Wearable Devices for Software Instruments

Team 17 — John Born, Kuang Wang, and Miguel Torres
ECE 445 Project Proposal — Fall 2019
TA: Kristina Miller

1 Introduction

1.1 Objective

Choreographed movement is known to enhance the expressivity of a musical piece. This is most evident in dance performances but can also be seen in musical performances spanning genres from pop to experimental. It is customary for music and choreography to be completed separately: in the case of a pop artist performance, choreography is used to enhance the music, and in the case of a dance performance, choreography is inspired by the music. Even though movement contributes to the experience of a musical performance, the dancer has no influence in the composition process of the music. Contributing factors to this separation are:

> 1) the performer must interact with a physical instrument during the performance, thereby limiting the performers range of motion.
>
> 2) the playability of typical instruments requires trained dexterity and coordination (i.e. the flute requires remembering note fingerings) from its performer.

With the advancement of computing technology, creating music has become more accessible to non-musicians by way of electronic instruments and computers. However, interacting with these newer instruments still requires the capability of pushing buttons, twisting knobs, and moving sliders during a performance. Unfortunately, this continues to exclude dancers from being an active participant in the composition in which they perform.

In order to include dancers in the process of composing music this project aims to make a set of wrist and ankle wearable devices that will use the performer’s movement to control a software instrument. The orientation data of these devices will be used to control sonic characteristics—such as: pitch, volume, modulation, filter cutoffs, and panning—of the software instrument. The orientation data will be transmitted wirelessly through Wi-Fi using the Open Sound Control (OSC) protocol—a URL style transmission—to a laptop. The laptop will use the program OSCulator to decode the incoming transmission and send it to the Supercollider IDE where the data will be mapped to a custom-made software instrument.

1.2 Background

Performances, musical and dance, are increasingly incorporating various forms of technology to captivate their audience through the use of projections, custom lighting, and lasers. These productions involve dancers performing to pre-arranged multimedia, which inadvertently excludes the dancers from the compositional aspect of said productions. Regarding music, the direction of inspiration between composer and choreographer tends to be one way, where the composition influences the choreography. Attempts to make this relationship more bidirectional have been explored as early as 1965 with *Variations V*, a collaboration between John Cage and Merce Cunningham which debuted at the New York Philharmonic's French-American Festival. This performance involved twelve antennas and photocells set throughout the stage and used to sense the proximity of seven dancers in order to trigger sound [1]. Experiments such as these change the dynamic between composition and choreography to be more interactive. The term Interactive Dance is now used to refer to such productions where the dancer, through the use of technology, is able to influence the musical composition and be affected in return [2]. Through this project, we seek to continue the evolution of such technology while making it more accessible to non-musicians.

1.3 High-level Requirements

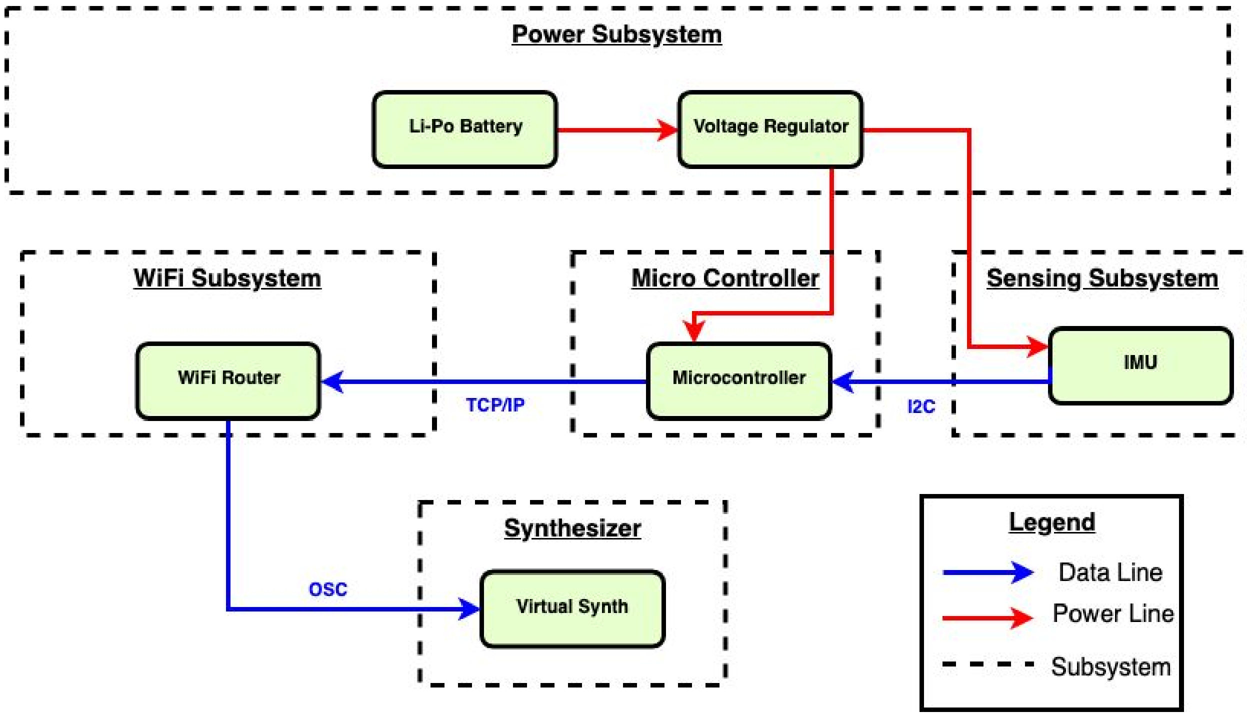
- In order for a performance to sound seamless we will need to insure that there are no abrupt jumps or discontinuities in the signal that may affect the sonic characteristics in a jarring manner (i.e. no unexpected jumps in pitch or volume). This will require maintaining the signals within tolerable limits and smoothing any abrupt spikes of data.

- The incoming sensor data must make meaningful changes to the sonic characteristics of the software instrument. This dictates a specified range of operation for each parameter, for instance: a pitch range spanning at least 2 octaves, filter cutoff frequencies between 50 Hz and 15 kHz, amplitude ranges from -60 dB to 0 dB, modulator frequencies from 0.01 Hz to 220 Hz, and panning from left to right. This will require the incoming data to be normalized to the respective range required by each mapping.

- Since the data from the sensors will be transmitted in real-time we expect to see some amount of data loss. We will monitor the data from the sensors and compare them to the data received over Wi-Fi to minimize how much data is lost while maintaining any latency between transmission and sound production under 10 ms [3].

2 Design

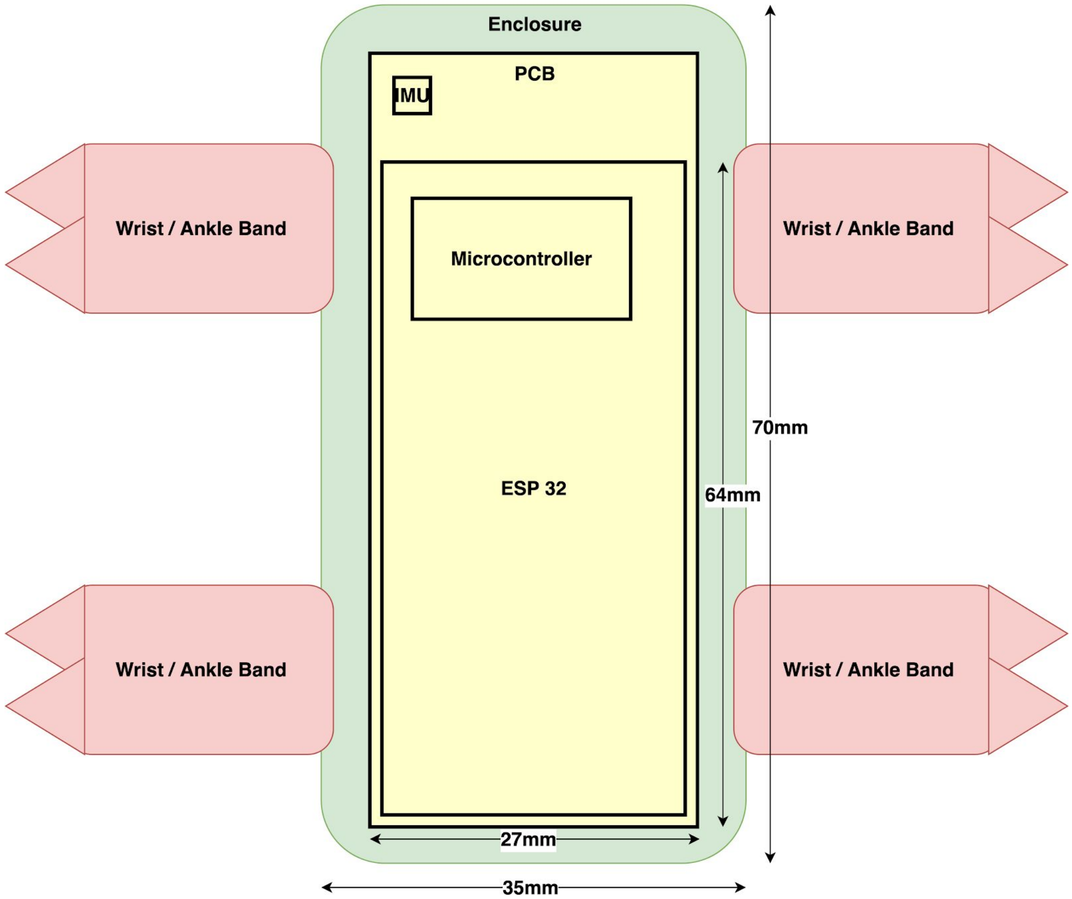
2.1 Block Diagram



The wearable devices require three subsystems to function properly: Power, Sensing, and Wi-Fi subsystems. The Power Subsystem will be comprised of a LiPo battery that will supply the necessary 3.3 V to the IMU and the Wi-Fi microcontroller and a voltage regulator to ensure there are no voltage fluctuations. The Sensing Subsystem, consisting of a 9 DoF IMU, will provide pitch, yaw, and roll data via I^2C to the Wi-Fi microcontroller. The Wi-Fi Subsystem will link the wearable device to a computer via a local Wi-Fi network adhering to the IEEE 802.11b/g/n standards. The sensor data will be packaged for transmission by the MCU using the OSC protocol.

External devices receiving the transmitted data consist of a wireless router and a computer. The wireless router will facilitate the wireless network, connecting each wearable device to the computer. The computer will receive the transmissions from the network and process the data in two steps: 1) identifying the device of origin of the data, and 2) mapping the data to synthesizer parameters. The program OSCulator will identify each device using their respective IP addresses and will forward the received data to the synthesizer. The Supercollider IDE will host the customized software synthesizer to produce the audio. Optional external devices will include speakers to amplify the computer's audio output.

2.2 Physical Design



The physical design shown is an approximation based on the dimensions for components currently in the design. The ESP 32 will be the main Wi-Fi controller mounted on a breakout board. The IMU sensor and microcontroller that communicates with the Wi-Fi controller will be mounted on a PCB connected to the Wi-Fi controller using wires. These boards will be attached to wrists and ankles using adjustable bands made of either fabric or elastic. Since these devices will need to be comfortable yet secure, they must be enclosed in a box with soft edges and a backing made of foam or some other soft material. Initially, one of these devices will be built as a proof of concept. Once a working prototype is finished, we expect to duplicate the work and produce at most 4 devices.

The devices will be worn on the wrists and ankles, where the arrows are pointing in the pictures above. To ensure that the enclosure fits comfortably and is easy to attach, we will be looking into nylon straps with velcro ends.

2.3 Functional Overview

2.3.1 Power

LiPo Battery

The battery will power the components of the wearable device. We expect to use a LiPo 5 V battery to provide the necessary power.

Requirement 1: Must be able to supply enough power for at least one hour of use.

Voltage Regulator

The voltage regulator will reduce the 5 volts from the power to 3.3 volts to directly supply the IMU and MCU with their required power.

Requirement 1: Must be able to supply a steady 3.3V for both IMU and MCU.

2.3.2 Inertial Measurement Units (IMU)

We are planning to use the MPU-9250 9-Axis Gyro-Accel-Magnet chip. It will collect inertial data (angular velocity, acceleration and geomagnetic field strength) of the devices, which will be used to calculate the orientation information (pitch, roll, yaw) required for music synthesis. The logic supply voltage should be between 2.4 to 3.6V and operating voltage supply should be between 1.7V to VDD for this unit.

Requirement 1: Must be able to detect correct inertial data of the devices, and send the data to the Microcontroller unit.

Requirement 2: Must be able to handle MotionFusion algorithm, such that we can get more accurate orientation information without too much noise and error.

Requirement 3: Must be able to handle runtime calibration, since sensors like gyroscope will get a more significant error by time. Able to reset data to zero or absolute values for calibration.

Requirement 4: Must report pitch, yaw, and roll to the accuracy within 20 degrees in any direction.

2.3.3 Microcontroller Unit (MCU)

The microcontroller we anticipate using is the ESP32. This is a Wi-Fi capable MCU for use with IoT devices. This microcontroller will be used to take in data from the IMU sensor, package the data for transmission according to OSC protocol requirements, and transmit that data with its onboard antenna. The ESP32 complies with the IEEE 802.11b/g/n standards allowing us to theoretically transmit data at 150 Mbps.

Requirement 1: Must use I^2C to communicate between microcontroller and IMU.

Requirement 2: Must be able to apply smoothing algorithms without adding significant latency to the transmission.

Requirement 3: Must package pitch, yaw, and roll data according to OSC protocol standards.

2.3.2 Wireless Router

A consumer grade wireless router will be used to facilitate a local wireless network for communication between the wearable devices and the computer.

Requirement 1: Must comply with IEEE 802.11b/g/n standards for transmission.

2.4 Risk Analysis

Several factors will pose a significant risk to the project. First, we need to maintain low latency for the entire system, as users will use their body motion to create different music effects. We need to make sure that latency does not influence the user experience of the devices. To ensure low latency, we need to make sure Wi-Fi controller can transmit data fast and stable. The programming of the MCU is an essential part of the project and will pose the most significant risk. The MCU we are using transmits data with TCP/IP between our devices and the computer. With TCP/IP transmissions, we need to make sure the connection between the devices and the computer is established properly, and it can handle any package loss. Since TCP/IP will make sure the data is completely transmitted in a specific order, the transmission time will be relatively longer than UDP/IP transmission. Thus, we may face a potential problem of larger latency. If the latency due to the Wi-Fi transmission is larger than 10ms, we will consider using UDP/IP instead and find a substitution for our current MCU that supports UDP/IP. Second, we also need to make sure the MCU can process the data fast enough to deliver it to the software synthesizer. The MCU needs to calculate the orientation from the raw inertial sensor data as well as smooth any abrupt spikes in that data. This could potentially add latency to the transmission as well.

3 Ethics and Safety

3.1 Ethics

Our design is such that the project complies with all of the Ethics requirements mentioned in the IEEE Code of Ethics [4]. There are several aspects we are primarily focusing on for the project. The IEEE Code of Ethics #5 states, “to improve the understanding by individuals and society of the capabilities and societal implications of conventional and emerging technologies, including intelligent systems” [4]. We believe our project satisfied this requirement because our project goal is to develop wearable devices for software instruments, which let not only dancers but also people without music composition experience to enjoy and compose music easier. We hope that with a combination of hardware and software technology, we can provide a better way for individuals to interact with music composition. The IEEE Code of Ethics #7 and #10 mentioned about how we accept criticism from others and assist co-workers in their professional development [4]. Just like how the ECE455 course designed, we worked as a group with assistance from the course’s staff. We will accept any honest criticism from the course’s staff, to acknowledgment and to credit each person’s contribution to the project. Moreover, we will make sure group members can assist each other to make sure we all get progress in professional development and get the project accomplished.

3.2 Safety

We think there are minimal chances to have safety issues for this project. However, we still have two potential safety issues to keep in mind for both the design and development process of the project. First, since we will use a LiPo battery as a power supply, we need to make sure the battery is connected correctly with each component requiring a power supply. This is crucial in order to avoid conditions such as a short circuit or where too much current is being drawn where the potential of causing severe burns becomes inevitable. Therefore, it will be necessary to make sure that our design remains under 26°C. Second, we need to make sure the enclosure of the devices can protect users’ limbs from scratching or cutting as well as ensuring comfort for extended wear.

References

[1] L. E. Miller, “Cage, Cunningham, and Collaborators: The Odyssey of Variations V,” *The Musical Quarterly*, vol. 85, no. 3, pp. 545–567, 2001.

[2] W. Siegel and J. Jacobsen, “The Challenges of Interactive Dance: An Overview and Case Study,” *Computer Music Journal*, vol. 22, no. 4, p. 29, 1998.

[3] *The complete MIDI 1.0 detailed specification: incorporating all recommended practices*. MIDI Manufacturers Association, 2006.

[4] “IEEE Code of Ethics”, Ieee.org. (2019) Online https://www.ieee.org/about/corporate/governance/p7-8.html [Accessed: 17 - Sept - 2019].