# Portable audio sampler · Embedded electronics and audio exploration

Jakub Krnac • krnac.jakub@gmail.com

## 1. Introduction

This project investigates the contributions of embedded electronics to the interaction design of auditory-based artefacts, with a primary focus on high-fidelity prototyping. By utilising commonly available and affordable development platforms such as Daisy Seed, which are slowly becoming a standard for audio applications, the project examines how these technologies can facilitate the creation of functional, interactive, and intuitive designs. The aim was to investigate the materiality of embedded systems and their potential to support innovative workflows in the context of auditory interactions.

As a designer, my aim was to gain understanding of the ways embedded electronics influence the prototyping process, balancing technical functionality with user experience. By working with high-fidelity prototyping, the project also questions whether alternative methods such as  hybrid prototyping approaches might provide comparable benefits, particularly in terms of efficiency.

This exploration highlights the values that embedded electronics bring to interaction design, emphasising their ability to enable designers to iterate and refine complex auditory-based prototypes. By questioning the reliance on embedded systems, the project seeks to uncover whether these technologies truly represent the optimal path forward or if alternative tools might offer new opportunities for innovation in auditory interface design.

## 2. Background

Audio sampling has become an essential tool in modern music production. Samplers allow manipulation and playback of pre-recorded sounds or audio clips. They enable artists to chop, loop, pitch shift, and layer samples, crafting unique soundscapes and beats. Additionally, samplers make it easy to tweak and alter projects quickly, allowing for faster adjustments and creative freedom compared to working with live musicians, where any changes would require additional recording sessions (McGuire, 2008). Furthermore, samplers allow producers to experiment with sounds and styles that might be difficult or impossible to achieve with traditional instruments. This opens up new creative possibilities and has contributed to the development of new genres and musical styles. Overall, audio sampling has revolutionised music production, making it more accessible and flexible for everyone involved, from seasoned professionals to budding amateurs.

Embedded electronics are crucial in modern interaction design, enabling designers to prototype and test new user interfaces effectively. Initially, the use of embedded electronics was challenging, requiring detailed manual coding and complex hardware setups. This often limited their accessibility and application in early design stages. Today, tools like Arduino and Raspberry Pi have revolutionised this process by making it easier and more affordable to create proof-of-concept prototypes. These platforms provide a user-friendly environment for experimenting with various hardware and software, facilitating rapid prototyping and iterative design (Electromaker, 2024). Additionally, embedded electronics are increasingly being used to work with audio, expanding their application in interaction design. Platforms like Daisy Seed have been developed specifically to support audio processing, enabling designers to integrate audio capabilities into their prototypes. These platforms often offer high quality audio processing and synthesis capabilities, making it easier for designers to experiment with sound in their projects. This integration of audio opens up new possibilities for creating more immersive and interactive experiences, further showcasing the versatility and power of embedded electronics in modern design practices.

This project works within the context of portable samplers that support control using a MIDI controller of choice as this was the area of my interest. The reason is that existing samplers usually feature at least one instrumental interface such as piano keys or drum pads (McGuire, 2008, p. 7), which I believe might be an unwanted feature compromising the portability of the sampler. By focusing on portable samplers that can be controlled via MIDI, the design can remain compact and lightweight, ideal for musicians who need to travel frequently or work in various locations. This approach also allows for greater flexibility, as users can choose their preferred MIDI controller, whether it be a keyboard, pad controller, or another device. This not only enhances portability but also personalises the user experience, making the sampler more adaptable to different workflows and preferences. Additionally, the use of MIDI controllers can provide more precise control over the sampler's functions, improving the overall efficiency and creativity in music production.

## 3. State of the art

The project started by looking at different samplers that were available on the market. Observations of typical usages such as sample recording, sample manipulation, audio effects, sequencing and performance were done mainly from video reviews and product manuals. The primary area of focus was on the first two steps of sampling. These are sample sourcing and sample manipulation. Sample sourcing can be done in two ways, either recording the audio directly or by importing an already prerecorded samples (McGuire, 2008, p. 12). From the observations, all samplers featured either or both of these methods and the process was generally similar across all of them. Sample manipulation / editing refers to a process of editing sample length and its start point, essentially trimming the sample to remove or include parts of the sample that the artist is indenting on using. This might be useful if the original audio sample includes multiple sounds that could be used separately (McGuire, 2008, p.12) or if the sample contains an unwanted noise. While sample manipulation differed across the observed samplers, I categorised them into two categories, this was based on the fact whether their interface allowed for real time manipulation - meaning the sample start and length parameters could be adjusted during performance or not (Table 1.). At the time, I was not aware how does this ability affect the experience, I was only looking at it from a technical point of view.

| **Sampler model** | **Supports sample recording** | **Supports MIDI playback** | **Supports real time trimming** |
|---|---|---|---|
| KORG Volca 2 | No | Yes | Yes |
| teenage engineering PO-33 K.O! | Yes | No | Yes |
| Elektron Model:Samples | No | Yes | Yes |
| KORG electribe sampler | No | No | No |
| Roland SPD-X | Yes | No | No |
| Polyend Tracker | Yes | Yes | No |
| 1010music Blackbox | Yes | Yes | - |
| Roland Verselab MV-1 | Yes | Yes | No |
| Elektron Digitakt | Yes | Yes | - |

*Table 1. Observed samplers*

Simultaneously, I started to  plan the prototyping process. Given my interest in exploring the material of embedded electronics, I searched for a suitable development platform that would meet the specific needs of my project. Working with audio in real time presents unique challenges, one of which is the necessity to store the audio in RAM (Random Access Memory) due to its quicker access speed compared to flash memory. Another crucial aspect to consider is the operating frequency of the microcontroller. Since the standard audio sample rate is 44.1 kHz (Adobe, n.d.), the operating frequency of the microcontroller should be higher than this rate to avoid any potential latency issues. After thorough research, I identified three main development platforms that seemed promising for my needs, as detailed in Table 2.

| **Development board** | **Operating frequency** | **RAM size** | **Maximal theoretical audio stored(48 khz)** |
|---|---|---|---|
| Raspberry Pi Pico | 133 MHz | 264 kb | 1.375 seconds |
| Electrosmith Daisy Seed | 480 MHz | 64 MB | 333.333 seconds |
| Teensy 4.1 | 600 MHz | 1024 kb | 5.333 seconds |
| Teensy 4.1 custom board | 600 MHz | 16 MB | 83.333 seconds |

*Table 2. Considered prototyping platforms*

Although it would be technically possible to create a sampler using all of the development boards above, I choose to explore Daisy Seed by Electrosmith. Not only does it provide the largest RAM memory but it is a development platform made specifically for audio applications (Electrosmith, n.d.), meaning it already contains necessary code for audio sampling at specific frequencies, necessary documentation and has a large community of people working with it.

As for the interface design, Calegario (2019) proposed a new concept in DMI (Digital Music Instruments) design called "instrumental inheritance", which I intended to utilise. The toolkit named Probatio (Figure 1.) offers a physical, modular system for prototyping DMIs, aiming to reduce the time for creating functional prototypes and promoting a rapid exploration of design ideas by having a pre-made modules of interfaces such as buttons, knobs, sensors, etc.

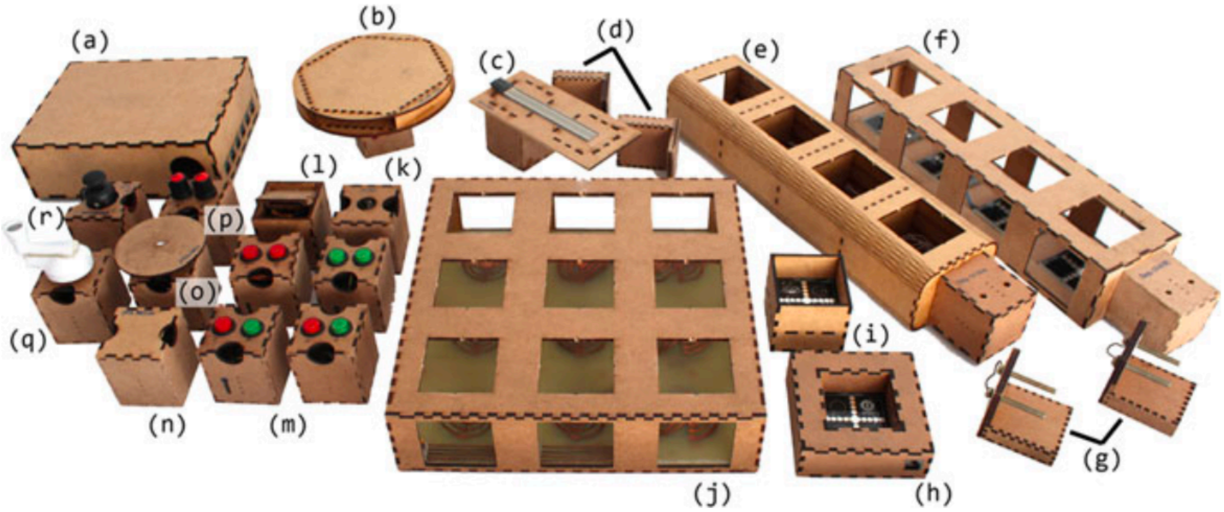


*Figure 1. Probatio toolkit (Calegario, 2019)*

**4. Methods**

This section lists the various methods and processes employed, detailing how each contributed to the development and refinement of the project. These methods ensured a comprehensive exploration of both technical and design challenges, guiding the project from research to material exploration and prototyping as can be seen on the design process timeline (Figure 2.).

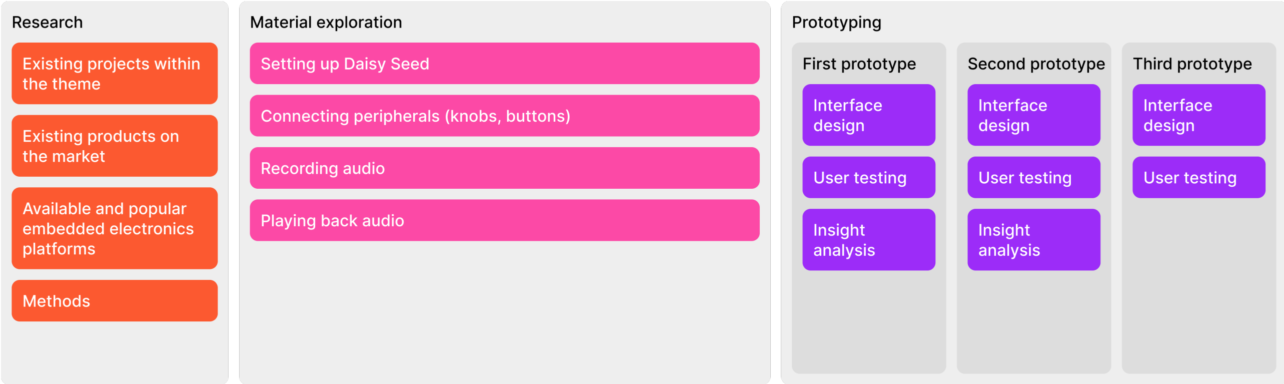


*Figure 2. Design process timeline*

### 4.1 Literature review

A literature review was conducted to establish a foundation for the portable audio sampler, synthesising relevant projects, research, and existing products in the field of audio sampling and embedded interaction design. Following the approach outlined in *Universal Methods of Design* (Martin & Hanington, 2012), the review identified trends, effective design strategies, and gaps in current solutions.

The review included an analysis of two existing sampler prototypes: Lee's *Chopping Board* and Argo's *Slidepipe*. These projects provided valuable insights into novel physical interfaces for sample manipulation, such as touch-sensitive surfaces and adjustable sliders, highlighting the potential of unconventional designs. Additionally, product documentation and academic research on real-time sample manipulation, portability, and interface design were studied. This informed key technical and interaction decisions for the project.

By organising findings thematically - covering areas such as audio processing, unconventional interfaces, and embedded systems - the review ensured the project was grounded in both practical and theoretical knowledge.

### 4.2 Observations

Observations played an important role in the early stages of the project, providing a systematic approach to studying the features and workflows of existing audio samplers. By analyzing user demonstrations, product manuals, and video reviews, I was able to gain valuable insights into current designs. As outlined in *Universal Methods of Design* (Martin & Hanington, 2012), observation was employed to closely examine interactions with products and environments, focusing on key features such as real-time sample manipulation, MIDI playback, and audio effects. While primarily informal, some structured components, such as a comparative checklist of sampler features (see Table 1.), were incorporated to ensure systematic analysis across products.

This approach aligns with the principles in *Universal Methods of Design*, which emphasise the importance of careful documentation and synthesis in observation to uncover meaningful themes or patterns in user behavior and product functionality. The insights gained through this method directly informed the design direction and priorities for the project.

### 4.3 Material exploration

Material exploration was a critical part of this project, enabling a deep understanding of the capabilities and limitations of the Daisy Seed platform as well as other interface components that were used. Following the *Material Driven Design (MDD)* method (Karana et al., 2015), this process emphasised hands-on experimentation to evaluate how the hardware and software could be integrated to achieve a balance between technical functionality and user experience.

The Daisy Seed platform played a key role in the creation of high-fidelity, fully functional prototypes, supporting real-time audio processing and interaction features like sample trimming and MIDI playback. However, it is worth questioning whether alternative materials or platforms could have been explored, as different hardware or software ecosystems might have offered similar opportunities or advantages in building the prototype.

### 4.4 Evaluative research and usability testing

Evaluative research and usability testing were integral methods used throughout the development of the portable audio sampler, ensuring the design evolved based on real user feedback and interaction. In line with *Universal Methods of Design* (Martin & Hanington, 2012), these methods focused on assessing both usability and user satisfaction through task-based scenarios and iterative refinement.

User testing sessions were designed around specific tasks, such as trimming audio samples and manipulating playback parameters, to evaluate how effectively the interface supported typical workflows. Observers collected both performance measures (e.g., task completion times, accuracy) and preference measures (e.g., user satisfaction, ease of use), combining objective and subjective feedback for a holistic evaluation.

This iterative process ensured that usability challenges, such as difficulties toggling between sample start and length adjustments, were identified and addressed in subsequent design iterations. By blending evaluative research with usability testing, the project ensured that the final design was both functional and intuitive, aligning with user needs and expectations.

### 4.5 Interviews

Interviews conducted during user testing followed a semi-structured format, allowing for both guided questions and open-ended responses. This approach, as outlined in *Universal Methods of Design* (Martin & Hanington, 2012), enabled the collection of specific feedback on tasks such as trimming audio samples and using rotary knobs while leaving room for participants to share broader observations and experiences.

Key questions were designed to probe usability and ergonomic challenges, such as the difficulty in switching between sample start and length adjustments. At the same time, the flexible format encouraged participants to express preferences and frustrations freely. The insights gained from these interviews informed subsequent iterations of the design, ensuring it addressed both functional and user-centered needs.

### 4.6 Bodystorming

Bodystorming, although not explicitly utilised as a formal method, emerged naturally during the testing and observation phases, particularly in evaluating the ergonomics of the first prototype. This method, as described in *Universal Methods of Design* (Martin & Hanington, 2012), combines role-playing and simulation to explore user interactions in a physical context, offering valuable insights into how users physically engage with a design.

One notable insight from this informal bodystorming occurred when a user highlighted the difficulty of pressing down and turning the rotary knob simultaneously. This challenge was especially pronounced because users typically operate the sampler with one hand while using the other for a MIDI controller, such as a piano keyboard. This feedback underscored the need for ergonomic refinements in the design, emphasising the importance of aligning interaction mechanics with natural user behaviors and physical constraints.

## 5. Prototyping

Max, developed by Cycling '74, and Pure Data, an open-source visual programming language, are widely used for interactive multimedia projects. Max offers user-friendly interface, extensive documentation, and integration with Ableton Live, making it a popular choice for artists and musicians who want an environment for many different audio projects (Cycling '74, 2022). On the other hand, Pure Data is flexible and works well on multiple platforms, including Linux and embedded systems like Raspberry Pi and Daisy Seed.

On the other hand, DaisyDuino and libDaisy, software libraries for the Daisy development platform, are specifically tailored for embedded systems, offering significant advantages in real-time audio processing and efficient memory management. DaisyDuino simplifies development through the Arduino IDE, making it accessible to those familiar with Arduino. This integration enables easy connection with various sensors and peripherals, facilitating the creation of interactive audio projects (Electro-Smith, 2023). LibDaisy, combined with DaisySP, provides more control and comprehensive documentation for experienced C++ developers, supporting more complex audio processing tasks. It allows for direct storage and manipulation of audio samples in memory, as well as the option to program various user flows and interactions, which might be challenging to achieve with Max or Pure Data. Overall, while Max and Pure Data are excellent for general audio applications on standard computing platforms, DaisyDuino and libDaisy are better suited for embedded audio projects like this one. Having experimented with both DaisyDuino and libDaisy, I found that while DaisyDuino offered an easy and quick setup, the lack of documentation compared to libDaisy led me to choose libDaisy for this project due to its greater detail and support.

### 5.1 Audio processing with Daisy Seed

Daisy Seed offers flexibility and robust support for audio processing. Its ability to handle audio input and output in real time is enabled through features like adjustable audio block sizes and sample rates, which are critical for minimising latency and maintaining high audio fidelity. Using its built-in functionality, the platform can process audio at a sample rate of 48 kHz, with the audio callback function being triggered multiple times per second to handle incoming and outgoing audio buffers. This allows for efficient real-time audio recording, manipulation, and playback. Additionally, Daisy Seed supports traditional components such as buttons and rotary encoders through an extensive code library, simplifying their integration into interactive prototypes. Its straightforward API and hardware configuration options make it accessible for both novice and advanced developers working on audio-focused projects. For more technical details, see the report appendix.

### 5.2 Building an interface

The next step was to create an interface for the sampler. Both Lee (2006) and Argo (2004) experiment with unconventional physical interfaces for sample manipulation and playback. Lee's Chopping Board (Figure 3.) prototype works with a single sample at a time, spreading it across a touch-sensitive, ruler-like pad. The prototype allows the user to then trigger different parts of the sample by placing their finger anywhere on the surface. This interaction provides a highly intuitive and tangible way to manipulate sound, enabling users to engage with the sample in real time. The author believes that skill development is essential for effectively using new musical interfaces such as the Chopping Board. They highlight that although the device is designed for intuitive use, developing a skill and refined control requires practice and experience. The prototype is crafted to be accessible for beginners while offering advanced users detailed control.

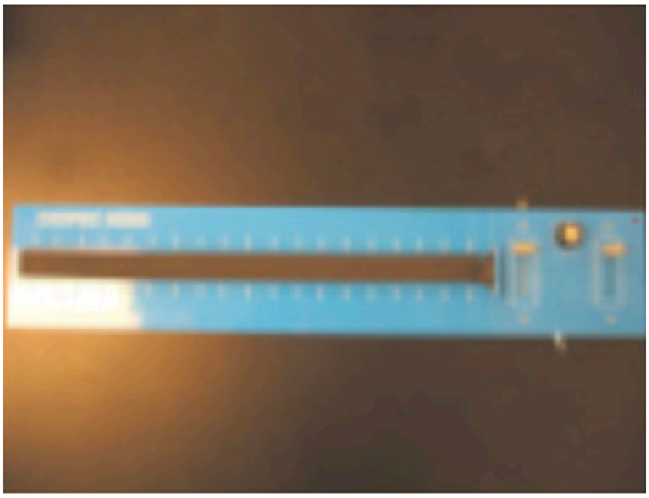

*Figure 3. Chopping Board (Lee, 2006)*

On the other hand, Argo presents a more static approach, where a pipe is utilised to represent the entire audio sample (Figure 4.). Two paddles slide along this pipe, marking the start and end points of the final sample. This physical interface allows performers to visually and physically engage with their material, providing an intuitive method to manipulate audio.

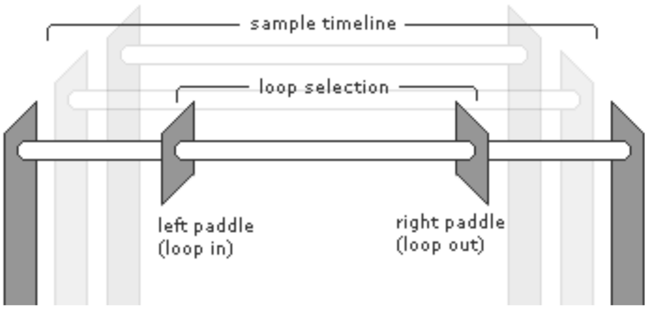


*Figure 4. The Slidepipe (Argo, 2004)*

I believe that both of these prototypes present audio samples in a physical and understandable manner, enhancing the users' ability to feel the material they are working with. At the same time, both of these setups were considerably complex, utilising multiple electronic components along with software for both the computer and the microcontroller. Instead of this, I chose to start with components that I had access to and begin from there. I managed to source a few rotary encoders with integrated switches as well as a few standard buttons, which provided a simpler yet effective starting point for my interface design.

A significant portion of the available time was dedicated learning and understanding the Daisy Seed platform. Various aspects such as audio input and output signals, memory management, MIDI messages, pitch shifting, and rotary encoders had to be understood, tested, and implemented before even starting the design of the interface itself. An ideating session with my supervisor followed, mapping the possible interactions achievable using rotary encoders. The objective of the interaction was to trim the audio sample and play it back using an external MIDI device. During a speculative discussion, an idea of single knob real-time trimming emerged. In theory, a single rotary encoder with a built-in switch could be used to trim both the start and end of the sample. This could be done in two ways, either by using the built-in switch as a toggle between sample start and sample length handle or by using the switch to access sample start handle by default and sample length handle while pressing down. To provide further and more understandable feedback, the knob would be accompanied by an LED indicator that would inform the user whether they are currently editing the sample start or the sample length parameter. As for the playback, the prototype was able to receive MIDI signals via the built-in USB port and pitch shift the recorded sample to a desired note.

### 5.3 First prototype and user testing

The initial prototype employed a single rotary encoder that could toggle between two parameters: sample start and sample end. While technically functional as a first prototype, I decided to prepare two interaction modes for comparison. In the first mode, users could switch between the two parameters by clicking the integrated button in the knob and then adjust the selected parameter by rotating the knob. In the second mode, users controlled the sample start by simply turning the knob, but adjusting the sample end required pressing the knob while rotating it.

To determine whether the proposed interactions would perform well, they had to be tested. Three individual testing sessions were carried out with three participants, ensuring a varied set of user experiences and interactions. Each user was first asked if they were familiar with the concept of audio sampling, and those who were not were given a comprehensive introduction to ensure they understood the task at hand. The task involved trimming a piano C3 note sample to remove any silent areas before attempting to play it.

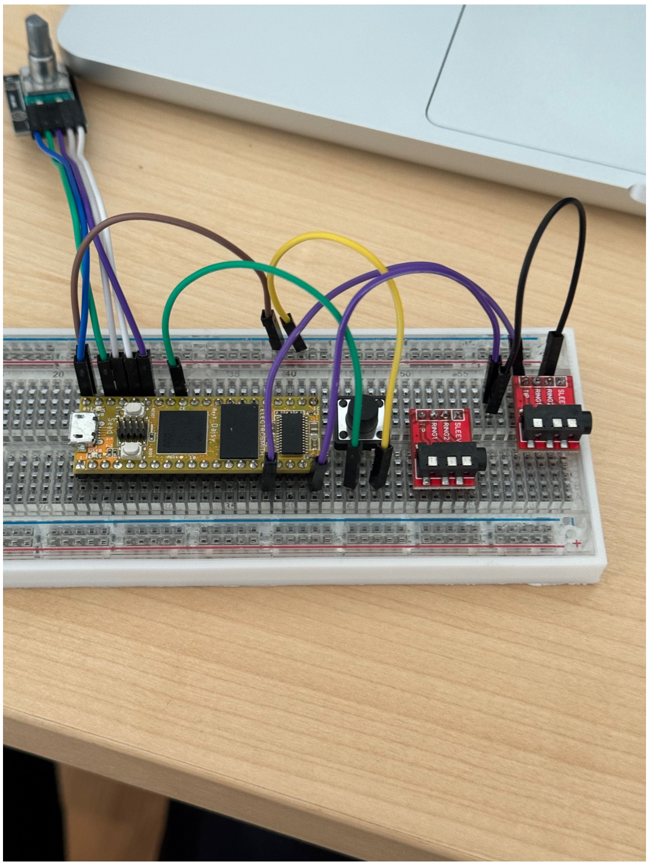

*Figure 5. First user testing prototype*

Although all users successfully managed to meet the objective, the experience and interaction differed greatly among them. The first mode of the prototype, which utilised an encoder button to toggle between handles, generally performed better. All users could switch between the handles without encountering major issues, showcasing the prototype's intuitive design. However, two of the three users required a slightly longer time to fully grasp the trimming process. This process also included pressing a key on the piano keyboard to hear the trimmed sample, adding an extra step that initially confused some participants. Despite this, they managed to trim the sample successfully in the end, demonstrating the prototype's overall effectiveness. With the second mode of the prototype, users were also able to trim the sample, but the process proved to be more cumbersome. The trimming was lengthy and often interrupted by the unintentional release of the knob while spinning. This unintended action frequently resulted in switching to the sample start handle. These issues highlighted significant areas for improvement in the design of the second prototype.

What follows are quotations from the testing of the second prototype, providing direct insights into the user experience and challenges encountered.

*Me: Which prototype do you like better?*

*User 2: Definitely the first one, since I had to twist a lot, it was much easier to do.*

*Me: Have you encountered any difficulties with the second prototype?*

*User 1: It was difficult to focus on the fact that I had to hold the knob down to adjust the end part and also play piano with my other hand.*

*User 2: Yes, the knob slipped few times while turning, it was a strange movement to use my left hand to both press and twist at the same time*

In the State of the art section, I split the existing samplers into two categories, ones that support real-time sample trimming and ones that do not. At this point, the prototype felt into the latter. I went back to the observations of existing samplers to find out what does this mean for the experience. Does it have an effect on the sampler capabilities, and if so, what exactly? Soon enough, I've noticed in one of the videos, when one user manipulated a snare sample that was being played back using the sampler's built-in sequencer to change the rhythm of a the composed track. They did this by moving the sample start knob to the left, meaning the actual drum sound was being played back with a slight delay. Although one could say that the sample was technically not trimmed properly, it resulted in a pleasing auditory experience. I was interested to see whether manipulating the sample length parameter, specifically of a long piano sample could be used to imitate sustain pedal of a real piano. This worked surprisingly well and I was able to manipulate the sound of the piano while playing a song, resulting in a sustain pedal-like auditory feedback.

### 5.4 Second prototype and user testing

With the second user testing I wanted to see the difference between the first successful prototype and a new prototype with two dedicated knobs to control sample start and sample length parameters at the same time. The intent was to compare a newly proposed single knob interaction and a two knob interaction which is already used by commercial samplers such as KORG volca sample 2, Elektron Model:Samples or Teenage Engineering PO-33 KO. The test was done with 4 subjects, from which 2 were present in the first testing. The testing itself was fairly similar to the first one, giving the user the objective to trim all silence from the original sample. After that, they were asked to try manipulating sample start and length parameters while a MIDI sequence was being played back using computer software. This aided to make the user testing fluent as majority of the test subjects did not know how to play piano. Despite this, I was still able to gain few valuable insights. The users mentioned that the second prototype (utilising two knobs) was easier to operate since they did not have to think about switching the sample start and sample length handles.

*Me: Which of the prototypes was easier to work with?*

*User 2: This one (pointing on the second prototype), It has been less stressful and I felt that I actually understood what I was doing with the sound.*

*User 3 (Intermediate piano player): After a while, it was much more intuitive for me to trim the samples since there were two separate knobs and I didn't have to think about switching with the button.*

*User 4: I liked the two knob version better, it was much more intuitive.*

Additionally, the users were able to trim the sample quicker using the two knob prototype, however this might also be due to the experience with the previous prototypes. Another observation with all prototypes was that users were almost always using their hands in the same manner, having their left hand on the sampler while operating the piano keys with their right one. I suppose this was to due to the nature of how the prototype is operated, but it was an interesting insight.

A few new issues arose when operating the sampler, the most noticeable being that navigating longer samples took slightly longer with the knob. This was because the knob was designed for small and precise steps of 1000 samples (1/48s or 20.83ms). While the precise control is still desired when trimming samples the ambiguous scrolling through long silent part is not, and I believed it could be mitigated. The interaction I've developed was inspired by my last year's project where I was using the speed of scrolling to change the density of browsed data. Essentially working with an assumption that if a set of data is scrolled through quickly, the resolution of it can be decreased as it is probably not important to the user, ultimately making scrolling through a larger dataset swift when needed while still providing full resolution if a slower browsing speed was applied. I was able to implement this by calculating a relative scrolling speed and multiplying the step size with it.

Furthermore, when working with longer audio samples, e.g. more than 4 seconds, adjustment of the sample length parameter became difficult since the user had to wait for the sample to play until the end to hear the changes they have made. In commercial samplers a display is typically used to visually represent the waveform, however I have not seen a solution to this issue among the samplers without displays. A short ideation session revealed few potential ideas on how to aid this issue. As mentioned before, the rotary encoders on the latest prototype have built-in switches, however their functionality has not been utilised in any way. I was interested to see whether playing a short part of the audio sample by pressing on the knob could be a way to inform the user of the audio present in that position. Meaning that by pressing the sample length knob down, the user would hear the few hundred last millisecond of the trimmed sample, making the interaction more understandable and quicker to execute.

Lastly, when working with short samples such as drums or percussion instruments, few instances occurred, when the important part of the audio sample was missed / scrolled past by the user in during the trimming process. Since, I've already worked with LEDs indicator in the project, I wanted to see whether such simple feedback could be used to inform the user about a presence of audio at the current position, ultimately, allowing the user to perceive the samples in a more understandable manner.

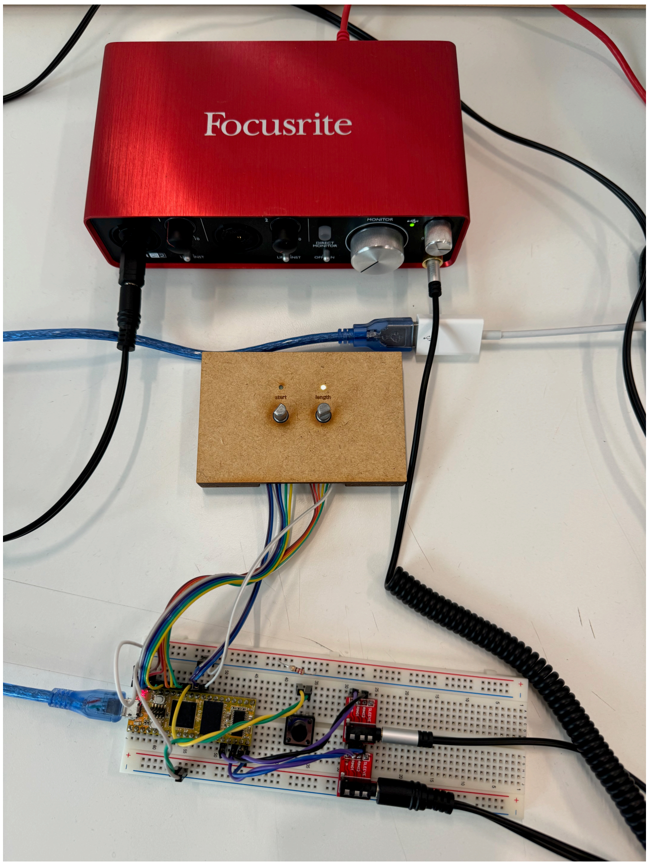


*Figure 6. Latest prototype*

**5.5 Third prototype and user testing**

For the last user testing, few test were designed to specifically test the individual features. Firstly, users were given the prototype with a prerecorded sample of a single kick drum. The sample itself was around 11 seconds long, however the actual drum sound lasted only roughly 300ms, leaving the rest silent. All users were assigned with a task of locating the drum sound and trimming all the silence away. In the first round, the users were given a prototype that did not had the LED indicators enabled, while in the second round, the LED indicators would turn on if the audio at the current position would pass a certain volume level. This threshold setting was essential since the audio samples always contained a minor noise. While users were instructed to find and trim the sample as usual, they were not told that their time was being tracked. This was done to prevent unnecessary stress on the users.

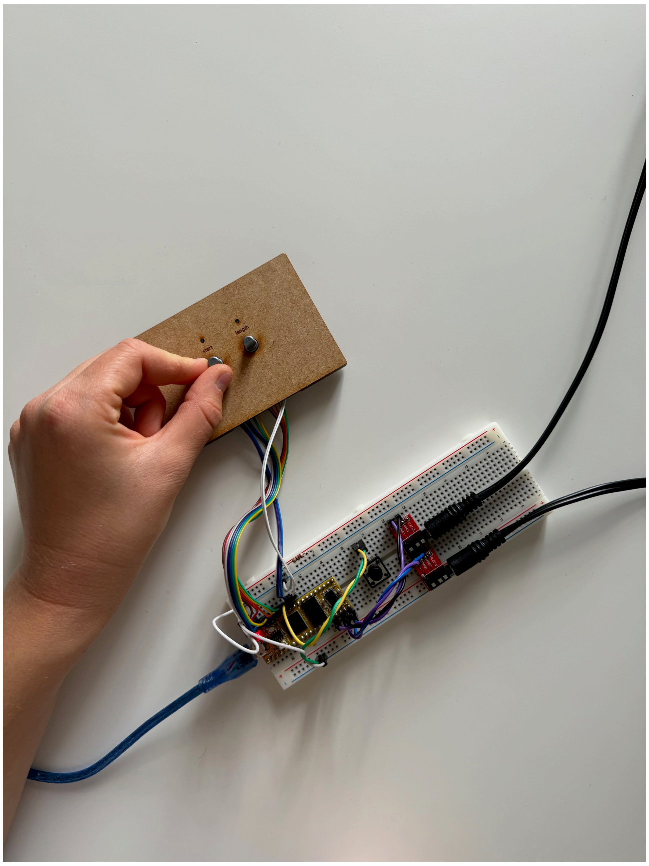

*Figure 7. Third user testing*

A comparison of the recorded times can be seen in the Table 3 bellow.

| User | Total trimming time with LED disabled | Total trimming time with LED enabled |
|---|---|---|
| 1 | 57 seconds | 33 seconds |
| 2 | 1 minute 13 seconds | 1 minute 17 seconds |
| 3 | 1 minute 27 seconds | 1 minute 3 seconds |
| 4 | 1 minute 4 seconds | 51 seconds |

*Table 3. LED indicators time comparison*

The users were then asked a few follow up questions.

*Me: Was is easier to trim the sample with the LED indicators on?*

*User 2: I feel like the second prototype made more sense. I've noticed that the light flashed occasionally, but I ignored that because it was very quick*

*User 4: I liked the LEDs, it was easy to adjust the end of the sample since I could see when the LED turned off.*

For the majority of users, the total trimming time was shorter using the second prototype which featured the LED indicators. Although the indicator did not work completely as intended, occasionally producing a short blinks even in the silent areas of the samples, I conclude that the indicator contribute to the overall interaction clarity and understandability.

Second part of the user testing involved testing whether the short preview function would be beneficial to the overall trimming process. Each of the users were again given a task of trimming two audio samples, one drum sound (300ms long audio) and one piano note (4s long audio). Users were first shown how to use the preview function and then instructed to trim the given samples, first without the preview and then with the preview enabled. This process was repeated over four rounds. As with the previous user testing, participants were unaware that their trimming times were being measured.

The table below summarises the total trimming times for each user, both with and without the preview function:

| User | Total trimming time without preview (drum) | Total trimming time without preview (piano) | Total trimming time with preview (drum) | Total trimming time with preview (piano) |
|---|---|---|---|---|
| 1 | 37 seconds | 1 minute 5 seconds | 29 seconds | 49 seconds |
| 2 | 1 minute 7 seconds | 1 minute 57 seconds | 1 minute 3 seconds | 1 minute 22 seconds |
| 3 | 53 seconds | 1 minute 24 seconds | 55 seconds | 1 minute 31 seconds |
| 4 | 1 minute 21 seconds | 2 minutes 2 seconds | 1 minute 19 seconds | 1 minute 16 seconds |

*Table 4. Preview function time comparisons*

Despite not all users fully understanding or utilising the preview function, most were able to use it at least once. In most cases, the preview function helped users trim the audio samples more quickly, which indicates that the preview feature can be a valuable tool in the audio trimming process, reducing the overall time needed to complete the task and potentially improving user efficiency. I also believe that user testing such as this one is not the ideal tool for evaluation since the sampler at it's current state is already a complex instrument that requires a certain level of skill development.

## 6. Discussion

### 6.1 Complexity of embedded systems and audio processing

While while platforms such as Daisy Seed are greatly beneficial when it comes to prototyping, they require substantial knowledge and effort to become truly useful. This project revealed that while high-fidelity prototypes can deliver functional and realistic designs, the technical investment required can detract from the ability to focus on interaction design. This highlights a need for tools or methods that simplify the use of embedded electronics, allowing designers to prioritise user interaction without being overwhelmed by technical challenges. An alternative approach worth exploring would a form of hybrid prototyping, where audio processing could be handled by a computer, while embedded systems focus on interface design. This method could potentially simplify development by reducing the technical challenges tied to real-time audio processing on embedded platforms. By leveraging the computer's more powerful processing capabilities and advanced software tools, designers might gain more flexibility to refine the interface. However, this approach is still speculative and would require further investigation to determine whether it truly offers these advantages in practice.

### 6.2 Extended and longitudinal testing

User testing sessions during this project were quite small, usually involving fewer than five participants. I believe this is not ideal for properly assessing the prototype's usability. Conducting more extensive user testing with a larger and more diverse group of participants would provide a wider range of feedback. This would help identify more usability issues and areas for improvement that might not be noticed with a smaller group. A larger sample size would give a better evaluation of the prototype, leading to a clearer understanding of its strengths and weaknesses and resulting in a more refined and user-friendly final product.

Since the prototype has became a considerably complex instrument, implementing a longitudinal user testing could also become beneficial. This approach would allow users to interact with the prototype over an extended period, providing more detailed insights into its usability and performance. Longitudinal testing would also enable for skill development, ensuring a more thorough assessment.

### 6.3 Tactility in real-time audio interfaces

Through iterative prototyping and testing, I gained a deeper appreciation for how users interact with auditory-based interfaces. For example, integrating real-time manipulation features and tactile feedback improved user engagement and efficiency. These experiences have broadened my understanding of how auditory and tactile elements can enhance interaction design, a lesson applicable to broader contexts beyond audio sampling.

### 6.4 Hardware comparison

The state of the art analysis provided valuable insights into the capabilities and limitations of different hardware tools. For example, platforms like Raspberry Pi Pico and Teensy 4.1 offer varying trade-offs in terms of RAM, operating frequency, and audio processing capacity. The comparison table could serve as a reference for designers and engineers when selecting hardware for specific scenarios, emphasising the need to align platform capabilities with project requirements. This structured evaluation of hardware tools could also guide future workflows in prototyping.

### 6.5 High fidelity prototyping

By focusing on high-fidelity prototyping with embedded systems, this project contributes to the discourse on creating functional and realistic prototypes in interaction design. The exploration of Daisy Seed's capabilities, combined with the iterative user testing process, provides a roadmap for other designers interested in pursuing similar projects. Specifically, the lessons learned about memory management, audio manipulation, interface design and prototyping effectivity can inform future work in both audio and other interaction domains.

## 7. Limitations

This project faced several limitations that influenced its scope and findings. The most significant challenge was the technical complexity of using platforms like Daisy Seed, which demanded substantial time and effort, limiting the ability to concentrate more on interaction design. User testing was limited to a small sample size and short-term sessions, which constrained the diversity and depth of feedback, leaving long-term usability and skill development unexplored. The prototype's use of basic components, such as rotary encoders and LEDs, limited the scope for innovation in interface design. Alternative prototyping methods, such as hybrid approaches that would utilise computer-based audio processing, were not fully investigated, potentially missing opportunities to balance technical complexity with design efficiency. These limitations highlight areas for future research to expand upon and refine the project's insights.

## 8. Future work

There are two potential directions for the continuation of this project. Given the technical challenges encountered during the prototyping process, one approach could involve exploring alternative materials and prototyping approaches that facilitate a smoother integration of technical functionality with a focus on interaction design. This could begin with experimenting with previously mentioned hybrid prototyping method or by revisiting foundational research to investigate prototyping practices adopted by others in the field. On the other hand the project could possibly look into different ways of presenting the audio sample, which is currently represented by an LED. Could a knob with haptic feedback be used to project the audio wave on the time axis, providing a resisting force or various vibrations that correspond to different parts of the audio sample? Would this give users a more tactile and intuitive way to interact with the audio? Additionally, would a slider work better than a knob for this purpose? Why do all commercial samplers use knobs? Could sliders offer a more straightforward and precise method for audio manipulation, or are there ergonomic or usability factors that make knobs the preferred option? If haptic feedback proves to be impractical or ineffective, what are the visual ways of displaying the audio? Could a round display that wraps around the knob provide a visual representation of the audio sample directly linked to the knob's rotation? Would this tight coupling enhance the user's ability to see and adjust the sample accurately?

## 9. Conclusion

The project has successfully developed a functional audio sampler, similar to existing commercial models, and has provided insights into their design and functionality. Although the latest prototype mostly resembles what is available on the market and one could argue that there is little to no novelty, it helped me understand why the available samplers are designed the way they are. Manufacturers usually do not disclose their design choices, making it hard to learn from them. I was able to carefully examine and test multiple interfaces for essential sampling functions. By doing so, I was able to determine which designs work best for sample recording, manipulation and playback. Through this process, I was able to see why certain designs are popular in commercial products.

Additionally, this project has greatly expanded my knowledge in the fields of audio, embedded electronics, and their use for prototyping in interaction design. I explored the complexities of real-time audio processing, such as setting audio block sizes and sample rates, and implementing audio callback functions to handle audio inputs and outputs. The project also required me to understand memory management, pitch shifting, and integrating components like rotary encoders and buttons.

A key finding, as discussed in section 6.1, is the significant technical complexity of working with embedded systems for real-time audio processing. While platforms like the Daisy Seed enable functional, high-fidelity prototypes, they demand substantial technical expertise and effort, which can detract from focusing on interaction design. This raises the question of whether alternative methods, such as hybrid prototyping approaches that leverage computer-based audio processing, might strike a better balance between technical functionality and user-centered design.

Overall, this project was an exploration of the material of embedded electronics and audio, providing a solid foundation for further exploration in the field of audio sampling.

## References


Hanington, B., & Martin, B. (2012). *Universal methods of design : 100 ways to research complex problems, develop innovative ideas, and design effective solutions*. Quarto Publishing Group USA.

Karana, E., Hekkert, P., & Kandachar, P. (2015). Material-driven design (MDD): A method to design for material experiences. *International Journal of Design, 9*(2), 35–54. https://ijdesign.org/index.php/IJDesign/article/view/1965/693

McGuire, S., & Pritts, R. (2008). *Audio sampling: A practical guide*. Focal Press.
Adobe. (n.d.). *Sample rates and audio sampling: A guide for beginners*. Adobe Creative Cloud. Retrieved May 28, *2024, from* https://www.adobe.com/uk/creativecloud/video/discover/audio-sampling.html

Electrosmith. (n.d.). Daisy Seed. Retrieved May 31, 2024, from https://electro-smith.com/products/daisy-seed

Electrosmith. (n.d.). *libDaisy: Getting started*. Electro-Smith. Retrieved May 28, 2024, from https://electro-smith.github.io/libDaisy/index.html

Lee, J. (2006). The Chopping Board: Real-time Sample Editor. Proceedings of the International Conference on New Interfaces for Musical Expression, 77--78. https://doi.org/10.5281/zenodo.117695

Argo, M. (2004). The Slidepipe: A Timeline-Based Controller for Real-Time Sample Manipulation. Proceedings of the International Conference on New Interfaces for Musical Expression, 189–192. https://doi.org/10.5281/zenodo.1176581

Calegario, F. (2019). *Designing Digital Musical Instruments Using Probatio.* [electronic resource] : A Physical Prototyping Toolkit (1st ed. 2019.). Springer International Publishing.

Electromaker. (2024, May 31). *The essential guide to prototyping your new electronic hardware product*. Electromaker. https://www.electromaker.io/blog/article/the-essential-guide-to-prototyping-your-new-electronic-hardware-product