# MetroGnome: Tactile Beats

December 4th, 2023

MSE 4803 - Smart Textiles

Sarah Al Bastaki, Jenny Gerber, Mir Jeffres, Joshua Hunter

Table of Contents

# Executive Summary

The MetroGnome aims to revolutionize the way musicians, athletes, and medical professionals engage with rhythm and timing. The project involves the development of a wristband Metronome to provide a wearable solution for maintaining precise timing in music, sports, and in other endeavors.

The MetroGnome is designed to offer rhythmic accuracy, with an adjustable BPM range to cater to various musical genres and necessary rhythmic patterns. It incorporates user-friendly controls, allowing users to set BPM and visually see the tempo with the lit up LED display. The wristband is compact, lightweight, and constructed from durable materials for optimal comfort during extended wear.

The MetroGnome Wristband targets a diverse market, including musicians, athletes, and individuals seeking a versatile timing tool. By offering a unique combination of precision, portability, and comfort, the product aims to capture a significant share in the wearable music technology market.

# Introduction

The origin of the project comes from the desire to keep a beat. Mir plays for the marching band and playing to the tempo is an essential part of a good musical performance. Tempo is usually distributed via a metronome, but the issue is that in a high volume environment like a marching band performance, it is hard to hear a metronome. He had always thought that a tactile metronome would be a good idea but had never had the opportunity to implement it. In discussing his idea, Jenny realized that she too had a good use case for the tactile metronome. Her friend had recently ran a marathon and complained to her that someone had a very loud metronome playing, so they could run to a beat per minute. Running to a beat per minute is something that is very popular for runners.[1]

With the central concept being a tactile BPM device, the group started planning how to orient it. The project had to be a smart wearable, so we had to determine how the device would be worn. A t-shirt was chosen at first, but was decided against in favor of a simpler implementation. There were concerns that a t-shirt would be too warm for marching band players and runners. Another issue arose that the t-shirt would be hard to interact with, as we wanted the user to be able to easily adjust the BPM. A wristband was chosen to keep the user cool as well as being easily accessible on the wrist.

Our skills were easily integrated into the project. Jenny had extensive textile experience, so she was put in charge of design and implementation of the fabric as well as the electrically conductive fabric and thread. Mir was our music base, who was expected to help design the BPM software and explain the music expectations to the other team members. We decided to implement an app because of Sarah's app experience. That left all of the leftover tasks to Josh, who is a chemical engineer (which does not lend itself to being very useful in this context). He handled the writing, wire cutting, and team motivation.

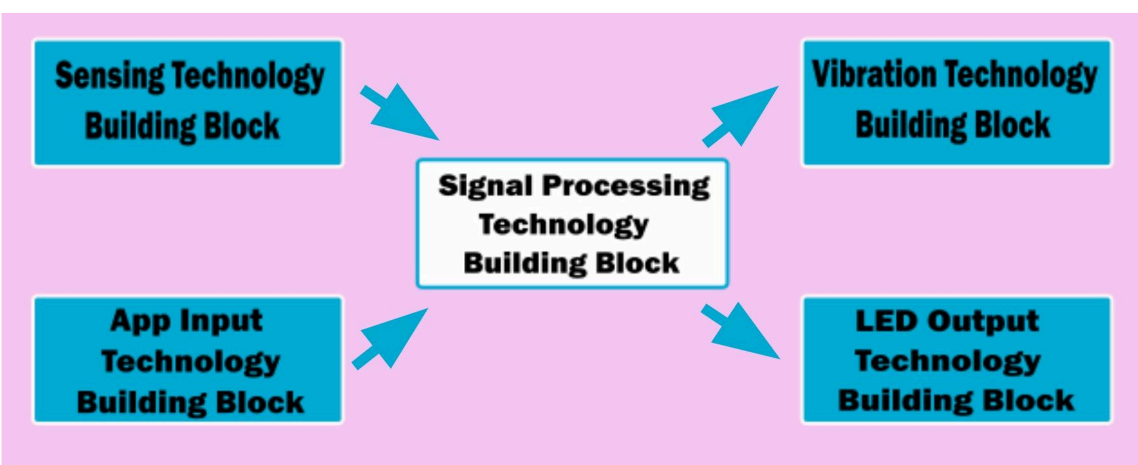
In terms of the use cases, the two major ones we were already aware of at the start of our project, runners and marching band people. As we thought more about BPM and the other applications, we realized more use cases. In physical therapy contexts, physical therapists have their patients complete a task on a specific beat, so with this device, physical therapists could easily provide a vibration for their patients. Medical workers often have to do CPR to a specific BPM, so this device could be used to help maintain that BPM. Lastly, there is a large portion of Americans that could benefit from having this device. 15% of Americans over the age of 18 report hearing loss.[2] These Americans can't use a traditional metronome due to their hearing loss, so this device would be a good alternative for them.

# System Building Blocks

The sensing component is the buttons on the device. The buttons receive inputs from presses and the time between button presses. The sensing also utilizes an app for inputs. The app incorporates a button and time between button press input as well as preset BPM inputs.

The processing component is the Adafruit Itsy Bitsy Microcontroller and CircuitPython code. The microcontroller fits inside the design of the wristband. The microcontroller runs a CircuitPython code that processes the parameters that are monitored and connected, and then produces our output.

The output comes in two forms. There is an LED network connected to the microcontroller that lights up at specific LEDs to produce a visual numeric understanding. Once input is set, the LEDs flash in order, starting with 100s, the 100 LED and the specific digit LED light up. Then the tens will blink along with the specific digit LED. Then the one's place digit will light up. The other output is the vibrational component. The buzzer is connected to the microcontroller and vibrates a specific BPM back to the user so they can experience a tactile BPM.



# System Specifications

The MetroGnome environment, which includes the device and user interface aims to present a tactile metronome experience. The project was designed with key target specifications which are outlined below:

**Device**:

A. Ability to calculate the tempo from a set of consecutive taps: the device must be able to accept taps from the user on the tap button and calculate the average tempo the user desires.
B. Haptic feedback with tempo: the device will trigger the buzzer component to vibrate to the tempo desired.
C. Visual display of tempo: LEDs will be assembled in a matrix labeled with digits. The digits corresponding to the tempo will be lit. For example, if the tempo is at 80 BPM, LEDs corresponding to digits 8 and 10 will be blinking and 10 will be lit stably.
D. Comfortable Design: hidden electronics to make the user feel comfortable wearing the device.

**Mobile App:**

A. Ability to communicate with the device: ensuring the device can communicate the tempo calculated in the app to the device using BLE UART protocol.

B. Ability to calculate the tempo from a set of 3 consecutive taps: using time between registered taps to calculate the tempo.
C. Serve as a manual for using the device: ensure enough information is on the app for users to feel comfortable using the app for the first time.

The team recognizes that target specifications are driven by customer needs and require elaborate design, but tradeoffs and feasibility drive the project to the final project and specifications. Therefore, the final specifications are the following:

**Device**:

A. Ability to calculate the tempo from a set of consecutive taps: the device is able to accept taps from the user on the tap button and calculate the average tempo the user desires.
B. Haptic feedback with tempo: the device triggers the buzzer component to vibrate to the tempo desired.
C. Visual display of tempo: LEDs are assembled in a matrix labeled with digits. The digits corresponding to the tempo are lit, but occasionally light up the wrong digits.
D. Comfortable Design: electronics are hidden. Electrical buttons are replaced with fabric buttons to reduce bulkiness and maintain flexibility throughout the device.

**Mobile App:**

A. Ability to communicate with the device: the app is currently unable to communicate with the device via BLE.

B. Ability to calculate the tempo from a set of 3 consecutive taps: the app successfully uses time between registered taps to accurately calculate the tempo.

C. Serve as a manual for using the device: the Instructions page on the app has the necessary information to operate the app and device.

# Prototype Development

The comfort component was one of the first things we considered in our prototype development. As this device would be used in athletic, sweaty contexts, we wanted to do a polyester fabric base because polyester absorbs sweat. Polyester is a very common fabric for consumers to wear on their arms, and we wanted consumers to be as comfortable as possible. Polyester also has a very good stretchability and flexibility, and as this would be used in high movement contexts, we wanted that as well.

For the sensing component, we knew we wanted to incorporate fabric buttons into our design after we watched the lecture by Clint Zeagler because of how interesting they looked. We designed our fabric buttons and found that they worked extremely well in our prototype.

The decision for our hardware and software was informed by prior experiences with smart wearables. The AdaFruit Itsy Bitsy was the microcontroller that Mir and Sarah both had experience with. It was also extremely small, which is what we desired in our microcontroller. Too bulky would have left an impression in the device or made the wearable feel bulky. The software wasn’t a decision we made; CircuitPython is the code for Adafruit Itsy Bitsy.

The output was designed around a haptic BPM response. To this end we implemented an AdaFruit Buzzer and AdaFruit Buzzer Controller. The Buzzer controller adjusts the buzzer’s vibrations, while the buzzer vibrates to the desired BPM. In conjunction with our desired BPM buzzer output, we decided to implement an LED array to let users see their current BPM. The LEDs are Neopixels implemented into a flexible protoboard.

# System Testing

The parameters monitored by the system are pressure, time, and app presets. The pressure parameter comes from the fabric buttons on the device, who respond to a pressure input i.e. a press down of the button. This is used for increasing the BPM, decreasing the BPM, and setting a desired BPM. The time parameter comes from the duration of the button press and the time between buttons. A long press of a button is used for turning on and off the device while the time measured between buttons is used the tap tempo algorithm to determine the desired BPM from the button taps. The last parameter is the app input, which comes from the preset BPMs on the app.

During the development process, testing was a routine process. Electronic connections were continuously tested using a multimeter. This was done to ensure that all connections and soldered components were proper. Also, this was the key tool used to ensure the fabric buttons work. Furthermore, laser cutting and engraving the buttons required testing different power levels, speeds, and points per inch (PPI). Using the wrong levels for each of the mentioned settings would have caused the fabric to burn if they were too high and not be visible if the settings were too low. Thus, using some of the fabric, we conducted swatches with varying settings to determine the best levels.

Testing also helped the team recognize what target specifications may be infeasible due to constraints such as time, experience, or hardware. Such specifications are the ability for the mobile app to communicate with the device using BLE. Testing was very much valued throughout the process and will continue to be a practice in the team’s future endeavors.

# Learning

In terms of learning, we all had different individual learning experiences. Josh is a chemical engineer and chemical engineers don’t take any courses in circuits, so this class was a lot of learning about how to build circuits for him. Jenny also developed her circuit experience by a lot, she learned how to incorporate conductive thread into circuits and how to build more complicated circuits. Sarah gained experience in app development that she wanted and feels more confident in her app creation experiences. Lastly, Mir gained knowledge in CircuitPython and developed his CircuitPython skills even further.

For our Aha moments, there were a few. We had each LED in our LED network have their own ground thread, which was proving difficult to sow. We realized that we could just run one thread through all of the grounds for the LEDs and the system would still work, which saved us a lot of headaches. We also started using wires for our LED network which pushed the LEDs too far apart to have a small design. Mir remembered that he had a Flexible Protoboard that could have LEDs soldered onto them, and that the Protoboard could be sowed with conductive thread to create an LED network that was tighter and easier to control.

# Conclusion

The MetroGnome is an innovative and versatile solution to address the need for precise timing in various fields, including music, sports, and healthcare. Our collaboration brought together diverse skills, from textile design and music expertise to app development and hanging out. The decision to focus on a wristband form factor was driven by considerations of comfort, accessibility, and ease of interaction for users in dynamic environments.

The device's building blocks were chosen for functionality, compactness, and user comfort. The integration of tactile and visual feedback through a buzzer and LED display enhances the user experience, providing both a numerical representation of the tempo and a tangible vibration synced to the desired BPM.

The development process involved thorough testing, ensuring the reliability of electronic connections, fabric buttons, and the overall functionality of the MetroGnome. Our commitment to iterative testing allowed them to refine their design and make informed decisions about the feasibility of certain specifications.

In conclusion, the MetroGnome project not only meets its original goal of creating a tactile metronome but also showcases the potential for interdisciplinary collaboration in developing wearable technology. Our dedication to user comfort, functionality, and continuous improvement positions the MetroGnome as a promising entrant in the wearable music technology market.

# References

1. Runners Need. (n.d.). https://www.runnersneed.com/expert-advice/training/running-and-music-finding-your-bpm.html
2. CDC. (n.d.). *2012 health statistics for U.S. adults: National Interview Survey,*. CDC.gov. https://www.cdc.gov/nchs/data/series/sr_10/sr10_252.pdf

# Appendix

To turn on the MetroGnome, long press on the on button. The device will start vibrating the last set BPM. To increase the BPM by 1, press the up arrow. To decrease the BPM by 1, press the down arrow. To set a specific BPM, press on the TAP button at the desired BPM. Four taps are all that's needed and the system will start vibrating the inputted BPM.

To read the LED display, examine the order of LED flashing. First, the 100s LED will flash along with the associated digit for the 100s place. Then the 10s LED will flash along with the associated digit for the 10s place. Lastly, the singles digit will flash.

To use the app for the MetroGnome device, simply input the desired BPM into the app. The MetroGnome will start vibrating to that bpm and the MetroGnome's LEDs will light up to the corresponding number. The MetroGnome device also has preset BPMs you can select from. You can also tap the desired BPM on the app to and the app will determine the BPM from your taps and then vibrate that BPM on the MetroGnome. Enjoy!