MUMT 619: Input Devices for Musical Expression Project Report

Patrick Ignoto
Student I.D. 260280956

April 2016

Contents

List of Figures

Abstract

A new form of gestural controller, called ”The Batons” was designed as part of the project for the course MUMT 619. The Batons are two wireless electronic devices that use accelerometers, force sensing resistors (FSR), and linear position sensors to gather input data. This input data is transmitted to a base station connected to a laptop computer. The data is then taken in and made available on a distributed mapping network created using the libmapper software library. This report documents the hardware design process and the input devices used to develop the controller. The resulting gestural controller is discussed and analyzed.

Chapter 1

Introduction

A new form of gestural interface, called ”The Batons” was developed as part of the final projects for the courses MUMT 619: Input Devices for Musical Expression and MUMT 620: Gestural Control of Sound Synthesis. The Batons are two wireless electronic devices that use accelerometers, force sensing resistors (FSR), and linear position sensors to gather input data. This input data is transmitted to a base station connected to a laptop computer. The data is then taken in and made available on a distributed mapping network created using the libmapper software library. With the data made available over this network, the user can then map the gestural data to a sound synthesizer available on the same network, to create a type of digital musical instrument (DMI). This report will discuss the hardware design and input devices used in order to create The Batons.

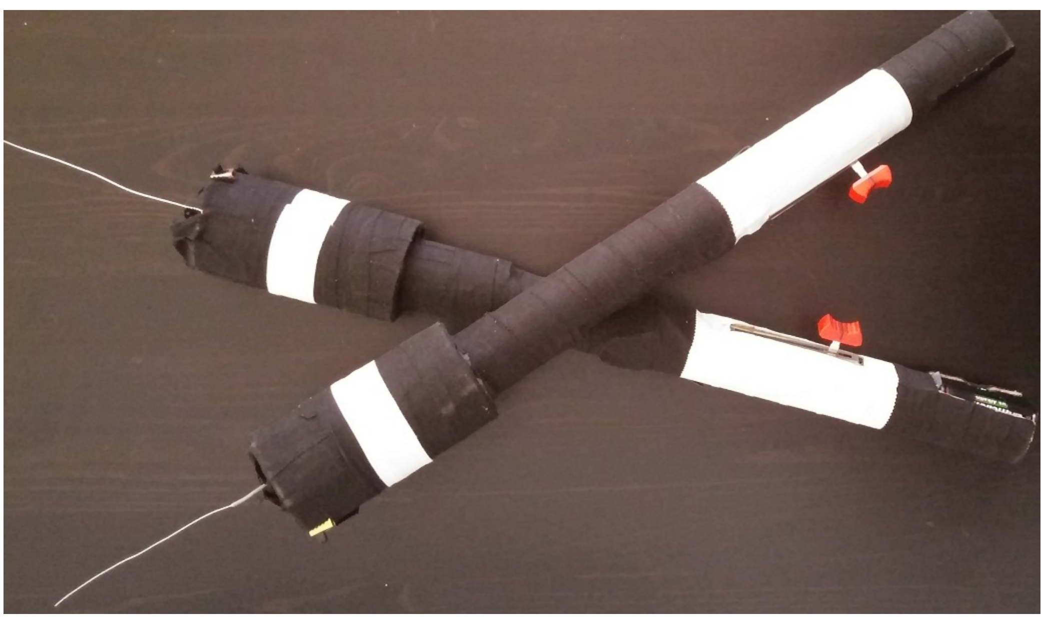

Figure 1.1: Image of The Batons

An overview of the sensors used and some background information on their use is discussed in Chapter 2. The hardware design work that was done is then discussed in Chapter 3. The resulting hardware is then discussed and analyzed in Chapter 4.

1.1 Motivation

The main motivation behind this project is to gain more experience in using accelerometers and wireless transmitter/receivers, and to apply concepts learned in the course through the design of a DMI. According to Medeiros and Wanderley[1], accelerometers and FSRs are the most widely used sensors in digital musical instruments presented at the New Instruments for Musical Expression (NIME) conference. As well, according to Cook[2], wireless technology has improved and become more viable for DMI design. Applying the two types of technology to a new design provides a much needed basis in DMI hardware design and provides a way of applying the concepts learned in the course.

Chapter 2

Background Information

The following chapter provides background information on accelerometers, force sensing resistors, and linear potentiometers as input devices.

2.1 Accelerometer

Accelerometers are sensors which measure linear acceleration in one or more axes and represent that as a voltage signal [3]. The accelerometer used to produce the Batons is the LM303D three-dimensional accelerometer by STMicroelectronics. The accelerometer is able to measure accelerations of $\pm 2g$ in each of the three axes and outputs these acceleration values as analog voltages. It is available in a breadboard compatible package and provides a digital interface with all the control circuitry necessary to send the data via I2C. The breadboard package is designed by Pololu Robotics and Electronics.

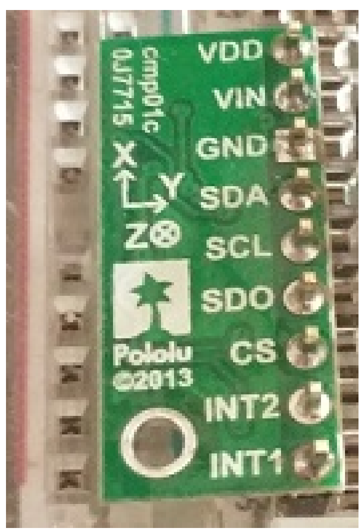

Figure 2.1: Image of accelerometer used in The Batons

The value of the acceleration in each of the three directions is sent via I2C as an analog value that is then quantized by the micro-controller. From the data sheet, this accelerometer has a sensitivity of 0.061 mg/LSB. We can therefore find the real value of acceleration in units of g-force with the following equation

$$a_{real} = a_{quantized} * (0.061mg/LSB) \tag{2.1}$$

With the acceleration in g-force units for each of the three directions of motion, the roll and pitch of the accelerometer can be calculated. Pitch can be calculated with the following equation [4]:

$$pitch = \tan^{-1}\left(\frac{-a_x}{\sqrt{a_y^2 + a_z^2}}\right) \tag{2.2}$$

Where a_x, a_y, a_z are the acceleration in the x-, y-, and z-axis in units of g-force, respectively.

The roll can be calculated with the following equation [4]:

$$roll = \tan^{-1}\left(\frac{a_y}{a_z}\right) \tag{2.3}$$

Where a_y and a_z are the acceleration in the y-, and z-axis in units of g-force, respectively.

The roll and pitch values (in degrees) are two of the signals sent to the distributed mapping network created by libmapper.

2.2 Force-sensing resistor

A force-sensing resistor (FSR) typically consists of a conductive film that increases its conductivity when a force is applied to it [3]. It is typically used as a variable resistance, whose value decreases when a force is applied to it. FSRs are typically used in a voltage divider configuration [1].

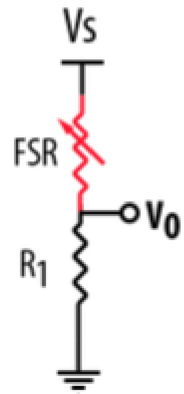

Figure 2.2: Voltage Divider conditioning circuit used for FSR [1]

When in the voltage divider configuration seen in Figure 2.2, the voltage measured at the output is:

$$V_o = \frac{R_1}{R_{FSR} + R_1} V_S \tag{2.4}$$

This means that when the FSR's resistance is very large (when no force is applied), the value V_o is very small. When the value of the FSR's resistance is very small (when the maximum force is applied), the value of V_o is approximately V_S. Therefore, the force applied to the FSR will change the voltage at V_o to a value between 0 and V_S.

Figure 2.3: Image of FSR used in The Batons

The FSR used for The Batons is the FSR-01 Force Sensing Resistor by Lynxmotion. It can be used with a V_S of $+5V$. The quantized value of the voltage representing the applied force (between 0 and 1024) is one of the signals sent to the distributed mapping network created by libmapper. This allows more flexibility in the final mapping, as any scaling factor can be performed in libmapper.

2.3 Slide Position Sensor

A slide position sensor is a type of linear position sensor, "typically consisting of a resistive element and a conductive wiper that slides along the element, making electrical contact with it" [3]. By measuring the voltage at the wiper, you can determine the relative position of the sliding element.

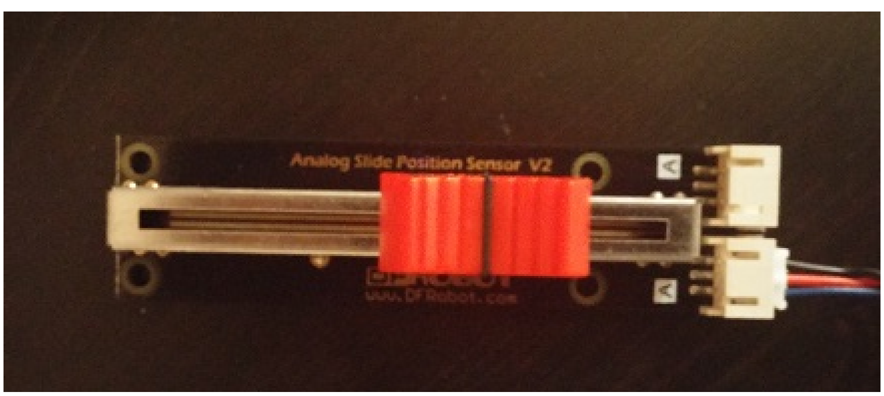

Figure 2.4: Image of slide position sensor used in The Batons

The Batons use the slide position sensor by DFRobot, which operates at $+5V$. The sensor includes two outputs that can be used to average the outputs and minimize fluctuations. However, the batons do not take advantage of this, because this level of accuracy was not necessary. The quantized value of the position (between 0 and 1024) is one of the signals created by The Batons that is then sent to the distributed mapping network created by libmapper. This allows more flexibility in the final mapping, as any scaling factor can be performed in libmapper.

Chapter 3

Hardware Design

The following chapter describes some aspects of the the hardware design of The Batons.

3.1 List of Materials

The following table lists all the materials and their quantities that were used in the hardware design of the Batons. This includes the quantity for both Batons and the Receiver Station.

Table 3.1: List of Materials

Materials	Quantity
Arduino Nano Microcontroller	2
Arduino Mega 2560 Microcontroller Rev3	1
LSM303D 3-Axis Accelerometer and Compass	2
Lynxmotion FSR-01 Force Sensing Resistor Kit	2
DFRobot Slide Position Sensor	2
10kΩ Resistor	2
Seeedstudio 315MHz Low Cost Transmitter/Receiver Pair	1
Seeedstudio 433MHz Low Cost Transmitter/Receiver Pair	1
Battery Holder 4xAA Cube BH-03	2
Three-way switch	2
AA Battery	8

The materials were chosen based on cost, availability, and ease of use. The Batons were meant to be a quick prototype of a DMI and so using Arduino microcontroller boards allowed quickly prototyping. The Arduino Nanos were chosen because of how compact the package is and to simplify and reduce programming time for each baton. This allowed there to be an Arduino for each baton and to quickly get a prototype baton up and running. The Arduino Mega 2560 has 3 auxiliary serial ports that can read serial data streaming in from several different devices, which simplified receiving data from each baton. Section 3.2 discusses sensor choices for The Batons in more detail.

3.2 Baton design

The Batons were designed to have the same hardware and input devices on each. The only fundamental difference between the electrical schematics of the two is the wireless transmitter operation frequency. The left one operates at 315 MHz and the right baton operates at 433 MHz.

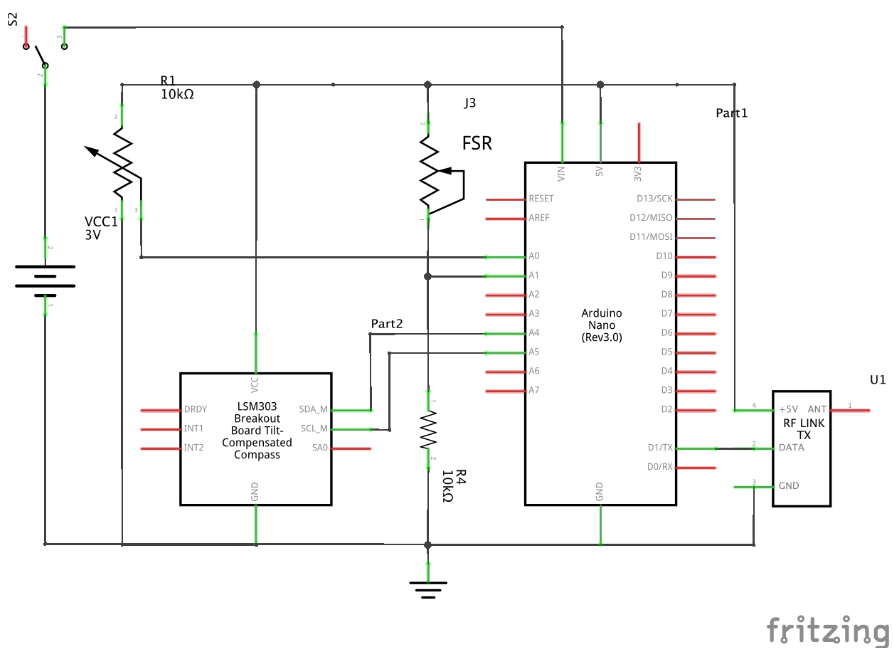

Figure 3.1: Schematic of one baton

The difference in transmission frequency was done to keep the Batons independent and prevent any co-channel interference from occurring when the Batons transmitted data. Co-channel interference is a form of crosstalk that occurs when two devices are transmitting data on the same frequency [5]. By having the two batons operating at different frequencies, the amount of co-channel interference should be minimized between the two signals. However, any radio signals being transmitted on the same frequency as one of the Batons could cause interference.

The LSM303D accelerometer was chosen for its low cost and because it had an open-source C++ library available, which helped simplify its use with the Arduino. The library was expanded to output the real acceleration value in g-force units, to filter the incoming data, and to calculate the pitch and roll. Equations to calculate the real acceleration values in g-force units and the pitch and roll are found in Section 2.1 of the previous chapter. The acceleration values were filtered with an exponential smoothing low-pass filter in the form:

$$y[n] = \alpha x[n] + (1 - \alpha) y[n-1] \tag{3.1}$$

Where $\alpha = 1/2$. This was done to smooth the data being generated, to remove any high frequency noise, and achieve better estimates of the pitch and roll [6].

A linear slide position sensor was used to provide user input. The user's thumb could easily control the slide position while holding a baton. The particular model was chosen for its aesthetic qualities and pre-installed connectors.

The FSR was a last minute addition to the design. Initially a button was meant to be on the opposite side of the slide position sensor. However, the button was difficult to mount onto the baton enclosure and had a limited ability of only representing two binary states. An FSR replaced the button as it can lay flat on the outside of the baton enclosure. It also allowed an extra degree of freedom in terms of controls for the Batons. The FSR was placed in a voltage divider configuration as similar circuitry was already in place for a pull-down resistor. However, according to Medeiros and Wanderley, a simple voltage divider is not the ideal configuration to use [1]. A better conditioning circuit which protects the FSR from exceeding the maximum power requirement and provides finer adjustment of the output signal would be better suited to the task [1]. However, time did not allow a reworking of the circuit. This is considered in more detail in the next chapter, where the design decisions are discussed.

3.3 Enclosing the hardware

The electronics shown in Figure 3.1 were soldered onto a protoboard, and the hardware was added to cardboard tubing. The battery pack was placed at the end of the cardboard tube, where the user would grip each baton. The linear position sensor were mounted close to this to make it easier to reach and modify them. The rest of the protoboarded electronics were placed at the top of the baton in a slightly larger cardboard tube. A wire antenna sticks out of the top to improve transmission of data. This helped in giving a good balance to each baton and made them quite light and easy to use. The cardboard tubing was then wrapped in black and white hockey tape to look aesthetically more pleasing.

Figure 3.2: Image of one baton

3.4 Receiver Station design

The receiver station for the Batons has the wireless receivers for the two Batons. Each receiver is connected to an Arduino Mega 2560, which has 3 auxiliary serial ports that can read serial data being transmitted by the two batons as two separate streams of data. The Arduino Mega is connected to a computer via USB. It verifies and organizes the data coming from both batons and then transmits it to a Max/MSP patch.

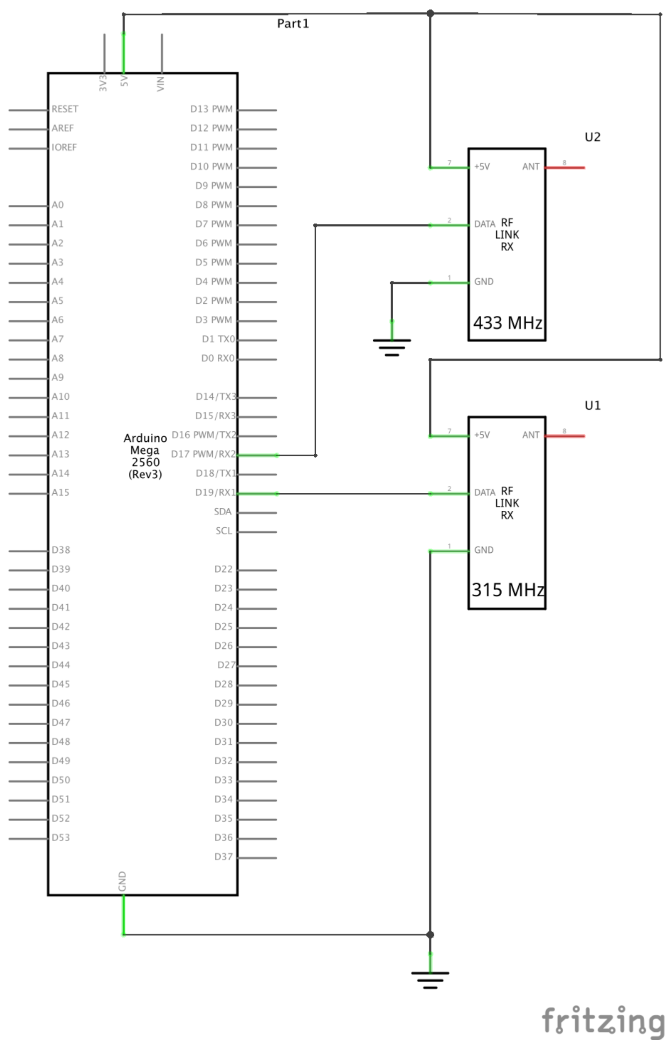

Figure 3.3: Schematic of Receiver Station

When the receiver station detects that serial data is available from both wireless receivers, the Arduino Mega verifies the data from the left baton is valid by checking for a sync character at the beginning of the packet of data transmitted. If this is the case it buffers the data and verifies it is accurate with a checksum. If the data is accurate, it is kept in an array and then it checks the data for the right baton. If that has the sync character and passes the checksum test, the data is kept. When accurate data from both batons is in, the data is written out on the primary serial port, where it is then read by a Max/MSP patch and sent to a distributed mapping network via libmapper objects in Max/MSP.

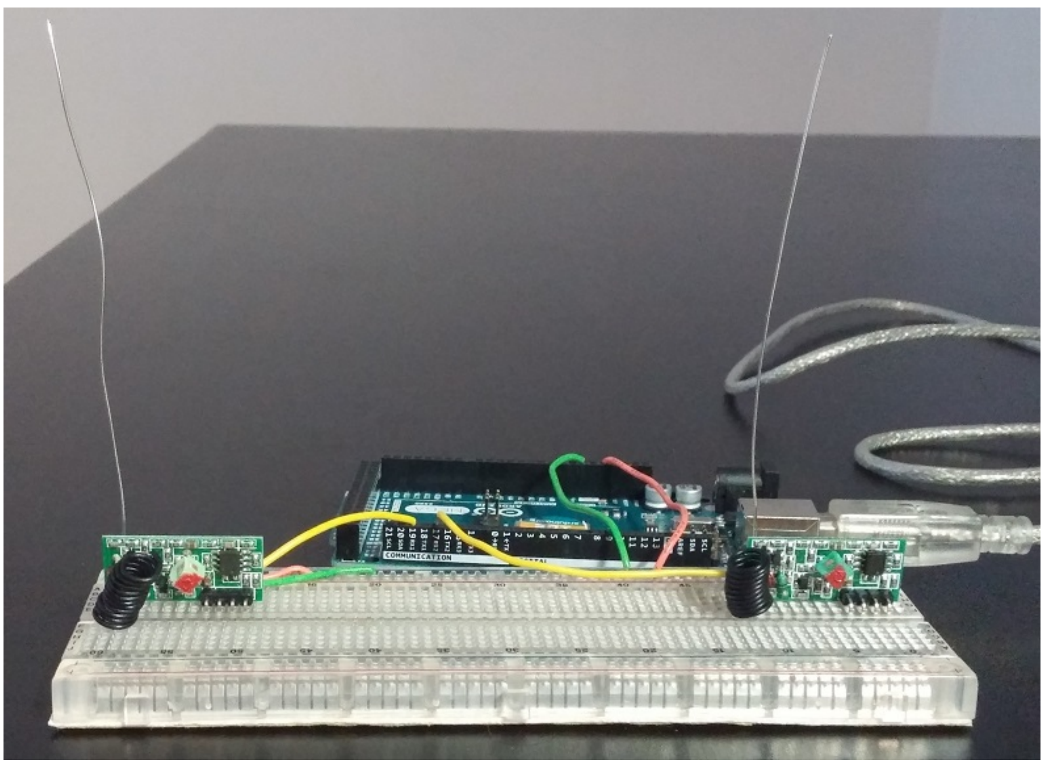

Figure 3.4: Image of Receiver Station

Chapter 4

Discussion

The Batons as a gestural controller are quite comfortable and easy to use. All degrees of freedom for the device are accessible and usable. However, there are some aspects that can be improved upon with later designs. This chapter discusses some issues that are present in the design of the Batons and offers some solutions to remedy these situations.

4.1 FSR Conditioning Improvements

As mentioned in Section 3.2, the FSR conditioning circuit is not ideal. A simple voltage divider circuit was used in the current design, but according to Mederios and Wanderley there the following conditioning circuit would improve performance [1].

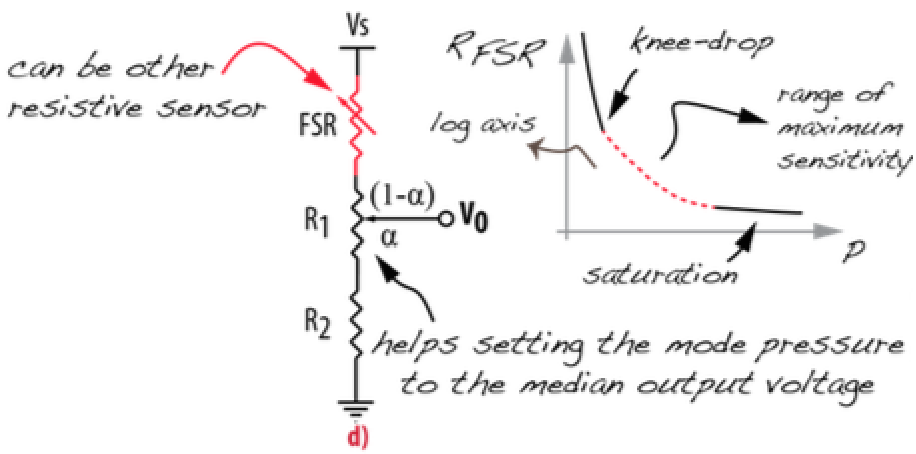

Figure 4.1: Improved voltage divider conditioning circuit for FSR [1]

The figure above shows the improved FSR conditioning circuit. The circuit protects the FSR from exceeding the maximum power requirement and provides finer adjustment of the output signal [1]. This design addresses additional engineering concerns compared to the current "DIY" conditioning circuit.

The circuit could have been added, however, a proto-board had already been assembled, where a button with pull-down resistor was replaced by the FSR. This left the conditioning circuit to be the simple voltage divider as seen in figure 2.2. If the FSR had been factored in to the original design, the improved conditioning circuit would have been used in its place to protect the FSR from exceeding the maximum power requirement and provide finer adjustment of the output signal.

4.2 Wireless jitter

Initially, The Batons were prone to significantly long moments of jitter. Jitter manifested itself as stale data not being updated in Max/MSP for a noticeable amount of time. This was due to the wireless receivers being physically placed too close together on the receiver station. By placing them further apart on a longer breadboard, the amount of jitter was significantly reduced. However, this did not completely solve the problem. A possible solution would be to attach stronger antennas onto the transmitter/receiver pairs. Similarly, the input voltage for the transmitter could be increased to improve range and signal strength. Both these measures would reduce the number of corrupt packets received and improve throughput. Finally, the firmware on the receiver station could also be modified to attempt to improve the throughput of data to Max/MSP, where data from one Baton would immediately be updated in Max/MSP even though the other Baton has yet to send new valid data.

4.3 Enclosures

Although the Baton enclosures are comfortable to hold, light, give easy access to the controls, they are quite fragile because they are made out of cardboard. If better resources and more time were available, the enclosures could have been drawn up in a computer aided design (CAD) program, such as Solid Works, and 3D printed. Also, a printed circuit board (PCB) could have been designed for the hardware, allowing parts to be miniaturized and reduce the length of the Batons.

What would be the most ideal in terms of aesthetic value is if the Batons were a solid-body device, with no indication of wire or circuitry visible on the outside, other than a power switch. This could potentially be done with some sort of capacitive sensing rather than the slide position sensor and FSR currently in use. Additionally, having a full Magnetic, angular rate, and gravity (MARG) sensor present would allow estimating the Yaw of the device, which would allow an additional degree of freedom.

Chapter 5

Conclusion

The Batons are a gestural controller that were developed for this project. They are two wireless batons that transmit pitch, roll, slide position, and FSR values to a receiver station connected to a computer via USB. This data is received by the station and sent to a Max/MSP patch, where it can then be sent to a distributed mapping network created with libmapper.

The resulting device used multiple sensors, provided various inputs to Max/MSP, and was an easy to use gestural controller. However, there are aspects that can be improved including better conditioning circuits, better wireless transmission/reception practices, and better enclosure design. These issues can all be addressed in the subsequent iteration of the design. As a proof of concept prototype, the Batons have provided a good starting point for an interesting form of DMI.

Bibliography

[1] Carolina Brum Medeiros and Marcelo M. Wanderley. A comprehensive review of sensors and instrumentation methods in devices for musical expression. *Sensors*, 14(8):13556, 2014.

[2] Perry R. Cook. Re-designing principles for computer music controllers : a case study of squeezevox maggie. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 218–221, Pittsburgh, PA, United States, 2009.

[3] Eduardo Reck Miranda and Marcelo M. Wanderley. *New Digital Instruments: Control and Interaction Beyond the Keyboard*. A-R Publications, Middleton, Wisconsin, 2006.

[4] Mark Pedley. *Tilt Sensing Using a Three-Axis Accelerometer*. Freescale Semiconductor Application Notes, 2013.

[5] Reinaldo Perez. Chapter 1 base stations, mobile {RF} communication systems, and antenna interferences. In *Terrestrial and Mobile Interference. Aspects of noise, interference, and environmental concerns*, volume 2 of *Wireless Communications Design Handbook*, pages 1 – 60. Academic Press, 1999.

[6] Wahyudi Wahyudi, Adhi Susanto, and Sasongko Pramono Hadi. Comparison between kalman filter and exponential filter on imu data acquisition. *The 12th International Conference on QiR (Quality in Research)*, 2011.