PRESSURE SENSING PIEZOELECTRIC INSOLE

By

Alan Lee

Gerald Kozel

Final Report for ECE 445, Senior Design, Spring 2019

TA: Kyle Michal

1 May 2019

Project No. 47

Abstract

This is a final report regarding team 47's project in ECE 445 Spring 2019 Senior Design at the University of Illinois at Urbana-Champaign. For the project, we designed and built a dynamic pressure sensing insole with a mobile app interface. Each insole gathers data on the change in pressure that a user's foot exerts on it in real time. That data is sent via Bluetooth to an Android app which displays the data on a color coded pressure mapping of the user's foot as they walk. Additionally, piezoelectric elements convert vibrations from the user's movement into voltage which is used to charge the battery which powers the device. Within the report we provide schematics for each aspect of the overall design. The main result of the project were a successful foot pressure distribution display which can be used for diagnostic purposes albeit the device did not end up gaining meaningful energy savings from the piezo elements.

Contents

1. Introduction

1.1 Objective

Over half of limb amputations (about 67 percent) in the United States are attributable to diabetes and related complications [4]. The majority of limb amputations are performed on the lower extremities. Diabetic neuropathy and subsequent damage to sensory nerves in the feet contribute greatly to deformities and ulcers, thereby increasing the risk for amputations if left untreated. This scenario is sadly all too common for diabetic individuals.

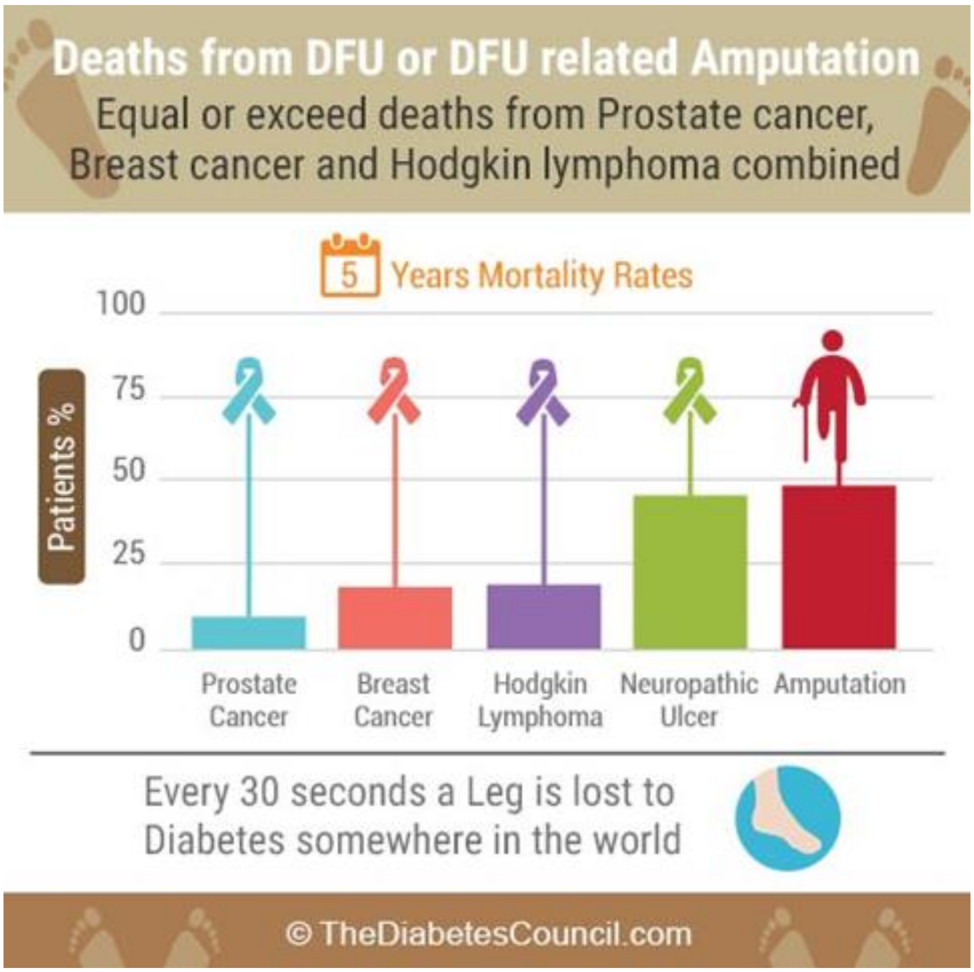

Figure 1 - Statistics on deaths from diabetes related amputation [4]

These statistics are alarming when compared with the CDC estimate that 30.3 million Americans currently have diabetes. From this data, we can draw that a cornerstone of diabetes self-management and education is foot care and continuous monitoring of patient foot physiologic symptoms. Clinical research has shown that monitoring plantar pressure is an important factor to have visibility of [7]. In addition, the diabetes medical devices market size worth will be $35.5 billion by 2024 with a CAGR 7.0% [5]. This makes a sizable market that is prime for disruption as there has not been much innovation with regards to medical care for those with diabetes. Medical devices in the area for the most part are still running on antiquated technology making pain points abundant for the population using them.

To address this, our project was designed to create a device that can provide actionable health information to a health care provider without interfering with daily life. Specifically, we decided to focus on gathering quantitative foot pressure distribution data in a way that is clinically useful for evaluating various foot or posture pathologies. This data can be invaluable for avoiding extremity amputation for diabetics. These insights will bring early detection and preventative measures to patients at high risk for such procedures and notify clinical professionals of patient statuses throughout monitoring their conditions.

We set out to achieve this by building a piezoelectric powered insole which will be embedded with pressure sensors that are connected to a microcontroller. This microcontroller communicates via Bluetooth with a phone application which processes the data into a format that displays the user's foot pressure distribution over a period of time. This can be a disruptive solution in monitoring whether neuropathic ulcers appear on the foot which is the primary reason for a foot amputation.

1.2 Background

Today, pressure mapping is mostly limited to lab analysis, kiosks, and existing commercial insoles. Both the kiosks and other insoles such as DigitSole have limited use cases. For example, Digitsole provides pressure distribution data but does not facilitate the clinical application of said data [8]. Rather, these focus on selling orthopedic insoles or providing runners with feedback in their techniques. TekScan, a producer of foot pressure sensing devices, offers a similar insole device called F-Scan. This solution is quoted at a price of $6,995 plus $35 per insole [1]. After completing our design and build of our insole we ended up with a very cheap production cost of $125.68. Our product ends up being much more competitive than the more expensive options which we've mentioned above.

2 Design

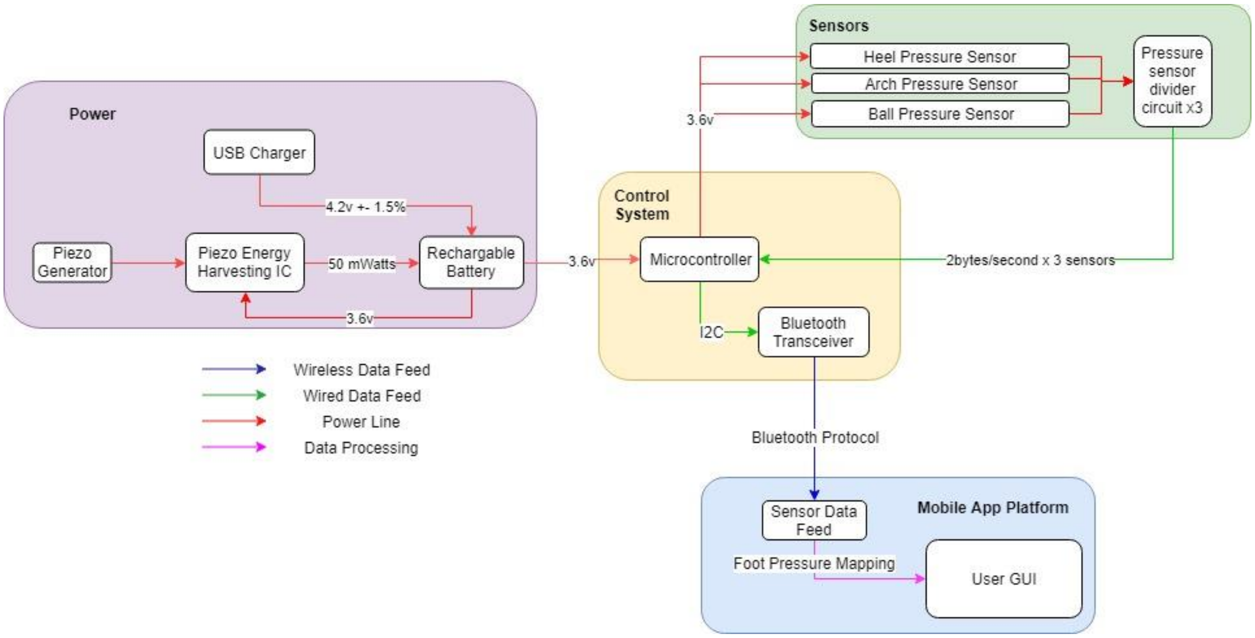

Figure 2 – Block Diagram of All Subsystems

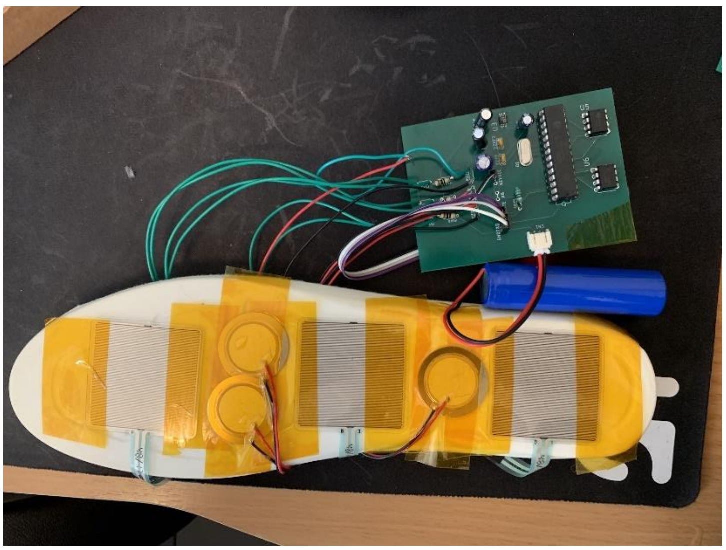

Figure 3 – Final Product Construction

2.1 Power Block

The first module is power, which includes our lithium ion battery and piezoelectric sensors. These connect to each other to enable charging capabilities as the user walks. This module primarily provides power to our sensor block where three pressure sensors are aligned on the sole, arch, and ball of the foot.

2.1.1 Piezo Energy Harvesting IC

This module handles both typical charging operation along with integrating the energy generated from the piezoelectric sensor. The charging capabilities will ensure we can get sufficient energy savings from our piezoelectrics and power our insole. This IC should also ensure the voltage output is regulated so that the battery charges steadily. Below is a schematic of how we utilized the LTC3588-1 to charge our battery.

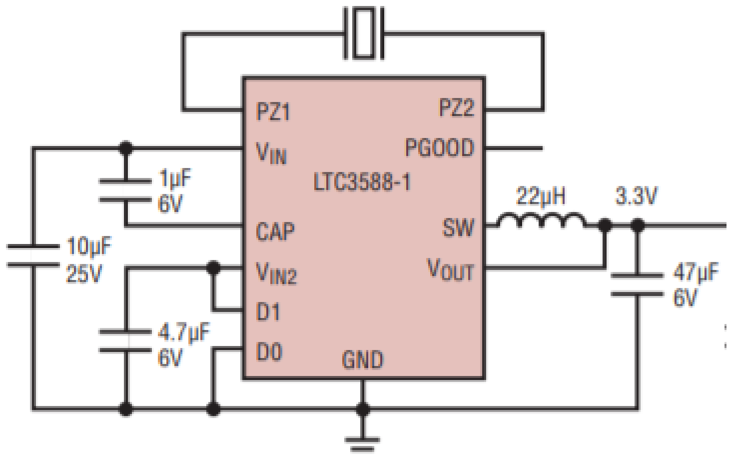

Figure 4 – Energy harvesting schematic [9]

2.1.2 USB Charger

The purpose of the USB Power Converter module is to utilize voltage from a wall socket via USB C which can be used in a commercial lithium ion charging circuit to charge the battery. This is the primary source of charge for our battery as the piezos will not generate enough power to sustain the battery for 8 hours. We used a breakout board produced by Adafruit which mainly utilizes a MCP7381/2 linear charge management controller which employ a constant-current/voltage charge algorithm.

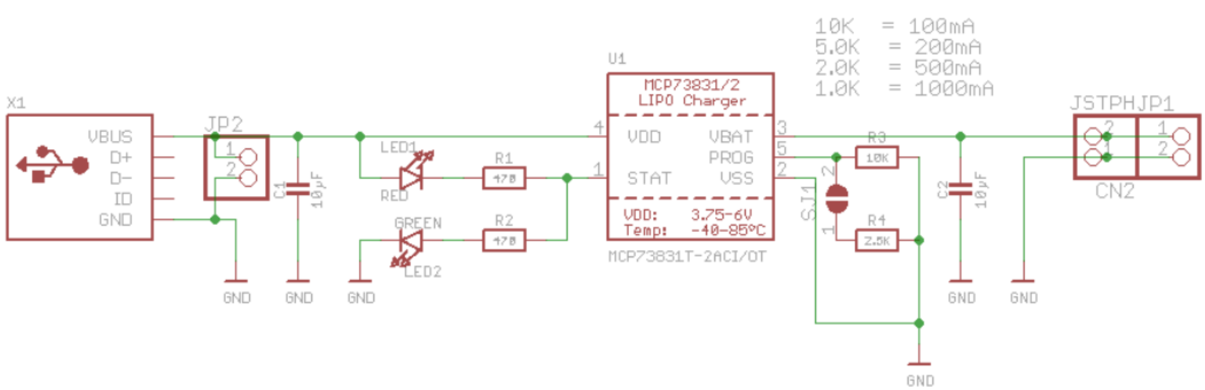

Figure 5 – USB charger schematic [2]

2.2 Control Block

2.2.1 Microcontroller

For our applications we utilized an ATMEGA328P microcontroller in order to consolidate the data feeds we were receiving from all three pressure sensors and to provide an interface with which we could communicate over UART with our Bluetooth transceiver. We included a 16 MHz oscillator instead of opting for the slower oscillator included with the microcontroller so that data could be transferred at a faster rate. As seen in the schematic below our microcontroller takes voltage readings from our sensors on pins ADC[0:2] and outputs that data to the HC-05 through TXD.

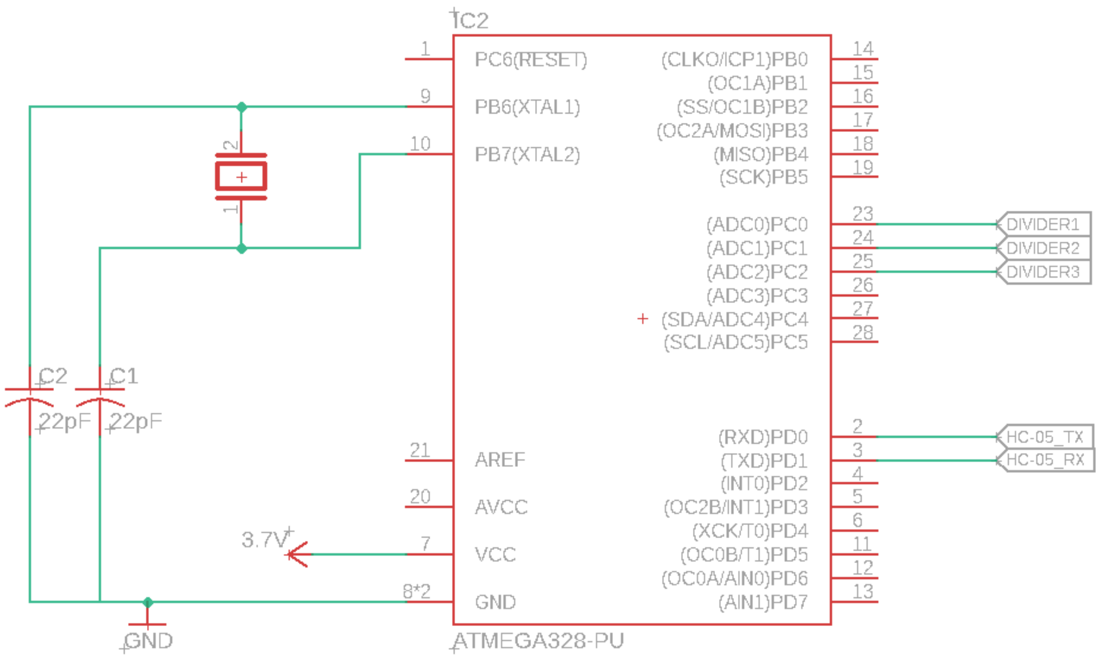

Figure 6 – Microcontroller schematic

Since we decided to go with the ATMEGA328P microcontroller it was simple to utilize an Arduino to program it. The code we wrote was mainly designed to loop through analog pin 1-3 and then perform a serial port send for each individually through the TXD pin. The code also converts the voltage level over each divider to a force by calculating conductance and dividing by a constant value. This data goes directly through to the Bluetooth module which then transmits that data to the app. A flowchart of this code is shown below:

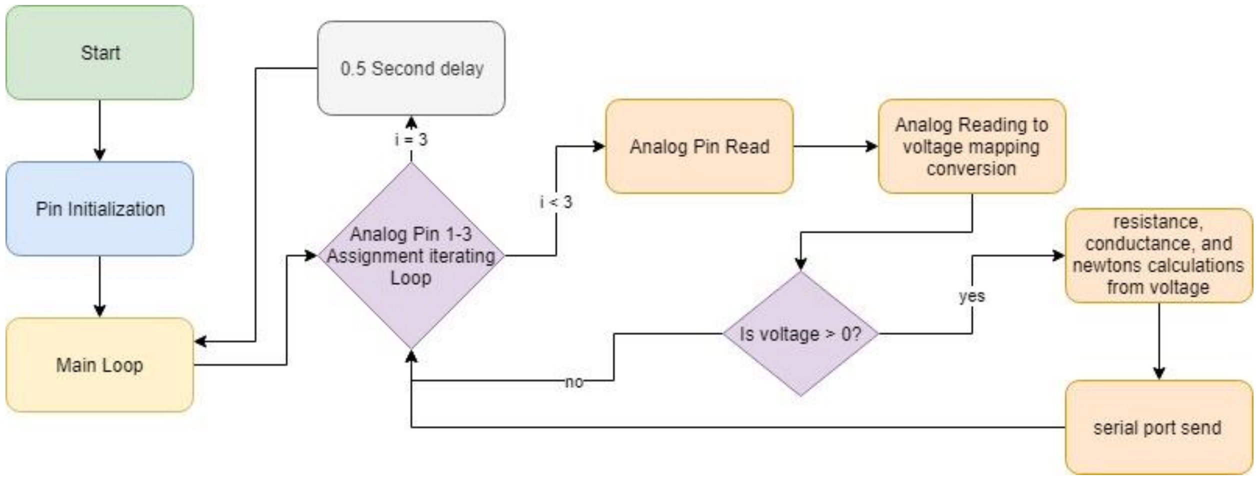

Figure 7 – Microcontroller software flowchart

2.2.2 Bluetooth Transceiver

The HC-05 Bluetooth Transceiver is a Serial Port Protocol module which interfaces serially over UART with our microcontroller to transmit the pressure sensor values to the mobile app.

The HC-05 has six pins as seen in the pinout diagram below:

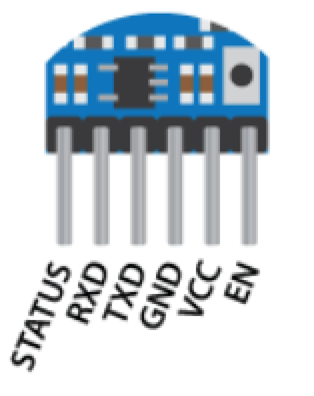

Figure 8 – HC-05 Pinout diagram [3]

In our design we only utilized four of those pins in order to satisfy all related requirements. VCC and GND was connected to the battery's respective lines while RXD and TXD were connected directly to the microcontrollers opposite TXD and RXD. This is apparent in the diagram of the microcontroller above.

2.3 Sensor Block

Within our sensor block is the three force sensitive resistors corresponding to the arch, ball, and heel of the foot along with the voltage divider circuits which allow us to quantify changing resistances through an output voltage.

2.3.1 Pressure Sensors

Each pressure sensor has a well-defined resistance versus force response as seen in the log-log graph that follows.

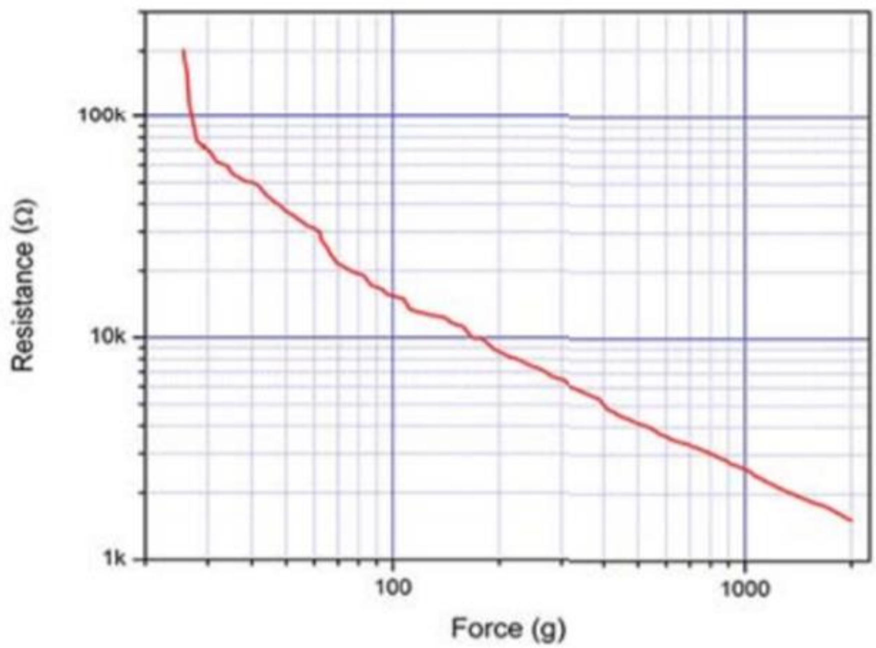

Figure 9 – Resistance vs. Force response for force sensitive resistors [6]

As one presses on each sensor there is a decrease in resistance across the two leads of the force sensitive resistor. We utilized this information in our voltage dividers in order to make increasing output voltage levels correspond to higher levels of pressure as less resistance in turn produce a larger voltage drop across the FSR in the divider circuit.

2.3.2 Sensor Voltage Divider

Each sensor is connected to a simple voltage divider circuit with an additional resistor and op amp. We utilize the op amp in order to maintain stability of the voltage output while the extra resistor experiences higher or lower voltage drops based on the pressure applied to each FSR. Below is a schematic for one of the three identical sensors:

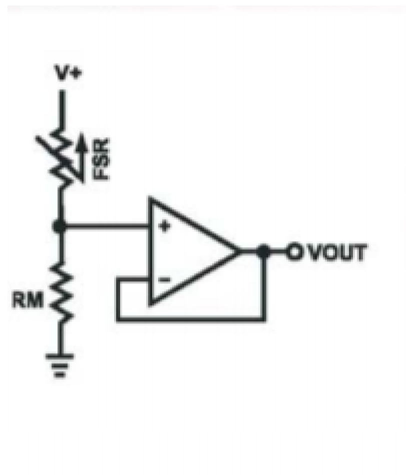

Figure 10 – Voltage Divider Schematic [6]

The voltage divider is characterized by the following equation:

$$V_{OUT} = \frac{R_M V+}{(R_M + R_{FSR})}$$

Our microcontroller can directly take this value and calculate a force using the processes mentioned in the previous section.

2.4 Mobile App Platform Block

2.4.1 Sensor Data Feed

The software on the mobile app side is primarily designed to parse pressure data provided by the HC-05 and update a display based on the magnitude of the data. When a user starts the app, they are presented with a list of Bluetooth devices with which they can connect to. This is within a Device Fragment function in the code. Once they click a device to connect to the code creates a Terminal Fragment which begins to read through a serial socket. After parsing the data received simple comparison statements map the value to one of four colors on the GUI display.

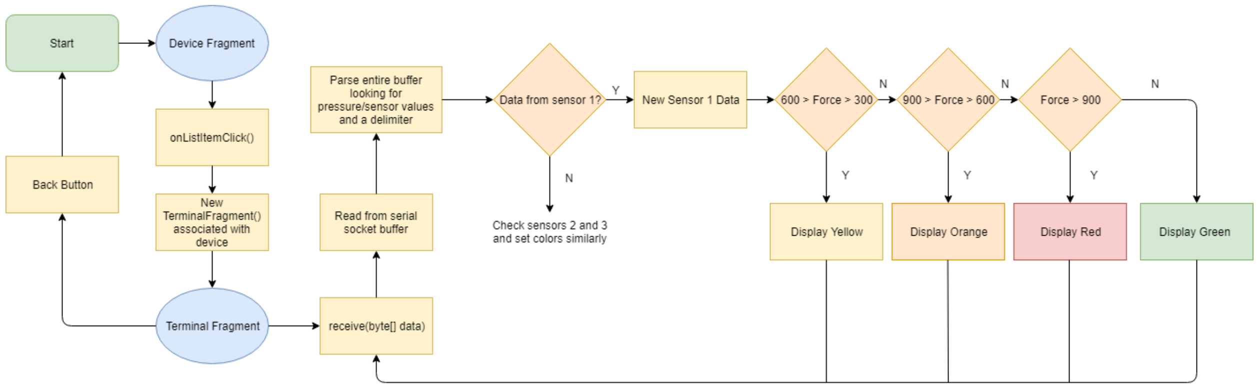

Figure 11 – Mobile App Software Flowchart

2.4.2 User GUI

We designed the user GUI to provide a live color-coded pressure distribution along with a debug feed which displays force values in half second intervals. This module depends on the code within the sensor data feed subblock which parses the data into key value pairs based on sensor number and force value. This allows us to create a simple ListItem which displays a live feed of parsed key value pairs and color mapped ImageViews for each sensor.

3. Design Verification

3.1 Power Block

3.1.1 Piezo Energy Harvesting IC
Due to complications regarding getting the IC to operation voltage from an input capacitor and insufficient energy generation, the design requirement for this section could not be verified. Further explanation for root cause is explained in section 5.2 Uncertainties.

3.1.2 USB Charger
With the chosen LiPO charger IC, the battery was charged to full within 5 hours. Verification from the OEM datasheet provided the info such that the IC plugged in to the battery with a USB C connection to a wall socket converted would provide ample charging capabilities. After verifying from the given data sheet, operating within parameters was verified as the battery was at full charge within the stated timeframe of 5 hours.

3.2 Control Block

3.2.1 Microcontroller
After choosing the AT Mega 328P, the microcontroller was confirmed to have 5 Analog inputs as well as RX, TX pins. This satisfied the original design requirement of compatibility of the 3 sensor input pins as well as Bluetooth functionality.

3.2.2 Bluetooth Transceiver
Choosing the HC-05 as the Bluetooth module, the technical specifications on the data sheet provided by the OEM stated operational limits of 9 meters (30 ft) which satisfied the original requirement of operational range. In addition, the HC-05 was verified to operate on Bluetooth V2.0 SPP with a baud rate of 9600 from the OEM data sheet.

3.3 Sensor Block

3.3.1 Pressure Sensors
Below shows the force response of full pressure of a spike of pressure on a single sensor input. As shown, the force was not saturated even at 600 newtons. As such, this satisfied the original requirement of kPa to newtons conversion.

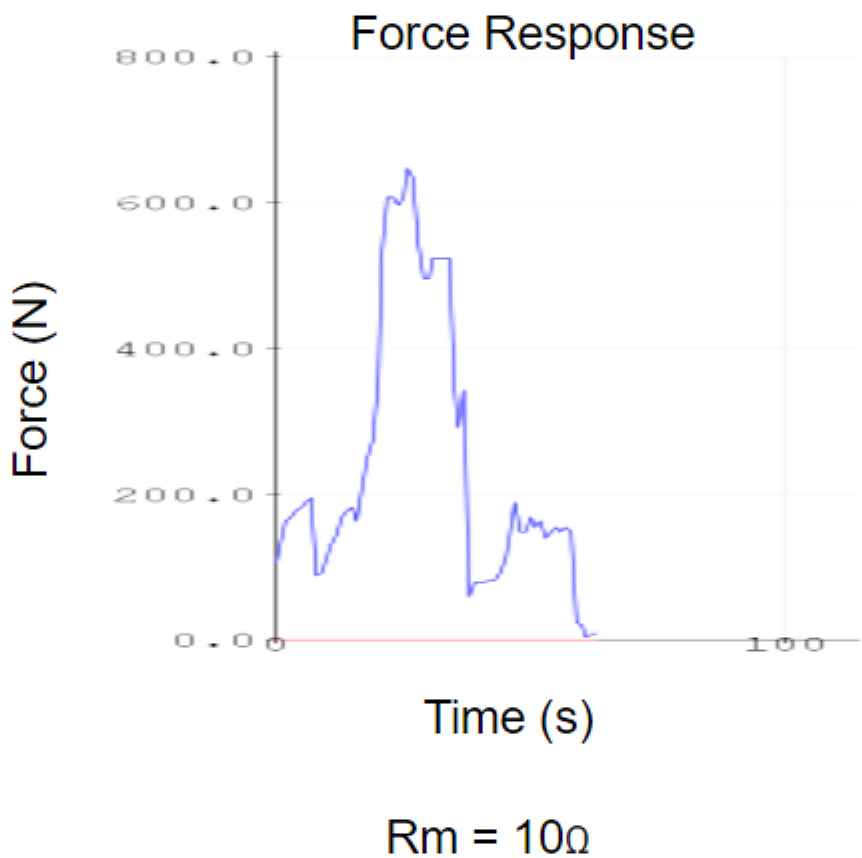

Figure 12 – Force response graph

3.3.2 Sensor Voltage Divider

The original requirement was for a sensitivity of having 2.63mV per 1 N of change. With the below graph, it is verified that 38mV of change corresponded to 14N. This ratio of 38/14 equates to 2.71 mV/N which exceeded the original requirement of a 2.61mV/1N sensitivity.

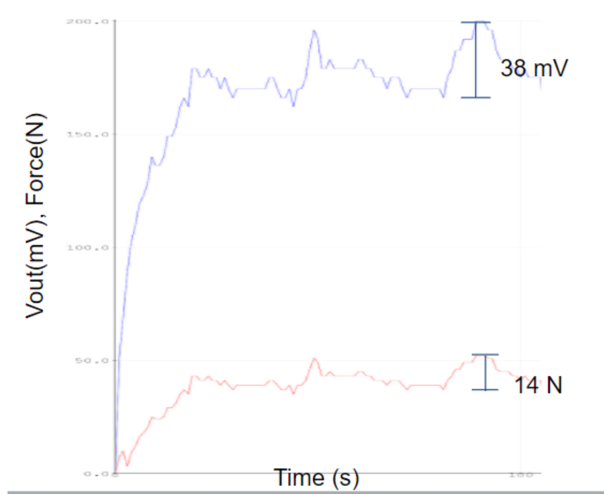

Figure 13 – Vout and Force versus time graphs

3.4 Mobile App Platform Block

3.4.1 Sensor Data Feed

The upper half of the below image is the display of dynamic sensor data feed. As seen, each sensor 1-3 provides constant updates of force application data.

3.4.2 User GUI

The second half of the below image displays the user GUI of all sensors color coded based on the applied force of the corresponding location. Each sensor display area updates dynamically with the input feed.

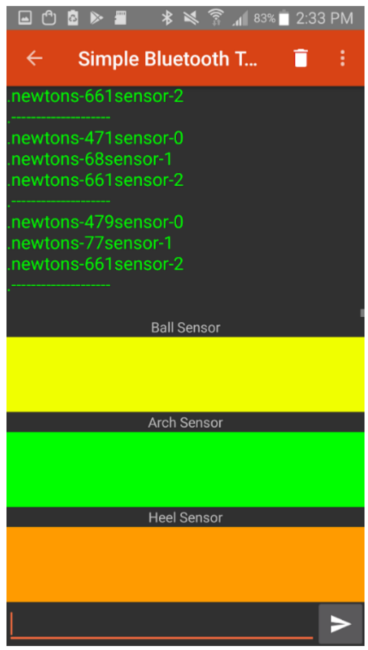

Figure 14 – App GUI

4. Costs

4.1 Parts

The table in this section shows the typical cost of each part used for constructing the project. Note that this total cost is for two insoles and the cost of a suitable Android device to run the mobile app is not included in the table.

Table 1 Parts cost

Part	Distributor and Number	Amount	Cost
Piezoelectric Sensors	Multicomp; ABT-441-RC	6	**$6.78**
Lithium Ion Battery	ShenZhen; ICR18650	2	**$25.00**
Microcontroller	Digikey; ATMEGA328P	2	**$4.28**
Wireless Bluetooth Transceiver	Amazon; HC-05	2	**$17.98**
Pressure Sensors	Interlink; FSR-406	6	**$53.70**
Resistors, capacitors, op-amps, diodes	Digikey	~30 in all	**$10.00**
Foam Insole	Target; Up&Up	1 (pair)	**$0.99**
Lithium Ion Battery Charger	Adafruit; MCP73831	1	**$6.95**
TOTAL			**$125.68**

4.2 Labor

Table 2 Labor cost

Name	Rate per Hour	Hours	Total Cost * 2
Alan Lee	$40	14 hr. /wk. * 12 wk. = 168 hrs.	**$13,440**
Gerald Kozel	$40	14 hr. /wk. * 12 wk. = 168 hrs.	**$13,440**
TOTAL			**$26,880**

4.3 Total Cost

With both labor and parts combined we get a total cost of $27,005.68. As we realized the majority of cost ends up being in the actual construction of the device. We took a great deal of time working on the design aspects as we went along with our project as we encountered some failures. By pivoting the design, we ended up spending much more time on the actual build phase which inflated our labor cost greatly. For the future builds of the product the labor cost would be reduced significantly making this product even more economical to produce and sell for a competitive price.

5. Conclusion

5.1 Accomplishments

Our accomplishments at the end of our product development lifecycle include successful integration of force sensitive resistors, a durable insole with all components onboard, a fully soldered PCB with no loose connections, a foot pressure mapping android application, and successful dynamic pressure data stream from the insole to the mobile app.

5.2 Uncertainties

The major uncertainty that remained at the end of our development cycle was the piezo charging mechanism. Through the development timeline, there was no comprehensive indicator that the piezo electrics could provide 2-5% of extended battery life to the device. There are a few possibilities as for the root cause of uncertainty. The first being that the IC used for energy harvesting originally indicated a specific piezo electric part optimized for the function of energy harvesting. Due to the part being no longer sold by any online distributor, the design opted for a different piezo electric which may not have been capable of generating sufficient power. Another consideration is the power harvesting IC required a 2.7v operating voltage. Because the IC was not connected to our main power supply, the piezos would have to build up a voltage of 2.7v on an input capacitor connected to the IC's voltage input. Due to the piezos having a minimal energy generation, charging the input capacitor to 2.7v was not feasible as it would have taken far longer than the considered use time of the device. These two reasons are the primary cause of uncertainty for the piezo electrics not offering a significant device lifetime extension of operation.

5.3 Ethical considerations

As with many products, there are a few possible safety hazards involved. Our lithium-ion battery rechargeable battery which powers the device has a possibility to explode. This usually occurs in the case of a short circuit, where all of the energy within the battery is released at once. Short circuits like this can happen when the plastic separator fails between the anode and cathode, allowing them to physically touch [10]. Over-charging and high temperatures must be avoided at all costs in operation. Additionally, proper insulation in the insole must be kept so that the battery is not subjected to extreme cold or heat.

The product also collects potential Protected Health Information (PHI), which is requires confidentiality, integrity, and availability of said data [11]. Diagnosis within the app is not provided to avoid violating #3 in the IEEE code of ethics by potentially misdiagnosing a user. Primarily, the collected data ends up in the hands of a medical doctor who could utilize it accurately. It's important to note that this product will not be doing any analysis on the data, instead giving a professional means to conduct that analysis themselves.

5.4 Future work

As with all products, improvements can be made. For this insole, the main room for improvement lies in the pressure sensing, piezo energy harvesting, and Bluetooth capabilities. Smaller pressure sensors could be implemented as the microcontroller used for the device has 5 analog inputs. In future models, to get a finer granularity of pressure points on the insole, smaller pressure sensors can be used to pinpoint more exact areas of pressure application. As for the piezoelectrics, as explained in the Uncertainties section 5.2, there could be potential improvements if considering changing the piezo electric part to one more optimized for the harvester IC being used. There is also more room on the insole to add more piezo electrics to the underside of the construction. This would increase the power gains as more of the pressure application would be converted into energy. The IC being used could also be compared to other harvester IC's to see which would offer better potential energy absorption. Lastly, for the Bluetooth and mobile app, in a future product release, the Bluetooth protocol could be switched to BTE 2.1 to allow two simultaneous connections on the mobile app to both insoles. This would provide a more comprehensive GUI with visuals on both feet while operating the device. As for the physical construction, the PCB can also be shrunk to a less clunky size so that users are not obstructed in the way they walk along with a battery compartment attached to the PCB. These possibilities would make for a promising next iteration of the insole product.

References

[1] Open Lab, 'Wearable Weight Sensing and Feedback Device'. [Online]. Available: http://www.openlab.psu.edu/wp-content/uploads/2014/04/Garneau_Wearable_Weight_Sensing_and_Feedback.pdf. [Accessed: 6-Feb-2019].

[2] Adafruit, 'Micro-Lipo charger'. [Online]. Available: https://learn.adafruit.com/assets/17043. [Accessed 2/20/2019].

[3] "Arduino and HC-05 Bluetooth Module Tutorial," HowToMechatronics, 08-Feb-2019. [Online]. Available: https://howtomechatronics.com/tutorials/arduino/arduino-and-hc-05-bluetooth-module-tutorial/. [Accessed: 01-May-2019].

[4] The Diabetes Council, 'diabetes-and-amputation-everything-you-need-to-know-to-avoid-amputation'. [Online]. Available: https://www.thediabetescouncil.com/diabetes-and-amputation-everything-you-need-to-know-to-avoid-amputation/. [Accessed: 2/18/2019].

[5] Grand View Research, 'global-diabetes-devices-market'. [Online]. Available: https://www.grandviewresearch.com/press-release/global-diabetes-devices-market. [Accessed: 2/20/2019].

[6] Adafruit, 'FSR 400 Series Data Sheet'. [Online]. Available: https://cdn-shop.adafruit.com/datasheets/FSR400Series_PD.pdf. [Accessed 2/20/2019].

[7] NCBI, 'Plantar Pressure as a Risk Assessment Tool for Diabetic Foot Ulceration in Egyptian Patients with Diabetes'. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4257475/ .[Accessed 3/6/2019].

[8] DigitSole 'DigitSole, the leader in connected insoles'. [Online]. Available: https://www.digitsole.com/. [Accessed 3/6/2019].

[9] Linear Technology, 'Nanopower Energy Harvesting Power Supply'. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/35881fc.pdf. [Accessed 4/29/2019].

[10] Battery University, 'Safety Concerns with Li-ion Battery'. [Online]. Available: https://batteryuniversity.com/learn/article/safety_concerns_with_li_ion. [Accessed 6-Feb-2019].

[11] HIPAA Journal, 'What is Protected Health Information?'. [Online]. Available: https://www.hipaajournal.com/what-is-protected-health-information/. [Accessed: 6-Feb-2019].

Appendix A Requirement and Verification Table

Table 3 System Requirements and Verifications

Requirement	Verification	Status
• Within 1 hour of walking, must provide a mean charging rate of 50 mW (2-5% savings)	A. Provide pressure to the sensors at various times over the course of an hour B. Use a voltmeter to get a voltage reading for the battery C. Do this with the piezoelectric module and without the piezoelectric module D. Calculate battery life with voltage reading based on battery's operating range. E. Compare both values to get energy savings.	N
• Must convert AC to DC (bridge rectifier) to store energy generated by the piezos	A. Make sure IC has a rectifier in the data sheet to ensure the piezo's AC voltage can be utilized to charge the battery.	Y
• Must regulate output voltage to 3.6V +/- 0.1V	A. This can be verified by consulting the IC's datasheet and observing the programmable output levels for the chip.	Y
• USB charger must charge the battery at a rate of 5 +/- 0.2 watts (5V at 1A) and to a level of 3.6V	A. Verifiable by IC's datasheet.	Y
• Battery must store 2000 mAh of charge	A. Verifiable by the battery's datasheet.	Y
• Battery must remain between 10 - 40 degrees Celsius during charging.	A. Verifiable by the battery's datasheet.	Y
• Sensors must react by changing resistance to different applications of pressure	A. Connect the sensors to a 1k Resistor in series. B. Connect voltmeter on both ends of 1k resistor. C. While sending a 5mA current through the series circuit, apply body pressure with hand on pressure sensors D. Verify the voltage readings change.	Y
	A. Choose an RM value around 1kΩ	Y

• Voltage divider must be able to output voltage 0.01V per kPa for up to 120 kPa (double the applicable weight)	or lower to increase sensitivity for higher pressure. B. Add increasing amounts of pressure to each sensor individually and observe the output voltage from the voltage divider. C. FSR should not be saturated at 120 kPa, check that voltage across each sensor still changes by at least 0.01V with pressures +/- 1kPa.	
• Microcontroller must be powered by a 1.8v - 5.25v +- 15%, 19mA +- 10mA power supply for 8 hours	A. Verify figures on data sheet. B. Must see the microcontroller turn on while measuring voltage input is within expected range. C. Let the device run for 8 hours. Check back to ensure there is still power at this point by checking battery voltage.	Y
• Microcontroller must have sensor data input ports and support the I2C protocol for communication with the Bluetooth Transceiver	A. Verify the microcontroller has sensor inputs on the datasheet/website.	Y
• Transceiver must be compatible with our microcontroller and the I2C protocol	A. Verify transceiver has I2C capability in datasheet	Y
• Transceiver must operate on 2.4GHz bandwidth	A. Verify the transceiver operates on Bluetooth's standard bandwidth of 2.4GHz on the datasheet	Y
• Transceiver must operate on a range of 30 meters	A. Separate device and insole by 10m B. Verify connection exists by checking app C. Repeat these steps for 5m increments up to the maximum 30 meters.	Y
• Transceiver must transfer data losslessly	A. Verify that data transfer of pressure sensors is lossless through visual inspection of data set.	Y

• App must be able to connect to the Bluetooth adapter on the insole	A. Call startDiscovery() within the app and then use getAddress() to receive the MAC address associated with the discovered device. B. Initiate the connection by calling connect and check to make sure it does not throw a C. Wait for a push notification popping up indicating that the connection is successful.	Y
• App must have a foot display for pressure distribution	A. Verify app has foot pressure UI that is displays data taken from the insole after a gather data button is pressed.	Y
• App must display accurate pressure sensor data	A. Check for malformed data by applying pressure on each sensor individually and inspecting the display in the app. Pressure should appear in the corresponding areas with a level that matches the weight.	Y
• App must have Android OS compatibility	A. Download the app to an actual android device B. Go through previous verification steps to ensure app works on hardware.	Y