Teensy 4.0 with its previous version. The fact that multiplexers are essentially input selectors could also pose a challenge while performing, as it would only read a singular input at a time from each multiplexer where for example playing tapping two or more fingers at once would have either a delayed response or only one tap would be recorded overriding any simultaneous taps, which could impair the glove's functionality. It should also be stated that a more advanced model of the Teensy boards, the 4.1, is also available which contains an appropriate number of pins for full operation, and will likely be used in the future.

Initially, the Teensy and any excess wiring were placed inside the carrier compartment of an armband, visible on the wrist in figure **??**, with a Velcro strap intended for joggers to carry MP3 players, provided by the thesis supervisor, to keep the hands as uncluttered as possible and the Teensy concealed, however, a noticeable drawback was the impracticality of having to remove these two items, as it required very careful handling to not disconnect the sensors and accelerometer of the glove from the Teensy. Moreover, there were several instances of sensor disconnections, and where the compartment on the armband was too small, the board had to be removed and reconnected. Another drawback from this approach was the necessity to use long wiring, which as already discussed, comes with the drawback of potential noisy signals. These concerns were revised by simply attaching the board directly onto the glove, which meant shorter wiring would be used and disconnects would be more infrequent, and even if they happened, they would be easily reconnected, and only one item would be required to be worn, facilitating the wearability of the glove.

2.2 Control Panel Design

Whereas the glove as described above functions to trigger notes and sounds in its designated modes, a complementary interface was developed to improve the functionality of the device. This interface has been dubbed the control panel, and it can both be wearable on the user's forearm or placed nearby the user's performing area. The panel features four buttons, three LED diodes with one being an RGB one and two potentiometers, each with their own use depending on the mode. A broad schematic of the control panel can be seen in figure 2.6. The dimensions of the panel are 10.8cm in length, 10cm wide and 2.3cm tall, and its assembly on stripboard can be observed in figure 2.7. The control panel has a Velcro strap underneath so that it can be strapped to the forearm. Where an elastic strap could fulfil the same purpose and perhaps even keep the board more stable on the arm, it would be more troublesome to wear and remove. This choice may seem contradictory to a previous statement of keeping the board on a strap, making wearability more difficult, however, the control panel is a complementary accessory that still requires connections and does not impair the normal usability of the glove. The panel can also be used without being strapped to the forearm, should the user desire.

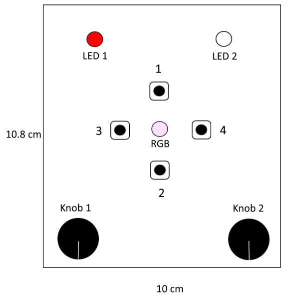

Figure 2.6: The control panel layout schematic.

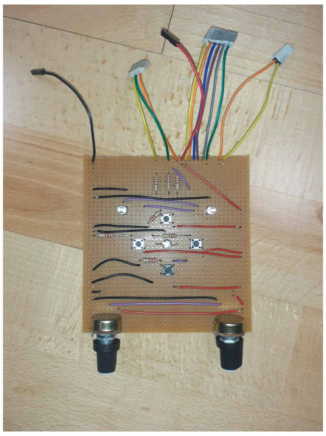

Figure 2.7: Top view from the control panel assembly on a stripboard.

When designing the panel, the potentiometers were kept at enough distance to ensure that while turning each knob, the other one does not interfere. This felt like a necessary decision as several market-available controllers tend to keep the knobs quite close which can get bothersome as the user can accidentally turn an adjacent knob. Each knob is approximately 2.8cm tall with a diameter of 1.5cm at the bottom and 1.3cm at the top. To calculate the distance apart at which the potentiometers should be, a potentiometer was held as if to turn them, and the total diameter of the potentiometer being held by the fingers was measured, at approximately 4cm. Since this measurement is biased as it was taken by a singular person, to this value, some more distance had to be allowed to ensure that fingers do not touch if turning both knobs at once, as that could potentially become uncomfortable, so an arbitrary value of 3.5cm was added to the spacing, also to account for larger fingers.

In MIDI mode, the potentiometers control undefined MIDI CC messages. Undefined MIDI CC allows the user to define a parameter to control within the user's DAW, and assigning these values arguably lets the user have more freedom with their performance, as they can control whichever parameters they desire. In Standalone mode, these potentiometers control the dry/wet values of the reverb and bitcrush effects. These effects were chosen solely under a subjective perspective as adequate effects for drum sounds, where reverb is a very common effect to use, controlled by knob 1 and bitcrush could be a pleasant effect to play with, controlled by knob 2.

The button placement was decided to follow a similar style to that of console controllers, in a D-pad formation since using a familiar layout could perhaps make the interaction more intuitive. The buttons being used are momentary switch pushbuttons, buttons that only activate when pushed down, being idle otherwise.

In MIDI mode, buttons 1 and 2 function to transpose up and down respectively the MIDI notes triggered by each finger by five notes. When mapping sounds to each finger, the sounds are being mapped to a MIDI note value, hence by transposing the note values, more sounds are available to map to the fingers allowing for more creative options. To illustrate, if the index finger is mapped to MIDI note value 50 (corresponding to D3 in western notation), pressing button 1 will increase its value to 55, and pressing it again to 60. Similarly, pressing button 2 once lowers its value to 45 and twice to 40. This system follows a cycle, wherein this example once note value 60 is reached, pressing button 1 again will make its value 40 and in contrast, if the value is 40, pressing button 2 will make its value 60. By following this cycle, the user has a total of five available modes – standard, transpose up once and twice, transpose down once and twice – which are displayed in five distinct colours by the RGB LED placed in the centre. As this mode attempts to focus more on using the MIDI notes as mappable items rather than using them for playing the keyboard notes, transposing in a cycle is more understandable. The decision to have the LED as the indicator of each mode relies on easily distinguishable visual feedback, which can be easily memorised and seen well enough as opposed to say a 7-segment display while providing a subjectively aesthetic appeal to the panel. The user should try and focus on mapping their desired sounds based on the colour being displayed on the LED.

In Standalone mode, these buttons are instead being used to control the volume of the sound outputted by the Teensy, button 1 raising the volume and button 2 does the opposite. The RGB LED is initially turned off but shines green when turned above the standard volume, becoming brighter the louder it is and shines red when below the standard volume, also becoming brighter the quieter it is. The inclusion of a volume controller was deemed a necessity as there are no other ways to control the volume in Standalone mode unless connected to speakers or headphones with their own volume control. Controlling the volume ensures that the user can enjoy a comfortable listening experience at the loudness they desire, and having the option to at least turn down the volume ensures a lesser likelihood of potential hearing damage [30].

Button 3 is a straightforward button that functions the same way regardless of modes, serving to enable or disable any inputs from the piezo sensors. Pressing the button also controls LED 1, where it lights up red if the inputs are active and turn off if they are inactive. This addition is convenient to avoid any false triggers that may occur when the user is performing other tasks whilst wearing the glove.

Finally, button 4 serves to enable or disable the accelerometer sensor in MIDI mode, where LED 2 is turned on to indicate if the accelerometer is enabled and vice-versa. This addition is important in this mode as the accelerometer is constantly sending data and is best to be disabled while attempting to link the potentiometers, since many DAWs have a linking by movement option when mapping MIDI CC, meaning that the selected parameter will be linked to whatever controlling tool (slider, knob, button) is engaged with first. Since the accelerometer is constantly engaged any potentiometer linking will likely be overridden by it, hence the necessity to disable it. As the accelerometer is not in use under the Standalone mode, this button and the LED controlled by it are to be repurposed in this mode, to control a record/stop button to record the audio onto an SD card. This feature is not yet implemented but is further discussed in chapter 5.

2.3 Implementation

This section aims to elaborate on how the design was achieved from software and hardware standpoints, detailing how the glove, the MIDI and Standalone modes and the control panel were conceived, programmed, and assembled as well as which methods were used. Here, the wiring methods, which other components were used, the algorithms developed, and some discussion is held.

The section is segmented into first elaborating the main hardware components present in both the glove and control panel, explaining how they work and why they work for their purpose and how they were implemented on hardware, and later the two modes of operation - MIDI and Standalone - are elaborated upon, detailing how they were programmed. It is relevant to note here that these two modes are not implemented in conjunction, meaning that two separate Arduino sketches are used, and to change between modes, one has to upload the desired sketch. Similarly, it should also be noted that if only MIDI operations are desired, the Audio Board Adapter can be removed entirely as it is only required for Standalone mode.

2.3.1 Glove and Sensors

As previously discussed, piezoelectric PZT sensors were used to sense the finger tapping motion. These sensors when triggered generate voltage through pressure and vibration that can be recorded and measured, where more intense pressure is proportional to more output voltage. The sensors need to be connected to the Teensy board's analog pins, as then the signal can be accurately read, which is necessary for making the inputs velocity or gain sensitive. Otherwise connecting to digital pins, would only register as triggered or not triggered (logic 1 and 0), similar to a switch, and the signal would have to be debounced due to noise, which by definition requires some delay to be performed. Alternatively, a capacitor could be used to filter quick voltage changes, however velocity and gain sensitivity are a core function of the project, or a resistor to protect the input pin and set the minimum threshold to activate logic high.

The voltage spikes emitted by the sensor when triggered can be overwhelming to the Teensy boards which can only handle 3.3V and as such, the inputs need to be protected. To do so, two methods were attempted, one using a sole 3.3V Zener diode and another using resistors, the former was used for the first iteration and the latter for the latest one. The Zener diode in reverse bias mode across the output and ground wires of the piezo works to protect the input as a voltage regulator, capping the voltage emitted by the sensor at the value of the Zener diode, in this case at 3.3V, which meets the limit of the board. It is recommended to use this component in conjunction with a resistor as the resistor filters out the dropped voltage, however it was deemed unnecessary as the 1cm piezos used here outputted a weak amount of voltage and the diode was mainly used as a precaution. Using just a resistor was applied to the 2.5cm piezo sensors, and proven efficient in the MIDI Glove project. In their glove, resistors valued at 1MΩ set in parallel worked for their intended purpose, however, it is worth mentioning that they were not aiming at velocity sensitivity. The key intention of using the resistor here is not only to protect the input but also to analogically calibrate the sensitivity of the sensor. As such a series of resistors were tested in the same method as the MIDI Glove, using 1MΩ, 100kΩ, 56kΩ, 47kΩ and 22kΩ resistors. Using the 1MΩ resistor, unlike the results displayed on the MIDI Glove, ended up being too much for the input values to be registered which led to weaker resistors being used, as it was deduced that the sensors were not outputting strong enough voltage. Since the FLORA did not get damaged even when the voltage was regulated by such resistance, it was assumed that using lower resistances would be acceptable. Comparing the results from the four other trial resistors, it was shown that 56 and 47kΩ were the most suitable, bearing in mind that the sensors would be placed on the back of the fingers rather than being directly tapped on. The 100kΩ resistance still did not make the readings sensitive enough and the 22kΩ conversely made them too sensitive. From the two suitable ones, the 47kΩ resistor ended up being chosen as it would be preferred to have a slightly more sensitive input to account for weaker tapping. The testing procedure for all of the resistances was carried out the same way, using a breadboard, attaching a sensor to the desired placement at the back of the finger and connecting the resistance in parallel to the input and ground. Ideally, the sensors and resistors would have been soldered and attached to the glove to replicate the performing conditions however not enough sensors and gloves were available to do so. It should also be noted that the analog calibration had to be performed in collaboration with the digital calibration performed in the peak detection algorithm, mentioned further on.

Each sensor and resistor was then soldered onto a piece of stripboard and wired to the input and a common ground, along with the remaining sensors. The stripboard piece was kept protected using a segment of a heat-shrinking tube that was heated to fit the piece, keeping the connections protected. The sensors and the protected piece were then hot glued directly onto the glove to keep them stable. This is an important step as loose components tend to result in either triggering false inputs or not trigger a real input, as observed in the first iteration. The remaining wiring was then sewn onto the glove to ensure stability. As opposed to the first iteration where the diode was soldered directly to the wiring, the stripboard approach kept the connections protected and compact, and proved to be much safer and less troublesome, as poor soldering would cause the diode to disconnect, disconnecting the sensors along with it. The inputs chosen are the analog pins A16 to A20 on the Teensy 3.6 and were chosen due to the implementation of the Audio Adapter Board, which requires most analog pins to function and these are outside the range of pins used by this adapter. The connections to the board are displayed in figure 2.8.

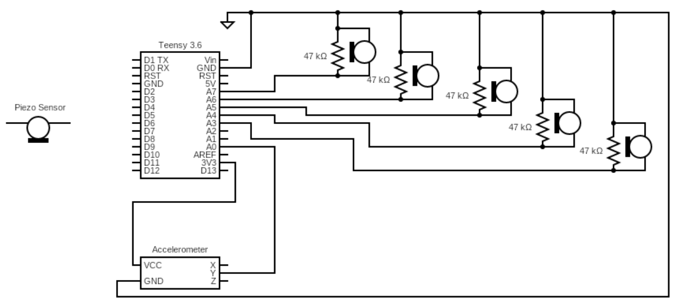

Figure 2.8: Circuit schematic of the connections of the sensors on the glove to the board.

With the inputs connected, an algorithm to detect finger taps had to be implemented, identifying voltage peaks when a tap occurred. The algorithm used was adapted from [36] and is a key part of the implementation, functioning to filter out noise, trigger an input and detecting the input value that can be mapped to the velocity and gain. The algorithm follows three states: idle, detection and aftershock states, elaborated below, and functions independently for each finger, allowing for simultaneous tapping. The entirety of this project was programmed using the Arduino IDE as Teensy has direct compatibility with it as well as complementary libraries, such as the USB MIDI library. Due to good familiarity with this software, no other options were considered.

- Idle State

  This state can be defined as the default state, ignoring any inputs if they are below a certain user-defined threshold, essentially denoising the data, preventing any false starts and background interference. A general threshold was initially used, applying the same threshold value for each finger tap, which has its drawbacks since it does not account for several external factors, namely solder and component quality as well as tap strength, as tapping with the index or the thumb arguably requires less physical strength than with the little finger to achieve the same peak. Hence, an individual threshold was set for each input, with a trial-and-error method until satisfactory results were achieved, making this the digital calibration. Digital calibration is an important step due to being able to account for these external issues to a certain extent, as broken connections or components cannot be corrected this way. This calibration mustn't set the threshold value too low as noise could be picked up, nor too high that it is not sensitive enough. Deciding on these values also depends on the input bit resolution which will be further explained in the MIDI mode and Standalone mode implementations. Once a tap is performed that exceeds the minimum threshold assigned to the respective sensor, the algorithm proceeds onto the detection state.

- Detection State

  This state occurs after a tap exceeds a minimum threshold, where the algorithm gathers the values acquired over a small fraction of time and accepts the highest reading. This fraction of time is also user-defined and much like the minimum threshold value, it should also be calibrated for each finger, keeping in mind that this fraction of time cannot be too large as it will cause delays on the input. The highest reading is then recorded and converted into the velocity or gain value, depending on the mode, and a trigger of the corresponding finger is sent. Upon detecting the peak, the algorithm shifts to the aftershock state.

- Aftershock State

  This final state determines the note duration in MIDI mode and is a final stage of denoising after the tap. As these sensors are vibration-sensitive, there is a chance for residual vibration, known as aftershocks, to be detected shortly after the finger tap occurred, so a timer Is placed, for which during that duration it ignores any value below the minimum threshold, being reset if the value is above the threshold. Much like in the detection state, this timer should be a small fraction of time to not impede taps performed in quick succession. The controlling of the MIDI note duration in this context is negligible, as the sounds are being sampled rather than synthesised, so whichever duration is used will not affect the sound. The calibration of this value should be coordinated with the calibration of the minimum threshold, as it could be argued that sensors with less placement stability could be more prone to these aftershocks since they are more likely to vibrate, so less stable sensors should have a longer timer to stabilise compared to more stable ones. Once this state is cleared, the algorithm reverts to the initial Idle state.

The algorithm is included within the *loop* function to be called upon *ad infinitum*. The algorithm functions independently for each finger, tracking these states for each individual input, so if the index is tapped, the algorithm performs the loop for the index only following the values assigned to their position on the arrays. If the reading from the index finger corresponds to the "third" analog pin, once there's a reading, the third values on each array from the calibration (being the minimum threshold value, the reading and aftershock times) as well as the MIDI note array are called upon. Triggering two or more fingers simultaneously causes no interference.

The last item on the glove is an ADXL335 3-axis accelerometer which is only being used in MIDI mode for MIDI CC modulation. An accelerometer is a tool that measures the proper acceleration from motion, converting the resulting mechanical energy into electrical energy which can be read analogically and processed. Although 3-axis are available to be used, only the Y-axis was used. The accelerometer is placed on the back of the hand facing opposite to the fingers, where along the chosen axis, by tilting the hand forwards or backwards the signal is obtained. This axis was chosen since the fingers can still be placed on a flat surface and performance can be multi-faceted, by finger drumming and modulation with a single hand. Along the X-axis it would imply moving the hand forwards and backwards which is an impractical movement, as is along the Z-axis by tilting the hand sideways, leading to an awkward positioning once the palm has to be facing outwards. Nonetheless, these axes could be implemented for free movement performance, where finger tapping is not desired, however, they were omitted as they did not fit within the scope of this project at the current stage.

The accelerometer data is constantly being tracked, which on one hand is practical as the readings are extremely precise, and on the other impractical as it is constantly being triggered within a DAW, making it difficult to automatically detect other MIDI CC controllers, as explained previously. The data requires calibration to then be mapped to MIDI values accurately, and more so to discover at what point the data rolls over, which can become an issue, as once it reaches the maximum value, it can roll over to the minimum and vice-versa, debilitating the controlling aspect of this sensor. To calibrate [14], the data obtained was simply observed at positions where the hand is as upright and facing downwards as much as possible, noticing when the data rolls over. A small amount is subtracted to the recorded values which are then constrained using the constrain function to clip data to these values, where any values exceeding either limit are capped to the limiting values. This is important to not make the user have to fully tilt their hand as it can become very uncomfortable over longer periods of time. On the other hand, once these values are mapped to the MIDI limits, they are technically perceived as more sensitive as the full motion is not being performed. An example of the output onto the DAW can be seen in figure 2.9.

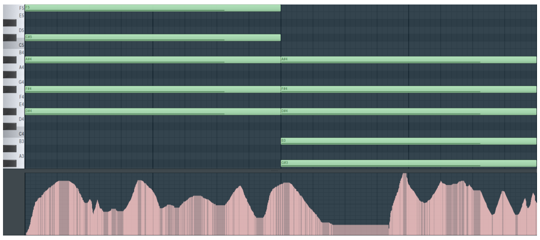

Figure 2.9: An example of how the accelerometer data can be used to modulate a parameter and be recorded in MIDI mode. Functioning like a real-time modulation "drawing tool", the volume here is being modulated denoted by the pink segments.

As evident in figure 2.8, the accelerometer was wired directly onto the board, to the necessary 3.3V pin as it requires an external power source to function, to the common ground along with the piezo sensors and the Y-axis to an analog pin, again, outside of the range of pins from the Audio Board. This sensor was covered in the same fabric as the glove and sewn directly onto the glove, to keep it as stable as possible, as any movement will generate undesired noise in the signal.

2.3.2 Control Panel

The control panel is a complementary tool created to boost the usability of the glove instrument, adding certain elements to interact with the sound in Standalone mode and with MIDI. The features have been highlighted in the previous section 2.2, and in this subsection, there is a focus on how they were implemented. The panel can be worn on the forearm or placed nearby as seen in 2.10 or outright omitted, as it is not a necessary feature for performance, even if it greatly improves it.

The current prototype for the panel has its components assembled on a sheet of stripboard, attached to a piece of cardboard using masking tape and has a Velcro strap attached to it so it can be worn. It is to be emphasised here that the panel shown is in an early prototype stage, shown for proof-of-concept motives, and in a future iteration will be further optimised with a 3D printed case. Although it is going to be replaced in the future, the cardboard sheet was used to protect the soldered side of the board and its interaction with a surface and the user's arm. Keeping the board on a surface without any cover will be guaranteed to damage the soldered joints, harming the panel, and direct contact between the board and the user's skin is uncomfortable as some soldered pins and joints stick outwards, which could scratch the user. Both of these

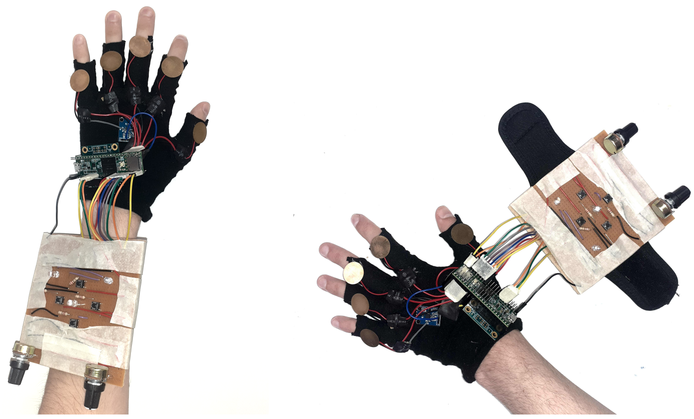

Figure 2.10: The glove instrument with the prototype control panel attached, showing how it can worn or placed nearby.

Following the diagram in 2.6, the component implementation is discussed. The figure 2.11 showcases the circuit diagram of the panel.

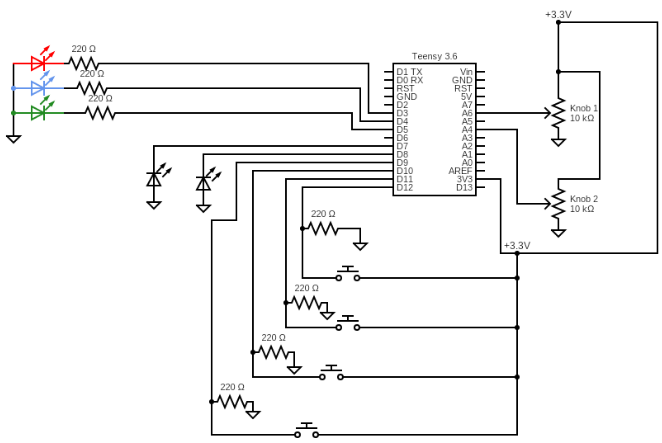

Figure 2.11: Control panel circuit diagram. The coloured diodes represent the RGB LED diode. All connections to ground are also connected to the board's ground pin.

Potentiometers are variable resistors, where the outermost pins are connected to ground and a voltage source, in this case, 3.3V from the Teensy board and the middle pin is connected to an analog pin in the board, which reads the amount of voltage applied to the pin. The resistance in these potentiometers is valued at 10kΩ. Using a higher voltage on this potentiometer causes it to reach the maximum value when being read prematurely, so if 5V were to be applied, the value read approximately halfway through turning the shaft would be the maximum and turning it further would stay that way. To implement these components onto Arduino, the pins connected are read using the analog read function. On the setup function, these pins are declared as "input pullup" pins, which is a necessary step. Declaring input pins this way prevents the occurrence of a floating pin, in which external conditions such as temperature or dust can interfere with the data reading if the wire is not connected, so this declaration uses internal resistors to help filter out these interferences. All input pins on the control panel were declared with the pullup resistor should the user decide not to use it to minimise the interference. The pots were implemented using the method in [2], where two variables are used, one for the current value and one for the previous value. The previous value initially takes the same value as the current value, and when a change occurs in which these two values are compared and differ, the previous value updates. This method prevents very small changes from occurring. To prevent any undesirable noise, the *ResponsiveAnalogRead* library [24] could be used, as it focuses on eliminating any noise from analog inputs whilst being extremely responsive, however it was not included as the potentiometers were not noticeably noisy to justify its use.

Push buttons work in a way that their state is only logic "high" when the buttons are pushed down, otherwise being logic "low". They follow a standard connection, where one of the rails is connected to voltage and the other is connected to ground with a 220Ω resistor on one side whilst the other connects to a digital pin.

Buttons 3 and 4 were implemented in such a way that the buttons act as switches, meaning that instead of the state reverting to logic "low" when the button is released, the state is instead toggled by storing the logic states and alternating between them whenever the button is pushed down. This switch effect is achieved using three variables - current state, current reading and previous reading. When declared, the current state is set to "high" and the previous reading set to "low".The current reading variable then takes the value from reading the corresponding digital pin and compares it with the previous reading value. If these two are different, the current state of the button is updated to its opposite value (if high before then it is updated to low), and the previous reading variable gets updated to the latest reading. The LED corresponding to each button is then turned off or on depending on the state of the button. These LEDs have their shorter leg, the cathode, connected to ground and the anode to a digital pin, and are defined as output pins in the setup. To implement the functionality of the buttons was quite straightforward: button 3 controls whether or not sensor inputs are read so by using an *if* statement inside the detection state of the algorithm, in which it only runs if the button state is high; and by using the same statement on the accelerometer in which the MIDI CC message is only sent if the state is high as well. It could be argued that using a simple two-state switch to perform this action would simplify this implementation as simply flicking it would simply switch states, however, buttons were used regardless due to being more in line with the desired design.

Buttons 1 and 2 follow a different implementation as they serve to control a counter. The variables declared to this implementation include two for the state of the individual buttons, one for the counter value (declared at value zero on startup), and another. When button 1 is pressed, and button 2 is not, the counter value adds one, and oppositely the counter value decreases 1. The RGB LED is associated with these two buttons, in which in MIDI mode the colour changes depending on the counter value and in Standalone mode the colour intensity increases or decrease depending on how much the counter value has changed. This LED has four pins, three corresponding to the colours red, green and blue respectively that are connected to digital pins and one cathode that is connected to ground. Just like the other LEDs, the connected pins are defined as outputs.

These processes are all included in their own separate function that is called upon in the loop function.

2.3.3 MIDI Mode

The MIDI Mode refers to the glove's mode of operation under the MIDI protocol, optimising the instrument to work directly with DAWs. This subsection focuses on how the project is optimised for MIDI functionality In this mode, the ADC resolution had to be changed from the standard 10-bit to 7-bit. Lower resolution means that the acquired analog data is digitally quantised in larger steps, meaning that some analog information is lost during capture. This effect is illustrated in figure 2.12.

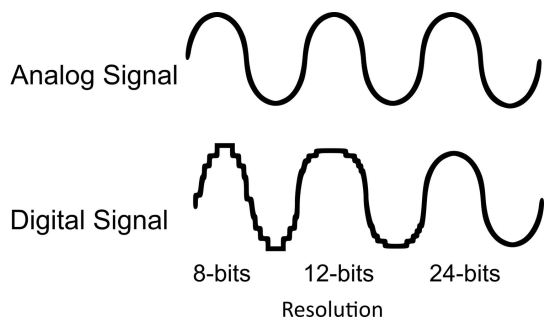

Figure 2.12: Illustration of analog data processed at different bit resolutions, adapted from [22].

This, however, is not intrinsically hindering in this mode as the MIDI protocol itself functions at 7-bit, so the acquired data can be directly reflected in the MIDI values without the need for mapping, as it would be required using any higher resolution, if considering the 10-bit format that ranges values from 0 to 1023, these values would have to be mapped to MIDI's standards. In MIDI mode the finger sensors are assigned MIDI note values, the accelerometer controls MIDI CC, as do the two potentiometers on the control panel and buttons 1 and 3 transposes the MIDI note values by five, up and down respectively. This mode has been mainly optimised to work with the FL Studio 12 DAW, whereby utilising the native FPC plugin, one can assign whichever samples they desire to whichever finger they prefer, assigning the pad trigger to the corresponding MIDI note played by the finger. This method is by far the easiest to work with this type of technology and has the advantage of letting the user have more creative freedom. The plugin's interface can be observed in figure 2.13.

Figure 2.13: The FPC plugin, the main bridge between the finger drumming and the DAW when in MIDI controller mode.

Assigning MIDI notes to each finger is fairly straightforward by creating an array of any values between 0 and 127, following the order of the analog pins reading the sensors corresponding to each finger. As previously explained, in this context, the values chosen can be arbitrary, however, percussive sounds are usually mapped between 21 and 108, so the values were chosen arbitrarily from 36 to 40, corresponding to the western notation of C3 to E3, from the little finger to the thumb so tapping the thumb will trigger MIDI note 40. These triggered notes are routed to MIDI channel 10, the usual channel for percussion sounds, however, any other channel can be used. To output these notes, in the detection state of the peak detection algorithm a function is used to call for MIDI "note on" which sends a MIDI note with the captured velocity to the chosen channel, and is terminated in the aftershock state using the MIDI "note off" function, having the length established in the aftershock calibration step. The MIDI velocity could be directly adapted from the value captured by the tap however, it was chosen to establish a minimum velocity value at 50, meaning that any value acknowledged exceeding the minimum threshold of the finger established in calibration, below this value will be set to this value. This value was chosen arbitrarily and serves also to make low readings more "audible" in the DAW. It could be argued that this step reduces the authenticity of the controller by not outputting the correct values however, having to set a minimum threshold to avoid noise could be a counterargument, as no values beneath that value will be acknowledged in MIDI. Since this value also varies between fingers, setting a value above the highest one was chosen, as otherwise a sensor with a higher value threshold than another, would never achieve lower values and this way it sets a level of equality is achieved.

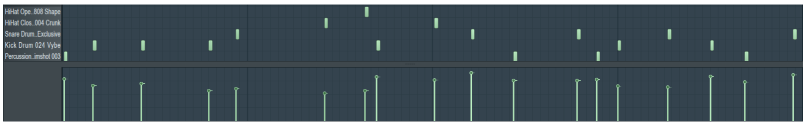

Figure 2.14: An example of how the drum output is presented in the FL Studio 12 piano roll. The velocity levels can also be observed.

The values of the potentiometers are continuously read as explained in subsection 2.3.2 similarly to the accelerometer while sending a MIDI CC note. For more creative freedom, the CC values chosen are classified as undefined [10], allowing for user-chosen parameters to be linked. To perform this, a "send MIDI CC" function is used, where the parameters are the analog input value, the CC value (9 and 14 in this case) and the desired MIDI channel.

Buttons 1 and 2 follow the counter procedure, however, in this mode, they function within a cycle, where the counter value can only vary between 2 and -2. To implement this, a constraint function was applied to the counter value variable to not exceed this range, however, there seemed to be an issue where the range values would be 1 unit above or the range limits (i.e. would range from 3 to -3), but this was easily circumvented by defining that if 3 was reached by the counter value, it would equal -2 instead and if -3 was reached, it would equal 2, therefore creating a cycle. Depending on the value of the counter, the RGB LED will display a different colour and the MIDI note array would be updated, being transposed either up or down by five, depending on the counter value (e.g. if the counter value is 1, it's transposed up by five; if the counter value is -2, it's transposed down by 10). An array of values all spaced between each other by five dubbed the "note lookup" is declared as a general array at the start, and within the peak detection algorithm, the MIDI note values set is an array of length five where each element is one of the of these "note lookup" values added by zero, one, two, three and four. The default value on the lookup is 36, making the values on the note array run from 36 to 40. Whenever the counter value is changed, the lookup note value is taken which consequently updates the MIDI note array. This is all implemented with a *switch case* statement, where if the counter value is a certain value, the note array is updated as described and the RGB LED is set to a certain colour. Magenta, cyan, dark blue, green and very light blue were the colours chosen, as they are all distinctive enough to not be confused with one another.

As previously mentioned, the toggle buttons 3 and 4 function in MIDI mode by simply not performing the MIDI message sending functions on the peak detection algorithm and on the accelerometer, if their states are low.

2.3.4 Standalone Mode

The Standalone mode refers to the mode of operation where the glove functions without the need of being connected to a computer and does not require any interfacing with a DAW, as everything is controlled on-site. The ADC resolution in this mode is the standard 10-bit, which mandatorily requires value mapping, and the Audio Adapter Board is required to function. The implementation of this mode, which was not present in the first iteration of the glove, stemmed from adding complete portability to the device, allowing users to make their own grooves wherever they desire. In this mode, the Teensy can be powered by an external battery outputting 5V, such as a portable charger, and portable wired headphones can be connected directly onto the Audio Board. The sounds assigned to each finger are royalty-free sampled percussion sounds, including a kick, snare and tom drums, a hi-hat and a gong. The sounds are sampled in 24-bit resolution WAV format for optimal quality and converted into readable code using the *wav2sketch* tool, converting the audio files into .h and .cpp files which are loaded when included within the Arduino code for this mode. The major drawback from this method is that loading other sounds is a laborious task as it requires conversion, naming and inclusion in the code instead of a straightforward method such as loading the audio files directly onto a microSD card, which is also possible, however it is warned by the Teensy team that very fast cards are recommended to trigger two sounds at once, as there can be latency between triggering and playing the sound. A note should be made that the code will only be able to include the sketch files converted from WAV if they are inside the same folder.

In this mode, a change from the MIDI mode is how most constants and variables are all set to be floating points rather than integers. The reason for this stems mainly from the range of values to control certain parameters like gain and effect dry/wet amount is between 0 and 1, meaning that the calculations necessary to map the tap velocity and the potentiometer in decimal format.

When programming for the Audio Board, the inputs and processing tools used need to follow the order of occurrence. This is easy to illustrate if thought of in the same way effect chain in a mixer works, where the effects added last are applied to everything that happens before it. This sequence should follow the order of inputs and audio sources first, followed by the audio processing used and lastly by the outputs. The order of connections is seen in figure 2.15. The progression follows inputting four sensors into the first 4-input mixer, combining the sounds, which is outputted onto the second mixer along with the remaining input. The effects are included in each mixer so that the input sounds can be processed with them, and later the parameters are included on the last mixer to affect everything that comes before it. It was decided to include the effects within the mixers since if only included in the last one, the effects would not be triggered in real-time, instead, the dry sound would play, followed by the processed audio with some delay. The last mixer then outputs into the board's DAC and headphone jack, where the DAC is necessary for sound to be outputted, converting the processed digital sound into analog.

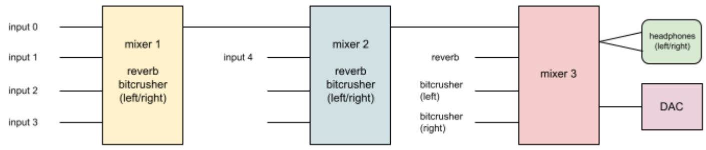

Figure 2.15: Diagram of the order of connections in Standalone mode. The inputs valued 0 to 4 correspond to the sensors placed on the fingers.

Sounds used require to be included within the Arduino sketch, as they are in sketch format as well. The peak detection algorithm in this mode is at its core the same, with the main difference that instead of calling a value within an array, a *switch case* statement is used for each finger, directly stating that if a certain sensor is triggered, a sound is played. To call each sound, a function needs to be in place for each individual sound, and these functions cannot be simply stored in an array as in the MIDI mode. This step is present inside the detection state of the algorithm, alongside the gain calculation, which is calculated by taking the value captured by the sensor in a 10-bit range, from the minimum threshold value defined for the finger to 1023 and proportionally mapped to the range of 0.1 to 1. Proportionally, the map does not range from 0 to 1, as it would be counterintuitive to trigger a sound that is triggered at threshold velocity to be essentially muted, so 0.1 was decided as the lower boundary. The mixer gain functions, where the gain of the sensor inputs is defined, is also included within the detection state of the algorithm.

The two potentiometers are implemented to control two effects separately, reverb and bitcrusher. It should be noted that these effects are not controlled dynamically, as they have multiple factors which are all controlled by the same knob. This comes with the drawback of not being able to adjust the parameters to one's liking fully but comes with the advantage of being able to use two effects simultaneously. Reverb was chosen due to how common the effect is present in all genres of music, and how well it works when paired with drum sounds [7]. The reverb effect being used is Teensy's own reverb. There are three parameters to consider when implementing this effect: room size, room damping and dry/wet value. Luckily, these do not have to be realised from scratch and have their own dedicated functions to call upon. These parameters can be summarised as room size controlling how much decay and space are present in the sound, and damping essentially controls the absorption of high frequencies in the reverb, where they are more dampened the more of this parameter is used, which could be seen as a warmer sound. The dry/wet volume just controls the amount of the effect present. The range of values much like the volume range between 0 and 1, so the assigned potentiometer analog readings are taken and divided by 1023 and defined as the amount of these parameters. The bitcrusher effect essentially functions by distorting the output signal by digitally downsampling it, reducing the bit rate and depth, larger quantisation steps are applied, resulting in a warm, quasi-clipping, heavily-distorted sound, following the same principle as seen in figure 2.12, but where the analog output is low-resolution due to the effect being applied on the DAC. The two parameters being included to modulate this effect are the bit resolution, defaulted at 16, and the sample rate, defaulted at 44.1kHz, however only the bit resolution is modulated at present, as the results are quite pleasant. The potentiometer value is processed the same way as in reverb, however, to calculate the downsampling, or to decrease bit resolution, to the initial value of 16 bits is subtracted the read value multiplied by 16, this way the more the knob is turned all the way around, the greater the downsampling. This effect needs to be defined for both left and right sides, and it was done so by simply declaring the mentioned process for each side.

Finally, the last control to be implemented is the volume control using buttons 1 and 2. The initial volume control is set to 0.5, being the halfway point. These buttons are again programmed to control a counter, where one button increases the counter value and the other decreases it. As such, the range of value that the counter can achieve range between -4 and 4, as the counter value is then divided by ten and added to the volume control value, which, since it's valued at 0.5, would mean the volume can be fully controlled. The RGB LED is implemented in a way that the values for the green and red diode are assigned to the counter value and its negative value respectively and simultaneously. This is so that if the counter value is positive, only the green pin is accounted for, as the LED cannot output negative values, which is what is being read for the red pin, and the same occurs when the counter value is negative, being turned into a value above zero for the red pin which can be outputted. This ensures that only one colour is outputted at a time, and its brightness increases, as the counter, reaches 5 or -5. If the volume is unchanged, the LED outputs no light. These two colours were chosen as green is usually associated with positive change and red with a negative change. For these changes to take place, the audio volume function was implemented in the loop function, so that it can be updated.

Chapter 3

Usability Testing & Evaluation

This chapter is dedicated to how the usability evaluation was performed, detailing the methodology, the process in which it was carried out and the results obtained, as well as a reflection on the gathered data. It is of urgent importance to note that the original intended usability testing method was very unfortunately not carried out due to reasons that were beyond control, most notably the COVID-19 pandemic during which this project was in development. This original method will still be detailed below along this chapter, alongside the benefits of having carried it out as intended, as there are still plans to put this method in practice in the near future. It is acknowledged that by not having executed this method, the usability evaluation of the glove controller is somewhat diminished, yet a new method was developed to still validate it. The chapter will be structured by first presenting potential usability evaluation methods in regard to literature, followed by displaying both evaluation strategies considered, the original one and the performed one, accompanied by details on why certain decisions were made. Afterwards, there will be a showcase of the results obtained from the test along with some discussion, followed by a comparison between both devised methods of usability testing.

It is of importance to reference that this evaluation procedure was performed in regard to the first iteration of the device, which led to the improvements presented in the latest iteration. Due to the same motives, an evaluation test was not performed for this latest iteration.

3.1 Evaluation Methodology

3.1.1 Usability Evaluation in Literature

This subsection aims at reflecting upon different usability evaluation procedures, showcasing their relevance to the topic at hand and discussed in which ways they will contribute to the direction of the usability tests that the prototype will be subjected to. An important point to be made here is that these methods were not considered for the applied method of evaluation that the first iteration was subjected to, as it is shown later that it is discussed later that it is not the optimal procedure of evaluation. The original method proposed in the next subsection 3.1.2 however takes some of these proposed evaluative methods into consideration, while being its own method.

The primary usability test tool that was considered is the SUS, standing for System Usability Scale [3], a proclaimed "quick and dirty" tool that focuses on presenting a set of 10 questions to the participants, each following the Likert five or seven-point scale of rating. The questions presented follow a specific pattern to adjust for the scoring system where each question values from 0 to 4, where odd numbered questions are scored based on the scale position subtracting 1, and even-numbered questions value at 5 subtracting the scale position number. The results are then added up and multiplied by 2.5 to reach a maximum score of 100. This allows for placement of negative and positive tone questions throughout the questionnaire, where odd numbered questions would focus on positive tone questions like "this device is very functional" and even numbered questions would have a negative connotation such as "this device is unwieldy". By using two tones evenly throughout a questionnaire, allows for less biased results, since the audience has to give an opinion in potential negative aspects as well. This scale is extremely well regarded, being used in a multitude of fields for how simple and effective it is.

When considering usability testing in this project, its important to look into intuitiveness of use. The participants are intended at some point during the test to perform certain tasks. Whereas the proposed tasks mentioned in the next subsection 3.1.2 mainly focus on how the wearable device's ergonomics and responsiveness, it would be interesting to provide some challenges after being provided with instructions and a cheat sheet to see how easy it is for a new user to grasp the basic interaction mechanics. A method proposed by [9], to test the intuitiveness of use of a new mobile phone, involved tasking the participants in the test with a series of six tasks of increasing difficulty and to evaluate how and if they completed them with ease. Perhaps applying this method and timing the tasks (up to a certain limit) could be beneficial, as if users spend less time getting the device to function, it should indicate good usability due to intuitive work.

Whereas important to define usability based on participant responses, an important factor to account for when designing a device that is meant to be extensively interacted with is the time spent interacting with it as noted in [28]. By providing the participants with a "free time" period in which they are allowed to interact with the device for as long as they want until a certain time limit, and also being notified that they can stop whenever they want, should provide two relevant aspects in evaluating usability: pleasure and good ergonomics. Pleasure can be measured this way as the participants are given a free choice to leave the test, yet they choose to keep playing with the glove. Comparatively, it is assumed that if the glove is not comfortable enough, the participants would not subject themselves to stay for long on the test time. Ultimately other factors come into play, and these can all be accounted for in, perhaps, a short post-interaction interview.

3.1.2 Original Method

The original method would follow a set of three usability tests, focusing on ergonomics, reliability, and performance. These tests would be done at different points in time during the development stage to help influence its progression and assess where improvements could be made and where focus should be placed before carrying out any further developing in some other aspect. These usability tests are further elaborated below, indicating their relevance to the evaluation of the project.

- Ergonomics Test

  This test focuses mainly on testing various glove textiles to settle on the final material being used. To gauge which material would be best fit, this test would involve presenting an array of gloves to an audience of participants. The test would then require the participants to wear each glove individually and perform various movements (e.g. tapping on a surface, clenching their fist, tilting their hand, etc.) whilst wearing it for a maximum period of time of five minutes, being allowed to remove the glove at any point. This time limit is necessary to not extend the test for too long, which could result in boredom for the participants and could perhaps lead them to rush through the test. No more than ten subjects would be required for this type of test, however the more comparative data, the better the deliberation. After testing each glove, they would be requested to rate them individually from most to least comfortable, as well as providing a short, written feedback for each one. Each participant on the test would be compensated with food and beverages. Providing compensation to participants serves to both entice more people to participate in the test and to make the test more worthwhile of their time.

  One of the main concerns arising from this test, especially during the COVID-19 pandemic, would be hygiene, as providing the same glove to different people would not be a very safe procedure, risking potential infections. To circumvent this issue, it was considered to provide hand sanitiser gel to every user, before and after wearing each glove, but this could still be unsafe enough to pose a threat. Another option would be to provide medical (nitrile) gloves to each user, renewing them between trials. To enforce further safety procedures, the testers would be asked to sanitise their hands and the nitrile gloves before and after every trial. This however could pose some bias against each material, as it does not accurately represent the real feel of the textile against the skin.

It must be stated that this test would also involve some monetary investment, not reimbursed by the university. This could prove to be a challenge since glove damaging could occur and would need replacement, increasing the costs. This was not the main reason for the non-realisation of the test but still a concern, nonetheless.

Between both iterations, two textiles ended up being tested by the researcher, as mentioned in **??**design), with the cotton glove being used in the latest iteration. This data, although still relevant, does not have enough comparative data to be reliable in the long run.

- Reliability Test

  This step would focus on how well the glove would fare against wear and tear caused by continuous usage. This part of the usability evaluation ended up morphing into a recurring process by experimenting with the glove. What is meant by this is that the usage by the researcher ended up serving as a way to measure the sturdiness and allowed for calibration.

  Shifting between both iterations caused a significantly noticeable difference in reliability, as the first iteration, due to the concerns already mentioned in section 2.1 was quite prone to damage. Instead the latest version of the glove mainly succeeded at surpassing these issues, with the implementation of a new design. Nonetheless, performing an actual reliability test where the glove and control panel are subjected to more unfavourable conditions, such as carrying them in a bag unprotected, extensive use and even misuse would provide more relevant data to improve the design. By subjecting it to these conditions repeatedly, one would be able to pinpoint where more common failures would happen and devise new ways of implementation to minimise these issues.

- Performance Test

  The final test to be implemented, would involve presenting the prototype to a series of participants for them to use, and could be considered the true usability test. The participants would be requested to wear the device and use the control panel, in both modes, and perform a series of tasks, followed by some free use period where the participants use the prototype as they desire, as well as a questionnaire to report on their experience. Prior to the test, they would be provided with instructions on how to operate the prototype and a cheat sheet of commands that would be present during the entirety of the testing. Doing so, allows for the user to learn as they perform, and in a real scenario, users are not expected to intuitively know all the commands from the beginning, however it is expected that the instructions provided are clear enough that intuitive interaction is encouraged. This test aims at covering various aspects of evaluation of a NIME project as seen in [4], primarily the usability, the user experience (UX), the engagement and the aesthetic appeal to participants. In MIDI mode, the tasks would require the participants to make use of the drumming capabilities of the glove, requested to perform some drum loops over pre-recorded samples at different tempos with different sounds. Prior to each task, they would be previewed the sounds for the task and asked which finger they would want associated to which sound, which would then be recorded over a minute. This could provide an interesting insight into which fingers are preferred for finger drumming, by correlating the frequency of taps for each finger. This section would also provide insight on how easily it is to finger drum directly on a surface and if the research aim of natural interaction is fulfilled, by evaluating how easily the participants perform. Following this step, the participants would be requested to use the accelerometer functions by assigning the MIDI CC to a certain effect of their choosing from a predetermined set - frequency filtering, reverb, and distortion - to a pre-recorded synth loop: first asked to just use the hand freely to modulate the effect followed by being requested to use both the accelerometer with the drumming functionalities. During this phase, the participants are allowed to change between the effects at any point.

Next, the users will be encouraged to interact with the control panel, where the features are to be explained in a simplified manner. Doing so detracts focus from the technical implementation and stimulates focus on the usability.

Finally, participants are encouraged to freely utilise the prototype, with the assistance of the researcher to ensure that the glove is used correctly. Users are given a time limit of a maximum five minutes. During this phase, their interactions are to be observed and noted down, and the time taken during this activity is to be timed. The idea behind this stems from an assumption that a longer free play time could correlate with better usability of the prototype as it would determine that the device is enjoyable to interact with.

Similarly the Standalone mode would be tested by following the free use method, as the functionalities do not require mapping since they are hard coded in the prototype. A difference would be that the users will be provided with a portable battery to run the device in a portable fashion and will be more encouraged to tap on diverse surfaces. By noting down what kind of surfaces participants choose to tap on (e.g. legs, tables, chairs) and noting that the participants are satisfied with the performance will further prove the usability of the prototype in the sense of portability, reinforcing its use outside a recording zone.

Afterwards the participants are asked to complete a questionnaire detailing their experience with the glove, with answers following the Likert scale system (i.e. how much the agree or disagree with a statement). Boilerplate questions about identifiers are to be used (i.e age, gender) as well as some questions about their experience with music production, such as if they are involved in music production, if they use MIDI controllers and drum samplers in their productions. The questionnaire then focuses on usability with questions focusing on how responsive the actions were, if the product performed as expected, if there were any problems during the tests, if the usage were easy to understand and set up, and if they would use it as a tool in their productions (if applicable). The final section is then dedicated to questions regarding the user's experience and feedback for both modes: if the glove was comfortable, if the controls were well positioned, easy to use and understand, and satisfying in both glove and the control panel, if the individual designs of glove and control panel and the overall design are pleasant and concluding with a section for written open feedback.

This test would be carried out with a test pool of at least ten participants and should not take longer than 15 minutes. The device in MIDI mode will be programmed with the latest iteration of its Arduino sketch, connected to the researcher's laptop running FL Studio 12. Every drum sound will be setup using the FPC native plugin (an MPC emulator of sorts), the pre-recorded samples are also to be made using the native plugin 3xOsc, a simple three oscillator synthesiser. To test the accelerometer functionality, it was linked to the wet slider on the *Fruity Reeverb 2* VST (a plugin for reverb), the frequency value knob on *Fruity Free Filter* VST (for frequency cutoff) and the "dry/wet" knob for the *CamelCrusher* VST which controls the amount of distortion. The control panel's knobs are to be assigned to the other two effects not being used by the accelerometer.

To ensure the minimal amount of latency, a *Focusrite Scarlett 2i2* audio interface is to be used, for which the respective driver has the low latency of only 2ms. With this setting, the participants are also provided with a set of open-back studio headphones, namely the *Beyerdynamic DT 990 PRO*.

3.1.3 Applied Method

As previously mentioned, an alternate method for evaluation had to be devised in order to validate this project. As such, a new method was carried out with a digital format. The main concerns stemming from this method was how it would not provide a completely accurate representation of how the project functions, nor a proper explanation on how to set it up, however, it had to be performed, nonetheless. This usability test is heavily based on the aforementioned Performance Test, replicated, and transformed into shorter video demo presentations, in which participants are required to answer a series of questions regarding the design, the finger drumming capabilities and the accelerometer modulation feature upon watching each demo video, followed by a more open feedback section upon watching a short demo video regarding these aspects with a brief explanation. These sections are presented below, on how they were conducted, how the demo videos were structured and the relevance of each question in the scope of this project. A transcription of the questionnaire, links to the videos and the answers received is provided in A.

It is reiterated here that this evaluation method was performed in regard to the first iteration of the device, where only MIDI capabilities were included. There is no evaluation data for the Standalone mode or the control panel as of yet. The feedback gathered was considered and then implemented in the design of the latest iteration.

For the showcase of the prototype, the performances were made with the FL Studio 12 digital audio workstation, using the *ASIO4ALL* sound driver as it provides minimal latency (2ms) with a speaker connected via direct AUX connection. Although it was mentioned that a *Focusrite Scarlett 2i2* sound card was to be used, this was not fulfilled as there were no cables available at the moment to fit the sound card, however, did not present any issue as both sound drivers appear to have the same latency. The demo videos were shot using a Samsung Galaxy A20e smartphone.

- Introduction

  The first part of the questionnaire aims at introducing the participants to the project and evaluation, giving a brief description of what the project is capable of doing and what is being presented along the questionnaire (i.e. that they are going to be shown three videos and will be prompted to answer some questions about what was observed). Participants are also made aware that the expected duration of the questionnaire should fall between five and ten minutes.

  Participants are then asked to disclose their age and gender for identifying purposes, if they tap their fingers to the beat of the music they listen to, if they have any background in beat-making and/or music production and if they do, if they use drum samplers to do it. These allow to analyse the user demographic, since the project is primarily aimed at users that fit the latter criteria.

- Design

  Participants are then informed of the first video to be shown which focuses on showcasing the design. In this short video, a brief showing of the controller and the features is done, showing where each component is placed and their functionality: explaining that there are piezo contact microphones at the fingertips to register finger taps, and the accelerometer on the back of the hand which allows for modulation by tilting.

The questions that follow focus on how the design is perceived by the participants, following the Likert scale. These questions regard: how appealing, comfortable, and complicated the design appeared to be, as well as if the components seemed to be in the right place. This verbiage was seen better fit rather than a more direct one (i.e. "is the design comfortable?") as in the scenario the project is presented, participants can only observe. By prompting these questions, a better understanding of what can be improved in future iterations can be obtained.

- Finger Drumming

  This section of the questionnaire prompts the participants to watch a demo video of how the controller can be used for finger drumming. The controller is used with the aforementioned FPC plugin. In the video, it is explained that each finger is associated to a drum sound, presented individually for credibility. It is required to state that prior to the recording, the piezo contact microphone located in the middle finger had stopped working and was disclosed in the demo. following this brief explanation, a short performance is done in a single take, without any processing, as an attempt to truly show how the prototype is currently functioning, emulating the feeling of it being used by the participants, hence some connection issues can arise. This decision was made to promote authenticity in the demonstration and to reflect the participant's opinion more accurately on the prototype.

  Once this video has been watched, participants are then presented with another set of questions focusing on what was just demonstrated, being questioned if the finger tapping concept seemed interesting, if they would use the controller for their own musical endeavours (both in a production and live performance standpoints), if they would use the controller as a replacement over a drum sampler, if the controls seemed responsive and finally they were asked to select which fingers they would most likely use whilst performing with the controller. All questions but the last also follow the Likert scale whereas the last one just follows a multiple choice scheme. These questions mainly help gauge out if the prototype fits the main goal of the project of creating a usable musical tool for production. By asking which fingers they would most likely use for performance, a better understanding of the usability can be derived and considered for a future implementation by, for example, introducing a hardware feature that allows the user to be able to select which fingers to use in their performance.

- Accelerometer Modulation

  This section aims at demonstrating how the glove can also be used as a modulator by tilting the hand along the y-axis. Participants are required to watch a short presentation video where the accelerometer is used as a modulator, displaying how the movement performance works. In the demonstration, a simple synth pad loop is used to exemplify the performance, where the modulator is linked to the frequency cutoff of this pad sound. While tilting the hand up and down, the effect modulation can be observed. In the video, it is explained that other effects can be linked to this component and how the movement affects the modulation, and it is also stated that it can be used in conjunction with other controllers. A single effect was used in the demo so as to keep the performance short.

  Upon watching the video, participants are once again asked a set of questions regarding this feature. How they felt about the importance of this feature and if it would be beneficial for live performance and to be used alongside other controllers were the questions that could validate the inclusion of a modulator in this prototype. As before, these questions still followed the Likert scale.

- Feedback

  The final section of the questionnaire takes a more open-ended approach, offering a space for participants to provide their own thoughts about the prototype in their own words. For the questions asked here, a short, written answer is expected, abandoning the Likert scale format. The participants are asked if they would add and/or change any aspect in the prototype's design, what other features they think could be implemented, a short reflection on how they felt about the prototype overall and are kindly requested to provide some additional comments as seen fit, however is not expected of the participants to provide any thorough insight.

  The inclusion of this section can be perceived as highly beneficial for consideration when developing the next iteration of the prototype, as it can provide a clearer focus on what features to focus on more.

3.2 Evaluation Results and Discussion

3.2.1 Data Analysis

The data from the questionnaires sent out was recorded from 14 participants, aged between 21 and 48 years of age, with a mean (μ) of 25.7 and standard deviation (σ)

of 6.62, where 12 participants identified with the male gender and two identified with the female gender. All 14 participants responded affirmatively to whether or not they tapped their fingers to the beat of the music they listened to. Eight of the participants claimed they had backgrounds in beat-making and music production but only three of them stated that they made use of drum samplers. This type of data leaves us with a somewhat diverse sample pool, albeit not the largest for extensive testing, where both musicians with and without experience with drum samplers and non-musicians responded to the questionnaire. This allows for a wider array of analysis by observing if there is any correlation within their group data.

Regarding the design of the glove, participants seemed to display a mixed reception towards it, with six participants having a neutral reaction to the design, while three others seemed to react more positively to the design and three others more negatively, however when in regard to component placement, there was a general consensus that they were correctly placed. When it came to the ergonomics of the design, a mixed reception was also observed, however positively leaning. The last question, on whether the design appeared complicated had the most scattered responses, as observed in 3.1. From the later feedback section, some comments touched on this topic by claiming the design did not seem ergonomic enough and even "overwhelming" by the exposed wiring.

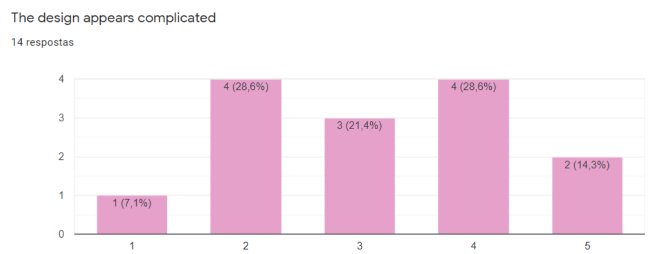

Figure 3.1: The responses to whether the design of the glove was complicated.

Since the main target audience for this type of product is musicians with backgrounds using drum samplers, their feedback will be considered slightly above others in regard to the finger drumming functionality. While it is a clear bias, it should be noted that adapting this type of product to fit the target demographic is not a far-fetched decision but, in any case, any and all feedback is considered and not discriminated.

The concept of a MIDI controller based on finger drumming was highly regarded as an interesting one, with the majority (nine participants) strongly agreeing with the statement. A single participant seemed to be neutral towards this statement. When asked if the participants would utilise the prototype as a tool for their musical endeavours, the response was generally favourable, with unfortunately one participant strongly disagreeing with this statement, however upon being asked if this controller would be used over a drum sampler tool, the responses were more skewed to the middle. On a brighter note, it seemed that the participants with drum sampler backgrounds seemed to lean more towards the positive side of responses. When prompted about the usage of the finger drumming techniques feature for live performances, half of the participants were strongly inclined to use it. When questioned about which fingers were preferred for finger tapping, the majority votes lied on the use of the thumb and the middle finger (both with ten picks), followed by the index finger (with seven picks). This was found interesting as during development, it was expected that the index and middle fingers would be most popular, as they seemed most dexterous to do so, followed by the thumb. The little finger was the least popular finger for this task as only three participants picked it. A graph elaborating on these results can be seen below on figure 3.2.

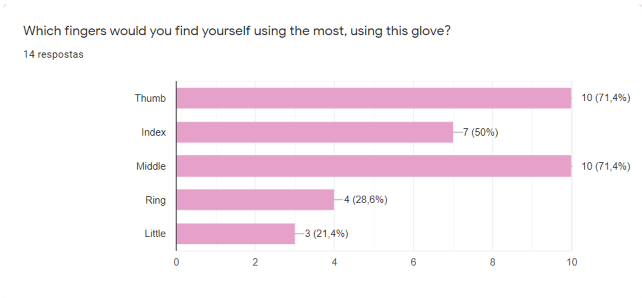

Figure 3.2: The participants' preferred finger for drum tapping.

Generally, regarding the accelerometer, most participants felt that it is a key feature and a very promising tool, especially for live performances. They also felt that this feature would be quite interesting for complementary performance alongside another controller. The responses to the questions posed in this section were majorly positive.

When in regard to the feedback section of this test, it was noted that a considerable number of comments were laid upon the design, with requests to make it "less robotic" looking, more "open" and to have less visible wiring. These requests were expected when presenting the demonstrations as they paralleled the developer's concerns. Some participants mentioned that wireless connection would be a beneficial feature and while it was considered at some point in development, it was later scrapped due to fear of latency. Other comments mentioned that the inclusion of visual feedback through lighting would be a helpful feature. Other mentioned features suggested the inclusion of a way to load samples directly onto the glove without the use of a DAW, different controls for the fingers and a way to turn off the glove controls (which is being included in the next iteration).

Overall, it has been observed from this test that while the concept of this controller is very promising, there is still some polishing to do, regarding the design mainly and the responsiveness of the components. Some suggestions even included the use of the glove for DJ performances by linking it to a controller. The overall response to the project was quite positive.

3.2.2 Method Comparison

In this subsection, a comparison between both methods is done, highlighting the advantages and disadvantages of each one, and providing some reflection on the choices that were followed through, as well as some additional reasoning for why different methods were devised. Here when the term *original method* is utilised, it should be noted that it is in reference to the original method Performance test detailed in 3.1.2.

It should go without saying that if the original method were followed through, it would make for a much more optimal testing option, allowing for participants to actually feel the glove and to control it rather than just observing a demo showcase as could be observed by the findings in [29] where hands-on usage could lead to a more proper evaluation. This more proper evaluation does not necessarily imply more satisfactory leaning results, as during testing users could have a lesser optimal experience for a myriad of reasons (e.g. component detachment, poor responsiveness, random triggers or even lack of functionality). Nevertheless, even with a less satisfactory experience it would still more accurately inform the level of usability of the prototype.

Whereas there is a clear inclination against the applied evaluation method, as it does not cover as many topics of evaluation seen in [4] as well, most notably the usability and engagement where they cannot be measured in any way, it cannot be denied that it does have its own advantages. The benefit of providing an online method of evaluation lays in the fact that a wider array of people can be reached, from different backgrounds and experiences with such devices, which can allow for a more diverse testing option albeit less accurate in the usability sense. An online testing method also allows for a safer, more convenient evaluation option, given the current pandemic situation.

Between both methods, it would still seem that a fairly accurate attempt at recreating the original evaluation through a different mean within the realm of possibility. Since there are no participants testing out the actual prototype, several testing pointers were rendered useless, such as how comfortable the glove is or how the performance actually feels like. Similarly, by doing a physical, perhaps a lower number of participants would be reached which would leave the evaluation with a smaller test pool.

Upon performing the actual evaluation, it has been noticed that truly the optimal method of evaluation would be a combination of both the original test and the performed test. The performed test could have been used mid-development to gauge out what to focus on further and what other features to implement before moving on to the final usability test.

Chapter 4

Conclusion

Upon thorough reflection on the proposed project, it can be said confidently that this thesis' goal has been successfully achieved, having been able to create an ergonomic wearable device that is both MIDI compatible and functional as its own entity in a portable manner, providing musicians with an affordable, powerful new tool to interact with their craft. Having implemented these two separate modes opens up many more possibilities and opportunities for further development as well as establishing new implementation methods for future devices, such as emphasising the use of compact piezoelectric sensors for tapping interaction, using accelerometers to modulate effects with freehand motion and reinforce the use of the Teensy line of boards for wearable devices as well as strengthening the belief that more musical devices should be multi-functional.

The research and literature review presented along this paper showcases and elaborates on new and alternative possibilities for the same results to be achieved, presenting their strengths and drawbacks in an understandable manner, deliberating on their potential usability within the scope of the project and giving possible implementation methods for these items. This deliberation is done in such a way that it provides a clear overview of the potential of these items in a way that can serve as a baseline for future endeavours. The methodology used for the actual design and implementation approaches utilised justifies their usability, having been proved to be functional. Displaying the differences between two versions of the same device also serves to illustrate how and why certain approaches are viable or not, further contributing to the literature.

Throughout the development process, some challenges were met and for the larger part were overcome. Reflecting upon the evaluation methods used, it is clear that for a wearable device, the timing of its creation was less than optimal, as attempting to evaluate the usability with an audience physically amidst a global pandemic could cause health risks that were not willing to be taken. As such the performed evaluation method served to attempt to circumvent this unavoidable issue, and whereas it is not the optimal method of evaluation, it still provided much relevant information as to how to evolve the project between the first iteration and the latest one being showcased. It would not be an incorrect assumption that the positive feedback already received at the time of the evaluation of the first iteration would be strengthened further once the latest iteration is presented, as it took into consideration several points of the feedback, such as the improved slicker design for instance. The original evaluation method will eventually be carried out once it is safe to do so and taken into consideration for further development. Amidst these challenges, hardware failure was prevalent throughout development, due to wire straining and component damaging, especially during the development stages of the first iteration which caused time constraints that impaired improvements. These challenges were eventually diminished upon implementation of the fingerless design, which facilitated progression.

In conclusion, this thesis successfully brings forth a new format of musical interaction for musicians, with several new possibilities to be explored in future iterations. Some of which are highlighted in the following chapter 5, with deliberation on how to be implemented and why these next steps are important to achieve a complete design.

Chapter 5

Future Work

Improvements to the research and new and revised features are planned to be added in future iterations of the glove instrument, as it feels like there are even more possibilities to be explored and other ways to interact with the instrument.

First and foremost, the main crucial improvement that can be made would be executing the originally proposed usability evaluation method, conducting a physical test where participants would get to actually play and interact with the instrument in its entirety. Doing so would help validate the creation and usage of the project and also receive adequate feedback which would help further improve the current design. Another test that would be interesting to carry out would be testing how the glove is compatible with other commercially available DAWs such as Ableton Live or Cubase, as testing was only carried out on FL Studio. This would be interesting to observe if the glove in MIDI mode can be used with these DAWs easily, which may improve its usability if so for being an easily transferable tool between these programs since some musicians enjoy using multiple ones in their craft. In addition, the next step forward includes merging both modes, removing the necessity to upload two separate sketches. This makes compatibility much easier as there is no need to upload the Standalone sketch beforehand.

The PVDF sensors are also to be tested and if the performance results obtained are more favourable than the current iteration, realised in the next iteration due to the benefits outlined in section 2.1. Testing of other sensors will also be realised if possible, which will allow for better data comparison, however, the PVDF film sensors are the more desirable ones currently. In the next iteration, the Teensy board will be updated to its latest version which at the time of writing is the Teensy 4.1, due to its benefits over the model currently in use.

In a future revision, a 3D printed casing will be manufactured in which to keep the control panel, not only keeping it protected but also improving the overall design quality drastically. The casing also allows for adequate labels to be placed over the panel for an easier and more intuitive understanding of how to interact with it. Regarding the control panel, some new and revised features are planned to be added. A linear potentiometer, or slider, is planned to be implemented for both modes and is intended to control the volume in Standalone mode, allowing the buttons currently filling that role to be repurposed, and in MIDI mode to function as another MIDI CC controller. This would be easily implemented as the principle is the same as the currently implemented potentiometers and proves to be more intuitive as a volume controller in Standalone mode as it would be reminiscent of a fader knob, commonly used in mixer boards to adjust the volume. Not only that but it would allow the user to choose the volume they start with even before they use the instrument. The control panel is also planned to be reworked into other shapes and dimensions until an optimal one is achieved, due to the current version not feeling as practical as it could be.

Adding SD card compatibility will be included to allow for a few possible new features in Standalone mode. One of these features is to replace the current sample loading format with loading samples directly from the SD card. By doing so, the user would be able to load their own sample banks to map onto each finger which is now standardised with the current state-of-the-art and removes the current laborious process of sample loading. In conjunction with the inclusion of the volume slider which frees up two buttons, the same process used for MIDI note transposition in MIDI mode can be applied where the user cycles through a set of custom sound banks. Another concept to be explored would be the possibility of introducing custom audio samples that can be loaded through SD card and looped where the user can improvise drumming patterns over, simulating even further the process of beat-making. The ability to record the drum grooves played by the user in this mode directly onto the SD card will be implemented. By making use of the unused button and LED in this mode, it would be possible to include this feature using the same process already programmed, by starting a recording when the button is toggled on, lighting up the LED as a visual signal, and stopping the recording once the button is toggled off. This would prove beneficial as some musicians tend to have ideas or concepts for songs that can be lost over the day if they are away from a recording environment and being a portable instrument, this feature could aid in jump-starting the creative process on the go, recording those ideas for later use.

Inclusion of more effects such as delay and chorus, that can be easily interchangeable would be a good addition to promote creative expression, allowing the user to not just be limited to the two included effects and create their own effect chains. Similarly, the ability to control the effects more dynamically, rather than simply the amount of dry/wet as it is presented in the current version, would complement this promotion, such as controlling the reverb room size and dampness individually for example. These ideas are to be included in some future version, although these are not a priority at the current time.

Visual feedback for the sensors will also be included in a future iteration, and it has already been conceptualised. Making use of RGB LEDs, it would be possible to assign a colour representing each finger, and the strength of the tap is calculated into each value of the red, green, and blue parameters of the diode, where stronger taps will result in brighter feedback. The LEDs could be fitted into clear, thin, flexible PVC tubing that extends from the components onto the fingers, being lit up whenever a tap occurs. Introducing this visual feedback could be a beneficial addition not only due to being aesthetically pleasing but, also due to this type of feedback is complementary with audio feedback [13], as evidenced in several market available products such as the Novation Launchpad Mk2 where each drum pad can be assigned a unique colour that flashes when tapped. To complete this concept, including an on/off toggle would also be a welcome addition, to account for users that do not desire this type of feedback.

Eventually, turning the glove into a fully wireless instrument will be realised. Although Teensy boards do not have native Bluetooth compatibility, there have been attempts by the Teensy community to implement it using USB connected Bluetooth dongles, as the 3.6 and 4.1 versions of the board have USB Host support, or another method would be to connect an ESP32 board via Serial line to the Teensy as they support both Bluetooth and Wi-Fi communication. The step to wireless connectivity will be the final step to achieving a fully portable and ergonomic device.

Bibliography

[1] Asheesh Kr. Sharma Arpit Jain Rashmi P.N Anjana Jain Prashanth K. J. “Dielectric and Piezoelectric Properties of PVDF/PZT Composites: A Review”. In: *Society of Plastics Engineers* (May 2015), p. 2. DOI: \url{https://doi.org/10.1002/pen.24088}.

[2] *Arduino Potentiometer*. https://www.arduino.cc/en/tutorial/potentiometer. Accessed: 2020-10-04.

[3] John Brooke. “SUS - a quick and dirty usability scale”. In: (Jan. 1996), pp. 189–194.

[4] Dominic Brown, Chris Nash, and Tom Mitchell. “A User Experience Review of Music Interaction Evaluations”. In: May 2017.

[5] Zachary Diaz. “Analysis of Sampling Techniques by J Dilla in Donuts”. In: *Electronic Theses and Dissertations* 197 (Aug. 2018), pp. 5–11. DOI: \url{https://scholarworks.sfasu.edu/etds/197}.

[6] Steven Dourmashkin. *Specdrums: Music at Your Fingertips*. https://www.kickstarter.com/projects/364756202/specdrums-music-at-your-fingertips. Accessed: 2021-03-21.

[7] Peter Doyle. “Echo and Reverb: Fabricating Space in Popular Music Recording, 1900–1960”. In: *[Oxford University Press, Society for Music Theory]* 32.1 (May 2010), pp. 89–92. DOI: \url{https://doi.org/10.1525/mts.2010.32.1.89}.

[8] Drumpants. *DrumPants*. http://www.drumpants.com/#how-it-works. Accessed: 2021-05-05.

[9] Henry Duh, Gerald Tan, and Vivian Chen. “Usability evaluation for mobile device: A comparison of laboratory and field tests”. In: vol. 159. Jan. 2006, pp. 181–186. DOI: 10.1145/1152215.1152254.

[10] Nick Fever. *MIDI CC List*. https://nickfever.com/music/midi-cc-list. Accessed: 2020-12-06.

[11] *Glover*. https://mimugloves.com/#glover-feature. Accessed: 2020-09-28.

[12] Moinul Haq. "Application of piezo transducers in biomedical science for health monitoring and energy harvesting problems". In: *Civil Engineering Department, Aligarh Muslim University* (June 2018). DOI: \url{https://doi.org/10.1088/2053-1591/aaefb8}.

[13] Sergi Jordà. "Interactive Music Systems for Everyone: Exploring Visual Feedback as a Way for Creating More Intuitive, Efficient and Learnable Instruments". In: *Stockholm Music Acoustics Conference* 03 (Aug. 2003), p. 3. DOI: \url{https://doi.org/10.1016/0041-624X(91)90091-L}.

[14] Arduino Learning. *ADXL335 Example*. http://arduinolearning.com/code/arduino-adxl335-accelerometer-example.php. Accessed: 2020-12-06.

[15] Peter Manning. "From Analog to Digital: The Evolution of MIDI Hardware". In: Oct. 2011.

[16] Andreas Fender Christian Holz Manuel Meier Paul Streli. "TapID: Rapid Touch Interaction in Virtual Reality using Wearable Sensing". In: *IEEE Virtual Reality and 3D User Interfaces* (May 2021), pp. 519–528. DOI: \url{https://doi.org/10.1109/VR50410.2021.00076}.

[17] *MI.MU Glove*. https://mimugloves.com/#key-features. Accessed: 2020-09-28.

[18] Rupa Mitra. "Effect of diameter-to-thickness ratio of crystal disks on the vibrational characteristics of ultrasonic ceramic transducers". In: *Applied Acoustics* 48 (May 1998), pp. 1–13. DOI: \url{https://doi.org/10.1016/0003-682X(95)00038-B}.

[19] P.s. Nicholson N.D. Patel. "Influence of effective diameter and piezoelectric and dielectric properties on the performance of ultrasonic transducers". In: *Ultrasonics 1991* 29 (Feb. 1991), p. 4. DOI: \url{https://doi.org/10.1016/0041-624X(91)90091-L}.

[20] Koka Nikoladze. *Koka's Pocket Beatbox is a pocket-sized drum — a portable multisampler designed by Koka*. https://www.nikoladze.one/. Accessed: 2021-04-11.

[21] John Park. *CLUE BLE MIDI Glove*. https://learn.adafruit.com/clue-midi-glove. Accessed: 2020-12-15.

[22] ProtoCentral. *What does 24-bit ADC resolution mean anyway?* https://www.crowdsupply.com/protocentral/sensything/updates/what-is-24-bit-acquisition-anyway. Accessed: 2021-05-18.

[23] B.H. Repp. "Sensorimotor synchronization: A review of the tapping literature". In: Dec. 2005. DOI: \url{https://doi.org/10.3758/BF03206433}.

[24] *ResponsiveAnalogRead*. https://github.com/dxinteractive/ResponsiveAnalogRead. Accessed: 2021-04-23.

[25] X. Zhu L. Dai S. Liu L. Chen and Y. Xin. "Digital piano keyboard based on PVDF piezoelectric film". In: *International Conference on Audio, Language and Image Processing* (Nov. 2010). DOI: \url{https://doi.org/10.1109/ICALIP.2010.5684555}.

[26] Dovid Schick. *TAP*. https://www.tapwithus.com/how-tap-works/. Accessed: 2021-04-16.

[27] J.G. Schloss and J. Chang. *Making beats: The art of sample-based hip-hop*. Jan. 2014, pp. 1–254.

[28] Jane Siegel and Malcolm Bauer. "A Field Usability Evaluation of a Wearable System". In: *Wearable Computers* (Nov. 1997), p. 2. DOI: \url{https://doi.org/10.1109/ISWC.1997.629914}.

[29] David Simm. "Experiential learning: assessing process and product". In: *Planet* 15 (Dec. 2015), pp. 16–19. DOI: \url{https://doi.org/10.11120/plan.2005.00150016}.

[30] Kim Kähäri Stephen E. Widén Claes Möller. "Headphone listening habits, hearing thresholds and listening levels in Swedish adolescents with severe to profound HL and adolescents with normal hearing". In: *INTERNATIONAL JOURNAL OF AUDIOLOGY* 57 (Apr. 2018), p. 733. DOI: \url{https://doi.org/10.1080/14992027.2018.1461938}.

[31] Becky Stern. *Getting Started with FLORA*. https://learn.adafruit.com/getting-started-with-flora. Accessed: 2020-12-01.

[32] Becky Stern. *MIDI Drum Glove*. https://learn.adafruit.com/midi-drum-glove. Accessed: 2020-12-01.

[33] *Teenage Engineering Pocket Operator PO-33*. https://teenage.engineering/products/po/metal. Accessed: 2020-12-15.

[34] Teensy. *Audio Adaptor Boards*. https://www.pjrc.com/store/teensy3_audio.html. Accessed: 2021-03-21.

[35] Teensy. *Teensy Technical Specifications*. https://www.pjrc.com/teensy/techspecs.html. Accessed: 2020-12-01.

[36] *Teensy Piezo Test Code - PJRC Forums*. https://forum.pjrc.com/threads/48308-test-code-for-piezo. Accessed: 2020-10-12.

Appendix A

Appendix A name

In this appendix the questionnaire performed for the evaluation is presented, as well as the answers obtained (for the open feedback questions).

A.1 Questionnaire

This is the questionnaire information and questions presented to participants. All questions follow the Likert scale unless otherwise specified.

Description:

A MIDI Controller Glove was developed as part of the Master's thesis final project. The glove allows for a user to connect with music by creating drum grooves with their fingertips and to modulate any effect by moving their hand. This questionnaire serves to evaluate the project in a feedback manner. You will be prompted to watch three short videos detailing the design, finger drumming and modulation aspects of the glove and will be required to answer some short questions regarding what was demonstrated. The questionnaire should take approximately 5-10 minutes to complete. Thank you for participating!

- Descriptors

  Age? (Short answer)

  Gender? (Male, Female, Prefer not to say, Other)

  Do you find yourself tapping your fingers to the beat of the music you listen to? (Yes/No)

  Do you have any background in beat-making and/or music production? (Yes/No)

  If yes, do you use drum samplers (e.g. Roland SP-404, Pocket Operators, MPC, etc.) (Yes/No)

- Section 1 - Design Showcase

In this section you will be shown a video succintly describing the controller and you will be asked some questions about the glove, design-wise.

The design appears to be appealing

The components seem to be in the right place for their purpose

The design appears comfortable

The design appears complicated

- Section 2 - Finger Drumming

  In this section you will be asked to comment on the performance seen in the following video, regarding the finger drumming capabilities of the glove.

  I find the finger drumming concept interesting

  I would use this glove for beat-making and music production purposes

  I would use this controller over a drum sampler

  I would use the drumming features for live performances

  The drumming controls appear responsive

  Which fingers would you find yourself using the most, using this glove? (Multiple selection)

- Section 3 - Accelerometer Modulation

  In this section you will be asked to comment on the performance seen in the following video, regarding the modulation done by tilting the hand up or down, by means of an accelerometer.

  I consider this an important feature of the glove

  I would use this feature in a live performance

  I would use this feature alongside other controllers

- Section 4 - Feedback

  In this section you will be asked to give some short feedback on the project

  What would you add and/or change in the design? (Short answer)

  What other features would you like to see implemented? (Short answer)

  What did you think of the project overall? (Short answer)

  Do you have any further comments? (Short answer)

A.2 Questionnaire Feedback Responses

The answers obtained for the open-ended questions regarding feedback of the glove are presented here unchanged.

A.2.1 What would you add and/or change in the design?

Less visible wiring

A more open design so the sweat is not an issue

Less robotic design

Sleeker more contained design, less exposed wires, make it look more beginner friendly

I like the design and the functionalities. Maybe I would hide a little bit the cables

Perhaps a better design/more subtle

I would add wireless midi connection, for better performance

The design is good, I'm looking forward to see the next implementation of the glove

Nothing

I would improve the visual appeal

The design seems great. It is a glove ;-) I don't know if maybe it is too heavy or fragile, but in the video looks great,

Just make it as ergonomic as possible

For now just the appearance and smoothness of the glove

I would add sensors on the side of the fingers, so one could use it with tilted hands

A.2.2 What other features would you like to see implemented?

Lighting

The distance sensing between fingers or the palm and the table for an additional control

Upload and play samples; bluetooth connection

N/a seems very well put together from a feature perspective

I really liked all the features

I dont know

I think it's fine as it is

Maybe it would be fun to be able to load samples directly to the glove, without the need of a DAW

I don't know

Any

I think there are enough sensors right now. Maybe in a second iteration it could be interesting to add new features, but ritght now it is great

Nothing

Maybe when you close your hand the music would stop or having diferent features like only having one finger up, the Accelerometer would do something else.

I would like to be able to disable single fingers' sensors when not needed

A.2.3 What did you think of the project overall?

Genius concept

Looks cool and promising , but it seems to need some more work to be great

Certo creative and innovative

Very interesting concept, could be a massive leap into music makers for beginners aswell as a useful tool for professionals

I found really interesting, it's a thing I'd like to use

Unique and looks very useful

It can be very usefull and more expressive in some cases, rather than using a pad controller

I liked it! good job!

Cool

It seems super interesting and a great idea

I want to test it!!!!

Very interesting and fun, if done correctly it can actually be practical and useful in live performances/beat making

I think its interesting and visualize a DJ for example using the accelerometer feature

It's really interesting, I want to try it in person

A.2.4 Do you have any further comments?

Love this

Who dont like egg pls leave

No

Nope.

More egg

No

Good luck bro

No

Not now (desculpa bro, acabei de acordar, espero que isto ainda esteja a tempo)

No

Great work!

No

Good luck

No