AirDrums: Real-Time Portable Drum Experience

Avi Ganguli
Department of Electrical and Computer Engineering
Carnegie Mellon University
aganguli@andrew.cmu.edu

Shaurya Khazanchi
Department Of Electrical and Computer Engineering
Carnegie Mellon University
skhazanc@andrew.cmu.edu

Samuel Lee
Department Of Electrical and Computer Engineering
Carnegie Mellon University
samuell1@andrew.cmu.edu

I. Table Of Contents

.

II. Introduction

AirDrums is a concept for smart portable drumsticks that would allow users to convert air drumming into a realistic experience - complete with haptic feedback from the drumsticks as responses to user actions and/or tempo correction. The concept allows for customization of drum sounds through inputs to the master device. Our approach to solving this problem mainly involves us fabricating drumsticks with PLA/ABS and embedding an Inertial Measurement Unit (IMU) to track the device that is capable of measuring the triaxial position, rotation vector and acceleration of the peripheral device. Upon sensing motion, the master device will receive the data via Bluetooth and output a corresponding haptic response throught the actuators onboard the drumsticks. The main goal of AirDrums is to conceive a low-latency device with a high battery life that can accurately detect and track drumstick motions and requires minimal instances of calibration.

Fundamentally, the project comprises of seven core layers that require integration - fabrication of drumsticks, drumstick motion sensing, master and peripheral device communication via Bluetooth, audio synthesis program, haptic feedback event generator and a frontend for the application to allow for user inputs. As extension goals, we plan on allowing some way of users to give inputs for the kick drum, but for now we plan on focusing on the the snare drums, cymbals and the tom-tom drums. As a result of this device, we hope to decrease the noise created by an actual drum-kit, increase the portability and completely eliminate the set up time.

III. Architecture

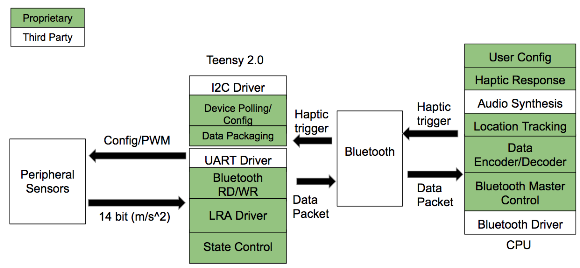

Fig. 1. System diagram of high level components

The block diagram above depicts a high level view of the system, with the arrows indicating the direction of information/data flow and the boxes representing components and sub-components. The system is comprised of 2 major components - drumstick and master device. The peripheral sensors, Teensy 2.0 microprocessor and the Bluetooth chip are embedded within the drumstick. These devices are responsible for collecting/transmitting sensor data and serializing/deserializing it. On the other hand, the master device is responsible for coordinating the Bluetooth communication protocol with the 2 drumsticks, processing received sensor data from the drumsticks and transmitting haptic sensor triggers. In addition, it will also be responsible for generating audio signals which will be played to the user.

IV. System Design

A. Drumsticks

The system will consist of 2 drumsticks embedded with 1 3.7V Lithium Ion rechargeable battery, 1 ENW-89849AZKF Bluetooth Chip, 1 Teensy 2.0, 1 BNO055 IMU, 1 NFP ELV1030AL LRA and 1 Push Button and 1 LED. The subsystems are depicted below.

1) Power and Digital I/O: This subsystem illustrates the interconnects concerning power delivery and GPIO signals. The Lithium Ion battery will be used to directly power the Teensy 2.0. External pins will also be accessible to allow the

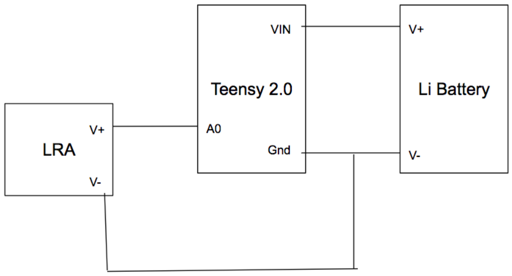

Fig. 2. Battery and GPIO connections

battery to be recharged. Pin A0 of the teensy will be used as a digital output pin for actuating the LRA. PWM signals will be generated by the Teensy to produce the desired response on the LRA.

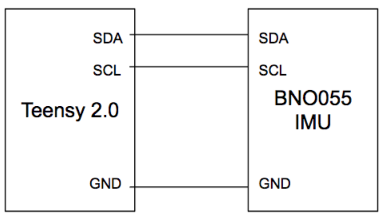

Fig. 3. I2C connection diagram between Teensy and IMU

2) I2C Bus and IMU: The IMU will be connected to the Teensy using I2C. I2C was chosen to allow additional peripheral sensors to be added to the bus easily. Pull-up resistors are embedded within the Teensy so external pull-ups are not required. The Teensy library comes with support for managing the I2C protocol which we will use. On boot, the Teensy will configure the IMU to the appropriate mode. After which, periodic reads on the IMU will be performed in order to extract the Euler angle and accelerometer data. This data will then be packaged and sent to the ENW-89849AZKF Bluetooth Chip for transmission.

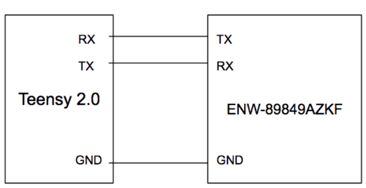

Fig. 4. UART connection diagram between Teensy and ENW-89849AZKF Bluetooth Chip

3) UART and Bluetooth: The ENW-89849AZKF Bluetooth Chip will be connected to the Teensy using the UART serial protocol. UART was chosen due to the potentially larger data packet sizes that will be transmitted between the Teensy and the bluetooth chip. The Teensy will be responsible for transmitting packaged sensor data to the ENW-89849AZKF Bluetooth Chip transmission buffers for querying by the master Bluetooth device. The Teensy will also be responsible for polling the Bluetooth chip for signals used to trigger the LRA.

B. Master Device

The master device represents external computation on which most of the processing in the system will be done. Initial development stages will use a laptop. After which, upon satisfactory performance and validation, the system will be ported over to an iPhone to meet out portability design goals. As depicted in the Architectural diagram, the master device is responsible for the following - user interface, application configuration, audio synthesis, location tracking, haptic signal triggering, data encoding/decoding and Bluetooth coordination.

1) Bluetooth master control: Using the on-device capabilities and modules, the master device will establish connections to the Bluetooth chips located on the drumsticks. Upon successful connection, the master device will be responsible for polling the devices for sensor information which will be used for location tracking/spatial orientation. The master will also transmit haptic trigger signals to the slave devices to actuate their respective LRAs.

2) Tracking: In order to track the drumsticks in 3D space, an initial calibration step is required. This involves holding the drumsticks side by side momentarily in order for the master device to establish an origin point. In order to perform real time tracking of the drumsticks the 3-axis accelerometer data from the drumsticks will be used to calculate displacement using the following kinematic relationship:

$$s = ut + \frac{1}{2}at^2 \tag{1}$$

where s is displacement, a acceleration, u initial velocity and t time period of the constant acceleration. Within the context of the system, u will be the velocity of the drumsticks calculated from the previous iteration, a will be the current acceleration measured by the IMU and t will be defined as the time step between 2 successive iterations which will be obtained from our sampling frequency. The Euler angle output of the IMU will also be incorporated in order to determine the orientation of the drumstick in 3D space so as to accurately determine the actual location of the drumstick tip. Using pre-defined displacement values to designate the drum surfaces in front of the player, the CPU will trigger the appropriate audio signals and haptic responses which will be relayed to their respective sub-systems for processing.

3) Audio Synthesis: In order to generate drum sounds, the system will utilize a lookup scheme. Upon detection of a hit during the tracking stage, the synthesizer will receive a data packet indicating the drum type and the force of the hit. These parameters will then be used to retrieve particular `.wav` files which will be played back to the user.

4) Haptic Trigger: A haptic trigger event will be generated whenever a hit on a drum is detected in the tracking phase. The master will encode the force of the hit as a haptic trigger event that will be encoded and sent via Bluetooth to the appropriate drumstick for actuation.

5) Application Interface: In order to allow user customization, the application will provide ways to change the type of drums and their location with respect to the calibration point. Changes to these options will be translated to changes in the parameters used to designate the location of the drums as well as the set of sound files used for audio synthesis.

V. Design Trade Studies

The design, as shown in Figure 1, is composed of 4 major functional subsystems: three dimensional position tracking, audio synthesis, haptic feedback and packaging/sending/receiving data over BLE. In order to meet the design requirements specified earlier, many alternatives were considered before determining the one that would provide the best functionality in our design.

A. 3 Dimensional Position Tracking

The accuracy of tracking the position of the drum stick is a critical design requirements. In addition, the location must be sampled with high frequency to avoid stale position data which may lead to incorrect results. To maximize the accuracy of the data from the sensor, it needs to provide high resolution data. After further research, we found that most small sensors provided up to 16 bits of resolution. Based on the experiences of the team members with I2C, a digital accelerometer with an I2C output was most suitable for our project. Based on prior research [**1**], the acceleration of the drumstick would never exceed $\pm 8g$ prior to contact. The sensor set should also fit into a drumstick and thus must be less than 3cm in dimension. The following alternatives were considered based on these criteria:

1) Gyroscope and 2-Axis Accelerometers: The Adafruit L3GD20H Triple-Axis Gyro Breakout Board is a gyroscope that fits the above criteria. In addition, the gyroscope can allow us to measure the angle of movement even when there is no acceleration which other alternatives may fail in. However, to actually measure the acceleration, two 2-axis accelerometers are necessary. The Memsic MXC6244AU satisfies the criteria and provides 8 bits of resolution for 2 axes which is sufficient. However, the challenge of this approach would be to integrate the 2 accelerometers and gyroscope readings and be able to synchronize them in a way that get pairs of readings per sample instead of sampling them individually. This would be time intensive and could lead to higher skew in our readings which would lead to more recalibrations proving it to be infeasible.

2) 3-Axis Accelerometer: The Adafruit MMA8451 Triple-Axis Accelerometer w/ 14-bit ADC fit the above criteria. As all the readings would be collected by this sensor, there is no need for synchronization with other readings. However, the challenge of obtaining movement vectors from these readings would prove substantial and be error-prone as none of the team-members have experience in the domain.

3) Inertial Motion Unit (IMU): The Adafruit 9-DOF Absolute Orientation IMU Fusion Breakout provides the features of all of the above. In addition, it outputs both the Euler angle and 3 axis acceleration as output. The position of the drumstick and the force of impact it would have on the drum set can be determined with a high degree of accuracy. The high level nature of the output also makes it easier to package and send over. Thus, this option was selected.

B. Audio Synthesis

Audio synthesis is another critical subsystem in this project. The synthesis needed to be done real-time (no buffering the data for later processing) and done quickly (under at most 45ms). The inputs to the synthesis program will be the acceleration and the angular velocity of the drum stick along with which drum is to be played. The synthesis program would output a sound based on this input. The following methods were considered:

1) Creating Audio via MIDI: Musical Instrument Digital Interface (MIDI) is an interface that can be used to play music via description of the instrument, notes played and their duration along with various other information. Though, there are multiple MIDI to audio synthesis libraries on Python and Swift, the libraries are built to synthesize existing MIDI from a file. File I/O is expensive and cannot be accommodated to real time processing. Also, the lack of familiarity our team-members have with musical concepts such as notes, pitch and duration make it difficult to accurately duplicate what a drum would actually sound like. Therefore, this alternative was not taken forward.

2) Playing Audio from a Soundbank: Using a soundbank to store the sounds that you will play has some significant advantages and disadvantages associated with it. The soundbank can be combined with a look-up table to enable fast audio synthesis. However, there is a strict limitation on the variety of sounds that can be played on the drums. As latency is a bigger concern than variety of sounds and the soundbank can be augmented further later on as necessary, this method was selected.

C. Haptic Feedback

The haptic feedback subsystem is necessary to provide feedback on impact to the drummer. By establishing the plane where the virtual drums exist, the feedback lets the drummer know where they need to hit and thus, helps determine their motion. The latency and accuracy of this response indirectly affects the success of audio synthesis as drummers may change the plane they play at if they receive an incorrect response. It is critical to get the haptic response in before the audio synthesis (significantly under 45 ms). The magnitude of vibration also represents the force of impact and volume of the sound generated so accuracy is critical to being able to play music at different volumes. These actuators were considered:

1) Eccentric Rotating Mass (ERM) Actuator: ERMs induces vibrations by utilizing a rotating mass. Due to the size requirement of the actuator being limited, the actuator can't use a large enough mass to create vibrations and may not be able to provide significant enough response. Also, the latency of ERMs are also higher compared to other haptic actuators. This violates too many of the design constrainsts specified.

2) Linear Resonant Actuator: LRAs induce vibration utilizing resonance thus, achieving a higher amplitude of vibration quickly and with less power. The time taken from turning on the actuator to vibration is typically less than 5-10 ms. NFP ELV1030AL provides $1.72g$ amplitude vibrations at a high frequency (205 Hz). This fits our specifications.

D. Bluetooth

The Bluetooth subsystem enables us to send data quickly over short distances. Ideally, we want to be able to transmit data with high frequency and reasonable throughput. However, due to the constant usage of the Bluetooth chip, power consumption would be significant adversely affecting battery life. Two alternatives were considered.

1) Bluetooth Classic: Bluetooth Classic can provide a higher throughput reaching up to 2Mbps. However, this is not critical for us as each packet will only need to contain Euler angle and acceleration and will be of small size. It also consumes power heavily due to its requirement of staying on the entire duration it has to operate. This would lead to a low battery life for the unnecessary benefit of high throughput.

2) Bluetooth Low Energy (BLE): BLE provides a relatively low throughput at 300kbps. This is sufficient for our purposes as the low latency (< 10ms) it provides is more critical for our application. The low power consumption allows for a longer battery life as well. The Panasonic NW89849A1KF is a BLE chip that measures less than 3cm in dimension and would be an ideal fit for embedding in the drumstick.

VI. Project Management

A. Schedule

As discussed before, this project comprises of seven core layers that require integration - fabrication of drumsticks, drumstick motion sensing, master and peripheral device communication via Bluetooth, audio synthesis program, haptic feedback event generator and a frontend for the application to allow for user inputs. And a project that extends across such a depth and breadth allows for very little miscalculation in project management. For this reason we have a very tight schedule that we need to abide by, we have allowed ourselves some buffers but according to our schedule we should be done with the complete integration of the project by around the end of April.

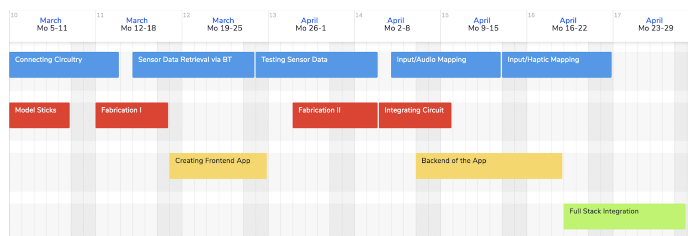

Fig. 5. Gantt Chart depicting high level project schedule

In the above diagram, there are three main timelines that run in parallel - the circuitry timeline, the fabrication timeline and the application timeline. We estimate that by the first week of April we would be ready to test the sensor data that we receive through Bluetooth and by mid-April be able to convert this data into meaningful audio signals and haptic outputs. On the application side of things, we should have the frontend and backend of the application running by around third week of April. We also estimate that we will be done with fabrication by mid to late-April. In the end, we plan on leaving two to three weeks to integrate these parts by end of April, but we also plan on working on integration of individual parts prior to this time if possible in any manner.

B. Team Member Responsibilities

As members of the AirDrums team, our division of work will be as follows: Shaurya Khazanchi - working mainly on application development, audio synthesis program and drumstick fabrication, Avishek Ganguli - working mainly on application development, audio synthesis and drivers, and Sam Lee- working mainly on circuitry, bluetooth communication and other drivers. However, all of us plan on being involved in all the parts that are not allocated to us specifically above too. The above division is a general majority of the work that all of us will be involved in at an individual level.

C. Budget Management

Even though our budget for this project is roughly \$600 we have not come remotely close to this expenditure. Our total expenditure comes out to roughly \$150 , with spare parts. This is in line with our vision of creating an inexpensive portable drumming device. We are using a Panasonic ENW-89849AZKF Bluetooth Chip that costs around \$15 , a Teensy 2.0 Microprocessor that costs \$16 , an NFP ELV1030AL LRA that costs \$8, a BNO055 IMU costing roughly \$35, a 3.7V Lithium Ion Polymer Battery costing \$15 and push buttons and LEDS that sum up to a maximum of \$10. In conclusion, we can see that our costs are far below the maximum allowed costs and even in case of usage of spare parts we will not exceed \$300 - half of our allocated budget.

D. Risk Management

1) Overhead Computation: Some of the overhead computation that we are doing might not be good enough to meet output latency bounds in that there may be a discernible difference between the time when user inputs an action and hears a sound. 45ms is the lower threshold where audio-video lag becomes noticeable so our aim is to keep our computations under this time frame. To cut down on this latency we will experiment with both aforementioned forms of Bluetooth and if that fail, try WiFi communication with HTTPS protocol.

2) Haptic Feedback Interference: We are also concerned that the actuators on the drumstick may in some way affect the readings observed by sensors on the drumstick. To avoid this we are going to place both of these components in different sections of the drumstick. We plan on placing the IMU at the top of the drumstick in order to capture the whole range of motion and place the actuators at the bottom so that the user can feel the vibration closely. This also has the pragmatic use of separating the two components so that they do not interfere with each other.

3) Fabrication Concerns: There are issues that we might run into due to fabrication concerns. Our main concern is to fit the sensors into the smart drumsticks. We also require the drumsticks to feel close enough in weight to actual drumsticks in order to emulate a true drumming experience. The drumsticks should also be flexible, durable, inexpensive and low density so it does not interfere with Bluetooth communication signals. For this reason we have decided upon PLA as the best medium for fabrication. In case this does not work out we will rely on ABS to realize the smart drumsticks.

E. Related Works

1) AeroDrums: Aerodrums involve attaching reflective silver balls at the end of regular drumsticks and tracking the position of these balls through an external PlayStation Camera. This setup is expensive, non-portable, inaccurate (has light dependencies), and has no haptic feedback.

2) Oculus Rift Apps: Certain applications compatible with Oculus Rift allow an immersive drumming experience at the cost of an expensive and bulky bundle of devices. The Oculus already comes to the user at a very expensive price, and the application itself costs a little more. This setup does include haptic feedback but is not portable. And so practicing a drumming session in another country, or while on the go is next to impossible.

3) FreeDrums: FreeDrums is the closest to our approach of air drumming, but lacks one of our main components-haptic feedback. FreeDrums involve attaching a strap onto a real drumstick in order to track the motion of the drumsticks. FreeDrums are accurate but lack the realistic experience of drumsticks due to a lack of haptic response and also have been facing criticisms of not being able to handle high velocity responses well due to slipping of the straps on the drumsticks.

REFERENCES

[1] Dahl, S. *Measurements of the motion of the hand and drumstick in a drumming sequence with interleaved accented strokes - a pilot study.* TMH-QPSR: Volume: 38.4, 1997.

[2] Wadell, Patrick, et al. Audio/Video Synchronization Standards and Solutions A Status Report. Audio/Video Synchronization, Advanced Television Systems Committee.

[3] FreeDrums. Freedrum, Freedrums, www.freedrum.rocks/.

[4] AeroDrums. Aerodrums, aerodrums.com/home/.